Instytut Podstawowych Problemów Techniki
Polskiej Akademii Nauk w Warszawie

ROZPRAWA DOKTORSKA

Metody obliczeniowe jedno- i wielokryterialnej optymalizacji rojem cząstek. Zastosowanie w bioinformatyce.

mgr Mateusz Banach

Praca wykonana pod kierunkiem
Pani prof. dr hab. Ireny Roterman-Koniecznej
w Zakładzie Technologii Informatycznych,
Wydział Fizyki, Astronomii i Informatyki Stosowanej,
Uniwersytet Jagielloński w Krakowie

Kraków, 2017

Streszczenie

Przedmiotem niniejszej rozprawy doktorskiej jest użycie metod optymalizacji jedno- i wielokryterialnej funkcjonujących w oparciu o zasadę działania algorytmu roju cząstek do sprawdzenia założeń modelu rozmytej kropli oliwy (fuzzy oil drop, FOD) dotyczących wpływu opisywanych przez ten model oddziaływań hydrofobowych na proces tworzenia się kompleksów typu białko-białko.

Oddziaływania hydrofobowe cząsteczki białka z otaczającym je środowiskiem wodnym są uznawane za mające istotne znaczenie w formowaniu się jego struktury trzecio- i czwartorzędowej. Pomimo tego, według Autorów modelu FOD, są one traktowane w sposób niewystarczający w obecnych podejściach do modelowania tych procesów, opartych na minimalizacji energii oddziaływań pomiędzy atomami.

W celu sprawdzenia poprawności założeń modelu FOD został opracowany eksperyment *in silico* przewidywania struktury kompleksu 200 białek homodimerycznych wybranych z bazy Protein Data Bank (PDB). Do wykonania tego eksperymentu zastosowano dwa algorytmy: algorytm optymalizacji rojem cząstek (PSO) oraz opracowany przez Autora rozprawy algorytm wielokryterialnych rodzin rojów (MOSF). Za pomocą tych algorytmów sprawdzono efekty osobnego i równoczesnego wpływu oddziaływań hydrofobowych (opisywanych przez model FOD) oraz oddziaływań niekowalencyjnych (opisywanych przez pole ECEPP/3) na układy par łańcuchów polipeptydowych. Ocena zgodności uzyskanych w tym eksperymencie kompleksów z ich strukturami natywnymi została wykonana przy pomocy miary RMSD i w przestrzeni krzywych ROC porównania map kontaktów niewiążących.

Algorytm MOSF został opracowany w celu uzyskania możliwości wykonywania w trakcie optymalizacji wielokryterialnej analizy skupień odnalezionych rozwiązań niezdominowanych, osiągania jednorodnej reprezentacji zawartości optymalnego zbioru Pareto oraz liniowej złożoności obliczeniowej ze względu na liczbę optymalizowanych kryteriów i cząstek. Wysoka przydatność algorytmu MOSF, również do rozwiązywania problemów optymalizacyjnych spoza bioinformatyki, została wykazana poprzez porównanie jego wyników z wynikami dwóch popularnych algorytmów: NSGA-II i NSPSO. Porównanie to zostało przeprowadzone na zbiorze wybranych funkcji testowych oraz kryteriów utworzonych w sposób losowy przez generator MPB.

W rozprawie znajduje się również wprowadzenie do bioinformatyki i tematu przewidywania kompleksów białkowych, optymalizacji wielokryterialnej i algorytmów optymalizacyjnych, w szczególności algorytmu PSO oraz jego modyfikacji.

Abstract

Aim of this doctoral thesis is to apply global and multiobjective optimization methods based on particle swarm theory, to verify the assumptions of the fuzzy oil drop model (FOD) regarding the influence of hydrophobic interactions on the process of formation of protein-protein complexes.

Hydrophobic interactions between the protein molecule and water environment are recognized to be influential forces in the process of formation of its tertiary and quaternary structure. However, according to the Authors of the FOD model, they are not addressed sufficiently by current approaches to protein folding and docking prediction based on potential energy minimization.

The assumptions of the FOD model were verified by an *in silico* protein complex prediction experiment. 200 homodimer proteins were selected from the Protein Data Bank (PDB) to take part in this experiment. Polypeptide chains comprising these complexes were separated from each other and then combined together using two optimization methods: particle swarm optimization (PSO) and multiobjective swarm families (MOSF). First is a well-known global optimization method, while the other was developed by the thesis' Author. Use of these algorithms allowed to observe separate and simultaneous influence of hydrophobic interactions (described by FOD model) and nonbonded interactions (described by ECEPP/3 force field) on the structure of a protein complex. Results of the simulation were compared with native structures using RMSD and contact map ROC curves.

MOSF algorithm, presented in this thesis, was developed in order to introduce online cluster analysis of nondominated solutions (in both search and objective space) into multiobjective optimization. Other features of this algorithm include: near-uniform representation of the Pareto optimal set, linear complexity in terms of numbers of objectives and particles. The ability of this algorithm to also solve problems unrelated to bioinformatics was proven by comparing its performance with two other, well-known algorithms: NSGA-II and NSPSO. This comparison was carried out using a group of selected multiobjective test functions and objectives randomly generated by the moving peaks benchmark (MPB).

This thesis also comprises introduction to bioinformatics and the field of protein-protein complex prediction, multiobjective optimization and optimization methods, in particular, the PSO algorithm and its modifications.

Struktura tekstu

Treść główna niniejszej rozprawy doktorskiej składa się z pięciu rozdziałów ułożonych w następującym porządku: wprowadzenie, materiały i metody, wyniki, dyskusja i wnioski oraz podsumowanie.

Rozdział wprowadzenie prezentuje motywację i cele pracy oraz osadza je w kontekście dotychczasowego stanu wiedzy z poruszanej tematyki.

W rozdziale materiały i metody znajduje się opis bazy danych białek, na których zostały przeprowadzone badania, a także kluczowych (ECEPP/3, FOD, PSO) oraz pomocniczych algorytmów zastosowanych w tych badaniach.

Rozdział wyniki przedstawia zaproponowany przez Autora rozprawy algorytm optymalizacji wielokryterialnej MOSF, jego porównanie z innymi metodami z tej dziedziny, modyfikację modelu FOD, analizę białek z bazy danych, a także opis oraz wyniki eksperymentu *in silico* kompleksowania typu białko-białko.

Za rozdziałami dyskusja i wnioski oraz podsumowanie znajdują się trzy rozdziały dodatkowe: rysunki, tabele i oprogramowanie.

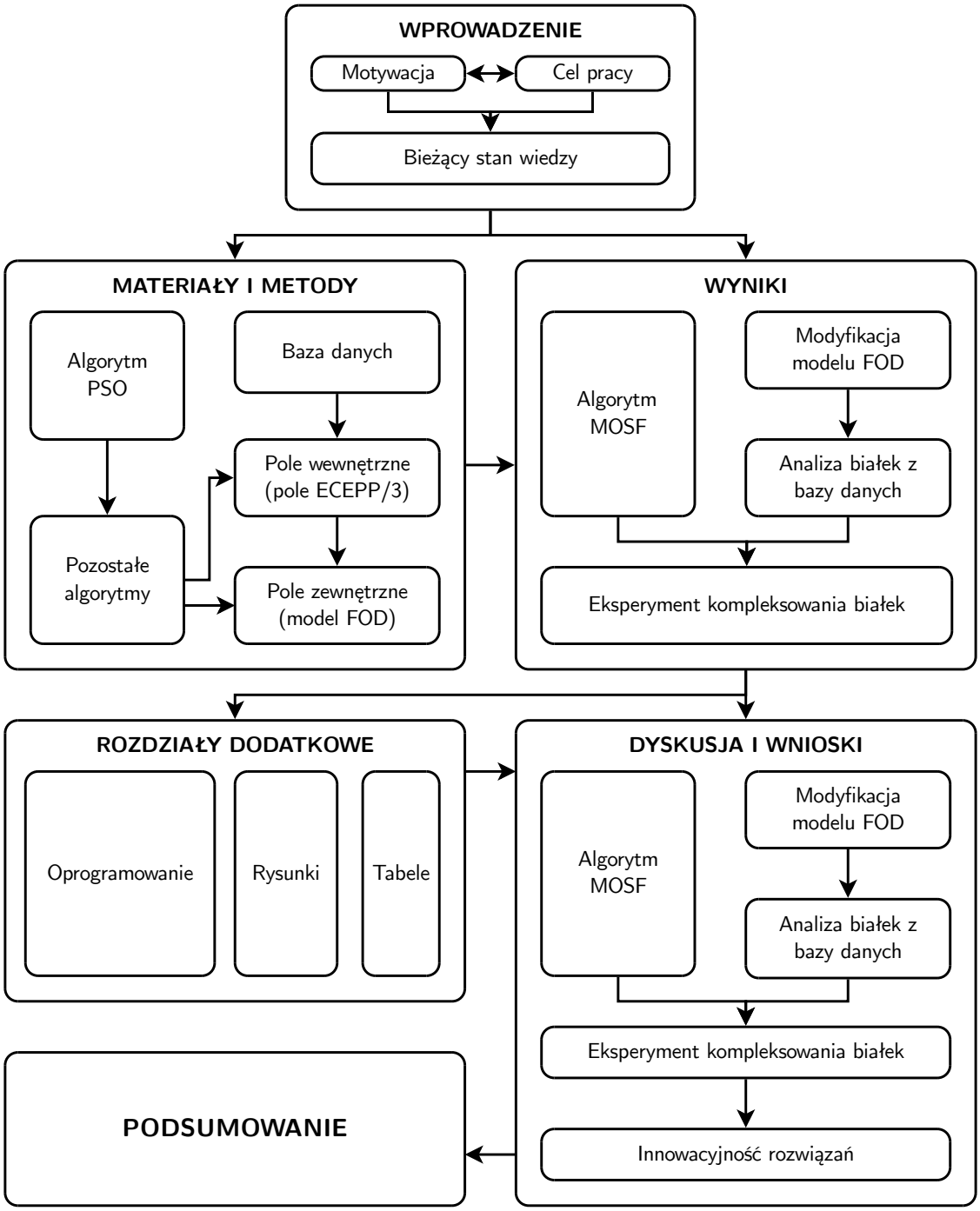
Pracę zamyka wykaz pozycji bibliograficznych, spisy rysunków, tabel, równań, definicji oraz wyjaśnienie często występujących skrótów.

Schemat nazewnictwa i układu rozdziałów

Pomimo poruszania tematów z różnych dziedzin, tam gdzie było to możliwe, Autor rozprawy starał się konsekwentnie stosować poniższy schemat nazewnictwa:

x	= przykład nazwy zmiennej, funkcji, wektora lub krotki
X	= przykład nazwy macierzy, rozkładu, listy lub funkcji
$\mathcal{X}$	= przykład nazwy zbioru
`1ABC`	= przykład nazwy identyfikatora struktury białka w bazie PDB
PROGRAM	= przykład nazwy programu, funkcji lub klasy

Elementy kolekcji oznaczane są indeksami dolnymi, na przykład x_i, X_{ij}, lub $\mathcal{X}_i$.



Ze względu na dwutorowość tematyki niniejszej rozprawy, dzielącą się na części informatyczną oraz biologiczną, dla ułatwienia poruszania się po jej tekście, opracowano powyższy schemat. Strzałki oznaczają kontynuację, podobieństwo tematyczne lub zastosowanie. Dla zachowania czytelności tego rysunku, powiązania pomiędzy zależnymi od siebie częściami różnych rozdziałów zostały pominięte.

Spis treści

1. Wprowadzenie

Bioinformatyka jest interdyscyplinarną dziedziną nauki, w której techniki informatyczne, matematyczne i statystyczne spotykają się z zagadnieniami natury biologicznej [1]. Metody te są stosowane w badaniach nad budową i funkcją podstawowych składników materii ożywionej, czyli kwasów nukleinowych oraz białek.

Białka pełnią krytyczne dla funkcjonowania komórki role: od enzymatycznych i regulacyjnych, przez strukturalne i transportowe, aż po biorące udział w odpowiedzi układu odpornościowego i rozmnażaniu. Aby mogły prawidłowo realizować powyższe i inne zadania, tworzące je łańcuchy polipeptydowe muszą osiągnąć właściwy kształt, a niekiedy również związać się ze sobą, tworząc kompleks. Pierwszy z tych procesów, czyli formacja struktury trzeciorzędowej białka, jest nazywany zwijaniem lub fałdowaniem [2], a drugi – czwartorzędowej – kompleksowaniem lub dokowaniem [3].

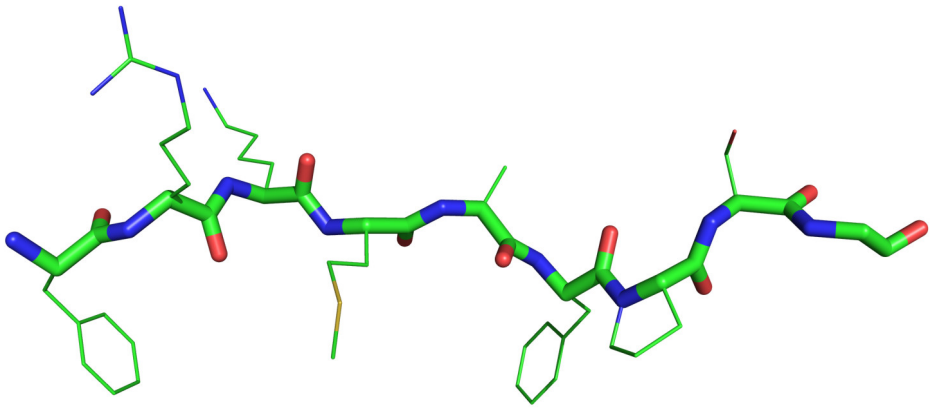
Nieprawidłowości w budowie białek, a przez to w pełnionej przez nie funkcji, są uznawane za przyczynę wielu poważnych i obecnie nieuleczalnych chorób, w tym Alzheimera, Parkinsona i Creutzfeldta-Jakoba [4], których występowanie wiąże się z gromadzeniem w tkankach (głównie w mózgu) nierozpuszczalnych złogów, tak zwanych blaszek amyloidowych [5]. W związku z tym, że schorzenia te prowadzą do zgonu lub znacznego pogorszenia jakości życia, wciąż poszukuje się modeli komputerowych pozwalających dokładnie symulować zjawiska zwijania i kompleksowania białek oraz analizować efekty wpływu na nie leków i innych czynników [6].

Wyzwaniem, przed którym stoi obecnie medycyna jest terapia indywidualna, uwzględniająca specyfikę organizmu pacjenta, a więc nastawiona na opracowywanie planów leczenia obejmujących zagadnienia związane z projektowaniem leków przeznaczonych dla konkretnej osoby [7]. Leki te muszą posiadać wysoką czułość i specyficzność. Innym czynnikiem jaki należy brać pod uwagę jest czas, który musi być odpowiednio krótki ze względu na postępującą chorobę. Dlatego badania nad komputerowymi technikami pozwalającymi na przyspieszenie prac laboratoryjnych związanych z wyborem najbardziej obiecujących kandydatów na leki spośród wielu możliwych cząsteczek są krytyczne do rozwoju spersonalizowanej medycyny [8].

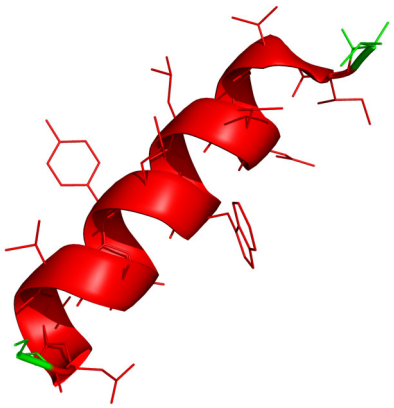
Informacje na temat białek, które może wyprodukować dana komórka są zapisane w sekwencji podwójnej helisy DNA znajdującej się w jej jądrze. Gdy zachodzi potrzeba utworzenia konkretnego białka, następuje transkrypcja kodującego je genu na pojedynczą nić mRNA, która jest następnie wydzielana do cytoplazmy, gdzie wchodzi do jednego z rybosomów. Rybosom – wielki kompleks białkowo-nukleotydowy – dokonuje wówczas translacji kolejnych trójek sekwencji nukleotydowej na sekwencję aminokwasową [9]. Reguły zarządzające przepływem informacji genetycznej zostały zawarte w „centralnym dogmacie biologii molekularnej", sformułowanym w 1956 i zaktualizowanym w 1970 roku przez Francisa Cricka [10]. Zgodnie z nim, informacja może być przekazywana wyłącznie z kwasów nukleinowych do białek, nigdy odwrotnie, lub pomiędzy tymi ostatnimi. Aktualizacja tej hipotezy wynikła z odkrycia mechanizmu odwrotnej transkrypcji [11, 12], stosowanego między innymi przez HIV i inne retrowirusy [13]. Nie jest on jednak sprzeczny z jej pierwotną postacią, gdyż polega wyłącznie na przekazie informacji z RNA do DNA. Podobnie, chorobotwórcze białka prionowe nie zaburzają tego schematu, ponieważ ich sposób działania polega na wymuszaniu zmian konformacyjnych w innych białkach, a nie na produkcji nowych [14, 15]. Czynność tę potrafią wykonywać syntetazy peptydów nierybosomalnych (NRPS) [16, 17], jednak również one nie stanowią tu wyjątku, gdyż tworzone przez nie cząsteczki są zbyt małe, aby mogły uzyskać status struktury białkowej.

Połączone ze sobą kowalencyjnie aminokwasy tworzą łańcuch polipeptydowy (struktura pierwszorzędowa), który po wyjściu z rybosomu zaczyna się zwijać poprzez zmiany wartości kątów dwuściennych wiązań swojego łańcucha głównego [18]. Powoduje to pojawienie się w nim motywów helis α i arkuszy β, stabilizowanych przez wiązania wodorowe (struktura drugorzędowa). Równocześnie łańcuch zapada się w sobie na skutek oddziaływań pomiędzy swoimi atomami i ze środowiskiem wodnym (struktura trzeciorzędowa). Zjawisko to może dotyczyć jego całości, lub niezależnie ewoluujących i indywidualnie zwijających się fragmentów sekwencji, czyli domen. Część białek osiąga swoją postać natywną na etapie samego łańcucha. Inne są natomiast kompleksami kilku takich łańcuchów (struktura czwartorzędowa), utrzymywanymi przez oddziaływania niekowalencyjne. Wizualizacja wszystkich czterech rzędów struktur przykładowego białka `1UJ1` [19] znajduje się na rysunku 1.1.

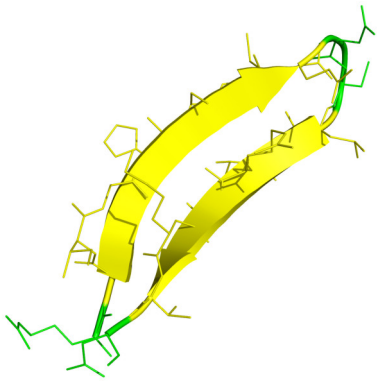
W zależności od liczby tworzących je łańcuchów, kompleksy nazywa się monomerami, dimerami, tetramerami, i tak dalej, a identyczność ich sekwencji określa, czy są homo-, czy też heteromerami. Na tej podstawie wyznacza się stechiometrię całej cząsteczki. Na przykład, może być ona homodimerem (A2), heterodimerem (AB), kompleksem dwóch homodimerów (A2B2), lub dowolną inną kombinacją [20].



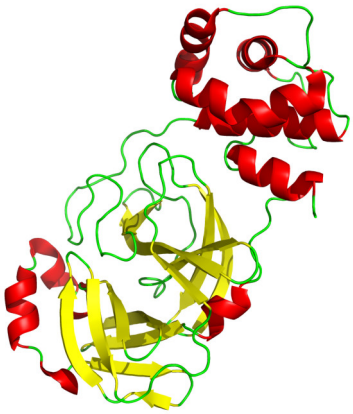
(a) Struktura pierwszorzędowa (pogrubione wiązania wskazują na łańcuch główny).



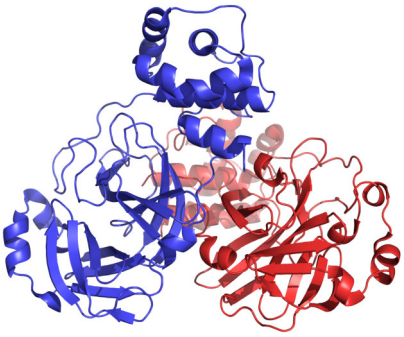
(b) Struktura drugorzędowa (zwinięcie fragmentu łańcucha w motyw helisy α).



(c) Struktura drugorzędowa (zwinięcie fragmentu łańcucha w motyw arkusza β).



(d) Struktura trzeciorzędowa (cały łańcuch zwinięty, wyraźnie widoczne trzy domeny).



(e) Struktura czwartorzędowa (kompleks dwóch łańcuchów, stechiometria A2).

Rysunek 1.1: Wizualizacja czterech rzędów struktury przykładowego białka `1UJ1`. Jest to białko homodimeryczne, posiadające 301 reszt i 3 domeny w każdym łańcuchu.

W jubileuszowym numerze, wydanym w 2005 roku z okazji swojego 125-lecia, czasopismo *Science* umieściło zagadnienia przewidywania struktury i kompleksów białek wśród 125 pytań, na które nauka wciąż nie potrafi udzielić w pełni satysfakcjonujących odpowiedzi [21, 22]. Ponieważ cząsteczki te nadal skrywają wiele tajemnic, w celu stymulowania rozwoju badań mogących przybliżyć ich odkrycie, powołane zostały otwarte, międzynarodowe inicjatywy, realizowane w postaci regularnie organizowanych konkursów. Wyniki wszystkich z nich są publicznie dostępne.

Pierwszym z tych eksperymentów, uznawanym za najbardziej prestiżowy, ponieważ mający na celu odnalezienie modelu przewidującego zwijanie się białek, jest CASP[1] (critical assessment of protein structure prediction) [23]. Odbywa się on co dwa lata, począwszy od 1994 roku. Zadaniem startujących w nim zespołów badawczych jest zaprezentowanie natywnej struktury białka, przewidzianej na podstawie danej sekwencji aminokwasowej. Modele są oceniane pod względem ich zgodności z nieopublikowanym do tej pory wzorcem i układane w rankingu [24, 25]. Do najważniejszych kryteriów tej oceny należy miara GDT-TS (global distance test - total score), stosowana od czwartej edycji w miejscu standardowej miary RMSD (root mean square deviation) ze względu na jej niższą wrażliwość na różnice w mniej istotnych fragmentach łańcucha [26]. Dodatkowo, białka-cele są dzielone na dwie kategorie trudności w zależności od liczby znanych, podobnych do nich sekwencji, decydujących o zakresie możliwości stosowania metod porównawczych.

Na podobnych zasadach funkcjonuje inna inicjatywa, tym razem dotycząca tematu przewidywania struktury kompleksów białkowych – CAPRI[2] (critical assessment of prediction of interactions) [27]. Co roku organizowana jest jedna runda tego eksperymentu – pierwsza odbyła się na początku wieku. Zespoły badawcze biorące w nim udział mają za zadanie wskazać strukturę kompleksu dwóch białek: pary receptor-ligand, których osobne modele trójwymiarowe zostały im przedstawione. Ranking uzyskanych wyników jest opracowywany na podstawie ich porównania z także nieopublikowanym wzorcem przy pomocy analizy map kontaktów niewiążących i wartości RMSD [28, 29]. CAPRI również wyróżnia dwie kategorie trudności białek-celów, klasyfikując je w zależności od tego, czy dane wejściowe zostały uzyskane z ich gotowego kompleksu, czy też przed jego utworzeniem [30]. Drugi z tych przypadków jest bardziej wymagający, ponieważ uczestnicy konkursu muszą liczyć się z możliwymi zmianami konformacyjnymi w łańcuchach.

[1] http://predictioncenter.org
[2] http://www.ebi.ac.uk/msd-srv/capri

Trzecim eksperymentem, nawiązującym w swojej idei do powyższych, jest CAFA[3] (critical assessment of function annotation) [31]. Został on powołany względnie niedawno – w 2010 roku. Jego celem jest poszukiwanie coraz dokładniejszych metod komputerowego przewidywania funkcji białek i procesów biologicznych, w których biorą one udział. Przyjęto, że realizacja tego zadania będzie polegać na próbach jak najdokładniejszego przydzielania sekwencji do klas określonych przez konsorcjum Gene Ontology [32]. Każda runda CAFA trwa dwa lata, a po niej następuje rok przerwy. Jej uczestnikom jest przedstawiane kilkadziesiąt tysięcy sekwencji o jeszcze nieokreślonych eksperymentalnie funkcjach. Po tym jak zakończą oni ich przewidywanie, część białek zostaje przekazana do laboratoriów, które w trakcie kolejnych miesięcy je opracowują. Dzięki temu, możliwa staje się ocena sprawności działania zastosowanych do przewidywania algorytmów, wyrażana w przestrzeni krzywych ROC. Już po pierwszej rundzie stwierdzono, że algorytmy te są w stanie osiągnąć wyższą dokładność od tradycyjnych podejść, takich jak poszukiwanie podobieństw z sekwencjami o znanych funkcjach za pomocą programu BLAST [33, 34].

Powszechnie przyjętym miejscem publikacji trójwymiarowych modeli uzyskanych eksperymentalnie struktur białek jest otwarta i ogólnodostępna baza Protein Data Bank (PDB) [35, 36]. Powstała ona na początku lat 70-tych w Brookhaven National Laboratory w USA, a obecnie zarządza nią organizacja Worldwide PDB (wwPDB[4]) [37], która skupia w sobie trzy instytucje zajmujące się przetwarzaniem i rozpowszechnianiem tego rodzaju danych: RCSB PDB[5], PDBe[6] oraz PDBj[7].

Struktury umieszczane w bazie PDB posiadają identyfikatory składające się z cyfry (od 1 do 9) oraz trzech cyfr lub liter, na przykład `1UJ1`. W 2017 roku było ich już ponad 130 tysięcy. Liczba ta systematycznie wzrasta, lecz nadal stanowi jedynie niewielki ułamek z przeszło 65 milionów uzyskanych do tej pory sekwencji aminokwasowych dostępnych w bazie UniProt[8] [38, 39]. Została ona utworzona w 2003 roku w wyniku ujednolicenia baz Swiss-Prot/TrEMBL [40] oraz PIR-PSD [41]. Zdobywaniem informacji na temat domen w białkach i klasyfikacji ich rodzin zajmują się natomiast inicjatywy CATH[9] [42, 43] oraz SCOP[10] i SCOP2[11] [44, 45].

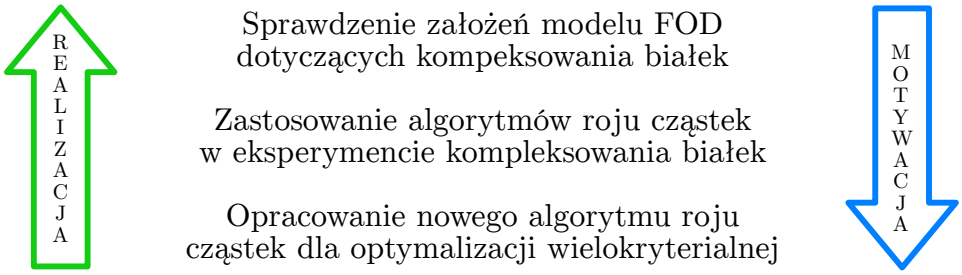
[3] http://biofunctionprediction.org
[4] http://www.wwpdb.org
[5] http://www.rcsb.org
[6] http://www.pdbe.org
[7] http://www.pdbj.org
[8] http://www.uniprot.org
[9] http://www.cathdb.info
[10] http://scop.mrc-lmb.cam.ac.uk
[11] http://scop2.mrc-lmb.cam.ac.uk

1.1. Motywacja

Sposób w jaki komórki produkują białka jest powszechnie znany. Nie są natomiast dokładnie znane siły powodujące, że łańcuch polipeptydowy o konkretnej sekwencji, który właśnie opuścił rybosom przyjmuje w ciągu kilku chwil określony kształt i ewentualnie wiąże się z innymi łańcuchami [46]. Problem ten wynika stąd, że istnieje wiele czynników mających wpływ na procesy związane z białkami. Na przykład, nie wszystkie z nich podążają tą samą ścieżką: część występuje w formie kompleksu, a innym wystarcza do pełnienia swojej funkcji wyłącznie osiągnięcie struktury trzeciorzędowej. Przedstawicielem pierwszej z tych grup jest hemoglobina [47], a drugiej – lizozym [48]. Oprócz tego, niektóre łańcuchy zwijają się samodzielnie [49], natomiast inne wymagają do uzyskania postaci natywnej obecności białek opiekuńczych, które zapewniają im odpowiednie w danym momencie środowisko [50]. Dochodzi tu również kwestia domen, stanowiących nie tylko ewolucyjnie odrębne i funkcjonujące niezależnie fragmenty łańcuchów, ale mogących także powstawać pomiędzy nimi, tworząc „quasi-domeny" [51], lub nawet być między nimi wymieniane [52]. Warto również wspomnieć o klasie białek częściowo nieuporządkowanych, charakteryzujących się brakiem w pełni wykształconej struktury trzeciorzędowej, ale posiadających pełną funkcjonalność [53]. Przykłady te pokazują jak różnorodny, a przez to niełatwy do modelowania, jest świat cząsteczek, którymi zajmuje się bioinformatyka.

Pomimo ciągłych postępów w badaniach nad procesami zwijania i kompleksowania białek [54, 55], wciąż nie udało się opracować ogólnego modelu zdolnego do wystarczająco dokładnego opisania sił nimi zarządzających [56]. Ze względu na skomplikowane relacje pomiędzy ich funkcją, strukturą i sekwencją [57] należy liczyć się z tym, że taki uniwersalny model nie powstanie w najbliższej przyszłości [58]. Nieustępowanie w jego poszukiwaniu może jednak doprowadzić do dokładniejszego poznania kolejnych rodzin tych cząsteczek.

Modelem opisującym reakcje zachodzące w białkach w ujęciu ich oddziaływań hydrofobowych ze środowiskiem wodnym jest model rozmytej kropli oliwy (fuzzy oil drop, FOD) [59]. Autor niniejszej rozprawy bierze udział w badaniach przy użyciu tego modelu i nad jego rozwojem [60]. Założenia modelu FOD dotyczące wpływu hydrofobowości na kształtowanie się struktury czwartorzędowej białek nie zostały jeszcze sprawdzone eksperymentalnie. Autor rozprawy podjął się tego zadania, opracowując eksperyment przewidywania tworzenia się kompleksów typu białko-białko, umożliwiający stwierdzenie, czy model FOD może być stosowany jako kryterium optymalizacyjne w symulacjach procesów biologicznych.



Rysunek 1.2: Graficzna prezentacja relacji pomiędzy głównymi celami pracy. Niewymieniony jest tu cel poboczny – modyfikacja sposobu obliczeń modelu FOD.

1.2. Cele pracy

Niniejsza rozprawa doktorska zakłada trzy główne cele do realizacji. Każdy z nich stanowi motywację dla następnego, a każdy następny umożliwia realizację wcześniejszego, co zostało przedstawione w sposób graficzny na rysunku 1.2.

1. Pierwszym celem jest sprawdzenie założeń modelu rozmytej kropli oliwy (FOD) dotyczących wpływu oddziaływań hydrofobowych na proces tworzenia się kompleksów typu białko-białko. Zdecydowano, że realizacja tego zadania będzie polegać na przeprowadzeniu eksperymentu *ab initio* przewidywania struktury czwartorzędowej 200 białek homodimerycznych wybranych z bazy PDB i porównaniu uzyskanych wyników z wynikami uzyskanymi przy pomocy chemicznych pól siłowych, na których reprezentanta zostało wybrane pole ECEPP/3 [61–63].

2. Drugim celem jest użycie algorytmów opartych na sposobie działania roju cząstek [64] do wykonania powyższego eksperymentu i sprawdzenie, czy możliwe jest ich skutecznie stosowanie w symulacjach procesów związanych z białkami.

3. Trzecim celem jest opracowanie autorskiego algorytmu optymalizacji wielokryterialnej opartego na sposobie działania roju cząstek (MOSF), umożliwiającego przeprowadzenie w powyższym eksperymencie symulacji równoczesnego wpływu sił opisywanych przez model FOD i pole ECEPP/3. W tym celu mieści się również wykazanie ogólnej przydatności tego algorytmu poprzez porównanie zwracanych przez niego wyników z wynikami powszechnie stosowanych metod oraz przedstawienie nowej funkcjonalności nieoferowanej przez te metody.

Oprócz powyższych celów, niniejsza rozprawa doktorska posiada cel poboczny, jakim jest modyfikacja sposobu obliczeń wykonywanych przez model FOD w celu umożliwienia jego efektywnego stosowania jako kryterium optymalizacyjnego.

1.3. Dotychczasowy stan wiedzy

Stanem, od którego rozpoczyna się modelowanie *in silico*[12] powstawania struktury trzeciorzędowej białek jest ich sekwencja. Wynika to z obserwacji, że zwijają się one spontanicznie, a w niektórych przypadkach potrafią nawet wracać do swojej aktywnej biologicznie postaci po usunięciu czynnika je denaturującego [66]. Na tej podstawie postawiona została hipoteza, że sekwencja powinna zawierać całość informacji potrzebnej siłom kierującym reakcją zwijania białek do jej przeprowadzenia [67]. Hipotezę tę popierają dwie dodatkowe obserwacje. Pierwszą z nich jest bardzo krótki czas w jakim białka osiągają swoją strukturę trzeciorzędową: rzędu mili- lub nawet mikrosekund [68]. Drugą natomiast – ogromna liczba wszystkich możliwych konformacji (wzajemnych położeń ich atomów), wśród których się one znajdują. Zakładając, w dużym w uproszczeniu, że zwijanie białek polega wyłącznie na zmianach wartości kątów φ i ψ w łańcuchu głównym [69], wynosi ona $p^{2(n-1)}$, gdzie p jest liczbą stopni swobody obrotu wokół pojedynczego wiązania, a n – długością sekwencji. Łatwo można sprawdzić, że czas potrzebny na wyczerpujące sprawdzenie wszystkich konformacji osiągalnych przez niedużą, 100-aminokwasową cząsteczkę, już przy 2 możliwych ustawieniach każdego kąta i tempie miliarda kandydatów na sekundę, wielokrotnie przekracza przyjmowany wiek Wszechświata. Jako pierwszy przedstawił te wyliczenia wraz wnioskami pod koniec lat 60-tych Cyrus Levinthal [70], od którego nazwiska wzięła się nazwa wynikającego z nich „paradoksu”.[13]

Aby można było mówić o paradoksie, należy wpierw założyć, że białka mogą dowolnie zmieniać swój kształt, swobodnie przechodząc pomiędzy równie prawdopodobnymi konformacjami, oceniając, która z nich jest tą właściwą. Fakt, że szybko kierują się ku strukturom natywnym, sugeruje jednak, że niektóre z tych stanów są faworyzowane. Na tej podstawie, Levinthal zaproponował hipotezę „ścieżek” zwijania (folding pathway) – odgórnie ustalonych, pośrednich etapów tego procesu, w których lokalne oddziaływania pomiędzy fragmentami cząsteczki powodują coraz silniejsze ograniczenie jej przestrzeni konformacyjnej [71]. Zapoczątkowało to serie badań w poszukiwaniu tych pośredników [72–74], w efekcie których opracowano nową hipotezę „leja” zwijania (folding funnel) [75–77].

[12] Pojęcie *in silico* („w krzemie”) oznacza eksperyment wykonywany w systemie komputerowym i nawiązuje do stosowania tego pierwiastka do masowej produkcji półprzewodnikowych układów scalonych. Zaczęło się ono pojawiać się w publikacjach z przełomu lat 80-tych i 90-tych w celu odróżnienia symulacji od eksperymentów *in vivo* i *in vitro* [65].

[13] Słowo „paradoks” jest ujęte w cudzysłów w celu podkreślenia tego, że – paradoksalnie – sam Levinthal nie widział w tych wyliczeniach paradoksu, proponując od razu swoją hipotezę.

Hipoteza, według której białka zwijają się zgodnie z informacją zawartą w ich sekwencji, będąca podstawą badań nad tym procesem [78], została sformułowana przez Christiana Anfinsena na podstawie eksperymentów, które przeprowadził ze współpracownikami w latach 60-tych [66, 67]. Nosi ona również nazwę „hipotezy termodynamicznej", gdyż zakłada, że struktura natywna każdej z tych cząsteczek, o ile przebywa ona w swoim naturalnym środowisku, jest określona przez całość oddziaływań pomiędzy atomami występującymi w opisującym ją układzie i znajduje się w stanie odpowiadającym minimum energii swobodnej Gibbsa (G) [79, 80]:

$$\Delta G = \Delta H - T\Delta S \tag{1.1}$$

W powyższym równaniu, H oznacza entalpię, której zmiana reprezentuje bilans energii układu przed i po reakcji, natomiast S jest entropią, wyrażającą prawdopodobieństwo jego fizycznego stanu, interpretowaną jako miara jego nieuporządkowania [81]. Minimalizacja pierwszej z nich oraz maksymalizacja drugiej nadaje procesom chemicznym spontaniczność [82].

Ze względu na wrażliwość komórek na wysoką temperaturę (T), energia cieplna nie ma zastosowania w większości zachodzących w nich reakcji [82]. Zamiast niej, do przeprowadzania procesów niespontanicznych wykorzystywana jest energia chemiczna, której biologiczny nośnik, adenozyno-5'-trifosforan (ATP) jest produkowany przez mitochondria w końcowym etapie łańcucha oddechowego poprzez wysoce spontaniczne utlenianie wodoru. Rośliny również produkują ten związek, ale w procesie odwrotnym do syntezy wody. Ponieważ reakcja ta jest niespontaniczna, energia potrzebna do jej przeprowadzenia jest uzyskiwana dzięki fotosyntezie.

Zwijanie białka polega na jego przejściu z konformacji niezwiniętej do zwiniętej, czyli z jednego stanu równowagi do drugiego, ale znajdującego się niżej od niego w sensie energii Gibbsa. Odwrócenie tego procesu jest możliwe, na przykład poprzez dostarczenie do układu dużej ilości ciepła. Powoduje to jednak denaturację białek i w konsekwencji utratę przez nie możliwości pełnienia funkcji.

Zgodnie z hipotezą Anfinsena, energia układu zawierającego białko jest wyrażana przez oddziaływania pomiędzy obecnymi w nim atomami: elektrostatyczne, van der Waalsa i hydrofobowe, a także potencjał wiązań wodorowych i torsyjny [83]. Za składową entropową uznaje się natomiast konfigurację tego układu. Jej zmiana jest możliwa poprzez translacje i obroty całych cząsteczek, a także zmiany konformacyjne wewnątrz nich, polegające na zmianach długości wiązań kowalencyjnych oraz tworzonych przez nie kątów płaskich i dwuściennych.

Wszystkie możliwe konfiguracje układu tworzą przestrzeń konfiguracyjną [84]. Liczba jej wymiarów, zależna od liczby stopni swobody, została już przybliżona podczas opisu „paradoksu" Levinthala. W rzeczywistości jest ona o wiele wyższa, choć w tym momencie nie ma to znaczenia. Zamiast tego, należy założyć, że podobne konfiguracje znajdują się w tej przestrzeni blisko siebie oraz że istnieje funkcja, która każdej z nich przypisuje odpowiadającą jej energię. Krajobraz wartości tej funkcji przyjmuje wówczas postać „leja", na którego spodzie powinna znajdować się struktura natywna. Z powodu faworyzowania przez układ niskich energii, białko „stacza" się po jego powierzchni, lokalnie zwijając się i rozwijając, aż do momentu zatrzymania w stanie równowagi. Sytuacja podczas procesu kompleksowania jest podobna – podlega on bowiem wpływowi tych samych oddziaływań [85], z tą różnicą, że zamiast zwijać się, łańcuchy łączą się ze sobą przy pomocy oddziaływań niekowalencyjnych.

Powierzchnia „leja" zwijania nie jest gładka – znajdują się na niej minima i maksima lokalne.[14] Nierówności te, występujące przede wszystkim w jego głębi, wynikają stąd, że zwinięty łańcuch posiada niższą energię od rozwiniętego, ale równocześnie charakteryzuje się większym uporządkowaniem. Oznacza to, że osiągnięcie struktury natywnej przez białko jest wynikiem równoczesnego obniżania entalpii oraz przeciwdziałającej temu i spowalniającej całą reakcję entropii [86]. Efektem tego jest występowanie barier na krajobrazie wartości energii swobodnej Gibbsa, uniemożliwiających białkom docieranie do niektórych konfiguracji [87] oraz powodujących dwuetapowe zwijanie małych białek [86, 88]. Niektóre cząsteczki, takie jak hemoglobina, mogą również posiadać kilka jej minimów [89, 90].

1.3.1. Mechanika molekularna

Do opisu układów cząsteczek chemicznych powszechnie stosuje się trzy techniki: metody *ab initio*, metody pół-empiryczne oraz mechanikę molekularną [91].

Metody *ab initio* („od początku") bazują na chemii kwantowej. Ich nazwa wynika stąd, że modele, którymi się posługują nie są dopasowywane do danych uzyskanych eksperymentalnie. Jedną z najczęściej stosowanych metod tego typu jest formalizm Hartree-Focka. Do opisu układu służą w nim orbitale atomowe i cząsteczkowe, co pozwala na osiągnięcie dokładności na poziomie elektronów. Kosztem tej dokładności jest jednak bardzo wysoka złożoność obliczeniowa, ograniczająca liczbę atomów w układzie do kilkudziesięciu (poniżej 100). Już względnie niewielkie białko, składające się ze 100 reszt, znajduje się zdecydowanie powyżej tego limitu.

[14] Dobrze widoczne na trójwymiarowych rysunkach z pracy Dilla i Chana [84].

Metody pół-empiryczne są modyfikacjami formalizmu HF i polegają na jego upraszczaniu poprzez wprowadzanie funkcji korzystających z empirycznie wyznaczonych parametrów. Ma to na celu przyspieszenie obliczeń kosztem ich dokładności. Jednym ze sposobów tego upraszczania jest ograniczenie się do bezpośredniego opisu wyłącznie elektronów walencyjnych. Pozostałe są uwzględnianie wraz z jądrem przez odpowiednie funkcje. Pozwala to na opisywanie układów zawierających setki atomów, czyli rząd wielkości powyżej „czystej" chemii kwantowej. Przykładem klasycznych metod pół-empirycznych jest CNDO [92–94].

Ze względu na ograniczenia powyższych reprezentacji, do opisu makrocząsteczek takich jak białka, stosuje się zazwyczaj mechanikę molekularną. Metoda ta jest spośród tu wymienionych najmniej dokładna, ale za to jako jedyna dobrze radzi sobie z układami zawierającymi tysiące atomów, bez potrzeby stosowania superkomputerów. Na przykład, kompleks `1UJ1` z rysunku 1.1 składa się z ponad 9000 z nich.

Idea mechaniki molekularnej opiera się na przybliżeniu Borna-Oppenheimera, zakładającym, że ruch jąder atomowych może być rozpatrywany oddzielnie od ruchu elektronów, które ze względu na swoją szybkość powinny być w stanie się do nich dostosowywać. Pozwala to na ich wspólnie przybliżanie w postaci sfer. Środek takiej sfery odpowiada położeniu jądra, a promień – promieniowi van der Waalsa wyznaczonemu dla danego pierwiastka [95, 96]. Wiązania pomiędzy atomami są natomiast interpretowane jako sprężynki, bez możliwości modelowania ich powstawania i rozrywania. Dzięki temu, opis ruchu i zmian konformacyjnych w cząsteczkach staje się możliwy za pomocą koncepcji mechaniki klasycznej (obroty, drgania i translacje), Na podstawie sfer van der Waalsa można również przybliżać objętość i kształt powierzchni cząsteczek, w tym ich część dostępną dla środowiska wodnego [97].

1.3.2. Chemiczne pola siłowe

Do zadań mechaniki molekularnej należy obliczanie struktur cząsteczek i odpowiadającej im energii, a także jej minimalizacja. W tym celu potrzebne są trzy narzędzia: pole siłowe, parametryzacja oraz algorytm optymalizacyjny [91].

Chemiczne pole siłowe jest funkcją, która na podstawie zbioru atomów i wiązań oraz parametryzacji zwraca wartość energii układu, E, będącej sumą kilku potencjałów. Typowy schemat jej równania wygląda następująco [98]:

$$E = \underbrace{E_b + E_a + E_t}_{\text{kowalencyjne}} + \underbrace{E_e + E_n + E_h}_{\text{niekowalencyjne}} \quad \left[\frac{\text{kcal}}{\text{mol}}\right] \tag{1.2}$$

gdzie:

E_b = energia potencjału długości wiązań (ściskania)
E_a = energia potencjału wartości kątów płaskich (wyginania)
E_t = energia potencjału wartości kątów dwuściennych (obracania)
E_e = energia potencjału oddziaływań elektrostatycznych (ładunek-ładunek)
E_n = energia potencjału oddziaływań van der Waalsa (dipol-dipol)
E_h = energia potencjału wiązań wodorowych (symbolizowanych przez $\mathrm{H}\cdots\mathrm{X}$)

Należy tutaj zwrócić uwagę na fakt, że oddziaływania van der Waalsa bywają również nazywane oddziaływaniami niewiążącymi (nonbonded interactions), stąd symbol E_n. Jednak potencjały elektrostatyczny i wiązań wodorowych również nie są obliczane dla wiązań kowalencyjnych, dlatego w celu uniknięcia nieporozumień, określenie „niewiążące" będzie stosowane wyłącznie do pojęcia kontaktów niewiążących, czyli faktu znajdowania się atomów w bliskiej odległości.

Energia układu wyrażona przy pomocy równania 1.2 może być również podzielona na dwie składowe – wewnątrzcząsteczkową (E_{intra}) i międzycząsteczkową (E_{inter}). W pierwszym przypadku pod uwagę brane są wiązania i pary atomów należące do indywidualnych cząsteczek, natomiast w drugim – potencjały niekowalencyjne opisujące oddziaływania, które występują pomiędzy tymi cząsteczkami.

Parametryzacja pola jest zestawem stałych charakteryzujących atomy należące do analizowanego układu. W przypadku białek, przypisane im wartości zależą między innymi od tego, w których resztach się one znajdują, a nawet tego, czy reszty te są położone w środku sekwencji łańcucha, czy też na którymś z jej końców. Parametryzację wyznacza się poprzez ekstrapolację wyników obliczeń chemii kwantowej lub eksperymentów przeprowadzanych na małych cząsteczkach. Ze względu na dopasowanie parametryzacji do konkretnego modelu, porównywanie ustawień różnych pól siłowych nie ma zazwyczaj sensu. Identyczna sytuacja dotyczy zwracanych przez nie wartości energii [99].

Każdemu potencjałowi są przypisywane typy oddziaływań, zapisywane w postaci dwóch indeksów. Pierwszym z nich jest zawsze 1, co oznacza bieżący atom, natomiast drugim – liczba określająca jak daleko w sensie wiązań kowalencyjnych znajduje się od niego w cząsteczce drugi z rozpatrywanej pary. Na przykład: 1-2 oznacza jedno wiązanie, 1-3 – dwa, 1-4 – trzy, 1-5 – cztery, i tak dalej. Typy oddziaływań określają, jaka relacja musi występować pomiędzy atomami, aby zmiana ich wzajemnego położenia miała wpływ na wartość danego potencjału, pod warunkiem, że energia tej zmiany nie jest w całości uwzględniona przez inne potencjały:

E_b – 1-2 (zmiana długości wiązania kowalencyjnego)

E_a – 1-3 (zmiana wartości kąta płaskiego)

E_t – 1-4 (zmiana wartości kąta dwuściennego, o ile jest możliwa)

E_e – 1-4, 1-5 i wyższe oraz pomiędzy cząsteczkami

E_n – 1-4, 1-5 i wyższe oraz pomiędzy cząsteczkami

E_h – 1-4, 1-5 i wyższe oraz pomiędzy cząsteczkami

Do oddziaływań 1-4 zaliczane są tylko te kąty dwuścienne, w których istnieje możliwość obrotu wokół wiązania pomiędzy atomami o indeksach 2 i 3. Przykładem sytuacji nie spełniającej tego warunku jest pierścień aromatyczny.

Wewnątrzcząsteczkowe potencjały niekowalencyjne oblicza się dla oddziaływań typu 1-4, 1-5 i wyższych – w pozostałych ustępują silniejszym od nich potencjałom kowalencyjnym. Wartości współczynników odpychania w potencjale van der Waalsa mogą być w przypadku oddziaływań typu 1-4 dodatkowo skalowane (osłabiane) ze względu na bliskie położenia atomów [100]. Międzycząsteczkowe potencjały niekowalencyjne nie mają narzuconych tego rodzaju ograniczeń.

Mając do dyspozycji układ zawierający badane białka, pole siłowe i parametryzację można przystąpić do wyznaczania wartości energii jego konfiguracji. Złożoność obliczeniowa tej procedury jest liniowa ze względu na liczbę wiązań (E_b, E_a i E_t) oraz kwadratowa ze względu na liczbę atomów (E_e, E_n i E_h). Z tego powodu, pomimo, że mechanika molekularna już w teorii stanowi duże przybliżenie rzeczywistego opisu cząsteczek, aby umożliwić praktyczne zastosowanie pól siłowych jako kryteriów optymalizacyjnych, niezbędne okazuje się podjęcie dalszych czynności skracających czas trwania obliczeń energii. Jednym ze sposobów osiągnięcia tego celu jest zmniejszanie liczby par atomów rozpatrywanych przez potencjały niekowalencyjne.

Ponieważ wszystkie potencjały niekowalencyjne słabną wraz z odległością pomiędzy atomami, najprostsze rozwiązanie polega na wprowadzeniu promienia odcięcia [101]. Pary znajdujące się od siebie w przestrzeni dalej niż wynosi jego długość są wówczas pomijane. W zależności od przyjętej wartości tego parametru, skutkuje to znacznym przyspieszeniem obliczeń, ale powoduje jednocześnie pojawianie się artefaktów w krajobrazie energetycznym i nieciągłości funkcji pola w jej pobliżu, co zmienia położenie minimum globalnego oraz może utrudnić jego poszukiwanie [102]. Najbardziej podatny na te zjawiska jest malejący liniowo wraz z odległością potencjał elektrostatyczny. Pozostałe dwa (van der Waalsa i wiązań wodorowych) zanikają wielokrotnie szybciej, dlatego stosowanie promienia odcięcia podczas obliczeń związanej z nimi energii jest powszechnie przyjętą praktyką [103].

Przyjmuje się, że akceptowalna dokładność obliczeń energii jest osiągalna wtedy, gdy promień odcięcia wynosi przynajmniej 12 Å [104], zarówno w białkach [105] jak i w kwasach nukleinowych [106]. W układach z okresowymi warunkami brzegowymi [107], oddziaływania elektrostatyczne dalekiego zasięgu są dodatkowo modelowane przy pomocy sumowania Ewalda [108], lub jego bardziej wydajnej obliczeniowo modyfikacji – particle mesh ewald (PME) [109].

Innym sposobem skrócenia czasu obliczeń energii jest uproszczenie reprezentacji cząsteczek [110, 111]. Klasyczne pola siłowe, analizujące wszystkie atomy, są nazywane polami typu all atom (AA) lub explicit atom (EA). Zakładając poprawność ich modeli, są one najbliższe rzeczywistości, oczywiście na tyle, na ile pozwala sama mechanika molekularna oraz dokładność danych wejściowych. Jednocześnie, pola te są najmniej wydajne obliczeniowo, przez co podczas symulacji dużych cząsteczek potrzebne jest stosowanie metod przedstawionych w poprzednich akapitach. Alternatywne pola „zjednoczone" (united atom, UA) łączą atomy węgla ze związanymi z nimi atomami wodoru,[15] tworząc pseudo-atomy. Ponieważ wodór stanowi około 50% wszystkich atomów w białkach, prowadzi to do znacznego zmniejszenia ich liczby. Jeszcze dalej posuwają się w tym względzie pola „gruboziarniste" (coarse-grained, CG), które ujednolicają całe łańcuchy boczne lub reszty. Efektem tego jest ograniczenie rozmiaru przestrzeni konformacyjnej białka kosztem zmiany równania energii opisującego inny układ, a więc potencjalnie posiadającej inny krajobraz wartości i inne minima niż mechanika molekularna w ujęciu klasycznym (AA).

Jako pierwsze zostały opracowane w latach 70-tych pola siłowe typu all atom. Spośród nich, do powszechnie znanych należą (w kolejności chronologicznej) [112]:

- ECEPP (empirical conformational energy program for peptides) [61–63],
- CHARMM (chemistry at harvard macromolecular mechanics) [113, 114],
- AMBER (assisted model building with energy refinement) [115, 116],
- OPLS-AA (optimized potentials for liquid simulations) [117, 118].

Przykładami pól stosujących uproszczoną reprezentację są natomiast „zjednoczone" OPLS-UA [119] oraz „gruboziarniste" UNRES (united-residue) [120, 121].

Implementacje powyższych funkcji znajdują się w dedykowanych dla nich pakietach obliczeniowych, a niektóre także w otwartym i darmowym oprogramowaniu GROMACS[16] (groningen machine for chemical simulations) [122, 123].

[15] Z wyłączeniem atomów węgla i wodoru, które mogą brać udział w tworzeniu wiązań wodorowych.
[16] http://www.gromacs.org

1.3.3. Modele wody i hydrofobowość

Bio-cząsteczki, takie jak białka, funkcjonują wyłącznie w wodzie. Oczywiście, nie są w niej osamotnione *in vivo*, ale to właśnie ona wyznacza sens ich istnienia. Z tego powodu, relacja białek z wodą ma istotne znaczenie dla badań nad ich pełnioną przez nie funkcją. Według Autorów modelu FOD, obecnie stosowane chemiczne pola siłowe traktują środowisko wodne w sposób niewystarczający, na przykład modelując oddziaływania z nim jako jeszcze jeden potencjał niekowalencyjny [124].

Stosowane są obecnie dwa sposoby reprezentacji wody w mechanice molekularnej i związanych z rozpuszczaniem się w niej cząsteczek zmian energii układu. Można je określić mianem dyskretnego lub jawnego oraz ciągłego lub niejawnego. Aktualny przegląd tych modeli znajduje się w pracy Skynera i współpracowników [125]. Tutaj przedstawione są tylko konceptualne różnice między nimi.

W modelu dyskretnym, opisywany układ jest zanurzany w całości w „pojemniku" szczelnie wypełnionym cząsteczkami wody (sztywnymi lub elastycznymi), z dodatkiem odpowiedniej liczby jonów Na+ lub Cl-. Typowo przyjmuje on kształt sześcianu. Stosowane przez dane pole siłowe potencjały niekowalencyjne mogą być wówczas użyte do modelowania oddziaływań pomiędzy wodą a białkiem. Wadą tego rozwiązania jest znacznie zwiększenie liczby par atomów oraz liczby stopni swobody konfiguracji układu, co przekłada się na znaczny wzrost czasu obliczeń energii.

W modelu ciągłym, środowisko wodne jest opisywane przez osobną funkcję jako nieskończone medium otaczające ze wszystkich stron znajdujące się w nim cząsteczki. Funkcja ta musi być więc dostosowana do pozostałych potencjałów konkretnego pola siłowego w celu połączenia ich we wspólnym równaniu. Dzięki zastąpieniu indywidualnych cząsteczek wody izotropowym polem elektrycznym uzyskuje się znaczne krótszy czas obliczeń energii niż w modelach dyskretnych, kosztem dokładności. Inne podejścia tego typu zajmują się analizą powierzchni białek, bazując na założeniu, że energia ich rozpuszczania jest proporcjonalna do jej części dostępnej dla środowiska wodnego [126]. Oblicza się ją poprzez toczenie sfery-próbnika (typowo o promieniu 1,4 Å, odpowiadającemu cząsteczce H_2O) po sferach van der Waalsa atomów, przybliżanych przez przez zbiory punktów rozmieszczone w pseudo-jednorodny sposób. W tym celu może być wykorzystana spirala Fibonacciego [127, 128].

Niezależnie od wybranego sposobu reprezentacji wody, relacje pomiędzy nią a białkiem w ujęciu pól siłowych sprowadzają się do lokalnych oddziaływań atomów, w których bierze udział zewnętrzna część cząsteczki [129]. Przedstawione dalej modele uznają jednak, że należy brać pod uwagę wszystkie tworzące je reszty.

Hydrofobowość / hydrofilność jest pojęciem mającym większe znaczenie w dziedzinie biochemii niż cząsteczek chemicznych. W przypadku białek wiąże się ona z powinowactwem aminokwasów do środowiska wodnego, wynikającym z polarności ich łańcuchów bocznych.

Obecność reszt hydrofobowych na powierzchni białka przeciwdziała jej rozpuszczaniu w wyniku wymuszania strukturalizacji cząsteczek wody znajdujących się w pobliżu [82]. Układ ma wówczas dwie możliwości usunięcia tego niekorzystnego termodynamicznie efektu entropowego. Pierwsza polega na zakopaniu (ukryciu) reszt hydrofobowych wewnątrz struktury białka oraz równoczesnej ekspozycji (odkryciu) reszt hydrofilnych na zewnątrz. Drugą możliwością jest natomiast utworzenie kompleksu z inną cząsteczką, powodującego wyparcie wody z miejsc, w których nastąpiło złączenie ich powierzchni ze sobą. Oddziaływania hydrofobowe przyczyniają się więc do samoorganizacji układu, przez co uznaje się je za jedne z najważniejszych sił wpływających na białka, choć wciąż nie zostały dokładnie poznane [130, 131].

Ponieważ woda otacza zanurzone w niej cząsteczki ze wszystkich stron, relacja pomiędzy nimi przybierają formę pola, które z tego powodu będzie w niniejszej rozprawie doktorskiej nazywane polem zewnętrznym. Dla kontrastu, chemiczne pola siłowe będą nazywane wewnętrznymi, gdyż skupiają się przede wszystkim na energii wewnętrznej układu, nadając jej priorytet przed efektami entropowymi.

Pod koniec lat 50-tych, w nawiązaniu do wcześniejszej idei „icebergów” [132], Walter Kauzmann zaproponował model wpływu środowiska wodnego na cząsteczkę białka będący analogią do zachowania kropli oleju [133, 134]. Umieszczona w wodzie, dąży bowiem do minimalizacji powierzchni kontaktu pomiędzy obydwoma cieczami. W sensie tego modelu, we wnętrzu struktury białka powinny znajdować się reszty hydrofobowe tworzące jądro hydrofobowe, szczelnie otoczone przez płaszcz złożony z reszt hydrofilnych, zapewniający całości stabilność i rozpuszczalność.

Hydrofobowość może być rozmieszczona w różnych miejscach w sekwencji łańcucha polipeptydowego. Dlatego w celu zbliżenia się do stanu opisywanego przez model Kauzmanna, muszą nastąpić w tym łańcuchu zmiany konformacyjne, ostatecznie doprowadzające do jego zwinięcia, a następnie – jeżeli zachodzi taka potrzeba – utworzenia kompleksu z innymi łańcuchami. Jest to najważniejsza różnica w interpretacji wpływu środowiska wodnego względem pól siłowych, polegająca na braniu pod uwagę statusu białka jako całości. Pomimo wyników badań potwierdzających sens idei tego modelu [135–137], nie znalazł on swojego zapisu w tych polach. Otworzył jednak drogę do powstania nowych, bardziej zaawansowanych sposobów interpretacji oddziaływań hydrofobowych. Jednym z nich jest omówiony poniżej model FOD.

Model rozmytej kropli oliwy (fuzzy oil drop, FOD) [59, 60] rozszerza ideę modelu Kauzmanna, zastępując binarny podział na jądro i płaszcz charakterystyką ciągłą. Korzysta w tym celu z dwóch rozkładów statusu hydrofobowości reszt: obserwowanego oraz teoretycznego. Dzięki ich porównaniu możliwa staje się analiza całego białka (lub jego wybranych fragmentów) pod względem stabilności, genezy, pełnionej funkcji oraz interakcji z innymi cząsteczkami.

Rozkład hydrofobowości obserwowanej modelu FOD prezentuje postrzeganą zmianę statusu hydrofobowości własnej reszt (pobieranej jako parametr tego modelu ze skal hydrofobowości) po ich osadzeniu w zwiniętej lub zwijającej się cząsteczce białka. Reszty, które znajdą się w sąsiedztwie względnie hydrofobowym dają się wówczas zauważyć jako bardziej hydrofobowe i odwrotnie.

Rozkład hydrofobowości teoretycznej przedstawia natomiast sytuację tych samych reszt, ale w ujęciu zbliżonym do modelu Kauzmanna. Rozumiana jest przez to sytuacja, w której reszty hydrofobowe przebywają w środku cząsteczki białka, przybliżanej za pomocą dopasowanej do niej elipsoidy „kropli", a hydrofilne w pobliżu jej powierzchni. Do modelowania wynikającego stąd spadku hydrofobowości służy w modelu FOD trójwymiarowa funkcja Gaussa.

Białko o obserwowanej charakterystyce hydrofobowej w pełni zgodnej z oczekiwaniami teoretycznymi modelu FOD byłoby całkowicie stabilne oraz doskonale rozpuszczane w wodzie. Jednak aby mogło pełnić swoją funkcję muszą istnieć pewne odchylenia od tego wyidealizowanego stanu. Pierwszym ich przykładem jest działalność enzymatyczna i związanie z nią występowanie kieszeni wiązania liganda [138]. W sensie modelu FOD, reprezentuje ją fragment cząsteczki o wysokiej hydrofobowości teoretycznej i niskiej hydrofobowości obserwowanej, co oznacza niedobór hydrofobowości we wnętrzu struktury białka.

Drugim przykładem zastosowania modelu FOD jest przewidywanie struktury kompleksów białkowych. Zgodnie z jego założeniami, białka powinny dążyć do ukrywania nadmiaru hydrofobowości występującego na ich powierzchni, poprzez łączenie tych powierzchni ze sobą [139]. Temat rozprawy doktorskiej dotyczy sprawdzenia poprawności tych założeń za pomocą metod optymalizacyjnych.

Od 2006 roku zostało opublikowane kilkadziesiąt artykułów naukowych oraz rozdziałów w kolekcjach wydawniczych prezentujących wyniki badań nad hydrofobowością w ujęciu modelu FOD [140–144]. Autor rozprawy jest współautorem łącznie ponad 25 z nich oraz twórcą stosowanego w tych badaniach oprogramowania. Jedno z zagadnień, którymi obecnie się zajmuje poza tematem niniejszej rozprawy, dotyczy relacji środowiska wodnego z tworzeniem się amyloidów [145, 146].

1.3.4. Kompleksowanie białek

Oddziaływania hydrofobowe mają kluczowe znaczenie dla kształtowania się struktury trzecio- i czwartorzędowej białek [147]. Niniejsza rozprawa doktorska ma na celu sprawdzenie poprawności założeń modelu FOD dotyczących tych procesów. Choć wykonane badania wykazały ich zgodność z podejściami stosowanymi przez inne ośrodki naukowe oraz wynikami eksperymentalnymi [148, 149], nie zostało stwierdzone, czy opisywany przez model FOD wpływ środowiska wodnego na cząsteczki białek faktycznie przyczynia się do osiągania przez nie struktury natywnej.

Zgodnie z ideą modelu FOD, podczas reakcji zwijania białka, reszty hydrofobowe powinny kierować się ku środkowi geometrycznemu cząsteczki, a hydrofilne ku jej powierzchni, natomiast w trakcie kompleksowania, reszty hydrofobowe należące do różnych cząsteczek powinny zmierzać ku sobie, a hydrofilne – ku środowisku. Obydwie te reakcje mogą być wyrażone jako dążenie układu do przyjęcia konfiguracji, w której obserwowany status hydrofobowości reszt jest jak najbliższy ich statusowi teoretycznemu. W modelu FOD, różnice pomiędzy rozkładami tych statusów są wyrażane przy pomocy entropii Kullbacka-Leiblera [150].

Dokładne modelowanie procesów zwijania i kompleksowania białek nadal pozostaje nierozwiązanym problemem nauki. Ponieważ model FOD opisuje wpływ środowiska wodnego, które ma znaczenie w każdym z tych procesów, wystarczy skupić się na jednym, aby uzyskać potwierdzenie jego przydatności w drugim. Ze względu na rozwojowy charakter tych badań, postanowiono zacząć od kompleksowania. Powodem tej decyzji była możliwość pracy z układami o mniejszej liczbie stopni swobody i mniej wymagających pod względem obliczeniowym, a przez to pozwalających na ich modelowanie bez potrzeby stosowania wysoce wydajnych komputerów [151].

W celu sprawdzenia założeń modelu FOD dotyczących wpływu opisywanych przez niego oddziaływań hydrofobowych na proces kształtowania się struktury czwartorzędowej białek, Autor rozprawy opracował eksperyment *in silico* polegający na wybraniu z bazy PDB grupy kompleksów, rozdzieleniu ich na tworzące je łańcuchy, a następnie złączeniu z powrotem posługując się kryterium minimalizacji różnicy pomiędzy rozkładami hydrofobowości. Jeżeli założenia modelu FOD są prawidłowe, powinno to doprowadzić do uzyskania kompleksów zbliżonych do natywnych.

Istnieją jeszcze dwa inne, nie poruszane tutaj zagadnienia związane z tematem kompleksowania białek. Pierwsze dotyczy oceny tego, czy dane łańcuchy mogą utworzyć kompleks, a drugie – wskazywania w nich potencjalnych miejsc tego kompleksowania. Model FOD również nadaje się do realizacji obydwu tych zadań [152].

Kompleksy białkowe są utrzymywane przez oddziaływania niekowalencyjne pomiędzy ich łańcuchami oraz środowisko wodne wokół nich [153]. Jeżeli dwa niezwiązane kowalencyjnie ze sobą atomy są położone dostatecznie blisko siebie, uznaje się, że znajdują się w kontakcie niewiążącym. Właściwość ta przekłada się również na ich macierzyste reszty. W zależności od tego, czy atomy należą do tego samego albo różnych łańcuchów, wyróżnia się kontakty wewnątrzcząsteczkowe i międzycząsteczkowe. W tym eksperymencie znaczenie mają tylko te drugie.

Zbiór reszt należących do danego łańcucha, które są zaangażowane w kontakty niewiążące z innymi łańcuchami nazywa się jego interfejsem [154], natomiast struktura przechowująca informację na temat wszystkich interfejsów w kompleksie jest nazywana mapą [155]. Stanowi ona podstawę do stwierdzenia faktu utworzenia tego kompleksu oraz umożliwia określenie jego symetrii. Mapy kontaktów niewiążących mogą być również wyznaczane dla oddziaływań białek z innymi cząsteczkami. Stąd, oprócz kontaktów typu białko-białko (P-P) wyróżnia się kontakty typu P-N (białko-DNA/RNA), P-L (białko-ligand) i P-I (białko-jon), i tak dalej.

Do wyznaczania kontaktów niewiążących wybrano kryterium używane przez serwis PDBsum[17] [156, 157]. Zgodnie z nim, dwie reszty znajdują się w kontakcie niewiążącym, jeżeli przynajmniej jedna para atomów ciężkich (innych niż wodór) do nich należących jest położona w odległości nie przekraczającej 3,9 Å [158].

Istnieje wiele baz materiałów dla eksperymentów przewidywania struktury czwartorzędowej białek [159–161]. Bazy te są wykorzystywane między innymi do sprawdzania i porównywania możliwości algorytmów biorących udział w inicjatywie CAPRI. Z tego powodu, kompleksy przechowywane w tych bazach są dobierane tak, aby były możliwie trudne do przewidzenia. Do potwierdzenia założeń modelu FOD potrzebna jest jednak grupa mniej nietypowych białek, pozwalająca na zmniejszenie liczby czynników wpływających na uzyskane wyniki, a przez to ułatwiająca wyciągnięcie wniosków na ich podstawie.

Zdecydowano się na pracę z białkami homodimerycznymi – najprostszymi i najbardziej rozpowszechnionymi w przyrodzie rodzajami kompleksów [162]. Ich prostota wynika stąd, że tworzy je łańcuch oraz jego kopia, identyczna pod względem sekwencji oraz niemal identyczna pod względem struktury trzeciorzędowej. Białka te charakteryzują się większymi i bardziej hydrofobowymi interfejsami [163], przez co stanowią dobry materiał do sprawdzenie założeń modelu FOD. Należy jednak podkreślić, że prostota kompleksów homodimerycznych, ze względu na różnorodność ich sekwencji i kształtów, nie przekłada się na łatwość ich przewidywania [164].

[17] http://www.ebi.ac.uk/pdbsum

Wyróżnia się dwa podejścia do problemu komputerowego przewidywania struktury białek: *ab inito*, zwane również modelowaniem *de novo*, oraz porównawcze (homologiczne) [165, 166]. Metody należące do pierwszej z tych kategorii, opierają się na modelach opracowanych na podstawie teorii oraz hipotez wynikających z obserwacji właściwości poznanych dotychczas cząsteczek. Zakładają one, że struktura natywna białka (trzecio- lub czwartorzędowa) odpowiada minimum globalnemu stosowanych przez nie kryteriów, a więc dotarcie do niej powinno być możliwe dzięki optymalizacji ich wartości [167]. Odkrycie czysto fizycznego modelu *ab initio* otworzyłoby drogę do udzielenia odpowiedzi na najważniejsze pytania dotyczące procesów zwijania i kompleksowania białek, a także pełnionych przez nie funkcji [168].

Ze względu na rozmiar przestrzeni konfiguracyjnej i występowanie w niej podobnych, a przez to trudnych do odróżnienia minimów lokalnych energii oddziaływań pomiędzy atomami, stosowane obecnie podejścia *ab initio* do tematyki przewidywania struktury białek są zazwyczaj mało precyzyjne [169]. Alternatywne metody porównawcze starają się obejść ten problem poprzez zmniejszenie liczby rozpatrywanych konfiguracji. Zamiast wpisywać badaną cząsteczkę bezpośrednio w zakładany model, poszukują znanej struktury, która jest do niej podobna pod względem ewolucyjnym. Własność tę wyraża się za pomocą identyczności ich sekwencji. Modelowanie danego procesu biologicznego polega wówczas na naśladowaniu motywów występujących w odnalezionym białku-wzorcu, bazując na założeniu, że łańcuchy o podobnych sekwencjach zwijają się w podobny sposób [170].

Dzięki pomocy zbiorów uczących, metody porównawcze osiągają lepsze wyniki od technik *ab initio* [171, 172]. Z drugiej strony, są one w pełni uzależnione od dostępności gotowych struktur, których liczba jest zdecydowanie mniejsza od liczby znanych sekwencji. Największe znaczenie ma jednak to, że będąc działaniami naśladowczymi, a nie twórczymi, nie mogą pomóc w zrozumieniu fizycznych podstaw symulowanych przez nie reakcji. W związku z tym, sprawdzenie założeń modelu FOD wymaga przeprowadzenia eksperymentu *ab initio*.

Procedura przewidywania struktury czwartorzędowej białka składa się z trzech kroków [173]. W pierwszej kolejności poszukiwana jest konformacja kompleksu zbliżona do natywnej. W tym czasie receptor i ligand są traktowane jako bryły sztywne [174]. Następnie, uzyskanie wyniki są oceniane, co pozwala na wybranie najbardziej obiecujących spośród nich [175]. W ostatnim kroku, konformacje podlegają dopracowaniu poprzez ich lokalną optymalizację za pomocą pól siłowych, uwzględniającą elastyczność cząsteczek, przynajmniej w obrębie interfejsu [176]. Podczas opracowanego eksperymentu ograniczono się do pierwszego z tych kroków.

Istnieją dwa sposoby przygotowania danych wejściowych do eksperymentu przewidywania struktury czwartorzędowej białek. W pierwszym, łańcuchy, których kompleks jest poszukiwany są uzyskiwane bezpośrednio z tego kompleksu, a w drugim – sprzed jego utworzenia. Traktowanie ich jako bryły sztywne jest możliwe w obydwu przypadkach, przy założeniu, że ich kształt przed jak i po utworzeniu kompleksu był podobny. W przeciwnym razie, na przykład podczas wymiany domen [177, 178], symulacja nadal będzie mogła być przeprowadzona, ale reprezentowany przez nią model będzie w mniejszym stopniu odpowiadać reakcjom zachodzącym *in vivo*.

Podczas próbkowania przestrzeni konfiguracyjnej kompleksu pary receptor-ligand zmienia się położenie tego drugiego – pierwszy pozostaje w bezruchu, co przekłada się na 6 stopni swobody ich układu. Metody realizujące to zadanie mogą być podzielone na dwie kategorie [179]: wyszukiwania wyczerpującego oraz bezpośredniego. W pierwszej, sprawdzane są wszystkie możliwe interfejsy w określonej rozdzielczości [180]. W drugiej natomiast, położenie liganda polega optymalizacji według przyjętego kryterium. Zaletą tego podejścia jest sprawdzanie mniejszej liczby rozwiązań, a przez to możliwość modelowania układów zawierających większą liczbę cząsteczek niż dwie. Wadą jest natomiast to, że algorytmy optymalizacyjne mogą utknąć w minimach lokalnych, oraz zwracać różne wyniki w różnych próbach dla tego samego białka. Dzieje się tak dlatego, że do sprawnego przeszukiwania przestrzeni konfiguracyjnej nadają się tylko metody stochastyczne, a więc niedeterministyczne.

Adaptacja wniosków z „paradoksu" Levinthala do reakcji powstawania kompleksów białkowych sugeruje, że metody oparte na optymalizacji są bliższe rzeczywistości od metod sprawdzających możliwe konfiguracje. W związku z tym, postanowiono w opracowanym eksperymencie poszukiwać minimum globalnego kryterium modelu FOD przy pomocy algorytmu optymalizacji rojem cząstek (PSO). Algorytm ten był już stosowany w bioinformatyce, ale głównie do dokowania ligandów [181–183], dlatego uznano, że pozwoli to również sprawdzić i potwierdzić jego przydatność w przeszukiwaniu większej przestrzeni konfiguracyjnej.

Dotychczas stosowane podejścia do oceny konformacji kompleksów opierają się przede wszystkim na kryteriach minimalizacji energii oddziaływań pomiędzy atomami [184] lub komplementarności powierzchni (w tym również hydrofobowych [185]). Przykładami metod dokonujących odpowiednio wyszukiwania wyczerpującego i optymalizacji są ZDOCK[18] [186, 187] i AutoDock[19] [188, 189].

[18] http://zdock.umassmed.edu
[19] http://autodock.scripps.edu

Model FOD skupia się na efektach entropowych w białkach, pomijając składowe entalpii, przez co może wskazywać niekorzystne energetycznie konformacje kompleksów. Dlatego, oprócz sprawdzenia poprawności jego założeń przy pomocy opracowanego eksperymentu, postanowiono porównać uzyskane wyniki z wynikami uzyskanymi podczas tego samego eksperymentu, ale wykonanego poprzez optymalizację kryterium jednego z pól siłowych reprezentujących obecnie stosowane podejścia oparte na minimalizacji energii oddziaływań. Wybór padł na pole ECEPP/3.

Decyzja o wyborze pola ECEPP/3 została podjęta z powodu stosowania przez nie stałych długości wiązań i kątów płaskich ($E_b = E_a = \text{const}$). Uznano, że traktowanie w opracowanym eksperymencie łańcuchów białek jako bryły sztywne, a więc gdy dodatkowo zachodzi $E_t = \text{const}$, będzie powodowało mniejsze zmiany w krajobrazie energetycznym ich kompleksu niż w przypadku innych pól, nastawionych na elastyczną reprezentację struktury cząsteczek. Pomimo tego, że pole ECEPP nie jest obecnie rozwijane, nie traci na znaczeniu. Badania przeprowadzone dla jego drugiej wersji wykazały, że jest ono tak dokładne jak inne pola [190, 191]. Używa się go również do optymalizacji konformacji cząsteczek uzyskanych przy pomocy pola UNRES.

Do osiągnięcia stanu równowagi odpowiadającego natywnej strukturze trzeciolub czwartorzędowej białka potrzebne jest osiągnięcie minimum energii swobodnej Gibbsa. W równaniu 1.1 występują dwie składowe: entalpii i entropii, reprezentowane w opracowanym eksperymencie odpowiednio przez pole wewnętrzne (pole ECEPP/3) oraz zewnętrzne (model FOD). Można jednak oczekiwać, że korzystanie z modeli bazujących na jednej z tych składowych okaże się niewystarczające do uzyskania kompleksu białka o oczekiwanej konformacji. Najprostsze podejście, polegające na połączeniu tych pól we wspólnym równaniu, podobnie do równania 1.2 jest niemożliwe ze względu na ich niekompatybilność wynikającą z faktu, że jedno działa w przestrzeni rzeczywistych atomów i posługuje się energią wyrażaną w $\frac{\text{kcal}}{\text{mol}}$, natomiast drugie korzysta z entropii Kullbacka-Leiblera do prezentacji różnic pomiędzy rozkładami hydrofobowości obliczanymi dla reszt. Ponieważ obydwa pola są uznawane za równoważne pod względem istotności, aby móc sprawdzić efekty ich równoczesnego wpływu na proces tworzenia się kompleksów białkowych, postanowiono wykonać ich optymalizację wielokryterialną.

Do wykonania optymalizacji wielokryterialnej użyto opracowanego przez Autora rozprawy, nowego algorytmu MOSF (wielokryterialne rodziny rojów, multi objective swarm families), opartego na zasadzie działania roju cząstek, którego główną zaletą jest zdolność do łączenia wynikowych konformacji kompleksu w grupy.

1.3.5. Optymalizacja

Zagadnienie problemu optymalizacyjnego dotyczy poszukiwania optymalnego elementu zbioru $\Omega \subseteq \mathcal{X}$ zgodnie ze wskazaniami funkcji $f : \mathcal{X} \to \mathcal{Y}$ [192]. Jej dziedzina, będąca podzbiorem przestrzeni rozwiązań $\mathcal{X}$, może być dowolnym, niepustym zbiorem liczb, wektorów, ciągów znaków, innych zbiorów, i tym podobnych obiektów, nazywanych ogólnie punktami. Analogicznie, zbiór $\mathcal{Y}$ określa się mianem przestrzeni wartości, zazwyczaj utożsamianej ze zbiorem liczb rzeczywistych.

Funkcja f jest nazywana kryterium optymalizacyjnym. Przyjmuje się, że poszukiwane rozwiązanie optymalne, $x^* \in \Omega$, stanowi taki jej argument, w którym – zależnie od sformułowania problemu – przyjmuje ona najniższą lub najwyższą wartość. Mówiąc inaczej, x^* jest jej minimum lub maksimum globalnym w Ω. Jeżeli w tym zbiorze znajduje się więcej punktów o identycznych właściwościach, zwrócony może być dowolny z nich. Pozostałe rozwiązania są nazywane suboptymalnymi. Tak zdefiniowana procedura optymalizacji jest nazywana odpowiednio minimalizacją lub maksymalizacją. Warto jednak zauważyć, że maksymalizacja kryterium f jest tożsama z minimalizacją $-f$. W związku z tym, w całej niniejszej rozprawie, pod pojęciem optymalizacji będzie rozumiana druga z tych czynności. Nie prowadzi to bowiem do utraty ogólnego brzmienia tej definicji.

Jeżeli punkty z przestrzeni rozwiązań są opisywane przez d mierzalnych cech (właściwości, zmiennych), w zależności ich typu, kryterium f będzie mieć inną postać, w szczególności jedną z następujących: $f : \mathbb{R}^d \to \mathbb{R}$, $f : \mathbb{Z}^d \to \mathbb{R}$ lub $f : \{0, 1\}^d \to \mathbb{R}$. Przykładem pierwszej z tych sytuacji jest optymalizacja orientacji bryły sztywnej (współrzędnych środka geometrycznego i kątów obrotu), drugiej – problem komiwojażera (poszukiwanie minimalnego cyklu Hamiltona w pełnym grafie ważonym), natomiast trzeciej – dyskretny problem plecakowy (maksymalizacja wartości „przedmiotów", których łączna „masa" nie może przekroczyć określonego progu) [193].

Rodzaje optymalizacji

Yu i Gen [194] wymieniają kilka kategorii, do których mogą zostać zakwalifikowane różne problemy optymalizacyjne. Choć wszystkie z nich są zgodne z powyższą definicją i mogą być sprowadzone do minimalizacji jakiegoś kryterium (w teorii, ale niekoniecznie w praktycznych zastosowaniach), ze względu na rodzaj oczekiwanej informacji zwrotnej oraz warunki w jakich optymalizacja jest przeprowadzana, sens ma posługiwanie się pewną taksonomią. Pozwala to również na podobny podział algorytmów służących do rozwiązywania tych problemów.

Poniżej znajduje się krótka charakterystyka następujących zagadnień:

- optymalizacji globalnej,
- optymalizacji wielomodalnej,
- optymalizacji kombinatorycznej,
- optymalizacji dynamicznej,
- optymalizacji z ograniczeniami,
- optymalizacji wielokryterialnej.

Poszukiwanie rozwiązania optymalnego wyłącznie na podstawie wartości kryterium f (bez dodatkowych wymagań, takich jak relacje pomiędzy zmiennymi) jest nazywane optymalizacją globalną. W przypadku braku informacji na temat typu cech je opisujących, można założyć, że są one rzeczywiste. Algorytmy rozwiązujące tego rodzaju zadania muszą więc odnaleźć, czyli osiągnąć zbieżność w którymś z minimów globalnych i nie utknąć w którymś z minimów lokalnych. Druga z tych sytuacji jest nazywana przedwczesną zbieżnością.

Funkcja wielomodalna posiada oprócz minimum globalnego kilka innych minimów, globalnych lub lokalnych. Klasyfikacja problemu jako wielomodalnego nie jest jednak związana z tym faktem, ale oczekiwaną postacią wyniku jego optymalizacji. Zamiast pojedynczego minimum globalnego, poszukiwane rozwiązanie stanowią bowiem wszystkie z nich, lub ich określona liczba. Alternatywnym, choć zaliczanym do tej samej kategorii zadaniem jest odnalezienie najniższych minimów lokalnych. Zgodnie ze swoją definicją, należą do nich również minima globalne. Algorytmy zajmujące się tą tematyką muszą być więc zdolne do równoczesnego zbiegania się w kilku rozwiązaniach oraz unikania zwracania duplikatów.

Optymalizacja kombinatoryczna pochodzi z dziedziny matematyki dyskretnej. Kryteria w niej występujące mają często postać $f : \mathbb{Z}^d \to \mathbb{R}$ lub $f : \{0,1\}^d \to \mathbb{R}$, a pomiędzy zmiennymi opisującymi rozwiązania występują relacje, ograniczające liczbę elementów w zbiorze Ω do skończonej wartości. Na przykład, punkt x może być kombinacją (nieuporządkowaną kolekcją) lub permutacją (sekwencją) o długości d indeksów elementów pewnego przeliczalnego zbioru. Oznacza to, że sposób poszukiwania rozwiązań optymalnych jest tutaj zależny od problemu. Dodatkowa trudność w poruszaniu się algorytmów po krajobrazie wartości kryteriów optymalizacji kombinatorycznej wynika z braku natywnej definicji sąsiedztwa, rozumianego jako zbiór wszystkich rozwiązań w przedziale $[x - \epsilon, x + \epsilon]$ dla danego $e > 0$ [194].

Optymalizacja dynamiczna wyróżnia się na tle pozostałych tym, że jej kryteria są niestacjonarne, czyli, że kształty krajobrazów ich wartości, a przez to minima mogą ulegać zmianie, w szczególności przemieszczać się, pojawiać i zanikać [195]. Reprezentują one sytuacje charakteryzujące się niepewnością, wynikającą z zależności od zmiennych, na które nie ma się wpływu. Przykładem takiej sytuacji jest optymalizacja orientacji ogniwa słonecznego w ciągu dnia. Wraz ze zmianą położenia miejsca, w którym urządzenie znajduje się na powierzchni Ziemi względem Słońca, zachodzi potrzeba ciągłego korygowania jego zwrotu i nachylenia w celu uzyskania maksymalnej sprawności. Algorytmy optymalizacyjne muszą więc mieć możliwość nie tylko poszukiwania minimów, ale również ich śledzenia, czyli zachowywania równowagi pomiędzy zbieżnością a zdolnością do przeszukiwania przestrzeni rozwiązań. Z tego powodu kolejne krajobrazy wartości kryteriów muszą być do siebie podobne. W przeciwnym przypadku, niezbędne może być rozpoczęcie całej procedury od początku [196]. Choć są to podobne tematy, do zagadnień optymalizacji dynamicznej nie zalicza się kryteriów, do których wartości dodawany jest szum, gdyż są one zazwyczaj stacjonarne, tak więc właściwości ich minimów nie ulegają zmianom [197].

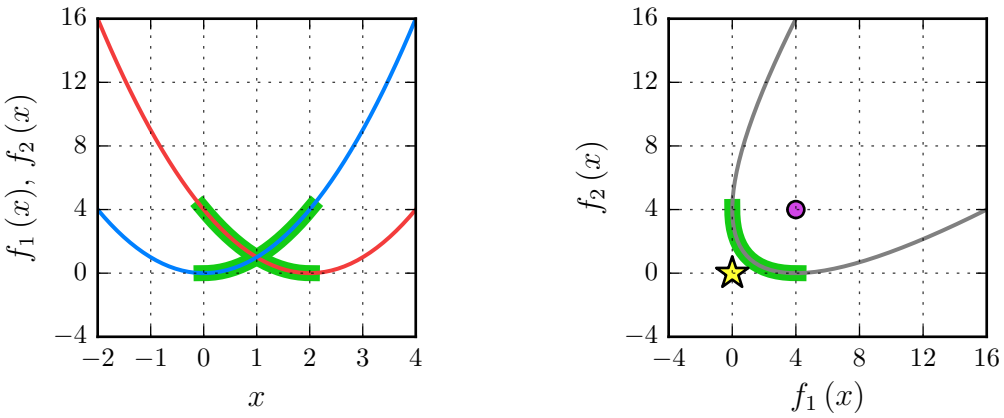
W optymalizacji z ograniczeniami, oprócz kryterium f występują dwa zbiory funkcji ograniczeń: nierówności $\mathcal{G} \equiv \{g_1 \dots g_k\}$ oraz równości $\mathcal{H} \equiv \{h_{k+1} \dots h_{k+n}\}$ [198, 199]. Powodują one podział zbioru Ω na dwa rozłączne podzbiory: rozwiązań dopuszczalnych $\mathcal{F}$ i niedopuszczalnych $\Omega \setminus \mathcal{F}$. Aby punkt x mógł być uznany za element pierwszego z nich, musi spełniać następujące warunki: $\forall i \in \{1, \dots, k\}\, g_i \leq 0$ oraz $\forall j \in \{k+1, \dots, k+n\}\, h_j = 0$. W przeciwnym przypadku trafia do drugiego podzbioru. Wartości zwracane przez funkcje ograniczeń pozwalają stwierdzić nie tylko to, które rozwiązania są niedopuszczalne, ale również jak bardzo odbiegają od oczekiwań. Poprzez ich minimalizację algorytmy optymalizacyjne mogą oszacowywać w którym kierunku powinny się udać, aby dotrzeć do właściwej części przestrzeni rozwiązań. Problem, na jaki przy tym natrafiają wynika z potencjalnie skomplikowanej relacji pomiędzy zbiorami $\mathcal{F}$ i Ω. W szczególności, pierwszy z nich nie musi być spójny, wypukły, lub dawać się opisać w jakikolwiek użyteczny dla optymalizacji sposób. Istnieje również możliwość, że algorytm, który rozpoczął swoje działanie w jednym z jego podzbiorów nie będzie w stanie „przeskoczyć” ponad rozwiązaniami niedopuszczalnymi do innego podzbioru zawierającego poszukiwane minimum globalne. Ponieważ funkcje ograniczeń równości są trudniejsze do usatysfakcjonowania i powodują znaczne zmniejszenie zbioru $\mathcal{F}$, utrudniające odnalezienie jakiegokolwiek jego elementu, często zastępuje się je mniej restrykcyjnymi funkcjami ograniczeń nierówności postaci $|h_j| - \delta \leq 0$, gdzie $j \in \{k+1, \dots, k+n\}$ oraz $\delta > 0$.

Ostatnim z wymienionych tu rodzajów optymalizacji, choć najważniejszym z punktu widzenia tematu niniejszej rozprawy, jest optymalizacja wielokryterialna. Jej wyróżniającą cechę stanowi obecność więcej niż jednego kryterium. Powstaje w ten sposób funkcja F zwracająca wartości wszystkich z nich dla danego rozwiązania [200]:

$$F(x) = \left[f_1(x), \ldots, f_k(x) \right] \tag{1.3}$$

Zadaniem algorytmów działających w tej dziedzinie jest jednoczesna optymalizacja kryteriów $f_1, \ldots, f_k$. Jako przykład takiej sytuacji można podać maksymalizację efektywności działania pewnego urządzenia przy minimalizacji kosztów jego eksploatacji. Zadanie to natrafia na problem wynikający z faktu, że kryteria $f_1, \ldots, f_k$ nie muszą być w jakikolwiek sposób ze sobą zgodne. Z tego powodu, mogą posiadać różne liczby inaczej rozmieszczonych minimów. Optymalizacja jednego z nich może (i prawdopodobnie będzie) powodować wzrost wartości pozostałych. Oznacza to, że nie da się na tej podstawie udzielić odpowiedzi na pytanie, które rozwiązania należy uznać za minima globalne funkcji F. Do zademonstrowania tego dylematu wystarczy posłużyć się następującymi dwoma kryteriami: $f_1(x) = x^2$ i $f_2(x) = (x-2)^2$. Wartość funkcji F w minimum globalnym pierwszego z nich ($x_1^* = 0$) wynosi $[0, 2]$, a drugiego ($x_2^* = 2$) – $[2, 0]$. Sytuacja ta jest widoczna na rysunku 1.3. W miarę przesuwania się po osi liczbowej argumentów od x_1^* do x_2^*, rośnie wartość kryterium f_1, a f_2 – maleje. Poza tym przedziałem, obydwa kryteria dążą do $+\infty$.

Pierwszym sposobem obsługi „niekompatybilności" kryteriów $f_1, \ldots, f_k$ jest użycie wag, przez które mnożone są zwracane przez nie ich wartości, efektywnie redukujące optymalizację wielokryterialną do globalnej. Wymaga to jednak wiedzy na temat konkretnego problemu. Optymalizowane kryteria mogą również nie dawać się w jakikolwiek sposób ze sobą porównać. Inne podejście wynika z obserwacji, że część rozwiązań jest „lepsza" od pozostałych. W powyższym przykładzie, punkty $[0, 2]$ i $[2, 0]$ znajdują się niżej na osiach wartości odpowiadających im kryteriów niż punkt $[4, 16]$. Nie można wskazać w tej trójce rozwiązania globalnie optymalnego w sensie funkcji F, ale można stwierdzić, które jest najsłabszym kandydatem do tego tytułu. Oznacza to, że zamiast szukać pojedynczego, zapewne nieistniejącego punktu „idealnego", warto skupić się na „najlepszym" podzbiorze Ω, stanowiącym dolne ograniczenie jego odpowiednika w przestrzeni wartości. Podzbiór ten jest nazywany optymalnym zbiorem Pareto F w Ω i oznaczany jako $\mathcal{PS}^*$. Jego nazwa pochodzi od nazwiska włoskiego ekonomisty Vilfredo Pareto (1848 – 1923) [201]. Wartości funkcji F elementów tego zbioru tworzą natomiast front Pareto – $\mathcal{PF}^*$.



(a) Podzbiór przestrzeni rozwiązań. (b) Podzbiór przestrzeni wartości.

Rysunek 1.3: Optymalny zbiór i front Pareto przykładowej funkcji wielokryterialnej $F : \mathbb{R} \to \mathbb{R}^2$, gdzie $f_1(x) = x^2$ i $f_2(x) = (x-2)^2$. Pierwsze z tych kryteriów ma na rysunku a kolor niebieski, a drugie – czerwony. Pary ich wartości dla kolejnych punktów z przedziału $[-2, 4]$ tworzą szarą krzywą na rysunku b. Położenia optymalnego zbioru i frontu Pareto są zaznaczone na zielono. Żółta gwiazda w punkcie $[0, 0]$ wskazuje punkt „idealny" – hipotetyczne minimum globalne funkcji F, nienależące do jej przestrzeni wartości, natomiast fioletowe koło na *nadir* – jego przeciwieństwo, $[4, 4]$.

Front Pareto zawdzięcza swoją nazwę temu, że stanowi zbiór wartości rozwiązań, które są ustawione „przodem" do hipotetycznego, wspólnego minimum globalnego kryteriów $f_1, \ldots, f_k$, „zasłaniając" przed nim pozostałe punkty. Dla $k = 2$, może przyjmować kształt niekoniecznie spójnej łamanej (wypukłej lub wklęsłej), tak jak to widać na rysunku 1.3, dla $k = 3$ – powierzchni, i tak dalej. Dalsze decyzje dotyczące tego, co należy zrobić z jego elementami wychodzą poza kompetencje algorytmów optymalizacji wielokryterialnej – podejmuje je użytkownik.

Przynależność rozwiązań do zbioru Pareto jest ściśle związana z pojęciami dominacji i niedominowania. Definicja pierwszego z nich jest następująca [202]:

Definicja 1.1. Dominacja w sensie Pareto

Niech x i y będą wektorami z przestrzeni $\mathbb{R}^d$. Mówi się, że x dominuje y, co oznaczamy $x \prec y$, jeżeli dla każdego $i \in \{1, \ldots, d\}$ wartość i-tej składowej x jest nie większa od wartości i-tej składowej y oraz istnieje takie $j \in \{1, \ldots, d\}$, dla którego wartość j-tej składowej x jest niższa od j-tej składowej y:

$$x \prec y \Leftrightarrow \forall\, i \in \{1, \ldots, d\} : x_i \leq y_i \wedge \exists\, j \in \{1, \ldots, d\} : x_j < y_j \tag{1.4}$$

Optymalny zbiór Pareto tworzą wszystkie rozwiązania niezdominowane w Ω, czyli te, dla których nie istnieją punkty dominujące je w przestrzeni wartości funkcji F:

Definicja 1.2. Optymalny zbiór Pareto
Niech $F(x) = \left[f_1(x), \ldots, f_k(x)\right] : \Omega \subseteq \mathbb{R}^d \to \mathbb{R}^k$. Mówi się, że $\mathcal{PS}^*$ jest optymalnym zbiorem Pareto funkcji F w zbiorze Ω, jeżeli zawiera wszystkie rozwiązania z tego zbioru, które są w nim niezdominowane:

$$\mathcal{PS}^* \equiv \left\{ x \,\middle|\, x \in \Omega \wedge \nexists y \in \Omega : F(y) \prec F(x) \right\} \tag{1.5}$$

Optymalny front Pareto jest zbiorem wartości funkcji F zbioru Pareto:

Definicja 1.3. Optymalny front Pareto
Niech $F(x) = \left[f_1(x), \ldots, f_k(x)\right] : \Omega \subseteq \mathbb{R}^d \to \mathbb{R}^k$. Mówi się, że $\mathcal{PF}^*$ jest optymalnym frontem Pareto funkcji F w zbiorze Ω, jeżeli zawiera jej wartości dla wszystkich rozwiązań z tego zbioru, które są w nim niezdominowane:

$$\mathcal{PF}^* \equiv \left\{ F(x) \,\middle|\, x \in \Omega \wedge \nexists y \in \Omega : F(y) \prec F(x) \right\} \tag{1.6}$$

Zakładając, że każde z kryteriów $f_1, \ldots, f_k$ posiada minimum globalne w Ω, zbiór Pareto może zawierać skończoną lub nieskończoną liczbę rozwiązań niezdominowanych (w szczególności – jedno). Oznacza to, że istnieją wyjątkowe sytuacje, w których poszukiwany wynik jest tożsamy z wynikiem optymalizacji globalnej lub wielomodalnej, aczkolwiek należą one do rzadkości. Pojawia się więc tu problem dotyczący sposobu przedstawienia zawartości nieskończonego zbioru, który na dodatek nie musi być spójny, zarówno w przestrzeni rozwiązań, jak i wartości.

Algorytmy optymalizacji wielokryterialnej działają iteracyjne, starając się w kolejnych krokach zwracać coraz lepsze przybliżenie optymalnego zbioru Pareto. Odnalezione przez nie rozwiązania niezdominowane są przechowywane w strukturze archiwum. Jej maksymalny rozmiar jest zazwyczaj stały, natomiast zawartością zarządza część algorytmu zwana archiwizatorem, któremu kandydatów na rozwiązania niezdominowane dostarcza inna część algorytmu określana mianem generatora. Przyjmuje się, że wspólnie powinny one realizować następujące trzy zadania [203]:

1. maksymalizować liczbę elementów frontu Pareto,
2. minimalizować odległość pomiędzy wynikowym a rzeczywistym frontem Pareto,
3. minimalizować odchylenie standardowe rozkładu elementów frontu Pareto.

Pierwsze zadanie polega na dążeniu do zapełnienia archiwum samymi rozwiązaniami niezdominowanymi. Nie muszą one jednak być optymalne w Ω. Zapewnienie tej właściwości jest bowiem celem zadania drugiego. Zgodnie z nim, algorytmy optymalizacyjne powinny starać się, aby rozwiązania przez nie zwracane znajdowały się najbliżej w przestrzeni wartości elementów optymalnego frontu Pareto. Ostatnie zadanie dotyczy natomiast ich rozkładu – oczekuje się, że powinien być on zbliżony do jednorodnego. Ma to na celu zapewnienie dobrej reprezentacji całego frontu, czyli unikanie skupisk punktów niedokładnie reprezentujących jego kształt.

Jak widać, powyższe zadania skupiają się przede wszystkim na froncie Pareto. Autor rozprawy dodaje do nich jeszcze jedno – dobrą reprezentację w przestrzeni rozwiązań, rozumianą jako takie rozmieszczenie w niej punktów stanowiących wynik działania algorytmu, które minimalizuje ich odległość od elementów rzeczywistego zbioru Pareto, co ważne – liczoną z perspektywy tego drugiego. Celem optymalizacji jest bowiem poszukiwanie argumentów badanego kryterium, które posiadają oczekiwane właściwości. Tutaj mają być one niezdominowane w Ω. Przykładem sytuacji, w której zadanie to jest potrzebne do przybliżenia wszystkich z nich jest optymalny zbiór Pareto składający się z dwóch rozłącznych podzbiorów, dla których elementów funkcja F przyjmuje identyczne wartości. Może się tak zdarzyć między innymi wtedy, gdy kryteria $f_1, \ldots, f_k$ są okresowe. Do zrealizowania zadań z poprzedniego akapitu wystarczy więc przybliżenie zwartości tylko jednego z tych podzbiorów. Dzięki dodatkowemu celowi zwrócone mogą być jednak obydwa, pomimo występującej redundancji. Więcej szczegółów na ten temat znajduje się w rozdziale 3.1.

Optymalizacja wielokryterialna ma istotne znaczenie nie tylko ze względu rodzaj uzyskiwanych w jej trakcie wyników, ale również dlatego, że może być pomocna podczas rozwiązywania innego rodzaju problemów, na przykład [204]:

- Jeżeli kryterium f jest tylko jedno, dzięki zastosowaniu archiwum, optymalizacja wielokryterialna sprowadza się do optymalizacji wielomodalnej lub globalnej.
- Niektóre problemy globalne, takie jak poszukiwanie najkrótszej ścieżki w grafie lub minimalnego drzewa rozpinającego mogą być dzielone na pod-problemy i efektywnie rozwiązywanie w sposób wielokryterialny [205].
- Funkcje ograniczeń mogą być traktowane jako osobne kryteria [206].
- Stosowanie dodatkowego kryterium utrzymującego algorytm wielokryterialny w fazie przeszukiwania przestrzeni rozwiązań, pozwala mu na reagowanie na dynamiczne zmiany w środowisku [207].

1.3.6. Algorytmy optymalizacyjne

Podobnie jak rodzaje optymalizacji, istnieje również taksonomia algorytmów działających w tej dziedzinie. Różnorodność ta wynika z faktu, że poszukiwanie minimów globalnych jest zadaniem trudnym [208]. Nie istnieje uniwersalny algorytm, zdolny do efektywnego rozwiązywania wszystkich możliwych problemów (wyjaśnienie tego stwierdzenia znajduje się na końcu rozdziału). Potrzebne są w związku z tym różne narzędzia, stosowane w zależności od rodzaju zadania do wykonania, oczekiwanej formy wyniku, dokładności, a także czasu jego realizacji.

Pojęcie trudności można rozumieć w trojaki sposób: poprzez złożoność obliczeniową i zapotrzebowanie na pamięć w sensie asymptotycznego tempa wzrostu, przynależność do klasy problemów NP-zupełnych, a także ilość przydatnej informacji dostarczanej przez optymalizowane kryteria. Pierwsze z tych zagadnień jest związane z rozmiarem zbioru danych wejściowych i niezbędnymi czynnościami do wykonania (na przykład mnożeniem macierzy), drugie – z naturą samego zadania, określającą, czy da się je rozwiązać dokładnie w rozsądnym czasie, czy potrzeba zwrócić się ku podejściom heurystycznym, natomiast ostatnie dotyczy pułapek, jakie czekają na algorytmy podczas pracy. Weise i współpracownicy [209] wymieniają wśród nich między innymi: wielokrotnie minima lokalne (przedwczesna zbieżność), brak użytecznego gradientu (trudność w określeniu kierunku działania), niewielkie baseny atrakcji minimów globalnych („igła w stogu siana” / „fałszywy trop”), płaskowyże na krajobrazie wartości (redukcja do losowego błądzenia), a także obecność szumu i nadmierne dopasowanie w przypadku stosowania zbiorów uczących.

Algorytmy optymalizacyjne mogą być przydzielone do dwóch głównych rodzin: deterministycznych oraz stochastycznych [210]. W skład pierwszej z nich wchodzi podkategoria metod numerycznych oraz wyszukiwanie wyczerpujące.

Algorytm wyszukiwania wyczerpującego, zwany również metodą siłową, polega na sprawdzaniu wszystkich możliwych, lub odpowiednio dużej próbki kandydatów na rozwiązanie danego problemu, licząc, że jego minimum globalne znajdzie się wśród nich, albo w ich bliskim sąsiedztwie. Jeżeli przestrzeń rozwiązań jest rzeczywista ($\mathbb{R}^d$), próbkę tę może tworzyć prostokątna siatka punktów o określonej gęstości w każdym wymiarze (n), rozpościerająca się pomiędzy wybranymi wartościami skrajnymi [211]. Wystarczy wówczas sprawdzić wszystkie z nich i znaleźć ten, w którym kryterium optymalizacyjne osiąga swoje minimum. Podejście to jest bardzo proste i znajduje zastosowanie tam, gdzie inne metody nie mogą być użyte, ale z powodu przekleństwa wymiarowości [212], jego praktyczne wykorzystanie podlega silnym ograniczeniom.

W optymalizacji wielokryterialnej, wyszukiwanie wyczerpujące pozwala na wygenerowanie referencyjnego przybliżenia optymalnego zbioru Pareto, służącego jako podstawa oceny wyników działania innych algorytmów. Odnalezienie rozwiązań niezdominowanych wymaga porównania ich par, co powoduje, że złożoność obliczeniowa tej procedury wynosi $O\left(n^{2d}\right)$. Już dla $n = 100$ i $d = 3$ lub $n = 1000$ i $d = 2$ zadanie to może okazać się bardzo czasochłonne. Do stwierdzenia, że wszystkie punkty tworzące te siatki są niezdominowane niezbędne jest bowiem wykonanie maksymalnie 10^{12} porównań. W zależności od kształtu optymalnego zbioru Pareto, względnie niska rozdzielczość może być jednak wystarczająca do uzyskania jego zadowalającego przybliżenia, na przykład wtedy, gdy tworzą go rozległe skupiska.

Jednym ze sposobów częściowego radzenia sobie z przekleństwem wymiarowości w problemach kombinatorycznych jest algorytm branch and bound [213]. Dzieli on przestrzeń rozwiązań na coraz mniejsze podzbiory i odrzuca te z nich, w których według niego nie będzie rozwiązania optymalnego. Zmniejsza to liczbę kandydatów sprawdzanych podczas wyszukiwania wyczerpującego, ale wymaga wiedzy na temat konkretnego problemu. Podejście to może być również użyte w kombinatorycznej optymalizacji wielokryterialnej [214]. Aktualny przegląd metod z tej dziedziny znajduje się w pracy Przybylskiego i Gandibleux [215].

Numeryczne algorytmy optymalizacyjne służą do lokalnego, iteracyjnego przeszukiwania przestrzeni rozwiązań. Również i one mogą być przydzielone do dwóch podkategorii: metod bezpośrednich, poruszających się w stronę malejących wartości badanego kryterium, oraz pośrednich, które w tym celu korzystają z jego pochodnych. Przedstawicielem pierwszej z nich jest algorytm wspinaczki (hill climbing) [216], traktujący wszystkie rozwiązania jako wierzchołki grafu. Krawędzie pomiędzy nimi wyznacza zależna od problemu funkcja sąsiedztwa. Algorytm wspinaczki rozpoczyna swoje działanie w jednym z punktów, sprawdzając jego najbliższych sąsiadów, czy w którymś optymalizowane kryterium przyjmuje niższą wartość. W wersji pierwszego wyboru, przechodzi do pierwszego, który spełnia ten warunek, natomiast w wersji najszybszego wzrostu analizowane są wszystkie z nich. Optymalizacja kończy się w chwili osiągnięcia najbliższego maksimum lokalnego i konsekwentnej niemożliwości przejścia do innego rozwiązania (przestrzeń dyskretna), lub spadku różnicy pomiędzy kolejnymi wartościami poniżej ustalonego progu $\epsilon > 0$ (przestrzeń rzeczywista).

Jednym z problemów algorytmu wspinaczki jest zmiana pojedynczych składowych wektorów rozwiązań powodująca trudność w poruszaniu się po dolinach ustawionych nierównolegle do osi układu współrzędnych. Problem ten nie występuje w metodach pośrednich, które mogą zmieniać wszystkie z nich na raz.

Powszechnie znanymi pośrednimi, numerycznymi algorytmami optymalizacyjnymi są metoda najszybszego spadku, metoda sprzężonych gradientów, metoda Newtona-Raphsona i metody quasi-Newtonowskie [217, 218]. Algorytmy te posługują się informacją zawartą w gradiencie wartości kryterium optymalizacyjnego do wyznaczania kierunku oraz zasięgu swojego przemieszczenia. Dzięki temu są dokładne, to znaczy, że mogą wskazać poszukiwane minimum z oczekiwaną dokładnością, zależną od liczby ich iteracji. Dlatego stosuje się je między innymi do rozwiązywania niektórych układów równań liniowych i nieliniowych. Ponieważ metody te kierują się w stronę najbliższego minimum, ich zastosowanie ogranicza się do optymalizacji lokalnej. Dodatkowo, przestrzeń wartości po której się poruszają musi być ciągła i gładka oraz muszą być znane równania pierwszej lub ewentualnie drugiej pochodnej funkcji, a także miejsce w którym ma rozpocząć się optymalizacja.

Jeżeli dany problem optymalizacyjny nie może zostać rozwiązany przy użyciu metod numerycznych z powodu braku ciągłości lub różniczkowalności jego kryterium, braku możliwości wskazania otoczenia minimum globalnego, z którego będzie można do niego dotrzeć, lub wtedy, gdy przestrzeń rozwiązań jest zbyt duża, do dyspozycji pozostaje ostatnia rodzina algorytmów – stochastycznych. Przykładem tego rodzaju podejść są metaheurystyki (gr. *szukać*, *odkrywać*), czyli niedeterministyczne strategie zarządzające procesem poszukiwania rozwiązania optymalnego [219]. Nie wymagają one wiedzy na temat właściwości optymalizowanego kryterium, a ich funkcjonowanie nie zależy od konkretnego problemu. Stosowanie wartości losowych pozwala na przeszukiwanie dużych podzbiorów przestrzeni rozwiązań, ale nie gwarantuje, że minimum globalne zostanie w nich odnalezione.

Popularne algorytmy metaheurystyczne czerpią inspirację z rzeczywistych zjawisk. Zaliczają się do nich między innymi: symulowane wyżarzanie [220], algorytmy ewolucyjne [221], optymalizacja rojem cząstek [64] i algorytm mrówkowy [222]. Charakterystyczną cechą części z tych metod jest korzystanie z populacji osobników, których wzajemne oddziaływania prowadzą je w stronę rozwiązań optymalnych.

Twierdzenia no free lunch Wolperta i Macready'ego [223, 224] wskazują, że nie istnieje algorytm „idealny", a nawet taki który rozwiązuje wszystkie problemy optymalizacyjne sprawniej od algorytmu wyszukiwania wyczerpującego. Każdy algorytm sprawdzający się w jednym przypadku musi okazać się słabszy w innym, zakładając, że nie posiada wiedzy na jego temat. Należy więc korzystać z takich algorytmów, o których wiadomo, że mogą być skuteczne do rozwiązywania danego problemu. Dlatego Autor rozprawy zdecydował się na użycie algorytmów opartych na zasadzie działania roju cząstek, sprawdzonego w wielu zastosowaniach [225].

2. Materiały i metody

Rozdział materiały i metody jest podzielony na dwie części: biologiczną i informatyczną. W pierwszej części jest przedstawiona baza danych białek homodimerycznych oraz pola wewnętrzne (ECEPP/3) i zewnętrzne (model FOD) użyte w eksperymencie przewidywania struktury czwartorzędowej białek. W części drugiej znajduje się natomiast opis algorytmu optymalizacji rojem cząstek (PSO) oraz pozostałych algorytmów zastosowanych do realizacji tego zadania.

2.1. Baza danych białek homodimerycznych

Materiałami użytymi do sprawdzenia założeń pola zewnętrznego dotyczących wpływu opisywanych przez to pole oddziaływań hydrofobowych na proces kompleksowania białek było 200 struktur homodimerycznych wybranych z bazy PDB.

Podstawowym kryterium umożliwiającym wyszukanie białek homodimerycznych w bazie PDB jest stechiometria A2 kompleksu. Ze względu na możliwość występowania mutacji, RCSB nie wymaga aby sekwencje łańcuchów były identyczne. Zamiast tego, dany kompleks jest klasyfikowany jako homodimer wtedy, gdy poziom identyczności sekwencji jego łańcuchów wynosi przynajmniej 95%.[1] Sprawdzenie tego warunku odbywa się przy pomocy algorytmów Needlemana-Wunscha [226] i Smitha-Watermana [227] lub programu BLAST2 SEQUENCES [228].

Porównanie sekwencji nie gwarantuje jednak, że – zgodnie z wymaganiami tego eksperymentu – łańcuchy tworzące kompleks uznany za homodimer będą również identyczne pod względem struktury trzeciorzędowej (z dokładnością do lokalnych zmian konformacyjnych). RCSB bierze bowiem pod uwagę tylko sekwencje zapisane w rekordach SEQRES, ignorując rzeczywisty skład aminokwasowy, wynikający z zawartości rekordów ATOM / HETATM obecnych w danym pliku PDB. Może się więc zdarzyć, że informacje na temat struktury jednego łańcucha będą niekompletne.

[1] http://www.rcsb.org/pdb/staticHelp.do?p=help/advancedsearch/stoichiometry.html

Oprócz różnic w strukturze trzeciorzędowej łańcuchów, istnieją także inne czynniki, takie jak wielokrotne domeny, reszty zmodyfikowane, międzycząsteczkowe mostki disiarczkowe, lub niedoskonałości eksperymentu krystalograficznego, które mogą utrudniać ustalenie parametrów symulacji, lub wyciągnięcie wniosków na podstawie uzyskanych wyników. W związku z tym, w celu ograniczenia liczby potencjalnych niewiadomych, postanowiono zastosować filtr złożony z kilku dodatkowych kryteriów. Dzięki niemu, zbiór prawie 30 tysięcy białek z bazy PDB charakteryzujących się stechiometrią A2[2] został zredukowany do 200 reprezentacyjnych kompleksów homodimerycznych o oczekiwanych właściwościach:

1. Łańcuchy tworzące kompleks musiały posiadać identyczny wpis w rekordzie COMPND oraz pochodzić z tego samego organizmu komórkowego lub wirusa (identyczny wpis w rekordzie SOURCE). Pozwoliło to na pominięcie modeli teoretycznych i struktur *de novo*.

2. Kompleks musiał składać się z dokładnie dwóch łańcuchów polipeptydowych. Gwarantowało to, że nie będzie on homodimeryczną podjednostką większego kompleksu. Przykładem takiej sytuacji jest hemoglobina, o stechiometrii A2B2.

3. Pomiędzy łańcuchami musiały występować kontakty niewiążące, obliczone zgodnie z kryterium stosowanym przez serwis PDBsum [156, 157] (pary atomów ciężkich znajdujące się w odległości nie większej niż 3,9 Å [158]). Ich obecność jest podstawą do stwierdzenia utworzenia kompleksu, a porównywanie map kontaktów niewiążących stanowi jedną z metod wymiernej oceny zgodności wyniku eksperymentu kompleksowania ze strukturą natywną.

4. Pomiędzy łańcuchami nie mogły występować kolizje (pary atomów ciężkich znajdujące się w odległości mniejszej niż 1,9 Å). Ponieważ zawierający je układ cząsteczek charakteryzuje się zazwyczaj wysoką wartością energii, ich obecność stanowi poważną przeszkodę podczas poszukiwań jego struktury natywnej w oparciu o kryterium pola wewnętrznego.

5. Model białka musiał być otrzymany na drodze eksperymentu którego wartość parametru rozdzielczości krystalograficznej nie przekraczała 2,9 Å. Na tym poziomie możliwe jest wyznaczenie prawidłowych konformacji łańcucha głównego i łańcuchów bocznych [229], niezbędnych do uzyskania sensownych wyników. Warunek ten nie dotyczył modeli opracowanych za pomocą techniki NMR.

[2] Stan w sierpniu 2014.

6. Plik PDB struktury białka musiał zawierać tylko jeden model (konformację atomów). Pozwoliło to na uniknięcie dylematów związanych z wyborem tej właściwej z dostępnego zestawu. W odróżnieniu od poprzedniego, warunek ten dotyczył przede wszystkim modeli uzyskanych za pomocą techniki NMR.

7. Łańcuchy musiały być spójne i kompletne: pomiędzy ich N- i C-końcami nie mogło brakować reszt lub ich atomów ciężkich (innych niż wodór). Obecność wszystkich elementów struktury jest wymagana przez pola wewnętrzne.

8. Łańcuchy musiały składać się wyłącznie ze standardowych aminokwasów. Pola wewnętrzne wymagają parametryzacji wszystkich atomów w białku, która nie zawsze jest dostępna dla reszt zmodyfikowanych (rekord MODRES), takich jak względnie często występująca naturalnie selenometionina [230].

9. Różnica pomiędzy liczbami reszt w łańcuchach nie mogła przekraczać 10% większej z nich, przy czym powodujące ją insercje/delecje mogły występować wyłącznie na N- lub C-końcach sekwencji, natomiast substytucje były zabronione. Zdecydowanie zwiększało to szansę na to, że łańcuchy te będą do siebie podobne pod względem ich struktury trzeciorzędowej [231, 232].

10. Liczba reszt aminokwasowych w całym kompleksie musiała należeć do przedziału $[200, 400]$, co w połączeniu z pozostałymi kryteriami przekładało się na około od 100 do 200 w każdym łańcuchu. Postanowiono pominąć niewielkie struktury, których status białka mógłby podlegać dyskusji oraz te, które charakteryzowały się zwiększonym prawdopodobieństwem występowania w nich wielu domen [233, 234] oraz były nieporęczne pod względem obliczeniowym.

11. Łańcuchy musiały posiadać tylko jedną domenę według bazy CATH, składającą się z pojedynczego segmentu tożsamego z większością ich sekwencji. Warunek uzupełniający do poprzedniego, niezbędny do umożliwienia traktowania łańcuchów przez pole zewnętrzne jako jednolite struktury.

12. Pomiędzy łańcuchami nie mogły występować mostki disiarczkowe. Pole siłowe ECEPP/3 nie obsługuje tworzenia i rozrywania tego rodzaju wiązań, a ustalenie ich *a priori* niemal jednoznaczne określałoby strukturę całego kompleksu, pozbawiając sensu jej poszukiwanie przy pomocy metod optymalizacyjnych. Dodatkowo oznaczałoby to potrzebę posiadania pewnej wiedzy na temat badanych cząsteczek, co jest sprzeczne z założeniami eksperymentu *ab initio*.

Po pobraniu z bazy PDB plików białek spełniających wymienione powyżej warunki, usunięto spośród nich te o redundantnych sekwencjach, przekraczających umowny poziom 33,3% identyczności. Dopasowanie sekwencji zostało wykonane przy pomocy programu NEEDLE z pakietu EMBOSS[3] (european molecular biology open software suite) [235], stosującego algorytm Needlemana-Wunscha. Użyto domyślnych wartości kar za delecje i substytucje, wynoszących odpowiednio 10 i 0,5.

Spośród pozostałych białek usunięto następnie wszystkie, których kształt wskazywał na to, że osiągnięcie przez nie struktury czwartorzędowej wiązało się z istotnymi zmianami konformacyjnymi w łańcuchach. Rozumiane przez to były dwie sytuacje: gdy ich łańcuchy główne były ze sobą splecione, lub gdy dochodziło pomiędzy nimi do wymiany domen. Czynność ta została podyktowana traktowaniem w tym eksperymencie białek jako bryły sztywne, uniemożliwiającym modelowanie innych zmian konformacyjnych niż obroty i translacje całych cząsteczek.

Ostatecznie uzyskano bazę 200 struktur, których identyfikatory PDB są przedstawione w tabeli 2.1. W każdym z nich, pierwszy łańcuch oznaczony był literą A, a drugi – B. Wizualizacja trzech przykładowych białek należących do tej bazy, o identyfikatorach `1ADW` [236], `1C77` [237] i `1FTP` [238], znajduje się na rysunku 2.1. Pierwsza z nich przedstawia kompleks symetryczny, natomiast pozostałe – dwie formy kompleksu niesymetrycznego: liniową oraz nieliniową.

2.1.1. Dodanie atomów wodoru

Pola wewnętrzne typu *all atom* wymagają dostępu do informacji na temat wszystkich atomów wskazywanych przez sekwencję białka. Brak tych danych powoduje uzyskiwanie niedokładnych wyników, lub całkowicie uniemożliwia obliczenia i minimalizację energii w przypadku rygorystycznych programów.

Większość modeli białek zdeponowanych w bazie PDB, została uzyskana przy pomocy techniki rentgenografii strukturalnej (XRD), dlatego zawiera dane tylko atomów ciężkich, a niekiedy nawet samych węgli Cα. Atomy wodoru, stanowiące około 50% wszystkich atomów w białkach, są widoczne w eksperymencie magnetycznego rezonansu jądrowego (NMR), który jest jednak ograniczony do mniejszych cząsteczek [239]. W związku z tym, liczba „kompletnych” struktur w bazie PDB wciąż pozostaje względnie niewielka. Ponieważ pliki PDB z danymi wszystkich białek z bazy danych rozprawy nie posiadały informacji na temat atomów wodoru, zdecydowano się na ich heurystyczne uzupełnienie.

[3] http://emboss.open-bio.org

137L	1A25	1A78	1ADW	1AG9	1AI9	1ATL	1AYO	1B78	1B88
1BD9	1BKZ	1BU5	1CO2	1C3I	1C77	1CBK	1COZ	1CSG	1DOQ
1D1G	1DHF	1DQE	1DZR	1EAJ	1EO6	1EX2	1EYV	1F08	1F1C
1F46	1FLM	1FQT	1FTP	1G17	1G2Q	1GE7	1GY6	1HFY	1HKQ
1HLC	1HPC	1I4S	1I6W	1IAZ	1IFV	1IPI	1IQ6	1J3M	1J3Q
1JR8	1K4Z	1KPT	1L8D	1LFA	1MO8	1M4I	1M4J	1M4R	1MK4
1MKA	1NA8	1NBC	1NCO	1NP8	1NWP	1NWW	1NXM	1OH0	1OPA
1P6O	1PPV	1Q98	1QAH	1QSD	1QZ8	1SGM	1SH8	1SQU	1TFP
1TLJ	1V5X	1V7L	1V8H	1VC1	1VH5	1VJ2	1VLT	1WPN	1WWZ
1X77	1XOX	1YAV	1YOC	1Z3A	1Z9M	1Z9P	1ZVF	2A4N	2A6P
2A9S	2AB0	2BPD	2CAR	2CC0	2D3K	2DC4	2DCT	2EAV	2F3G
2FBN	2FZF	2GJA	2H29	2HJ3	2IDL	2IGI	2J8M	2J96	2O7M
2OE3	2OFC	2OMD	2P5R	2PBR	2Q20	2QSQ	2QV0	2QZT	2SPC
2W2A	2W31	2WCU	2WLV	2XHF	2XOL	2YEM	2YVE	2Z5D	2Z76
2Z9D	2ZB9	2ZGL	2ZOW	2ZWM	3AIA	3CPQ	3CQR	3CT6	3CXK
3D7A	3EVI	3F81	3FOU	3FQC	3FU1	3G46	3GLV	3GRN	3GWN
3HPE	3HUP	3HV2	3I4S	3IA1	3IIR	3IQ3	3IX3	3K3K	3K9U
3L18	3LB2	3LBB	3LYN	3MGK	3N4K	3N7H	3N8E	3NBC	3OCP
3P9X	3PH4	3QU1	3RD3	3RFB	3RHC	3RQ3	3SLZ	3SZJ	3TRF
3TW2	3UJM	3UMZ	3V6G	3VRC	4AUU	4DF0	4E7P	4EC7	4EP4

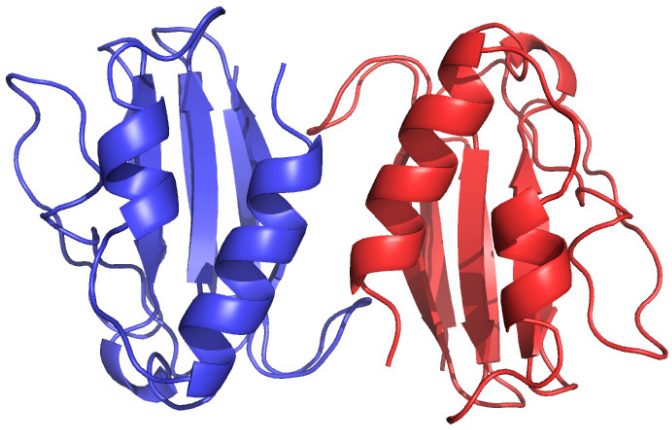
Tabela 2.1: Identyfikatory PDB 200 białek należących do bazy danych rozprawy.

Heurystyczny sposób dodawania atomów wodoru do modeli białek polega na obliczeniu ich współrzędnych na podstawie danych istniejących atomów ciężkich. Do realizacji tego zadania wybrano program REDUCE[4] [240], który uruchomiono z parametrem -ROTEXIST oraz pozostałymi, domyślnymi ustawieniami. Oznacza to, że po uzupełnieniu atomów wodoru została przeprowadzona przez ten program optymalizacja energii potencjału oddziaływań van der Waalsa, usuwająca poprzez obroty grup -OH, -SH i -CH_3 potencjalne kolizje przez niego wywołane.

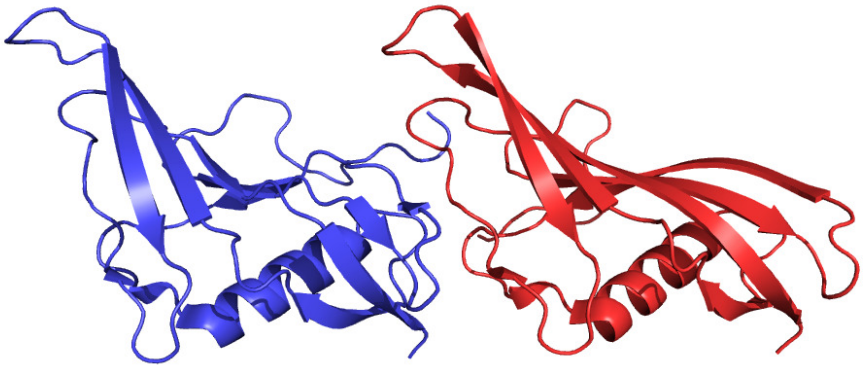
Aby zachować pełną zgodność wyników pola zewnętrznego oraz map kontaktów niewiążących z oryginalnymi strukturami, pozostawiono wyłączony drugi sposób eliminacji kolizji atomów, polegający na wykonywanie odbić lustrzanych kompletnych łańcuchów bocznych w resztach ASN, GLN i HIS.

Wpływ dodania w powyższy sposób atomów wodoru do białek z bazy danych rozprawy na odległości pomiędzy ich łańcuchami oraz wynikające stąd inne ich właściwości jest szczegółowo omówione w rozdziale 3.4.

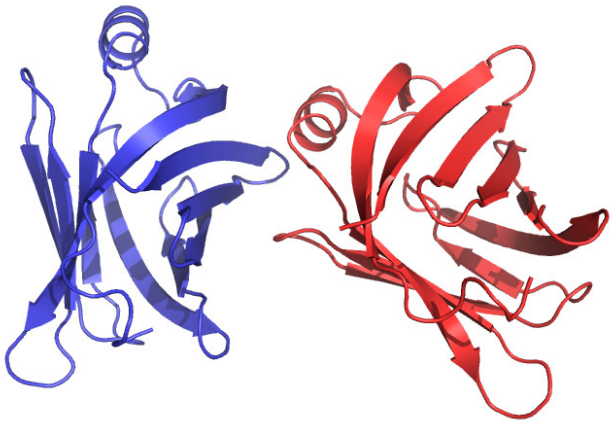
[4] http://kinemage.biochem.duke.edu/software/reduce.php



(a) Kompleks symetryczny – białko `1ADW`.



(b) Kompleks niesymetryczny (forma liniowa) – białko `1C77`.



(c) Kompleks niesymetryczny (forma nieliniowa) – białko `1FTP`.

Rysunek 2.1: Wizualizacja przykładowych białek homodimerycznych z bazy danych rozprawy, prezentujących trzy różne formy ułożenia łańcuchów względem siebie. Na wszystkich rysunkach łańcuch A ma kolor niebieski, a B – czerwony.

2.2. Pole wewnętrzne – pole siłowe ECEPP/3

ECEPP (empirical conformational energy program for peptides) [61–63] jest pierwszym, pół-empirycznym chemicznym polem siłowym typu *all atom*, opracowanym w latach 70-tych na Cornell University w USA przez zespół Harolda Scheragi.

W odróżnieniu od pól siłowych wprowadzonych później, wszystkie wersje pola ECEPP (1, 2, 3 oraz 05 [241]) stosują podczas obliczeń energii oraz konstrukcji łańcuchów polipeptydowych stałe długości wiązań i wartości kątów płaskich ($E_b = E_a = \text{const}$). Każda reszta posiada własny, uniwersalny zestaw danych atomów. Konsekwencją tego podejścia jest zawężenie możliwości wpływania na konformacje łańcuchów wyłącznie do działań na kątach dwuściennych w ich łańcuchach głównych (φ, ψ, ω) i w łańcuchach bocznych (χ_1, χ_2, ...). Pominięcie zmiennych opisujących elastyczność cząsteczki wprowadza silne ograniczenia dla symulacji mechaniki molekularnej, co przekłada się jednak na znaczne zmniejszenie rozmiaru przestrzeni konformacyjnej oraz skrócenie czasu obliczeń i optymalizacji wartości energii. Autorzy pola argumentują swoją decyzję tym, że te same aminokwasy występujące w różnych białkach posiadają zbliżone wartości kątów płaskich i długości wiązań [61]. Dzięki temu, równanie 1.2 w rodzinie pól ECEPP posiada tylko cztery składowe:

$$E = E_e + E_n + E_h + E_t \quad \left[\frac{\text{kcal}}{\text{mol}}\right] \tag{2.1}$$

Różnice pomiędzy pierwszymi trzema wersjami pola ECEPP są względnie niewielkie. Główną motywacją do ich powstania była potrzeba aktualizacji parametryzacji na podstawie coraz większej liczby dostępnych struktur w bazie PDB. Opublikowane na początku 2006 roku ECEPP-05 nie jest kompatybilne z wcześniejszymi wersjami – posiada własną parametryzację i równania energii potencjałów.

Poniżej przedstawiony jest sposób obliczeń wartości potencjałów z równania 2.1 w polu ECEPP/3 dla cząsteczki białka zbudowanej z a atomów, pomiędzy którymi występują wiązania kowalencyjne tworzące b kątów dwuściennych, których wartość może się zmieniać (oddziaływania typu 1-4). Ze względu na ograniczoną ilość miejsca, szczegóły dotyczące parametryzacji, takie jak wartości ładunków cząstkowych, zostały tu pominięte. Można je jednak odnaleźć w cytowanych powyżej artykułach [61–63] oraz w kodzie źródłowym programu ECEPPAK[5]. Więcej informacji na temat rodziny pól ECEPP znajduje się w serii prac opublikowanych pod przewodnim tytułem „energy parameters in polypeptides” [242–248].

[5] http://cbsu.tc.cornell.edu/software/eceppak

2.2.1. Potencjał oddziaływań elektrostatycznych

Do obliczeń energii potencjału oddziaływań elektrostatycznych pary atomów w polu ECEPP/3, tak jak w innych polach siłowych, korzysta się z prawa Coulomba:

$$U_e(i,j) = 332\frac{q_i q_j}{\epsilon r_{ij}} \tag{2.2}$$

gdzie:

i, j = indeksy atomów w białku
r_{ij} = odległość pomiędzy współrzędnymi atomów i oraz j
q_i, q_j = ładunki cząstkowe atomów i oraz j
332 = czynnik konwersji, powodujący wyrażenie U_e w $\frac{\text{kcal}}{\text{mol}}$
ϵ = parametr względnej przenikalności elektrycznej: $\epsilon = 2$

Wartość ładunków cząstkowych zależą od stanu jonizacji reszty oraz tego, czy reszta ta znajduje się ona na jednym z końców łańcucha głównego.

Całkowita energia potencjału elektrostatycznego jest równa sumie wartości równania 2.2 dla oddziaływań typu 1-4 lub wyższych, zapisywanych tu jako i-$j \geq$ 1-4:

$$E_e(a) = \sum_{i=1}^{a-1}\sum_{j=i+1}^{a}\begin{cases} U_e(i,j) & i\text{-}j \geq \text{1-4} \\ 0 & \text{w przeciwnym przypadku} \end{cases} \tag{2.3}$$

2.2.2. Potencjał oddziaływań van der Waalsa

Energia potencjału oddziaływań van der Waalsa pary atomów w polu ECEPP/3 jest wyrażana za pomocą klasycznej funkcji 6–12 Lennarda-Jonesa:

$$U_n(i,j) = FA_{ij}r_{ij}^{-12} - C_{ij}r_{ij}^{-6} \tag{2.4}$$

gdzie:

i,j = indeksy atomów w białku
r_{ij} = odległość pomiędzy współrzędnymi atomów i oraz j
A_{ij} = współczynnik odpychania typów atomów i oraz j
C_{ij} = współczynnik przyciągania typów atomów i oraz j
F = współczynnik korygujący, równy 0,5 dla oddziaływań typu 1-4 oraz 1,0 dla oddziaływań typu 1-5 i wyższych oraz międzycząsteczkowych

W parametryzacji pola ECEPP/3, każdy atom ma przypisany stały typ. Jest to liczba naturalna z przedziału od 1 do 21, wskazująca gdzie w tablicach danych znajdują się wartości niezbędne do obliczenia współczynników A i C, takie jak optymalna odległość oraz odpowiadająca jej minimum energii. Typy atomów są zapisywane w postaci indeksów dolnych, na przykład C_6.

Jeżeli atomy i oraz j mogą brać udział w tworzeniu wiązań wodorowych, w miejscu równania 2.4 stosuje się równanie 2.6, niezależnie od tego, czy spełnione są warunki do powstania tego wiązania (kąty, odległości, itd.). Rozróżnienie pomiędzy tymi dwoma sytuacjami następuje więc wyłącznie na podstawie typów atomów.

Analogicznie do potencjału elektrostatycznego, całkowita energia potencjału van der Waalsa w białku jest tożsama z sumą wartości równania 2.4 dla oddziaływań typu 1-4 lub wyższych:

$$E_n(a) = \sum_{i=1}^{a-1} \sum_{j=i+1}^{a} \begin{cases} U_n(i,j) & i\text{-}j \geq \text{1-4 oraz } i \cdots j \overset{?}{\neq} \mathrm{H} \cdots \mathrm{X} \\ 0 & \text{w przeciwnym przypadku} \end{cases} \tag{2.5}$$

Wyrażenie $i \cdots j \overset{?}{\neq} \mathrm{H} \cdots \mathrm{X}$ oznacza, że atomy i i j nie mogą brać udziału w tworzeniu wiązań wodorowych. Pole ECEPP-05 zastępuje w równaniu 2.4 funkcję Lennarda-Jonesa funkcją 6-exp Buckinghama [249].

2.2.3. Potencjał wiązań wodorowych

Potencjał wiązań wodorowych jest stosowany w miejscu potencjału oddziaływań van der Waalsa w przypadku tych par atomów, które mogą brać udział w tworzeniu wiązań wodorowych: wodór H_2 i H_4 oraz azot N_{14}, tlen O_{17} i tlen O_{18}.

Energia potencjału wiązań wodorowych powyższych par atomów jest wyrażana za pomocą funkcji 10-12, wzmacniającej ich wzajemne przyciąganie:

$$U_{\mathrm{h}}(i,j) = A'_{ij} r_{ij}^{-12} - C'_{ij} r_{ij}^{-10} \tag{2.6}$$

gdzie:

i, j = indeksy atomów w białku
r_{ij} = odległość pomiędzy współrzędnymi atomów i oraz j
A'_{ij} = współczynnik odpychania atomów typów i oraz j
C'_{ij} = współczynnik przyciągania atomów typów i oraz j

Równanie 2.6 nie wyraża całkowitej energii danego wiązania wodorowego, ale jedynie tę jej część, która nie jest określona przez pozostałe oddziaływania, w które są zaangażowane tworzące je atomy. Wartość tej energii w całym białku jest obliczana podobnie jak w przypadku potencjału van der Waalsa:

$$E_h(a) = \sum_{i=1}^{a-1} \sum_{j=i+1}^{a} \begin{cases} U_h(i,j) & i\text{-}j \geq \text{1-4 oraz } i \cdots j \stackrel{?}{=} \text{H} \cdots \text{X} \\ 0 & \text{w przeciwnym przypadku} \end{cases} \tag{2.7}$$

Wyrażenie $i \cdots j \stackrel{?}{=} \text{H} \cdots \text{X}$ oznacza, że atomy i i j mogą brać udział w tworzeniu wiązań wodorowych. Pole ECEPP-05 włącza potencjał wiązań wodorowych do potencjału oddziaływań van der Waalsa.

2.2.4. Potencjał torsyjny

Potencjał torsyjny w polu ECEPP/3 dotyczy barier dla obrotów wokół wiązań kowalencyjnych w cząsteczkach. Jego ogólne równanie ma następującą postać:

$$U_t(k) = \frac{u_k}{2}\Big(1 \pm \cos(n_k \theta_k)\Big) \tag{2.8}$$

gdzie:

k = indeks wiązania w białku
θ_k = kąt obrotu wokół wiązania k
u_k = wysokość bariery energetycznej dla obrotu wokół wiązania k
n_k = krotność bariery energetycznej dla obrotu wokół wiązania k

Analogicznie do potencjału wiązań wodorowych, równanie 2.8 nie wyraża całkowitej energii związanej z barierami dla obrotu wokół danego wiązania kowalencyjnego, a jedynie tę jej część, która nie jest określona przez pozostałe oddziaływania, w które są zaangażowane tworzące je atomy. Wartość tej energii w całym białku jest obliczana tylko dla tych wiązań, wokół których ów obrót jest możliwy ($\theta_k \neq \text{const}$):

$$E_t(b) = \sum_{k=1}^{b} \begin{cases} U_t(k) & \text{jeżeli } \theta_k \neq \text{const} \\ 0 & \text{w przeciwnym przypadku} \end{cases} \tag{2.9}$$

Pole ECEPP zalicza do potencjału torsyjnego również energię związaną z mostkami disiarczkowymi. Do obliczeń jej wartości stosowane jest osobne równanie.

2.2.5. Potencjał mostków disiarczkowych

Mostki disiarczkowe są wiązaniami powstającymi w białkach pomiędzy atomami siarki należącymi do dwóch cystyn. Optymalizacją ich konfiguracji w polu ECEPP/3 zarządza algorytm analizujący współrzędne czterech atomów tworzących razem kąt dwuścienny: $C\beta_i - S\gamma_i - S\gamma_j - C\beta_j$, gdzie i oraz j są indeksami reszt.

Równanie energii mostku disiarczkowego jest podzielone na dwie składowe: obrotu wokół wiązania $S\gamma_i - S\gamma_j$ oraz pętli. Spełniającą one tę samą rolę co równanie 2.8, równocześnie kierując wszystkie cztery atomy w stronę ich optymalnych współrzędnych bez potrzeby obliczania wartości kątów płaskich i dwuściennych między nimi:

$$U_s(i,j) = B\left(r_{4,ij} - r_{4,*}\right)^2 + D\sum_{l=1}^{3}\left(r_{l,ij} - r_{l,*}\right)^2 \tag{2.10}$$

gdzie:

i, j = indeksy reszt w białku
B = współczynnik obrotu
D = współczynnik pętli
$r_{1,ij}$ = aktualna odległość pomiędzy atomami $S\gamma_i$ i $S\gamma_j$
$r_{2,ij}$ = aktualna odległość pomiędzy atomami $C\beta_i$ i $S\gamma_j$
$r_{3,ij}$ = aktualna odległość pomiędzy atomami $S\gamma_i$ i $C\beta_j$
$r_{4,ij}$ = aktualna odległość pomiędzy atomami $C\beta_i$ i $C\beta_j$
$r_{1,*}$ = optymalna odległość pomiędzy atomami $S\gamma_i$ i $S\gamma_j$
$r_{2,*}$ = optymalna odległość pomiędzy atomami $C\beta_i$ i $S\gamma_j$
$r_{3,*}$ = optymalna odległość pomiędzy atomami $S\gamma_i$ i $C\beta_j$
$r_{4,*}$ = optymalna odległość pomiędzy atomami $C\beta_i$ i $C\beta_j$

Pary reszt tworzące mostki disiarczkowe muszą być wskazane *a priori* przez użytkownika. Kwadratowa postać równania 2.10 uniemożliwia symulowanie ich dynamicznego powstawania i rozrywania, co nakłada na konformację cząsteczki kolejne ograniczenie. W przypadku symulacji zwijania białek jest to pożądane zjawisko, ale stanowi przeszkodę podczas eksperymentu kompleksowania, gdyż niemal jednoznacznie wskazuje na natywną strukturę czwartorzędową, która powinna być w trakcie tego eksperymentu odnaleziona bez dodatkowej wiedzy na temat badanego białka.

2.3. Pole zewnętrzne – model FOD

Model rozmytej kropli oliwy (fuzzy oil drop, FOD), opracowany w Uniwersytecie Jagiellońskim – Collegium Medicum w Krakowie przez prof. Leszka Koniecznego i prof. Irenę Roterman-Konieczną [59], opisuje oddziaływania hydrofobowe w białkach, wynikające z ich budowy i reakcji ze środowiskiem wodnym.

Autor niniejszej rozprawy od ponad 8 lat aktywnie uczestniczy w pracach grupy bioinformatycznej UJCM prowadzonej przez Autorów tego modelu [60].

Idea modelu FOD wywodzi się od klasycznego modelu „kropli oliwy" Waltera Kauzmanna [133]. Bazuje on na hipotezie twierdzącej, że jednym z czynników biorących udział w tworzeniu się oraz odpowiedzialnych za stabilizację struktury trzecio- i czwartorzędowej białek są ich oddziaływania z wodą. Zgodnie z tą hipotezą, w trakcie procesu zwijania, reszty o hydrofobowych łańcuchach bocznych powinny kierować się ku środkowi geometrycznemu cząsteczki, tworząc jądro hydrofobowe, natomiast reszty hydrofilne powinny je osłaniać, tworząc wyeksponowany do środowiska wodnego płaszcz hydrofilny. Identyczne zasady zarządzają kompleksowaniem białek. Model FOD oczekuje, że będą one łączyć się ze sobą w taki sposób, aby zasłaniać wyeksponowane powierzchnie hydrofobowe przed wodą, a także uzupełniać hydrofilne kieszenie pasującymi do nich ligandami.

Słowo „rozmyty" w nazwie modelu FOD wynika stąd, że w miejscu binarnych cech reszt z modelu Kauzmanna (hydrofilna / hydrofobowa oraz jądro / płaszcz), stosuje wartości ciągłe, pozwalające na bardziej szczegółową obserwację ich właściwości. Do obliczeń wymagana jest znajomość struktury trzeciorzędowej, co może stanowić przeszkodę w przypadku posiadania wyłącznie sekwencji białek. Z drugiej strony, dzięki temu, że model FOD działa w przestrzeni reszt, ignorując wiązania pomiędzy nimi, obliczenia te mogą być wykonywane dla dowolnej kompozycji aminokwasów, w szczególności kompleksu, łańcucha, lub domeny.

Reszty w modelu FOD są reprezentowane przez atomy efektywne. Do wyznaczania ich „rozmytego" statusu przynależności do jądra lub płaszcza analizowanej struktury służy trójwymiarowa funkcja Gaussa [250]. Aby móc jej użyć, białko musi zostać wpisane w elipsoidę „kropli", symulującą globularny kształt cząsteczki. Bryła ta jest dopasowywana do atomów efektywnych w taki sposób, aby jej średnice wskazywały na kierunki największej zmienności ich współrzędnych. Na podstawie tych średnic są obliczane odchylenia standardowe funkcji Gaussa, powodujące, że jej wartości w pobliżu powierzchni elipsoidy stają się bliskie 0, natomiast maksimum, tożsame ze środkiem geometrycznym tej bryły, umieszcza się w początku układu współrzędnych.

Obliczony przy pomocy funkcji Gaussa status reszt jest nazywany teoretycznym, gdyż wskazuje jaka powinna być ich hydrofobowość, gdyby zbudowana z nich struktura była w pełni zgodna z założeniami modelu FOD. Wartości te są następnie konfrontowane z ich odpowiednikami, wyrażającymi drugi z „rozmytych" statusów. Przedstawia on obserwowaną zmianę hydrofobowości własnej reszt, wynikającą z ich oddziaływań hydrofobowych z sąsiednimi resztami, których siłę, zależną od odległości między nimi, wyznacza się przy pomocy wielomianu Michaela Levitta [251].

Porównanie rozkładów hydrofobowości teoretycznej i obserwowanej umożliwia analizę charakterystyki hydrofobowej białek. Podobieństwo tych rozkładów sugeruje zgodność struktury z wyidealizowanym stanem, co w opinii modelu oznacza, że posiada ona stabilne jądro hydrofobowe. Analiza ta może być również prowadzona lokalnie, gdy rozpatrywany jest status indywidualnych reszt. Różnice w rozkładach hydrofobowości mogą wskazywać na potencjalne miejsca kompleksowania [252, 253] oraz inne obszary mające znaczenie z punktu widzenia funkcji biologicznej [254].

2.3.1. Przygotowanie struktury

Przed przystąpieniem do obliczeń rozkładów hydrofobowości, niezbędne jest wstępne przygotowanie struktury białka, polegające na uporządkowaniu jej danych, obliczeniu atomów efektywnych oraz wpisaniu ich w elipsoidę „kropli".

Uporządkowania danych

Modele białek są pobierane z bazy PDB. Ponieważ stosowany przez RCSB format ich zapisu jest dość swobodny, aby można było prawidłowo obliczyć atomy efektywne reszt i przypisać im wartości hydrofobowości własnej, w niektórych przypadkach zachodzi potrzeba wykonania poniższych czynności:

1. Jeżeli w eksperymencie krystalograficznym zauważono kilka alternatywnych współrzędnych danego atomu (kolumna ALTLOC), wybierane jest to, które zostało stwierdzone jako zajmowane najczęściej (kolumna OCCUPANCY), a pozostałe są usuwane. W sytuacji, gdy wszystkie są pod tym względem równoważne, pozostawiane jest pierwsze z nich, typowo oznaczane literą A.

2. Jeżeli białko zawiera reszty zmodyfikowane (rekordy MODRES), których aminokwasy nie posiadają wpisów w stosowanej skali hydrofobowości, na czas obliczeń są im nadawane nazwy ich standardowych 20 pierwowzorów [255], na przykład, selenometionina (MSE) jest zastępowana metioniną (MET).

Wyznaczenie atomów efektywnych

Ponieważ obliczenia rozkładów hydrofobowości modelu FOD są wykonywane dla reszt, każda z nich jest upraszczana do reprezentującego ją atomu efektywnego. Duże ligandy, takie jak czerwień Kongo, mogą mieć kilku takich reprezentantów.

Atom efektywny reszty jest to wektor odpowiadający średniej arytmetycznej współrzędnych jej atomów ciężkich (wszystkich oprócz wodoru). Ewentualne braki, wynikające najczęściej z niedokładności eksperymentu krystalograficznego, są ignorowane. Oznacza to, że model FOD jest w stanie zaakceptować każdą resztę, o ile określony został dla niej parametr hydrofobowości własnej. Dzięki temu, możliwa staje się również praca z białkami zapisanymi w plikach PDB zawierających wyłącznie współrzędne atomów Cα, na przykład fragmentami dużych kompleksów.

W tym miejscu następuje również wybór interesującego użytkownika fragmentu struktury, dla którego mają być wyznaczone rozkłady hydrofobowości modelu.

Wpisanie atomów efektywnych w elipsoidę „kropli”

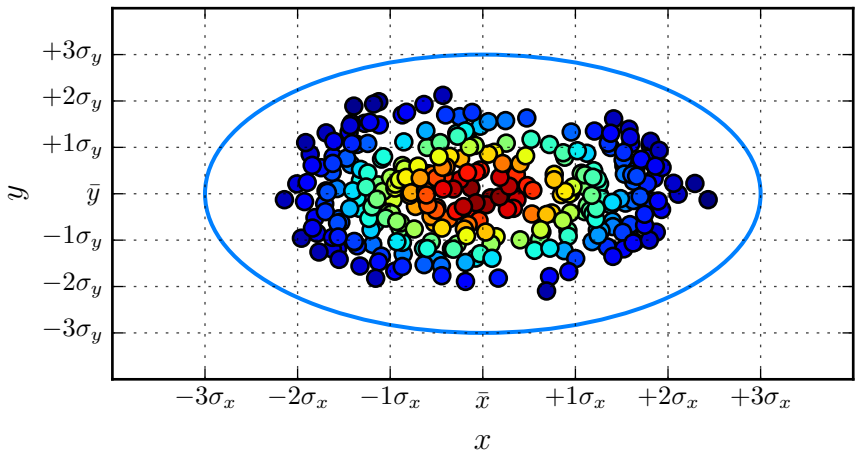
Ostatnią czynnością przygotowawczą jest wpisanie atomów efektywnych w elipsoidę „kropli”, której środek ma znaleźć się w początku układu współrzędnych, a średnice mają być równoległe do osi tego układu. Celem tej czynności jest umożliwienie obliczeń rozkładu hydrofobowości teoretycznej przy pomocy trójwymiarowej funkcji Gaussa. Do jej wykonania stosowany jest poniższy algorytm, dążący do maksymalizacji wariancji atomów efektywnych w kolejnych wymiarach przestrzeni:

1. translacja środka geometrycznego do początku układu współrzędnych,

2. obrót powodujący ułożenie średnicy (odcinka łączącego najbardziej odległą od siebie parę atomów efektywnych) równolegle do osi X,

3. obrót wokół osi X powodujący ułożenie średnicy rzutu atomów efektywnych na płaszczyznę YZ równolegle do osi Y.

Do wpisania tak ułożonego zbioru atomów efektywnych w elipsoidę „kropli” pozostaje już tylko wyznaczenie długości jej promieni. W związku z tym, odnajdywane są te atomy efektywne, które w każdym wymiarze przestrzeni znajdują się najdalej od początku układu współrzędnych. Promienie elipsoidy „kropli” stają się wówczas równe wartościom bezwzględnym ich odpowiednich współrzędnych, powiększonych o maksymalny zasięg oddziaływań hydrofobowych wynoszący 9 Å [251].



Rysunek 2.2: Prezentacja rzutu na płaszczyznę XY atomów efektywnych łańcucha A z przykładowego białka `1UJ1` po wpisaniu ich w elipsoidę „kropli" przez model FOD. Kolory znaczników wskazują na odpowiadające im wartości funkcji Gaussa (dla uproszczenia – dwuwymiarowej): czerwony – wysokie, niebieski – niskie. Na osiach wykresu zaznaczone są wielokrotności odchyleń standardowych, równych $\frac{1}{3}$ długości promieni elipsoidy. Punkt $[\bar{x}, \bar{y}]$ jest tożsamy z początkiem układu współrzędnych.

Na podstawie promieni elipsoidy „kropli" obliczane są w oparciu o regułę 3-sigma odchylenia standardowe funkcji Gaussa. Więcej informacji na ten temat znajduje się w części rozdziału dotyczącej hydrofobowości teoretycznej. Prezentacja wyników zastosowania algorytmu wpisywania w elipsoidę „kropli" atomów efektywnych łańcucha A z przykładowego białka `1UJ1` znajduje się na rysunku 2.2. Choć jest to homodimer, nie został dołączony do bazy danych rozprawy ze względu na zbyt dużą liczbę reszt (301) oraz obecność trzech domen (za CATH: 3–14+100–197, 15–99 i 198–301).

2.3.2. Rozkłady hydrofobowości

Podstawowym zadaniem modelu FOD jest wyznaczenie wartości trzech rozkładów hydrofobowości dla n reszt tworzących strukturę wpisaną w elipsoidę „kropli":

1. rozkładu hydrofobowości własnej: $\tilde{H}r \equiv \left\{\tilde{H}r_1, \ldots, \tilde{H}r_n\right\}$,
2. rozkładu hydrofobowości obserwowanej: $\tilde{H}o \equiv \left\{\tilde{H}o_1, \ldots, \tilde{H}o_n\right\}$,
3. rozkładu hydrofobowości teoretycznej: $\tilde{H}t \equiv \left\{\tilde{H}t_1, \ldots, \tilde{H}t_n\right\}$.

Wartości w tych rozkładach są umieszczane w kolejności występowania reszt w sekwencji i zgodnie z porządkiem alfabetycznym identyfikatorów łańcuchów.

Reszta	$\tilde{H}r$	Reszta	$\tilde{H}r$
LYS	0,001	ALA	0,572
GLU	0,083	HIS	0,628
ASP	0,167	TYR	0,700
GLN	0,250	LEU	0,783
ARG	0,272	VAL	0,811
ASN	0,278	MET	0,828
PRO	0,300	TRP	0,856
SER	0,422	ILE	0,883
THR	0,478	PHE	0,906
GLY	0,550	CYS	1,000

Tabela 2.2: Skala parametru hydrofobowości własnej modelu FOD. Aminokwasy są ułożone od najbardziej hydrofilnego (LYS) do najbardziej hydrofobowego (CYS).

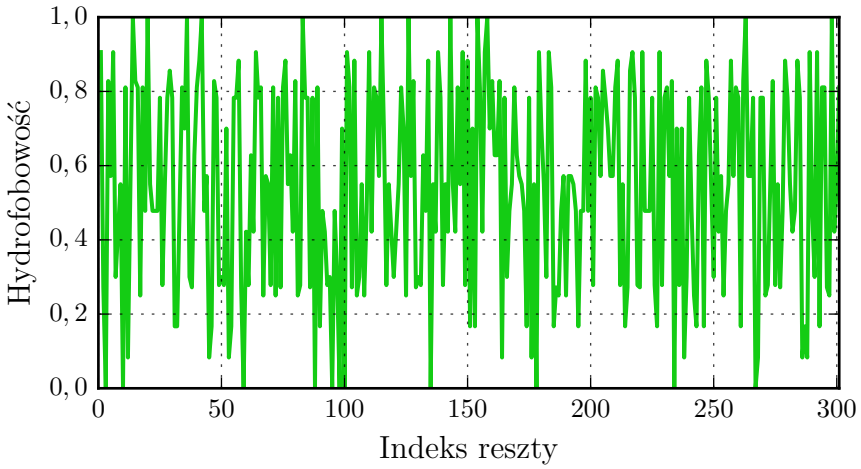
Hydrofobowość własna

Rozkład hydrofobowości własnej, $\tilde{H}r$, zawiera wartości hydrofobowości własnej reszt. Są one pobierane ze skal hydrofobowości i stanowią jedyną parametryzację modelu FOD. Oznacza to, że reszty tworzące analizowaną strukturę, które nie posiadają odpowiadających im wpisów w tych skalach nie mogą brać udziału w dalszych obliczeniach. Jest to powód, dla którego wykonuje się zmianę nazw reszt zmodyfikowanych. Powodowane przez tę czynność przybliżenie jest uznawane za mniej problematyczne od ich usunięcia, wprowadzającego luki w sekwencji, które zaburzają ciągłość łańcuchów i mogą przez to jeszcze bardziej zmieniać wartości elementów wszystkich trzech rozkładów.

Model FOD dysponuje własną, domyślnie stosowaną skalą hydrofobowości, zawierającą informacje na temat standardowych 20 aminokwasów. Przypisane im wartości, należące do przedziału od 0 do 1, są przedstawione w tabeli 2.2. Skala ta została opracowana empirycznie, na podstawie analizy nieredundantnego zbioru jednodomenowych białek wybranych z bazy PDB [256, 257].

W miejscu skali hydrofobowości modelu FOD mogą być również stosowane inne skale, których przegląd oraz klasyfikację prezentuje publikacja Simma i współpracowników [258]. Jedną z nich jest klasyczna skala Kyte-Doolittle [259], z którą model FOD utrzymuje kompatybilność w sensie zwracanych przez niego wyników [142].

Rozkład $\tilde{H}r$ dla łańcucha A z białka `1UJ1` jest widoczny na rysunku 2.3. Białko to jest konsekwentnie stosowane jako przykład w całej tej części rozdziału.



Rysunek 2.3: Rozkład hydrofobowości własnej modelu FOD obliczony dla łańcucha A z przykładowego białka `1UJ1`. Przypisuje on kolejnym resztom w jego sekwencji wartości hydrofobowości pobrane ze stosowanej skali hydrofobowości (tabela 2.2).

Hydrofobowość obserwowana

Rozkład hydrofobowości obserwowanej, $\tilde{Ho}$, przedstawia wpływ oddziaływań hydrofobowych pomiędzy resztami z analizowanej struktury na przypisane im wartości hydrofobowości własnej. Odróżnia to model FOD od innych podejść, które nie wychodzą poza stałe ze skal hydrofobowości. Dzięki temu, że uznaje on hydrofobowość za cechę całej struktury, reszty, w zależności od ich sąsiedztwa, mogą zostać zaobserwowane jako mniej lub bardziej hydrofobowe niż wynikałoby to z ich parametryzacji. Stąd pochodzi nazwa tego rozkładu, określanego również mianem empirycznego.

Siłę oddziaływań hydrofobowych w modelu FOD wyznacza wielomian Michaela Levitta [251]. Jest to funkcja odległości pomiędzy dwoma atomami efektywnymi, r:

$$g(r,c) = 1 - \frac{1}{2}\left(7\left(\frac{r}{c}\right)^2 - 9\left(\frac{r}{c}\right)^4 + 5\left(\frac{r}{c}\right)^6 - \left(\frac{r}{c}\right)^8\right) \tag{2.11}$$

Drugi parametr występujący w powyższym równaniu, c, wskazuje na maksymalny zasięg oddziaływań hydrofobowych, wynoszący 9 Å [251].

Wielomian Levitta osiąga swoje maksimum równe 1 dla $r = 0$ oraz 0 dla $r = c$. Gdy r staje się większe od c, jego wartości zaczynają dążyć do $-\infty$. Kształt wykresu wielomianu Levitta dla $r \in [0, c]$ bardzo przypomina kształt funkcji Gaussa o wartości oczekiwanej 0 i odchyleniu standardowym $\frac{c}{3}$.

Wartość hydrofobowości obserwowanej i-tej reszty jest obliczana w następujący sposób:

$$\tilde{H}o_i = \frac{1}{\tilde{H}o_{\text{sum}}} \sum_{j=1}^{n} \begin{cases} \left(\tilde{H}r_i + \tilde{H}r_j\right) g\left(r_{ij}, c\right) & \text{jeżeli } i \neq j \text{ oraz } r_{ij} \leq c \\ 0 & \text{w przeciwnym przypadku} \end{cases} \tag{2.12}$$

gdzie:

$\tilde{H}r_i$ = hydrofobowość własna reszty i
$\tilde{H}r_j$ = hydrofobowość własna reszty j
r_{ij} = odległość pomiędzy atomami efektywnymi reszt i i j
$\tilde{H}o_{\text{sum}}$ = czynnik normalizujący hydrofobowość obserwowaną:

$$\tilde{H}o_{\text{sum}} = \sum_{i=1}^{n} \tilde{H}o_i \tag{2.13}$$

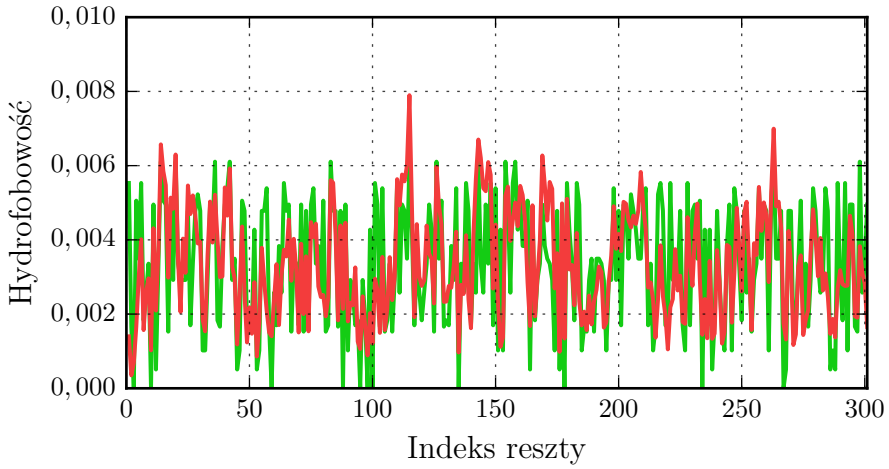
Interpretacja równania 2.12 jest taka, że każda reszta „zbiera” hydrofobowość z reszt w promieniu $c = 9\,\text{Å}$ od niej i dodaje ją do swojej hydrofobowości własnej. Nie zależy to od orientacji całości zbioru atomów efektywnych w przestrzeni.

Aby stwierdzić w którą stronę oraz jak daleko nastąpiło przesunięcie reszt na skali hydrofobowości własnej po ich osadzeniu w analizowanej strukturze, wykonywana jest normalizacja rozkładu $\tilde{H}o$. Polega ona na podzieleniu wszystkich jego elementów przez wartość czynnika $\tilde{H}o_{\text{sum}}$ (równanie 2.13). Resztom, które dają się wówczas zaobserwować na tle pozostałych jako hydrofilne przydzielane są względnie niskie wartości, a hydrofobowym – wysokie. Zachowanie przez nie swojej własnej charakterystyki jest natomiast wskazywane przez $\tilde{H}o_i$ zbliżone do $\tilde{H}r_i/\tilde{H}r_{\text{sum}}$.

Rozkład $\tilde{H}o$ dla łańcucha A z białka `1UJ1` jest przedstawiony na rysunku 2.4. Dla porównania, znajduje się na nim również znormalizowany rozkład hydrofobowości własnej reszt z rysunku 2.3.

Hydrofobowość teoretyczna

Rozkład hydrofobowości teoretycznej, $\tilde{H}t$, jest ostatnim z trzech rozkładów modelu FOD. Nazywa się go również rozkładem wyidealizowanym. Określenie to wynika stąd, że prezentuje on charakterystykę analizowanej struktury odzwierciedlającą hipotezę, na której bazuje model FOD. Dla przypomnienia, zgodnie z nią, we wnętrzu białka powinno znajdować się jądro hydrofobowe, otoczone przez izolujący je od wody płaszcz hydrofilny, zapewniające całości stabilność oraz rozpuszczalność.



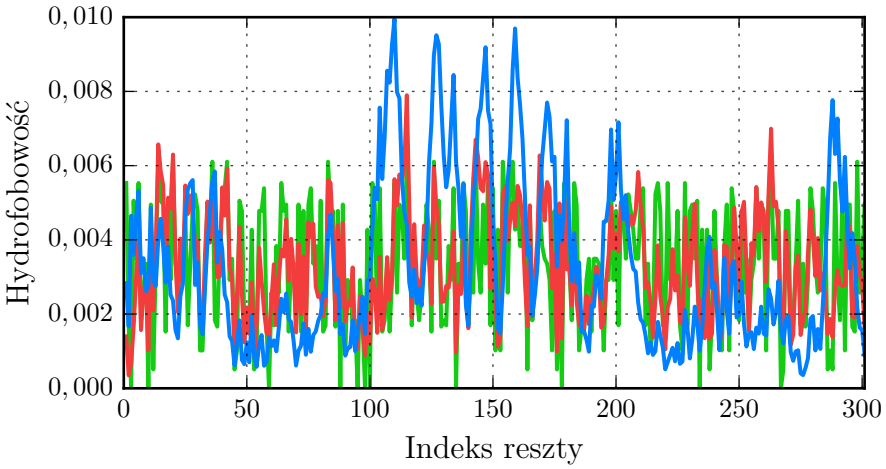
Rysunek 2.4: Rozkład hydrofobowości obserwowanej modelu FOD (kolor czerwony), obliczony dla łańcucha A z białka `1UJ1`. Znajduje się tu również rozkład $\tilde{H}r$ z rysunku 2.3 (kolor zielony) w postaci znormalizowanej. Dzięki temu widać, jak sąsiedztwo każdej reszty w strukturze wpływa na postrzeganie jej hydrofobowości.

Resztom, które znajdują się w środku struktury białka nadawana jest najwyższa wartości hydrofobowości teoretycznej, a tym, które znajdują się coraz dalej od niej – coraz niższe. Do modelowania tego spadku służy wspominana już wcześniej trójwymiarowa funkcja Gaussa. Wektor wartości oczekiwanej, $[\bar{x}, \bar{y}, \bar{z}]$, jest tożsamy z początkiem układu współrzędnych, natomiast wektor odchyleń standardowych, $[\sigma_x, \sigma_y, \sigma_z]$, jest dobierany w taki sposób, aby wartości tej funkcji zbliżały się do 0 przy powierzchni elipsoidy „kropli" i poza nią. Osiągane jest to poprzez ustawienie tych odchyleń standardowych na $\frac{1}{3}$ długości jej promieni. Zgodnie z regułą 3-sigma, powoduje to, że 99,7% pola pod wykresem funkcji Gaussa znajduje się w każdym wymiarze w przedziale $[-3\sigma, +3\sigma]$.

Hydrofobowość teoretyczna jest niezależna od parametryzacji reszt. Decydują o niej wyłącznie współrzędne atomów efektywnych, a w szczególności ich otoczka wypukła. Jej elementy mają bowiem największe znaczenie podczas układania cząsteczki zgodnie z osiami układu współrzędnych. Konsekwencje tego zjawiska są omówione szczegółowo w rozdziale 3.3.

Wartość hydrofobowości teoretycznej i-tej reszty jest obliczana w następujący sposób:

$$\tilde{H}t_i = \frac{1}{\tilde{H}t_{\text{sum}}} \exp\left(\frac{-(x_i - \bar{x})^2}{2\sigma_x^2}\right) \exp\left(\frac{-(y_i - \bar{y})^2}{2\sigma_y^2}\right) \exp\left(\frac{-(z_i - \bar{z})^2}{2\sigma_z^2}\right) \tag{2.14}$$



Rysunek 2.5: Rozkład hydrofobowości teoretycznej modelu FOD (kolor niebieski), obliczony dla łańcucha A z białka `1UJ1`. Znajdują się tu również rozkłady $\tilde{H}o$ (kolor czerwony) i $\tilde{H}r$ (kolor zielony) z rysunku 2.4.

gdzie:

$[x_i, y_i, z_i]$ = wektor współrzędnych atomu efektywnego i-tej reszty
$[\bar{x}, \bar{y}, \bar{z}]$ = wektor środka elipsoidy „kropli”, równy $[0, 0, 0]$
$[\sigma_x, \sigma_y, \sigma_z]$ = wektor $\frac{1}{3}$ promieni elipsoidy „kropli”
$\tilde{H}t_{\text{sum}}$ = czynnik normalizujący:

$$\tilde{H}t_{\text{sum}} = \sum_{i=1}^{n} \tilde{H}t_i \tag{2.15}$$

Rozkład $\tilde{H}t$ jest normalizowany w analogiczny sposób do rozkładu $\tilde{H}o$, przy pomocy czynnika $\tilde{H}t_{\text{sum}}$ z równania 2.15. Pozwala to dodatkowo na traktowanie go jako rozkładu gęstości prawdopodobieństwa, określającego oczekiwaną szansę na napotkanie reszty hydrofobowej w danym miejscu w przestrzeni [124].

Na rysunku 2.2 widać, że wydłużenie promieni znajdującej się na nim elipsy o 9 Å powoduje, że zawiera ona w sobie wszystkie atomy efektywne białka. Ponieważ funkcja Gaussa maleje asymptotycznie do 0, dopuszczalne jest aby niektóre z nich znajdowały się poza nią. Wydłużanie promieni elipsoidy „kropli” ma również inny cel, jakim jest rozciąganie, a przez to spłaszczanie kształtu funkcji Gaussa, zmierzające do zmniejszania zbyt dużych różnic w wartościach hydrofobowości teoretycznej pomiędzy resztami położonymi w pobliżu środka tej bryły, a pozostałymi.

Rozkład $\tilde{H}t$ dla łańcucha A z białka `1UJ1` jest przedstawiony na rysunku 2.5. Ponownie, dla porównania zostały dołączone do niego poprzednie dwa rozkłady.

2.3.3. Analiza rozkładów hydrofobowości

Białko o obserwowanej charakterystyce hydrofobowej identycznej z rozkładem teoretycznym ($\tilde{H}t \equiv \tilde{H}o$) byłoby doskonale rozpuszczalne w wodzie. Jednocześnie nie posiadałoby cech aktywności biologicznej, rozumianej tutaj jako oddziaływanie enzymu z substratem lub tworzenie kompleksów [124]. W naturze istnieją rodziny białek o zbliżonych właściwościach, takie jak białka antifreeze (typ II – do 100 reszt), których funkcja wymaga właśnie silnego powinowactwa do wody [260, 261]. Owa „bierna" aktywność biologiczna polega bowiem na przeciwdziałaniu strukturalizacji wody, co w przypadku ryb umożliwia im przeżycie w niskich temperaturach.

Funkcja białek musi wiązać się z występowaniem pewnych odstępstw od wyidealizowanej „kropli". Mogą być one lokalne – ograniczone do konkretnego zbioru reszt, lub globalne – zmieniające sposób postrzegania całej struktury [124]. Zakładając, że wpływ oddziaływań hydrofobowych na procesy zwijania i kompleksowania białek może być modelowany poprzez dążenie do maksymalizacji podobieństwa rozkładów $\tilde{H}t$ i $\tilde{H}o$, na podstawie analizy różnic pomiędzy tymi rozkładami można wnioskować na temat stabilności, funkcji oraz genezy tych cząsteczek, a także symulować te procesy za pomocą metod optymalizacyjnych.

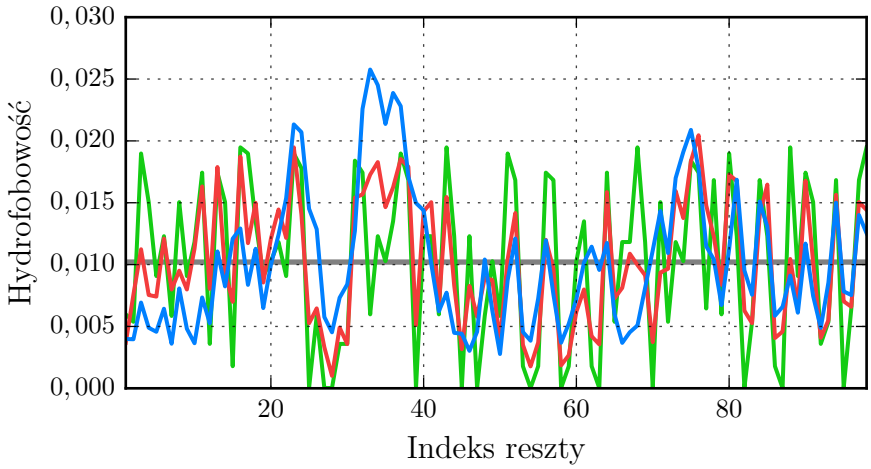
Na rysunku 2.5 są widoczne wyraźnie różnice pomiędzy rozkładami $\tilde{H}t$ i $\tilde{H}o$ w białku `1UJ1`. W jego przypadku, powodem tych rozbieżności jest obecność trzech domen, które powinny być rozpatrywane osobno. Indywidualnie, każda z nich wykazuje wysoką zgodność pomiędzy odpowiadającymi jej rozkładami hydrofobowości, podobną do tej prezentowanej przez białko `1MB1` [262] na rysunku 2.6.

Podczas porównywania rozkładów hydrofobowości, każda reszta może znaleźć się w jednej z poniższych sytuacji:

1. jeżeli $\tilde{H}t_i \approx \tilde{H}o_i$, wówczas i-ta reszta jest traktowana jako zgodna z hipotezą modelu FOD, ponieważ jej hydrofobowość obserwowana, wynikająca z oddziaływań hydrofobowych z jej najbliższymi sąsiadami, odpowiada teoretycznym oczekiwaniom wobec zajmowanego przez nią miejsca w strukturze,

2. jeżeli $\uparrow\tilde{H}t_i$ oraz $\downarrow\tilde{H}o_i$, wówczas i-ta reszta znajdująca się we wnętrzu struktury wykazuje niedobór hydrofobowości obserwowanej, co sugeruje możliwą obecność w tym miejscu kieszeni wiązania liganda [252],

3. jeżeli $\downarrow\tilde{H}t_i$ oraz $\uparrow\tilde{H}o_i$, wówczas i-ta reszta znajdująca się na powierzchni białka wykazuje nadmiar hydrofobowości obserwowanej, co sugeruje możliwą obecność w tym miejscu interfejsu kompleksowania typu białko-białko [253].



Rysunek 2.6: Rozkłady hydrofobowości modelu FOD obliczone dla białka `1MB1`: kolor niebieski – $\tilde{H}t$, czerwony – $\tilde{H}o$, zielony – $\tilde{H}r$ (znormalizowany). Szara pozioma linia odpowiada jednorodnemu rozkładowi $\tilde{R}$. W odróżnieniu od białka `1UJ1`, charakterystyka hydrofobowa tej struktury jest w sensie dywergencji Kullbacka-Leiblera zgodna z modelem FOD. Nawet bez niej widać tu duże podobieństwo rozkładów $\tilde{H}t$ i $\tilde{H}o$.

Kwartyle hydrofobowości

Pierwszy, lokalny, sposób porównywania ze sobą rozkładów hydrofobowości polega na podziale przedziałów ich wartości przy pomocy kwartyli [149]. Reszty, które znajdują się w obydwu przypadkach w tych samych ćwiartkach są uznawane za zgodne z hipotezą modelu FOD. Szczególne znaczenie mają dwie poniższe sytuacje:

- jeżeli wartości $\tilde{H}t_i$ i $\tilde{H}o_i$ należą do górnych 25% swoich rozkładów oznacza, że i-ta reszta jest częścią jądra hydrofobowego,
- jeżeli wartości $\tilde{H}t_i$ i $\tilde{H}o_i$ należą do dolnych 25% przedziałów wartości swoich rozkładów oznacza, że i-ta reszta jest częścią płaszcza hydrofilnego.

Analogicznie mogą być wyznaczone reszty niezgodne z hipotezą modelu FOD:

- $\tilde{H}t_i$ w czwartej ćwiartce i $\tilde{H}o_i$ w pierwszej ćwiartce oznacza $\uparrow\tilde{H}t_i$ oraz $\downarrow\tilde{H}o_i$, czyli potencjalne miejsce wiązania liganda,
- $\tilde{H}t_i$ w pierwszej ćwiartce i $\tilde{H}o_i$ w czwartej ćwiartce oznacza $\downarrow\tilde{H}t_i$ oraz $\uparrow\tilde{H}o_i$, czyli potencjalne miejsce kompleksowania typu białko-białko.

Powyższe podejście zastępuje stosowaną wcześniej miarę $\Delta\tilde{H}$ [263].

Dywergencja Kullbacka-Leiblera

Porównywanie wartości rozkładów hydrofobowości umożliwia ocenę statusu indywidualnych reszt, ale nie pozwala na stwierdzenie, czy białko, łańcuch, lub domena, do których te reszty należą wykazuje jako całość cechy podobieństwa do swojego odpowiednika charakteryzowanego przez trójwymiarową funkcję Gaussa. Mówiąc inaczej – czy cząsteczka ta posiada wykształcone jądro hydrofobowe.

Do wymiernej oceny podobieństwa rozkładów $\tilde{H}t$ i $\tilde{H}o$ służy w modelu FOD dywergencja Kullbacka-Leiblera [264, 265]. Zastosowanie jej jest możliwe dzięki normalizacji tych rozkładów oraz dodatnich wartości ich elementów.

Dywergencja Kullbacka-Leiblera, D_{KL}, zwana również entropią względną, określa ilość informacji utraconej w wyniku przybliżenia rzeczywistego stanu układu (P) przez model (Q) [150]. Dla danych dyskretnych, oblicza się ją w następujący sposób:

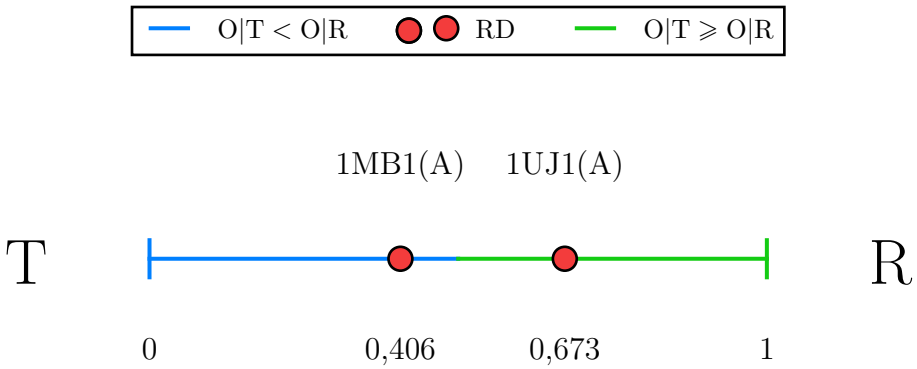
$$D_{KL}(P\|Q) = \sum_{i=1}^{n} \left(P_i \cdot \log_2\left(\frac{P_i}{Q_i}\right)\right) \tag{2.16}$$

W modelu FOD, P jest rozkładem obserwowanym, a Q – teoretycznym. Wartość $D_{KL}(\tilde{H}o\|\tilde{H}t)$ oznacza więc „odległość" pomiędzy tymi rozkładami.[6]

Rozkłady $\tilde{H}t$ i $\tilde{H}o$ są tym bardziej do siebie podobne im obliczona dla nich wartość D_{KL} jest niższa. Entropia nie może być jednak rozpatrywana w kategoriach absolutnych – nie istnieje uniwersalny próg określający kiedy można uznać to podobieństwo za znaczące dla stabilności struktury. Mówiąc inaczej, aby móc wnioskować na podstawie D_{KL}, niezbędne jest przyjęcie jakiegoś punktu odniesienia. W modelu FOD stanowi go inny rozkład teoretyczny, o przeciwnej charakterystyce do $\tilde{H}t$. Traktuje on wszystkie reszty jako jednorodnie hydrofobowe, co w efekcie uniemożliwia podział cząsteczki pod tym względem na jądro i płaszcz. Rozkład ten jest nazywany „random" ($\tilde{R}$). Można go zaobserwować na rysunku 2.6 w postaci szarej poziomej linii przecinającej oś pionową w punkcie odpowiadającym wartości $\frac{1}{n}$, co wynika stąd, że w celu porównania z pozostałymi, również musi być znormalizowany:

$$\tilde{R} = \underbrace{\left\{n^{-1}, \ldots, n^{-1}\right\}}_{n \text{ razy}} \tag{2.17}$$

[6] Słowo odległość zostało tu wzięte w cudzysłów, ponieważ miara D_{KL} nie jest metryką. Wprawdzie osiąga ona minimum równe 0 gdy $P \equiv Q$ i rośnie wraz z powiększaniem się różnic pomiędzy tymi rozkładami, ale nie spełnia warunków nierówności trójkąta oraz symetryczności. W przypadku modelu FOD jest to jednak bez znaczenia.



Rysunek 2.7: Oś liczbowa miary RD z umieszczonymi na niej znacznikami odpowiadającym wartościom obliczonym na podstawie rozkładów z rysunków 2.5 i 2.6.

Zgodność charakterystyki hydrofobowej białka z hipotezą modelem FOD może być wyrażona w postaci relacji pomiędzy wartościami $O\|T$ i $O\|R$, gdzie:

$$O\|T = D_{KL}\left(\tilde{H}o\|\tilde{H}t\right) \tag{2.18}$$

$$O\|R = D_{KL}\left(\tilde{H}o\|\tilde{R}\right) \tag{2.19}$$

Jeżeli zachodzi $O\|T < O\|R$, wówczas rozkład $\tilde{H}t$ lepiej przybliża rozkład $\tilde{H}o$ od rozkładu $\tilde{R}$. Z drugiej strony, nierówność $O\|T \geq O\|R$ oznacza, że analizowana struktura wykazuje więcej cech nieuporządkowania, czyli jest mniej stabilna.

Stosowanie dwóch symboli jest jednak niepraktyczne, w szczególności, że znaczenie ma wyłącznie relacja pomiędzy nimi. Dlatego wprowadzona została wiążąca je ze sobą miara odległości względnej (relative distance, RD) [53]:

$$\mathrm{RD} = \frac{O\|T}{O\|T + O\|R} \tag{2.20}$$

Miara RD przyjmuje wartości z przedziału $[0, 1]$, gdzie 0 oznacza, że $\tilde{H}o \equiv \tilde{H}t$, a 1 – że $\tilde{H}o \equiv \tilde{R}$. W ten sposób, nierówność $O\|T < O\|R$ zostaje zastąpiona przez RD $< 0{,}5$, a $O\|T \geq O\|R$ – przez RD $\geq 0{,}5$.

Status różnych struktur może być teraz przedstawiany i porównywany na skali RD, której przykład, wraz z zaznaczonymi białkami `1UJ1` i `1MB1` znajduje się na rysunku 2.7. Wysoką zgodność z modelem FOD obserwuje się dla RD $< 0{,}4$.

2.4. Optymalizacja rojem cząstek

Optymalizacja rojem cząstek (particle swarm optimization, PSO) jest metaheurystycznym algorytmem optymalizacji globalnej funkcji zmiennych rzeczywistych, przedstawionym w 1995 roku przez Russella Eberharta i Jamesa Kennedy'ego [64]. Tak jak w przypadku innych algorytmów z dziedziny inteligencji obliczeniowej, algorytm ten wzoruje się na zjawiskach występujących w naturze [266]. Inspiracją dla jego Autorów były bowiem modele zachowań występujących wewnątrz grup zwierząt poruszających się w poszukiwaniu źródła pożywienia [267, 268].

Każda cząstka należąca do roju reprezentuje pojedynczego kandydata na rozwiązanie problemu optymalizacji kryterium $f : \mathbb{R}^d \to \mathbb{R}$. Posiada ona trzy właściwości będące wektorami z przestrzeni $\mathbb{R}^d$: położenie, prędkość oraz pamięć, których każda składowa odpowiada jednej spośród d zmiennych.

Definicja 2.1. Cząstka w algorytmie PSO

Cząstka, p, jest to trójka (x, v, m), gdzie:

x = położenie cząstki: $x \in \mathbb{R}^d$

v = prędkość cząstki: $v \in \mathbb{R}^d$

m = pamięć cząstki: $m \in \mathbb{R}^d$

Definicja 2.2. Rój cząstek w algorytmie PSO

Rój cząstek, PSO, jest to trójka $(\mathcal{P}, f, \Phi)$, gdzie:

$\mathcal{P}$ = niepusty zbiór cząstek: $\mathcal{P} \equiv \{p_1, \ldots, p_n\}$

f = kryterium optymalizacyjne: $f : \mathbb{R}^d \to \mathbb{R}$

Φ = zbiór parametrów algorytmu: $\Phi \equiv \{\phi_v, \phi_m, \phi_l, Vmax, \tau, t\}$

Położenie cząstki wskazuje na punkt w przestrzeni rozwiązań, w którym obecnie się ona znajduje, będący równocześnie reprezentowanym przez nią kandydatem na rozwiązanie problemu optymalizacji. Prędkość cząstki określa kierunek, zwrot i zasięg zmiany jej położenia, natomiast pamięć jest tożsama z jednym z jej wcześniejszych położeń, w którym została przez nią zaobserwowana najniższa wartość kryterium f. Z tego powodu, wektor pamięci bywa również nazywany osobistym najlepszym rozwiązaniem (personal best, pbest).

Liczebność roju, n, jest zazwyczaj stała w trakcie trwania optymalizacji, aczkolwiek cząstki mogą być w dowolnym momencie do niego dodawane i z niego usuwane, o ile jest to obsługiwane przez wybraną topologię roju, τ. Stosuje się ją wraz z pozostałymi parametrami ze zbioru Φ podczas aktualizacji algorytmu.

Dzięki temu, że PSO jest algorytmem metaheurystycznym, jedynym warunkiem, jaki musi spełniać kryterium f jest bycie określonym w podzbiorze przestrzeni rozwiązań, w którym działają cząstki. Ponieważ ich położenia odpowiadają bezpośrednio elementom tej przestrzeni, nie muszą zawierać się w tym obszarze i mogą go dowolnie opuszczać. Jeżeli wymagane jest, aby bezwzględnie w nim pozostawały, niezbędne może być użycie funkcji ograniczeń oraz obsługującej je modyfikacji algorytmu.

Ponieważ wartości kryterium f są obliczane *ad hoc*, również ono może być zmieniane w trakcie optymalizacji, teoretycznie bez negatywnych konsekwencji, dopóki rój nie zacznie się zbiegać w którymś z rozwiązań. Jeżeli jednak to już nastąpi, do zachowania sprawności działania algorytmu, potrzebne jest zastosowanie innej z jego modyfikacji, pozwalającej na efektywne dostosowywanie się do kształtu nowego środowiska, czyli podążaniu za dotychczas śledzonym minimum.

Algorytm PSO nie potrafi wykonywać optymalizacji wielomodalnej i wielokryterialnej. Modyfikacje pozwalające na rozwiązywanie przy jego pomocy tych oraz innych rodzajów problemów są omówione w dalszej części tego rozdziału.

Pojedyncza cząstka może liczyć wyłącznie na siebie, błądząc pomiędzy przypadkowymi rozwiązaniami. Sprawność algorytmu PSO wynika z zastosowania ich roju, w którym, dzięki współpracy, uzyskują zdolność do wspólnego przemieszczania się w stronę coraz niższych wartości kryterium f. Sposób, w jaki to następuje zależy od dwóch czynników: indywidualnego i zbiorowego, określanych również jako świadomość cząstek oraz relacje społeczne między nimi.

Świadomość cząstek jest modelowana przy pomocy mechanizmu pamięci. Dzięki niemu, każda z nich może wracać w do odwiedzonego przez nią wcześniej rozwiązania jeżeli wartość kryterium f w jej bieżącym położeniu jest wyższa. Zachowania społeczne polegają natomiast na rozpowszechnianiu w roju informacji o odnalezionych przez cząstki rozwiązaniach. Umożliwia to im na podążanie za innymi cząstkami, które znalazły się w miejscach, gdzie kryterium f przyjmuje niższe wartości niż w punktach przez nie zapamiętanych.

Cząstki, za którymi podążają inne cząstki są nazywane liderami. Ich przydziałem zajmuje się funkcja topologii roju, τ. Efektem wymiany informacji jest częściowe skoordynowanie ruchu cząstek i jej ewentualne zbiegnięcie się ich w pobliżu jednego z punktów, stanowiącego wynik procedury optymalizacji.

Oprócz czynników świadomości i społeczności, cząstki są także poddawane wpływowi zdarzeń losowych („turbulencji"), powodujących nieoczekiwane odchylenia od obranych przez nie trajektorii. Zwiększa to ich zdolność do przeszukiwania przestrzeni rozwiązań, wysyłając je w miejsca, do których inaczej mogłyby nie dotrzeć.

Pomimo tego, że rój cząstek nie jest zaliczany do rodziny algorytmów ewolucyjnych, posiada cechy, które go do nich upodobniają [269]. Dwie najbardziej zauważalne to: sposób działania wzorujący się na naturalnych zjawiskach oraz stosowanie populacji oddziałujących ze sobą osobników reprezentujących kandydatów na rozwiązanie problemu optymalizacyjnego. Algorytmy ewolucyjne korzystają z trzech operatorów: krzyżowania, mutacji i selekcji. Do pierwszego z nich zbliżony jest mechanizm liderów, który decyduje o zmianie położenia każdej cząstki poprzez wymianę posiadanej przez nią informacji z pozostałymi cząstkami. Operatorowi mutacji odpowiada natomiast turbulencja, wykonywana dokładnie w tym samym celu: przemieszczania roju w nowe, niewynikające z krzyżowania miejsca oraz uciekania z pułapek minimów lokalnych. Algorytm PSO nie posiada za to odpowiednika operatora selekcji, gdyż nie występuje w nim dobór naturalny i konkurencja pomiędzy cząstkami. Te słabiej przystosowane zawsze mają szansę na poprawę swojej sytuacji dzięki informacji przekazywanej *pro bono* przez pozostałe, a nawet na stanie się ich liderami.

Wspomnianą wcześniej zaletą algorytmu PSO jest to, że cząstki reprezentują rozwiązania bezpośrednio. Oprócz tego, zakresy wartości i wzajemne relacje pomiędzy optymalizowanymi zmiennymi są dla niego bez znaczenia (każda jest *de facto* optymalizowana osobno, choć w tym samym czasie), co powoduje że algorytm ten stanowi jedną z najprostszych i uniwersalnych metod obliczeniowych.

2.4.1. Inicjalizacja

Inicjalizacja algorytmu PSO polega na rozmieszczeniu w d-wymiarowej przestrzeni rozwiązań roju złożonego z n cząstek. Pozostałe dane, takie jak kryterium f oraz parametry ze zbioru Φ nie są na tym etapie potrzebne.

Początkowe położenia cząstek mogą być ustalane bezpośrednio przez użytkownika, ale najczęściej są wybierane losowo z interesującego go podzbioru przestrzeni rozwiązań. Podzbiór ten jest zazwyczaj wskazywany przy pomocy dwóch d-wymiarowych wektorów: $Xmin$ i $Xmax$, wyznaczających kształt hiperprostokąta o bokach równoległych do osi układu współrzędnych, w którym ma znaleźć się rój. W tym celu, j-ta składowa położenia każdej cząstki, odpowiadająca j-tej optymalizowanej zmiennej, jest wybierana z rozkładu jednorodnego $U\left(Xmin_j, Xmax_j\right)$. Poszukiwane minimum globalne powinno znajdować się w tym hiperprostokącie, ale nie ma takiego obowiązku, w szczególności, gdy kształt krajobrazu wartości kryterium f pozwala rojowi na przemieszczenie się we właściwym kierunku. Wektory $Xmin$ i $Xmax$ nie są potrzebne po inicjalizacji algorytmu i nie muszą być przechowywane w zbiorze Φ.

Początkowa prędkość roju także może być ustalona ręcznie (w szczególności na 0), lub wybrana w sposób losowy. W drugim przypadku, do jej wyznaczenia wystarcza jednak tylko jeden wektor – $Vmax$. Na jego podstawie, j-tej składowej prędkości każdej cząstki jest przypisywana wartość wybrana losowo z rozkładu jednorodnego $U\left(-Vmax_j, Vmax_j\right)$. Jeżeli rój ma początkowo przebywać w hiperprostokącie określonym przez wektory $Xmin$ i $Xmax$, do jego odpowiedniego rozpędzenia, pozwalającego mu na osiągnięcie wszystkich rozwiązań we wnętrzu tego hiperprostokąta, można przyjąć za $Vmax$ różnicę tych wektorów ($Vmax = Xmax - Xmin$) [270]. Parametr ten jest również stosowany do odgórnego ograniczania prędkości cząstek, a przez to kontroli zasięgu możliwości zmiany ich położenia.

Początkowa pamięć cząstek jest tożsama z ich położeniem ($m_i = x_i$). Efektem jej losowego wyboru byłaby jeszcze większa zmienność ich ruchu w pierwszych iteracjach algorytmu, ale wymagałoby to wykonania dodatkowego sprawdzenia, który z tych dwóch punktów powinien być przez każdą z nich zapamiętany.

2.4.2. Aktualizacja

Procedura optymalizacji kryterium f przy pomocy algorytmu PSO polega na aktualizacji właściwości cząstek przez maksymalnie t iteracji w taki sposób, aby zbiegły się w pobliżu rozwiązania uznanego przez nie za minimum globalne.

W pierwszej kolejności zmianie podlega prędkość cząstek, następnie położenie, a na końcu podejmowana jest decyzja o ewentualnym zapamiętaniu tego położenia.

Aktualizacja prędkości

Nowy wektor i-tej cząstki jest wyznaczany w następujący sposób:

$$v_i \leftarrow \phi_v v_i + \phi_m \left(m_i - x_i\right) \odot r_{m,i} + \phi_l \left(l_i - x_i\right) \odot r_{l,i} \tag{2.21}$$

gdzie:

ϕ_v = parametr bezwładności: $\phi_v \in \mathbb{R}$
ϕ_m = parametr świadomości: $\phi_m \in \mathbb{R}$
ϕ_l = parametr społeczności: $\phi_l \in \mathbb{R}$
l_i = wektor pamięci lidera: $l_i \in \{m_1, \ldots, m_n\}$
$r_{m,i}, r_{l,i}$ = wektory turbulencji: $r_{m,i}, r_{l,i} \in U\left(0,1\right)^d$
$\odot$ = iloczyn wektorów po współrzędnych

Nowy wektor prędkości każdej cząstki jest sumą trzech czynników. Pierwszy z nich stanowi jej obecna trajektoria, wskazująca na dotychczasowy kierunek ruchu i pozwalająca na jego częściową kontynuację. Stosowany w tym miejscu parametr bezwładności ma istotne znaczenie dla osiągnięcia zbieżności przez rój. Mianowicie, służy on do jego spowalniania, przeciwdziałając przyspieszaniu wywoływanemu przez pozostałe dwa czynniki. Aby tak się działo wartość parametru ϕ_v musi należeć do przedziału $[0, 1)$. Parametr ten nie występował w oryginalnym równaniu prędkości cząstek z 1995 roku – został do niego wprowadzony dopiero trzy lata później [271]. Jego wartość może być stała (typowo należąca do przedziału od 0,7 do 0,8), wybierana za każdym razem w sposób losowy, lub zmniejszana stopniowo, co pozwala rojowi na początkowe przeszukiwanie większego podzbioru przestrzeni rozwiązań, a następnie szybkie zbiegnięcie się w pobliżu wybranego przez nie punktu [272].

Kolejne elementy równania 2.21 modelują indywidualne oraz społeczne zachowania cząstek. Pierwszy przyciąga każdą z nich do zapamiętanego przez nią rozwiązania $(m_i - x_i)$, natomiast drugi kieruje ją ku wektorowi pamięci jej lidera $(l_i - x_i)$. Siłę tych oddziaływań określają parametry ϕ_m i ϕ_l. Domyślne, ich wartości wynoszą 2, co w połączeniu z turbulencją, powoduje, że rój może trafiać w miejsca znajdujące się w przestrzeni rozwiązań bliżej lub nawet dwukrotnie dalej niż te, które wynikałyby z jego ruchu pozbawionego wpływu zdarzeń losowych.

Przydziałem liderów zarządza funkcja topologii roju, τ. Może ona należeć do jednej z dwóch rodzin: globalnej (global best, gbest), w której wszystkie cząstki podążają za obecnie najlepiej przystosowaną, lub lokalnej (local best, lbest), w której liderzy są im przydzielani z określonych podzbiorów roju. Szczegółowe informacje na ten temat znajdują się w dalszej części rozdziału.

Po wyznaczeniu nowych wektorów prędkości roju, następuje ich ograniczenie. Również ta czynność nie znajdowała się w pierwszej publikacji algorytmu PSO, choć została do niego wprowadzona jeszcze w tym samym roku, kiedy zauważono tendencję cząstek do nadmiernego przyspieszania, uniemożliwiającego im zbiegnięcie się [273]. Do realizacji tego zadania służy wspomniany wcześniej parametr $Vmax$. Wektor ten wyznacza maksymalną dozwoloną prędkość z jaką mogą poruszać się cząstki: j-ta składowa wektora v każdej z nich musi zawierać się w przedziale $[-Vmax_j, Vmax_j]$. Jeżeli znajduje się poza nim, jest do niego ograniczana z odpowiedniej strony. Parametr $Vmax$ wskazuje więc jak daleko rój może przemieścić się w pojedynczej iteracji algorytmu. Rozwiązuje to problem jego niekontrolowanego przyspieszania, ale nie gwarantuje, że zwolni on na tyle, aby osiągnąć zbieżność. Zapewnienie tego jest zadaniem parametru ϕ_v, a dokładnie jego relacji z parametrami ϕ_m i ϕ_l.

Relacja pomiędzy parametrami ϕ_v, ϕ_m i ϕ_l była obiektem badań, które przeprowadził Clerc [274]. Zaproponował on wyeliminowanie parametru ϕ_v i zastąpienie go współczynnikiem „zaciskania", χ, zależnym od sumy wartości parametrów ϕ_m i ϕ_l:

$$v_i \leftarrow \chi\Big[v_i + \phi_m\,(m_i - x_i) \odot r_{m,i} + \phi_l\,(l_i - x_i) \odot r_{l,i}\Big] \tag{2.22}$$

gdzie:

$$\chi = \frac{2}{\left|2 - \phi - \sqrt{\phi^2 - 4\phi}\right|}$$
$$\phi = \phi_m + \phi_l$$
$$\phi > 4$$

Wpływ współczynnika χ na ruch cząstek jest taki sam jak parametru ϕ_v, z tą różnicą, że spowolnienie ich następuje tu po wyznaczeniu nowych wektorów prędkości, zamiast przed. Dzięki temu, użycie parametru $Vmax$ również przestaje być potrzebne, przy założeniu, że $\chi < 1$. Clerc i Kennedy [275] sprawdzili, że w odróżnieniu od podejścia stosującego wyłącznie parametr $Vmax$ (przy ustawieniu $\phi_v = 1$), „zaciskanie" jest faktycznie w stanie wymusić zbieżność roju.

Standardowo przyjmuje się $\phi_m = \phi_l = 2{,}05$, co daje $\chi \approx 0{,}729$. Warto jednak zauważyć, że ten sam efekt można osiągnąć przy pomocy równania 2.21: jeżeli ϕ_v ma być równe χ, to musi zachodzić $\phi_m = \phi_l = 2{,}05 \cdot 0{,}729 \approx 1{,}494$.

Aktualizacja położenia

Następnym krokiem aktualizacji roju jest zmiana położenia każdej cząstki, polegająca na dodaniu do niego jej nowego wektora prędkości:

$$x_i \leftarrow x_i + v_i \tag{2.23}$$

Położenie cząstek nie podlega ograniczeniom – wszystkie elementy sterujące zachowaniem roju odnoszą się wyłącznie do ich prędkości. Nawet jeśli któraś cząstka opuści podzbiór przestrzeni rozwiązań, w którym przebywają obecnie pozostałe, zostanie do niego zawrócona dzięki mechanizmom pamięci i liderów. Do kontroli zasięgu jej ucieczki służą parametry ϕ_v i $Vmax$. Natomiast gdy trafi ona poza tym obszarem na punkt, dzięki któremu stanie się liderem, wówczas rój podąży za nią w odpowiednim czasie. Oznacza to, że jeżeli nie ma takiej potrzeby, modyfikacje algorytmu PSO powinny skupiać się na zmianach w równaniu prędkości cząstek, pozostawiając domyślny sposób aktualizacji ich położenia.

Aktualizacja pamięci

Ostatnim krokiem aktualizacji roju jest podjęcie przez każdą cząstkę decyzji o zapamiętaniu swojego nowego położenia. Dzieje się tak wtedy, gdy wartość kryterium f jest w nim niższa od jej wartości w położeniu zapamiętanym poprzednio:

$$m_i \leftarrow \begin{cases} x_i & \text{jeżeli } f(x_i) < f(m_i) \\ m_i & \text{jeżeli } f(x_i) \geq f(m_i) \end{cases} \tag{2.24}$$

Mechanizm pamięci powoduje, że wynik zwracany przez algorytm PSO w każdej iteracji zawsze znajduje się w zbiorze $\{m_1, \ldots, m_n\}$. Inną konsekwencją stosowania równania 2.24 jest gwarancja tego, że po zakończeniu optymalizacji, ów wynik będzie nie gorszy od rozwiązania, w którym się ona rozpoczęła, przy założeniu, że kryterium f było stacjonarne, a liczba cząstek się nie zmniejszyła.

2.4.3. Topologie roju

Topologia roju w algorytmie PSO modeluje zachowania społeczne cząstek. Jest to funkcja $\tau : \mathcal{P} \to \mathcal{P}$, przydzielająca każdej cząstce lidera, czyli cząstkę prowadzącą ją w danej iteracji w kierunku niższych wartości kryterium f.

Schematy topologii są przedstawiane w postaci grafów nieskierowanych, których wierzchołki reprezentują cząstki, a krawędzie – możliwość wymiany informacji między nimi. Ich położenia nie są brane pod uwagę. Topologie mogą należeć do jednej z dwóch rodzin: globalnej (global best, gbest) lub lokalnej (local best, lbest).

W domyślnie stosowanej w klasycznym algorytmie PSO rodzinie topologii gbest, cały rój podąża za tą cząstką, która odnalazła rozwiązanie o najniższej wartości kryterium f, co oznacza, że jej schematem jest graf pełny K^n, widoczny na rysunku 2.8a. Podejście to prowadzi do możliwie najszybszej zbieżności algorytmu, przy jednocześnie zwiększonej szansie na jego utknięcie w minimum lokalnym.

Topologia grafu pełnego jest jedynym przedstawicielem rodziny gbest. Jej zaletą jest wysoka precyzja, natomiast wadą – możliwie niska dokładność. Jej przeciwieństwo stanowią topologie lbest, nastawione na opóźnianie zbieżności roju, polegające na ograniczaniu możliwości wymiany informacji pomiędzy cząstkami.

Schematem pierwszej opracowanej topologii należącej do rodziny lbest jest graf pierścienia C^n [273], widoczny na rysunku 2.8b. Zgodnie z nim, że i-ta cząstka ma dostęp wyłącznie do informacji posiadanej przez siebie oraz jej najbliższych sąsiadów, czyli cząstek o indeksach $(i-1) \bmod n$ i $(i+1) \bmod n$.

Stosowanie topologii lbest powoduje efekt przeciwny do topologii gbest: możliwe zwiększenie dokładności procedury optymalizacji kosztem jej precyzji. Choć graf pierścienia jest czasami utożsamiany z tą rodziną, zaliczają się do niej również inne topologie. Poniżej wymienione jest kilka ich popularnych przykładów [276]:

1. topologia gwiazdy, w której całość informacji przepływa przez centralną cząstkę o stałym indeksie (rysunek 2.8c),
2. topologia von Neumanna, rozmieszczająca cząstki na dwuwymiarowej siatce i łącząca je z ich sąsiadami w pionie oraz w poziomie (rysunek 2.8d),
3. topologia zrównoważonego drzewa binarnego, w której cząstki układane są hierarchii, choć nadal pozostają sobie równoważne,
4. topologia izolowanych klik, dzieląca rój na rozłączne, a przez to niemające na siebie wpływu podzbiory,
5. topologia grafu pustego, osiągana poprzez ustawienie parametru ϕ_l na 0.

Kennedy i Mendes [277] sprawdzili jak wybór topologii roju wpływa na możliwości poszukiwania przez algorytm PSO minimów globalnych klasycznych funkcji testowych. Zgodnie z przyjętymi w tych badaniach kryteriami oceny, najsłabsze okazały się topologie grafu pełnego oraz gwiazdy, natomiast topologia pierścienia została sklasyfikowany niedaleko od nich ze względu na zbyt powolne osiąganie zbieżności przez korzystający z niej rój. Bardzo dobre wyniki zaobserwowano za to dla topologii von Neumanna, łączącej w sobie zalety obydwu rodzin, pozwalających jej na zachowanie równowagi pomiędzy fazami przeszukiwania i zbieżności. Do podobnych wniosków doszli Reyes Medina i współpracownicy [278], dodatkowo argumentując, że słabością topologii gwiazdy są niewielkie możliwości wymiany informacji pomiędzy cząstkami, wynikające z tego, że wszystkie z nich muszą podążać zawsze za tym samym liderem, co powoduje negatywnie na nie działającą synchronizację ich ruchu.

Wszystkie wymienione tu topologie pozwalają na to, aby liderem każdej cząstki mogła być również ona sama. Powoduje to dwukrotne zwiększenie siły przyciągania do jej wektora pamięci, przez co może dotrzeć do rozwiązań położonych dwukrotnie dalej za nim, lub pozostać w miejscu, jeżeli długość jej wektora prędkości wynosi 0 oraz $x_i = m_i$. Ponieważ cząstka, której dotyczy druga z tych sytuacji zostaje chwilowo odcięta od reszty roju, istnieje podejście alternatywne, zmuszające ją do podążania za następnym w kolejności sąsiadem wskazanym przez stosowaną topologię. Kennedy i Mendes [277] sprawdzili, że w niektórych przypadkach może przynieść to pozytywne efekty (graf pierścienia), a w innych – zupełnie przeciwne (graf siatki).

2.4.4. Warunki STOP

Podstawowym warunkiem STOP algorytmu PSO jest wykonanie założonej liczby aktualizacji roju, t. Gwarantuje on, że niezależnie od tego, co wydarzy się w trakcie optymalizacji, przerwanie całej procedury nastąpi najpóźniej we wskazanym momencie. Warunek ten pozwala także na oszacowywanie maksymalnej liczby obliczeń wartości kryterium f jaka może być w tym czasie wykonana, lub liczby rozwiązań jakie mogą być sprawdzone przez cząstki.

Oprócz warunku liczby iteracji mogą być również stosowane inne, dodatkowe warunki STOP, których zadaniem jest wcześniejsze zatrzymanie algorytmu w chwili stwierdzenia osiągnięcia przez niego rozwiązania lub stanu roju o satysfakcjonujących właściwościach [279]. Należą do nich między innymi:

1. brak zmiany położenia bieżącego wyniku w ciągu ostatnich ω iteracji,
2. spadek różnicy pomiędzy wartościami kryterium f w jego faktycznym minimum globalnym i w bieżącym wyniku poniżej progu ϵ,
3. spadek odległości pomiędzy położeniami faktycznego minimum globalnego kryterium f i bieżącego wyniku poniżej progu δ,
4. spadek rozmiaru roju poniżej progu $Rmin$, trwający przez ω iteracji,
5. spadek prędkości roju poniżej progu $Vmin$ trwający przez ω iteracji.

Warunek 1 przerywa optymalizację w chwili, gdy cząstki nie mogą odnaleźć kolejnego przybliżenia minimum globalnego kryterium f w ciągu określonej liczby iteracji. Dobór parametru ω nie jest łatwy, ponieważ algorytm PSO, tak jak inne algorytmy metaheurystyczne, nie gwarantuje, że znajdzie jakiekolwiek rozwiązanie lepsze od tego, w którym rozpoczął swoje działanie.

Warunek 2 może być stosowany wtedy, gdy znana jest minimalna wartość kryterium f. Uznaje się, że rój wystarczająco się do niej zbliżył wtedy, gdy wartość bezwzględna różnicy tej wartości i wartości odpowiadającej bieżącemu wynikowi zwracanemu przez algorytm jest niższa od ustalonego progu ϵ.

Warunek 3 może być stosowany wtedy, gdy znane jest położenie minimum globalnego kryterium f. Uznaje się, że rój wystarczająco się do niej zbliżył wtedy, gdy wartości bezwzględne wszystkich składowych wektora różnicy tego położenia i położenia bieżącego wyniku zwracanego przez algorytm są niższe od odpowiadających im składowych ustalonego wektora-progu δ.

Warunek 4 dotyczy rozmiaru roju, wyrażanego jako długość boków opisanego na nim hiperprostokąta, lub jego promień, czyli wektor średnich albo median odległości wektorów pamięci cząstek od bieżącego wyniku zwracanego przez algorytm w każdym wymiarze. Obliczony w ten sposób wektor jest porównywany z wektorem-progiem $Rmin$ identycznie jak w warunku 3. Parametr liczby iteracji ω pełni tu rolę wspomagającą. Jego zadanie polega na zapewnieniu, że oczekiwany stan algorytmu nie jest chwilowy, ale utrzymuje się przez odpowiednio długi czas.

Piąty spośród wymienionych tutaj warunków STOP jest niemal taki sam jak poprzedni. Różnica między nimi polega na tym, że bierze on pod uwagę prędkość cząstek zamiast ich wektorów pamięci. Spadek tej prędkości poniżej progu określonego przez wektor $Vmin$ oznacza, że rój zaczyna zwalniać i jeżeli nie przyspieszy w ciągu ω iteracji, to zapewne nie będzie się już oddalać od swojej obecnej lokalizacji. Autor rozprawy uważa, że w porównaniu z pozostałymi czterema, warunek ten jest uniwersalny, gdyż dotyczy cechy roju niezwiązanej z położeniami cząstek i rozwiązywanym problemem optymalizacyjnym. Położenie minimum globalnego wraz z odpowiadającą mu wartością kryterium f, wymagane przez warunki 2 i 3, są zazwyczaj nieznane, jak również nie można zagwarantować, że cząstki zbiegną się w jednym punkcie, czego oczekuje warunek 4. Spadek prędkości jest natomiast sygnałem, że dalsza optymalizacja nie będzie skutkować istotnymi zmianami w zwracanych przez algorytm wynikach.

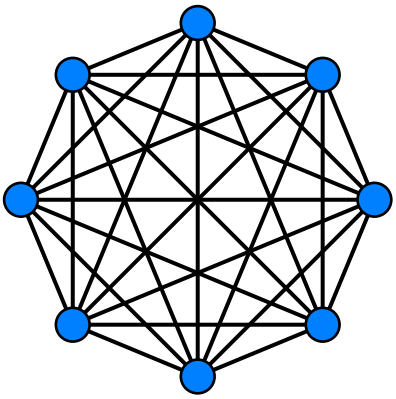
Uniwersalna przydatność warunku 5 nie oznacza jednak, że pozostałe cztery są bezużyteczne, lub mogą być przez niego zastąpione. Według Autor rozprawy, stanowią one raczej uzupełnienie tego warunku, dostarczając dodatkowych informacji uzyskanych w trakcie lub po zakończeniu działania algorytmu:

- warunek numer 1 pozwala na sprawdzenie czy rój nie utknął w zbiorze rozwiązań niedopuszczalnych,

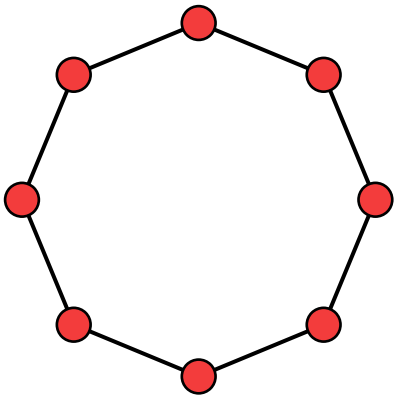
- warunki 2 i 3 znajdują zastosowanie w wymiernej prezentacji i porównywaniu uzyskanych wyników z wynikami innych algorytmów optymalizacyjnych,

- warunek czwarty umożliwia ocenę poprawności doboru parametrów ze zbioru Φ, rozróżniając pomiędzy rojami skupionymi wokół jednego rozwiązania oraz tymi, których cząstki pozostały w zbyt dużej odległości od siebie, pomimo wystarczającej do osiągnięcia tego stanu liczby iteracji.

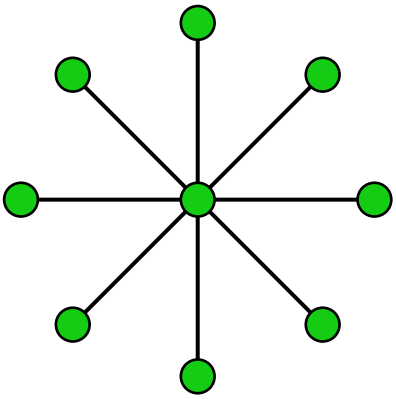
Na koniec pozostaje tylko kwestia doboru wartości parametrów δ, $Rmin$ i $Vmin$.



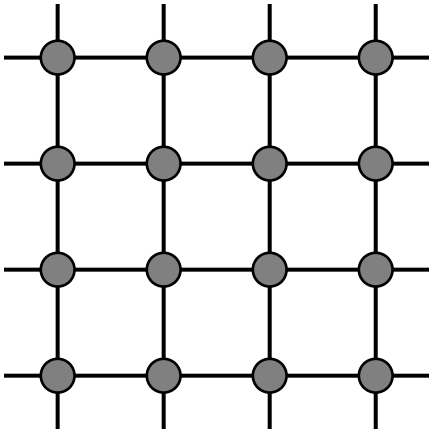
(a) Topologia grafu pełnego.



(b) Topologia pierścienia.



(c) Topologia gwiazdy.



(d) Topologia von Neumanna.

Rysunek 2.8: Wizualizacje wybranych topologii roju w algorytmie PSO. Wierzchołki grafów reprezentują cząstki, natomiast krawędzie wskazują na możliwość wymiany informacji pomiędzy nimi na temat niskich wartości optymalizowanego kryterium.

Najprostszy sposób ustalenia wartości parametrów warunków STOP polega na powiązaniu ich z ustalonymi wcześniej wartościami, na przykład elementami zbioru Φ. Ponieważ zarówno δ, jak i $Rmin$ oraz $Vmin$ są wektorami, najbardziej odpowiednim kandydatem jest $Vmax$ – wystarczy tylko określić jego akceptowalny ułamek. Na przykład, pętla aktualizacji roju zostaje przerwana wtedy, gdy średnia prędkość cząstek spadnie poniżej 10% jej maksymalnej wartości we wszystkich wymiarach przestrzeni rozwiązań. Podobna relacja może wiązać parametr ω z t.

2.4.5. Modyfikacje

Ze względu na prostotę konstrukcji i łatwość implementacji, a także skuteczność działania oraz liniową złożoność obliczeniową, algorytm PSO został wielokrotnie zmodyfikowany w celu rozszerzenia jego funkcjonalności na inne dziedziny optymalizacji niż globalna [280, 281]. Ponieważ nie jest możliwe wymienienie wszystkich z tych modyfikacji, poniżej przedstawione są wybrane podejścia, które w uznaniu Autora rozprawy dobrze prezentują sposoby rozwiązywania problemów z rozdziału 1.3.5.

Optymalizacja kombinatoryczna

Najprostszy sposób użycia algorytmu PSO w sytuacji, gdy przestrzeń rozwiązań jest dyskretna, ale nie występują relacje pomiędzy jej wymiarami, polega na zaokrąglaniu bieżących i zapamiętanych położeń cząstek pod koniec każdej iteracji. Składowe Laskari, Parsopoulos i Vrahatis [282] wykazali, że nie wpływa to negatywnie na zdolność roju do minimalizacji wybranych funkcji testowych.

Zaokrąglanie do liczb całkowitych jest jednak niewystarczające do rozwiązywania problemów kombinatorycznych, w których istnieją relacje pomiędzy optymalizowanymi zmiennymi, lub mogą one przyjmować wartości wyłącznie ze zbioru $\{0, 1\}$.

Pierwszą modyfikację algorytmu PSO umożliwiającą mu poruszania się w przestrzeni binarnej (BPSO) zaproponowali Kennedy i Eberhart [283]. Składowe wektorów prędkości cząstek są w niej traktowane jako wartości prawdopodobieństwa: wartość x_{ij} staje się równa 1 jeżeli liczba wybrana z rozkładu jednorodnego $U(0, 1)$ jest niższa od wartości sigmoidy $S(v_{ij})$. Modyfikacja ta nie zmienia samego równania prędkości, natomiast sugerowana w niej wartość parametru $Vmax$ wynosi w każdym wymiarze 6. Dzięki temu, sigmoida może przyjmować wartości z przedziału $[0{,}0025, 0{,}9975]$, co oznacza, że prawdopodobieństwo mutacji jest zawsze dodatnie.

Problem w tym, ze sigmoida powoduje, że gdy cząstki nabierają prędkości, osłabia się ich zdolność do przeszukiwania przestrzeni rozwiązań i odwrotnie – dla prędkości bliskich 0, prawdopodobieństwo zmiany ich położenia wynosi 50%, co odpowiada błądzeniu losowemu. Problemem tym zajęli się Bansal i Deep [284], którzy zaproponowali linową funkcję S, zależną dodatkowo od położenia cząstek.

Jeżeli problem optymalizacyjny polega na poszukiwaniu permutacji pewnego zbioru trzeba na nowo zdefiniować prędkość i położenie cząstek oraz sposób w jaki pierwsza z nich powoduje zamianę składowych wektorów drugiej. Kwestie związane z realizacją tych czynności, umożliwiającą rozwiązanie przykładowego problemu komiwojażera są szczegółowo omówione pracy, którą opublikował Clerc [285].

Optymalizacja wielomodalna

Jeżeli optymalizowane kryterium posiada kilka minimów globalnych lub lokalnych, które są do nich podobne w przestrzeni wartości, klasyczny algorytm PSO zbiegnie się w jednym z tych rozwiązań (nie wiadomo w którym), lub będzie rzucać się pomiędzy nimi, nie mogąc podjąć decyzji o wybraniu jednego z nich w ciągu określonej liczby iteracji. Niezależnie od tego, co się wydarzy, nie jest on w stanie zwrócić wyniku zgodnego za założenia optymalizacji wielomodalnej.

Umożliwienie algorytmowi PSO odnajdywania wszystkich lub określonej liczby minimów staje się możliwe po zastosowaniu wielu rojów. Parsopoulos i Vrahatis [286, 287] zaproponowali podejście, w którym w chwili wykrycia zbiegania się roju w którym z rozwiązań, najbliższa temu punktowi cząstka jest izolowana, a pozostałe od niej odsuwane za pomocą algorytmu rozciągania krajobrazu wartości kryterium [288], zapobiegającemu ich ponownemu zbiegnięciu w jej pobliżu. Wokół tej odizolowanej cząstki tworzony jest następnie rój, dokonujący lokalnej optymalizacji. W roju głównym zastępuje ją natomiast inna cząstka. Kontrola liczby minimów do zwrócenia jest możliwa poprzez ustalanie liczby rojów pochodnych.

Inne podejście do tematu optymalizacji wielomodalnej polega na równoczesnym uruchomieniu stałej liczby rojów. Aby każdy z nich mógł skupić się na innym rozwiązaniu, niezbędne jest wprowadzenie mechanizmu odpychania ich od siebie [289]. Cząstki, które znajdą się zbyt blisko cząstek z innego roju są odpychane, podobnie jak atomy przez oddziaływania van der Waalsa, co zapobiega zbiegnięciu się ich w rozwiązaniu, którym zajmuje się ten rój.

Podejścia wielorojowe są przykładem sytuacji, w której algorytm PSO może być efektywnie wykonany równolegle [290]. Każdy rój, a nawet pojedyncze cząstki mogą być aktualizowany osobno. Synchronizacja pomiędzy nimi jest potrzebna wtedy gdy dochodzi do interakcji. Algorytm PSO ma niską złożoność obliczeniową, dlatego zysk z jego wykonania równoległego wynika przede wszystkim z możliwości równoczesnego obliczania wartości kryterium optymalizacyjnego dla wielu rozwiązań.

Optymalizacja dynamiczna

Algorytm PSO może być łatwo przystosowany do wykonywania optymalizacji dynamicznej. Ponieważ nie przechowuje wartości optymalizowanego kryterium, nie występuje w nim efekt zdezaktualizowanej pamięci. Pozostają więc do rozwiązania wyłącznie kwestie wykrywania zmian w kształcie tego kryterium oraz dostosowywania się do nich, czyli śledzenia poruszających się minimów [291, 292].

Do wykrycia zmiany krajobrazu wartości optymalizowanego kryterium wystarczy ponowne obliczenie jego wartości w rozwiązaniu stanowiącym bieżący wynik zwracany przez algorytm i porównaniu go z wartością zapamiętaną poprzednio [293]. Można również sprawdzić wszystkie wektory pamięci cząstek. Zdolność algorytmu PSO do poszukiwania minimów globalnych jest bezpośrednio związana z prędkością cząstek. Im szybciej cząstki się poruszają, tym większy podzbiór przestrzeni rozwiązań może być przez nie przeszukiwany. W związku z tym, w chwili, gdy minimum globalne, w którym się zbiegały zostało przemieszczone w pobliże jego dotychczasowego położenia, cząstki muszą być odpowiednio rozpędzone, aby móc za nim podążyć. Jeżeli rój zdążył się wcześniej zbiec, należy go rozpędzić w losowych kierunkach, co pozwoli mu na ucieczkę z powstałego minimum lokalnego.

Dodatkowe obliczanie wartości optymalizowanego kryterium w każdej iteracji, w szczególności dla wszystkich cząstek, powoduje spadek efektywności algorytmu, nie gwarantując równocześnie, że zmiana kształtu krajobrazu wartości optymalizowanego kryterium zostanie zarejestrowana. Blackwell i współpracownicy [291, 292] zaproponowali podejście alternatywne, w którym występują dwa roje cząstek. Pierwszy z nich zbiega się standardowo w wybranym minimum globalnym, natomiast drugi jest od niego odpychany, podobnie jak oddziaływania elektrostatyczne odpychają od siebie ładunki jednoimienne. Ponieważ obydwa roje są połączone wspólną topologią, powoduje to upodobnienie się przez nie do modelu atomu. Dzięki temu, drugi z tych rojów porusza się w pewnej odległości od pierwszego (nie zbiega się), co pozwala im na śledzenie przemieszczającego się minimum globalnego.

Optymalizacja z ograniczeniami

Najprostszy, „zerowy” sposób obsługi funkcji ograniczeń polega na zastosowaniu wag, pozwalających na połączenie tych funkcji z optymalizowanym kryterium we wspólnym równaniu. Dzięki temu, zagadnienie optymalizacji z ograniczeniami sprowadza się do zagadnienia optymalizacji globalnej. Wagi te są jednak zazwyczaj nieznane, co wymusza potrzebę stosowania innych podejść [294].

Pozostałe podejścia do tematu optymalizacji z ograniczeniami korzystają z funkcji kary. Funkcje te zwracają odpowiednio wysokie wartości na podstawie tego, jak bardzo rozwiązania są niedopuszczalne. Im wyższa jest wartość funkcji ograniczeń, tym wyższa jest wartość funkcji kary. Dla rozwiązań dopuszczalnych wynosi ona 0. W zależności od sposobu użycia funkcji kary, podejścia z nich korzystające mogą być podzielone na dwie kategorie [295].

Podejścia należące do pierwszej kategorii sposobu użycia funkcji ograniczeń dodają zwracane przez nie wartości do wartości optymalizowanego kryterium [296–298]. Powoduje to preferowanie przez algorytm rozwiązań dopuszczalnych przed niedopuszczalnymi. W ten sposób zagadnienie optymalizacji z ograniczeniami zostaje ponownie sprowadzone do zagadnienia optymalizacji globalnej. Problem związany z takim stosowaniem funkcji kary wynika z potrzeby ich dopasowywania do konkretnego kryterium optymalizacyjnego. Idealnie, wartości tych funkcji powinny być na tyle wysokie, aby przeciwdziałać zbieganiu się algorytmu w rozwiązaniach niedopuszczalnych, ale też na za wysokie, aby zupełnie nie zniechęcać go do przeszukiwania zawierających je podzbiorów przestrzeni rozwiązań. W przeciwnym razie, może mieć on trudność w odnajdywaniu minimów znajdujących się w ich pobliżu oraz przemieszczaniu się pomiędzy rozłącznymi podzbiorami rozwiązań dopuszczalnych [299, 300].

Podejścia należące do drugiej kategorii obsługują funkcje ograniczeń bez łączenia ich z optymalizowanym kryterium. Jednym tych sposobów jest przekształcenie zagadnienia optymalizacji z ograniczeniami w zagadnienie optymalizacji wielokryterialnej [301]. Funkcje ograniczeń są wówczas traktowane jako osobne kryteria. Ich wartości mogą być także sumowane, łącząc się w pojedynczym kryterium. Podejścia te różnią się od typowej optymalizacji wielokryterialnej tym, że nie jest w nich poszukiwany cały zbiór Pareto, ale jego pojedynczy element. Ich zaletą jest brak potrzeby dopasowywania funkcji ograniczeń do kryterium optymalizacyjnego.

Innym podejściem, znajdującym się pomiędzy powyższymi kategoriami jest binarna strategia turniejowa Deba [302]. Z jednej strony, nie łączy ona funkcji ograniczeń z optymalizowanym kryterium, ale z drugiej nie dąży do minimalizacji ich wartości w sensie optymalizacji wielokryterialnej. Stosuje się ją w sytuacji, gdy należy dokonać wyboru pomiędzy dwoma rozwiązaniami: x_1 i x_2, na przykład podczas podejmowania przez cząstki decyzji o zapamiętaniu swojego nowego położenia. Wybór pomiędzy nimi podlega poniższym zasadom:

1. jeżeli x_1 i x_2 są dopuszczalne, wybierane jest to z tych rozwiązań, w którym kryterium optymalizacyjne osiąga niższą wartość,

2. jeżeli x_1 jest dopuszczalne a x_2 – nie, wybierane jest x_1,

3. jeżeli x_2 jest dopuszczalne a x_1 – nie, wybierane jest x_2,

4. jeżeli x_1 i x_2 są niedopuszczalne, wybierane jest to z tych rozwiązań, w którym suma wartości funkcji ograniczeń jest niższa.

W strategii Deba rozwiązania dopuszczalne mają pierwszeństwo przed niedopuszczalnymi. Za pomysłodawców tej idei uznaje się Richardsona i współpracowników [303], a jako pierwsi zastosowali ją w praktyce Powell i Skolnick [304]. Deb zaproponował ich uniwersalną, pozbawioną parametrów modyfikację, w której algorytm optymalizacyjny jest kierowany w stronę rozwiązań dopuszczalnych przez wszystkie funkcje ograniczeń, bez potrzeby posiadania wiedzy na temat relacji pomiędzy tymi funkcjami a optymalizowanym kryterium. Użycie tego podejścia w algorytmie PSO przedstawili Fuentes Cabrera i Coello Coello [305]. Zaletą strategii Deba jest to, że pozwala obsługiwać dowolną liczbę dowolnych funkcji ograniczeń bez potrzeby wprowadzania istotnych zmian w stosujących ją algorytmach. Oprócz tego, jeżeli dane rozwiązanie jest niedopuszczalne, wówczas wartość optymalizowanego kryterium nie musi być dla niego obliczana, co miało szczególne znaczenie w niniejszej rozprawie.

Optymalizacja wielokryterialna

Pierwsza modyfikacja algorytmu PSO do zastosowań w optymalizacji wielokryterialnej (MOPSO) została opublikowana w 2002 roku przez Coello Coello i współpracowników [306]. Inne wczesne modyfikacje powstały poprzez adaptacje rozwiązań pochodzących z algorytmów ewolucyjnych: VEGA [307] → VEPSO [308], NSGA [309] → NSPSO [310] i micro-GA [311] → micro-MOPSO [312].

Trzy zadania, jakie stoją przed algorytmami optymalizacji wielokryterialnej zostały przedstawione w rozdziale 1.3.5. Realizacja tych zadań przy pomocy roju cząstek wymaga zastanowienia się nad sposobem wyboru liderów promujących rozwiązania globalnie niezdominowane, przechowywania informacji o tych rozwiązaniach i zapobiegania zbieżności w jednym z nich [313]. Reyes-Sierra i Coello Coello [313] zaproponowali również następującą taksonomię metod realizujących te zadania:

- podejścia agregujące, łączące funkcje wielokryterialne w jednokryterialne,
- podejścia leksykograficzne, wykonujące optymalizację globalną kolejnych kryteriów ustawionych przez użytkownika w kolejce priorytetowej,
- podejścia wielorojowe, korzystające z grupy kilku rojów, z których każdy optymalizuje inne kryterium (algorytm MOSF należy do tej kategorii),
- podejścia oparte na analizie frontu Pareto, najbardziej popularne, wybierające liderów cząstek na podstawie jego właściwości.

Wyróżnia się również podejścia łączące możliwości powyższych metod, a także hybrydowe, wprowadzające do PSO mechanizmy algorytmów ewolucyjnych [314].

2.5. Pozostałe algorytmy

Poniżej znajduje się krótki opis najważniejszych algorytmów, niezwiązanych bezpośrednio z tematem rozprawy, które zostały zastosowane przez jej Autora w celu otrzymania wyników przedstawionych w następnym rozdziale. Są to: poszukiwanie najbliższych sąsiadów przy pomocy struktury drzewa k-d, analiza skupień k-średnich i hierarchiczna, analiza składowych głównych i algorytm Kabscha, transformacja Householdera oraz generator MPB (moving peaks benchmark).

2.5.1. Drzewo k-d

Drzewo k-d (k-d tree), opracowane przez Jona Bentley-a, jest binarnym drzewem służącym do podziału przestrzeni $\mathbb{R}^k$ [315]. Korzystają z niego algorytmy zajmujące się poszukiwaniem najbliższych sąsiadów, czyli efektywnym pod względem czasu wykonania oraz zapotrzebowania na pamięć wskazywaniem m wektorów należących do n-elementowego zbioru, które według stosowanej metryki, znajdują się najbliżej badanego wektora v.

Konstrukcja drzewa k-d jest następująca: każdemu jego wierzchołkowi przypisuje się komórkę – k-wymiarowy hiperprostokąt o bokach równoległych do osi układu współrzędnych, wszystkie punkty, które się w nim zawierają, a także hiperpłaszczyznę jego podziału. Korzeń drzewa obejmuje cały wejściowy zbiór danych, natomiast liśćmi są te wierzchołki, w których komórkach leży nie więcej punktów niż wynosi wartość parametru zwanego rozmiarem kubełka (bucket size). Służy on do wypracowywania kompromisu pomiędzy czasem spędzanym na lokalnym, wyczerpującym poszukiwaniu najbliższych sąsiadów, a czasem potrzebnym na poruszanie się po strukturze drzewa. Kolejne jego poziomy powstają w wyniku podziału komórek przez hiperpłaszczyzny. Friedman i współpracownicy [316] przyjęli, że normalną każdej z nich będzie ta oś układu współrzędnych, wzdłuż której rozstęp punktów (różnica pomiędzy najwyższymi i najniższymi współrzędnymi) jest największy, a wektorem translacji – mediana tego rozkładu. Dane znajdujące się w wybranym wymiarze poniżej hiperpłaszczyzny podziału trafiają do lewego poddrzewa, a powyżej – do prawego. Punkt, przez który przechodzi ta hiperpłaszczyzna może być umieszczony w dowolnym poddrzewie. Podejście to, nazywane podziałem standardowym (standard split), skutkuje utworzeniem drzewa o wysokości $O(\log n)$ i zajmującego $O(k\,n)$ miejsca w pamięci. Oczekiwana złożoność obliczeniowa tej procedury jest taka sama jak posortowania danych we wszystkich wymiarach, czyli $O(k\,n\log n)$ [317].

Poszukiwanie m najbliższych sąsiadów wektora v rozpoczyna się od korzenia drzewa k-d. Do ich przechowywania służy dynamiczna lista, której elementy są zawsze utrzymywane w porządku rosnących odległości od v. W chwili natrafienia przez algorytm na wierzchołek nieterminalny, jest on uruchamiany rekursywnie dla tego poddrzewa, po którego stronie hiperpłaszczyzny podziału danej komórki znajduje się v. Dotarcie do liścia powoduje przejście w tryb wyszukiwania wyczerpującego, polegającego na obliczeniu odległości od v do wszystkich elementów komórki i odpowiedniej aktualizacji listy najbliższych sąsiadów. Po zwróceniu kontroli przez wywołanie rekursywne, następuje sprawdzenie, czy należy w taki sam sposób zbadać również drugie poddrzewo. Dzieje się tak wtedy, gdy liczba zapamiętanych do tej pory najbliższych sąsiadów jest mniejsza niż m, lub gdy hipersfera o środku w v i promieniu równym odległości do najbliższego sąsiada przecina się z bieżącą hiperpłaszczyzną podziału. Z badań jakie przeprowadzili Friedman i współpracownicy wynika, że najbardziej wydajne pod względem czasu działania tego algorytmu są wartości parametru rozmiaru kubełka z przedziału pomiędzy 4 a 32 (optymalnie 10), natomiast jego oczekiwana złożoność obliczeniowa, zależna od liczby danych wejściowych, jest proporcjonalna do $\log n$ [316]. Występuje tu jednak ukryty czynnik rosnący wykładniczo wraz z liczbą wymiarów przestrzeni, dlatego przyjmuje się, że zysk wynikający ze stosowania drzewa k-d względem algorytmu wyczerpującego będzie znaczący tylko wtedy, gdy n będzie większe od 2^k [318].

Wadą standardowego podziału przestrzeni jest jego tendencja do tworzenia długich i wąskich komórek. Powoduje to przecinanie przez hipersferę algorytmu poszukiwania najbliższych sąsiadów wielu hiperpłaszczyzn, co niepotrzebnie zmusza go do odwiedzania dużej liczby poddrzew. Podejście alternatywne, zwane midpoint split, ustawia te hiperpłaszczyzny za każdym razem dokładnie w połowach najdłuższych boków komórek. W przypadku kilku równoważnych pod tym względem kandydatów, wybierany jest ten wymiar, w którym punkty mają największy rozstęp. Powstają w ten sposób komórki o maksymalnym stosunku boków $2:1$, ale jednocześnie zwiększa się ich liczba. Innym negatywnym efektem ubocznym tego podziału jest to, że część liści lub nawet całych poddrzew może okazać się pusta. W związku z tym, Maneewongvatana i Mount [319] zaproponowali jeszcze inne rozwiązanie, które nazwali sliding midpoint split. Jest ono modyfikację poprzedniego i polega na przesuwaniu hiperpłaszczyzny podziału ze środka boku komórki na pozycję najbliższego jej punktu jeżeli jedno z poddrzew okazuje się puste, co pozwala na nadanie mu od razu statusu liścia. Podejście to nie gwarantuje jednak, że wysokość drzewa będzie równa $O(\log n)$, co – jak przekonują jego Autorzy – nie ma zazwyczaj istotnego znaczenia.

Procedura poszukiwanie m najbliższych sąsiadów może być łatwo przekształcona w taki sposób, aby zwracała wszystkie punkty znajdujące się w promieniu r od wektora v [320]. Podstawowa różnica w realizacji tego zadania polega na tym, że pomija się tutaj warunek zapełnienia listy wyników. Dodatkowo, zawartość każdej komórki może być zwrócona od razu, bez potrzeby analizowania jej elementów, jeżeli zawiera się ona w całości w hipersferze wyśrodkowanej w v.

Tam, gdzie czas działania jest ważniejszy od dokładności obliczeń, może być wykonane przybliżone poszukiwanie najbliższych sąsiadów. Znajduje ono szczególne zastosowanie w przestrzeniach o wysokiej liczbie wymiarów. Sprawdzanie dalszych poddrzew podczas poruszania się w stronę korzenia następuje w nim nadal wtedy, gdy hiperpłaszczyzna podziału obecnej komórki znajduje się nie dalej od v niż wynosi odległość do jego najbliższego sąsiada, ale podzielona przez $1 + \epsilon$, gdzie $\epsilon \geq 0$ [321]. Oczekiwana złożoność obliczeniowa procedury przybliżonego poszukiwania najbliższych sąsiadów w drzewie skonstruowanym przy pomocy metody sliding midpoint split wynosi wówczas $O\left(1/\epsilon^k \log n\right)$ [322].

2.5.2. Analiza skupień

Analiza skupień jest techniką eksploracyjną, służącą do podziału (partycji) zbioru n obiektów $\mathcal{O} \equiv \{o_1 \ldots o_n\}$ zwanych obserwacjami, na k podzbiorów zwanych skupiskami $\mathcal{C}_1 \ldots \mathcal{C}_k$, gdzie $n, k \geq 1$, w oparciu o następujące cztery zasady [323]:

1. każdy obiekt należy do jakiegoś skupiska: $\mathcal{C}_1 \cup \cdots \cup \mathcal{C}_k = \mathcal{O}$,
2. wszystkie skupiska są niepuste: $\forall\, i \in \{1 \ldots k\} : \mathcal{C}_i \neq \emptyset$,
3. wszystkie skupiska są rozłączne: $\forall\, i,j \in \{1 \ldots k\}\,, i \neq j : \mathcal{C}_i \cap \mathcal{C}_j = \emptyset$,
4. obiekty do siebie podobne według przyjętych kryteriów powinny należeć do tego samego skupiska, a niepodobne – do różnych skupisk.

Analiza skupień umożliwia kategoryzację danych oraz uzyskiwanie zorganizowanej ich reprezentacji, uwidaczniającej relacje pomiędzy nimi.

Podobieństwo obiektów jest wyrażane w sposób wymierny przy pomocy miary niepodobieństwa – funkcji $\mathcal{O} \times \mathcal{O} \to \mathbb{R}$, określającej jak bardzo dwa z nich się od siebie różnią. Powinna być ona symetryczna i zwracać jakąś ustaloną wartość minimalną (zazwyczaj 0) gdy porównywane obiekty są identyczne. Jeżeli spełnia również nierówność trójkąta, staje się metryką w $\mathcal{O}$.

Najczęściej stosowanymi miarami niepodobieństwa są metryki Minkowskiego, w szczególności norma euklidesowa. Każdy obiekt jest wówczas reprezentowany przez d-wymiarowy wektor, którego kolejne składowe przechowują wartości opisujących go cech. Ponieważ cechy te mogą posiadać różne zakresy, jeżeli nie są znane ich wagi, przez przystąpieniem do analizy skupień, dokonuje się zazwyczaj wstępnej standaryzacji danych wejściowych. Czynność ta polega na podzieleniu składowych reprezentujących je wektorów przez odchylenia standardowe odpowiadających im cech, co powoduje zrównanie tych odchyleń ze sobą.

Istnieją dwa sposoby oceny jakości przypisania obiektów do skupisk: wewnętrzne (analizujące cechy wynikowej partycji) oraz zewnętrzne (porównujące wynikową partycję z partycją-wzorcem). Przedstawicielem pierwszej z tych grup jest Silhouette autorstwa Petera Rousseeuwa [324]. Algorytm ten wskazuje jak bardzo pojedynczy obiekt jest podobny do elementów swojego skupiska na tle pozostałych skupisk. Tworzy on rozkład o długości n i wartościach z przedziału $[-1, 1]$. Im więcej z nich jest bliskich 1, tym lepiej wynikowa partycja odpowiada badanemu zbiorowi obserwacji i odwrotnie. Przykładem techniki zewnętrznej jest natomiast indeks Williama Randa (Rand index, RI) [325], wyrażający podobieństwo dwóch partycji w postaci pojedynczej liczby z przedziału $[0, 1]$. Podobnie, 1 oznacza tu pełną identyczność. Statystyka ta ma jednak kilka wad: jej wartość oczekiwana dla losowych wyników nie jest stała, a także dąży do 1 wraz ze wzrostem liczby skupisk [326]. Obydwa te problemy zostały rozwiązane wraz z wprowadzeniem skorygowanego indeksu Randa (adjusted Rand index, ARI) [327], obliczanego za pomocą poniższego równania [328]:

$$\mathrm{ARI}\,(U, V) = \frac{2\,(ad - bc)}{(a + b)\,(b + d) + (a + c)\,(c + d)} \tag{2.25}$$

gdzie:

U, V = porównywane partycje tego samego zbioru obiektów
a = liczba par obiektów w tych samych skupiskach w U i V
b = liczba par obiektów w tych samych skupiskach w U i różnych w V
c = liczba par obiektów w tych samych skupiskach w V i różnych w U
d = liczba par obiektów w różnych skupiskach w U i V

Poniżej omówione są krótko są dwa popularne i powszechnie stosowane algorytmy analizy skupień: k-średnich oraz aglomeracyjny, będące przedstawicielami dwóch głównych rodzin tych metod: kombinatorycznych oraz hierarchicznych [329].

Algorytm k-średnich

Algorytm k-średnich jest kombinatoryczną metodą analizy skupień, działającą na wektorach o składowych rzeczywistych [330–332]. Przypisuje on każdą obserwację do jednego z k skupisk, którego centroid (średnia arytmetyczna elementów tego skupiska) znajduje się najbliżej niej w sensie kwadratu wartości wybranej miary niepodobieństwa, typowo normy euklidesowej.

Klasyfikacja algorytmu k-średnich do rodziny metod kombinatorycznych wynika z zasady jego działania: przenosi on obiekty pomiędzy skupiskami, poszukując takiej partycji, dla której miara błędu, równa sumie odległości od wszystkich obiektów do najbliższych im centroidów, jest najmniejsza. Tej zasadzie oraz faktowi, że liczba skupisk musi być być znana *a priori*, podejście to zawdzięcza swoją nazwę.

Ponieważ wskazanie optymalnej partycji jest problemem NP-trudnym już dla $k \geq 2$ [333], realizacja algorytmu k-średnich polega na wybraniu początkowego zbioru centroidów (zazwyczaj w sposób losowy spośród wszystkich obserwacji), a następnie kierowaniu się w stronę najbliższego minimum lokalnego miary błędu. Osiągnięcie go jest możliwe dzięki wykonaniu poniższych czynności:

1. przypisania obiektów do skupisk, których centroidy znajdują się najbliżej nich,
2. wyznaczenia nowych centroidów skupisk,
3. zakończenia procedury jeżeli wszystkie obiekty pozostały w tych samych skupiskach, w których znajdowały się poprzednio, lub wykonana została określona liczba iteracji, t; w przeciwnym przypadku – powrotu do punktu 1.

Minimalizacja miary błędu będącej sumą odległości od obserwacji do centroidów powoduje, że skupiska algorytmu k-średnich są zawsze wypukłe. Dowód zbieżności tego i podobnych algorytmów znajduje się w pracy Selima i Ismaila [334].

Atrakcyjność metody k-średnich wynika przede wszystkim z jej liniowej złożoności obliczeniowej i zapotrzebowania na pamięć ze względu na wszystkie czynniki, wynoszących odpowiednio $O(n\,k\,d\,t)$ oraz $O((n+k)d)$, a także prostoty implementacji i możliwości wykonania równoległego [335].

Do wad metody k-średnich należy natomiast zaliczyć wrażliwość na położenia odstające, zastosowanie tylko do cech, dla których pojęcie średniej ma sens, wymóg znajomości docelowej liczby skupisk, ograniczoną możliwość reprezentacji niektórych kształtów zbiorów danych, takich jak krzywe lub pierścienie, a także silną zależność od początkowych położeń centroidów [323, 329].

Problem położeń odstających może być rozwiązany dzięki niemal identycznemu algorytmowi k-medoidów (PAM) [336]. Reprezentantem każdego skupiska jest w nim medoid – obiekt znajdujący się najbliżej pozostałych elementów tego skupiska. Podejście to ma jednak dużo wyższą złożoność obliczeniową, $O\left((n-k)^2\, k\, d\, t\right)$, choć kwadratowa zależność od n może być usunięta poprzez losowe próbkowanie danych wejściowych [337]. Praca z obiektami posiadającymi cechy jakościowe jest możliwa dzięki algorytmowi k-prototypów [338], natomiast tematy estymacji liczby skupisk oraz lepszego sposobu wyboru ich początkowych centroidów zostały poruszone między innymi w metodach x-średnich [339] oraz k-średnich++ [340].

Algorytm aglomeracyjny

Algorytm aglomeracyjny jest przedstawicielem rodziny hierarchicznych metod analizy skupień. Jego sposób działania polega na tworzeniu binarnego drzewa skupisk [341]. Na początku, wszystkie obiekty są umieszczane w n liściach tego drzewa. Następnie, każde dwa skupiska znajdujące się najbliżej siebie są ze sobą łączone, aż do dotarcia do korzenia obejmującego cały zbiór danych wejściowych. Wybór wynikowej partycji polega wówczas na przekształceniu drzewa w las poprzez określenie poziomu odcięcia, powyżej którego nie następuje już łączenie skupisk ze sobą.

Miary niepodobieństwa służą w algorytmie aglomeracyjnym do porównywania ze sobą całych skupisk. Dwa najbardziej znane sposoby ich użycia są nazywane metodami najbliższych (single-link) [342] oraz najdalszych sąsiadów (complete-link) [343]. W pierwszej z nich, niepodobieństwo pary skupisk jest równe najmniejszemu niepodobieństwu pary obiektów, natomiast w drugiej – najwyższemu.

W odróżnieniu od algorytmu k-średnich, nie jest tu wymagana wiedza na temat liczby skupisk – o liczbie elementów wynikowej partycji decyduje bowiem poziom odcięcia. Mogą one być wykrywane prawidłowo nawet wtedy, gdy ich kształt odbiega od wypukłego, na przykład, gdy przypomina pierścień lub krzywą. Metoda najbliższych sąsiadów ma większe możliwości w tym względzie, choć jest bardziej czuła na szum występujący w zbiorze danych wejściowych. Metoda najdalszych sąsiadów tworzy za to bardziej zwarte skupiska, przez co ma większe znaczenie praktyczne [341].

Podobnie jak w algorytmie k-medoidów, hierarchicznej analizie skupień mogą podlegać dowolne obiekty, nie tylko te opisywane przez wymierne cechy. Największą jej wadą jest jednak podobnie wysoka złożoność obliczeniowa, wynosząca $O\left(d\, n^2 \log n\right)$, a także kwadratowe zapotrzebowanie na pamięć w przypadku korzystania ze wstępnie obliczanej macierzy niepodobieństwa.

2.5.3. Transformacja Householdera

Transformacja autorstwa Alstona Householdera jest przekształceniem liniowym dokonującym odbicia lustrzanego w przestrzeni $\mathbb{R}^d$ względem hiperpłaszczyzny o wektorze normalnej v i przechodzącej przez początek układu współrzędnych [344]. Realizującą je macierz H, o rozmiarze $d \times d$, oblicza się w następujący sposób:

$$H = I - 2vv^T \tag{2.26}$$

gdzie I jest macierzą jednostkową, również o rozmiarze $d \times d$.

Macierz H posiada następujące właściwości [345]:

1. jest symetryczna i ortogonalna: $H = H^T = H^{-1}$
2. ma ujemny wyznacznik: $\det H = -1$
3. dla wektora v z równania 2.26 zachodzi: $Hv = -v$
4. dla każdego wektora x prostopadłego do v zachodzi: $Hx = x$

Jednym z zastosowań transformacji Householdera jest anihilacja składowych wektorów, czyli przekształcanie ich w taki sposób, aby wszystkie te składowe, poza jedną, były równe 0. Stosuje się ją między innymi w algorytmie QR [346–348].

Aby uzyskać macierz H odbijającą wektor s na wektor t, należy przyjąć $v = s - t$ i dokonać jego normalizacji, dzieląc przez $\|s - t\|$. Powoduje to umieszczenie tych wektorów symetrycznie po obydwu stronach hiperpłaszczyzny odbicia, dzięki czemu zachodzą oczekiwane zależności: $Hs = t$ oraz $Ht = s$ [349].

Złożenie parzystej liczby odbić lustrzanych pomiędzy wektorami należącymi do tej samej hiperpłaszczyzny jest tożsame z obrotem pierwszego na ostatni z nich w tej hiperpłaszczyźnie [350]. W związku z tym, aby uzyskać macierz R obracającą wektor s na wektor t, należy wykonać dwie transformacje Householdera: z s na pośredni wektor u oraz z u na t. Ponieważ u ma zawierać się w hiperpłaszczyźnie wyznaczonej przez s i t, najprostszymi kandydatami są $u = -s$ i $u = -t$.

Autor rozprawy zauważa, że powyższe podejście zawodzi gdy $t = -s$, kiedy zamiast obrotu nastąpi odbicie s na t. Aby temu zaradzić, wystarczy przyjąć za u wektor z dowolnej hiperpłaszczyzny, której normalną stanowi wektor różnicy s i t. Należy pamiętać, że dla $d > 2$ wybór ten wpływa na wynik całego obrotu. Jeżeli jednak nie ma to znaczenia, można go wybrać spośród wektorów własnych macierzy odbijającej s na t, którym odpowiada wartość własna równa 1. Zgodnie z oczekiwaniami, tyle będzie również wynosić wyznacznik macierzy R [351].

2.5.4. Analiza składowych głównych

Analiza składowych głównych (principal component analysis, PCA) jest metodą czynnikową przekształcającą zbiór n punktów z przestrzeni $\mathbb{R}^d$, reprezentujących grupę obserwacji opisywanych przez d możliwie skorelowanych cech, do przestrzeni nieskorelowanych składowych głównych $\mathbb{R}^c$, gdzie $c \leq d$ [352, 353]. Czynność ta pozwala na uwidacznianie zależności między danymi oraz ich kompresję.

Składowe główne są interpretowane w sposób geometryczny jako średnice hiperelipsoidy dopasowanej do danych w taki sposób, aby najdłuższa wskazywała kierunek ich największej wariancji, a kolejne – coraz niższych [354].

Dwie cechy są nieskorelowane wtedy, gdy ich kowariancja wynosi 0. W tym przypadku warunek ten dotyczy macierzy kowariancji o rozmiarze $d \times d$. Zadaniem PCA jest dokonanie jej diagonalizacji, powodującej ułożenie składowych głównych równolegle do osi układu współrzędnych. Obrócone w ten sposób dane wejściowe zostają wyrażone za pomocą nowych, nieskorelowanych cech.

Wariancja obserwacji w każdym wymiarze nowej przestrzeni jest wprost proporcjonalna do ilości informacji dostarczanej na ich temat przez odpowiadającą mu składową główną. Dzięki tej wiedzy możliwa staje się kompresja danych. Bazuje ona na założeniu, że najwyższą wariancją charakteryzują się początkowe składowe główne, co w wyniku pominięcia pozostałych, przy niewielkiej jej stracie, pozwala na zaoszczędzenie sporej ilości miejsca. O wartości parametru c decyduje użytkownik. Może ją wskazać bezpośrednio, lub ustalić na taką, która zachowuje określony ułamek całkowitej wariancji. Po przywróceniu obserwacji do oryginalnej przestrzeni cech, artefakty kompresji prezentują się w postaci „spłaszczeń”. Na przykład, efektem usunięcia jednej z trzech składowych głównych jest rzut prostopadły danych na płaszczyznę o normalnej równoległej do tej składowej.

Składowe główne są tożsame z wektorami własnymi macierzy kowariancji obserwacji, a odpowiadające im wartości własne – z wariancjami opisujących te obserwacje cech we wskazywanych przez nie kierunkach. Przyjmując, że dane wejściowe są zapisane w macierzy X o rozmiarze $n \times d$, pierwszym krokiem PCA jest ich wyśrodkowanie, czyli odjęcie od każdej kolumny wartości jej średniej arytmetycznej [355]:

$$\tilde{X} = X - \bar{X} \tag{2.27}$$

Jeżeli zakresy wartości cech nie są ze sobą porównywalne, identycznie jak w analizie skupień, może zachodzić potrzeba zrównania ich istotności, polegającego na podzieleniu każdej kolumny macierzy $\tilde{X}$ przez jej odchylenie standardowe.

W następnym kroku obliczana jest macierz kowariancji cech C:

$$C = \frac{1}{n-1}\tilde{X}^T\tilde{X} \tag{2.28}$$

Macierz C jest symetryczna, co oznacza, że istnieje ortonormalna macierz wektorów własnych V oraz diagonalna macierz wartości własnych D, które ją diagonalizują, czyli spełniają następującą zależność:

$$C = VDV^{-1} \tag{2.29}$$

Wartość własna $D_{i,i}$ jest tożsama z wariancją danych wejściowych wzdłuż i-tego wektora własnego, odpowiadającego i-tej kolumnie macierzy V. Do rozwiązywania równania 2.29 stosuje się algorytm QR lub inne metody numeryczne [356].

Po diagonalizacji macierzy C, zadaniem użytkownika pozostaje tylko posortowanie kolumn macierzy D i V w kolejności malejącej wariancji. Rzutowanie obserwacji na składowe główne sprowadza się wówczas do prostego mnożenia macierzy $\tilde{X}$ i V:

$$\tilde{Y} = \tilde{X}V \tag{2.30}$$

Ponieważ $V^{-1} = V^T$, przywrócenie obserwacji z przestrzeni składowych głównych do przestrzeni oryginalnych cech jest równie proste:

$$\tilde{X} = \tilde{Y}V^T \tag{2.31}$$

Aby skompresować dane wejściowe należy obliczyć rozkład sumy kolejnych wartości własnych z macierzy D podzielonych przez ich sumę. Na tej podstawie można stwierdzić ile nieskorelowanych cech, c, jest potrzebne do zachowania określonego ułamka całości informacji, na przykład 90%. Wystarczy wówczas zamienić w równaniach 2.29 i 2.31 macierz V na macierz W zawierającą jej pierwsze c kolumn.

Inny, preferowany, sposób wykonywania analizy składowych głównych polega na obliczeniu rozkładu według wartości osobliwych (singular value decomposition, SVD). Macierz $\tilde{X}$ może być bowiem przedstawiona w postaci następującego iloczynu:

$$\tilde{X} = UDV^T \tag{2.32}$$

gdzie kolumny z U są wektorami własnymi $\tilde{X}\tilde{X}^T$, V – $\tilde{X}^T\tilde{X}$, a podniesione do kwadratu i podzielone przez $n-1$ elementy diagonali D – wartościami własnymi $\tilde{X}$.

Zaletą analizy składowych głównych w oparciu o SVD jest brak potrzeby wykonywania mnożenia $\bar{X}^T\bar{X}$, co ma znaczenie w przypadku dużej liczby cech. Powszechnie stosowanie podejście do rozwiązywania równania 2.32, będące modyfikacją algorytmu QR, zostało zaproponowane przez Goluba i Reinscha [357] jako rozwinięcie wcześniejszej pracy Goluba i Kahana [358].

Należy pamiętać, że macierze U i V mogą w przypadku niektórych danych wejściowych okazać się złożeniami nie tylko obrotów, ale również odbić lustrzanych, co może stanowić problem dla aplikacji oczekujących zachowania ich oryginalnej orientacji, na przykład poniższego algorytmu Kabscha. Istnieje jednak łatwy sposób na rozwiązanie tego problemu, również przedstawiony poniżej.

2.5.5. Algorytm Kabscha

Algorytm autorstwa Wolfganga Kabscha służy do wyznaczania przekształceń liniowych minimalizujących pierwiastek średniej kwadratów różnicy (root mean square deviation, RMSD) dwóch równolicznych zbiorów wektorów: $\mathcal{X} \equiv \{x_1, \ldots, x_n\}$ oraz $\mathcal{Y} \equiv \{y_1, \ldots, y_n\}$, gdzie dla każdego $i \in \{1, \ldots, n\}$ $x_i, y_i \in \mathbb{R}^d$ [359, 360].

Wartość RMSD stosuje się w bioinformatyce do oceny podobieństwa struktur trzeciorzędowych białek [361]. W celu zmniejszenia wpływu położeń łańcuchów bocznych, jego dane wejściowe stanowią zazwyczaj współrzędne atomów Cα.

Wartość RMSD dla zbiorów $\mathcal{X}$ i $\mathcal{Y}$ jest obliczana w następujący sposób:

$$\mathrm{RMSD}\left(\mathcal{X}, \mathcal{Y}\right) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|x_i - y_i\|^2} \tag{2.33}$$

Ponieważ $\mathrm{RMSD}\left(\mathcal{X}, \mathcal{Y}\right) = 0$ wtedy i tylko wtedy gdy $\mathcal{X} \equiv \mathcal{Y}$, algorytm Kabscha ma za zadanie odnalezienie optymalnych macierzy obrotu i translacji przekształcających elementy jednego z tych zbiorów tak, aby znalazły się jak najbliżej swoich odpowiedników w drugim. Bazuje on na tej samej koncepcji co analiza składowych głównych i składa się z bardzo podobnych kroków [362].

Jeżeli punkty należące do zbiorów $\mathcal{X}$ i $\mathcal{Y}$ są zapisane odpowiednio w macierzach X i Y o rozmiarze $n \times d$, pierwszy krok algorytmu Kabscha polega na ich wyśrodkowaniu w początku układu współrzędnych. Od każdej kolumny X i Y jest więc odejmowana jej średnia arytmetyczna:

$$\begin{aligned} \tilde{X} &= X - \bar{X} \\ \tilde{Y} &= Y - \bar{Y} \end{aligned} \tag{2.34}$$

Zakładając, że macierz X zawiera docelowe (nieruchome) położenie, obliczana jest następnie macierz kowariancji C w następujący sposób:

$$C = \tilde{X}^T \tilde{Y} \tag{2.35}$$

Kolejny, najważniejszy etap polega na obliczeniu rozkładu według wartości osobliwych macierzy C:

$$C = UDV^T \tag{2.36}$$

Optymalna macierz obrotu, R, ma wówczas postać [363]:

$$R = SVU^T \tag{2.37}$$

gdzie S jest macierzą korygującą – odmianą macierzy jednostkowej o rozmiarze $d \times d$, której ostatni element zawiera iloczyn wyznaczników macierzy V i U:

$$S = \operatorname{diag}(1, 1, \ldots, 1, \det V \det U) \tag{2.38}$$

W zależności od danych wejściowych, macierz VU^T może okazać się złożeniem nie tylko obrotów, ale również pewnej liczby odbić lustrzanych. Iloczyn wyznaczników V i U pozwala stwierdzić czy liczba ta jest parzysta (1) czy też nieparzysta (-1). W pierwszym przypadku odbicia składają się w obrót, co nie wymaga korekty, a więc $S = I$, natomiast w drugim należy ją wykonać. Nieparzysta liczba odbić powoduje bowiem, że przekształcany zbiór danych staje się chiralny wobec swojego oryginału, powodując wzrost zamiast minimalizacji wartości RMSD. Korekta tego stanu polega na pomnożeniu najmniej istotnej (d-tej) składowej głównej z macierzy V przez -1, co wprowadza dodatkowe odbicie przywracające prawidłową orientację układu współrzędnych oraz zmienia znak wyznacznika macierzy R na dodatni.

Na koniec, do obliczenia pozostaje tylko optymalna macierz translacji T, wskazująca wektor przemieszczenia środka geometrycznego macierzy Y:

$$T = \bar{X} - \bar{Y}R \tag{2.39}$$

Teraz można wyznaczyć zbiór $\mathcal{Z}$, reprezentowany przez macierz Z i zawierający przekształcone dane ze zbioru $\mathcal{Y}$, dla którego wartość $\operatorname{RMSD}(\mathcal{X}, \mathcal{Z})$ jest najmniejsza:

$$Z = YR + T \tag{2.40}$$

2.5.6. Test ruchomych wierzchołków

Test ruchomych wierzchołków (moving peaks benchmark, MPB)[7] jest opracowanym przez Jurgena Branke-go generatorem kryteriów dla badań z dziedziny optymalizacji dynamicznej, tworzącym wielowymiarowe i wielomodalne funkcje o losowym oraz zmiennym w czasie krajobrazie wartości [195, 364].

Motywacją dla powstania tego algorytmu była chęć jego Autora do wypełnienia luki na skali złożoności problemów optymalizacyjnych, znajdującej się pomiędzy względnie prostymi zagadnieniami testowymi, a bardziej skomplikowanymi modelami rzeczywistych zjawisk. Dostosowywanie stopnia trudności do wymagań eksperymentu (liczba i rozmieszczenie minimów lokalnych, okres i amplituda zmiany kształtu, itd.) jest możliwe dzięki dostępnemu zestawowi parametrów.

Konstrukcja generatora MPB bardzo przypomina rój cząstek. Składa się on bowiem z populacji obiektów zwanych wierzchołkami, z których każdy jest charakteryzowany przez kilka właściwości zmieniających się według określonych reguł podczas jego aktualizacji. W szczególności, posiadają one swoje położenia w przestrzeni rozwiązań, modyfikowane poprzez dodawanie do nich wektorów prędkości.

Definicja 2.3. Generator MPB

Generator MPB jest to czwórka $(\mathcal{P}, f, b, \Phi)$, gdzie:

$\mathcal{P}$ = niepusty zbiór wierzchołków: $\mathcal{P} \equiv \{p_1, \ldots, p_n\}$

f = funkcja kształtu wierzchołków: $f : \mathbb{R}^d \times \mathcal{P} \to \mathbb{R}$

b = funkcją kształtu krajobrazu bazowego: $b : \mathbb{R}^d \to \mathbb{R}$

Φ = zbiór parametrów: $\Phi \equiv \{t, \lambda, \phi_v, \phi_w, \phi_h, Xmax, Wmax, Hmax\}$

Definicja 2.4. Wierzchołek generatora MPB

Wierzchołek MPB jest to czwórka (x, v, w, h), gdzie:

x = położenie wierzchołka: $x \in [0, Xmax]^d$

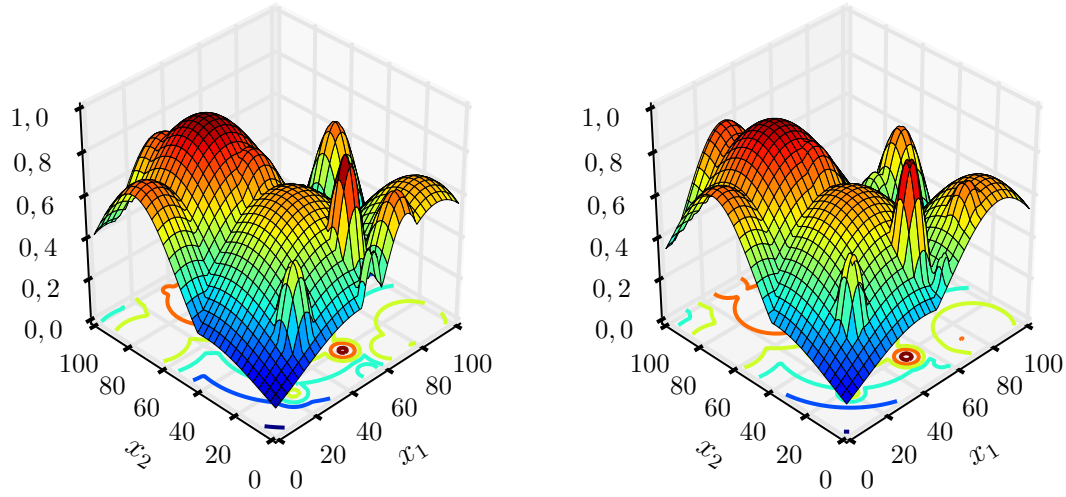
v = prędkość wierzchołka: $v \in \mathbb{R}^d$, $\|v\| = \phi_v$

w = szerokość wierzchołka: $w \in [0, Wmax]$

h = wysokość wierzchołka: $h \in [0, Hmax]$

Krajobraz wartości przykładowego kryterium optymalizacyjnego $\mathbb{R}^2 \to \mathbb{R}$ wygenerowanego przez MPB znajduje się na rysunku 2.9.

[7] Zgodnie ze stanem wiedzy Autora rozprawy, nazwa „moving peaks benchmark" nie posiada odpowiednika w języku polskim. „Test ruchomych wierzchołków" jest proponowanym tłumaczeniem.



(a) Krajobraz przed aktualizacją.

(b) Krajobraz po aktualizacji.

Rysunek 2.9: Krajobraz wartości przykładowego kryterium optymalizacyjnego wygenerowanego przy pomocy MPB, złożonego z 20 wierzchołków o kształcie opisanym przez funkcję gauss, widoczny przed i po wykonaniu jego pojedynczej aktualizacji.

Generacja

MPB generuje kryteria optymalizacyjne poprzez przypisywanie punktom z przestrzeni $\mathbb{R}^d$ wartości obliczanych na podstawie całego zbioru $\mathcal{P}$ przy pomocy funkcji f oraz b. Pierwsza z nich nadaje kształt wierzchołkom, natomiast druga wyznacza niezmienny krajobraz bazowy, po którym mogą się one przemieszczać.

Wartość wygenerowanego przez MPB kryterium w punkcie $x \in \mathbb{R}^d$ stanowi maksimum spośród wartości $b(x)$ oraz $f(x, p_i)$ dla wszystkich $i \in \{1, \ldots, n\}$:

$$MPB(x) = \max\left\{b(x), f(x, p_1), \ldots, f(x, p_n)\right\} \tag{2.41}$$

Aby uzyskać kryterium dla zagadnienia minimalizacji, wystarczy pomnożyć otrzymany wynik przez -1, co spowoduje „odbicie” wierzchołków w stronę $-\infty$.

Najważniejszym czynnikiem decydującym o kształcie krajobrazu wartości generowanych kryteriów jest funkcja f. Branke przedstawił jej cztery propozycje: function1 (nazywaną również sharp), cone, hilly oraz twin, spośród których pierwsze dwie są stosowane w standardowych scenariuszach[8,9,10] [365].

[8] http://www.aifb.uni-karlsruhe.de/~jbr/MovPeaks

[9] Kopie powyższej witryny WWW, zawierającej pliki źródłowe języka C z implementacjami tych funkcji oraz szczegóły scenariuszy, można odnaleźć w witrynie Internet Archive (http://archive.org).

[10] Funkcje hilly oraz twin są modyfikacjami funkcji cone. W pierwszej, krajobraz jest lekko zaburzony przez sinusoidę, a w drugiej wierzchołki dzielą się na dwa bliźniacze piki o wysokości h_i.

Równania funkcji function1 (f_s) i cone (f_c) są następujące:

$$f_s(x, p_i) = \frac{h_i}{1 + w_i \|x_i - x\|^2} \tag{2.42}$$

$$f_c(x, p_i) = h_i - w_i \|x_i - x\| \tag{2.43}$$

Funkcje te osiągają swoje maksima równe h_i w x_i, a ich wartości maleją do $-\infty$ proporcjonalnie do w_i wraz z oddalaniem się od x_i. Dzięki temu, ekstremum globalne wygenerowanego kryterium zawsze znajduje się w zbiorze $\{x_1, \ldots, x_n\}$.

Autor rozprawy zwraca uwagę na fakt, że we wszystkich czterech funkcjach zaproponowanych przez Branke-go, szerokość wierzchołków jest traktowana w sposób sprzeczny z intuicją. Mianowicie, stają się one coraz szersze dla coraz niższych jej wartości, co może stanowić problem podczas przewidywania zasięgu zmiany kształtu krajobrazu wartości generowanych kryteriów. Zamiast nich, może być stosowana bardziej intuicyjna (choć bardziej kosztowna) funkcja Gaussa z modelu FOD:

$$f_g(x, p_i) = \exp\left(\frac{\|x_i - x\|^2}{2\left(\frac{1+w_i}{3}\right)^2}\right) h_i \tag{2.44}$$

Dzięki regule 3-sigma, w promieniu $1 + w_i$ od x_i znajduje się 99,7% pola pod powierzchnią i-tego wierzchołka, którego maksimum w x_i pozostaje równe h_i. Liczba 1 zapewnia, że odchylenie standardowe będzie dodatnie dla dowolnych $w_i \geq 0$.

Inicjalizacja

Inicjalizację generatora MPB wykonuje się w tym samym celu i w taki sam sposób jak ma to miejsce w roju cząstek, czyli poprzez ustalenie początkowych właściwości wszystkich wierzchołków. Może je wskazać bezpośrednio użytkownik, lub mogą być wybrane w sposób losowy z dozwolonych przedziałów ich wartości:

- $x_i \in U(0, Xmax)^d$
- $w_i \in U(0, Wmax)$
- $h_i \in U(0, Hmax)$

Jedynym ograniczeniem dla wektorów prędkości jest zawarty w definicji 2.4 warunek długości każdego z nich równej ϕ_v. W szczególności może ona wynosić 0, co należy traktować jako sygnał do wyłączenia jej zmiany podczas aktualizacji.

Aktualizacja

Drugim zadaniem generatora MPB, obok tworzenia kryteriów o losowym kształcie, jest dynamiczna zmiana tego kształtu. Kluczowe znaczenie ma tu zachowywanie jego podobieństwa pomiędzy kolejnymi, wykonywanymi co t obliczeń wartości aktualizacjami, co pozwala algorytmom optymalizacyjnym na śledzenie przemieszczających się optimów. Jeżeli $t = 0$, wówczas kryteria stają się stacjonarne. Tak jak w przypadku roju cząstek, aktualizacja MPB polega na zmianie właściwości wierzchołków, aczkolwiek bez oddziaływań pomiędzy nimi.

Nowe wektory prędkości są wyznaczane na podstawie ich obecnych składowych oraz czynników losowych, po czym podlegają skalowaniu do długości równej ϕ_v:

$$v_i \leftarrow \phi_v \frac{\lambda v_i + (1-\lambda)\, r_i}{\|\lambda v_i + (1-\lambda)\, r_i\|} \tag{2.45}$$

Składowe wektora r_i są wybierane losowo z rozkładu $N(0,1)$ oraz również skalowane do długości równej ϕ_v. Parametr ten określa jak daleko wierzchołki mogą się przemieszczać podczas aktualizacji generatora. Relacja pomiędzy ich bieżącą trajektorią a wektorem r_i jest kontrolowana przez parametr λ: wartość 1 oznacza pełną kontynuację dotychczasowego kierunku, a 0 – ruch całkowicie losowy.

Wektory prędkości wierzchołków są następnie dodawane do ich bieżących położeń:

$$x_i \leftarrow x_i + v_i \tag{2.46}$$

W odróżnieniu od cząstek, których położenie nie podlega ograniczeniom, wierzchołki muszą przebywać w hiperprostokącie wyznaczonym przez parametr $Xmax$. Jest to niezbędne do zapewnienia, że ekstremum globalne wygenerowanego kryterium będzie cały czas znajdować się w tym obszarze. Kolizje wierzchołków ze ścianami tego hiperprostokąta są obsługiwane zgodnie z zasadą odbicia fali.

Na koniec pozostaje tylko zmiana szerokości i wysokości wierzchołków, polegająca na dodaniu do ich bieżących wartości liczb wybranych w sposób losowy oraz ograniczeniu odpowiednio do przedziałów $[0, Wmax]$ i $[0, Hmax]$:

$$w_i \leftarrow \min\left\{0, \max\left\{w_i + \phi_w N(0,1), Wmax\right\}\right\} \tag{2.47}$$

$$h_i \leftarrow \min\left\{0, \max\left\{h_i + \phi_h N(0,1), Wmax\right\}\right\} \tag{2.48}$$

Wizualizacja zmiany krajobrazu wartości wygenerowanego przez MPB kryterium znajduje sie na rysunku 2.9 (parametry: $\phi_v = 5$, $\phi_w = 1$, $\phi_h = \frac{1}{3}$, $\lambda = 0$, $f = \text{gauss}$).

3. Wyniki

Badania, których wyniki są przedstawione w niniejszej rozprawie doktorskiej miały na celu sprawdzenie założeń pola zewnętrznego (modelu FOD) dotyczących tworzenia się kompleksów typu białko-białko. Zgodnie z nimi, układ łańcuchów polipeptydowych dąży do usunięcia niekorzystnego termodynamicznie efektu entropowego wynikającego z ekspozycji reszt hydrofobowych do środowiska wodnego. Rozwiązaniem tego problemu jest zetknięcie ze sobą zawierających te reszty powierzchni łańcuchów, powodujące wyparcie wody z ich okolic. Reakcja ta może być więc wyrażona jako poszukiwanie stabilnej konformacji kompleksu, w której miara różnicy pomiędzy obserwowanym i teoretycznym rozkładem hydrofobowości, RD, osiąga najniższą wartość. W związku z tym, postanowiono przeprowadzić eksperyment *in silico* przewidywania struktury czwartorzędowej białek homodimerycznych za pomocą symulacji mechaniki molekularnej. Jeżeli założenia modelu FOD są poprawne, optymalizacja kryterium wartości RD powinna spowodować, że dwa rozdzielone łańcuchy przyjmą na powrót konformację zbliżoną do natywnej postaci ich kompleksu. W eksperymencie wzięło udział 200 takich struktur wybranych z bazy PDB.

W przeciwieństwie do pola zewnętrznego, skupiającego się na hydrofobowych efektach entropowych, pola wewnętrzne stosowane obecnie do przewidywania struktury kompleksów białkowych poszukują najczęściej minimum energii potencjalnej układu łańcuchów, wyrażanej za pomocą oddziaływań niekowalencyjnych między atomami: elektrostatycznych, van der Waalsa i wiązań wodorowych. Według Autorów modelu FOD, wpływ środowiska wodnego jest w nich traktowany w sposób niewystarczający. Oznacza to również, że w danej chwili optymalizacji może podlegać tylko jedna właściwość układu: energia lub hydrofobowość, pomimo tego, że obydwie te siły mają znaczenie dla osiągnięcia przez niego stanu równowagi. Ponieważ relacja między tymi siłami jest nieznana, postanowiono powtórzyć powyższy eksperyment kompleksowania, ale przy użyciu kryterium wybranego reprezentanta pól wewnętrznych – pola ECEPP/3. Do wymiernej oceny wyników użyto miary RMSD oraz porównania map kontaktów niewiążących w przestrzeni krzywych ROC.

Zarówno energia potencjalna oddziaływań niekowalencyjnych jak i wpływ środowiska wodnego opisywany przez model FOD nie są tożsame z energią swobodną Gibbsa układu, przez co nie można zagwarantować, że ich minima globalne będą odpowiadać jej minimom, a przez to poszukiwanej, stabilnej konformacji natywnej struktury kompleksu białkowego. Nie ma obecnie możliwości bezpośredniego połączenia tych funkcji ze sobą, na przykład poprzez zastosowanie wag, ponieważ zwracają niekompatybilne wartości: z jednej strony występują kilokalorie na mol, których sens ograniczony jest do konkretnego pola wewnętrznego, a z drugiej – pozbawione jednostek rozkłady hydrofobowości, porównywane przy pomocy dywergencji Kullbacka-Leiblera. Niewykluczone, że kiedyś powstanie model, który tego dokona, lub w inny sposób przybliży rzeczywiste siły zarządzające procesami biologicznymi związanymi z białkami, a przez to pozwoli na ich trafne przewidywanie *in silico*. Aby zbliżyć się do jego odkrycia, niezbędne jest więc jak najlepsze poznanie relacji pomiędzy tymi polami. Pierwszy krok ku temu stanowi sprawdzenie ich przydatności jako kryteriów optymalizacyjnych w mechanice molekularnej, polegające na obserwacji czy istnieją białka, które tworzą kompleksy zgodnie z założeniami modelu FOD. Druga sprawa dotyczy porównania uzyskanych w ten sposób wyników z wynikami osiągniętymi przy pomocy pól wewnętrznych. Skoro nie istnieje równanie łączące w sobie energię i hydrofobowość, należy zaobserwować jak zachowuje się układ, gdy obydwa pola działają na niego osobno, choć tym samym czasie. Każde pole będzie wówczas kierować kompleks ku konformacjom o swoich niskich wartościach, ale tylko w taki sposób, na ile pozwala na to drugie pole. Wynikiem symulacji staje się wówczas zbiór Pareto. Obecność konformacji natywnej kompleksu w nim lub w jego pobliżu będzie sygnałem, że równowaga termodynamiczna danego kompleksu wynika ze „starcia" sił reprezentowanych przez wybrane modele energii i hydrofobowości.

Zbiór Pareto jest wyznaczany w trakcie optymalizacji wielokryterialnej. Istnieje wiele algorytmów realizujących to zadanie, ale nie oferujących przydatnej w tym eksperymencie funkcjonalności. Potrzebna jest tu bowiem możliwość automatycznego dzielenia wyników na podzbiory reprezentujące różne grupy podobnych konformacji, które mogą być dalej analizowane i oceniane przez inne kryteria, a także uzyskiwania ich jednorodnej reprezentacji. Aby sprostać temu zadaniu, został opracowany przez Autora rozprawy nowy algorytm o nazwie MOSF (wielokryterialne rodziny rojów). Pomimo tego, że motywacją do jego utworzenia jest bioinformatyka, ma on również zastosowanie ogólne. Porównanie algorytmu MOSF z algorytmami NSGA-II i NSPSO wykazało, że nie ustępuje on pod względem dokładności powszechnie stosowanym technikom, jednocześnie oferując do tej pory niedostępną funkcjonalność.

3.1. Algorytm MOSF – prezentacja

Poniżej przedstawiony jest opracowany samodzielnie przez Autora rozprawy nowy algorytm optymalizacji wielokryterialnej o nazwie wielokryterialne rodziny rojów (multi objective swarm families, MOSF), stanowiący rozszerzenie klasycznego algorytmu PSO i służący do poszukiwania optymalnego frontu Pareto dowolnej funkcji wielokryterialnej $F(x) = \left[f_1(x) \dots f_k(x)\right] : \Omega \subseteq \mathbb{R}^d \to \mathbb{R}^k$, gdzie $d, k \geq 1$.

3.1.1. Motywacja

Motywacją do opracowania nowego algorytmu była chęć uzyskania możliwości przeprowadzania w trakcie optymalizacji wielokryterialnej automatycznej „analizy skupień" elementów odnajdywanego przybliżenia optymalnego zbioru Pareto. Poprzez automatyzację tej procedury jest tutaj rozumiana niezależność od zewnętrznych (niezwiązanych z samą optymalizacją) parametrów, takich jak liczba skupisk. Zgodnie ze stanem wiedzy Autora rozprawy, funkcjonalność ta oraz inne możliwości algorytmu MOSF nie są oferowane przez dotychczas opublikowane algorytmy.

Słowa „analiza skupień" zostały ujęte w cudzysłów, ponieważ realizacja tej czynności w algorytmie MOSF nie jest identyczna z jej powszechnie przyjętą definicją, przedstawioną w rozdziale 2.5.2. Cel tej czynności pozostaje jednak taki sam: podział zbioru (w tym przypadku – Pareto) na podzbiory zawierające podobne do siebie rozwiązania. Najważniejsza różnica między tymi podejściami polega właśnie na sposobie wykonania tego podziału. Zanim jednak będzie można go przedstawić, należy wyjaśnić, dlaczego w ogóle warto zajmować się tym tematem.

Ponieważ wszystkie rozwiązania niezdominowane są sobie równoważne w sensie optimum Pareto, podzielenie ich na skupiska pozwala na uzyskanie dodatkowej informacji na ich temat, a przez to wyciągnięcie wniosków dotyczących badanego układu i optymalizowanych kryteriów. Wiedza ta udostępnia także nowe heurystyki dla dalszej (bardziej precyzyjnej) inicjalizacji algorytmów optymalizacyjnych. Aby to zobrazować najlepiej posłużyć się przykładem, nieprzypadkowo związanym z tematem niniejszej rozprawy. Otóż, jeżeli wynik symulacji tworzenia się kompleksu typu białko-białko stanowi zbiór Pareto, każdy jego element reprezentuje jedną z wielu alternatywnych konformacji łańcuchów. Podział tego zbioru na skupiska wyróżnia grupy podobnych do siebie i różnych od innych ułożeń cząsteczek, które mogą być następnie wspólnie poddawane ocenie przy pomocy innych kryteriów, pozwalającej na wybranie ich najlepszych reprezentantów lub odrzucenie w całości.

Elementy przybliżenia optymalnego zbioru Pareto są „analizowane” przez algorytm MOSF na bieżąco w trakcie trwania optymalizacji, dzięki czemu mogą dynamicznie łączyć się w skupiska, podobnie jak ma to miejsce w algorytmie aglomeracyjnym. Ponieważ oczekiwana liczba tych skupisk jest nieznana, przyjęto dwie zasady określające kiedy para rozwiązań będzie do siebie wystarczająco podobna. Do umieszczenia ich we wspólnym skupisku wystarczy, że zostanie spełniony przynajmniej jeden z tych warunków. W pierwszym przypadku, decydującym czynnikiem jest odległość w przestrzeni rozwiązań. Za podobne uznawane są te rozwiązania, które znajdują się względnie blisko siebie. Stosowana tu miara niepodobieństwa jest omówiona w dalszej części rozdziału. Alternatywny warunek dotyczy natomiast kształtu krajobrazu wartości funkcji F. Umożliwia on umieszczanie we wspólnych skupiskach tych rozwiązań, pomiędzy którymi przemieścił się algorytm optymalizacyjny poszukujący minimum globalnego któregoś z kryteriów $f_1, \ldots, f_k$. Mówiąc inaczej, gdy kryterium to prowadzi ten algorytm pomiędzy nimi. W połączeniu ze sprawnym generatorem rozwiązań niezdominowanych, tak zdefiniowana „analiza skupień” stanowi przydatne narzędzie zdolne do odkrywania niewidocznych właściwości badanego układu.

Do realizacji powyższego zadania niezbędne było opracowanie takiego podejścia, które będzie brało pod uwagę zarówno rozwiązania niezdominowane, jak i każde z kryteriów $f_1, \ldots, f_k$ osobno. W tym celu Autor rozprawy postanowił zmodyfikować powszechnie znany algorytm PSO, wzorując się na metodzie VEPSO (vector-evaluated particle swarm optimization) Parsopoulosa i współpracowników [308]. Metoda VEPSO stosuje do poszukiwania rozwiązań niezdominowanych grupę jednokryterialnych rojów cząstek, z których każdy posiada taki sam zestaw parametrów i optymalizuje inne kryterium $f_1, \ldots, f_k$, podążając jednocześnie za liderem następnego w kolejności roju, tworząc w ten sposób topologiczny pierścień wyższego poziomu. Ponieważ minima globalne są zawsze niezdominowane, działanie to przyczynia się do kierowania algorytmu w pobliże optymalnego zbioru Pareto. Przybliżenie jego kształtu oraz zawartości stanowi jednak wyłącznie efekt uboczny tej procedury, w szczególności, że wyniki przechowywane są w wektorach pamięci cząstek.

Algorytm MOSF rozszerza ideę algorytmu VEPSO, oferując możliwość ustawiania parametrów indywidualnych rojów, a dzięki zastosowaniu zewnętrznego archiwum – na sprawniejsze kierowanie ruchem cząstek, prowadzące do dokładnego odwzorowania rozkładu elementów optymalnego zbioru Pareto, wraz z ich automatyczną „analizą skupień”. Podobnie jak swój pierwowzór, może być on również wykonywany równolegle, co znacznie przyspiesza jego działanie [366]. Szczegóły dotyczące wykonania równoległego oraz wyniki testów są przedstawione w dodatku C.3.

3.1.2. Definicja

Zgodnie z taksonomią przedstawioną w rozdziale 2.4.5, algorytm MOSF jest techniką wielorojową, wykonującą optymalizację wielokryterialną za pomocą zbioru rodzin rojów cząstek, których definicja jest następująca:

Definicja 3.1. Rodzina rojów cząstek w algorytmie MOSF

Rodzina rojów cząstek, PSF jest to piątka $(F, \mathcal{G}, \mathcal{A}, \mathcal{S}, \Phi)$, gdzie:

F = funkcja wielokryterialna: $F : \Omega \subseteq \mathbb{R}^d \to \mathbb{R}^k$

$\mathcal{G}$ = zbiór funkcji ograniczeń nierówności: $\mathcal{G} \equiv \{g_1, \ldots, g_m\}$

$\mathcal{A}$ = archiwum rozwiązań niezdominowanych: $\mathcal{A} \subseteq \Omega$

$\mathcal{S}$ = zbiór rojów cząstek: $\mathcal{S} \equiv \{PSO_1 \ldots, PSO_k\}$

Φ = zbiór parametrów: $\Phi \equiv \{a, n, s, t_d, t_m, r\}$

Identycznie jak w metodzie VEPSO, każdy rój PSO_i poszukuje rozwiązania stanowiącego minimum globalne kryterium f_i w dopuszczalnym przez wszystkie funkcje ograniczeń podzbiorze zbioru Ω. Standardowo, roje te są identyczne pod względem ustalonych *a priori* parametrów cząstek (dlatego niewystępujących w zbiorze Φ). Możliwe jest również ich indywidualne dopasowywanie. W szczególności, każdy rój może wykonywać optymalizację wielokryterialną, lub nawet zajmować się zupełnie inną funkcją niż F. Takie podejście przekłada się na otwartą konstrukcję algorytmu, przypominającą luźno połączone bloki. Jeszcze inną zaletą tego podejścia jest fakt, że jeżeli kryterium f jest tylko jedno, to znaczy $F = f_1 : \Omega \subseteq \mathbb{R}^d \to \mathbb{R}$, wówczas optymalizacja wielokryterialna w wykonaniu algorytmu MOSF samoistnie zmienia się globalną, bez potrzeby wprowadzania dodatkowych modyfikacji.

Relacja algorytmu MOSF z podległymi mu rojami może być określona mianem komensalizmu (współbiesiadnictwa), gdyż stara się on ingerować w sposób ich działania w minimalnym stopniu. Jego praca polega bowiem na pobieraniu i zapisywaniu w archiwum $\mathcal{A}$ kandydatów na rozwiązania niezdominowane, tożsamych ze zmieniającymi się położeniami cząstek. Archiwum to stanowi jednocześnie jedyną formę pośredniej komunikacji pomiędzy rojami w danej rodzinie. Choć nie są do tego zobowiązane, mogą i powinny korzystać z jego zawartości, co pozwala im na swobodne poszukiwanie minimów globalnych przydzielonych im kryteriów $f_1, \ldots, f_k$ oraz kierowanie się w stronę bieżącego przybliżenia optymalnego zbioru Pareto bez potrzeby modyfikowania swoich mechanizmów pamięci. Oznacza to, że cząstki w algorytmie MOSF mają *de facto* dwóch liderów przyciągających je w możliwie przeciwstawnych kierunkach, pomiędzy którymi musi być zachowana równowaga.

W poprzednich paragrafach można odnaleźć odniesienia wyłącznie do zbioru Pareto. Wynika to stąd, że roje cząstek w algorytmie MOSF nie analizują zawartości bieżącego przybliżenia optymalnego frontu Pareto, bazując na założeniu, że nie musi występować jakakolwiek nadająca się do wykorzystania relacja pomiędzy tymi zbiorami. Autor rozprawy przyjął, że większe znaczenie ma osiągnięcie dokładnej reprezentacji punktów w przestrzeni rozwiązań, co powinno przełożyć się na ich satysfakcjonujący rozkład w przestrzeni wartości, a przynajmniej dostarczyć solidnych podstaw do osiągnięcia tego celu. Redukcja roli frontu Pareto wyłącznie do narzędzia określającego niezdominowanie punktów powoduje, że algorytm MOSF unika problemów związanych z kształtem, ciągłością i gęstością jego elementów. Natomiast dzięki temu, że każdemu kryterium $f_1, \ldots, f_k$ jest przypisany jeden rój, który nie oddziałuje bezpośrednio z innymi, wpływ liczby optymalizowanych kryteriów na złożoność obliczeniową całej procedury optymalizacji jest liniowy.

Algorytm MOSF przechowuje bieżące przybliżenie optymalnego zbioru Pareto w zewnętrznych archiwach rodzin rojów, których maksymalny rozmiar kontroluje parametr a. Ponieważ potencjalnych kandydatów na rozwiązana niezdominowane dostarczanych w pojedynczej iteracji jest tyle, ile wynosi całkowita liczba cząstek we wszystkich rojach, w momencie przepełnienia musi zostać wykonana procedura przycinania. Zadanie to polega na wybraniu a-elementowego podzbioru każdego archiwum w taki sposób, aby zapewniona była jak najlepsza reprezentacja jego oryginalnej zawartości, która będzie kierować cząstki we właściwą stronę. Pod tym pojęciem rozumiane jest osiągnięcie rozkładu danych w przestrzeni rozwiązań zbliżonego do jednorodnego (w sensie wierzchołków diagramu Woronoja [317, 367]), ale jednocześnie zachowującego kluczowe dla optymalizacji informacje na temat struktury zbioru Pareto, czyli izolowanych punktów oraz ich gęstych skupisk.

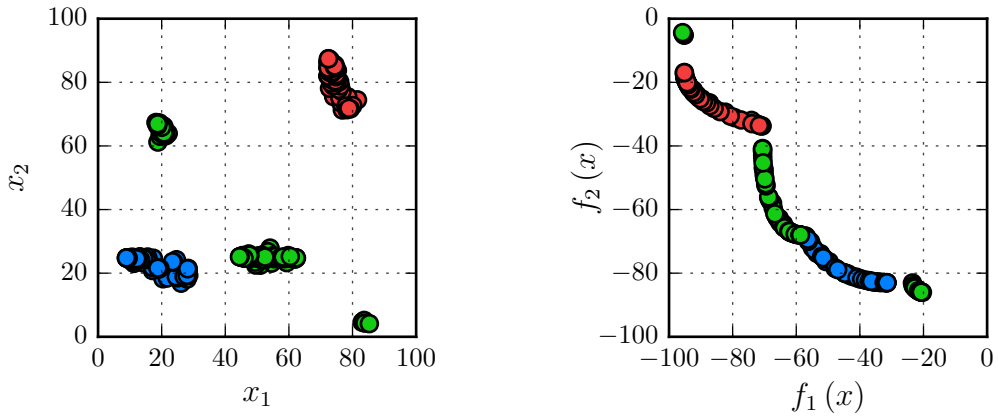
Główną motywacją do opracowania algorytmu MOSF była wspomniana wcześniej chęć przeprowadzania na bieżąco „analizy skupień” zawartości aktualnego przybliżenia optymalnego zbioru Pareto. Jak można jednak zauważyć, pojedyncza rodzina rojów będzie zawsze produkować pojedyncze skupisko, tożsame z jego archiwum. Do uzyskania zamierzonego efektu należy więc użyć kilku takich rodzin. Każda z nich staje się wówczas niezależnym od pozostałych nośnikiem jednego skupiska, samodzielnie zmieniającego swoje położenie w przestrzeni rozwiązań wraz z cząstkami podległych mu rojów. Jeżeli elementy dwóch różnych archiwów znajdą się w tym samym obszarze, będzie to oznaczało, że mogą zostać połączone, gdyż doprowadziły je w to miejsce malejące wartości kryteriów $f_1, \ldots, f_k$, lub były umieszczone blisko siebie od początku trwania procedury optymalizacji.

„Analiza skupień” w wykonaniu algorytmu MOSF stanowi naturalną konsekwencję sposobu generacji przez niego rozwiązań niezdominowanych i nie wymaga wprowadzania dodatkowych modyfikacji w sposobie poruszania się cząstek. Do jej realizacji potrzebne są jednak dwie specjalne procedury: łączenia i dzielenia. Idea pierwszej z nich została już omówiona w poprzednim paragrafie, tak więc pozostaje przedstawić drugą. Czynność ta – zgodnie ze swoją nazwą – pozwala tworzyć potomne rodziny rojów na podstawie zawartości archiwów bieżącego „pokolenia”. Dzięki temu, pierwsza z nich staje się korzeniem drzewa, którego kolejne poziomy odpowiadają coraz bardziej zlokalizowanym poszukiwaniom elementów optymalnego zbioru Pareto. O liczbie tych podziałów decyduje użytkownik. Nie są one obowiązkowe, ale do wykonania „analizy skupień” niezbędny jest przynajmniej jeden. Oprócz tych mechanizmów, algorytm MOSF dokonuje również wzajemnej dominacji rodzin rojów. Polega ona na usuwaniu z ich archiwów punktów zdominowanych przez pozostałe rodziny oraz na eliminacji tych, które zostały całkowicie opróżnione.

Dalsze części tego rozdziału przedstawiają szczegóły funkcjonowania algorytmu MOSF, natomiast poniżej znajduje się podsumowanie najważniejszych cech wyróżniających go na tle alternatywnych metod obliczeniowych:

1. możliwość wykonywania optymalizacji wielokryterialnej i globalnej,
2. niezależność od kształtu i rozkładu elementów we froncie Pareto oraz liczby i zakresów wartości optymalizowanych kryteriów,
3. możliwość dostosowywania ustawień indywidualnych rojów cząstek,
4. automatyczna „analiza skupień” elementów zbioru Pareto,
5. możliwość systematycznego zwiększania precyzji optymalizacji,
6. archiwizacja wyników zachowująca istotne informacje na temat zbioru Pareto, a przez to jednorodnie reprezentująca rozkład jego elementów,
7. możliwość efektywnego wykonania równoległego.

Prezentacja przykładowego wyniku działania algorytmu MOSF, użytego do odnalezienia optymalnego zbioru i frontu Pareto funkcji $F : \mathbb{R}^2 \to \mathbb{R}^2$, której kryteria zostały wygenerowane w sposób losowy przy pomocy testu ruchomych wierzchołków, znajduje się na rysunku 3.1. Zbiór Pareto został podzielony na trzy skupiska. Jedno z nich, zielone, pomimo tego, że znajduje się w pobliżu skupiska niebieskiego, nie zostało z nim połączone ze względu na rozdzielające je maksima lokalne. Za to trafiły do niego końce frontu Pareto, do których zaprowadziły roje kryteria f_1 i f_2.



(a) Zbiór Pareto w przestrzeni rozwiązań. (b) Front Pareto w przestrzeni wartości.

Rysunek 3.1: Przykładowy wynik działania algorytmu MOSF, prezentujący podział optymalnego zbioru rozwiązań niezdominowanych na trzy skupiska, oznaczone kolorami czerwonym, zielonym i niebieskim. Kryteria optymalizacyjne f_1 i f_2 zostały wygenerowane w sposób losowy przy pomocy testu ruchomych wierzchołków.

3.1.3. Inicjalizacja

Do inicjalizacji algorytmu MOSF potrzebna jest przynajmniej jedna rodzina rojów. Zgodnie z definicją 3.1, aby ją utworzyć, wymagane są: funkcja wielokryterialna F, zbiór funkcji ograniczeń $\mathcal{G}$ oraz niepusty zbiór rojów cząstek $\mathcal{S}$. Archiwum $\mathcal{A}$ jest na początku optymalizacji zazwyczaj puste, ale nie ma takiego wymogu.

Roje należące do każdej rodziny rojów również muszą być zainicjalizowane. Czynność ta jest wykonywana na wszystkich z nich jednocześnie, w sposób właściwy dla realizowanych przez nie algorytmów. Standardowo, odbywa się to tak jak w klasycznym roju cząstek: początkowe wektory położeń i pamięci są wybierane losowo z wnętrza hiperprostokąta, którego równoległe do osi układu współrzędnych boki określają dodatkowo początkową prędkość. Może być on podany przez użytkownika lub dopasowany do najbardziej skrajnych elementów archiwum. Druga z tych sytuacji dotyczy zazwyczaj inicjalizacji wykonywanej po podziale rodzin rojów.

Na tym etapie ustalana jest również maksymalna prędkość, identyczna dla dla wszystkich rojów. Ma ona istotne znaczenie dla „analizy skupień”, gdyż wpływa na zasięg ruchu cząstek, a przez to na możliwość odwiedzania przez nie miejsc, w których działają inne rodziny i łączenia się z nimi. Ponownie, najprostszym sposobem ustalenia tego parametru jest odniesienie się do rozmiaru początkowego hiperprostokąta.

3.1.4. Aktualizacja

Pojedyncza iteracja algorytmu MOSF sprowadza się do aktualizacji zarządzanych przez niego rodzin rojów oraz wykonaniu na nich pozostałych procedur: dominacji, łączenia i dzielenia. Pierwsze z tych zadań jest realizowane w każdej iteracji, natomiast pozostałe są opcjonalne i zależą od preferencji użytkownika.

Podczas aktualizacji pojedynczej rodziny rojów następuje aktualizacja należących do niej rojów cząstek, zapisanie ich położeń w archiwum, a następnie usunięcie z tego archiwum punktów zdominowanych oraz tych, które zostały odrzucone podczas jego przycinania do ustalonego rozmiaru. Czynności te są powtarzane w pętli do czasu aż wszystkie roje spełnią ustalone dla nich warunki STOP. Maksymalna liczba iteracji jest zazwyczaj wspólna. Stosowany jest tu także warunek spadku średniej prędkości cząstek poniżej określonego progu będącego ułamkiem jej maksymalnej wartości, czyli tak jak zostało to przedstawione w rozdziale 2.4.4.

Ponieważ rodziny rojów nie mają na siebie wpływu podczas ich aktualizacji, możliwe jest ich przetwarzanie w dowolnej kolejności. Oznacza to, że czynność ta nadaje się do wykonania równoległego. Do jednostek roboczych mogą być przekazywane całe rodziny, pojedyncze roje, lub indywidualne cząstki. Coraz mniejsza ziarnistość maksymalizuje użycie procesora kosztem jednoczesnego zwiększenia użycia mechanizmów synchronizacji i wynikającego z tego skomplikowania programu oraz wykorzystania zasobów systemowych. W związku z tym, że efektywność wykonania równoległego jest zależna od implementacji algorytmu, kwestie związane z tym tematem zostały przeniesione do dodatku C.3.

Generacja rozwiązań niezdominowanych

Procedura generacji kandydatów na rozwiązania niezdominowane jest tożsama z aktualizacją wszystkich rojów należących do danej rodziny, które nie spełniają swoich warunków STOP, a następnie zapisaniu w archiwum położeń ich cząstek. Funkcje ograniczeń są obsługiwane przy pomocy strategii turniejowej Deba.

Ponieważ domyślnie stosowane roje cząstek wykonują optymalizację globalną, w celu przemieszczenia ich w stronę innych elementów zbioru Pareto niż minima globalne, algorytm MOSF stosuje zmodyfikowaną wersję równania 2.21:

$$v_i \leftarrow \phi_v v_i + \phi_m \left(m_i - x_i\right) \odot r_{m,i} + \phi_l \left(l_i - x_i\right) \odot r_{l,i} + \phi_e \left(e_i - x_i\right) \odot r_{e,i} \tag{3.1}$$

Aktualizacja położenia i pamięci jest taka sama jak w klasycznym algorytmie PSO.

Równanie 3.1 różni się od swojego pierwowzoru obecnością parametru ϕ_e oraz wektora e_i. Litera e jest skrótem od słowa external, oznaczającego zewnętrzny bodziec, który stanowi jeszcze jedną, obok pamięci oraz „standardowego" lidera, składową nowego kierunku ruchu cząstek. Parametr ϕ_e oraz wektor turbulencji $r_{e,i}$ są stosowane identycznie jak ich klasyczne odpowiedniki. Ponieważ ów zewnętrzny bodziec jest traktowany jako drugi lider, będzie od teraz nazywany liderem zewnętrznym w celu odróżnienia go lidera wewnętrznego, wskazywanego przez topologię roju.

Wektory e_i są elementami archiwum $\mathcal{A}$. W algorytmie MOSF archiwum jest strukturą danych, która przechowuje niezdominowane położenia cząstek, utrzymując je w kolejności rosnącego zagęszczenia ich sąsiadów w przestrzeni rozwiązań. Własność tę zapewnia algorytm archiwizacji uruchamiany po aktualizacji rojów.

Lider zewnętrzny i-tej cząstki jest wyznaczany w następujący sposób:

$$e_i = \begin{cases} \mathcal{A}_j, \quad j = \min\left\{ \left\lfloor \left| N\left(0, \frac{|\mathcal{A}|}{6}\right) \right| + \frac{1}{2} \right\rfloor + 1, |\mathcal{A}| \right\} & |\mathcal{A}| > 0 \\ x_i, & |\mathcal{A}| = 0 \end{cases} \tag{3.2}$$

Dzięki użyciu rozkładu normalnego o odchyleniu standardowym $\frac{|\mathcal{A}|}{6}$, za każdym razem istnieje większe prawdopodobieństwo, że cząstki zostaną skierowane ku bardziej izolowanym rozwiązaniom spośród wszystkich obecnie zapamiętanych w archiwum. Ów losowy wybór, będący pewną formą prostej topologii, zapewnia ciągłe poruszanie się wszystkich rojów, przeciwdziałając ich zbieganiu się w pojedynczych punktach. W połączeniu z faktem, że cząstki nadal mogą poruszać się w stronę swoich liderów wewnętrznych, prowadzi to do coraz dokładniejszego przybliżania zawartości optymalnego zbioru Pareto. Właściwość ta jest szczególnie pomocna podczas pierwszej iteracji algorytmu, w której archiwum może być puste. Liderami zewnętrznymi cząstek stają się wówczas ich bieżące położenia, co redukuje wpływ tego czynnika do zera, ale nie stanowi problemu właśnie ze względu na dostępność liderów wewnętrznych.

Komunikacja pomiędzy rojami za pośrednictwem wspólnego archiwum pozwala im dodatkowo na wzajemne sugerowanie niskich wartości optymalizowanych przez nie kryteriów $f_1, \ldots, f_k$. Choć każdy interesuje się tylko jednym, algorytm MOSF musi obliczać wartości wszystkich z nich w celu sprawdzenia czy położenia cząstek są niezdominowane. W ten sposób, rozwiązane odnalezione i wpisane do archiwum dzięki jednemu z rojów może po pewnym czasie przyciągnąć pozostałe, które znajdą w jego pobliżu swoich nowych liderów wewnętrznych. Jest to powód, dla którego nie ma potrzeby wprowadzania zmian w sposobie działania mechanizmu zapamiętywania rozwiązań przez cząstki.

Inną konsekwencją stosowania równania 3.1 jest zmiana sposobu doboru wartości parametrów rojów. Na podstawie obserwacji wyników działania algorytmu MOSF w przypadku wybranych funkcji testowych i generatora MPB, Autor rozprawy opracował zestaw reguł zwiększających szansę uzyskania zadowalającego przybliżenia optymalnego zbioru Pareto o oczekiwanych właściwościach rozkładu jego elementów:

$$\begin{cases} \phi_v \in U\,(0,1) \\ \phi_l + \phi_e \leq \phi_m \\ \phi_m = 2{,}0 \\ \phi_l \leq \phi_e = 1{,}0 \end{cases} \tag{3.3}$$

Zależność pomiędzy bezwładnością cząstek a pozostałymi współczynnikami nie jest jeszcze w dokładnie znana, dlatego ustalono, że będą one spowalniane do ułamka swojej bieżącej prędkości, za każdym razem wybieranego losowo z przedziału $[0, 1]$. Następnie, ze względu na obecność dwóch liderów: wewnętrznego i zewnętrznego, którzy mogą przyciągać cząstki w różnych kierunkach, w celu uniknięcia wynikającego stąd potencjalnego rzucania się utrudniającego dotarcie im w pobliże któregokolwiek z tych wektorów, przyjęto, że suma $\phi_l + \phi_e$ nie powinna być większa od wartości parametru ϕ_m, wynoszącego domyślnie 2,0.

Optymalizacja wielokryterialna jest realizowana przez rodziny rojów algorytmu MOSF wtedy, gdy parametr ϕ_e ma dodatnią wartość. Oczekiwane właściwości rozkładu elementów przybliżenia optymalnego zbioru Pareto mogą być osiągnięte już dla wartości $\phi_e = 0{,}1$. Wraz ze wzrostem liczby wykonanych iteracji i przy odpowiednio wysokiej wartości parametru ϕ_l, cząstki będą jednak coraz bardziej zbiegać się w pojedynczych punktach, co wynika z nakładania się na siebie wektorów pamięci i liderów wewnętrznych. Optymalizacja wielokryterialna może się wówczas zredukować do globalnej lub wielomodalnej. Z drugiej strony, liderzy wewnętrzni nie są niezbędni do jej przeprowadzenia, co oznacza, że ustawienie parametru ϕ_l na 0 jest możliwe, ale powoduje spowolnienie docierania algorytmu do optymalnego frontu Pareto i utrudnia przeprowadzenie „analizy skupień”. W związku z tym, przyjęto, że równość pomiędzy wartościami ϕ_l, ϕ_e oraz liczbą 1 powinna zachowywać równowagę w dążeniu rojów do minimalizacji indywidualnych kryteriów i maksymalizacji liczby odnalezionych rozwiązań niezdominowanych, skutkującą utrzymywaniem cząstek w ciągłym ruchu. Alternatywną do tego podejścia, podobnie jak w algorytmie PSO, jest stopniowe zmniejszanie wartości parametru ϕ_l z 1, przez 0,5, aż do 0,1, lub użycie drugiej albo trzeciej z nich od początku procedury optymalizacji.

Archiwizacja rozwiązań niezdominowanych

Algorytm MOSF przechowuje bieżące przybliżenie optymalnego zbioru Pareto w archiwach swoich rodzin rojów. Po dodaniu do nich w poprzednim kroku nowych położeń cząstek następuje usunięcie z każdego rozwiązań zdominowanych. Na tym etapie różne rodziny rojów nie są ze sobą pod tym względem porównywane. Podejście to wynika z kwestii ekonomicznych (archiwa szybko się zapełniają, a złożoność obliczeniowa ich wzajemnej dominacji jest kwadratowa) oraz w celu zapewnienia cząstkom swobody w wyborze liderów zewnętrznych, przekładającej się na większą płynność ich ruchu i większy zasięg przeszukiwania przestrzeni rozwiązań.

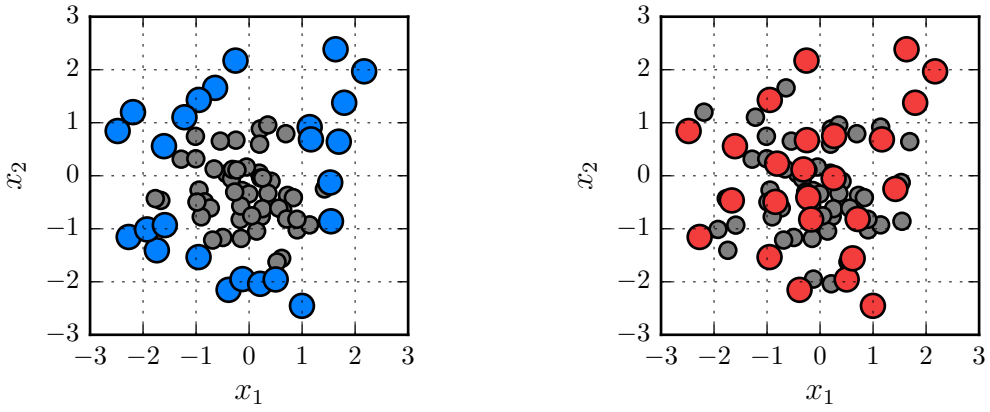
Funkcje ograniczeń są traktowane przez archiwizator jako dodatkowe kryteria optymalizacyjne postaci $\tilde{g}_i(x) = \max\left\{0, g_i(x)\right\}$. Do ich obsługi stosowana jest również strategia turniejowa Deba. Zgodnie z nią, rozwiązania dopuszczalne zawsze dominują niedopuszczalne, co oznacza, że w danej chwili mogą być zapamiętani przedstawiciele tylko jednego z tych zbiorów. Dzięki temu nie ma potrzeby obliczania wartości kryteriów $f_1, \ldots, f_k$ dla tych drugich (zastępuje je $+\infty$), co istotnie skraca czas działania algorytmu, jednocześnie gwarantując, że archiwa nie pozostaną puste, niezależnie od tego, gdzie będą znajdować się cząstki.

Archiwizator algorytmu MOSF realizuje dwa zadania: oblicza zagęszczenie sąsiedztwa każdego z zapamiętanych rozwiązań i układa je w rosnącym porządku tej własności, a następnie usuwa tyle, aby nie pozostało ich więcej niż wynosi wartość parametru a. Gęstość otoczenia elementów pojedynczego archiwum jest wyrażana jako suma odwrotności odległości od każdego z nich do jego n najbliższych sąsiadów w przestrzeni rozwiązań (parametr ze zbioru Φ). Odnajdywani są oni przy pomocy struktury drzewa k-d, stosując metrykę euklidesową (szczegóły w rozdziale 2.5.1). Z tego powodu niezbędna jest wstępna standaryzacja, polegająca na podzieleniu zawartości archiwum przez odchylenia standardowe wszystkich optymalizowanych zmiennych. Uniemożliwia to użycie drzewa z poprzedniej iteracji, chyba, że nie nastąpiła jego zmiana. Ponieważ algorytm poszukiwania najbliższych sąsiadów jest dotknięty przekleństwem wymiarowości, uznano, że większe znaczenie od dokładności powinien mieć czas jego działania, w związku z czym postanowiono, że będzie on zwracać wyniki przybliżone, stosując wartość ϵ równą 1. Alternatywnym podejściem, charakteryzującym się jeszcze niższą dokładnością, ale za to zależnym liniowo od liczby wymiarów przestrzeni rozwiązań jest użycie algorytmu niszowania [309]. Pozwala ono na uproszczenie tej części procedury archiwizacji do sortowania punktów we wszystkich tych wymiarach, a więc posiadającego złożoność obliczeniową $O(d\,n \log n)$.

Po posortowaniu zapamiętanych rozwiązań niezdominowanych zgodnie z miarą gęstości ich sąsiedztwa, jeżeli rozmiar archiwum nie przekracza wartości parametru a, praca archiwizatora się kończy. W przeciwnym razie uruchamiana jest procedura przycinania. Ma ona na celu wybranie tych elementów archiwum, które możliwie najdokładniej odwzorowują jego zawartość, niezależnie od ich liczby, zarówno wejściowej, jak i wyjściowej. Zgodnie z wcześniejszymi ustaleniami, oznacza to, że docelowy rozkład danych powinien być w przestrzeni rozwiązań zbliżony do jednorodnego: unikającego nadmiernej reprezentacji gęstych skupisk i jednocześnie zachowującego informację na temat punktów izolowanych, czyli rzadko odwiedzanych, a więc możliwie trudnych do odnalezienia przez bieżące lub pochodne rodziny rojów. Dzięki temu cząstki są kierowane ku dynamicznie zmieniającym się liderom zewnętrznym: gdy jakiś obszar zostanie przez nie dostatecznie dobrze sprawdzony, ich uwaga kieruje się w inną stronę. Efektem tego działania jest dążenie do maksymalizacji pokrycia aktualnego przybliżenia optymalnego zbioru Pareto.

Rozwiązania niezdominowane stanowiące poszukiwaną jednorodną reprezentację zawartości archiwum są w algorytmie MOSF utożsamiane z wierzchołkami diagramu Woronoja. Do ich wyznaczenia wystarczy posłużyć się algorytmem Lloyda, czyli wykonać analizę skupień k-średnich. Oczekiwana liczba skupisk jest równa a, natomiast liczbę iteracji kontroluje inny parametr ze zbioru Φ – s. Kombinatoryczne metody analizy skupień są silnie zależne od wyboru początkowych centroidów, które również w tym przypadku mają istotne znaczenie. To one decydują bowiem o tym, czy wynik zwrócony przez archiwizator będzie posiadać oczekiwane właściwości. W tym celu przydają się dane uzyskane w kroku poprzednim. Mianowicie, na początkowe centroidy wybierane jest a najbardziej izolowanych rozwiązań w sensie metryki euklidesowej w zestandaryzowanym archiwum. Ponieważ każdy punkt może należeć w danej chwili do tylko jednego skupiska, podejście to przeciwdziała zbieganiu się centroidów w obszarach o wysokiej gęstości, utrzymując je w pewnej odległości od siebie. Po wykonaniu s iteracji, przy pomocy utworzonej wcześniej struktury drzewa k-d odnajdywani są najbliżsi sąsiedzi tych centroidów. Reprezentowane przez nie skupiska nie mają w tym momencie znaczenia. Oznacza to, że archiwizator może zwrócić mniej niż a rozwiązań ponieważ najbliższymi sąsiadami centroidów tych skupisk mogą okazać się te same punkty. Powstałe z tego powodu braki są uzupełniane w sposób losowy spośród pozostałych elementów archiwum.

Prezentacja obydwu etapów działania archiwizatora algorytmu MOSF jest przedstawiona na rysunku 3.2. Poniżej natomiast omówione są wszystkie trzy parametry kontrolujące ich działanie, wraz z sugerowanymi wartościami domyślnymi:



(a) Poszukiwanie k-najbliższych sąsiadów.

(b) Analiza skupień k-średnich.

Rysunek 3.2: Przykładowy wynik działania archiwizatora algorytmu MOSF na losowo wygenerowanym zbiorze danych. Każdy znacznik odpowiada jednemu z 75 elementów archiwum. Zadaniem było przycięcie jego rozmiaru do $\frac{1}{3}$ tej liczby. Na rysunku a zaznaczone są punkty wskazane przez algorytm k-najbliższych sąsiadów jako te, które znajdują się w obszarach ich najmniejszej gęstości. Rysunek b przedstawia natomiast wynik analizy skupień k-średnich, której początkowym zbiorem centroidów były te punkty. Wynik ten stanowi względnie dokładną reprezentację danych wejściowych w postaci rozkładu zbliżonego do jednorodnego, zachowującego informację o gęstych skupiskach oraz izolowanych rozwiązaniach.

- Rozmiar archiwum (a) powinien wynosić przynajmniej tyle, ile jest cząstek w całej rodzinie rojów. Choć nie ma tu dolnego ograniczenia, nie należy go za bardzo zmniejszać, gdyż może to powodować zbyt szybką wymianę zapamiętanych rozwiązań, wprowadzającą niepotrzebne zamieszanie w ruchu cząstek, lub zwracać niedokładną reprezentację bieżącego przybliżenia zbioru Pareto.

- Liczba sąsiadów (n) nie musi być zbyt wysoka, gdyż najważniejsze dla miary zagęszczenia są punkty położone najbliżej każdego punktu. Ponieważ bardziej istotny od dokładności jest czas ich poszukiwania, domyślnie przyjmuje się $n = 10$. Dla dużych archiwów wartość ta może być jeszcze niższa.

- Liczba iteracji (s) również wynosi domyślnie 10, co wynika stąd, że największe przemieszczenia centroidów następują w trakcie pierwszych kroków analizy skupień. Z drugiej strony, stosowany tu algorytm k-średnich ma liniową złożoność obliczeniową ze względu na wszystkie zmienne, tak więc nie ma powodu aby zmniejszać wartość tego parametru ze względów ekonomicznych.

Dominacja zbiorów rojów cząstek

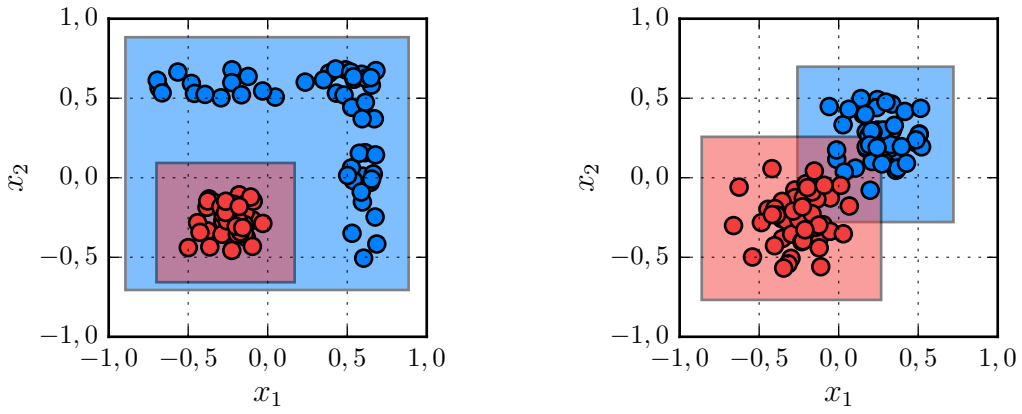
Dominacja, łączenie oraz dzielenie są trzema opcjonalnymi składowymi pojedynczej iteracji algorytmu MOSF. Działają one na wyższym poziomie od generacji i archiwizacji rozwiązań niezdominowanych, gdyż zarządzają całymi rodzinami rojów cząstek, nie interesując się tym w jaki sposób rozwiązania te są przez odkrywane i zapamiętywane. Zamiast tego, zajmują się porównywaniem ze sobą zawartości przechowujących je archiwów i podejmowaniem na tej podstawie stosownych akcji. Ponieważ czynności te charakteryzują się względnie wysoką złożonością obliczeniową, dominacja i łączenie są wykonywane odpowiednio co t_d i t_m iteracji, natomiast dzielenie następuje tylko na polecenie użytkownika, typowo po zakończeniu optymalizacji przez wszystkie roje, dlatego nie posiada własnego parametru w zbiorze Φ.

Po wygenerowaniu i zapamiętaniu nowych położeń cząstek, wszystkie rozwiązania zapisane w archiwum danej rodziny rojów są względem siebie niezdominowane. Dominacja jest procedurą przekształcającą ów „lokalny" status na „globalny", czyli wyrażany w sensie wszystkich rodzin. Realizacja tego zadania polega na usunięciu z każdego archiwum punktów zdominowanych przez inne archiwa. Rodziny rojów, które w ten sposób całkowicie utracą swoich liderów zewnętrznych są eliminowane z algorytmu MOSF. Puste archiwum oznacza bowiem, że korzystające z jego zawartości cząstki prawdopodobnie utknęły w izolowanym obszarze względnie wysoko położonych na krajobrazie wartości funkcji F minimów lokalnych, przez co przestały być pomocne w dalszych poszukiwaniach optymalnego zbioru Pareto. Ich usunięcie pozwala więc na uniknięcie niepotrzebnych obliczeń. Problem w tym, że dominacja jest czynnością kosztowną, w związku z czym wykonuje się ją tylko co t_d iteracji. Inny powód stosowania tego opóźnienia został już wspomniany w części dotyczącej archiwizatora. Mianowicie, zbyt częste usuwanie „globalnie" zdominowanych rozwiązań prowadzi do ograniczeń w ruchu cząstek, a także wiąże się z ryzykiem przedwczesnego wyeliminowania słabiej przystosowanych w danym momencie rojów. Z drugiej strony, dominacja umożliwia wskazywanie tym rojom, gdzie, według innych rodzin, znajdują się rozwiązania bliższe optymalnemu zbiorowi Pareto, a przez to stymulowanie ich do poruszania się w jego stronę. Jedną z głównych zalet algorytmu PSO jest zdolność do uciekania z pułapek minimów lokalnych, dlatego należy pozwolić mu działać przynajmniej przez chwilę, nawet wtedy, gdy znajduje się w zdominowanym lub niedopuszczalnym obszarze. W tym celu potrzebne jest zachowanie równowagi, osiągane poprzez odpowiedni dobór wartości parametru t_d, ustalanej obecnie na 10% maksymalnej liczby iteracji przewidzianej na całą procedurę optymalizacji.

Łączenie rodzin rojów cząstek

Inną cechą algorytmu MOSF, która wyróżnia go na tle pozostałych metod optymalizacji wielokryterialnej jest możliwość przeprowadzania automatycznej „analizy skupień" odnalezionych rozwiązań niezdominowanych bez potrzeby znajomości *a priori* docelowej liczby skupisk. Realizacja tego zadania staje się możliwa poprzez utożsamienie ich z archiwami rodzin rojów cząstek. W ten sposób mogą one poruszać się w przestrzeni rozwiązań i łączyć ze sobą jeżeli spełnione zostaną określone warunki. Działanie to przypomina swoim sposobem funkcjonowania aglomeracyjny algorytm analizy skupień. Najważniejszą różnicę pomiędzy tymi podejściami stanowią czynniki decydujące o tworzeniu się hierarchii skupisk. Pierwszy z nich dotyczy odległości w przestrzeni rozwiązań: dwie rodziny rojów mogą zostać połączone w jedną jeżeli zapamiętane przez nie rozwiązania znajdują się dostatecznie blisko siebie. Drugi jest natomiast związany z dotarciem w pobliże tych samych punktów przez należące do nich cząstki, wskazującym, że doprowadziło je tam podążanie za coraz niższymi wartościami kryteriów $f_1, \ldots, f_k$. Dzięki temu, uzyskana zostaje dodatkowa, niedostępna po zakończeniu optymalizacji informacja o rozkładzie danych w zbiorze i froncie Pareto. Pozwala to również na zmniejszanie liczby rodzin rojów, które w przeciwnym razie pracowałyby w tych samych miejscach.

Spełnienie drugiego warunku łączenia skupisk implikuje spełnienie pierwszego. W związku z tym sprawdzenie ich odbywa się w tym samym czasie. Dwa archiwa są uznawane za położone odpowiednio blisko siebie wtedy, gdy elementy jednego znajdują się w przestrzeni rozwiązań wśród elementów drugiego i odwrotnie. Najprostsze rozwiązanie, czyli bezpośrednie porównanie ich zawartości jest niepraktyczne, ponieważ nie można zakładać, że różne roje odwiedzą kiedykolwiek dokładnie te same punkty. Obliczanie odległości pomiędzy tymi punktami w sensie metryki euklidesowej byłoby natomiast zbyt kosztowne i wymagało normalizacji zapamiętanych rozwiązań. Nie stanowi ona problemu podczas archiwizacji, ponieważ poszukiwani są tam kolejni najbliżsi sąsiedzi, którzy mają znaczenie wyłącznie w lokalnym kontekście. Łączenie jest jednak procedurą globalną, dlatego musi być kontrolowana przez parametr mający zastosowanie do wszystkich rodzin rojów. Jest nim ostatni element zbioru Φ – r. Ma on postać d-wymiarowego wektora rozdzielczości, o który wydłużane są we wszystkich kierunkach równoległe do osi układu współrzędnych boki najmniejszych hiperprostokątów opisanych na archiwach. Połączenie ich może nastąpić wtedy, gdy bryły te będą się przecinać. Wartość parametru r jest typowo ustalana jako ułamek rozmiaru obszaru, w którym rozpoczęła się optymalizacja (domyślnie 10%).



(a) Sytuacja pierwsza – połączenie rodzin rojów niemożliwe: brak elementów z obydwu archiwów w części wspólnej.

(b) Sytuacja druga – połączenie rodzin rojów możliwe: elementy z obydwu archiwów obecne w części wspólnej.

Rysunek 3.3: Graficzna prezentacja sposobu łączenia rodzin rojów w algorytmie MOSF. Przedstawione są tu dwie sytuacje, jakie mogą zdarzyć się w trakcie tej procedury. Każdy kwadrat odpowiada hiperprostokątowi opisanemu na jednym z archiwów, powiększonemu o wartość parametru rozdzielczości $r = [0{,}2, 0{,}2]$ (10%).

Warunek przecinania się hiperprostokątów nie jest jednak wystarczający. Jako pochodne rozkładu elementów archiwów, nie muszą bowiem dokładnie ich reprezentować. Istnieje przez to ryzyko zajścia sytuacji przedstawionej na rysunku 3.3a, gdzie zbiór o mniejszej objętości, pomimo tego, że jest oddalony od drugiego, znajduje się w całości w jego hiperprostokącie. Aby tego uniknąć, sprawdzane są także same punkty. Połączenie dwóch rodzin rojów następuje więc tylko wtedy, gdy przynajmniej jeden element archiwum każdego z nich znajdują się wewnątrz hiperprostokąta drugiego. Sytuację tę obrazuje rysunek 3.3b. Złożoność obliczeniowa tej procedury jest liniowa ze względu na liczbę rodzin, punktów oraz wymiarów przestrzeni rozwiązań.

Łączenie jest podobne do dominacji. Również tutaj rodziny rojów są usuwane, ale w ich miejscu pojawiają się nowe. Każda z nich jest inicjalizowana w hiperprostokącie swojego archiwum, z maksymalną prędkością proporcjonalną do r, choć otrzymuje tylko tyle cząstek, ile posiadały jej macierzyste rodziny. Powoduje to, że wszystkie z nich, należące do tego samego pokolenia, utworzonego w wyniku przeprowadzenia procedury dzielenia, pozostają identyczne pod względem swoich ustawień.

Ponieważ łączenie powinno być poprzedzane dominacją, przyjmuje się domyślną wartość parametru kontrolującego częstość jego wykonywana, t_m, za równą t_d.

Dzielenie rodzin rojów cząstek

Dzielenie rodzin rojów cząstek jest procedurą odwrotną do łączenia. Polega ona na utworzeniu ich nowego pokolenia na podstawie zawartości archiwów obecnego. Czynność jest wykonywana z dwóch powodów. Po pierwsze, ma za zadanie zwiększać precyzję optymalizacji poprzez ponowną inicjalizację algorytmu w mniejszych, badanych osobno podzbiorach przestrzeni rozwiązań. Drugim jest natomiast uzyskanie danych potrzebnych do przeprowadzenia „analizy skupień".

Opracowanie sposobu podziału rodzin rojów, umożliwiającego realizację powyższych zadań, wydaje się nieoczywiste. Trudno bowiem stwierdzić, które rozwiązania niezdominowane powinny być archiwizowane wspólnie, a które wymagają dedykowanego zestawu cząstek. Informacja na temat skupisk, jakie tworzyły wcześniej jest tracona podczas ich ewentualnych połączeń, czemu nie sprzyja archiwizator algorytmu MOSF, dodatkowo starający się je rozmieszczać względnie jednorodnie. W związku z tym, Autor rozprawy przyjął, że podejściem, które nie będzie wymagało podejmowania tego rodzaju decyzji będzie rozdzielenie elementów każdego archiwum pomiędzy tyle nowych rodzin rojów, ile wynosi liczba jego elementów. Każda rodzina otrzymuje więc jeden punkt. Zakładając, że większość z nich połączy się przy najbliższej okazji na powrót w większe skupiska, a także w celu uniknięcia nadmiaru obliczeń spowodowanego bardzo dużą liczbą cząstek, zamiast czekać t_m iteracji, natychmiast po podziale wykonywane jest dodatkowe łączenie. Dzięki temu, powstaje tylko tyle rodzin, na ile pozwala parametr r. Użytkownik decyduje o liczbie ich cząstek. Typowo wynosi ona mniej niż we wcześniejszym pokoleniu, ponieważ do przeszukiwania mniejszych podzbiorów przestrzeni rozwiązań nie są potrzebne tak duże roje jak te, które je wcześniej odkryły.

Jeżeli rodzin rojów powstałych w wyniku dzielenia ma być tyle co wszystkich zapamiętanych rozwiązań, złożoność obliczeniowa następującego po nim łączenia staje się kwadratowa, tak jak w aglomeracyjnym algorytmie analizy skupień. Z tego powodu, oraz z faktu, że powstają w tym momencie zupełnie nowe cząstki, wykonywanie dzielenia następuje tylko na żądanie użytkownika.

Standardowy scenariusz optymalizacji wielokryterialnej przy użyciu algorytmu MOSF jest następujący: na początku zostaje uruchomiona pojedyncza rodzina rojów. Po odnalezieniu przez nią przybliżenia optymalnego zbioru Pareto, jej archiwum ulega podziałowi. Pochodne rodziny rojów, składające się z mniejszej liczby cząstek, kontynuują optymalizację, ewentualnie łącząc się ze sobą („analiza skupień"). Zbiór ich archiwów stanowi końcowy wynik działania algorytmu.

3.2. Algorytm MOSF – porównanie

Algorytm MOSF został opracowany przez Autora niniejszej rozprawy w celu przeprowadzenia symulacji równoczesnego wpływu pól zewnętrznego i wewnętrznego na układ łańcuchów w eksperymencie kompleksowania. Algorytm ten nie jest jednak ograniczony do bioinformatyki, dlatego postanowiono sprawdzić jego przydatność również w ogólnych zastosowaniach.

Aby wykazać, że algorytm MOSF jest przydatnym narzędziem optymalizacji wielokryterialnej, posiadającym możliwości, których nie oferują obecne dostępne metody, ale jednocześnie zachowującym zbliżoną lub lepszą od nich sprawność w poszukiwaniu optymalnego zbioru Pareto, porównano zwracane przez niego wyniki z wynikami dwóch innych algorytmów: NSGA-II [309] oraz NSPSO [310].

NSGA-II stanowi *de facto* standardowy wzorzec w tego rodzaju badaniach, przez co odniesienie się do niego pozwala na stwierdzenie zalet i ograniczeń proponowanego podejścia [314]. NSPSO jest natomiast jego rojowym odpowiednikiem, realizującym rozwiązania opracowane z myślą o algorytmach genetycznych przy pomocy modelu zachowań cząstek. Z tego powodu, stanowi pewnego rodzaju „łącznik" pomiędzy tymi dwoma podejściami, ułatwiający porównanie opartych na nich algorytmach optymalizacji wielokryterialnej.

Podstawowe porównanie algorytmu MOSF z NSGA-II i NSPSO zostało wykonane przy pomocy czterech funkcji testowych, pozwalających na łatwe powielanie tego eksperymentu oraz ocenę wyników: jednej $\mathbb{R}^2 \to \mathbb{R}^2$, jednej $\mathbb{R}^6 \to \mathbb{R}^2$ oraz dwóch $\mathbb{R}^2 \to \mathbb{R}^3$. Funkcje te dobrano tak, aby dzięki różnym właściwościom problemów optymalizacyjnych uwidaczniały silne i słabe strony tych metod. W szczególności brano pod uwagę kwestie dotyczące nieciągłości, symetrii oraz nieprzewidywalnych relacji pomiędzy ich optymalnymi zbiorami i frontami Pareto [368].

Sprawdzono również, jak badane algorytmy radzą sobie w środowisku stacjonarnych kryteriów generowanych przez test ruchomych wierzchołków. Zdecydowano się na jego użycie w celu uzyskania lepszego wglądu w możliwości ich zastosowania do bardziej zaawansowanych problemów, których charakterystyka jest nieznana *a priori*. Choć przestrzeń rozwiązań generowanych kryteriów była ze względów praktycznych za każdym razem dwuwymiarowa, wykorzystano inne właściwości MPB pozwalające na modyfikowanie poziomu trudności zagadnień optymalizacyjnych: liczbę wierzchołków i kryteriów, a także możliwość ich szybkiego, wielokrotnego tworzenia (inny zestaw funkcji w każdej próbie). Zgodnie ze stanem wiedzy Autora rozprawy, generator ten został po raz pierwszy zastosowany w taki sposób właśnie w tej pracy.

3.2.1. NSGA-II i NSPSO

Opublikowany przez Deba i współpracowników [309], algorytm NSGA-II (non-dominated sorted genetic algorithm) jest aktualizacją wcześniejszej metody optymalizacji wielokryterialnej Srinivasa i Deba [369]. NSGA-II charakteryzuje się niższą od niej złożonością obliczeniową pojedynczej iteracji, wynoszącą $O\left(kn^2\right)$ zamiast $O\left(kn^3\right)$, gdzie k oznacza liczbę kryteriów, a n – rozmiar populacji, a także brakiem dodatkowych parametrów. Ze względu na swoją skuteczność i łatwość implementacji, metoda ta jest dostępna w wielu bibliotekach programistycznych [370, 371].

NSGA-II (oraz podobny do niego algorytm SPEA2 [372, 373]) stosuje elitarną strategię selekcji osobników, ocenianych przez funkcję przystosowania, której wartości są obliczane na podstawie dwóch miar: rankingu niezdominowania oraz niszowania. Pierwsza promuje punkty znajdujące się bliżej optymalnego frontu Pareto od pozostałych, natomiast druga – bardziej odległe od swoich najbliższych sąsiadów w przestrzeni wartości. Wspólnie, powodują one kierowanie populacji w stronę rozwiązań globalnie niezdominowanych, utrzymywanie odpowiedniego zasięgu ich poszukiwań oraz dążenie do uzyskania jednorodnego rozkładu wyników.

NSGA-II stosuje do ustawiania osobników w rankingu procedurę sortowania niezdominowanego. Dzieli ona populację na rozłączne podzbiory, których elementom nadawane są rangi – kolejne liczby naturalne. Wartość 1 odpowiada rozwiązaniom globalnie niezdominowanym, 2 – rozwiązaniom niezdominowanym wśród zdominowanych przez te o randze 1, i tak dalej, aż do uzyskania ciągu lokalnych frontów Pareto, coraz bardziej oddalających się od pierwszego. Niszowanie jest natomiast realizowane poprzez przypisywanie każdemu osobnikowi sumy odległości pomiędzy jego dwoma najbliższymi sąsiadami we wszystkich wymiarach przestrzeni wartości. Suma ta przybliża rozmiar największego hiperprostokąta, który zawiera wyłącznie tego osobnika, zastępując stosowany wcześniej parametr rozmiaru niszy (σ_{share}).

W algorytmie NSGA-II archiwum odnalezionych rozwiązań niezdominowanych stanowi sama populacja. Podczas selekcji wybierane są w pierwszej kolejności osobniki o najniższych rangach (bliższych optymalnemu frontowi Pareto), a następnie o większym rozproszeniu w przestrzeni wartości. Jak wykazały badania, elitarne podejście przyspiesza zbieżność oraz zapobiega eliminacji dobrych rozwiązań [374].

W 2014 roku ukazała się nowa wersja tego algorytmu, NSGA-III [375, 376], usprawniająca jego działanie w przypadku większej liczby kryteriów niż trzy, a niedługo później – wersja ujednolicona: U-NSGA-III [377], umożliwiająca również optymalizację globalną. Nie jest jeszcze jednak dostępny jej kod źródłowy.

Algorytm NSPSO [310] stosuje te same mechanizmy poszukiwania rozwiązań niezdominowanych co NSGA-II, ale realizuje je za pomocą roju cząstek. Wszystkie równania aktualizacji ich właściwości z rozdziału 2.4 zostały pozostawione w nim bez zmian. Zmianie uległ natomiast sposób wyboru liderów oraz pojawił się nowy mechanizm „ewolucji" populacji.

Liderzy w algorytmie NSPSO są wybierani losowo spośród wybranego podzbioru cząstek wskazanych przez funkcję przystosowania zaadaptowaną z metody NSGA-II, na przykład, najlepszych 5%. Po przemieszczeniu, tworzony jest zbiór zawierający zarówno ich nowe położenia jaki i wektory pamięci. Do następnej iteracji algorytmu przechodzą losowo wybrane, niezdominowane elementy tego zbioru. Jeżeli jest ich mniej niż powinna wynosić liczba cząstek w roju, dołączane są do nich pozostałe (zdominowane), w kolejności wartości funkcji przystosowania. Li zaproponował również dodatkowy mechanizm usuwania skupisk punktów we froncie Pareto polegający na reinicjalizacji cząstek o największej gęstości sąsiadów. Przeprowadzone eksperymenty nie wykazały jednak istotnego wpływu tej procedury na sprawność poszukiwania rozwiązań niezdominowanych przez algorytm [310].

Porównania algorytmów NSPSO i NSGA-II [310] wykazało, że zwracają one podobne pod względem dokładności wyniki, co oznacza, że są wobec siebie mocno konkurencyjne, a także, że PSO stanowi tak samo przydatną jak algorytmy genetyczne podstawę do tworzenia sprawnych narzędzi optymalizacji wielokryterialnej.

3.2.2. Funkcje testowe

Podczas wyboru funkcji testowych do porównania algorytmów MOSF, NSGA-II i NSPSO kierowano się krytyką ze strony Okabe i współpracowników [378], którzy wskazują, że spora część z nich posiada ciągły, liniowy zbiór Pareto, a odpowiadający mu front jest regularny, najczęściej w kształcie wypukłej lub wklęsłej krzywej lub powierzchni. Ponieważ nie należy zakładać, że rzeczywiste problemy optymalizacyjne będą charakteryzować się jakąkolwiek prawidłowością, postanowiono skupić się na reprezentacyjnej grupie czterech funkcji, w przypadku których relacja pomiędzy zbiorem i frontem Pareto jest nietrywialna. Jedna została zaproponowana przez Autora rozprawy, a pozostałe trzy zaczerpnięto z pracy Coello Coello i współpracowników [379]. Poniżej przedstawione są ich równania oraz krótkie charakterystyki, natomiast wizualizacje optymalnych zbiorów i frontów Pareto zostały w celu poprawy czytelności tekstu przeniesione do dodatku A.1. Tam też znajdują się rysunki rozkładów miar oceny wyników działania porównywanych tu algorytmów.

Region	$\mathbf{x_1}$	$\mathbf{x_2}$	$\mathbf{x_3}$	$\mathbf{x_4}$	$\mathbf{x_5}$	$\mathbf{x_6}$
AB	5	1	$[1, 5]$	0	5	0
BC	5	1	$[1, 5]$	0	1	0
CD	$[4{,}06, 5]$	$[0{,}68, 1]$	1	0	1	0
DE	0	2	$[1, 3{,}73]$	0	1	0
EF	$[0, 1]$	$[1, 2]$	1	0	1	0

Tabela 3.1: Regiony funkcji Osyczka 2 zawierające jej optymalny zbiór Pareto.

Funkcja testowa $\mathbf{F_1}$ – Banach 1

Funkcja $F_1 : \mathbb{R}^2 \to \mathbb{R}^2$ (Banach 1), zaproponowana przez Autora rozprawy, posiada zbiór Pareto złożony z pięciu koncentrycznych okręgów wyśrodkowanych w początku układu współrzędnych, których promienie zwiększają się za każdym razem o 2π (rysunek A.1a). Odpowiadający mu front jest ciągły i wypukły (rysunek A.1b).

$$\begin{aligned} F_1(x) &= \left[f_1(x), f_2(x) \right] \\ f_1(x) &= \cos|x| \\ f_2(x) &= \sin|x| \\ g_1(x) &= |x| - 10\pi \leq 0 \\ x_1, x_2 &\in [-10\pi, 10\pi] \end{aligned} \tag{3.4}$$

Choć funkcja ta ma proste równanie, trudność w jej optymalizacji sprawia wielomodalność tworzących ją kryteriów. Powodują one, że każdy z pięciu okręgów z których składa się zbiór Pareto, a nawet ich fragmenty, odpowiadają całemu frontowi. W związku z tym, do odnalezienia wszystkich z nich nie wystarczy tylko poleganie na analizie zawartości przestrzeni wartości.

Funkcja testowa $\mathbf{F_2}$ – Osyczka 2

Funkcja $F_2 : \mathbb{R}^6 \to \mathbb{R}^2$ (Osyczka 2 [380]) posiada zbiór Pareto składający się z pięciu regionów przedstawionych w tabeli 3.1. Z powodu wysokiej liczby optymalizowanych zmiennych, jest on widoczny na rysunku A.2a w przestrzeni pierwszych dwóch składowych głównych. Odpowiadający mu front Pareto tworzy natomiast pięć fragmentów wypukłej łamanej (A.2b). Punkt A odpowiada wartości minimum globalnego kryterium f_1, a punkt F – wartości minimum globalnego kryterium f_2.

$$
\begin{aligned}
F_2(x) &= \left[f_1(x), f_2(x)\right] \\
f_1(x) &= -\left(25(x_1-2)^2 + (x_2-2)^2 + (x_3-1)^2 + (x_4-4)^2 + (x_5-1)^2\right) \\
f_2(x) &= x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 + x_6^2 \\
g_1(x) &= x_1 + x_2 - 2 \geq 0 \\
g_2(x) &= 6 - x_1 - x_2 \geq 0 \\
g_3(x) &= 2 + x_1 - x_2 \geq 0 \\
g_4(x) &= 2 - x_1 + 3x_2 \geq 0 \\
g_5(x) &= 4 - (x_3-3)^2 - x_4 \geq 0 \\
g_6(x) &= (x_5-3)^2 + x_6 - 4 \geq 0 \\
x_1, x_2, x_6 &\in [0, 10] \\
x_3, x_5 &\in [1, 5] \\
x_4 &\in [0, 6]
\end{aligned}
\tag{3.5}
$$

Funkcja Osyczki reprezentuje kryteria optymalizacyjne o wysokiej liczbie zmiennych i funkcji ograniczeń. Jednych i drugich jest tutaj 6. Przecięcie tych ostatnich, rozmieszczone w różnych częściach przestrzeni rozwiązań, wyznacza położenie optymalnego zbioru Pareto wśród rozwiązań niedopuszczalnych. Najtrudniejszym zadaniem dla algorytmów optymalizacyjnych jest dotarcie do każdego z tych fragmentów.

Funkcja testowa F_3 – Viennet 3

Funkcja $F_3 : \mathbb{R}^2 \to \mathbb{R}^3$ (Viennet 3 [381]) posiada zbiór Pareto składający się z trzech rozłącznych powierzchni o różnych kształtach i rozmiarach (rysunek A.3a). Odpowiadający mu front jest natomiast złożoną z kilku odcinków krzywą, wijącą się w różnych kierunkach w przestrzeni wartości (rysunek A.3b).

$$
\begin{aligned}
F_3(x) &= \left[f_1(x), f_2(x), f_3(x)\right] \\
f_1(x) &= \tfrac{1}{2}\left(x_1^2 + x_2^2\right) + \sin\left(x_1^2 + x_2^2\right) \\
f_2(x) &= \frac{(3x_1 - 2x_2 + 4)^2}{8} + \frac{(x_1 - x_2 + 1)^2}{27} + 15 \\
f_3(x) &= \frac{1}{x_1^2 + x_2^2 + 1} - 1{,}1\exp\left(-x_1^2 - x_2^2\right) \\
x_1, x_2 &\in [-3, 3]
\end{aligned}
\tag{3.6}
$$

W trzeciej funkcji Vienneta nie jest zauważalna prosta relacja pomiędzy zbiorem a frontem Pareto. W szczególności, utrudnia to odnalezienie podzbioru pierwszego z nich, znajdującego się najbliżej punktu $[3, 3]$ – odpowiada mu bowiem niewielki fragment frontu w pobliżu maksimum wartości kryterium f_1.

Funkcja testowa $\mathbf{F_4}$ – Viennet 4

Funkcja $\mathrm{F}_4 : \mathbb{R}^2 \to \mathbb{R}^3$ (Viennet 3 [381]) posiada zbiór Pareto składający się z powierzchni połączonej z odcinkiem (rysunek A.4a). Odpowiadający mu front jest natomiast powierzchnią przechodzącą w krzywą (rysunek A.4b).

$$
\begin{aligned}
F_4(x) &= \left[f_1(x), f_2(x), f_3(x) \right] \\
f_1(x) &= \frac{(x_1 - 2)^2}{2} + \frac{(x_2 + 1)^2}{13} + 3 \\
f_2(x) &= \frac{(x_1 + x_2 - 3)^2}{175} + \frac{(2x_2 - x_1)^2}{17} - 13 \\
f_3(x) &= \frac{(3x_1 - 2x_2 + 4)^2}{8} + \frac{(x_1 - x_2 + 1)^2}{27} + 15 \\
g_1(x) &= 4x_1 + x_2 - 4 < 0 \\
g_2(x) &= -x_1 - 1 < 0 \\
g_3(x) &= x_1 - x_2 - 2 < 0 \\
x_1, x_2 &\in [-4, 4]
\end{aligned}
\tag{3.7}
$$

W czwartej funkcji Vienneta trzy funkcje ograniczeń przycinają optymalny zbiór Pareto w taki sposób, aby jego kształt przypominał wielokąt, a także powodują silniejszą ekspozycję wystającego z niego odcinka, przyciągającą algorytmy optymalizacyjne. Symbol ostrej nierówności nie ma tu istotnego znaczenia.

Test ruchomych wierzchołków

Dzięki temu, że test ruchomych wierzchołków pozwala na szybkie generowanie kryteriów o różnych kształtach, sprawdzono jak badane algorytmy radzą sobie ze zwiększającymi się ich liczbami (2, 3, 4 i 5). We wszystkich przypadkach stosowane były następujące ustawienia: 50 wierzchołków, f = gauss, $Xmax = 100$, $Wmax = 100$, $Hmax = 100$. Pozostałe ustawienia były bez znaczenia, ponieważ nie wykonywano aktualizacji generatora. Dzięki funkcji kształtu wierzchołków gauss nie było również potrzeby stosowania funkcji krajobrazu bazowego (b).

3.2.3. Funkcje oceny

Algorytmy optymalizacji wielokryterialnej mają następujące trzy zadania:

1. minimalizację odległości pomiędzy elementami wynikowego frontu ($\mathcal{PF}$) i zbioru ($\mathcal{PS}$) Pareto, a ich optymalnymi odpowiednikami: $\mathcal{PF}^*$, i $\mathcal{PS}^*$,
2. maksymalizację liczby odnalezionych elementów $\mathcal{PF}^*$ i $\mathcal{PS}^*$,
3. maksymalizację odległości pomiędzy odnalezionymi elementami $\mathcal{PF}^*$.

W celu dobrania odpowiednich miar pozwalających na sprawdzenie, jak algorytmy MOSF, NSGA-II i NSPSO wywiązały się z powyższych zadań, odwołano się do innej pracy Okabe i współpracowników [382]. Na jej podstawie przyjęto, że istotne znaczenie powinien mieć brak potrzeby podawania dodatkowych parametrów, a także niezależność od liczby zmiennych, kryteriów i zwracanych rozwiązań. Autor rozprawy zdecydował się w związku z tym na użycie czterech miar: dwóch wprowadzonych przez siebie i dwóch stosowanych w innych publikacjach [379].

Miara approximation distance – set (ADS)

Miara approximation distance – set (ADS), zaproponowana przez Autora rozprawy, ocenia jak dobrze wynik optymalizacji przybliża rozkład elementów optymalnego zbioru Pareto. Dzieje się tak wtedy, gdy punkty z pierwszego z nich znajdują się w przestrzeni rozwiązań jak najbliżej punktów z drugiego:

$$\mathrm{ADS} = \sum_{i=1}^{|\mathcal{PS}^*|} \|\mathcal{PS}_i^* - \mathcal{PS}_j\| \tag{3.8}$$

gdzie j jest indeksem elementu $\mathcal{PS}$, który znajduje się najbliżej $\mathcal{PS}_i^*$ w sensie metryki euklidesowej. Obydwa te zbiory są skalowane do przedziału $[0, 1]$ na podstawie zawartości drugiego z nich. Dzięki temu, że analiza odbywa się z perspektywy optymalnego zbioru Pareto, miara ADS pozwala na porównywanie wyników zwracanych przez różne algorytmy niezależnie od liczby odnalezionych przez nie rozwiązań niezdominowanych. Inną zaletą tej miary jest wskazywanie czy rozwiązania z $\mathcal{PS}$ dobrze reprezentują rozkład elementów $\mathcal{PS}^*$. Ich brak w pobliżu jakiegoś jego fragmentu będzie bowiem skutkować szybkim wzrostem sumy odległości.

Miara ADS osiąga swoje minimum wtedy, gdy $\mathcal{PS} \equiv \mathcal{PS}^*$. Oznacza to, że nawet jeśli wynik optymalizacji jest niezdominowany względem podzbioru optymalnego zbioru Pareto, ale zawiera inne punkty, wartość tej miary będzie dodatnia.

Miara approximation distance – front (ADF)

Miara approximation distance – front (ADF), również zaproponowana przez Autora rozprawy, jest odpowiednikiem miary ADS w przestrzeni wartości:

$$\mathrm{ADF} = \sum_{i=1}^{|\mathcal{PF}^*|} \|\mathcal{PF}_i^* - \mathcal{PF}_j\| \tag{3.9}$$

gdzie j jest indeksem elementu $\mathcal{PF}$, który znajduje się najbliżej $\mathcal{PF}_i^*$ w sensie metryki euklidesowej. Tak jak w przypadku miary ADS, obydwa te zbiory są skalowane do przedziału $[0, 1]$ na podstawie zawartości drugiego z nich.

Miara ADF jest podobna do miary GD (generational distance) Van Veldhuizena [383]. Różnica pomiędzy nimi polega na tym, że odległości są tutaj obliczane z perspektywy optymalnego frontu Pareto, zamiast wyniku optymalizacji. Pozwala to na ocenę pokrycia $\mathcal{PF}^*$ niezależnie od liczby elementów $\mathcal{PF}$, a dzięki stosowaniu sumy zamiast średniej – również jednorodności ich rozkładów.

Miara error ratio (ER)

Miara error ratio (ER) odpowiada ułamkowi liczby elementów wyniku optymalizacji zdominowanych przez optymalny front Pareto [383]:

$$\mathrm{ER} = \frac{1}{|\mathcal{PF}|} \sum_{i=1}^{|\mathcal{PF}|} \begin{cases} 1 & \exists j \in \{1, \ldots, |\mathcal{PF}^*|\} : \mathcal{PF}_j^* \prec \mathcal{PF}_i \\ 0 & \nexists j \in \{1, \ldots, |\mathcal{PF}^*|\} : \mathcal{PF}_j^* \prec \mathcal{PF}_i \end{cases} \tag{3.10}$$

Miara ER jest powszechnie stosowana do oceny wyników optymalizacji wielokryterialnej, jednak, jako bardzo radykalna, może być myląca, gdyż nie zwraca informacji na temat rozkładu elementów $\mathcal{PF}$ i jego odległości od $\mathcal{PF}^*$ [382].

Miara niche count (NC)

Ostatnia miara, niche count (NC), służy do szacowania jednorodności rozkładu elementów wynikowego przybliżenia frontu Pareto, wyrażanego w postaci odchylenia standardowego liczby ich sąsiadów znajdujących się w sferycznej niszy o stałym promieniu [384]. Ponieważ promień ten musi zależeć od badanego problemu [382], postanowiono ustalić jego wartość na 0,05 oraz skalować $\mathcal{PF}$ do przedziału $[0, 1]$, ale na podstawie zawartości $\mathcal{PF}^*$. Po tej modyfikacji, miara ta staje się uzupełnieniem do miary ADF. Z powodu stosowania średniej, jest ona jednak wrażliwa na punkty odstające, przez co jej wartości mogą być w niektórych przypadkach mylące.

3.2.4. Wyniki porównania

Wszystkie funkcje testowe były optymalizowane przez każdą z porównywanych metod 100 razy, stosując domyślne wartości ich parametrów. W przypadku algorytmu MOSF oznaczało to: $\phi_v \in U(0,1)$, $\phi_m = 2$, $\phi_l = 1$, $\phi_e = 1$ oraz graf siatki (von Neumanna) jako topologia roju. Pozostałe parametry są omówione poniżej.

Rozmiar pojedynczego roju należącego do początkowej rodziny algorytmu MOSF ustalono na 64 cząstki. Dla NSGA-II i NSPSO liczba ta była mnożona przez liczbę optymalizowanych kryteriów. Dzięki temu, podczas pojedynczej iteracji, każdy algorytm mógł sprawdzić tyle samo rozwiązań.

Aby wszystkie trzy algorytmy miały równe szanse, można ustalić maksymalną liczbę obliczeń wartości optymalizowanych kryteriów. Nie ma wówczas znaczenia w jaki sposób oraz ile iteracji zostanie przez te algorytmy wykonanych. Jednak ze względu na różnice w ich implementacji, w szczególności dotyczące takich kwestii jak przechowywanie obliczonych wartości w pamięci podręcznej, nie można zagwarantować, że będzie tak w praktyce. Ponieważ przyjęto tu identyczne rozmiary populacji, postanowiono rozwiązać ten problem poprzez zrównanie również liczby iteracji. Jedyną kwestią pozostało wówczas tylko dwuetapowe wykonanie algorytmu MOSF. Nie jest bowiem wiadome ile pochodnych rodzin rojów zostanie utworzonych oraz ile z nich przetrwa do końca obliczeń. Dlatego był on za każdym razem uruchamiany jako pierwszy. W pierwszej kolejności wykonywano 64 aktualizacji jego początkowej rodziny rojów. Następnie rodzina ta była dzielona stosując wartość promienia r równą 10% długości boków początkowego hiperprostokąta, po czym zliczano ile rozwiązań mogły zbadać cząstki należące do jej rodzin pochodnych w trakcie kolejnych 64 iteracji. Rozmiar każdego roju na tym etapie wynosił 16, natomiast rozmiar archiwum był bez zmian (64 razy liczba kryteriów). Wartości parametrów t_d i t_m zostały ustawione na 12. W ten sposób można było ustalić liczbę iteracji pozostałych dwóch metod na taką samą jak w algorytmie MOSF, z dokładnością do rozmiaru populacji.

Po przeprowadzeniu wszystkich prób optymalizacji zostały obliczone dla otrzymanych wyników średnie i odchylenia standardowe oraz minimalne i maksymalne wartości wszystkich czterech funkcji oceny (ADS, ADF, ER, NC). Oprócz tego, odnotowano również liczby rodzin algorytmu MOSF, które pozostały na końcu jego działania, a także czas trwania obliczeń. Choć czas ten nie stanowi wiążącej oceny sprawności działania porównywanych metod, pozwala na ocenę możliwości ich implementacji, czyli stwierdzenie, które z nich mogą być wydajnie zaprogramowane.

Liczba elementów przybliżeń optymalnych zbiorów Pareto

Elementy referencyjnych przybliżeń optymalnych zbiorów Pareto funkcji testowych zostały wyznaczone poprzez wyczerpujące próbkowanie przestrzeni rozwiązań za pomocą siatki o gęstości 200 punktów w każdym wymiarze. Wyjątek stanowiła funkcja Osyczka 2, gdzie liczba ta dotyczyła każdego z pięciu jej regionów. Po usunięciu rozwiązań niedopuszczalnych, pozostało w nich łącznie 995 punktów. Liczby elementów referencyjnych zbiorów Pareto funkcji Vienneta wyniosły natomiast odpowiednio 552 i 1665, a funkcji Banach 1 – 8039.

W każdej ze 100 prób, generator MPB tworzył dla całej trójki algorytmów nowy problem optymalizacyjny. Ze względu na losowy kształt krajobrazów wartości jego kryteriów oraz występujące na nich wielokrotne minima lokalne, wraz ze wzrostem ich liczby, pojawiało się coraz więcej „niezgodności" pomiędzy nimi, powodujących przeistaczanie ich zbiorów Pareto z formy kilku rozłącznych skupisk lub odcinków w jedno duże skupisko, co sugeruje poniższa statystyka:

- 2 kryteria: średnio 121 rozwiązań ($\sigma = 54$, min $= 9$, max $= 251$),
- 3 kryteria: średnio 962 rozwiązania ($\sigma = 473$, min $= 103$, max $= 2324$),
- 4 kryteria: średnio 3116 rozwiązań ($\sigma = 1191$, min $= 1002$, max $= 7275$),
- 5 kryteriów: średnio 6936 rozwiązań ($\sigma = 2239$, min $= 2090$, max $= 12725$).

Liczba sprawdzonych rozwiązań i rodzin rojów algorytmu MOSF

Podstawowa liczba rozwiązań, które mogły sprawdzić wszystkie trzy algorytmy optymalizacyjne była równa iloczynowi liczb osobników (64), iteracji (64) oraz kryteriów (od 2 do 5). Tyle różnych punktów mogła bowiem odwiedzić początkowa rodzina rojów algorytmu MOSF. Końcowa ich liczba była natomiast zależna od tego, ile rodzin pochodnych zostało utworzonych i ile z nich dotrwało do końca procedury optymalizacji. Szczegóły znajdują się w tabelach 3.2 i 3.3.

Wiedząc ile punktów mogło być łącznie sprawdzonych przez algorytm MOSF, ustalano jak długo mają w danej próbie działać pozostałe dwie metody, z dokładnością do rozmiaru populacji. Dzięki temu, przeprowadzone porównanie było sprawiedliwe pod względem możliwości przeszukiwania przez nie przestrzeni rozwiązań.

Warto zauważyć, że podejście to działało na korzyść algorytmów NSGA-II i NSPSO, ponieważ mogły pracować tak samo długo jak MOSF, ale bez potrzeby ponownej inicjalizacji rojów, potrzebnej mu do wykonania „analizy skupień".

Funkcja	Średnia	Odch. std.	Minimum	Maksimum
Banach 1	1,0	0,0	1,0	1,0
Osyczka 2	2,6	0,7	1,0	5,0
Viennet 3	1,0	0,0	1,0	1,0
Viennet 4	1,0	0,0	1,0	1,0
MPB 2	2,9	1,1	1,0	5,0
MPB 3	3,3	1,2	1,0	6,0
MPB 4	2,3	1,2	1,0	6,0
MPB 5	1,4	0,7	1,0	4,0

Tabela 3.2: Liczba wynikowych rodzin rojów algorytmu MOSF.

Funkcja	Podstawa	Średnia	Odch. std.	Minimum	Maksimum
Banach 1	8192	10382	538	10240	14464
Osyczka 2	8192	31505	3148	22496	40032
Viennet 3	12288	20981	1153	18432	21504
Viennet 4	12288	15360	0	15360	15360
MPB 2	8192	15055	2738	10240	24576
MPB 3	12288	27009	4010	18432	36864
MPB 4	16384	36113	7556	20480	61440
MPB 5	20480	36204	8847	25600	56320

Tabela 3.3: Liczba sprawdzonych rozwiązań przez algorytm MOSF.

Z tabeli 3.2 można odczytać, że algorytm MOSF uznał, że w rozdzielczości 10% rozmiaru początkowego hiperprostokąta, funkcje Banach 1 i Viennet 4 posiadają optymalne zbiory Pareto złożone tylko z jednego fragmentu. W przypadku drugiej jest to zgodne z rzeczywistością, natomiast rozwiązania niezdominowane wskazywane przez pierwszą są ułożone w koncentrycznych pierścieniach, które nie są wykrywalne przez algorytm k-średnich oraz nierozróżnialne w przestrzeni wartości. Są one jednak położone na tyle blisko siebie, że mogą być uznane za jedno duże skupisko.

Warto w tym miejscu zwrócić również uwagę na fakt, że odchylenie standardowe liczby sprawdzonych rozwiązań funkcji Viennet 4 jest równe 0 (tabela 3.3). Oznacza to, że pochodne rodziny rojów łączyły się w jedną natychmiast po ich utworzeniu. Wytłumaczenie tego zjawiska jest takie, że algorytm MOSF podczas pierwszej fazy swojego działania zawsze rozmieszczał elementy archiwum w sposób zbliżony do jednorodnego, tak, że każdy z nich posiadał sąsiadów we wszystkich wymiarach przestrzeni rozwiązań, w promieniu wyznaczonym przez parametr rozdzielczości.

Elementy optymalnego zbioru Pareto trzeciej funkcji Vienneta tworzą trzy skupiska, choć rodziny pochodne algorytmu MOSF za każdym razem łączyły się w jedną. Wynika to stąd, że kryteria z tej funkcji prowadzą cząstki pomiędzy tymi skupiskami, w szczególności dwoma większymi, znajdującymi się bliżej punktu $[-1, -1]$.

W funkcji Osyczki również występują trzy skupiska rozwiązań globalnie niezdominowanych, składające się z trzech odcinków i dwóch płaszczyzn. Algorytm MOSF najczęściej wskazywał, że izolowane od siebie są dwa z tych skupisk, co wynika stąd, że jedno z nich jest przedłużeniem innego, w którego stronę kieruje cząstki kryterium f_1. Wszystkie pochodne rodziny rojów połączyły się tylko raz. Tak samo, tylko raz pozostało ich pięć. Oznacza to, że wyniki również dla tej funkcji potwierdzają zdolność algorytmu MOSF do uzyskiwania informacji na temat rozkładu danych w przestrzeni rozwiązań zgodnej z jego rzeczywistą charakterystyką.

Na koniec pozostał tylko generator MPB. Przy dwóch lub trzech kryteriach tworzonych na podstawie 50 wierzchołków i funkcji gauss, optymalny zbiór Pareto tworzyło zazwyczaj kilka rozłącznych, owalnych zbiorów punktów, najczęściej występujących w izolowanych grupach, lub – rzadziej – w postaci pojedynczego odcinka, ułożonego skośnie do osi układu współrzędnych. Algorytm MOSF zawsze prawidłowo kończył optymalizację w drugim z tych przypadków z jedną rodziną rojów. Przy czterech i pięciu kryteriach pojawiało się coraz więcej „niezgodności" pomiędzy nimi, czego efektem było powiększanie się skupisk rozwiązań niezdominowanych oraz łączenie ich w jeden duży zbiór o skomplikowanym kształcie. Występowanie tego zjawiska potwierdzają zmniejszające się wartości średnie w dolnych wierszach tabeli 3.2.

Czas trwania optymalizacji

Statystyka czasu wykonywania optymalizacji wielokryterialnej przez algorytmy MOSF, NSGA-II i NSPSO znajduje się w tabeli 3.4. Podczas każdej próby miały one swobodny dostęp do dedykowanego im rdzenia procesora.

Ponieważ ważniejsze od czasów trwania optymalizacji (zależnych od liczby iteracji, liczby elementów optymalnego zbioru Pareto oraz rozwiązywanego problemu optymalizacyjnego) są relacje pomiędzy tymi czasami, dane w tabeli 3.4 zostały przedstawione w procentach. Istotne znaczenie miało również to, że algorytmy NSGA-II i NSPSO zostały dostarczone przez tę samą bibliotekę. Ponieważ korzystają z identycznej funkcji przystosowania i chwilowo zwiększają dwukrotnie rozmiar swojej populacji w każdej iteracji, powodu różnic w czasie ich działania należy poszukiwać w innych sposobach użycia tej funkcji przez algorytmy genetyczne i roje cząstek.

Z tabeli 3.4 można odczytać, że przeciętne różnice w czasie działania pomiędzy algorytmami NSGA-II i NSPSO w przypadku funkcji F_1, F_3 i F_4 wyniosły zaledwie kilka procent na korzyść tego drugiego. Inaczej wyglądała ta relacja podczas optymalizacji pozostałych kryteriów – tutaj NSPSO uzyskał średnio do około 20% przewagi. Może to wynikać z prostszego sposobu tworzenia nowych populacji lub przechowywania wyników w pamięci podręcznej roju. Funkcja F_2 ma bowiem aż 6 funkcji ograniczeń, a do obliczenia wartości kryteriów MPB dla pojedynczego rozwiązania niezbędne było sprawdzenie wszystkich wierzchołków, których liczba wynosiła za każdym razem 50. Przechowanie wyników odpowiadających wektorom pamięci cząstek może więc pozwolić na oszczędność czasu trwania pojedynczej iteracji.

Bieżąca implementacja algorytmu MOSF przechowuje obliczone wartości optymalizowanych kryteriów w archiwach rodzin rojów, natomiast cząstki, którymi się posługuje nie korzystają z pamięci podręcznej. Autor rozprawy podjął taką decyzję w celu umożliwienia stosowania tych rojów w optymalizacji dynamicznej.[1] Pomimo tego, algorytm MOSF był w stanie sprawdzić tyle samo kandydatów na rozwiązania niezdominowane co NSGA-II i NSPSO najczęściej w czasie od 10 do ponad 30 procent krótszym. Ponieważ nie skupia się on na przestrzeni wartości, najbardziej wymagającym pod względem czasu elementem jego działania było odnajdywanie rozwiązań niezdominowanych w archiwach rodzin rojów.

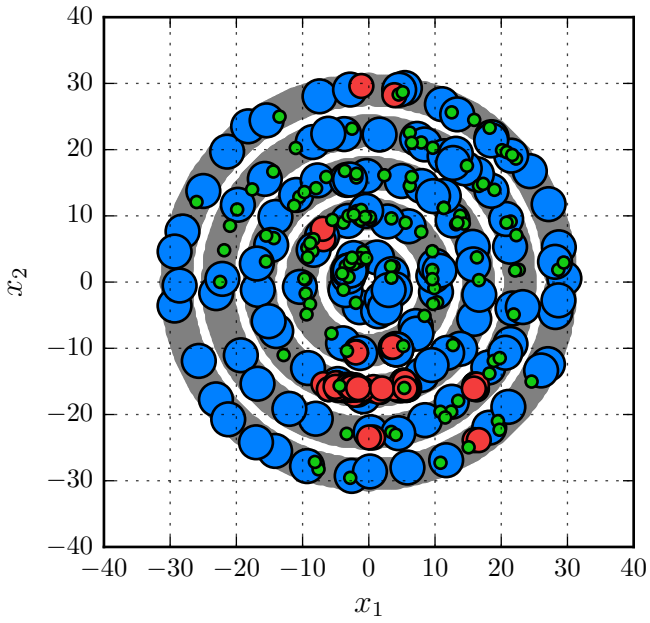
Choć dzielenie rodzin rojów jest czasochłonne i skaluje się kwadratowo wraz z rozmiarem archiwum (podobnie jak w algorytmie hierarchicznym analizy skupień), wykonywane było jednak tylko raz w każdej próbie. Z drugiej strony, dominacja rodzin, przeprowadzana co $t_d = 12$ iteracji, polega na sprawdzaniu wszystkich ich par. Czynność ta wymaga większego zaangażowania niż w przypadku analizy pojedynczego archiwum, dlatego jeżeli w danym momencie uruchomiona była duża liczba rojów, mogło to prowadzić do chwilowych spowolnień działania algorytmu. Potwierdzeniem występowania tej sytuacji są najwyższe średnie w wierszach tabeli 3.3 odpowiadające funkcjom MPB 2 i MPB 3, zbiegające się z wartościami ułamków MOSF/NSGA-II i MOSF/NSPSO w tabeli 3.4 przekraczającymi wartość 1.

Najważniejszy wniosek płynący z powyższej analizy jest taki, że algorytm MOSF może być wydajnie zaprogramowany i tak samo dobrze nadaje się pod tym względem czasu działania do rozwiązywania problemów wielokryterialnych jak powszechnie stosowane metody optymalizacyjne.

[1] Podczas eksperymentu kompleksowania białek pamięć podręczna była jednak stosowana, ale na poziomie kryteriów optymalizacyjnych, a nie rojów cząstek. Dzięki temu oraz obsłudze funkcji ograniczeń przy pomocy strategii turniejowej Deba możliwe było znaczne zmniejszenie (około sześciokrotne) czasu trwania symulacji bez negatywnego wpływu na otrzymane wyniki.

Funkcja		$\frac{\textbf{MOSF}}{\textbf{NSGA-II}}$	$\frac{\textbf{MOSF}}{\textbf{NSPSO}}$	$\frac{\textbf{NSPSO}}{\textbf{NSGA-II}}$
Banach 1	μ	82,1	84,3	97,7
	σ	3,0	5,2	7,6
	$\downarrow$	72,2	48,4	75,7
	$\uparrow$	98,6	104,0	169,3
Osyczka 2	μ	75,5	90,1	83,9
	σ	3,5	4,1	3,5
	$\downarrow$	63,4	76,9	71,6
	$\uparrow$	97,1	113,3	98,1
Viennet 3	μ	70,8	70,4	100,6
	σ	7,2	7,0	2,7
	$\downarrow$	42,0	51,1	82,2
	$\uparrow$	118,1	117,5	104,5
Viennet 4	μ	73,1	73,8	99,2
	σ	2,0	3,1	4,6
	$\downarrow$	62,2	57,3	78,6
	$\uparrow$	76,8	84,4	131,5
MPB 2	μ	100,3	129,9	77,5
	σ	4,9	12,3	3,3
	$\downarrow$	79,5	107,5	52,4
	$\uparrow$	122,9	234,5	89,6
MPB 3	μ	86,4	109,7	79,0
	σ	4,5	9,6	3,8
	$\downarrow$	74,1	86,4	59,3
	$\uparrow$	110,3	186,1	95,2
MPB 4	μ	74,7	93,5	80,1
	σ	3,0	4,8	3,8
	$\downarrow$	66,3	67,8	74,9
	$\uparrow$	82,7	104,6	110,6
MPB 5	μ	65,1	82,3	79,3
	σ	3,2	5,2	4,7
	$\downarrow$	56,5	54,6	59,2
	$\uparrow$	78,1	101,7	118,8

Tabela 3.4: Czas wykonania optymalizacji przez porównywane algorytmy. Wszystkie wartości są podane w procentach jako ułamki postaci: wynik dla algorytmu 1 dzielony przez wynik dla algorytmu 2. Wyjaśnienie symboli: μ – średnia σ – odchylenie standardowe, $\downarrow$ – minimum, $\uparrow$ – maksimum.



Rysunek 3.4: Przykładowy wynik optymalizacji funkcji Banach 1 (przestrzeń rozwiązań). Kolory znaczników wskazują na algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Optymalny zbiór Pareto ma kolor szary.

Wyniki optymalizacji funkcji testowych

Statystyki oceny wyników optymalizacji wybranych funkcji testowych i generowanych przez MPB znajdują się w tabelach 3.5 i 3.6, natomiast wykresy indywidualnych wartości wszystkich czterech miar – w rozdziale A.1. Na podstawie rodzaju zwracanej przez nie informacji, przyjęto, że najwyższą rangę należy nadać ADS i ADF, natomiast pozostałe dwie potraktować jako ich uzupełnienie.

W przypadku funkcji Banach 1, algorytm MOSF uzyskał zdecydowanie najniższe wartości miary ADS, rozmieszczając elementy archiwum we wszystkich pierścieniach tworzących optymalny zbiór Pareto w sposób zbliżony do jednorodnego, co widać na rysunku 3.4. Podobnie, choć w tym sensie prawie dwukrotnie słabiej poradził sobie z tym zadaniem algorytm NSPSO, natomiast NSGA-II skupił się wyłącznie na punktach bardzo dobrze reprezentujących front Pareto, ale niewystarczających do przedstawienia ich rozkładu w przestrzeni rozwiązań.

Funkcja F_1 jest przykładem skutecznego i konsekwentnego odnajdywania oraz archiwizacji rozwiązań niezdominowanych przez algorytm MOSF.

Miara		MOSF	NSGA-II	NSPSO
ADS	μ	**250,564**	1541,967	414,497
	σ	**20,466**	432,116	95,206
	$\downarrow$	**214,621**	828,897	292,054
	$\uparrow$	**338,381**	2762,005	735,653
ADF	μ	54,738	**32,918**	50,336
	σ	6,357	**1,185**	5,194
	$\downarrow$	44,799	**30,560**	40,946
	$\uparrow$	79,905	**35,915**	66,400
ER	μ	**0,000**	**0,000**	0,003
	σ	**0,000**	0,002	0,004
	$\downarrow$	**0,000**	**0,000**	**0,000**
	$\uparrow$	**0,000**	0,016	0,016
NC	μ	3,108	**1,615**	2,767
	σ	0,707	**0,198**	0,457
	$\downarrow$	2,083	**1,185**	1,895
	$\uparrow$	6,282	**2,119**	3,943

(a) Banach 1 (rozkłady na rysunku A.5)

Miara		MOSF	NSGA-II	NSPSO
ADS	μ	535,606	**298,433**	498,726
	σ	**79,649**	203,032	272,509
	$\downarrow$	343,255	**29,834**	210,076
	$\uparrow$	**756,574**	1102,013	1408,516
ADF	μ	**30,369**	76,785	104,299
	σ	**6,035**	69,479	71,535
	$\downarrow$	18,756	**5,712**	14,389
	$\uparrow$	**47,606**	309,486	345,294
ER	μ	1,000	**0,809**	0,883
	σ	0,000	**0,112**	0,167
	$\downarrow$	1,000	0,539	**0,344**
	$\uparrow$	1,000	1,000	1,000
NC	μ	6,640	**2,456**	5,574
	σ	2,199	**0,776**	2,811
	$\downarrow$	2,315	**1,250**	2,586
	$\uparrow$	14,753	**4,451**	20,995

(b) Osyczka 2 (rozkłady na rysunku A.6)

Miara		MOSF	NSGA-II	NSPSO
ADS	μ	**5,324**	9,274	8,410
	σ	1,112	0,561	**0,441**
	$\downarrow$	**4,130**	8,253	7,378
	$\uparrow$	**8,368**	13,350	9,446
ADF	μ	5,796	**5,145**	5,292
	σ	0,782	0,789	**0,578**
	$\downarrow$	4,322	4,426	**4,047**
	$\uparrow$	8,698	12,381	**7,410**
ER	μ	0,197	0,140	**0,084**
	σ	0,045	**0,027**	**0,027**
	$\downarrow$	0,094	0,078	**0,026**
	$\uparrow$	0,417	0,208	**0,172**
NC	μ	23,289	**1,940**	10,258
	σ	6,543	**0,191**	2,558
	$\downarrow$	6,807	**1,530**	4,792
	$\uparrow$	43,017	**2,489**	17,000

(c) Viennet 3 (rozkłady na rysunku A.7)

Miara		MOSF	NSGA-II	NSPSO
ADS	μ	**37,062**	70,203	49,946
	σ	**1,325**	6,290	3,866
	$\downarrow$	**34,597**	58,716	41,656
	$\uparrow$	**40,044**	92,550	61,164
ADF	μ	**38,319**	67,586	49,651
	σ	**1,043**	5,405	3,144
	$\downarrow$	**35,811**	57,554	43,903
	$\uparrow$	**41,231**	85,072	59,056
ER	μ	0,072	0,050	**0,042**
	σ	**0,017**	0,018	**0,017**
	$\downarrow$	0,026	0,021	**0,000**
	$\uparrow$	0,109	0,109	**0,083**
NC	μ	**1,847**	2,456	3,370
	σ	**0,193**	0,252	0,758
	$\downarrow$	**1,432**	1,917	1,826
	$\uparrow$	**2,302**	3,257	5,622

(d) Viennet 4 (rozkłady na rysunku A.8)

Tabela 3.5: Statystyka wyników optymalizacji funkcji testowych. Mniej znaczy lepiej. Pogrubiony krój wskazuje na najniższe wartości w danym wierszu. Wyjaśnienie symboli: μ – średnia, σ – odchylenie standardowe, $\downarrow$ – minimum, $\uparrow$ – maksimum.

Miara		MOSF	NSGA-II	NSPSO
ADS	μ	**1,545**	3,878	3,915
	σ	**1,915**	5,705	5,459
	$\downarrow$	0,071	**0,052**	0,081
	$\uparrow$	**9,772**	33,786	30,537
ADF	μ	**1,248**	1,450	1,731
	σ	**1,763**	1,969	2,134
	$\downarrow$	0,047	**0,042**	0,046
	$\uparrow$	13,641	**12,201**	15,714
ER	μ	**0,144**	0,201	0,182
	σ	**0,112**	0,182	0,204
	$\downarrow$	**0,000**	**0,000**	**0,000**
	$\uparrow$	**0,680**	0,898	0,961
NC	μ	9,416	**2,648**	7,273
	σ	5,299	**1,626**	6,225
	$\downarrow$	2,080	**1,167**	1,964
	$\uparrow$	31,646	**10,949**	33,345

(a) MPB 2 (rozkłady na rysunku A.9)

Miara		MOSF	NSGA-II	NSPSO
ADS	μ	**17,267**	24,289	48,174
	σ	**18,309**	21,740	59,887
	$\downarrow$	**0,569**	1,625	4,207
	$\uparrow$	**119,941**	156,690	301,261
ADF	μ	37,211	**33,105**	42,000
	σ	35,891	**21,745**	24,620
	$\downarrow$	**1,345**	1,994	6,171
	$\uparrow$	214,267	**102,024**	111,950
ER	μ	**0,140**	0,198	0,252
	σ	**0,078**	0,082	0,137
	$\downarrow$	**0,016**	0,036	0,052
	$\uparrow$	**0,370**	0,464	0,693
NC	μ	9,120	**2,841**	6,512
	σ	8,301	**1,386**	6,566
	$\downarrow$	**1,245**	1,706	1,508
	$\uparrow$	43,744	**15,239**	42,966

(b) MPB 3 (rozkłady na rysunku A.10)

Miara		MOSF	NSGA-II	NSPSO
ADS	μ	**57,170**	85,874	89,128
	σ	**40,542**	63,537	59,480
	$\downarrow$	**8,656**	19,544	16,929
	$\uparrow$	**212,664**	353,724	289,011
ADF	μ	**161,303**	188,339	167,970
	σ	105,680	107,535	**83,183**
	$\downarrow$	**23,256**	39,463	45,374
	$\uparrow$	609,658	541,811	**456,957**
ER	μ	**0,207**	0,255	0,281
	σ	0,104	**0,082**	0,101
	$\downarrow$	**0,047**	0,109	0,074
	$\uparrow$	**0,559**	0,598	0,633
NC	μ	4,920	**2,236**	2,972
	σ	5,616	**0,448**	1,924
	$\downarrow$	**0,764**	1,305	1,013
	$\uparrow$	40,572	**3,530**	13,818

(c) MPB 4 (rozkłady na rysunku A.11)

Miara		MOSF	NSGA-II	NSPSO
ADS	μ	**126,922**	207,021	176,889
	σ	**69,689**	99,999	94,278
	$\downarrow$	**15,945**	22,589	42,312
	$\uparrow$	**413,526**	564,546	653,140
ADF	μ	**441,441**	623,801	497,547
	σ	219,180	271,170	**201,036**
	$\downarrow$	**50,809**	66,695	113,731
	$\uparrow$	1091,757	1375,628	**1062,347**
ER	μ	**0,246**	0,268	0,269
	σ	0,088	0,075	**0,073**
	$\downarrow$	**0,056**	0,113	0,141
	$\uparrow$	0,491	**0,478**	0,487
NC	μ	2,158	1,802	**1,559**
	σ	2,605	**0,395**	0,760
	$\downarrow$	**0,419**	1,173	0,694
	$\uparrow$	17,283	**3,473**	4,485

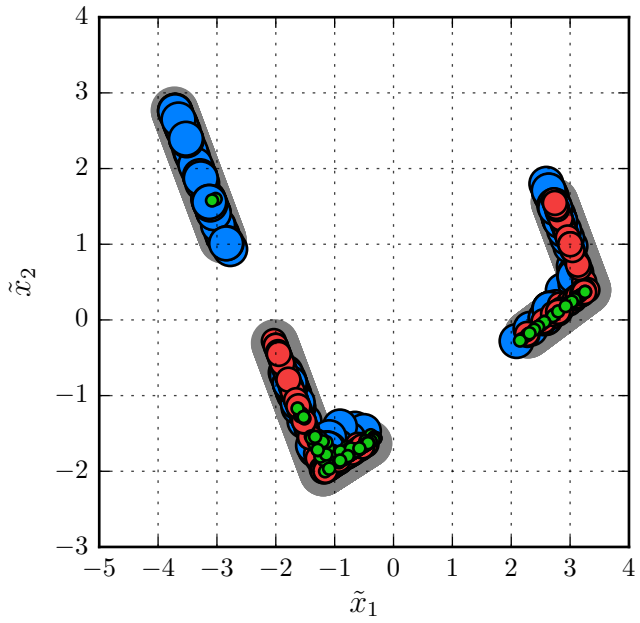
(d) MPB 5 (rozkłady na rysunku A.12)

Tabela 3.6: Statystyka wyników optymalizacji funkcji generowanych. Mniej znaczy lepiej. Pogrubiony krój wskazuje na najniższe wartości w danym wierszu. Wyjaśnienie symboli: μ – średnia, σ – odchylenie standardowe, $\downarrow$ – minimum, $\uparrow$ – maksimum.

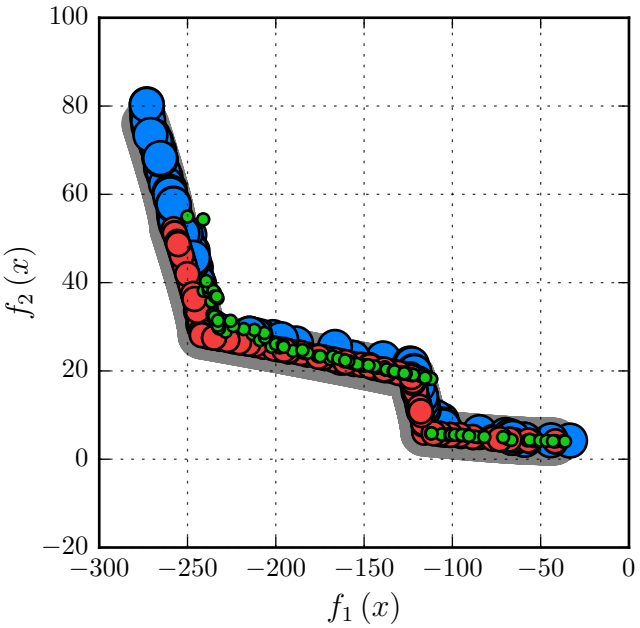
Najniższe wartości miary ADF w przypadku funkcji F_1, a przez to również NC, zaobserwowano dla algorytmu NSGA-II, co wskazuje na skuteczność tego podejścia w odnajdywaniu jednorodnej reprezentacji liniowego frontu Pareto. Algorytm MOSF nie interesuje się przestrzenią wartości, w związku z czym jego wyniki są w tej kwestii słabsze, lecz nadal porównywalne z NSPSO. Mówiąc inaczej, pokrywają one całość frontu Pareto, choć nie aż tak równomiernie jak w przypadku NSGA-II. Sugeruje to ewentualną potrzebę wprowadzenia dodatkowego czynnika przy wyborze liderów, promującego różnorodność położeń punktów w przestrzeni wartości. Warto również zauważyć, że w wynikowych archiwach algorytmu MOSF zawsze znajdowały się rozwiązania niezdominowane przez referencyjny zbiór Pareto.

W funkcji Osyczki sytuacja okazała się odwrotna. To algorytm NSGA-II uzyskał najniższe wartości miary ADS – różne średnio o około połowę od pozostałych dwóch metod. Wynik ten jest efektem głównie tego, że najłatwiej przychodziło mu odnajdywanie regionów CD i EF. Z drugiej strony, czasami omijał pozostałe, zwłaszcza AB lub DE, co widać na rysunku 3.5. Jednak nawet wtedy jego przeciętne ADS pozostawało najniższe. Przyczyny tej sytuacji należy więc szukać w rozmieszczeniu optymalnego zbioru Pareto w przestrzeni rozwiązań. Po wykonaniu rzutowania na jego pierwsze dwie składowe główne, najniższe wartości miary ADS uzyskał algorytm MOSF. Interesujące jest również to, że w przeciwieństwie do NSGA-II, miał on największą trudność w dotarciu właśnie do regionów CD i EF, co także widać na rysunku 3.5. Pomimo tego, jako jedyny odnajdywał za każdym razem wszystkie pięć regionów, łącząc je zazwyczaj w dwa lub trzy skupiska, dzięki czemu wartości miary ADF dla zwróconych przez niego wyników okazały się najniższe.

Minimum miary EC równe 1 w przypadku funkcji F_2 dla algorytmu MOSF nie wskazuje na problem w poruszaniu się pomiędzy rozwiązaniami niezdominowanymi lub obsłudze funkcji ograniczeń, ale w dokładnym osiągnięciu zawierających je odcinków. Pozostałe metody lepiej sobie z tym poradziły, pod warunkiem, że wcześniej dotarły w ich pobliże, co nie zawsze im się udawało. Wnioski z tej obserwacji są takie, że sortowanie niezdominowane jest skuteczniejsze w przybliżaniu liniowych fragmentów frontu Pareto, natomiast analiza gęstości punktów w przestrzeni rozwiązań – na wskazywaniu położenia, kształtu oraz liczby podzbiorów optymalnego zbioru Pareto. W związku z tym, uzyskanie najlepszych wyników powinno być możliwe dzięki połączeniu dwóch podejść: w pierwszej kolejności należałoby zastosować algorytm MOSF, który wyznaczyłby i podzielił na podzbiory wstępne przybliżenie zbioru Pareto, a następnie uruchomić algorytm NSGA-II lub NSPSO w tych podzbiorach, które osiągnęłyby ich dobrą reprezentację w przestrzeni wartości.

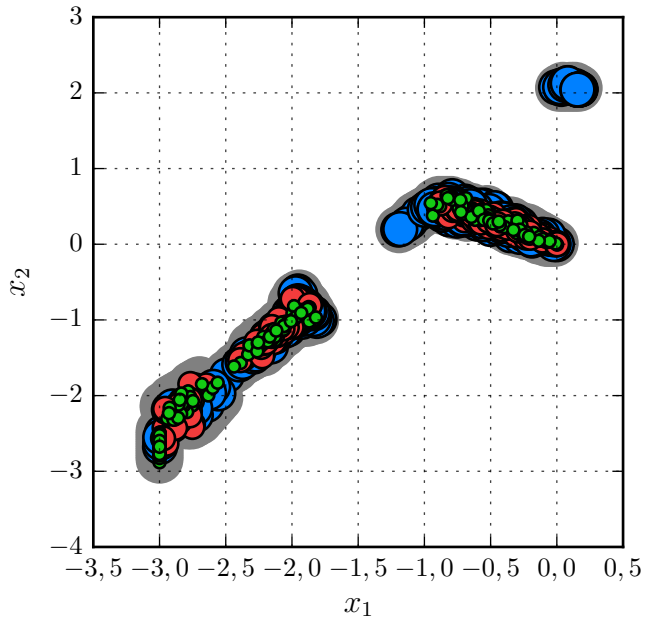


(a) Pierwsze dwie składowe główne przestrzeni rozwiązań.



(b) Przestrzeń wartości.

Rysunek 3.5: Przykładowy wynik optymalizacji funkcji Osyczka 2. Kolory znaczników wskazują na algorytm: niebieski – MOSF (wyniki w postaci jednego zbioru), czerwony – NSGA-II, zielony – NSPSO. Optymalny zbiór i front Pareto mają kolor szary.

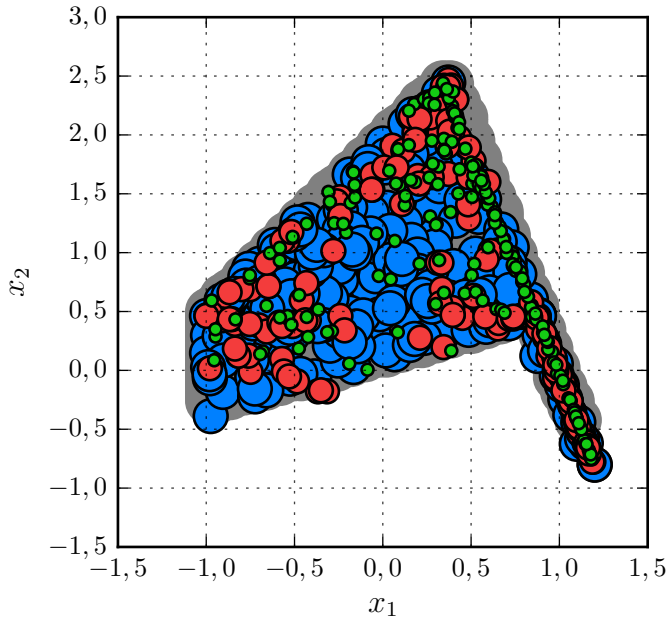


Rysunek 3.6: Przykładowy wynik optymalizacji funkcji Viennet 3 (przestrzeń rozwiązań). Kolory znaczników wskazują na algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Optymalny zbiór Pareto ma kolor szary.

Optymalny zbiór Pareto trzeciej funkcji Vienneta składa się z trzech skupisk o różnych kształtach. Algorytm MOSF najczęściej znajdował wszystkie z nich, natomiast pozostałe dwie metody miały problem z jednym – najmniejszym, a także krótkim, liniowym fragmentem drugiego (środkowego). Wykazały one również tendencję do dzielenia trzeciego skupiska na dwie części. Sytuację tę przedstawia rysunek 3.6.

Pomijanym przez algorytmy NSGA-II i NSPSO fragmentom zbioru Pareto funkcji F_3 odpowiada niewielki zwrot we froncie Pareto oraz środek odcinka znajdującego się w pobliżu maksimum wartości kryterium f_1. Dlatego ich średnie wartości miary ADF były porównywalne z MOSF, który uzyskał najwyższe pomimo najniższej średniej miary ADS. Wynikło to z jego pełnej, choć bardziej oddalonej od jednorodnej reprezentacji danych w przestrzeni wartości, będącej efektem tego, że kryterium f_1 prowadzi cząstki pomiędzy wszystkimi trzema skupiskami, powodując łączenie rodzin rojów w jedną. Co ciekawe, gdy algorytm MOSF nie docierał do najmniejszego skupiska, wartości miar ADF i NC zwracanych przez niego wyników malały.

Funkcja F_3 pokazuje, że wszystkie trzy metody dobrze radzą sobie z optymalizacją trzech kryteriów o nietrywialnej relacji pomiędzy ich zbiorem i frontem Pareto.



Rysunek 3.7: Przykładowy wynik optymalizacji funkcji Viennet 4 (przestrzeń rozwiązań). Kolory znaczników wskazują na algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Optymalny zbiór Pareto ma kolor szary.

Czwarta funkcja Vienneta również składa się z trzech kryteriów, których Pareto ma postać wygiętej powierzchni zamiast łamanej. Odnalezienie jej nie stanowi problemu, nawet pomimo stosowania trzech funkcji ograniczeń. Ich zadanie polega bowiem wyłącznie na nadawaniu optymalnemu zbiorowi Pareto oczekiwanego kształtu. Problem stanowi za to uzyskanie jego jednorodnej reprezentacji.

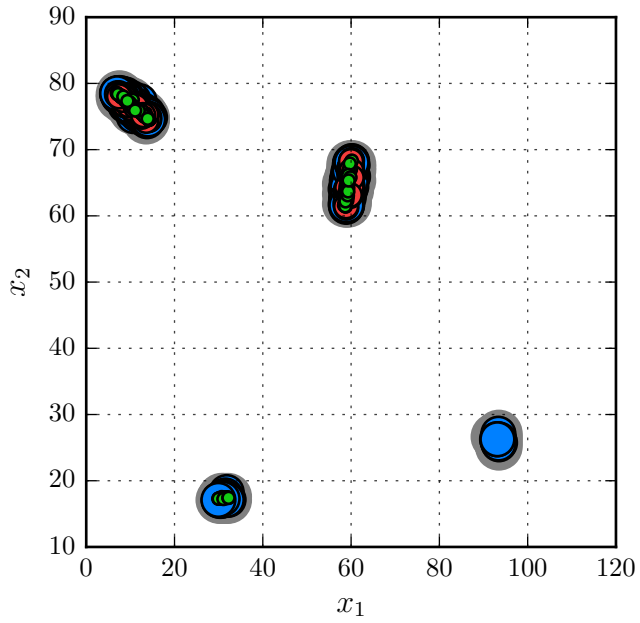
Wcześniejsze wyniki sugerują, że algorytmy oparte na sortowaniu niezdominowanym dobrze radzą sobie z liniowymi frontami Pareto. W przypadku funkcji F_4 zadziałało to jednak na ich niekorzyść: zarówno NSGA-II jak i NSPSO wykazały tendencję do skupiania się na niewielkim odcinku oraz przechodzącej w ten odcinek krawędzi zbioru Pareto, pomijając częściowo resztę jego zawartości (w szczególności środek), co widać na rysunku 3.7. Z drugiej strony, algorytm MOSF rozmieszczał zapamiętane przez siebie rozwiązania w sposób zbliżony do jednorodnego, uzyskując dzięki temu maksymalne wartości miar ADS i ADF, niższe nawet od ich minimów w przypadku pozostałych dwóch metod. Ponieważ kształty optymalnego zbioru i frontu Pareto tej funkcji są do siebie podobne, wartości miary NC dla algorytmu MOSF również okazały się najniższe spośród porównywanych metod.

Na koniec pozostała analiza wyników optymalizacji kryteriów generowanych przez MPB. W odróżnieniu od poprzednich czterech funkcji, do których algorytmy MOSF, NSGA-II i NSPSO podchodziły 100 razy, tutaj miały tylko jedną szansę na odnalezienie zbioru rozwiązań niezdominowanych danego zestawu. Dzięki stukrotnemu zwiększeniu liczby problemów optymalizacyjnych, możliwe było zaobserwowanie jak metody te radzą sobie w różnych, nieprzewidywalnych i niepozwalających na wcześniejsze przygotowanie się sytuacjach. Generacja wielu losowych kryteriów pozwoliła również na sprawdzenie czy ich wysoka liczba (powyżej trzech) istotnie wpływa na wyniki uzyskiwane przez porównywane algorytmy.

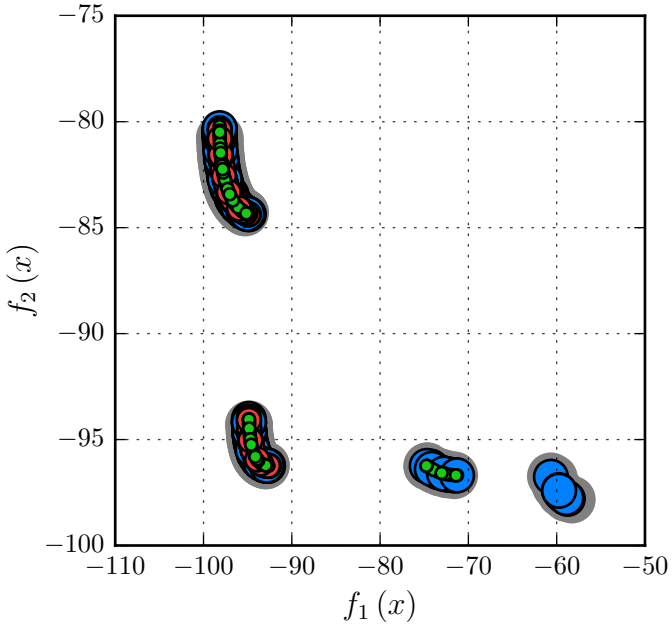
Ponieważ referencyjne przybliżenie optymalnego zbioru Pareto było za każdym razem wyznaczane przez próbkowanie względnie dużego podzbioru przestrzeni rozwiązań (co 0,5 jednostek), istniała możliwość umieszczenia w nim punktów fałszywie dodatnich. W związku z tym, jeżeli wyniki zwrócone przez wszystkie algorytmy dominowały łącznie więcej niż 50% jego zawartości, cała próba była ponawiana.

Optymalny zbiór Pareto funkcji wielokryterialnych tworzonych przez dwa lub trzy generatory MPB składał się zazwyczaj z kilku rozłącznych skupisk, mających najczęściej owalną postać, którym odpowiadał wypukły i rozłączny front Pareto. Przykład wraz z wynikami jego optymalizacji znajduje się na rysunku 3.8. Choć nakładające się na siebie znaczniki utrudniają zauważenie tego, tylko algorytm MOSF wskazał wszystkie cztery fragmenty widocznego na tym rysunku zbioru Pareto, NSPSO – trzy, a NSGA-II – dwa. Potwierdzeniem, że nie było to dziełem przypadku, ale konsekwencją skuteczności działania algorytmu MOSF, są najniższe wartości miar ADS, ADF i ER odpowiadające jego wynikom. Skupianie się na podzbiorze optymalnego frontu frontu Pareto umożliwiło natomiast algorytmowi NSGA-II ponownie uzyskanie rozkładu elementów jego przybliżenia najbliższego do jednorodnego, na co wskazuje miara NC.

Przy dwóch i trzech kryteriach, algorytm MOSF kończył optymalizację zazwyczaj z dwoma lub trzema pochodnymi rodzinami rojów. Przy czterech i pięciu następowało ich łączenie, najczęściej do dwóch lub jednej. Więcej losowo wygenerowanych, kryteriów oznacza, że rozwiązania niezdominowane będą zajmować coraz większy podzbiór w przestrzeni rozwiązań. Przykładowy wynik optymalizacji obrazujący to zjawisko jest przedstawiony na rysunku 3.9. Gdy $k \geq 10$, zbiór Pareto przybiera formę dużej, jednolitej „plamy”. Zadanie odnalezienia go w dwuwymiarowej przestrzeni rozwiązań staje się wówczas trywialne, przez co ustępuje miejsca zadaniu uzyskania jak najdokładniejszego rozkładu elementów jego przybliżenia. Ma to szczególne znaczenie wtedy, gdy ustalony został niewielki rozmiar populacji lub archiwum.

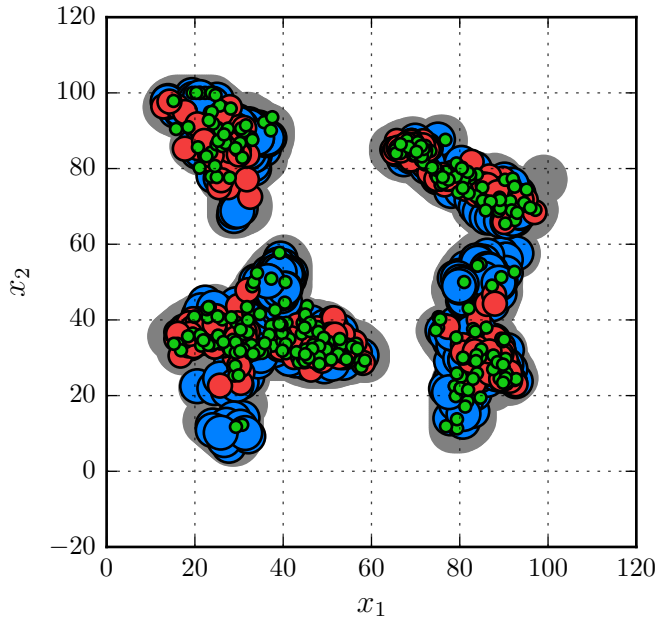


(a) Przestrzeń rozwiązań.

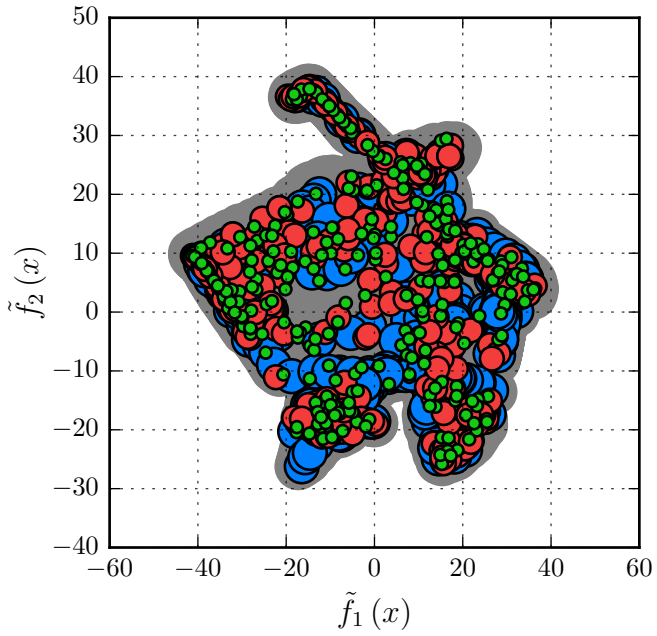


(b) Przestrzeń wartości.

Rysunek 3.8: Przykładowy wynik optymalizacji dwóch kryteriów wygenerowanych przez MPB. Kolory znaczników wskazują na algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Optymalny zbiór i front Pareto mają kolor szary.

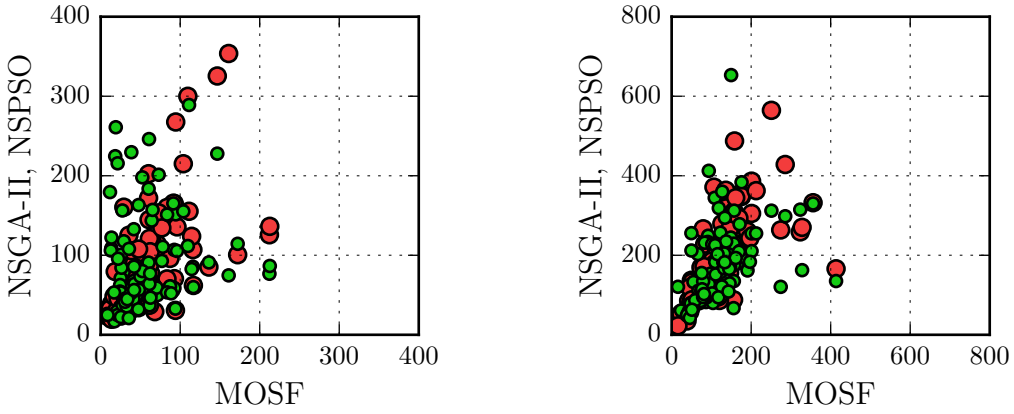


(a) Przestrzeń rozwiązań.



(b) Pierwsze dwie składowe główne przestrzeni wartości.

Rysunek 3.9: Przykładowy wynik optymalizacji pięciu kryteriów wygenerowanych przez MPB. Kolory znaczników wskazują na algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Optymalny zbiór i front Pareto mają kolor szary.



(a) Cztery kryteria.

(b) Pięć kryteriów.

Rysunek 3.10: Porównanie wartości miary ADS oceny wyników optymalizacji kryteriów wygenerowanych przez MPB i uzyskanych przez algorytmy MOSF i NSGA-II (kolor czerwony) oraz NSPSO (kolor zielony).

Algorytm MOSF uzyskał najniższe wartości wszystkich miar (poza NC) również w przypadku optymalizacji czterech i pięciu kryteriów wygenerowanych przez MPB. Pozostałe dwie metody osiągnęły słabsze wyniki. Ponownie widać, że ich priorytetem są liniowe fragmenty frontu Pareto. Wprowadzenie algorytmu (U-)NSGA-III miało za zadanie usprawnić rozmieszczanie wyników w przestrzeni wartości, ale nadal nie gwarantuje to, że ich odpowiedniki w przestrzeni rozwiązań będą równie dobrze reprezentować optymalny zbiór Pareto. Obecny brak jego implementacji w bibliotekach uniemożliwił jednak sprawdzenie tych hipotez w niniejszej rozprawie.

Na podstawie przeprowadzonej analizy można stwierdzić, że algorytm MOSF jest skutecznym narzędziem optymalizacji wielokryterialnej, pozwalającym na szybkie i dokładne wskazywanie niezdominowanych obszarów przestrzeni rozwiązań oraz automatyczne dzielenie ich na skupiska. Nie dla niego znaczenia liczba zmiennych oraz kryteriów, choć należy zwracać uwagę na wartość parametru rozdzielczości, od którego zależy ile pochodnych rodzin rojów zostanie utworzonych.

Warto jeszcze zwrócić uwagę na rozkłady miary ADS widoczne na rysunku 3.10. Sugerują one, że problemy optymalizacyjne wygenerowane przez MPB były zazwyczaj tak samo trudne lub łatwe do rozwiązania dla wszystkich trzech algorytmów (najmniej dla MOSF), co świadczy o przydatności tego generatora w tworzeniu problemów o zmiennym poziomie skomplikowania.

3.3. Modyfikacja modelu FOD

Zastosowanie modelu FOD jako kryterium optymalizacyjnego wymaga wykonywania wielokrotnych obliczeń rozkładów hydrofobowości w celu uzyskania odpowiadających im wartości RD. Do uzyskania efektywnej czasowo realizacji tego zadania, okazało się niezbędne wprowadzenie zmian w algorytmie układania atomów efektywnych białka zgodnie z osiami układu współrzędnych.

Przedstawiona poniżej modyfikacja modelu FOD została opracowana przez Autora rozprawy i polega na zastąpieniu algorytmu bazującego na średnicach przez analizę składowych głównych (PCA). Podstawową korzyścią z tego wynikającą jest zmniejszenie złożoności obliczeniowej procedury przygotowującej białko do obliczeń rozkładu $\tilde{H}t$ ze względu na liczbę reszt z liniowo-logarytmicznej (oczekiwanej, a w najgorszym przypadku kwadratowej) do liniowej. Podejście to posiada również inne zalety, przydatne z punktu widzenia pracy z modelem FOD:

1. uproszczenie teorii obliczeń (mniej algorytmów do wyjaśniania),
2. uproszczenie praktyki obliczeń (mniej algorytmów do implementacji),
3. zwiększenie dokładności uzyskiwanych wyników – maksymalizacja wariancji współrzędnych atomów efektywnych na podstawie całego ich zbioru zamiast najbardziej odległych par (mniejsza podatność na punkty skrajne),
4. uniezależnienie od użytkownika – brak potrzeby ręcznego wskazywania osi symetrii białek (pełna automatyzacja),
5. umożliwienie kolejnych udoskonaleń (w dalszych planach).

Zaproponowane zmiany dotyczą wyłącznie sposobu obliczeń hydrofobowości teoretycznej, gdyż przekształcenia afiniczne całości zbioru atomów efektywnych nie mają wpływu na rozkład $\tilde{H}o$. Procedura uzyskiwania wartości tego rozkładu jest obecnie uważana za optymalną i dlatego nie była modyfikowana. Wynika to stąd, że sprowadza się ona do poszukiwania przy użyciu drzewa k-d wszystkich sąsiadów atomów efektywnych w stałym promieniu $c = 9\,\text{Å}$ wokół nich.

Białka są obiektami trójwymiarowymi, tak więc utworzenie struktury drzewa k-d atomów efektywnych zajmuje $O(n \log n)$ czasu i $O(n)$ miejsca, gdzie n jest liczbą reszt. Złożoność obliczeniowa poszukiwania w drzewie k-d najbliższych sąsiadów pojedynczego punktu jest natomiast proporcjonalna do $\log n$, co przekłada się na $O(n \log n)$ w skali całej cząsteczki białka.

3.3.1. Obecna metoda – FOD-MAX

Do obliczania wartości rozkładu hydrofobowości teoretycznej służy w modelu FOD trójwymiarowa funkcja Gaussa. Aby móc ją zastosować, niezbędne jest wstępne ułożenie zbioru atomów efektywnych białka zgodnie z osiami układu współrzędnych w taki sposób, aby wariancja ich położeń w kolejnych wymiarach przestrzeni stała się jak największa. Obecna realizacja tego zadania polega na wykonaniu następujących trzech przekształceń:

1. translacji środka geometrycznego do początku układu współrzędnych,
2. obrotu w taki sposób, aby najdłuższa średnica stała się równoległa do osi X,
3. obrotu w taki sposób, aby średnica w rzucie na płaszczyznę YZ stała się równoległa do osi Y.

Pierwsza z powyższych czynności zeruje wartość oczekiwaną, natomiast pozostałe dwie starają się zmaksymalizować wariancję atomów efektywnych. Ponieważ podejście to bazuje na średnicach (odcinkach łączących pary najbardziej oddalonych od siebie punktów), zostało tu roboczo nazwane FOD-MAX. Wiążą się z nim następujące dwa problemy: efektywne obliczeniowo wyznaczenie średnic nie jest łatwym zadaniem, a ich układanie równolegle do osi układu współrzędnych nie musi maksymalizować wariancji atomów efektywnych zgodnie z oczekiwaniami.

Najprostsza, wyczerpująca realizacja metody FOD-MAX sprawdza wszystkie możliwe pary atomów efektywnych w czasie $O\left(n^2\right)$. Stosowane obecnie podejście charakteryzuje się niższą złożonością obliczeniową dzięki obserwacji, że dwa najbardziej oddalone od siebie punkty w danym zbiorze muszą należeć do jego otoczki wypukłej [317]. Użycie algorytmu Quickhull pozwala na jej wyznaczenie w czasie oczekiwanym $O\left(n \log n\right)$ [385]. Pozostaje wówczas do wyczerpującego sprawdzenia tylko $O\left(h^2\right)$ par, gdzie h jest liczbą atomów efektywnych należących do wynikowej otoczki.

Istotną wadą tego podejścia jest zależność danych wyjściowych od ich rozkładu w zbiorze wejściowym. Mówiąc inaczej, zysk wynikający z zastosowania algorytmu otoczki wypukłej następuje tylko wtedy, gdy h jest odpowiednio mniejsze od n. Dzieje się tak jednak w przypadku białek globularnych, posiadających wiele reszt w swoim wnętrzu, które mogą być w ten sposób pominięte bez negatywnych konsekwencji. Dobrze to widać na przykładzie struktur z bazy danych rozprawy: przeciętna wartość ułamka h^2/n^2 dla kompleksów wyniosła tutaj 3,19% ($\sigma = 1{,}19\%$), natomiast dla łańcuchów – 7,64% ($\sigma = 2{,}45\%$), co wskazuje na zmniejszenie maksymalnej liczby par atomów efektywnych do rozpatrzenia średnio o ponad 90%.

Przestawiona powyżej analiza dotyczy poszukiwań najdłuższej średnicy białka. Postępowanie w przypadku rzutu atomów efektywnych na płaszczyznę YZ jest bardzo podobne, z tą różnicą, że otoczka wypukła może być wyznaczona na podstawie wyniku poprzedniej, a więc w czasie oczekiwanym $O(h \log h)$. Zakładając, że składa się ona z l elementów, średnia wartość ułamka l^2/h^2 dla kompleksów z bazy danych rozprawy wyniosła 8,00% ($\sigma = 2{,}98\%$), a dla łańcuchów – 10,1% ($\sigma = 3{,}28\%$).

Zastosowanie algorytmu otoczki wypukłej ma na celu wyłącznie znaczne zmniejszenie liczby par atomów efektywnych, których odległości muszą być obliczone aby odnaleźć obydwie średnice ich zbioru. Warto jednak zauważyć, że istnieje alternatywne podejście do tego zagadnienia, obecnie nie stosowanie w modelu FOD, ale umożliwiające całkowite wyeliminowanie wyszukiwania wyczerpującego. W przestrzeni $\mathbb{R}^3$, może być ono zastąpione jedną z dwóch metod charakteryzujących się liniowo-logarytmicznym oczekiwanym czasem działania: stochastyczną Clarksona i Shora [386] lub deterministyczną Ramosa [387], a w przypadku rzutu na płaszczyznę – algorytmem Shamosa o liniowej złożoności obliczeniowej [388]. Rozwiązanie to nie unika jednak obliczeń otoczek wypukłych oraz jest bardziej skomplikowane pod względem programistycznym.

Po każdorazowym wyznaczeniu średnicy zbioru atomów efektywnych, następuje jego obrót w taki sposób, aby odcinek ten stał się równoległy do wskazanej osi układu współrzędnych: odpowiednio X lub Y. Przekształcenie to jest realizowane poprzez złożenie dwóch macierzy przekształceń Householdera, a więc bez potrzeby obliczania jakichkolwiek kątów. Pierwsza z nich odbija wektor różnicy końców danej średnicy na wektor pośredni należący do płaszczyzny obrotu, natomiast druga – z tego wektora na docelową oś. Szczegóły tej procedury są omówione w rozdziale 2.5.3. Wybór wektora pośredniego w trzech wymiarach nie ma tu istotnego znaczenia (o ile całe przekształcenie zostało wykonane prawidłowo) ze względu na następny obrót, który posiada jeden stopień swobody.

Wadą podejścia opartego na średnicach jest jego silna zależność od punktów skrajnych, które mogą niedokładnie reprezentować kierunki najwyższych wariancji w całym zbiorze atomów efektywnych. Staje się to szczególnie widoczne w przypadku symetrycznych cząsteczek, takich jak białko `1AON` [389]. Efekt zastosowania metody FOD-MAX do tego kompleksu jest widoczny na rysunku 3.11a. Aby osiągnąć oczekiwane przez model FOD ułożenie, czyli takie, w którym oś symetrii tej cząsteczki staje się równoległa do osi X, zachodzi potrzeba ingerencji ze strony użytkownika, polegająca na ręcznym wskazaniu najdłuższej średnicy [50]. Oznacza to, że metoda FOD-MAX nie pozwala na w pełni automatyczne obliczenia rozkładu $\tilde{H}t$.

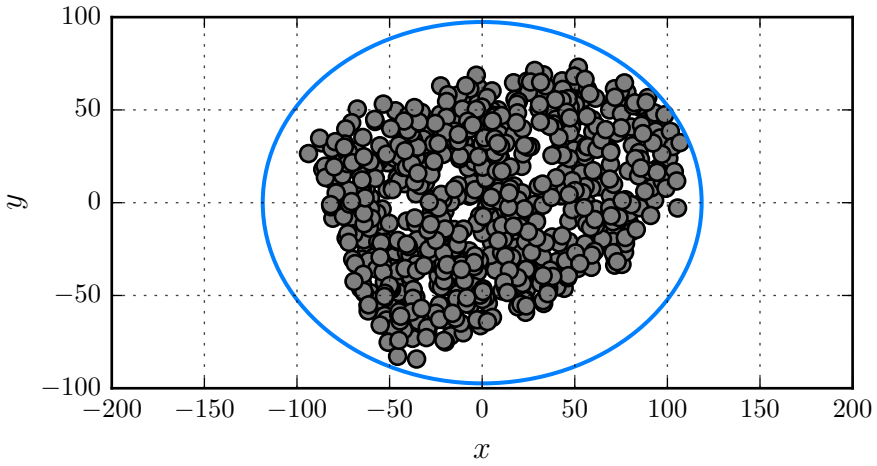
3.3.2. Nowa metoda – FOD-PCA

Opracowany przez Autora rozprawy, alternatywny sposób układania struktury białka zgodnie z osiami układu współrzędnych eliminuje wszystkie problemy wynikające ze stosowania metody FOD-MAX. Został on roboczo nazwany FOD-PCA, gdyż zamiast poszukiwania średnic zbioru atomów efektywnych, wykonuje analizę składowych głównych. Podejście to opiera się na geometrycznej interpretacji tej procedury, czyli dopasowaniu do danych elipsoidy, wzdłuż której kolejnych osi występuje największa zmienność ich położeń.

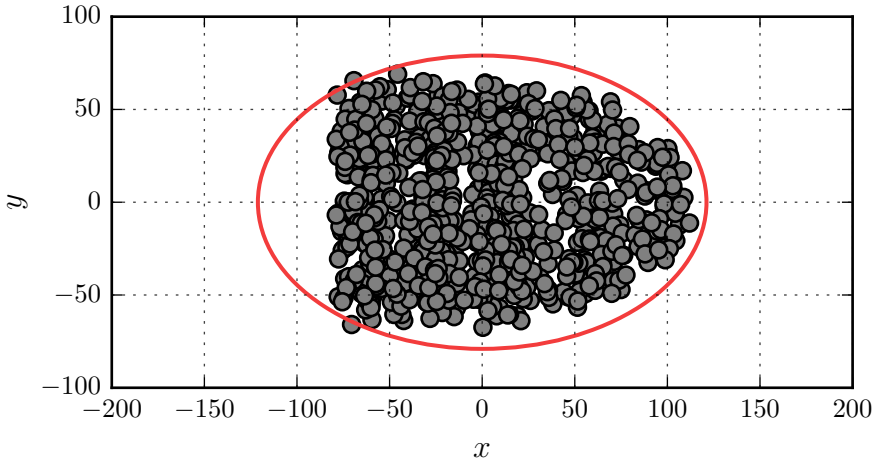
Algorytm analizy składowych głównych jest szczegółowo omówiony w rozdziale 2.5.4. Dane wejściowe dla niego stanowi macierz E o rozmiarze $n \times 3$, przechowująca położenia wszystkich atomów efektywnych, których zbiór został *a priori* wyśrodkowany w początku układu współrzędnych. Pomnożenie E^T przez E, a następnie podzielenie przez $n - 1$ daje macierz kowariancji C o 3 wierszach i 3 kolumnach. Składowe główne, czyli poszukiwane osie elipsoidy „kropli", są tożsame z wektorami własnymi tej macierzy. Rzutowanie na nie całego zbioru atomów efektywnych powoduje obrót białka w taki sposób, że ich wariancja staje się największa w pierwszym wymiarze przestrzeni, potem w drugim, a na końcu – w trzecim.

Metoda FOD-PCA realizuje to samo zadanie, co FOD-MAX, ale przy pomocy pojedynczego algorytmu, którego najbardziej skomplikowanym krokiem jest wykonanie rozkładu według wartości osobliwych (SVD). Nie stanowi to jednak problemu, ponieważ procedurę tę można odnaleźć w bibliotekach numerycznych większości popularnych języków programowania. Złożoność obliczeniowa metody FOD-PCA, wyrażona w odniesieniu do liczby reszt, jest więc taka sama jak mnożenia E^T przez E, czyli $O(n)$. Należy pamiętać, że macierz PCA może w niektórych przypadkach okazać się złożeniem obrotu i odbicia lustrzanego. Choć nie wpływa to na rozkłady hydrofobowości, musi być każdorazowo poprawione w celu zachowania identyczności kształtu oraz sensu biologicznego cząsteczek.

Dzięki temu, że analiza składowych głównych bierze pod uwagę wszystkie atomy efektywne, a nie samą otoczkę wypukłą ich zbioru, potrafi „zauważyć" symetrię struktur białkowych, co widać na rysunku 3.11b. Oznacza to, że metoda FOD-PCA jest zarówno szybka jak i dokładna, przez co stanowi przydatną modyfikację modelu FOD, usprawniającą obliczenia rozkładu hydrofobowości teoretycznej oraz eliminującą potrzebę angażowania w nie użytkownika. Aby móc przejść do dyskusji na temat możliwości zastąpienia przez nią metody FOD-MAX należy sprawdzić, jak bardzo różnią się od siebie wyniki zwracane przez te podejścia.

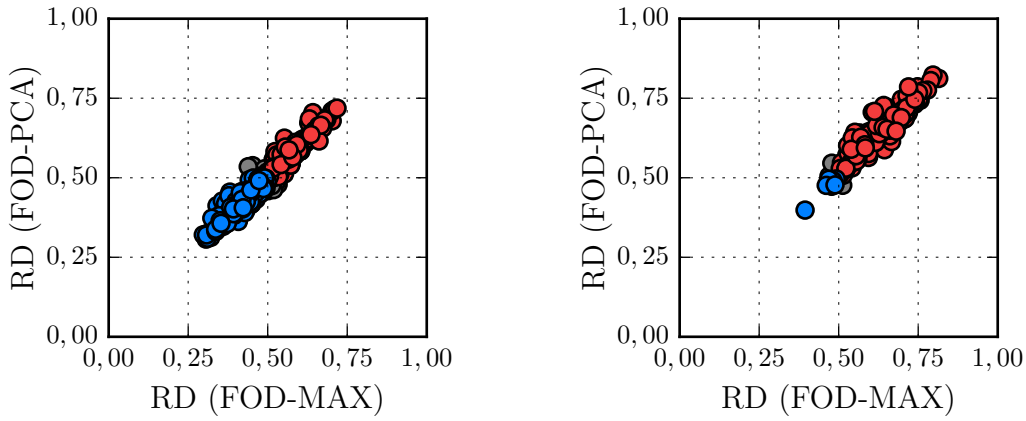


(a) Wynik działania metody FOD-MAX. Ułamki całkowitej wariancji atomów efektywnych: $\sigma_x^2 = 47{,}7\%$, $\sigma_y^2 = 27{,}6\%$, $\sigma_z^2 = 24{,}5\%$. Wartość RD: 0,799.



(b) Wynik działania metody FOD-PCA. Ułamki całkowitej wariancji atomów efektywnych: $\sigma_x^2 = 50{,}8\%$, $\sigma_y^2 = 24{,}6\%$, $\sigma_z^2 = 24{,}7\%$. Wartość RD: 0,804.

Rysunek 3.11: Porównanie efektów działania metod FOD-MAX i FOD-PCA zastosowanych do ułożenia zgodnie z osiami układu współrzędnych atomów efektywnych kompleksu `1AON`. Metoda FOD-PCA maksymalizuje ich wariancję w kolejnych wymiarach przestrzeni oraz „rozpoznaje” siedmiokrotną symetrię osiową cząsteczki, czego metoda FOD-MAX, korzystająca ze średnic zamiast składowych głównych, nie jest w stanie wykonać. Zastosowanie nowego podejścia umożliwiło około siedmiokrotne skrócenie czasu obliczeń rozkładu hydrofobowości teoretycznej.



(a) Łańcuchy. (b) Kompleksy.

Rysunek 3.12: Porównanie wyników zwracanych przez metody FOD-MAX i FOD-PCA. Każdy znacznik odpowiada jednemu łańcuchowi lub kompleksowi. Kolory znaczników oznaczają status zgodności struktury z modelem FOD: niebieski – zachowana zgodność, czerwony – zachowana niezgodność, szary – zmiana zgodności.

Dzięki temu, że sposób obliczeń hydrofobowości obserwowanej nie uległ zmianie, różnice w ułożeniu cząsteczek mogą być wyrażone za pomocą wartości RD. W związku z tym, obliczono je dla wszystkich łańcuchów i kompleksów z bazy danych, stosując każdą z metod osobno. Uzyskano w ten sposób cztery rozkłady, przedstawione parami na rysunku 3.12. Łańcuchy były analizowane niezależnie od swoich partnerów w kompleksie. Do określenia poziomu podobieństwa wartości RD użyto współczynnika korelacji liniowej Pearsona [352]. W przypadku łańcuchów, jego wartość wyniosła 0,963 ($p = 0{,}0$), a dla kompleksów – 0,945 ($p = 0{,}0$). Wobec braku zauważalnych punktów odstających, pozwoliło to na stwierdzenie, że w wybranej grupie białek, niezależnie od rozmiaru i kształtu ich cząsteczek, najważniejsze czynniki decydujące o hydrofobowości teoretycznej reszt, czyli najdłuższa średnica zbioru atomów efektywnych oraz pierwsza składowa główna, są do siebie bardzo zbliżone. Alternatywna interpretacja tej obserwacji mówi, że w sensie dywergencji Kullbacka-Leiblera, wybór stosowanego podejścia nie ma istotnego statystycznie wpływu na otrzymywane wyniki. Oczywiście, indywidualne wartości $\tilde{H}t$ mogą się od siebie nieznacznie różnić. Średnia wartość współczynników korelacji par tych rozkładów we wszystkich białkach z bazy danych wyniosła 0,988 ($\sigma = 0{,}009$), zarówno dla kompleksów jak i dla łańcuchów. Do rozpatrzenia pozostaje teraz kwestia struktur o RD bliskim 0,5, których interpretacja przez model FOD może ulec zmianie wraz z metodą obliczeń.

Zmiana statusu zgodności z modelem FOD dotyczyła wyłącznie tych cząsteczek, których RD było bliskie 0,5, a więc najbardziej podatnych na niewielkie różnice w sposobie ich ułożenia. Stało się tak w przypadku 32 łańcuchów i 4 kompleksów. Wśród elementów pierwszej z tych grup, metoda FOD-PCA uznała 21 za zgodne z modelem FOD, natomiast pozostałe 11 za niezgodne. Pomimo tego, całkowita liczba białek, których łańcuchy posiadają odmienną charakterystykę jądra hydrofobowego się nie zmieniła i nadal wynosiła 16. Sytuacja kompleksów była natomiast odwrotna: metoda FOD-PCA wskazała na niezgodność trzech i zgodność jednego z nich. Normalizacja rozkładów hydrofobowości powoduje, że wymuszenie zmiany statusu dużych struktur wymaga równie dużych zmian w ich ułożeniu, lub może okazać się w ogóle niemożliwe. Widać to dobrze na przykładzie białka `1AON`, składającego się z 8015 reszt, w którym obrót o około 45 stopni spowodował przyrost wartości RD o około 0,005.

239 (RD $<$ 0,5) + 129 (RD $\geq$ 0,5) łańcuchów oraz 8 (RD $<$ 0,5) + 188 (RD $\geq$ 0,5) kompleksów nie zmieniło statusu swojego jądra hydrofobowego. Metoda FOD-PCA wykazała jednak niewielką tendencję do kierowania wyników ku niezgodności z modelem FOD. Średnia różnicy pomiędzy rozkładami wartości RD (FOD-PCA minus FOD-MAX) dla łańcuchów wyniosła bowiem 0,003 ($\sigma = 0{,}0221$), a dla kompleksów 0,013 ($\sigma = 0{,}025$). Największą zmianę, z 0,605 na 0,706 zaobserwowano w białku `2YVE` [390], którego kształt w rzucie atomów efektywnych na płaszczyznę XY przypomina kwadrat. Metoda FOD-MAX ułożyła równolegle do osi X jego przekątną, natomiast FOD-PCA – bok, czego efektem było skrócenie promieni elipsoidy „kropli" w obydwu wymiarach z $\approx 44\,\text{Å}$ do $\approx 37\,\text{Å}$, powodujące odsunięcie się od siebie rozkładów $\tilde{H}t$ i $\tilde{H}o$. Dotyczyło to w szczególności reszt znajdujących się w środku tej cząsteczki. Białko `1AON` jest wewnątrz puste, przez co wpływ jego ułożenia na wartość RD jest względnie niewielki. Za to wyraźnie odstający od pozostałych na rysunku 3.12b kompleks posiada identyfikator `2D3K` [391]. Jego RD wyniosło odpowiednio 0,394 dla FOD-MAX oraz 0,398 dla FOD-PCA. Podobne, choć trochę niższe wartości uzyskano za każdym razem dla obydwu tworzących go łańcuchów. Jest to hydrolaza, tak więc wyniki te są zgodne z założeniami modelu FOD.

Podsumowując, uzyskane wyniki sugerują, że metoda FOD-PCA może być z powodzeniem stosowania w miejscu metody FOD-MAX, pozwalając na szybsze i dokładniejsze obliczanie rozkładu $\tilde{H}t$. Wysoka korelacja pomiędzy wartościami hydrofobowości pozwala założyć, że będzie ona zachowywać wsteczną kompatybilność z wnioskami przedstawionymi w dotychczas opublikowanych pracach. Metoda FOD-PCA wnosi również nową funkcjonalność, jaką jest wyeliminowanie potrzeby nadzoru ze strony użytkownika dzięki automatycznemu wykrywaniu osi symetrii białek.

3.4. Analiza białek z bazy danych

Przed przystąpieniem do badań nad wpływem pól zewnętrznego i wewnętrznego na proces tworzenia się kompleksów białkowych, w celu dostosowania parametrów symulacji, niezbędne było poznanie struktur należących do bazy danych rozprawy oraz sprawdzenie czy spełniają wymagania określone w rozdziale 2.1. W związku z tym, poniżej przedstawione są wyniki następującej analizy:

1. podstawowych informacji,
2. dopasowania struktury pierwszorzędowej,
3. dopasowania struktury trzeciorzędowej,
4. kontaktów niewiążących,
5. wartości energii oddziaływań pola wewnętrznego,
6. wartości RD pola zewnętrznego.

Kolejność ta jest nieprzypadkowa – wyniki z wcześniejszych punktów stanowią bowiem pomoc w badaniach przeprowadzonych później. Najważniejsze dane są dodatkowo umieszczone na końcu rozprawy, w tabeli B.1.

3.4.1. Podstawowe informacje

Baza danych rozprawy składała się 200 białek pobranych z bazy PDB. Warunkiem koniecznym do ich wyboru była stechiometria A2 kompleksu.

W zaokrągleniu do wartości całkowitych, przeciętna liczba reszt aminokwasowych w łańcuchach wyniosła 147 ($\sigma \approx 28$). Par, które zawierały ich dokładnie tyle samo było 125. Dwie największe różnice w liczbie reszt, 8,1% i 6,9%, zaobserwowano odpowiednio w białkach `3HV2` i `2OMD`. Wśród pozostałych 73 liczba ta nie przekroczyła 5% – średnio 1,5% ($\sigma = 0{,}9\%$).

Tylko jedna struktura, `1XOX` [392], została uzyskana przy pomocy techniki magnetycznego rezonansu jądrowego (NMR). Wszystkie pozostałe uzyskano na drodze eksperymentu rentgenografii strukturalnej (XRD), którego przeciętna wartość parametru rozdzielczości wyniosła 1,89 Å ($\sigma = 0{,}43$ Å). Pozwala to założyć, że zapisane w plikach PDB położenia atomów stanowią akceptowane przybliżenie ich położeń w kryształach. Bardzo wysoką dokładnością (wartością parametru rozdzielczości mniejszą lub równą 1,5 Å [229]) charakteryzowały się 33 struktury.

Najczęściej występującym organizmem źródłowym białek z bazy danych był *Homo sapiens*, od którego pochodziło 26 cząsteczek. Następne pod tym względem były *Escherichia coli* (11) oraz *Saccharomyces cerevisiae* (9). Odpowiada to sytuacji w bazie PDB, gdzie człowiek jest również najbardziej powszechnym źródłem struktur (obecnie powyżej 45%[2]), a bakteria *E. coli* i drożdże znajdują się w pierwszej piątce.[3] Łącznie, 200 kompleksów reprezentowało 103 różne organizmy, z których zostały bezpośrednio wyizolowane 23 razy.

Cztery spośród wybranych białek posiadały strukturę fibrylarną, choć tylko jedno z nich, `2SPC` [393], było faktycznym elementem konstrukcyjnym komórki (spektryna). Pozostałe 196 miało kształt zbliżony do globularnego.

137 białek z bazy danych zawierało w swoich strukturach ligandy. W 25 przypadkach były nimi wyłącznie jony. Spośród pozostałych 112, 76 tworzyło kompleksy z cząsteczkami zbudowanymi z przynajmniej 10 atomów ciężkich. Największym zaobserwowanym ligandem był związek TL-3 (identyfikator PDB `3TL`) w białku `3SLZ` [394]. Liczba jego atomów wyniosła aż 66.

Na podstawie informacji uzyskanych z bazy CATH w wersji 4.0.0 stwierdzono, że 111 domen w wybranych białkach charakteryzuje się mieszaną strukturą drugorzędową. Przewaga helis alfa wystąpiła w 37 z nich, a arkuszy beta – w 52. Własność ta przekłada się również na całe cząsteczki, ponieważ domeny te były tożsame z kompletnymi łańcuchami 189 białek. Wśród pozostałych 11, w 10 przypadkach tworzyła je zdecydowana większość sekwencji – średnio powyżej 90% ich długości. Jedynym wyjątkiem było tutaj helikalne białko `2ZB9` [395], którego wpis w bazie CATH pokrywał się z mniej niż $\frac{1}{3}$ całkowitej liczby jego reszt. Zostało ono jednak zaakceptowane, gdyż domena ta posiada wystarczająco dużą liczbę kontaktów niewiążących spajających ją z resztą struktury.[4] Dotyczyło to również fragmentu pętli łączącego te podjednostki, biegnącego wzdłuż dwóch równoległych helis.

Liczba unikatowych topologii CATH wyniosła 84, z czego 50 posiadało w bazie danych w jednego reprezentanta. Do pozostałych 34 należało więc średnio od 4 do 5 białek. Zbliżone do tych wartości odchylenie standardowe było spowodowane przez trzy najczęściej występujące tu topologie: *Rossmann fold* (24 razy), *Immunoglobulin-like* (18 razy) i *Jelly rolls* (11 razy). 24 białka nie miały ustalonej funkcji.

[2] Stan w maju 2017.

[3] Należy podkreślić, że statystyka ta dotyczy wyłącznie pochodzenia materiału genetycznego. W przypadku białek wyizolowanych bezpośrednio z organizmu, najbardziej powszechnym źródłem jest obecnie *Thermus thermophilus* – co szóstego. Następne w kolejności są *Escherichia coli* i *Saccharomyces cerevisiae*, natomiast człowiek znajduje się dopiero na czwartej pozycji.

[4] CATH w wersji 4.1.0 posiada już wpis większej domeny w tym białku (reszty od 67 do 203), ale postanowiono nie zmieniać z tego powodu zawartości bazy danych rozprawy.

3.4.2. Dopasowanie sekwencji

Średnia wartość współczynnika identyczności rzeczywistych sekwencji par łańcuchów, czyli uzyskanych na podstawie dostępnego składu aminokwasowego w każdym z 200 wybranych białek, wyniosła 99,3% długości ich dopasowania ($\sigma = 1{,}2\%$). Potwierdza to, że zgodnie z kryterium stosowanym przez RCSB, wszystkie z nich są faktycznie homodimerami. W połączeniu z wymogiem braku mutacji punktowych, wynik ten wskazuje dodatkowo, że wszelkie niezgodności pomiędzy sekwencjami są efektem insercji/delecji na N- lub C- końcach łańcuchów, powstałych najprawdopodobniej z powodu niedoskonałości eksperymentu krystalograficznego.

Przyjęto, że identyczność par sekwencji różnych białek z bazy danych nie może przekraczać poziomu 33,3%. Spośród 19900 możliwych kombinacji, wartość jej współczynnika była wyższa od 30% tylko w 8 przypadkach. Jak można było się tego spodziewać, białka tworzące każdą z tych par należały do tych samych topologii CATH. Wartość większa lub równa 20% wystąpiła również we względnie niewielkiej ich liczbie – 319, choć tworzyło je już 185 struktur o unikatowych identyfikatorach. Oznacza to, że prawie każde z wybranych białek posiadało w bazie danych przynajmniej jednego partnera (średnio pomiędzy 3 a 4, $\sigma \approx 2$), gdzie na jednej na pięć pozycji po dopasowaniu do siebie ich sekwencji znajdowała się ta sama reszta. Przeciętna wartość współczynnika tej identyczności wyniosła 9,1% ($\sigma = 6{,}1\%$), przez co można uznać bazę danych rozprawy pod tym względem za nieredundantną.

3.4.3. Dopasowanie strukturalne

Po potwierdzeniu na podstawie dopasowania sekwencji, że wybrane białka są homodimerami, wykonana została analiza struktur trzeciorzędowych ich łańcuchów przy użyciu miary RMSD. Analiza ta miała na celu sprawdzenie czy są one również w tej kwestii do siebie podobne.

Ponieważ taka sama liczba reszt występowała w parach łańcuchów 125 białek, klasyczny algorytm Kabscha mógł być zastosowany jedynie do niecałych $\frac{2}{3}$ całości bazy danych. W związku z tym, postanowiono zastąpić go algorytmem CE (combinatorial extension) [396], pozwalającym na porównywanie struktur o dowolnym składzie aminokwasowym. W odróżnieniu od metody Kabscha, która minimalizuje wartość RMSD wyłącznie pomiędzy dwoma równolicznymi zbiorami atomów, algorytm CE poszukuje najdłuższej ścieżki zbudowanej z dopasowanych do siebie par fragmentów sekwencji (aligned fragment pair, AFP).

Zgodnie z sugerowanymi ustawieniami, sekwencje zostały podzielone na odcinki złożone z 8 reszt. Oznacza to, że długość każdej ścieżki dopasowania była zawsze wielokrotnością tej liczby, stanowiąc drugi obok obliczonej na jej podstawie wartości RMSD wynik działania algorytmu CE. Dzięki temu, możliwa stała się ocena podobieństwa struktur łańcuchów niezależnie od występowania insercji/delecji na końcach ich sekwencji. Każda para była uznawana za niemal identyczną jeżeli długość wynikowej ścieżki dopasowania była zbliżona do liczby reszt, a odpowiadająca jej wartość RMSD znajdowała się na odpowiednio niskim poziomie. W taki sam sposób mogą być porównywane ze sobą również różne białka.

Przeciętna długość ścieżki dopasowania do siebie łańcuchów tworzących kompleksy z bazy danych, wyrażona w postaci ułamka liczby reszt w krótszym z nich, wyniosła 97,6% ($\sigma = 1{,}7\%$). W każdym przypadku przekroczyła ona 90%, co pozwoliło uznać wartości RMSD zwrócone przez algorytm CE za wiążące dla całości analizowanych struktur. Średnia ich rozkładu równa 0,536 Å, wraz z odchyleniem standardowym 0,405 Å, oznacza, że łańcuchy z każdego białka były niemal identyczne zarówno pod względem sekwencji jak i struktury trzeciorzędowej. Wynika stąd, że utworzenie przez nie kompleksu nie wiązało się z wprowadzeniem w nich istotnych, niesymetrycznych zmian konformacyjnych. Należy się więc spodziewać, że taka sama identyczność będzie występować wśród wyników pola zewnętrznego.

Identycznie jak w przypadku analizy sekwencji, wykonane zostało również dopasowanie strukturalne różnych białek. Przyjęto, że za podobne do siebie mogą być uznane te z nich, dla których ułamek ścieżki dopasowania algorytmu CE będzie większy lub równy 90%, a odpowiadająca jej wartość RMSD nie przekroczy 5% liczby reszt w mniejszej cząsteczce. Sprawdzono w ten sposób pojedyncze łańcuchy oraz całe kompleksy. Przynajmniej jedna para łańcuchów spełniła powyższy warunek w 102 z 19900 możliwych kombinacji. W zbiorze tym znalazło się 89 unikatowych identyfikatorów PDB. Poza dwoma wyjątkami (8 i 7), białka do niego należące miały od 1 do 4 partnerów. W bazie danych znalazło się również 21 par podobnych do siebie kompleksów. Utworzyły je 33 spośród powyższych 89 białek. Dla każdego z nich istniało jedno lub dwa o niemal identycznej strukturze. Wszystkie podobne do siebie białka należały do tych samych topologii CATH. Średni poziom dopasowania ich sekwencji wyniósł około 20% ($\sigma \approx 7$), choć zdarzyły się też wyjątki – wartość tego współczynnika nie przekroczyła 10% w przypadku 10 par łańcuchów i dwóch kompleksów: `1MKA` [397] oraz `1YOC`. Stanowią one interesujący materiał do badań, gdyż pozwalają na obserwację tego jak białka o różnych strukturach pierwszorzędowych mogą posiadać podobne struktury trzecio- i czwartorzędowe.

3.4.4. Kontakty niewiążące

Symetria kompleksu jest zjawiskiem powszechnie występującym wśród białek homodimerycznych [398]. Jako główne powody tego zachowania wskazuje się kompresję informacji genetycznej [399] oraz wyższą stabilizację cząsteczki, a przez to ochronę przed sytuacjami patologicznymi [400]. Brak tej właściwości, czyli asymetryczność, wynika natomiast z konkretnej funkcji biologicznej jaką pełni dane białko, lub z charakterystyki środowiska, w którym działa [401].

Kompleks homodimeryczny może być uznany za symetryczny jeżeli zdecydowana większość reszt należących do interfejsów jego łańcuchów posiada takie same identyfikatory [402]. Potrzebne do stwierdzenia tego mapy kontaktów niewiążących zostały obliczone zgodnie z kryterium przyjętym przez serwis PDBsum, czyli poprzez poszukiwanie najbliższych sąsiadów atomów ciężkich w promieniu 3,9 Å [158].

W celu przeprowadzenia wymiernej oceny symetrii kompleksów z bazy danych, Autor rozprawy wprowadził dwie miary zwracające dla każdej pary łańcuchów wartości z przedziału $[0, 1]$. Pierwsza z nich została nazwana ułamkiem wspólnego interfejsu (interface common fraction, ICF). Określa ona jaką część mniejszego z nich stanowią te same reszty zaangażowane w kontakty w obydwu łańcuchach:

$$\mathrm{ICF}\,(A,B) = \frac{|I_A \cap I_B|}{\min\{|I_A|, |I_B|\}} \tag{3.11}$$

gdzie:

I_A = zbiór identyfikatorów reszt z interfejsu łańcucha A

I_B = zbiór identyfikatorów reszt z interfejsu łańcucha B

A = zbiór identyfikatorów wszystkich reszt z łańcucha A

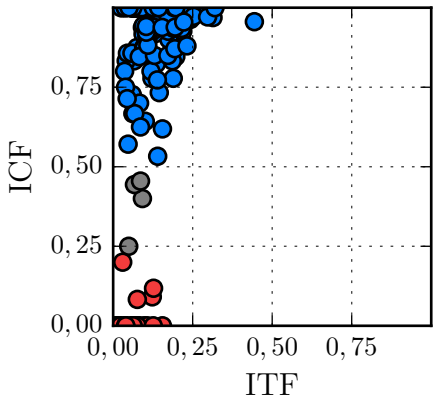
B = zbiór identyfikatorów wszystkich reszt z łańcucha B

ICF równe 1 oznacza pełną symetrię, a 0 – pełną asymetrię kompleksu, z dokładnością do różnicy pomiędzy liczbami reszt w łańcuchach.

Druga miara, ułamek całości reszt zaangażowanych w interfejsie (interface total fraction, ITF) wskazuje natomiast jaką część struktury kompleksu stanowi liczba wszystkich występujących w niej kontaktów:

$$\mathrm{ITF}\,(A,B) = \frac{|I_A| + |I_B|}{|A| + |B|} \tag{3.12}$$

Gdy ITF wynosi 1, wówczas każda reszta posiada przynajmniej jeden kontakt z drugim łańcuchem. Przypadek przeciwny (ITF równy 0) oznacza pusty interfejs.

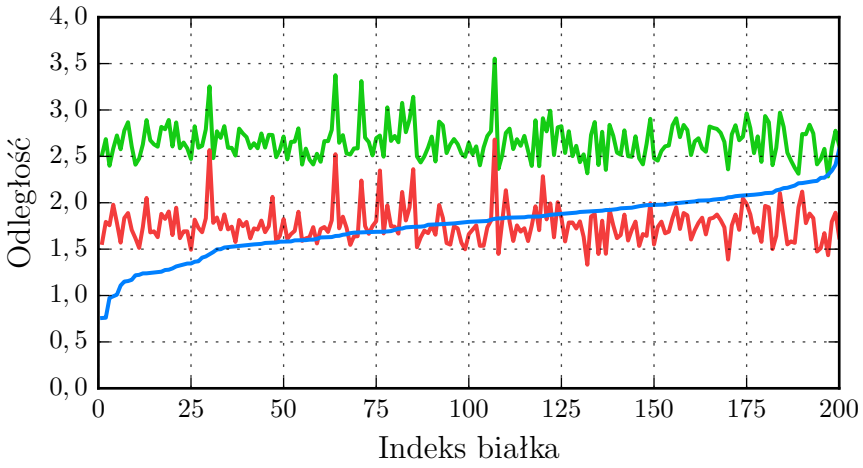


Rysunek 3.13: Miary ICF oraz ITF interfejsów pomiędzy łańcuchami tworzącymi kompleksy z bazy danych rozprawy. Każdy znacznik odpowiada jednemu białku. Kolory znaczników oznaczają status kompleksu: niebieski – symetryczny (ICF $>$ 0,5), czerwony – asymetryczny (ICF $<$ 0,25), szary – status niepewny (szczegóły w tekście).

Ponieważ miary ICF i ITF mają taki sam zakres wartości, charakterystyka interfejsów analizowanych białek może być przedstawiona w postaci punktów w jednostkowym kwadracie. Kwadrat ten znajduje się na rysunku 3.13.

Podczas obserwacji wybranych białek zauważono, że kompleksy symetryczne posiadały ICF wyższe od 0,5, niezależnie od wartości ITF. W związku z tym przyjęto, że połowa wspólnego interfejsu łańcuchów powinna być wystarczająca do stwierdzenia tej cechy struktury. Z drugiej strony, brakiem symetrii charakteryzowały się białka o ICF mniejszym od 0,25. Pozostałe wymagały sprawdzenia wizualnego.

Kompleksów symetrycznych (ICF $>$ 0,5) było w bazie danych 174 (średnia wartość 0,937 przy odchyleniu standardowym 0,094). W 80 przypadkach wyniosła ona dokładnie 1, natomiast różnica tylko jednej reszty pomiędzy częścią wspólną interfejsów a większym z danej pary wystąpiła w 39 spośród pozostałych 94 białek. Z drugiej strony, na rysunku 3.13 widać 22 kompleksy niesymetryczne, charakteryzujące się ICF mniejszym od 0,25. 18 z nich osiągnęło minimum tej miary (0), a w pozostałych czterech łańcuchy kontaktowały się za pośrednictwem nie więcej niż dwóch reszt o tych samych identyfikatorach. Do trzeciej grupy, o niepewnym statusie, trafiły również cztery białka. Ustalono, że trzy z nich (bliższe 0,5) są symetryczne, a jedno (bliższe 0,25) – niesymetryczne. Niska liczba elementów tej grupy wskazuje na przydatność miary ICF do oszacowywania symetrii kompleksów oraz potwierdza założenia dotyczące symetrii białek homodimerycznych.



Rysunek 3.14: Najmniejsze odległości pomiędzy łańcuchami tworzącymi kompleksy z bazy danych rozprawy. Kolor rozkładu oznacza typ pary atomów: niebieski – wodoru (H-H), czerwony – wodoru i atomów ciężkich (H-X), a zielony – tylko dla par atomów ciężkich (X-X). Dane są posortowane rosnąco zgodnie z wartościami rozkładu H-H.

Na podstawie obserwacji rozkładu miary ITF, stwierdzono, że reszty zaangażowane w kontakty niewiążące stanowią średnio 12,5% ($\sigma = 6{,}4\%$) liczby wszystkich reszt w kompleksach. Jest to niemała wartość biorąc pod uwagę ich kształt, co sugeruje hydrofobowość interfejsów łańcuchów [163]. Największy z nich posiadało wspomniane wcześniej fibrylarne białko `2SPC`, zbudowane z położonych równolegle helis α. Występuje ono na rysunku 3.13 jako skrajnie prawy punkt, o ITF równym 0,444. Po przeciwnej stronie znalazł się natomiast kompleks `1G17` [403] posiadający tylko jeden, na dodatek niesymetryczny, kontakt pomiędzy dwoma resztami.

Po wykonaniu ogólnej analizy kontaktów niewiążących, skupiono się na atomach reszt w nie zaangażowanych, przede wszystkim w celu sprawdzenia wpływu dodania atomów wodoru przez program REDUCE na odległości pomiędzy łańcuchami. Odnaleziono więc we wszystkich kompleksach trzy ich pary znajdujące się najbliżej siebie, z których każda należała do jednej z poniższych kategorii:

- H-H: odległości pomiędzy parami atomów wodoru,
- H-X: odległości pomiędzy parami atomów wodoru i ciężkich,
- X-X: odległości pomiędzy parami atomów ciężkich.

Powstałe w ten sposób rozkłady są przedstawione na rysunku 3.14.

Średnia wartość rozkładu X-X wyniosła 2,661 Å ($\sigma = 0{,}190$ Å). Z powodu przyjętych kryteriów wyboru, nie mogło być w bazie danych rozprawy białek, których łańcuchy posiadałyby atomy ciężkie znajdujące się w odległości poniżej 1,9 Å. Choć takie struktury istnieją w bazie PDB, najmniejsza zaobserwowana tu wartość X-X była równa 2,283 Å, co umieszcza ją niewiele powyżej optymalnej długości wiązań wodorowych w polu ECEPP/3. Z drugiej strony, najbliższe sobie atomy ciężkie położone dalej niż 3 Å odnaleziono tylko w 7 kompleksach. Jak można było się tego spodziewać, odpowiadająca im wartość ITF nie przekroczyła poziomu 0,07.

Średnie rozkładów H-X i H-H różniły się dopiero na trzecim miejscu po przecinku wynosząc odpowiednio 1,761 Å ($\sigma = 0{,}196$ Å) oraz 1,760 Å ($\sigma = 0{,}310$ Å). Interpretacja tej obserwacji jest taka, że dodanie atomów wodoru przez program REDUCE spowodowało częściowe wypełnienie nimi przestrzeni pomiędzy łańcuchami, zbliżając je przeciętnie do siebie o około 0,9 Å ($\sigma = 0{,}109$ Å), czyli liczbę porównywalną z długością wiązań jakie tworzy w białkach ten pierwiastek. Sugeruje to wysoki współczynnik korelacji rozkładów H-X i X-X, wynoszący 0,841. Wartości pierwszego z nich były niższe od 1,9 Å 169 razy. Z drugiej strony, odnaleziono 129 białek w których odległość par atomów wodoru również nie przekroczyła tego progu. Jednocześnie stwierdzono korelację bliską zeru pomiędzy rozkładem H-H i pozostałymi dwoma.

3.4.5. Pole wewnętrzne

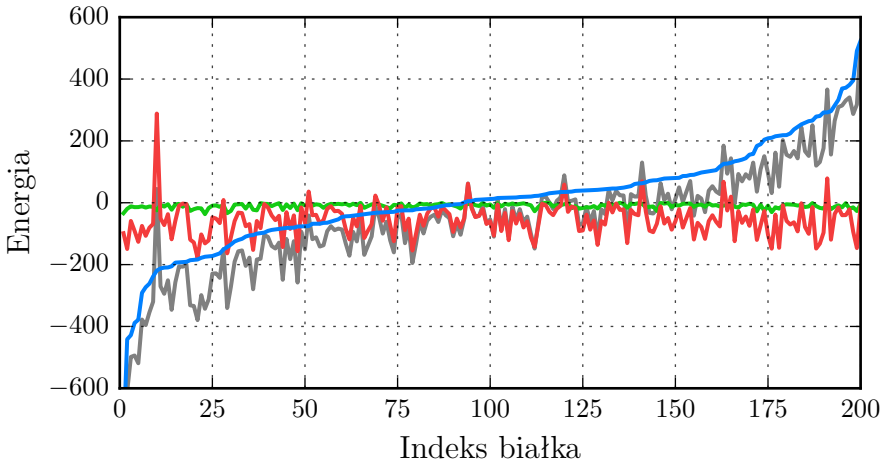
Przedostatnim krokiem analizy białek z bazy danych rozprawy była obserwacja odpowiadających im wartości pola zewnętrznego, czyli energii oddziaływań pola ECEPP/3. Oprócz samych wyników, zaprezentowany jest tu również sposób w jaki zostały one uzyskane, konsekwentnie stosowany w całej rozprawie.

Ponieważ struktury białek były traktowane w tym eksperymencie jako bryły sztywne, wszystkie składowe energii oddziaływań wewnątrzcząsteczkowych można było potraktować jako stałe: $E_{\text{intra}} = \text{const}$. Tym samym, jedyną zmienną stanowiły oddziaływania pomiędzy łańcuchami, wynikające z ich wzajemnej orientacji, reprezentujące część całkowitej energii układu związanej z utrzymaniem kompleksu w całości. Ponieważ międzycząsteczkowe mostki disiarczkowe zostały wyeliminowane przez przyjęte kryteria wyboru, do obliczenia pozostały wyłącznie wartości potencjałów niekowalencyjnych: $E = E_{\text{inter}} = E_e + E_n + E_h$. Obecność mostków disiarczkowych pozbawiałaby sensu poszukiwanie konformacji natywnej kompleksu, gdyż równanie 2.10 silnie wskazuje na konkretne rozwiązanie, które musiałoby być znane *a priori*, co stoi w sprzeczności z ideą eksperymentu *ab initio*.

Jeżeli dwa łańcuchy tworzące dany kompleks są zbudowane odpowiednio z a i b atomów, złożoność obliczeniowa algorytmu analizującego wszystkie możliwe ich pary na potrzeby równań 2.3, 2.5 i 2.7 wynosi $O(ab)$. Zwracane wówczas wyniki byłyby w pełni zgodne z definicją pola ECEPP/3, ale aby mogło być ono stosowane jako kryterium optymalizacyjne w tym eksperymencie, niezbędne okazało się zrezygnowanie z części tej dokładności na rzecz przyspieszenia obliczeń, umożliwiającego uzyskanie wyników w rozsądnym czasie. Ponieważ potencjały U_e, U_n i U_h maleją wraz z odległością pomiędzy atomami, coraz bardziej oddalone od siebie pary mają coraz mniejszy wpływ na ich wartości. W związku z tym, zdecydowano się na użycie stałego promienia odcięcia, przyjmując jego sugerowaną długość równą 12 Å [404]. Zastosowano go we wszystkich trzech przypadkach. Podyktowane zostało to względami praktycznymi, pomimo tego, że dla szybko zanikających oddziaływań van der Waalsa i wiązań wodorowych wystarczyłby krótszy promień.

Inny problem związany z obliczeniami wartości pola wewnętrznego wynika z heurystycznego dodania atomów wodoru do białek. Spowodowane przez tą czynność zbyt silne zbliżenie łańcuchów do siebie, przekładające się na nadmierny wzrost wartości potencjałów U_n i U_h, mógłby utrudnić dotarcie do struktur natywnych podczas optymalizacji tego kryterium. Już dla dwóch atomów znajdujących się w odległości 1 Å, potencjały te są mierzone w dziesiątkach lub setkach tysięcy $\frac{\text{kcal}}{\text{mol}}$. Jeżeli kolizja jest poważna, ale dotyczy tylko jednej pary, do jej usunięcia może wystarczyć niewielkie przesunięcie. W przypadku większej ich liczby („zakleszczonego” interfejsu) zapewne będzie to niewykonalne, przez co łańcuchy zostaną odsunięte od siebie, powodując zmniejszenie wartości miary ITF i w konsekwencji prawdopodobnie również ICF, lub przyjmą inną, bardziej atrakcyjną od natywnej konformację. Możliwa jest również koniunkcja tych zdarzeń.

Brak możliwości optymalizacji orientacji łańcuchów bocznych jest jedną z wad podejścia traktującego białka jako bryły sztywne, w których zostały wprowadzone modyfikacje. Aby temu zaradzić bez modelowania elastyczności ich struktury, ustalono, że każda para atomów, z których przynajmniej jeden jest wodorem, znajdująca się w odległości mniejszej niż 1,9 Å będzie traktowana jakby znajdowała się dokładnie w tej odległości. Podejście to zastosowano do wszystkich trzech potencjałów. Dzięki temu, maksymalna wartość U_h mogła wynieść w przybliżeniu 1,4 $\frac{\text{kcal}}{\text{mol}}$, a U_n: od 3 $\frac{\text{kcal}}{\text{mol}}$ dla pary atomów wodoru do 120 $\frac{\text{kcal}}{\text{mol}}$ dla wodoru i siarki. Potencjał elektrostatyczny w $r_{ij} = 1{,}9$ zawierał się natomiast w przedziale $[-16, 16]$ $\frac{\text{kcal}}{\text{mol}}$. Promień ten jest krótszy od optymalnych odległości pomiędzy dowolnymi atomami w polu ECEPP/3, dzięki czemu wszystkie miały teoretycznie szansę na ich osiągnięcie.



Rysunek 3.15: Wartości energii oddziaływań niekowalencyjnych pomiędzy łańcuchami tworzącymi kompleksy z bazy danych rozprawy. Kolor rozkładu oznacza potencjał: niebieski – E_e, czerwony – E_n, zielony – E_h, szary – E (suma). Dane są posortowane rosnąco zgodnie z wartościami rozkładu E_e. Wartości energii dla białka znajdującego się poza przedziałem wartości osi pionowej można odczytać z tabeli 3.7.

Energia	**Średnia**	**Odch. std.**	**Minimum**	**Maksimum**
E_e	10,852	164,592	−769,854	523,537
E_n	−56,357	52,931	−178,781	288,349
E_h	−10,011	7,526	−36,938	−0,004
E_{inter}	−55,516	177,762	−902,385	509,698

Tabela 3.7: Statystyka wartości energii oddziaływań niekowalencyjnych pomiędzy łańcuchami tworzącymi kompleksy z bazy danych rozprawy.

Wartości energii białek z bazy danych rozprawy są przedstawione na rysunku 3.15, natomiast ich statystyka znajduje się w tabeli 3.7. W 127 kompleksach suma energii wszystkich trzech potencjałów była mniejsza od 0, ale nie zaobserwowano korelacji pomiędzy nią a liczbą reszt lub symetrią interfejsu. Widoczne na rysunku 3.15 dodatnie piki rozkładu E_n dotyczą 11 białek. Poza jednym wyjątkiem (ponownie `2SPC` – 288,349 $\frac{\text{kcal}}{\text{mol}}$), ich wysokość nie przekroczyła 100 $\frac{\text{kcal}}{\text{mol}}$. Wyższa wartość energii w białku `2SPC` wynikła z jego dużego interfejsu (50 reszt – najwięcej w całej bazie danych). Z rysunku 3.15 można również odczytać, że wiązania wodorowe pomiędzy łańcuchami nie miały istotnego znaczenia dla stabilności kompleksów, w odróżnieniu od najwolniej słabnących wraz z odległością oddziaływań elektrostatycznych.

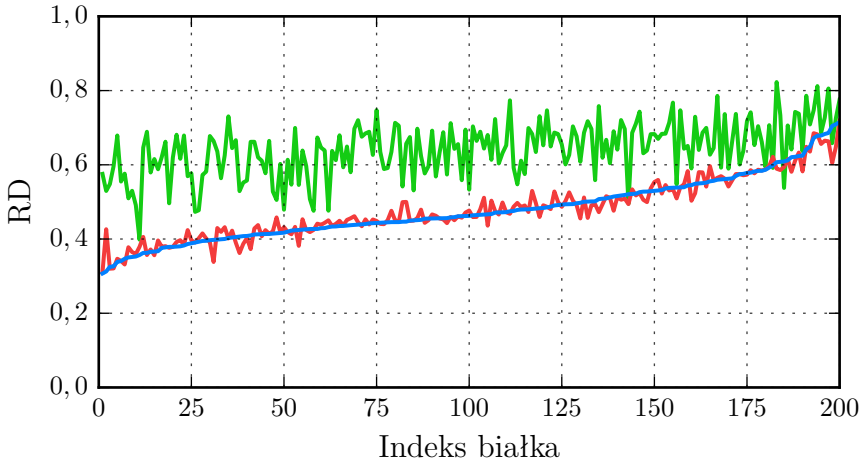
3.4.6. Pole zewnętrzne

Na koniec pozostała analiza charakterystyki białek z bazy danych rozprawy w sensie pola zewnętrznego. Stosując przedstawioną w rozdziale 3.3 metodę FOD-PCA, obliczono dla każdej struktury trzy wartości RD, reprezentujące status jądra hydrofobowego w kompleksie oraz w tworzących go łańcuchach. Otrzymane w ten sposób rozkłady są przedstawione na rysunku 3.16a.

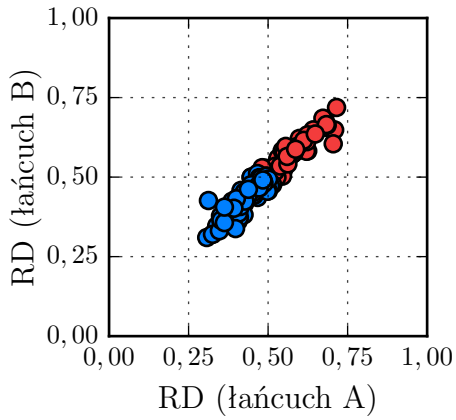
Wykształcone jądro hydrofobowe zaobserwowano w obydwu łańcuchach 122 białek, których średnie RD wyniosło 0,427 ($\sigma = 0{,}043$). Wysoką stabilność, rozumianą jako RD $< 0{,}4$, wykazało 25 par. Brak wykształconego jądra hydrofobowego zauważono natomiast w jednym i drugim łańcuchu należącym do 62 kompleksów. Ich średnie RD okazało się równe 0,577, przy odchyleniu standardowym 0,053. Największą różnicę pomiędzy rozkładami $\tilde{H}t$ i $\tilde{H}o$ wykazały znów łańcuchy białka `2SPC`, co jest zgodne z interpretacją struktur fibrylarnych przez model FOD. Wartość RD każdego z nich przekroczyła 0,71, dzięki czemu znalazły się pod tym względem nawet powyżej większości kompleksów. Status mieszany, czyli taki, gdy jeden łańcuch jest zgodny, a drugi niezgodny z modelem FOD, nadano pozostałym 16 parom. RD wszystkich z nich było bliskie 0,5, przy czym w 13 przypadkach mniejsza z tych wartości nie różniła się od tego granicznego poziomu o więcej niż 0,01. Ponieważ pary łańcuchów z analizowanych białek były bardzo do siebie podobne pod względem struktury trzeciorzędowej, za możliwy powód tych rozbieżności uznano niewielkie zmiany konformacyjne, rozdzielczość eksperymentu krystalograficznego, niewystarczającą w niektórych przypadkach do dokładnego zlokalizowania atomów efektywnych, lub zaokrąglenia podczas obliczeń rozkładów hydrofobowości i D_{KL}.

W związku z tym, że białka o mieszanej charakterystyce łańcuchów stanowiły tylko 8% bazy danych i znajdowały się w sensie wartości RD w pobliżu pozostałych grup, w celu zmniejszenia liczby typów zgodności z modelem FOD z trzech (2/1/0)do dwóch (2/0), postanowiono przydzielić je do pozostałych na podstawie średniej ich wartości RD. Jeżeli była ona niższa od 0,5, wówczas obydwa uznawano za zgodne z modelem i odwrotnie. Dzięki temu, 9 spośród tych struktur dołączyło do grupy o stabilnym jądrze hydrofobowym, a pozostałe 7 – o niestabilnym.

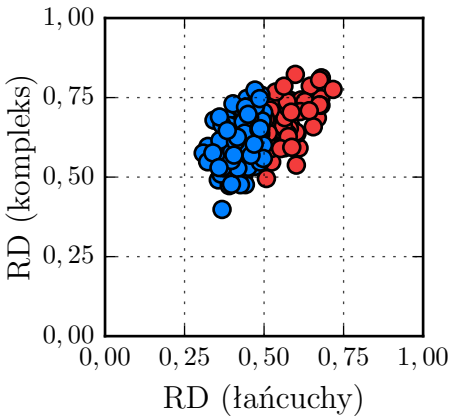
Współczynnik korelacji RD par łańcuchów we wszystkich białkach wyniósł 0,963 (rysunek 3.16b). Potwierdza to wcześniejsze oczekiwania wynikające z obserwacji ich struktur trzeciorzędowych. Tak jak w przypadku pola wewnętrznego, nie stwierdzono korelacji pomiędzy wartościami RD i ICF, choć do uzyskania pierwszej z nich poniżej 0,5, niezbędne było posiadanie drugiej bliskiej 1 (najmniej 0,846).



(a) Wartości RD łańcucha A (rozkład niebieski), B (rozkład czerwony) oraz kompleksu (rozkład zielony). Dane są posortowane rosnąco, zgodnie z rozkładem wartości dla łańcucha A.

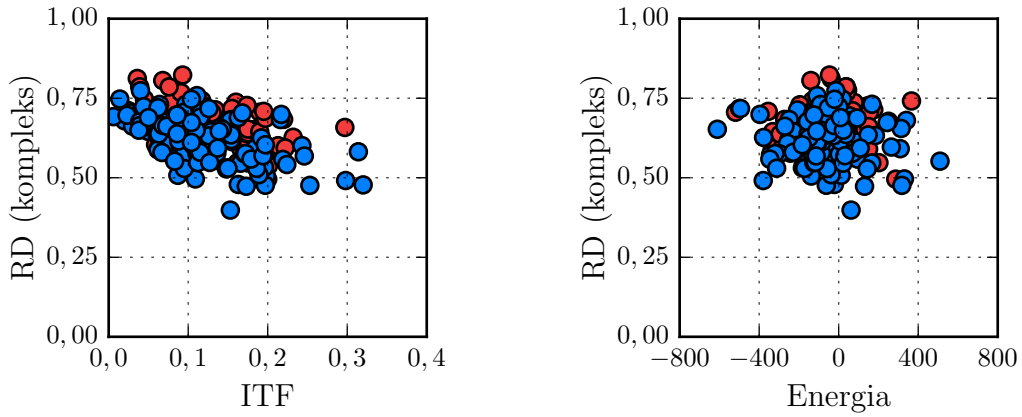


(b) Relacja pomiędzy wartościami RD łańcuchów.



(c) Relacja pomiędzy wartościami RD łańcuchów (średnia) i kompleksu.

Rysunek 3.16: Wartości RD łańcuchów i kompleksów z bazy danych rozprawy, obliczone przy pomocy metody FOD-PCA. Każdy znacznik odpowiada jednemu białku. Kolory znaczników oznaczają zgodność łańcuchów z modelem FOD: niebieski – obydwa zgodne, czerwony – obydwa niezgodne.



(a) Relacja pomiędzy RD i ITF.

(b) Relacja pomiędzy RD i energią.

Rysunek 3.17: Relacja pomiędzy wartościami RD kompleksów a miarą ITF i energią w kompleksach z bazy danych rozprawy. Każdy znacznik odpowiada jednemu białku. Kolory znaczników oznaczają zgodność par łańcuchów z modelem FOD: niebieski – obydwa zgodne, czerwony – obydwa niezgodne.

Tylko w 9 białek z bazy danych stwierdzono charakterystykę jądra hydrofobowego kompleksu zgodną z modelem FOD. Identyczny status wykazały tworzące je łańcuchy. Par, które utraciły zgodność z modelem po utworzeniu kompleksu było natomiast 122. Oznacza to, że wzrost hydrofobowości obserwowanej w środku dwukrotnie większej elipsoidy „kropli" dopasowanej do struktury białka-homodimeru nie był w stanie spełnić jej oczekiwań teoretycznych. W celu osiągnięcia stabilizacji jądra hydrofobowego w sensie modelu FOD, kompleksy powinny dążyć do minimalizacji wartości RD poprzez zwiększanie liczby kontaktów pomiędzy resztami hydrofobowymi, układają jednocześnie te postrzegane jako bardziej hydrofilne bliżej środowiska wodnego. Na poparcie tej hipotezy można wskazać niedużą, ale zauważalną ujemną korelację pomiędzy miarami RD a ITF, widoczną na rysunku 3.17a. Pomijając znów białko `2SPC`, jej współczynnik wyniósł $-0{,}477$.

Sytuacja, w której łańcuchy uzyskały zgodność z modelem FOD jako kompleks, choć były z nim niezgodne osobno nie wystąpiła w bazie danych. Niewykluczone jest, że wybrane białka nie mogły osiągnąć już niższego poziomu wartości RD. Sugeruje to widoczna na rysunku 3.16b korelacja tej wartości pomiędzy łańcuchami i kompleksami, której współczynnik wyniósł 0,517. Nie zaobserwowano natomiast korelacji pomiędzy wynikami zwróconymi przez obydwa pola (rysunek 3.17b), sugerującej posiadanie przez nie podobnych krajobrazów wartości i wspólnych minimów.

3.5. Kompleksowanie białek – opis eksperymentu

W celu sprawdzenia założeń pola zewnętrznego (modelu FOD) dotyczących wpływu opisywanych przez niego oddziaływań hydrofobowych na proces tworzenia się kompleksów białkowych, a także wyciągnięcia na tej podstawie wniosków na temat możliwości jego zastosowania jako kryterium optymalizacyjnego w mechanice molekularnej, opracowano i przeprowadzono eksperyment *ab initio* polegający na podzieleniu każdej z 200 struktur homodimerycznych należących do bazy danych rozprawy na tworzące ją łańcuchy, a następnie próbie osiągnięcia na powrót ich konformacji natywnej poprzez optymalizację globalną wartości RD. Uzyskane w ten sposób wyniki zostały następnie skonfrontowane z wynikami identycznej symulacji, ale przeprowadzonej przy użyciu pola wewnętrznego (pola ECEPP/3), reprezentującego obecne podejścia do problemu przewidywania struktury czwartorzędowej białek, polegające na minimalizacji wartości energii oddziaływań pomiędzy atomami. Na koniec sprawdzono, czy symulacja równoczesnego wpływu pól zewnętrznego i wewnętrznego, wykonana poprzez optymalizację wielokryterialną ich kryteriów, może także przyczynić się do osiągnięcia natywnych konformacji kompleksów.

3.5.1. Reprezentacja układu

Aby ograniczyć liczbę czynników mogących wpływać na stanu symulowanego układu i umożliwić wydajną pod względem czasu trwania optymalizację opisujących go kryteriów, a także ułatwić interpretację uzyskanych dzięki niej wyników, przyjęto, że łańcuchy białek z bazy danych będą traktowane jako bryły sztywne. Ponieważ nie dochodziło w nich do istotnych zmian konformacyjnych podczas tworzenia przez nie kompleksów, takich jak wymiana domen, uznano, że na tym etapie badań powinno stanowić to akceptowalne przybliżenie. Zmiana konformacji kompleksu mogła więc wynikać wyłącznie ze zmiany orientacji jednego łańcucha (liganda). Drugi (receptor) był nieruchomy. Optymalnie, w stałym położeniu powinna znaleźć się największa cząsteczka, ale ponieważ w białkach homodimerycznych nie ma to znaczenia, uznano, że receptorem będzie zawsze łańcuch A, a ligandem – łańcuch B.

Do określenia orientacji bryły sztywnej w trójwymiarowej przestrzeni potrzebne jest sześć zmiennych: trzy składowe wektora położenia jej środka geometrycznego: $\{x, y, z\}$ oraz trzy kąty jej obrotu wokół osi układu współrzędnych: $\{\alpha, \beta, \gamma\}$ [405]. Do wymiernej reprezentacji kompleksu ligand-receptor wystarczył więc pojedynczy wektor $[x, y, z, \alpha, \beta, \gamma]$, reprezentujący ich konformację w przestrzeni rozwiązań.

Problem jaki wiąże się z powyższą reprezentacją wynika stąd, że w niektórych przypadkach, ta sama orientacja bryły sztywnej może być osiągnięta na podstawie więcej niż jednego zestawu kątów $\{\alpha, \beta, \gamma\}$ [405]. Efektem tego jest zwielokrotnienie minimów globalnych optymalizowanych kryteriów, stanowiących dodatkową trudność dla poszukujących ich algorytmów oraz dla późniejszej analizy wyników. Dlatego postanowiono zastosować inny sposób kodowania, będący funkcją faktycznie wzajemnie jednoznaczną pomiędzy przestrzeniami rozwiązań i konformacyjną.

Zamiast kątów $\{\alpha, \beta, \gamma\}$ posłużono się w tym eksperymencie kątami $\{\theta, \phi, \psi\}$, z których pierwszy określał kąt obrotu wokół wektora wodzącego liganda, a pozostałe dwa – jego współrzędne sferyczne [406]. Za ów wektor wodzący przyjęto kierunek największej zmienności położeń atomów. Aby go wyznaczyć, ligand był układany *a priori* w położeniu początkowym przy pomocy algorytmu FOD-PCA. Identycznie przygotowywano cząsteczkę receptora. Algorytm orientacji liganda na podstawie danego wektora konformacji $[x, y, z, \theta, \phi, \psi]$ był więc następujący:

1. obrót wokół osi X (wektora wodzącego) o θ radianów,
2. obrót wokół osi Z o ϕ radianów,
3. obrót wokół osi Y' (wektora $[0, 1, 0]$ obróconego w kroku drugim) o ψ radianów,
4. translacja do punktu $[x, y, z]$.

Macierze realizujące przekształcenia z punktów 1, 2 i 3, nazwane odpowiednio R_X, R_Z i $R_{Y'}$, były wyznaczane przy pomocy formuły Rodriguesa [407]:

$$R = I + \sin\omega N + (1 - \cos\omega)\, N^2 \tag{3.13}$$

gdzie I oznacza macierz jednostkową, a ω – kąt obrotu zgodnie z regułą prawej dłoni wokół wektora v, będącego podstawą konstrukcji antysymetrycznej macierzy N:

$$N = \begin{pmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{pmatrix} \tag{3.14}$$

Dzięki temu algorytmowi, orientacja liganda składającego się z n atomów, których położenia były zapisane w macierzy A o rozmiarze $3 \times n$, odpowiadająca wektorowi konformacji $[x, y, z, \theta, \phi, \psi]$, mogła być obliczona przy pomocy poniższego równania:

$$A' = (R_{Y'} R_Z R_X)\, A + [x, y, z] \tag{3.15}$$

3.5.2. Kryteria optymalizacyjne

Poszukiwanie natywnej struktury kompleksu receptora i liganda polegało w tym eksperymencie na optymalizacji dwóch kryteriów: f_1 – pola wewnętrznego oraz f_2 – pola zewnętrznego, których argumentami były wektory konformacji.

Kryterium f_1 zwracało wartość energii oddziaływań pola ECEPP/3 obliczaną identycznie jak w części 3.4.5 tego rozdziału, czyli według poniższych zasad:

- brane pod uwagę były wyłącznie oddziaływania międzycząsteczkowe,
- brane pod uwagę były wyłącznie pary atomów znajdujące się w odległości nie większej niż 12 Å,
- pary atomów, w których występował wodór, znajdujące się w odległości mniejszej niż 1,9 Å były traktowane jakby znajdowały się dokładnie w tej odległości.

Kryterium f_2 zwracało natomiast wartość RD modelu FOD obliczoną przy użyciu algorytmu FOD-PCA dla całego kompleksu. Hipoteza, którą miał sprawdzić ten eksperyment, zakłada, że natywna struktura kompleksu homodimerycznego powinna posiadać najniższą wartość RD spośród wszystkich konformacji jego łańcuchów dopuszczalnych przez poniższe funkcje ograniczeń.

3.5.3. Funkcje ograniczeń

Kryteria f_1 i f_2 wykonywały swoje obliczenia nie interesując się tym, czy łańcuchy, których konformacja podlegała w tym czasie optymalizacji faktycznie tworzyły kompleks. Nie miała dla nich również znaczenia poprawność struktury tego kompleksu (występowanie kolizji). Dlatego w celu motywacji algorytmu optymalizacyjnego do poruszania się w stronę rozwiązań o oczekiwanych właściwościach zastosowano trzy funkcje ograniczeń nierówności: g_1, g_2 i g_3. Każda z nich przyjmowała jako swój argument wektor konformacji i zwraca 0 w przypadku jego dopuszczenia.

Pierwsza funkcja ograniczeń, g_1, sprawdzała, czy wartości zmiennych $\{\theta, \phi, \psi\}$ zawierały się w przedziale $(-\pi, \pi]$. Było to niezbędne do uniknięcia problemów związanych z obrotami cząsteczki liganda. Ponieważ dopuszczalny przez tę funkcję podzbiór przestrzeni rozwiązań był wypukły, algorytm optymalizacyjny mógł bez utrudnień poruszać się w jego wnętrzu. W związku z tym przyjęto, że wartość funkcji g_1 dla punktów spoza tego sześcianu będzie równa $+\infty$ [408]. Pozwoliło to na uniknięcie obliczania dla nich wartości pozostałych funkcji ograniczeń.

Funkcja ograniczeń g_2 sprawdzała, czy istniał przynajmniej jeden kontakt niewiążący pomiędzy łańcuchami. W razie niespełnienia tego warunku zwracany był kwadrat odległości pomiędzy ich środkami geometrycznymi. Gdyby symulowanych cząsteczek było więcej niż dwie, wówczas wartość ta byłaby równa sumie odległości od każdej izolowanej z nich do jej najbliższego sąsiada. Tak jak poprzednio, podczas wyznaczania kontaktów niewiążących brane były pod uwagę tylko atomy ciężkie. Wyjątkowo jednak postanowiono wydłużyć promień ich poszukiwania do 4,9 Å, a więc o 1 Å więcej niż wynosi kryterium serwisu PDBsum. Podyktowane zostało to chęcią zwiększenia liczby dostępnych rozwiązań dopuszczalnych, a przez to ułatwienia algorytmowi optymalizacyjnemu przemieszczania się pomiędzy nimi.

Funkcja ograniczeń g_3 dbała natomiast o to, aby pomiędzy łańcuchami nie występowały kolizje, czyli pary atomów ciężkich znajdujące w odległości mniejszej niż 1,9 Å. Każda taka para dodawała do zwracanej przez tę funkcję wartości ujemny logarytm ilorazu swojej odległości i tego promienia.

Koniunkcja warunków określonych przez funkcje g_2 i g_3 powodowała że dopuszczalne były wyłączne te konformacje kompleksów, w których przynajmniej jedna para atomów ciężkich pochodzących z różnych łańcuchów znajdowała się w odległości zawierającej się w przedziale od 1,9 Å do 4,9 Å. Warto również zauważyć, że nie jest możliwe aby funkcje g_2 i g_3 jednocześnie uznały tę samą konformację za niedopuszczalną: brak spełnienia oczekiwań jednej z nich implikuje bowiem usatysfakcjonowanie drugiej: $g_2 > 0 \Rightarrow g_3 = 0$ oraz $g_3 > 0 \Rightarrow g_2 = 0$. W praktyce nie miało jednak znaczenia to, który warunek z tej pary był sprawdzany jako pierwszy. Aby funkcja g_2 mogła zwrócić 0, wystarczył bowiem tylko jeden kontakt niewiążący pomiędzy łańcuchami. Z drugiej strony, poszukiwanie najbliższych sąsiadów atomów w krótszym promieniu przez funkcję g_3 było szybsze.

3.5.4. Funkcje oceny

Ocena wyników optymalizacji konformacji pary łańcuchów zgodnie ze wskazaniami kryteriów f_1 i f_2 oraz funkcji ograniczeń g_1, g_2 i g_3 polegała na porównaniu uzyskanych kompleksów z ich strukturami natywnymi. W tym celu przyjęto dwie miary podobnie do stosowanych w eksperymencie CAPRI: wartość RMSD oraz analizę map kontaktów niewiążących w przestrzeni krzywych ROC. Podstawowa różnica między nimi polegała na tym, że CAPRI jest konkursem, w związku z czym stosuje bardziej surowe zasady oceniania, które dostarczałyby zbyt małej ilości informacji potrzebnej do udzielenia odpowiedzi na stawiane w niniejszej rozprawie pytania.

Ponieważ materiałami dla eksperymentu były białka homodimeryczne, do wymiernej oceny wyników przewidywania ich struktur natywnych w oparciu o mapy kontaktów niewiążących wystarczyło, że każdej reszcie został przypisany jeden z dwóch statusów: „tak" albo „nie". Pierwszy z nich oznaczał, że znajdowała się w kontakcie niewiążącym z dowolną resztą z przeciwnego łańcucha, a drugi – że nie. Dzięki temu, para: algorytm optymalizacyjny i kryterium optymalizacyjne mogła być traktowana jako binarny „klasyfikator", starający się przypisać każdej reszcie z wyniku symulacji jej natywny status. Porównywanie par rozkładów takich statusów (wynikowego i natywnego) zostało przeprowadzone w przestrzeni krzywych ROC [409]. W tym celu, dla każdego białka zliczano zajścia następujących czterech sytuacji:

- prawdziwie dodatnie (true positive, TP): liczba reszt, które posiadały kontakty zarówno w kompleksie natywnym, jak i w wyniku symulacji,
- fałszywie dodatnie (false positive, FP): liczba reszt, które nie posiadały kontaktów w kompleksie natywnym, ale posiadały je w wyniku symulacji,
- fałszywie ujemne (false negative, FN): liczba reszt, które posiadały kontakty w kompleksie natywnym, ale nie posiadały ich w wyniku symulacji,
- prawdziwie ujemne (true negative, TN): liczba reszt, które nie posiadały kontaktów zarówno w kompleksie natywnym, jak i w wyniku symulacji.

Wartości TP, FP, FN I TN prezentują zgodności i różnice pomiędzy mapami kontaktów z wyniku symulacji i z kompleksu natywnego konkretnego białka. Aby móc porównać pod tym względem różne białka, obliczono na podstawie tych wartości dwie wartości pochodne. Pierwszą z nich jest czułość (true positive rate, TPR):

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3.16}$$

natomiast drugą jest 1-swoistość (false positive rate, FPR):

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{3.17}$$

Wartość TPR jest ułamkiem liczby kontaktów natywnych znajdujących się w wynikowym kompleksie. Wartość FPR wskazuje natomiast jaka część reszt, które natywnie nie posiadały kontaktów, uzyskały je w wyniku symulacji. Wspólnie, stanowią one parę współrzędnych w jednostkowym kwadracie przestrzeni krzywych ROC. Wartości FPR umieszcza się na jego osi poziomej, a TPR – pionowej.

Uzyskanie przez kompleks będący wynikiem symulacji pełnej zgodności z jego strukturą natywną odpowiada czułości i swoistości równym 1, czyli punktowi znajdującemu się w lewym górnym wierzchołku przestrzeni krzywych ROC. Oddalanie się od niego świadczy o coraz większej niedokładności, powodowanej przez niedobór prawdziwie dodatnich kontaktów (malejąca wartość TPR), lub nadmiar ich fałszywych odpowiedników (rosnąca wartość FPR). Dotarcie do przekątnej oznacza wynik porównywalny z losowym dopasowaniem łańcuchów do siebie [409], natomiast przejście na jej drugą stronę jest natomiast sygnałem, że resztom nadawany był status przeciwny do natywnego. Ze względu na użycie funkcji ograniczeń g_3 oraz globularny kształt większości białek z bazy danych, reszty z wnętrza łańcuchów nie miały możliwości znalezienia się kontaktach w niewiążących, co zapobiegało osiąganiu niskiej swoistości przez procedurę przewidywania struktury ich kompleksów.

Analogicznie do modelu FOD, w którym miara RD służy do przedstawiania relacji pomiędzy wartościami O‖T i O‖R, również wartości TPR i FPR mogły być zastąpione jedną, także znormalizowaną miarą – polem pod krzywą ROC (area under curve, AUC), powstałą w tym przypadku w wyniku połączenia dwoma odcinkami punktu $[\mathrm{FPR}, \mathrm{TPR}]$ z wierzchołkami $[0,0]$ i $[1,1]$ przestrzeni [409]:

$$\mathrm{AUC} = \frac{\mathrm{TPR} \cdot \mathrm{FPR}}{2} + \frac{(1+\mathrm{TPR})\,(1-\mathrm{FPR})}{2} = \frac{1+\mathrm{TPR}-\mathrm{FPR}}{2} \tag{3.18}$$

Wartość AUC równa 1 oznacza pełną zgodność map kontaktów, 0 – pełną niezgodność, a 0,5 – poziom losowego kompleksowania (przekątnej kwadratu).

Drugim narzędziem służącym do oceny wyników eksperymentu była miara RMSD, obliczana przy pomocy algorytmu Kabscha na podstawie współrzędnych wszystkich atomów węgla Cα. Stanowiła ona podstawę do stwierdzenia podobieństwa pomiędzy wynikowymi i natywnymi strukturami kompleksów, a także uzupełnienie dla miary ARC, pozwalające na wykrywanie dwóch niezauważalnych przez nią sytuacji. Pierwszą z nich były obroty liganda, które nie zmieniają stanu mapy kontaktów przechowującej informację na temat tego „czy”, a nie „z czym” kontaktują się reszty. Druga sytuacja dotyczyła natomiast łańcuchów, które zostały prawidłowo ułożone względem siebie, ale nie na tyle blisko, aby uzyskać wysoką wartość TPR.

Przyjęto, że wynikowy kompleks może być uznany za zbliżony do konformacji natywnej danego białka wtedy, gdy ich wartość RMSD będzie mniejsza od 10 Å, lub gdy pole pod krzywą ROC będzie większe od 0,75, czyli gdy punkt $[\mathrm{FPR}, \mathrm{TPR}]$ znajdzie się w sensie tej miary bliżej wierzchołka $[0,1]$ niż przekątnej kwadratu. Na tej podstawie, wynik ten mógł być zakwalifikowany do jednej z trzech kategorii:

1. RMSD $< 10\,\text{Å}$ i AUC $> 0{,}75$ – wysoka zgodność ze strukturą natywną,
2. RMSD $< 10\,\text{Å}$ albo AUC $> 0{,}75$ – potencjalna zgodność ze strukturą natywną,
3. RMSD $\geq 10\,\text{Å}$ i AUC $\leq 0{,}75$ – niezgodność ze strukturą natywną.

Ustalono, że wartości RMSD będą obliczane dla wszystkich atomów węgla Cα w kompleksach po nałożeniu na strukturę natywną całej struktury wynikowej. Zdecydowano się na takie podejście, zamiast typowego nałożenia na siebie samych receptorów, ponieważ eksperyment ten nie miał na celu sprawdzenia, czy możliwe jest dokładnie osiągnięcie konformacji natywnej, ale czy w wyniku optymalizacji kryterium pola zewnętrznego lub wewnętrznego może dojść do ułożenia łańcuchów względem siebie w taki sposób, że dalsza optymalizacja ich układu przy użyciu tych lub innych kryteriów będzie mogła do niej doprowadzić. Przyjęto, że uzyskanie takiego stanu powinno być na tym etapie badań wystarczające do stwierdzenia, że dane pole lub pola mogą mieć istotny wpływ na powstawanie danego kompleksu *in vivo*. Z tego powodu, zaprezentowane w niniejszej rozprawie wartości RMSD są niższe niż sugerują to wizualizacje odpowiadających im struktur, wykonane właśnie po nałożeniu na siebie samych receptorów. Takie pomieszanie pozwoliło jednak na ich łatwiejszą interpretację. Poniżej 10 Å, lub przy pokryciu ponad 75% przestrzeni krzywych ROC, do przekształcenia struktury wynikowej do natywnej okazały się bowiem zazwyczaj wystarczające proste translacje lub obroty cząsteczek ligandów.

Z użyciem dwóch miar do oceny wyników wiąże się jeszcze jeden problem. Nie można bowiem bezpośrednio na ich podstawie stwierdzić, który wynik jest bliższy strukturze natywnej: ten o niższej wartości RMSD, czy ten o wyższej wartości AUC? Niska wartość RMSD implikuje wysoką wartość AUC, ale może również okazać się, że do jej zmniejszenia wystarczy obrócić łańcuchy w sposób nie zmieniający zawartości mapy kontaktów. Postanowiono więc potraktować te miary jako równoważne. Pozwoliło to na wprowadzenie jeszcze jednej funkcji oceny wiążącej je ze sobą, która została z tego powodu nazwana ARC, od słów AUC and RMSD combined:

$$\text{ARC} = \sqrt{(1 - \text{AUC})^2 + \left(\frac{\text{RMSD}}{40\,\text{Å}}\right)^2} \tag{3.19}$$

Wartość ARC jest odległością punktu [RMSD, AUC] od punktu, w którym RMSD jest równe 0 Å, a AUC równe 1. Podzielenie RMSD przez 40 Å powoduje, że 10 Å zamienia się w 0,25, czyli tyle samo co $1 - 0{,}75$. Niektóre wyniki z kategorii pierwszej mogą przez to uzyskać niższe wartości ARC od tych z kategorii drugiej lub trzeciej.

liczba cząstek w roju	100
maksymalna liczba iteracji	600
współczynnik bezwładności (ϕ_v)	$U(0,1)$
współczynnik świadomości (ϕ_m)	2,0
współczynnik społeczności (ϕ_l)	2,0
topologia wyboru liderów roju	Von Neumanna (siatka 10×10 cząstek)
początkowe położenie cząstek	sześciowymiarowy hiperprostokąt
początkowa prędkość cząstek	rozmiar początkowego hiperprostokąta
maksymalna prędkość cząstek	rozmiar początkowego hiperprostokąta
warunek STOP	wykonanie maksymalnej liczby iteracji lub spadek średniej prędkości cząstek poniżej 0,5% rozmiaru początkowego hiperprostokąta

Tabela 3.8: Ustawienia algorytmu PSO w eksperymencie kompleksowania.

3.6. Kompleksowanie białek – jednokryterialne

W pierwszej części eksperymentu została wykonana analiza osobnego wpływu pól zewnętrznego i wewnętrznego na proces kompleksowania białek, czyli wtedy, gdy o konformacji danej pary łańcuchów decydowało jedno z tych kryteriów. Drugie było w tym czasie pomijane. Uzyskano w ten sposób 400 modeli, po dwa dla każdego kompleksu z bazy danych. Następnie obliczono odpowiadające im wartości RD i energii oraz porównano je ze strukturami natywnymi za pomocą miary RMSD i w przestrzeni krzywych ROC. Dane te znajdują się w tabeli B.2.

Optymalizacja globalna kryteriów f_1 i f_2 została przeprowadzona za pomocą klasycznego algorytmu PSO, obsługującego funkcje ograniczeń g_1, g_2 i g_3 przy pomocy strategii turniejowej Deba oraz stosującego ustawienia przedstawione w tabeli 3.8. Początkowe położenia cząstek były wybierane w sposób losowy z wnętrza hiperprostokąta o bokach równoległych do osi układu współrzędnych. Jego ściany ograniczające wartości zmiennych kątów obrotu $\{\theta, \phi, \psi\}$ zostały umieszczone w punktach $-\pi$ oraz π. Podstawą do wyznaczenia początkowych współrzędnych środka każdego liganda $\{x, y, z\}$ była natomiast suma długości najdłuższych średnic jego i receptora. Zdecydowano się na takie podejście w celu umożliwienia rojom cząstek dotarcia do wszystkich możliwych konformacji łańcuchów. Średnia długości połowy boków tak skonstruowanych hiperześcianów wyniosła 107,5 Å ($\sigma = 18{,}8$ Å). Jedynym punktem odstającym było ponownie białko `2SPC` (234,6 Å).

Algorytm PSO za każdym razem zakończył swoje działanie w dopuszczalnym rozwiązaniu. Tylko trzynastokrotnie nastąpiło to przed osiągnięciem maksymalnej liczby iteracji i tylko wtedy, gdy optymalizowane było kryterium pola zewnętrznego. Średnia prędkość roju spadła w jego przypadku poniżej progu warunku STOP we wszystkich wymiarach zmiennych położenia 141 razy, a w przynajmniej jednym – 194, osiągając średnio 0,62% ($\sigma = 0{,}4\%$) oczekiwanej wartości. Podczas optymalizacji kryterium pola wewnętrznego, liczby te wyniosły odpowiednio: 10, 73 oraz 1,76% i 0,92%. Średnia prędkość cząstek w wymiarach obrotu okazała się natomiast kilkukrotnie wyższa. Dla pola zewnętrznego wyniosła bowiem 2,14% ($\sigma = 1{,}45\%$) wartości warunku STOP, a dla pola wewnętrznego – 5,04% ($\sigma = 2{,}42\%$). Rojom optymalizującym pierwsze z tych kryteriów udało się zwolnić do 0,5% maksymalnej prędkości w trzech wymiarach 13 razy, a w przynajmniej jednym – 70. Pole wewnętrzne doprowadziło wyłącznie dwukrotnie do drugiej z tych sytuacji. Wyniki te potwierdzają, że krajobraz wartości RD modelu FOD, działającego na mniej licznym zbiorze danych wejściowych, zawiera w porównaniu z krajobrazem energetycznym mniejszą liczbę minimów lokalnych, a część z nich wyraźnie odróżnia się od pozostałych, mocniej przyciągając cząstki do siebie. Pozwala to algorytmowi optymalizacyjnemu na szybsze osiągnięcie zbieżności w wybranym przez niego punkcie oraz – teoretycznie – na łatwiejsze zbliżenie się do konformacji struktury natywnej.

3.6.1. Kontakty niewiążące

Efektem optymalizacji układu cząsteczek według kryterium pola zewnętrznego, biorącego pod uwagę całą jego zawartość, ale wyłącznie na poziomie reszt, było zwiększenie liczby kontaktów niewiążących w kompleksach. Przeciętna wartość ułamka ITF wyników symulacji i struktur natywnych wyniosła bowiem 1,319 ($\sigma = 0{,}871$). W statystyce tej nie zostało uwzględnione białko `1G17`, w którym zamiast jednego kontaktu wystąpiło ich aż 25. 87 par łańcuchów utworzyło kompleksy o mniej licznych interfejsach niż białka natywne (ułamek poniżej 1). Zgodnie z obserwacjami z rozdziału 3.4.6, w celu osiągnięcia minimum wartości RD, hydrofobowość obserwowana reszt musiała zbliżyć się do ich hydrofobowości teoretycznej. Mogło to nastąpić poprzez rozciąganie elipsoidy „kropli”, przybliżanie jej kształtu, lub zwiększanie liczby kontaktów. Funkcja ograniczeń g_2 nie pozwala na oddalanie się łańcuchów od siebie, co pozostawiło tylko drugą i trzecią z tych możliwości. Dopasowanie ich do kształtu elipsoidy „kropli” było możliwe wtedy, gdy były one podłużne lub wygięte. Niskie $\tilde{H}t$ reszt rekompensowała wówczas niska liczba kontaktów między nimi.

Porównując wartości ITF wyników kompleksowania według kryterium pola wewnętrznego i struktur natywnych stwierdzono, że przeciętna wartość ich ułamka wyniosła 0,447 ($\sigma = 0{,}388$). Oznacza to, że preferuje ono układy w których liczba kontaktów pomiędzy cząsteczkami jest niewielka, czyli wykazuje tendencję przeciwną do modelu FOD. Średnia ITF okazała się w jego przypadku równa 0,042 ($\sigma = 0{,}019$), co stanowi około $\frac{1}{3}$ jej średniej dla pola zewnętrznego: 0,127 ($\sigma = 0{,}041$).

Oddziaływania elektrostatyczne pomiędzy atomami wodoru powodują wzrost energii, któremu przeciwdziała odsuwanie łańcuchów od siebie. Powstaje w ten sposób na krajobrazie jej wartości jeszcze więcej atrakcyjnych minimów lokalnych, odciągających algorytm optymalizacyjny od bardziej zwartych konformacji. Nie należy jednak zakładać, że obliczenia energii dla kompletnej cząsteczki (bez stosowania promienia odcięcia) uniknęłyby tych samych problemów. Rozwiązaniem byłoby wprowadzenie dodatkowych funkcji ograniczeń wymuszających obecność określonej liczby kontaktów, choć mogłoby to za bardzo ingerować w kryteria f_1 i f_2, lub jeszcze bardziej ograniczać już i tak mocno ograniczony zbiór dopuszczalnych rozwiązań. Obserwacje te potwierdzają słabość pól wewnętrznych w kwestiach dotyczących samodzielnego rozróżniania pomiędzy konformacjami podobnymi do siebie pod względem energii, zwłaszcza, gdy cząsteczki są traktowane jako bryły sztywne.

3.6.2. Symetria kompleksu

Kolejnym etapem analizy wyników tej części eksperymentu była kwestia symetrii uzyskanych kompleksów. Posługując się wcześniej przyjętymi kryteriami, wartość ICF wyższą od 0,5 zaobserwowano 60 razy dla pola zewnętrznego i 8 dla pola wewnętrznego, natomiast niższą od 0,25 – w odpowiednio 106 oraz 176 przypadkach. Osiągnęła ona dokładnie 0 podczas 74 i 159 symulacji. Ponownie nie zaobserwowano korelacji pomiędzy symetrią interfejsów a liczbą tworzących je reszt. Identyczna zależność dotyczyła rozkładów wartości ICF i RD (zarówno kompleksu jak i łańcuchów). Jednak w odróżnieniu od struktur natywnych, spośród których wszystkie 9 zgodnych z modelem FOD było symetryczne, 24 wyniki symulacji o RD mniejszym od 0,5 wykazały tę cechę, ale pojawiło się również 27 o przeciwnej charakterystyce. Oznacza to, że dążenie do stabilizacji jądra hydrofobowego pary łańcuchów nie musi prowadzić do ich układania w sposób zwiększający wartość ICF, nawet jeśli są identyczne. O RD kompleksu decydują bowiem wszystkie tworzące go reszty. Miara ICF jest jednak dość rygorystyczna, tak więc możliwe, że część wyników o niskiej wartości ITF może charakteryzować się również przynajmniej częściową symetrią.

3.6.3. Wartości RD i energii

Podczas optymalizacji kryterium pola zewnętrznego, 195 razy zostały uzyskane konformacje o niższym RD niż w kompleksach natywnych. Przeciętna różnica wyniosła w ich przypadku 0,104 ($\sigma = 0{,}056$), a u pozostałych pięciu – 0,017 ($\sigma = 0{,}009$). Sytuację tę obrazuje rysunek 3.18a. Największa różnicę, aż 0,308: z 0,723 do 0,416, zaobserwowano w przypadku białka `3N8E`, w którym model FOD „zamknął" rozwartą strukturę kompleksu, zwiększając liczbę kontaktów pomiędzy łańcuchami z 7 do 25. W skali całej bazy danych oznacza to, że dla niemal każdej cząsteczki można wskazać konformację charakteryzującą się wyższą stabilnością od jej postaci natywnej. Zauważono również dodatnią korelację pomiędzy ich wartościami RD, o współczynniku równym 0,657, która implikacje są omówione poniżej.

Zaobserwowano następujące relacje statusu zgodności z modelem FOD:

- 7 razy kompleksy natywne oraz wynikowe były zgodne,
- 147 razy kompleksy natywne oraz wynikowe były niezgodne,
- 44 razy kompleksy natywne były niezgodne, ale wynikowe okazały się zgodne,
- 2 razy kompleksy natywne były zgodne, ale wynikowe okazały się niezgodne.

W pierwszej z powyższych grup, za każdym razem osiągnięte zostały niższe wartości RD niż w białkach natywnych, średnio o 0,035 ($\sigma = 0{,}027$), natomiast w ostatniej bliskie 0,5: 0,509 i 0,513. Choć 9 kompleksów to niewielka próba, wyniki te sugerują, że odnalezienie konformacji zgodnych z modelem FOD, gdy na ich obecność wskazuje eksperyment krystalograficzny, jest możliwe dzięki optymalizacji globalnej kryterium pola zewnętrznego. Wśród nich znalazła się najbardziej pasująca do oczekiwań teoretycznych w bazie danych hydrolaza `2D3K` (z 0,398 do 0,373).

Druga grupa zawiera białka, w przypadku których nie udało się osiągnąć RD mniejszego niż 0,5 jeżeli było ono wyższe od tej wartości w strukturach natywnych. Sugeruje to, że łańcuchy je tworzące nie są w stanie utworzyć kompleksu zgodnego z modelem FOD, a tym samym wspólnie wykształcić stabilnego jądra hydrofobowego, o ile nie dochodzi w nich do nieprzewidywalnych w tym eksperymencie zmian konformacyjnych, takich jak dynamika kształtu odcinków pętli. Średnia wartości RD w tej grupie wyniosła 0,571 ($\sigma = 0{,}043$). Białek znajdujących się w jej sensie w odległości mniejszej niż 0,01 od 0,5 było tylko 14. Zjawisko to zdaje się również nie zależeć od stabilności samych łańcuchów: wartości RD mniejsza od 0,5 charakteryzowała przynajmniej jeden łańcuch w 106 parach z 147.

Najciekawsza wydaje się jednak trzecia grupa. Znajdują się w niej 44 przykłady sytuacji gdy istnieje możliwość utworzenia kompleksu zgodnego z modelem FOD, pomimo tego, że jego natywna postać jest z nim niezgodna. Średnia różnica wartości RD wyniosła tutaj 0,140 ($\sigma = 0{,}06$). Wynika to stąd, że niezależnie od tego, jak wysoka była ta wartość w kompleksie natywnym (nawet ponad 0,7 w przypadku wspomnianego wcześniej białka `3N8E`), algorytm PSO był w stanie odnaleźć konformację, dla której zawierała się ona w przedziale od 0,4 do 0,5. Dlatego po ich pominięciu, korelacja wartości RD wśród pozostałych wzrosła do 0,704. Jedno białko osiągnęło w jej sensie wyższy poziom zgodności modelem niż `2D3K`: 0,338 z 0,582. Ma ono identyfikator `2DCT` i pełni do tej pory nieustaloną funkcję w *Thermus thermopilus*. Również w tej grupie białek nie zaobserwowano związku pomiędzy stabilnością łańcuchów i kompleksów. Wartością RD poniżej 0,5 charakteryzował się przynajmniej jeden z nich w 27 parach. W odróżnieniu od poprzedniej grupy, której elementy nie mają możliwości osiągnięcia struktury o stabilnym jądrze hydrofobowym, do pełnienia właściwej funkcji, kompleksy te muszą przyjąć i utrzymać konformacje celowo niezgodne z modelem FOD.

Optymalizacja kryterium pola wewnętrznego doprowadziła za każdym razem do uzyskania ujemnej energii oddziaływań niekowalencyjnych pomiędzy łańcuchami. Przeciętnie wniosła ona $-275{,}632\ \frac{\text{kcal}}{\text{mol}}$ ($\sigma = 79{,}238\ \frac{\text{kcal}}{\text{mol}}$), a 182 razy okazała się niższa niż w strukturach natywnych, średnio o $253{,}918\ \frac{\text{kcal}}{\text{mol}}$ ($\sigma = 151{,}205\ \frac{\text{kcal}}{\text{mol}}$). Największą różnicę, $729{,}143\ \frac{\text{kcal}}{\text{mol}}$ (pomiędzy $509{,}698\ \frac{\text{kcal}}{\text{mol}}$ a $-219{,}445\ \frac{\text{kcal}}{\text{mol}}$) zaobserwowano w przypadku białka `3IQ3` [410], natomiast najniższą wartość w białku `1I4S` [411]: $-674{,}565\ \frac{\text{kcal}}{\text{mol}}$. Sytuacja ta, widoczna na rysunku 3.19a, jest zbliżona do wyników optymalizacji pola zewnętrznego, ale w odróżnieniu od nich, nie zaobserwowano tu korelacji pomiędzy wartościami energii. W pozostałych 18 strukturach natywnych była ona niższa od $-250\ \frac{\text{kcal}}{\text{mol}}$ (najmniej, $-902{,}385\ \frac{\text{kcal}}{\text{mol}}$, w białku `3TW2` [412]). Względnie liczne interfejsy P-P ich łańcuchów sugerują, że konformacje te odpowiadają trudno dostępnym, możliwie globalnym minimom energii, otoczonym przez wysokie maksima lokalne oddziaływań van der Waalsa. Algorytm PSO nie dotarł do nich, gdyż zgodnie z wcześniejszymi ustaleniami, jest mocniej przyciągany do stanów układu o niewielkiej liczbie kontaktów pomiędzy łańcuchami. Sugeruje to trzynaście białek, w których energia obniżyła się w trakcie optymalizacji do wartości niższych od $-400\ \frac{\text{kcal}}{\text{mol}}$, ale wartość ITF przekroczyła 0,1 tylko raz.

Po wykonaniu eksperymentu kompleksowania według kryterium każdego z pól, sprawdzono również ile wyniosły wartości drugiego z nich dla otrzymanych przez nie struktur (energia wyniku pola zewnętrznego i RD wyniku pola wewnętrznego).

Podczas analizy liczby kontaktów niewiążących pomiędzy łańcuchami, stwierdzono, że kryteria modelu FOD i pola ECEPP/3 wykazują tendencje przeciwne – kierują algorytm optymalizacyjny w stronę konformacji o odpowiednio bardziej lub mniej licznych interfejsach łańcuchów. Porównanie uzyskanych przy ich pomocy wyników miało na celu zaobserwowanie wpływu tego zachowania na wartości RD i energii. Gdy jedna rośnie, druga powinna maleć i odwrotnie.

W przypadku pola zewnętrznego, przeciętna suma energii wszystkich trzech potencjałów okazała się równa 2714,852 $\frac{\text{kcal}}{\text{mol}}$, przy odchyleniu standardowym 959,813 $\frac{\text{kcal}}{\text{mol}}$. Tylko dwa razy osiągnęła ona wartość poniżej 100 $\frac{\text{kcal}}{\text{mol}}$, nigdy poniżej zera. Powodem tego były kolizje, w których brały udział atomy wodoru, przekładające się na wzrost wartości potencjału van der Waalsa. Bez ustalenia minimalnych odległości pomiędzy atomami, energia ta byłaby jeszcze wyższa, liczona nawet w miliardach $\frac{\text{kcal}}{\text{mol}}$. Kolizje atomów wynikają z ignorowania ich przez model FOD oraz tego, że hydrofobowość obserwowana reszt rośnie wraz ze zwiększaniem się liczby najbliższych sąsiadów atomów efektywnych. Tym samym, optymalizacja kryterium pola zewnętrznego wiąże się ze ścisłym upakowaniem łańcuchów w kompleksie. Dwa białka w których to nie nastąpiło, `1VLT` [413] i `4EC7` [414], przyjęły strategię alternatywną, polegającą na uzyskaniu przez ich kompleksy kształtu zbliżonego do elipsoidy „kropli”. Po odjęciu od całkowitej energii sumy oddziaływań van der Waalsa, jej średnia zmalała do 50,865 $\frac{\text{kcal}}{\text{mol}}$ ($\sigma = 115{,}739\ \frac{\text{kcal}}{\text{mol}}$), choć tylko 69 razy stała się ujemna. Powodem tego były oddziaływania elektrostatyczne – energia potencjału wiązań wodorowych tylko cztery razy nieznacznie przekroczyła 0 (do 2 $\frac{\text{kcal}}{\text{mol}}$).

W przypadku optymalizacji kryterium pola wewnętrznego, tylko jeden kompleks osiągnął wartość RD niewiele niższą od 0,5: 0,499. Okazało się nim wspomniane kilka paragrafów wcześniej białko `3IQ3`. Wysoka średnia RD wszystkich wyników, wynosząca 0,694 ($\sigma = 0{,}055$), sugeruje, że zgodnie z wcześniejszymi przewidywaniami, optymalizacja wartości energii spowodowała utworzenie kompleksów wysoce niezgodnych z modelem FOD, czego główną przyczyną było odsunięcie łańcuchów od siebie, a przez to ograniczenie dostępu do wzajemnie dostarczanej przez nie hydrofobowości. Pięć razy uzyskana została nawet wartość wyższa od 0,8. Nie był to jednak efekt samego oddalenia łańcuchów od siebie, ale również ich specyficznego ułożenia rozszerzającego elipsoidę „kropli” przy jednoczesnym utrudnieniu dopasowania jej do atomów efektywnych. Sytuacja ta dotyczyła kompleksów, w których łańcuchy zostały ułożone prostopadle do siebie lub w kształcie litery „T”.

Wniosek z powyższych obserwacji jest taki, że jeżeli obydwa pola oddziałują na układ niezależnie, to kierują go w stronę stanów o przeciwnej charakterystyce.

3.6.4. Zgodność ze strukturami natywnymi

Ostatnim i zarazem najważniejszym etapem analizy wyników optymalizacji globalnej kryteriów pól zewnętrznego i wewnętrznego było sprawdzenie ich zgodności ze strukturami natywnymi. Dla każdego z nich wyznaczono wartości RMSD i AUC w taki sposób jak zostało to przedstawione w części 3.5.4. Na podstawie obserwacji uzyskanych danych, z trzeciej z wymienionych w niej kategorii (gdy RMSD ≥ 10 Å i AUC $\leq 0{,}75$ – brak zgodności) zostały dodatkowo wydzielone jeszcze dwie podkategorie. Podstawowym kryterium zakwalifikowania wyników do nich była wartość AUC jednego łańcucha większa, a drugiego mniejsza lub równa właśnie 0,75. Jego spełnienie oznaczało, że tylko połowa kompleksu została ułożona w sposób wyraźnie zbliżony do oczekiwanego. Ponieważ analizowane są tu białka homodimeryczne, nie ma znaczenia, który z danej pary znalazł się w tej sytuacji. Drugie kryterium dotyczyło natomiast faktu, że nie można traktować tak samo trzech czwartych przestrzeni krzywych ROC. Łańcuch, którego wartość AUC wyniosła 0,74 znajduje się bowiem w zupełnie innej relacji ze swoim partnerem niż ten, który nie przekroczył pod tym względem 0,5. Dlatego użyto właśnie tego progu do rozróżnienia pomiędzy obydwoma tymi podkategoriami. Przyjęto, że jeżeli wartość AUC drugiego łańcucha należy do przedziału $(0{,}5, 0{,}75)$, należy uznać, że wykazuje tendencję podobną do pierwszego, choć nie osiągnął tak samo wysokiej zgodności. Oczywiście, jeśli wystarczyło to do przekroczenia 0,75 w skali całej mapy kontaktów, taki wynik był przydzielany do kategorii trzeciej. Jeżeli wartość AUC drugiego łańcucha okazała się jednak mniejsza od 0,5, uznawano, że według stosowanego kryterium, dany kompleks miał powstać w sposób połowicznie zgodny ze strukturą natywną. Jest to inna sytuacja niż w przypadku kategorii drugiej. Implikuje ona bowiem wartość RMSD wyższą od 10 Å. Do oceny zmiany stanu symetrii posłużono się miarą ICF.

Wartości RMSD oraz AUC oceny zgodności wyników eksperymentu ze strukturami natywnymi są przedstawione w sposób graficzny na rysunkach 3.18b i 3.18c oraz 3.19b i 3.19c. Pierwsza para dotyczy pola zewnętrznego a druga – wewnętrznego. Te z nich, w przypadku których stwierdzono RMSD mniejsze od 10 Å lub AUC większe od 0,75 (dla kompleksu lub jednego łańcucha) zostały dodatkowo umieszczone odpowiednio na rysunkach 3.18a i 3.19a. Wybrane informacje na ich temat z tabeli B.2 znajdują się w tabelach 3.9, 3.10, 3.11, 3.12, a wizualizacje – w rozdziale A.2. Wykonana została również analiza zgodności pomiędzy wynikami dla tych samych białek uzyskanymi poprzez optymalizację różnych pól. Ich graficzna reprezentacja jest widoczna na rysunku 3.20, a odpowiadające im dane z tabeli B.2 – w tabeli 3.13.

Optymalizacja kryterium pola zewnętrznego przy pomocy algorytmu PSO doprowadziła do utworzenia 7 kompleksów o wysokiej zgodności z ich strukturami natywnymi (RMSD < 10 Å i AUC $> 0{,}75$), 9 o RMSD niższym od 10 Å oraz 3 o AUC większym od 0,75. Wizualizacje wszystkich z nich znajdują się na rysunkach A.13, A.14 i A.15. Na tej podstawie można stwierdzić, że oddziaływania hydrofobowe mają kluczowe znaczenie dla procesu powstawania struktury czwartorzędowej *in vivo* 7 spośród 200 wybranych białek homodimerycznych (rozumianych jako te o wartości RMSD mniejszej od 5 Å), natomiast łącznie niemal w co dziesiątym z nich kierują układ w stronę konformacji zbliżonych do natywnych. Do uzyskania jeszcze wyższej zgodności wystarczające były bowiem proste zmiany orientacji liganda względem receptora. Pokazuje to w jaki sposób przestrzeń konformacyjna eksperymentu *ab initio* może być skutecznie ograniczona przez metody heurystyczne i pole zewnętrzne do jej podzbioru, umożliwiającego rozpoczęcie w nim dokładnej optymalizacji przy pomocy innych metod optymalizacyjnych lub kryteriów.

Wynik eksperymentu kompleksowania białka `2W2A` [415] jako jedyny osiągnął wartość RMSD niższą od 1 Å, dokładnie 0,941 Å. Oznacza to, że ten pochodzący z *Lactobacillus plantarum* kompleks powstaje *in vivo* zgodnie z kryterium modelu FOD. Jego natywna wartość RD wyniosła 0,531 (0,509 podczas symulacji) co wnika stąd, że pełni on funkcję enzymu – dekarboksylazy.

Za wyjątkiem białka `1G17`, które posiadało wyłącznie jeden kontakt niewiążący pomiędzy łańcuchami, wszystkie omawiane tu struktury były natywnie symetryczne (najniższa wartość ICF wyniosła 0,857) i o względnie licznych interfejsach (średnia ITF 0,147 przy odchyleniu standardowym 0,041). Symetrię wynikowych kompleksów stwierdzono 13 razy. Wartość ICF 11 z nich była wyższa od 0,5, natomiast pełną asymetrię (ICF $= 0$) zaobserwowano trzykrotnie. Średnia i odchylenie standardowe wartości ITF okazały się zbliżone do białek natywnych i wyniosły odpowiednio 0,146 i 0,032. Nie wystąpiła jednak korelacja pomiędzy jej rozkładami.

Czternastokrotnie średnia wartość RD łańcuchów była niższa od 0,5. Wśród tych białek, stwierdzono zgodność kompleksu natywnego z modelem FOD 3 razy. Wyniki eksperymentu dla każdego z nich również osiągnęły wartości RD niższe od 0,5. W przypadku pozostałych, czterokrotnie uzyskana została konformacja zgodna z modelem FOD, choć ich struktury natywne były z nim niezgodne. W przeciwieństwie do miary ITF, zaobserwowano wysoką korelację pomiędzy wartościami RD. Jej współczynnik dla struktur natywnych i wynikowych wyniósł bowiem 0,916. Podobnie wyglądała sytuacja w przypadku wartości energii. Pomijając oddziaływania van der Waalsa, współczynnik korelacji jej rozkładów okazał się równy 0,733.

Na rysunku 3.18a widać, że większość omawianych tu wyników charakteryzowała się względnie niską wartością RD kompleksu natywnego (bliską 0,5 lub niższą). Sugeruje to, że im bardziej rozkłady hydrofobowości teoretycznej i obserwowanej danego białka są do siebie podobne w sensie miary D_{KL}, tym wyższe jest prawdopodobieństwo trafnego przewidzenia jego struktury poprzez optymalizację kryterium pola zewnętrznego. Za przyczynę tego zjawiska należy uznać ograniczone możliwości przyjmowania przez nie konformacji, które posiadałyby zdecydowanie niższe wartości RD. W związku z tym, w przypadku takich białek, problemu nie stanowi wybór właściwego ułożenia tworzących je łańcuchów względem siebie, ale ich dokładne dopasowane (liczny interfejs kontaktów niewiążących), satysfakcjonujące jednocześnie funkcję ograniczeń g_2. Z drugiej strony, wraz z rosnącą wartością RD kompleksu natywnego, rośnie liczba konformacji alternatywnych lub siła z jaką odciągają algorytm optymalizacyjny od tej właściwej. Oznacza to, że hydrofobowość nie decyduje samodzielnie o strukturze czwartorzędowej tych cząsteczek. Osiągnięcie i utrzymanie przez nie równowagi musi więc wynikać z równoczesnego wpływu innych czynników. Odpowiedź na pytanie, czy zalicza się do nich energia potencjałów niekowalencyjnych znajduje się w następnej części rozdziału.

Jeżeli struktura natywna kompleksu nie jest znana, co ma miejsce w eksperymencie CAPRI, do oszacowania możliwości jej przewidzenia w oparciu o samo pole zewnętrzne można posłużyć się zgodnością łańcuchów z modelem FOD. Nawet wtedy, gdy nie będą traktowane jako bryły sztywne, ich charakterystyka w jego ujęciu nie powinna ulegać zbyt silnym zmianom. Na tej podstawie można wnioskować, jaki może być status tworzonej przez nie cząsteczki. Podczas analizy białek z bazy danych zauważono korelację pomiędzy wartościami RD łańcuchów i kompleksów, której współczynnik wyniósł 0,517. Tutaj wzrósł on do 0,799.

Pozostaje jeszcze kwestia wyników, w których jeden łańcuch uzyskał wartość AUC wyższą od 0,75, a drugi niższą. Sytuacja ta wystąpiła podczas 44 symulacji. Wartość AUC drugiego łańcucha przekroczyła poziom 0,5 17 razy. Spowodowało to zakwalifikowanie uzyskanych kompleksów do drugiej kategorii zgodności ze strukturami natywnymi trzykrotnie, a trzeciej – dwukrotnie. Wśród pozostałych dwunastu, AUC kompleksu okazało się wyższe od 0,7 sześć, a od 0,74 – dwa razy. Oznacza to, że białka te wykazały tendencję do kierowania się pod wpływem oddziaływań hydrofobowych ku swoim strukturom natywnym. Średnia wartość RMSD wyniosła w ich przypadku jednak 15,429 Å ($\sigma = 2{,}180$), co sugeruje, że gdyby osiągnęły zgodność ze strukturami natywnymi w sensie przyjętych funkcji oceny, prawdopodobnie zostałyby umieszczone w jej trzeciej kategorii (RMSD ≥ 10 Å, AUC $> 0{,}75$).

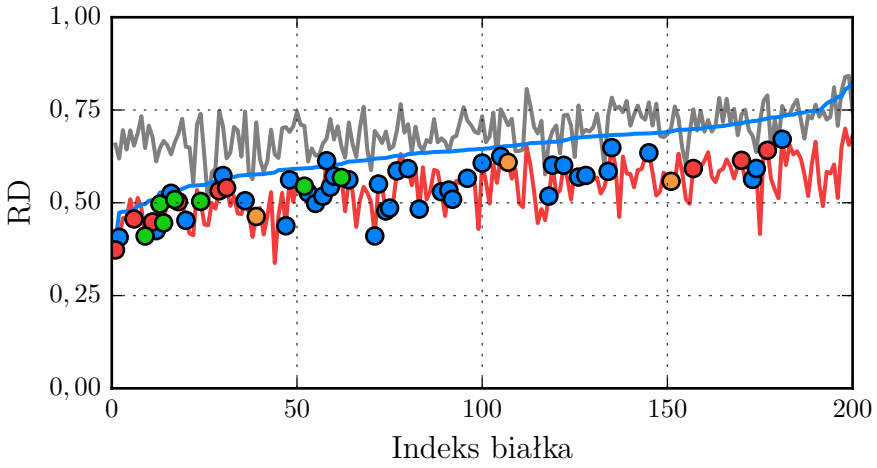
Zaobserwowano 27 wyników, w których wartość AUC jednego łańcucha była wyższa od 0,75 a drugiego niższa od 0,5. Stwierdzono asymetrię 24 z nich. W tej grupie, kompleks natywny były asymetryczny dwukrotnie. Oznacza to, że 22 razy jego symetria została utracona w trakcie symulacji. Wartość ICF osiągnęła 0 w 16 z tych przypadków. Sytuacja przeciwna, gdy kompleks natywny był asymetryczny, a wynik eksperymentu symetryczny wystąpiła trzy razy. Jak można było się spodziewać, wartość RMSD niemal wszystkie tych wyników była względnie wysoka (średnia 14,936 Å przy odchyleniu standardowym 1,753 Å) – tylko raz nie przekroczyła 11 Å. Również wszystkie poza jednym białkiem były natywnie niezgodne z modelem FOD – zgodne z nim stało się 7. Nie stwierdzono, aby w którymkolwiek z tych wyników doszło do utworzenia liniowej formy kompleksu. Należy przez to rozumieć konformacje eksponujące wolny interfejs natywny jednego łańcucha pozwalający na dołączenie kolejnych cząsteczek. Jest to zrozumiałe, gdyż w białkach homodimerycznych, ułożenie łańcuchów w sposób liniowy, jeden za drugim, nie może prowadzić do wyraźnego zmniejszenia różnic pomiędzy rozkładami hydrofobowości.

Liczba wyników, w których wartość AUC obydwu łańcuchów okazała się mniejsza od 0,5 wyniosła 62. Oznacza to, że ich natywne interfejsy zostały skierowane w przeciwne strony, z czego w 43 razy wszystkie kontakty były niewłaściwe (TPR = 0). Przykładem takiej struktury jest jedyne białko, które osiągnęło zgodność z modelem FOD podczas optymalizacji kryterium pola wewnętrznego – `3IQ3` [410]. Znajduje się ono najbliżej punktu $[0, 0]$ na rysunku 3.18c.

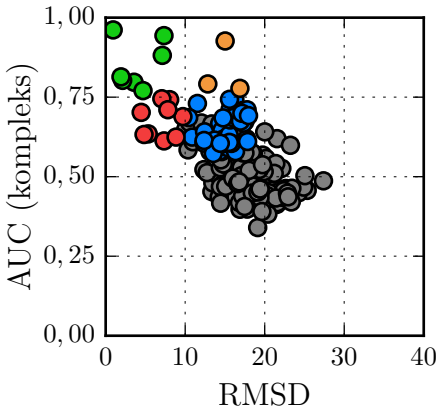
Skoro poruszona została kwestia pola wewnętrznego, optymalizacja tego kryterium doprowadziła do utworzenia tylko 7 kompleksów, których wartość RMSD oceny zgodności ze strukturami natywnymi była niższa od 10 Å. Najniższa spośród nich wyniosła 4,140 Å. Jest to ponad cztery razy więcej niż najlepszy wynik uzyskany przez pole zewnętrzne (0,941 Å – `2W2A`). We wszystkich siedmiu przypadkach, wartość AUC kompleksu nie przekroczyła progu 0,75. Próg ten osiągnął jeden łańcuch tylko w dwóch białkach. Na rysunku 3.19a widać, że wyniki te charakteryzowały się natywnie niską energią. Nie znajdują się one jednak na samym początku wykresu, gdyż jak już ustalono, dotarcie do tych konformacji jest trudnym zadaniem dla algorytmu optymalizacyjnego. 6 białek posiadało łańcuchy zgodne z modelem FOD oraz kompleksy natywnie niezgodne. 6 ich par charakteryzowało się symetrycznym interfejsem, który w 4 przypadkach stał się niesymetryczny. Obserwacje te potwierdzają niską dokładność metody przewidywania struktury czwartorzędowej białek w oparciu o samo pole wewnętrzne, choć pokazują również, że istnieją wśród nich takie, których kształt krajobrazu wartości energii pozwala na dotarcie do ich struktur natywnych.

Podczas analizy dwóch wyników charakteryzujących się wartością AUC jednego łańcucha wyższą od 0,75, a drugiego niższą od 0,5, zauważono, że białko `1FTP`, którego AUC kompleksu przekroczyło 0,7, a RMSD wyniosło 11,273 Å, miało najwyższą szansę spośród wszystkich 200 na zakwalifikowanie nawet do pierwszej kategorii zgodności ze swoją strukturą natywną. Sytuację tę widać dobrze na rysunku 3.19b. Nie udało mu się to jednak z powodu stwierdzonej wcześniej tendencji pola wewnętrznego do odsuwania łańcuchów od siebie. Drugie białko, `2OFC`, prezentuje natomiast inną właściwość tego kryterium, czyli dopuszczanie do tworzenia się liniowych form kompleksów. Optymalizacja konformacji jego łańcuchów doprowadziła bowiem do ustawienia ich obok siebie i obrócenia w identyczny sposób (symetria translacyjna). Powoduje to, że bez wpływu innych sił, destabilizujących ten stan układu, kolejne łańcuchy mogłyby być do nich dołączane z jednej lub z drugiej strony, tworząc oligomer. Obserwacja pozostałych kompleksów potwierdziła obecność wśród wyników symulacji jeszcze 7 takich struktur. Wśród białek natywnych były tylko dwa kompleksy posiadające tę właściwość: `1G17` i `1C77` [237].

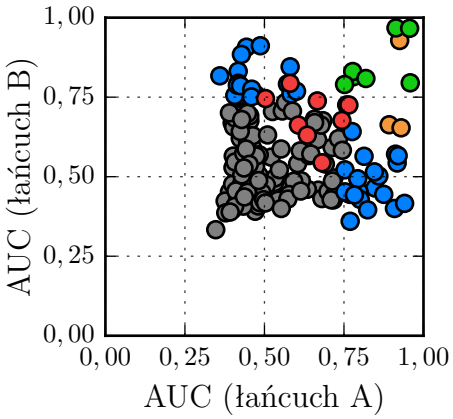
Ostatnia kwestia tej analizy dotyczyła zgodności pomiędzy samymi wynikami eksperymentu. Jej celem było sprawdzenie, czy istnieją kompleksy, w przypadku których pola zewnętrzne i wewnętrzne doprowadziły do przyjęcia przez ich łańcuchy podobnych konformacji. Wykonane poprzednio porównanie wartości RD i energii wykazało, że kryteria te prowadzą algorytm optymalizacyjny w przeciwnych kierunkach. Zgodnie z oczekiwaniami, okazało się, że zjawisko to dotyczy również struktur kompleksów jakie zostały dzięki nim uzyskane. Prezentacja graficzna ich porównania przy pomocy miar RMSD i AUC znajduje się na rysunku 3.20, a wizualizacje – na rysunku A.16. Średnia wartość pierwszej z tych miar wyniosła 17,778 Å ($\sigma = 3{,}149$ Å), a drugiej 0,507 ($\sigma = 0{,}042$). Ze względu na większą liczbę kontaktów niewiążących, za wzorzec przyjęto wyniki pola zewnętrznego. Na rysunku 3.20 widać tylko jedno białko zakwalifikowane do pierwszej kategorii zgodności. Jego RMSD było faktycznie niższe od 10 Å, ale tylko jeden łańcuch osiągnął wartość AUC wyższą od 0,75 – w drugim nie przekroczyła ona 0,5. Jest to jedyna taka sytuacja w całej bazie danych. W pozostałych AUC obydwu łańcuchów znajdowało się w najwyższej ćwiartce przedziału wartości tej miary. W związku z tym, wynik ten został przeniesiony do kategorii drugiej, w której dołączył do 8 innych struktur. Wszystkie z nich okazały się niezgodne z odpowiadającymi im natywnymi postaciami kompleksów. Średnia wartość RD łańcuchów była niższa od 0,5 w 7 przypadkach. Każda para była natomiast natywnie niezgodna z modelem FOD, choć podczas optymalizacji kryterium pola zewnętrznego, cztery z nich stały się zgodne.



(a) Wartości RD.

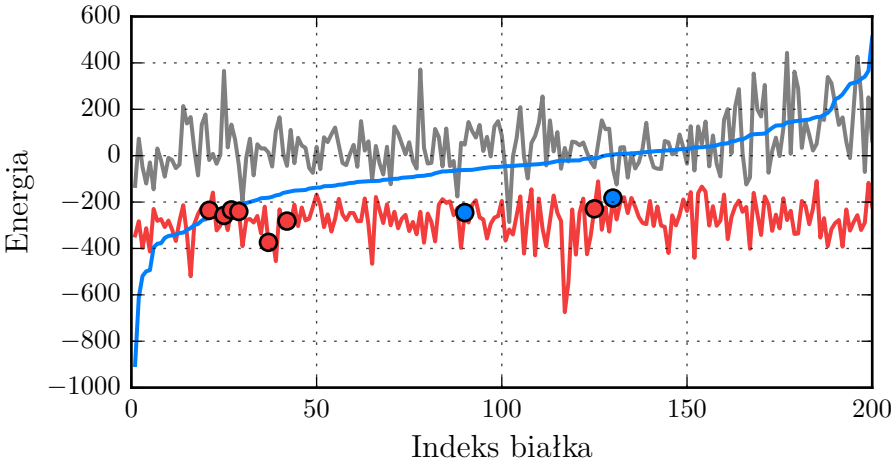


(b) Zgodność kompleksów.

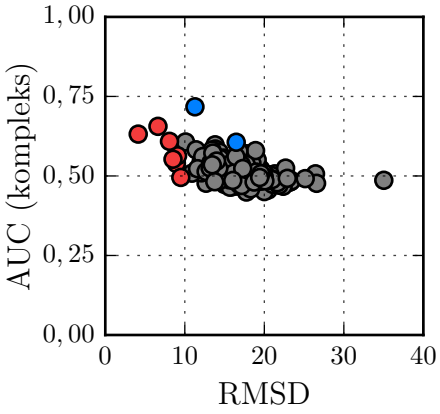


(c) Zgodność łańcuchów.

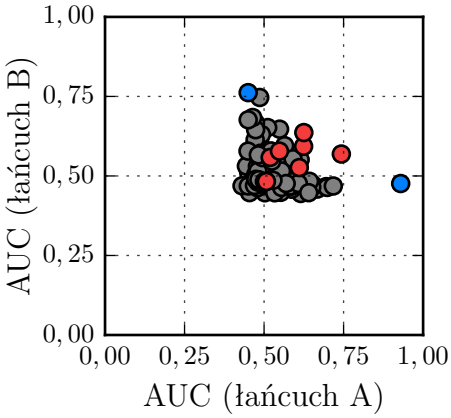
Rysunek 3.18: Wartości RD oraz miar oceny wyników optymalizacji globalnej kryterium pola zewnętrznego. Kolory rozkładów na rysunku a oznaczają rodzaj struktury: niebieski – natywna, czerwony – wynikowa, szary – wynikowa, ale uzyskana podczas optymalizacji kryterium pola wewnętrznego. Wszystkie rozkłady są posortowane zgodnie z rosnącymi wartościami RD struktur natywnych. Każdy znacznik odpowiada jednemu białku. Ich kolory wskazują natomiast na kategorie zgodności ze strukturami natywnymi: zielony – RMSD $< 10\,\text{Å}$ i AUC $> 0{,}75$, czerwony – RMSD $< 10\,\text{Å}$ i AUC $\leq 0{,}75$, pomarańczowy – RMSD $\geq 10\,\text{Å}$ i AUC $> 0{,}75$, niebieski – AUC $> 0{,}75$ tylko w jednym łańcuchu, szary – brak zgodności.



(a) Wartości energii.



(b) Kompleksy.



(c) Łańcuchy.

Rysunek 3.19: Wartości RD oraz miar oceny wyników optymalizacji globalnej kryterium pola wewnętrznego. Kolory rozkładów na rysunku a oznaczają rodzaj struktury: niebieski – natywna, czerwony – wynikowa, szary – wynikowa, ale uzyskana podczas optymalizacji kryterium pola zewnętrznego (tylko $E_e + E_h$). Wszystkie rozkłady są posortowane zgodnie z rosnącymi wartościami energii struktur natywnych. Każdy znacznik odpowiada jednemu białku. Ich kolory wskazują natomiast na kategorie zgodności ze strukturami natywnymi: czerwony – RMSD $< 10\,\text{Å}$ i AUC $\leq 0{,}75$, niebieski – AUC $> 0{,}75$ tylko w jednym łańcuchu, szary – brak zgodności.

PDB	Kryteria		Ocena					Rodzaj cząsteczki
ID	RD	Energia	RMSD	C	A	B	ARC	(pełniona funkcja)
2W2A	0,509	31,429	0,941	0,961	0,956	0,966	0,05	p-coumaric acid decarboxylase
2OE3	0,446	78,063	7,331	0,943	0,914	0,968	0,19	thioredoxin-3
1VC1	0,410	265,688	1,926	0,814	0,819	0,809	0,19	putative anti-sigma factor antagonist tm1442
1TLJ	0,504	312,466	2,136	0,803	0,777	0,831	0,20	hypothetical upf0130 protein sso0622
1NWP	0,568	49,265	7,119	0,881	0,958	0,795	0,21	azurin
1SGM	0,545	87,141	3,567	0,797	0,780	0,814	0,22	putative hth-type transcriptional regulator yxaf
3AIA	0,497	443,178	4,695	0,771	0,752	0,790	0,26	upf0217 protein mj1640

Tabela 3.9: Wartości RD i energii oraz miar oceny wyników optymalizacji globalnej kryterium pola zewnętrznego, dla których RMSD < 10 Å i AUC $> 0{,}75$. Wiersze są posortowane zgodnie z rosnącymi wartościami miary ARC. Wyjaśnienie nagłówków kolumn: *PDB ID* – identyfikator struktury w bazie PDB; *Kryteria* – wartości RD i energii; *Ocena* – wartości RMSD i AUC oceny zgodności kompleksu (C) i łańcuchów (A, B) oraz obliczona na ich podstawie wartość ARC; *Rodzaj cząsteczki (pełniona funkcja)* – rodzaj cząsteczki białka wskazany w rekordzie COMPND z pliku PDB.

PDB	Kryteria		Ocena					Rodzaj cząsteczki
ID	RD	Energia	RMSD	C	A	B	ARC	(pełniona funkcja)
1Z9P	0,448	-28,802	7,063	0,745	0,765	0,725	0,31	superoxide dismutase [cu-zn]
1D1G	0,541	-40,803	4,466	0,702	0,666	0,738	0,32	dihydrofolate reductase
1I4S	0,456	33,446	8,048	0,742	0,759	0,726	0,33	ribonuclease iii
3RD3	0,641	-123,245	7,821	0,711	0,744	0,677	0,35	probable transcriptional regulator
1VH5	0,503	152,379	4,826	0,633	0,635	0,631	0,39	hypothetical protein ydii
1FLM	0,532	304,912	5,351	0,635	0,607	0,662	0,39	protein (fmn-binding protein)
1OH0	0,592	-145,116	9,715	0,689	0,581	0,794	0,39	steroid delta-isomerase
1MK4	0,614	153,623	7,339	0,612	0,681	0,544	0,43	hypothetical protein yqjy
2D3K	0,373	9,734	8,852	0,624	0,503	0,746	0,44	peptidyl-trna hydrolase

Tabela 3.10: Wartości RD i energii oraz miar oceny wyników optymalizacji globalnej kryterium pola zewnętrznego, dla których RMSD < 10 Å i AUC $\leq 0{,}75$. Wiersze są posortowane zgodnie z rosnącymi wartościami miary ARC. Nagłówki kolumn są identyczne z tabelą 3.9.

PDB	Kryteria		Ocena					Rodzaj cząsteczki
ID	RD	Energia	RMSD	C	A	B	ARC	(pełniona funkcja)
1G17	0,558	-50,092	15,021	0,927	0,925	0,928	0,38	ras-related protein sec4
2ZOW	0,463	287,043	12,845	0,792	0,930	0,654	0,38	superoxide dismutase [cu-zn]
1A78	0,609	141,688	16,882	0,778	0,892	0,663	0,48	galectin-1

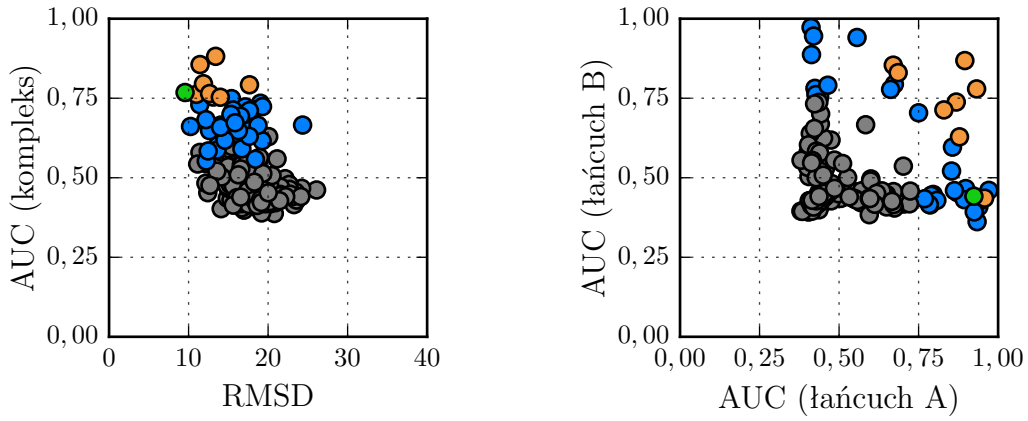
Tabela 3.11: Wartości RD i energii oraz miar oceny wyników optymalizacji globalnej kryterium pola zewnętrznego, dla których RMSD ≥ 10 Å i AUC $> 0{,}75$. Wiersze są posortowane zgodnie z rosnącymi wartościami miary ARC. Nagłówki kolumn są identyczne z tabelą 3.9.

PDB	Kryteria		Ocena					Rodzaj cząsteczki
ID	RD	Energia	RMSD	C	A	B	ARC	(pełniona funkcja)
1F08	0,753	-229,123	6,625	0,656	0,743	0,569	0,38	replication protein e1
3HV2	0,687	-235,580	4,140	0,632	0,625	0,636	0,38	response regulator/hd domain protein
3GLV	0,697	-233,923	8,074	0,609	0,624	0,593	0,44	lipopolysaccharide core biosynthesis protein
1NWW	0,754	-257,599	9,090	0,563	0,548	0,578	0,49	limonene-1,2-epoxide hydrolase
3F81	0,686	-282,067	8,516	0,553	0,611	0,526	0,50	dual specificity protein phosphatase 3
1BKZ	0,682	-240,516	8,943	0,539	0,519	0,558	0,51	galectin-7
3VRC	0,704	-373,691	9,498	0,495	0,507	0,483	0,56	cytochrome c'

Tabela 3.12: Wartości RD i energii oraz miar oceny wyników optymalizacji globalnej kryterium pola wewnętrznego zgodnych ze strukturami natywnymi. Wiersze są posortowane zgodnie z rosnącymi wartościami miary ARC. Wyjaśnienie nagłówków kolumn: *PDB ID* – identyfikator struktury w bazie PDB; *Kryteria* – wartości RD i energii; *Ocena* – wartości RMSD i AUC oceny zgodności kompleksu (C) i łańcuchów (A, B) oraz obliczona na ich podstawie wartość ARC; *Rodzaj cząsteczki (pełniona funkcja)* – rodzaj cząsteczki białka wskazany w rekordzie COMPND z pliku PDB.

PDB	Kryteria		Ocena					Rodzaj cząsteczki
ID	RD	Energia	RMSD	C	A	B	ARC	(pełniona funkcja)
1IPI	0,413	-109,528	11,445	0,856	0,932	0,779	0,32	holliday junction resolvase
3HUP	0,408	-339,355	9,558	0,768	0,924	0,442	0,33	early activation antigen cd69
3IQ3	0,440	-219,445	13,404	0,882	0,895	0,868	0,36	phospholipase a2 homolog bothropstoxin-1
1MO8	0,501	-219,155	11,873	0,796	0,868	0,738	0,36	colicin e7
1K4Z	0,483	-366,177	10,972	0,762	0,670	0,854	0,36	adenylyl cyclase-associated protein
1NCO	0,679	-194,826	12,642	0,765	0,958	0,436	0,39	holo-neocarzinostatin
1KPT	0,639	-117,213	13,109	0,754	0,879	0,629	0,41	kp4 toxin
2QVO	0,556	-272,723	13,993	0,753	0,686	0,830	0,43	protein mrke
2IGI	0,564	-288,175	17,661	0,792	0,830	0,713	0,49	oligoribonuclease

Tabela 3.13: Wartości RD i energii oraz miar oceny wyników optymalizacji globalnej kryterium pola wewnętrznego zgodnych z wynikami optymalizacji globalnej kryterium pola zewnętrznego. Wartości RD pochodzą z wyników optymalizacji kryterium pola zewnętrznego, a wartości energii – pola zewnętrznego. Wiersze są posortowane zgodnie z rosnącymi wartościami miary ARC. Wyjaśnienie nagłówków kolumn: *PDB ID* – identyfikator struktury w bazie PDB; *Kryteria* – wartości RD i energii; *Ocena* – wartości RMSD i AUC oceny zgodności kompleksu (C) i łańcuchów (A, B) oraz obliczona na ich podstawie wartość ARC; *Rodzaj cząsteczki (pełniona funkcja)* – rodzaj cząsteczki białka wskazany w rekordzie COMPND z pliku PDB.



(a) Kompleks. (b) Łańcuchy.

Rysunek 3.20: Wartości miar oceny wyników optymalizacji globalnej kryterium pola wewnętrznego zgodnych z wynikami optymalizacji globalnej kryterium pola zewnętrznego. Każdy znacznik odpowiada jednemu białku. Ich kolory wskazują natomiast na kategorie zgodności ze strukturami natywnymi: zielony – RMSD < 10 Å i AUC $> 0{,}75$, pomarańczowy – RMSD ≥ 10 Å i AUC $> 0{,}75$, niebieski – AUC $> 0{,}75$ tylko w jednym łańcuchu, szary – brak zgodności.

3.7. Kompleksowanie białek – wielokryterialne

W drugiej części eksperymentu została wykonana analiza równoczesnego wpływu pól zewnętrznego i wewnętrznego na proces kompleksowania białek, polegającego na ich optymalizacji wielokryterialnej. Oznacza to, że zamiast pojedynczego modelu, wynikiem dla każdego kompleksu był zbiór Pareto zawierający konformacje jego łańcuchów niezdominowane w przestrzeni wartości RD i energii. Tak jak poprzednio, po zakończeniu symulacji, sprawdzono zgodność tych wyników ze strukturami natywnymi przy pomocy miar RMSD i AUC.

Do przeprowadzenia optymalizacji wielokryterialnej kryteriów f_1 i f_2 wybrano algorytm MOSF. Decyzję o jego użyciu podjęto na podstawie obserwacji jego osiągnięć przedstawionych w części 3.2 tego rozdziału oraz zdolności do wykonywania „analizy skupień” odnalezionego przybliżenia optymalnego zbioru Pareto, czyli w tym przypadku podziału konformacji kompleksów na grupy o podobnych właściwościach. Innym powodem użycia tego algorytmu była chęć sprawdzenia tego, jak poradzi sobie w rzeczywistych zastosowaniach, wychodzących poza proste funkcje testowe. Ustawienia symulacji są przedstawione w tabeli 3.14.

liczba cząstek w każdym roju	100 (1), 16 (2)
maksymalna liczba iteracji	600 (1), 100 (2)
maksymalny rozmiar archiwów (a)	100
współczynnik przyciągania do liderów wewnętrznych (ϕ_l)	1,0
współczynnik przyciągania do liderów zewnętrznych (ϕ_e)	1,0
liczba najbliższych sąsiadów (n)	10
liczba iteracji algorytmu k-średnich (s)	10
okres łączenia zbiorów rojów (t_m)	20
okres usuwania zbiorów rojów (t_d)	20
rozdzielczość łączenia zbiorów rojów (r)	10% maksymalnej prędkości cząstek
pozostałe ustawienia	tak jak w tabeli 3.8

Tabela 3.14: Ustawienia algorytmu MOSF w eksperymencie kompleksowania. Liczby w nawiasach oznaczają numer pokolenia rodzin rojów: pierwsze (1) lub drugie (2).

Algorytm MOSF, podobnie jak algorytm PSO, przeszukiwał w pierwszej fazie swojego działania przestrzeń rozwiązań przez maksymalnie 600 iteracji. Oznacza to, że na tym etapie symulacji, mógł sprawdzić dwukrotnie więcej konformacji danego kompleksu niż rój cząstek dokonujący optymalizacji globalnej, ale musiał jednocześnie brać pod uwagę dwa kryteria zamiast jednego. Następnie, każda rodzina rojów cząstek była dzielona wraz ze swoim archiwum na rodziny pochodne, które mogły pracować przez maksymalnie 100 iteracji. Ich zadanie polegało na doprecyzowaniu wyników w otrzymanych przez nie podzbiorach rozwiązań niezdominowanych oraz umożliwić algorytmowi MOSF wykonanie „analizy skupień”.

Podczas eksperymentu wskazane zostały łącznie 4452 niezdominowane konformacje kompleksów. Przeciętnie, w wynikowym zbiorze Pareto dla każdego białka znalazły się 22 z nich ($\sigma \approx 7$), najmniej – 10, a najwięcej – 73. W 23 przypadkach utworzyły one jedną grupę konformacji, dwie – w 44, trzy – w 55, cztery – w 44, pięć – w 24, sześć – w 7 i siedem – w 3, łącznie 635. Średnia liczba modeli w każdej z tych grup wyniosła 7 ($\sigma \approx 6$). 76 grup, z czego 66 w różnych białkach, zawierało tylko jeden element. Była również jedna składająca się właśnie z 73.

Liczba rozwiązań niższa od maksymalnego rozmiaru archiwum sugeruje możliwą łatwość w dominowaniu jednych konformacji przez drugie, wynikającą z szybko zmieniających się wartości energii, lub krótkiego okresu uruchamiania procedur łączenia i usuwania rodzin rojów. Nie można również wykluczyć udziału w tym funkcji ograniczeń oraz sposobu w jaki obsługuje je algorytm MOSF.

Aby sprawdzić w jaki sposób konformacje kompleksów zostały połączone przez algorytm MOSF w grupy, obliczono dla każdej z nich różnice pomiędzy jej elementami przy pomocy miary RMSD. Również tutaj były brane pod uwagę obydwa łańcuchy. Pomijając 76 jednoelementowych skupisk, uzyskano następujące wyniki:

- średnia średnich: 2,018 Å ($\sigma = 1{,}430$ Å),
- średnia odchyleń standardowych: 0,786 Å ($\sigma = 0{,}841$ Å),
- średnia najniższych wartości: 0,836 Å ($\sigma = 1{,}049$ Å),
- średnia najwyższych wartości: 3,492 Å ($\sigma = 2{,}623$ Å).

W 101 grupach należących do 83 białkach o unikatowych identyfikatorach największa różnica w sensie miary RMSD pomiędzy dwoma konformacjami przekroczyła 5 Å, a w 18 – 10 Å. Oznacza to, że w większości z nich ligandy zostały ułożone w podobny sposób, z dokładnością do lokalnych obrotów lub translacji.

Następnym etapem analizy było sprawdzenie różnic pomiędzy grupami konformacji. Zastosowano w tym celu podejście najbliższych sąsiadów z algorytmu hierarchicznego analizy skupień. Zgodnie z nim, każdej grupie konformacji przypisana została najmniejsza wartość RMSD obliczona dla jednego z jej elementów i jednego elementu z innej grupy. Pomijając 23 białka, w których wszystkie konformacje znalazły się we wspólnym skupisku, średnia tej wartości wyniosła 10,599 Å ($\sigma = 5{,}029$ Å). 517 razy była ona wyższa od 5 Å, a 363 – od 10 Å.

W 72 grupach z 44 białek najmniejsza różnica w sensie wartości RMSD z inną grupą okazała się niższa od największej różnicy pomiędzy ich elementami. Analiza wizualna wykazała, że znalazły się w tej sytuacji głównie te konformacje, w których ligand został obrócony, ale tylko nieznacznie zmienił swoje położenie względem receptora. Z drugiej strony, 13 razy wartość RMSD dla konformacji należących do różnych grup była niższa od najniższej wartości tej miary wewnątrz jednej z nich. Jedenastokrotnie nie przekroczyła ona 2 Å. Powodem niepowodzenia w łączeniu tych skupisk była niewielka liczba ich elementów – w 10 poniżej czterech.

W dalszej kolejności sprawdzono odległości pomiędzy przeciętnymi środkami geometrycznymi łańcuchów B w grupach konformacji. 335 razy były one mniejsze od 5 Å, a 151 – większe od 10 Å. Sugeruje to, że podczas symulacji ligand był w pierwszej kolejności umieszczany w wybranym miejscu w pobliżu receptora, a następnie obracany w poszukiwaniu konformacji niezdominowanych. Na to zachowanie wskazują również różnice w średnich prędkościach cząstek w wymiarach położenia i obrotu zmierzone w trakcie eksperymentu optymalizacji globalnej.

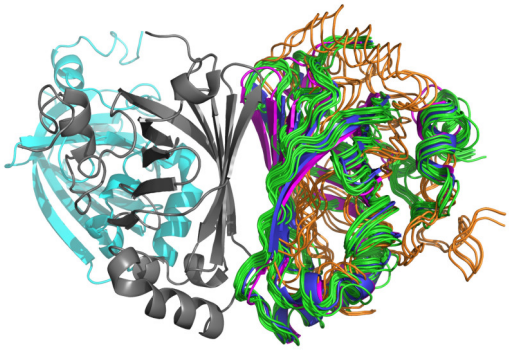
Powyższa analiza sugeruje, że w przypadku większości białek z bazy danych rozprawy, algorytm MOSF doprowadził do wskazania kilku różnych sposobów utworzenia kompleksu przez ich łańcuchy, rozumianych jako grupy, do których zostały przez niego przydzielone odnalezione konformacje niezdominowane. Umożliwiło to ułatwienie dalszych badań dzięki możliwości skupienia się od razu na nich, zamiast na kilkudziesięciu indywidualnych rozwiązaniach, oszczędzając w ten sposób czas potrzebny na ich ręczne podzielenie i unikając związanych z tym błędów. Analiza skupień *a posteriori* mogłaby prawdopodobnie zwrócić podobne wyniki, aczkolwiek byłaby mniej pewna ze względu na brak informacji na temat kształtu krajobrazu wartości optymalizowanych kryteriów oraz oczekiwanej liczby skupisk.

Zgodnie z wcześniejszymi obserwacjami, konformacje skupione w grupach wskazanych przez algorytm MOSF były do siebie podobne w sensie wartości RMSD. Oznacza to, że z każdej z nich mógł być wybrany i przekazany do dalszej analizy pojedynczy element reprezentacyjny. Przyjęto, że będzie nim ta konformacja, dla której wartość miary ARC okazała się najniższa. Ponieważ średnia odchyleń standardowych tej miary we wszystkich grupach wyniosła tylko 0,015 ($\sigma = 0{,}013$), uznano, że taki sposób reprezentacji nie wpłynie istotnie na postrzeganie ich zgodności ze strukturami natywnymi. Dane odpowiadające wybranym w ten sposób konformacjom reprezentacyjnym są umieszczone w tabeli B.3. Identycznie byli wybierani przedstawiciele całych zbiorów Pareto, jako minima globalne miary ARC.

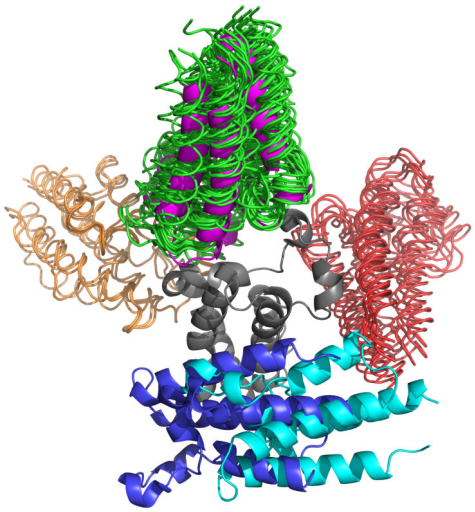
Dwa przykłady łączenia wyników w skupiska przez algorytm MOSF są przedstawione na rysunku 3.21 w dwojaki sposób: w postaci wizualizacji konformacji tworzących zbiory Pareto oraz w przestrzeni wartości, ukazując kształt i zawartość odpowiadających im frontów Pareto. Pierwszy z tych przykładów prezentuje białko, które osiągnęło najwyższą zgodność ze strukturą natywną podczas optymalizacji globalnej kryterium pola zewnętrznego – `2W2A`. Wynik ten został poprawiony w trakcie optymalizacji wielokryterialnej w sensie miary RMSD ponad czterokrotnie: z 0,941 Å do 0,217 Å. Na dotyczących go rysunkach widać dwie grupy konformacji: wysoce zgodną oraz taką, w której ligand został obrócony o 180 stopni.

Wynikiem optymalizacji wielokryterialnej układu łańcuchów drugiego białka, `2YEM` [416], były natomiast trzy grupy konformacji, z których każda okazała się niezgodna ze strukturą. Najbliżej niej znalazł się kompleks uzyskany podczas optymalizacji globalnej pola wewnętrznego.

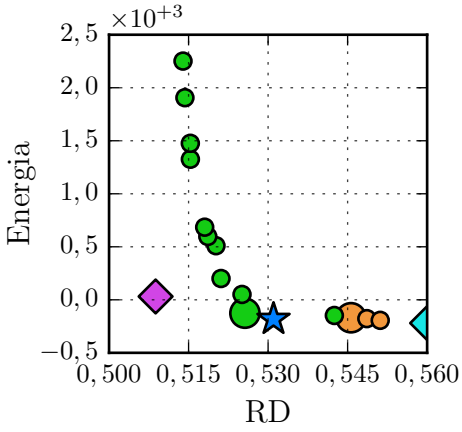
Białka `2W2A` i `2YEM` są przykładami zgodności pomiędzy wynikami z obydwu części eksperymentu: konformacje otrzymane podczas optymalizacji globalnej znalazły się w grupach konformacji niezdominowanych odnalezionych przez algorytm MOSF.



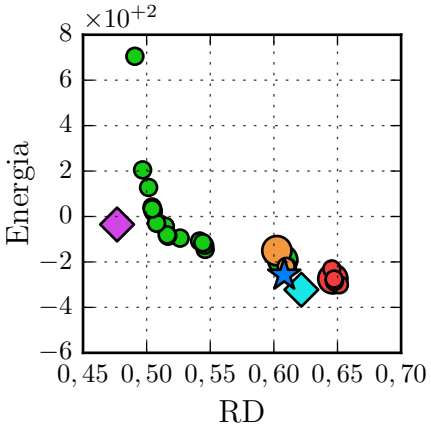
(a) 2W2A – przestrzeń konformacyjna.



(b) 2YEM – przestrzeń konformacyjna.



(c) 2W2A – przestrzeń wartości.



(d) 2YEM – przestrzeń wartości.

Rysunek 3.21: Przykładowe wizualizacje wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego oraz odpowiadające im fronty Pareto. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski. Na zielono, czerwono i pomarańczowo są natomiast pokolorowane łańcuchy B należące do grup konformacji zwróconych przez algorytm MOSF. Dodatkowo, dla porównania, zostały tu umieszczone wyniki optymalizacji globalnej pola zewnętrznego (magenta – wartości energii bez E_n) i wewnętrznego (cyjan). Dobór kolorów na rysunkach b i d nie jest zgodny ze schematem stosowanym w dodatku A.2. Każdy znacznik na rysunkach c i d odpowiada jednej konformacji. Ich kolory są takie same jak struktur powyżej. Znaczniki w kształcie kół o większych rozmiarach oznaczają przedstawicieli grup konformacji (o najniższej wartości AUC). W celu zachowania czytelności rysunku c współrzędna wartości RD wyniku optymalizacji globalnej pola wewnętrznego została umieszczona na jego prawej krawędzi – w rzeczywistości wyniosła ona 0,663.

3.7.1. Kontakty niewiążące

Średnia wartość ułamka ITF wyników symulacji i struktur natywnych wyniosła 0,686, przy odchyleniu standardowym 0,753. Wskazuje to na działalność pola wewnętrznego, odsuwającego łańcuchy od siebie. Wynik ten jest o połowę wyższy niż w przypadku optymalizacji globalnej tego kryterium (0,447), ale równocześnie dwukrotnie niższy od średniej dla pola zewnętrznego (1,319). Znajduje się on jednak najbliżej wartości oczekiwanej, czyli 1. Ponieważ jest od niej niższy, oznacza to, że podczas optymalizacji wielokryterialnej, łańcuchy mogły przyjąć konformacje zgodne ze strukturami natywnymi, ale nie należy się spodziewać dużej liczby kontaktów niewiążących między nimi, zarówno tych prawdziwie, jak i fałszywie dodatnich. Ponownie nie zauważono korelacji pomiędzy wartościami ITF a liczbą reszt. Zwrócono natomiast uwagę na fakt, że najwyższe z nich uzyskały białka o najkrótszej sekwencji.

3.7.2. Symetria kompleksu

W 72 grupach konformacji z 51 białek stwierdzono wartość ICF wyższą od 0,5, a w 475 – poniżej 0,25. Są to wartości zbliżone do tych, które zaobserwowano w przypadku wyników optymalizacji pola zewnętrznego (74). Ze względu na niedużą liczbę kontaktów pomiędzy łańcuchami, nie należy zakładać, że kompleksy niesymetryczne w sensie miary ICF faktycznie nie posiadały tej cechy. W połączeniu z wynikami analizy rozmiaru interfejsu łańcuchów, jest to jednak wystarczające do wyciągnięcia wniosku na temat tego jak siły modelowane przez obydwa pola kierują procesem powstawania kompleksów. Mianowicie, za ułożenie cząsteczek względem siebie odpowiada pole zewnętrzne, które swoim zasięgiem obejmuje całość układu, natomiast dopasowaniem ich powierzchni zajmuje się pole wewnętrzne, działające w przestrzeni rzeczywistych atomów, zamiast wirtualnych atomów efektywnych.

3.7.3. Wartości RD i energii

Po zakończeniu symulacji obliczone zostały wartości RD i energii dla wszystkich wynikowych konformacji kompleksów zwróconych przez algorytm MOSF. Ich wykresy dla przedstawicieli całych zbiorów Pareto znajdują się na rysunku 3.22. Cztery ostatnie kolumny tabeli B.3 zawierają natomiast dane końców fragmentów frontów Pareto odpowiadających kolejnym grupom konformacji, co pozwala na oszacowanie położenia każdej z nich w przestrzeni wartości.

Współczynnik korelacji wartości RD wyników optymalizacji globalnej i białek natywnych wyniósł 0,657, a średnia różnica między nimi – 0,102 ($\sigma = 0{,}057$). Tutaj, dla przedstawicieli grup konformacji okazał się on równy 0,534, a dla całych zbiorów Pareto – 0,666. Przeciętna różnica wartości RD zmniejszyła się natomiast z 0,102 ($\sigma = 0{,}057$) do 0,068 ($\sigma = 0{,}048$), podobnie jak liczba przypadków, w których wynik eksperymentu osiągnął tę wartość niższą niż jego struktura natywna: ze 195 do 165. Ponownie, najwyższą zgodność pomiędzy rozkładami hydrofobowości zaobserwowano w przypadku białka `2D3K`: 0,406. W obecności tylko kilku nieznacznie odstających punktów, oznacza to, że wśród konformacji odnalezionych przez algorytm MOSF znalazły się takie, które w sensie wartości kryterium pola zewnętrznego były zbliżone do tych posiadanych przez kompleksy natywne. Nie oznacza to oczywiście, że taka sama zależność musi również dotyczyć zgodności ich struktur.

Współczynnik korelacji wartości RD wyników optymalizacji globalnej i wielokryterialnej wyniósł 0,757 dla przedstawicieli grup konformacji i 0,817 w przypadku całych zbiorów Pareto. Sugeruje to, że algorytm MOSF, biorący pod uwagę obydwa pola, zwrócił struktury kompleksów podobne pod względem zgodności z modelem FOD również do wskazanych przez rój cząstek zajmujący się tylko jednym z tych pól. Obecność pola wewnętrznego zdaje się więc nie mieć wpływu na przeszukiwanie przestrzeni rozwiązań. Staje się on jednak widoczny po sprawdzeniu zgodności uzyskanych wyników z modelem FOD. Przedstawicieli grup konformacji, w przypadku których wartość RD okazała się mniejsza od 0,5 było 61. Należały one do 26 białek o unikatowych identyfikatorach. Stanowi to około połowę liczby struktur zgodnych z modelem FOD uzyskanych podczas optymalizacji pola zewnętrznego. Oprócz tego, tylko 11 razy zostały osiągnięte wartości RD niższe niż w poprzedniej części eksperymentu. Ponieważ zdecydowana większość kompleksów z bazy danych rozprawy była niezgodna z modelem FOD, pole wewnętrzne może mieć więc korzystny wpływ na ich poszukiwanie. Wskazują na to następujące relacje statusu tej zgodności:

1. 6 razy kompleksy natywne oraz wynikowe były zgodne (poprzednio 7),

2. 171 razy kompleksy natywne oraz wynikowe były niezgodne (poprzednio 147),

3. 20 razy kompleksy natywne były niezgodne, ale wynikowe okazały się zgodne (poprzednio 44),

4. 3 razy kompleksy natywne były zgodne, ale wynikowe okazały się niezgodne (poprzednio 2).

Tylko jedno białko zostało przeniesione podczas optymalizacji wielokryterialnej z pierwszej do czwartej grupy: `1I4S`. Uzyskało ono jednak wartość RD równą 0,515, a więc bliską statusu zgodności z modelem FOD. W przypadku pozostałych dwóch grup, 24 białka przeszły z trzeciej do drugiej i ani jedno w drugą stronę. Jest to wyraźny sygnał, że optymalizacja wielokryterialna może być bardziej skuteczna od globalnej w poszukiwaniu struktur kompleksów niezgodnych z modelem FOD. Jak zostało wykazane w poprzedniej części eksperymentu, pole wewnętrzne przeciwdziała polu zewnętrznemu, kierując algorytm optymalizacyjny w stronę innych konformacji. Dzięki temu możliwe było dotarcie do nisko położonych minimów jednego i drugiego kryterium, co widać na przykładzie białka `2W2A`, przy jednoczesnym zachowaniu w archiwum części rozwiązań o RD większym od 0,5.

W 93 kompleksach, energia ich przedstawicieli okazała się niższa niż strukturach natywnych, z czego 88 razy osiągnęła wartość niższą od 0, średnio o 171,773 $\frac{\text{kcal}}{\text{mol}}$ ($\sigma = 92{,}218\,\frac{\text{kcal}}{\text{mol}}$). Jest to niemal połowa liczby wyników, które uzyskały ten sam status podczas optymalizacji globalnej kryterium pola wewnętrznego (182 w jednym i drugim przypadku). Tylko w jednym białku, o identyfikatorze `3I4S`, energia okazała się niższą niż poprzednio. Sugeruje to przewagę pola zewnętrznego nad wewnętrznym w przyciąganiu rojów cząstek do swoich niskich wartości. Z drugiej strony, dzięki obecności tego drugiego możliwe było zmniejszenie liczby kolizji pomiędzy atomami, skutkujących poprzednio wysoką wartością energii. Jej średnia wyniosła tutaj bowiem 123,738 $\frac{\text{kcal}}{\text{mol}}$ ($\sigma = 511{,}364\,\frac{\text{kcal}}{\text{mol}}$), a więc znacznie mniej niż ponad 2700 $\frac{\text{kcal}}{\text{mol}}$ podczas optymalizacji samego pola zewnętrznego. Wyników, których reprezentanci charakteryzowali się energią wyższą od 500 $\frac{\text{kcal}}{\text{mol}}$ było 35. Próg 3000 $\frac{\text{kcal}}{\text{mol}}$ został natomiast przekroczony przez tylko dwie z 4452 konformacji. Ponownie nie zaobserwowano jakiejkolwiek korelacji pomiędzy wartościami energii.

Fronty Pareto utworzone przez wyniki optymalizacji wielokryterialnej są przedstawione na rysunku 3.23. Punkty odpowiadające strukturom natywnym znalazły się w ich dolnych częściach, czyli bliżej konformacji o niskich wartościach energii. Struktury te zostały zdominowane przez konformacje wynikowe w 154 białkach. Sytuacja odwrotna, gdy front Pareto był całkowicie przez nie zdominowany zdarzyła się 4 razy. W pozostałych przypadkach stanowiły jego część. 9 razy nie wiązało się to ze zdominowaniem jakichkolwiek wynikowych konformacji. Średnia różnicy pomiędzy strukturami natywnymi a najbliższymi elementami frontów Pareto wyniosła 0,033 ($\sigma = 0{,}038$) w wymiarze wartości RD i 30,948 $\frac{\text{kcal}}{\text{mol}}$ ($\sigma = 52{,}799\,\frac{\text{kcal}}{\text{mol}}$) w wymiarze energii. Jest to względnie niewiele, co wskazuje na możliwość ich odnajdywania przy pomocy optymalizacji wielokryterialnej pól zewnętrznego i wewnętrznego.

3.7.4. Zgodność ze strukturami natywnymi

Wartości RMSD oraz AUC miar oceny zgodności wyników eksperymentu ze strukturami natywnymi są przedstawione w sposób graficzny na rysunku 3.24. Wyniki, w przypadku których stwierdzono wartość RMSD mniejszą od 10 Å lub wartość AUC większą od 0,75 zostały dodatkowo umieszczone na rysunku 3.22. Wybrane z tabeli B.3 informacje na temat tych wyników znajdują się w tabelach 3.15, 3.16 i 3.17.

Optymalizacja wielokryterialna kryteriów pól zewnętrznego i wewnętrznego doprowadziła do utworzenia 11 kompleksów o wysokiej zgodności z ich strukturami natywnymi (RMSD < 10 Å i AUC $> 0{,}75$), 31 o wartości RMSD niższej od 10 Å oraz 5 o wartości AUC wyższej od 0,75. Wizualizacje tych wyników są przedstawione na rysunkach A.18, A.19, A.20 i A.15. W porównaniu z poprzednią częścią eksperymentu, oznacza to wzrost liczb kompleksów zgodnych ze strukturami natywnymi o odpowiednio 4 (z 7), 22 (z 9) i 2 (z 3). Dzięki symulacji równoczesnego wpływu środowiska wodnego oraz energii oddziaływań między atomami na układ pary łańcuchów możliwe było więc doprowadzenie do przyjęcia przez nie konformacji zbliżonych do natywnych w przypadku 47 białek, czyli w prawie $\frac{1}{4}$ zawartości bazy danych rozprawy. Optymalizacja kryterium samego pola zewnętrznego umożliwiła osiągnięcie podobnego stanu 19 razy, a wewnętrznego – 7.

Wszystkie 11 wyników o RMSD < 10 Å i AUC $> 0{,}75$ było natywnie niezgodne z modelem FOD. Sytuacja w przypadku ich łańcuchów była odwrotna: tylko jedna para, pochodząca z białka `1Q98` posiadała wartość RD większą od 0,5. Średnia różnicy wartości RD pomiędzy natywnymi i wynikowymi strukturami z tej kategorii wyniosła natomiast 0,027 ($\sigma = 0{,}033$). Wszystkie kompleksy były natywnie symetryczne i dziesięciokrotnie takie też okazały się po zakończeniu symulacji.

Trzy białka osiągnęły ten sam status zgodności ze strukturami natywnymi co wyniki uzyskane podczas optymalizacji globalnej kryterium pola zewnętrznego: `2OE3`, `2W2A` i `3AIA` [417]. Wartości RMSD dla białek `2OE3` i `3AIA` w obydwu częściach eksperymentu okazały się bardzo podobne, natomiast w białku `2W2A` wartość ta zmniejszyła się z 0,941 Å do 0,217 Å, osiągając niemal konformację natywną. Oznacza to, że pole zewnętrzne jest wiodącą siłą w formowaniu się kompleksu tego białka, natomiast pole wewnętrzne w tym nie przeszkadza. W pozostałych dwóch pełni ono role wspomagającą, blokując konformacje o niskich wartościach RD. Zaobserwowano również dwa białka, w przypadku których osiągnięcie struktury niemal natywnej kompleksu wynikało ze „starcia” pól zewnętrznego i wewnętrznego: `3I4S` (RMSD 0,499 Å) i `3GLV` (RMSD 0,727 Å). Innych wyników o RMSD < 5 Å było 7.

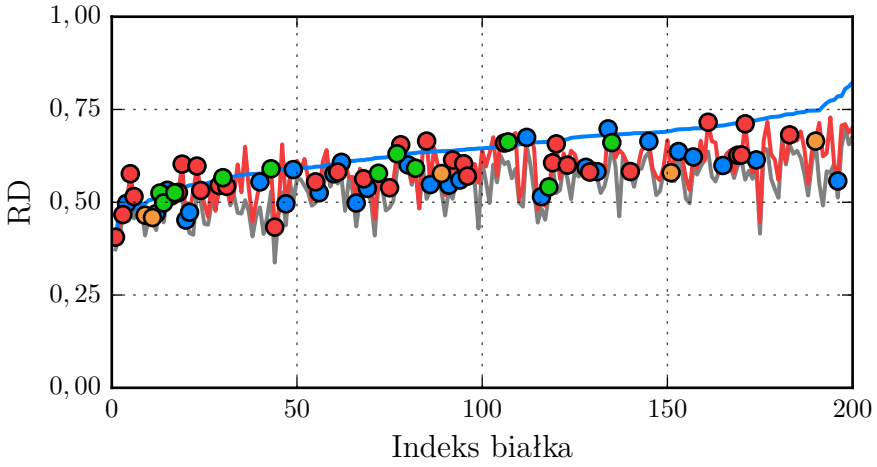
W 4 wynikach o RMSD < 10 Å i AUC $> 0{,}75$ stwierdzono obecność więcej niż jednej grupy konformacji, których przedstawiciele byli zgodni ze strukturami natywnymi. W białku `2W2A` została odnaleziona konformacja kompleksu z receptorem obróconym o 180 stopni w płaszczyźnie interfejsu pomiędzy łańcuchami, charakteryzująca się niższą energią, ale wyższą wartością RD. W białku `2OE3` były natomiast dwie takie grupy: obróconego receptora i poddanego translacji. Konformacje obróconych receptorów zaobserwowano również w białkach `1A78` [418] i `1ADW` [236]. Wśród pozostałych wyników o RMSD < 10 Å albo AUC $> 0{,}75$ wystąpiły odpowiednio 6 oraz 2 takie sytuacje. W każdej z nich, grupy konformacji zgodnych ze strukturami natywnymi znajdowały się blisko siebie (co implikuje niska wartość RMSD). Oznacza to, że należy je potraktować jako skupiska wyników optymalizacji wielokryterialnej, które nie zostały, choć powinny, być połączone przez algorytm MOSF.

Wśród 31 wyników, dla których stwierdzono RMSD < 10 Å i AUC $\leq 0{,}75$ było 6 białek, które uzyskały identyczny status podczas optymalizacji globalnej kryterium pola zewnętrznego. Jedno, białko `1TLJ` uzyskało wcześniej wartość AUC większą od 0,75. Nie było natomiast w tej grupie białek, w przypadku których optymalizacja globalna pola zewnętrznego doprowadziła do uzyskania wartości AUC $> 0{,}75$. Poza trzema białkami, `2D3K`, `2DCT` i `3PH4` [419], wyniki należące do tej grupy okazały się niezgodne z modelem FOD.

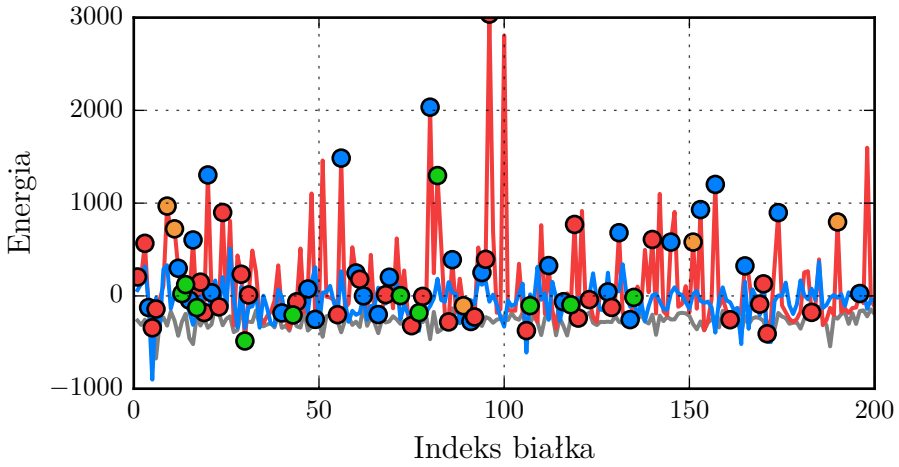
Wyników o RMSD ≥ 10 Å i AUC $> 0{,}75$ było tylko 5, przy czym wysoka wartość AUC była efektem prawidłowego ułożenia jednego z łańcuchów. Niewielka liczba wyników należących do tej kategorii sugeruje, że konformacje o tych właściwościach nie są preferowane przez pola zewnętrzne i wewnętrzne.

Wśród konformacji reprezentujących 28 wyników zaobserwowano wartość AUC tylko jednego łańcucha większą od 0,75. W drugim była ona mniejsza od 0,5 19 razy. Jest ich mniej niż w przypadku optymalizacji globalnej pola zewnętrznego, podczas której odpowiednio 44 i 27 wyników charakteryzowało się tą właściwością. Oznacza to, że dzięki optymalizacji wielokryterialnej możliwe było zbliżenie rzeczywistych interfejsów łańcuchów do siebie. Widać to szczególne w 5 białkach, które uzyskały dzięki temu najwyższy status zgodności ze strukturą natywną.

Graficzna prezentacja porównania wyników optymalizacji globalnej pola zewnętrznego i wielokryterialnej w przestrzeni wartości ARC znajduje się na rysunku 3.25. Widać na nim, że podejścia bazujące na samym polu zewnętrznym i przy udziale pola wewnętrznego wzajemnie się uzupełniają w tym sensie, że pozwalają na odnalezienie różnych struktur natywnych, o odpowiednio niskiej lub wysokiej wartości RD.

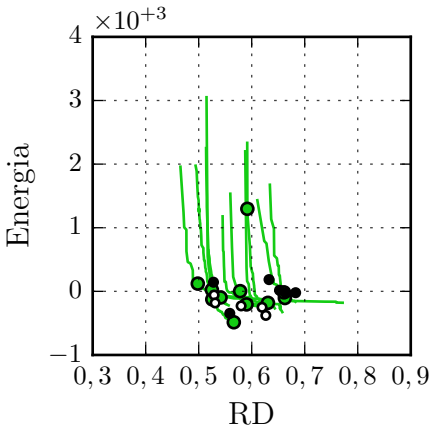


(a) Wartości RD.

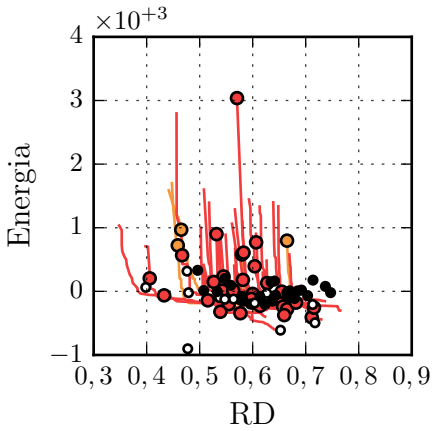


(b) Wartości energii.

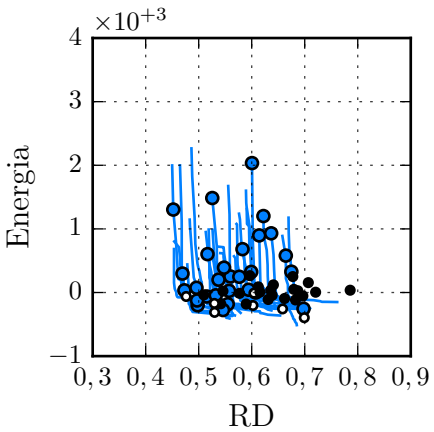
Rysunek 3.22: Wartości RD i energii wyników optymalizacji wielokryterialnej pól zewnętrznego i wewnętrznego. Każde białko jest reprezentowane przez konformację jego kompleksu o najniższej wartości miary ARC. Kolory rozkładów oznaczają rodzaj struktury: niebieski – natywna, czerwony – wynikowa, szary – wynikowa, ale uzyskana podczas optymalizacji globalnej pola zewnętrznego (rysunek a) lub wewnętrznego (rysunek b). Wszystkie rozkłady są posortowane zgodnie z rosnącymi wartościami RD dla struktur natywnych. Każdy znacznik odpowiada jednemu białku. Ich kolory wskazują natomiast na kategorie zgodności ze strukturami natywnymi: zielony – RMSD $< 10\,\text{Å}$ i AUC $> 0{,}75$, czerwony – RMSD $< 10\,\text{Å}$ i AUC $\leq 0{,}75$, pomarańczowy – RMSD $\geq 10\,\text{Å}$ i AUC $> 0{,}75$, niebieski – AUC $> 0{,}75$ tylko w jednym łańcuchu, szary – pozostałe.



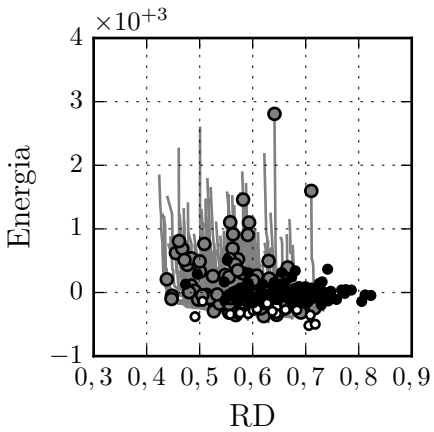
(a) RMSD < 10 Å i AUC $> 0{,}75$.



(b) RMSD < 10 Å albo AUC $> 0{,}75$.

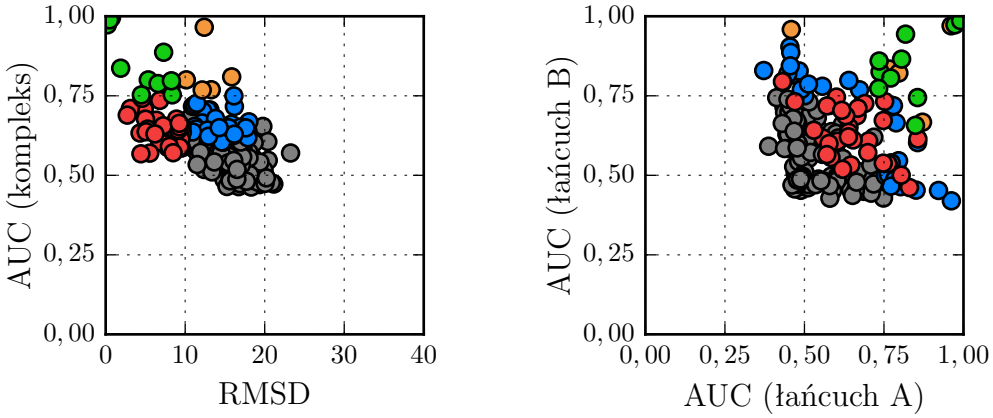


(c) AUC $> 0{,}75$ w jednym łańcuchu.



(d) Pozostałe wyniki.

Rysunek 3.23: Fronty Pareto wyników optymalizacji wielokryterialnej pól zewnętrznego i wewnętrznego, podzielone ze względu na ich przynależność do kategorii zgodności ze strukturami natywnymi. Kolorowe znaczniki wskazują na konformacje im najbliższe w sensie miary ARC. Nadają one również jednolity kolor całym frontom Pareto, niezależnie od obecności w nich grup konformacji należących do innych kategorii (sytuacja ta dotyczy wyłącznie rysunku 3.23a). Pozostałe znaczniki wskazują natomiast na położenia w przestrzeni wartości niezdominowanych (kolor biały) oraz zdominowanych (kolor czarny) przez wyniki symulacji struktur natywnych.



(a) Wartości RD. (b) Wartości Energii.

Rysunek 3.24: Wartości miar oceny wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego. Każdy znacznik odpowiada jednemu białku. Ich kolory wskazują natomiast na kategorie zgodności ze strukturami natywnymi: zielony – RMSD $< 10\,\text{Å}$ i AUC $> 0{,}75$, czerwony – RMSD $< 10\,\text{Å}$ i AUC $\leq 0{,}75$, pomarańczowy – RMSD $\geq 10\,\text{Å}$ i AUC $> 0{,}75$, niebieski – AUC $> 0{,}75$ tylko w jednym łańcuchu, szary – brak zgodności.

PDB	Kryteria		Ocena					Rodzaj cząsteczki
ID	RD	Energia	RMSD	C	A	B	ARC	(pełniona funkcja)
3I4S	0,566	-485,767	0,499	0,989	0,991	0,986	0,02	histidine triad protein
3GLV	0,590	-205,355	0,727	0,995	0,995	0,995	0,02	lipopolysaccharide core biosynthesis protein
2W2A	0,526	-126,018	0,217	0,973	0,973	0,973	0,03	p-coumaric acid decarboxylase
1M4R	0,631	-180,599	1,886	0,837	0,805	0,865	0,17	interleukin-22
2OE3	0,498	121,731	7,289	0,887	0,818	0,944	0,21	thioredoxin-3
1F1C	0,592	1297,195	5,358	0,800	0,856	0,744	0,24	cytochrome c549
1Q98	0,661	-15,762	6,589	0,788	0,771	0,806	0,27	thiol peroxidase
3AIA	0,525	24,885	4,487	0,753	0,733	0,773	0,27	upf0217 protein mj1640
3RHC	0,541	-95,919	8,262	0,797	0,735	0,859	0,29	glutaredoxin-c5, chloroplastic
1A78	0,663	-104,952	8,237	0,781	0,738	0,825	0,30	galectin-1
1ADW	0,578	3,224	8,376	0,753	0,849	0,657	0,32	pseudoazurin

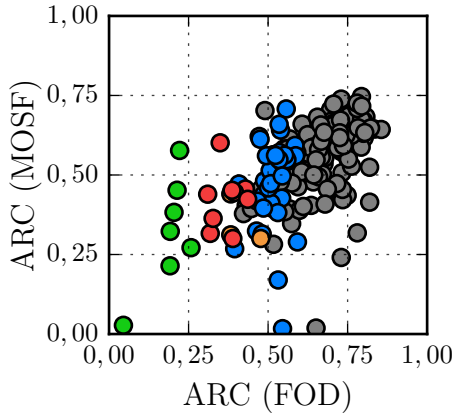
Tabela 3.15: Wartości RD i energii oraz miar oceny wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego, dla których RMSD $< 10\,\text{Å}$ i AUC $> 0{,}75$. Wiersze są posortowane zgodnie z rosnącymi wartościami miary ARC. Wyjaśnienie nagłówków kolumn: *PDB ID* – identyfikator struktury w bazie PDB; *Kryteria* – wartości RD i energii; *Ocena* – wartości RMSD i AUC oceny zgodności kompleksu (C) i łańcuchów (A, B) oraz obliczona na ich podstawie wartość ARC; *Rodzaj cząsteczki (pełniona funkcja)* – rodzaj cząsteczki białka wskazany w rekordzie COMPND z pliku PDB.

PDB	Kryteria			Ocena				Rodzaj cząsteczki
ID	RD	Energia	RMSD	C	A	B	ARC	(pełniona funkcja)
3PH4	0,467	566,994	4,574	0,742	0,752	0,732	0,28	ribose-5-phosphate isomerase
1FLM	0,545	233,647	3,109	0,710	0,748	0,672	0,30	protein (fmn-binding protein)
1OPA	0,606	770,408	6,801	0,735	0,857	0,614	0,31	cellular retinol binding protein ii
1D1G	0,543	10,455	4,851	0,708	0,690	0,726	0,32	dihydrofolate reductase
2XOL	0,711	-407,703	2,715	0,689	0,668	0,711	0,32	chaperone protein ttrd
1J3M	0,603	-179,039	5,011	0,676	0,665	0,687	0,35	the conserved hypothetical protein tt1751
1I4S	0,515	-141,446	6,377	0,673	0,601	0,746	0,36	ribonuclease iii
2DCT	0,433	-62,953	5,280	0,645	0,640	0,651	0,38	hypothetical protein ttha0104
1TLJ	0,532	899,473	4,321	0,633	0,653	0,611	0,38	hypothetical upf0130 protein sso0622
3TW2	0,577	-343,804	5,091	0,637	0,639	0,635	0,38	histidine triad nucleotide-binding protein 1
3EVI	0,657	-238,970	7,229	0,656	0,630	0,687	0,39	phosducin-like protein 2
1DQE	0,582	-124,235	8,915	0,677	0,830	0,462	0,39	pheromone-binding protein
4DFO	0,659	-373,741	6,198	0,630	0,638	0,621	0,40	orotidine 5'-phosphate decarboxylase
1BKZ	0,627	-88,866	8,353	0,654	0,702	0,608	0,40	galectin-7
3IIR	0,682	-177,444	9,254	0,663	0,617	0,704	0,41	trypsin inhibitor
1M4J	0,614	-226,941	8,504	0,648	0,571	0,718	0,41	a6 gene product
2OFC	0,570	3037,775	9,257	0,652	0,804	0,501	0,42	sclerotium rolfsii lectin
2BPD	0,563	13,747	9,602	0,653	0,762	0,537	0,42	dectin-1
2D3K	0,406	204,333	9,125	0,643	0,748	0,538	0,42	peptidyl-trna hydrolase
1V5X	0,655	-4,837	6,796	0,605	0,603	0,607	0,43	phosphoribosylanthranilate isomerase
1CO2	0,555	-202,014	9,646	0,635	0,699	0,571	0,44	phosphotransferase ypd1p
3FOU	0,598	-118,193	4,432	0,568	0,570	0,565	0,45	quinol-cytochrome c reductase, rieske iron-sulfur subunit
1VH5	0,527	148,412	5,549	0,570	0,584	0,556	0,45	hypothetical protein ydii
1BD9	0,583	608,628	8,832	0,604	0,604	0,605	0,45	phosphatidylethanolamine binding protein
1TFP	0,604	394,434	9,456	0,613	0,431	0,795	0,45	transthyretin
3FU1	0,665	-280,287	8,008	0,591	0,646	0,533	0,46	general secretion pathway protein g
1MK4	0,627	130,079	7,808	0,588	0,558	0,617	0,46	hypothetical protein yqjy
1NP8	0,539	-320,863	9,050	0,601	0,470	0,731	0,46	calcium-dependent protease, small subunit
2Z76	0,716	-256,472	9,333	0,589	0,577	0,600	0,47	putative steroid isomerase
1QAH	0,600	-39,790	9,544	0,586	0,530	0,642	0,48	perchloric acid soluble protein
3UJM	0,582	179,633	8,481	0,571	0,620	0,519	0,48	rasputin

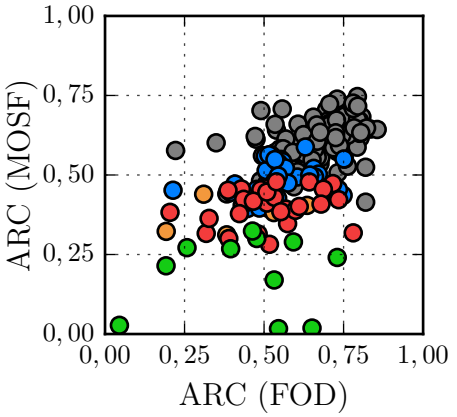
Tabela 3.16: Wartości RD i energii oraz miar oceny wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego, dla których RMSD < 10 Å i AUC $\leq 0{,}75$. Wiersze są posortowane zgodnie z rosnącymi wartościami miary ARC. Nagłówki kolumn są identyczne z tabelą 3.15.

PDB	Kryteria			Ocena				Rodzaj cząsteczki
ID	RD	Energia	RMSD	C	A	B	ARC	(pełniona funkcja)
1G17	0,580	580,208	12,397	0,966	0,961	0,970	0,31	ras-related protein sec4
1VC1	0,465	968,432	10,139	0,800	0,766	0,834	0,32	putative anti-sigma factor antagonist tm1442
2Z5D	0,577	-105,656	12,140	0,768	0,459	0,958	0,38	ubiquitin-conjugating enzyme e2 h
1IFV	0,665	797,108	13,281	0,769	0,871	0,667	0,40	protein llr18b
1Z9P	0,459	722,963	15,846	0,809	0,797	0,822	0,44	superoxide dismutase [cu-zn]

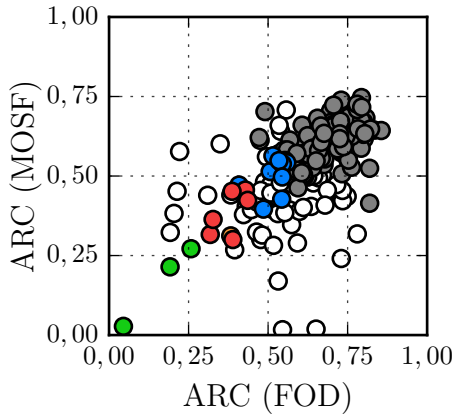
Tabela 3.17: Wartości RD i energii oraz miar oceny wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego, dla których RMSD ≥ 10 Å i AUC $> 0{,}75$. Wiersze są posortowane zgodnie z rosnącymi wartościami miary ARC. Nagłówki kolumn są identyczne z tabelą 3.15.



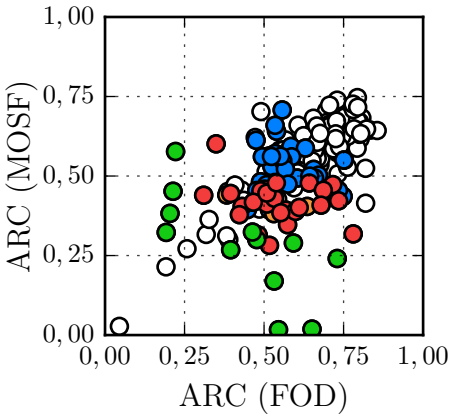
(a) Kolor statusu zgodności według wyników optymalizacji globalnej.



(b) Kolor statusu zgodności według wyników optymalizacji wielokryterialnej.



(c) Identyczny status zgodności.



(d) Różne statusy zgodności.

Rysunek 3.25: Porównanie wyników optymalizacji globalnej kryterium pola zewnętrznego i optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego. Każdy znacznik odpowiada jednemu białku. Ich kolory wskazują natomiast na kategorie zgodności ze strukturami natywnymi: zielony – RMSD < 10 Å i AUC $> 0{,}75$, czerwony – RMSD < 10 Å i AUC $\leq 0{,}75$, pomarańczowy – RMSD ≥ 10 Å i AUC $> 0{,}75$, niebieski – AUC $> 0{,}75$ tylko w jednym łańcuchu, szary – brak zgodności. Znaczniki w kolorze białym oznaczają na rysunku c białka o różnym statusie zgodności, a na rysunku d – o identycznym, czyli przeciwnie do podpisów pod tymi rysunkami.

4. Dyskusja i wnioski

Niniejsza rozprawa doktorska zakładała do realizacji trzy główne cele i jeden cel poboczny. Wszystkie cztery cele zostały zrealizowane.

Pierwszym celem głównym było sprawdzenie założeń pola zewnętrznego (modelu FOD) dotyczących wpływu opisywanych przez niego oddziaływań hydrofobowych na proces tworzenia się kompleksów typu białko-białko poprzez wykonanie eksperymentu *ab initio* przewidywania struktury czwartorzędowej 200 białek homodimerycznych wybranych z bazy PDB i porównanie uzyskanych wyników z wynikami uzyskanymi przy pomocy pola wewnętrznego (pola ECEPP/3).

Drugim celem było zastosowanie algorytmów opartych na sposobie działania roju cząstek do wykonania powyższego eksperymentu i sprawdzenie, czy możliwe jest ich skutecznie stosowanie w symulacjach procesów związanych z białkami, w tym przypadku – kompleksowaniem. Użyte zostały dwa algorytmy: klasyczny algorytm PSO oraz autorski algorytm MOSF. Za pomocą algorytmu PSO została wykonana optymalizacja globalna konformacji kompleksu według osobnych kryteriów pola zewnętrznego i wewnętrznego, natomiast za pomocą algorytmu MOSF wykonano optymalizację wielokryterialną, symulującą równoczesny wpływ tych sił.

Trzecim celem było opracowanie autorskiego algorytmu MOSF (wielokryterialne rodziny rojów, multi objective swarm families), będącego modyfikacją algorytmu PSO do zadań optymalizacji wielokryterialnej. Motywacją do opracowania nowego algorytmu była potrzeba uzyskania funkcjonalności przydatnej podczas analizy wyników przeprowadzonego przy jego pomocy eksperymentu kompleksowania. Funkcjonalności tej („analizy skupień” przybliżenia optymalnego zbioru Pareto oraz jego jednorodnej reprezentacji) nie oferują dotychczas stosowane metody. Aby wykazać, że algorytm MOSF ma zastosowanie również poza bioinformatyką, porównano zwrócone przez niego wyniki z wynikami algorytmów NSGA-II i NSPSO.

Czwartym, pobocznym celem rozprawy była aktualizacja sposobu przygotowania cząsteczki białka do obliczeń rozkładu hydrofobowości teoretycznej modelu FOD umożliwiająca jego efektywnego stosowanie jako kryterium optymalizacyjne.

4.1. Algorytm MOSF

Motywacją Autora rozprawy do opracowania nowego algorytmu optymalizacji wielokryterialnej bazującego na zasadzie działania roju cząstek było uzyskania możliwości przeprowadzania automatycznej „analizy skupień" rozwiązań stanowiących wynikowe przybliżenie optymalnego zbioru Pareto, a także jego jednorodnej reprezentacji. Okazało się to bardzo przydatne podczas eksperymentu kompleksowania, gdyż umożliwiło połączenie uzyskanych konformacji w grupy, reprezentujących różne sposoby utworzenia kompleksów, co skróciło czas ich przetwarzania i ułatwiło ich interpretację dzięki skupieniu się od razu na tych grupach, zamiast na indywidualnych rozwiązaniach, nierozróżnialnych w sensie optimum Pareto.

„Analiza skupień" wykonywana przez algorytm MOSF dostarcza informacji na temat elementów przybliżenia optymalnego zbioru Pareto i rozwiązywanego problemu, które mogą być trudne do uzyskania po zakończeniu optymalizacji. Mianowicie, oprócz łączenia w skupiska blisko położonych rozwiązań, łączy również te rozwiązania, pomiędzy którymi poprowadziły go optymalizowane kryteria. Pozwala to na wnioskowanie na temat kształtu krajobrazu wartości tych kryteriów, relacji pomiędzy nimi, a także wskazywanie mniejszych obszarów przestrzeni rozwiązań, w których może być rozpoczęta nowa, lokalna, a przez to bardziej precyzyjna procedura optymalizacji. Algorytm MOSF umożliwia zwiększanie tej rozdzielczości do dowolnego poziomu wskazanego przez użytkownika przy pomocy mechanizmu podziału rodzin rojów. Zgodnie ze stanem wiedzy Autora rozprawy dotychczas stosowane algorytmy optymalizacji wielokryterialnej nie oferują tej możliwości.

Realizacja „analizy skupień" w algorytmie MOSF polega na użyciu kilku rodzin rojów cząstek (skąd pochodzi jego nazwa), powstałych zazwyczaj z podziału jednej, początkowej rodziny. Każda rodzina składa się z tylu rojów, ile kryteriów podlega optymalizacji, a każdy z nich wykonuje optymalizację globalną jednego z tych kryteriów. Jednocześnie wszystkie cząstki z danej rodziny są przyciągane do elementów ich wspólnego archiwum rozwiązań niezdominowanych. Na tych dodatkowych liderów są wybierane izolowane punkty w przestrzeni rozwiązań. Motywuje to roje do poruszania się pomiędzy nimi, zapobiegając zbieganiu się w pojedynczych punktach i prowadzi do coraz dokładniejszego przybliżania rzeczywistego zbioru Pareto. Archiwa rodzin rojów reprezentują skupiska w „analizie skupień". Gdy dwa z nich znajdą się w pobliżu siebie, następuje ich połączenie. Analiza wyników optymalizacji wybranych problemów wykazała, że skupiska zwracane przez algorytm MOSF odpowiadają rzeczywistym skupiskom optymalnych zbiorów Pareto tych problemów.

Podczas wyboru liderów cząstek algorytm MOSF skupia się na elementach bieżącego przybliżenia optymalnego zbioru Pareto, pomijając ich odpowiedniki z przestrzeni wartości, dzięki czemu jest niezależny od ich rozkładu. Nie ma dla niego znaczenia, czy rozkład ten przypomina krzywą, powierzchnię oraz czy jest ciągły. Analiza wyników w oparciu o miarę ADF wykazała, że podejście to nie wpływa negatywnie na zdolność algorytmu MOSF do dokładnego przybliżania zawartości optymalnego frontu Pareto, gdyż okazał się w tym bardziej skuteczny od algorytmów NSGA-II i NSPSO, w szczególności przy większych liczbach optymalizowanych kryteriów.

Również opracowany przez Autora rozprawy archiwizator algorytmu MOSF ma za zadanie – oprócz ograniczania liczby zapamiętanych rozwiązań niezdominowanych – zapewnianie jednorodnej reprezentacji optymalnego zbioru Pareto. Pod tym pojęciem rozumiane jest zachowywanie w archiwum punktów izolowanych oraz reprezentantów ich gęstych skupisk. Analiza wyników w oparciu o miarę ADS wykazała, że podejście to pozwala na dokładniejsze odwzorowanie zawartości optymalnego zbioru Pareto wybranych funkcji testowych od algorytmów NSGA-II i NSPSO. Widać to szczególne w przypadku okresowej funkcji F_1 (Banach 1).

Złożoność obliczeniowa pojedynczej iteracji algorytmu MOSF jest liniowa ze względu na liczbę rodzin, rojów, cząstek i optymalizowanych kryteriów. Pozwala to na efektywne rozwiązywanie problemów optymalizacyjnych składających się w teorii z dowolnej liczby kryteriów. Złożoność obliczeniowa algorytmów NSGA-II i NSPSO jest kwadratowa ze względu na liczbę osobników. W algorytmie MOSF również występuje kwadratowa zależność, ale od liczby elementów archiwum, której maksymalna wartość może być konfigurowana niezależnie od liczby cząstek. Na przykład, pozwala to na uruchomienie rodzin składających się z wielotysięcznych rojów, ale posługujących się archiwami o maksymalnym rozmiarze nie przekraczającym 100.

Procedura archiwizacji algorytmu MOSF także charakteryzuje się nieliniową złożonością obliczeniową ze względu na liczbę wymiarów przestrzeni rozwiązań spowodowaną użyciem algorytmu przybliżonego poszukiwania najbliższych sąsiadów. Analiza czasu trwania optymalizacji wykazała jednak, że nie prowadzi to do spowolnień jego działania. W szczególności, w przypadku funkcji F_2 (Osyczka 2), algorytm MOSF okazał się średnio od 15 do 25 procent szybszy algorytmów od NSGA-II i NSPSO, co świadczy również o możliwości jego wydajnej implementacji.

Kryteria tworzone przez algorytm MPB mogą być z powodzeniem stosowane w badaniach nad algorytmami optymalizacji wielokryterialnej, choć ich użycie podlega ograniczeniom z powodu potrzeby wyczerpującego wyszukiwania referencyjnego przybliżenia optymalnego zbioru Pareto.

Podsumowując, zaproponowany przez Autora niniejszej rozprawy algorytm MOSF, jest efektywnym narzędziem optymalizacji wielokryterialnej, którego przydatność w ogólnych zastosowaniach została potwierdzona poprzez porównanie zwróconych przez niego wyników z wynikami algorytmów NSGA-II i NSPSO. Oferuje on funkcjonalność, której te metody nie posiadają, a stosowanie binarnej strategii turniejowej Deba pozwala mu na sprawną obsługę funkcji ograniczeń.

Do najbliższych planów rozwoju algorytmu MOSF należy sprawdzenie relacji pomiędzy zewnętrznymi i wewnętrznymi liderami cząstek, zmniejszenie wartości miary NC zwracanych wyników oraz porównanie go z algorytmem U-NSGA-III.

4.2. Modyfikacja modelu FOD

Opracowanie przez Autora rozprawy sposobu zastąpienia algorytmu opartego na średnicach analizą składowych głównych do układania atomów efektywnych białka zgodnie z osiami układu współrzędnych pozwoliło na zmniejszenie złożoności obliczeniowej procedury wyznaczania rozkładu hydrofobowości teoretycznej modelu FOD ze względu na liczbę reszt z liniowo-logarytmicznej (oczekiwanej, a w najgorszym przypadku kwadratowej) do liniowej we wszystkich przypadkach.

Oprócz nawet siedmiokrotnego przyspieszenia obliczeń, przekładającego się na bardziej efektywne stosowanie modelu FOD jako kryterium optymalizacyjnego, umożliwiło to uproszczenie jego implementacji dzięki wyeliminowaniu potrzeby stosowania algorytmów wyszukiwania wyczerpującego i otoczki wypukłej.

Ponieważ analiza składowych głównych jest mniej wrażliwa na położenia odstające, układanie atomów efektywnych zgodnie z osiami układu współrzędnych przy jej pomocy prowadzi do faktycznej maksymalizacji wariancji ich położeń w kolejnych wymiarach przestrzeni, zgodnie z potrzebami trójwymiarowej funkcji Gaussa. Na przykładzie białka `1AON` wykazano również, że umożliwia to wykrywanie osi symetrii cząsteczek, dzięki czemu, nie musi ich wskazywać użytkownik.

Porównanie wartości RD białek z bazy danych rozprawy, obliczonych stosując bieżący (FOD-MAX) oraz nowy (FOD-PCA) algorytm wykazało bardzo wysoką korelację pomiędzy uzyskanymi wynikami: 0,963 dla łańcuchów i 0,945 dla kompleksów. Wniosek z tej obserwacji jest taki, że zaproponowane rozwiązanie przynosi korzyść w postaci przyspieszenia obliczeń bez negatywnych efektów ubocznych, gdyż zachowuje wsteczną kompatybilność z dotychczas opublikowanymi pracami.

Wprowadzenie analizy składowych głównych do modelu FOD jest kolejnym krokiem w jego udoskonalaniu. Inne rozwiązania są w dalszych planach.

4.3. Analiza białek z bazy danych

Autor rozprawy opracował zestaw 12 kryteriów, dzięki którym możliwe było wybranie z bazy PDB 200 białek homodimerycznych spełniających założenia eksperymentu przewidywania ich struktury natywnej spośród prawie 30 tysięcy charakteryzujących się według RCSB stechiometrią A2. Kryteria te mogą być modyfikowane w celu wybieranie podstawowych zbiorów białek na potrzeby innych eksperymentów.

Analiza odległości pomiędzy atomami należącymi do różnych łańcuchów w białkach z bazy danych wykazała, że heurystyczne dodanie atomów wodoru powoduje wypełnienie nimi przestrzeni pomiędzy tymi łańcuchami. W 169 kompleksach zbliżyły się one do siebie na odległość niższą niż wynosił przyjęty promień kolizji (1,9 Å) – średnio około 1,761 Å. Spowodowany przez to wzrost wartości potencjału van der Waalsa wymusił wprowadzenie zmian w sposobie obliczeń energii.

W celu umożliwienia dotarcia podczas eksperymentu do struktury natywnej kompleksów posługując się kryterium pola wewnętrznego, Autor rozprawy zaproponował wprowadzenie „dolnego" promienia odcięcia, powodującego, że jeżeli dwa atomy znajdą się bliżej siebie niż 1,9 Å, traktowane będą jakby znajdowały się w tej odległości. Pozwoliło to na obniżenie średniej wartości energii w kompleksach natywnych do $-55{,}516\,\frac{\text{kcal}}{\text{mol}}$. Wartości te uzyskano poprzez zastosowane również „górnego" promienia odcięcia, wynoszącego 12 Å, którego zadaniem było skrócenie czasu trwania obliczeń poprzez ograniczenie liczby par atomów branych pod uwagę przez potencjały niekowalencyjne.

Prawie trzy czwarte (131) łańcuchów białek z bazy danych było zgodne z modelem FOD. Tylko w 9 przypadkach utworzyły one zgodny kompleks, natomiast nie zdarzyła się sytuacja, w której stabilny kompleks został utworzony przez niestabilne łańcuchy. Przypuszcza się, że powodem tego może być kształt wybranych białek homodimerycznych uniemożliwiający im uzyskanie dostatecznie niskich wartości RD. Nie jest to sprzeczne z założeniami modelu FOD, ponieważ według nich, kompleks powinien dążyć do minimalizacji tej wartości. Ponieważ łańcuchy były traktowane jako bryły sztywne, minimum to mogło znajdować się w górnej połowie osi wartości RD. Opracowany eksperyment miał wykazać czy jest tak w rzeczywistości.

Korelacja wartości RD łańcuchów i ich kompleksów o współczynniku wynoszącym 0,517 sugeruje natomiast możliwość oszacowania tego, czy kompleks będzie zgodny z modelem. Ma to znaczenie podczas wyboru sposobu jego przewidywania: jeżeli RD jest niskie, większą skuteczność może osiągnąć optymalizacja globalna, a w przeciwnym przypadku – wielokryterialna.

4.4. Kompleksowanie białek

Autor rozprawy opracował eksperyment przewidywania struktury czwartorzędowej białek polegający na optymalizacji globalnej i wielokryterialnej kryteriów pola zewnętrznego i wewnętrznego. Zastosowanie funkcji jednoznacznie odwzorowującej elementy przestrzeni rozwiązań na elementy przestrzeni konformacyjnej kompleksu pozwoliło na uniknięcie problemów związanych z obrotami bryły liganda. Oprócz tego, wprowadzone zostały trzy funkcje ograniczeń zapewniające utworzenie prawidłowego (pozbawionego kolizji) kompleksu. Dzięki ich obsłudze przy pomocy strategii turniejowej Deba, pozwalającej na pominięcie obliczeń wartości optymalizowanych kryteriów dla rozwiązań niedopuszczalnych (w szczególności tych, w których łańcuchy się przeniknęły), umożliwiło około sześciokrotne skrócenie przeciętnego czasu symulacji: w trybie wielokryterialnym, z 90 do 15 minut.

Stwierdzono, że użycie w opracowanym eksperymencie algorytmu PSO i algorytmu MOSF doprowadziło do utworzenia kompleksów par łańcuchów dopuszczalnych przez wszystkie funkcje ograniczeń. Podczas optymalizacji globalnej, we wszystkich białkach z bazy danych rozprawy, cząstki osiągnęły zbieżność w wybranych przez nie rozwiązaniach. Podczas optymalizacji wielokryterialnej odnalezione zostały natomiast konformacje niezdominowane w przestrzeni wartości RD i energii. Potwierdza to możliwość skutecznego stosowania metod optymalizacyjnych opartych na roju cząstek w badaniach nad przewidywaniem kompleksów typu białko-białko.

Inną zaletą stosowania algorytmów optymalizacyjnych, dającą im przewagę nad metodami wyszukiwania wyczerpującego, jest możliwość symulowania zjawisk zachodzących w układach zawierających więcej niż dwie cząsteczki. Do opisu konformacji kompleksu składającego się z n z nich wystarcza bowiem $6\,(n-1)$ zmiennych.

Optymalizacja globalna kryterium pola zewnętrznego wykazała tendencję modelu FOD do tworzenia kompleksów o licznych interfejsach pomiędzy łańcuchami. Zachowanie to jest zgodne z jego założeniami. Ze względu na kształt większości kompleksów z bazy danych, z nielicznymi wyjątkami, maksymalizacja kontaktów pomiędzy resztami była jedną możliwością zwiększania hydrofobowości w ich środku geometrycznym. Pole wewnętrzne wykazało natomiast tendencję przeciwną, zmierzającą do odsuwania łańcuchów od siebie. Minimalizacja wartości RD prowadziła do uzyskiwania wysokich energii, a minimalizacja energii prowadziła do uzyskiwania wysokich wartości RD. Z tego powodu, obydwa pola wskazały tylko kilka konformacji kompleksu podobnych do siebie pod względem położenia liganda, ale różnych od natywnych. Wniosek z tego jest taki, że pola te reprezentują przeciwne siły.

W 19 spośród 200 białek z bazy danych rozprawy optymalizacja globalna kryterium pola zewnętrznego doprowadziła do utworzenia struktur kompleksów zbliżonych do natywnych. Wśród nich, 7 charakteryzowało się wartością RMSD niższą od 5 Å. Oznacza to, że proces powstawania *in vivo* niemal 10% analizowanych kompleksów może być (przynajmniej częściowo) odwzorowany *in silico* poprzez modelowanie oddziaływań hydrofobowych przy pomocy modelu FOD. Wartość współczynnika korelacji pomiędzy wartościami RD przed i po symulacji wynosząca powyżej 0,7 sugeruje, że w konformacjach natywnych tych białek różnice pomiędzy rozkładami hydrofobowości faktycznie należą do jednych z najniższych.

Optymalizacja globalna kryterium pola wewnętrznego doprowadziła do osiągnięcia przez łańcuchy konformacji zbliżonej do natywnej w 7 przypadkach, aczkolwiek nie aż tak jak w przypadku pola zewnętrznego. Oznacza to, że posługiwanie się wyłącznie energią oddziaływań w obrębie interfejsu pomiędzy łańcuchami może być niewystarczające do wybrania właściwej konformacji.

Optymalizacja wielokryterialna kryteriów pól zewnętrznego i wewnętrznego, symulująca ich równoczesny wpływ na układ pary łańcuchów, doprowadziła do utworzenia 46 kompleksów o strukturze zbliżonej do natywnej, czyli niemal 25% białek z bazy danych rozprawy. Wniosek z tego jest taki, że w sytuacji, gdy pod uwagę brane są oddziaływania hydrofobowe oraz energia oddziaływań niekowalencyjnych, ich przeciwne dążenia powodują, że łańcuchy docierają do konformacji o wartościach RD wyższych od 0,5. Ponieważ prawie wszystkie kompleksy z bazy danych rozprawy były niezgodne z modelem FOD, umożliwiło to na odnalezienie takich frontów Pareto w pobliżu, których znajdowały się struktury natywne.

Wynikiem optymalizacji wielokryterialnej były zbiory Pareto, zawierające niezdominowane konformacje kompleksów. Dzięki algorytmowi MOSF, który je odnalazł w przestrzeni wartości RD i energii, możliwe było połączenie ich w skupiska odpowiadające różnym sposobom ułożenia receptorów względem ligandów. Pozwoliło to na zmniejszenie liczby wyników z kilkunastu lub kilkudziesięciu nierozróżnialnych w sensie optimum Pareto konformacji do kilku różnych od siebie grup. Grupy te mogą być następnie poddane osobnej analizie przy pomocy innych metod, decydujących, która z nich powinna być tą reprezentującą wynik symulacji.

Wyniki przedstawione w niniejszej rozprawie potwierdzają, że istnieją kompleksy białkowe, które spełniają założenia modelu FOD i mogą być przy jego pomocy przewidziane. Daje to podstawę do stwierdzenia jego przydatności jako kryterium w symulacjach procesów biochemicznych związanych z białkami, a także stanowi impuls do dalszych prac nad nim i przy jego pomocy.

4.5. Innowacyjność rozwiązań

Niniejsza rozprawa doktorska prezentuje nowatorskie i – zgodnie ze stanem wiedzy jej Autora – do tej pory niepodejmowane podejścia do tematyki optymalizacji wielokryterialnej oraz badań nad wpływem oddziaływań hydrofobowych na procesy związane z białkami. Ponieważ zaproponowane w niniejszej rozprawie doktorskiej rozwiązania wraz ze wskazaniem ich przydatności w nauce zostały szczegółowo przedstawione w tym rozdziale, poniżej znajduje się ich wyszczególnienie:

1. Opracowano autorski algorytm optymalizacji wielokryterialnej o nazwie wielokryterialne rodziny rojów (multi objective swarm families, MOSF), oparty na zasadzie działania roju cząstek, zdolny do wykonywania „analizy skupień" odnalezionych rozwiązań niezdominowanych oraz jednorodnej reprezentacji optymalnego zbioru Pareto.

2. Wykazano wyższą dokładność algorytmu MOSF od algorytmów NSGA-II i NSPSO w przybliżaniu zawartości optymalnego zbioru i frontu Pareto wybranych funkcji testowych i kryteriów wygenerowanych przez MPB. Wykazano również, że osiągnięcie tej dokładności było możliwe w porównywalnym lub krótszym czasie.

3. Wykazano dokładność procedury „analizy skupień" algorytmu MOSF we wskazywaniu faktycznych skupisk elementów optymalnego zbioru Pareto.

4. Zaproponowano dwie nowe, uniwersalne miary oceny wyników zwracanych przez algorytmy optymalizacji wielokryterialnej: ADS i ADF. Miary te umożliwiły porównanie algorytmu MOSF z algorytmami NSGA-II i NSPSO z perspektywy referencyjnych zbiorów i frontów Pareto, co pozwoliło na dokładne oszacowanie ich przybliżenia przez te algorytmy.

5. Zaprezentowano użycie generatora MPB do tworzenia kryteriów testowych do porównywania sprawności algorytmów optymalizacji wielokryterialnej.

6. Wykazano, że algorytm MOSF może być wykonywany równolegle i że zysk czasowy z tego wynikający jest proporcjonalny do liczby jednostek obliczeniowych.

7. Zaproponowano modyfikację modelu FOD dotyczącą sposobu układania atomów efektywnych białka zgodnie z osiami układu współrzędnych opartą na analizie składowych głównych. Umożliwiło to znaczne zmniejszenie złożoności obliczeniowej procedury wyznaczania rozkładów hydrofobowości.

8. Stwierdzono, że modyfikacja modelu FOD nie wpływa istotnie na zmianę statusu jądra hydrofobowego większości białek z bazy danych rozprawy. Pozwala to założyć, że stosowanie tej modyfikacji w przyszłości będzie zachowywać wsteczną kompatybilność z wnioskami z dotychczas opublikowanych prac.

9. Zastosowano zestaw kryteriów umożliwiających wybranie reprezentacyjnego zbioru różnorodnych białek homodimerycznych z bazy PDB, którego modyfikacje mogą być używane do tworzenia baz danych dla innych eksperymentów.

10. Opracowano eksperyment *in silico* przewidywania struktury czwartorzędowej białek polegający na optymalizacji globalnej i wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego przy pomocy algorytmów roju cząstek.

11. Opracowano sposób reprezentacji konformacji kompleksu w przestrzeni rozwiązań oraz zestaw funkcji gwarantujących uzyskiwanie jego prawidłowej struktury, rozumianej jako obecność kontaktów niewiążących i brak kolizji atomów.

12. Wykazano skuteczność stosowania algorytmów opartych na zasadzie działania roju cząstek (jedno- i wielokryterialnych) w opracowanym eksperymencie.

13. Stwierdzono, że poszukiwanie natywnej struktury czwartorzędowej białek w oparciu o oddziaływania hydrofobowe (w ujęciu modelu FOD) osiąga wyższą skuteczność od podejścia opartego wyłącznie na energii oddziaływań pomiędzy atomami, obliczają zgodnie z własną implementacją pola ECEPP/3.

14. Stwierdzono, że modelowanie równoczesnego wpływu pola zewnętrznego i wewnętrznego przy pomocy optymalizacji wielokryterialnej prowadzi uzyskania większej liczby struktur kompleksów zbliżonych do natywnych niż w przypadku optymalizacji globalnej tych pól.

15. Stwierdzono istnienie kompleksów homodimerycznych, których powstaniem kierują oddziaływania hydrofobowe, takich, które powstają w wyniku starcia oddziaływań hydrofobowych z niekowalencyjnymi, a także takich, dla których oddziaływania niekowalencyjne są neutralne.

16. Potwierdzono założenia modelu FOD dotyczące wpływu oddziaływań hydrofobowych na proces tworzenia się kompleksów białkowych poprzez wykazanie, że istnieją białka, w których te założenia są spełnione. W niektórych przypadkach osiągnięcie struktury bliskiej natywnej jest możliwe dzięki samemu polu zewnętrznemu, a w innych potrzebne jest dołączenie do niego pola wewnętrznego.

5. Podsumowanie

Inspiracją do podjęcia tematu niniejszej rozprawy doktorskiej było zainteresowanie Autora bioinformatyką, w szczególności proteomiką.

Wciąż trwają poszukiwania modeli komputerowych umożliwiających coraz bardziej dokładne odwzorowywanie reakcji chemicznych, w których biorą udział białka, mających na celu lepsze poznanie tych cząsteczek, a przez to opracowanie coraz bardziej skutecznych leków. Autor rozprawy jest zaangażowany w rozwój i badania przy użyciu jednego z tych modeli – modelu rozmytej kropli oliwy (FOD).

Model FOD opisuje wpływ oddziaływań hydrofobowych na strukturę i funkcję białek, wynikających z ich reakcji z otaczającym je środowiskiem wodnym. Wyniki badań, w których model ten był stosowany zostały przedstawione w wielu pracach opublikowanych w znaczących czasopismach i wydaniach książkowych.

Woda jest czynnikiem decydującym o aktywności biologicznej białek. Symulowanie jej wpływu na te cząsteczki umożliwia między innymi przewidywanie tworzenia się ich struktury trzecio- i czwartorzędowej. Założenia modelu FOD dotyczące tych procesów nie zostały do tej pory sprawdzone eksperymentalnie, co jest warunkiem koniecznym do jego dalszego rozwoju. Z tego powodu, Autor rozprawy, jako osoba posiadająca wykształcenie informatyczne oraz umiejętności programistyczne przydatne w pracy badawczej, podjął się wykonania eksperymentu realizującego ten cel.

Opracowany i wykonany przez Autora rozprawy eksperyment *in silico* przewidywania tworzenia się kompleksów typu białko-białko wykazał, że istnieją białka homodimeryczne, których struktura czwartorzędowa wpisuje się w założenia modelu FOD. Nie oczekuje się, że kompleksy białkowe będą powstawać według tego samego schematu, dającego opisać się przy pomocy jednego modelu. Proces ten jest najprawdopodobniej bardzo zróżnicowany. Uzyskane wyniki świadczą o tym, że model FOD może być z powodzeniem stosowany w przypadku części białek, co sugeruje, że należy kontynuować badania w tym temacie, w szczególności nad identyfikacją struktur, które mogą wpisywać się w założenia modelu FOD, a także możliwościami stosowania ich wspólnie z innymi podejściami.

Wychodząc na przeciw zagadnieniom możliwości włączenia modelu FOD do zbioru kryteriów oceny stosowanych w symulacjach procesów biologicznych, Autor rozprawy zastosował optymalizację wielokryterialną do zaobserwowania równoczesnego wpływu oddziaływań hydrofobowych oraz elektrostatycznych, van der Waalsa i wiązań wodorowych na tworzenie się kompleksów białkowych.

Optymalizacja wielokryterialna umożliwiła „złączenie" dwóch odmiennych, a przez to niekompatybilnych podejść: modelu FOD i pola ECEPP/3. Zostało wykazane, że efektem tego było znaczne zwiększenie precyzji przewidywania struktury czwartorzędowej analizowanych białek homodimerycznych.

Modelowanie układu cząsteczek przy pomocy metod optymalizacyjnych pozwala na obserwację tworzenia się ich kompleksu w sposób bardziej zbliżony do rzeczywistości w porównaniu z metodami przeszukiwania wyczerpującego. Umożliwia to także symulacje większej liczby struktur niż dwie. Wykazano, że użycie algorytmów opartych na zasadzie działania roju cząstek może być z powodzeniem stosowanie do przeszukiwania podzbiorów przestrzeni konfiguracyjnej wychodzących poza dokowanie w obrębie kieszeni wiązania liganda.

Przeprowadzenie eksperymentu kompleksowania typu białko-białko było możliwe dzięki użyciu algorytmu MOSF. Algorytm ten został opracowany przez Autora rozprawy i również w niej przedstawiony. Wykazano, że charakteryzuje się on wyższą dokładnością w odwzorowywaniu optymalnego zbioru Pareto od powszechnie stosowanych metod optymalizacji wielokryterialnej w przypadku wybranych problemów, co przemawia o jego możliwościach w ogólnych zastosowaniach.

Metody informatyczne są nierozłącznym elementem badań naukowych z dziedziny bioinformatyki. Umożliwiają one modelowanie zjawisk zachodzących *in vivo* za pomocą technik komputerowych, obniżając koszty i skracając czas trwania prac laboratoryjnych. Niniejsza rozprawa doktorska jest jednym z etapów ciągłego dążenia do poszerzania wiedzy na temat białek. Dalsze plany związane z poruszoną w niej tematyką dotyczą:

- udoskonalania algorytmu MOSF i porównanie jego możliwości z algorytmem U-NSGA-III,
- kontynuacji badań nad hydrofobowością w białkach przy użyciu modelu FOD,
- opracowywania kolejnych modyfikacji usprawniających działanie modelu FOD,
- rozszerzania opracowanego eksperymentu kompleksowa na inne zbiory białek,
- wzięcia udziału w inicjatywie CAPRI.

A. Rysunki

Dodatek A zawiera graficzne prezentacje optymalnych zbiorów i frontów Pareto wybranych funkcji testowych, profile miar oceny wyników ich optymalizacji przez algorytmy MOSF, NSGA-II i NSPSO, a także wizualizacje wyników eksperymentu przewidywania struktury czwartorzędowej białek, które okazały się zgodne z ich strukturami natywnymi. Ponieważ każdy z tych rysunków zajmuje wraz z podpisem całą stronę, umieszczenie ich tutaj zostało podyktowane chęcią zapewnienia lepszego przepływu tekstu w rozdziale 3, a przez to poprawy czytelności całej rozprawy.

A.1. Algorytm MOSF

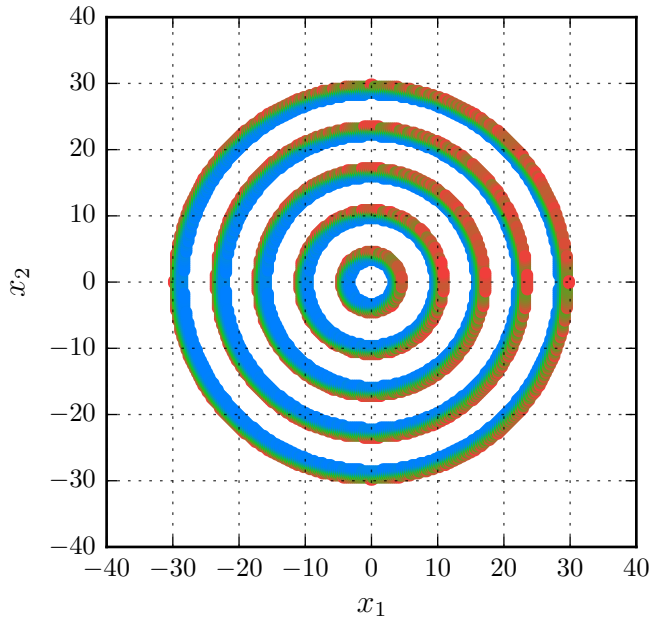
W pierwszej części tego dodatku są przedstawione dokładne przybliżenia optymalnych zbiorów i frontów Pareto czterech wybranych funkcji testowych:

- F_1 (Banach 1) – rysunek A.1
- F_2 (Osyczka 2) – rysunek A.2
- F_3 (Viennet 3) – rysunek A.3
- F_4 (Viennet 4) – rysunek A.4

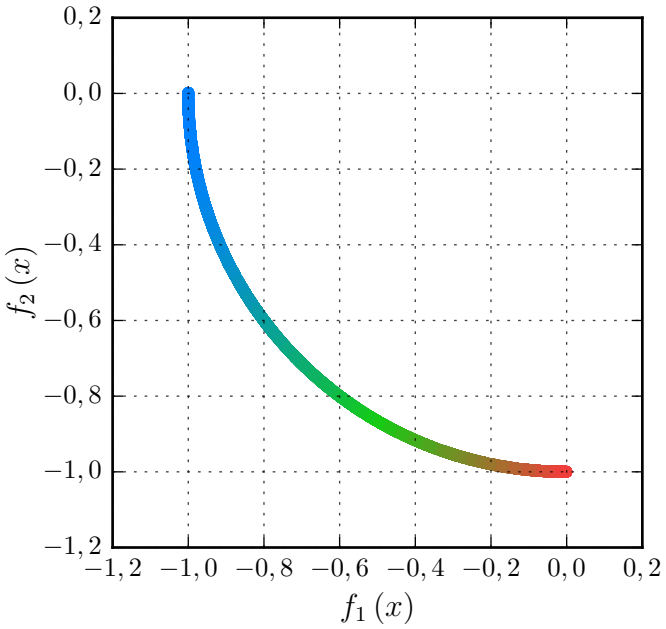
Dalej natomiast znajdują się rozkłady miar ADS, ADF, ER i NC użytych do oceny wyników uzyskanych przez algorytmy MOSF, NSGA-II i NSPSO podczas optymalizacji powyższych funkcji oraz kryteriów wygenerowanych przez MPB:

- Banach 1 – rysunek A.5
- Osyczka 2 – rysunek A.6
- Viennet 3 – rysunek A.7
- Viennet 4 – rysunek A.8
- MPB (2 kryteria) – rysunek A.9
- MPB (3 kryteria) – rysunek A.10
- MPB (4 kryteria) – rysunek A.11
- MPB (5 kryteriów) – rysunek A.12

Przybliżenia optymalnych zbiorów i frontów Pareto były poszukiwane wśród elementów siatek o gęstości 200 punktów w każdym wymiarze przestrzeni rozwiązań.

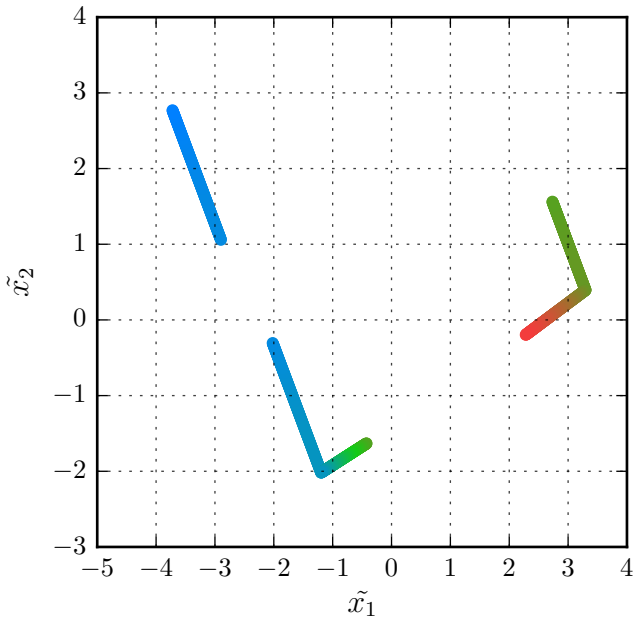


(a) Przestrzeń rozwiązań.

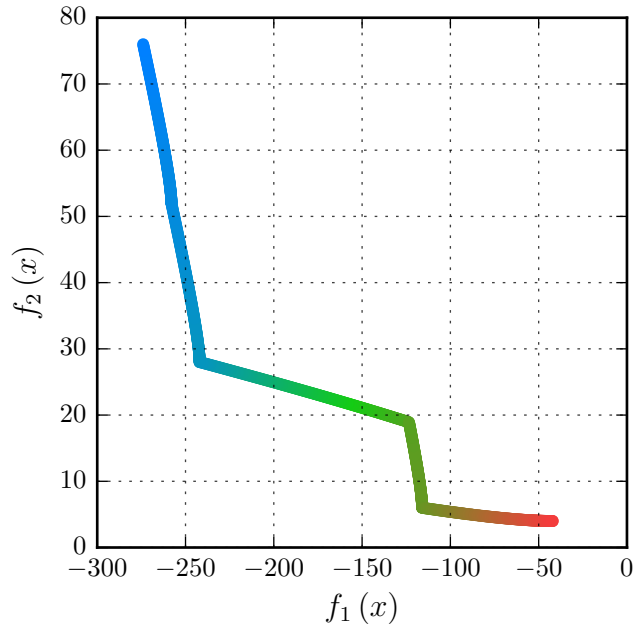


(b) Przestrzeń wartości.

Rysunek A.1: Optymalny zbiór i front Pareto funkcji Banach 1. Kolory znaczników wskazują na wartości kryterium f_1: niebieski – niskie, czerwony – wysokie.

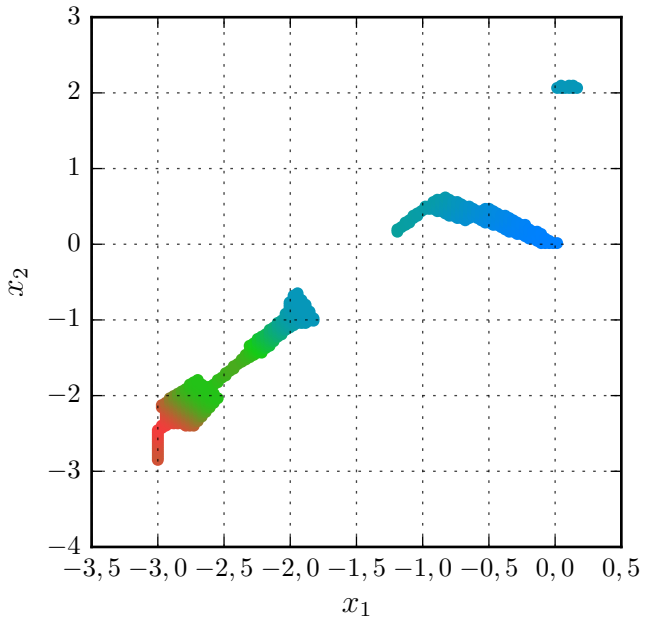


(a) Pierwsze dwie składowe główne przestrzeni rozwiązań.

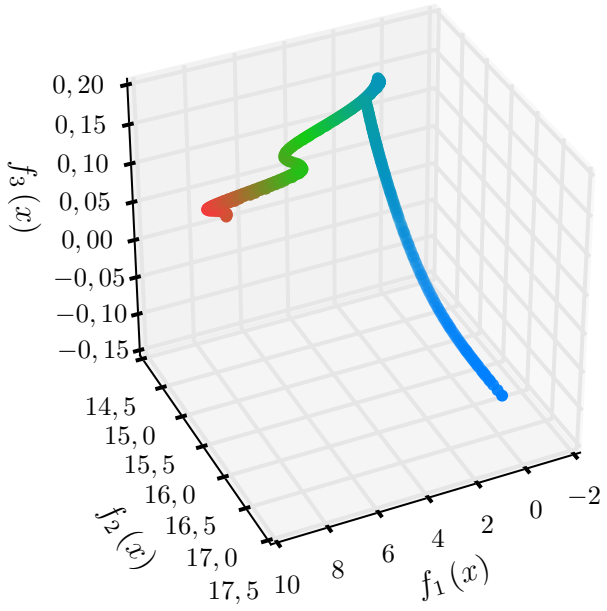


(b) Przestrzeń wartości.

Rysunek A.2: Optymalny zbiór i front Pareto funkcji Osyczka 2. Kolory znaczników wskazują na wartości kryterium f_1: niebieski – niskie, czerwony – wysokie.

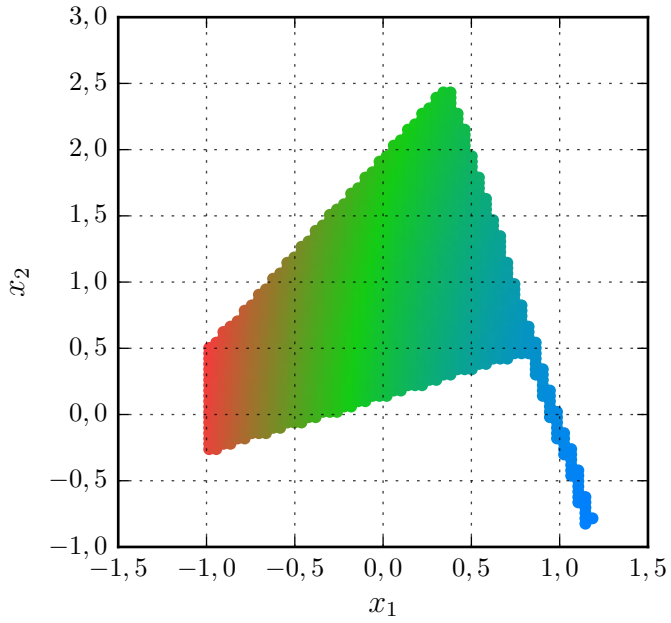


(a) Przestrzeń rozwiązań.

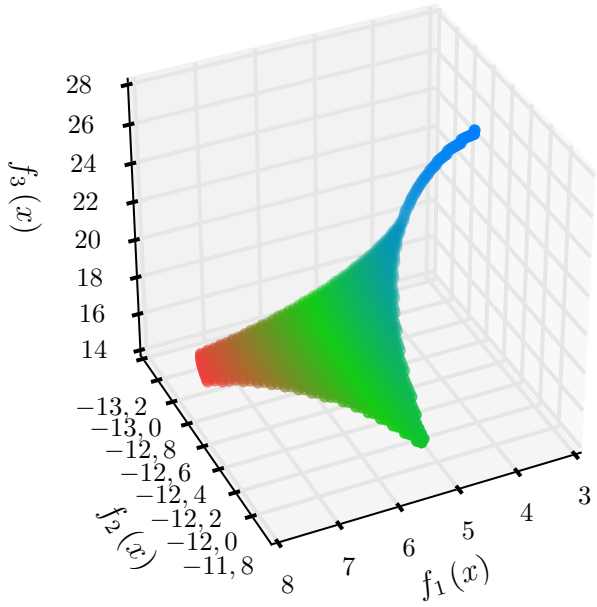


(b) Przestrzeń wartości.

Rysunek A.3: Optymalny zbiór i front Pareto funkcji Viennet 3. Kolory znaczników wskazują na wartości kryterium f_1: niebieski – niskie, czerwony – wysokie.

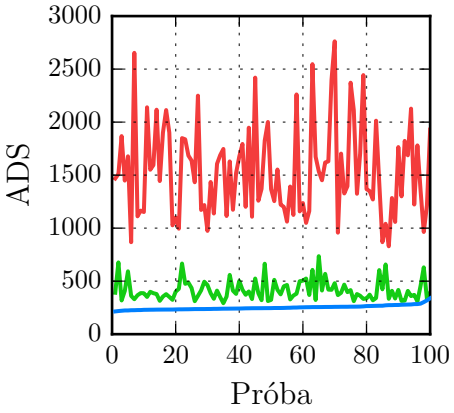


(a) Przestrzeń rozwiązań.

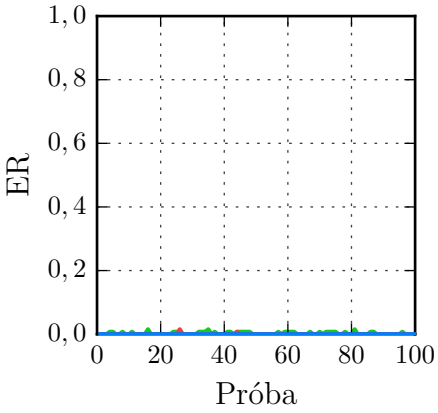


(b) Przestrzeń wartości.

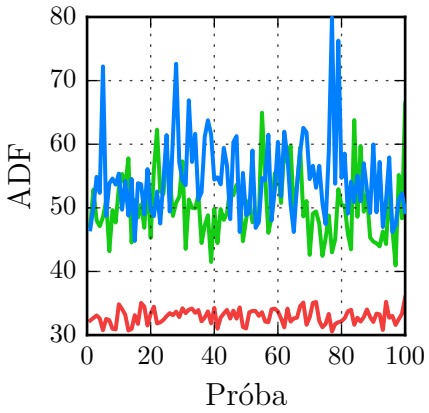
Rysunek A.4: Optymalny zbiór i front Pareto funkcji Viennet 4. Kolory znaczników wskazują na wartości kryterium f_1: niebieski – niskie, czerwony – wysokie.



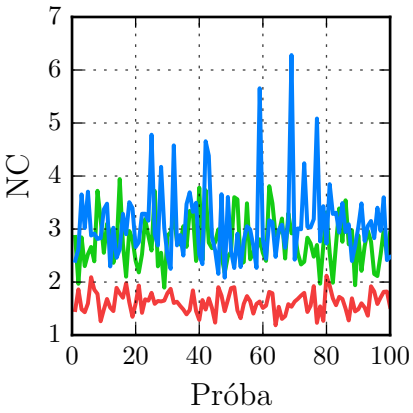
(a) Miara ADS.



(b) Miara ER.

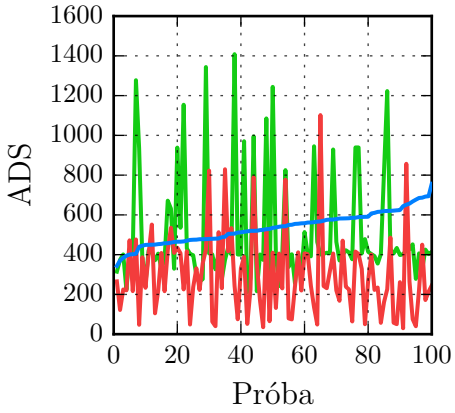


(c) Miara ADF.

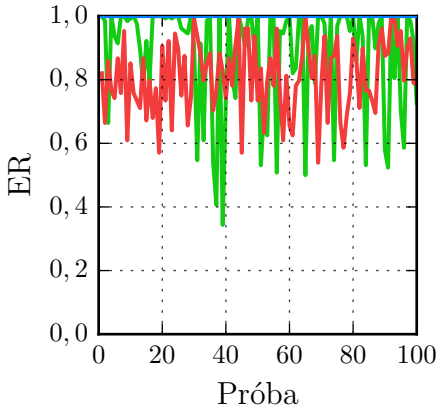


(d) Miara NC.

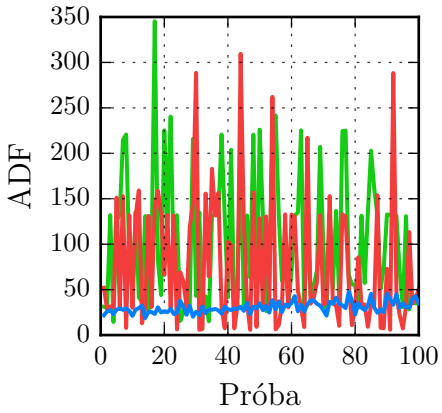
Rysunek A.5: Wartości miar oceny wyników optymalizacji funkcji Banach 1. Kolory rozkładów oznaczają algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Rozkłady są posortowane rosnąco zgodnie z rosnącymi wartościami miary ADS dla MOSF.



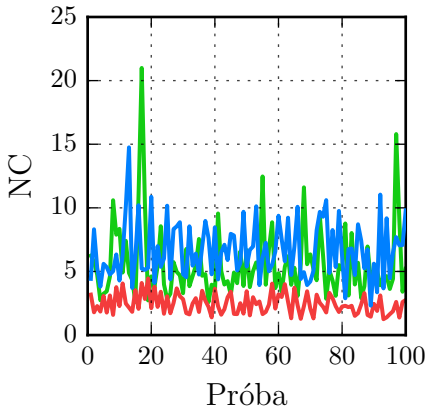
(a) Miara ADS.



(b) Miara ER.

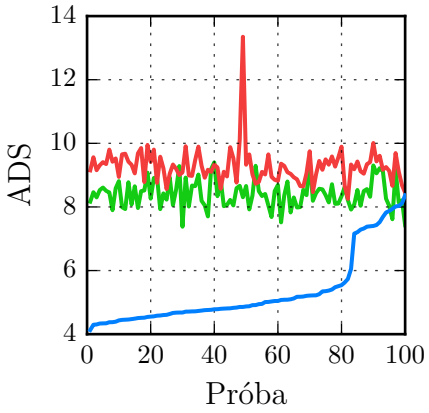


(c) Miara ADF.

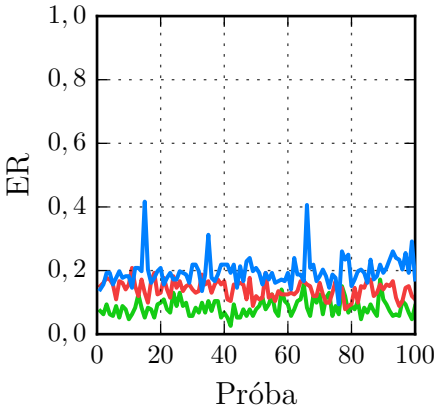


(d) Miara NC.

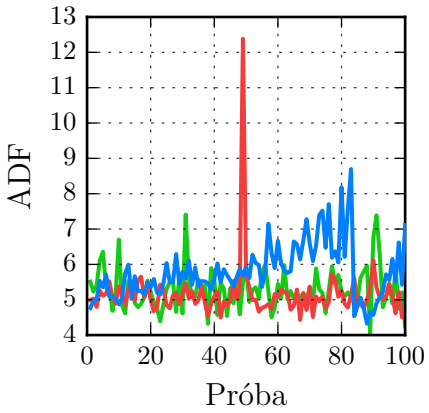
Rysunek A.6: Wartości miar oceny wyników optymalizacji funkcji Osyczka 2. Kolory rozkładów oznaczają algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Rozkłady są posortowane rosnąco zgodnie z rosnącymi wartościami miary ADS dla MOSF.



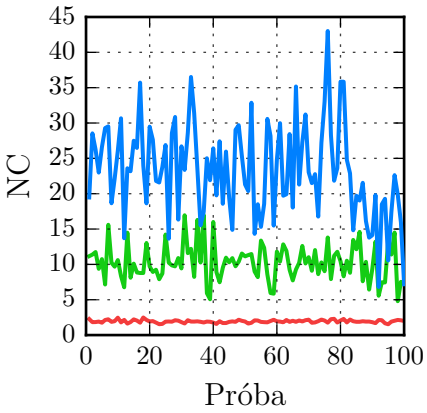
(a) Miara ADS.



(b) Miara ER.

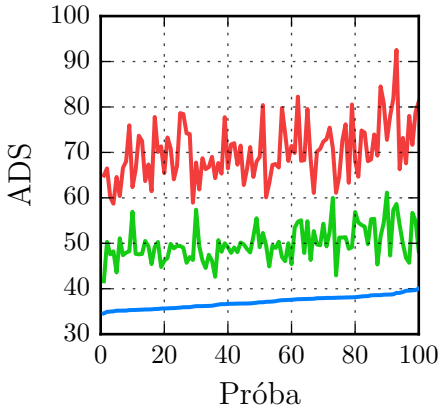


(c) Miara ADF.

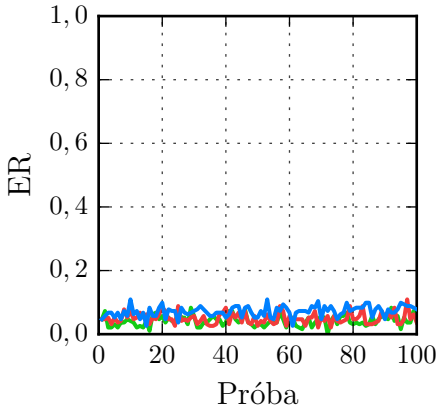


(d) Miara NC.

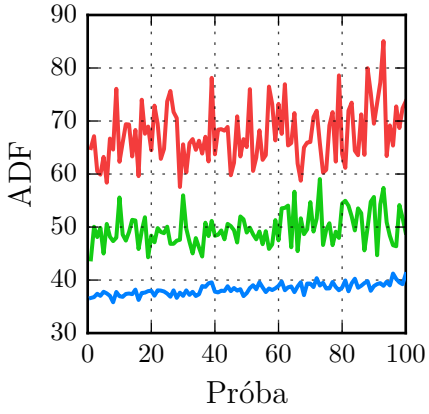
Rysunek A.7: Wartości miar oceny wyników optymalizacji funkcji Viennet 3. Kolory rozkładów oznaczają algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Rozkłady są posortowane rosnąco zgodnie z rosnącymi wartościami miary ADS dla MOSF.



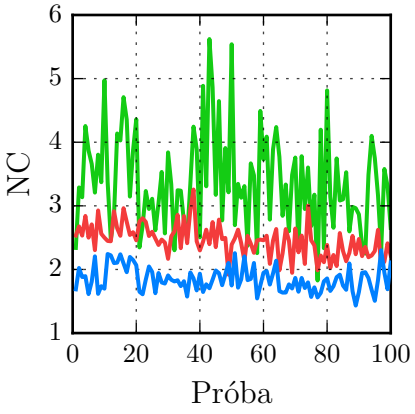
(a) Miara ADS.



(b) Miara ER.

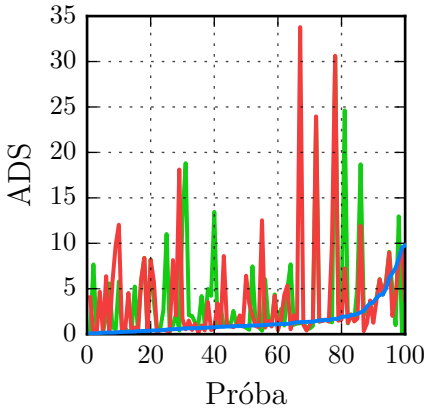


(c) Miara ADF.

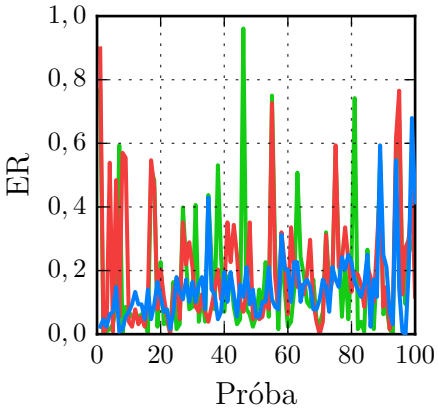


(d) Miara NC.

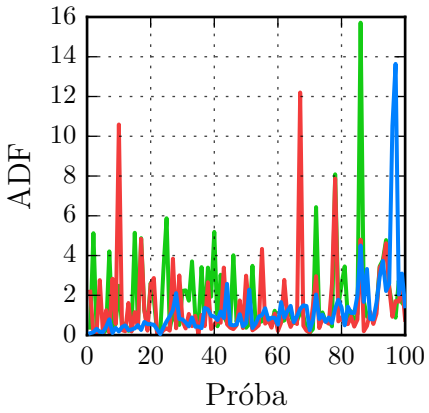
Rysunek A.8: Wartości miar oceny wyników optymalizacji funkcji Viennet 4. Kolory rozkładów oznaczają algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Rozkłady są posortowane rosnąco zgodnie z rosnącymi wartościami miary ADS dla MOSF.



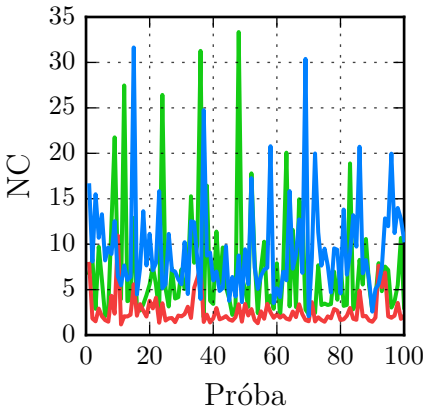
(a) Miara ADS.



(b) Miara ER.

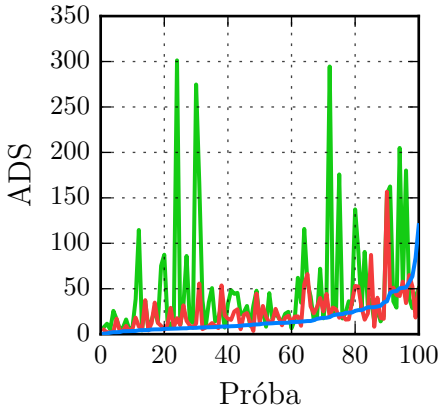


(c) Miara ADF.

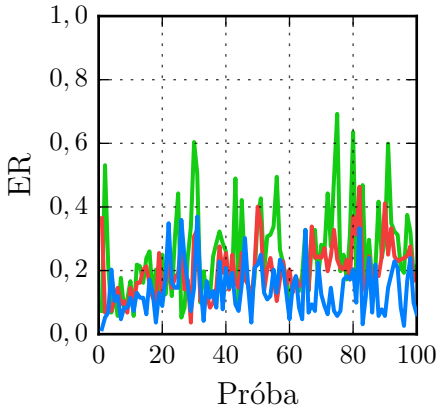


(d) Miara NC.

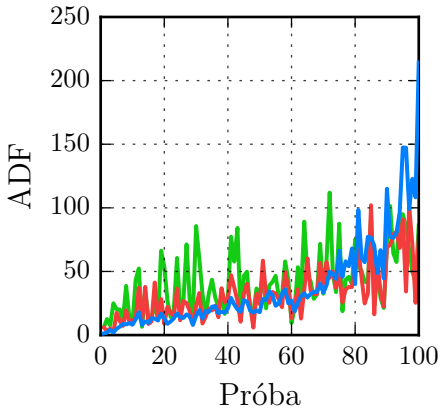
Rysunek A.9: Wartości miar oceny wyników optymalizacji dwóch kryteriów wygenerowanych przez MPB. Kolory rozkładów oznaczają algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Rozkłady są posortowane rosnąco zgodnie z rosnącymi wartościami miary ADS dla MOSF.



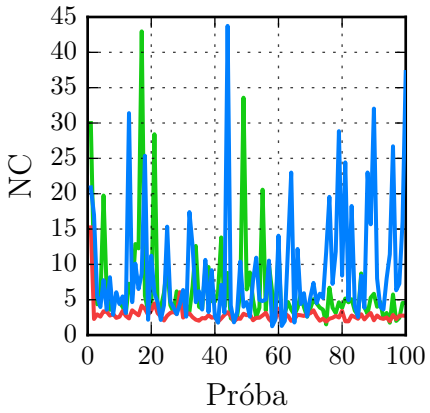
(a) Miara ADS.



(b) Miara ER.

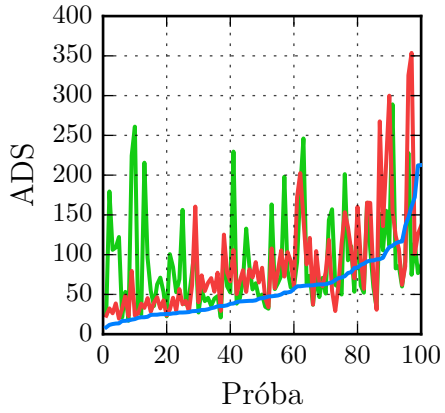


(c) Miara ADF.

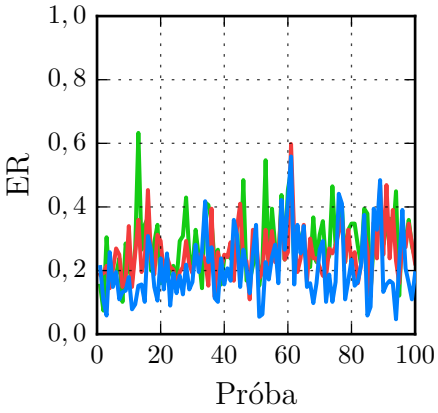


(d) Miara NC.

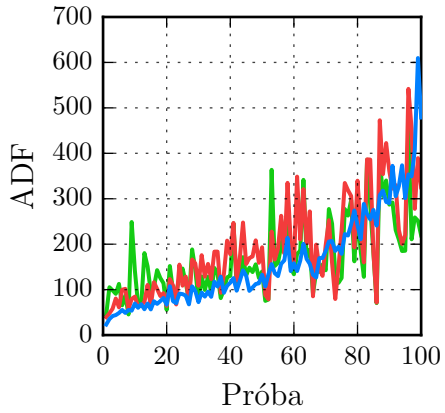
Rysunek A.10: Wartości miar oceny wyników optymalizacji trzech kryteriów wygenerowanych przez MPB. Kolory rozkładów oznaczają algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Rozkłady są posortowane rosnąco zgodnie z rosnącymi wartościami miary ADS dla MOSF.



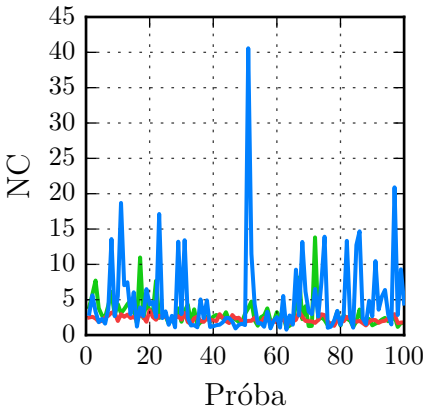
(a) Miara ADS.



(b) Miara ER.

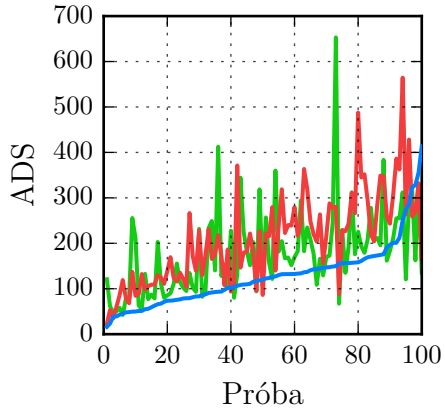


(c) Miara ADF.

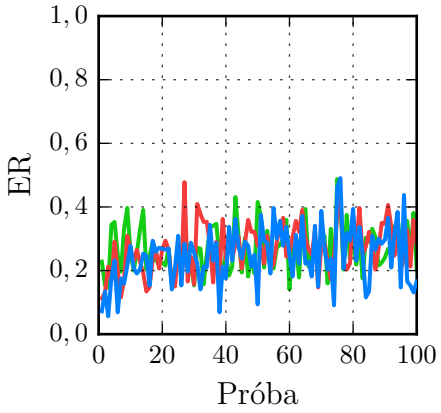


(d) Miara NC.

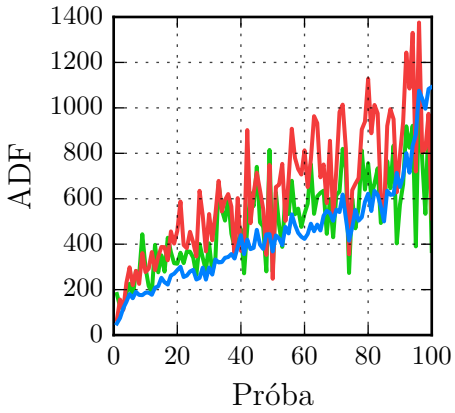
Rysunek A.11: Wartości miar oceny wyników optymalizacji czterech kryteriów wygenerowanych przez MPB. Kolory rozkładów oznaczają algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Rozkłady są posortowane rosnąco zgodnie z rosnącymi wartościami miary ADS dla MOSF.



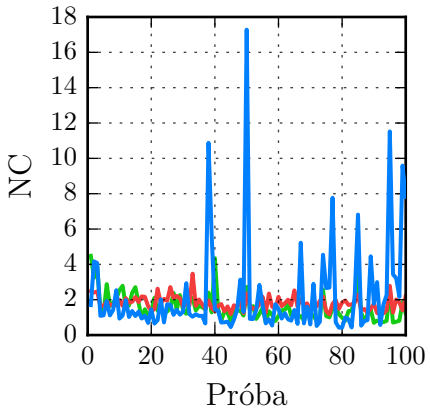
(a) Miara ADS.



(b) Miara ER.



(c) Miara ADF.



(d) Miara NC.

Rysunek A.12: Wartości miar oceny wyników optymalizacji pięciu kryteriów wygenerowanych przez MPB. Kolory rozkładów oznaczają algorytm: niebieski – MOSF, czerwony – NSGA-II, zielony – NSPSO. Rozkłady są posortowane rosnąco zgodnie z rosnącymi wartościami miary ADS dla MOSF.

A.2. Kompleksowanie białek

W drugiej części tego dodatku są przedstawione wizualizacje wyników eksperymentu przewidywania struktury czwartorzędowej białek homodimerycznych, dla których zaobserwowano wartość miary RMSD niższą od 10 Å lub wartość miary AUC wyższą od 0,75, oznaczające osiągnięcie zgodności z ich strukturami natywnymi. Sugeruje to, że siły wpływające na powstawanie tych kompleksów *in vivo* są dobrze opisywane przez jedno lub obydwa badane pola (zewnętrzne lub wewnętrzne).

Wyniki zostały podzielone na uzyskane w trakcie optymalizacji globalnej:

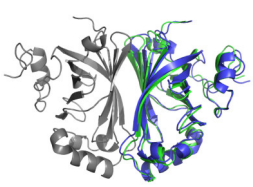
- pola zewnętrznego, RMSD < 10 Å i AUC $> 0{,}75$ – rysunek A.13
- pola zewnętrznego, RMSD < 10 Å i AUC $\leq 0{,}75$ – rysunek A.14
- pola zewnętrznego, RMSD ≥ 10 Å i AUC $> 0{,}75$ – rysunek A.15
- pola wewnętrznego, RMSD < 10 Å i AUC $\leq 0{,}75$ – rysunek A.16
- pola wewnętrznego i zewnętrznego (podobne konformacje) – rysunek A.17

oraz wielokryterialnej, wykonanej przy użyciu algorytmu MOSF:

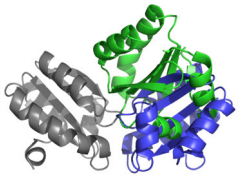
- RMSD < 10 Å i AUC $> 0{,}75$ – rysunek A.18
- RMSD < 10 Å i AUC $\leq 0{,}75$ – rysunki A.19 i A.20
- RMSD ≥ 10 Å i AUC $> 0{,}75$ – rysunek A.21

Algorytm MOSF podzielił wynikowe przybliżenia optymalnych zbiorów Pareto na podzbiory, które w tym eksperymencie odpowiadały grupom podobnych do siebie, ale różnych od pozostałych konformacji kompleksów. W celu zachowania czytelności tych rysunków, umieszczono na nich wyłącznie struktury reprezentacyjne, czyli te, którym w każdej grupie odpowiadała najniższa wartość miary ARC. Z tego samego powodu, grupy, w których nie stwierdzono obecności konformacji charakteryzujących się RMSD < 10 Å lub AUC $> 0{,}75$ również nie zostały tu uwzględnione.

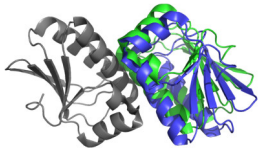
Każdy wynik jest przedstawiony w konformacji ze strukturą natywną kompleksu po nałożeniu na siebie atomów Cα z ich receptorów (łańcuchów A). Powoduje to w niektórych przypadkach zauważalne różnice w orientacjach ligandów (łańcuchów B). Należy jednak pamiętać, że zgodnie z założeniami eksperymentu, wartości RMSD znajdujące się w podpisach pod rysunkami były obliczane po nałożeniu na siebie całych kompleksów, przez co są niższe niż sugerują to te wizualizacje.



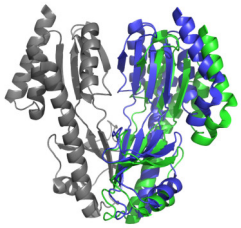
(a) Kompleks `2W2A`:
0,96 × 0,94 Å (0,05)



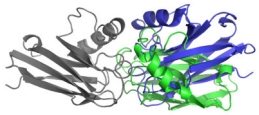
(b) Kompleks `2OE3`:
0,94 × 7,33 Å (0,19)



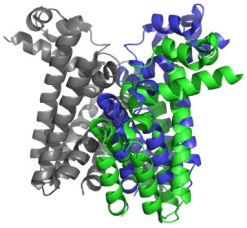
(c) Kompleks `1VC1`:
0,81 × 1,93 Å (0,19)



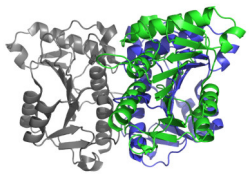
(d) Kompleks `1TLJ`:
0,80 × 2,14 Å (0,20)



(e) Kompleks `1NWP`:
0,88 × 7,12 Å (0,21)

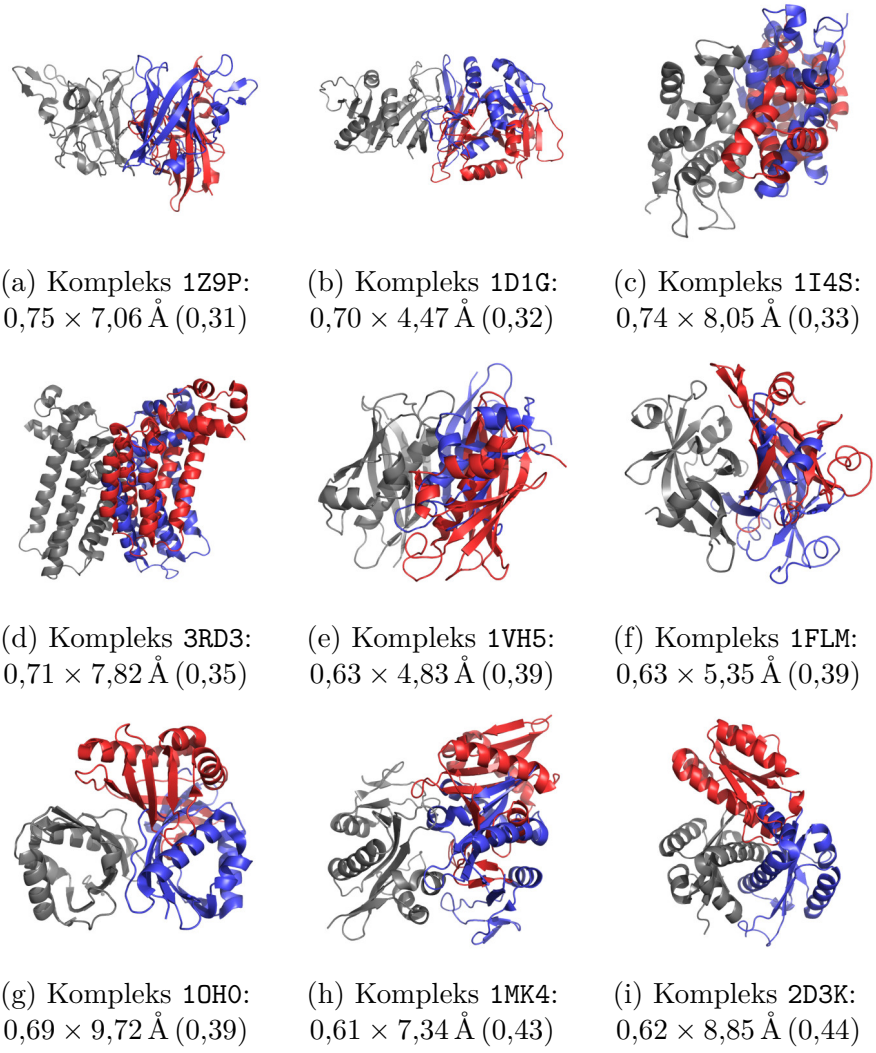


(f) Kompleks `1SGM`:
0,80 × 3,57 Å (0,22)



(g) Kompleks `3AIA`:
0,77 × 4,70 Å (0,26)

Rysunek A.13: Wizualizacja wyników optymalizacji globalnej kryterium pola zewnętrznego, w przypadku których wartość RMSD była mniejsza od 10 Å, a wartość AUC większa od 0,75. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, a z wyników symulacji – zielony. Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC).



(a) Kompleks `1Z9P`: 0,75 × 7,06 Å (0,31)

(b) Kompleks `1D1G`: 0,70 × 4,47 Å (0,32)

(c) Kompleks `1I4S`: 0,74 × 8,05 Å (0,33)

(d) Kompleks `3RD3`: 0,71 × 7,82 Å (0,35)

(e) Kompleks `1VH5`: 0,63 × 4,83 Å (0,39)

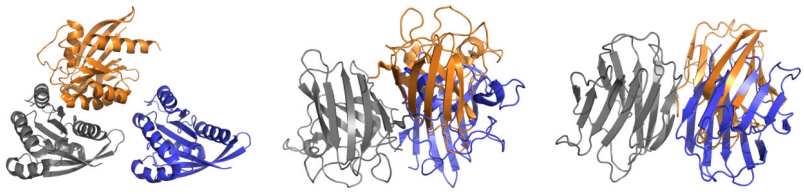
(f) Kompleks `1FLM`: 0,63 × 5,35 Å (0,39)

(g) Kompleks `1OH0`: 0,69 × 9,72 Å (0,39)

(h) Kompleks `1MK4`: 0,61 × 7,34 Å (0,43)

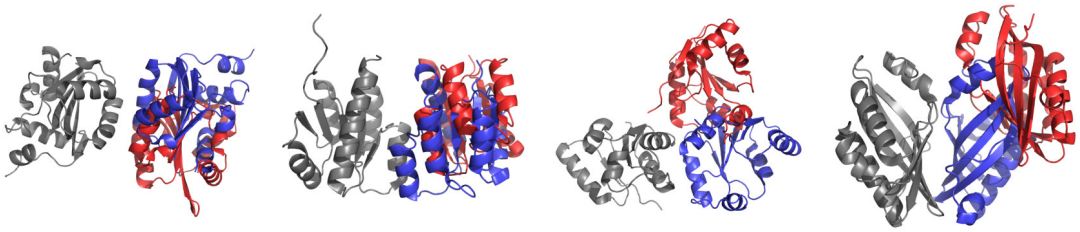
(i) Kompleks `2D3K`: 0,62 × 8,85 Å (0,44)

Rysunek A.14: Wizualizacja wyników optymalizacji globalnej kryterium pola zewnętrznego, w przypadku których wartość RMSD była mniejsza od 10 Å, a wartość AUC mniejsza lub równa 0,75. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, a z wyników symulacji – czerwony. Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC).

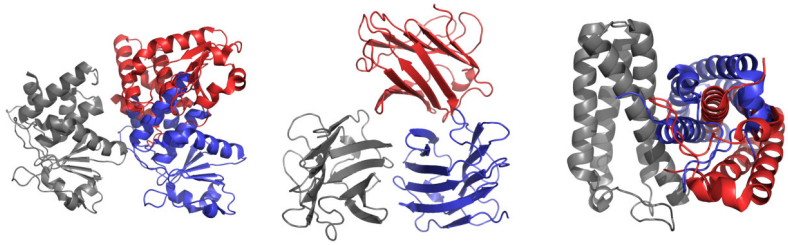


(a) Kompleks `1G17`: 0,93 × 15,02 Å (0,38) (b) Kompleks `2ZOW`: 0,79 × 12,85 Å (0,38) (c) Kompleks `1A78`: 0,78 × 16,88 Å (0,48)

Rysunek A.15: Wizualizacja wyników optymalizacji globalnej kryterium pola zewnętrznego, w przypadku których wartość RMSD była większa lub równa 10 Å, a wartość AUC większa od 0,75. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, a z wyników symulacji – pomarańczowy. Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC).

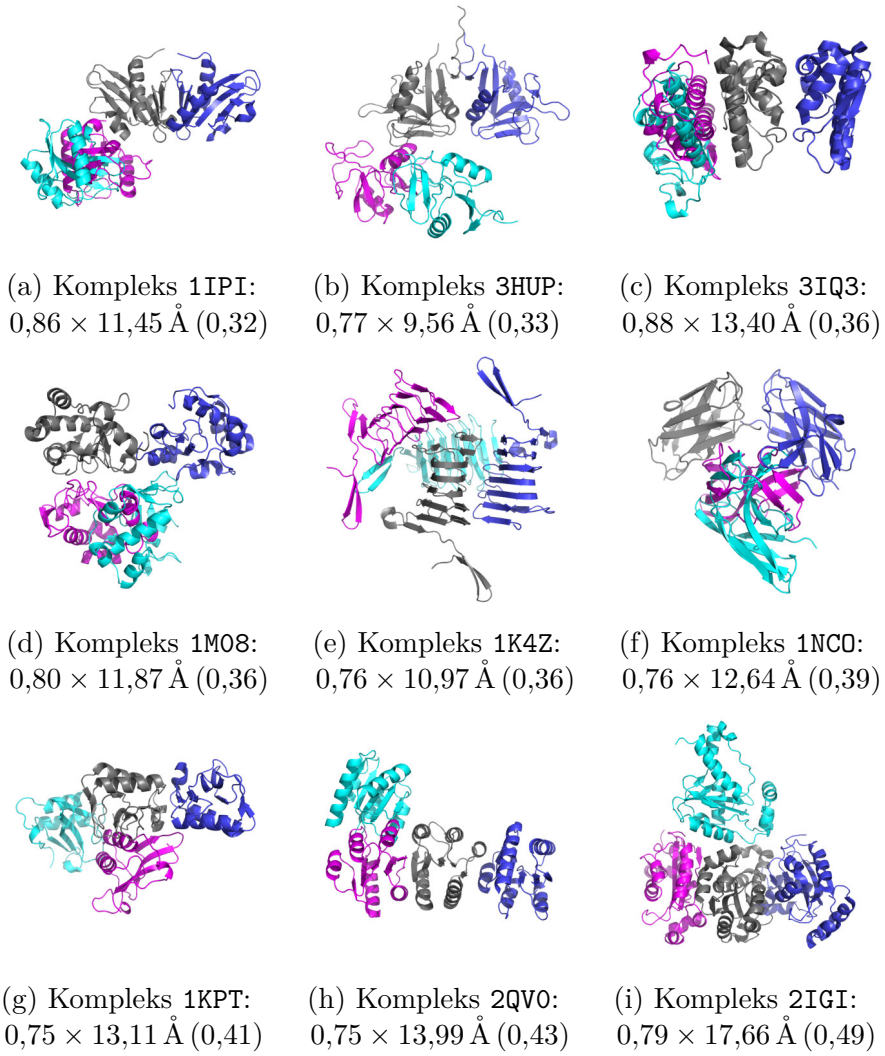


(a) Kompleks `1F08`: 0,66 × 6,62 Å (0,38) (b) Kompleks `3HV2`: 0,63 × 4,14 Å (0,38) (c) Kompleks `3GLV`: 0,61 × 8,07 Å (0,44) (d) Kompleks `1NWW`: 0,56 × 9,09 Å (0,49)



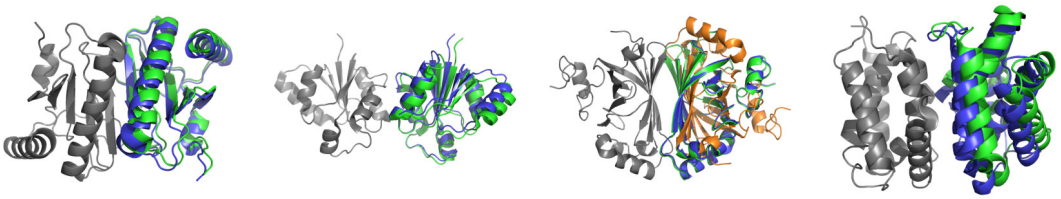
(e) Kompleks `3F81`: 0,55 × 8,52 Å (0,50) (f) Kompleks `1BKZ`: 0,54 × 8,94 Å (0,51) (g) Kompleks `3VRC`: 0,50 × 9,50 Å (0,56)

Rysunek A.16: Wizualizacja wyników optymalizacji globalnej kryterium pola wewnętrznego, w przypadku których wartość RMSD była mniejsza od 10 Å (wartość AUC była zawsze mniejsza lub równa 0,75). Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, a z wyników symulacji – czerwony. Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC).

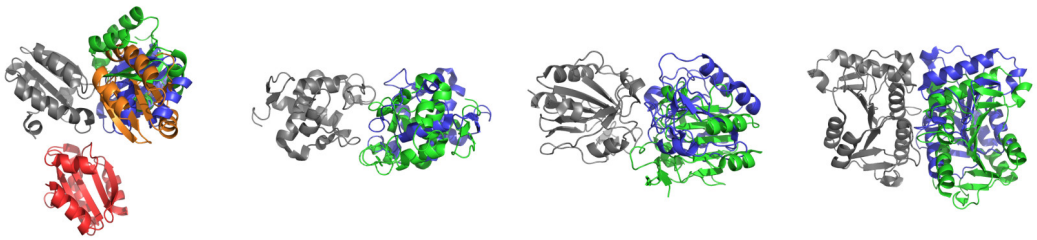


(a) Kompleks `1IPI`: 0,86 × 11,45 Å (0,32)
(b) Kompleks `3HUP`: 0,77 × 9,56 Å (0,33)
(c) Kompleks `3IQ3`: 0,88 × 13,40 Å (0,36)
(d) Kompleks `1M08`: 0,80 × 11,87 Å (0,36)
(e) Kompleks `1K4Z`: 0,76 × 10,97 Å (0,36)
(f) Kompleks `1NCO`: 0,76 × 12,64 Å (0,39)
(g) Kompleks `1KPT`: 0,75 × 13,11 Å (0,41)
(h) Kompleks `2QV0`: 0,75 × 13,99 Å (0,43)
(i) Kompleks `2IGI`: 0,79 × 17,66 Å (0,49)

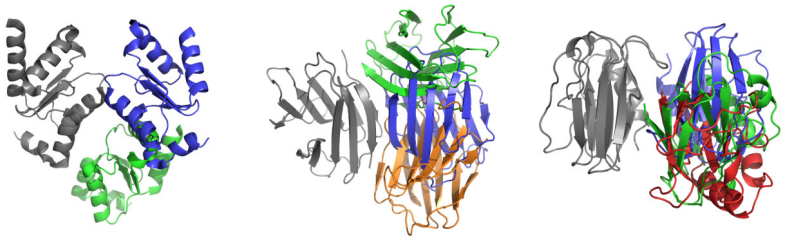
Rysunek A.17: Wizualizacja wyników optymalizacji globalnej kryteriów pól zewnętrznego i wewnętrznego, w przypadku których wartość RMSD drugich względem pierwszych była mniejsza od 10 Å lub wartość AUC większa od 0,75. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, z wyników symulacji – magenta (pole zewnętrzne) i cyjan (pole wewnętrzne). Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC).



(a) Kompleks `3I4S`: 0,99 × 0,50 Å (0,02) (b) Kompleks `3GLV`: 1,00 × 0,73 Å (0,02) (c) Kompleks `2W2A`: 0,97 × 0,22 Å (0,03) (d) Kompleks `1M4R`: 0,84 × 1,89 Å (0,17)

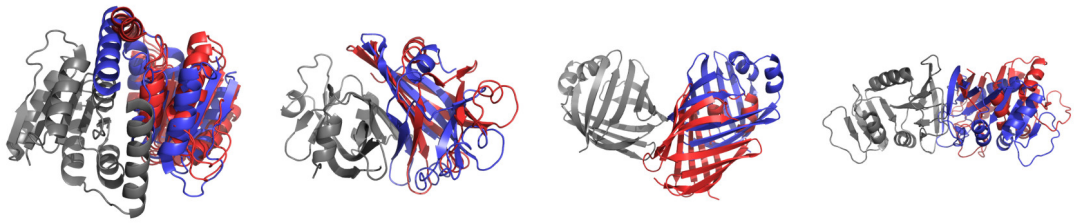


(e) Kompleks `2OE3`: 0,89 × 7,29 Å (0,21) (f) Kompleks `1F1C`: 0,80 × 5,36 Å (0,24) (g) Kompleks `1Q98`: 0,79 × 6,59 Å (0,27) (h) Kompleks `3AIA`: 0,75 × 4,49 Å (0,27)

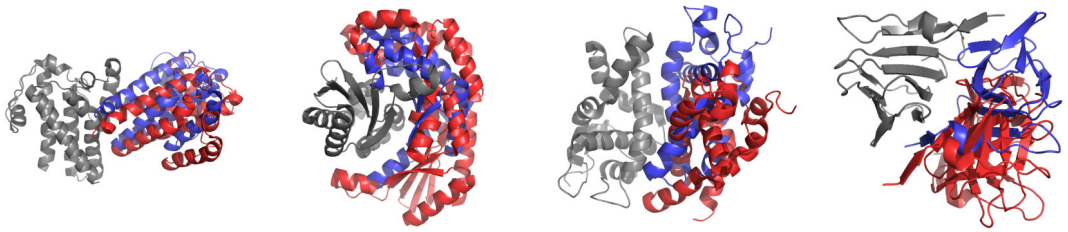


(i) Kompleks `3RHC`: 0,80 × 8,26 Å (0,29) (j) Kompleks `1A78`: 0,78 × 8,24 Å (0,30) (k) Kompleks `1ADW`: 0,75 × 8,38 Å (0,32)

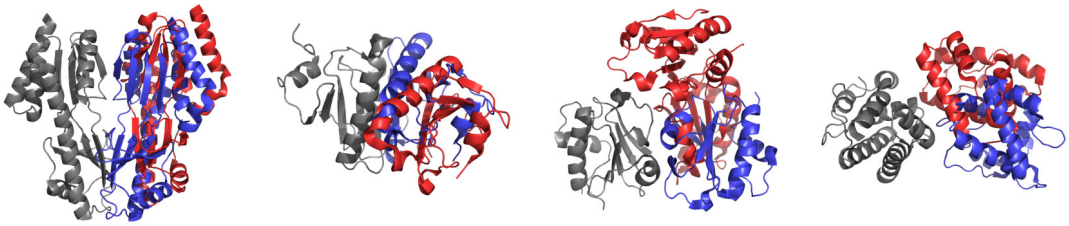
Rysunek A.18: Wizualizacja wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego, wśród których wartość RMSD była mniejsza od 10 Å, a wartość AUC większa od 0,75. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, a z wyników symulacji – zielony (RMSD < 10 Å i AUC $> 0{,}75$), czerwony (tylko RMSD < 10 Å) i pomarańczowy (tylko AUC $> 0{,}75$). Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC). W przypadku obecności na danym rysunku więcej niż jednej wynikowej konformacji kompleksu, podpis pod nim dotyczy tej o najniższej wartości ARC.



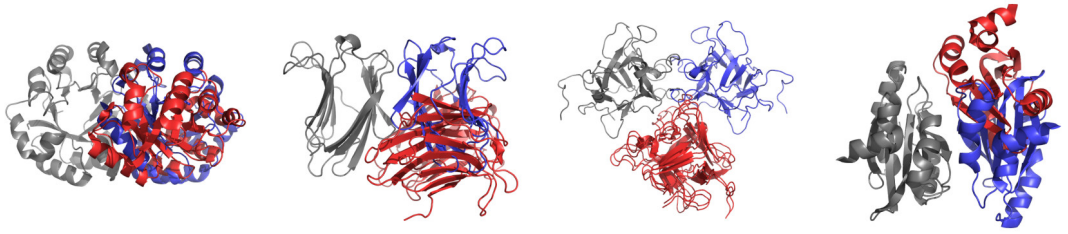
(a) Kompleks `3PH4`: 0,74 × 4,57 Å (0,28)
(b) Kompleks `1FLM`: 0,71 × 3,11 Å (0,30)
(c) Kompleks `1OPA`: 0,74 × 6,80 Å (0,31)
(d) Kompleks `1D1G`: 0,71 × 4,85 Å (0,32)



(e) Kompleks `2XOL`: 0,69 × 2,72 Å (0,32)
(f) Kompleks `1J3M`: 0,68 × 5,01 Å (0,35)
(g) Kompleks `1I4S`: 0,67 × 6,38 Å (0,36)
(h) Kompleks `2DCT`: 0,65 × 5,28 Å (0,38)

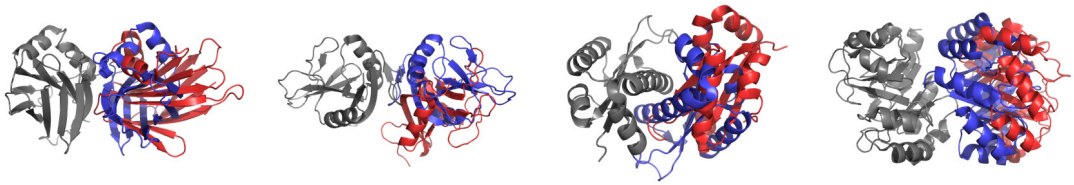


(i) Kompleks `1TLJ`: 0,63 × 4,32 Å (0,38)
(j) Kompleks `3TW2`: 0,64 × 5,09 Å (0,38)
(k) Kompleks `3EVI`: 0,66 × 7,23 Å (0,39)
(l) Kompleks `1DQE`: 0,68 × 8,92 Å (0,39)

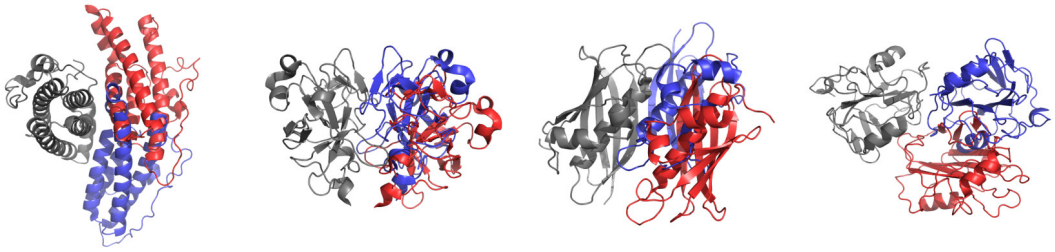


(m) Kompleks `4DF0`: 0,63 × 6,20 Å (0,40)
(n) Kompleks `1BKZ`: 0,65 × 8,35 Å (0,40)
(o) Kompleks `3IIR`: 0,66 × 9,25 Å (0,41)
(p) Kompleks `1M4J`: 0,65 × 8,50 Å (0,41)

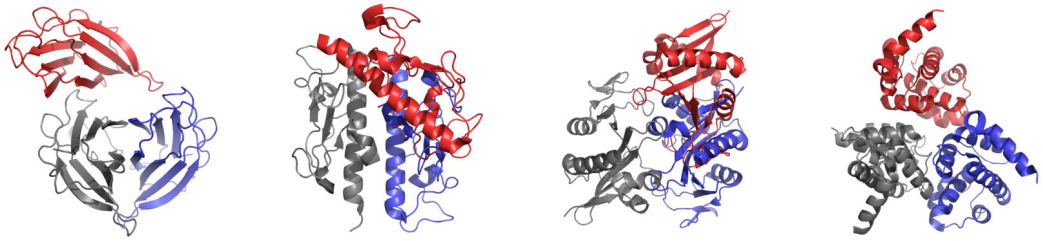
Rysunek A.19: Część 1 wizualizacji wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego, wśród których wartość RMSD była mniejsza od 10 Å, a wartość AUC mniejsza lub równa 0,75. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, a z wyników symulacji – czerwony. Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC).



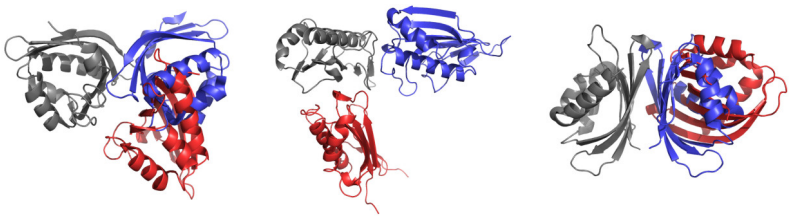
(a) Kompleks 2OFC: 0,65 × 9,26 Å (0,42) (b) Kompleks 2BPD: 0,65 × 9,60 Å (0,42) (c) Kompleks 2D3K: 0,64 × 9,12 Å (0,42) (d) Kompleks 1V5X: 0,61 × 6,80 Å (0,43)



(e) Kompleks 1CO2: 0,64 × 9,65 Å (0,44) (f) Kompleks 3FOU: 0,57 × 4,43 Å (0,45) (g) Kompleks 1VH5: 0,57 × 5,55 Å (0,45) (h) Kompleks 1BD9: 0,60 × 8,83 Å (0,45)

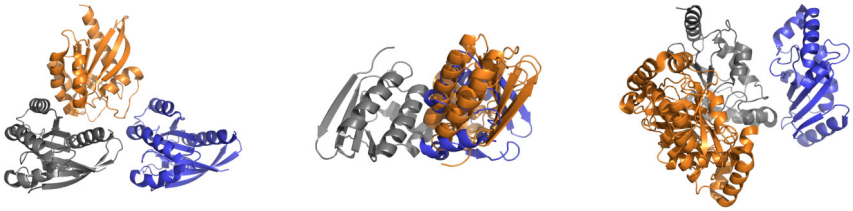


(i) Kompleks 1TFP: 0,61 × 9,46 Å (0,45) (j) Kompleks 3FU1: 0,59 × 8,01 Å (0,46) (k) Kompleks 1MK4: 0,59 × 7,81 Å (0,46) (l) Kompleks 1NP8: 0,60 × 9,05 Å (0,46)



(m) Kompleks 2Z76: 0,59 × 9,33 Å (0,47) (n) Kompleks 1QAH: 0,59 × 9,54 Å (0,48) (o) Kompleks 3UJM: 0,57 × 8,48 Å (0,48)

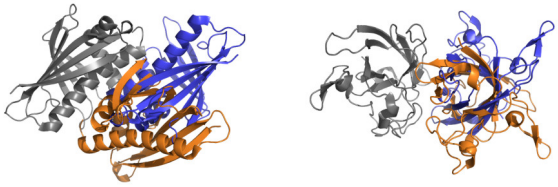
Rysunek A.20: Część 2 wizualizacji wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego, wśród których wartość RMSD była mniejsza od 10 Å, a wartość AUC mniejsza lub równa 0,75. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, a z wyników symulacji – czerwony. Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC).



(a) Kompleks `1G17`: 0,97 × 12,40 Å (0,31)

(b) Kompleks `1VC1`: 0,80 × 10,14 Å (0,32)

(c) Kompleks `2Z5D`: 0,77 × 12,14 Å (0,38)



(d) Kompleks `1IFV`: 0,77 × 13,28 Å (0,40)

(e) Kompleks `1Z9P`: 0,81 × 15,85 Å (0,44)

Rysunek A.21: Wizualizacja wyników optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego, wśród których wartość RMSD była większa lub równa 10 Å, a wartość AUC większa od 0,75. Modele są przedstawione w konformacji nałożenia na siebie ich łańcuchów A, widocznych w kolorze szarym. Łańcuchy B ze struktur natywnych mają kolor niebieski, a z wyników symulacji – pomarańczowy. Format wartości w podpisach pod rysunkami jest następujący: AUC × RMSD (ARC).

B. Tabele

Dodatek B zawiera tabele danych dotyczących białek z bazy danych rozprawy. Tabela B.1 przedstawia wyniki ich analizy, natomiast tabele B.2 i B.3 – wyniki eksperymentu przewidywania ich struktury czwartorzędowej, polegającego na optymalizacji globalnej i wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego.

B.1. Analiza białek z bazy danych

Tabela B.1: Analiza białek z bazy danych. Wyjaśnienie nagłówków kolumn: *PDB ID* – identyfikator struktury w bazie PDB; *Interfejs* – wartości miar ICF oraz ITF interfejsu kontaktów niewiążących pomiędzy łańcuchami; *Pole zewnętrzne* – wartości RD modelu FOD dla kompleksu (C) i łańcuchów (A i B); *Pole wewnętrzne* – wartości energii potencjałów pola ECEPP/3; *Rodzaj cząsteczki (pełniona funkcja)* – rekord COMPND z pliku PDB. Podkreślenie wskazuje na wartości RD mniejsze od 0,5.

PDB	**Interfejs**		**Pole zewnętrzne**			**Pole wewnętrzne**			**Rodzaj cząsteczki**
ID	**ICF**	**ITF**	**C**	**A**	**B**	$\mathbf{E_n}$	$\mathbf{E_n}$	$\mathbf{E_h}$	**(pełniona funkcja)**
137L	0,00	0,08	0,700	0,552	0,541	143,965	-44,523	-7,140	t4 lysozyme
1A25	0,94	0,12	0,591	0,445	0,451	332,909	-15,040	-8,358	protein kinase c (beta)
1A78	1,00	0,09	0,652	0,472	0,472	34,543	-9,858	-7,608	galectin-1
1ADW	1,00	0,07	0,619	0,379	0,379	-209,175	-36,923	-4,642	pseudoazurin
1AG9	1,00	0,06	0,596	0,326	0,321	314,254	-36,354	-12,942	flavodoxin
1AI9	0,25	0,05	0,646	0,429	0,438	-25,822	-14,672	-8,099	dihydrofolate reductase
1ATL	1,00	0,03	0,704	0,464	0,459	-38,843	-9,817	-0,384	atrolysin c
1AYO	0,00	0,04	0,684	0,536	0,545	-90,985	-15,251	-3,977	alpha-2-macroglobulin
1B78	0,94	0,09	0,643	0,540	0,547	97,734	-78,558	-5,723	pyrophosphatase
1B88	0,88	0,07	0,577	0,388	0,397	-32,544	23,212	-3,702	t cell receptor v-alpha domain
1BD9	0,73	0,06	0,685	0,440	0,453	32,970	-19,354	-7,169	phosphatidylethanolamine binding protein
1BKZ	0,86	0,11	0,713	0,446	0,450	-191,297	-2,622	-12,967	galectin-7
1BU5	0,00	0,02	0,679	0,383	0,384	75,647	-11,802	-0,706	protein (flavodoxin)
1C02	0,94	0,10	0,596	0,455	0,451	-191,088	-4,105	-4,910	phosphotransferase ypd1p
1C3I	1,00	0,06	0,676	0,439	0,435	-86,111	-56,265	-5,593	stromelysin-1
1C77	0,00	0,08	0,612	0,434	0,441	34,725	58,799	-4,834	staphylokinase
1CBK	0,94	0,12	0,689	0,442	0,439	116,846	-76,548	-12,196	protein (7,8-dihydro-6-hydroxymethylpterin- pyrophosphokinase)
1COZ	1,00	0,14	0,555	0,341	0,341	70,596	-90,170	-13,215	protein (glycerol-3-phosphate cytidylyltransferase)
1CSG	0,93	0,12	0,743	0,526	0,524	-19,654	-74,441	-5,161	granulocyte colony-stimulating factor receptor

PDB	Interfejs		Pole zewnętrzne			Pole wewnętrzne			Rodzaj cząsteczki
ID	ICF	ITF	C	A	B	E_n	E_n	E_h	(pełniona funkcja)
1DOQ	0,09	0,12	0,619	0,509	0,494	157,697	-21,588	-6,272	dna primase
1D1G	0,97	0,19	0,564	0,400	0,419	-30,543	-72,654	-18,071	dihydrofolate reductase
1DHF	0,83	0,07	0,666	0,405	0,390	-8,055	-25,559	-6,965	dihydrofolate reductase
1DQE	0,00	0,06	0,679	0,338	0,347	-67,871	-39,039	-4,972	pheromone-binding protein
1DZR	0,93	0,15	0,680	0,467	0,436	381,060	-21,267	-19,096	dtdp-4-dehydrorhamnose 3 5-epimerase
1EAJ	1,00	0,12	0,690	0,514	0,476	-65,121	-55,825	-16,489	coxsackie virus and adenovirus receptor
1EO6	0,00	0,15	0,613	0,467	0,503	10,023	-42,219	-10,506	golgi-associated atpase enhancer of 16 kd
1EX2	1,00	0,06	0,655	0,493	0,487	369,441	-40,269	-14,860	protein maf
1EYV	0,82	0,14	0,686	0,500	0,512	-67,847	-67,039	-6,798	n-utilizing substance protein b homolog
1F08	0,83	0,04	0,712	0,458	0,459	17,046	-26,991	-2,250	replication protein e1
1F1C	1,00	0,07	0,633	0,456	0,455	125,943	67,846	-9,131	cytochrome c549
1F46	0,00	0,10	0,621	0,376	0,379	-125,312	-59,468	-5,855	cell division protein zipa
1FLM	1,00	0,18	0,559	0,461	0,457	208,369	-103,812	-9,394	protein (fmn-binding protein)
1FQT	0,00	0,06	0,699	0,424	0,433	-17,642	-17,097	-10,152	rieske-type ferredoxin of biphenyl dioxygenase
1FTP	0,00	0,06	0,721	0,517	0,506	35,261	-29,521	-1,008	muscle fatty acid binding protein
1G17	0,00	0,01	0,692	0,455	0,466	27,466	-7,878	-0,086	ras-related protein sec4
1G2Q	0,97	0,17	0,641	0,516	0,513	278,743	-148,134	-10,615	adenine phosphoribosyltransferase 1
1GE7	0,80	0,04	0,812	0,678	0,682	-15,078	-17,210	-5,973	peptidyl-lys metalloendopeptidase
1GY6	0,88	0,20	0,505	0,415	0,458	-36,515	-90,445	-10,821	nuclear transport factor 2
1HFY	1,00	0,10	0,597	0,427	0,424	17,063	-67,322	-5,780	alpha-lactalbumin
1HKQ	1,00	0,14	0,577	0,413	0,424	-163,710	-65,829	-13,333	replication protein
1HLC	0,92	0,10	0,695	0,462	0,471	102,981	-40,452	-9,046	human lectin
1HPC	0,40	0,09	0,683	0,529	0,536	-81,341	-30,566	-5,105	h protein of the glycine cleavage system
1I4S	1,00	0,16	0,479	0,417	0,432	72,830	-67,686	-26,465	ribonuclease iii
1I6W	0,00	0,01	0,747	0,443	0,449	24,658	-3,232	-2,477	lipase a
1IAZ	1,00	0,06	0,679	0,435	0,449	80,520	-21,076	-9,680	equinatoxin ii
1IFV	0,86	0,04	0,747	0,672	0,685	32,093	-40,048	-12,391	protein llr18b
1IPI	0,95	0,17	0,546	0,423	0,413	258,847	-78,513	-13,815	holliday junction resolvase
1IQ6	0,93	0,22	0,592	0,598	0,622	-184,656	-128,484	-20,704	(r)-specific enoyl-coa hydratase
1J3M	0,96	0,20	0,538	0,609	0,594	-31,156	-58,301	-10,237	the conserved hypothetical protein tt1751
1J3Q	0,97	0,22	0,558	0,408	0,403	-143,205	-163,585	-33,707	phosphoglucose isomerase
1JR8	0,96	0,22	0,542	0,447	0,499	-105,781	-68,981	-2,674	erv2 protein, mitochondrial
1K4Z	1,00	0,06	0,634	0,482	0,497	9,177	-46,505	-7,839	adenylyl cyclase-associated protein
1KPT	1,00	0,11	0,741	0,579	0,586	292,173	78,787	-4,730	kp4 toxin
1L8D	1,00	0,14	0,686	0,679	0,656	65,935	-27,503	-15,520	dna double-strand break repair rad50 atpase
1LFA	0,20	0,03	0,662	0,412	0,428	-76,632	-15,220	-1,591	cd11a
1MO8	1,00	0,08	0,616	0,380	0,397	60,632	-36,621	-4,127	colicin e7
1M4I	0,95	0,12	0,631	0,528	0,506	12,702	-106,132	-16,234	aminoglycoside 2'-n-acetyltransferase
1M4J	0,92	0,09	0,639	0,430	0,444	-56,654	-27,363	-4,627	a6 gene product
1M4R	0,62	0,15	0,627	0,443	0,453	-291,264	-74,920	-11,499	interleukin-22
1MK4	0,95	0,13	0,714	0,538	0,531	292,167	-96,557	-19,079	hypothetical protein yqjy
1MKA	0,97	0,21	0,601	0,575	0,575	-104,424	-152,627	-22,701	beta-hydroxydecanoyl thiol ester dehydrase
1NA8	0,78	0,12	0,637	0,625	0,582	62,417	-40,468	-1,683	adp-ribosylation factor binding protein gga1;
1NBC	0,75	0,04	0,786	0,642	0,649	46,035	-2,282	-5,387	cellulosomal scaffolding protein a
1NCO	1,00	0,09	0,823	0,607	0,589	-32,262	-8,164	-5,842	holo-neocarzinostatin
1NP8	0,00	0,11	0,623	0,452	0,479	-192,608	-9,024	-3,741	calcium-dependent protease, small subunit
1NWP	0,86	0,06	0,604	0,512	0,511	10,832	-20,438	-0,114	azurin

PDB	Interfejs		Pole zewnętrzne			Pole wewnętrzne			Rodzaj cząsteczki
ID	ICF	ITF	C	A	B	E_n	E_n	E_h	(pełniona funkcja)
1NWW	0,94	0,22	0,683	0,492	0,500	-82,293	-143,360	-23,433	limonene-1,2-epoxide hydrolase
1NXM	1,00	0,12	0,589	0,444	0,440	216,607	-145,682	-19,171	dtdp-6-deoxy-d-xylo-4-hexulose 3,5-epimerase
1OHO	1,00	0,22	0,699	0,480	0,473	-273,101	-107,022	-15,582	steroid delta-isomerase
1OPA	1,00	0,07	0,664	0,465	0,460	-53,689	-26,087	-3,643	cellular retinol binding protein ii
1P6O	1,00	0,20	0,573	0,395	0,415	-174,602	-131,367	-36,938	cytosine deaminase
1PPV	0,88	0,10	0,733	0,608	0,586	78,295	-67,312	-2,694	isopentenyl-diphosphate delta-isomerase
1Q98	0,88	0,10	0,683	0,530	0,554	43,019	-59,065	-5,441	thiol peroxidase
1QAH	0,00	0,08	0,670	0,531	0,559	45,969	-31,331	-7,232	perchloric acid soluble protein
1QSD	0,94	0,16	0,661	0,584	0,585	39,506	-38,322	-14,455	protein (beta-tubulin binding post-chaperonin cofactor)
1QZ8	0,53	0,14	0,657	0,447	0,500	79,215	-49,287	-0,565	polyprotein 1ab
1SGM	0,97	0,17	0,592	0,561	0,569	45,294	0,318	-3,954	putative hth-type transcriptional regulator yxaf
1SH8	0,92	0,17	0,640	0,550	0,580	296,022	-119,438	-20,764	hypothetical protein pa5026
1SQU	0,96	0,16	0,689	0,465	0,481	-4,514	-51,228	-12,182	chex protein
1TFP	0,90	0,17	0,641	0,621	0,581	214,960	-38,438	-14,527	transthyretin
1TLJ	1,00	0,16	0,547	0,538	0,509	218,414	-7,766	-8,215	hypothetical upf0130 protein sso0622
1V5X	0,93	0,14	0,627	0,494	0,526	99,461	-116,852	-13,118	phosphoribosylanthranilate isomerase
1V7L	0,67	0,06	0,588	0,367	0,396	126,604	-25,092	-7,528	3-isopropylmalate dehydratase small subunit
1V8H	0,92	0,13	0,644	0,467	0,511	-41,814	-49,246	-3,107	sulfur oxidation protein soxz
1VC1	0,92	0,11	0,496	0,379	0,376	371,840	-39,890	-1,363	putative anti-sigma factor antagonist tm1442
1VH5	1,00	0,20	0,533	0,462	0,477	133,980	-124,320	-12,895	hypothetical protein ydii
1VJ2	1,00	0,30	0,491	0,353	0,359	-183,517	-178,781	-17,249	novel manganese-containing cupin tm1459
1VLT	0,80	0,12	0,726	0,709	0,648	129,144	19,828	-5,298	aspartate receptor
1WPN	1,00	0,09	0,675	0,450	0,449	72,288	-95,206	-11,945	manganese-dependent inorganic pyrophosphatase
1WWZ	0,44	0,07	0,579	0,365	0,372	-45,842	-35,673	-5,013	hypothetical protein ph1933
1X77	1,00	0,15	0,667	0,510	0,515	-4,403	-80,277	-18,855	conserved hypothetical protein
1XOX	1,00	0,08	0,656	0,461	0,461	-75,403	36,198	-1,826	apoptosis inhibitor survivin
1YAV	0,85	0,20	0,476	0,432	0,450	-12,809	-45,064	-4,477	hypothetical protein bsu14130
1YOC	0,96	0,18	0,651	0,544	0,583	280,484	-132,247	-16,663	hypothetical protein pa1835
1Z3A	0,93	0,17	0,473	0,390	0,390	63,075	69,373	-2,348	trna-specific adenosine deaminase
1Z9M	0,92	0,13	0,645	0,404	0,423	-168,332	-51,322	-8,672	gapa225
1Z9P	1,00	0,09	0,508	0,427	0,419	79,676	-59,995	-7,087	superoxide dismutase [cu-zn]
1ZVF	0,97	0,17	0,653	0,582	0,577	38,671	-91,507	-19,273	3-hydroxyanthranilate 3,4-dioxygenase
2A4N	0,97	0,18	0,592	0,476	0,487	183,802	-120,200	-27,827	aac(6')-ii
2A6P	1,00	0,11	0,747	0,539	0,540	38,365	-16,808	-10,673	possible phosphoglycerate mutase gpm2
2A9S	1,00	0,19	0,645	0,557	0,558	-185,791	-119,231	-26,599	competence/damage-inducible protein cina
2ABO	1,00	0,15	0,620	0,413	0,438	51,302	-130,532	-16,133	yajl
2BPD	1,00	0,11	0,614	0,420	0,419	-24,377	-51,473	-6,061	dectin-1
2CAR	0,94	0,10	0,743	0,622	0,611	-21,254	-99,080	-11,325	inosine triphosphate pyrophosphatase
2CCO	1,00	0,07	0,806	0,686	0,663	-87,906	-40,856	-11,063	acetyl-xylan esterase
2D3K	0,94	0,15	0,398	0,357	0,379	5,055	61,721	-4,527	peptidyl-trna hydrolase
2DC4	1,00	0,10	0,737	0,561	0,563	43,802	-70,744	-9,701	165aa long hypothetical protein
2DCT	0,97	0,31	0,582	0,395	0,401	-24,585	-88,606	-17,632	hypothetical protein ttha0104
2EAV	0,85	0,08	0,687	0,491	0,528	53,797	-23,167	-2,756	peptidoglycan recognition protein-i-beta
2F3G	0,57	0,05	0,678	0,397	0,391	265,678	-14,158	-0,959	glucose-specific phosphocarrier
2FBN	1,00	0,10	0,574	0,456	0,443	86,354	-80,295	-3,512	70 kda peptidylprolyl isomerase, putative

PDB	Interfejs		Pole zewnętrzne			Pole wewnętrzne			Rodzaj cząsteczki
ID	ICF	ITF	C	A	B	E_n	E_n	E_h	(pełniona funkcja)
2FZF	0,85	0,13	0,611	0,479	0,486	-96,734	-3,981	-3,437	hypothetical protein
2GJA	1,00	0,10	0,731	0,483	0,459	237,034	-63,240	-6,418	trna modification gtpase trme
2H29	1,00	0,11	0,602	0,415	0,419	-109,299	-75,888	-18,973	probable nicotinate-nucleotide adenylyltransferase
2HJ3	0,78	0,19	0,510	0,350	0,378	-24,859	-10,792	-2,106	sulfhydryl oxidase erv1p
2IDL	0,95	0,17	0,546	0,424	0,382	94,271	-64,913	-7,088	hypothetical protein
2IGI	1,00	0,16	0,730	0,549	0,523	-24,373	-154,515	-16,118	oligoribonuclease
2J8M	1,00	0,22	0,683	0,532	0,522	-135,230	-110,096	-26,344	acetyltransferase pa4866 from p, aeruginosa
2J96	0,96	0,16	0,738	0,624	0,633	90,407	-118,985	-13,282	phycoerythrocyanin alpha chain
2O7M	1,00	0,13	0,637	0,576	0,572	-209,678	-72,229	-12,527	dna endonuclease i-crei
2OE3	1,00	0,10	0,529	0,406	0,362	-18,414	-35,285	-4,290	thioredoxin-3
2OFC	1,00	0,15	0,643	0,459	0,451	40,546	-87,010	-15,640	sclerotium rolfsii lectin
2OMD	0,96	0,20	0,496	0,521	0,494	395,450	-93,612	-14,831	molybdopterin-converting factor subunit 2
2P5R	1,00	0,09	0,615	0,366	0,356	36,059	-12,055	-17,626	glutathione peroxidase 5
2PBR	0,00	0,02	0,697	0,501	0,491	-60,984	-11,185	-3,608	thymidylate kinase
2Q2O	0,91	0,11	0,758	0,508	0,472	-34,413	-69,406	-5,139	vk1 o18/o8 germline light chain variable domain
2QSQ	0,73	0,14	0,636	0,399	0,430	105,435	-61,689	-7,513	carcinoembryonic antigen-related cell adhesion molecule 5
2QVO	1,00	0,09	0,705	0,446	0,446	-85,277	-42,182	-1,405	protein mrke
2QZT	0,91	0,11	0,686	0,522	0,534	-20,030	-38,617	-2,426	sterol carrier protein 2-like 2
2SPC	0,96	0,44	0,776	0,715	0,719	-219,219	288,349	-24,484	spectrin
2W2A	0,96	0,15	0,531	0,450	0,450	-80,800	-84,601	-17,082	p-coumaric acid decarboxylase
2W31	1,00	0,19	0,590	0,545	0,503	203,973	-26,777	-26,669	globin
2WCU	0,94	0,11	0,575	0,480	0,491	-43,852	-84,963	-2,868	protein fucu homolog
2WLV	0,64	0,10	0,707	0,593	0,597	218,722	-58,628	-4,889	gag polyprotein
2XHF	1,00	0,10	0,635	0,444	0,441	18,996	-79,947	-1,835	peroxiredoxin 5
2XOL	0,00	0,13	0,718	0,501	0,457	-388,531	-91,098	-14,236	chaperone protein ttrd
2YEM	0,93	0,13	0,608	0,498	0,455	-193,595	-58,539	-5,266	bromodomain-containing protein 4
2YVE	0,94	0,19	0,706	0,612	0,616	-378,502	-126,401	-14,050	transcriptional regulator
2Z5D	0,00	0,06	0,638	0,434	0,435	-116,739	-33,385	-4,072	ubiquitin-conjugating enzyme e2 h
2Z76	1,00	0,17	0,704	0,494	0,504	39,051	-89,593	-17,234	putative steroid isomerase
2Z9D	1,00	0,11	0,661	0,563	0,558	263,879	-100,465	-12,027	fmn-dependent nadh-azoreductase
2ZB9	0,97	0,20	0,687	0,551	0,580	21,910	-142,584	-27,519	putative transcriptional regulator
2ZGL	0,88	0,11	0,696	0,519	0,509	11,648	-59,806	-5,530	anti-tumor lectin
2ZOW	1,00	0,11	0,577	0,455	0,444	139,416	-65,096	-5,141	superoxide dismutase [cu-zn]
2ZWM	1,00	0,17	0,576	0,307	0,310	-178,018	-102,409	-18,242	transcriptional regulatory protein yycf
3AIA	0,97	0,15	0,528	0,414	0,421	245,197	-92,642	-8,989	upf0217 protein mj1640
3CPQ	1,00	0,08	0,662	0,377	0,381	15,651	-32,893	-3,250	50s ribosomal protein l30e
3CQR	1,00	0,13	0,632	0,472	0,498	40,978	-57,276	-10,647	violaxanthin de-epoxidase, chloroplast
3CT6	0,83	0,19	0,530	0,313	0,426	-43,319	-118,485	-9,391	pts-dependent dihydroxyacetone kinase, phosphotransferase subunit dham
3CXK	1,00	0,07	0,681	0,380	0,392	47,172	-23,102	-0,004	methionine-r-sulfoxide reductase
3D7A	1,00	0,09	0,697	0,450	0,456	-30,715	-15,030	-15,473	upf0201 protein ph1010
3EVI	1,00	0,07	0,664	0,414	0,414	18,581	-35,624	-3,789	phosducin-like protein 2
3F81	0,00	0,04	0,677	0,496	0,491	-120,397	-31,908	-4,581	dual specificity protein phosphatase 3
3FOU	1,00	0,11	0,547	0,478	0,497	-40,739	-76,457	-7,934	quinol-cytochrome c reductase, rieske iron-sulfur subunit
3FQC	1,00	0,06	0,730	0,403	0,402	-64,665	-19,454	-7,782	trimethoprim-sensitive dihydrofolate reductase
3FU1	1,00	0,15	0,635	0,446	0,476	224,125	-70,318	-5,416	general secretion pathway protein g
3G46	1,00	0,11	0,649	0,437	0,451	15,637	-91,566	-11,997	globin-1

PDB	Interfejs		Pole zewnętrzne			Pole wewnętrzne			Rodzaj cząsteczki
ID	ICF	ITF	C	A	B	E_n	E_n	E_h	(pełniona funkcja)
3GLV	0,93	0,12	0,580	0,437	0,456	-172,227	-47,320	-9,400	lipopolysaccharide core biosynthesis protein
3GRN	0,12	0,13	0,548	0,325	0,320	-67,919	-40,380	-9,458	mutt related protein
3GWN	0,96	0,24	0,601	0,494	0,502	-29,359	-124,416	-6,733	probable fad-linked sulfhydryl oxidase r596
3HPE	1,00	0,15	0,718	0,578	0,573	-428,114	-58,862	-11,507	conserved hypothetical secreted protein
3HUP	0,85	0,17	0,576	0,347	0,331	-238,366	-68,533	-13,199	early activation antigen cd69
3HV2	1,00	0,07	0,662	0,410	0,373	-212,471	-50,823	-7,184	response regulator/hd domain protein
3I4S	1,00	0,18	0,559	0,501	0,494	-205,531	-118,169	-22,535	histidine triad protein
3IA1	0,95	0,14	0,593	0,435	0,437	189,147	-74,619	-6,349	thio-disulfide isomerase/thioredoxin
3IIR	1,00	0,08	0,737	0,570	0,575	155,549	-74,606	-3,113	trypsin inhibitor
3IQ3	0,70	0,08	0,552	0,408	0,387	523,537	-8,821	-5,018	phospholipase a2 homolog bothropstoxin-1
3IX3	1,00	0,10	0,611	0,413	0,408	-5,570	-94,393	-8,471	transcriptional activator protein lasr
3K3K	0,67	0,07	0,639	0,528	0,499	19,579	-59,692	-4,789	abscisic acid receptor pyr1
3K9U	1,00	0,03	0,711	0,522	0,518	89,735	-17,050	-2,459	paia acetyltransferase
3L18	1,00	0,09	0,688	0,456	0,453	136,165	-27,238	-15,900	intracellular protease i
3LB2	0,00	0,04	0,774	0,475	0,468	5,296	-18,074	-1,294	dehaloperoxidase a
3LBB	1,00	0,25	0,568	0,455	0,463	41,720	-136,339	-18,679	putative uncharacterized protein smu,793
3LYN	0,94	0,13	0,704	0,584	0,587	-76,108	-65,768	-8,889	sperm lysin
3MGK	1,00	0,15	0,530	0,352	0,365	-173,299	-121,883	-16,982	intracellular protease/amidase related enzyme (thij family)
3N4K	0,87	0,19	0,569	0,387	0,424	210,106	-148,504	-30,332	rna methyltransferase
3N7H	0,92	0,10	0,674	0,427	0,441	253,429	-1,764	-7,886	odorant binding protein
3N8E	0,71	0,04	0,723	0,471	0,475	30,693	-29,664	-0,225	stress-70 protein, mitochondrial
3NBC	1,00	0,09	0,768	0,538	0,535	100,526	-59,632	-2,080	ricin b-like lectin
3OCP	0,46	0,09	0,638	0,425	0,453	-7,882	-36,951	-4,846	prkg1 protein
3P9X	1,00	0,06	0,720	0,438	0,445	0,979	-49,815	-3,365	phosphoribosylglycinamide formyltransferase
3PH4	0,97	0,25	0,476	0,427	0,425	491,022	-146,597	-26,846	ribose-5-phosphate isomerase
3QU1	0,94	0,10	0,747	0,485	0,487	79,503	-87,021	-13,735	peptide deformylase 2
3RD3	0,97	0,17	0,727	0,683	0,665	20,059	-112,586	-8,228	probable transcriptional regulator
3RFB	1,00	0,19	0,625	0,400	0,432	-150,984	8,856	-7,441	putative uncharacterized protein
3RHC	1,00	0,04	0,664	0,398	0,338	-10,156	-12,789	-0,684	glutaredoxin-c5, chloroplastic
3RQ3	0,94	0,15	0,683	0,481	0,530	11,646	-68,718	-9,159	t cell immunoreceptor with ig and itim domains
3SLZ	0,88	0,23	0,627	0,555	0,597	-37,967	-113,993	-16,543	gag-pro-pol polyprotein
3SZJ	0,97	0,30	0,658	0,705	0,605	-76,938	-156,894	-24,203	avidin/streptavidin
3TRF	1,00	0,04	0,649	0,487	0,481	48,852	-21,566	-3,658	shikimate kinase
3TW2	1,00	0,32	0,478	0,393	0,402	-769,854	-98,519	-34,012	histidine triad nucleotide-binding protein 1
3UJM	1,00	0,20	0,604	0,472	0,466	-99,884	-58,960	-24,611	rasputin
3UMZ	0,62	0,09	0,697	0,438	0,462	-80,593	-40,827	-2,811	mediator of dna damage checkpoint protein 1
3V6G	0,94	0,20	0,709	0,648	0,636	-261,669	-80,704	-12,213	probable transcriptional regulatory protein (probably deor family)
3VRC	1,00	0,13	0,590	0,562	0,541	-97,913	-82,254	-1,225	cytochrome c'
4AUU	0,08	0,08	0,786	0,560	0,565	86,776	-47,422	-3,225	fimh
4DF0	1,00	0,14	0,652	0,484	0,491	-441,332	-149,531	-19,586	orotidine 5'-phosphate decarboxylase
4E7P	1,00	0,05	0,689	0,363	0,357	15,386	-5,445	-1,207	response regulator
4EC7	0,95	0,22	0,594	0,587	0,588	22,574	-75,230	-13,564	venom nerve growth factor
4EP4	0,77	0,14	0,646	0,363	0,406	16,913	-121,101	-6,307	crossover junction endodeoxyribonuclease ruvc

B.2. Kompleksowanie białek – jednokryterialne

Tabela B.2: Wyniki eksperymentu przewidywania struktury czwartorzędowej białek, polegającego na optymalizacji globalnej kryteriów pól zewnętrznego i wewnętrznego przy pomocy algorytmu PSO. Wyjaśnienie nagłówków kolumn: *PDB ID* – identyfikator struktury w bazie PDB; *Pole zewnętrzne* – wartość RD modelu FOD, wartość RMSD oraz wartości AUC dla kompleksu (C) i łańcuchów (A, B); *Pole wewnętrzne* – jak wyżej, z tą różnicą, że w miejscu wartości RD znajduje się wartość energii pola ECEPP/3. W ostatnich kolumnach wyników dla każdego pola (*energia* i *RD*) znajdują się odpowiadające im wartości kryterium drugiego pola. W przypadku pola zewnętrznego, przedstawiona jest wyłącznie suma energii potencjałów elektrostatycznego i wiązań wodorowych. Podkreślone liczby wskazują na wartości RD mniejsze od 0,5, wartości RMSD mniejsze od 10 Å, lub wartości AUC większe od 0,75.

PDB	**Pole zewnętrzne**						**Pole wewnętrzne**					
ID	**RD**	**RMSD**	**C**	**A**	**B**	**Energia**	**Energia**	**RMSD**	**C**	**A**	**B**	**RD**
137L	0,597	20,754	0,484	0,437	0,527	354,025	-160,348	17,278	0,518	0,542	0,497	0,748
1A25	0,503	20,407	0,449	0,471	0,427	-10,615	-322,973	17,215	0,544	0,480	0,608	0,706
1A78	0,609	16,882	0,778	0,892	0,663	141,688	-295,457	14,835	0,496	0,480	0,513	0,709
1ADW	0,551	10,876	0,624	0,417	0,832	3,074	-301,772	16,818	0,480	0,483	0,478	0,679
1AG9	0,524	16,785	0,565	0,454	0,676	165,016	-392,396	18,843	0,477	0,479	0,476	0,630
1AI9	0,557	14,713	0,462	0,465	0,459	148,886	-260,857	22,082	0,485	0,484	0,486	0,663
1ATL	0,613	21,055	0,461	0,474	0,448	125,757	-266,995	18,601	0,494	0,495	0,492	0,692
1AYO	0,579	16,073	0,522	0,424	0,620	61,883	-297,565	17,165	0,526	0,468	0,584	0,775
1B78	0,560	21,548	0,620	0,525	0,705	30,022	-289,737	23,298	0,482	0,482	0,482	0,709
1B88	0,486	17,802	0,398	0,400	0,396	-26,234	-344,416	12,058	0,545	0,486	0,611	0,617
1BD9	0,578	19,382	0,506	0,456	0,556	35,384	-201,637	16,217	0,552	0,619	0,486	0,708
1BKZ	0,598	10,088	0,650	0,713	0,592	-40,369	-240,516	8,943	0,539	0,519	0,558	0,682
1BU5	0,562	13,358	0,453	0,451	0,455	210,898	-219,077	20,910	0,490	0,490	0,490	0,704
1CO2	0,498	15,669	0,713	0,581	0,845	73,116	-272,949	13,808	0,598	0,549	0,647	0,627
1C3I	0,569	15,813	0,665	0,443	0,907	33,929	-279,193	14,952	0,489	0,494	0,485	0,762
1C77	0,435	19,739	0,568	0,669	0,458	-93,571	-401,756	14,812	0,533	0,615	0,445	0,623
1CBK	0,519	21,325	0,443	0,451	0,436	34,585	-188,888	19,884	0,505	0,517	0,493	0,767
1COZ	0,500	18,628	0,487	0,445	0,529	-42,263	-390,675	13,025	0,563	0,667	0,459	0,582
1CSG	0,576	21,509	0,413	0,401	0,425	96,471	-312,927	19,215	0,479	0,481	0,476	0,673
1DOQ	0,411	17,513	0,673	0,847	0,465	77,690	-254,235	15,525	0,543	0,519	0,574	0,693
1D1G	0,541	4,466	0,702	0,666	0,738	-40,803	-277,320	14,177	0,523	0,565	0,481	0,620
1DHF	0,533	20,936	0,451	0,450	0,453	-98,359	-249,550	22,201	0,494	0,497	0,491	0,633
1DQE	0,559	12,356	0,522	0,545	0,495	148,977	-282,339	17,707	0,486	0,484	0,488	0,692
1DZR	0,525	23,095	0,435	0,422	0,449	-70,289	-289,026	17,852	0,546	0,484	0,611	0,680
1EAJ	0,568	17,827	0,419	0,423	0,415	-58,176	-214,407	18,918	0,488	0,486	0,491	0,685
1EO6	0,507	17,022	0,424	0,390	0,452	75,793	-257,255	17,739	0,470	0,465	0,474	0,703
1EX2	0,590	25,341	0,457	0,454	0,460	201,086	-286,029	22,122	0,487	0,489	0,486	0,639
1EYV	0,590	15,888	0,445	0,456	0,434	-53,319	-271,988	19,533	0,522	0,575	0,474	0,733
1FO8	0,565	15,170	0,548	0,635	0,460	-49,767	-229,123	6,625	0,656	0,743	0,569	0,753
1F1C	0,590	18,645	0,438	0,400	0,476	339,323	-390,572	16,657	0,469	0,471	0,467	0,705
1F46	0,478	14,575	0,579	0,416	0,753	49,237	-259,874	19,123	0,536	0,472	0,604	0,692
1FLM	0,532	5,351	0,635	0,607	0,662	304,912	-176,236	18,802	0,465	0,470	0,460	0,747
1FQT	0,575	13,795	0,503	0,520	0,488	-3,260	-339,834	17,174	0,487	0,451	0,513	0,695
1FTP	0,592	15,377	0,617	0,768	0,444	-56,535	-182,899	11,273	0,717	0,929	0,476	0,637
1G17	0,558	15,021	0,927	0,925	0,928	-50,092	-303,060	14,755	0,491	0,491	0,491	0,701
1G2Q	0,553	13,714	0,631	0,556	0,707	20,903	-324,013	20,640	0,503	0,493	0,513	0,718
1GE7	0,656	18,338	0,439	0,441	0,438	54,719	-144,776	19,991	0,484	0,485	0,484	0,842
1GY6	0,458	17,107	0,603	0,452	0,749	37,997	-168,191	21,544	0,477	0,475	0,480	0,708
1HFY	0,518	14,343	0,668	0,875	0,445	132,523	-266,116	15,983	0,468	0,463	0,472	0,660

PDB	Pole zewnętrzne						Pole wewnętrzne					
ID	**RD**	**RMSD**	**C**	**A**	**B**	**Energia**	**Energia**	**RMSD**	**C**	**A**	**B**	**RD**
1HKQ	0,414	20,835	0,424	0,425	0,422	35,654	-321,033	16,848	0,570	0,565	0,575	0,584
1HLC	0,557	15,502	0,446	0,456	0,435	200,802	-360,043	14,559	0,503	0,526	0,479	0,760
1HPC	0,585	14,400	0,581	0,763	0,449	46,436	-247,480	16,689	0,468	0,467	0,470	0,735
1I4S	0,456	8,048	0,742	0,759	0,726	33,446	-674,565	17,354	0,499	0,455	0,543	0,654
1I6W	0,550	21,041	0,459	0,460	0,458	1,706	-195,446	18,240	0,483	0,480	0,486	0,732
1IAZ	0,568	15,505	0,658	0,566	0,733	-89,431	-280,364	17,780	0,570	0,695	0,466	0,740
1IFV	0,664	17,449	0,538	0,429	0,647	44,691	-245,826	20,727	0,478	0,483	0,473	0,764
1IPI	0,413	17,685	0,430	0,435	0,426	50,913	-109,528	18,156	0,484	0,484	0,484	0,555
1IQ6	0,562	17,329	0,574	0,604	0,541	-38,130	-280,834	21,744	0,502	0,470	0,537	0,745
1J3M	0,508	16,574	0,603	0,640	0,566	125,345	-241,758	19,587	0,502	0,525	0,480	0,726
1J3Q	0,484	22,151	0,526	0,462	0,592	-53,814	-367,964	23,139	0,480	0,481	0,479	0,593
1JR8	0,453	15,275	0,592	0,360	0,817	-46,965	-344,021	19,017	0,466	0,470	0,463	0,651
1K4Z	0,483	17,727	0,619	0,463	0,759	-67,530	-366,177	15,367	0,483	0,483	0,483	0,656
1KPT	0,639	13,477	0,480	0,402	0,565	251,639	-117,213	18,259	0,489	0,489	0,489	0,777
1L8D	0,542	20,003	0,641	0,667	0,616	57,384	-419,677	20,015	0,469	0,472	0,466	0,738
1LFA	0,483	16,108	0,516	0,578	0,463	57,953	-325,042	17,557	0,530	0,480	0,572	0,721
1MO8	0,501	16,123	0,542	0,521	0,561	-56,076	-219,155	14,658	0,517	0,587	0,458	0,588
1M4I	0,592	14,888	0,609	0,474	0,757	-103,587	-466,288	16,968	0,546	0,596	0,492	0,712
1M4J	0,510	14,611	0,645	0,909	0,400	-55,491	-338,869	12,577	0,557	0,559	0,556	0,676
1M4R	0,586	14,813	0,618	0,797	0,453	-19,648	-273,439	13,838	0,585	0,511	0,652	0,696
1MK4	0,614	7,339	0,612	0,681	0,544	153,623	-269,442	20,480	0,456	0,449	0,463	0,737
1MKA	0,613	14,509	0,601	0,753	0,452	134,267	-222,133	18,732	0,518	0,478	0,556	0,673
1NA8	0,589	14,731	0,676	0,733	0,620	-57,296	-301,193	21,309	0,507	0,477	0,536	0,726
1NBC	0,649	19,122	0,441	0,439	0,444	147,244	-155,330	14,346	0,575	0,486	0,747	0,801
1NCO	0,679	14,075	0,506	0,532	0,482	86,872	-194,826	15,007	0,473	0,466	0,480	0,747
1NP8	0,486	12,363	0,606	0,417	0,794	-194,108	-390,311	20,030	0,470	0,462	0,477	0,645
1NWP	0,568	7,119	0,881	0,958	0,795	49,265	-110,675	14,862	0,556	0,492	0,630	0,652
1NWW	0,596	20,538	0,436	0,430	0,443	365,528	-257,599	9,090	0,563	0,548	0,578	0,754
1NXM	0,527	20,441	0,444	0,441	0,447	154,849	-167,618	23,221	0,488	0,491	0,485	0,701
1OH0	0,592	9,715	0,689	0,581	0,794	-145,116	-229,701	17,326	0,472	0,474	0,470	0,752
1OPA	0,601	16,248	0,741	0,912	0,571	-34,095	-249,596	16,387	0,472	0,468	0,476	0,694
1P6O	0,471	14,792	0,478	0,440	0,515	-19,146	-319,203	18,437	0,508	0,524	0,492	0,695
1PPV	0,671	15,524	0,631	0,753	0,522	-94,713	-247,685	18,870	0,484	0,484	0,485	0,761
1Q98	0,649	11,536	0,730	0,917	0,543	-56,329	-282,607	19,269	0,481	0,483	0,480	0,783
1QAH	0,565	17,498	0,530	0,447	0,613	36,900	-179,021	18,479	0,500	0,471	0,530	0,737
1QSD	0,531	27,389	0,487	0,424	0,547	13,333	-425,115	12,674	0,477	0,453	0,500	0,704
1QZ8	0,488	17,953	0,467	0,401	0,529	-34,641	-238,680	16,551	0,492	0,474	0,510	0,643
1SGM	0,545	3,567	0,797	0,780	0,814	87,141	-218,078	21,028	0,489	0,487	0,490	0,707
1SH8	0,559	19,839	0,537	0,398	0,669	88,965	-295,572	17,578	0,484	0,480	0,488	0,680
1SQU	0,581	16,543	0,550	0,616	0,487	-46,494	-187,607	18,457	0,506	0,538	0,477	0,677
1TFP	0,570	10,360	0,584	0,420	0,748	22,150	-228,112	14,630	0,481	0,484	0,479	0,757
1TLJ	0,504	2,136	0,803	0,777	0,831	312,466	-316,525	23,397	0,484	0,481	0,487	0,740
1V5X	0,632	17,037	0,680	0,684	0,676	152,187	-282,110	21,723	0,490	0,485	0,494	0,766
1V7L	0,465	14,925	0,571	0,552	0,587	168,820	-341,886	20,330	0,474	0,467	0,480	0,621
1V8H	0,550	20,135	0,427	0,424	0,430	5,659	-255,048	13,349	0,529	0,585	0,468	0,691
1VC1	0,410	1,926	0,814	0,819	0,809	265,688	-320,458	16,446	0,534	0,516	0,553	0,647
1VH5	0,503	4,826	0,633	0,635	0,631	152,379	-319,626	17,129	0,570	0,473	0,667	0,686
1VJ2	0,513	15,476	0,555	0,429	0,685	30,981	-280,905	18,188	0,485	0,468	0,503	0,687
1VLT	0,646	12,744	0,513	0,533	0,496	8,005	-237,683	17,220	0,545	0,629	0,476	0,776
1WPN	0,585	20,733	0,444	0,438	0,450	255,378	-314,532	20,534	0,499	0,482	0,515	0,624
1WWZ	0,463	21,576	0,418	0,412	0,424	34,121	-375,544	19,100	0,551	0,653	0,472	0,612
1X77	0,601	12,436	0,641	0,793	0,493	38,536	-209,849	13,820	0,506	0,477	0,535	0,735
1XOX	0,499	17,738	0,477	0,407	0,546	-4,940	-422,245	17,653	0,465	0,463	0,468	0,640
1YAV	0,461	20,162	0,557	0,681	0,431	55,486	-397,584	12,091	0,510	0,500	0,520	0,643
1YOC	0,625	16,328	0,579	0,407	0,755	116,407	-183,945	19,533	0,490	0,487	0,492	0,752
1Z3A	0,407	13,648	0,601	0,415	0,788	-1,711	-313,258	17,222	0,518	0,481	0,556	0,619
1Z9M	0,429	16,893	0,398	0,376	0,423	69,047	-247,123	13,626	0,536	0,472	0,614	0,687
1Z9P	0,448	7,063	0,745	0,765	0,725	-28,802	-264,874	21,371	0,477	0,475	0,479	0,677

PDB	Pole zewnętrzne						Pole wewnętrzne					
ID	**RD**	**RMSD**	**C**	**A**	**B**	**Energia**	**Energia**	**RMSD**	**C**	**A**	**B**	**RD**
1ZVF	0,565	21,396	0,541	0,592	0,486	79,912	-211,136	21,611	0,481	0,476	0,486	0,726
2A4N	0,562	18,559	0,573	0,454	0,692	-64,030	-440,309	17,371	0,476	0,473	0,479	0,710
2A6P	0,605	18,335	0,577	0,700	0,453	-55,423	-219,857	13,110	0,530	0,488	0,571	0,727
2A9S	0,607	17,204	0,708	0,776	0,642	213,935	-213,449	17,683	0,497	0,505	0,489	0,733
2AB0	0,541	17,350	0,546	0,464	0,629	52,731	-271,644	20,324	0,488	0,488	0,488	0,661
2BPD	0,543	18,595	0,431	0,471	0,391	-31,019	-286,079	16,226	0,471	0,464	0,478	0,759
2CAR	0,620	18,203	0,448	0,461	0,435	58,506	-252,934	19,451	0,486	0,489	0,483	0,725
2CC0	0,700	21,324	0,449	0,427	0,475	-74,162	-220,114	14,206	0,512	0,475	0,556	0,839
2D3K	0,373	8,852	0,624	0,503	0,746	9,734	-265,322	17,985	0,475	0,470	0,480	0,657
2DC4	0,602	17,038	0,597	0,612	0,581	140,870	-184,936	20,168	0,502	0,518	0,486	0,735
2DCT	0,338	10,554	0,670	0,678	0,662	92,213	-327,191	16,398	0,475	0,502	0,447	0,630
2EAV	0,635	10,437	0,686	0,572	0,791	91,144	-254,607	18,874	0,493	0,490	0,497	0,771
2F3G	0,574	15,644	0,678	0,941	0,416	132,894	-277,541	19,055	0,490	0,490	0,490	0,697
2FBN	0,470	24,246	0,449	0,453	0,445	116,379	-327,635	19,491	0,492	0,514	0,471	0,624
2FZF	0,531	19,659	0,434	0,447	0,420	-80,974	-290,591	20,107	0,484	0,482	0,486	0,659
2GJA	0,512	20,529	0,450	0,457	0,441	166,811	-353,346	15,546	0,541	0,605	0,476	0,668
2H29	0,572	13,689	0,615	0,801	0,429	-8,918	-277,231	19,400	0,484	0,485	0,482	0,670
2HJ3	0,425	13,531	0,573	0,769	0,360	85,070	-429,868	16,829	0,486	0,444	0,531	0,612
2IDL	0,417	12,049	0,669	0,657	0,681	63,457	-307,023	15,042	0,500	0,552	0,448	0,649
2IGI	0,564	23,165	0,445	0,436	0,454	-85,025	-288,175	20,602	0,490	0,487	0,493	0,646
2J8M	0,625	22,244	0,435	0,445	0,425	177,683	-275,447	18,161	0,489	0,474	0,504	0,744
2J96	0,631	20,388	0,508	0,435	0,574	209,039	-175,660	19,255	0,483	0,489	0,478	0,762
2O7M	0,585	13,336	0,666	0,745	0,583	-85,493	-251,254	21,089	0,479	0,481	0,478	0,663
2OE3	0,446	7,331	0,943	0,914	0,968	78,063	-192,462	15,907	0,466	0,459	0,473	0,667
2OFC	0,566	15,575	0,743	0,920	0,566	47,640	-245,463	16,464	0,606	0,450	0,762	0,721
2OMD	0,486	15,472	0,609	0,646	0,571	60,157	-231,749	26,433	0,507	0,525	0,487	0,733
2P5R	0,527	18,797	0,436	0,435	0,438	-124,398	-274,953	22,099	0,483	0,483	0,484	0,642
2PBR	0,582	24,897	0,467	0,468	0,466	115,170	-369,840	18,719	0,487	0,484	0,490	0,648
2Q20	0,491	18,347	0,422	0,421	0,423	41,981	-178,107	14,134	0,503	0,479	0,530	0,749
2QSQ	0,532	13,118	0,579	0,717	0,427	50,934	-133,365	13,400	0,554	0,473	0,645	0,663
2QV0	0,556	20,393	0,382	0,378	0,386	121,405	-272,723	19,482	0,471	0,468	0,473	0,708
2QZT	0,588	14,419	0,528	0,484	0,569	105,871	-211,970	18,954	0,467	0,469	0,464	0,762
2SPC	0,603	41,517	0,491	0,506	0,474	-34,755	-302,412	35,025	0,486	0,502	0,471	0,811
2W2A	0,509	0,941	0,961	0,956	0,966	31,429	-219,766	19,078	0,482	0,480	0,483	0,663
2W31	0,562	15,817	0,605	0,783	0,443	289,443	-335,864	22,370	0,468	0,460	0,475	0,691
2WCU	0,506	15,968	0,686	0,860	0,502	79,458	-308,103	17,366	0,549	0,522	0,579	0,746
2WLV	0,591	18,311	0,526	0,403	0,657	147,326	-217,000	18,439	0,540	0,596	0,481	0,718
2XHF	0,596	20,269	0,441	0,443	0,438	-175,867	-226,214	18,880	0,482	0,486	0,478	0,639
2XOL	0,561	22,644	0,463	0,459	0,467	-28,888	-413,424	18,895	0,580	0,565	0,595	0,751
2YEM	0,477	18,024	0,494	0,483	0,504	-35,064	-322,260	13,409	0,581	0,699	0,464	0,622
2YVE	0,586	25,057	0,503	0,472	0,532	-49,292	-397,414	26,568	0,477	0,476	0,478	0,783
2Z5D	0,530	17,783	0,710	0,428	0,884	111,127	-288,617	10,904	0,509	0,486	0,520	0,674
2Z76	0,636	17,149	0,427	0,417	0,436	117,302	-203,663	16,873	0,520	0,486	0,554	0,723
2Z9D	0,603	21,195	0,510	0,577	0,444	33,532	-195,304	21,767	0,483	0,480	0,486	0,763
2ZB9	0,622	18,926	0,444	0,447	0,441	43,524	-223,456	25,138	0,492	0,490	0,493	0,724
2ZGL	0,632	10,969	0,610	0,718	0,496	20,107	-274,603	19,996	0,451	0,454	0,448	0,727
2ZOW	0,463	12,845	0,792	0,930	0,654	287,043	-193,662	16,372	0,483	0,485	0,481	0,717
2ZWM	0,474	14,281	0,436	0,415	0,457	24,018	-294,251	14,412	0,556	0,650	0,465	0,581
3AIA	0,497	4,695	0,771	0,752	0,790	443,178	-277,632	20,928	0,501	0,491	0,511	0,647
3CPQ	0,453	16,823	0,418	0,423	0,414	57,315	-426,678	15,671	0,465	0,456	0,473	0,576
3CQR	0,547	19,934	0,516	0,453	0,576	18,362	-171,824	20,020	0,508	0,538	0,480	0,636
3CT6	0,507	14,416	0,562	0,429	0,696	105,832	-454,780	15,617	0,490	0,533	0,447	0,655
3CXK	0,543	19,745	0,418	0,415	0,422	11,699	-300,860	17,430	0,482	0,480	0,484	0,599
3D7A	0,509	14,429	0,469	0,429	0,505	145,067	-288,272	16,810	0,578	0,464	0,684	0,602
3EVI	0,525	15,172	0,537	0,547	0,522	60,710	-231,961	10,091	0,607	0,718	0,468	0,708
3F81	0,579	23,412	0,464	0,471	0,456	-43,979	-282,067	8,516	0,553	0,611	0,526	0,686
3FOU	0,534	13,255	0,614	0,691	0,540	44,814	-186,151	20,734	0,493	0,496	0,489	0,711
3FQC	0,535	22,386	0,456	0,459	0,452	-23,121	-285,636	15,545	0,488	0,490	0,486	0,723
3FU1	0,509	14,460	0,417	0,418	0,416	362,360	-337,465	16,913	0,471	0,479	0,463	0,669

PDB	Pole zewnętrzne						Pole wewnętrzne					
ID	**RD**	**RMSD**	**C**	**A**	**B**	**Energia**	**Energia**	**RMSD**	**C**	**A**	**B**	**RD**
3G46	0,602	14,696	0,546	0,408	0,684	371,371	-242,042	11,419	0,582	0,613	0,551	0,743
3GLV	0,529	15,869	0,484	0,542	0,426	133,872	-233,923	8,074	0,609	0,624	0,593	0,697
3GRN	0,442	18,493	0,471	0,492	0,450	7,329	-246,878	12,253	0,561	0,643	0,483	0,626
3GWN	0,542	15,548	0,608	0,424	0,786	9,455	-313,013	21,053	0,474	0,471	0,476	0,726
3HPE	0,585	21,413	0,433	0,410	0,455	-120,167	-312,707	21,857	0,471	0,471	0,471	0,695
3HUP	0,408	19,404	0,434	0,436	0,433	138,767	-339,355	17,746	0,450	0,431	0,469	0,563
3HV2	0,445	17,216	0,440	0,480	0,408	-18,746	-235,580	4,140	0,632	0,625	0,636	0,687
3I4S	0,573	17,321	0,668	0,583	0,760	-9,345	-279,791	17,724	0,536	0,595	0,473	0,707
3IA1	0,527	12,373	0,614	0,460	0,775	-73,634	-210,869	20,739	0,486	0,480	0,492	0,611
3IIR	0,656	19,331	0,523	0,474	0,565	-124,124	-221,217	18,945	0,512	0,533	0,494	0,739
3IQ3	0,440	19,117	0,340	0,347	0,333	64,530	-219,445	18,293	0,459	0,450	0,468	0,499
3IX3	0,562	14,413	0,605	0,459	0,751	-33,657	-232,857	16,816	0,481	0,479	0,483	0,717
3K3K	0,534	16,946	0,675	0,846	0,515	-286,387	-318,099	14,485	0,527	0,569	0,488	0,639
3K9U	0,598	22,862	0,448	0,448	0,448	10,398	-242,159	16,571	0,481	0,481	0,481	0,718
3L18	0,582	14,571	0,574	0,510	0,631	199,006	-253,329	15,440	0,535	0,594	0,483	0,708
3LB2	0,579	18,443	0,449	0,447	0,451	56,975	-185,196	19,829	0,487	0,489	0,485	0,731
3LBB	0,492	17,036	0,529	0,424	0,642	-48,060	-282,205	21,628	0,479	0,484	0,474	0,635
3LYN	0,569	23,334	0,421	0,434	0,407	-32,395	-211,102	19,574	0,484	0,486	0,481	0,692
3MGK	0,526	14,764	0,685	0,601	0,769	166,454	-519,735	17,111	0,535	0,555	0,516	0,631
3N4K	0,518	17,231	0,575	0,447	0,699	126,916	-157,015	20,981	0,489	0,489	0,488	0,705
3N7H	0,508	17,469	0,406	0,406	0,406	224,342	-290,558	11,601	0,523	0,482	0,564	0,708
3N8E	0,416	19,142	0,452	0,486	0,418	113,841	-222,505	16,170	0,509	0,555	0,464	0,721
3NBC	0,583	17,100	0,504	0,574	0,430	-46,268	-301,493	19,241	0,483	0,481	0,485	0,724
3OCP	0,465	19,647	0,391	0,392	0,389	-3,046	-294,342	17,149	0,470	0,470	0,470	0,664
3P9X	0,563	17,128	0,699	0,487	0,912	81,272	-357,351	14,237	0,484	0,481	0,486	0,681
3PH4	0,458	15,715	0,664	0,658	0,670	426,565	-256,539	14,276	0,524	0,581	0,468	0,696
3QU1	0,518	22,944	0,437	0,437	0,436	-24,980	-543,805	13,253	0,540	0,608	0,477	0,735
3RD3	0,641	7,821	0,711	0,744	0,677	-123,245	-300,304	19,983	0,498	0,481	0,515	0,789
3RFB	0,503	19,426	0,461	0,497	0,427	77,190	-303,600	18,713	0,481	0,481	0,481	0,652
3RHC	0,518	17,848	0,611	0,825	0,396	101,684	-246,468	12,679	0,514	0,450	0,578	0,663
3RQ3	0,462	19,419	0,461	0,503	0,418	-45,109	-196,838	16,431	0,561	0,450	0,677	0,759
3SLZ	0,608	14,789	0,540	0,390	0,701	165,666	-231,494	13,181	0,525	0,495	0,558	0,689
3SZJ	0,651	14,483	0,553	0,429	0,680	7,939	-158,890	17,284	0,523	0,569	0,476	0,807
3TRF	0,554	18,041	0,520	0,442	0,587	5,710	-257,213	20,067	0,485	0,491	0,478	0,623
3TW2	0,509	14,373	0,580	0,719	0,441	-130,341	-344,165	13,521	0,572	0,591	0,554	0,695
3UJM	0,550	16,187	0,644	0,588	0,706	30,798	-335,093	13,996	0,547	0,640	0,447	0,706
3UMZ	0,497	15,096	0,470	0,553	0,400	-31,629	-253,527	13,769	0,482	0,474	0,489	0,689
3V6G	0,617	23,235	0,599	0,714	0,486	-90,553	-307,860	22,675	0,525	0,482	0,566	0,780
3VRC	0,438	17,962	0,692	0,813	0,564	13,595	-373,691	9,498	0,495	0,507	0,483	0,704
4AUU	0,513	15,984	0,495	0,452	0,539	126,053	-161,783	19,569	0,507	0,528	0,486	0,707
4DF0	0,603	17,513	0,574	0,710	0,427	73,298	-282,827	13,448	0,535	0,524	0,547	0,732
4E7P	0,530	16,607	0,478	0,435	0,527	10,405	-182,235	16,029	0,486	0,484	0,488	0,680
4EC7	0,491	16,796	0,484	0,482	0,488	-22,787	-256,634	22,887	0,493	0,501	0,483	0,647
4EP4	0,498	19,726	0,538	0,614	0,455	21,779	-184,407	19,416	0,495	0,507	0,483	0,679

B.3. Kompleksowanie białek – wielokryterialne

Tabela B.3: Wyniki eksperymentu przewidywania struktury czwartorzędowej białek, polegającego na optymalizacji wielokryterialnej kryteriów pól zewnętrznego i wewnętrznego przy pomocy algorytmu MOSF. Dla każdego białka przedstawione są dane reprezentantów grup konformacji (o najniższych wartościach miary ARC), na które zostały podzielone przez algorytm MOSF odnalezione przez niego przybliżenia optymalnych zbiorów Pareto. Wyjaśnienie nagłówków kolumn: *PDB ID* – identyfikator struktury w bazie PDB; *Grupa* – indeks (I) oraz liczba elementów (N) grupy konformacji; *Kryteria* – wartość RD modelu FOD oraz wartość energii pola ECEPP/3; *Ocena* – wartość RMSD oraz wartości AUC dla kompleksu (C) i łańcuchów (A, B); *Min. RD* – wartości RD i energii konformacji, dla której pole zewnętrzne osiągnęło swoje minimum w danej grupie (pierwszy koniec fragmentu frontu Pareto); *Min. energii* – wartości RD i energii konformacji, dla której pole wewnętrzne osiągnęło swoje minimum w danej grupie (drugi koniec fragmentu frontu Pareto). Podkreślone liczby wskazują na wartości RD mniejsze od 0,5, wartości RMSD mniejsze od 10 Å, lub wartości AUC większe od 0,75.

PDB	**Grupa**		**Kryteria**		**Ocena**				**Min. RD**		**Min. energii**	
ID	**I**	**N**	**RD**	**Energia**	**RMSD**	**C**	**A**	**B**	**RD**	**Energia**	**RD**	**Energia**
137L	1	8	0,615	226,915	20,941	0,552	0,487	0,612	0,612	314,243	0,637	22,607
	2	1	0,731	-132,742	22,446	0,493	0,493	0,493	0,731	-132,742	0,731	-132,742
	3	1	0,708	-102,468	18,108	0,488	0,483	0,493	0,708	-102,468	0,708	-102,468
	4	8	0,697	-88,753	17,790	0,493	0,490	0,497	0,686	-48,959	0,726	-110,811
	5	6	0,666	-24,326	12,380	0,518	0,497	0,538	0,647	9,984	0,669	-28,534
1A25	1	12	0,564	-131,963	18,427	0,491	0,491	0,491	0,513	1144,760	0,564	-131,963
	2	5	0,588	-252,588	16,135	0,626	0,501	0,750	0,583	-197,665	0,588	-252,588
	3	1	0,565	-162,012	18,725	0,463	0,457	0,470	0,565	-162,012	0,565	-162,012
	4	4	0,577	-187,401	16,265	0,617	0,773	0,461	0,577	-187,401	0,611	-293,152
	5	6	0,542	-111,533	19,044	0,492	0,518	0,466	0,512	1187,676	0,542	-130,814
1A78	1	7	0,657	-48,725	16,105	0,590	0,475	0,704	0,641	180,481	0,658	-56,424
	2	7	0,628	753,086	15,192	0,765	0,767	0,763	0,611	1432,027	0,641	119,893
	3	2	0,651	34,520	16,503	0,698	0,654	0,742	0,648	52,337	0,651	34,520
	4	7	0,663	-104,952	8,237	0,781	0,738	0,825	0,662	-57,408	0,683	-162,991
1ADW	1	9	0,658	-325,190	18,762	0,461	0,448	0,474	0,653	-200,730	0,658	-325,190
	2	4	0,586	-119,560	7,514	0,635	0,733	0,536	0,583	-100,938	0,593	-164,323
	3	14	0,578	3,224	8,376	0,753	0,849	0,657	0,560	1534,316	0,599	-195,808
1AG9	1	5	0,539	247,837	16,955	0,574	0,466	0,682	0,539	247,837	0,562	-32,208
	2	9	0,526	1485,321	13,346	0,636	0,457	0,815	0,526	1485,321	0,551	-13,796
	3	2	0,569	-202,876	12,519	0,579	0,479	0,679	0,569	-202,876	0,570	-218,407
	4	4	0,566	-153,854	18,561	0,636	0,509	0,764	0,566	-153,854	0,579	-320,460
	5	3	0,546	2,488	16,565	0,536	0,591	0,482	0,546	2,488	0,563	-39,028
1AI9	1	3	0,587	-127,743	12,899	0,519	0,495	0,537	0,565	263,924	0,587	-127,743
	2	4	0,628	-198,894	19,183	0,503	0,541	0,475	0,628	-198,894	0,630	-258,688
	3	4	0,596	-156,582	15,108	0,611	0,473	0,713	0,596	-156,582	0,626	-191,071
	4	8	0,565	900,366	18,808	0,475	0,476	0,475	0,563	1284,068	0,575	-89,056
1ATL	1	13	0,611	-162,963	16,894	0,469	0,477	0,461	0,602	580,793	0,616	-244,798
	2	2	0,681	-305,542	21,053	0,485	0,482	0,487	0,676	-253,155	0,681	-305,542
1AYO	1	1	0,606	516,589	14,298	0,474	0,524	0,424	0,606	516,589	0,606	516,589
	2	2	0,675	-206,725	18,313	0,474	0,484	0,464	0,675	-206,725	0,710	-213,998
	3	10	0,609	39,160	11,557	0,596	0,540	0,652	0,604	991,008	0,643	-161,641
1B78	1	31	0,562	691,527	19,764	0,654	0,580	0,720	0,562	691,527	0,597	-283,968
1B88	1	15	0,552	-150,766	17,086	0,462	0,467	0,458	0,513	950,275	0,552	-150,766
	2	1	0,601	-271,888	10,056	0,612	0,633	0,587	0,601	-271,888	0,601	-271,888

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
	3	6	0,555	-183,400	12,854	0,705	0,962	0,420	0,551	-124,193	0,583	-232,215
1BD9	1	5	0,587	270,242	13,329	0,654	0,649	0,659	0,587	270,242	0,607	-141,480
	2	1	0,583	608,628	8,832	0,604	0,604	0,605	0,583	608,628	0,583	608,628
	3	7	0,619	-206,413	16,619	0,551	0,625	0,477	0,617	-173,985	0,624	-218,703
	4	7	0,588	97,446	15,370	0,683	0,701	0,665	0,581	1095,946	0,610	-166,150
1BKZ	1	11	0,627	-88,866	8,353	0,654	0,702	0,608	0,623	162,445	0,646	-175,223
	2	10	0,617	507,132	9,503	0,538	0,597	0,483	0,613	1332,558	0,633	-128,129
1BU5	1	10	0,645	-185,861	18,289	0,483	0,483	0,483	0,619	-57,603	0,645	-185,861
	2	9	0,606	-23,883	16,242	0,493	0,493	0,493	0,578	879,211	0,611	-42,559
	3	7	0,588	140,446	16,145	0,464	0,462	0,465	0,579	509,458	0,592	78,275
	4	5	0,603	7,851	16,084	0,490	0,490	0,490	0,591	116,043	0,603	7,851
1CO2	1	4	0,555	-202,014	9,646	0,635	0,699	0,571	0,555	-202,014	0,562	-285,543
	2	2	0,518	434,641	11,656	0,552	0,594	0,509	0,518	434,641	0,526	55,480
	3	13	0,515	557,108	15,541	0,668	0,810	0,525	0,508	1599,152	0,546	-162,551
1C3I	1	9	0,595	82,691	15,187	0,599	0,475	0,735	0,593	111,038	0,606	-46,492
	2	13	0,614	-193,345	17,022	0,477	0,472	0,482	0,577	1559,189	0,614	-193,345
	3	1	0,656	-212,073	17,285	0,470	0,478	0,463	0,656	-212,073	0,656	-212,073
1C77	1	9	0,498	-200,491	15,814	0,661	0,851	0,453	0,483	-101,873	0,522	-233,378
	2	8	0,470	-83,471	17,575	0,511	0,537	0,483	0,453	683,239	0,477	-94,513
	3	6	0,453	152,530	19,878	0,568	0,669	0,458	0,453	152,530	0,482	-95,635
1CBK	1	1	0,569	-117,078	17,903	0,479	0,479	0,479	0,569	-117,078	0,569	-117,078
	2	2	0,558	-97,852	19,512	0,465	0,465	0,465	0,558	-97,852	0,559	-103,192
	3	6	0,572	-121,339	19,733	0,477	0,479	0,475	0,571	-118,836	0,578	-155,382
	4	11	0,540	-31,078	20,873	0,470	0,472	0,468	0,527	759,431	0,542	-91,520
1COZ	1	3	0,556	-318,150	15,729	0,569	0,688	0,450	0,556	-318,150	0,566	-323,948
	2	3	0,543	-207,296	17,296	0,506	0,520	0,491	0,533	-152,928	0,543	-207,296
	3	22	0,513	180,673	15,466	0,541	0,624	0,459	0,510	527,475	0,525	-114,923
1CSG	1	2	0,599	-179,754	19,302	0,467	0,472	0,462	0,594	-174,335	0,599	-179,754
	2	3	0,591	-124,687	19,503	0,455	0,462	0,448	0,577	881,172	0,591	-124,687
	3	8	0,669	-315,688	18,704	0,474	0,476	0,472	0,653	-264,464	0,686	-347,420
	4	10	0,643	-261,637	18,353	0,486	0,486	0,486	0,578	539,870	0,643	-261,637
	5	2	0,584	-16,166	19,089	0,455	0,462	0,448	0,583	59,876	0,584	-16,166
1DOQ	1	18	0,455	620,135	14,592	0,604	0,727	0,451	0,439	1500,492	0,536	-185,065
1D1G	1	6	0,662	-243,542	16,025	0,555	0,629	0,481	0,648	-194,164	0,674	-262,459
	2	13	0,543	10,455	4,851	0,708	0,690	0,726	0,543	10,455	0,575	-148,210
	3	6	0,582	-157,580	12,051	0,573	0,501	0,645	0,582	-157,580	0,601	-185,257
1DHF	1	12	0,602	-262,274	20,646	0,493	0,494	0,491	0,566	373,147	0,604	-272,911
	2	5	0,563	916,256	19,358	0,471	0,489	0,453	0,561	1028,118	0,569	-5,574
1DQE	1	18	0,582	-124,235	8,915	0,677	0,830	0,462	0,567	667,722	0,592	-170,508
	2	7	0,626	-221,574	12,745	0,508	0,534	0,473	0,606	-198,137	0,626	-292,406
1DZR	1	4	0,584	-184,331	20,442	0,547	0,603	0,487	0,582	-166,416	0,584	-244,136
	2	11	0,553	-140,896	23,816	0,479	0,471	0,487	0,537	417,807	0,557	-166,414
1EAJ	1	6	0,667	-177,797	12,237	0,596	0,710	0,481	0,664	-146,170	0,684	-188,735
	2	1	0,652	-136,689	17,067	0,461	0,468	0,453	0,652	-136,689	0,652	-136,689
	3	20	0,591	-129,111	18,178	0,481	0,482	0,481	0,555	1526,119	0,591	-129,111
	4	2	0,627	-134,931	17,319	0,493	0,491	0,495	0,624	-134,197	0,627	-134,931
1EO6	1	9	0,601	-156,462	20,062	0,485	0,485	0,485	0,601	-156,462	0,609	-238,600
	2	23	0,590	-149,355	16,717	0,464	0,470	0,459	0,522	496,797	0,590	-149,355
1EX2	1	1	0,687	-278,096	20,394	0,606	0,719	0,483	0,687	-278,096	0,687	-278,096
	2	9	0,603	-27,501	24,566	0,481	0,480	0,483	0,593	473,201	0,613	-98,160
	3	10	0,627	-261,259	21,560	0,639	0,804	0,460	0,608	-85,585	0,627	-261,259
1EYV	1	1	0,718	-191,809	19,957	0,471	0,474	0,469	0,718	-191,809	0,718	-191,809
	2	3	0,652	-171,902	19,319	0,487	0,487	0,487	0,617	-138,854	0,652	-171,902
	3	5	0,632	-164,038	19,853	0,478	0,474	0,482	0,610	-98,392	0,632	-164,038
	4	1	0,640	-166,089	15,865	0,533	0,482	0,579	0,640	-166,089	0,640	-166,089
	5	4	0,691	-174,864	21,041	0,491	0,491	0,491	0,675	-174,607	0,702	-191,268
	6	11	0,594	120,168	16,276	0,471	0,447	0,491	0,590	675,950	0,611	-121,631
1F08	1	18	0,579	1,004	17,094	0,595	0,729	0,460	0,573	885,023	0,599	-181,902
1F1C	1	3	0,626	-247,068	18,051	0,473	0,479	0,467	0,626	-247,068	0,627	-264,804

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
	2	24	0,592	1297,195	5,358	0,800	0,856	0,744	0,592	1297,195	0,615	-118,934
	3	3	0,592	2333,816	10,055	0,683	0,868	0,497	0,592	2333,816	0,620	-184,675
1F46	1	9	0,505	-123,218	19,613	0,460	0,468	0,453	0,483	559,027	0,506	-150,190
	2	12	0,529	-201,208	19,377	0,484	0,492	0,476	0,478	1398,105	0,529	-201,208
1FLM	1	16	0,545	233,647	3,109	0,710	0,748	0,672	0,538	978,872	0,569	-71,125
	2	10	0,606	-128,192	16,191	0,543	0,621	0,465	0,603	-71,375	0,609	-198,787
1FQT	1	4	0,701	-238,708	16,149	0,552	0,476	0,610	0,682	-228,452	0,723	-268,083
	2	2	0,607	-162,724	13,645	0,504	0,471	0,528	0,606	-136,307	0,607	-162,724
	3	4	0,622	-177,803	15,632	0,449	0,456	0,441	0,622	-165,430	0,622	-223,492
	4	14	0,592	-118,931	13,019	0,558	0,535	0,576	0,569	286,228	0,598	-121,258
1FTP	1	2	0,692	-127,387	15,389	0,511	0,480	0,548	0,672	-124,929	0,692	-127,387
	2	10	0,614	897,760	12,224	0,701	0,921	0,452	0,614	897,760	0,641	-103,181
1G17	1	18	0,580	580,208	12,397	0,966	0,961	0,970	0,578	645,231	0,586	-194,535
	2	1	0,602	-241,636	17,600	0,484	0,485	0,482	0,602	-241,636	0,602	-241,636
	3	1	0,595	-203,545	22,508	0,485	0,488	0,482	0,595	-203,545	0,595	-203,545
1G2Q	1	6	0,643	-163,003	17,439	0,563	0,453	0,673	0,634	-154,070	0,663	-252,460
	2	21	0,560	251,922	18,010	0,668	0,556	0,780	0,556	1673,539	0,581	-80,930
	3	5	0,601	-88,262	15,038	0,550	0,486	0,613	0,601	-88,262	0,608	-117,762
1GE7	1	12	0,688	-41,040	19,329	0,480	0,481	0,478	0,676	729,494	0,689	-91,107
1GY6	1	1	0,682	-113,489	18,231	0,482	0,480	0,485	0,682	-113,489	0,682	-113,489
	2	1	0,624	-111,544	20,498	0,480	0,485	0,474	0,624	-111,544	0,624	-111,544
	3	17	0,467	688,838	14,336	0,577	0,436	0,714	0,464	924,142	0,487	-57,678
	4	2	0,487	-70,062	17,258	0,536	0,490	0,580	0,487	-70,062	0,507	-104,859
	5	6	0,474	194,821	17,113	0,582	0,455	0,705	0,474	194,821	0,527	-110,466
1HFY	1	1	0,587	-117,654	14,255	0,566	0,565	0,568	0,587	-117,654	0,587	-117,654
	2	2	0,647	-198,300	11,025	0,557	0,463	0,659	0,647	-198,300	0,656	-233,784
	3	3	0,617	-174,325	16,783	0,523	0,569	0,472	0,615	-160,954	0,618	-192,997
	4	12	0,611	-131,887	10,912	0,603	0,690	0,509	0,556	725,010	0,611	-131,887
1HKQ	1	1	0,612	-272,350	15,486	0,528	0,481	0,580	0,612	-272,350	0,612	-272,350
	2	3	0,586	-251,515	14,385	0,533	0,514	0,553	0,584	-241,309	0,586	-251,515
	3	18	0,487	-240,977	16,710	0,483	0,444	0,526	0,460	2255,565	0,487	-240,977
	4	1	0,482	-228,842	19,707	0,472	0,472	0,472	0,482	-228,842	0,482	-228,842
1HLC	1	8	0,585	30,200	16,987	0,493	0,491	0,495	0,583	150,973	0,604	-102,756
	2	6	0,620	-137,712	12,894	0,572	0,504	0,645	0,620	-137,712	0,633	-251,018
1HPC	1	1	0,698	-255,412	16,986	0,652	0,796	0,546	0,698	-255,412	0,698	-255,412
	2	3	0,700	-275,576	17,116	0,633	0,746	0,550	0,699	-262,194	0,703	-279,447
	3	5	0,614	-185,767	15,216	0,546	0,642	0,474	0,614	-185,767	0,643	-254,985
	4	13	0,567	569,761	16,030	0,585	0,800	0,427	0,567	569,761	0,590	-185,239
1I4S	1	4	0,612	-476,912	20,135	0,503	0,476	0,530	0,612	-476,912	0,624	-508,766
	2	5	0,563	-212,489	12,122	0,525	0,501	0,550	0,555	-171,699	0,567	-234,254
	3	2	0,548	-168,119	11,277	0,565	0,538	0,592	0,548	-168,119	0,575	-273,757
	4	5	0,515	-141,446	6,377	0,673	0,601	0,746	0,513	-86,703	0,534	-161,344
	5	7	0,577	-300,311	14,888	0,608	0,579	0,638	0,567	-250,220	0,640	-595,396
	6	14	0,502	795,568	12,745	0,738	0,692	0,784	0,502	1000,135	0,515	-135,239
1I6W	1	2	0,714	-174,282	16,913	0,477	0,489	0,466	0,713	-156,892	0,714	-174,282
	2	4	0,629	-136,240	17,579	0,484	0,480	0,489	0,611	-102,734	0,631	-138,115
	3	17	0,597	-64,122	21,583	0,482	0,480	0,483	0,552	1566,640	0,607	-76,081
	4	4	0,608	-97,459	21,895	0,483	0,480	0,486	0,606	-73,261	0,608	-97,459
1IAZ	1	17	0,583	681,353	10,918	0,717	0,856	0,603	0,576	1238,685	0,687	-222,401
	2	6	0,609	-167,298	11,364	0,682	0,932	0,476	0,595	12,633	0,710	-250,084
1IFV	1	12	0,665	797,108	13,281	0,769	0,871	0,667	0,665	797,108	0,677	-103,009
	2	8	0,677	-116,028	18,281	0,616	0,473	0,759	0,677	-116,028	0,682	-209,870
1IPI	1	1	0,529	-85,078	14,802	0,527	0,479	0,575	0,529	-85,078	0,529	-85,078
	2	13	0,488	-59,803	17,644	0,489	0,484	0,495	0,460	256,796	0,513	-78,817
	3	10	0,499	-74,578	16,969	0,484	0,484	0,484	0,441	752,613	0,499	-74,578
1IQ6	1	5	0,612	-147,741	11,445	0,586	0,581	0,593	0,590	-109,377	0,612	-147,741
	2	5	0,646	-278,258	13,440	0,619	0,635	0,601	0,575	-13,733	0,646	-278,258
	3	2	0,621	-156,732	10,752	0,558	0,545	0,574	0,621	-156,732	0,639	-171,043
1J3M	1	1	0,563	-117,341	6,350	0,578	0,580	0,576	0,563	-117,341	0,563	-117,341

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
	2	8	0,603	-179,039	5,011	0,676	0,665	0,687	0,580	-161,445	0,610	-218,941
	3	2	0,561	-81,361	7,030	0,598	0,630	0,566	0,561	-81,361	0,563	-156,600
	4	12	0,516	21,797	16,488	0,627	0,635	0,619	0,508	414,717	0,560	-63,937
1J3Q	1	4	0,534	-130,320	16,756	0,527	0,479	0,575	0,534	-130,320	0,683	-348,684
	2	8	0,535	-155,255	22,818	0,483	0,479	0,486	0,535	-155,255	0,551	-277,230
	3	13	0,477	436,439	17,889	0,639	0,619	0,659	0,470	1306,563	0,529	-107,898
1JR8	1	5	0,543	-236,336	16,652	0,483	0,415	0,549	0,541	-215,913	0,553	-308,195
	2	3	0,536	-169,310	15,908	0,491	0,457	0,523	0,536	-169,310	0,537	-208,064
	3	16	0,452	1303,926	14,808	0,604	0,372	0,829	0,450	1997,200	0,515	-142,488
1K4Z	1	6	0,484	537,494	16,973	0,586	0,668	0,513	0,484	537,494	0,489	-50,420
	2	16	0,486	203,385	17,888	0,570	0,682	0,469	0,486	203,385	0,521	-268,732
1KPT	1	5	0,689	-45,947	15,507	0,487	0,489	0,484	0,681	5,155	0,703	-66,140
	2	11	0,666	393,784	14,374	0,569	0,499	0,645	0,664	898,362	0,688	-29,185
	3	5	0,697	-53,613	15,783	0,489	0,489	0,489	0,671	301,131	0,699	-54,061
	4	1	0,764	-67,503	16,806	0,487	0,468	0,508	0,764	-67,503	0,764	-67,503
1L8D	1	1	0,709	-372,519	27,712	0,483	0,483	0,483	0,709	-372,519	0,709	-372,519
	2	5	0,645	-306,361	10,717	0,541	0,626	0,460	0,639	-302,907	0,659	-371,072
	3	18	0,581	-189,897	11,095	0,584	0,604	0,566	0,552	179,393	0,638	-246,412
1LFA	1	13	0,616	-295,161	14,850	0,529	0,586	0,480	0,534	-146,316	0,653	-361,633
	2	4	0,506	10,686	16,575	0,521	0,578	0,474	0,486	2265,104	0,506	10,686
	3	5	0,499	382,371	16,059	0,526	0,583	0,477	0,488	1824,375	0,504	184,349
	4	8	0,515	-62,564	17,880	0,618	0,786	0,477	0,506	165,535	0,538	-161,901
1M08	1	6	0,542	-133,157	13,383	0,538	0,484	0,583	0,542	-133,157	0,570	-194,257
	2	13	0,519	257,479	15,385	0,573	0,558	0,586	0,519	257,479	0,598	-203,913
1M4I	1	14	0,601	2034,428	15,424	0,648	0,483	0,828	0,601	2034,428	0,628	-189,994
	2	15	0,677	-375,327	20,260	0,474	0,478	0,471	0,671	-263,518	0,685	-512,287
1M4J	1	9	0,571	-92,749	14,895	0,650	0,884	0,433	0,549	889,157	0,584	-222,707
	2	1	0,614	-226,941	8,504	0,648	0,571	0,718	0,614	-226,941	0,614	-226,941
1M4R	1	1	0,588	2200,602	14,759	0,606	0,789	0,436	0,588	2200,602	0,588	2200,602
	2	16	0,631	-180,599	1,886	0,837	0,805	0,865	0,590	364,911	0,644	-267,327
1MK4	1	4	0,726	-139,781	18,436	0,485	0,485	0,485	0,719	-130,835	0,736	-154,231
	2	2	0,661	-95,920	18,788	0,527	0,502	0,553	0,661	-95,920	0,664	-124,518
	3	10	0,627	130,079	7,808	0,588	0,558	0,617	0,626	482,836	0,638	-39,890
	4	3	0,644	-67,746	15,828	0,496	0,496	0,496	0,640	-60,147	0,644	-75,169
	5	4	0,643	-67,698	15,954	0,504	0,533	0,474	0,642	-61,593	0,649	-85,287
1MKA	1	15	0,641	-82,011	18,496	0,556	0,452	0,658	0,632	658,691	0,646	-210,360
1NA8	1	2	0,672	-203,109	20,782	0,507	0,466	0,548	0,672	-203,109	0,674	-210,332
	2	3	0,685	-215,651	21,410	0,529	0,470	0,587	0,685	-215,651	0,689	-227,506
	3	5	0,647	-182,116	18,235	0,465	0,462	0,469	0,639	-101,111	0,649	-184,661
	4	10	0,625	-64,554	22,256	0,488	0,485	0,492	0,610	931,048	0,632	-77,504
1NBC	1	19	0,695	-67,160	19,640	0,497	0,497	0,497	0,664	382,612	0,695	-67,160
	2	1	0,665	360,150	19,800	0,453	0,459	0,447	0,665	360,150	0,665	360,150
	3	2	0,709	-79,037	17,871	0,473	0,473	0,474	0,709	-79,037	0,735	-79,280
	4	3	0,739	-89,912	20,368	0,480	0,483	0,477	0,739	-89,912	0,742	-98,549
	5	1	0,706	-78,719	17,495	0,560	0,480	0,724	0,706	-78,719	0,706	-78,719
1NCO	1	21	0,701	-117,072	16,490	0,485	0,511	0,461	0,669	1134,412	0,701	-117,072
1NP8	1	7	0,616	-355,577	13,414	0,495	0,477	0,512	0,615	-336,827	0,623	-397,640
	2	15	0,539	-320,863	9,050	0,601	0,470	0,731	0,505	407,847	0,541	-333,265
1NWP	1	2	0,665	-112,897	12,653	0,506	0,467	0,551	0,665	-94,965	0,665	-112,897
	2	2	0,633	-80,561	15,418	0,575	0,475	0,689	0,633	-80,561	0,647	-88,154
	3	16	0,608	0,318	12,130	0,665	0,454	0,904	0,591	1314,070	0,657	-91,501
1NWW	1	3	0,640	-76,608	18,531	0,546	0,474	0,616	0,639	-55,961	0,643	-108,951
	2	1	0,715	-211,652	16,752	0,506	0,474	0,538	0,715	-211,652	0,715	-211,652
	3	2	0,663	-157,109	21,989	0,491	0,491	0,491	0,663	-157,109	0,670	-190,025
	4	10	0,621	100,559	12,094	0,568	0,636	0,501	0,605	511,078	0,634	-39,306
	5	2	0,655	-113,582	16,996	0,553	0,477	0,627	0,655	-113,582	0,658	-129,243
1NXM	1	3	0,656	-178,878	16,002	0,540	0,584	0,497	0,654	-175,058	0,658	-203,145
	2	11	0,533	-107,266	20,254	0,484	0,485	0,482	0,525	633,969	0,562	-171,856
	3	1	0,693	-228,876	22,272	0,491	0,509	0,474	0,693	-228,876	0,693	-228,876

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
1OH0	1	2	0,685	-181,679	15,874	0,480	0,480	0,480	0,685	-181,679	0,699	-235,681
	2	18	0,622	1201,726	11,201	0,654	0,773	0,539	0,622	1201,726	0,670	-173,299
1OPA	1	8	0,606	770,408	6,801	0,735	0,857	0,614	0,606	904,391	0,617	-30,498
	2	8	0,620	-88,516	15,772	0,653	0,766	0,539	0,616	-8,632	0,625	-164,953
1P6O	1	1	0,511	-154,322	14,227	0,486	0,500	0,473	0,511	-154,322	0,511	-154,322
	2	2	0,595	-170,814	15,972	0,579	0,528	0,629	0,595	-170,814	0,601	-186,962
	3	3	0,602	-205,300	14,195	0,611	0,585	0,637	0,602	-205,300	0,678	-222,189
	4	14	0,510	-148,906	18,907	0,490	0,488	0,492	0,476	796,117	0,510	-148,906
1PPV	1	5	0,716	-253,964	20,251	0,477	0,478	0,475	0,709	-162,533	0,716	-253,964
	2	1	0,691	-139,446	15,603	0,491	0,487	0,494	0,691	-139,446	0,691	-139,446
	3	19	0,656	253,108	13,902	0,547	0,603	0,497	0,653	856,474	0,680	-122,131
1Q98	1	24	0,661	-15,762	6,589	0,788	0,771	0,806	0,634	1675,355	0,675	-150,998
	2	4	0,771	-179,138	18,428	0,488	0,490	0,486	0,738	-161,799	0,771	-179,138
1QAH	1	2	0,625	-102,522	18,711	0,482	0,480	0,484	0,625	-102,522	0,629	-109,572
	2	1	0,601	-53,272	10,813	0,500	0,530	0,471	0,601	-53,272	0,601	-53,272
	3	2	0,634	-112,270	13,832	0,619	0,509	0,730	0,634	-112,270	0,665	-123,427
	4	6	0,600	-39,790	9,544	0,586	0,530	0,642	0,568	642,863	0,603	-55,258
	5	11	0,574	231,470	14,122	0,572	0,476	0,667	0,561	1062,379	0,605	-92,991
1QSD	1	3	0,581	-96,733	27,241	0,537	0,483	0,588	0,575	-67,930	0,581	-96,733
	2	7	0,605	-172,013	25,207	0,543	0,619	0,471	0,588	-120,390	0,605	-172,013
	3	10	0,556	-66,428	27,317	0,501	0,520	0,482	0,540	475,866	0,556	-66,428
	4	6	0,645	-351,636	15,608	0,489	0,471	0,506	0,630	-249,690	0,645	-351,636
	5	5	0,547	-40,724	28,242	0,494	0,494	0,494	0,545	19,466	0,552	-54,752
1QZ8	1	20	0,541	-196,004	18,295	0,463	0,469	0,457	0,511	1458,098	0,544	-215,111
1SGM	1	13	0,617	-8,057	13,645	0,535	0,480	0,590	0,615	57,278	0,693	-182,271
1SH8	1	3	0,643	-218,793	20,852	0,466	0,469	0,463	0,643	-160,808	0,654	-224,716
	2	14	0,572	522,114	14,019	0,571	0,438	0,698	0,572	522,114	0,611	-141,977
1SQU	1	17	0,597	105,783	12,957	0,613	0,540	0,681	0,580	1882,784	0,607	-136,017
	2	8	0,596	243,698	12,781	0,577	0,527	0,623	0,596	243,698	0,633	-199,153
1TFP	1	16	0,593	1443,396	13,562	0,553	0,576	0,531	0,593	1443,396	0,635	-92,794
	2	5	0,667	-104,349	17,224	0,638	0,814	0,461	0,667	-104,349	0,674	-228,738
	3	4	0,604	394,434	9,456	0,613	0,431	0,795	0,604	394,434	0,621	-32,488
1TLJ	1	1	0,685	-255,564	23,900	0,476	0,478	0,475	0,685	-255,564	0,685	-255,564
	2	5	0,591	-122,125	18,999	0,476	0,481	0,472	0,591	-122,125	0,613	-253,639
	3	9	0,532	899,473	4,321	0,633	0,653	0,611	0,532	899,473	0,580	-114,520
1V5X	1	4	0,713	-221,998	20,826	0,475	0,474	0,477	0,707	-183,732	0,713	-221,998
	2	3	0,655	-4,837	6,796	0,605	0,603	0,607	0,655	-4,837	0,659	-35,123
	3	13	0,644	352,162	16,341	0,668	0,649	0,688	0,639	1398,986	0,658	-18,161
	4	8	0,663	-103,063	18,108	0,604	0,701	0,504	0,660	-40,124	0,664	-142,803
	5	6	0,644	454,467	17,124	0,537	0,471	0,604	0,644	454,467	0,700	-181,846
1V7L	1	17	0,471	511,126	18,676	0,602	0,507	0,681	0,470	1011,351	0,489	-106,860
	2	2	0,498	-194,278	18,328	0,515	0,490	0,536	0,498	-194,278	0,505	-233,098
	3	1	0,526	-281,152	19,028	0,479	0,484	0,474	0,526	-281,152	0,526	-281,152
	4	9	0,491	-153,306	18,516	0,537	0,464	0,597	0,491	-153,306	0,524	-274,407
1V8H	1	12	0,603	-182,078	18,440	0,489	0,484	0,495	0,569	-42,805	0,603	-182,078
1VC1	1	3	0,619	-236,534	16,160	0,532	0,553	0,511	0,619	-236,534	0,622	-252,505
	2	4	0,468	743,415	12,290	0,766	0,855	0,678	0,460	1158,041	0,474	527,254
	3	6	0,558	-96,247	12,268	0,522	0,584	0,459	0,558	-96,247	0,566	-179,197
	4	6	0,561	-129,519	12,031	0,529	0,594	0,464	0,556	-60,739	0,576	-224,096
	5	23	0,465	968,432	10,139	0,800	0,766	0,834	0,442	1588,206	0,505	-57,368
1VH5	1	8	0,528	145,975	16,246	0,623	0,602	0,644	0,524	183,868	0,547	-122,823
	2	11	0,527	148,412	5,549	0,570	0,584	0,556	0,517	229,941	0,597	-238,147
1VJ2	1	1	0,584	-199,289	17,588	0,484	0,424	0,545	0,584	-199,289	0,584	-199,289
	2	4	0,542	-157,150	16,335	0,517	0,445	0,591	0,536	-114,960	0,542	-157,150
	3	14	0,513	-37,138	15,536	0,591	0,459	0,727	0,503	443,183	0,545	-181,518
1VLT	1	1	0,674	-200,615	22,834	0,512	0,457	0,559	0,674	-200,615	0,674	-200,615
	2	7	0,649	-68,082	12,488	0,523	0,567	0,488	0,647	-26,790	0,662	-96,589
	3	2	0,661	-90,831	23,716	0,519	0,559	0,488	0,659	-83,951	0,661	-90,831
	4	7	0,666	-173,969	12,893	0,536	0,488	0,575	0,665	-118,658	0,672	-187,240

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
1WPN	1	7	0,629	-282,352	19,920	0,516	0,544	0,488	0,587	600,808	0,629	-282,352
	2	15	0,606	-209,076	19,838	0,490	0,491	0,488	0,585	1222,897	0,606	-209,076
1WWZ	1	1	0,568	-355,018	21,144	0,473	0,473	0,472	0,568	-355,018	0,568	-355,018
	2	11	0,478	-45,942	22,044	0,483	0,486	0,479	0,471	1103,320	0,519	-262,142
	3	5	0,523	-295,568	22,287	0,473	0,473	0,472	0,520	-275,906	0,542	-322,279
	4	8	0,476	18,001	21,844	0,481	0,480	0,483	0,472	456,654	0,507	-255,351
1X77	1	7	0,632	-94,275	16,712	0,552	0,490	0,612	0,626	-76,921	0,637	-117,432
	2	3	0,679	-163,774	18,022	0,501	0,480	0,522	0,647	-130,719	0,679	-163,774
	3	15	0,623	-74,137	18,853	0,506	0,540	0,473	0,611	394,829	0,623	-74,137
1XOX	1	8	0,591	-424,742	19,547	0,447	0,449	0,444	0,579	-294,177	0,591	-424,742
	2	8	0,517	-211,177	17,720	0,484	0,454	0,514	0,514	-100,122	0,517	-232,003
	3	8	0,509	762,849	16,950	0,567	0,625	0,509	0,509	762,849	0,512	-77,085
	4	3	0,525	-233,371	17,796	0,463	0,468	0,458	0,525	-233,371	0,526	-253,562
1YAV	1	9	0,546	-322,409	12,287	0,541	0,500	0,582	0,544	-290,805	0,558	-408,847
	2	7	0,497	-125,038	13,533	0,628	0,490	0,770	0,497	-125,038	0,504	-290,073
	3	6	0,478	441,010	12,425	0,571	0,439	0,707	0,478	441,010	0,487	-101,440
1YOC	1	8	0,651	-115,656	16,974	0,535	0,617	0,450	0,649	-85,688	0,661	-163,339
	2	12	0,634	1234,460	18,294	0,613	0,594	0,633	0,634	1234,460	0,645	-42,937
	3	3	0,677	-176,409	14,199	0,566	0,667	0,463	0,676	-175,207	0,682	-186,635
1Z3A	1	19	0,438	207,240	14,779	0,621	0,539	0,704	0,430	1219,123	0,443	-82,843
	2	4	0,457	-159,676	18,130	0,491	0,510	0,473	0,457	-159,676	0,476	-229,193
1Z9M	1	1	0,553	-145,443	14,130	0,542	0,478	0,620	0,553	-145,443	0,553	-145,443
	2	2	0,555	-194,067	17,107	0,539	0,472	0,620	0,555	-194,067	0,601	-204,224
	3	3	0,515	-115,426	16,633	0,584	0,444	0,750	0,510	-94,964	0,520	-121,102
	4	8	0,485	-71,747	16,543	0,486	0,489	0,484	0,460	330,402	0,526	-142,002
1Z9P	1	1	0,605	-193,170	19,977	0,501	0,479	0,522	0,605	-193,170	0,605	-193,170
	2	19	0,459	722,963	15,846	0,809	0,797	0,822	0,448	1700,386	0,481	-183,835
1ZVF	1	15	0,676	-194,611	18,720	0,535	0,590	0,476	0,571	305,688	0,676	-194,611
	2	5	0,597	-162,032	21,453	0,553	0,479	0,634	0,586	-96,585	0,597	-162,032
	3	2	0,654	-193,702	24,922	0,488	0,486	0,490	0,653	-172,678	0,654	-193,702
2A4N	1	2	0,630	-319,388	16,482	0,529	0,510	0,548	0,625	-289,765	0,630	-319,388
	2	2	0,651	-369,399	19,110	0,496	0,517	0,476	0,651	-369,399	0,674	-390,385
	3	11	0,582	1460,257	16,094	0,522	0,564	0,481	0,582	1460,257	0,605	-271,101
2A6P	1	5	0,636	-207,620	16,893	0,537	0,445	0,629	0,636	-207,620	0,653	-306,081
	2	15	0,623	17,826	18,284	0,564	0,465	0,664	0,609	755,975	0,632	-152,011
2A9S	1	17	0,641	2808,001	15,255	0,587	0,748	0,429	0,641	2808,001	0,670	-125,461
	2	3	0,696	-169,978	16,090	0,554	0,629	0,481	0,696	-169,978	0,721	-207,274
	3	1	0,675	-134,667	16,286	0,527	0,581	0,474	0,675	-134,667	0,675	-134,667
2AB0	1	7	0,564	-63,749	17,974	0,552	0,473	0,630	0,561	-38,197	0,614	-174,940
	2	8	0,548	274,196	15,190	0,579	0,467	0,691	0,547	861,471	0,559	4,016
	3	3	0,584	-134,208	16,345	0,539	0,502	0,577	0,567	-92,783	0,584	-134,208
	4	3	0,657	-180,232	21,178	0,485	0,482	0,488	0,650	-178,141	0,658	-195,191
2BPD	1	1	0,664	-174,350	19,257	0,471	0,478	0,465	0,664	-174,350	0,664	-174,350
	2	8	0,563	13,747	9,602	0,653	0,762	0,537	0,555	187,723	0,581	-74,650
	3	12	0,583	-88,669	13,452	0,657	0,862	0,439	0,583	-88,669	0,610	-169,838
2CAR	1	8	0,691	-314,838	17,054	0,496	0,466	0,527	0,666	-195,736	0,702	-369,134
	2	14	0,637	-37,097	17,273	0,487	0,501	0,472	0,630	398,953	0,664	-194,441
2CC0	1	2	0,711	1597,555	14,051	0,532	0,449	0,628	0,711	1597,555	0,731	-138,560
	2	15	0,724	-95,854	14,526	0,523	0,557	0,483	0,711	530,947	0,724	-109,368
	3	1	0,730	-117,064	14,615	0,489	0,489	0,489	0,730	-117,064	0,730	-117,064
	4	4	0,743	-191,851	18,385	0,480	0,480	0,481	0,737	-140,437	0,743	-191,851
2D3K	1	3	0,613	-178,589	17,994	0,493	0,493	0,493	0,609	-165,007	0,613	-178,589
	2	11	0,406	204,333	9,125	0,643	0,748	0,538	0,398	708,293	0,447	-111,970
	3	5	0,476	-137,632	13,901	0,536	0,568	0,503	0,471	-123,108	0,492	-163,030
2DC4	1	18	0,608	266,102	13,773	0,620	0,709	0,532	0,606	716,261	0,624	-139,424
	2	1	0,659	-167,996	21,010	0,541	0,605	0,476	0,659	-167,996	0,659	-167,996
2DCT	1	2	0,543	-212,345	17,472	0,495	0,523	0,466	0,535	-198,257	0,543	-212,345
	2	8	0,568	-316,341	17,711	0,488	0,444	0,533	0,545	-238,618	0,571	-323,881
	3	2	0,433	-62,953	5,280	0,645	0,640	0,651	0,433	-62,953	0,456	-154,693

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
	4	21	0,363	407,532	10,511	0,653	0,634	0,674	0,348	1034,112	0,420	-61,110
	5	2	0,581	-337,486	15,786	0,503	0,552	0,452	0,581	-337,486	0,585	-344,246
	6	5	0,435	-76,876	7,155	0,614	0,625	0,603	0,423	-62,736	0,450	-112,117
2EAV	1	7	0,664	580,222	13,816	0,679	0,457	0,886	0,663	604,872	0,677	-120,058
	2	9	0,658	888,755	14,836	0,652	0,447	0,844	0,658	888,755	0,693	-165,866
2F3G	1	11	0,594	43,492	14,402	0,633	0,448	0,819	0,584	578,352	0,601	-126,564
	2	6	0,620	-218,552	14,924	0,642	0,829	0,455	0,620	-218,552	0,637	-281,964
2FBN	1	1	0,584	-300,699	17,887	0,480	0,482	0,478	0,584	-300,699	0,584	-300,699
	2	9	0,505	-201,724	21,594	0,570	0,675	0,471	0,493	-172,778	0,514	-272,629
	3	2	0,527	-294,508	11,671	0,564	0,518	0,608	0,525	-284,583	0,527	-294,508
	4	9	0,483	-85,888	24,214	0,482	0,482	0,482	0,474	576,033	0,491	-141,248
2FZF	1	3	0,569	-220,609	19,044	0,512	0,482	0,543	0,561	-189,484	0,569	-220,609
	2	15	0,560	-177,522	19,446	0,473	0,471	0,475	0,539	872,144	0,560	-177,522
	3	1	0,573	-259,616	21,712	0,516	0,464	0,571	0,573	-259,616	0,573	-259,616
2GJA	1	16	0,533	35,176	20,479	0,463	0,467	0,459	0,517	1475,577	0,548	-97,569
	2	3	0,559	-169,432	14,467	0,586	0,467	0,705	0,559	-169,432	0,562	-251,121
	3	8	0,557	-113,623	18,712	0,475	0,480	0,469	0,540	22,221	0,557	-113,623
2H29	1	3	0,591	-124,795	14,503	0,527	0,477	0,576	0,591	-124,795	0,593	-173,760
	2	1	0,609	-175,822	14,728	0,529	0,513	0,544	0,609	-175,822	0,609	-175,822
	3	4	0,577	307,551	15,699	0,503	0,458	0,548	0,571	1078,169	0,579	230,329
	4	7	0,577	247,486	17,638	0,649	0,763	0,535	0,572	1069,958	0,583	-107,979
	5	5	0,613	-180,319	22,359	0,483	0,495	0,470	0,613	-180,319	0,617	-211,733
2HJ3	1	6	0,469	298,073	14,607	0,605	0,432	0,793	0,468	349,141	0,564	-360,682
	2	12	0,487	-279,380	19,069	0,429	0,438	0,421	0,465	1997,861	0,488	-304,983
2IDL	1	23	0,473	34,172	11,162	0,721	0,673	0,769	0,454	802,502	0,557	-255,325
2IGI	1	4	0,645	-317,445	20,210	0,536	0,606	0,460	0,644	-282,983	0,646	-324,735
	2	2	0,600	-133,796	23,551	0,480	0,473	0,487	0,584	-105,024	0,600	-133,796
	3	4	0,630	-282,832	17,011	0,553	0,645	0,454	0,626	-227,737	0,630	-282,832
	4	12	0,618	-187,550	21,200	0,473	0,486	0,460	0,568	1068,998	0,618	-187,550
2J8M	1	3	0,653	-128,220	20,919	0,477	0,481	0,474	0,649	-47,586	0,654	-140,797
	2	6	0,652	-113,889	21,083	0,466	0,470	0,462	0,622	2171,258	0,656	-160,918
	3	3	0,663	-233,180	20,204	0,462	0,451	0,474	0,662	-205,785	0,663	-233,180
	4	1	0,660	-186,716	20,666	0,468	0,462	0,474	0,660	-186,716	0,660	-186,716
	5	9	0,630	496,810	16,015	0,570	0,479	0,663	0,630	496,810	0,649	-79,282
2J96	1	6	0,689	-210,165	20,682	0,478	0,475	0,481	0,661	-78,637	0,689	-210,165
	2	4	0,667	-117,167	18,173	0,543	0,604	0,489	0,662	-104,268	0,671	-130,025
	3	12	0,652	-63,648	23,096	0,496	0,496	0,496	0,634	686,241	0,652	-63,648
	4	2	0,651	-11,039	17,439	0,499	0,489	0,507	0,651	-11,039	0,654	-74,367
2O7M	1	23	0,601	185,906	18,026	0,573	0,451	0,699	0,597	1102,670	0,639	-253,695
2OE3	1	2	0,636	-196,409	14,284	0,511	0,514	0,508	0,633	-139,251	0,636	-196,409
	2	2	0,601	-112,650	8,907	0,609	0,530	0,676	0,599	-92,069	0,601	-112,650
	3	6	0,502	60,598	11,137	0,684	0,651	0,711	0,477	968,636	0,599	-99,403
	4	17	0,516	8,573	10,925	0,842	0,995	0,717	0,466	1958,359	0,551	-74,731
	5	4	0,507	49,491	11,732	0,739	0,707	0,767	0,471	1241,433	0,521	-4,294
	6	2	0,627	-131,573	15,131	0,654	0,879	0,468	0,624	-125,985	0,627	-131,573
	7	5	0,498	121,731	7,289	0,887	0,818	0,944	0,498	121,731	0,572	-88,981
2OFC	1	15	0,570	3037,775	9,257	0,652	0,804	0,501	0,570	3037,775	0,606	-183,014
	2	3	0,586	197,491	16,023	0,645	0,793	0,498	0,583	707,419	0,593	69,799
2OMD	1	7	0,505	-131,866	16,502	0,574	0,624	0,520	0,497	-90,853	0,510	-178,177
	2	1	0,533	-178,889	15,496	0,522	0,530	0,514	0,533	-178,889	0,533	-178,889
	3	13	0,484	108,538	16,391	0,587	0,621	0,553	0,484	108,538	0,496	-49,564
2P5R	1	8	0,537	201,426	11,393	0,646	0,754	0,499	0,526	985,642	0,537	201,426
	2	6	0,541	-77,043	20,118	0,485	0,469	0,509	0,538	88,953	0,584	-222,748
	3	8	0,556	-177,563	19,821	0,487	0,486	0,487	0,541	70,991	0,556	-177,563
2PBR	1	5	0,621	-370,873	10,677	0,683	0,718	0,620	0,601	-160,890	0,621	-370,873
	2	6	0,608	-238,407	15,943	0,529	0,557	0,474	0,606	-195,306	0,614	-330,788
	3	8	0,598	-150,212	19,996	0,475	0,479	0,471	0,587	354,011	0,598	-150,212
2Q2O	1	13	0,598	-121,855	17,722	0,464	0,463	0,464	0,508	750,057	0,598	-121,855
	2	4	0,624	-135,043	17,273	0,586	0,710	0,444	0,614	-130,086	0,634	-145,689

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
2QSQ	1	8	0,658	-102,772	18,440	0,487	0,489	0,484	0,653	-85,201	0,663	-118,099
	2	9	0,548	390,599	12,798	0,642	0,802	0,464	0,547	614,857	0,572	-62,881
	3	10	0,541	619,126	14,653	0,645	0,778	0,498	0,541	619,126	0,594	-83,087
2QV0	1	5	0,661	-251,851	17,169	0,486	0,486	0,486	0,653	-243,274	0,666	-256,795
	2	6	0,616	-0,008	18,366	0,473	0,477	0,468	0,591	658,869	0,616	-0,008
	3	7	0,619	-14,981	20,098	0,475	0,477	0,473	0,587	1701,294	0,619	-20,258
	4	11	0,632	-242,552	17,762	0,450	0,441	0,459	0,620	-73,786	0,632	-242,552
	5	3	0,599	302,240	18,893	0,450	0,446	0,455	0,596	458,408	0,600	259,240
2QZT	1	3	0,751	-149,943	16,924	0,474	0,474	0,474	0,743	-144,406	0,754	-151,183
	2	1	0,682	-118,263	14,805	0,558	0,631	0,490	0,682	-118,263	0,682	-118,263
	3	1	0,656	-105,993	16,044	0,485	0,485	0,485	0,656	-105,993	0,656	-105,993
	4	6	0,593	1099,991	10,347	0,616	0,599	0,632	0,593	1099,991	0,655	-93,119
	5	13	0,646	-91,861	13,911	0,593	0,525	0,656	0,595	963,722	0,646	-91,861
2SPC	1	13	0,605	-125,779	41,248	0,506	0,501	0,511	0,604	-105,485	0,733	-280,839
	2	3	0,702	-210,338	41,930	0,490	0,492	0,487	0,676	-177,209	0,702	-210,338
	3	2	0,623	-145,688	42,032	0,517	0,511	0,522	0,623	-145,688	0,713	-220,492
2W2A	1	11	0,526	-126,018	0,217	0,973	0,973	0,973	0,514	2251,506	0,542	-147,782
	2	3	0,546	-169,087	18,191	0,790	0,801	0,778	0,546	-169,087	0,551	-193,535
2W31	1	6	0,571	104,022	15,843	0,586	0,725	0,459	0,570	151,811	0,593	-31,619
	2	2	0,683	-214,832	19,578	0,491	0,509	0,475	0,683	-214,832	0,698	-263,938
	3	1	0,663	-205,945	16,837	0,543	0,503	0,579	0,663	-205,945	0,663	-205,945
	4	5	0,624	-161,266	21,676	0,486	0,488	0,484	0,615	-153,688	0,626	-181,885
	5	6	0,558	1104,247	14,663	0,592	0,456	0,717	0,558	1104,247	0,583	4,047
	6	3	0,691	-234,695	22,306	0,490	0,492	0,488	0,682	-206,589	0,697	-236,191
	7	6	0,604	-93,481	21,710	0,487	0,499	0,475	0,595	-44,986	0,612	-131,089
2WCU	1	1	0,615	-187,007	12,074	0,509	0,492	0,527	0,615	-187,007	0,615	-187,007
	2	5	0,712	-274,973	15,956	0,532	0,547	0,516	0,702	-236,148	0,714	-283,506
	3	3	0,650	-216,787	10,802	0,612	0,606	0,617	0,647	-187,361	0,652	-223,575
	4	19	0,525	922,691	15,506	0,691	0,897	0,475	0,521	1637,572	0,554	-164,731
2WLV	1	23	0,599	323,729	16,187	0,716	0,640	0,798	0,589	1599,916	0,615	-80,295
	2	7	0,652	-173,842	19,429	0,498	0,481	0,516	0,652	-173,842	0,706	-272,253
	3	1	0,670	-190,338	18,302	0,486	0,488	0,485	0,670	-190,338	0,670	-190,338
	4	6	0,624	-110,202	17,723	0,617	0,759	0,465	0,622	-103,206	0,630	-149,531
2XHF	1	21	0,649	-208,604	18,097	0,484	0,486	0,482	0,604	630,738	0,666	-241,920
2XOL	1	3	0,696	-234,309	17,200	0,486	0,499	0,474	0,695	-232,251	0,698	-250,505
	2	7	0,711	-407,703	2,715	0,689	0,668	0,711	0,704	-257,180	0,729	-440,963
	3	3	0,609	-187,262	17,892	0,533	0,584	0,480	0,604	-182,505	0,611	-214,154
	4	15	0,582	-138,960	20,157	0,568	0,656	0,477	0,565	403,791	0,600	-181,004
	5	1	0,655	-228,251	21,656	0,485	0,486	0,483	0,655	-228,251	0,655	-228,251
2YEM	1	6	0,646	-274,382	20,055	0,497	0,469	0,525	0,646	-230,741	0,651	-300,445
	2	19	0,607	-190,913	16,078	0,479	0,479	0,479	0,491	705,143	0,607	-190,913
	3	3	0,602	-150,191	19,447	0,505	0,490	0,520	0,602	-150,191	0,611	-220,738
2YVE	1	12	0,578	-131,399	14,877	0,490	0,511	0,471	0,578	-131,399	0,630	-462,821
2Z5D	1	9	0,601	-289,242	16,598	0,498	0,472	0,513	0,592	-176,025	0,602	-314,018
	2	7	0,567	-25,438	17,147	0,763	0,462	0,944	0,550	92,322	0,572	-67,050
	3	4	0,577	-105,656	12,140	0,768	0,459	0,958	0,577	-105,656	0,589	-165,020
2Z76	1	3	0,716	-256,472	9,333	0,589	0,577	0,600	0,702	-211,777	0,716	-256,472
	2	4	0,659	-100,985	12,299	0,531	0,627	0,436	0,659	-100,985	0,662	-185,625
	3	2	0,689	-186,650	13,029	0,527	0,582	0,472	0,689	-186,650	0,696	-195,928
	4	11	0,653	-33,813	14,606	0,536	0,623	0,450	0,648	1336,329	0,655	-85,244
	5	4	0,656	-85,584	15,128	0,543	0,623	0,463	0,648	1145,701	0,660	-128,324
2Z9D	1	2	0,681	-153,380	17,365	0,496	0,494	0,497	0,675	-136,732	0,681	-153,380
	2	1	0,665	-87,128	20,286	0,485	0,486	0,483	0,665	-87,128	0,665	-87,128
	3	7	0,657	-78,910	21,157	0,520	0,503	0,537	0,618	145,814	0,665	-81,144
2ZB9	1	17	0,649	-209,286	20,835	0,476	0,480	0,472	0,623	635,566	0,665	-256,577
2ZGL	1	15	0,637	930,435	10,929	0,636	0,764	0,503	0,637	930,435	0,658	-197,350
2ZOW	1	2	0,677	-155,120	16,872	0,476	0,474	0,478	0,670	-132,065	0,677	-155,120
	2	1	0,652	-130,912	15,830	0,496	0,466	0,525	0,652	-130,912	0,652	-130,912
	3	6	0,490	903,568	13,720	0,637	0,503	0,772	0,490	903,568	0,618	-128,059

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
	4	3	0,535	-53,559	13,235	0,547	0,463	0,632	0,535	-53,559	0,559	-85,027
	5	7	0,505	211,096	11,744	0,632	0,673	0,592	0,503	214,070	0,520	22,592
	6	5	0,501	329,369	13,108	0,706	0,683	0,728	0,501	329,369	0,518	46,218
2ZWM	1	1	0,611	-207,914	15,723	0,517	0,485	0,548	0,611	-207,914	0,611	-207,914
	2	3	0,554	-197,351	17,964	0,470	0,460	0,480	0,554	-186,599	0,593	-199,559
	3	1	0,537	-173,062	14,100	0,570	0,470	0,667	0,537	-173,062	0,537	-173,062
	4	3	0,500	-101,235	15,883	0,477	0,455	0,497	0,494	-15,552	0,500	-101,235
	5	3	0,500	-131,731	14,609	0,489	0,515	0,465	0,493	21,528	0,500	-131,731
	6	20	0,500	-111,651	14,576	0,477	0,500	0,455	0,479	935,436	0,502	-171,049
3AIA	1	4	0,613	-236,699	16,144	0,541	0,597	0,482	0,613	-236,699	0,620	-258,399
	2	6	0,578	-111,689	17,295	0,555	0,617	0,491	0,578	-111,689	0,585	-203,115
	3	5	0,525	24,885	4,487	0,753	0,733	0,773	0,525	24,885	0,551	-65,102
	4	7	0,515	3051,078	19,167	0,749	0,799	0,698	0,515	3051,078	0,552	-86,492
	5	1	0,566	-110,425	19,402	0,507	0,497	0,517	0,566	-110,425	0,566	-110,425
3CPQ	1	6	0,568	-374,161	16,787	0,467	0,467	0,468	0,542	-234,950	0,568	-374,161
	2	1	0,533	-200,113	16,038	0,451	0,445	0,457	0,533	-200,113	0,533	-200,113
	3	6	0,491	28,267	14,879	0,486	0,484	0,489	0,482	93,273	0,491	6,229
	4	11	0,505	-124,697	16,086	0,470	0,473	0,468	0,479	419,021	0,520	-144,262
3CQR	1	5	0,562	-65,878	20,539	0,528	0,575	0,483	0,559	29,300	0,573	-132,786
	2	11	0,555	245,360	19,407	0,541	0,602	0,483	0,551	1087,804	0,570	-104,852
3CT6	1	5	0,532	-44,320	15,474	0,649	0,513	0,787	0,532	-44,320	0,560	-231,789
	2	5	0,520	234,665	15,584	0,588	0,455	0,721	0,519	989,488	0,526	135,668
	3	10	0,565	-343,936	18,953	0,457	0,448	0,466	0,560	-245,337	0,568	-360,167
	4	4	0,529	1,269	14,048	0,592	0,448	0,736	0,529	1,269	0,547	-213,233
3CXK	1	5	0,596	-267,873	16,838	0,484	0,488	0,480	0,585	-239,934	0,597	-295,229
	2	8	0,556	36,015	18,593	0,457	0,459	0,455	0,539	756,463	0,556	36,015
	3	6	0,577	-199,286	15,814	0,516	0,588	0,451	0,572	-155,466	0,585	-222,460
	4	7	0,561	-69,825	19,637	0,476	0,480	0,471	0,557	28,597	0,562	-149,840
3D7A	1	4	0,566	-270,209	13,646	0,521	0,452	0,584	0,565	-167,532	0,570	-280,036
	2	10	0,535	-103,922	12,676	0,470	0,480	0,459	0,520	342,021	0,542	-147,031
	3	6	0,551	-153,581	15,977	0,474	0,472	0,475	0,515	1057,816	0,551	-153,581
3EVI	1	4	0,657	-238,970	7,229	0,656	0,630	0,687	0,621	-190,914	0,657	-238,970
	2	2	0,543	-98,846	9,769	0,502	0,542	0,455	0,543	-87,656	0,543	-98,846
	3	17	0,532	45,247	14,253	0,570	0,621	0,508	0,518	1394,079	0,559	-175,478
3F81	1	12	0,597	-21,312	16,863	0,474	0,474	0,473	0,592	785,170	0,620	-178,411
3FOU	1	4	0,716	-160,709	18,735	0,478	0,482	0,475	0,704	-119,981	0,716	-173,583
	2	1	0,598	-118,193	4,432	0,568	0,570	0,565	0,598	-118,193	0,598	-118,193
	3	18	0,550	-57,286	14,658	0,580	0,471	0,684	0,536	921,029	0,555	-111,281
3FQC	1	8	0,581	-198,352	20,252	0,483	0,483	0,483	0,546	949,035	0,581	-198,352
	2	6	0,615	-249,334	19,188	0,478	0,480	0,476	0,583	-216,658	0,619	-264,053
3FU1	1	4	0,572	-122,319	10,429	0,575	0,696	0,447	0,571	-55,323	0,583	-154,185
	2	3	0,665	-280,287	8,008	0,591	0,646	0,533	0,665	-280,287	0,671	-314,920
	3	8	0,571	-64,328	11,620	0,513	0,550	0,474	0,551	31,196	0,571	-64,328
	4	5	0,547	223,358	14,482	0,471	0,469	0,474	0,531	952,023	0,549	191,835
	5	8	0,611	-232,232	14,050	0,493	0,507	0,479	0,589	-172,312	0,622	-265,264
3G46	1	3	0,675	-159,933	11,916	0,572	0,555	0,590	0,673	-133,923	0,678	-188,333
	2	16	0,627	556,182	15,829	0,581	0,458	0,703	0,623	762,491	0,674	-141,131
3GLV	1	7	0,546	639,701	15,611	0,587	0,609	0,564	0,546	639,701	0,588	-150,468
	2	9	0,566	-42,764	14,600	0,522	0,558	0,486	0,547	622,611	0,566	-42,764
	3	5	0,545	1178,600	15,364	0,483	0,503	0,463	0,545	1178,600	0,557	-14,879
	4	3	0,590	-205,355	0,727	0,995	0,995	0,995	0,584	-132,406	0,596	-215,028
3GRN	1	2	0,618	-164,640	18,583	0,487	0,488	0,487	0,578	-148,113	0,618	-164,640
	2	4	0,459	63,293	19,648	0,467	0,471	0,462	0,445	863,709	0,459	63,293
	3	2	0,450	348,177	19,667	0,477	0,496	0,458	0,450	348,177	0,451	236,830
	4	10	0,497	-110,854	16,924	0,492	0,492	0,492	0,460	1,172	0,501	-130,600
3GWN	1	2	0,650	-201,677	19,221	0,477	0,465	0,488	0,650	-201,677	0,658	-214,121
	2	3	0,675	-248,649	21,307	0,482	0,488	0,476	0,654	-201,803	0,682	-267,977
	3	19	0,557	524,890	14,973	0,580	0,413	0,744	0,554	1479,227	0,574	-201,115
3HPE	1	13	0,616	-245,261	20,042	0,475	0,471	0,478	0,595	1096,381	0,616	-245,261

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
3HUP	1	3	0,448	-50,887	16,197	0,524	0,490	0,556	0,444	85,838	0,448	-50,887
	2	1	0,494	-184,013	19,711	0,460	0,460	0,459	0,494	-184,013	0,494	-184,013
	3	11	0,461	-146,081	19,229	0,472	0,480	0,464	0,433	1184,705	0,495	-206,106
	4	5	0,443	163,397	18,098	0,454	0,466	0,443	0,434	371,624	0,443	163,397
3HV2	1	6	0,490	-155,011	18,519	0,473	0,473	0,474	0,485	-90,863	0,490	-155,011
	2	3	0,480	5,697	15,257	0,465	0,461	0,469	0,477	98,471	0,480	-15,807
	3	3	0,496	-157,820	18,095	0,469	0,469	0,469	0,487	-93,585	0,496	-157,820
	4	14	0,482	-80,549	17,595	0,461	0,457	0,465	0,462	1211,613	0,482	-80,549
3I4S	1	73	0,566	-485,767	0,499	0,989	0,991	0,986	0,494	1974,796	0,570	-517,163
3IA1	1	2	0,587	-194,374	20,070	0,476	0,480	0,472	0,587	-194,374	0,589	-195,952
	2	8	0,564	-16,895	14,453	0,571	0,476	0,672	0,555	716,782	0,565	-33,843
	3	10	0,583	-170,155	19,215	0,472	0,467	0,476	0,555	830,902	0,583	-170,155
	4	15	0,565	-36,392	14,466	0,563	0,467	0,664	0,554	1295,349	0,568	-90,472
3IIR	1	3	0,680	-132,265	9,137	0,628	0,562	0,687	0,680	-132,265	0,686	-212,451
	2	12	0,671	418,206	17,919	0,660	0,550	0,758	0,669	579,747	0,680	-157,472
	3	2	0,682	-177,444	9,254	0,663	0,617	0,704	0,682	-177,444	0,688	-214,257
3IQ3	1	4	0,516	-205,722	19,509	0,450	0,446	0,455	0,511	-182,613	0,521	-220,746
	2	1	0,461	808,168	15,655	0,489	0,387	0,591	0,461	808,168	0,461	808,168
	3	9	0,482	-172,712	18,890	0,457	0,455	0,459	0,463	392,621	0,491	-177,681
3IX3	1	9	0,581	-91,408	17,596	0,499	0,538	0,459	0,578	-45,733	0,583	-128,789
	2	1	0,692	-193,268	22,636	0,488	0,486	0,490	0,692	-193,268	0,692	-193,268
	3	4	0,599	-175,066	20,604	0,490	0,490	0,490	0,591	-130,889	0,599	-175,066
	4	9	0,567	442,978	17,381	0,504	0,580	0,428	0,561	1510,678	0,573	-31,198
3K3K	1	22	0,546	-275,651	16,170	0,629	0,783	0,486	0,527	512,564	0,562	-367,488
3K9U	1	8	0,645	-186,084	20,072	0,479	0,477	0,481	0,643	-181,660	0,650	-264,324
	2	4	0,640	-153,107	20,911	0,492	0,494	0,490	0,620	183,173	0,640	-153,107
	3	8	0,640	-114,412	20,207	0,492	0,494	0,490	0,628	-31,742	0,640	-114,412
3L18	1	8	0,611	-162,553	17,569	0,484	0,484	0,483	0,602	-11,419	0,611	-162,553
	2	2	0,591	906,023	14,665	0,548	0,519	0,572	0,591	906,023	0,593	407,875
	3	2	0,630	-187,368	20,416	0,475	0,477	0,474	0,628	-175,484	0,630	-187,368
	4	4	0,596	31,579	18,678	0,461	0,464	0,457	0,593	659,629	0,625	-162,801
	5	2	0,673	-247,664	15,671	0,485	0,487	0,483	0,664	-246,623	0,673	-247,664
	6	3	0,633	-238,087	18,192	0,475	0,477	0,474	0,633	-195,138	0,633	-238,087
	7	8	0,593	562,233	15,869	0,526	0,536	0,516	0,593	562,233	0,609	-158,920
3LB2	1	2	0,729	-130,420	15,462	0,560	0,557	0,566	0,726	-126,924	0,729	-130,420
	2	2	0,602	685,167	17,965	0,452	0,454	0,451	0,602	700,573	0,602	685,167
	3	10	0,621	-75,407	19,807	0,471	0,473	0,470	0,601	866,835	0,624	-110,661
3LBB	1	2	0,532	-169,818	16,094	0,501	0,435	0,572	0,532	-169,818	0,559	-226,281
	2	1	0,652	-236,251	17,386	0,547	0,624	0,463	0,652	-236,251	0,652	-236,251
	3	12	0,505	76,245	16,910	0,556	0,435	0,686	0,490	740,563	0,538	-214,273
	4	5	0,500	489,230	17,072	0,600	0,717	0,473	0,499	521,884	0,514	-64,093
3LYN	1	6	0,665	-182,814	17,555	0,524	0,580	0,468	0,646	-109,864	0,665	-182,814
	2	14	0,623	-37,873	20,116	0,470	0,462	0,477	0,589	402,106	0,638	-61,726
3MGK	1	13	0,517	603,278	11,451	0,726	0,788	0,665	0,517	603,278	0,534	-332,787
3N4K	1	2	0,643	-104,210	22,378	0,479	0,481	0,477	0,643	-104,210	0,651	-109,185
	2	4	0,541	-45,251	18,383	0,560	0,455	0,664	0,541	-23,269	0,550	-101,311
	3	7	0,525	257,548	13,795	0,624	0,514	0,731	0,516	495,124	0,540	-12,424
3N7H	1	3	0,639	-227,908	13,357	0,511	0,438	0,584	0,639	-227,908	0,645	-241,788
	2	10	0,611	-181,237	16,863	0,475	0,482	0,469	0,531	1096,514	0,613	-201,745
	3	5	0,545	114,524	18,421	0,478	0,478	0,478	0,545	114,524	0,557	-45,439
	4	2	0,559	-48,187	17,312	0,464	0,464	0,464	0,559	-48,187	0,565	-53,178
	5	10	0,571	-72,245	17,073	0,475	0,478	0,473	0,540	434,990	0,582	-121,800
3N8E	1	22	0,447	-101,841	18,126	0,503	0,535	0,470	0,424	1834,675	0,451	-124,583
	2	5	0,551	-206,221	18,488	0,501	0,532	0,470	0,507	-125,117	0,551	-206,221
3NBC	1	1	0,704	-240,865	19,781	0,474	0,466	0,481	0,704	-240,865	0,704	-240,865
	2	7	0,632	-183,452	17,701	0,481	0,478	0,485	0,622	-113,491	0,643	-232,735
	3	14	0,626	-125,982	19,219	0,476	0,470	0,481	0,593	675,715	0,626	-125,982
3OCP	1	6	0,557	-192,437	17,300	0,470	0,470	0,470	0,557	-192,437	0,576	-330,537
	2	15	0,519	-88,615	15,646	0,483	0,478	0,487	0,504	904,656	0,540	-138,333

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
3P9X	1	6	0,586	-170,073	16,418	0,580	0,556	0,603	0,582	58,636	0,592	-252,152
	2	4	0,655	-302,584	12,044	0,478	0,478	0,478	0,648	-291,520	0,658	-354,498
	3	3	0,604	-252,338	17,158	0,493	0,512	0,475	0,604	-252,338	0,617	-288,192
	4	5	0,579	805,195	17,343	0,631	0,462	0,800	0,576	937,124	0,611	-252,830
	5	1	0,580	304,567	14,504	0,571	0,464	0,678	0,580	304,567	0,580	304,567
	6	2	0,579	597,687	15,252	0,581	0,475	0,686	0,579	675,615	0,579	597,687
3PH4	1	2	0,612	-249,129	16,050	0,553	0,532	0,574	0,611	-190,045	0,612	-249,129
	2	11	0,467	566,994	4,574	0,742	0,752	0,732	0,456	2796,375	0,557	-114,237
	3	6	0,465	910,192	3,648	0,698	0,703	0,693	0,465	910,192	0,510	-57,107
	4	7	0,543	-94,745	18,064	0,542	0,595	0,490	0,543	-94,745	0,577	-178,341
3QU1	1	6	0,669	-374,779	15,716	0,499	0,496	0,501	0,653	-267,569	0,669	-374,779
	2	5	0,632	-262,244	18,310	0,529	0,460	0,594	0,621	-221,238	0,634	-265,178
	3	5	0,663	-340,936	20,111	0,480	0,483	0,477	0,663	-340,936	0,674	-407,088
	4	14	0,537	-74,631	23,080	0,483	0,480	0,487	0,518	788,965	0,543	-112,194
	5	7	0,551	-128,013	18,025	0,555	0,474	0,632	0,549	-120,885	0,561	-216,718
3RD3	1	1	0,717	-246,875	15,449	0,539	0,481	0,600	0,717	-246,875	0,717	-246,875
	2	1	0,694	-179,467	17,784	0,507	0,477	0,537	0,694	-179,467	0,694	-179,467
	3	5	0,670	405,678	14,229	0,511	0,451	0,571	0,670	405,678	0,672	83,699
	4	7	0,680	-169,513	23,537	0,522	0,477	0,568	0,673	51,397	0,680	-169,513
3RFB	1	3	0,556	-182,050	17,076	0,533	0,477	0,586	0,550	-163,333	0,561	-203,853
	2	2	0,559	-198,681	18,864	0,474	0,446	0,501	0,559	-198,681	0,561	-270,581
	3	20	0,530	77,572	18,093	0,496	0,549	0,447	0,515	1778,533	0,549	-102,998
	4	1	0,630	-289,528	18,782	0,485	0,469	0,501	0,630	-289,528	0,630	-289,528
3RHC	1	1	0,556	-219,992	14,063	0,474	0,485	0,464	0,556	-219,992	0,556	-219,992
	2	11	0,533	-19,856	15,759	0,706	0,975	0,438	0,518	642,855	0,533	-19,856
	3	2	0,546	-179,510	16,082	0,469	0,480	0,458	0,544	-120,224	0,546	-179,510
	4	4	0,541	-95,919	8,262	0,797	0,735	0,859	0,535	-31,222	0,544	-121,418
3RQ3	1	4	0,641	-113,350	17,238	0,558	0,660	0,451	0,641	-113,350	0,660	-155,596
	2	14	0,520	535,526	19,225	0,526	0,474	0,581	0,507	1077,754	0,614	-102,536
3SLZ	1	3	0,654	-180,992	16,129	0,553	0,661	0,438	0,654	-173,031	0,656	-209,891
	2	3	0,653	-172,525	16,373	0,521	0,465	0,580	0,647	-167,193	0,653	-172,525
	3	15	0,623	15,380	13,660	0,542	0,430	0,663	0,609	743,062	0,644	-149,765
3SZJ	1	20	0,675	327,185	16,124	0,750	0,780	0,719	0,669	1176,082	0,761	-150,010
3TRF	1	23	0,572	346,425	16,986	0,605	0,448	0,739	0,565	1351,028	0,630	-289,398
3TW2	1	6	0,577	-343,804	5,091	0,637	0,639	0,635	0,576	-276,427	0,597	-440,694
	2	6	0,554	-261,395	11,965	0,579	0,585	0,574	0,541	-150,098	0,554	-261,395
	3	3	0,533	-101,358	14,659	0,530	0,439	0,622	0,533	0,953	0,541	-150,145
	4	3	0,539	-107,499	15,870	0,567	0,507	0,628	0,539	-107,499	0,540	-138,340
	5	7	0,515	124,616	15,497	0,605	0,541	0,669	0,506	301,618	0,532	25,876
3UJM	1	3	0,760	-234,024	21,575	0,476	0,473	0,479	0,760	-234,024	0,765	-302,922
	2	3	0,669	-133,486	13,496	0,563	0,661	0,458	0,669	-133,486	0,671	-179,479
	3	1	0,694	-214,647	15,354	0,511	0,468	0,558	0,694	-214,647	0,694	-214,647
	4	3	0,566	1287,967	16,281	0,699	0,713	0,683	0,566	1287,967	0,590	25,627
	5	15	0,582	179,633	8,481	0,571	0,620	0,519	0,578	411,246	0,646	-122,945
3UMZ	1	6	0,533	-31,403	11,504	0,534	0,651	0,437	0,528	51,109	0,538	-82,946
	2	8	0,612	-199,900	12,228	0,536	0,651	0,442	0,601	-134,985	0,613	-208,508
	3	7	0,522	150,374	15,364	0,493	0,568	0,432	0,503	1295,478	0,522	150,374
	4	1	0,550	-99,972	12,329	0,466	0,474	0,458	0,550	-99,972	0,550	-99,972
	5	1	0,563	-126,182	13,026	0,469	0,469	0,468	0,563	-126,182	0,563	-126,182
	6	3	0,523	59,624	15,244	0,501	0,579	0,437	0,523	59,624	0,528	-7,779
3V6G	1	10	0,631	218,234	23,252	0,570	0,664	0,479	0,629	805,899	0,643	-213,561
3VRC	1	5	0,603	-231,971	14,939	0,560	0,654	0,461	0,603	-231,971	0,606	-281,916
	2	3	0,510	-86,637	13,690	0,584	0,659	0,505	0,508	-85,609	0,510	-97,183
	3	12	0,496	74,548	12,939	0,623	0,772	0,466	0,483	1145,673	0,540	-223,010
4AUU	1	4	0,551	388,102	13,718	0,626	0,445	0,806	0,548	607,408	0,565	21,994
	2	4	0,582	-29,760	14,679	0,497	0,497	0,497	0,573	-18,022	0,596	-48,311
	3	8	0,557	26,011	14,274	0,650	0,455	0,844	0,538	720,010	0,583	-40,402
	4	8	0,547	711,633	13,156	0,582	0,445	0,719	0,547	711,633	0,659	-120,365
4DF0	1	6	0,659	-373,741	6,198	0,630	0,638	0,621	0,656	-240,408	0,667	-390,870

PDB	Grupa		Kryteria			Ocena			Min. RD		Min. energii	
ID	I	N	RD	Energia	RMSD	C	A	B	RD	Energia	RD	Energia
	2	12	0,619	295,379	17,689	0,605	0,755	0,442	0,619	295,379	0,649	-173,442
4E7P	1	3	0,591	-191,979	20,596	0,460	0,464	0,456	0,580	-157,030	0,591	-191,979
	2	12	0,549	81,574	17,014	0,462	0,464	0,460	0,531	654,701	0,557	-86,081
	3	6	0,559	-88,205	16,098	0,547	0,468	0,639	0,547	154,392	0,578	-143,024
4EC7	1	6	0,505	-45,120	16,497	0,488	0,488	0,488	0,496	9,465	0,505	-45,120
	2	7	0,590	-173,512	17,552	0,499	0,520	0,476	0,590	-165,683	0,602	-220,325
	3	4	0,506	-74,711	25,592	0,484	0,476	0,494	0,506	-74,711	0,509	-118,822
	4	1	0,522	-160,313	24,265	0,494	0,520	0,465	0,522	-160,313	0,522	-160,313
4EP4	1	6	0,548	-127,590	16,414	0,492	0,472	0,512	0,546	-109,309	0,551	-142,663
	2	4	0,633	-154,041	16,522	0,519	0,552	0,483	0,628	-148,717	0,648	-205,150
	3	14	0,501	2588,012	18,364	0,550	0,465	0,642	0,501	2588,012	0,535	-108,484

C. Oprogramowanie

Dodatek C zawiera opis utworzonego i rozwijanego przez Autora rozprawy oprogramowania, przy pomocy którego uzyskano wyniki przedstawione w tej pracy.

Wszystkie użyte narzędzia zostały napisane w języku Python w zgodności z jego standardowym interpreterem – CPythonem[1], przy użyciu zaawansowanej powłoki IPython[2] [420], a także wymienionych poniżej bibliotek rozszerzeń.

Na końcu rozdziału poruszone są kwestie związane z wykonaniem równoległym algorytmu MOSF oraz wyniki przeprowadzonych testów.

C.1. Biblioteki rozszerzeń

Programy, których użyto w badaniach związanych z tematem rozprawy oraz do prezentacji otrzymanych wyników korzystają z następujących bibliotek zewnętrznych rozszerzających funkcjonalność języka Python:

C.1.1. NumPy

NumPy[3] [421] jest biblioteką dostarczającą klasę homogenicznych, wielowymiarowych tablic. Pozostałe funkcje przez nią oferowane znajdują zastosowanie w takich dziedzinach jak algebra liniowa (przy użyciu LAPACK[4]), szybka transformacja Fouriera (przy użyciu FFTPACK[5]) oraz generacja liczb pseudolosowych. NumPy stanowi obecnie *de facto* standardowe narzędzie do obliczeń numerycznych w Pythonie, głównie ze względu na powszechną użyteczność oferowanej przez nie funkcjonalności, łatwość integracji w programach oraz zdecydowanie krótszy czas wykonania w porównaniu z kodem korzystającym wyłącznie z funkcji biblioteki standardowej.

[1] http://www.python.org
[2] http://ipython.org
[3] http://www.numpy.org
[4] http://www.netlib.org/lapack
[5] http://www.netlib.org/fftpack

Na NumPy opierają się wszystkie metody numeryczne stosowane przez Autora rozprawy, jak również większość przedstawionych tu bibliotek rozszerzeń.

C.1.2. SciPy

SciPy[6] [421] jest biblioteką zawierającą kolekcję różnorodnych klas i funkcji przydatnych w naukach ścisłych, a konkretnie w takich dziedzinach jak: analiza skupień, całkowanie, interpolacja, optymalizacja, obsługa macierzy rzadkich, przetwarzanie sygnałów oraz geometria obliczeniowa. Stanowi ona darmowy i otwarty odpowiednik komercyjnego oprogramowania oferującego podobne możliwości. SciPy korzysta przede wszystkim z NumPy, aczkolwiek część modułów jest napisana bezpośrednio w C, co pozwala na jeszcze szybsze ich wykonywanie, choć wiążące się jednak z ograniczeniami w tworzeniu klas potomnych dostarczanych przez nie klas.

W rozprawie zostały użyte moduły analizy skupień k-średnich oraz poszukiwania najbliższych sąsiadów przy pomocy struktury drzewa k-d.

C.1.3. Matplotlib

Matplotlib[7] [422] jest biblioteką graficzną służącą do programistycznego tworzenia wysokiej jakości wykresów 2D i 3D, ich animacji oraz osadzania ich w graficznych interfejsach użytkownika. Wymaga NumPy. Jakość na poziomie publikacji jest zapewniana przez napisaną w C++ bibliotekę Anti-Grain Geometry,[8] zdolną do renderowania podpikselowego i z antyaliasingiem w wysokiej rozdzielczości. Produkowane przez Matplotlib rysunki mogą być zapisywane zarówno w formatach rastrowych (PNG, JPG, TIFF) jak i wektorowych (EPS, PDF, SVG). Biblioteka ta oferuje również bezpośredni rendering wyrażeń matematycznych. Alternatywnie może być w tym celu wykorzystany LaTeX[9] (jego użycie jest w pełni zautomatyzowane, choć da się zmieniać preambułę dokumentu, na przykład aktywując obsługę różnych języków). Drugie z tych rozwiązań pozwala na składanie dokumentów o profesjonalnym wyglądzie, których elementy tworzą wizualnie spójną całość.

Wszystkie wykresy umieszczone w tekście rozprawy zostały utworzone przy użyciu tej biblioteki, wykorzystując LaTeX i zapis w formacie wektorowym, dzięki czemu, opisy ich osi stosują ten sam krój fontu co treść tekstu pracy.

[6] http://www.scipy.org
[7] http://matplotlib.org
[8] http://agg.sourceforge.net
[9] http://www.tug.org/texlive

C.1.4. PyGMO

PyGMO[10] (python parallel global multiobjective optimizer) [423] jest biblioteką zorientowaną na rozwiązywanie ciągłych i dyskretnych, jedno- i wielokryterialnych problemów optymalizacyjnych z ograniczeniami. Stanowi ona interfejs pomiędzy Pythonem a napisaną w C++ biblioteką PaGMO (parallel global multiobjective optimizer), oryginalnie stworzoną przez Europejską Agencję Kosmiczną. Wspólnie, oferują one dostęp do kilkunastu powszechnie znanych metod optymalizacyjnych.

W rozprawie została użyta implementacja algorytmów NSGA-II i NSPSO.

C.1.5. NetworkX

NetworkX[11] [424] jest biblioteką dostarczającą klasy, funkcje i algorytmy związane z teorią grafów. Umożliwia ona ich tworzenie, analizę oraz prezentację. Do realizacji ostatniego z tych zadań niezbędny jest Matplotlib.

NetworkX został użyty w rozprawie do utworzenia wykresów topologii roju cząstek znajdujących się na rysunku 2.8.

C.1.6. PyMOL

PyMOL[12] jest programem służącym do trójwymiarowej wizualizacji modeli białek i innych cząsteczek, którego interpreter poleceń, interfejs użytkownika oraz wtyczki są napisane w Pythonie, natomiast silnik graficzny w C. Ponieważ silnik ten działa w oparciu o język shaderów, wymaga obecności w systemie komputerowym akceleratora graficznego obsługującego GLSL, a więc OpenGL w wersji 2.0 lub wyższej. Najważniejsze cechy PyMOLa, wyróżniające go na tle podobnych narzędzi, to zdolność do wielowątkowego renderowania modeli w wysokiej rozdzielczości, z antyaliasingiem i efektami przezroczystości, a także możliwość ich edycji, na przykład poprzez zmiany wartości kątów dwuściennych. Do renderingu służy w tym programie wbudowany algorytm śledzenia promieni. Posiada on także możliwość eksportu modeli do plików w formacie VRML2, które po konwersji przez osobne narzędzia do formatu STL, nadają się do wykorzystania przez drukarki przestrzenne [425].

PyMOL został użyty w rozprawie do utworzenia rysunków struktur białkowych i wykonania dopasowania strukturalnego przy pomocy algorytmu CE.

[10] http://esa.github.io/pygmo/index.html
[11] http://networkx.github.io
[12] http://pymol.org

C.2. Biblioteka modułów

Oprogramowanie, dzięki któremu otrzymano wyniki przedstawione w niniejszej rozprawie dzieli się na bibliotekę klas i funkcji oraz zbiór korzystających z nich programów. Wszystkie moduły z tej biblioteki zostały opracowane i są samodzielnie rozwijane przez Autora rozprawy w ramach prowadzonej pracy badawczej oraz osobistych zainteresowań. Poniżej znajduje się ich krótki opis, pomijający szczegóły języka programowania. Należy podkreślić, że wymienione są tu tylko te moduły, które dotyczą tematu rozprawy. Ich kolejność jest następująca: najpierw użytkowe (wspomagające), następnie związane z optymalizacją, a na końcu – z bioinformatyką.

C.2.1. Moduł transform

Moduł TRANSFORM zawiera klasy dokonujące przekształceń przy pomocy macierzy oraz inne algorytmy z dziedziny geometrii obliczeniowej.

Klasa TRANSFORM
Klasa przekształcająca dane zapisane w macierzy o rozmiarze $n \times d$ przy pomocy wskazanej macierzy transformacji. Użytkownik decyduje, czy mnożenie przez tę macierz ma być lewo- czy prawostronne. Dzięki przeciążaniu operatorów, możliwe jest również proste składanie wielu przekształceń w jedno.

Klasa ROTATE2D	klasa bazowa: TRANSFORM
Klasa obracająca w dwóch wymiarach wokół początku układu współrzędnych o θ radianów zgodnie z regułą prawej dłoni.	

Klasa ROTATE3D	klasa bazowa: TRANSFORM
Klasa obracająca w trzech wymiarach wokół wektora v o θ radianów zgodnie z regułą prawej dłoni.	

Klasa REFLECT	klasa bazowa: TRANSFORM
Klasa odbijająca wektor s na wektor t przy pomocy transformacji Householdera.	

Klasa REFLECTROTATE	klasa bazowa: TRANSFORM
Klasa obracająca wektor s na wektor t w wyznaczonej przez nie hiperpłaszczyźnie przy pomocy złożenia dwóch transformacji Householdera i dbająca o prawidłową obsługę wszystkich sytuacji jakie mogą się przy tym zdarzyć.	

Klasa RMSD	klasa bazowa: TRANSFORM

Klasa implementująca algorytm Kabscha – obliczająca macierze obrotu i translacji minimalizujące wartość RMSD dwóch równolicznych zbiorów punktów.

Klasa PCA	klasa bazowa: TRANSFORM

Klasa wykonująca analizę składowych głównych przy pomocy SVD. Możliwe jest rzutowanie na dowolną ich liczbę, przywracanie danych do oryginalnej przestrzeni, a także obliczanie ile wektorów cech jest potrzebne do uzyskania oczekiwanego ułamka całkowitej wariancji danych.

Moduł TRANSFORM udostępnia również następujące funkcje:

ANGLE	oblicza kąt płaski utworzony przez trzy punkty
DIHEDRAL	oblicza kąt dwuścienny utworzony przez cztery punkty
LINE_PROJECTION	oblicza wektor rzutu prostopadłego punktu na prostą
EULER_ANGLES	oblicza kąty Eulera na podstawie macierzy obrotu
RMSD	oblicza wartość RMSD dla dwóch zbiorów punktów

C.2.2. Moduł roc

Moduł ROC służy do obliczania tabel kontyngencji i krzywych ROC.

Klasa ROC

Klasa obliczająca tablice kontyngencji porównania rozkładu-wzorca z dowolną liczbą wyników klasyfikacji binarnej. Na ich podstawie są przez nią obliczane wartości TPR i FPR oraz pola pod krzywymi ROC.

C.2.3. Moduł mpb

Moduł MPB zawiera implementację testu ruchomych wierzchołków.

Klasa MPB

Klasa implementująca generator MPB. Użytkownik ma możliwość ustalania liczby wierzchołków, funkcji ich kształtu oraz funkcji krajobrazu bazowego. Właściwości wierzchołków mogą być inicjalizowane w sposób losowy lub podawane bezpośrednio przez użytkownika. Ta sama zasada dotyczy zmiany krajobrazu wartości wygenerowanego kryterium: może następować samoczynnie (po obliczeniu jego wartości określoną liczbę razy), lub na żądanie użytkownika.

Moduł MPB udostępnia również cztery funkcje kształtu wierzchołków zaproponowane przez Brankego: function1 (sharp), cone, hilly i twin, a także funkcję gauss. Funkcje hilly i twin zostały tu zmodyfikowane w celu nadania im uniwersalności poprzez zastąpienie stosowanych w nich stałych zależnością od prędkości. Dzięki temu, nim szybciej dany wierzchołek się porusza, tym bardziej jego krajobraz się zaburza (hilly), lub bardziej odsuwają się od siebie jego dwa piki (twin).

C.2.4. Moduł problem

Moduł PROBLEM zawiera różne testowe problemy optymalizacyjne.

Klasa PROBLEM

Klasa – szablon dla problemów optymalizacyjnych. Pozwala na definiowanie dowolnej liczby kryteriów, ich dziedzin oraz funkcji ograniczeń.

Moduł PROBLEM udostępnia obecnie następujące funkcje wielokryterialne: Belegundu, Binh 1, Binh 2, Binh 3, Binh 4, Deb 1, Deb 2, Deb 3, Fonseca 1, Fonseca 2, Jimenez, Kita, Kursawe, Laumanns, Lis, Murata, Obayashi, Okabe 1, Okabe 2, Osyczka 1, Osyczka 2, Poloni, Quagliarella, Rendon 1, Rendon 2, Schaffer 1, Schaffer 2, Srinivas, Tamaki, Tanaka, Viennet 1, Viennet 2, Viennet 3 i Viennet 4. Każda z tych funkcji jest klasą pochodną klasy PROBLEM. Znajduje się tu również klasa generująca losowe kryteria przy pomocy MPB.

C.2.5. Moduł pso

Moduł PSO zawiera implementację klasycznego algorytmu optymalizacji rojem cząstek oraz jego czterech popularnych topologii.

Klasa PSO

Klasa implementująca algorytm PSO. Właściwości cząstek mogą być inicjalizowane w sposób losowy lub podawane bezpośrednio przez użytkownika. Funkcje ograniczeń są obsługiwane przy pomocy strategii turniejowej Deba. Klasa ta dopuszcza również możliwość wskazywania zewnętrznych liderów cząstek oraz definiuje parametr kontrolujący siłę ich przyciągania. Umożliwia to jej wykorzystanie przez algorytm MOSF.

Topologie roju są klasami pochodnymi klasy PSO: PSOFULL (grafu pełnego), PSORING (pierścienia), PSOSTAR (gwiazdy) oraz PSOGRID (von Neumanna).

C.2.6. Moduł pareto

Moduł PARETO służy do obsługi danych należących do zbioru i frontu Pareto.

Klasa ParetoFront

Klasa przechowująca dane zbioru i frontu Pareto w dwóch tabelach o rozmiarach $n \times d$ i $n \times k$. Dopuszczalna jest obecność powtórzeń – za zapewnienie unikalności danych odpowiada użytkownik. Klasa umożliwia następujące działania: dodawanie i usuwanie elementów, sumę z innymi instancjami, a także wybieranie niezdominowanego podzbioru wśród swojej zawartości lub niezdominowanego przez elementy innych instancji.

Moduł PARETO udostępnia również następujące funkcje:

GENERATE	generuje prostokątną siatkę punktów o wskazanej gęstości w różnych wymiarach przestrzeni
CALCULATE	oblicza wartości wskazanych kryteriów i funkcji ograniczeń dla danego zbioru punktów
SAVE	zapisuje instancję klasy ParetoFront w pliku binarnym
LOAD	odczytuje instancję klasy ParetoFront z pliku binarnego

C.2.7. Moduł mosf

Moduł MOSF zawiera implementację algorytmu MOSF.

Klasa SwarmFamily — klasa bazowa: ParetoFront

Klasa reprezentująca rodzinę rojów cząstek, umożliwiająca wykonywane przy ich pomocy optymalizacji wielokryterialnej. Użytkownik przekazuje do instancji zainicjalizowane roje, parametry oraz początkową zawartość archiwum, które podczas aktualizacji jest wypełniane odnalezionymi rozwiązaniami niezdominowanymi. Przycinanie jego zawartości następuje automatycznie, ale może być również uruchomione na żądanie użytkownika. Wywołanie metody podziału tworzy na podstawie bieżącej instancji instancję klasy SwarmFamilies.

Klasa SwarmFamilies

Klasa reprezentująca grupę rodzin rojów cząstek. Umożliwia szybkie wywoływanie metod należących do niej instancji klasy SwarmFamily, a także ich łączenie. Czynność ta może być wykonywana automatyczne lub na żądanie użytkownika. Zawartość archiwów wszystkich rodzin stanowi wynik optymalizacji wielokryterialnej zwracany przez algorytm MOSF.

C.2.8. Moduł pdb

Moduł PDB służy do obsługi klasycznego formatu PDB oraz do pobierania plików z serwera FTP należącego do wwPDB. Autor rozprawy wzorował się podczas jego tworzenia na bibliotece Biopython[13] [426], z której zaczerpnął ideę drzewiastej reprezentacji elementów struktury cząsteczek. Wynikają stąd zauważalne podobieństwa w nazewnictwie klas, aczkolwiek ich implementacja jest zupełnie inna. Utworzenie od podstaw własnego modułu zostało podyktowane względami praktycznymi: chęcią uniezależnienia się od rozbudowanej biblioteki, centralizacją kodu w pojedynczym pliku oraz usprawnieniem obsługi struktur składających się z wielu modeli. Dodatkowo, moduł ten nie wymaga NumPy, co zwiększa jego przenośność.

Klasa ENTITY
Klasa niezwiązana bezpośrednio z formatem PDB, dostarczająca mechanizmy umożliwiające tworzenie hierarchii instancji jej klas pochodnych. Każda instancja posiada wskaźnik swojego rodzica i kolekcję wskaźników potomków, a także funkcje umożliwiające poruszanie się pomiędzy nimi (w górę i w dół drzewa) oraz ich szybkie wyszukiwanie na podstawie ich identyfikatorów. Klasa ENTITY nie posiada własnych instancji. Zamiast tego, dziedziczą z niej pozostałe klasy, reprezentujące różne poziomy strukturalne cząsteczki.

Klasa ATOM	klasa bazowa: ENTITY
Klasa reprezentująca atom PDB. Identyfikatorami jej instancji są pary (nazwa, położenie alternatywne), rodzicami – reszty, natomiast nie posiadają one określonego typu potomków. Klasa ta umożliwia zapis do pliku pojedynczego rekordu ATOM lub HETATM. Potrzebne do tego dane reszty i łańcucha są uzyskiwane automatycznie lub mogą być podane przez użytkownika.	

Klasa RESIDUE	klasa bazowa: ENTITY
Klasa reprezentująca resztę PDB. Identyfikatorami jej instancji są trójki (nazwa, numer, kod insercji), rodzicami – łańcuchy, a potomkami – atomy. Klasa ta decyduje o identyfikatorze rekordu PDB (ATOM / HETATM) oraz umożliwia wykrywanie typu cząsteczki: aminokwas (P), kwas nukleinowy (N), ligand (L), jon (I) lub woda (S). Za jony są uznawane wszystkie ligandy inne niż woda, zbudowane tylko z jednego atomu, którego nazwa składa się z dwóch znaków. Choć nie jest to bezbłędny sposób klasyfikacji, jego czułość w przypadku danych w pełni zgodnych z formatem PDB jest bardzo wysoka.	

[13] http://biopython.org

Klasa Chain — klasa bazowa: Entity

Klasa reprezentująca łańcuch PDB. Identyfikatorami jej instancji są pojedyncze litery, rodzicami – modele, a potomkami – reszty. Podobnie jak Residue, klasa ta umożliwia wykrywanie typu cząsteczki (białko, kwas nukleinowy, itd.). Ponieważ jej instancja może zawierać reszty różnego typu, stosowany jest w tym celu następujący priorytet ich obecności: P $\prec$ N $\prec$ L $\prec$ I $\prec$ S. Mieszanie reszt typu P i N w tym samym łańcuchu jest niedozwolone przez RCSB.

Klasa Model — klasa bazowa: Entity

Klasa reprezentująca model PDB. Identyfikatorami jej instancji są liczby naturalne, rodzicami – struktury, a potomkami – łańcuchy. W odróżnieniu od swoich potomków, klasa ta potrafi analizować pojedyncze rekordy atom / hetatm i na nich podstawie tworzyć lub aktualizować podległe im w hierarchii instancje.

Klasa Structure — klasa bazowa: Entity

Klasa reprezentująca strukturę PDB. Identyfikatorami jej instancji są ciągi znaków (typowo nazwy plików), potomkami – modele, natomiast nie posiadają one określonego typu rodziców. Klasa ta odtwarza kompletną reprezentację danych zapisanych w pliku w formacie PDB. Rekordy nienależące do sekcji współrzędnych są umieszczane w instancji klasy Header. Istnieje również możliwość wczytania wszystkich albo tylko wybranego podzbioru modeli z pliku PDB, co znacznie przyspiesza czas jego przetwarzania.

Klasa Header

Klasa reprezentująca nagłówek PDB. W jej instancjach przechowywane są rekordy, które nie należą do sekcji współrzędnych formatu PDB. Wszystkie, za wyjątkiem remark, są parsowane i prezentowane w sposób przystępny dla użytkownika oraz zgodny z właściwościami pozostałych klas. Klasa ta nie posiada obecnie możliwości zapisu tych rekordów z powrotem w formacie PDB.

Klasa PDBFTP

Klasa umożliwiająca pobieranie plików w formacie PDB z serwera FTP należącego do wwPDB, lub innego, posiadającego identyczną strukturę katalogową. Dostęp do danych jest uzyskiwany z poszanowaniem zasobów komputerowych. Pliki są pobierane w postaci skompresowanej (gzip), ale mogą być rozpakowywane w pamięci przed ich zapisem w lokalnym systemie plików. Jeżeli białko o danym identyfikatorze zostało zastąpione w bazie PDB innym, w zależności od decyzji użytkownika, w jego miejscu może być pobrana nowsza struktura. Informacje na ten temat są uzyskiwane z osobnego pliku na serwerze (obsolete.dat) i umieszczane w pamięci podręcznej. Dzięki temu, nie ma potrzeby pobierania struktur tylko po to, aby dowiedzieć się, że są już nieaktualne.

Moduł PDB udostępnia również następujące funkcje:

ALTLOC	pozostawia najczęściej zajmowane położenia alternatywne przez wskazane atomy i usuwa pozostałe (pierwsze w przypadku ich identycznego prawdopodobieństwa)
MODRES	zmienia nazwy reszt zmodyfikowanych na nazwy ich pierwowzorów zgodnie z informacją z rekordów MODRES
FOLD	„zwija" elementy struktury do wskazanego poziomu nadrzędnego, na przykład atomy do łańcuchów
UNFOLD	„rozwija" elementy struktury do wskazanego poziomu podrzędnego, na przykład modele do reszt

C.2.9. Moduł contacts

Moduł CONTACTS służy do obliczeń międzycząsteczkowych map kontaktów niewiążących zgodnie z kryteriami stosowanymi przez serwis PDBsum.

Klasa ContactMap

Klasa wyznaczająca mapę kontaktów niewiążących pomiędzy dwoma zbiorami reszt. Dwie reszty znajdują się w kontakcie niewiążącym jeżeli należą do różnych łańcuchów, a dwa spośród ich atomów ciężkich są położone w odległości nie przekracza wskazanej wartości. Zgodnie z serwisem PDBsum, wynosi ona domyślnie 3,9 Å. Kontakty są również dzielone ze względu na typy reszt: aminokwas (P), kwas nukleinowy (N), ligand (L), jon (I) oraz woda (S). Istnieją również dwie formy zwracanych wyników: zwięzła (ile każda reszta ma kontaktów danego typu) oraz pełna (która reszta ma kontakt z którą resztą).

C.2.10. Moduł fod

Moduł FOD zawiera skalę hydrofobowości własnej oraz umożliwia obliczenia i porównywanie rozkładów hydrofobowości teoretycznej i obserwowanej modelu FOD.

Klasa FOD

Klasa obliczająca wartości rozkładów $\tilde{H}r$, $\tilde{H}t$ i $\tilde{H}o$ dla wskazanego zbioru reszt. Wartości hydrofobowości własnej oraz współrzędne atomów efektywnych są wyznaczane automatycznie lub mogą być przekazywane przez użytkownika. Ma on również możliwość wyboru pomiędzy standardową skalą hydrofobowości modelu lub swoją własną. Atomy efektywne są układane zgodnie z osiami układu współrzędnych przy pomocy metody opartej na średnicach (FOD-MAX).

Klasa FODPCA — klasa bazowa: FOD

Modyfikacja klasy FOD, stosująca analizę składowych głównych do układania atomów efektywnych zgodnie z osiami układu współrzędnych (FOD-PCA).

Klasa FODIO

Klasa zapisująca i odczytująca z plików tekstowych dane uzyskane przez klasę FOD. Dane są zapisywane w tekstowym formacie kolumn o stałej szerokości (5 miejsc dziesiętnych), dzięki czemu są czytelne dla użytkownika i mogą być łatwo importowane przez programy arkuszy kalkulacyjnych.

Moduł FOD udostępnia również następujące funkcje:

ENTROPY	Oblicza rozkłady wartości $O\|\|T$ i $O\|\|R$
CLASSIFICATION	klasyfikuje reszty na podstawie kwartyli rozkładów $\tilde{H}t$ i $\tilde{H}o$

C.2.11. Moduł ecepp

Moduł ECEPP zawiera parametryzację oraz umożliwia obliczenia energii międzycząsteczkowych potencjałów niekowalencyjnych pola siłowego ECEPP/3.

Klasa ECEPP

Klasa obliczająca energię międzycząsteczkowych potencjałów niekowalencyjnych pola ECEPP/3 w sposób przedstawiony w rozdziale 3.4.5. Użytkownik ma możliwość wskazywania promienia poszukiwania najbliższych sąsiadów atomów, a także promienia kolizji dla atomów wodoru. Wartości energii są zwracane dla każdego potencjału osobno. Obecnie, klasa ta nie posiada możliwości obliczania wewnątrzcząsteczkowych składowych energii, w tym potencjału torsyjnego.

C.2.12. Moduł docking

Moduł DOCKING umożliwia przeprowadzanie symulacji tworzenia się kompleksów dwóch lub więcej cząsteczek traktowanych jako bryły sztywne.

Klasa RIGIDBODY

Klasa przechowująca współrzędne zbioru atomów, umożliwiająca układanie ich w położeniu początkowym przy pomocy metody FOD-PCA, a następnie zmianę ich orientacji w przestrzeni na podstawie wskazanego wektora konformacji w sposób przedstawiony w rozdziale 3.5.1. Każde takie przekształcenie tworzy nową instancję klasy. Klasa ta umożliwia również tworzenie instancji klasy MODEL, której współrzędne atomów odpowiadają jej współrzędnym atomów.

Klasa RigidComplex
Klasa reprezentująca konformację kompleksu brył sztywnych, złożonego z dowolnej liczby instancji klasy RigidBody. Umożliwia zmianę orientacji wszystkich z nich (poza pierwszą) na podstawie wskazanego wektora konformacji. Każde takie przekształcenie tworzy nową instancję klasy. Klasa ta umożliwia również obliczenia map kontaktów niewiążących w reprezentowanym przez nią kompleksie, a także tworzenie na jego podstawie instancji klasy Model.

Klasa RigidDocker
Klasa umożliwiająca obliczenia wartości kryteriów pól zewnętrznego i wewnętrznego oraz funkcji ograniczeń odpowiadające orientacji kompleksu (instancji klasy RigidComplex) wyznaczonej na podstawie wskazanego wektora konformacji. Wszystkie obliczone wartości są przechowywane w pamięci podręcznej, co pozwala na znaczne skrócenie czasu ponownego dostępu do nich. Użytkownik ma możliwość ustalania rozmiaru tego bufora. Wartości funkcji ograniczeń są obliczane przez jedną metodę, co również skraca czas uzyskiwania ich wartości.

C.3. Wykonanie równoległe

Zaproponowany w niniejszej rozprawie i przedstawiony w rozdziale 3.1 algorytm optymalizacji wielokryterialnej MOSF może być częściowo wykonywany równolegle. Każda rodzina rojów cząstek poszukuje rozwiązań niezdominowanych niezależnie od pozostałych, co oznacza, że ich aktualizacja nie musi następować w ustalonej kolejności. Mogą być więc przetwarzanie w tym samym czasie. W związku z tym, o ile kryteria optymalizacyjne i funkcje ograniczeń nie mają osobnych wymagań, synchronizacja staje się niezbędna tylko podczas oczekiwania na zakończenie pracy wszystkich rodzin oraz rzadziej przeprowadzanych na nich operacji: inicjalizacji, łączenia, dzielenia i dominacji. Zysk z wykonania równoległego wynika stąd, że czynności te zajmują łącznie mniej czasu niż aktualizacja właściwości cząstek.

Poprzez wykonanie równoległe rozumiane jest tutaj uruchomienie w pojedynczym procesie kilku wątków roboczych, które pobierają informacje o rodzinach rojów ze zsynchronizowanej struktury danych (na przykład kolejki FIFO) i samodzielnie je aktualizują. Ich liczba odpowiada typowo liczbie rdzeni procesora. Warto zauważyć, że w zależności od implementacji algorytmu, jednostki robocze mogą pracować na całych rodzinach, pojedynczych rojach lub nawet indywidualnych cząstkach. Większa ziarnistość maksymalizuje użycie procesora, choć wymaga większego skomplikowania kodu programu oraz podnosi koszt związany z wykorzystaniem przez niego systemowych mechanizmów synchronizacji (pamięć dzielona, semafory, muteksy, itd.).

Implementacja wykonania równoległego algorytmu MOSF w Pythonie natrafia jednak na problem natury technicznej, związany z obecnością globalnej blokady interpretera (global interpreter lock, GIL) w CPythonie. Powoduje ona, że w tym samym czasie bajktod może być wykonywany tylko przez jeden wątek należący do danego procesu. Blokada ta jest domyślnie zwalniania podczas oczekiwania na zakończenie operacji wejścia/wyjścia oraz na żądanie funkcji z dynamicznie ładowanych, zewnętrznych bibliotek C/C++. Z tego powodu, CPython dobrze nadaje się do interpretacji i realizacji jednowątkowych skryptów, lub takich w których wątki przez większość czasu przebywają w stanie uśpienia, na przykład w graficznych interfejsach użytkownika. Jeżeli jednak próbują one w pełni wykorzystać możliwości procesora, co ma miejsce w przypadku optymalizacji, sprawność tak napisanego programu zdecydowanie się obniża. W szczególności, jego osiągi mogą okazać się słabsze nawet od wersji jednowątkowych ze względu na czas tracony na przełączanie pomiędzy jednostkami roboczymi. GIL jest podstawowym elementem funkcjonowania CPythona, dlatego obecnie nie ma możliwości prostego rozwiązania tego problemu na poziomie implementacji samego interpretera (optymalna modyfikacja wciąż nie została zaproponowana).

Aby móc jednak przeprowadzić test wykonania równoległego algorytmu MOSF napisanego w Pythonie, zastosowano należący do standardowej biblioteki tego języka moduł MULTIPROCESSING. Obchodzi on ograniczenia blokady interpretera poprzez imitację sposobu działania modułu THREADING przy użyciu procesów zamiast wątków. Pozwala to na faktyczne wykonanie równoległe, kosztem utrudnień w wymianie danych pomiędzy jednostkami roboczymi.

Test wykonania równoległego algorytmu MOSF został przeprowadzony na komputerze typu IBM PC o następujących parametrach sprzętowych i programowych:

- procesor: Intel Core i7-6700 @ 3,4 GHz (4,0/3,9/3,8/3,7 GHz turbo)
- RAM: Kingston HyperX FURY DDR4 2×8 GB @ 2133 MHz (CL 14)
- system operacyjny: GNU/Linux 4.4.19 x86_64 (glibc 2.46.2, gcc 5.3.0)
- Python: 2.7.11 (NumPy 1.11.0, SciPy 0.18.0)

Test miał wykazać, czy równoczesna aktualizacja więcej niż jednej rodziny rojów skutkuje skróceniem czasu oczekiwania na zakończenie procedury optymalizacji. Ponieważ wyżej wymieniony procesor posiada 4 rdzenie i obsługuje do 8 współbieżnych wątków, najpierw uruchomiono tylko jeden proces roboczy (wykonanie sekwencyjne – referencyjne), następnie dwa, potem cztery, a na końcu właśnie osiem.

Procesy	Średnia	Odch. std.	Minimum	Maksimum	Skrócenie czasu
1	16,094 s	0,195 s	15,523 s	16,235 s	100% (16,094 s)
2	8,419 s	0,027 s	8,368 s	8,470 s	52,3% z 16,094 s
4	4,496 s	0,093 s	4,394 s	4,739 s	27,9% z 16,094 s
8	3,697 s	0,019 s	3,657 s	3,725 s	23,0% z 16,094 s

Tabela C.1: Statystyka czasu wykonania równoległego algorytmu MOSF.

Do testu wybrano funkcję Banach 1, ale z pominięciem występującej w niej funkcji ograniczeń, dzięki czemu wszystkie rodziny rojów mogły sprawdzić dokładnie tę samą liczbę kandydatów na rozwiązania niezdominowane, czyli teoretycznie wykonywać optymalizację przez taki sam okres czasu. Choć nie należy oczekiwać takiej prawidłowości w rzeczywistych zastosowaniach, podejście to umożliwiło w prosty sposób sprawdzenie i potwierdzenie hipotezy z poprzedniego paragrafu.

Algorytm MOSF został uruchomiony 10 razy dla każdej liczby procesów. Początkowa rodzina rojów, z których wszystkie składały się ze 100 cząstek, była aktualizowana 100 razy, analizując w ten sposób 20000 rozwiązań. Maksymalny rozmiar jej archiwum wynosił 200 punktów. Następnie rodzina ta była dzielona na rodziny pochodne, spośród których wybierano w sposób losowy 8, a pozostałe usuwano. Roje należące do każdej z nich również składały się ze 100 cząstek i również działały przez 100 iteracji, korzystając z archiwów również o maksymalnym rozmiarze 200 punktów. Procedury łączenia, dominacji i usuwania nie były stosowane. Był natomiast mierzony czas działania programu od momentu podziału do zakończenia optymalizacji.

Statystyka uzyskanych wyników jest przedstawiona w tabeli C.1. Można z niej odczytać, że zwiększenie liczby jednostek roboczych z jednej do dwóch skraca czas trwania optymalizacji o około połowę. Ta sama zależność jest zauważalna w przypadku czterech procesów, ale już nie dla ośmiu. Dzieje się tak dlatego, że tylko fizyczne rdzenie procesora są w stanie zapewnić „prawdziwe” wykonanie równoległe, których taktowanie w trybie turbo zmniejsza się dodatkowo wraz z obciążeniem. Technologia wielowątkowości współbieżnej nadal pozwala na uzyskanie pewnego przyspieszenia (do około 30% [427]), choć najbardziej mogą na niej skorzystać aplikacje zoptymalizowane pod kątem jej maksymalnego wykorzystania.

Uzyskane wyniki pozwalają na stwierdzenie, że algorytm MOSF może być efektywnie wykonywany równolegle, a zysk na czasie trwania optymalizacji jest proporcjonalny do liczby uruchomionych jednostek roboczych. Inny sposób przyspieszenia obliczeń może być możliwy dzięki przeniesieniu części z nich na procesor graficzny [428].

Bibliografia

1. Hogeweg P (2011). The Roots of Bioinformatics in Theoretical Biology. *PLoS Computational Biology* 7.3, e1002021. DOI: 10.1371/journal.pcbi.1002021.
2. Dill K, Ozkan S, Weikl T i in. (2007). The protein folding problem: when will it be solved? *Current Opinion in Structural Biology* 17.3, s. 342–346. DOI: 10.1016/j.sbi.2007.06.001.
3. Ritchie D (2008). Recent Progress and Future Directions in Protein-Protein Docking. *Current Protein & Peptide Science* 9.1, s. 1–15. DOI: 10.2174/138920308783565741.
4. Dobson C (2006). The Generic Nature of Protein Folding and Misfolding. *Protein Misfolding, Aggregation, and Conformational Diseases*. T. 4. Protein Reviews. Springer Boston, s. 21–41. DOI: 10.1007/0-387-25919-8_2.
5. Dobson C (2003). Protein folding and misfolding. *Nature* 426.6968, s. 884–890. DOI: 10.1038/nature02261.
6. Dobson C (2001). The structural basis of protein folding and its links with human disease. *Philosophical Transactions of the Royal Society B: Biological Sciences* 356.1406, s. 133–145. DOI: 10.1098/rstb.2000.0758.
7. Fernald G, Capriotti E, Daneshjou R i in. (2011). Bioinformatics challenges for personalized medicine. *Bioinformatics* 27.13, s. 1741–1748. DOI: 10.1093/bioinformatics/btr295.
8. Kitchen D, Decornez H, Furr J i in. (2004). Docking and scoring in virtual screening for drug discovery: methods and applications. *Nature Reviews Drug Discovery* 3.11, s. 935–949. DOI: 10.1038/nrd1549.
9. Ramsden J (2009). Bioinformatics. An Introduction. T. 10. Computational Biology. Springer London. DOI: 10.1007/978-1-84800-257-9.
10. Crick F (1970). Central Dogma of Molecular Biology. *Nature* 227.5258, s. 561–563. DOI: 10.1038/227561a0.
11. Baltimore D (1970). Viral RNA-dependent DNA Polymerase: RNA-dependent DNA Polymerase in Virions of RNA Tumour Viruses. *Nature* 226.5252, s. 1209–1211. DOI: 10.1038/2261209a0.
12. Temin H, Mizutani S (1970). Viral RNA-dependent DNA Polymerase: RNA-dependent DNA Polymerase in Virions of Rous Sarcoma Virus. *Nature* 226.5252, s. 1211–1213. DOI: 10.1038/2261211a0.
13. Smith J, René D (2006). Following the Path of the Virus: The Exploitation of Host DNA Repair Mechanisms by Retroviruses. *ACS Chemical Biology* 1.4, s. 217–226. DOI: 10.1021/cb600131q.
14. Prusiner S (1982). Novel proteinaceous infectious particles cause scrapie. *Science* 216.4542, s. 136–144. DOI: 10.1126/science.6801762.
15. Puławski W, Ghoshdastider U, Andrisano V i in. (2012). Ubiquitous Amyloids. *Applied Biochemistry and Biotechnology* 166.7, s. 1626–1643. DOI: 10.1007/s12010-012-9549-3.
16. Strieker M, Tanović A, Marahiel M (2010). Nonribosomal peptide synthetases: structures and dynamics. *Current Opinion in Structural Biology* 20.2, s. 234–240. DOI: 10.1016/j.sbi.2010.01.009.

17. Walton J, Panaccione D, Hallen H (2004). Peptide Synthesis without Ribosomes. *Advances in Fungal Biotechnology for Industry, Agriculture, and Medicine*. Springer Boston, s. 127–162. DOI: 10.1007/978-1-4419-8859-1_7.
18. Buxbaum E (2007). Fundamentals of Protein Structure and Function. Springer US. DOI: 10.1007/978-0-387-68480-2.
19. Yang H, Yang M, Ding Y i in. (2003). The crystal structures of severe acute respiratory syndrome virus main protease and its complex with an inhibitor. *Proceedings of the National Academy of Sciences* 100.23, s. 13190–13195. DOI: 10.1073/pnas.1835675100.
20. Matthews J, Sunde M (2012). Dimers, Oligomers, Everywhere. *Advances in Experimental Medicine and Biology*. T. 747. Advances in Experimental Medicine and Biology. Springer New York, s. 1–18. DOI: 10.1007/978-1-4614-3229-6_1.
21. Kennedy D, Norman C (2005). What Don't We Know? *Science* 309.5731, s. 75. DOI: 10.1126/science.309.5731.75.
22. So Much More to Know... (2005). *Science* 309.5731, s. 78–102. DOI: 10.1126/science.309.5731.78b.
23. Moult J, Pedersen J, Judson R i in. (1995). A large-scale experiment to assess protein structure prediction methods. *Proteins: Structure, Function, and Genetics* 23.3, s. ii–iv. DOI: 10.1002/prot.340230303.
24. Cozzetto D, Kryshtafovych A, Fidelis K i in. (2009). Evaluation of template-based models in CASP8 with standard measures. *Proteins* 77.S9, s. 18–28. DOI: 10.1002/prot.22561.
25. Ben-David M, Noivirt-Brik O, Paz A i in. (2009). Assessment of CASP8 structure predictions for template free targets. *Proteins* 77.S9, s. 50–65. DOI: 10.1002/prot.22591.
26. Zemla A, Češlovas V, Moult J i in. (2001). Processing and evaluation of predictions in CASP4. *Proteins: Structure, Function, and Genetics* 45.S5, s. 13–21. DOI: 10.1002/prot.10052.
27. Janin J, Henrick K, Moult J i in. (2003). CAPRI: A Critical Assessment of PRedicted Interactions. *Proteins: Structure, Function, and Genetics* 52.1, s. 2–9. DOI: 10.1002/prot.10381.
28. Méndez R, Leplae R, De Maria L i in. (2003). Assessment of blind predictions of protein-protein interactions: Current status of docking methods. *Proteins* 52.1, s. 51–67. DOI: 10.1002/prot.10393.
29. Méndez R, Leplae R, Lensink M i in. (2005). Assessment of CAPRI predictions in rounds 3-5 shows progress in docking procedures. *Proteins* 60.2, s. 150–169. DOI: 10.1002/prot.20551.
30. Lensink M, Méndez R, Wodak S (2007). Docking and scoring protein complexes: CAPRI 3rd Edition. *Proteins* 69.4, s. 704–718. DOI: 10.1002/prot.21804.
31. Radivojac P, Clark W, Oron T i in. (2013). A large-scale evaluation of computational protein function prediction. *Nature Methods* 10.3, s. 221–227. DOI: 10.1038/nmeth.2340.
32. Ashburner M, Ball C, Blake J i in. (2000). Gene Ontology: tool for the unification of biology. *Nature Genetics* 25.1, s. 25–29. DOI: 10.1038/75556.
33. Altschul S, Gish W, Miller W i in. (1990). Basic local alignment search tool. *Journal of Molecular Biology* 215.3, s. 403–410. DOI: 10.1016/s0022-2836(05)80360-2.
34. Altschul S, Madden T, Schäffer A i in. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research* 25.17, s. 3389–3402. DOI: 10.1093/nar/25.17.3389.
35. Bernstein F, Koetzle T, Williams G i in. (1977). The protein data bank: A computer-based archival file for macromolecular structures. *Journal of Molecular Biology* 112.3, s. 535–542. DOI: 10.1016/s0022-2836(77)80200-3.
36. Berman H, Westbrook J, Feng Z i in. (2000). The Protein Data Bank. *Nucleic Acids Research* 28.1, s. 235–242. DOI: 10.1093/nar/28.1.235.

37. Berman H, Henrick K, Nakamura H (2003). Announcing the worldwide Protein Data Bank. *Nature Structural & Molecular Biology* 10.12, s. 980–980. DOI: 10.1038/nsb1203-980.
38. Apweiler R, Bairoch A, Wu C i in. (2004). UniProt: the Universal Protein knowledgebase. *Nucleic Acids Research* 32.Database issue, s. D115–D119. DOI: 10.1093/nar/gkh131.
39. Bairoch A (2004). The Universal Protein Resource (UniProt). *Nucleic Acids Research* 33, s. D154–D159. DOI: 10.1093/nar/gki070.
40. Bairoch A (1996). The SWISS-PROT protein sequence data bank and its new supplement TREMBL. *Nucleic Acids Research* 24.1, s. 21–25. DOI: 10.1093/nar/24.1.21.
41. Wu C, Yeh LS, Huang H i in. (2003). The Protein Information Resource. *Nucleic Acids Research* 31.1, s. 345–347. DOI: 10.1093/nar/gkg040.
42. Orengo C, Michie A, Jones S i in. (1997). CATH – a hierarchic classification of protein domain structures. *Structure* 5.8, s. 1093–1109. DOI: 10.1016/s0969-2126(97)00260-8.
43. Sillitoe I, Lewis T, Cuff A i in. (2015). CATH: comprehensive structural and functional annotations for genome sequences. *Nucleic Acids Research* 43.D1, s. D376–D381. DOI: 10.1093/nar/gku947.
44. Murzin A, Brenner S, Hubbard T i in. (1995). SCOP: A structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology* 247.4, s. 536–540. DOI: 10.1016/s0022-2836(05)80134-2.
45. Andreeva A, Howorth D, Chothia C i in. (2013). SCOP2 prototype: a new approach to protein structure mining. *Nucleic Acids Research* 42.D1, s. D310–D314. DOI: 10.1093/nar/gkt1242.
46. Dill K, MacCallum J (2012). The Protein-Folding Problem, 50 Years On. *Science* 338.6110, s. 1042–1046. DOI: 10.1126/science.1219021.
47. Hardison R (1996). A brief history of hemoglobins: plant, animal, protist, and bacteria. *Proceedings of the National Academy of Sciences* 93.12, s. 5675–5679. DOI: 10.1073/pnas.93.12.5675.
48. Banach M, Kalinowska B, Konieczny L i in. (2016). Role of Disulfide Bonds in Stabilizing the Conformation of Selected Enzymes—An Approach Based on Divergence Entropy Applied to the Structure of Hydrophobic Core in Proteins. *Entropy* 18.3, s. 67. DOI: 10.3390/e18030067.
49. Roterman I, Konieczny L, Jurkowski W i in. (2011a). Two-intermediate model to characterize the structure of fast-folding proteins. *Journal of Theoretical Biology* 283.1, s. 60–70. DOI: 10.1016/j.jtbi.2011.05.027.
50. Banach M, Stąpor K, Roterman I (2009). Chaperonin Structure - The Large Multi-Subunit Protein Complex. *International Journal of Molecular Science* 10.3, s. 844–861. DOI: 10.3390/ijms10030844.
51. Dygut J, Kalinowska B, Banach M i in. (2016). Structural Interface Forms and Their Involvement in Stabilization of Multidomain Proteins or Protein Complexes. *International Journal of Molecular Sciences* 17.10, s. 1741. DOI: 10.3390/ijms17101741.
52. Rousseau F, Schymkowitz J, Itzhaki L (2012). Implications of 3D Domain Swapping for Protein Folding, Misfolding and Function. *Advances in Experimental Medicine and Biology*. T. 747. Advances in Experimental Medicine and Biology. Springer New York, s. 137–152. DOI: 10.1007/978-1-4614-3229-6_9.
53. Kalinowska B, Banach M, Konieczny L i in. (2014). Intrinsically Disordered Proteins—Relation to General Model Expressing the Active Role of the Water Environment. *Advances in Protein Chemistry and Structural Biology*. T. 94. Advances in Protein Chemistry and Structural Biology. Elsevier BV, s. 315–346. DOI: 10.1016/B978-0-12-800168-4.00008-1.
54. Kryshtafovych A, Moult J, Bales P i in. (2014). Challenging the state of the art in protein structure prediction: Highlights of experimental target structures for the 10th Critical Assessment of Techniques for Protein Structure Prediction Experiment CASP10. *Proteins* 82.S2, s. 26–42. DOI: 10.1002/prot.24489.

55. Lensink M, Wodak S (2013). Docking, scoring, and affinity prediction in CAPRI. *Proteins* 81.12, s. 2082–2095. DOI: 10.1002/prot.24428.
56. Samish I, Bourne P, Najmanovich R (2014). Achievements and challenges in structural bioinformatics and computational biophysics. *Bioinformatics* 31.1, s. 146–150. DOI: 10.1093/bioinformatics/btu769.
57. Dessailly B, Orengo C (2009). Function Diversity Within Folds and Superfamilies. *From Protein Structure to Function with Bioinformatics*. Springer Dordrecht, s. 143–166. DOI: 10.1007/978-1-4020-9058-5_6.
58. Rooman M, Dehouck Y, Kwasigroch J i in. (2002). What is Paradoxical about Levinthal Paradox? *Journal of Biomolecular Structure and Dynamics* 20.3, s. 327–329. DOI: 10.1080/07391102.2002.10506850.
59. Konieczny L, Brylinski M, Roterman I (2006). Gauss-Function-Based Model of Hydrophobicity Density in Proteins. *In Silico Biology* 6.1–2, s. 15–22.
60. Banach M, Konieczny L, Roterman I (2012a). The late-stage intermediate. *Protein Folding in Silico: Protein Folding Versus Protein Structure Prediction*. T. 22. Woodhead Publishing Series in Biomedicine. Woodhead Publishing, s. 21–37. DOI: 10.1533/9781908818256.21.
61. Momany F, McGuire R, Burgess A i in. (1975). Energy parameters in polypeptides. VII. Geometric parameters, partial atomic charges, nonbonded interactions, hydrogen bond interactions, and intrinsic torsional potentials for the naturally occurring amino acids. *Journal of Physical Chemistry* 79.22, s. 2361–2381. DOI: 10.1021/j100589a006.
62. Nemethy G, Pottle M, Scheraga H (1983). Energy parameters in polypeptides. 9. Updating of geometrical parameters, nonbonded interactions, and hydrogen bond interactions for the naturally occurring amino acids. *Journal of Physical Chemistry* 87.11, s. 1883–1887. DOI: 10.1021/j100234a011.
63. Nemethy G, Gibson K, Palmer K i in. (1992). Energy parameters in polypeptides. 10. Improved geometrical parameters and nonbonded interactions for use in the ECEPP/3 algorithm, with application to proline-containing peptides. *Journal of Physical Chemistry* 96.15, s. 6472–6484. DOI: 10.1021/j100194a068.
64. Kennedy J, Eberhart R (1995). Particle swarm optimization. *IEEE International Conference on Neural Networks*, s. 1942–1948. DOI: 10.1109/ICNN.1995.488968.
65. Danchin A, Médigue C, Gascuel O i in. (1991). From data banks to data bases. *Research in Microbiology* 142.7-8, s. 913–916. DOI: 10.1016/0923-2508(91)90073-j.
66. White FJ (1961). Regeneration of native secondary and tertiary structures by air oxidation of reduced ribonuclease. *Journal of Biological Chemistry* 236.5, s. 1353–1360.
67. Anfinsen C, Haber E, Sela M i in. (1961). The kinetics of formation of native ribonuclease during oxidation of the reduced polypeptide chain. *Proceedings of the National Academy of Sciences* 47.9, s. 1309–1314. DOI: 10.1073/pnas.47.9.1309.
68. Englander S (2000). Protein Folding Intermediates and Pathways Studied by Hydrogen Exchange. *Annual Review of Biophysics and Biomolecular Structure* 29.1, s. 213–238. DOI: 10.1146/annurev.biophys.29.1.213.
69. Ramachandran G, Ramakrishnan C, Sasisekharan V (1963). Stereochemistry of polypeptide chain configurations. *Journal of Molecular Biology* 7.1, s. 95–99. DOI: 10.1016/s0022-2836(63)80023-6.
70. Levinthal C (1969). How to Fold Graciously. *Mössbauer Spectroscopy in Biological Systems: Proceedings of a meeting held at Allerton House, March 17 and 18, 1969, Monticello, Illinois*. T. 67. University of Illinois bulletin 41. University of Illinois, Urbana, s. 22–24.
71. Levinthal C (1968). Are there pathways for protein folding? *Journal de Chimie Physique* 65.1, s. 44–45.
72. Zwangiz R, Szabo A, Bagchi B (1992). Levinthal's paradox. *Proceedings of the National Academy of Sciences* 89.1, s. 20–22. DOI: 10.1073/pnas.89.1.20.

73. Baldwin R (1995). The nature of protein folding pathways: The classical versus the new view. *Journal of Biomolecular NMR* 5.2, s. 103–109. DOI: 10.1007/bf00208801.
74. Baldwin R (2008). The Search for Folding Intermediates and the Mechanism of Protein Folding. *Annual Review of Biophysics* 37.1, s. 1–21. DOI: 10.1146/annurev.biophys.37.032807.125948.
75. Wolynes P, Onuchic J, Thirumalai D (1995). Navigating the folding routes. *Science* 267.5204, s. 1619–1620. DOI: 10.1126/science.7886447.
76. Baldwin R (1999). Protein folding from 1961 to 1982. *Nature Structural Biology* 6.9, s. 814–817. DOI: 10.1038/12268.
77. Englander S, Mayne L (2014). The nature of protein folding pathways. *Proceedings of the National Academy of Sciences* 111.45, s. 15873–15880. DOI: 10.1073/pnas.1411798111.
78. Lee J, Liwo A, Ripoll D i in. (1999). Calculation of protein conformation by global optimization of a potential energy function. *Proteins: Structure, Function, and Bioinformatics* 37.S3, s. 204–208. DOI: 10.1002/(SICI)1097-0134(1999)37:3+<204::AID-PROT26>3.0.CO;2-F.
79. Anfinsen C (1972). The formation and stabilization of protein structure. *Biochemical Journal* 128.4, s. 737–749. DOI: 10.1042/bj1280737.
80. Anfinsen C (1973). Principles that Govern the Folding of Protein Chains. *Science* 181.4096, s. 223–230. DOI: 10.1126/science.181.4096.223.
81. Greiner W, Neise L, Stöcker H (1995). Thermodynamics and Statistical Mechanics. Springer New York. DOI: 10.1007/978-1-4612-0827-3.
82. Konieczny L, Roterman I, Spólnik P (2014). Systems Biology. Springer, Cham. DOI: 10.1007/978-3-319-01336-7.
83. Dill K (1997). Additivity Principles in Biochemistry. *Journal of Biological Chemistry* 272.2, s. 701–704. DOI: 10.1074/jbc.272.2.701.
84. Dill K, Chan H (1997). From Levinthal to pathways to funnels. *Nature Structural & Molecular Biology* 4.1, s. 10–19. DOI: 10.1038/nsb0197-10.
85. O'Toole N, Vakser I (2008). Large-scale characteristics of the energy landscape in protein–protein interactions. *Proteins* 71.1, s. 144–152. DOI: 10.1002/prot.21665.
86. Karplus M (2011). Behind the folding funnel diagram. *Nature Chemical Biology* 7.7, s. 401–404. DOI: 10.1038/nchembio.565.
87. Ben-Naim A (2012). Levinthal's question revisited, and answered. *Journal of Biomolecular Structure and Dynamics* 30.1, s. 113–124. DOI: 10.1080/07391102.2012.674286.
88. Roterman I, Konieczny L, Banach M i in. (2011b). Intermediates in the Protein Folding Process: A Computational Model. *International Journal of Molecular Science* 12.8, s. 4850–4860. DOI: 10.3390/ijms11084850.
89. Ben-Naim A (2011). Pitfalls in Anfinsen's thermodynamic hypothesis. *Chemical Physics Letters* 511.1-3, s. 126–128. DOI: 10.1016/j.cplett.2011.05.049.
90. Shortle D, Simons K, Baker D (1998). Clustering of low-energy conformations near the native structures of small proteins. *Proceedings of the National Academy of Sciences* 95.19, s. 11158–11162. DOI: 10.1073/pnas.95.19.11158.
91. Ramachandran K, Gopakumar D, Namboori K (2008). Computational Chemistry and Molecular Modeling. Principles and Applications. Springer Berlin Heidelberg. DOI: 10.1007/978-3-540-77304-7.
92. Pople J, Santry D, Segal G (1965). Approximate Self-Consistent Molecular Orbital Theory. I. Invariant Procedures. *Journal of Chemical Physics* 43.10, S129–S135. DOI: 10.1063/1.1701475.
93. Pople J, Segal G (1965). Approximate Self-Consistent Molecular Orbital Theory. II. Calculations with Complete Neglect of Differential Overlap. *Journal of Chemical Physics* 43.10, S136–S151. DOI: 10.1063/1.1701476.

94. Pople J, Segal G (1966). Approximate Self-Consistent Molecular Orbital Theory. III. CNDO Results for AB2 and AB3 Systems. *Journal of Chemical Physics* 44.9, s. 3289–3296. DOI: 10.1063/1.1727227.
95. Bondi A (1964). van der Waals Volumes and Radii. *The Journal of Physical Chemistry* 68.3, s. 441–451. DOI: 10.1021/j100785a001.
96. Rowland R, Taylor R (1996). Intermolecular Nonbonded Contact Distances in Organic Crystal Structures: Comparison with Distances Expected from van der Waals Radii. *Journal of Physical Chemistry* 100.18, s. 7384–7391. DOI: 10.1021/jp953141.
97. Connolly M (1983). Solvent-accessible surfaces of proteins and nucleic acids. *Science* 221.4612, s. 709–713. DOI: 10.1126/science.6879170.
98. Schlick T (2010). Molecular Modeling and Simulation: An Interdisciplinary Guide. T. 21. Interdisciplinary Applied Mathematics. Springer New York. DOI: 10.1007/978-1-4419-6351-2.
99. Lewars E (2011). Computational Chemistry. Introduction to the Theory and Applications of Molecular and Quantum Mechanics. Springer Netherlands. DOI: 10.1007/978-90-481-3862-3.
100. Pang YP (2015). Use of 1–4 interaction scaling factors to control the conformational equilibrium between α-helix and β-strand. *Biochemical and Biophysical Research Communications* 457.2, s. 183–186. DOI: 10.1016/j.bbrc.2014.12.084.
101. Brooks C, Pettitt B, Karplus M (1985). Structural and energetic effects of truncating long ranged interactions in ionic and polar fluids. *The Journal of Chemical Physics* 83.11, s. 5897. DOI: 10.1063/1.449621.
102. Ruvinsky A, Vakser I (2007). Interaction cutoff effect on ruggedness of protein-protein energy landscape. *Proteins* 70.4, s. 1498–1505. DOI: 10.1002/prot.21644.
103. Piana S, Lindorff-Larsen K, Dirks R i in. (2012). Evaluating the Effects of Cutoffs and Treatment of Long-range Electrostatics in Protein Folding Simulations. *PLoS ONE* 7.6, e39918. DOI: 10.1371/journal.pone.0039918.
104. Steinbach P, Brooks B (1994). New spherical-cutoff methods for long-range forces in macromolecular simulation. *Journal of Computational Chemistry* 15.7, s. 667–683. DOI: 10.1002/jcc.540150702.
105. Loncharich R, Brooks B (1989). The effects of truncating long-range forces on protein dynamics. *Proteins: Structure, Function, and Genetics* 6.1, s. 32–45. DOI: 10.1002/prot.340060104.
106. Norberg J, Nilsson L (2000). On the Truncation of Long-Range Electrostatic Interactions in DNA. *Biophysical Journal* 79.3, s. 1537–1553. DOI: 10.1016/s0006-3495(00)76405-8.
107. Toukmaji A, Board J (1996). Ewald summation techniques in perspective: a survey. *Computer Physics Communications* 95.2-3, s. 73–92. DOI: 10.1016/0010-4655(96)00016-1.
108. Ewald P (1921). Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Annales de Physique* 369.3, s. 253–287. DOI: 10.1002/andp.19213690304.
109. Darden T, York D, Pedersen J (1993). Particle mesh Ewald: An Nfflog(N) method for Ewald sums in large systems. *Journal of Chemical Physics* 98.12, s. 10089. DOI: 10.1063/1.464397.
110. Chen C, Depa P, Sakai V i in. (2006). A comparison of united atom, explicit atom, and coarse-grained simulation models for poly(ethylene oxide). *Journal of Chemical Physics* 124.23, s. 234901. DOI: 10.1063/1.2204035.
111. Chen C, Depa P, Maranas J i in. (2008). Comparison of explicit atom, united atom, and coarse-grained simulations of poly(methyl methacrylate). *The Journal of Chemical Physics* 128.12, s. 124906. DOI: 10.1063/1.2833545.
112. Ponder J, Case D (2003). Force Fields for Protein Simulations. *Protein Simulations*. T. 66. Advances in Protein Chemistry. Academic Press, s. 27–85. DOI: 10.1016/s0065-3233(03)66002-x.

113. Brooks B, Bruccoleri R, Olafson B i in. (1983). CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. *Journal of Computational Chemistry* 4.2, s. 187–217. DOI: 10.1002/jcc.540040211.

114. Brooks B, Brooks C, Mackerell A i in. (2009). CHARMM: The biomolecular simulation program. *Journal of Computational Chemistry* 30.10, s. 1545–1614. DOI: 10.1002/jcc.21287.

115. Weiner S, Kollman P, Case D i in. (1984). A new force field for molecular mechanical simulation of nucleic acids and proteins. *Journal of the American Chemical Society* 106.3, s. 765–784. DOI: 10.1021/ja00315a051.

116. Cornell W, Cieplak P, Bayly C i in. (1995). A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids, and Organic Molecules. *Journal of the American Chemical Society* 117.19, s. 5179–5197. DOI: 10.1021/ja00124a002.

117. Jorgensen W, Maxwell D, Tirado-Rives J (1996). Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids. *Journal of the American Chemical Society* 118.45, s. 11225–11236. DOI: 10.1021/ja9621760.

118. Kaminski G, Friesner R, Tirado-Rives J i in. (2001). Evaluation and Reparametrization of the OPLS-AA Force Field for Proteins via Comparison with Accurate Quantum Chemical Calculations on Peptides. *Journal of Physical Chemistry B* 105.28, s. 6474–6487. DOI: 10.1021/jp003919d.

119. Jorgensen W, Tirado-Rives J (1988). The OPLS potential functions for proteins, energy minimizations for crystals of cyclic peptides and crambin. *Journal of the American Chemical Society* 110.6, s. 1657–1666. DOI: 10.1021/ja00214a001.

120. Liwo A, Ołdziej S, Pincus M i in. (1997a). A united-residue force field for off-lattice protein-structure simulations. I. Functional forms and parameters of long-range side-chain interaction potentials from protein crystal data. *Journal of Computational Chemistry* 18.7, s. 849–873. DOI: 10.1002/(sici)1096-987x(199705)18:7<849::aid-jcc1>3.0.co;2-r.

121. Liwo A, Pincus M, Wawak R i in. (1997b). A united-residue force field for off-lattice protein-structure simulations. II. Parameterization of short-range interactions and determination of weights of energy terms by Z-score optimization. *Journal of Computational Chemistry* 18.7, s. 874–887. DOI: 10.1002/(sici)1096-987x(199705)18:7<874::aid-jcc2>3.0.co;2-o.

122. Berendsen H, Spoel D van der, Drunen R van (1995). GROMACS: A message-passing parallel molecular dynamics implementation. *Computer Physics Communications* 91.1-3, s. 43–56. DOI: 10.1016/0010-4655(95)00042-e.

123. Páll S, Abraham M, Kutzner C i in. (2015). Tackling Exascale Software Challenges in Molecular Dynamics Simulations with GROMACS. *Solving Software Challenges for Exascale*. T. 8759. Lecture Notes in Computer Science. Springer Cham, s. 3–27. DOI: 10.1007/978-3-319-15976-8_1.

124. Banach M, Konieczny L, Roterman I (2014). The fuzzy oil drop model, based on hydrophobicity density distribution, generalizes the influence of water environment on protein structure and function. *Journal of Theoretical Biology* 359, s. 6–17. DOI: 10.1016/j.jtbi.2014.05.007.

125. Skyner R, McDonagh J, Groom C i in. (2015). A review of methods for the calculation of solution free energies and the modelling of systems in solution. *Physical Chemistry Chemical Physics* 17.9, s. 6174–6191. DOI: 10.1039/c5cp00288e.

126. Ooi T, Oobatake M, Nemethy G i in. (1987). Accessible surface areas as a measure of the thermodynamic parameters of hydration of peptides. *Proceedings of the National Academy of Sciences* 84.10, s. 3086–3090. DOI: 10.1073/pnas.84.10.3086.

127. Swinbank R, Purser R (2006). Fibonacci grids: A novel approach to global modelling. *Quarterly Journal of the Royal Meteorological Society* 132.619, s. 1769–1793. DOI: 10.1256/qj.05.227.

128. Keinert B, Innmann M, Sänger M i in. (2015). Spherical fibonacci mapping. *ACM Transactions on Graphics* 34.6, s. 1–7. DOI: 10.1145/2816795.2818131.

129. Vakser I, Aflalo C (1994). Hydrophobic docking: A proposed enhancement to molecular recognition techniques. *Proteins: Structure, Function, and Genetics* 20.4, s. 320–329. DOI: 10.1002/prot.340200405.

130. Dill K (1990). Dominant forces in protein folding. *Biochemistry* 29.31, s. 7133–7155. DOI: 10.1021/bi00483a001.

131. Pettitt B (2013). The unsolved "solved-problem" of protein folding. *Journal of Biomolecular Structure and Dynamics* 31.9, s. 1024–1027. DOI: 10.1080/07391102.2012.748547.

132. Frank H, Evans M (1945). Free Volume and Entropy in Condensed Systems III. Entropy in Binary Liquid Mixtures; Partial Molal Entropy in Dilute Solutions; Structure and Thermodynamics in Aqueous Electrolytes. *Journal of Chemical Physics* 13.11, s. 507–532. DOI: 10.1063/1.1723985.

133. Kauzmann W (1959). Some Factors in the Interpretation of Protein Denaturation. *Advances in Protein Chemistry* 14, s. 1–63. DOI: 10.1016/S0065-3233(08)60608-7.

134. Kauzmann W (1987). Thermodynamics of unfolding. *Nature* 325.6107, s. 763–764. DOI: 10.1038/325763a0.

135. Baldwin R (2014). Dynamic hydration shell restores Kauzmann's 1959 explanation of how the hydrophobic factor drives protein folding. *Proceedings of the National Academy of Sciences* 111.36, s. 13052–13056. DOI: 10.1073/pnas.1414556111.

136. Baldwin R, Rose G (2016). How the hydrophobic factor drives protein folding. *Proceedings of the National Academy of Sciences* 113.44, s. 12462–12466. DOI: 10.1073/pnas.1610541113.

137. Grdadolnik J, Merzel F, Avbelj F (2016). Origin of hydrophobicity and enhanced water hydrogen bond strength near purely hydrophobic solutes. *Proceedings of the National Academy of Sciences* 114.2, s. 322–327. DOI: 10.1073/pnas.1612480114.

138. Alejster P, Banach M, Jurkowski W i in. (2013). Comparative Analysis of Techniques Oriented on the Recognition of Ligand Binding Area in Proteins. *Identification of Ligand Binding Site and Protein-Protein Interaction Area*. T. 8. Focus on Structural Biology. Springer Dordrecht, s. 55–86. DOI: 10.1007/978-94-007-5285-6_4.

139. Banach M, Konieczny L, Roterman I (2013). Can the Structure of the Hydrophobic Core Determine the Complexation Site? *Identification of Ligand Binding Site and Protein-Protein Interaction Area*. T. 8. Focus on Structural Biology. Springer Dordrecht, s. 41–54. DOI: 10.1007/978-94-007-5285-6_3.

140. Piwowar M, Banach M, Konieczny L i in. (2013). Structural role of exon-coded fragment of polypeptide chains in selected enzymes. *Journal of Theoretical Biology* 337, s. 15–23. DOI: 10.1016/j.jtbi.2013.07.016.

141. Piwowar M, Banach M, Konieczny L i in. (2014). Hydrophobic core formation in protein complex of cathepsin. *Journal of Biomolecular Structure and Dynamics* 32.7, s. 1023–1032. DOI: 10.1080/07391102.2013.801784.

142. Kalinowska B, Banach M, Konieczny L i in. (2015). Application of Divergence Entropy to Characterize the Structure of the Hydrophobic Core in DNA Interacting Proteins. *Entropy* 17.3, s. 1477–1507. DOI: 10.3390/e17031477.

143. Gadzała M, Kalinowska B, Banach M i in. (2017). Determining protein similarity by comparing hydrophobic core structure. *Heliyon* 3.2, e00235. DOI: 10.1016/j.heliyon.2017.e00235.

144. Kalinowska B, Banach M, Wiśniowski Z i in. (2017). Is the hydrophobic core a universal structural element in proteins? *Journal of Molecular Modeling* 23.7, s. 205. DOI: 10.1007/s00894-017-3367-z.

145. Roterman I, Banach M, Kalinowska B i in. (2016). Influence of the Aqueous Environment on Protein Structure—A Plausible Hypothesis Concerning the Mechanism of Amyloidogenesis. *Entropy* 18.10, s. 351. DOI: 10.3390/e18100351.

146. Roterman I, Banach M, Konieczny L (2017). Application of the Fuzzy Oil Drop Model Describes Amyloid as a Ribbonlike Micelle. *Entropy* 19.4, s. 167. DOI: 10.3390/e19040167.

147. Roterman I, Konieczny L, Banach M i in. (2014). Simulation of the Protein Folding Process. *Computational Methods to Study the Structure and Dynamics of Biomolecules and Biomolecular Processes*. T. 1. Springer Series in Bio-/Neuroinformatics. Springer Berlin Heidelberg, s. 559–638. DOI: 10.1007/978-3-642-28554-7_18.

148. Banach M, Roterman I, Prudhomme N i in. (2014). Hydrophobic core in domains of immunoglobulin-like fold. *Journal of Biomolecular Structure and Dynamics* 32.10, s. 1583–1600. DOI: 10.1080/07391102.2013.829756.

149. Banach M, Prudhomme N, Carpentier M i in. (2015). Contribution to the Prediction of the Fold Code: Application to Immunoglobulin and Flavodoxin Cases. *PLOS ONE* 10.4, e0125098. DOI: 10.1371/journal.pone.0125098.

150. Kullback S, Leibler R (1951). On Information and Sufficiency. *Annals of Mathematical Statistics* 22.1, s. 79–86. DOI: 10.1214/aoms/1177729694.

151. Marshall G, Vakser I (2005). Protein-Protein Docking Methods. Biology, Chemistry, Bioinformatics, and Drug Design. *Proteomics and Protein-Protein Interactions*. T. 3. Protein Reviews. Springer Boston, s. 115–146. DOI: 10.1007/0-387-24532-4_6.

152. Marchewka D, Jurkowski W, Banach M i in. (2013). Prediction of Protein-Protein Binding Interfaces. *Identification of Ligand Binding Site and Protein-Protein Interaction Area*. T. 8. Focus on Structural Biology. Springer Dordrecht, s. 105–133. DOI: 10.1007/978-94-007-5285-6_6.

153. Leckband D, Israelachvili J (2001). Intermolecular forces in biology. *Quarterly Reviews of Biophysics* 34.02, s. 105–267. DOI: 10.1017/s0033583501003687.

154. Vakser I (2004). Protein-Protein Interfaces Are Special. *Structure* 12.6, s. 910–912. DOI: 10.1016/j.str.2004.05.003.

155. Atilgan A, Turgut D, Atilgan C (2007). Screened Nonbonded Interactions in Native Proteins Manipulate Optimal Paths for Robust Residue Communication. *Biophysical Journal* 92.9, s. 3052–3062. DOI: 10.1529/biophysj.106.099440.

156. Laskowski R (2001). PDBsum: summaries and analyses of PDB structures. *Nucleic Acids Research* 29.1, s. 221–222. DOI: 10.1093/nar/29.1.221.

157. Beer T de, Berka K, Thornton J i in. (2014). PDBsum additions. *Nucleic Acids Research* 42.D1, s. 292–296. DOI: 10.1093/nar/gkt940.

158. Laskowski R, MacArthur M, Moss D i in. (1993). PROCHECK: a program to check the stereochemical quality of protein structures. *Journal of Applied Crystallography* 26, s. 283–291. DOI: 10.1107/S0021889892009944.

159. Chen R, Mintseris J, Janin J i in. (2003). A protein-protein docking benchmark. *Proteins: Structure, Function, and Genetics* 52.1, s. 88–91. DOI: 10.1002/prot.10390.

160. Hwang H, Vreven T, Janin J i in. (2010). Protein-protein docking benchmark version 4.0. *Proteins* 78.15, s. 3111–3114. DOI: 10.1002/prot.22830.

161. Kastritis P, Moal I, Hwang H i in. (2011). A structure-based benchmark for protein-protein binding affinity. *Protein Science* 20.3, s. 482–491. DOI: 10.1002/pro.580.

162. Levy E, Erba E, Robinson C i in. (2008). Assembly reflects evolution of protein complexes. *Nature* 453.7199, s. 1262–1265. DOI: 10.1038/nature06942.

163. Lensink M, Velankar S, Kryshtafovych A i in. (2016). Prediction of homo- and hetero-protein complexes by protein docking and template-based modeling: a CASP-CAPRI experiment. *Proteins*. DOI: 10.1002/prot.25007.

164. Nooren I, Thornton J (2003). Diversity of protein-protein interactions. *The EMBO Journal* 22.14, s. 3486–3492. DOI: 10.1093/emboj/cdg359.

165. Lee J, Wu S, Zhang J (2009). Ab Initio Protein Structure Prediction. *From Protein Structure to Function with Bioinformatics*. Springer Dordrecht, s. 3–25. DOI: 10.1007/978-1-4020-9058-5_1.

166. Fiser A (2009). Comparative Protein Structure Modelling. *From Protein Structure to Function with Bioinformatics*. Springer Dordrecht, s. 57–90. DOI: 10.1007/978-1-4020-9058-5_3.

167. Floudas C, Klepeis J, Pardalos P (1999). Global Optimization Approaches in Protein Folding and Peptide Docking. *Mathematical Support for Molecular Biology*. T. 47. DIMACS Series in Discrete Mathematics and Theoretical Computer Science. American Mathematical Society, s. 141–171.

168. Zhang Y (2008). Progress and challenges in protein structure prediction. *Current Opinion in Structural Biology* 18.3, s. 342–348. DOI: 10.1016/j.sbi.2008.02.004.

169. Khoury G, Smadbeck J, Kieslich C i in. (2014). Protein folding and de novo protein design for biotechnological applications. *Trends in Biotechnology* 32.2, s. 99–109. DOI: 10.1016/j.tibtech.2013.10.008.

170. Floudas C (2007). Computational methods in protein structure prediction. *Biotechnology and Bioengineering* 97.2, s. 207–213. DOI: 10.1002/bit.21411.

171. Kopp J, Bordoli L, Battey J i in. (2007). Assessment of CASP7 predictions for template-based modeling targets. *Proteins* 69.S8, s. 38–56. DOI: 10.1002/prot.21753.

172. Jauch R, Yeo H, Kolatkar P i in. (2007). Assessment of CASP7 structure predictions for template free targets. *Proteins* 69.S8, s. 57–67. DOI: 10.1002/prot.21771.

173. Smith G, Sternberg M (2002). Prediction of protein–protein interactions by docking methods. *Current Opinion in Structural Biology* 12.1, s. 28–35. DOI: 10.1016/s0959-440x(02)00285-3.

174. Janin J (2013). Docking Predictions of Protein-Protein Interactions and Their Assessment: The CAPRI Experiment. *Identification of Ligand Binding Site and Protein-Protein Interaction Area*. T. 8. Focus on Structural Biology. Springer Dordrecht, s. 87–104. DOI: 10.1007/978-94-007-5285-6_5.

175. Kastritis P, Bonvin A (2010). Are Scoring Functions in Protein-Protein Docking Ready To Predict Interactomes? Clues from a Novel Binding Affinity Benchmark. *Journal of Proteome Research* 9.5, s. 2216–2225. DOI: 10.1021/pr9009854.

176. Andrusier N, Mashiach E, Nussinov R i in. (2008). Principles of flexible protein-protein docking. *Proteins* 73.2, s. 271–289. DOI: 10.1002/prot.22170.

177. Bennett M, Schlunegger M, Eisenberg D (1995). 3D domain swapping: A mechanism for oligomer assembly. *Protein Science* 4.12, s. 2455–2468. DOI: 10.1002/pro.5560041202.

178. Cho S, Levy Y, Onuchic J i in. (2005). Overcoming residual frustration in domain-swapping: the roles of disulfide bonds in dimerization and aggregation. *Physical Biology* 2.2, S44–S55. DOI: 10.1088/1478-3975/2/2/s05.

179. Huang SY (2014). Search strategies and evaluation in protein–protein docking: principles, advances and challenges. *Drug Discovery Today* 19.8, s. 1081–1096. DOI: 10.1016/j.drudis.2014.02.005.

180. Katchalski-Katzir E, Shariv I, Eisenstein M i in. (1992). Molecular surface recognition: determination of geometric fit between proteins and their ligands by correlation techniques. *Proceedings of the National Academy of Sciences* 89.6, s. 2195–2199. DOI: 10.1073/pnas.89.6.2195.

181. Liu BF, Chen HM, Huang HL i in. (2005). Flexible protein-ligand docking using particle swarm optimization. *2005 IEEE Congress on Evolutionary Computation*. IEEE. DOI: 10.1109/cec.2005.1554692.

182. Namasivayam V, Günther R (2007). pso@autodock: A Fast Flexible Molecular Docking Program Based on Swarm Intelligence. *Chemical Biology & Drug Design* 70.6, s. 475–484. DOI: 10.1111/j.1747-0285.2007.00588.x.

183. Janson S, Merkle D, Middendorf M (2008). Molecular docking with multi-objective Particle Swarm Optimization. *Applied Soft Computing* 8.1, s. 666–675. DOI: 10.1016/j.asoc.2007.05.005.

184. Pons C, Grosdidier S, Solernou A i in. (2009). Present and future challenges and limitations in protein-protein docking. *Proteins* 78.1, s. 95–108. DOI: 10.1002/prot.22564.

185. Berchanski A, Shapira B, Eisenstein M (2004). Hydrophobic complementarity in protein-protein docking. *Proteins: Structure, Function, and Bioinformatics* 56.1, s. 130–142. DOI: 10.1002/prot.20145.

186. Chen R, Li L, Weng Z (2003). ZDOCK: An initial-stage protein-docking algorithm. *Proteins: Structure, Function, and Genetics* 52.1, s. 80–87. DOI: 10.1002/prot.10389.

187. Pierce B, Wiehe K, Hwang H i in. (2014). ZDOCK server: interactive docking prediction of protein-protein complexes and symmetric multimers. *Bioinformatics* 30.12, s. 1771–1773. DOI: 10.1093/bioinformatics/btu097.

188. Morris G, Goodsell D, Halliday R i in. (1998). Automated docking using a Lamarckian genetic algorithm and an empirical binding free energy function. *Journal of Computational Chemistry* 19.14, s. 1639–1662. DOI: 10.1002/(sici)1096-987x(19981115)19:14<1639::aid-jcc10>3.0.co;2-b.

189. Morris G, Huey R, Lindstrom W i in. (2009). AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. *Journal of Computational Chemistry* 30.16, s. 2785–2791. DOI: 10.1002/jcc.21256.

190. Roterman I, Gibson K, Scheraga H (1989). A Comparison of the CHARMM, AMBER and ECEPP Potentials for Peptides. I. Conformational Predictions for the Tandemly Repeated Peptide (Asn-Ala-Asn-Pro) 9. *Journal of Biomolecular Structure and Dynamics* 7.3, s. 391–419. DOI: 10.1080/07391102.1989.10508502.

191. Roterman I, Lambert M, Gibson K i in. (1989). A Comparison of the CHARMM, AMBER and ECEPP Potentials for Peptides. II. Phi-Psi Maps for N-Acetyl Alanine N-Methyl Amide: Comparisons, Contrasts and Simple Experimental Tests. *Journal of Biomolecular Structure and Dynamics* 7.3, s. 421–453. DOI: 10.1080/07391102.1989.10508503.

192. Hiriart-Urruty JB (1995). Conditions for Global Optimality. *Handbook of Global Optimization*. T. 2. Nonconvex Optimization and Its Applications. Springer Boston, s. 1–26. DOI: 10.1007/978-1-4615-2025-2_1.

193. Korte B, Vygen J (2008). Combinatorial Optimization. Theory and Algorithms. Theory and Algorithms. T. 21. Algorithms and Combinatorics. Springer Berlin Heidelberg.

194. Yu X, Gen M (2010). Introduction to Evolutionary Algorithms. Decision Engineering. Springer London. DOI: 10.1007/978-1-84996-129-5.

195. Branke J (2002). Optimization in Dynamic Environments. *Evolutionary Optimization in Dynamic Environments*. T. 3. Genetic Algorithms and Evolutionary Computation. Springer Boston, s. 13–29. DOI: 10.1007/978-1-4615-0911-0_2.

196. Eberhart R, Shi Y (2001). Tracking and optimizing dynamic systems with particle swarms. *Proceedings of the 2001 Congress on Evolutionary Computation*. IEEE. DOI: 10.1109/cec.2001.934376.

197. Branke J, Schmeck H (2003). Designing Evolutionary Algorithms for Dynamic Optimization Problems. *Advances in Evolutionary Computing*. Natural Computing Series. Springer Berlin Heidelberg, s. 239–262. DOI: 10.1007/978-3-642-18965-4_9.

198. Michalewicz Z (1995). A Survey of Constraint Handling Techniques in Evolutionary Computation Methods. *Proceedings of the 4th Annual Conference on Evolutionary Programming*. The MIT Press, s. 135–155.

199. Michalewicz Z, Schoenauer M (1996). Evolutionary Algorithms for Constrained Parameter Optimization Problems. *Evolutionary Computation* 4.1, s. 1–32. DOI: 10.1162/evco.1996.4.1.1.

200. Abraham A, Jain L (2005). Evolutionary Multiobjective Optimization. *Evolutionary Multiobjective Optimization*. Advanced Information and Knowledge Processing. Springer London, s. 1–6. DOI: 10.1007/1-84628-137-7_1.

201. Luc D (2008). Pareto Optimality. *Pareto Optimality, Game Theory And Equilibria*. T. 17. Springer Optimization and Its Applications. Springer New York, s. 481–515. DOI: 10.1007/978-0-387-77247-9_18.

202. Coello Coello C (2005). Recent Trends in Evolutionary Multiobjective Optimization. *Evolutionary Multiobjective Optimization*. Advanced Information and Knowledge Processing. Springer London, s. 7–32. DOI: 10.1007/1-84628-137-7_2.

203. Coello Coello C (2011). An Introduction to Multi-Objective Particle Swarm Optimizers. *Soft Computing in Industrial Applications*. T. 96. Advances in Intelligent and Soft Computing. Springer Berlin Heidelberg, s. 3–12. DOI: 10.1007/978-3-642-20505-7_1.

204. Segura C, Coello Coello C, Miranda G i in. (2013). Using multi-objective evolutionary algorithms for single-objective optimization. *4OR* 11.3, s. 201–228. DOI: 10.1007/s10288-013-0248-x.

205. Neumann F, Wegener I (2008). Can Single-Objective Optimization Profit from Multiobjective Optimization? *Multiobjective Problem Solving from Nature*. Natural Computing Series. Springer Berlin Heidelberg, s. 115–130. DOI: 10.1007/978-3-540-72964-8_6.

206. Greeff M, Engelbrecht A (2010). Dynamic Multi-objective Optimisation Using PSO. *Multi-Objective Swarm Intelligent Systems*. T. 261. Studies in Computational Intelligence. Springer Berlin Heidelberg, s. 105–123. DOI: 10.1007/978-3-642-05165-4_5.

207. Bui L, Nguyen MH, J. B i in. (2008). Tackling Dynamic Problems with Multiobjective Evolutionary Algorithms. *Multiobjective Problem Solving from Nature*. Natural Computing Series. Springer Berlin Heidelberg, s. 77–91. DOI: 10.1007/978-3-540-72964-8_4.

208. Macready W, Wolpert D (1996). What makes an optimization problem hard? *Complexity* 1.5, s. 40–46. DOI: 10.1002/cplx.6130010511.

209. Weise T, Zapf M, Chiong R i in. (2009). Why Is Optimization Difficult? *Nature-Inspired Algorithms for Optimisation*. Studies in Computational Intelligence 193. Springer Berlin Heidelberg, s. 1–50. DOI: 10.1007/978-3-642-00267-0_1.

210. Nobakhti A (2010). On Natural Based Optimization. *Cognitive Computation* 2.2, s. 97–119. DOI: 10.1007/s12559-010-9039-2.

211. Powell M (1998). Direct search algorithms for optimization calculations. *Acta Numerica* 7, s. 287–336. DOI: 10.1017/s0962492900002841.

212. Indyk P, Motwani R (1998). Approximate nearest neighbors: towards removing the curse of dimensionality. *Proceedings of the thirtieth annual ACM symposium on Theory of computing - STOC '98*, s. 604–613. DOI: 10.1145/276698.276876.

213. Land A, Doig A (1960). An Automatic Method of Solving Discrete Programming Problems. *Econometrica* 28.3, s. 497–520. DOI: 10.2307/1910129.

214. Kiziltan G, Yucaoğlu E (1983). An Algorithm for Multiobjective Zero-One Linear Programming. *Management Science* 29.12, s. 1444–1453. DOI: 10.1287/mnsc.29.12.1444.

215. Przybylski A, Gandibleux X (2017). Multi-objective branch and bound. *European Journal of Operational Research* 260.3, s. 856–872. DOI: 10.1016/j.ejor.2017.01.032.

216. Johnson A, Jacobson S (2002). On the convergence of generalized hill climbing algorithms. *Discrete Applied Mathematics* 119.1-2, s. 37–57. DOI: 10.1016/s0166-218x(01)00264-5.

217. Bonnans J, Gilbert J, Lemaréchal C i in. (2006). Numerical Optimization. Springer Berlin Heidelberg. DOI: 10.1007/978-3-540-35447-5.

218. Stoer J, Bulirsch R (2002). Introduction to Numerical Analysis. T. 12. Texts in Applied Mathematics. Springer New York. DOI: 10.1007/978-0-387-21738-3.

219. Gendreau M, Potvin JY (2010). Handbook of Metaheuristics. T. 146. International Series in Operations Research & Management Science. Springer US. DOI: 10.1007/978-1-4419-1665-5.

220. Kirkpatrick S, Gelatt C, Vecchi M (1983). Optimization by Simulated Annealing. *Science* 220.4598, s. 671–680. DOI: 10.1126/science.220.4598.671.

221. Eiben A, Smith J (2003). What is an Evolutionary Algorithm? *Introduction to Evolutionary Computing*. Natural Computing Series. Springer Berlin Heidelberg, s. 15–35. DOI: 10.1007/978-3-662-05094-1_2.

222. Dorigo M (1992). Optimization, Learning and Natural Algorithms. Prac. dokt. Politecnico di Milano.

223. Wolpert D, Macready W (1997). No Free Lunch Theorems for Optimization. *IEEE Transactions on Evolutionary Computation* 1.1, s. 67–82. DOI: 10.1109/4235.585893.

224. Koppen M, Wolpert D, Macready W (2001). Remarks on a recent paper on the "no free lunch" theorems. *IEEE Transactions on Evolutionary Computation* 5.3, s. 295–296. DOI: 10.1109/4235.930318.

225. Zhang Y, Wang S, Ji G (2015). A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications. *Mathematical Problems in Engineering* 2015, s. 1–38. DOI: 10.1155/2015/931256.

226. Needleman S, Wunsch C (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology* 48.3, s. 443–453. DOI: 10.1016/0022-2836(70)90057-4.

227. Smith T, Waterman M (1981). Identification of common molecular subsequences. *Journal of Molecular Biology* 147.1, s. 195–197. DOI: 10.1016/0022-2836(81)90087-5.

228. Tatusova T, Madden T (1999). BLAST 2 Sequences, a new tool for comparing protein and nucleotide sequences. *FEMS Microbiology Letters* 174.2, s. 247–250. DOI: 10.1111/j.1574-6968.1999.tb13575.x.

229. Blow D (2002). Outline of Crystallography for Biologists. Oxford University Press (OUP). ISBN: 978-0-19-851051-2.

230. Schrauzer G (2000). Selenomethionine A Review of Its Nutritional Significance, Metabolism and Toxicity. *The Journal of Nutrition* 130.7, s. 1653–1656.

231. Zhou X, Alber F, Folkers G i in. (2000). An analysis of the helix-to-strand transition between peptides with identical sequence. *Proteins: Structure, Function, and Genetics* 41.2, s. 248–256. DOI: 10.1002/1097-0134(20001101)41:2<248::aid-prot90>3.0.co;2-j.

232. Jacoboni I, Martelli P, Fariselli P i in. (2000). Predictions of protein segments with the same aminoacid sequence and different secondary structure: A benchmark for predictive methods. *Proteins: Structure, Function, and Genetics* 41.4, s. 535–544. DOI: 10.1002/1097-0134(20001201)41:4<535::aid-prot100>3.0.co;2-c.

233. Xu D, Nussinov R (1998). Favorable domain size in proteins. *Folding and Design* 3.1, s. 11–17. DOI: 10.1016/s1359-0278(98)00004-2.

234. Lin M, Zewail A (2012). Hydrophobic forces and the length limit of foldable protein domains. *Proceedings of the National Academy of Sciences* 109.25, s. 9851–9856. DOI: 10.1073/pnas.1207382109.

235. Rice P, Longden I, Bleasby A (2000). EMBOSS: The European Molecular Biology Open Software Suite. *Trends in Genetics* 16.6, s. 276–277. DOI: 10.1016/s0168-9525(00)02024-2.

236. Williams P, Fülöp V, Yun-Chung L i in. (1995). Pseudospecific docking surfaces on electron transfer proteins as illustrated by pseudoazurin, cytochrome c550 and cytochrome cd1 nitrite reductase. *Nature Structural Biology* 2.11, s. 975–982. DOI: 10.1038/nsb1195-975.

237. Chen Y, Song G, Jiang F i in. (2002). Crystal structure of a staphylokinase variant. *European Journal of Biochemistry* 269.2, s. 705–711. DOI: 10.1046/j.0014-2956.2001.02706.x.

238. Haunerland N, Jacobson B, Wesenberg G i in. (1994). Three-Dimensional Structure of the Muscle Fatty-Acid-Binding Protein Isolated from the Desert Locust Schistocerca gregaria. *Biochemistry* 33.41, s. 12378–12385. DOI: 10.1021/bi00207a004.

239. Ogata H, Nishikawa K, Lubitz W (2015). Hydrogens detected by subatomic resolution protein crystallography in a [NiFe] hydrogenase. *Nature* 520.7548, s. 571–574. DOI: 10.1038/nature14110.

240. Word J, Lovell S, Richardson J i in. (1999). Asparagine and glutamine: using hydrogen atom contacts in the choice of side-chain amide orientation. *Journal of Molecular Biology* 285.4, s. 1735–1747. DOI: 10.1006/jmbi.1998.2401.

241. Arnautova Y, Jagielska A, Scheraga H (2006). A New Force Field (ECEPP-05) for Peptides, Proteins, and Organic Molecules. *Journal of Physical Chemistry B* 110.10, s. 5025–5044. DOI: 10.1021/jp054994x.

242. Poland D, Scheraga H (1967). Energy Parameters in Polypeptides. I. Charge Distributions and the Hydrogen Bond. *Biochemistry* 6.12, s. 3791–3800. DOI: 10.1021/bi00864a024.

243. Yan J, Momany F, Hoffmann R i in. (1970). Energy parameters in polypeptides. II. Semiempirical molecular orbital calculations for model peptides. *Journal of Physical Chemistry* 74.2, s. 420–433. DOI: 10.1021/j100697a031.

244. Momany F, McGuire R, Yan J i in. (1970). Energy parameters in polypeptides. III. Semiempirical molecular orbital calculations for hydrogen-bonded model peptides. *Journal of Physical Chemistry* 74.12, s. 2424–2438. DOI: 10.1021/j100706a003.

245. Momany F, McGuire R, Yan J i in. (1971). Energy parameters in polypeptides. IV. Semiempirical Molecular Orbital Calculations of Conformational Dependence of Energy and Partial Charge in Di- and Tripeptides. *Journal of Physical Chemistry* 75.15, s. 2286–2297. DOI: 10.1021/j100684a011.

246. McGuire R, Momany F, Scheraga H (1972). Energy parameters in polypeptides. V. Empirical hydrogen bond potential function based on molecular orbital calculations. *Journal of Physical Chemistry* 76.3, s. 375–393. DOI: 10.1021/j100647a017.

247. Lewis P, Momany F, Scheraga H (1973). Energy Parameters in Polypeptides. VI. Conformational Energy Analysis of the N-Acetyl N′-Methyl Amides of the Twenty Naturally Occurring Amino Acids. *Israel Journal of Chemistry* 11.2-3, s. 121–152. DOI: 10.1002/ijch.197300017.

248. Dunfield L, Burgess A, Scheraga H (1978). Energy parameters in polypeptides. 8. Empirical potential energy algorithm for the conformational analysis of large molecules. *Journal of Physical Chemistry* 82.24, s. 2609–2616. DOI: 10.1021/j100513a014.

249. Buckingham R (1938). The Classical Equation of State of Gaseous Helium, Neon and Argon. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 168.933, s. 264–283. DOI: 10.1098/rspa.1938.0173.

250. Gut A (2009). An Intermediate Course in Probability. Springer Texts in Statistics. Springer New York. DOI: 10.1007/978-1-4419-0162-0.

251. Levitt M (1976). A simplified representation of protein conformations for rapid simulation of protein folding. *Journal of Molecular Biology* 104.1, s. 59–107. DOI: 10.1016/0022-2836(76)90004-8.

252. Banach M, Konieczny L, Roterman I (2012b). Ligand-binding-site recognition. *Protein Folding in Silico: Protein Folding Versus Protein Structure Prediction*. T. 22. Woodhead Publishing Series in Biomedicine. Woodhead Publishing, s. 79–93. DOI: 10.1533/9781908818256.79.

253. Banach M, Konieczny L, Roterman I (2012c). Use of the "fuzzy oil drop" model to identify the complexation area in protein homodimers. *Protein Folding in Silico: Protein Folding Versus Protein Structure Prediction*. T. 22. Woodhead Publishing Series in Biomedicine. Woodhead Publishing, s. 95–122. DOI: 10.1533/9781908818256.95.

254. Prymula K, Jadczyk T, Roterman I (2010). Catalytic residues in hydrolases: analysis of methods designed for ligand-binding site prediction. *Journal of Computer-Aided Molecular Design* 25.2, s. 117–133. DOI: 10.1007/s10822-010-9402-0.

255. Markley J, Bax A, Arata Y i in. (1998). Recommendations for the presentation of NMR structures of proteins and nucleic acids. *Pure and Applied Chemistry* 70.1, s. 117–142. DOI: 10.1351/pac199870010117.

256. Bryliński M, Konieczny L, Roterman I (2007). Is the protein folding an aim oriented process? Human haemoglobin as example. *International Journal of Bioinformatics Research and Applications* 3.2, s. 234–260. DOI: 10.1504/ijbra.2007.013605.

257. Bryliński M, Prymula K, Jurkowski W i in. (2007). Prediction of Functional Sites Based on the Fuzzy Oil Drop Model. *PLoS Computational Biology* 3.5, e94. DOI: 10.1371/journal.pcbi.0030094.

258. Simm S, Einloft J, Mirus O i in. (2016). 50 years of amino acid hydrophobicity scales: revisiting the capacity for peptide classification. *Biological Research* 49, s. 1–19. DOI: 10.1186/s40659-016-0092-5.

259. Kyte J, Doolittle R (1982). A simple method for displaying the hydropathic character of a protein. *Journal of Molecular Biology* 157.1, s. 105–132. DOI: 10.1016/0022-2836(82)90515-0.

260. Banach M, Prymula K, Konieczny L i in. (2011). "Fuzzy oil drop" model verified positively. *Bioinformation* 5.9, s. 375–377. DOI: 10.6026/97320630005375.

261. Banach M, Prymula K, Jurkowski W i in. (2012a). Fuzzy oil drop model to interpret the structure of antifreeze proteins and their mutants. *Journal of Molecular Modeling* 18.1, s. 229–237. DOI: 10.1007/s00894-011-1033-4.

262. Taylor I, Treiber M, Olivi L i in. (1997). The X-ray structure of the DNA-binding domain from the Saccharomyces cerevisiae cell-cycle transcription factor Mbp1 at 2.1 A resolution. *Journal of Molecular Biology* 272.1, s. 1–8. DOI: 10.1006/jmbi.1997.1229.

263. Banach M, Roterman I (2009). Recognition of protein complexation based on hydrophobicity distribution. *Bioinformation* 4.3, s. 98–100. DOI: 10.6026/97320630004098.

264. Marchewka D, Banach M, Roterman I (2011). Internal force field in proteins seen by divergence entropy. *Bioinformation* 6.8, s. 300–302. DOI: 10.6026/97320630006300.

265. Banach M, Marchewka D, Piwowar M i in. (2012b). The divergence entropy characterizing the internal force field in proteins. *Protein Folding in Silico: Protein Folding Versus Protein Structure Prediction*. T. 22. Woodhead Publishing Series in Biomedicine. Woodhead Publishing, s. 55–77. DOI: 10.1533/9781908818256.55.

266. Steer K, Wirth A, Halgamuge S (2009). The Rationale Behind Seeking Inspiration from Nature. *Nature-Inspired Algorithms for Optimisation*. Studies in Computational Intelligence 193. Springer Berlin Heidelberg, s. 51–76. DOI: 10.1007/978-3-642-00267-0_2.

267. Reynolds C (1987). Flocks, herds and schools: A distributed behavioral model. *SIGGRAPH Computer Graphics* 21.4, s. 25–34. DOI: 10.1145/37402.37406.

268. Heppner H, Grenander U (1990). A stochastic non-linear model for coordinated bird flocks. *The Ubiquity of Chaos*. American Association for the Advancement of Science, s. 233–238. ISBN: 978-0871683502.

269. Eberhart R, Shi Y (1998). Comparison between genetic algorithms and particle swarm optimization. *Evolutionary Programming VII*. Proceedings of the 7th International Conference, EP98 San Diego, California, USA, March 25–27, 1998. T. 1447. Lecture Notes in Computer Science. Springer Berlin Heidelberg, s. 611–616. DOI: 10.1007/bfb0040812.

270. Eberhart R, Shi Y (2000). Comparing inertia weights and constriction factors in particle swarm optimization. *Proceedings of the 2000 Congress on Evolutionary Computation*. DOI: 10.1109/cec.2000.870279.

271. Shi Y, Eberhart R (1998a). A modified particle swarm optimizer. *1998 IEEE International Conference on Evolutionary Computation Proceedings*. DOI: 10.1109/icec.1998.699146.

272. Shi Y, Eberhart R (1998b). Parameter selection in particle swarm optimization. *Evolutionary Programming VII*. Proceedings of the 7th International Conference, EP98 San Diego, California, USA, March 25–27, 1998. T. 1447. Lecture Notes in Computer Science. Springer Berlin Heidelberg, s. 591–600. DOI: 10.1007/bfb0040810.

273. Eberhart R, Kennedy J (1995). A new optimizer using particle swarm theory. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, s. 39–43. DOI: 10.1109/mhs.1995.494215.

274. Clerc M (1999). The swarm and the queen: towards a deterministic and adaptive particle swarm optimization. *Proceedings of the 1999 Congress on Evolutionary Computation*. Institute of Electrical i Electronics Engineers (IEEE). DOI: 10.1109/cec.1999.785513.

275. Clerc M, Kennedy J (2002). The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation* 6.1, s. 58–73. DOI: 10.1109/4235.985692.

276. Kameyama K (2009). Particle Swarm Optimization - A Survey. *IEICE Transactions on Information and Systems* E92-D.7, s. 1354–1361. DOI: 10.1587/transinf.e92.d.1354.

277. Kennedy J, Mendes R (2002). Population structure and particle swarm performance. *Proceedings of the 2002 Congress on Evolutionary Computation, CEC'02*. Institute of Electrical & Electronics Engineers (IEEE), s. 1671–1676. DOI: 10.1109/cec.2002.1004493.

278. Reyes Medina A, Toscano Pulido G, Ramirez Torres J (2009). A Comparative Study of Neighborhood Topologies for Particle Swarm Optimizers. *Proceedings of the International Joint Conference on Computational Intelligence, IJCCI 2009*, s. 152–159.

279. Zielinski K, Laur R (2007). Stopping Criteria for a Constrained Single-Objective Particle Swarm Optimization Algorithm. *Informatica* 31.1, s. 51–59.

280. Banks A, Vincent J, Anyakoha C (2007a). A review of particle swarm optimization. Part I: background and development. *Natural Computing* 6.4, s. 467–484. DOI: 10.1007/s11047-007-9049-5.

281. Banks A, Vincent J, Anyakoha C (2007b). A review of particle swarm optimization. Part II: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications. *Natural Computing* 7.1, s. 109–124. DOI: 10.1007/s11047-007-9050-z.

282. Laskari E, Parsopoulos K, Vrahatis M (2002). Particle swarm optimization for integer programming. *Proceedings of the 2002 Congress on Evolutionary Computation, CEC'02*. T. 2. Institute of Electrical & Electronics Engineers (IEEE), s. 1582–1587. DOI: 10.1109/cec.2002.1004478.

283. Kennedy J, Eberhart R (1997). A discrete binary version of the particle swarm algorithm. *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*. T. 5. Institute of Electrical & Electronics Engineers (IEEE), s. 4104–4108. DOI: 10.1109/icsmc.1997.637339.

284. Bansal J, Deep K (2012). A Modified Binary Particle Swarm Optimization for Knapsack Problems. *Applied Mathematics and Computation* 218.22, s. 11042–11061. DOI: 10.1016/j.amc.2012.05.001.

285. Clerc M (2004). Discrete Particle Swarm Optimization, illustrated by the Traveling Salesman Problem. *New Optimization Techniques in Engineering*. T. 141. Studies in Fuzziness and Soft Computing. Springer Berlin Heidelberg, s. 219–239. DOI: 10.1007/978-3-540-39930-8_8.

286. Parsopoulos K, Vrahatis M (2001). Modification of the Particle Swarm Optimizer for Locating All the Global Minima. *Artificial Neural Nets and Genetic Algorithms*. Springer Vienna, s. 324–327. DOI: 10.1007/978-3-7091-6230-9_80.

287. Parsopoulos K, Vrahatis M (2002a). Recent approaches to global optimization problems through Particle Swarm Optimization. *Natural Computing* 1.2/3, s. 235–306. DOI: 10.1023/a:1016568309421.

288. Parsopoulos K, Plagianakos V, Magoulas G i in. (2001). Objective function "stretching" to alleviate convergence to local minima. *Nonlinear Analysis: Theory, Methods & Applications* 47.5, s. 3419–3424. DOI: 10.1016/s0362-546x(01)00457-6.

289. El Dor A, Clerc M, Siarry P (2011). A multi-swarm PSO using charged particles in a partitioned search space for continuous optimization. *Computational Optimization and Applications* 53.1, s. 271–295. DOI: 10.1007/s10589-011-9449-4.

290. Venter G, Sobieszczanski-Sobieski J (2006). Parallel Particle Swarm Optimization Algorithm Accelerated by Asynchronous Evaluations. *Journal of Aerospace Computing, Information, and Communication* 3.3, s. 123–137. DOI: 10.2514/1.17873.

291. Blackwell T (2007). Particle Swarm Optimization in Dynamic Environments. *Evolutionary Computation in Dynamic and Uncertain Environments*. T. 51. Studies in Computational Intelligence. Springer Berlin Heidelberg, s. 29–49. DOI: 10.1007/978-3-540-49774-5_2.

292. Blackwell T, J. B, Li X (2008). Particle Swarms for Dynamic Optimization Problems. *Swarm Intelligence*. Natural Computing Series. Springer Berlin Heidelberg, s. 193–217. DOI: 10.1007/978-3-540-74089-6_6.

293. Hu X, Eberhart R (2002). Adaptive particle swarm optimization: detection and response to dynamic systems. *Proceedings of the 2002 Congress on Evolutionary Computation*. T. 2, s. 1666–1670. DOI: 10.1109/cec.2002.1004492.

294. Kramer O (2010). A Review of Constraint-Handling Techniques for Evolution Strategies. *Applied Computational Intelligence and Soft Computing* 2010, s. 1–11. DOI: 10.1155/2010/185063.

295. Mezura-Montes E, Flores-Mendoza J (2009). Improved Particle Swarm Optimization in Constrained Numerical Search Spaces. *Nature-Inspired Algorithms for Optimisation*. Studies in Computational Intelligence 193. Springer Berlin Heidelberg, s. 299–332. DOI: 10.1007/978-3-642-00267-0_11.

296. Courant R (1943). Variational methods for the solution of problems of equilibrium and vibrations. *Bulletin of the American Mathematical Society* 49.1, s. 1–24. DOI: 10.1090/s0002-9904-1943-07818-4.

297. Carroll C (1961). The Created Response Surface Technique for Optimizing Nonlinear, Restrained Systems. *Operations Research* 9.2, s. 169–184. DOI: 10.1287/opre.9.2.169.

298. Fiacco A, McCormick G (1966). Extensions of SUMT for Nonlinear Programming: Equality Constraints and Extrapolation. *Management Science* 12.11, s. 816–828. DOI: 10.1287/mnsc.12.11.816.

299. Coello Coello C (1999). *A Survey of Constraint Handling Techniques used with Evolutionary Algorithms*. Spraw. tech. Lania-RI-99-04. Laboratorio Nacional de Informática Avanzada.

300. Coello Coello C (2002). Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer Methods in Applied Mechanics and Engineering* 191.11-12, s. 1245–1287. DOI: 10.1016/s0045-7825(01)00323-1.

301. Mezura-Montes E, Coello Coello C (2008). Constrained Optimization via Multiobjective Evolutionary Algorithms. *Multiobjective Problem Solving from Nature*. Natural Computing Series. Springer Berlin Heidelberg, s. 53–75. DOI: 10.1007/978-3-540-72964-8_3.

302. Deb K (2000). An efficient constraint handling method for genetic algorithms. *Computer Methods in Applied Mechanics and Engineering* 186.2-4, s. 311–338. DOI: 10.1016/s0045-7825(99)00389-8.

303. Richardson J, M. P, Liepins G i in. (1989). Some Guidelines for Genetic Algorithms with Penalty Functions. *Proceedings of the Third International Conference on Genetic Algorithms*. Los Altos, CA, Morgan Kaufmann Publishers, s. 191–197.

304. Powell D, Skolnick M (1993). Using genetic algorithms in engineering design optimization with non-linear constraints. *Proceedings of the Fifth International Conference on Genetic Algorithms*. Morgan Kaufmann, San Mateo, CA, s. 424–431.

305. Cabrera J, Coello Coello C (2007). Handling Constraints in Particle Swarm Optimization Using a Small Population Size. *MICAI 2007: Advances in Artificial Intelligence*. T. 4827. Lecture Notes in Computer Science. Springer Berlin Heidelberg, s. 41–51. DOI: 10.1007/978-3-540-76631-5_5.

306. Coello Coello C, Toscano Pulido G, Lechuga M (2004). Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation* 8.3, s. 256–279. DOI: 10.1109/tevc.2004.826067.

307. Schaffer J (1985). Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. *Proceedings of the First International Conference on Genetic Algorithms*. 93–100.

308. Parsopoulos K, Vrahatis M (2002b). Particle swarm optimization method in multiobjective problems. *Proceedings of the 2002 ACM symposium on Applied computing*. ACM Press. DOI: 10.1145/508791.508907.

309. Deb K, Agrawal S, Pratap A i in. (2000). A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II. *Parallel Problem Solving from Nature PPSN VI*. T. 1917. Lecture Notes in Computer Science. Springer Berlin Heidelberg, s. 849–858. DOI: 10.1007/3-540-45356-3_83.

310. Li X (2003). A Non-dominated Sorting Particle Swarm Optimizer for Multiobjective Optimization. *Genetic and Evolutionary Computation — GECCO 2003*. T. 2723. Lecture Notes in Computer Science. Springer Berlin Heidelberg, s. 37–48. DOI: 10.1007/3-540-45105-6_4.

311. Coello Coello C, Toscano Pulido G (2001). A Micro-Genetic Algorithm for Multiobjective Optimization. *Evolutionary Multi-Criterion Optimization*. T. 1993. Lecture Notes in Computer Science. Springer Berlin Heidelberg, s. 126–140. DOI: 10.1007/3-540-44719-9_9.

312. Fuentes Cabrera J, Coello Coello C (2010). Micro-MOPSO: A Multi-Objective Particle Swarm Optimizer That Uses a Very Small Population Size. *Multi-Objective Swarm Intelligent Systems*. T. 261. Studies in Computational Intelligence. Springer Berlin Heidelberg, s. 83–104. DOI: 10.1007/978-3-642-05165-4_4.

313. Reyes-Sierra M, Coello Coello C (2006). Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art. *International Journal of Computational Intelligence Research* 2.3, s. 287–308. DOI: 10.5019/j.ijcir.2006.68.

314. Reyes-Sierra M, Coello Coello C (2007). A Study of Techniques to Improve the Efficiency of a Multi-Objective Particle Swarm Optimizer. *Evolutionary Computation in Dynamic and Uncertain Environments*. T. 51. Studies in Computational Intelligence. Springer Berlin Heidelberg, s. 269–296. DOI: 10.1007/978-3-540-49774-5_12.

315. Bentley J (1975a). Multidimensional binary search trees used for associative searching. *Communications of the ACM* 18.9, s. 509–517. DOI: 10.1145/361002.361007.

316. Friedman J, Bentley J, Finkel R (1977). An Algorithm for Finding Best Matches in Logarithmic Expected Time. *ACM Transactions on Mathematical Software* 3.3, s. 209–226. DOI: 10.1145/355744.355745.

317. Berg M de, Cheong O, Kreveld M van i in. (2008). Computational Geometry. Algorithms and Applications. Springer Berlin Heidelberg. DOI: 10.1007/978-3-540-77974-2.

318. Arya S, Mount D, Narayan O (1995). Accounting for boundary effects in nearest neighbor searching. *Proceedings of the eleventh annual symposium on Computational geometry - SCG '95*. Association for Computing Machinery (ACM). DOI: 10.1145/220279.220315.

319. Maneewongvatana S, Mount D (2002). Analysis of Approximate Nearest Neighbor Searching with Clustered Point Sets. *Data Structures, Near Neighbor Searches, and Methodology: Fifth and Sixth DIMACS Implementation Challenges*. T. 59. DIMACS Series in Discrete Mathematics and Theoretical Computer Science. AMS, s. 105–123. ISBN: 0-8218-2892-2.

320. Bentley J (1975b). *A Survey of Techniques for Fixed Radius Near Neighbor Searching*. Spraw. tech. Stanford Linear Accelerator Center.

321. Arya S, Mount D, Netanyahu N i in. (1998). An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *Journal of the ACM* 45.6, s. 891–923. DOI: 10.1145/293347.293348.

322. Arya S, Fu HY (2003). Expected-Case Complexity of Approximate Nearest Neighbor Searching. *SIAM Journal on Computing* 32.3, s. 793–815. DOI: 10.1137/s0097539799366340.

323. Rokach L (2010). A survey of Clustering Algorithms. *Data Mining and Knowledge Discovery Handbook*. Springer Boston, s. 269–298. DOI: 10.1007/978-0-387-09823-4_14.

324. Rousseeuw P (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* 20, s. 53–65. DOI: 10.1016/0377-0427(87)90125-7.

325. Rand W (1971). Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association* 66.336, s. 846–850. DOI: 10.1080/01621459.1971.10482356.

326. Santos J, Embrechts M (2009). On the Use of the Adjusted Rand Index as a Metric for Evaluating Supervised Classification. *Artificial Neural Networks – ICANN 2009*. T. 5769. Lecture Notes in Computer Science. Springer Berlin Heidelberg, s. 175–184. DOI: 10.1007/978-3-642-04277-5_18.

327. Hubert L, Arabie P (1985). Comparing partitions. *Journal of Classification* 2.1, s. 193–218. DOI: 10.1007/bf01908075.

328. Park HS, Jun CH (2009). A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications* 36.2, s. 3336–3341. DOI: 10.1016/j.eswa.2008.01.039.

329. Jain A (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters* 31.8, s. 651–666. DOI: 10.1016/j.patrec.2009.09.011.

330. MacQueen J (1967). Some methods for classification and analysis of multivariate observations. *Some methods for classification and analysis of multivariate observations*, s. 281–297.

331. Hartigan J, Wong M (1979). Algorithm AS 136: A K-Means Clustering Algorithm. *Applied Statistics* 28.1, s. 100. DOI: 10.2307/2346830.

332. Lloyd S (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory* 28.2, s. 129–137. DOI: 10.1109/tit.1982.1056489.

333. Aloise D, Deshpande A, Hansen P i in. (2009). NP-hardness of Euclidean sum-of-squares clustering. *Machine Learning* 75.2, s. 245–248. DOI: 10.1007/s10994-009-5103-0.

334. Selim S, Ismail M (1984). K-Means-Type Algorithms: A Generalized Convergence Theorem and Characterization of Local Optimality. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 6.1, s. 81–87. DOI: 10.1109/tpami.1984.4767478.

335. Hadian A, Shahrivari S (2014). High performance parallel k-means clustering for disk-resident datasets on multi-core CPUs. *The Journal of Supercomputing* 69.2, s. 845–863. DOI: 10.1007/s11227-014-1185-y.

336. Kaufman L, Rousseeuw P (1987). Clustering by means of Medoids. *Statistical Data Analysis Based on the L1–Norm and Related Methods*. Elsevier Science Ltd, s. 405–416.

337. Ng R, Han J (1994). Efficient and Effective Clustering Methods for Spatial Data Mining. *Proceedings of the 20th International Conference on Very Large Data Bases*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, s. 144–155.

338. Huang Z (1998). Extensions to the k-Means Algorithm for Clustering Large Data Sets with Categorical Values. *Data Mining and Knowledge Discovery* 2.3, s. 283–304. DOI: 10.1023/a:1009769707641.

339. Pelleg D, Moore A (2000). X-means: Extending K-means with Efficient Estimation of the Number of Clusters. *Proceedings of the Seventeenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, s. 727–734.

340. Arthur D, Vassilvitskii S (2007). k-means++: the advantages of careful seeding. *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms SODA '07*, s. 1027–1035.

341. Jain A, Murty M, Flynn P (1999). Data clustering: a review. *ACM Computing Surveys* 31.3, s. 264–323. DOI: 10.1145/331499.331504.

342. Sneath P, Sokal R (1973). Numerical taxonomy: the principles and practice of numerical classification. Freeman San Francisco.

343. King B (1967). Step-Wise Clustering Procedures. *Journal of the American Statistical Association* 62.317, s. 86–101. DOI: 10.1080/01621459.1967.10482890.

344. Householder A (1958). Unitary Triangularization of a Nonsymmetric Matrix. *Journal of the ACM* 5.4, s. 339–342. DOI: 10.1145/320941.320947.

345. Gentle J (2007). Matrix Algebra. Theory, Computations, and Applications in Statistics. Springer Texts in Statistics. Springer New York. DOI: 10.1007/978-0-387-70873-7.

346. Francis J (1961). The QR Transformation A Unitary Analogue to the LR Transformation–Part 1. *The Computer Journal* 4.3, s. 265–271. DOI: 10.1093/comjnl/4.3.265.

347. Francis J (1962). The QR Transformation–Part 2. *The Computer Journal* 4.4, s. 332–345. DOI: 10.1093/comjnl/4.4.332.

348. Kublanovskaya V (1962). On some algorithms for the solution of the complete eigenvalue problem. *USSR Computational Mathematics and Mathematical Physics* 1.3, s. 637–657. DOI: 10.1016/0041-5553(63)90168-x.

349. Cuppen J (1984). On updating triangular products of Householder reflections. *Numerische Mathematik* 45.3, s. 403–409. DOI: 10.1007/bf01391416.

350. Aragon-Gonzalez G, Aragon J, Rodriguez-Andrade M i in. (2008). Reflections, Rotations, and Pythagorean Numbers. *Advances in Applied Clifford Algebras* 19.1, s. 1–14. DOI: 10.1007/s00006-008-0129-0.

351. Cederberg J (2001). A Course in Modern Geometries. Springer New York. DOI: 10.1007/978-1-4757-3490-4.

352. Pearson K (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series* 2.11, s. 559–572. DOI: 10.1080/14786440109462720.

353. Hotelling H (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology* 24.7, s. 498–520. DOI: 10.1037/h0070888.

354. Jolliffe I (2002). Principal Component Analysis. Springer Series in Statistics. Springer New York. DOI: 10.1007/b98835.

355. Gallier J (2011). Geometric Methods and Applications. Springer New York. DOI: 10.1007/978-1-4419-9961-0.

356. Golub G, Vorst H van der (2000). Eigenvalue computation in the 20th century. *Journal of Computational and Applied Mathematics* 123.1-2, s. 35–65. DOI: 10.1016/s0377-0427(00)00413-1.

357. Golub G, Reinsch C (1970). Singular value decomposition and least squares solutions. *Numerische Mathematik* 14.5, s. 403–420. DOI: 10.1007/bf02163027.

358. Golub G, Kahan W (1965). Calculating the Singular Values and Pseudo-Inverse of a Matrix. *Journal of the Society for Industrial and Applied Mathematics Series B Numerical Analysis* 2.2, s. 205–224. DOI: 10.1137/0702016.

359. Kabsch W (1976). A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica* A32, s. 922–923. DOI: 10.1107/S0567739476001873.

360. Kabsch W (1978). A discussion of the solution for the best rotation to relate two sets of vectors. *Acta Crystallographica* A34, s. 827–828. DOI: 10.1107/S0567739478001680.

361. Maiorov V, Crippen G (1994). Significance of Root-Mean-Square Deviation in Comparing Three-dimensional Structures of Globular Proteins. *Journal of Molecular Biology* 235.2, s. 625–634. DOI: 10.1006/jmbi.1994.1017.

362. Arun K, Huang T, Blostein S (1987). Least-Squares Fitting of Two 3-D Point Sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 9.5, s. 698–700. DOI: 10.1109/tpami.1987.4767965.

363. Umeyama S (1991). Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13.4, s. 376–380. DOI: 10.1109/34.88573.

364. Branke J (1999). Memory enhanced evolutionary algorithms for changing optimization problems. *Proceedings of the 1999 Congress on Evolutionary Computation*, s. 1875–1882. DOI: 10.1109/CEC.1999.785502.

365. Moser I, Chiong R (2013). Dynamic Function Optimization: The Moving Peaks Benchmark. *Metaheuristics for Dynamic Optimization*. T. 433. Studies in Computational Intelligence 333. Springer Berlin Heidelberg, s. 35–59. DOI: 10.1007/978-3-642-30665-5_3.

366. Parsopoulos K, Tasoulis D, Vrahatis M (2004). Multiobjective Optimization using Parallel Vector Evaluated Particle Swarm Optimization. *Proceedings of the International Conference on Artificial Intelligence and Applications*.

367. Voronoi G (1908). Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Premier mémoire. Sur quelques propriétés des formes quadratiques positives parfaites. *Journal für die reine und angewandte Mathematik (Crelle's Journal)* 1908.133, s. 97–102. DOI: 10.1515/crll.1908.133.97.

368. Deb K (1999). Multi-objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems. *Evolutionary Computation* 7.3, s. 205–230. DOI: 10.1162/evco.1999.7.3.205.

369. Srinivas N, Deb K (1994). Muiltiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation* 2.3, s. 221–248. DOI: 10.1162/evco.1994.2.3.221.

370. De Rainville FM, Fortin FA, Gardner MA i in. (2012). DEAP: A Python Framework For Evolutionary Algorithms. *Proceedings of the fourteenth international conference on Genetic and Evolutionary Computation Conference Companion - GECCO Companion '12*. ACM New York, s. 85–92. DOI: 10.1145/2330784.2330799.

371. Hadka D, Reed P (2012). Diagnostic Assessment of Search Controls and Failure Modes in Many-Objective Evolutionary Optimization. *Evolutionary Computation* 20.3, s. 423–452. DOI: 10.1162/evco_a_00053.

372. Zitzler E, Laumanns M, Thiele L (2002). SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. *Evolutionary Methods for Design, Optimisation, and Control*, s. 95–100.

373. Harada K, Ikeda K, Kobayashi S (2006). Hybridization of genetic algorithm and local search in multiobjective function optimization. *Proceedings of the 8th annual conference on Genetic and evolutionary computation - GECCO '06*. ACM New York, s. 667–674. DOI: 10.1145/1143997.1144116.

374. Zitzler E, Deb K, Thiele L (2000). Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation* 8.2, s. 173–195. DOI: 10.1162/106365600568202.

375. Deb K, Jain H (2014). An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. *IEEE Transactions on Evolutionary Computation* 18.4, s. 577–601. DOI: 10.1109/tevc.2013.2281535.

376. Jain H, Deb S (2014). An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. *IEEE Transactions on Evolutionary Computation* 18.4, s. 602–622. DOI: 10.1109/tevc.2013.2281534.

377. Seada H, Deb K (2015). U-NSGA-III: A Unified Evolutionary Optimization Procedure for Single, Multiple, and Many Objectives: Proof-of-Principle Results. *Lecture Notes in Computer Science*. T. 9019. Lecture Notes in Computer Science. Springer Cham, s. 34–49. DOI: 10.1007/978-3-319-15892-1_3.

378. Okabe T, Jin Y, Olhofer M i in. (2004). On Test Functions for Evolutionary Multi-objective Optimization. *Parallel Problem Solving from Nature - PPSN VIII*. T. 3242. Lecture Notes in Computer Science. Springer Berlin Heidelberg, s. 792–802. DOI: 10.1007/978-3-540-30217-9_80.

379. Coello Coello C, Lamont G, Van Veldhuizen D (2007). Evolutionary Algorithms for Solving Multi-Objective Problems. Genetic and Evolutionary Computation Series. Springer US. DOI: 10.1007/978-0-387-36797-2.

380. Osyczka A, S. K (1995). A new method to solve generalized multicriteria optimization problems using the simple genetic algorithm. *Structural Optimization* 10.2, s. 94–99. DOI: 10.1007/bf01743536.

381. Viennet R, Fonteix C, Marc I (1996). Multicriteria optimization using a genetic algorithm for determining a Pareto set. *International Journal of Systems Science* 27.2, s. 255–260. DOI: 10.1080/00207729608929211.

382. Okabe T, Yaochu J, Sendhoff B (2003). A critical survey of performance indices for multi-objective optimisation. *The 2003 Congress on Evolutionary Computation, 2003. CEC '03.* DOI: 10.1109/cec.2003.1299759.

383. Van Veldhuizen D (1999). Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations. Prac. dokt. Department of Electrical i Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio, USA.

384. Zitzler E, Thiele L (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation* 3.4, s. 257–271. DOI: 10.1109/4235.797969.

385. Barber C, Dobkin D, Huhdanpaa H (1996). The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software* 22.4, s. 469–483. DOI: 10.1145/235815.235821.

386. Clarkson K, Shor P (1989). Applications of random sampling in computational geometry, II. *Discrete & Computational Geometry* 4.1, s. 387–421. DOI: 10.1007/bf02187740.

387. Ramos E (2001). An Optimal Deterministic Algorithm for Computing the Diameter of a Three-Dimensional Point Set. *Discrete & Computational Geometry* 26.2, s. 233–244. DOI: 10.1007/s00454-001-0029-8.

388. Shamos M (1975). Geometric complexity. *Proceedings of seventh annual ACM symposium on Theory of computing - STOC '75*, s. 224–233. DOI: 10.1145/800116.803772.

389. Xu Z, Horwich A, Sigler P (1997). The crystal structure of the asymmetric GroEL – GroES – (ADP)7 chaperonin complex. *Nature* 388.6644, s. 741–750. DOI: 10.1038/41944.

390. Itou H, Watanabe H, Yao M i in. (2010). Crystal Structures of the Multidrug Binding Repressor Corynebacterium glutamicum CgmR in Complex with Inducers and with an Operator. *Journal of Molecular Biology* 403.2, s. 174–184. DOI: 10.1016/j.jmb.2010.07.042.

391. Shimizu K, Kuroishi C, Sugahara M i in. (2008). Structure of peptidyl-tRNA hydrolase 2 from Pyrococcus horikoshii OT3: insight into the functional role of its dimeric state. *Acta Crystallographica Section D* 64.4, s. 444–453. DOI: 10.1107/s0907444908002850.

392. Sun C, Nettesheim D, Liu Z i in. (2005). Solution Structure of Human Survivin and Its Binding Interface with Smac/Diablo. *Biochemistry* 44.1, s. 11–17. DOI: 10.1021/bi0485171.

393. Yan Y, Winograd E, Viel A i in. (1993). Crystal structure of the repetitive segments of spectrin. *Science* 262.5142, s. 2027–2030. DOI: 10.1126/science.8266097.

394. Li M, Gustchina A, Matúz K i in. (2011). Structural and biochemical characterization of the inhibitor complexes of xenotropic murine leukemia virus-related virus protease. *FEBS Journal* 278.22, s. 4413–4424. DOI: 10.1111/j.1742-4658.2011.08364.x.

395. Okada U, Kondo K, Hayashi T i in. (2008). Structural and functional analysis of the TetR-family transcriptional regulator SCO0332 from Streptomyces coelicolor. *Acta Crystallographica Section D* 64.2, s. 198–205. DOI: 10.1107/s0907444907059835.

396. Shindyalov I, Bourne P (1998). Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Engineering Design & Selection* 11.9, s. 739–747. DOI: 10.1093/protein/11.9.739.

397. Leesong M, Henderson B, Gillig J i in. (1996). Structure of a dehydratase–isomerase from the bacterial pathway for biosynthesis of unsaturated fatty acids: two catalytic activities in one active site. *Structure* 4.3, s. 253–264. DOI: 10.1016/s0969-2126(96)00030-5.

398. Andre I, Strauss C, Kaplan D i in. (2008). Emergence of symmetry in homooligomeric biological assemblies. *Proceedings of the National Academy of Sciences* 105.42, s. 16148–16152. DOI: 10.1073/pnas.0807576105.

399. Crick F, Watson J (1956). Structure of Small Viruses. *Nature* 177.4506, s. 473–475. DOI: 10.1038/177473a0.

400. Goodsell D, Olson A (2000). Structural Symmetry and Protein Function. *Annual Review of Biophysics and Biomolecular Structure* 29.1, s. 105–153. DOI: 10.1146/annurev.biophys.29.1.105.

401. Brown J (2006). Breaking symmetry in protein dimers: Designs and functions. *Protein Science* 15.1, s. 1–13. DOI: 10.1110/ps.051658406.

402. Swapna L, Srikeerthana K, Srinivasan N (2012). Extent of Structural Asymmetry in Homodimeric Proteins: Prevalence and Relevance. *PLoS ONE* 7.5, e36688. DOI: 10.1371/journal.pone.0036688.

403. Stroupe C, Brunger A (2000). Crystal Structures of a Rab Protein in its Inactive and Active Conformations. *Journal of Molecular Biology* 304.4, s. 585–598. DOI: 10.1006/jmbi.2000.4236.

404. Maleki M, Vasudev G, Rueda L (2013). The role of electrostatic energy in prediction of obligate protein-protein interactions. *Proteome Science* 11.Suppl 1, S11. DOI: 10.1186/1477-5956-11-s1-s11.

405. Vince J (2011). 3D Rotation Transforms. *Quaternions for Computer Graphics*. Springer London, s. 73–88. DOI: 10.1007/978-0-85729-760-0_6.

406. Kanatani K (1990). Representation of 3D Rotations. *Group-Theoretical Methods in Image Understanding*. T. 20. Springer Series in Information Science. Springer Berlin Heidelberg, s. 197–235. DOI: 10.1007/978-3-642-61275-6_6.

407. Rodrigues O (1816). De l'attraction des sphéroïdes. *Correspondence sur l'École Impériale Polytechnique* 3.3, s. 361–385.

408. IEEE Standard for Floating-Point Arithmetic (2008). DOI: 10.1109/ieeestd.2008.4610935.

409. Fawcett T (2006). An introduction to ROC analysis. *Pattern Recognition Letters* 27.8, s. 861–874. DOI: 10.1016/j.patrec.2005.10.010.

410. Fernandes C, Marchi-Salvador D, Salvador G i in. (2010). Comparison between apo and complexed structures of bothropstoxin-I reveals the role of Lys122 and Ca2+-binding loop region for the catalytically inactive Lys49-PLA2s. *Journal of Structural Biology* 171.1, s. 31–43. DOI: 10.1016/j.jsb.2010.03.019.

411. Blaszczyk J, Tropea J, Bubunenko M i in. (2001). Crystallographic and Modeling Studies of RNase III Suggest a Mechanism for Double-Stranded RNA Cleavage. *Structure* 9.12, s. 1225–1236. DOI: 10.1016/s0969-2126(01)00685-2.

412. Dolot R, Ozga M, Wlodarczyk A i in. (2012). A new crystal form of human histidine triad nucleotide-binding protein 1 (hHINT1) in complex with adenosine 5'-monophosphate at 1.38 AA resolution. *Acta Crystallographica Section F Structural Biology and Crystallization Communications* 68.8, s. 883–888. DOI: 10.1107/s1744309112029491.

413. Yeh J, Biemann HP, Prive G i in. (1996). High-resolution Structures of the Ligand Binding Domain of the Wild-type Bacterial Aspartate Receptor. *Journal of Molecular Biology* 262.2, s. 186–201. DOI: 10.1006/jmbi.1996.0507.

414. Tong Q, Wang F, Zhou HZ i in. (2012). Structural and functional insights into lipid-bound nerve growth factors. *The FASEB Journal* 26.9, s. 3811–3821. DOI: 10.1096/fj.12-207316.

415. Rodriguez H, Angulo I, Rivas B de las i in. (2010). P-Coumaric acid decarboxylase from Lactobacillus plantarum: Structural insights into the active site and decarboxylation catalytic mechanism. *Proteins* 78.7, s. 1662–1676. DOI: 10.1002/prot.22684.

416. Chung C, Coste H, White J i in. (2011). Discovery and Characterization of Small Molecule Inhibitors of the BET Family Bromodomains. *Journal of Medicinal Chemistry* 54.11, s. 3827–3838. DOI: 10.1021/jm200108t.

417. Chen HY, Yuan Y (2010). Crystal Structure of Mj1640/DUF358 Protein Reveals a Putative SPOUT-Class RNA Methyltransferase. *Journal of Molecular Cell Biology* 2.6, s. 366–374. DOI: 10.1093/jmcb/mjq034.

418. Bianchet M, Ahmed H, Vasta G i in. (2000). Soluble β-galactosyl-binding lectin (galectin) from toad ovary: Crystallographic studies of two protein-sugar complexes. *Proteins: Structure, Function, and Genetics* 40.3, s. 378–388. DOI: 10.1002/1097-0134(20000815)40:3<378::aid-prot40>3.0.co;2-7.

419. Jung J, Kim JK, Yeom SJ i in. (2011). Crystal structure of Clostridium thermocellum ribose-5-phosphate isomerase B reveals properties critical for fast enzyme kinetics. *Applied Microbiology and Biotechnology* 90.2, s. 517–527. DOI: 10.1007/s00253-011-3095-8.

420. Perez F, Granger B (2007). IPython: A System for Interactive Scientific Computing. *Computing in Science & Engineering* 9.3, s. 21–29. DOI: 10.1109/mcse.2007.53.

421. Oliphant T (2007). Python for Scientific Computing. *Computing in Science & Engineering* 9.3, s. 10–20. DOI: 10.1109/MCSE.2007.58.

422. Hunter J (2007). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering* 9.3, s. 90–95. DOI: 10.1109/MCSE.2007.55.

423. Izzo D (2012). PyGMO and PyKEP: Open Source Tools for Massively Parallel Optimization in Astrodynamics (the case of interplanetary trajectory optimization). *Proceedings of the International Conference on Astrodynamics Tools and Techniques - ICATT, 2012.*

424. Hagberg A, Schult D, Swart P (2008). Exploring Network Structure, Dynamics, and Function using NetworkX. *Proceedings of the 7th Python in Science Conference*, s. 11–15.

425. Da Veiga Beltrame E, Tyrwhitt-Drake J, Roy I i in. (2017). 3D Printing of Biomolecular Models for Research and Pedagogy. *Journal of Visualized Experiments* 121, e55427. DOI: 10.3791/55427.

426. Cock P, Antao T, Chang J i in. (2009). Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* 25.11, s. 1422–1423. DOI: 10.1093/bioinformatics/btp163.

427. Marr D, Binns F, Hill D i in. (2002). Hyper-Threading Technology Architecture and Microarchitecture. *Intel Technology Journal* 6.1, s. 11–15.

428. Du P, Weber R, Luszczek P i in. (2012). From CUDA to OpenCL: Towards a performance-portable solution for multi-platform GPU programming. *Parallel Computing* 38.8, s. 391–407. DOI: 10.1016/j.parco.2011.10.002.

Spis rysunków

Spis tabel

Spis równań

Spis definicji

Wykaz skrótów

Å	ångström	$1\,\text{Å} = 10^{-10}$ m = 0,1 nm
ADF	approximation distance – front	odległość przybliżenia optymalnego frontu Pareto
ADS	approximation distance – set	odległość przybliżenia optymalnego zbioru Pareto
AUC	area under curve	pole pod krzywą ROC
ARC	AUC and RMSD combined	miara łącząca miary AUC i RMSD
CATH	class architecture topology/fold homologous superfamily	baza domen CATH
ECEPP	empirical conformational energy program for peptides	chemiczne pole siłowe ECEPP
ER	error ratio	miara błędu (liczba rozwiązań zdominowanych)
FOD	fuzzy oil drop	model rozmytej kropli oliwy
FOD-MAX	fuzzy oil drop – maximum distance	algorytm układania zbioru atomów efektywnych białka oparty na średnicach
FOD-PCA	fuzzy oil drop – principal component analysis	algorytm układania zbioru atomów efektywnych białka oparty na PCA
FPR	false positive rate	1-swoistość
ICF	interface common fraction	ułamek wspólnego interfejsu
ITF	interface total fraction	ułamek całości reszt w interfejsie
MPB	moving peaks benchmark	test ruchomych wierzchołków
MOPSO	multiobjective particle swarm optimization	wielokryterialna optymalizacja rojem cząstek
MOSF	multi objective swarm families	wielokryterialne rodziny rojów
NC	niche count	miara niszowania
NMR	nuclear magnetic resonance	magnetyczny rezonans jądrowy

NSGA	nondominated sorting genetic algorithm	algorytm optymalizacji wielokryterialnej NSGA-II
NSPSO	nondominated sorting particle swarm optimization	algorytm optymalizacji wielokryterialnej NSPSO
PCA	principal component analysis	analiza składowych głównych
PDB	protein data bank	baza struktur białek PDB
PSO	particle swarm optimization	optymalizacja rojem cząstek
RCSB	research collaboration for structural biology	instytucja zarządzająca główną bazą PDB
RD	relative distance	odległość względna rozkładów hydrofobowości
RMSD	root mean square deviation	pierwiastek średniej kwadratów różnicy
ROC	receiver operating characteristic	krzywa ROC
TPR	true positive rate	czułość
SVD	singular value decomposition	rozkład według wartości osobliwych
XRD	x-ray diffraction	rentgenografia strukturalna