mgr Mateusz Pikora, MWB UG

Promotor: prof. UG dr hab. Rajmund Kaźmierkiewicz, MWB UG

Promotor pomocniczy: dr Artur Giełdoń, WChem UG

Zastosowanie modelu Markova do badania ścieżek zwijania białek

21 czerwca 2019

Bardzo dziękuję moim promotorom, profesorowi Rajmundowi Kaźmierkiewiczowi oraz doktorowi Arturowi Giełdoniowi za opiekę, poświęcony czas, wyrozumiałość oraz wiele cennych rad otrzymanych podczas wykonywania niniejszej pracy.

Spis treści

IV. Wyniki

Spis rysunków

Wykaz skrótów

GCC GNU Compiler Collection.

MD dynamika molekularna (ang. Molecular Dynamics).

MPI Message Passing Interface.

MREMD multipleksowa dynamika molekularna z wymianą replik (ang. Multiplexed Replica Exchange Molecular Dynamics).

NMR magnetyczny rezonans jądrowy (ang. Nuclear Magnetic Resonance).

OpenMP Open Multi-Processing.

PDB Protein Data Bank.

REMD dynamika molekularna z wymianą replik (ang. Replica Exchange Molecular Dynamics).

RMSD odchylenie średniokwadratowe (ang. Root Mean Square Deviation).

Streszczenie

Białka są liniowymi polimerami najczęściej pochodzenia biologicznego składającymi się z reszt aminokwasowych połączonych wiązaniami peptydowymi. Przyjmują one złożoną, kilkupoziomową strukturę trójwymiarową warunkująca pełnienie przez białka wyspecjalizowanych funkcji. Badanie tej struktury jest istotnym zagadnieniem w biologii molekularnej i biochemii. W żywych organizmach białka pełnią wiele różnorakich funkcji, do najważniejszych możemy zaliczyć enzymatyczną, strukturalną, mechaniczną i receptorową.

Dynamika molekularna jest techniką pozwalającą na symulowanie ruchu atomów w danym systemie przy wykorzystaniu równań dynamiki Newtona. Dynamika molekularna z wymianą replik jest odmianą tej metody, w której jednocześnie symulowane jest kilka kopii (replik) układu w różnych temperaturach. W określonych odstępach czasu temperatury są wymieniane pomiędzy replikami, co pomaga w wyjściu z lokalnych minimów energetycznych. Wynikiem takiej symulacji jest seria kolejnych struktur reprezentujących ewolucję badanego układu molekularnego w czasie, razem z informacjami o ich energii i temperaturze występowania. Jednym ze sposobów analizy tych danych jest analiza skupień (grupowanie). To proces dzielenia struktur w grupy pod względem określonych kryteriów, takich jak podobieństwo strukturalne. Poszczególne grupy reprezentują stany przejściowe układu, a przejścia pomiędzy nimi mogą być analizowane z wykorzystaniem metody Łańcuchów Markova. Łańcuch Markova to matematyczny, stochastyczny model opisujący zmiany stanu układu w czasie.

Celem niniejszej pracy doktorskiej było zbadanie mechanizmu fałdowania, czyli sposobu uzyskiwania struktury trzeciorzędowej, dla kilku wybranych białek. Chciałem zweryfikować następującą hipotezę badawczą: Białka nie posiadają jednej dominującej ścieżki fałdowania rozumianej jako jednoznaczna i powtarzalna sekwencja zdarzeń prowadząca od struktury całkowicie rozwiniętej do konformacji natywnej. Dodatkowo struktura natywna nie jest idealnie stabilna i białko może z niej rozwijać się w różne,

częściowo zwinięte struktury. Tworzy się w ten sposób sieć częściowo stabilnych stanów którą chciałem pokazać.

Moją strategią umożliwiającą osiągnięcie tego celu było przeprowadzenie analizy skupień struktur pochodzących z symulacji, budowa modelu Markova na jej podstawie i uzyskanie diagramów przejść pomiędzy otrzymanymi grupami struktur. Użyłem do tego zmodyfikowanego przeze mnie Algorytmu Najbliższego Sąsiada. Środkiem technicznym do realizacji celu pracy był program, który zaprojektowałem i napisałem w języku C z wykorzystaniem biblioteki OpenMP służącej do przeprowadzenia obliczeń równoległych. We współpracy z Wydziałem Chemii UG przeprowadziłem symulacje dynamiki molekularnej z wymianą replik wybranych białek, które następnie przeanalizowałem przy pomocy wspomnianego programu. Udało mi się pozytywnie zweryfikować moją hipotezę badawczą dla kilku wybranych białek w dwóch różnych polach siłowych.

Summary

Proteins are linear polymers, usually of biological origins. They consist of amino-acid residues linked by peptide bonds. They have complicated, multi-level, three-dimensional structure, which determines their specialized functions. Studying this structure is an important issue in molecular biology and biochemistry. In living organisms proteins fulfill many different roles, most importantly enzymatic, structural, mechanical and signaling.

Molecular dynamics is a technique enabling simulation of atoms in a given system using Newtonian mechanics. Replica exchange molecular dynamics is a subtype of this method, where several copies (replicas) of given system are simulated simultaneously in different temperatures. Temperatures are exchanged between replicas at defined time intervals, which helps in escaping local energy minima. The outcome of such simulation is a time series of structures representing evolution of studied molecular system, with information about each structure energy and temperature. One method used to analyze this data is clustering. It is a process of dividing structures into groups by defined criteria, such as structural similarity. Different groups represent different system states and transitions between them can be analyzed with Markov Chains. Markov Chain is a mathematical, stochastic model of behavior of given system.

The aim of this doctoral thesis was to investigate the folding mechanism, i.e. the path molecules follow to accomodate a tertiary structure, for several selected proteins. I wanted to verify the following research hypothesis: Proteins do not have one dominant folding pathway understood as an unambiguous and repetitive sequence of events leading from a fully extended structure to a native conformation. Moreover, the native structure is not perfectly stable and a protein can unfold into various, partially folded structures. This creates a network of partially stable states, which I wanted to show.

My strategy to achieve this goal was to use the clustering analysis of molecular structures obtained from simulations, and to build a Markov model using obtained clusters and subsequent preparation of transition graphs between obtained groups

of structures. To reach it I used a modified Nearest Neighbor Algorithm. The technical means to achieve my goal was a computer program, which I designed and written using C programming language with OpenMP library for parallelization. In cooperation with the Chemistry Faculty of UG I carried out replica exchange molecular dynamics simulations of chosen proteins and analyzed their results with mentioned program. I succesfully verified my research hypothesis for several selected proteins in two different force fields.

Część I

Wstęp

1. Białka

Białka są jednym z rodzajów makromolekuł biologicznych, czyli dużych cząsteczek organicznych pełniących różnorakie funkcje w organizmach żywych. Poza białkami zaliczamy do nich także kwasy nukleinowe, lipidy i polisacharydy. Białka składają się z połączonych liniowo reszt aminokwasowych. Przyjmują one skomplikowaną, kilkupoziomową strukturę przestrzenną, która pozwala im na pełnienie określonych funkcji biologicznych. Są, między innymi, ważnymi katalizatorami reakcji chemicznych zachodzących w organizmie[1].

1.1. Budowa

Białka są liniowymi polimerami składającymi się z reszt aminokwasowych połączonych wiązaniami peptydowymi. Najmniejsze zbudowane są z około 50 reszt aminokwasowych, natomiast największe mogą się składać z kilku tysięcy. Krótsze polimery złożone z reszt aminokwasowych nazywane są peptydami. Granica wielkości pomiędzy peptydami a białkami jest nieostra. Zwykle przyjmuje się, że białka posiadają zdefiniowaną strukturę trzeciorzędową, natomiast peptydy posiadają dużą swobodę konformacyjną i ich struktura jest reprezentowana przez zbiór konformacji[1, 2].

1.1.1. Aminokwasy

Aminokwasy są związkami organicznymi posiadającymi grupę aminową (-NH_2) i grupę karboksylową (-COOH). W aminokwasach białkowych (czyli występujących naturalnie w białkach) te 2 grupy funkcyjne przyłączone są kowalencyjnie do tego samego atomu węgla - tzw. węgla α. Takie aminokwasy nazywamy α-aminokwasami. Aminokwasy w których te grupy przyłączone są do różnych atomów węgla nazywamy w zależności od liczby atomów pomiędzy tymi grupami: β-aminokwasami, γ-aminokwasami

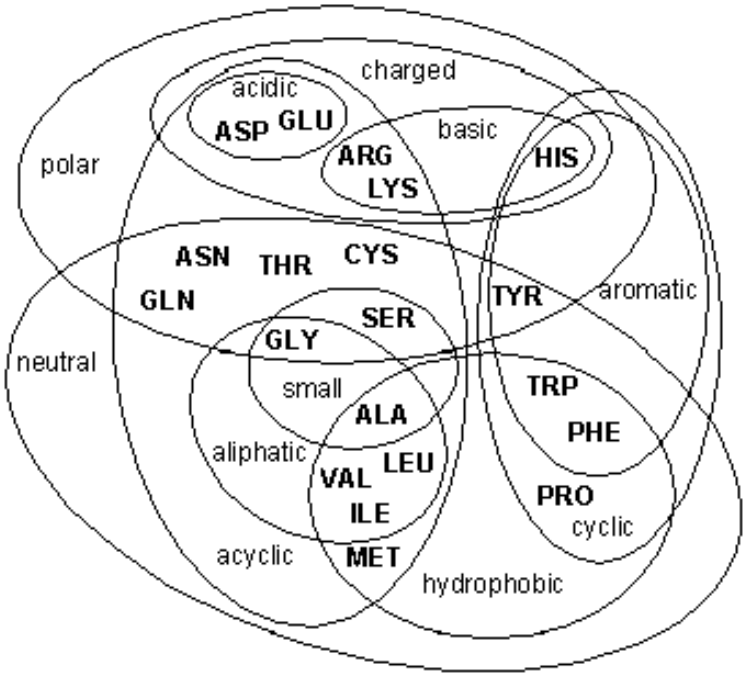
itd. W aminokwasach białkowych do atomu węgla α przyłączone są kowalencyjnie również atom wodoru oraz łańcuch boczny (oznaczany R), czyli grupa atomów decydująca o jego właściwościach chemicznych w białku. W pH obojętnym aminokwasy występują w formie jonu obojnaczego, czyli ze sprotonowaną grupą aminową (-NH_3^+) i zjonizowaną grupą karboksylową (-COO^-). Dzięki temu mogą dobrze rozpuszczać się w wodzie[3].

W przyrodzie wyróżniamy 20 podstawowych aminokwasów białkowych. Wszystkie poza glicyną posiadają izomery optyczne (enancjomery) oznaczane jako L i D. Wynika to z obecności asymetrycznego (chiralnego) atomu węgla α. W białkach występujących naturalnie znajdują się przeważnie izomery L aminokwasów. Izomery D występują w organizmach żywych dużo rzadziej. Wchodzą na przykład w skład ścian komórkowych bakterii Gram-dodatnich i odkryto je w tkance mózgowej[1, 2]. Istnieje niewielka grupa dodatkowych aminokwasów białkowych wykorzystywanych przez niektóre organizmy. Zaliczają się do nich selenocysteina i pirolizyna[4]. Po utworzeniu białka łańcuchy boczne reszt aminokwasowych mogą ulegać wielu różnym dodatkowym modyfikacjom chemicznym zmieniającym ich właściwości (stabilność, rozpuszczalność, sposób oddziaływania z innymi aminokwasami). Nazywamy je modyfikacjami posttranslacyjnymi[5].

Podstawowe aminokwasy białkowe dzielimy na kategorie w zależności od właściwości ich łańcuchów bocznych. Pod względem budowy wyróżniamy grupy aminokwasów aromatycznych i alifatycznych. Pod względem ładunku i polarności niepolarne (hydrofobowe), polarne bez ładunku, kwaśne i zasadowe. Wyróżniamy także grupę aminokwasów zawierających siarkę. Każdy aminokwas białkowy posiada dwa skrótowe identyfikatory: jednoliterowy i trzyliterowy[1]. Ich wzory strukturalne oraz obydwa rodzaje skróconych oznaczeń przedstawione są na rysunku 1.2. Podział aminokwasów białkowych na kategorie pokazuje rysunek 1.1.

Oprócz bycia budulcem białek aminokwasy pełnią także inne funkcje biologiczne. Są między innymi:

— Substratami biosyntezy zasad azotowych wchodzących w skład kwasów nukleinowych (puryn i pirymidyn) oraz innych związków.
— Składnikami peptydów będących hormonami, neuromodulatorami i neuroprzekaźnikami.



Rysunek 1.1. Diagram Venna przedstawiający podział aminokwasów białkowych na kategorie. Źródło: [6]

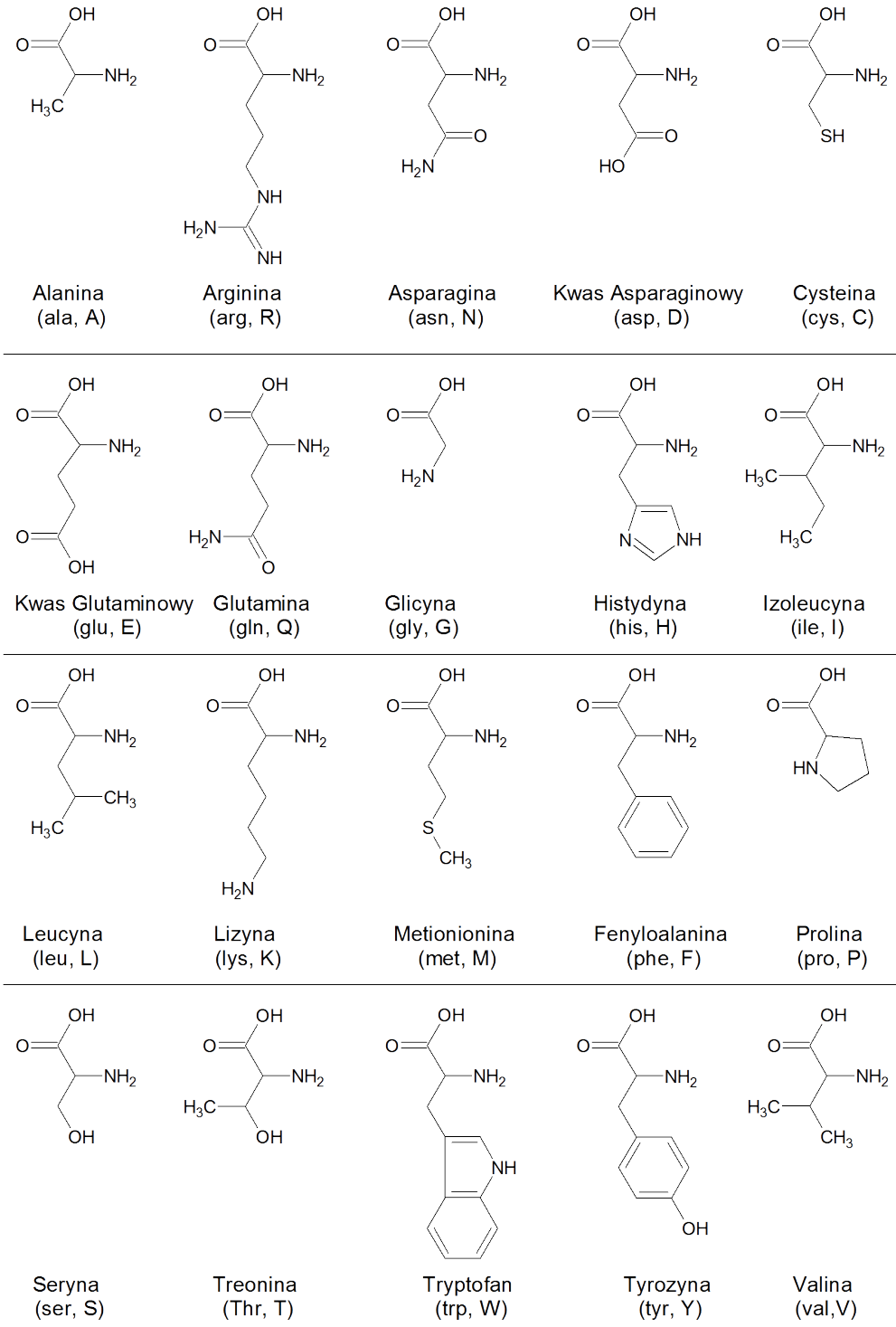
— Składnikami peptydów będących antybiotykami i lekami przeciwnowotworowymi.

— Składnikami peptydów będących toksynami mikroorganizmów[2].

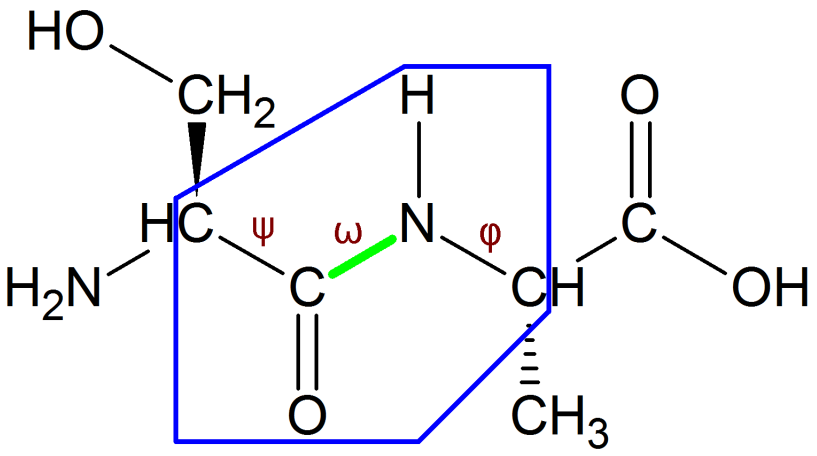
1.1.2. Wiązanie peptydowe

Grupa α-aminowa jednego i α-karboksylowa następującego po nim aminokwasu uczestniczą w tworzeniu charakterystycznego wiązania amidowego, nazywanego także wiązaniem peptydowym. Ze względu na ten sposób łączenia monomerów określa się białka jako liniowe polimery. Powstanie wiązania peptydowego wymaga dostarczenia energii, ale samo wiązanie jest stabilne kinetycznie. Reakcja ta powoduje także uwolnienie cząsteczki wody (H_2O). W związku z tym aminokwasy połączone wiązaniem peptydowym formalnie nazywa się resztami aminokwasowymi[1].

Wiązanie peptydowe jest płaskie. Ma ono częściowo charakter wiązania podwójnego, co uniemożliwia rotację wokół niego i nakłada ograniczenia na konformacje jakie może przyjąć łańcuch polipeptydowy. W dwóch resztach aminokwasowych nim połączonych w jednej płaszczyźnie występuje sześć atomów: obydwa atomy węgla α oraz grupy CO



Rysunek 1.2. Wzory strukturalne przedstawiające 20 podstawowych aminokwasów występujących w białkach, wraz z ich oznaczeniami jedno i trzyliterowymi. Źródło: Własne na podstawie [3].

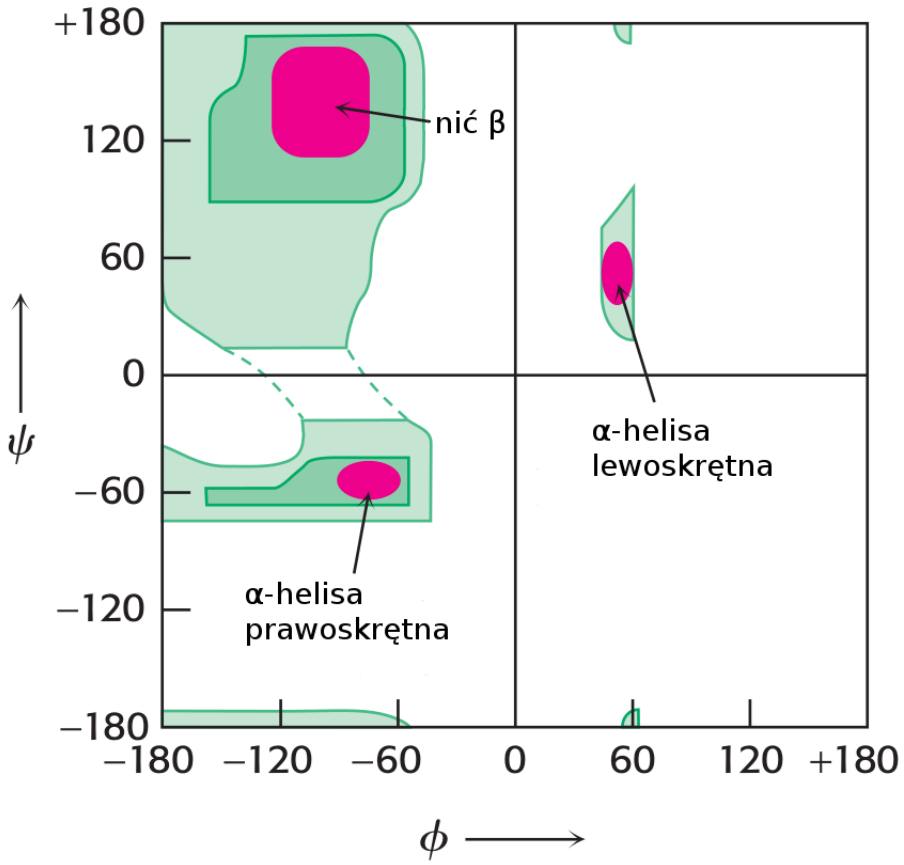


Rysunek 1.3. Dwie reszty aminokwasowe połączone wiązaniem peptydowym (zielony) w konformacji *trans*. Niebieskim obramowaniem zaznaczono płaski fragment cząsteczki. Czerwonym kolorem oznaczono kąty torsyjne występujące w łańcuchu głównym. Źródło: Własne na podstawie [1].

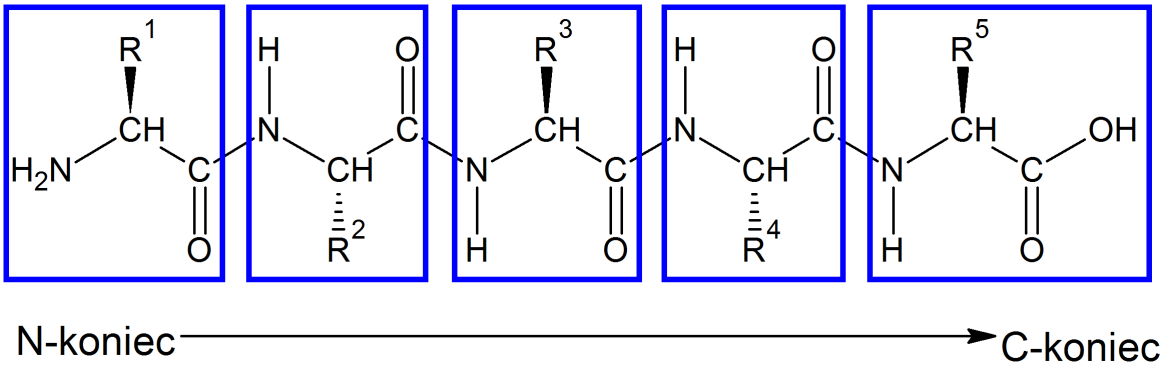
i NH tworzące wiązanie peptydowe. W wyniku tego wiązanie peptydowe może przyjmować dwie konformacje różniące się od siebie obrotem o 180° wokół tego wiązania: *cis* i *trans*. Ze względu na zawady steryczne w konformacji *cis* konformacja *trans* występuje w białkach dużo częściej. Ilustracja 1.3 przedstawia dwie reszty aminokwasowe połączone wiązaniem peptydowym w konformacji *trans* i płaski fragment cząsteczki. W łańcuchu głównym w każdej reszcie aminokwasowej możliwy jest obrót wokół dwóch wiązań. Pierwszym jest wiązanie pomiędzy grupą aminową a atomem węgla α. Kąt rotacji wokół niego nazywamy *fi* (φ). Drugim jest wiązanie pomiędzy atomem węgla α a atomem węgla należącym do grupy karbonylowej. Kąt rotacji wokół niego nazywamy *psi* (ψ). Z powodu zawad sterycznych w białkach występują tylko pewne, określone kombinacje wartości tych kątów[1]. Wizualizuje to wykres Ramachandrana. Dozwolone kombinacje wartości kątów φ i ψ skupiają się w kilku fragmentach tego wykresu[7]. Wykres ten pokazany jest na rysunku 1.4.

1.1.3. Struktura pierwszorzędowa białek

Polimer zbudowany z reszt aminokwasowych połączonych wiązaniem peptydowym nazywamy łańcuchem polipeptydowym. Na jednym jego końcu znajduje się grupa α-aminowa i nazywamy go N-końcem. Na drugim znajduje się grupa α-karboksylowa i nazywamy go C-końcem. Przez obecność tych dwóch różnych końców mówimy, że łań-



Rysunek 1.4. Wykres Ramachandrana. Na osiach x i y przedstawiono odpowiednio wartości kątów φ i ψ. Kolor zielony pokazuje ich kombinacje występujące w białkach. Ciemniejszym kolorem zaznaczono najczęstsze wartości. Kolorem różowym oznaczono obszary wykresu odpowiadające strukturom drugorzędowym. Źródło: [1].



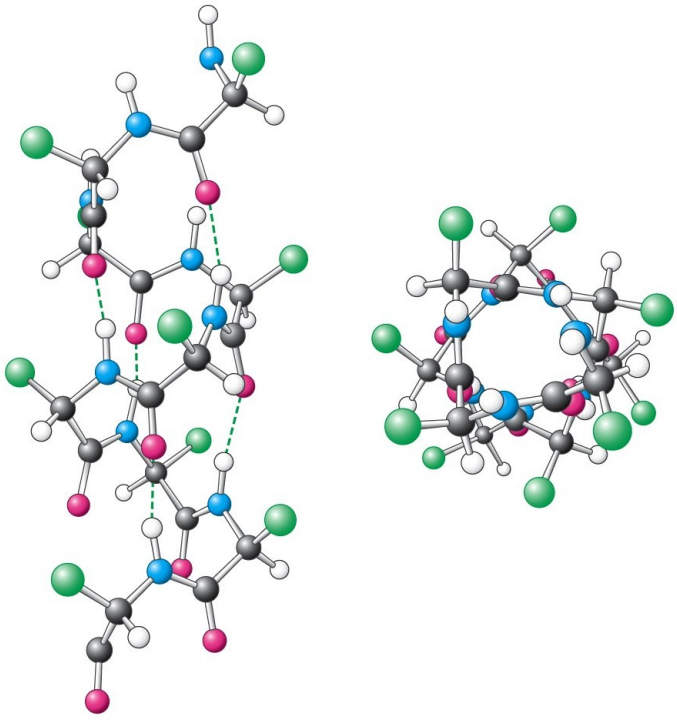
Rysunek 1.5. Schematycznie przedstawiony łańcuch peptydowy. Niebieskim obramowaniem zaznaczono poszczególne reszty aminokwasowe. Źródło: Własne na podstawie [1].

cuch polipeptydowy jest spolaryzowany. Łańcuch polipeptydowy dzielimy na łańcuch główny i łańcuchy boczne. W skład łańcucha głównego wchodzą atomy węgla α oraz grupy peptydowe łączące kolejne reszty aminokwasowe. Kolejność reszt aminokwasowych w łańcuchu polipeptydowym, rozpoczynając od N-końca nazywamy strukturą pierwszorzędową białka[3]. Przykładowy łańcuch peptydowy pokazuje rysunek 1.5.

W wielu białkach pomiędzy resztami aminokwasowymi występują wiązania kowalencyjne inne niż peptydowe. Najczęściej występującym wiązaniem tego typu jest mostek disulfidowy. Powstaje on na skutek utlenienia dwóch reszt tiolowych (-SH) w wyniku czego powstaje wiązanie pomiędzy atomami siarki. W białkach tworzy się ono pomiędzy dwoma łańcuchami bocznymi reszt cysteiny. Powstały w ten sposób związek nazywamy cystyną[1, 2]. Innym przykładem jest wiązanie sulfiliminowe występujące pomiędzy metioniną a pochodną lizyny - hydroksylizyną. Występuje w nim podwójne wiązanie pomiędzy siarką a azotem[8].

1.1.4. Struktura drugorzędowa

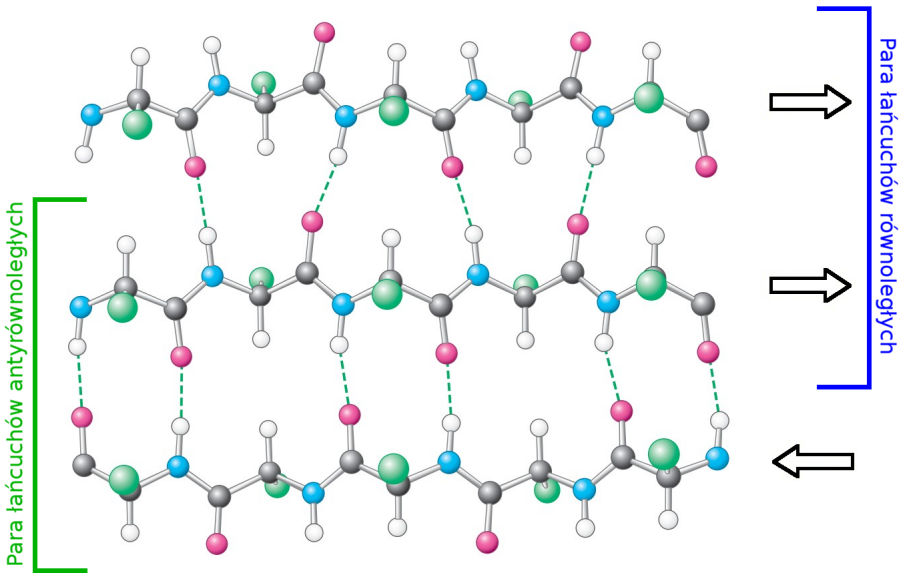
Struktura drugorzędowa białka określa strukturę przestrzenną aminokwasów znajdujących się blisko siebie w sekwencji łańcucha polipeptydowego. Jest ona stabilizowana głównie przez wiązania wodorowe powstające pomiędzy grupami peptydowymi i definiowana przez zestawy kątów φ i ψ występujące w danym fragmencie białka. Najpowszechniejszymi strukturami drugorzędowymi są α helisa i β kartka. Oprócz nich występują także pętle, zwroty (ang. *turns*) i inne struktury[1].



Rysunek 1.6. Schematycznie przedstawiona struktura α helisy. Po lewej rzut z boku struktury, po prawej z góry. Liniami przerywanymi zaznaczone są wiązania wodorowe stabilizujące strukturę. Źródło:[1].

Helisa α jest ciasno upakowaną, cylindryczną, skręconą strukturą. Jej rdzeń tworzony jest przez łańcuch główny, a łańcuchy boczne wystają z niego na zewnątrz cząsteczki. Jest ona stabilizowana przez wiązania wodorowe tworzące się pomiędzy grupą α karbonylową jednej reszty aminokwasowej a fragmentem NH grupy peptydowej reszty aminokwasowej znajdującej się o cztery pozycje dalej w sekwencji polipeptydowej. Każda kolejna reszta aminokwasowa jest obrócona o kąt 100° względem poprzedniej (na pełny obrót formalnie przypada 3,6 reszt aminokwasowych) i przesunięta o 0.15 nm wzdłuż osi struktury. α helisa może być zarówno prawo jak i lewoskrętna. Ze względu na zawady steryczne struktura prawoskrętna jest energetycznie uprzywilejowana i ogromna większość α helis w białkach występuje w tej formie[1]. Odpowiadające jej obszary na wykresie Ramachandrana pokazane są na rysunku 1.4. Jej schematyczny model przedstawiam na rysunku 1.6.

Struktura β, nazywana zazwyczaj β kartką albo β harmonijką składa się z co najmniej dwóch odcinków łańcucha polipeptydowego nazywanych wstążkami lub nićmi β. Pojedyncza nić β jest rozciągniętą strukturą w której łańcuchy boczne sąsiednich reszt



Rysunek 1.7. Schematycznie przedstawiona struktura β kartki. Strzałki po prawej stronie wskazują kierunek w jakim biegnie łańcuch polipeptydowy. Liniami przerywanymi zaznaczone są wiązania wodorowe stabilizujące strukturę. Źródło:[1].

aminokwasowych są zwrócone w przeciwnych kierunkach. Odległość pomiędzy sąsiednimi aminokwasami wynosi 0.35 nm. Dwie nici β oddziałują ze sobą poprzez wiązania wodorowe pomiędzy grupami α karbonylowymi i fragmentami NH grup peptydowych należących do różnych nici. Fragmenty łańcucha polipeptydowego tworzące poszczególne nici β mogą względem siebie biec w tym samym kierunku (równoległa β kartka) albo w przeciwnych kierunkach (antyrównoległa β kartka). β kartkę może tworzyć różna liczba nici. Najczęściej jest ich 4 albo 5, ale może być ich nawet kilkanaście. Struktura ta jako całość może być zarówno płaska jak i skręcona[1]. Odpowiadający jej obszar na wykresie Ramachandrana pokazany jest na rysunku 1.4. Schematyczny model tej struktury przedstawiam na rysunku 1.7.

Zwroty (ang. *turns*) tworzone są przez dwie do sześciu reszt aminokwasowych. Dzieli się je na klasy, oznaczane greckimi literami, w zależności od liczby reszt aminokwasowych je tworzących. Najczęściej występują zwroty β składające się z czterech reszt aminokwasowych. Pierwsza połączona jest z czwartą przez wiązanie wodorowe, co skutkuje ostrym zwrotem o 180° przestrzennego przebiegu łańcucha polipeptydowego. Poszczególne klasy zwrotów dzielimy na podtypy w zależności od zestawów wartości kątów torsyjnych występujących w ich łańcuchu głównym. Cechą charakterystyczną zwrotów jest ich regularność. Poszczególne klasy są dobrze zdefiniowane pod względem

struktury i tworzonych wiązań wodorowych. Pętle składają się z sześciu lub więcej reszt aminokwasowych, więcej niż jest to potrzebne do połączenia fragmentów innych struktur drugorzędowych. Od zwrotów różnią się przede wszystkim tym, że przyjmują nieregularne, ale specyficzne konformacje stabilizowane przez oddziaływania z innymi regionami białka. Często pełnią ważne funkcje biologiczne, uczestnicząc w wiązaniu substratu i katalizie. Obydwa te typy struktur powodują zmianę kierunku w którym łańcuch polipeptydowy biegnie w przestrzeni, a zwroty dodatkowo stabilizują nagłe zmiany tego kierunku. Często znajdują się one pomiędzy dwoma fragmentami łańcucha polipeptydowego tworzącymi struktury α albo β. Przykładem może być motyw strukturalny nić β - zwrot β - nić β, nazywana strukturą spinki do włosów. Obydwa typy struktur występują najczęściej na powierzchni zwiniętego białka i często biorą udział w jego oddziaływaniach z innymi cząsteczkami[1, 2, 9].

Opisane powyżej struktury są najczęściej występującymi strukturami drugorzędowymi białek. Poza nimi występują też inne, takie jak:

— helisa 3_{10} - prawoskrętna helisa w której kolejne reszty aminokwasowe obrócone są o 120° względem poprzednich i przesunięte o 0.20 nm wzdłuż osi struktury. Wiązania wodorowe tworzą się pomiędzy grupą α karbonylową jednej reszty aminokwasowej a fragmentem NH grupy peptydowej reszty aminokwasowej znajdującej się o trzy pozycje dalej w sekwencji polipeptydowej[10].

— helisa π - prawoskrętna helisa w której kolejne reszty aminokwasowe obrócone są o 83° względem poprzednich i przesunięte o 0.115 nm wzdłuż osi struktury. Wiązania wodorowe tworzą się pomiędzy grupą α karbonylową jednej reszty aminokwasowej a fragmentem NH grupy peptydowej reszty aminokwasowej znajdującej się o pięć pozycji dalej w sekwencji polipeptydowej[11].

— helisy poliprolinowa - helisy składające się z reszt proliny. Nie posiadają wewnętrznych wiązań wodorowych. W helisie prawoskrętnej kolejne reszty obrócone są o około 110° względem poprzednich i przesunięte wzdłuż osi struktury o 0.19 nm. W lewoskrętnej kolejne reszty obrócone są o 120° i przesunięte o 0.31 nm[12].

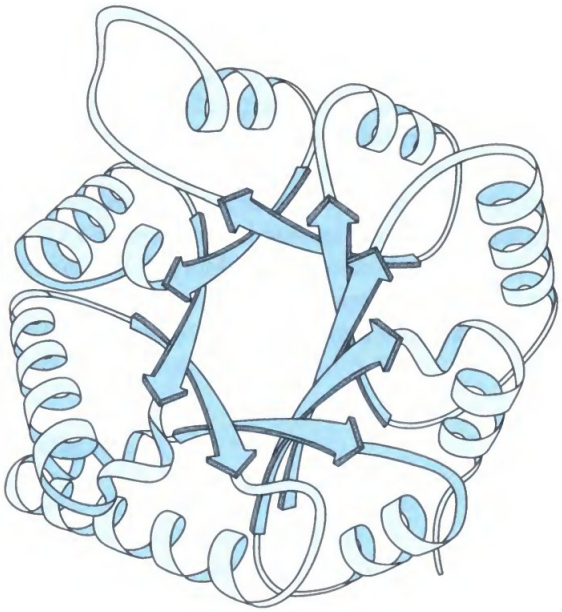
1.1.5. Struktura trzecio i czwartorzędowa

Strukturą trzeciorzędową białka nazywamy specyficzne ułożenie w przestrzeni całego pofałdowanego łańcucha polipeptydowego. Stabilizowana jest ona przez oddziaływania wewnątrzcząsteczkowe reszt aminokwasowych. Reszty aminokwasowe o hydrofobowych łańcuchach bocznych mają tendencję do występowania we wnętrzu białka, natomiast polarne występują w większości na jego powierzchni. Od tej reguły są wyjątki. Na przykład białka znajdujące się w błonach biologicznych posiadają hydrofobowe łańcuchy boczne zlokalizowane na powierzchniach znajdujących się na granicy białko-błona. Reszty hydrofilowe znajdujące się wewnątrz białek pełnią różnorodne funkcje, na przykład stabilizujące, enzymatyczne i kompleksujące jony metali[1, 2].

W łańcuchu głównym grupy aminowe i karbonylowe które nie tworzą wiązań wodorowych są hydrofilowe i występują zazwyczaj na powierzchni cząsteczek. α helisy i β kartki mają te grupy precyzyjnie sparowane i często posiadają charakter amfipatyczny. Oznacza to, że łańcuchy boczne po jednej stronie struktury są hydrofobowe i skierowane do wnętrza białka, a po drugiej stronie charakter hydrofilowy i są skierowane na zewnątrz, w kierunku środowiska[1]. Głównymi czynnikami stabilizującymi strukturę przestrzenną są wiązania wodorowe, mostki solne (oddziaływania pomiędzy przeciwnie naładowanymi łańcuchami bocznymi) i opisane przeze mnie już wcześniej wiązania disulfidowe[2]. Przykładowy model pokazujący strukturę trzeciorzędową białka przedstawiam na rysunku 1.8.

Pojedynczy, zwinięty łańcuch polipeptydowy może pod względem strukturalnym składać się z kilku domen. Domena jest rejonem łańcucha, który może zwijać się i istnieć niezależnie od reszty białka. Pojedyncza domena pełni określoną funkcję chemiczną lub fizyczną, jak wiązanie substratu lub kotwiczenie w błonach komórkowych. Domeny składają się z od około 30 do ponad 400 reszt aminokwasowych. Domeny na jednym łańcuchu polipeptydowym połączone są jego krótkimi, elastycznymi odcinkami. W skład jednego łańcucha polipeptydowego mogą wchodzić zarówno takie same jak i różne domeny[1, 2].

Białka mogą składać się z jednego albo kilku osobnych łańcuchów polipeptydowych nazywanych podjednostkami albo protomerami. Strukturą czwartorzędową nazywamy



Rysunek 1.8. Model wstęgowy struktury trzeciorzędowej izomerazy fosfotrioz. Źródło:[2].

wzajemne ułożenie przestrzenne tych podjednostek i sposób w jaki oddziałują one między sobą. Białka mogą się składać zarówno z takich samych jak i różnych podjednostek. O białku składającym się z dwóch podjednostek mówimy, że jest dimerem, trzech - trimerem itd. Jeśli podjednostki białka są identyczne mówimy o nim, że jest homodimerem, homotrimerem, itd. W przeciwnym wypadku, jeśli podjednostki są różne, mówimy o heterodimerze, heterotrimerze itd[1, 2].

1.2. Proces zwijania białek

Łańcuch polipeptydowy tworzący białko pełni określoną funkcję biologiczną tylko wtedy, kiedy jest prawidłowo zwinięty (pofałdowany) w określoną konformację struktury trzecio- lub czwartorzędowej. Konformację taką nazywamy strukturą natywną. Jest ona zazwyczaj najbardziej uprzywilejowana energetycznie spośród wszystkich konformacji krajobrazu energetycznego białka. Jeśli łańcuch polipeptydowy jest rozwinięty albo przypadkowo zwinięty w tzw. kłębek statystyczny (na przykład na skutek denaturacji przez temperaturę lub związki chemiczne) nie jest w stanie pełnić swojej funkcji. Niewłaściwie zwinięte białka mogą nawet w pewnych okolicznościach być toksyczne. Przykładowo choroby Alzheimera, Huntingtona i Parkinsona powodowane są tworze-

niem agregatów białkowych powstających w wyniku nieprawidłowego zwijania białek[1, 2].

Proces w którym białko przyjmuje określoną konformację nazywamy fałdowaniem albo zwijaniem się białka. W komórkach zachodzi on zazwyczaj w czasie do 1 sekundy. Pojedyncza makromolekuła może w trakcie swojego istnienia w komórce wielokrotnie się zwijać i (zazwyczaj częściowo) rozwijać. Proces ten nie jest obecnie w pełni poznany i zrozumiany. Wiadomo, że nie może zachodzić całkowicie przypadkowo, gdyż nawet w przypadku prostych białek przeszukanie wszystkich możliwych konformacji w poszukiwaniu natywnej zajęłoby czas znacząco przekraczający wiek Wszechświata. Tę różnicę pomiędzy obliczonym a faktycznym czasem potrzebnym do zwinięcia białka nazywamy paradoksem Levinthala. Pokazuje on, że na opisywany proces mają wpływ oddziaływania ukierunkowujące go i ograniczające liczbę testowanych konformacji[13].

Hipoteza termodynamiczna Anfinsena mówi, że trójwymiarowa struktura natywna białka w jego normalnym fizjologicznym środowisku jest strukturą o najniższej entalpii swobodnej. Oznacza to, że struktura natywna zależy od sumy oddziaływań pomiędzy atomami białka, a przez to od jego sekwencji aminokwasowej[14]. Poszczególne rodzaje aminokwasów różnią się zdolnością do tworzenia poszczególnych struktur drugorzędowych. Alanina, kwas glutaminowy i leucyna mają tendencję do tworzenia α helis, walina i izoleucyna do β kartek, glicyna, asparagina i prolina do zwrotów. Podane aminokwasy mogą występować w pozostałych strukturach drugorzędowych, ale pojawiają się w nich rzadziej. Te tendencje wynikają zarówno z zawad przestrzennych jak i właściwości chemicznych ich łańcuchów bocznych. Pozwala to obecnie przewidywać strukturę drugorzędową fragmentu łańcucha polipeptydowego o danej sekwencji z dość dużą dokładnością. Fakt, że nie jesteśmy w stanie przewidzieć tej struktury z całkowitą pewnością wskazuje że w jej definiowaniu i stabilizacji biorą udział nie tylko bezpośrednie oddziaływania pomiędzy sąsiednimi resztami aminokwasowymi w sekwencji łańcucha polipeptydowego, ale także oddziaływania pomiędzy bardziej odległymi od siebie fragmentami cząsteczki[1, 2].

Wiele białek jest w stanie zwinąć się prawidłowo ze stanu zdenaturowanego w roztworze laboratoryjnym (*in vitro*), ale proces ten zachodzi wolniej i z mniejszą wydaj-

nością niż w środowisku wnętrza komórki (*in vivo*). Inne białka nie są w stanie w ogóle uzyskać struktury natywnej *in vitro* tworząc częściowo zwinięte struktury, nierozpuszczalne agregaty czy przypadkowe kompleksy. Powodem tego jest istnienie w komórce białek wspomagających zwijanie. Białka opiekuńcze uczestniczą u ssaków w zwijaniu ponad połowy białek. Zapobiegają one agregacji łańcuchów polipeptydowych przez osłanianie hydrofobowych fragmentów formującej się struktury i stabilizują częściowo zwinięte białka. Należą do nich m.in. białka szoku cieplnego (HSP - heat shock proteins). Innym białkiem wspomagającym fałdowanie jest izomeraza dwusiarczkowa katalitycznie przyspieszająca rozrywanie i tworzenie nowych mostków disulfidowych[1, 2].

1.3. Modele procesu zwijania białek

1.3.1. Model dwustanowy

Dwustanowy model kinetyki zwijania białek zakłada, że proces ten można opisać odwołując się tylko do dwóch stanów konformacji białka. Pierwszym z nich jest prawidłowo zwinięte białko, czyli struktura natywna (F). Drugim jest stan rozwinięty (U), który jest rozumiany nie jako pojedyncza konformacja, ale jako zbiór wszystkich konformacji różnych od natywnej. Rozwinięte białka znajdują się w równowadze dynamicznej obejmującej wiele rozwiniętych struktur reprezentujących lokalne minima energetyczne tego stanu. Przebywają w nim dopóki nie zostaną całkowicie zwinięte. Model ten można przedstawić poniższym, prostym schematem:

$$F \rightleftharpoons U$$

Zastosowanie tego modelu dobrze opisuje kinetykę białek w których istnieje jeden prawidłowy, sfałdowany stan, a niesfałdowanych stanów jest wiele, żaden nie jest uprzywilejowany i białko w trakcie fałdowania może swobodnie pomiędzy nimi przechodzić[15].

1.3.2. Model trzystanowy

Kolejnym modelem kinetyki zwijania białka jest model trzystanowy. Poza stanami zwiniętym (F) i rozwiniętym (U), takimi jak w modelu dwustanowym, występuje też stan pośredni (I). Stan pośredni to konformacja lub zbiór konformacji przez które białko przechodzi w trakcie zwijania i rozwijania się. Konformacje te reprezentują lokalne minima energetyczne molekuły. Cząsteczki w tym stanie zazwyczaj są kompaktowe i zwinięte w struktury drugorzędowe ale nie mają poprawnej struktury trzeciorzędowej. Przejścia $U \rightleftharpoons I$ są zazwyczaj dużo szybsze niż $F \rightleftharpoons I$. Model ten można przedstawić następującym schematem:

$$F \rightleftharpoons I \rightleftharpoons U$$

Zastosowanie tego modelu dobrze opisuje kinetykę białek w których istnieje wyraźny, częściowo sfałdowany i stabilny, bogaty w struktury drugorzędowe stan pośredni pomiędzy białkiem całkowicie zwiniętym a rozwiniętym[16, 17].

1.3.3. Model kondensacji wokół zarodków strukturalnych

Model kondensacji wokół zarodków strukturalnych jest kolejnym modelem zwijania białek. Zakłada on, że proces ten zachodzi poprzez stabilizację form pośrednich. Rozwinięte białko ma określoną energię swobodną, która zmniejsza się w trakcie zwijania. Zachodzi to na skutek tworzenia się wiązań wodorowych, agregacji hydrofobowych łańcuchów bocznych wewnątrz cząsteczki, tworzenia mostków disulfidowych itd. Fragmenty cząsteczki tworzą najpierw struktury drugorzędowe, co powoduje lokalne uporządkowanie struktury. Następnie te struktury drugorzędowe zaczynają się układać względem siebie tworząc formę pośrednią nazywaną fazą roztopionej kuli. Proces ten zachodzi głównie dzięki oddziaływaniom hydrofobowym i nie jest sztywno ustalony (może prowadzić różnymi drogami przez rożne stadia pośrednie). Z tej formy białko zwija się ostatecznie do struktury natywnej[13].

1.4. Rola biologiczna

Białka spełniają wiele funkcji w organizmach żywych. Do najważniejszych z nich zaliczamy:

— Enzymatyczna. Katalizują reakcje chemiczne dostarczające energii, syntetyzujące i degradujące cząsteczki i makromolekuły biologiczne.
— Strukturalna. Utrzymują kształt komórek oraz połączenia pomiędzy nimi.
— Mechaniczna. Tworzą aparat kurczliwy mięśni.
— Transportowa. Rozprowadzają cząsteczki po organizmie, na przykład tlen przez hemoglobinę.
— Obronna. Jako przeciwciała uczestniczą w zwalczaniu patogenów.
— Receptorowa. Reagują na czynniki środowiskowe[2].

1.5. Determinowanie struktur białek

Białka mogą spełniać swoją funkcję biologiczną tylko kiedy są prawidłowo sfałdowane. Sprawia to, że poznawanie ich struktur jest istotnym zagadnieniem w biochemii i biologii molekularnej. Uzyskane modele struktur makromolekuł przechowywane są w bazach danych, takich jak Protein Data Bank (PDB)[18].

1.5.1. Krystalografia rentgenowska

Krystalografia rentgenowska jest najstarszą instrumentalną metodą wykorzystywaną do poznawania struktur przestrzennych białek i jest powszechnie wykorzystywana do dziś. Uznawana jest za najlepszą metodę wyznaczania struktury białka ze względu na możliwość uzyskania wysokiej rozdzielczości otrzymywanych struktur[1]. Pierwszą strukturą przestrzenną białka uzyskaną za jej pomocą pod koniec lat 50 XX wieku była struktura mioglobiny[19].

Wymaga ona uzyskania wysokiej jakości kryształu danego białka. Otrzymuje się go powoli wysalając je z roztworu w odpowiednich warunkach, dobranych eksperymentalnie do danego białka. Białka różnią się zdolnością do tworzenia kryształów. Niektóre tworzą je łatwo, inne wymagają bardzo specyficznych warunków. Proces ten powoduje

pewne zmiany w strukturze przestrzennej makromolekuły. Mimo to kryształy często zachowują aktywność biologiczną i znajdujące się w nich białka są podobne do struktury natywnej[20].

Uzyskany kryształ, schłodzony w ciekłym azocie i wprawiony w ruch obrotowy, jest oświetlany wiązką promieniowania rentgenowskiego. Daje ono możliwość uzyskania rozdzielczości wystarczającej do ustalenia położenia większości atomów, ponieważ długość jego fali jest porównywalna z długością wiązania kowalencyjnego. Uzyskuje się je w synchrotronach albo bombardując miedzianą anodę strumieniem elektronów. Część promieniowania ulega w krysztale dyfrakcji, rozpraszając się na różne strony, co można zarejestrować stosując odpowiedni detektor. Dyfrakcję tę powodują elektrony atomów znajdujących się w krysztale. Im więcej elektronów posiada dany atom, tym większa jest amplituda powstałej fali. Poszczególne fale interferują ze sobą wzmacniając się lub wygaszając, a sposób w jaki to robią zależy od układu atomów w krysztale[21].

W ten sposób na detektorze uzyskuje się wzór dyfrakcyjny w postaci punktów (plamek) i przy pomocy przekształcenia Fouriera tworzy się z niego obraz białka. Na podstawie tego obrazu i dodatkowych danych tworzy się mapę gęstości elektronowej białka, mierząc ją w regularnie rozmieszczonych punktach wewnątrz kryształu. W zależności od rozdzielczości uzyskanej mapy można na jej podstawie otrzymać mniej lub bardziej dokładną strukturę przestrzenną białka. Wadą tej metody jest to, że w modelach uzyskanych za jej pomocą nie są widoczne ruchliwe pętle znajdujące się w niektórych białkach[21].

1.5.2. Spektroskopia magnetycznego rezonansu jądrowego

Spektroskopia magnetycznego rezonansu jądrowego (NMR spectroscopy) jest techniką pozwalającą na poznanie struktury przestrzennej białka w jego stężonym roztworze wodnym. Wykorzystuje ona kwantową własność niektórych jąder atomowych nazwaną spinem. Jednym z takich jąder jest proton, czyli jądro atomu wodoru. Pod wpływem pola magnetycznego jego spin może przybrać jeden z dwóch stanów, pomiędzy którymi istnieje niewielka różnica w energii. Fale radiowe o odpowiedniej częstotliwości, zależnej od różnicy w energii pomiędzy tymi dwoma stanami, mogą spowodować przejście

spinu w stan o wyższej energii (pobudzony), co oznacza uzyskanie tzw. rezonansu. Tę częstotliwość można zmierzyć[22].

Obecność innych atomów w pobliżu powoduje niewielkie zmiany częstotliwości rezonansu danego atomu. Mierzone są one w częściach na milion (ppm) względem wzorca i nazywane przesunięciami chemicznymi. Widmo rezonansu całej cząsteczki uzyskuje się zmieniając długość fal radiowych przy zachowaniu stałego pola magnetycznego. Można w ten sposób uzyskać informacje o większości protonów w białku. Powstaje w ten sposób wykres zależności intensywności rezonansu od przesunięcia chemicznego. Tę technikę określa się jako jednowymiarowy NMR[22].

Bardziej zaawansowane techniki pozwalają na zmianę spinu pojedynczego jądra atomowego i analizę jego wpływu na spiny jąder atomowych sąsiadujących w przestrzeni poprzez tzw. efekt Overhausera. Analiza dostatecznie dużej liczby atomów pozwala na utworzenie dwuwymiarowego wykresu pokazującego które jądra mogą wpływać na inne i określenie struktury przestrzennej białka. Podobnie jak w krystalografii otrzymana struktura nie jest doskonała, ze względu na istnienie w roztworze nieznacznych różnic pomiędzy konformacjami białka pozostającymi w równowadze, a wyznaczone odległości pomiędzy atomami są tylko przybliżone. Techniki te określa się jako dwuwymiarowy NMR[22].

Technikę tę zaczęto stosować do poznawania struktur białek na początku lat 80. XX wieku. Pierwszą strukturą poznaną za jej pomocą była struktura przyłączonego do miceli glukagonu, który jest polipeptydowym hormonem[23]. Największa wadą tej metody jest to, że uzyskane modele są zazwyczaj gorszej jakości niż uzyskane przy pomocy krystalografii rentgenowskiej[1].

1.5.3. Kriomikroskopia elektronowa 3D

Kriomikroskopia elektronowa 3D jest jedną z nowszych eksperymentalnych technik poznawania struktur białek i innych makromolekuł. Zyskuje ona szybko na popularności i w 2017 roku do bazy PDB dodano więcej struktur uzyskanych za jej pomocą niż struktur uzyskanych przy pomocy opisanej powyżej metody NMR[24].

W tej technice cienka warstwa roztworu zawierającego badaną makromolekułę (białko

lub kompleks białek) jest błyskawicznie zamrażana na filmie węglowym do temperatury ciekłego azotu, wodoru lub helu. W ten sposób cząsteczki zostają zamrożone w amorficznym lodzie. W przeciwieństwie do lodu posiadającego strukturę krystaliczną nie odkształca on struktury badanych molekuł. W ten sposób uzyskuje się próbkę która powinna zawierać identyczne cząsteczki w różnych orientacjach przestrzennych[25, 26].

Następnie przy pomocy mikroskopu elektronowego uzyskiwane jest wiele obrazów badanego związku w różnych konformacjach przestrzennych. Silny strumień elektronów mikroskopu działa niszcząco na badany materiał, zrywając wiązania i tworząc wolne rodniki. Zamrożenie cząsteczek zwiększa ich odporność na to promieniowanie i pozwala na uzyskiwanie obrazów przy pomocy strumienia o wyższej energii, co zwiększa ich rozdzielczość. Podobnie uzyskanie wielu obrazów pozwala uzyskać pewniejsze wyniki[25, 26].

Z uzyskanych dwuwymiarowych obrazów tworzony jest trójwymiarowy model badanej struktury. Wykorzystuje się przy tym transformację Fouriera i fakt, że każda transformacja Fouriera dwuwymiarowych projekcji trójwymiarowego obiektu jest wycinkiem trójwymiarowej transformacji Fouriera tego obiektu. Wadą tej metody jest to, że ze względu na uśrednianie informacji z wielu obrazów nie są w niej widoczne ruchliwe pętle znajdujące się w niektórych białkach[25, 26].

Jedną z pierwszych struktur o wysokiej rozdzielczości uzyskanych tą metodą była bakteriorodopsyna. Uzyskano ją w 1990 roku z 72 obrazów co pozwoliło uzyskać rozdzielczość struktury do 3,5Å, w zależności od kierunku[27].

1.5.4. Metody obliczeniowe

Dzięki rosnącej mocy obliczeniowej komputerów, także rosnącej ilości dostępnych danych biomolekularnych oraz doskonaleniu modeli cząsteczek próby obliczeniowego wyznaczania struktur białek dają coraz lepsze wyniki[28].

W podejściu *ab initio* strukturę białka próbuje się uzyskać bezpośrednio z jego sekwencji aminokwasowej, poprzez przeszukiwanie przestrzeni konformacyjnej. Jedną z technik jest "build-up" w którym peptyd budowany jest przez cykliczne przyłączanie

kolejnych reszt aminokwasowych i minimalizację energii powstałej struktury. Daje ona dobre wyniki dla krótkich peptydów, ale nie radzi sobie z dłuższymi.

W metodach Monte Carlo strukturę białka próbuje się uzyskać poprzez iteracyjne wprowadzanie losowych zmian wartości poszczególnych parametrów (długości wiązań, kątów walencyjnych i torsyjnych) określonej struktury startowej. Następnie oblicza się energię zmienionej struktury i albo odrzuca się ją powracając do struktury startowej, albo zastępuje się startową strukturę nowo uzyskaną. Pozwala ona na eksplorację krajobrazu energetycznego molekuły, ale nie daje gwarancji osiągnięcia struktury natywnej[28].

Kolejną metodą jest budowanie modelu białka z fragmentów o znanej strukturze. Wykorzystuje ona założenie, że lokalne interakcje w znaczącym stopniu definiują strukturę makromolekuły. Dzieli ona sekwencję badanego białka na fragmenty, które są wyszukiwane w bazie znanych modeli. Są one ze sobą łączone aby uzyskać model całego białka. Jest on następnie optymalizowany i oceniany przy użyciu odpowiednich funkcji. Trudnością tej metody jest wybór właściwej funkcji oceniającej oraz ilość i jakość danych w bazie fragmentów[29].

W modelowaniu homologicznym (nazywanym też modelowaniem porównawczym) znana struktura białka jest wykorzystywana do zbudowania modelu innego białka o podobnej sekwencji. Najpierw przeszukuje się dostępne bazy danych w celu znalezienia białek - wzorców o znanych strukturach i sekwencjach podobnych do badanej. Prawidłowe porównanie sekwencji jest istotnym zagadnieniem, ze względu na możliwość nie tylko zamiany reszty aminokwasowej, ale także insercji i delecji. Obecnie najczęściej wykorzystuje się do tego algorytmy BLAST i FASTA. W uproszczeniu uznaje się, że 25% lub większe podobieństwo sekwencji wskazuje na homologię. Następnie na podstawie znalezionych wzorców konstruuje się i ocenia grupę struktur, które prawdopodobnie mogłaby przyjmować badana sekwencja. Wadą tej metody jest to, że nie zawsze dostępna jest struktura białka o podobieństwie wystarczającym do uzyskania wysokiej jakości modelu[13, 28].

Kolejną techniką jest mechanika molekularna, opisana szerzej w dalszej części niniej-

szej pracy, która może służyć do badania procesu zwijania się i zmian konformacyjnych białek[28].

2. Modelowanie molekularne

2.1. Wstęp

Modele różnych zjawisk i obiektów ułatwiają ich zrozumienie i są użytecznym narzędziem w pracy naukowej. Dla związków chemicznych przyjmują one formy od prostych, plastikowych fizycznych modeli, przez komputerowe, graficzne reprezentacje cząsteczek (tzw. grafika molekularna) po zaawansowane modele matematyczne. Do tych ostatnich zaliczamy między innymi mechanikę molekularną[30].

Mechanika molekularna zajmuje się uproszczonymi modelami cząsteczek typu „kule i sprężyny". Kule reprezentują w nich poszczególne atomy, a sprężyny, które mogą się kurczyć i rozciągać, wiązania pomiędzy nimi. Rozmiar kul-atomów i sztywność sprężyn-wiązań są determinowane empirycznie tak, aby właściwości modelu miały przełożenie na rzeczywiste układy molekularne. W ramach tak uproszczonych modeli nie jest możliwe na przykład badanie procesów w których rozrywane są lub powstają wiązania chemiczne. W zamian pozwalają one na symulowanie zachowania większych układów w dłuższym czasie. Mimo swoich ograniczeń metody mechaniki molekularnej są dzisiaj powszechnie używane do badania zachowania wielu układów biomolekularnych[28, 31].

Inną klasą matematycznych modeli molekuł są modele kwantowomechaniczne. Pozwalają one na modelowanie zachowania chmur elektronowych poszczególnych atomów. Dzięki temu możliwe jest symulowanie za ich pomocą procesów, w których następuje tworzenie i rozrywanie wiązań chemicznych. Ze względu na znacznie większe ich skomplikowanie symuluje się za ich pomocą mniejsze układy w krótszym czasie[28, 32]. W niniejszej pracy wykorzystuję modele bazujące na mechanice molekularnej i na nich skupia się dalsza część tego rozdziału.

2.2. Opis struktury cząsteczki

W celu utworzenia modelu określonej cząsteczki muszą być znane pozycje jej atomów w przestrzeni. Najczęściej są one zapisywane w formie współrzędnych kartezjańskich, czyli zestawu trzech wartości określających odległości danego atomu od środka układu współrzędnych w trzech wymiarach. Alternatywnym sposobem są współrzędne wewnętrzne, które określają pozycje atomów na podstawie odległości oraz wartości kątów walencyjnych i torsyjnych pomiędzy nimi[30].

Same atomy mogą być opisywane na dwa różne sposoby. W modelach pełnoatomowych każdy atom traktowany jest jako osobny obiekt. W modelach uproszczonych, tzw. gruboziarnistych, pojedyncze obiekty odpowiadają grupom atomów[33].

2.3. Pole sił

Pole sił, albo empiryczna funkcja energii to konkretny opis modelu oddziaływań w mechanice molekularnej. Składają się na nie następujące elementy:

— Matematyczny wzór określający energię potencjalną układu.
— Baza standardowych elementów budulcowych cząsteczek, tak zwanych reszt (aminokwasy, nukleotydy, cukry proste, woda itp.), zawierająca informacje na temat ich struktury. Pozwala to na łatwe operowanie większymi układami zawierającymi te standardowe elementy budulcowe.
— Zestaw parametrów zoptymalizowany dla danej klasy cząsteczek, pozwalający na obliczenie właściwości układu. Parametrami tymi są przykładowo właściwości poszczególnych rodzajów atomów i wiązań pomiędzy nimi. Są one ustalane eksperymentalnie albo na podstawie obliczeń kwantowomechanicznych[32].

2.3.1. Energia potencjalna układu

W empirycznych polach siłowych funkcja energii potencjalnej układu stanowi sumę składników równania reprezentujących różne oddziaływania i efekty:

$$E_{total} = E_{bonds} + E_{angles} + E_{dihedrals} + E_{VDV} + E_{elec}$$

Różnią się one pomiędzy poszczególnymi polami sił[28]. Poniżej opisano najczęściej występujące składniki razem z ich przykładowymi wzorami matematycznymi.

2.3.2. Oddziaływania wiążące

Oddziaływania wiążące (ang. *bonded interactions*) zachodzą pomiędzy atomami połączonymi wiązaniami kowalencyjnymi. Mogą być to atomy bezpośrednio połączone ze sobą takim wiązaniem, albo takie pomiędzy którymi istnieje ścieżka zawierająca określoną liczbę wiązań (zazwyczaj 2-3 wiązania). Do najczęściej modelowanych oddziaływań wiążących należą:[32]

Rozciąganie i ściskanie wiązań kowalencyjnych

Zachodzi pomiędzy parami atomów połączonych bezpośrednio wiązaniem kowalencyjnym i oznacza energię związaną z długością wiązań. Jego wartość zależy od rodzaju wiązania i różnicy jego długości względem optymalnej. Najprostszym sposobem modelowania wiązania jest traktowanie go jako klasyczny oscylator harmoniczny. Metoda ta daje wyniki zgodne z rzeczywistością dla małych odchyleń od stanu równowagi. W przypadku dużych odchyleń wartość potencjału energii rośnie w nim zbyt szybko. Alternatywami dla niego są potencjały kwadratowe i sześcienne.[28]. W typowych polach sił pakietu AMBER jego energia przedstawia się wzorem:

$$E_{bonds} = \sum_{bonds} K_b(b - b_0)^2$$

Gdzie

K_b - stała sprężystości danego wiązania wynikająca z prawa Hooke'a

b - obecna długość wiązania

b_0 - długość wiązania w stanie równowagi[30, 34]

Zmiany kątów walencyjnych pomiędzy wiązaniami

Zachodzi w grupach trzech atomów połączonych liniowo wiązaniami kowalencyjnymi i oznacza energię związaną z kątem pomiędzy dwoma wiązaniami. Jego wartość zależy od rodzajów atomów i wiązań oraz różnicy wartości tego kąta względem warto-

ści optymalnej. To oddziaływanie również jest najczęściej modelowane jako oscylator harmoniczny[28]. W typowych polach sił pakietu AMBER jego energia przedstawia się wzorem:

$$E_{angles} = \sum_{angles} K_\theta(\theta - \theta_0)^2$$

Gdzie

K_θ - stała sprężystości deformacji danego kąta

θ - obecna wartość kąta

θ_0 - wartość kąta w stanie równowagi[30, 34]

2.3.3. Oddziaływania niewiążące

Oddziaływania niewiążące (ang. *non-bonded interactions*) formalnie mają nieograniczony zasięg i zachodzą pomiędzy wszystkimi atomami w układzie. W ich obliczaniu najczęściej pomija się pary atomów połączone bezpośrednio wiązaniem kowalencyjnym lub ścieżką dwóch wiązań ze względu na znacząca dominację energii oddziaływań wiążących w tych przypadkach. W symulacjach zazwyczaj ogranicza się także ich zasięg, najczęściej do 8-14Å. Powoduje to uproszczenie obliczeń bez znaczącej utraty dokładności, ze względu na szybki spadek znaczenia tych oddziaływań wraz z odległością. Wadą tego rozwiązania jest powstanie nieciągłości na wprowadzonej granicy oddziaływania. Istnieje kilka algorytmów ograniczania wpływu tej nieciągłości na symulację. Do najczęściej modelowanych oddziaływań niewiążących należą:[32]

Oddziaływania Van der Waalsa

Zachodzą pomiędzy wszystkimi parami atomów w układzie i wynikają z niesymetryczności rozmieszczenia ładunków elektrycznych wokół atomu. Ich wartość zależy od rodzaju atomów i odległości pomiędzy nimi. Najczęściej oblicza się je wykorzystując potencjał Lennard-Jones'a, który przyjmuje postać następującego równania:

$$E_{VDV} = \sum_{i,j} 4\epsilon_{ij}[(\frac{\sigma_{ij}}{r_{ij}})^{12} - (\frac{\sigma_{ij}}{r_{ij}})^{6}]$$

Gdzie

ϵ_{ij} - głębokość studni potencjału dla danej pary atomów

σ_{ij} - odległość pomiędzy danymi atomami w której potencjał Lennard-Jones'a ma wartość 0

r_{ij} - obecna odległość pomiędzy danymi atomami[1, 28]

Oddziaływania elektrostatyczne

Zachodzą pomiędzy parami atomów posiadającymi ładunek elektryczny. Ich wartość zależy od wielkości ładunków, odległości pomiędzy atomami oraz stałej dielektrycznej środowiska. Oblicza się je korzystając z prawa Coulomba, które przyjmuje postać następującego równania:

$$E_{elec} = \sum_{charged\ i,j} (\frac{q_i q_j}{4\pi\epsilon r_{ij}})$$

Gdzie

q_i, q_j - wartości ładunków elektrycznych atomów

ϵ - stała dielektryczna środowiska

r_{ij} - odległość pomiędzy danymi atomami[1, 28]

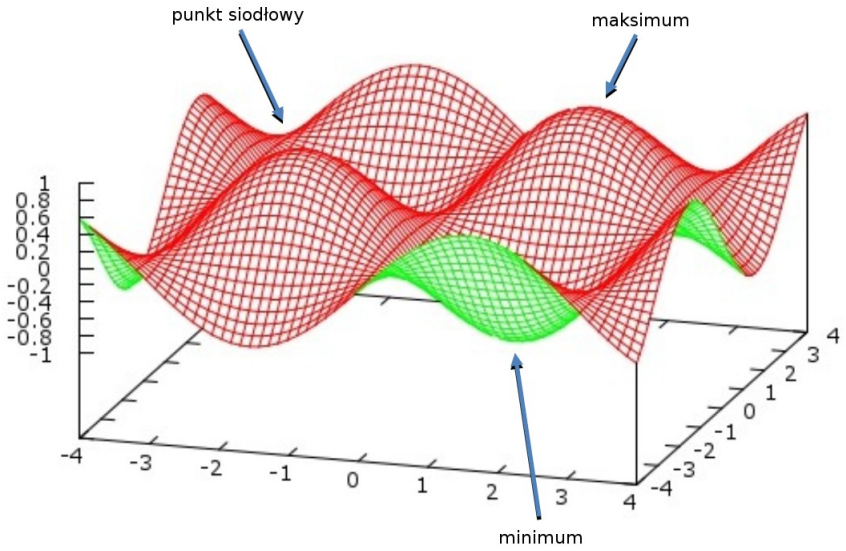
2.3.4. Potencjał torsyjny

Zachodzi w grupach czterech atomów połączonych liniowo wiązaniami kowalencyjnymi i oznacza energię związaną z kątem torsyjnym, czyli kątem obrotu wokół wiązania. Jego wartość zależy od wysokości bariery energetycznej i wartości tego kąta. Najczęściej modelowany jest z wykorzystaniem rozwinięcia szeregu cosinusowego[28, 30]. W typowych polach sił pakietu AMBER jego energia przedstawia się wzorem:

$$E_{dihedrals} = \sum_{dihedrals} (\frac{V_n}{2})(1 + \cos[n\phi - \delta])$$

Gdzie

V_n - wysokość bariery energetycznej



Rysunek 2.1. Schematyczne przedstawienie hiperpowierzchni energii układu. Zaznaczono maksimum, minimum i punkt siodłowy. Źródło:[32].

n - periodyczność kąta torsyjnego

ϕ - obecna wartość kąta torsyjnego

δ - kąt fazowy[30, 34]

2.4. Optymalizacja energii potencjalnej struktury

W ramach pola sił wartość energii potencjalnej układu może być przedstawiona w formie jej hiperpowierzchni o 3N wymiarach, nazywanej także krajobrazem energetycznym cząsteczki. N to liczba atomów w układzie, a każdy atom reprezentują 3 wymiary opisujące jego położenie w przestrzeni. Tylko niewielki zbiór wyróżnionych punktów tej hiperpowierzchni jest interesujący z biochemicznego punktu widzenia. Lokalne minima energii reprezentują stabilne stany układu. Minimum globalne odpowiada najbardziej stabilnemu stanowi, często będącym natywną konformacją w przypadku białek. Punkty siodłowe pomiędzy minimami odpowiadają stanom pośrednim układu[32]. Schematyczne przedstawienie wykresu krajobrazu energetycznego przedstawiam na rysunku 2.1.

Dostępne struktury białek zazwyczaj nie są idealnie zgodne z rzeczywistą strukturą natywną ze względu na ograniczenia metod ich pozyskiwania. Podobnie każde pole sił jest tylko pewnym przybliżeniem praw natury, co ma wpływ na energię poszczegól-

nych stanów układu. W związku z tym każda struktura, która ma być symulowana w danym polu sił powinna być najpierw dla niego zoptymalizowana. Proces ten polega na znalezieniu w krajobrazie energetycznym lokalnego minimum leżącego w pobliżu wejściowej konfiguracji układu. Powoduje to, że przyjmie on bardziej prawdopodobną i realistyczną konfigurację w kontekście danego pola sił. Dopiero tak przygotowany układ powinien brać udział w dalszych symulacjach[32]. Przykładowymi technikami minimalizacji energii układu są:

— Minimalizacja wzdłuż kolejnych osi współrzędnych
— Metoda Newtona poszukiwania minimum funkcji
— Metoda najszybszego spadku
— Metoda gradientów sprzężonych[28]

2.5. Dynamika molekularna

Dynamika molekularna (MD) pozwala na obliczanie ewolucji układu molekularnego w czasie i dzięki temu umożliwia poznanie jego właściwości dynamicznych i termodynamicznych. Polega na obliczaniu sił działających na poszczególne atomy układu i analizie ich ruchu. Typowa symulacja MD zaczyna się od powolnego podgrzania zoptymalizowanego układu do zadanej temperatury. Pozwala to na przyjęcie przez układ właściwych prędkości poszczególnych atomów i osiągnięcie stanu równowagi. Następnie przeprowadza się właściwą część symulacji MD, tzw. (ang.) *production run*, z którego to etapu zbierane są informacje do dalszej analizy. Początkowe prędkości atomów w symulacji są ustalane na podstawie rozkładu Maxwella dla danej temperatury. Proces ten jest losowy i jest to jedyna część symulacji MD która nie jest deterministyczna. Jeśli przeprowadza się kilka różnych symulacji tego samego układu należy za każdym razem wylosować prędkości na nowo[32].

Przeprowadzanie symulacji MD wymaga rozwiązywania równań ruchu dla poszczególnych atomów układu. W przypadku korzystania z kartezjańskiego układu współrzędnych sprowadzają się one do równania dynamiki Newtona $m_i \cdot r_i = F_i$, gdzie m_i jest masą danego atomu, r_i jego przyspieszeniem, a F_i siłą na niego działającą. W praktyce

równania te rozwiązuje się numerycznie z wykorzystaniem metody różnic skończonych. Polega ona na rozwiązywaniu równań po kolei dla określonych, niewielkich kroków czasowych Δt [28, 35].

Istotną rzeczą jest ustalenie właściwego kroku czasowego Δt. Jeśli będzie on zbyt krótki obliczenia dla danego czasu symulacji będą trwały dłużej. Jeśli będzie zbyt długi otrzymane wyniki będą niefizyczne, a układ może się nawet rozpaść. Uznaje się, że powinien być on około 10 razy krótszy niż okres wibracji o najwyższej częstotliwości w układzie[28].

Prędkości atomów w danym kroku czasowym Δt obliczane są na podstawie ich pozycji. Atomy są następnie przemieszczane o odległości obliczone na podstawie tych prędkości. Proces ten jest powtarzany dla każdego kolejnego kroku symulacji. Istnieją różne algorytmy przeprowadzania tych obliczeń. Należą do nich[32]:

Algorytm Verleta

$$v(t) = \frac{r(t + \Delta t) - r(t - \Delta t)}{2\Delta t}$$

Algorytm Skokowy Verleta

$$v(t) = \frac{v(t + \frac{1}{2}\Delta t) + v(t - \frac{1}{2}\Delta t)}{2}$$

Algorytm Prędkościowy Verleta

$$v(t + \Delta t) = v(t + \frac{1}{2}\Delta t) + \frac{1}{2}\Delta t a(t + \Delta t)$$

Gdzie

$v(t)$ - prędkość w danym czasie

$t(t)$ - położenie w danym czasie

Δt - krok czasowy

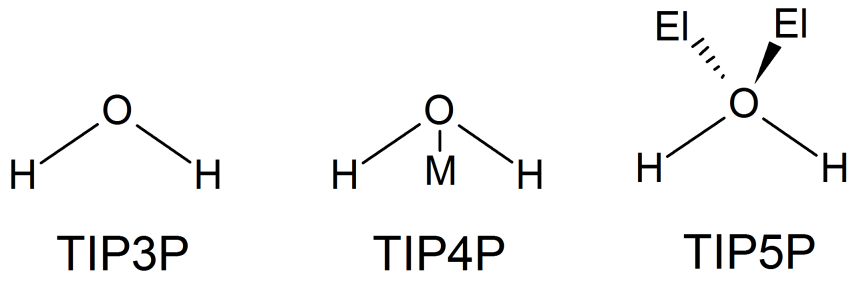
$a(t)$ - przyspieszenie w danym czasie

Poszczególne odmiany Algorytmu Verleta różnią się między sobą wielkością wprowadzanych błędów i złożonością obliczeniową. Z zaprezentowanych algorytmów najmniejsze błędy wprowadza Algorytm Skokowy Verleta[32].

2.6. Symulacja rozpuszczalnika (wody)

Cząsteczka wody jest najprawdopodobniej najczęściej symulowaną molekułą w mechanice molekularnej ze względu na jej rolę rozpuszczalnika makromolekuł biologicznych. Z tego powodu powstało wiele modeli tej cząsteczki stosowanych w symulacjach. Najczęściej mają one od 3 do 5 centrów oddziaływań. Często stosowanym modelem jest TIP3P składający się z 3 centrów oddziaływań, takich jak atomy w rzeczywistej cząsteczce. Kolejnym modelem jest TIP4P, posiadający dodatkowy wirtualny atom reprezentujący ładunek znajdujący się na atomie tlenu. Następnym jest TIP5P, w którym występują dwa dodatkowe, wirtualne atomy reprezentujące wolne pary elektronowe atomu tlenu[28, 36, 37]. Modele te schematycznie zaprezentowano na rysunku 2.2. Alternatywą dla symulowania poszczególnych cząsteczek wody jest traktowanie jej jako ciągłe środowisko, czyli tzw. model ciągły rozpuszczalnika. Przykładem może być uogólniony model Borna[28, 30].

Model ciągły jest najszybszy obliczeniowo, ale traci się w nim wpływ niektórych oddziaływań rozpuszczalnika na symulowaną molekułę. W modelach atomowych stopień komplikacji obliczeń rośnie wraz ze wzrostem liczby atomów. Ze względu na konieczność uniknięcia efektów brzegowych niezbędna jest obecność dużej liczby cząsteczek rozpuszczalnika, przez co jego skomplikowanie ma znaczący wpływ na tempo obliczeń[28, 30].

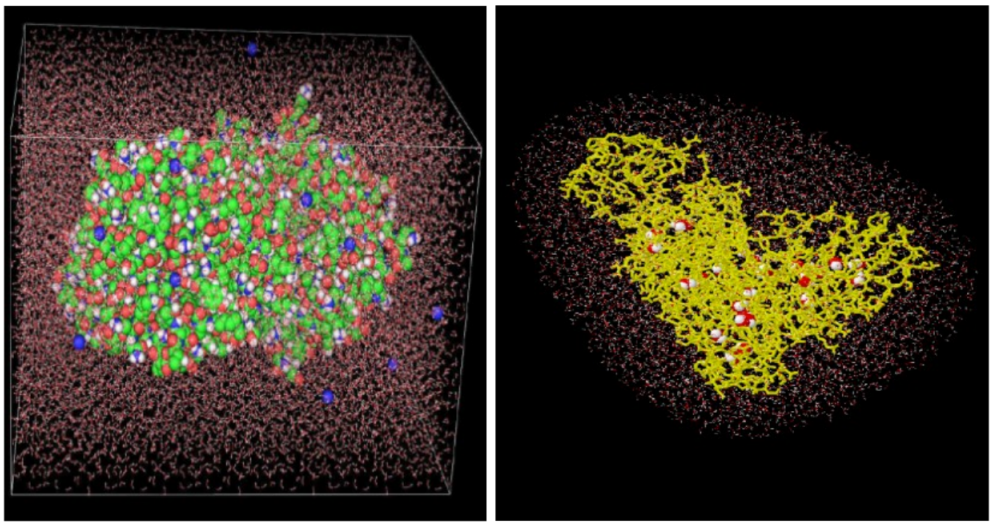


Rysunek 2.2. Schematyczne przedstawienie kilku modeli cząsteczki wody wykorzystywanych w mechanice molekularnej. **M** oznacza wirtualny atom na którym znajduje się ładunek atomu tlenu. **El** oznacza wirtualne atomy reprezentujące wolne pary elektronowe. Źródło: Własne na podstawie [28, 36, 37].

2.7. Algorytm SHAKE

Algorytm SHAKE służy do wymuszania ograniczeń zmian wartości pewnych parametrów podczas symulacji (tzw. (ang.) *constraints*). Wykorzystywany jest przede wszystkim do ograniczania ruchów atomów cząsteczki tak, aby długości wiązań nie przekraczały określonych wartości. Najczęściej stosowany jest dla wiązań w których jeden z atomów jest atomem wodoru, ale może być także stosowany dla innych wiązań. Istnieją także wersje tego algorytmu dla ograniczania kątów pomiędzy wiązaniami i kątów torsyjnych. Polega on na dynamicznym korygowaniu zmian długości poszczególnych wiązań poddanych działaniu algorytmu tak, aby zadane ograniczenia były spełnione. Zmiana długości jednego wiązania może spowodować, że inne będą miały nieprawidłowe długości. W związku z tym proces ten jest wykonywany iteracyjnie aż wszystkie nałożone na układ ograniczenia będą spełnione[28, 38, 39].

Algorytm SHAKE pozwala na przyspieszenie symulacji dynamiki molekularnej bez znaczącej utraty dokładności. Umożliwia zwiększenie kroku czasowego w symulacji, nawet kiedy jest stosowany tylko dla wiązań zawierających atom wodoru. Generalnie nie jest stosowany w trakcie minimalizacji, poza pewnymi szczególnymi przypadkami[28, 31].



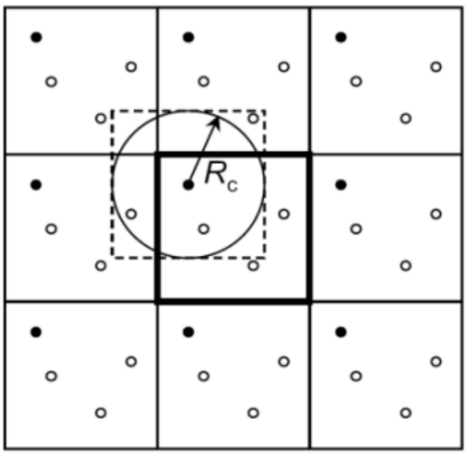
Rysunek 2.3. Symulacja rozpuszczalnika w przestrzeni periodycznej (pokazano jedną kopię) i jako kroplę. Źródło:[32].

2.8. Warunki brzegowe i periodyczność układu

Symulowanie makromolekuł biologicznych razem z rozpuszczalnikiem jest istotnym zagadnieniem w mechanice molekularnej. W przypadku braku rozpuszczalnika (symulacje "w próżni") albo użycia ciągłego modelu rozpuszczalnika symuluje się tylko samą makromolekułę, a problemy związane z warunkami brzegowymi nie występują. W przypadku układu symulującego poszczególne cząsteczki rozpuszczalnika sytuacja jest inna. Można go symulować na dwa sposoby: jako kroplę albo w przestrzeni periodycznej co pokazuje rysunek 2.3 [38].

W symulacji układu jako kropli występuje ograniczona liczba cząsteczek wody otaczająca makromolekułę (najczęściej sferycznie), a wielkość tej kropli wyznacza granice układu. Powoduje to wystąpienie artefaktów na tej granicy, które mogą wpływać na zachowanie symulowanej cząsteczki. Metoda ta ogranicza przemieszczanie się i zmiany kształtu makromolekuły do wnętrza kropli, co może spowodować znaczące problemy, jeśli makromolekuła znajdzie się bardzo blisko granicy układu[38].

W symulacji periodycznej (tzw. periodic box) makromolekuła razem z cząsteczkami rozpuszczalnika znajduje się w ograniczonej przestrzeni o regularnym kształcie, na przykład sześcianu albo ściętego ośmiościanu. Układ ten symulowany jest tak, jakby cała przestrzeń była wypełniona jego kopiami. Atomy znajdujące się na jednej ścianie układu oddziałują z atomami na przeciwległej ścianie tak, jakby były one tuż obok.



Rysunek 2.4. Symulacja w przestrzeni periodycznej przedstawiona schematycznie w dwóch wymiarach. Żaden atom nie powinien oddziaływać z więcej niż jedną kopią każdego innego atomu, więc oddziaływania niewiążące muszą być ograniczane do wartości mniejszej niż R_c. Źródło:[32].

Podobnie atomy mogą przekraczać granice układu, ale zachowują się tak, jakby pojawiły się po jego przeciwnej stronie. Technika ta eliminuje artefakty brzegowe, ale może doprowadzić do pojawienia się artefaktów związanych z periodycznością. Ryzyko ich wystąpienia zmniejsza się ustalając odpowiednio dużą wielkość układu oraz ograniczając odległość na jaką symulowane są oddziaływania niewiążące (ang. *cut-off*). Żaden atom układu nie powinien oddziaływać ze swoją własną kopią ani z więcej niż jedną kopią jakiegokolwiek innego atomu. Ten sposób symulacji jest obecnie wykorzystywany najczęściej[35, 38]. Jego schematyczne przedstawienie znajduje się na rysunku 2.4.

2.9. Kontrola temperatury

Temperatura jest właściwością układu zależną od energii kinetycznej jego atomów. W mechanice molekularnej symulowany układ może mieć stałą energię przez co temperatura nie jest regulowana i zmienia się w zależności od energii potencjalnej układu. Alternatywą jest utrzymywanie w przybliżeniu stałej temperatury układu poprzez zmniejszanie lub zwiększanie jego energii kinetycznej w trakcie symulacji. Wykorzystuje się do tego algorytmy modelujące wymianę ciepła z otoczeniem zwane termostatami. Przykładowe termostaty to:

1. Termostat Berendsena. Dodaje on do równań ruchu wyrażenie przypominające tar-

cie, które prowadzi układ do zadanej temperatury. Jest prosty do zaimplementowania i szybki obliczeniowo, ale nie tworzy prawidłowego zespołu statystycznego w układzie.

2. Termostat Nose-Hoovéra. Działa podobnie jak termostat Berendsena, ale jest od niego bardziej wymagający obliczeniowo. Tworzy w układzie prawidłowy zespół statystyczny.
3. Termostat Anderséna. W każdym kroku czasowym z pewnym niewielkim prawdopodobieństwem poszczególnym atomom układu może zostać przypisana nowa prędkość, zgodna z rozkładem Maxwella dla danej temperatury. Tworzy w układzie prawidłowy zespół statystyczny.

Analogicznie istnieją algorytmy pozwalające na regulację ciśnienia w symulacji[32].

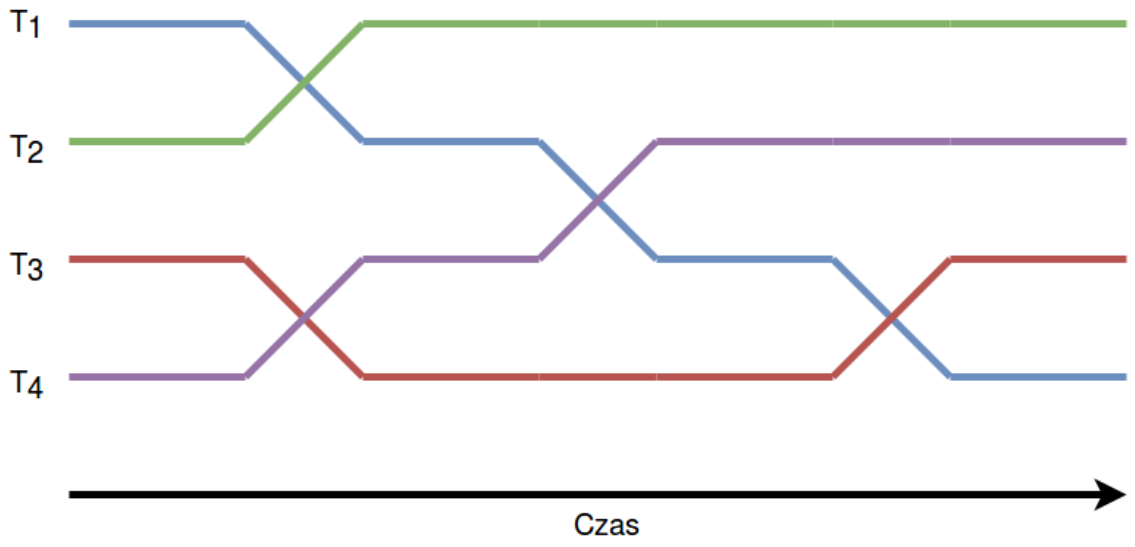
2.10. Termodynamika układu

Pod względem termodynamiki badany układ może być symulowany na kilka sposobów:

1. Układ mikrokanoniczny - ma stałą objętość oraz nie wymienia atomów i energii z otoczeniem (nie ma kontaktu z termostatem). Jest rzadko wykorzystywany w dynamice molekularnej ze względu na konieczność zachowania stałej energii, co sprawia że lokalnie niezależne elementy układu stają się ze sobą sprzężone.
2. Układ kanoniczny - ma stałą objętość, nie wymienia atomów z otoczeniem ale wymienia z nim energię (ma kontakt z termostatem). Taki układ jest często wykorzystywany w dynamice molekularnej ze względu na możliwość wymiany energii (ciepła) z otoczeniem.
3. Układ wielki kanoniczny - ma stałą objętość, może wymieniać z otoczeniem zarówno atomy jak i energię. Taki układ jest wykorzystywany kiedy system musi być w równowadze zarówno termicznej jak i chemicznej ze środowiskiem.
4. Układ izotermalno-izobaryczny - ma stałą temperaturę i ciśnienie oraz nie wymienia atomów z otoczeniem, ale jego objętość może się zmieniać. Taki układ jest wykorzystywany kiedy system musi zachowywać stałe ciśnienie.[40].

2.11. Dynamika molekularna z wymianą replik

W celu zwiększenia tempa eksploracji krajobrazu energetycznego układu stosuje się często dynamikę molekularną z wymianą replik (REMD). Polega ona na symulowaniu kilku kopii (replik) układu, każdej w określonej temperaturze odmiennej od pozostałych replik. Zakres stosowanych temperatur jest różny, często od około 280K do około 500K. Temperatury poszczególnych replik są parami wymieniane co pewien czas symulacji. Umożliwia to łatwiejsze wychodzenie układu z lokalnych minimów energetycznych, co jak wspomniano przyspiesza eksplorację jego krajobrazu energetycznego[41, 42]. Schemat przedstawiający wymianę temperatur pomiędzy replikami w trakcie takiej symulacji przedstawiony jest na rysunku 2.5.



Rysunek 2.5. Schematyczne przedstawienie wymiany temperatur podczas przeprowadzania REMD. Źródło: Własne na podstawie [42]

Odmianą tej metody jest multipleksowa dynamika molekularna z wymianą replik (MREMD). Polega ona na tym, że w każdej wybranej temperaturze symulowane jest kilka replik, a wymiany mogą zachodzić zarówno wśród replik o różnych temperaturach jak i wśród replik o tej samej temperaturze[43].

3. Analiza skupień i jej algorytmy

3.1. Wstęp

Analiza skupień (grupowanie, ang. *clustering*) jest procesem nienadzorowanej klasyfikacji danych (najczęściej wektorów albo punktów w wielowymiarowej przestrzeni) w grupy. Klasyfikacja ta odbywa się pod względem podobieństwa pomiędzy danymi, zdefiniowanego w określony, właściwy dla danego problemu i zbioru danych sposób. Dane przypisane do jednej grupy są bardziej podobne do siebie nawzajem niż dane przypisane do innych grup. Istnieje duża liczba algorytmów grupowania wykorzystywanych do różnych zadań[44]. Trzy z nich zostaną krótko omówione w dalszej części tego rozdziału.

Typowy protokół analizy skupień składa się z 3 do 5 kroków:

1. Przygotowanie właściwej reprezentacji danych wejściowych.
2. Zdefiniowanie sposobu pomiaru podobieństwa pomiędzy danymi.
3. Przeprowadzenie grupowania.
4. Abstrahowanie danych (opcjonalnie).
5. Ocena wyniku (opcjonalnie)[44].

Metoda ta jest wykorzystywana w wielu różnych dziedzinach jako jeden ze sposobów eksploracyjnej analizy danych. Na przykład:

— Segmentacja (podział na regiony) obrazów, będąca jednym z etapów maszynowej analizy zdjęć.
— Rozpoznawanie obiektów trójwymiarowych i osób.
— Przechowywanie i wyszukiwanie informacji (dokumentów) w ich zbiorach.

— Eksploracja danych (ang. *data mining*), na przykład grupowanie stron internetowych na podstawie ich treści i ocenianiu dostępności depozytów zasobów naturalnych[44].

3.2. Podobieństwo struktur molekularnych

W przypadku molekuł biologicznych danymi są najczęściej współrzędne poszczególnych atomów w ich strukturach. Mogą w tym celu zostać użyte wszystkie jej atomy, ale najczęściej zbiór ten jest ograniczany. W przypadku białek do łańcucha głównego albo tylko do węgli α. Pomiar podobieństwa pomiędzy strukturami odbywa się zazwyczaj poprzez obliczanie RMSD pomiędzy ich parami. dla dwóch optymalnie nałożonych na siebie struktur i oraz j:

$$RMSD_{i,j} = \sqrt{\frac{1}{N}\sum_{k=1}^{k=N}(x_{i,k} - x_{j,k})^2}$$

gdzie N to liczba atomów w strukturach, a $x_{i,k}$ oraz $x_{j,k}$ są współrzędnymi atomu k w obydwu strukturach po ich wycentrowaniu i obrocie do uzyskania najlepszego dopasowania[45]. Zmierzone podobieństwa struktur są następnie używane do przeprowadzenia podziału zbioru danych (struktur białek) na grupy[44].

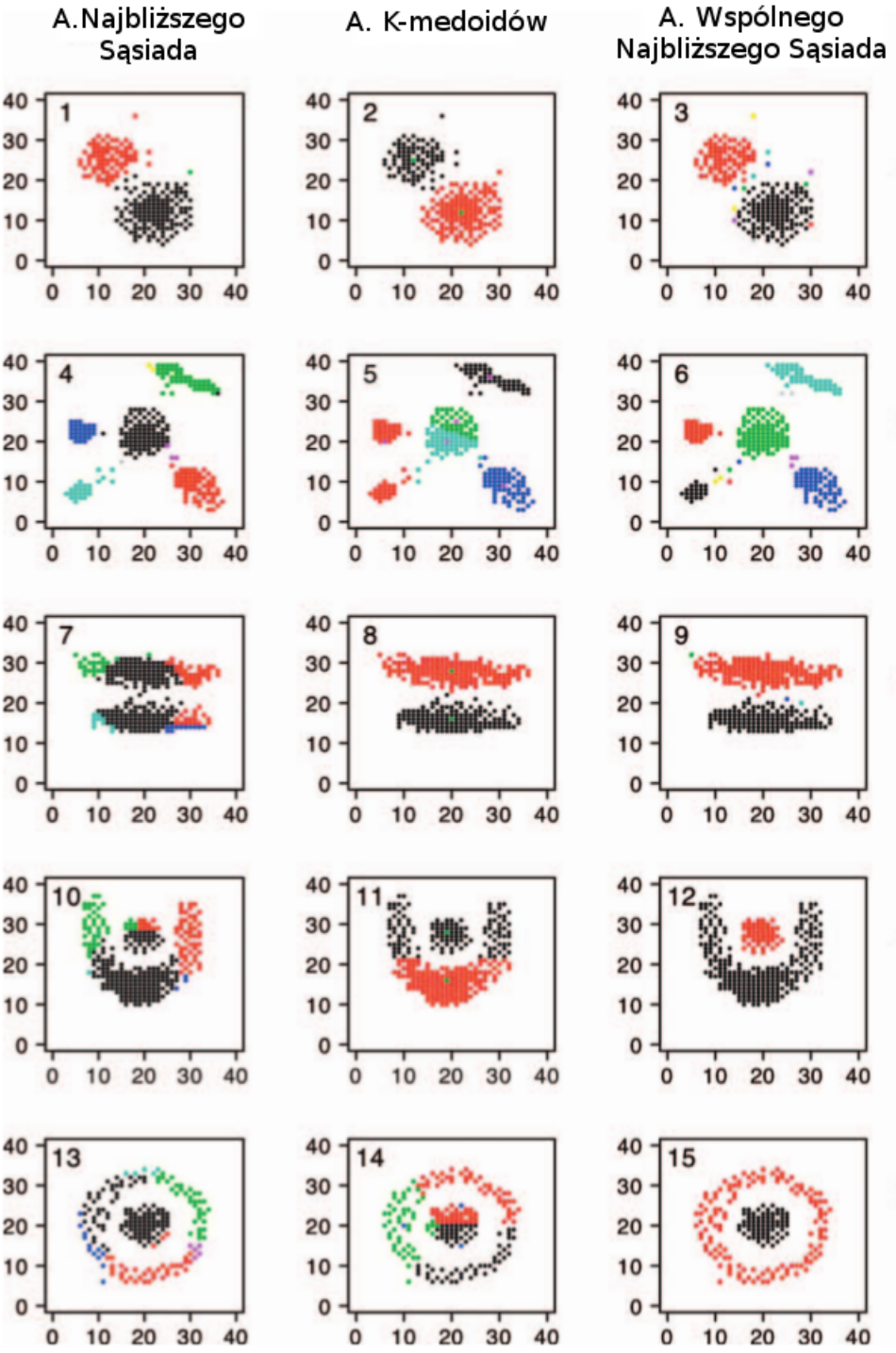
3.3. Przegląd algorytmów grupowania

Algorytm Najbliższego Sąsiada (ang. *Nearest Neighbor Algorithm*) jest szybkim i prostym algorytmem, w którym sąsiadami danego punktu danych są wszystkie inne punkty znajdujące się do określonej odległości od niego. Punkt mający największą liczbę sąsiadów staje się centrum grupy do której należą wszyscy jego sąsiedzi. Wszystkie te punkty są usuwane ze zbioru danych i wyszukiwany jest następny punkt z największą liczbą sąsiadów. Proces ten jest powtarzany póki wszystkie punkty nie zostaną przypisane do grup. Jedynym parametrem tego algorytmu jest maksymalny dystans sąsiedztwa. Jego wadą jest to, że daje słabe wyniki dla zbiorów danych o skomplikowanej strukturze wewnętrznej[45].

Algorytm K-medoidów (ang. *K-Medoids Algorithm*) dzieli zbiór danych na z góry określoną liczbę grup. Najpierw losowo przypisuje on dane do grup, a następnie iteracyjnie poprawia to przypisanie według określonych kryteriów zbieżności. Proces ten pozwala na poprawę błędnych przypisań w następnych krokach. Algorytm, jak wspomniano, wymaga liczby grup jako parametru wejściowego a na ostateczny podział ma wpływ początkowe, losowe przypisanie do nich danych. Mimo tych wad jeśli liczba grup jest znana może on dać lepsze wyniki niż Algorytm Najbliższego Sąsiada[45].

Algorytm Wspólnego Najbliższego Sąsiada (ang. *Common Nearest Neighbor Algorithm*) jest odmianą Algorytmu Jarvisa-Patricka. Opiera się on na lokalnej gęstości punktów danych. Podział zaczyna się od losowego wyboru jednego z punktów który w ten sposób staje się zalążkiem grupy. Kolejne punkty danych są przyłączane do tej grupy kiedy są z nią połączone ciągłym obszarem o zadanej gęstości danych. Kiedy do danej grupy nie da się dodać nowych punktów jest ona usuwana ze zbioru. Proces ten jest powtarzany póki nie zostaną przetworzone wszystkie dane. Algorytm ten tworzy grupy będące obszarami o wysokiej gęstości, podobnie jak intuicyjnie czyni to człowiek. Jednocześnie jest to najbardziej skomplikowany i najwolniejszy z prezentowanych tu algorytmów[45].

Efekty działania tych trzech algorytmów (uzyskany podział na grupy) na kilku przykładowych, dwuwymiarowych zbiorach danych znajdują się na rysunku 3.1.



Rysunek 3.1. Porównanie wyników grupowania przy użyciu trzech opisanych w tekście algorytmów, dla pięciu przykładowych zbiorów dwuwymiarowych danych. Źródło: [45]

4. Modele Markova

Proces stochastyczny to funkcja zależna od czasu, której wartości są zmiennymi losowymi. Jeżeli parametr czasowy przyjmuje tylko pewne dyskretne wartości (na przykład pomiar jest dokonywany co minutę) to proces ten staje się ciągiem zmiennych losowych i nazywamy go łańcuchem. Jeśli wartości tej funkcji również należą do dyskretnego zbioru nazywamy je stanami układu, a cały proces to zjawisko kolejnych zmian stanu układu[46].

Modele Markova w teorii prawdopodobieństwa są modelami stochastycznych procesów wykorzystywanymi do modelowania losowo zmieniających się systemów. Zakładają one że badany system posiada tzw. własność Markova która mówi, że rozkład prawdopodobieństwa następnego (przyszłego) stanu układu zależy tylko od jego obecnego stanu, a poprzednie stany nie mają na to żadnego wpływu (układ nie posiada pamięci). Z tego warunku wynika, że można wyznaczyć rozkład prawdopodobieństwa dowolnego stanu w przyszłości na podstawie tylko obecnego stanu układu. Modele Markova są często wykorzystywane do badania serii czasowych. Dzielimy je na kategorie ze względu na to, czy badany proces jest autonomiczny czy kontrolowany i czy stan systemu może zostać w całości zaobserwowany[46, 47].

4.1. Łańcuch Markova

Łańcuch Markova jest typem modelu Markova w którym badany system jest autonomiczny a jego stan może zostać w całości zaobserwowany. Zarówno zbiór argumentów jak i zbiór wartości funkcji go opisującej są dyskretne, a sama funkcja posiada opisaną powyżej własność Markova. Przykładem procesów spełniających te cechy mogą być serie czasowe. Możliwe wartości zmiennych losowych (wartości funkcji) określa zbiór zwany przestrzenią stanów. Analiza sekwencji zmiennych pozwala na określenie praw-

dopodobieństwa przejścia układu pomiędzy parami stanów. Im więcej danych (dłuższy łańcuch) zostanie poddanych analizie tym dokładniejszy będzie powstały model. Prawdopodobieństwa wszystkich możliwych przejść przedstawiane są często w tabeli zwanej macierzą przejść (albo macierzą stochastyczną), albo schematycznie na tzw. grafie przejść[47]. Poszczególne elementy macierzy przejść ($T_{i,j}$, gdzie i oraz j są stanami układu) są obliczane następującym wzorem[48]:

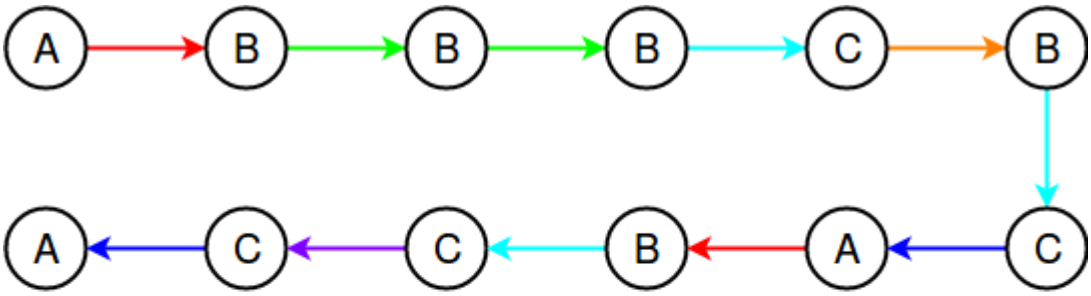
$$T_{i,j} = \frac{\text{liczba przejść z i do j}}{\text{liczba wszystkich przejść z i}}$$

Przykładowy, prosty Łańcuch Markova wraz z reprezentującymi go macierzą i grafem przejść znajduje się na rysunku 4.1. Reprezentuje on prosty układ posiadający zbiór stanów $\{A, B, C\}$. Na górze rysunku znajduje się łańcuch - seria kolejnych obserwacji stanu układu. Takie same przejścia oznaczono w nim takimi samymi kolorami strzałek. Poniżej znajduje się macierz przejścia w postaci tabeli. Otrzymuje się ją zliczając wszystkie przejścia z danego stanu do poszczególnych elementów zbioru stanów i dzieląc je przez ich sumę. W ten sposób powstaje jeden wiersz macierzy przejścia (tzw. wektor stochastyczny), który powinien sumować się do wartości 1. Proces ten jest powtarzany dla wszystkich stanów. Kolory elementów macierzy odpowiadają kolorom przejść w łańcuchu. Na dole rysunku znajduje się graf przejścia narysowany na podstawie tej macierzy. Jest to graf skierowany w którym mogą występować pętle, a wartości przypisane krawędziom są prawdopodobieństwami danego przejścia. Kiedy prawdopodobieństwo danego przejścia wynosi 0 na grafie zazwyczaj nie umieszcza się krawędzi mu odpowiadającej. Na tym grafie ponownie kolory krawędzi odpowiadają kolorom przejść w łańcuchu. Wielkość okręgów reprezentujących wierzchołki zależy od populacji odpowiadającego mu stanu układu.

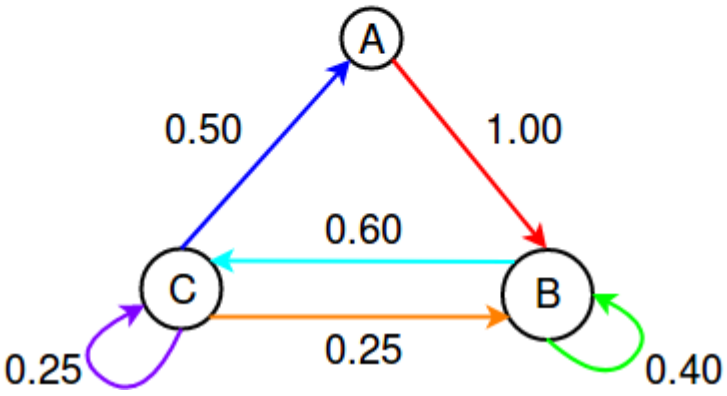
4.2. Zastosowanie Łańcuchów Markova

Łańcuchy Markova wykorzystywane są m.in. w meteorologii (do przewidywania wielkości opadów z dnia na dzień), informatyce (w systemach rozpoznawania mowy), biologii (do badania dynamiki populacji i ich genetyki), czy w ekonomii finansowej[47].

W biologii molekularnej są używane m.in do tworzenia sztucznych peptydów o określonych właściwościach[49] i wykrywania promotorów w sekwencjach DNA[50]. Używane też są bardziej żartobliwie np. do tworzenia losowych tekstów na podstawie dostatecznie dużego zbioru tekstów wejściowych[51]. Łańcuchy Markova są także wykorzystywane w mechanice molekularnej, co opisuję w rozdziale 5.



	A	B	C
A	0	1,00	0
B	0	0,40	0,60
C	0,50	0,25	0,25



Rysunek 4.1. Przykładowy łańcuch Markova. Na górze pokazano serię czasową zdarzeń (stanów). Poniżej utworzone na jej podstawie macierz i graf przejścia. Kolorami zaznaczono różne rodzaje przejść. Szczegółowy opis w tekście. Źródło: Własne.

5. Zastosowanie analizy skupień i łańcuchów Markova w mechanice molekularnej

5.1. Wstęp

W ostatnich latach analiza skupień i konstruowanie na jej podstawie modelu Markova stają się coraz bardziej popularną metodą analizy danych pochodzących z symulacji dynamiki molekularnej. Technika ta pozwala na analizę stabilnych i metastabilnych konformacji systemu, ich energii oraz przejść pomiędzy nimi. W ten sposób można uzyskać dane na temat ewolucji systemu i jego stanu równowagi w stosunkowo długich skalach czasowych[52]. Umożliwia to badanie powolnych procesów zachodzących w makromolekułach lub z ich udziałem, takich jak zwijanie białek, aktywacja kinaz oraz receptorów związanych z białkami G i związane z nimi złożone zmiany konformacyjne makromolekuł[53].

5.2. Konstrukcja modeli Markova

Wykorzystanie tej techniki wymaga prawidłowo skonstruowanego modelu Markova, najlepiej opartego na zbiorze trajektorii (n.p. pochodzących z symulacji typu REMD), ponieważ pojedyncza trajektoria może na długi czas symulacji utknąć w lokalnym minimum energetycznym układu. Najczęściej przy konstruowaniu modelu Markova odrzuca się informacje kinetyczne (o prędkości atomów w symulacji) i zakłada się, że konformacje podobne do siebie strukturalnie są też podobne kinetycznie. Prawidłowy Model Markova pozwala na modelowanie kinetyki układu w czasie znacznie dłuższym (na-

wet o kilka rzędów wielkości) niż długość trajektorii użytych do jego skonstruowania. Umożliwia to uzyskanie ogólnego obrazu ścieżek i stanów pośrednich jakie przyjmuje badana makromolekuła. Metastabilne stany uzyskane w modelu Markova mogą posłużyć jako konformacje startowe w następnym cyklu dynamiki molekularnej, analizy skupień i budowy modelu Markova. Pozwala to na łatwiejsze zebranie lepszej jakości danych w porównaniu z symulacjami startującymi z losowych, rozwiniętych konformacji albo ze struktury natywnej[54, 55].

W trakcie tworzenia łańcucha Markova mogą pojawić się dwa rodzaje błędów. Pierwszym są błędy deterministyczne, które wynikają z trudności prawidłowego podziału danych z symulacji na stany. Nieprawidłowy podział może sprawić, że uzyskany model nie będzie spełniał własności Markova - jeden jego stan będzie reprezentował więcej niż jeden niezależny stan symulowanego układu. Błąd ten można zmniejszyć polepszając podział, ale jego właściwe wyznaczenie jest trudne. Zbyt płytki podział (zbyt mała liczba stanów) powoduje utratę dokładności uzyskanego modelu. Zbyt głęboki podział (zbyt duża liczba stanów) utrudnia obliczenia nie polepszając jakości uzyskanego modelu, a niekiedy może nawet pogorszyć jakość modelu, jeśli układ spędza w jednym stanie zbyt mało czasu żeby zachować własność Markova. Wielkość (i przez to liczba) uzyskanych grup ma wpływ na informacje jakie można odczytać z modelu. Bardziej szczegółowy podział może dostarczyć lepszych informacji o kinetyce układu. Gruboziarnisty model (mniejsza liczba grup) pozwala wizualizować zmiany konformacyjne układu w sposób bardziej zrozumiały dla człowieka, co pozwala na łatwiejsze formułowanie kolejnych hipotez o zachowaniu badanego układu. Z tych powodów prawidłowe przeprowadzenie podziału jest bardzo istotnym krokiem w konstruowaniu modelu Markova[54, 56, 57].

Drugim rodzajem błędów są błędy statystyczne, które wynikają z ograniczonego zakresu danych zebranych w trakcie symulacji. Zmniejsza się go zbierając więcej danych (przeprowadzając dłuższe symulacje lub symulując więcej trajektorii w przypadku REMD) i przygotowując na ich podstawie kilka różnych modeli Markova[56].

W konstruowaniu modelu Markova znaczenie ma użyty algorytm grupowania. dla niewielkich układów różne algorytmy często dają podobne wyniki. W przypadku więk-

szych makromolekuł użyty algorytm może wywierać duży wpływ na uzyskany podział na stany, a przez to na zbudowany model Markova. Co istotne algorytmy geometryczne (k-średnich, grupowanie hierarchiczne i Bayesowskie) sprawdzają się równie dobrze, a w niektórych sytuacjach nawet lepiej niż algorytmy kinetyczne (Perron-cluster). Potwierdza to wspomniane wyżej założenie, że konformacje podobne strukturalnie są też podobne kinetycznie[54].

Istotnym zagadnieniem jest także dobranie właściwego czasu opóźnienia (ang. *lag time*). Jest to odstęp, liczony po czasie symulacji, pomiędzy parami struktur wykorzystywanymi do zliczania przejść w trakcie tworzenia matrycy przejścia modelu Markova. Zbyt duża wartość tego parametru może spowodować pogorszenie jakości modelu ze względu na mniejszą liczbę branych pod uwagę przejść. Zbyt mała również może spowodować pogorszenie jakości modelu ponieważ uzyskany łańcuch przejść może nie spełniać własności Markova (może zachowywać pamięć o poprzednich stanach)[57]. W przypadku tworzenia modeli z których wywodzone mają być własności kinetyczne układu najczęściej używa się czasu pomiędzy 1 ns a 100 ns[54, 58].

5.3. Literaturowe przykłady zastosowania metody

W 2009 roku Vijay S. Pande z zespołem wykorzystali analizę skupień i łańcuch Markova do analizy wielu różnej długości trajektorii pochodzących z symulacji dynamiki molekularnej białka NTL9(1-39), składającego się z 39 reszt aminokwasowych. Symulacje przeprowadzili przy pomocy platformy Folding@Home, czego wynikiem była duża liczba bardzo krótkich trajektorii, które następnie były ze sobą łączone w długie trajektorie. W grupowaniu wykorzystali algorytm k-średnich i hierarchiczne budowanie stanów. Najpierw podzielili struktury uzyskane z trajektorii na 100 000 mikrostanów z których zbudowali 2000 makrostanów. W konstruowaniu modelu Markova użyli trajektorii o temperaturze 370K i czasów opóźnienia od 1 ns do 32 ns. Następnie przy użyciu zachłannego algorytmu z nawrotami ustalili 10 najczęstszych ścieżek zwijania ze struktury rozwiniętej w natywną. Odkryli istnienie dwóch głównych stanów układu - rozwiniętego i zwiniętego oraz dwóch istotnych stanów pośrednich pomiędzy nimi.

Te stany pośrednie istnieją równolegle wobec siebie, jako elementy dwóch różnych ścieżek zwijania białka i różnią się upakowaniem regionów hydrofobowych. Poza nimi istnieją inne stany, tworzące złożoną sieć. W przeanalizowanym białku autorzy nie znaleźli jednej, dominującej ścieżki zwijania, ale całą sieć stanów pośrednich z których kilka jest bardziej znaczących od pozostałych[58].

Ten sam zespół zastosował tę metodę w analizie wcześniej opublikowanych, bardzo długich (100 µs) trajektorii zwijania domeny FiP35 WW. Uzyskali oni nowe, niewidoczne bezpośrednio w trajektoriach, możliwe ścieżki zwijania tego białka. Odkryli także zbiór grup zawierających struktury dość podobne do natywnej, pomiędzy którymi badany układ może bardzo szybko i często przechodzić. W tym zbiorze zawierają się stany istotne z punktu widzenia funkcjonalności białka, a zmiany konformacji struktury pomiędzy nimi mogą odpowiadać za mechanizm jego działania[59].

Technika ta jest też używana do badania zmian konformacyjnych białek związanych z pełnionymi przez nie funkcjami. Za jej pomocą zbadano uwalnianie jonu difosforowego przez bakteryjną polimerazę RNA. Odkryto w ten sposób prosty, dwustanowy mechanizm tego procesu[60]. Innym przykładem jest analiza działania bakteryjnego, zależnego od jonów sodu, transportera leucyny. Zidentyfikowała ona 6 metastabilnych konformacji i prawdopodobieństwa przejść pomiędzy nimi[61].

Analiza skupień w połączeniu z budowaniem modeli Markova może być stosowana nie tylko do białek. W 2016 roku użyto jej do analizy rozwijania aptameru łączącego się z trombiną. Cząsteczka ta jest G-kwadrupleksem, czyli krótką, czteroniciową strukturą DNA. Bierze udział w kaskadzie krzepnięcia krwi. Odkryto dwie główne ścieżki rozwijania tej cząsteczki. Jedna jest zgodna z danymi uzyskanymi z eksperymentów NMR i przechodzi przez wiele stanów pośrednich. Istotne w niej są dwa ważne stany pośrednie: jeden przypomina G-triplex, a drugi jest obligatoryjnym stanem przejściowym pomiędzy zwiniętą a rozwiniętą konformacją struktury. Jest to główna ścieżka zwijania. Druga ścieżka jest prostsza, zawiera mniej stanów pośrednich i nie przechodzi przez dwa opisane wyżej istotne dla pierwszej ścieżki stany[62].

Część II

Cel pracy

Celem niniejszej pracy doktorskiej było zbadanie mechanizmu fałdowania, czyli sposobu uzyskiwania struktury trzeciorzędowej, dla kilku wybranych, niewielkich białek. W tych badaniach przyjąłem następującą hipotezę badawczą: Białka nie posiadają jednej dominującej ścieżki fałdowania rozumianej jako jednoznaczna i powtarzalna sekwencja zdarzeń prowadząca od struktury całkowicie rozwiniętej do konformacji natywnej. Dodatkowo struktura natywna nie jest idealnie stabilna i białko może z niej rozwijać się w różne, częściowo zwinięte struktury. Tworzy się w ten sposób sieć częściowo stabilnych stanów którą chciałem pokazać. Istotnym elementem moich badań jest zaproponowanie i implementacja nowej metody obliczeniowej mającej za zadanie dostosować procedurę analizy skupień do charakterystyki danych pochodzących z symulacji. Została ona opisana w rozdziale 6.3.3.

Moją strategią mającą na celu weryfikację opisanej hipotezy było podzielenie pracy na następujące etapy:

1. Wybranie kilku niewielkich białek oraz przeprowadzenie dla nich, we współpracy z doktorem Arturem Giełdoniem z Wydziału Chemii UG, symulacji multipleksowej dynamiki molekularnej z wymianą replik wykorzystując dwa różne pola sił - pełnoatomowe (AMBER) oraz gruboziarniste (UNRES).
2. Przygotowanie środka technicznego do realizacji celu w postaci programu, który zaprojektowałem i napisałem w języku C z wykorzystaniem biblioteki OpenMP służącej do przeprowadzenia obliczeń równoległych. Jego zadaniem jest wykonywanie analizy skupień, przy użyciu zmodyfikowanego przeze mnie Algorytmy Najbliższego Sąsiada, oraz budowa Łańcucha Markova na podstawie struktur pochodzących z symulacji. W swoim założeniu mój program ma umożliwiać szybką analizę nawet bardzo dużych zestawów danych.
3. Przeanalizowanie serii czasowych struktur, pochodzących z uzyskanych w trakcie symulacji trajektorii dynamiki molekularnej, przy pomocy opisanego powyżej programu, co umożliwiło mi wizualizację sieci stanów pośrednich i procesu fałdowania białka.

Część III

Materiały i metody

6. Program *pdbclust*

6.1. Wstęp

Program pdbclust jest utworzonym przeze mnie uniwersalnym programem służącym do przeprowadzenia analizy skupień i budowy gruboziarnistego modelu Markova na podstawie danych pochodzących z dynamiki molekularnej. W swoim założeniu ma pomagać w analizie ścieżek zwijania białek. Dzięki włączeniu znanej struktury natywnej w budowanie modelu Markova pozwala on na badanie jak działa pole sił oraz czy i w jaki sposób prowadzi do struktury natywnej. Generalnie nie ma za zadanie uzyskania modelu pozwalającego na badanie kinetyki układu.

W tym programie zastosowałem znaczącą modyfikację jednego ze standardowych algorytmów analizy skupień, co opisałem w rozdziale 6.3.3. Jej celem jest dostosowanie algorytmu do charakterystyki użytych danych, czyli trajektorii molekularnych, co powinno doprowadzić do uzyskania lepszych wyników. Drugą istotną rzeczą była możliwość grupowania bardzo dużej liczby struktur. Wzrost ilości danych, czyli m.in. struktur, uzyskiwanych z symulacji sprawia, że potrzebne są nowe podejścia do ich analizy. Przeprowadzone przeze mnie pomiary wydajności mojego programu, znajdujące się w rozdziale 10 pokazują, że program może być użyty do analizy setek tysięcy i więcej struktur i przeprowadzić ją w rozsądnym czasie.

Program ten dzieli struktury na grupy pod względem podobieństwa strukturalnego, następnie tworzy łańcuch Markova z przejść pomiędzy określoną liczbą najliczniejszych grup. Przejścia te są sumowane w matrycy przejść i na jej podstawie tworzona jest prosta graficzna reprezentacja zachowania układu. Program pozwala na jednoczesną analizę pojedynczej albo wielu trajektorii, pochodzących na przykład z symulacji typu REMD. Może on działać zarówno sekwencyjnie jak i równolegle. Instrukcja obsługi w języku angielskim znajduje się w dodatku do niniejszej pracy. W tym rozdziale

postawiłem sobie za zadanie szczegółowo opisać sposób działania napisanego przeze mnie programu.

6.2. Używane technologie i narzędzia programistyczne

6.2.1. Język C

Język C jest strukturalnym, imperatywnym językiem programowania używającym statycznego typowania. Utworzony został w AT&T Bell Labs pomiędzy rokiem 1969 a 1973. Jego głównym twórcą był Dennis Ritchie. W roku 1978 Brian Kernighan i Dennis Ritchie opublikowali pierwszą edycję książki "The C Programming Language", która stała się nieformalnym standardem tego języka. Jego pierwszy oficjalny standard pojawił się w 1989 roku. Kolejne jego wersje opublikowano w latach 1999 oraz 2011. Język C jest typowym przykładem języka wysokiego poziomu ówczesnych czasów ze względu na multiplatformowość oraz uniwersalność zastosowań. W języku tym został napisany m.in system UNIX. Ze względu na tworzenie wydajnych programów wynikowych jest on, razem z językami FORTRAN i C++, stosowany do zadań wymagających dużej liczby obliczeń, między innymi w modelowaniu molekularnym[63, 64]. Mimo swojego wieku pozostaje wciąż popularnym językiem programowania[65, 66].

Istnieją napisane w tym języku biblioteki programistyczne przeznaczone do obsługi plików w formacie PDB. Zdecydowałem się ich nie wykorzystywać ze względu na wydajność pod względem wykorzystywanej pamięci. Dostępne biblioteki wczytują większość lub wszystkie informacje z pliku PDB, podczas gdy mój program potrzebuje tylko typu i współrzędnych atomów. Ma to istotne znaczenie w przypadku przetwarzania bardzo dużej liczby struktur pochodzących z symulacji[67].

6.2.2. OpenMP

OpenMP jest interfejsem programowania aplikacji (API), czyli zestawem funkcji, protokołów i danych przeznaczonym do tworzenia programów komputerowych. Umożliwia on tworzenie aplikacji przeprowadzających wielowątkowe, równoległe obliczenia. Wykorzystuje przy tym model pamięci współdzielonej, w którym wszystkie wątki da-

nego programu posiadają dostęp do wspólnego obszaru pamięci. Ze względu na tą decyzję projektową przeznaczony jest głównie do działania na pojedynczych maszynach wyposażonych w procesory wielordzeniowe. Obecnie OpenMP dostępny jest dla języków programowania C, C++ oraz FORTRAN dla wielu platform zarówno sprzętowych jak i systemowych[68, 69]. W mojej pracy wykorzystałem implementację OpenMP będącą częścią pakietu GCC.

Główną alternatywą dla OpenMP jest MPI. Jest on protokołem przesyłania danych pomiędzy wątkami programów komputerowych uruchomionymi na jednej lub wielu maszynach w sieci. Wykorzystuje on pamięć rozproszoną, osobną dla każdego wątku. Podobnie jak OpenMP jest on dostępny dla języków programowania C, C++, FORTRAN, a także C#, Python i innych, dla wielu platform zarówno sprzętowych jak i systemowych[70, 71].

Ze względu na wykorzystywanie przez niego pamięci współdzielonej z tych dwóch możliwości zdecydowałem się wybrać OpenMP. Charakterystyka danych i sposobu działania mojego programu wymagałyby przesyłania bardzo dużych ilości informacji pomiędzy wątkami w przypadku wykorzystania standardu MPI. Wpłynęłoby to bardzo negatywnie na wydajność aplikacji. Dodatkowo zdecydowana większość obliczeń w programie wykonywanych jest niezależna od siebie nawzajem. Pozwala to na łatwe unikanie tzw. (ang.) *race conditions*, czyli zależności końcowego stanu programu od tego w jakiej kolejności zostały wykonane zrównoleglone obliczenia, mimo stosowania współdzielonego modelu pamięci.

6.2.3. Pozostałe technologie i narzędzia

Wizualizację wyników programu przeprowadzam przy pomocy programu Graphviz. Jest to otwarty (typu (ang.) *open-source*) program przeznaczony do wizualizacji danych w postaci grafów. Przyjmuje on tekstowy plik wejściowy opisujący dane i na jego podstawie tworzy ich graficzną reprezentację. Pozwala w szerokim stopniu kontrolować wygląd wynikowej grafiki. Dostępny jest on w większości dystrybucji systemu Linux, a także w systemach Windows, Mac OS i Solaris[72].

Zadania programistyczne będące częścią mojej pracy wykonywałem przy wykorzy-

staniu komputera działającego pod kontrolą systemu operacyjnego Linux Mint[73]. Kod programu tworzyłem przy pomocy otwartych zintegrowanych środowisk programistycznych (IDE) Code::Blocks[74] oraz Geany[75]. Program kompilowałem przy użyciu pakietu kompilatorów i narzędzi GCC[76]. W celu umożliwienia automatycznej kompilacji programu napisałem skrypt dla programu make (tzw. Makefile)[77]. W celu zidentyfikowania problemów z zarządzaniem pamięcią przez mój program użyłem aplikacji Valgrind[78]. Wykorzystywanym przeze mnie systemem kontroli wersji był Mercurial[79] a do wykonania kopii zapasowej danych w postaci zdalnego repozytorium wykorzystałem sieciową usługę BitBucket[80].

6.3. Opis działania programu

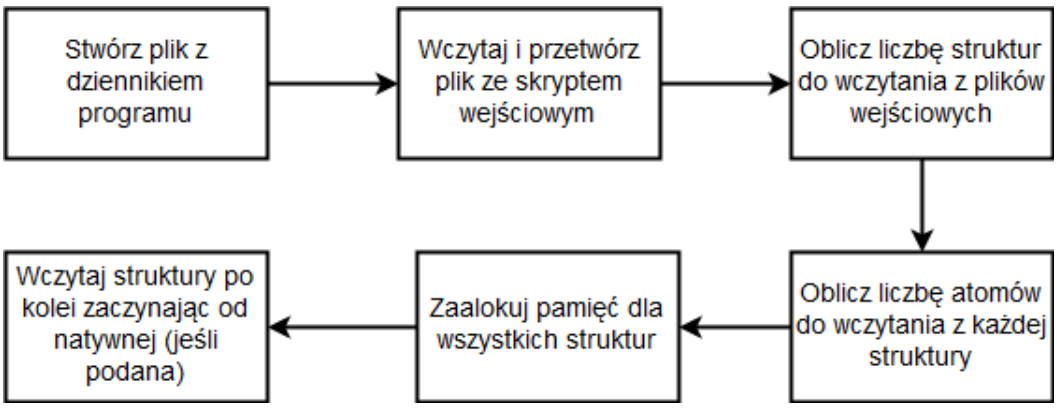
6.3.1. Dane wejściowe

Program pdbclust przyjmuje jako parametry wejściowe utworzony przez użytkownika plik konfiguracyjny zawierający parametry programu oraz pliki wynikowe pochodzące z symulacji biomolekularnych w jednej z trzech form:

— Pliki wynikowe programu UNRES skonwertowane do plików multiPDB przy pomocy narzędzia będącego częścią UNRESa. Zawierają one zarówno współrzędne atomów jak i informacje o temperaturze i energii struktur.
— Pliki wynikowe programu sander będącego częścią pakietu oprogramowania AMBER, skonwertowane do plików multiPDB razem z plikami mdout zawierającymi informacje o temperaturze i energii struktur.
— Pliki multipdb pochodzące z dowolnego źródła razem z dodatkowymi plikami zawierającymi informacje o temperaturze i energii struktur.

6.3.2. Wczytanie i przygotowanie danych

Program zaczyna działanie od utworzenia pliku do którego będzie zapisywany dziennik działania programu. W kolejnym kroku wczytywany jest podany przez użytkownika plik konfiguracyjny i ustawiane są zdefiniowane w nim opcje. Następnie tworzona jest, również na podstawie pliku konfiguracyjnego, lista plików wejściowych zawierających



Rysunek 6.1. Schemat inicjalizacji programu pdbclust. Źródło: Własne.

dane z symulacji razem z ich podziałem na poszczególne trajektorie. Dalej obliczana jest całkowita liczba struktur oraz liczba atomów w pojedynczej strukturze które zostaną wczytane. Na podstawie tego alokowana jest odpowiednia ilość pamięci operacyjnej komputera. Następnie do tej pamięci wczytywane są po kolei wszystkie struktury z plików wejściowych. Struktura natywna, jeśli jest podana, jest wczytywana jako pierwsza. W celu zmniejszenia ilości pamięci potrzebnej programowi z przetwarzanych struktur są wczytywane tylko współrzędne i identyfikatory atomów biorących udział w analizie skupień i inne niezbędne informacje (takie jak energia i temperatura danej struktury). Schematyczne przedstawienie tego procesu znajduje się na rysunku 6.1.

6.3.3. Analiza skupień

Analiza skupień struktur jest przeprowadzana przy użyciu zmodyfikowanego algorytmu najbliższego sąsiada (ang. *nearest neighbor algorithm*). Jest to szybki algorytm pozwalający na uzyskanie w jednym kroku gruboziarnistego podziału na grupy. Bazuje on na obliczaniu RMSD pomiędzy parami struktur. Użytkownik definiuje jakie atomy powinny być brane pod uwagę w tych obliczeniach. Domyślnie są to atomy węgla α aminokwasów. Jeśli struktury we wczytanych trajektoriach mają różne temperatury analiza skupień może zostać przeprowadzona na wszystkich z nich albo tylko na tych o określonej temperaturze.

Aplikacja wykorzystuje struktury o najniższej energii jako centra albo ziarna (ang. *kernel*) grup. Takie podejście jest uzasadnione, gdyż struktura o najniższej energii znajduje się w pobliżu minimum lokalnego. W przypadku braku informacji o energii

analiza skupień może zostać przeprowadzona losowo. Każda struktura może należeć tylko do jednej grupy. Wielkość uzyskanych grup jest modyfikowana przez parametr określający promień grupy, czyli maksymalną wartość RMSD obliczoną pomiędzy centrum i strukturą przynależną do grupy. Domyślnie wartość tego promienia wynosi 4Å. Po zakończeniu analizy skupień zdefiniowana przez użytkownika liczba największych grup jest wykorzystywana w dalszych obliczeniach oraz zapisywane są na dysku dwa pliki. Pierwszy zawiera dane o wszystkich strukturach (ich numery, energię, przynależność do grupy i RMSD względem centrum). Drugi zawiera informacje o strukturach tworzących największe grupy wraz z liczbą struktur do nich przynależących.

Ta część programu jest najbardziej wymagająca obliczeniowo i przez to czasochłonna, w związku z tym użytkownikowi wyświetlany jest przybliżony pasek postępu przy pomocy semigrafiki.

Modyfikacja Algorytmu Najbliższego Sąsiada

W oryginalnej wersji algorytmu najbliższego sąsiada centra grup są strukturami, które mają największą liczbę sąsiadów (struktur o RMSD względem nich nie większym od danego). Jak wspomniano w akapicie powyżej program pdbclust w swoim głównym założeniu wykorzystuje inną metodę. Centrami grup są struktury o najniższej energii, czyli znajdujące się w lokalnych minimach (lub ich pobliżu) krajobrazu energetycznego układu. Zgodnie z tym co opisałem w części niniejszej pracy na temat modelowania molekularnego takie punkty reprezentują stabilne stany układu i są interesujące z biochemicznego punktu widzenia. Algorytm ten wybrałem ze względu na małą złożoność obliczeniową i niewielkie zużycie pamięci, co umożliwia przetwarzanie bardzo dużej ilości struktur w rozsądnym czasie.

Szczegóły algorytmu analizy skupień

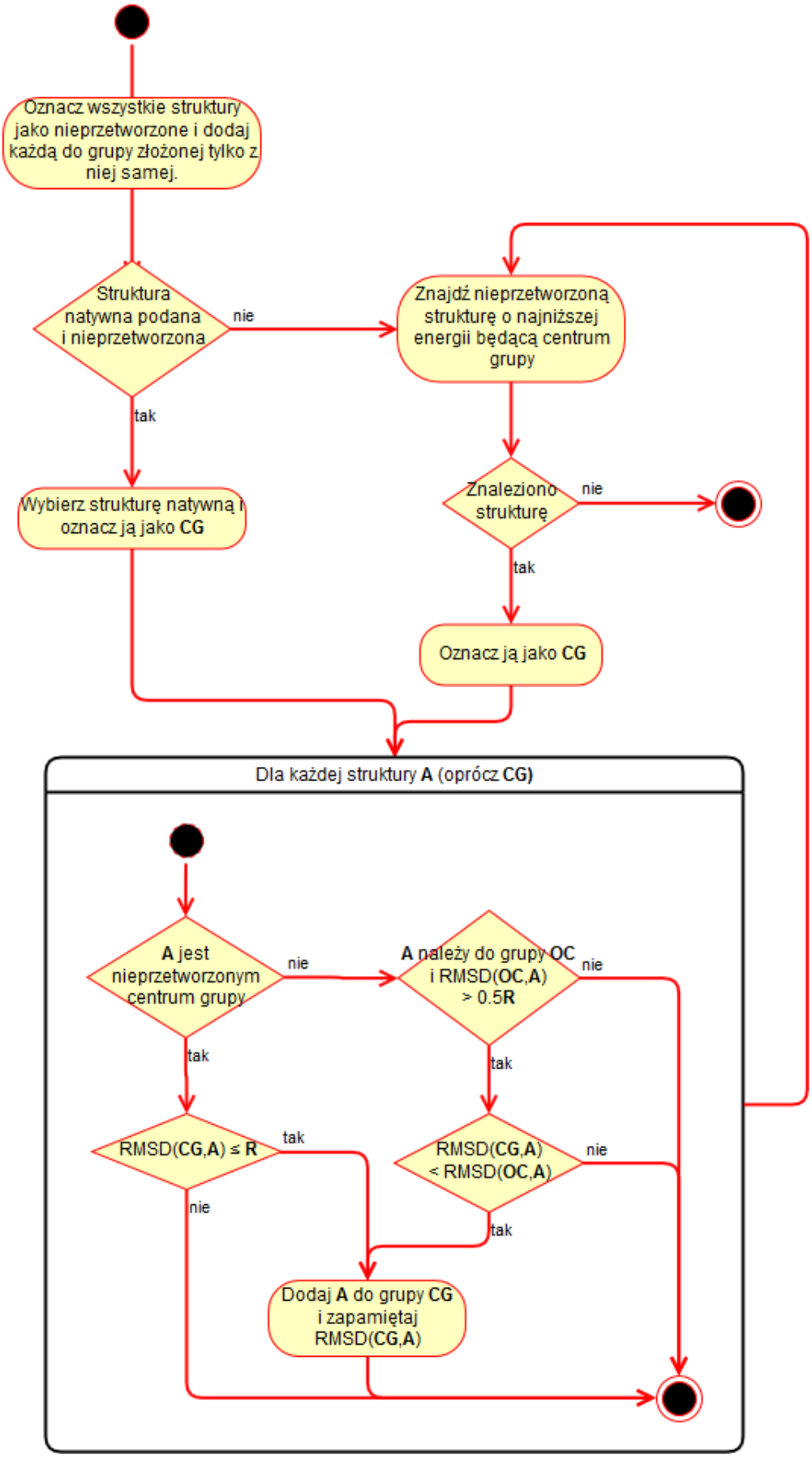
Początkowo wszystkie struktury są oznaczone jako nieprzetworzone i każda z nich należy do grupy składającej się tylko z niej samej. Jeśli podana jest struktura natywna przetwarzana jest ona jako pierwsza. Poza tym wyjątkiem program działa w następujący sposób w pętli dopóki istnieją odpowiednie struktury:

1. Znajdź nieprzetworzoną strukturę o najniższej energii, będącą centrum grupy. Weź ją jako nowe centrum grupy **CG**.
2. dla każdej pozostałej struktury **A**:
   a) Jeśli **A** jest nieprzetworzonym centrum grupy oblicz jej RMSD względem **CG**.
      i. Jeśli jest on mniejszy lub równy promieniowi grupy **R** dodaj **A** do grupy należącej do **CG** i zapisz jej RMSD względem niego.
      ii. W przeciwnym wypadku nie rób nic ze strukturą **A**.
   b) Jeśli **A** należy już do innej grupy i jej RMSD względem obecnego centrum grupy **OC** jest większy od połowy promienia grupy **R** oblicz jej RMSD względem **CG**.
      i. Jeśli jest on mniejszy niż RMSD względem **OC** dodaj **A** do grupy należącej do **CG** i zapisz jej RMSD względem niego nadpisując RMSD względem **OC**.
      ii. W przeciwnym wypadku nie rób nic ze strukturą **A**.
   c) W innych przypadkach nie rób nic ze strukturą **A**.
3. Oznacz **CG** jako przetworzoną.

Diagram UML aktywności opisujący ten algorytm znajduje się na rysunku 6.2. Zarówno poszukiwania nowego centrum grupy jak i poszukiwanie struktur które powinny do niej należeć można bardzo łatwo przeprowadzić równolegle. Wystarczy podzielić zbiór wszystkich struktur pomiędzy wątki, gdyż w obu tych przypadkach obliczenia dla pojedynczej struktury są niezależne od obliczeń na innych strukturach. W ten sposób zrównolegliłem tą część programu.

Obliczanie RMSD

Program pdbclust oblicza RMSD pomiędzy dwoma strukturami przy pomocy algorytmu Kabscha zaproponowanego przez Wolfganga Kabsha w 1976 roku[81]. Algorytm ten tworzy optymalną macierz rotacji która jest wykorzystywana do obliczenia minimalnego RMSD pomiędzy dwoma zbiorami odpowiadających sobie punktów. Składa się on z trzech etapów. Najpierw struktury reprezentowane przez oba zbiory punktów są nakładane na siebie. Odbywa się to poprzez przesunięcie (translację) centroidów obydwu zbiorów punktów do środka układu współrzędnych. W kolejnym kroku obliczana jest macierz kowariancji obydwu zbiorów punktów. Następnie przy pomocy dekompozycji



Rysunek 6.2. Schemat algorytmu grupowania wykorzystywanego przez program pdbclust. Źródło: Własne.

głównych składowych (SVD, singular-value decomposition) obliczana jest optymalna macierz rotacji. Wykorzystywana jest ona w końcu aby optymalnie nałożyć na siebie dwa zbiory punktów, zarówno pod względem translacji jak i rotacji i na podstawie tego obliczyć RMSD[81]. Program używa implementacji tego algorytmu wykorzystywanej przez pakiet oprogramowania UNRES. Została ona utworzona przez doktora Kennetha D. Gibsona z Cornell University w języku FORTRAN jako podprogram o nazwie *fitsq*[82].

Struktura natywna

Struktura natywna, o czym wspomina powyższy opis, jest traktowana w sposób uprzywilejowany. Przedstawia ona "domyślną" konfigurację przestrzenną danej struktury w związku z czym dostaje ona możliwość utworzenia grupy jako pierwsza. W dalszych obliczeniach, związanych z tworzeniem modelu Markova, również jest ona traktowana specjalnie i zawsze pojawia się jako jedna ze struktur wykorzystywanych do utworzenia matrycy przejścia. Pozwala to na stwierdzenie jakimi ścieżkami pole sił zwija badaną makromolekułę oraz jaki jest w nim związek pomiędzy największymi grupami a strukturą natywną. Potencjalnie może to zostać wykorzystane do ulepszania parametrów pola sił. Użytkownik wskazuje, jaka struktura powinna być traktowana jako natywna. Może to być struktura uzyskana eksperymentalnie, kandydat na stabilną strukturę natywną lub przejściową lub inna, w zależności od wymagań.

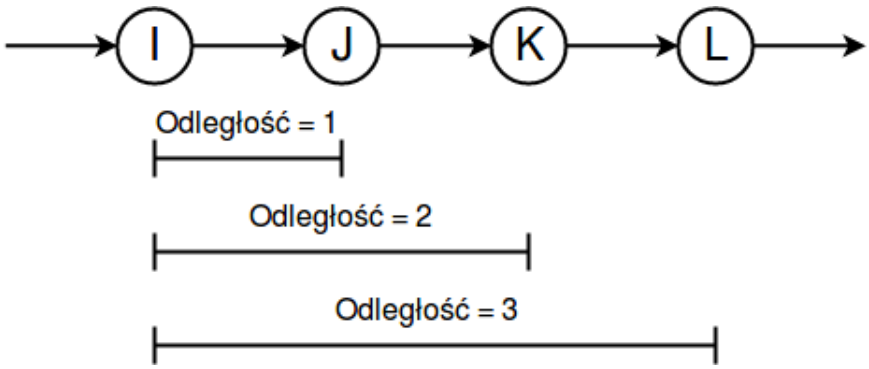
Ze względu na swoją charakterystykę obecne pola siłowe nigdy nie oddają doskonale fizycznej rzeczywistości, co wpływa na konformacje struktur uzyskanych w trakcie symulacji. Podobnie wiele struktur natywnych jest uzyskiwanych przy pomocy krystalografii, która także wpływa na przestrzenne rozmieszczenie atomów w strukturze. Struktura natywna może być też niskiej jakości bez względu na technikę, przy pomocy której została uzyskana. W związku z tym może ona nie pasować dobrze do warunków istniejących w trakcie symulacji. W takiej sytuacji użycie struktury natywnej zminimalizowanej, albo zminimalizowanej i podgrzanej wewnątrz danego pola sił może dać lepsze rezultaty, czyli liczniejszą grupę przez nią utworzoną.

6.3.4. Konstruowanie modelu Markova

Konstruowanie modelu Markova przez program rozpoczyna się znalezieniem grup zawierających największą liczbę struktur oraz alokacją pamięci dla struktur danych. Użytkownik może zdefiniować zarówno minimalną wielkość jak i maksymalną liczbę grup które powinny być użyte w tych obliczeniach. Następnie program zlicza przejścia pomiędzy tymi grupami następujące chronologicznie w trajektoriach. Może się ono odbywać dla wszystkich struktur albo tylko dla tych o określonej temperaturze. Aby takie przejście zostało zarejestrowane muszą być spełnione dwa warunki. Po pierwsze obydwie struktury muszą należeć do tej samej trajektorii. Po drugie struktura do której następuje przejście nie może być zbyt odległa chronologicznie (w serii czasowej) od struktury z której następuje przejście.

Odległość definiowana jest tutaj jako liczba struktur w trajektorii pomiędzy dwoma strukturami. Dwie struktury następujące bezpośrednio po sobie mają odległość równą jeden. Dwie struktury pomiędzy którymi istnieje jedna inna struktura, należąca do grupy niebranej pod uwagę przy konstrukcji modelu Markova, mają odległość równą dwa, itd. Graficzna reprezentacja tej definicji znajduje się na rysunku 6.3. Maksymalną odległość przy której przejście będzie brane pod uwagę definiuje użytkownik w pliku konfiguracyjnym. Z danej struktury może nastąpić co najwyżej jedno przejście, do najbliższej z następujących po niej struktur należącej do jednej z grup tworzących model Markova. Zliczanie przejść w ten sposób ignoruje małe grupy pośrednie które mogły utworzyć się pomiędzy największymi grupami. Pozwala to na zwiększenie liczby zliczonych przejść i przez to ilości danych użytych do budowy modelu. Zliczanie przejść w ten sposób, w przeciwieństwie do stosowania czasu opóźnienia, nie pozwala na modelowanie kinetyki układu. Upraszcza za to uzyskany model przez ignorowanie mniej ważnych stanów i pozwala na zliczenie większej liczby przejść pomiędzy największymi grupami.

Zliczone przejścia są zbierane w macierzy przejść która następnie zapisywana jest do pliku. Jej pierwsza kolumna zawiera numer grupy (struktury będącej jej centrum) z której następuje przejście, natomiast pierwszy wiersz numer grupy do której następuje przejście. Wartości w macierzy oznaczają liczbę przejść pomiędzy danymi grupami. W związku z tym nie jest to macierz przejścia w ścisłym matematycznym sensie,



Rysunek 6.3. Graficzna wizualizacja odległości w łańcuchu obserwacji. Źródło: Własne.

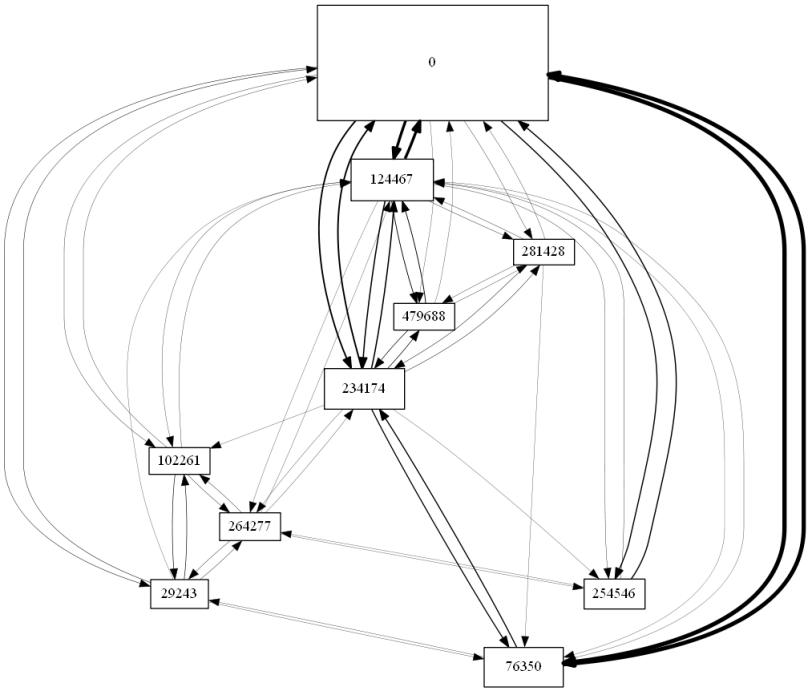
gdzie poszczególne elementy macierzy reprezentują wartości prawdopodobieństwa zajścia przejścia (macierz jest znormalizowana). Liczba przejść jest istotną informacją, potrzebną do pełnej analizy wyników, w związku z czym program nie przeprowadza normalizacji.

Również tę część programu zrównolegliłem przy pomocy OpenMP. Uzyskałem to ponownie poprzez podzielenie zbioru wszystkich struktur pomiędzy wątki programu i przetwarzanie ich w następujący sposób:

1. Sprawdź czy dana struktura należy do grupy biorącej udział w tworzeniu modelu Markova. Jeśli nie zignoruj ją.
2. Sprawdzaj po kolei następujące po niej chronologicznie struktury aż znajdziesz należącą do którejś z grup tworzących model Markova albo przekroczysz maksymalną odległość.
   a) Jeśli przekroczyłeś maksymalna odległość nie rób nic.
   b) Jeśli znalazłeś strukturę należącą do którejś z grup tworzących model Markova dodaj to przejście do macierzy przejść.

6.3.5. Graficzna reprezentacja wyników

Ostatnim zadaniem wykonywanym przez program jest wizualizacja danych. Bazuje ona zarówno na wynikach analizy skupień, z których wykorzystywane są rozmiary największych grup, jak i na modelu Markova, z którego wykorzystywana jest liczba przejść pomiędzy poszczególnymi grupami. Odbywa się ona przez utworzenie pliku wejściowego dla programu Graphviz. W utworzonej przez niego grafice wierzchołki grafu



Rysunek 6.4. Przykładowy graf wygenerowany ze skryptu utworzonego przez pdbclust. Liczby w prostokątach oznaczają numery poszczególnych grup.

oznaczają poszczególne grupy, natomiast krawędzie oznaczają przejścia pomiędzy nimi. Im liczniejsza grupa tym większy reprezentujący ją wierzchołek grafu (symbolizowany większym rozmiarem na grafice). Podobnie grubsze krawędzie pomiędzy wierzchołkami reprezentują większą liczbę przejść pomiędzy danymi dwiema grupami.

Użytkownik mojego programu może na tym etapie wyłączyć wyświetlanie pętli grafu (czyli krawędzi zaczynających i kończących się w tym samym wierzchołku) i włączyć skalowanie wartości, gdyż często pojedyncze wysokie wartości liczby przejść dominują nad pozostałymi, co zmniejsza czytelność grafu. Jest także możliwe zdefiniowanie minimalnej i maksymalnej wyświetlanej wartości liczby przejść. Liczby przejść mniejsze od minimalnej nie są wyświetlane, natomiast większe od maksymalnej są zmniejszane do jej wartości. Ze względu na to, że generowany jest plik wejściowy dla programu Graphviz, a nie gotowa grafika, można dokonać w nim ręcznych zmian, co pozwala dostosować wygenerowaną grafikę do wymagań użytkownika. Przykładowa grafika wygenerowana ze skryptu utworzonego przez pdbclust znajduje się na rysunku 6.4.

6.3.6. Wykonywanie poszczególnych zadań

Powyżej opisałem pełen proces działania programu. Możliwe jest także wykonywanie poszczególnych, głównych zadań pojedynczo, na podstawie plików wynikowych poprzednich zadań. Umożliwia to, na przykład, utworzenie kilku modeli Markova o różnych parametrach na podstawie tych samych wyników analizy skupień. Oszczędza to czas i zasoby, gdyż nie jest wtedy konieczne kilkukrotne wykonywanie tych samych kosztownych obliczeń. Możliwe są cztery różne sposoby wykonywania programu:

1. Pełny przebieg programu. Wykorzystuje on pliki wejściowe pochodzące z symulacji i zapisuje wszystkie rodzaje plików wynikowych.
2. Wykonanie tylko analizy skupień. Wykorzystuje on pliki wejściowe pochodzące z symulacji i zapisuje pliki wynikowe potrzebne do tworzenia modelu Markova i wizualizacji.
3. Utworzenie modelu Markova. Wykorzystuje on pliki wejściowe pochodzące z analizy skupień i zapisuje pliki wynikowe potrzebne do wizualizacji.
4. Utworzenie skryptu do wizualizacji. Wykorzystuje on pliki wejściowe pochodzące z analizy skupień oraz modelu Markova i zapisuje plik wynikowy zawierający skrypt wizualizacji.

6.4. Licencjonowanie programu

Kod źródłowy programu pdbclust postanowiłem rozpowszechniać nieodpłatnie dla użytkowników akademickich i innych niekomercyjnych, na licencji analogicznej do używanej przez program UNRES[82]. Poza wymogiem nie korzystania z programu w celach komercyjnych i właściwego cytowania w publikacjach nie nakłada ona żadnych zobowiązań na użytkowników. Uzyskanie licencji do zastosowań komercyjnych wymaga skontaktowania się z dr Giełdoniem, który koordynował prace nad tym programem.

7. Symulacje dynamiki molekularnej białek

Symulacje dynamiki molekularnej przeprowadziłem na trzech polipeptydach pochodzących z bazy danych RCSB PDB o następujących kodach ID:

— 1BDD (rekombinowana domena B białka A Gronkowca Złocistego)
— 1L2Y (sztucznie zaprojektowane minibiałko posiadające tzw. motyw "*klatki tryptofanowej, ang. Trp-cage*")
— 2MQ8 (sztucznie zaprojektowane białko posiadające tzw. motyw ang. "*ferredoxin-lile fold*")

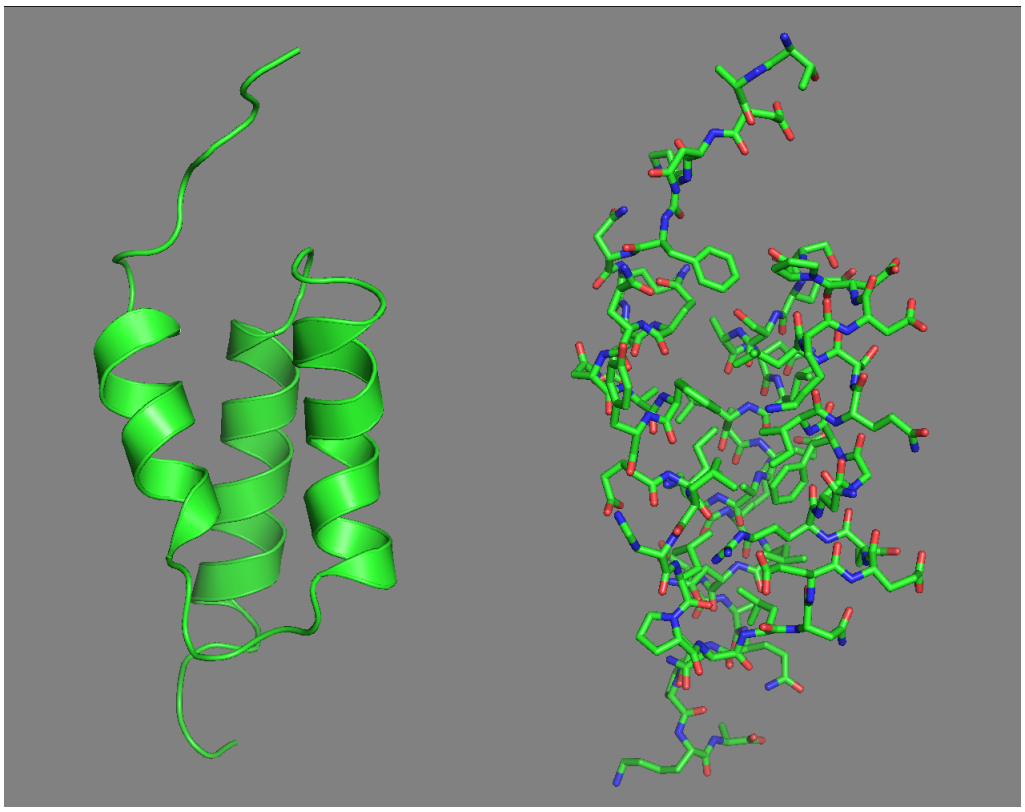
7.1. 1BDD

Struktura o kodzie ID 1BDD obejmuje pojedynczy łańcuch o długości 60 reszt aminokwasowych i następującej sekwencji aminokwasowej:

THR ALA ASP ASN LYS PHE ASN LYS GLU GLN GLN ASN ALA PHE TYR GLU ILE LEU HIS LEU PRO ASN LEU ASN GLU GLU GLN ARG ASN GLY PHE ILE GLN SER LEU LYS ASP ASP PRO SER GLN SER ALA ASN LEU LEU ALA GLU ALA LYS LYS LEU ASN ASP ALA GLN ALA PRO LYS ALA

Jego struktura drugorzędowa składa się z trzech α helis o długościach od 10 do 14 reszt aminokwasowych. Plik PDB z tą strukturą zawiera jeden model, który w trakcie analizy wykorzystuję jako referencyjną strukturę natywną[83]. Dwie reprezentacje graficzne modelu tej struktury znajdują się na rysunku 7.1.

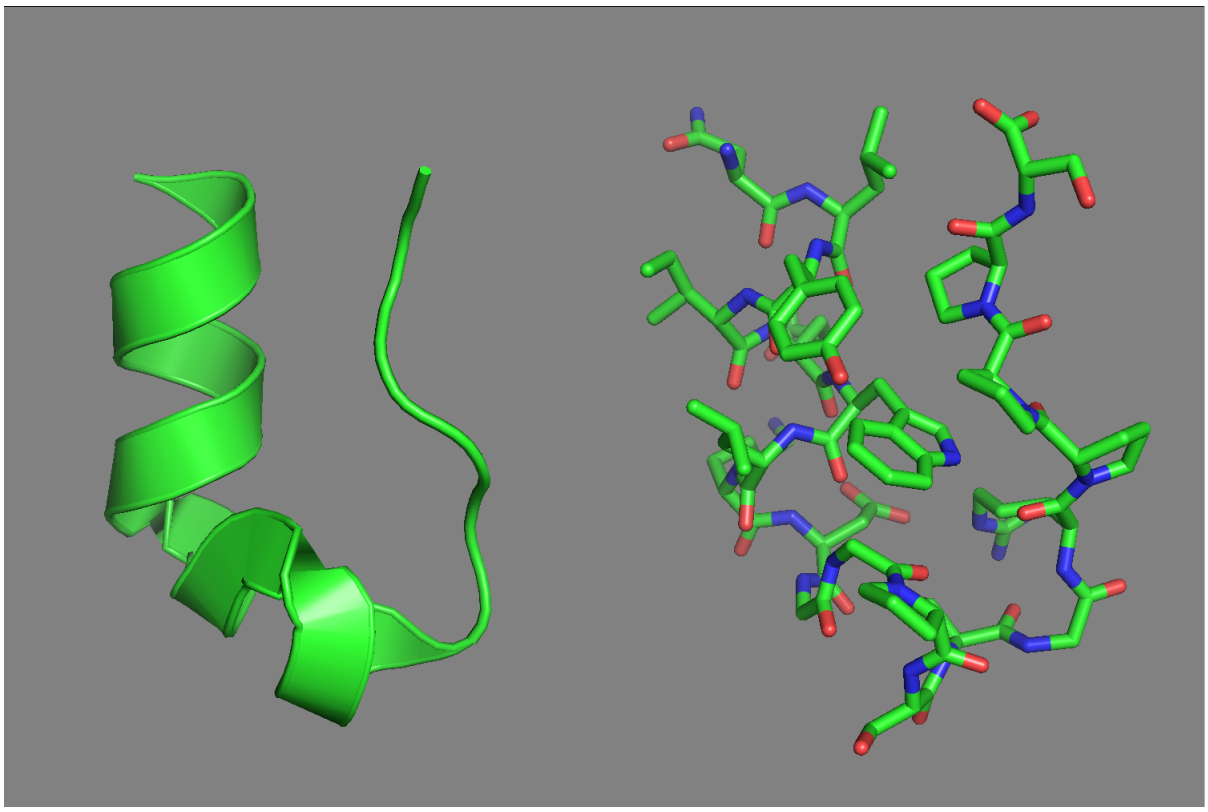
Jest to rekombinowana domena B białka A pochodzącego z Gronkowca Złocistego (*Staphylococcus aureus*). Łączy się ona z fragmentem Fc immunoglobuliny G. Próbkę białka do stwierdzenia struktury uzyskano przez ekspresję syntetycznego genu w bak-



Rysunek 7.1. Dwie reprezentacje graficzne modelu struktury natywnej symulowanej cząsteczki o kodzie ID 1BDD. Źródło: Własne z wykorzystaniem programu PyMOL[84].

terii *E. Coli* hodowanej na minimalnym medium zawierającym jako jedyne źródło azotu jego izotop ^{15}N. Następnie wyizolowano je i oczyszczono. Jego struktura została uzyskana przy pomocy opisanej we wstępie spektroskopii magnetycznego rezonansu jądrowego wykorzystującej efekt Overhausera (NOESY). Autorzy otrzymali 587 ograniczeń na odległości pomiędzy atomami oraz dodatkowe ograniczenia związane z kątami torsyjnymi i wiązaniami wodorowymi. Wyniki zostały następnie ulepszone przy pomocy hybrydowej metody ang. distance geometry-simulated annealing.

Dwie z trzech α helis tej struktury przebiegają antyrównolegle, a ostatnia jest nachylona do nich pod kątem około 30°. Większość hydrofobowych aminokwasów znajduje się wewnątrz struktury składającej się z tych 3 helis tworząc niepolarny rdzeń cząsteczki[85].



Rysunek 7.2. Dwie reprezentacje graficzne modelu struktury natywnej symulowanej cząsteczki o kodzie ID 1L2Y. Źródło: Własne z wykorzystaniem programu PyMOL[84].

7.2. 1L2Y

Struktura o kodzie ID 1L2Y obejmuje pojedynczy łańcuch o długości 20 reszt aminokwasowych i następującej sekwencji aminokwasowej:

ASN LEU TYR ILE GLN TRP LEU LYS ASP GLY GLY PRO SER SER GLY ARG PRO PRO PRO SER

Jego struktura drugorzędowa składa się z dwóch α helis o długościach 6 i 9 reszt aminokwasowych. Plik z tą strukturą zawiera 38 modeli. Autorzy nie wskazali najbardziej reprezentatywnego, więc w trakcie analizy wykorzystuję pierwszy z nich jako referencyjną strukturę natywną[86]. Dwa modele tej struktury znajdują się na rysunku 7.2.

Jest to sztucznie zaprojektowane minibiałko posiadające tzw. motyw "klatki tryptofanowej, ang. Trp-cage". Zostało ono utworzone poprzez skracanie i wprowadzanie mutacji w exendynie-4. Jest ona 39 aminokwasowym peptydem, hormonem znajdują-

cym się w ślinie Helodermy Arizońskiej. Białko to ma zastosowanie lecznicze w terapii cukrzycy. Posiada α helisę długości 20 aminokwasów w środku sekwencji i zwinięty w kłębek struktury trzeciorzędowej C-koniec składający się głównie z aminokwasów aromatycznych: fenyloalaniny, tryptofanu i proliny. Peptyd ten był następnie skracany z obydwu końców w celu ustalenia reszt tworzących ten motyw oraz wprowadzano w nim mutacje punktowe, aby zwiększyć jego stabilność przez uzyskanie korzystniejszych oddziaływań niekowalencyjnych[87].

Wynikiem tej procedury jest opisywany peptyd. W fizjologicznym pH w środowisku wodnym jest on zwinięty w więcej niż 95% i stabilniejszy od minibiałek opublikowanych przed nim. Wiadomo, że tworzy monomery. Jego fałdowanie przebiega kooperatywnie w wyniku oddziaływań hydrofobowych powodujących otaczanie łańcucha bocznego tryptofanu pierścieniami proliny. Jego struktura została uzyskana przy pomocy spektroskopii magnetycznego rezonansu jądrowego wykorzystującej efekt Overhausera (NOESY)[87].

7.3. 2MQ8

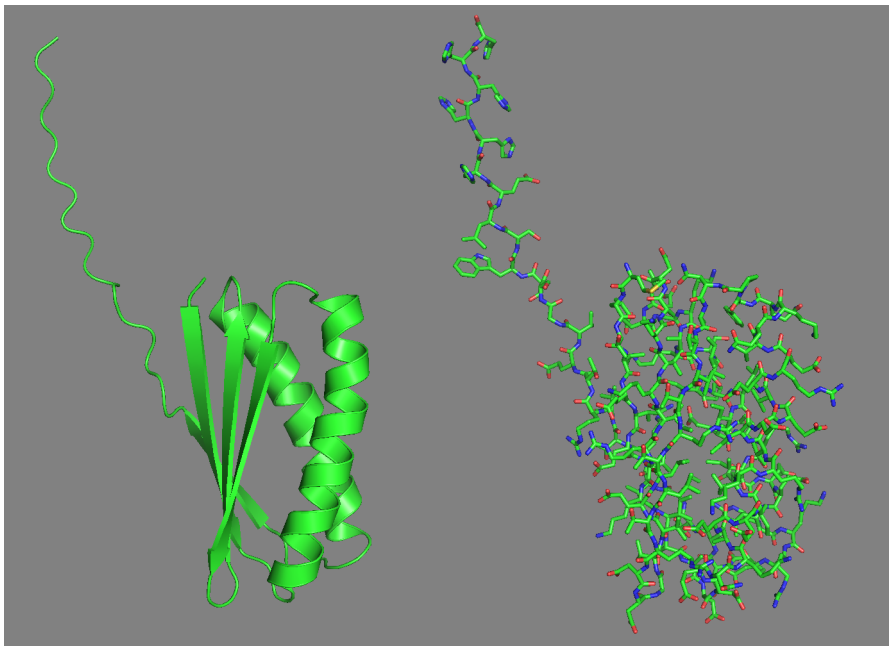
Struktura o kodzie ID 2MQ8 obejmuje pojedynczy łańcuch o długości 112 reszt aminokwasowych i następującej sekwencji:

MET LEU THR VAL GLU VAL GLU VAL LYS ILE THR ALA ASP ASP GLU ASN LYS ALA GLU GLU ILE VAL LYS ARG VAL ILE ASP GLU VAL GLU ARG GLU VAL GLN LYS GLN TYR PRO ASN ALA THR ILE THR ARG THR LEU THR ARG ASP ASP GLY THR VAL GLU LEU ARG ILE LYS VAL LYS ALA ASP THR GLU GLU LYS ALA LYS SER ILE ILE LYS LEU ILE GLU GLU ARG ILE GLU GLU GLU LEU ARG LYS ARG ASP PRO ASN ALA THR ILE THR ARG THR VAL ARG THR GLU VAL GLY SER SER TRP SER LEU GLU HIS HIS HIS HIS HIS HIS

Jego struktura drugorzędowa składa się z dwóch α helis o długościach 22 i 24 reszt aminokwasowych oraz czterech struktur *beta* o długościach od 6 do 10 reszt aminokwasowych. Plik z tą strukturą zawiera 20 modeli. Autorzy wskazali pierwszy z nich

jako najbardziej reprezentatywny, w trakcie analizy wykorzystuję go jako referencyjną strukturę natywną[88]. Dwa modele tej struktury znajdują się na rysunku 7.3.

Jest sztucznie zaprojektowanym białkiem posiadającym w strukturze trzeciorzędowej tzw. motyw "ferredoxin-lile fold". Składa się on z występujących kolejno następujących fragmentów struktury drugorzędowej: $\beta\alpha\beta\beta\alpha\beta$ i kształtem struktury trzeciorzędowej przypomina spinkę do włosów. Przy jego tworzeniu zaprojektowano zarówno regiony struktur α i β jak i pętle oraz zwroty pomiędzy nimi. Dobrano także odpowiednie aminokwasy aby stworzyć hydrofobowy rdzeń i hydrofilową powierzchnię cząsteczki oraz zdestabilizować możliwe alternatywne konformacje. Symulacje zwijania struktur - kandydatów zostały przeprowadzone przy użyciu oprogramowania Rosetta. Został utworzony sztuczny gen kodujący jego sekwencję, który następnie podlegał ekspresji w odpowiednim systemie. Białko to zostało następnie oczyszczone a jego struktura została uzyskana przy pomocy odmiany spektroskopii magnetycznego rezonansu jądrowego HSQC-NMR (ang. *Heteronuclear Single Quantum Coherence NMR*)[89, 90].



Rysunek 7.3. Dwie reprezentacje graficzne modelu struktury natywnej symulowanej cząsteczki o kodzie ID 2MQ8. Źródło: Własne z wykorzystaniem programu PyMOL[84].

7.4. Wykorzystane pakiety oprogramowania do symulacji dynamiki molekularnej

7.4.1. AMBER

AMBER jest zestawem oprogramowania przeznaczonym do przygotowywania, przeprowadzania i analizy symulacji biomolekularnych. Jest to także nazwa zestawu pełnoatomowych empirycznych pól siłowych mechaniki molekularnej wykorzystywanego przez ten pakiet[91, 92]. Zawiera on zestawy parametrów pozwalające na symulacje wszystkich czterech podstawowych typów biomoleku: aminokwasów (białek), kwasów nukleinowych, węglowodanów i fosfolipidów oraz uogólniony zestaw pozwalający na symulację cząsteczek organicznych. Pola sił pakietu AMBER domyślnie używają stałych ładunków na atomach, ale istnieją też ich modyfikacje umożliwiające polaryzowanie[34, 93, 94].

Ten pakiet oprogramowania składa się z dwóch części. Pierwsza, nazwana AMBERTOOLS, jest bezpłatna. Pozwala ona na przeprowadzenie symulacji w podstawowym zakresie, przygotowanie układu i analizę wyników. Druga, współpracująca z pierwszą i nazwana AMBER, jest płatna. Daje ona dostęp do wielu dodatkowych opcji przy przeprowadzaniu symulacji[34]. Do przeprowadzenia symulacji został użyty pakiet AMBER w wersji 17.

Pakiet ten składa się z wielu różnych programów współpracujących ze sobą. Do najważniejszych z nich należą:

pdb4amber

Służy do automatycznej edycji plików PDB tak, aby mogły być wykorzystane przez inne narzędzia pakietu.

LEaP

Służy do tworzenia, edycji i przygotowywania układów do symulacji.

antechamber

Służy do przygotowywania parametrów pola sił dla molekuł organicznych które nie są obecne w standardowych zestawach dostarczanych przez AMBER.

sander

To podstawowy program do przeprowadzania minimalizacji i dynamiki molekularnej. Ma on wiele opcji konfiguracji symulacji, w tym przeprowadzenie dynamiki z wymianą replik. Istnieją dwie jego wersje - sekwencyjna i zrównoleglona przy pomocy biblioteki MPI.

pmemd

To bardziej zaawansowany program do przeprowadzania minimalizacji i dynamiki molekularnej. Jest szybszy i wydajniej zrównoleglony od programu sander. W pakiecie istnieje jego dodatkowa wersja pozwalająca na przeprowadzanie obliczeń przy pomocy kart graficznych poprzez technologię CUDA.

cpptraj

Służy do analizy trajektorii będącej wynikiem symulacji. W pakiecie istnieje jego wersja zrównoleglona przy pomocy biblioteki OpenMP[34, 95].

Pole sił AMBER wraz z towarzyszącymi mu programami jest rozwijane od lat 70 XX wieku. Początkowo odbywało się to pod kierunkiem profesora Petera Kollmana z Uniwersytetu Kalifornijskiego w San Francisco. Obecnie projekt ten jest rozwijany w ramach współpracy pomiędzy wieloma zespołami z różnych uczelni, głównie w USA[91, 96].

7.4.2. UNRES

UNRES (ang. *UNited RESidue*) jest zestawem oprogramowania przeznaczonym do przeprowadzania i analizy symulacji biomolekularnych. Ten pakiet oprogramowania jest dostępny bezpłatnie dla zastosowań akademickich[82]. Jest to także nazwa uproszczonego, gruboziarnistego (ang. *coarse-grained*) pola sił mechaniki molekularnej wykorzystywanego przez ten pakiet. W tym polu sił każdy aminokwas składa się z dwóch elementów: grupy peptydowej i łańcucha bocznego. Umożliwia to nawet kilka tysięcy razy szybsze przeprowadzanie symulacji. Mimo tych uproszczeń symulacje w tym polu sił są dobrym systemem do badania procesu zwijania białek, w tym *ab initio*[97, 98, 99].

Pakiet oprogramowania UNRES składa się z kilku współpracujących ze sobą programów. Należą do nich:

nares

To podstawowy program do przeprowadzania symulacji, w tym minimalizacji, dynamiki molekularnej, dynamiki molekularnej z wymianą replik i multipleksowej dynamiki molekularnej z wymianą replik.

wham

Służy do analizy wyników symulacji przy użyciu ważonych histogramów.

cluster

Służy do analizy skupień danych z symulacji i opisanego powyżej programu wham.

xdrf2pdb

Służy do konwersji wewnętrznego formatu trajektorii pakietu UNRES (pliki cx) do formatu PDB.

Pole sił UNRES i jego oprogramowanie są rozwijane od lat 90 XX wieku przez zespoły profesora Harolda Scheragi z Uniwersytetu Cornella w USA i profesora Adama Liwo z Uniwersytetu Gdańskiego[82].

7.5. Protokół symulacji

Przeprowadziłem w sumie symulacje 64 replik każdego opisanego białka zarówno przy pomocy pakietu AMBER jak i UNRES. Dynamikę molekularną z wymianą replik przeprowadziłem w temperaturach 280K, 290K, 300K, 310K, 320K, 330K, 340K, 360K (razem 8 temperatur). W UNRESie przeprowadziłem jedną MREMD. W każdej temperaturze symulowałem jednocześnie po 8 replik[100]. AMBER obecnie nie pozwala na przeprowadzenie MREMD, w związku z tym przeprowadziłem 8 odrębnych symulacji dynamiki molekularnej, w których symulowałem po jednej replice w każdej temperaturze. Poszczególne symulacje używały różnych liczb pseudolosowych do ustalenia początkowych prędkości. Szczegóły przeprowadzonych symulacji w obydwu pakietach znajdują się poniżej w rozdziałach 7.5.1 i 7.5.2.

Parametry wejściowe w obydwu programach dobrałem tak, aby uzyskać podobne długości czasu symulacji. W ten sam sposób dobrałem też liczbę i odstępy pomiędzy zapisywanymi stanami układu (ang. *snapshot*) w trakcie symulacji. Dzięki temu uzyskałem odpowiadające sobie dane które mogę łatwo analizować i porównywać pomiędzy programami AMBER i UNRES. Użyłem stosunkowo długiego czasu pomiędzy próbami wymiany replik, aby uzyskać długie próbki trajektorii w jednej temperaturze, co zwiększa liczbę możliwych zliczonych przejść w uzyskanym modelu Markova.

7.5.1. AMBER

Symulacje w pakiecie AMBER przeprowadziłem przy pomocy zasobów obliczeniowych Pracowni Symulacji Układów Biomolekularnych MWB UG oraz Wydziału Chemii UG.

Pliki wejściowe *prmtop* i *inpcrd* zawierające rozciągnięte łańcuchy aminokwasowe o właściwych sekwencjach przygotowałem przy użyciu skryptów programu LEaP. Użyłem pola sił ff14SB[101], a promienie Borna ustaliłem zgodnie z użytym modelem ciągłego rozpuszczalnika.

Minimalizację układu przeprowadziłem przez 10 000 cykli, przełączając algorytm minimalizacji z metody najszybszego spadku na metodę gradientów sprzężonych po osiągnięciu przez program 5 000 cykli. Użyłem ciągłego modelu rozpuszczalnika którego autorami są A. Onufriev, D. Bashford i D.A. Case ze zmodyfikowanymi parametrami polepszającymi zgodność modelu z rzeczywistością. Jest to odmiana modelu GBSA (ang. *Generalized Born / Surface Area*)[102, 103]. Stężenie soli dla potrzeb modelu ciągłego ustawiłem na fizjologicznym poziomie 0.154 mol/l[34]. Tego modelu rozpuszczalnika użyłem też podczas symulacji procesu podgrzewania układu oraz we właściwej symulacji dynamiki molekularnej. Program zapisywał informacje o energii układu do plików co 250 cykli symulacji.

Podgrzewanie poszczególnych replik przeprowadziłem od 0K do właściwej dla nich temperatury przez 160 000 kroków symulacji. Program zapisywał współrzędne układu oraz informacje o jego energii co 4 000 kroków. Krok czasowy ustaliłem na 0.002 ps, co w sumie daje 0.32 ns czasu podgrzewania z informacjami o symulacji zapisywanymi

co 8 ps. Do kontroli temperatury użyłem dynamiki Langevina ustalając liczbę kolizji na 5 na pikosekundę. Użyłem algorytmu SHAKE do ograniczania zmian długości wiązań w skład których wchodzą atomy wodoru. Ze względu na jego użycie nie były obliczane siły związane z tymi wiązaniami.

Zbudowane modele symulowałem przy użyciu MD przez 48 000 000 kroków. Próby wymian pomiędzy parami replik były przeprowadzane co 120 000 kroków, razem tych prób było 400. Pozostałe parametry symulacji ustawiłem tak, jak przy podgrzewaniu. W sumie pojedynczą replikę symulowałem przez 96 ns, a próby wymiany replik były przeprowadzane co 0.24 ns.

Z tych symulacji uzyskałem dla każdego białka w sumie 768 000 struktur, które użyłem w analizie skupień i budowie modelu Markova.

7.5.2. UNRES

Symulacje w pakiecie UNRES przeprowadziłem poprzez udostępniony serwer sieciowy wykorzystujący zasoby obliczeniowe Wydziału Chemii UG[104].

Początkowe współrzędne układu zostały utworzone automatycznie przez program po podaniu sekwencji aminokwasowej. Symulację dynamiki molekularnej przeprowadziłem przez 20 000 000 kroków. Program zapisywał do pliku współrzędne układu i informacje o jego energii co 2 000 kroków. Próby wymian pomiędzy parami replik były przeprowadzane co 60 000 kroków symulacji. Pakiet UNRES korzysta ze swojej wewnętrznej jednostki czasu wynoszącej 0.0489 ps. Jako krok symulacji użyłem 0,1 tej wartości. W związku z tym symulacja pojedynczej repliki trwała 97.8 ns, próby wymiany replik były przeprowadzane co około 0.293 ns, a informacje o symulacji zapisywane były co 9.78 ps. Pozostałe opcje programu pozostawiłem z domyślnymi wartościami.

Z tych symulacji uzyskałem dla każdego białka w sumie 640 000 struktur, które użyłem w analizie skupień i budowie modelu Markova.

Część IV

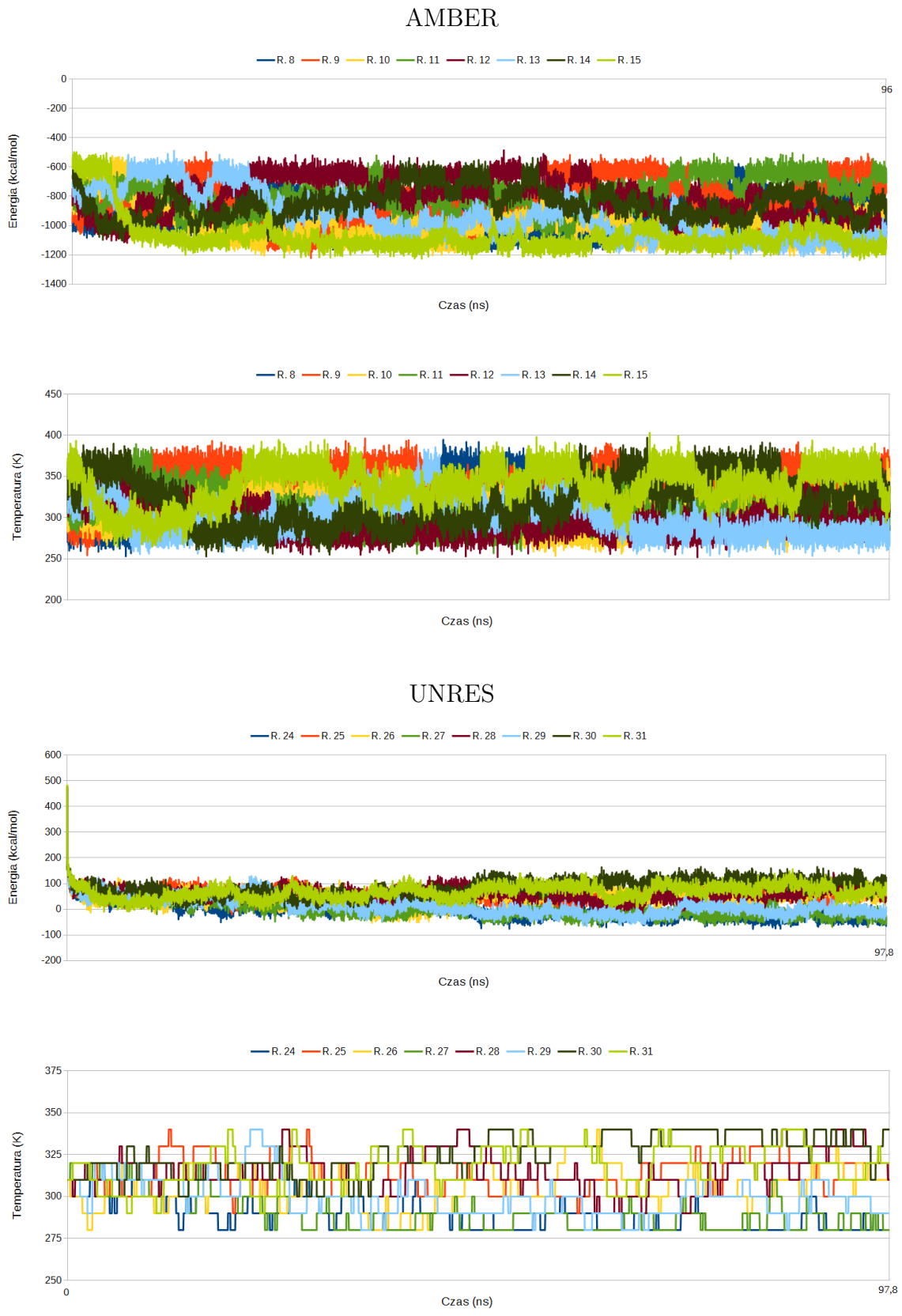
Wyniki

8. Analiza przeprowadzonych symulacji

Do analizy stabilności przeprowadzonych symulacji REMD użyłem następujących parametrów: RMSD względem struktury natywnej, energii całkowitej i współczynnika żyroskopowego (promienia żyracji) układu. W niniejszym rozdziale umieściłem pojedyncze, reprezentatywne wykresy tych danych. Wszystkie wykresy znajdują się w dodatku do niniejszej pracy w rozdziale 12. dla większej czytelności każdą symulację rozbiłem na 8 wykresów ilustrujących po 8 replik.

8.1. Energia całkowita

Na rysunku 8.1 znajdują się przykładowe wykresy zależności energii oraz temperatury układu od czasu dla 8 replik pochodzących z przeprowadzonych symulacji. W symulacji z UNRESa widać początkowy gwałtowny spadek energii całkowitej związany z szybkimi początkowymi zmianami konformacji rozciągniętego łańcucha aminokwasowego do stanów o niższej energii. Przyczynami tego spadku energii są tworzenie oddziaływań łańcuch-łańcuch oraz wiązań wodorowych w strukturze drugorzędowej. W polu sił UNRES wiązania wodorowe nie występują wprost, ze względu na gruboziarnistość modeli białek, ale wynikają z parametrów pola sił. W obydwu przypadkach energia całkowita układu wykazuje duże fluktuacje wartości co pokazuje przechodzenie układu pomiędzy różnymi temperaturami replik. Na wykresach w dodatku widać pojedyncze bardzo wysokie piki energii (np. replika 51 struktury 1L2Y na wykresie 12.4), ale są to pojedyncze konformacje o bardzo wysokiej energii które nie mają wpływu na ogólną poprawność symulacji. Na wykresach temperatury widać różnicę w podejściu do jej regulacji w tych dwóch polach sił. W programie UNRES jest ona bardzo sztywna, podczas gdy AMBER pozwala na zmienność temperatury w znaczącym zakresie. Wynika to z zastosowanego termostatu, który reguluje temperaturę poprzez symulację kolizji



Rysunek 8.1. Przykładowe wykresy zależności energii oraz temperatury od czasu w 8 replikach pochodzących z symulacji w programach AMBER (na górze) i UNRES (na dole).

z ośrodkiem zewnętrznym co 0.2 ps. W związku z tą zmiennością temperatury w polu sił AMBER w trakcie analizy skupień przyjmowałem nominalną temperaturę danej repliki jako temperaturę poszczególnych struktur. Widać też, że zależność energii od temperatury repliki jest bardzo wyraźna w programie UNRES, a w polu sił programu AMBER jej nie widać.

8.2. Współczynnik żyroskopowy

Na rysunku 8.2 znajdują się przykładowe wykresy zależności współczynnika żyroskopowego układu od czasu w 8 replikach pochodzących z przeprowadzonych symulacji. W obu symulacjach widać dość szybki początkowy spadek tego współczynnika związany z początkowymi zmianami konformacji rozciągniętego i napiętego łańcucha aminokwasowego do stanów bardziej zwiniętych. Współczynnik ten jest tym większy im większy jest układ, co jest zgodne z oczekiwaniami. Waha się on w trakcie symulacji i występują jego pojedyncze, większe fluktuacje co pokazuje rozwijanie się i zwijanie struktur do różnych, częściowo stabilnych stanów pośrednich.

Wiele replik osiągnęło podczas symulacji wartości współczynnika żyroskopowego bliskie natywnym (12,1Å dla 1BDD, 7,5Å dla 1L2Y, 18,4Å dla 2MQ8). Repliki te mogły osiągnąć strukturę bliską natywnej albo utknąć w głębokim minimum lokalnym. Istotnym wyjątkiem jest tutaj symulacja struktury 2MQ8 w programie UNRES, w której wiele replik osiągnęło wartości tego współczynnika około 12Å. Wynik ten może sugerować, że w tym polu siłowym pojawiła się inna konformacja, stabilniejsza i korzystniejsza energetycznie od natywnej. Może to sugerować, że parametry pola sił UNRES wciąż wymagają lepszej parametryzacji. Struktura natywna tego białka zawiera na końcu cząsteczki nieposiadający struktury drugorzędowej łańcuch około 15 reszt aminokwasowych. Być może pole sił zdołało go w jakiś sposób zwinąć.

8.3. RMSD względem struktury natywnej

Na rysunku 8.3 znajdują się przykładowe wykresy zależności RMSD układu względem struktury natywnej od czasu w 8 replikach pochodzących z przeprowadzonych sy-

Rysunek 8.2. Przykładowy wykres zależności współczynnika żyroskopowego od czasu w 8 replikach pochodzących z symulacji w programach AMBER (na górze) i UNRES (na dole).

Rysunek 8.3. Przykładowy wykres zależności RMSD względem struktury natywnej od czasu w 8 replikach pochodzących z symulacji w programach AMBER (na górze) i UNRES (na dole).

mulacji. W obu symulacjach widać dość szybki początkowy spadek tego współczynnika związany z początkowymi zmianami konformacji rozciągniętego i napiętego łańcucha aminokwasowego do stanów bardziej zwiniętych stanów. Wartość tego współczynnika waha się w trakcie symulacji co pokazuje rozwijanie się i zwijanie struktur do różnych, częściowo stabilnych stanów o różnym podobieństwie do struktury natywnej. RMSD był obliczany na podstawie tylko węgli α.

Ten parametr przyjmował różne wartości w zależności od symulacji.

— W symulacji struktury o kodzie ID 1BDD w programie AMBER (wykresy pokazałem na znajdującej się w dodatku ilustracji 12.13) przez większą część czasu symulacji znajdował się pomiędzy 10Åa 20Å. Ta wartość wskazuje struktury dość różne od natywnej. Pojedyncze repliki osiągnęły niższe wartości, około 8Åi 5Å, co wskazuje na struktury podobne do natywnej.

— W symulacji struktury o kodzie ID 1BDD w programie UNRES (wykresy pokaza-

łem na znajdującej się w dodatku ilustracji 12.14) przez prawie cały czas trwania symulacji znajdował się pomiędzy 8Åa 15Å. Struktury w pobliżu dolnej granicy tego przedziału mogą być podobne do natywnej.

— W symulacji struktury o kodzie ID 1L2Y w programie AMBER (wykresy pokazałem na znajdującej się w dodatku ilustracji 12.15) znajdował się w zakresie od 2Ådo 9Å, z pikami dochodzącymi do 14Å. Układ ten jest niewielki w związku z czym tak niska wartość tego współczynnika jest możliwa i wskazuje, że wiele replik mogło osiągnąć strukturę podobną do natywnej.

— W symulacji struktury o kodzie ID 1L2Y w programie UNRES (wykresy pokazałem na znajdującej się w dodatku ilustracji 12.16) znajdował się w zakresie od 5Ådo 8Å, z pikami do 14Å. Są to wartości wyraźnie większe niż w symulacji w programie AMBER co wskazuje, że pole sił UNRES nie zdołało zwinąć tej struktury do stanu przypominającego strukturę natywną.

— W symulacji struktury o kodzie ID 2MQ8 w programie AMBER (wykresy pokazałem na znajdującej się w dodatku ilustracji 12.17) znajdował się w zakresie od 20Ådo 40Å, co wskazuje na struktury znacznie różniące się od natywnej.

— W symulacji struktury o kodzie ID 2MQ8 w programie UNRES (wykresy pokazałem na znajdującej się w dodatku ilustracji 12.18) znajdował się w zakresie od 10Ådo 20Å, skupiając się bliżej środka tego przedziału. Wskazuje to na obecność struktur bardziej podobnych do natywnej niż uzyskane w polu sił AMBER, ale wciąż znacząco od niej rożnych.

RMSD umożliwia stwierdzenie na ile podobne do natywnej struktury zostały osiągnięte w symulacji, ale nie pozwala ustalić ścieżki do niej prowadzącej ani liczby stanów pośrednich.

9. Analiza skupień z wykorzystaniem programu pdbclust oraz modelu Markova

9.1. Protokół analizy

W celu przeprowadzenia analizy skupień dla trajektorii uzyskanych z symulacji przekonwertowałem ich pliki wynikowe na pliki multiPDB z wykorzystaniem narzędzi wchodzących w skład pakietów AMBER (cpptraj) i UNRES (xdrf2pdb-m). Aby uzyskać pełnoatomowe modele symulowanych polipeptydów z gruboziarnistych wyników programu UNRES użyłem programu PULCHRA, przeznaczonego do odbudowywania łańcuchów polipeptydowych[105]. Program ten nie jest perfekcyjny i czasem w jego wynikach pojawiają się niefizyczne naprężenia i odległości pomiędzy atomami. W celu ich naprawy uzyskane struktury zminimalizowałem przy użyciu parametrów ff14SB pola sił AMBER w ten sam sposób co startowe struktury symulacji w programie AMBER. Do wizualizacji struktur będących centrami grup użyłem programu PyMOL[84].

Analizę skupień przeprowadziłem używając wszystkich struktur uzyskanych z symulacji. Wykorzystałem struktury natywne pochodzące bezpośrednio z plików PDB. Użyłem różnych promieni grup dla poszczególnych struktur. W pierwszej próbie przyjąłem promień (w Å) w przybliżeniu równy pierwiastkowi kwadratowemu z liczby reszt aminokwasowych w strukturze. Następnie próbowałem go zmniejszać i zwiększać tak, aby uzyskać rozsądnie spopulowane grupy. W ten sposób ustaliłem następujące promienie grup dla poszczególnych struktur:

ID Struktury	Liczba reszt	Pierwiastek kwadratowy	Promień grupy
1BDD	60	$\sim 7,75$	9Å
1L2Y	20	$\sim 4,47$	5Å
2MQ8	112	$\sim 10,58$	13Å

Łańcuch Markova zbudowałem używając 5 jako maksymalnej chronologicznej odle-

głości pomiędzy strukturami należącymi do największych grup (patrz 6.3.4). Oznacza to 40 ps dla symulacji w programie AMBER i 48.98 ps w programie UNRES. dla każdej symulacji przygotowałem model Markova ze struktur o temperaturze 310K, która jest najbliższa temperaturze fizjologicznej spośród wykorzystanych w symulacji. W niektórych przypadkach (kiedy uznałem, że to może dostarczyć dodatkowych informacji) przygotowałem drugi model złożony ze wszystkich struktur. Do utworzenia macierzy i grafu przejścia użyłem 10 największych grup. W uzyskanych grafach umieściłem miniatury struktur będących centrami grup razem z informacjami o ich RMSD względem struktury natywnej (w indeksie górnym), współczynniku żyroskopowym (w indeksie dolnym) oraz kolejnym numerem (kolejność zgodnie z rosnącym RMSD względem struktury natywnej). Te stany, które w trakcie symulacji były osiągane przez poszczególne repliki jako pierwsze spośród stanów modelu Markova są poprzedzane literą S w indeksie dolnym. Razem te dane mają format ${}_{S}Nr_{Rgyr}^{RMSD}$. Struktury natywne oznaczyłem literą N.

Jednym ze sposobów weryfikacji poprawności uzyskanego modelu Markova jest analiza symetryczności matrycy przejścia[57]. W wyidealizowanym przypadku (to znaczy dla nieskończenie długiej symulacji rozpoczętej w stanie równowagi) powinna być ona idealnie symetryczna. Ze względu na ograniczoną długość symulacji oraz rozpoczęcie ich z rozprostowanej, rozwiniętej konformacji możemy jedynie oczekiwać, że macierz przejścia będzie w przybliżeniu symetryczna. Znaczące odstępstwa powinny zostać sprawdzone gdyż mogą wskazywać na niepoprawną konstrukcję modelu. Uzyskany graf przejścia powinien być spójny, czyli powinna na nim istnieć ścieżka pozwalająca na przejście pomiędzy dwoma dowolnymi wierzchołkami (stanami)[57]. W praktyce mogą zaistnieć sytuacje, w których obserwujemy znacząco więcej przejść w jedną stronę. Ich przykładem może być rozpoczęcie symulacji, tak jak ja to zrobiłem, ze stanu kompletnie rozciągniętego łańcucha polipeptydowego, który jest niekorzystny energetycznie, mało prawdopodobny i raczej nie pojawia się normalnie w przebiegu symulacji. Innym przykładem może być istnienie w danym układzie stanu znajdującego się na dnie bardzo stromej, wąskiej doliny energetycznej z której układ nie jest w stanie wyjść. Taki stan może reprezentować nieprawidłowo zwinięte, ale stabilne białko, podobne do wspo-

mnianych w rozdziale 1.2 nieprawidłowo zwiniętych białek. Symetryczność macierzy (to znaczy taka sama liczba przejść w obie strony pomiędzy parami stanów) nie oznacza, że w każdym stanie znajduje się tyle samo struktur. Jeden ze stanów może być stabilniejszy od drugiego i białko może przebywać w nim dłużej zanim przejdzie do innego stanu[13].

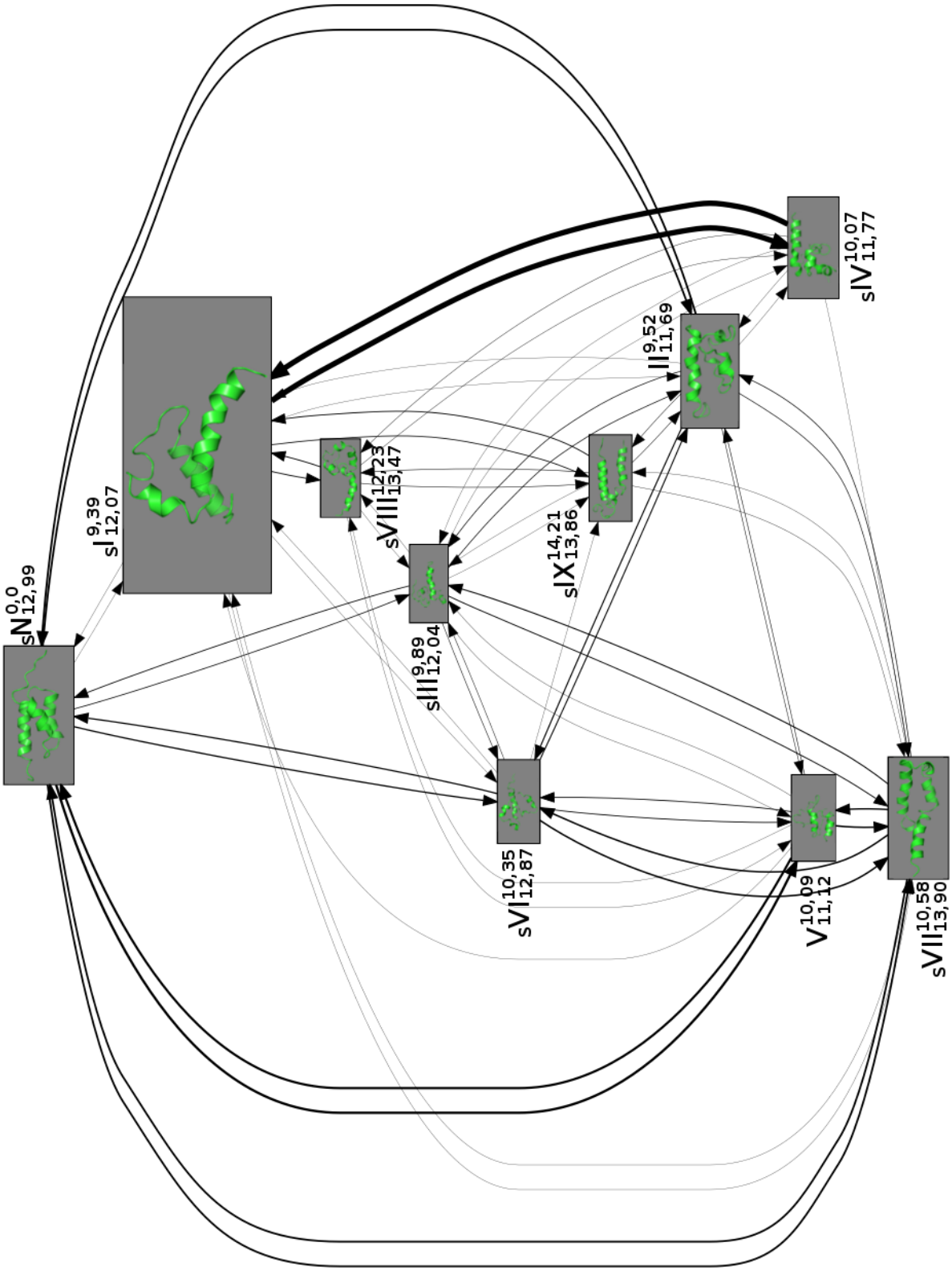
9.2. UNRES

9.2.1. 1BDD

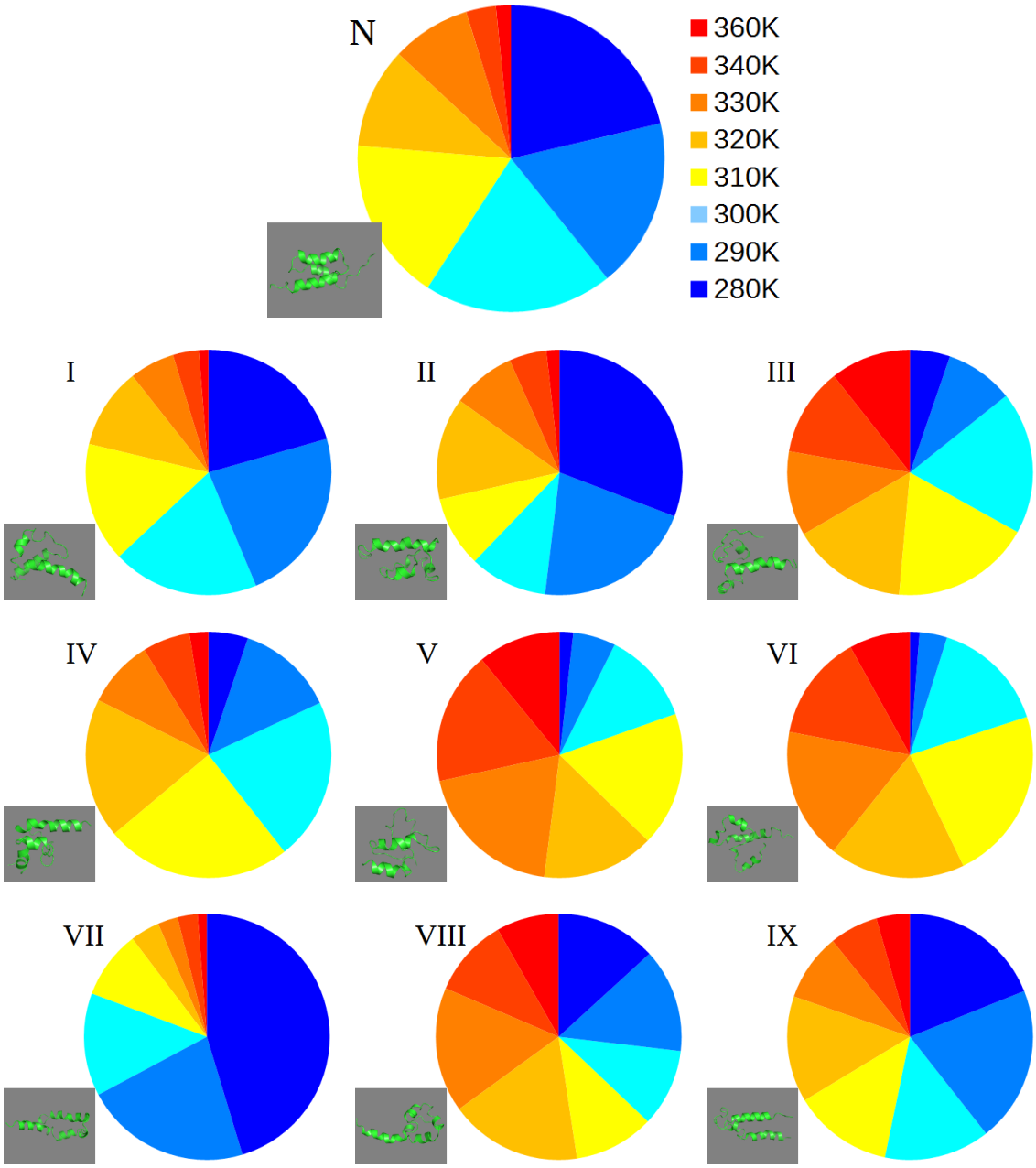
Na rysunku 9.1 znajduje się graf przejścia dla symulacji białka o ID 1BDD w programie UNRES. Do jego przygotowania wykorzystałem tylko przejścia pomiędzy strukturami w temperaturze 310K. Jego macierz przejścia znajduje się poniżej. Jest ona bliska symetrycznej, jedynie kilka wartości różni się o więcej niż 10% względem ich symetrycznego odpowiednika. Wskazuje to na prawidłowe przygotowanie modelu Markova. Na rysunku 9.2 znajdują się proporcje populacji struktur w poszczególnych temperaturach w grupach wykorzystanych do stworzenia Modelu Markova. Na rysunku 9.3 znajdują się modele struktury natywnej oraz centrów największych uzyskanych grup. Tabela 9.4 przedstawia liczbę replik które osiągnęły poszczególne stany modelu Markova jako pierwsze w czasie tej symulacji.

Z\Do	**N**	**I**	**VII**	**II**	**IV**	**V**	**IX**	**VI**	**VIII**	**III**
N	**7477**	1	160	162	0	217	0	106	0	53
I	1	**21648**	1	1	518	0	58	1	66	0
VII	160	1	**2734**	46	0	122	6	132	0	81
II	165	1	40	**2635**	2	26	21	90	0	45
IV	0	517	1	1	**5684**	0	0	0	15	1
V	201	1	130	30	0	**2498**	0	60	1	6
IX	0	65	9	23	0	0	**1876**	0	31	1
VI	102	1	133	82	0	59	2	**2913**	0	23
VIII	0	67	0	0	12	2	27	0	**1032**	1
III	61	0	69	46	1	5	1	28	2	**1825**

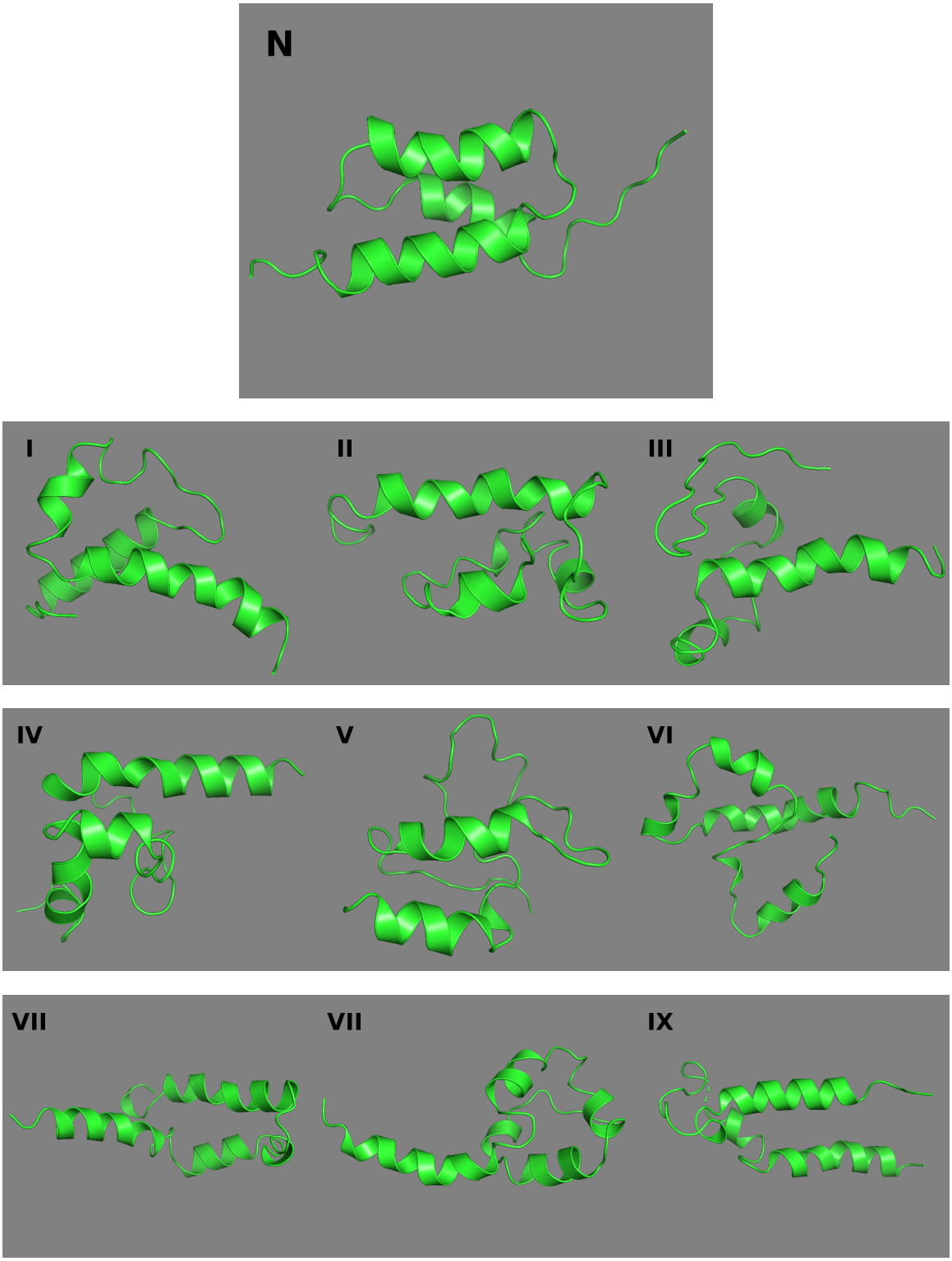
W tej symulacji grupa utworzona przez strukturę natywną (N) jest drugą największą grupą i obejmuje około 7,5% wszystkich struktur użytych w analizie skupień.



Rysunek 9.1. Graf wynikowy dla białka o ID 1BDD symulowanego w programie UNRES dla przejść w temperaturze 310K. Obok każdego numeru struktury w indeksie górnym znajduje się jej RMSD względem struktury natywnej, a w indeksie dolnym jej współczynnik żyroskopowy. Stany osiągane przez repliki jako pierwsze są poprzedzone literą S w indeksie dolnym, a liczba tych replik znajduje się w tabeli 9.4. Powiększone modele struktur znajdują się na rysunku 9.3.



Rysunek 9.2. Wykresy kołowe populacji największych grup w poszczególnych temperaturach dla symulacji struktury o ID 1BDD przeprowadzonej w pakiecie UNRES. W lewym dolnym rogu każdego wykresu znajduje się model struktury której dany wykres dotyczy. Powiększone modele struktur znajdują się na rysunku 9.3.



Rysunek 9.3. Modele struktury natywnej oraz struktur centralnych największych grup dla symulacji struktury o ID 1BDD przeprowadzonej w pakiecie UNRES. Pochodzą one z grafu na rysunku 9.1.

Stan	N	I	II	III	IV	V	VI	VII	VIII	IX
Liczba replik	11	11	0	3	11	0	4	10	7	7

Rysunek 9.4. Tabela przedstawiająca pierwsze osiągnięte przez poszczególne repliki stany wchodzące w skład modelu Markova pochodzące z symulacji struktury o ID 1BDD przeprowadzonej w pakiecie UNRES.

Zawiera ona 3 helisy α, które dalej będą oznaczane α_1, α_2, α_3. Obserwujemy też kilka stanów pośrednich do i z których ta struktura przechodzi. Około 75% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że dolina energetyczna związana z tym stanem jest głęboka i układ nie mógł z niej wyjść w niskich temperaturach. Oznaczałoby to dużą stabilność tej struktury, co jest oczekiwane dla struktury natywnej.

Największym stanem do i z którego przechodzi struktura natywna jest stan I, w którym znajduje się około 22,5% wszystkich struktur. Jest to jednocześnie najbardziej spopulowany stan w całym modelu. Pod względem struktury drugorzędowej konformacja będąca centrum tej grupy ma 3 helisy α, tak jak struktura natywna. α_1 jest dłuższa od odpowiadającej jej helisie w strukturze natywnej, α_2 jest krótsza a α_3 identyczna. W strukturze trzeciorzędowej widać, że α_1 ułożona jest pod innym kątem względem pozostałych niż w strukturze natywnej, co może wynikać z jej większej długości. Znajduje się też ona po przeciwnej stronie pozostałych dwóch helis względem struktury natywnej, co tłumaczy dlaczego przejścia pomiędzy nią i strukturą natywną były rzadkie. Trzy wspomniane helisy α i fragmenty pętli tworzą wyraźnie hydrofobowy rdzeń cząsteczki, co wpływa na jej stabilizację. Jest to najbardziej podobna (pod względem RMSD struktury centralnej) do natywnej grupa powstała w tej symulacji. Rozkład temperatur jest w niej podobny do struktury natywnej.

Blisko związany ze stanem I jest stan IV, w którym znajduje się około 4% struktur. Również w strukturze będącej jego centrum widać 3 helisy α. W porównaniu ze stanem I α_2 znajduje się dalej w sekwencji (reszty aminokwasowe 38-43 zamiast 22-28), a pozostałe dwie są podobne. Aminokwasy hydrofobowe α_1 i α_3 razem z fragmentami pętli tworzą rdzeń cząsteczki, ale hydrofobowe reszty α_2 zwrócone są w kierunku roztworu, co może obniżać stabilność tej konformacji. Jej struktura trzeciorzędowa jest podobna, ale bardziej rozwinięta od struktury centralnej stanu I. Przejścia pomiędzy tymi dwoma

stanami były w symulacji bardzo częste. Jego rozkład temperatur sugeruje nieco mniejszą stabilność ze względu na większy udział wyższych temperatur w populacji (około 64% struktur jest w temperaturze fizjologicznej i niższej). Wskazuje to na jego niższą stabilność i sugeruje, że stan I rozwija się w stan IV w wyższych temperaturach.

Kolejnym istotnym stanem pośrednim jest stan VII, w którym znajduje się około 6% struktur. Reprezentuje on trzecią najliczniejszą grupę uzyskaną w analizie skupień. Jego struktura centralna zawiera 5 helis α. Pierwsza z nich odpowiada w przybliżeniu α_1, ale jest mocno przesunięta w kierunku N-końca cząsteczki. Druga i trzecia odpowiadają bardzo dobrze α_2 z niedużą przerwą w środku. Czwarta odpowiada α_3, a piąta nie ma swojego odpowiednika w strukturze natywnej. Istotna jest struktura trzeciorzędowa tego stanu. Przypomina strukturę natywną, a jedyną różnicą jest to, że pierwsza helisa jest odchylona o około 120° od reszty cząsteczki. Druga, trzecia i czwarta helisa, razem z fragmentami pętli, tworzą hydrofobowy rdzeń cząsteczki. Jest on częściowo otwarty do środowiska zewnętrznego ze względu na wspomniane odchylenie pierwszej helisy. Przejścia pomiędzy tą strukturą a natywną są częste i można zasugerować prosty mechanizm tego procesu polegający na zmianach wartości kąta pomiędzy pierwszą helisą a resztą cząsteczki. Rozkład temperatur tego stanu sugeruje bardzo dużą stabilność - około 90% struktur w jego grupie znajdowało się w temperaturze fizjologicznej lub niższej. Oznacza to, że dolina energetyczna związana z tym stanem jest głęboka i układ nie mógł z niej wyjść w niskich temperaturach.

Następnym stanem, do i z którego często przechodziła struktura natywna jest stan II, w którym znajduje się około 5% struktur. Struktura go reprezentująca posiada jedną długą helisę α, bardzo blisko odpowiadającą α_3 i kilka krótkich helis które słabo odpowiadają fragmentom struktury natywnej. Mimo to jej RMSD względem struktury natywnej jest niski i wynosi 9,5Å. Można zauważyć, że struktura trzeciorzędowa fragmentu białka tworzącego pierwszą i drugą helisę przypomina bardziej rozwiniętą strukturę natywną, a trzecia helisa jest względem tego fragmentu obrócona o około 90°. Helisy oraz wiele fragmentów pętli tworzy wyraźny, hydrofobowy rdzeń cząsteczki, co stabilizuje tę strukturę. Konformacja struktury centralnej tego stanu sugeruje przechodzenie do i ze struktury natywnej poprzez szybkie zwijanie i rozwijanie pierwszej

i drugiej helisy α. Rozkład temperatur struktur w tym stanie sugeruje stabilność minimalnie niższą niż struktury natywnej.

Kolejnym stanem pośrednim jest stan V, w którym znajdowało się około 2,5% struktur. Jego struktura centralna zawiera helisy α dość dobrze odpowiadające α_1 i α_3, przy czym pierwsza helisa jest ponownie wydłużona i składa się z dwóch fragmentów rozdzielonych krótka pętlą. Układają się one blisko siebie, ale jedna z nich jest obrócona o około 180° względem drugiej wobec ich położenia w strukturze natywnej (biegną równolegle zamiast antyrównolegle). Fragment białka tworzący α_2 i pętle łączące ją z pierwszą i trzecią są rozwinięte i nie tworzą wyraźnej struktury. Konformacja ta posiada wyraźny rdzeń hydrofobowy, którego jedna strona jest w wyraźnym kontakcie ze środowiskiem zewnętrznym. Wpływa to negatywnie na jej stabilność i sprawia, że możliwe są jej oddziaływania hydrofobowe z innymi molekułami. Potencjalnie może przechodzić w strukturę bliższą natywnej przez obrót jednej z helis i zwinięcie środkowego fragmentu cząsteczki. Temperatury struktur w tym stanie sugerują niską stabilność (38% struktur jest w temperaturze fizjologicznej i niższej). Jego minimum energetyczne jest prawdopodobnie płytkie. Może on być często, ale na krótko osiąganym stanem pośrednim.

Stany VI, VIII i IX przypominają pod pewnymi względami stan V i znajdowało się w nich odpowiednio około 2,4%, 1,8% i 2,5% struktur. Ich struktury centralne, podobnie jak stan V, posiadają helisy α dość dobrze odpowiadające α_1 i α_3, ułożone względem siebie w podobny sposób ale bardziej oddalone na różne sposoby i rozwinięty środkowy fragment białka. Wyróżnia się z nich stan VIII w którym wspomniane helisy α są znacznie przesunięte względem siebie wzdłuż wspólnej osi. Podobnie struktury te posiadają wyraźne rdzenie hydrofobowe, będące częściowo w kontakcie ze środowiskiem zewnętrznym. Mają także podobne rozkłady temperatur, jedynie stan IX jest nieco bardziej stabilny.

Ostatnim stanem w modelu jest stan III, w którym znajdowało się około 1,8% struktur. Jego struktura centralna posiada zwiniętą α_1, wydłużoną względem struktury natywnej, i dwa krótkie fragmenty helisy dalej w cząsteczce. W strukturze trzeciorzędowej widać fragmenty biegnące równolegle względem siebie, ale nie odpowiadają one heli-

som znajdującym się w strukturze natywnej. Konformacja ta posiada dwa oddzielone od siebie fragmenty rdzenia hydrofobowego, jeden około 3 razy mniejszy od drugiego. Oba są częściowo w kontakcie ze środowiskiem zewnętrznym, co destabilizuje ją. Rozkład temperatur w tym stanie sugeruje stabilność podobną do trzech stanów opisanych w poprzednim akapicie.

Tabela pierwszych osiąganych stanów (9.4) pokazuje, że większość z nich była osiągana jako pierwsze przez więcej niż jedną replikę. Jest to istotne zwłaszcza dla struktury natywnej, którą osiągnęło jako pierwszą 11 replik. Wskazuje to na istnienie innych ścieżek zwijania, których nie obejmuje niniejszy model Markova. Mogłyby one zostać przeanalizowane w toku dalszych badań. Pozostałe stany pokazują, że istnieje rozległa sieć dodatkowych, mniej istotnych stanów pośrednich które doprowadzają strukturę do zwinięcia w jeden ze stanów modelu Markova.

Uzyskany model sieci stanów struktury białka pokazuje zarówno różne możliwe ścieżki zwijania, jak i częściowe rozwijanie struktury natywnej w sieć form pośrednich, zgodnie z przyjętą hipotezą badawczą. Pozwala on, opierając się na powyższych obserwacjach, formułować ogólne hipotezy na temat przebiegu zwijania tego białka w polu sił UNRES. α_1 i α_3 obecne są w prawie wszystkich stanach utworzonego modelu Markova, można więc założyć, że tworzą się najszybciej i są najbardziej stabilnymi elementami jego struktury. α_2 jest mniej stabilna i tworzy się w późniejszym czasie, kiedy dwie pozostałe helisy są już utworzone i zbliżyły się do siebie w przestrzeni. Wskazują na to przejścia pomiędzy strukturą natywną, a stanami II, V i VI, w których α_1 i α_3 są blisko siebie w przestrzeni. W wielu opisanych stanach pośrednich α_1 i α_3 są ułożone względem siebie równolegle zamiast antyrównolegle. Stany VII i VIII pokazują, że te helisy mogą przesuwać się względem siebie. Sugeruje to, że w czasie zwijania α_2 pierwsza helisa jest przez nią przesuwana, a następnie obraca się zbliżając do pozostałych dwóch helis. Udział w tym ostatnim kroku mogą mieć oddziaływania hydrofobowe, na co wskazuje częściowe wystawienie takich reszt w kierunku środowiska w stanie VII. Ten proces mógłby reprezentować jedną z możliwych ścieżek zwijania tego białka. Stan I może reprezentować inny mechanizm prowadzący do struktury natywnej, w którym α_1 i α_3 obracają się względem siebie, co pozwala na powstanie helisy α_2. Osiągnięcie dopiero

stanu natywnego jako pierwszego w wielu replikach pokazuje, że istnieją ścieżki zwijania które nie znalazły się w niniejszym modelu. Mogłyby one zostać przeanalizowane w toku dalszych badań.

Pole sił UNRES zwinęło to białko w strukturę bliską natywnej. Jedną z cech wielu opisanych struktur pośrednich jest wydłużona pierwsza helisa α, co może sugerować, że w tym polu sił helisy te tworzą się zbyt łatwo lub są zbyt stabilne. Niektóre z opisanych struktur wykazują niższy współczynnik żyroskopowy od struktury natywnej. Może to wynikać z bardziej zwartego ułożenia obydwu końców łańcucha polipeptydowego, które to w strukturze natywnej biegną na zewnątrz struktury. Innym powodem mogły być zmiany wprowadzone przez proces odbudowywania struktur z gruboziarnistych do pełnoatomowych.

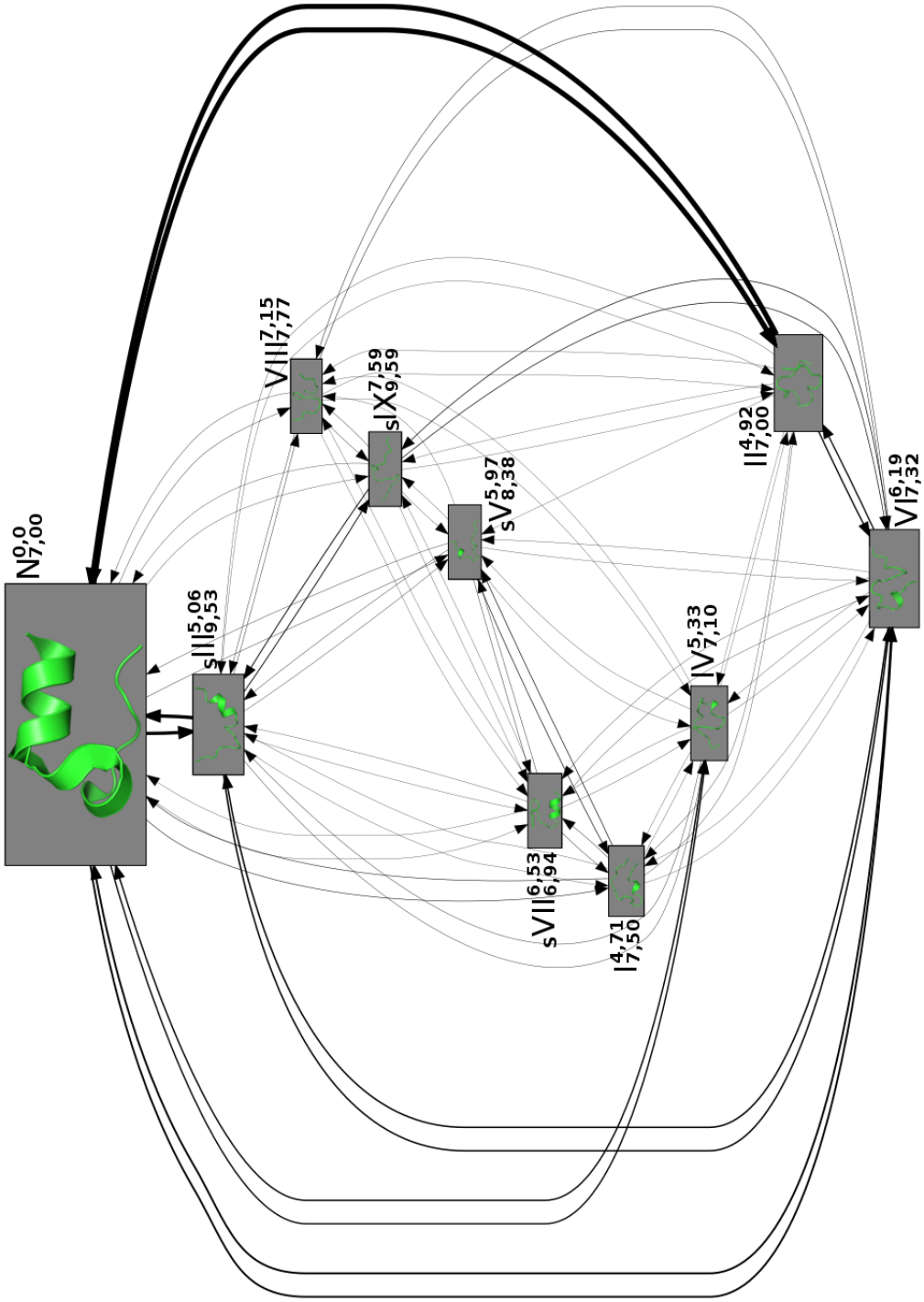
9.2.2. 1L2Y

Na rysunku 9.5 znajduje się graf przejścia dla symulacji białka o ID 1L2Y w programie UNRES. Do jego przygotowania wykorzystałem tylko przejścia pomiędzy strukturami w temperaturze 310K. Jego macierz przejścia znajduje się poniżej. Jest ona bliska symetrycznej, jedynie kilka niskich wartości różni się o znaczący procent od ich symetrycznego odpowiednika. Wskazuje to na prawidłowe przygotowanie modelu Markova. Na rysunku 9.6 znajdują się proporcje populacji struktur w poszczególnych temperaturach w grupach wykorzystanych do stworzenia Modelu Markova. Na rysunku 9.7 znajdują się modele struktury natywnej oraz centrów największych uzyskanych grup. Tabela 9.8 przedstawia liczbę replik które osiągnęły poszczególne stany modelu Markova jako pierwsze w czasie tej symulacji.

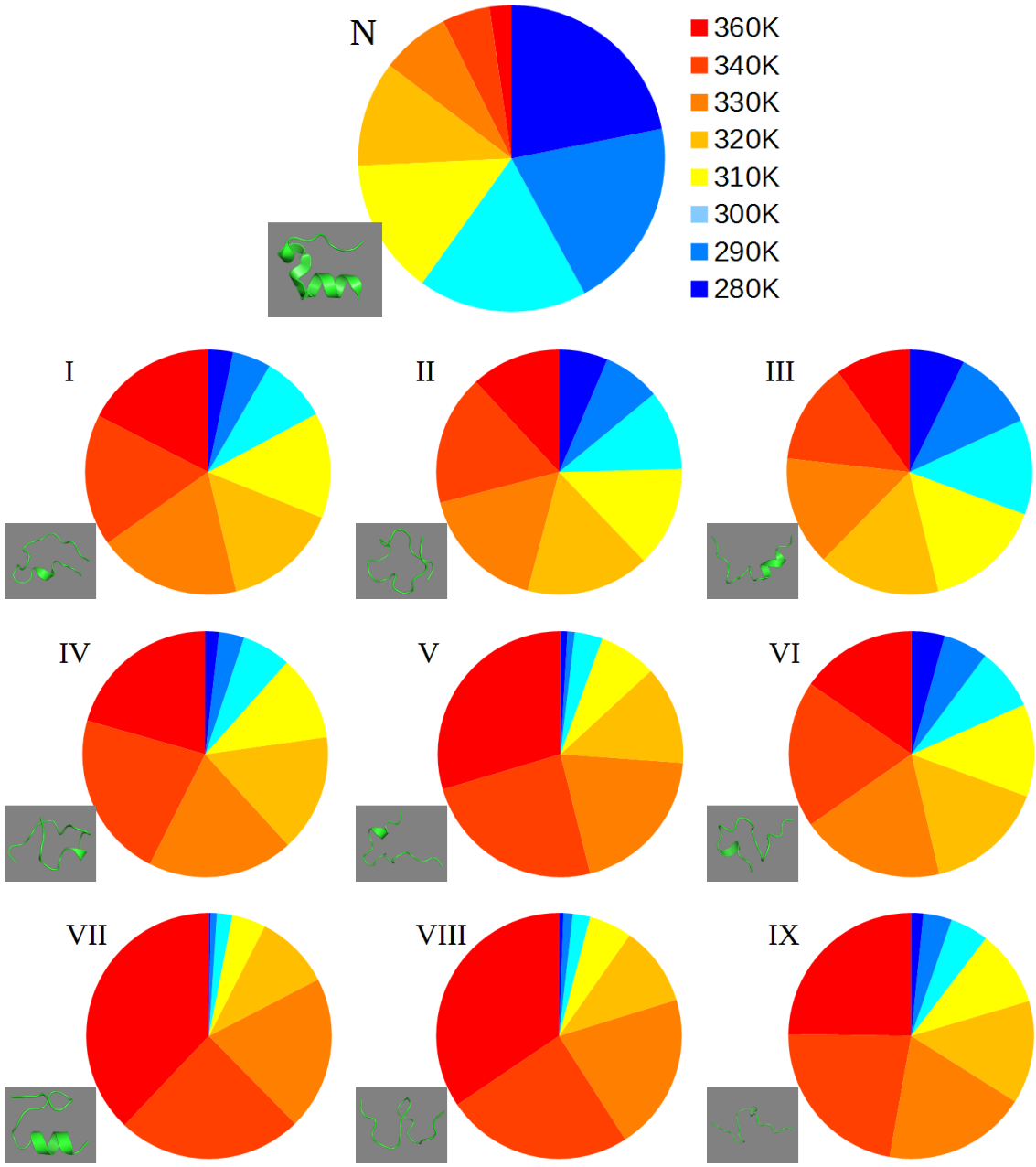
Z\Do	N	III	VI	II	IV	I	VII	IX	V	VIII
N	**36797**	1763	933	2859	661	143	9	26	66	56
III	1800	**6184**	728	14	60	10	12	375	43	78
VI	944	737	**4146**	598	11	4	22	251	13	116
II	2809	27	633	**3677**	1	36	0	5	2	11
IV	677	63	10	2	**1821**	8	12	0	10	1
I	147	11	4	32	11	**2195**	73	0	227	0
VII	6	15	28	0	18	71	**442**	8	76	5
IX	29	393	239	7	0	0	8	**520**	2	47
V	55	52	10	0	6	226	82	3	**380**	1
VIII	43	69	121	11	4	0	6	50	0	**150**

W tej symulacji grupa utworzona przez strukturę natywną (N) jest wyraźnie największą grupą i obejmuje około 48% wszystkich struktur użytych w analizie skupień. Struktura natywna zawiera 2 helisy α, które dalej będą oznaczane α_1 i α_2. Obserwujemy też cztery stany pośrednie do i z których ta struktura przechodzi najczęściej (II, III, IV, VI). Dwa z nich (II, III) zauważalnie wyróżniają się pod względem liczby przejść. Około 75% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest bardzo stabilna, a dolina energetyczna związana z tym stanem jest głęboka i układ nie mógł z niej łatwo wyjść w niskich temperaturach. Cechą charakterystyczną struktury natywnej tego białka jest C-koniec składający się głównie z aminokwasów aromatycznych, zwinięty w kłębek struktury trzeciorzędowej.

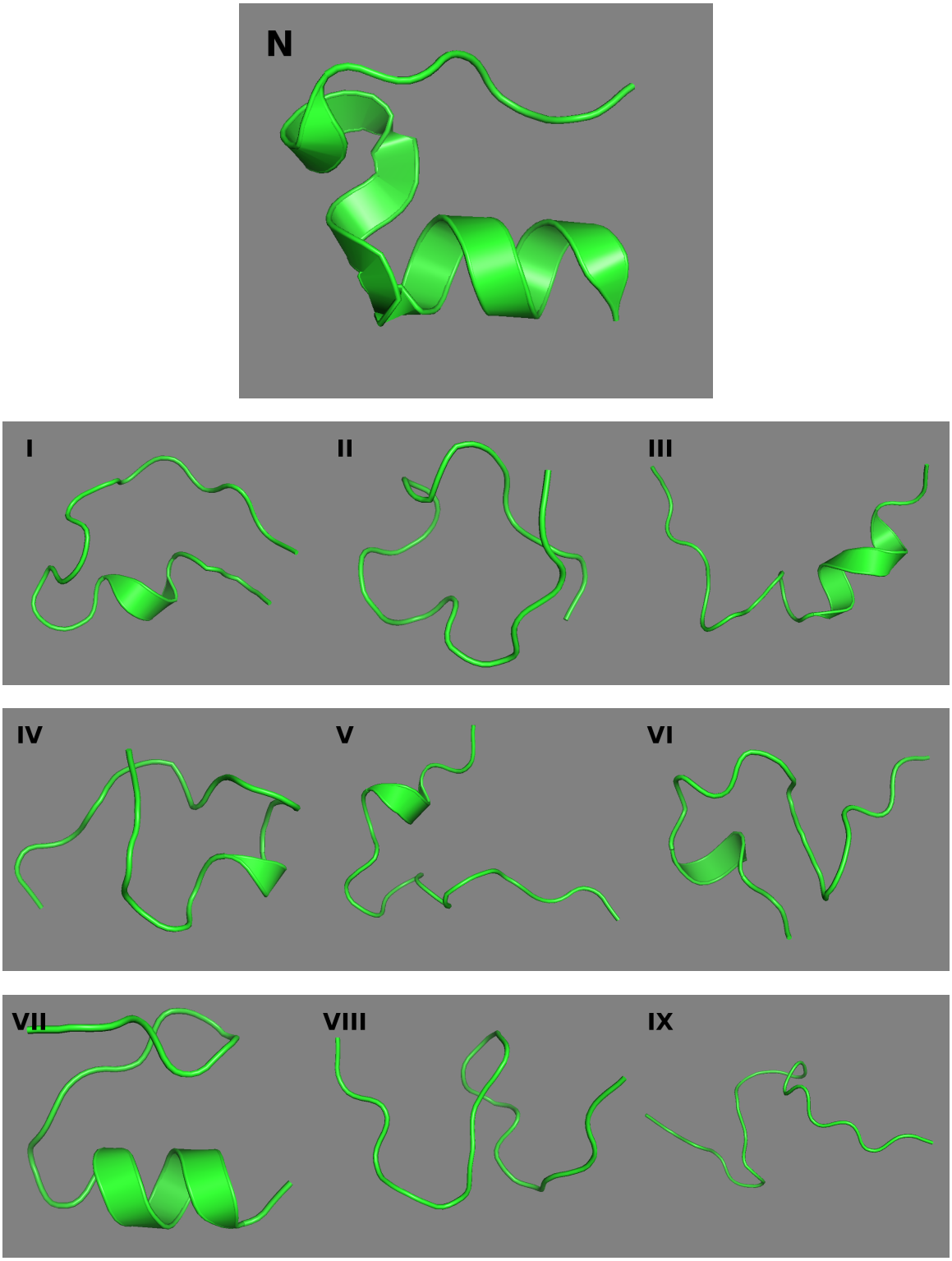
Pierwszym stanem do i z którego często przechodzi struktura natywna jest stan II, w którym znajduje się około 8,6% struktur. Struktura będąca jego centrum nie posiada zwiniętych helis α, ale końcowe 2/3 jej długości przypomina kształtem strukturę natywną. Jej początek jest zauważalnie skręcony co sprawia że jako całość struktura ta przypomina pętlę, podczas gdy struktura natywna przypomina trzy krawędzie równoległoboku. Sugeruje to, że w zwijaniu tego białka duże znaczenie mają oddziaływania pomiędzy odległymi elementami łańcucha polipeptydowego, a tworzenie struktur drugorzędowych ma mniejsze znaczenie. W tej konformacji wspomniana struktura C-końca pojawia się częściowo. Tryptofan istotny dla niej znajduje się po przeciwnej stronie łańcucha względem pozostałych reszt. Stan ten jest mało stabilny, jedynie około 38%



Rysunek 9.5. Graf wynikowy dla białka o ID 1L2Y symulowanego w programie UNRES dla przejść w temperaturze 310K. Obok każdego numeru struktury w indeksie górnym znajduje się jej RMSD względem struktury natywnej, a w indeksie dolnym jej współczynnik żyroskopowy. Stany osiągane przez repliki jako pierwsze są poprzedzone literą S w indeksie dolnym, a liczba tych replik znajduje się w tabeli 9.8. Powiększone modele struktur znajdują się na rysunku 9.7.



Rysunek 9.6. Wykresy kołowe populacji największych grup w poszczególnych temperaturach dla symulacji struktury o ID 1L2Y przeprowadzonej w pakiecie UNRES. W lewym dolnym rogu każdego wykresu znajduje się model struktury której dany wykres dotyczy. Powiększone modele struktur znajdują się na rysunku 9.7.



Rysunek 9.7. Modele struktury natywnej oraz struktur centralnych największych grup dla symulacji struktury o ID 1L2Y przeprowadzonej w pakiecie UNRES. Pochodzą one z grafu na rysunku 9.5.

Stan	N	I	II	III	IV	V	VI	VII	VIII	IX
Liczba replik	0	0	0	49	0	1	0	2	0	12

Rysunek 9.8. Tabela przedstawiająca pierwsze osiągnięte przez poszczególne repliki stany wchodzące w skład modelu Markova pochodzące z symulacji struktury o ID 1L2Y przeprowadzonej w pakiecie UNRES.

struktur wchodzących w jego skład ma temperaturę fizjologiczną lub niższą. Sugeruje to, że jego minimum energetyczne jest płytkie, ale jest często na krótko osiągany jako stan pośredni, ze względu na istotny udział w populacji.

Drugim stanem do i z którego najczęściej przechodzi struktura natywna jest stan III, w którym znajduje się około 9,5% struktur. Jest to najbardziej spopulowany stan po stanie natywnym. Struktura go reprezentująca posiada jedną zwinięta helisę α, która swoim położeniem bardzo dobrze odpowiada α_1. Reszta struktury pozostaje niezwinięta, ale jej środkowa część przypomina kształtem strukturę natywną. Reszty aminokwasowe tworzące strukturę C-końca są do siebie zbliżone, ale ułożone zupełnie inaczej niż w strukturze natywnej. Stan ten jest nieco stabilniejszy od stanu II, 46% jego struktur znajduje się w temperaturze fizjologicznej i niższej.

Kolejnym istotnym stanem, do i z którego często przechodzi struktura natywna, jest stan IV, w którym znajduje się około 3,7% struktur. Jego centralna struktura posiada jeden krótki fragment helisy α, odpowiadający końcówce α_1. Mimo to jej struktura trzeciorzędowa przypomina strukturę natywną, choć jest bardziej rozwinięta a jej końce zauważalnie rozchodzą się w przestrzeni. Struktura na C-końcu jest częściowo obecna. 4 z 5 jej najważniejszych reszt są względem siebie w pozycjach bliskich natywnym. Rozkład temperatur tego stanu wskazuje na bardzo niską stabilność, jedynie około 23% struktur znajduje się w temperaturze fizjologicznej lub niższej.

Podobny do stanu IV jest stan VI, w którym znajduje się około 9% struktur. Struktura reprezentująca go posiada jeden krótki fragment helisy α, odpowiadający środkowi α_1. Jej struktura trzeciorzędowa przypomina częściowo strukturę natywną, ale jej C-koniec odstaje od reszty cząsteczki. Ponownie 4 z 5 najważniejszych reszt aminokwasowych struktury na C-końcu są blisko siebie, ale nie są to te same reszty co w strukturze reprezentującej stan IV. Stan ten często przechodzi w stan III, w którym to pierwsza helisa jest całkowicie zwinięta, więc przejście pomiędzy tymi stanami

jest najprawdopodobniej związane z jej zwijaniem i rozwijaniem. Rozkład temperatur tego stanu wskazuje na niską stabilność, jedynie około 30% struktur znajduje się w temperaturze fizjologicznej lub niższej.

Stan I jest następnym stanem pośrednim. Znajduje się w nim około 3% struktur. Struktura będąca jego centrum posiada pojedynczy, krótki fragment helisy α, odpowiadający środkowi α_1. Reszty aminokwasowe tworzące strukturę na C-końcu są blisko siebie w przestrzeni, ale mają kompletnie inne ułożenie niż w strukturze natywnej. Jej struktura trzeciorzędowa jest zwarta i najbardziej podobna do struktury natywnej pod względem RMSD spośród wszystkich stanów modelu Markova. Mimo tak dużego podobieństwa do struktury natywnej przejścia pomiędzy tymi dwoma stanami są rzadkie. Ten stan jest raczej stanem pośrednim pomiędzy różnymi stanami prowadzącymi do struktury natywnej. Stan ten jest mało stabilny, około 30% jego struktur znajduje się w temperaturze fizjologicznej lub niższej.

Stan VII jest kolejnym stanem pośrednim. Znajduje się w nim około 2,4% struktur. Jego struktura centralna posiada jedną helisę α, dobrze odpowiadająca α_1. Pozostała część cząsteczki jest zwarta, ale raczej nie przypomina konformacji natywnej. Reszty aminokwasowe tworzące strukturę na C-końcu ponownie są blisko siebie w przestrzeni, ale mają kompletnie inne ułożenie niż w strukturze natywnej. Stan ten jest niestabilny i tylko około 7,5% jego struktur znajduje się w temperaturze fizjologicznej lub niższej.

Stan V jest nieco podobny do stanu VI. Znajduje się w nim około 1,7% struktur. Struktura go reprezentująca posiada pojedynczy, krótki fragment helisy α, odpowiadający środkowi α_1. Pozostała część tej struktury jest słabo zwinięta i nie przypomina struktury natywnej. Struktura na C-końcu jest również nieobecna. Stan ten jest mało stabilny i tylko około 13% jego struktur znajduje się w temperaturze fizjologicznej lub niższej.

Stany VIII i IX są do siebie podobne i znajduje się w nich odpowiednio około 1,3% i 2% struktur. Struktury je reprezentujące nie posiadają zwiniętych fragmentów struktury drugorzędowej i są najodleglejsze pod względem RMSD od struktury natywnej. W obydwu struktura na C-końcu jest częściowo obecna. Są słabo zwiniętymi stanami pośrednimi które mogą przechodzić w wiele z pozostałych stanów, ale żadne konkretne

przejście nie jest w tym wyróżnione. Oba stany są mało stabilne i tylko odpowiednio 10% i 20% struktur znajduje się w temperaturze fizjologicznej lub niższej.

Tabela pierwszych osiąganych stanów (9.8) pokazuje cztery startowe stany. Najliczniejszy jest stan III który jest pierwszym stanem modelu osiąganym przez około 75% replik. Drugim, który osiąga około 18% replik jest stan IX. Obydwa są wyraźnie rozwinięte i częściowo liniowe, co tłumaczy ich pojawienie się jako startowych. Znacząca przewaga stanu III sugeruje, że α_1 tworzy się bardzo szybko na początku procesu zwijania białka. Pojedyncze repliki osiągnęły też stany V i VII. Brak replik które osiągnęły jako pierwszy stan natywny sugeruje, że uzyskany model Markova obejmuje wszystkie najważniejsze ścieżki zwijania tego białka w polu sił UNRES.

Uzyskany model sieci stanów struktury białka pokazuje zarówno różne możliwe ścieżki zwijania, jak i częściowe rozwijanie struktury natywnej w sieć form pośrednich, zgodnie z przyjętą hipotezą badawczą. Pozwala on, opierając się na powyższych obserwacjach, zaproponować dwie ogólne drogi zwijania tego białka. W pierwszej najistotniejsze jest formowanie się α_1. Jest ona obecna w całości w dwóch stanach pośrednich (III, VII) i częściowo w czterech innych. Dopiero po jej uformowaniu formowałaby się α_2 i struktura trzeciorzędowa. Alternatywną ścieżką, której istotnym elementem jest stan II, jest tworzenie się najpierw struktury trzeciorzędowej bliskiej natywnej. Zwinięcie helis α następowałoby w kolejnym kroku. Należy zaznaczyć, że te dwie ścieżki nie są liniowe, ale każda z nich jest mniejszą siecią stanów pośrednich. Nie są też one całkowicie od siebie oddzielone i stany należące do jednego sposobu zwijania mogą przechodzić w stany drugiego. Warto też zauważyć, że w żadnym stanie pośrednim nie pojawiła się zwinięta α_2, co wskazuje na jej niższą stabilność w tym polu sił. Struktura na C-końcu pojawia się częściowo zwinięta na rożne sposoby w wielu stanach modelu, ale trudno jest na ich podstawie stawiać hipotezy o tym procesie. Być może gruboziarnistość pola sił UNRES utrudnia lub uniemożliwia jej tworzenie. Możliwe jest też, że proces odbudowy i minimalizacji pełnego łańcucha polipeptydowego działa na nią niszcząco. Współczynnik żyroskopowy większości struktur jest zbliżony do struktury natywnej (mniej niż 0,8Å różnicy) co pokazuje, że są równie zwarte co struktura

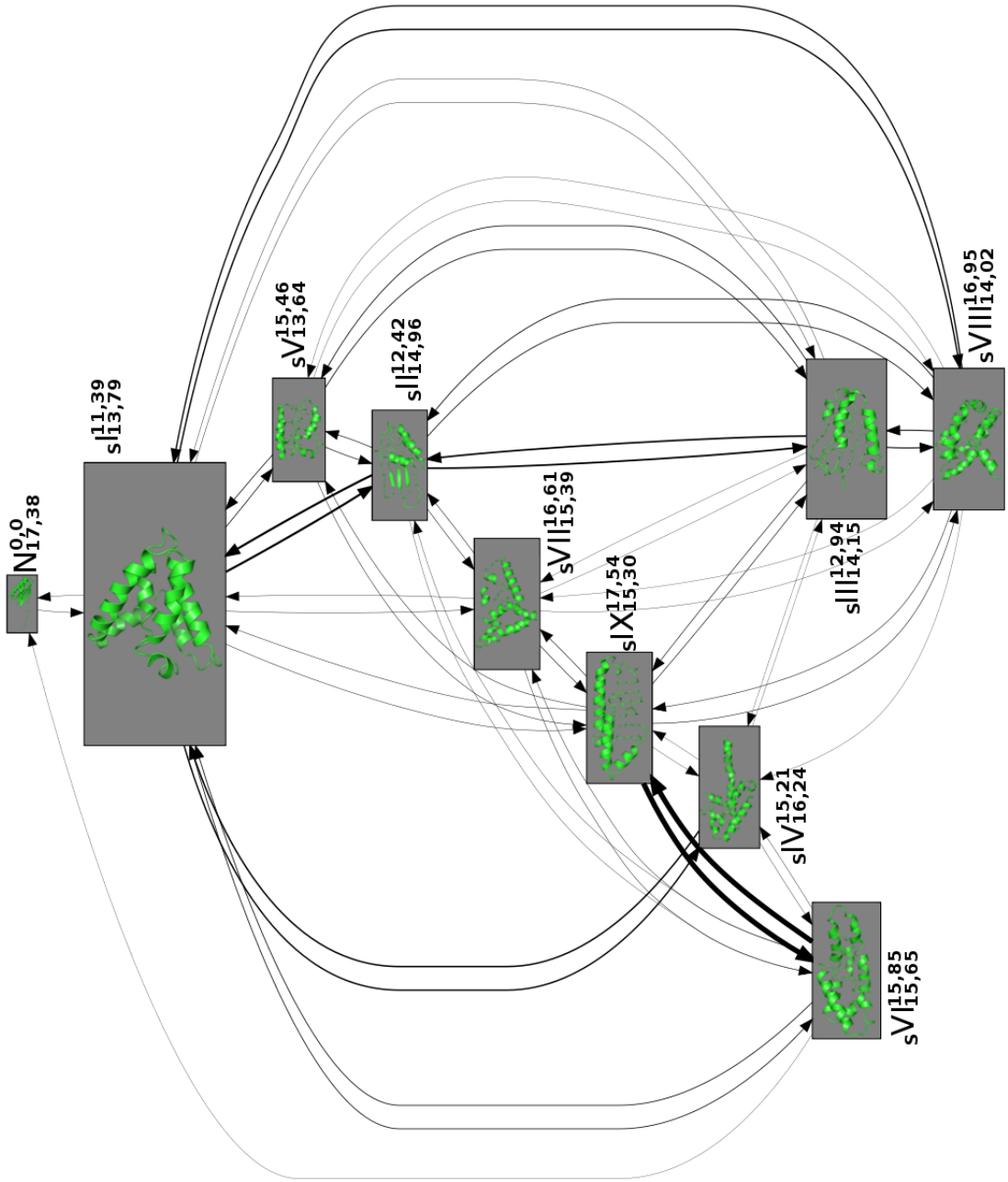
natywna. Struktury mające wyższą wartość tego współczynnika są wyraźnie słabiej zwinięte co wskazuje, że reprezentują wcześniejsze stadia zwijania tego białka.

9.2.3. 2MQ8

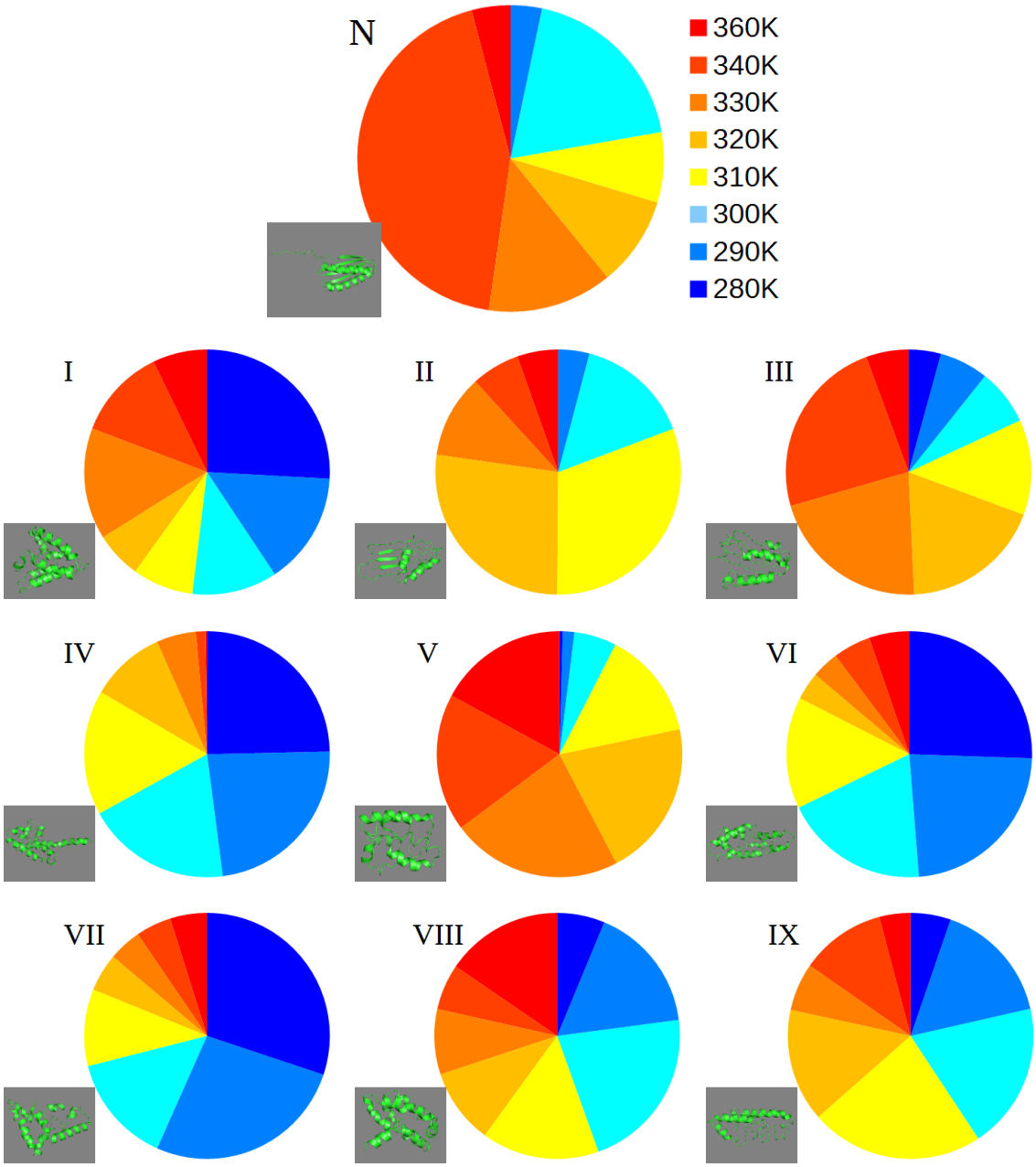
Na rysunku 9.9 znajduje się graf przejścia dla symulacji białka o ID 2MQ8 w programie UNRES. Do jego przygotowania wykorzystałem tylko przejścia pomiędzy strukturami w temperaturze 310K. Jego macierz przejścia znajduje się poniżej. Jest ona bliska symetrycznej, co wskazuje na prawidłowe przygotowanie modelu Markova. Na rysunku 9.10 znajdują się wielkości populacji struktur w poszczególnych temperaturach w grupach wykorzystanych do stworzenia Modelu Markova. Na rysunku 9.11 znajdują się modele struktury natywnej oraz centrów największych uzyskanych grup. Tabela 9.12 przedstawia liczbę replik które osiągnęły poszczególne stany modelu Markova jako pierwsze w czasie tej symulacji.

Z\Do	**N**	**I**	**III**	**VIII**	**VI**	**VII**	**IX**	**IV**	**II**	**V**
N	**6**	8	0	0	0	0	0	0	0	0
I	5	**6434**	9	78	35	10	12	81	129	32
III	0	8	**4742**	51	0	2	24	1	97	37
VIII	0	72	50	**4859**	0	1	17	1	39	2
VI	1	36	0	0	**4126**	11	335	8	5	0
VII	0	7	2	3	12	**2834**	22	0	17	0
IX	0	12	24	18	337	22	**6126**	2	0	15
IV	0	85	1	0	6	0	2	**4003**	0	0
II	0	130	101	36	4	16	0	0	**5968**	25
V	0	32	36	2	0	0	14	0	26	**2466**

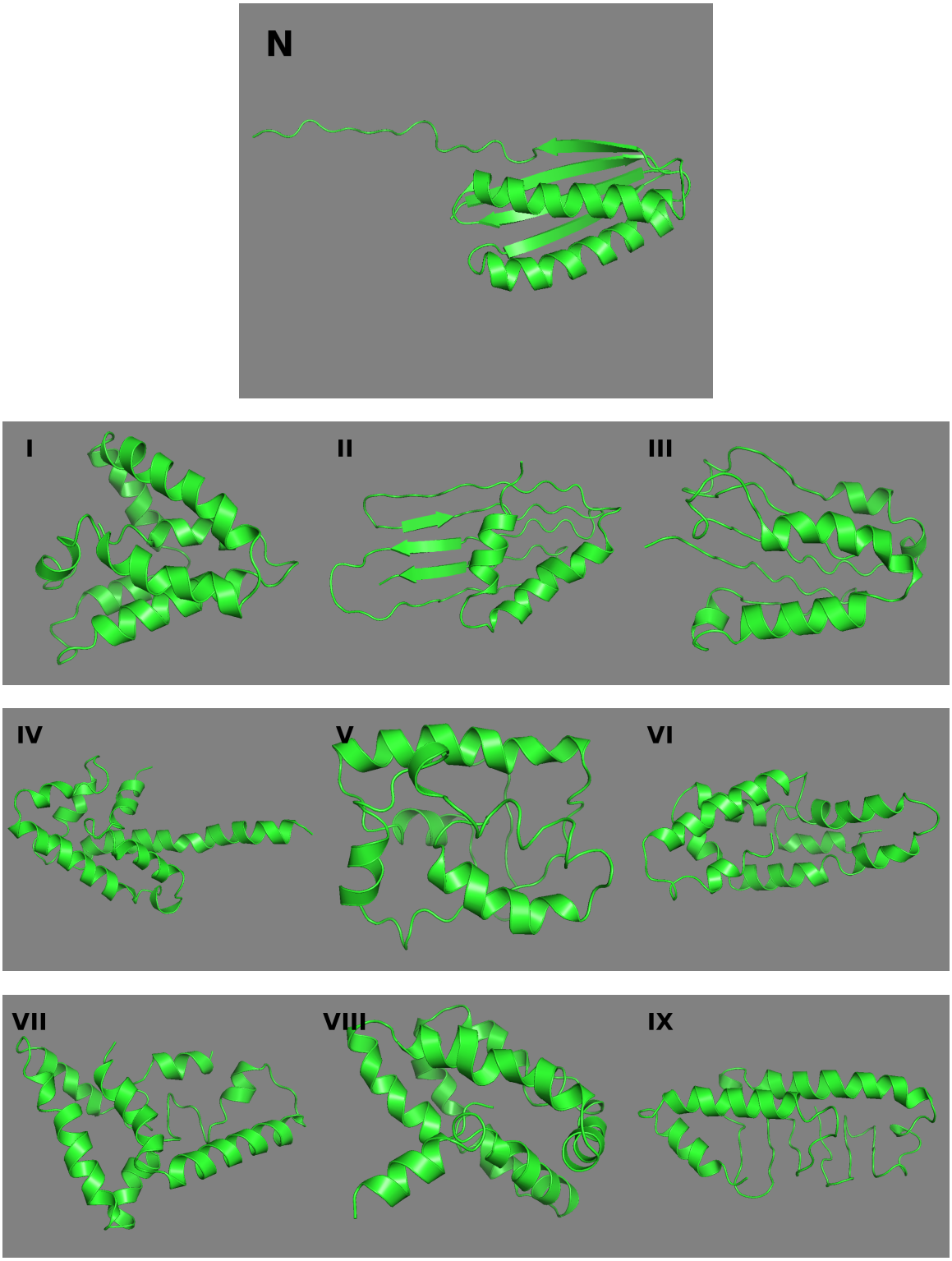
W tej symulacji grupa utworzona przez strukturę natywną (N) jest bardzo mała i obejmuje jedynie 0,038% struktur. Przechodzi ona do i ze stanu I oraz ze stanu VI. To jednostronne przejście jest najprawdopodobniej artefaktem statystycznym wynikającym z niewielkiej liczby zliczonych przejść pomiędzy stanem natywnym i VI. Przy zliczaniu przejść we wszystkich temperaturach pojawiają się przejścia w obie strony. Jedynie 30% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest mało stabilna i może wskazywać, że pole sił UNRES nie jest w stanie jej zwinąć. Innym powodem może być zbyt krótki



Rysunek 9.9. Graf wynikowy dla białka o ID 2MQ8 symulowanego w programie UNRES dla przejść w temperaturze 310K. Obok każdego numeru struktury w indeksie górnym znajduje się jej RMSD względem struktury natywnej, a w indeksie dolnym jej współczynnik żyroskopowy. Stany osiągane przez repliki jako pierwsze są poprzedzone literą S w indeksie dolnym, a liczba tych replik znajduje się w tabeli 9.12. Powiększone modele struktur znajdują się na rysunku 9.11.



Rysunek 9.10. Wykresy kołowe populacji największych grup w poszczególnych temperaturach dla symulacji struktury o ID 2MQ8 przeprowadzonej w pakiecie UNRES. W lewym dolnym rogu każdego wykresu znajduje się model struktury której dany wykres dotyczy. Powiększone modele struktur znajdują się na rysunku 9.11.



Rysunek 9.11. Modele struktury natywnej oraz struktur centralnych największych grup dla symulacji struktury o ID 2MQ8 przeprowadzonej w pakiecie UNRES. Pochodzą one z grafu na rysunku 9.9.

Stan	N	I	II	III	IV	V	VI	VII	VIII	IX
Liczba replik	0	6	2	6	9	4	3	12	4	18

Rysunek 9.12. Tabela przedstawiająca pierwsze osiągnięte przez poszczególne repliki stany wchodzące w skład modelu Markova pochodzące z symulacji struktury o ID 2MQ8 przeprowadzonej w pakiecie UNRES.

czas symulacji, który nie pozwolił żadnej z replik całkowicie się zwinąć, a struktury które znajdują się w tej grupie nie osiągnęły doliny energetycznej związanej ze strukturą natywną, a tylko płytko przekroczyły promień grupy. W tej strukturze wszystkie elementy struktury drugorzędowej układają się w uporządkowany sposób niemal równolegle (lub antyrównolegle) względem siebie z odchyleniami do około 30° i są częściowo wystawione zarówno na zewnątrz jak i do środka cząsteczki, tworząc zamkniętą strukturę przypominającą beczkę lub kanał.

Stan I jest najliczniejszym stanem uzyskanym w analizie skupień i obejmuje około 13,8% struktur. Jego struktura centralna posiada siedem helis α, z których dwie dobrze odpowiadają obydwu helisom struktury natywnej. Nie posiada ona łańcuchów β, ale w miejscach w których występują one w strukturze natywnej pojawiły się helisy α. Pojawiła się też dodatkowa helisa α na C-końcu cząsteczki. Jego struktura trzeciorzędowa, mimo niskiego RMSD, znacząco różni się od struktury natywnej. Elementy struktury drugorzędowej są nieuporządkowane i poukładane względem siebie pod różnymi kątami tworząc kłębek. W tej strukturze istnieje wyraźny, hydrofobowy rdzeń. 52% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest dość stabilna. Stan ten często przechodzi do i ze stanów II, IV, VI, VIII.

Stan VI obejmuje około 4,9% struktur wykorzystanych w analizie i jest drugim stanem z którego zliczono przejścia do struktury natywnej. Jego struktura centralna posiada osiem helis α, z których trzy odpowiadają swoim położeniem dwóm helisom struktury natywnej (z przerwą w środku jednej z nich). Ponownie w miejscu struktur β pojawiają się helisy α, ale zgodność ze strukturą natywną jest mniejsza niż w stanie I. Ponownie pojawiła się dodatkowa helisa α na C-końcu cząsteczki. Jej struktura trzeciorzędowa nie przypomina struktury natywnej. Jej rdzeń jest znacznie dłuższy, węższy i nie tworzy zamkniętej przestrzeni. Utworzył się częściowo rdzeń hydrofobowy,

choć występuje pewna liczba reszt hydrofobowych wystawionych w stronę środowiska zewnętrznego. 82,5% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest bardzo stabilna, a jej dolina energetyczna jest głęboka.

Stan II obejmuje około 3,2% struktur użytych w analizie. Jego struktura centralna posiada dwie helisy α i trzy krótkie łańcuchy β. Jest to jedyna struktura pośrednia posiadająca struktury β. Tylko jedna z nich odpowiada częściowo łańcuchowi β znajdującemu się w strukturze natywnej. Jedna z helis α odpowiada helisie struktury natywnej, a druga jednemu z łańcuchów β. Pojawiła się też helisa na C-końcu, ale w nieco innym miejscu. Pozostałe struktury drugorzędowe są w przypadkowych fragmentach cząsteczki. W strukturze trzeciorzędowej łańcuchy β i fragmenty nieposiadające struktury drugorzędowej tworzą duży, płaski fragment pod którym znajdują się obydwie helisy α. Pozwala to na częściowe utworzenie hydrofobowego rdzenia cząsteczki. 50% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest dość stabilna.

Stan IV obejmuje około 4% struktur. Jego struktura centralna posiada siedem helis α. Dwie z nich odpowiadają dość dobrze helisom struktury natywnej, przy czym jedna z nich jest znacząco (o 5 reszt aminokwasowych) wydłużona w obie strony. Istnieją też helisy odpowiadające strukturom β struktury natywnej i dodatkowa helisa na C-końcu cząsteczki. Struktura trzeciorzędowa jest kłębkiem z którego wystaje najdłuższa helisa. Jako całość posiada duży hydrofobowy rdzeń który ją stabilizuje. 83% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest bardzo stabilna, a jej dolina energetyczna głęboka.

Stan VIII obejmuje około 5,2% struktur. Jego struktura centralna posiada osiem helis α. Trzy z nich odpowiadają dobrze dwóm helisom struktury natywnej. Pozostałe znajdują się częściowo w miejscach odpowiadających łańcuchom β struktury natywnej. Pojawiła się też dodatkowa helisa na C-końcu. Jej struktura trzeciorzędowa jest zwarta i tworzy zamknięta strukturę, która jednak nie przypomina struktury natywnej. W strukturze tej utworzył się wyraźny rdzeń hydrofobowy. 60% struktur w tej

grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest stabilna.

Stan III jest istotnym stanem pośrednim, do i z którego przechodzi wiele innych stanów. Znajduje się w nim około 6,2% struktur użytych w analizie. Jego struktura centralna posiada pięć helis α. Trzy z nich częściowo odpowiadają dwóm helisom struktury natywnej. Pozostałe dwie znajdują się w pobliżu miejsc występowania struktur β, ale nie pokrywają się dobrze z nimi. Jego struktura trzeciorzędowa tworzy dwie równoległe płaszczyzny. Większa z nich złożona jest z dwóch helis i fragmentów bez struktury natywnej, które przypominają swoim ułożeniem β kartkę. Mniejszą tworzą dwie helisy α. W strukturze tej utworzył się częściowo rdzeń hydrofobowy, ale dużo hydrofobowych łańcuchów bocznych jest wciąż wystawionych w kierunku środowiska. 31% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest mało stabilna.

Stan V jest następnym stanem pośrednim. Znajduje się w nim około 2,9% struktur. Jego struktura centralna posiada pięć helis α. Dwie z nich częściowo odpowiadają pierwszej, a kolejna bardzo dobrze odpowiada drugiej helisie struktury natywnej. Jedna z pozostałych helis odpowiada jednemu z łańcuchów β. W strukturze trzeciorzędowej tworzy ciasno zwinięty kłębek, nieprzypominający struktury natywnej. Utworzył się w niej wyraźny rdzeń hydrofobowy, ale część reszt hydrofobowych wystawiona jest w kierunku środowiska. Tylko 22% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest mało stabilna i jest raczej wysokoenergetycznym stanem pośrednim osiąganym przez repliki w wyższych temperaturach.

Stan VII obejmuje około 4,5% struktur użytych w analizie skupień. Jego struktura centralna posiada osiem helis α. Dwie z nich odpowiadają dobrze dwóm helisom, a trzy następne trzem łańcuchom β struktury natywnej. Pojawiła się ponownie dodatkowa helisa na C-końcu. Struktura trzeciorzędowa przypomina dwa trójkąty złożone z helis α, położone jeden na drugim i przesunięte względem siebie. Istnieje w nim wyraźny rdzeń obejmujący większość hydrofobowych aminokwasów. 81% struktur w tej grupie znaj-

duje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest bardzo stabilna, a jej dolina energetyczna głęboka.

Stan IX obejmuje około 4,5% struktur. Jego struktura centralna posiada pięć helis α. Dwie z nich odpowiadają dobrze dwóm helisom struktury natywnej, dwie kolejne częściowo odpowiadają jej dwóm łańcuchom β. Charakterystycznym elementem struktury trzeciorzędowej jest płaski fragment przypominający β kartkę utworzony przez reszty aminokwasowe nie należące do żadnej helisy. Trzy najdłuższe helisy α znajdują się pod tą strukturą. Pozwoliło to na utworzenie się znaczącego rdzenia hydrofobowego. 63% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że struktura ta jest stabilna. Stan ten bardzo często przechodzi do i ze stanu VI.

Tabela pierwszych osiąganych stanów (9.12) pokazuje, że wszystkie stany poza natywnym były osiągane jako pierwsze przez repliki w symulacji. Wskazuje to, że istnieje rozległa sieć dodatkowych, mniej istotnych stanów pośrednich które doprowadzają strukturę do zwinięcia w jeden ze stanów modelu Markova. Brak replik które osiągnęły jako pierwszy stan natywny sugeruje natomiast, że uzyskany model Markova obejmuje wszystkie najważniejsze ścieżki zwijania tego białka w polu sił UNRES. Liczbą replik w pewnym stopniu wyróżnia się stan IX (około 28% replik), który jednocześnie jest najbardziej rozwiniętym stanem modelu Markova.

Wszystkie stany pośrednie są mocno zwinięte i posiadają po kilka (5-8) helis α. W zdecydowanej większości pojawiają się helisy odpowiadające helisom struktury natywnej. Powstają one również we fragmentach cząsteczki, w których struktura natywna posiada struktury β. Może to być spowodowane tendencjami pola sił do nadmiernej stabilizacji helis. Hipotezę tą wspiera też częste pojawianie się dodatkowej helisy na C-końcu cząsteczki. Struktury β pojawiają się tylko w jednym stanie pośrednim. Kilka tych stanów posiada płaskie, złożone z kilku łańcuchów struktury przypominające β-kartki (III i IX). Być może są one wprost β-kartkami, które zostały na tyle zaburzone przez proces odbudowy łańcucha, że program PyMOL przestał je wykrywać. Struktury te jednak tylko częściowo tworzą się w regionach, w których struktura natywna posiada łańcuchy β. Wszystkie uzyskane stany przejściowe mają współczynnik

żyroskopowy mniejszy od struktury natywnej, co wynika z obecności w niej długiego, prostego, wystającego znacznie poza resztę cząsteczki C-końca.

Uzyskany model sieci stanów struktury białka pokazuje zarówno różne możliwe ścieżki zwijania, jak i częściowe rozwijanie struktury natywnej w sieć form pośrednich, zgodnie z przyjętą hipotezą badawczą. Pozwala on, opierając się na powyższych obserwacjach, formułować ogólne hipotezy na temat przebiegu zwijania tego białka w polu sił UNRES. Uzyskane wyniki wskazują, że białko to szybko zwija się w zwartą strukturę, gdyż żaden stan pośredni nie ma bezpośredniego związku ze startową, liniową strukturą. Dwie helisy α struktury natywnej istnieją częściowo lub w całości we wszystkich stanach pośrednich, co wskazuje na ich dużą stabilność. Najliczniejszym, głównym stanem prowadzącym do struktury natywnej jest stan I, który jest kłębkiem helis. Jest możliwe, że wzajemne oddziaływania odległych fragmentów łańcucha w tym kłębku promują zmianę helis w struktury β, co mogłoby doprowadzić do uzyskania struktury natywnej i reprezentować jedną z możliwych ścieżek zwijania. Inną ścieżkę zwijania może reprezentować kilka stanów które, jak wspomniano, tworzą strukturę układającą się w dwie płaszczyzny. Jeśli jedna z tych płaszczyzn byłaby utworzona ze struktur β a druga z helis α to po ich skręceniu i zgięciu powstałaby struktura natywna. Stany II, III i IX najbardziej przypominają taką strukturę. Należy też zauważyć, że niniejsze białko jest największe z symulowanych i czas symulacji mógł być niewystarczający aby dostateczna liczba replik osiągnęła strukturę bliską natywnej.

9.3. AMBER

9.3.1. 1BDD

Na rysunku 9.13 znajduje się graf przejścia dla symulacji białka o ID 1BDD w programie AMBER. Do jego przygotowania wykorzystałem tylko przejścia pomiędzy strukturami w temperaturze 310K. Jego macierz przejścia znajduje się poniżej. Jest ona bliska symetrycznej, co wskazuje na prawidłowe przygotowanie modelu Markova. Na rysunku 9.15 znajdują się wielkości populacji struktur w poszczególnych temperaturach w grupach wykorzystanych do stworzenia Modelu Markova. Na rysunku 9.16 znajdują

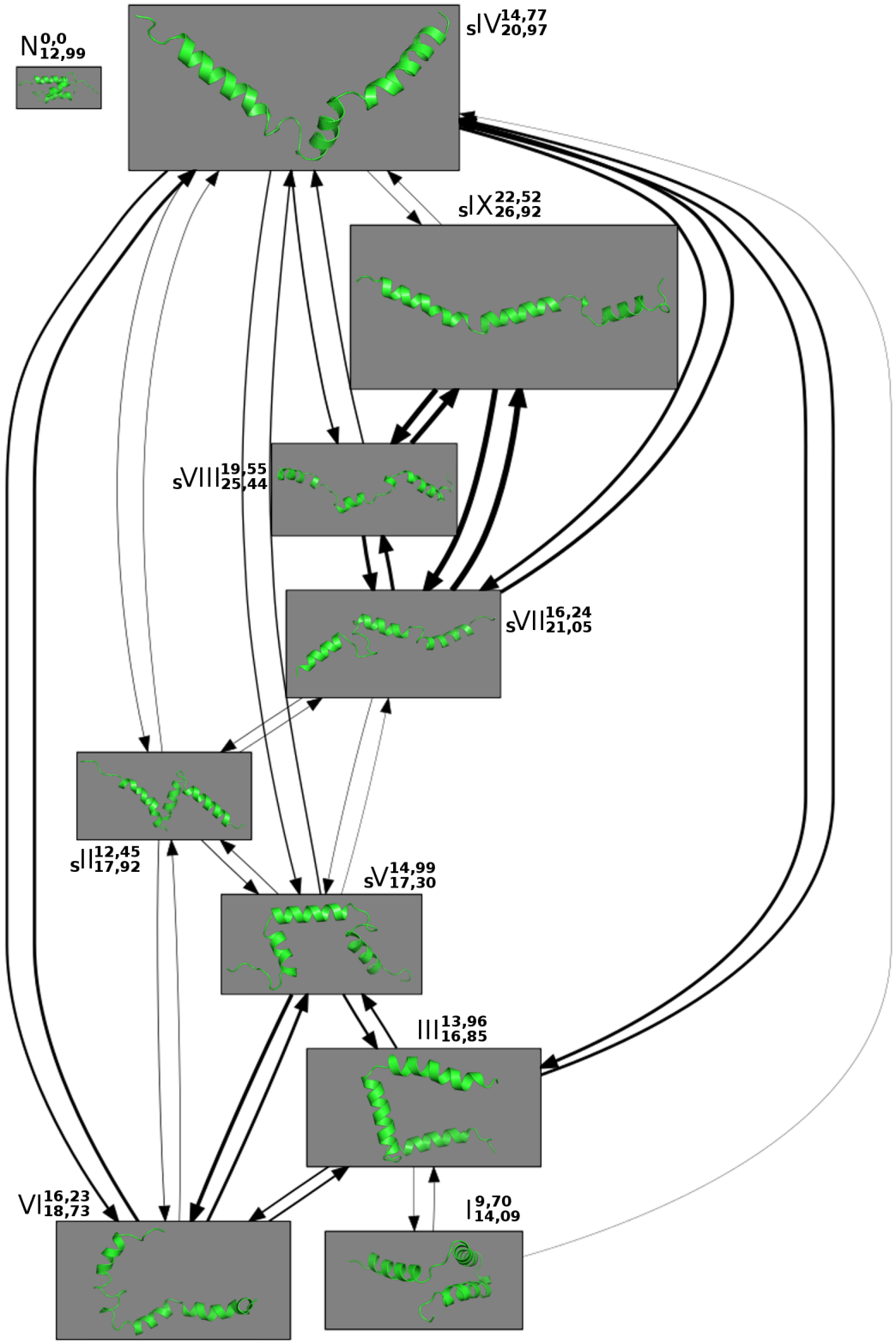
się modele struktury natywnej oraz centrów największych uzyskanych grup. Tabela 9.17 przedstawia liczbę replik które osiągnęły poszczególne stany modelu Markova jako pierwsze w czasie tej symulacji.

Z\Do	**N**	**IV**	**IX**	**VI**	**III**	**VII**	**V**	**I**	**VIII**	**II**
N	**159**	0	0	0	0	0	0	0	0	0
IV	0	**3940**	5	34	43	46	18	0	23	7
IX	0	5	**3691**	0	0	80	0	0	76	0
VI	0	44	0	**2766**	26	0	41	0	0	7
III	0	40	0	23	**2486**	0	30	4	0	0
VII	0	50	86	0	0	**3167**	5	0	49	7
V	0	19	0	52	34	3	**1543**	0	0	9
I	0	1	0	0	6	0	0	**874**	0	0
VIII	0	21	71	0	0	51	0	0	**1533**	0
II	0	6	0	9	0	6	11	0	0	**1890**

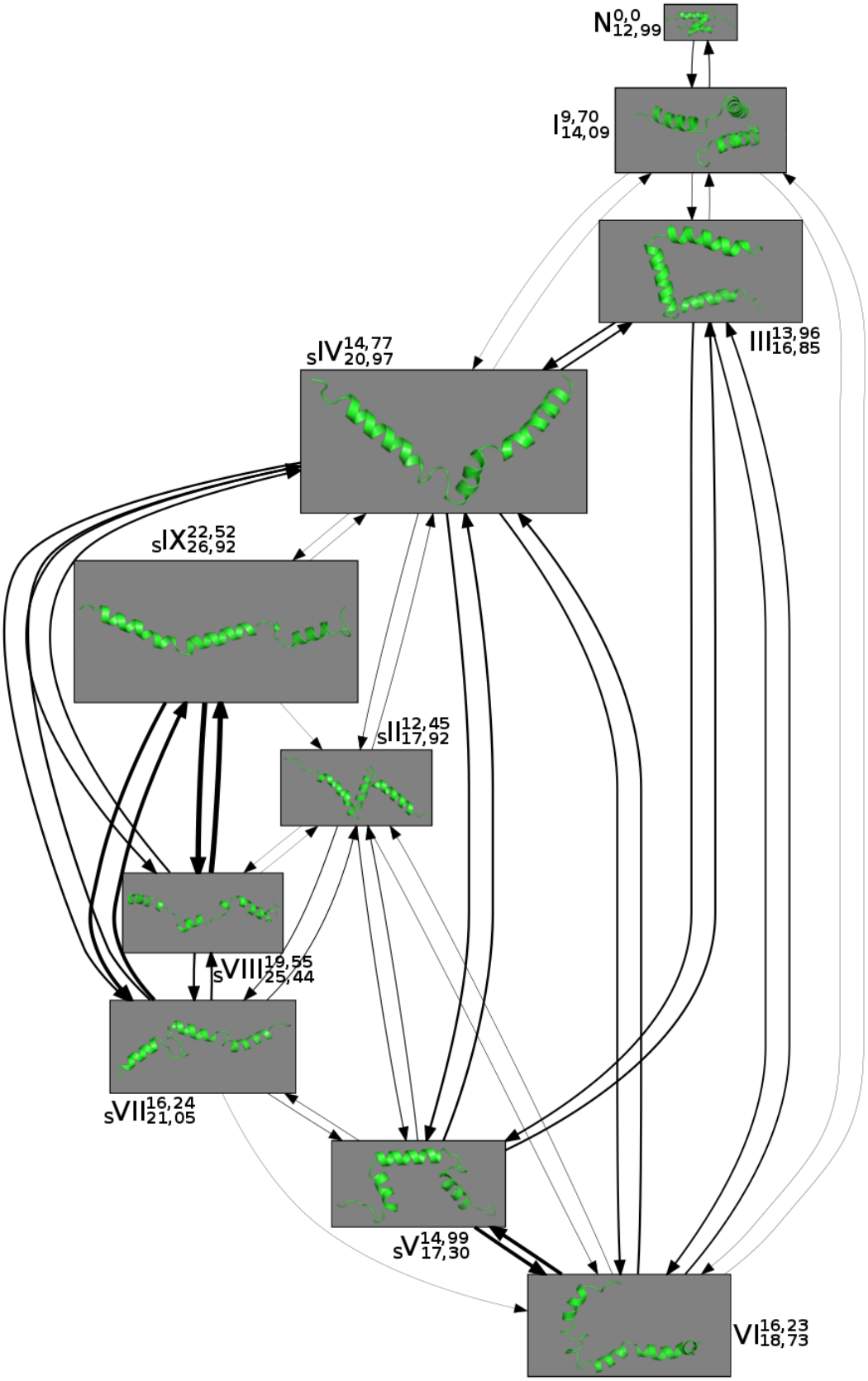
Graf 9.13 dla przejść zarejestrowanych w temperaturze 310K nie jest spójny (żadna krawędź nie dochodzi do struktury natywnej). W związku z tym do analizy zostanie użyty graf 9.14 zawierający przejścia zliczone dla wszystkich temperatur.

W tej symulacji grupa utworzona przez strukturę natywną (N) jest niewielka i obejmuje około 0,3% struktur użytych w analizie skupień. Struktura natywna zawiera 3 helisy α, które dalej będą oznaczane α_1, α_2, α_3. Struktura ta przechodzi do i z jednego stanu pośredniego (numer I). 91% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że dolina energetyczna związana z tym stanem jest głęboka i wskazuje na dużą stabilność tej struktury, co jest oczekiwane dla struktury natywnej.

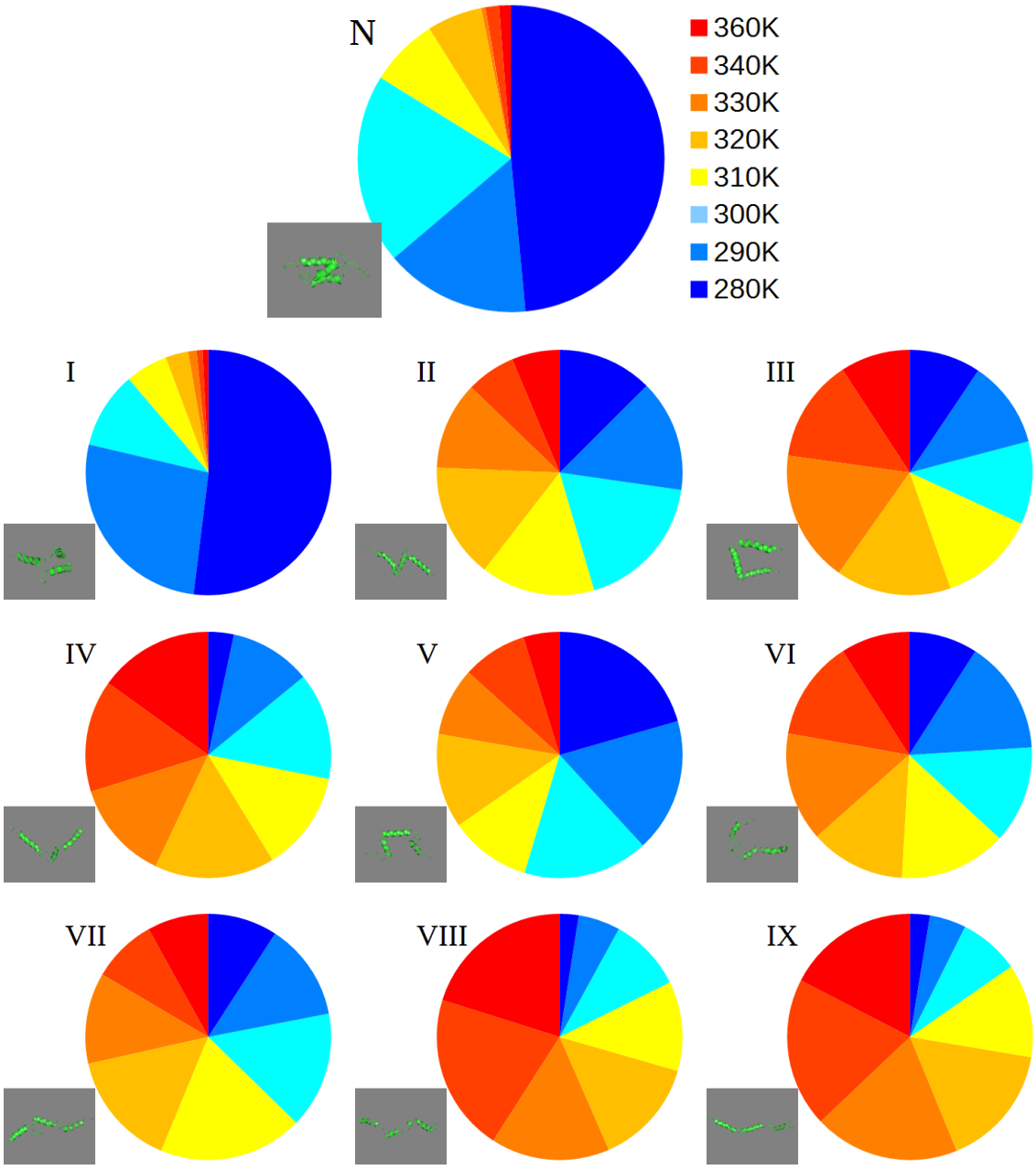
Jedynym stanem do i z którego najczęściej przechodzi struktura natywna jest stan I, w którym znajduje się około 2,2% struktur. Struktura go reprezentująca posiada trzy zwinięte helisy α. Druga i trzecia z nich swoim położeniem na łańcuchu polipeptydowym bardzo dobrze odpowiadają α_2 i α_3, choć są nieco wydłużone w porównaniu z nią. Pierwsza jest znacząco wydłużona i przesunięta w stronę N-końca cząsteczki względem α_1. α_2 i α_3 są ułożone względem siebie prawie dokładnie tak, jak w strukturze natywnej. α_1 jest odchylona od nich, i przez to też od swojego natywnego położenia, o około 90°. Wskazuje to na prosty mechanizm przechodzenia do i ze struktury na-



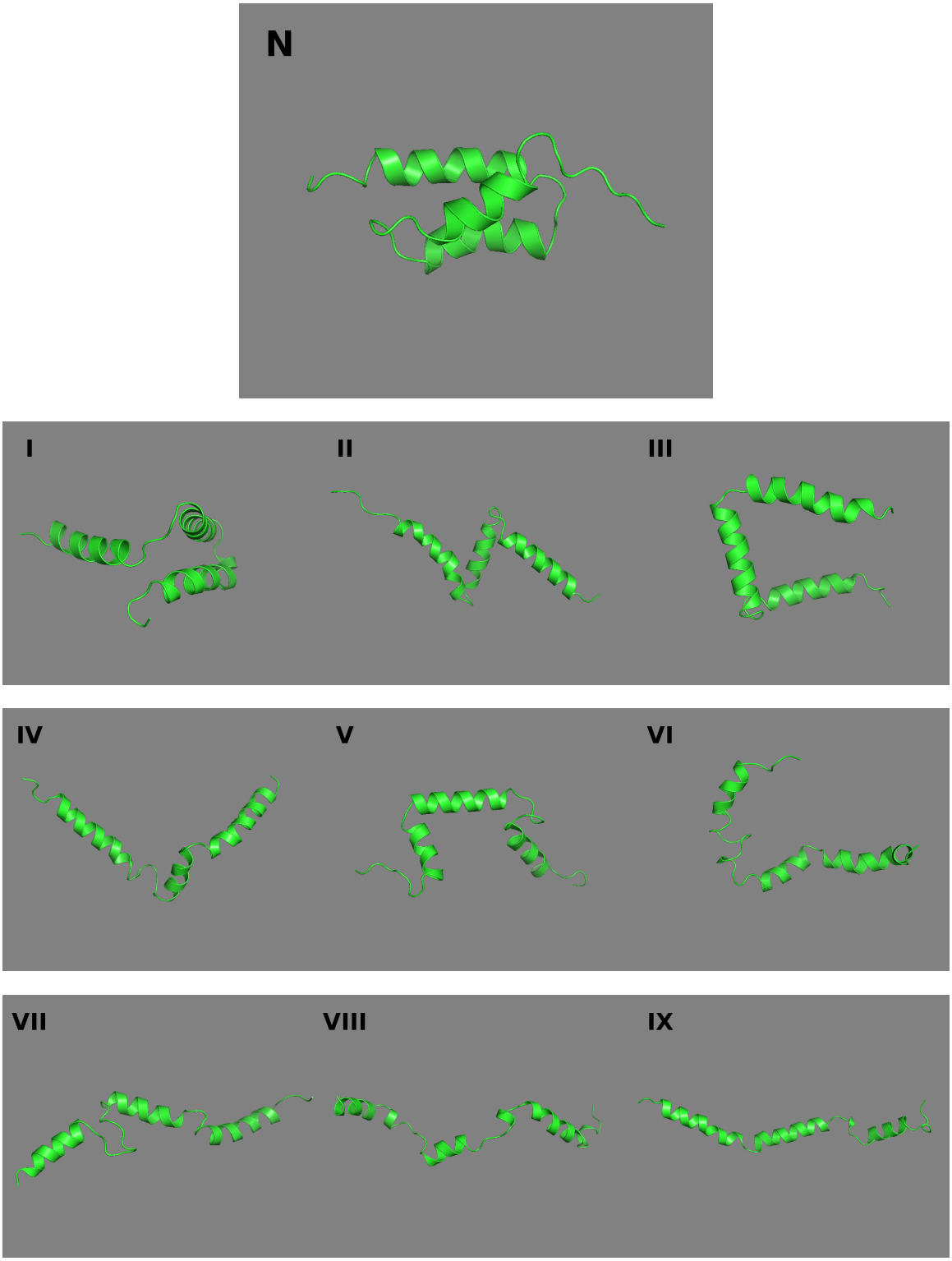
Rysunek 9.13. Graf wynikowy dla białka o ID 1BDD symulowanego w programie AMBER dla przejść w temperaturze 310K. Obok każdego numeru struktury w indeksie górnym znajduje się jej RMSD względem struktury natywnej, a w indeksie dolnym jej współczynnik żyroskopowy. Stany osiągane przez repliki jako pierwsze są poprzedzone literą S w indeksie dolnym, a liczba tych replik znajduje się w tabeli 9.17. Powiększone modele struktur znajdują się na rysunku 9.16.



Rysunek 9.14. Graf wynikowy dla białka o ID 1BDD symulowanego w programie AMBER dla przejść we wszystkich temperaturach. Obok każdego numeru struktury w indeksie górnym znajduje się jej RMSD względem struktury natywnej, a w indeksie dolnym jej współczynnik żyroskopowy. Stany osiągane przez repliki jako pierwsze są poprzedzone literą S w indeksie dolnym, a liczba tych replik znajduje się w tabeli 9.17. Powiększone modele struktur znajdują się na rysunku 9.16.



Rysunek 9.15. Wykresy kołowe populacji największych grup w poszczególnych temperaturach dla symulacji struktury o ID 1BDD przeprowadzonej w pakiecie AMBER. W lewym dolnym rogu każdego wykresu znajduje się model struktury której dany wykres dotyczy. Powiększone modele struktur znajdują się na rysunku 9.16.



Rysunek 9.16. Modele struktury natywnej oraz struktur centralnych największych grup dla symulacji struktury o ID 1BDD przeprowadzonej w pakiecie AMBER. Pochodzą one z grafu na rysunku 9.14.

Stan	N	I	II	III	IV	V	VI	VII	VIII	IX
Liczba replik	0	0	1	0	4	1	0	2	11	45

Rysunek 9.17. Tabela przedstawiająca pierwsze osiągnięte przez poszczególne repliki stany wchodzące w skład modelu Markova pochodzące z symulacji struktury o ID 1BDD przeprowadzonej w pakiecie AMBER.

tywnej, polegający na zmianach kąta tego odchylenia. Hydrofobowy rdzeń cząsteczki jest w pewnym stopniu wystawiony do środowiska ze względu na opisane odchylenie helisy. 94% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co wskazuje na jej bardzo dużą stabilność i dużą głębokość jej doliny energetycznej. W uzyskanym modelu ten stan przechodzi do i z trzech innych stanów (poza natywnym): III, IV, VI.

Stan III obejmuje około 2,8% struktur użytych w analizie skupień. Struktura będąca jego centrum posiada trzy zwinięte helisy α. Są one zgodne z helisami obecnymi w strukturze natywnej, jedynie pierwsza jest znacząco wydłużona w kierunku N-końca łańcucha polipeptydowego. Te trzy helisy są ułożone względem siebie w przestrzeni w przybliżeniu jak trzy sąsiednie krawędzie kwadratu. Stan ten może przechodzić w stan I poprzez wzajemne zbliżenie się drugiej i trzeciej helisy. Struktury należące do tej grupy są rozłożone prawie równo pod względem temperatur. Nie tworzy rdzenia hydrofobowego ze względu na wysoki stopień rozwinięcia. Około 45% z nich znajduje się w temperaturze fizjologicznej lub niższej (310K i mniej). Wskazuje to, że dolina energetyczna z nim związana jest raczej płytka i repliki w każdej temperaturze mogły ją łatwo opuścić.

Stan IV obejmuje około 4,4% struktur i pod wieloma względami jest podobny do stanu III. Struktura go reprezentująca posiada trzy zwinięte helisy α z których pierwsza jest wydłużona w kierunku N-końca, druga przesunięta w tym samym kierunku, a położenie trzeciej zgadza się z natywną. Układają się one w kształt przypominający literę 'L' nieco skręconą w trzecim wymiarze. Stan ten może przechodzić w stan I również poprzez wzajemne zbliżenie się drugiej i trzeciej helisy. Nie tworzy rdzenia hydrofobowego ze względu na wysoki stopień rozwinięcia. Jedynie na zgięciach pomiędzy helisami znajdują się małe grupy reszt hydrofobowych, które mogą stabilizować strukturę. Rozkład temperatur jest w nim bardzo podobny do stanu III

i można na jego podstawie przedstawić identyczne wnioski. Jedynie procent struktur w temperaturze fizjologicznej i niższej jest minimalnie mniejszy i wynosi 41%.

W stanie VI znajduje się około 2,8% struktur użytych w analizie skupień. Struktura go reprezentująca posiada cztery helisy α. Pierwsza odpowiada α_1, ale ponownie jest znacząco wydłużona w kierunku N-końca cząsteczki. Druga i trzecia helisa odpowiadają dość dobrze α_2, przedłużonej nieco w kierunku N-końca z niewielką przerwą w środku. Czwarta helisa odpowiada dobrze α_3, ale jest od niej nieco krótsza. Jego struktura trzeciorzędowa jest podobna do stanu IV. Może przechodzić w stan I przez prawidłowe zwinięcie do końca drugiej helisy i zbliżenie się jej do helisy trzeciej. Nie tworzy rdzenia hydrofobowego ze względu na wysoki stopień rozwinięcia. Rozkład temperatur jest w nim bardzo podobny do stanu III i można na jego podstawie przedstawić identyczne wnioski. Jedynie procent struktur w temperaturze fizjologicznej i niższej jest nieco większy i wynosi 51%, co może wskazywać na nieco większą stabilność tej struktury. Te 3 stany (III, IV, VI) dość często przechodzą jeden w drugi.

Stan V jest blisko związany ze stanem VI przez częste wzajemne przejścia. Znajduje się w nim 2,2% struktur użytych w analizie. Jego struktura drugorzędowa zawiera trzy helisy α, które bardzo dobrze odpowiadają helisom struktury natywnej, jedynie druga z nich jest nieco krótsza. W strukturze trzeciorzędowej pierwsza i druga helisa są niemal w jednej płaszczyźnie i znajdują się pod kątem około 90° względem siebie. Trzecia helisa odchyla się od nich na zewnątrz cząsteczki i poza ich płaszczyznę. Jest to jedyny uzyskany stan pośredni w którym pierwsza helisa jest prawidłowo zwinięta. Fakt, że nie przechodzi on w strukturę natywną sugeruje, że tymczasowe przedłużenie pierwszej helisy może odgrywać istotną rolę w ostatecznym zwinięciu cząsteczki do struktury natywnej. Może też być ślepym zaułkiem procesu zwijania. Poza stanem VI przechodzi dość często do i z opisanych wcześniej stanów III i IV. Ponownie nie istnieje rdzeń hydrofobowy cząsteczki, ale na zgięciach struktury znajdują się niewielkie grupy reszt hydrofobowych. 65% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co wskazuje na jej dużą stabilność i znaczącą głębokość jej doliny energetycznej. Zgadzałoby się to z zaproponowaną rolą ślepego zaułka procesu zwijania białka.

Stany VIII i IX są najbardziej rozwinięte, na co wskazują ich współczynniki żyroskopowe, i reprezentują początkowe stadia procesu zwijania białka. Często przechodzą jeden w drugi. Znajduje się w nich odpowiednio 2% i 4,3% struktur użytych w analizie. Struktura reprezentująca stan VIII składa się z sześciu helis α, które parami odpowiadają helisom struktury natywnej z przerwą w środku. Wszystkie są wydłużone w kierunku N-końca, najbardziej pierwsza helisa. Struktura reprezentująca stan VIII składa się z czterech helis α. Pierwsza i druga odpowiadają α_1 i α_2, natomiast trzecia i czwarta odpowiadają α_3. Wszystkie helisy są przesunięte w kierunku N-końca cząsteczki. Struktura trzeciorzędowa obu tych struktur jest przede wszystkim liniowa, choć widać też w nich początek procesu zbliżania się do siebie helis α. Z tego powodu nie posiadają rdzeni hydrofobowych. Jedynie odpowiednio 29% i 28% struktur w tych grupach znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co wskazuje że ich doliny energetyczne są bardzo płytkie i że są krótkotrwałymi stanami pośrednimi.

Stan VII przypomina opisane powyżej stany VIII i IX. Znajduje się w nim około 2,4% struktur użytych w analizie. Struktura go reprezentująca posiada trzy helisy α. Druga i trzecia bardzo dobrze odpowiadają α_2 i α_3, natomiast pierwsza jest mocno przesunięta w kierunku N-końca. Pętla pomiędzy pierwszą a druga helisą jest zwarta i zawiera zwrot. Wspólnie z fragmentem drugiej helisy tworzy mały hydrofobowy region. Jej struktura trzeciorzędowa jest również przede wszystkim liniowa. 56% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co wskazuje na jej dość dużą stabilność, zwłaszcza w porównaniu ze strukturami VIII i IX. Może ona wynikać zarówno z uformowania większej części helis α jak i z wpływu wspomnianej zwiniętej pętli pomiędzy pierwszą a drugą helisą.

Ostatnim stanem użytym do budowy modelu Markova jest stan II. Znajduje się w nim około 2,4% struktur użytych w analizie. Struktura go reprezentująca posiada trzy helisy α. Druga i trzecia dość dobrze odpowiadają α_2 i α_3, natomiast pierwsza jest mocno wydłużona w kierunku N-końca. Jej struktura trzeciorzędowa przypomina literę 'Z'. Widać w niej zbliżanie się do siebie poszczególnych helis α w porównaniu ze stanami VII, VIII, IX. Jest ona stanem pośrednim istniejącym pomiędzy bardziej i mniej zwiniętymi stanami. Nie posiada ona rdzenia hydrofobowego. 60% struktur w tej

grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co wskazuje na jej dużą stabilność i obecność zauważalnej doliny energetycznej.

Tabela pierwszych osiąganych stanów (9.17) pokazuje, że sześć stanów było osiągane jako pierwsze przez repliki w symulacji. Wskazuje to na istnienie różnych ścieżek które prowadzą strukturę od rozwiniętej do jednego ze stanów modelu Markova. Brak replik które osiągnęły jako pierwszy stan natywny sugeruje natomiast, że uzyskany model Markova obejmuje wszystkie najważniejsze ścieżki zwijania tego białka w polu sił AMBER. Liczbą replik wyróżnia się stan IX (około 70% replik) i w mniejszym stopniu stan VIII (około 17% replik) które są najbardziej rozwiniętymi stanami modelu Markova.

Uzyskany model sieci stanów struktury białka pokazuje zarówno różne możliwe ścieżki zwijania, jak i częściowe rozwijanie struktury natywnej w sieć form pośrednich, zgodnie z przyjętą hipotezą badawczą. Pozwala on, opierając się na powyższych obserwacjach, formułować ogólne hipotezy na temat przebiegu zwijania tego białka w polu sił AMBER. Wszystkie helisy α są obecne w każdym stanie, co sugeruje że powstają bardzo szybko. Stan IX wskazuje, że każda z nich zaczyna się tworzyć w dwóch miejscach które wydłużają się i łączą ze sobą. W prawie każdym stanie, poza natywnym i V pierwsza helisa jest znacząco przesunięta lub wydłużona w kierunku N-końca cząsteczki. Być może stabilizuje to formy pośrednie, albo pole sił ma tendencje do tworzenia takiej struktury. Pod koniec tworzenia helis zaczynają się one zbliżać do siebie na rożne sposoby, czego wynikiem jest szeroka sieć różnych konformacji pośrednich (stany II, III, IV, V, VI). Stan I pokazuje, że α_2 i α_3 zbliżają się do siebie najpierw i dopiero po tym jak osiągną swoją natywną konformację zbliża się do nich α_1.

Pole sił programu AMBER zwinęło to białko w strukturę bliską natywnej, choć stan natywny był bardzo słabo spopulowany. Jak wspomniano charakterystyczną cechą wielu struktur pośrednich jest wydłużona pierwsza helisa α. To samo zjawisko zaobserwować można w opisanej powyżej symulacji w programie UNRES co sugeruje, że może to być rzeczywiste zjawisko, a nie artefakt pola sił. Struktura natywna posiada najmniejszy współczynnik żyroskopowy co jest zgodne z oczekiwaniami ze względu na wyraźnie mniejszą zwartość pozostałych stanów.

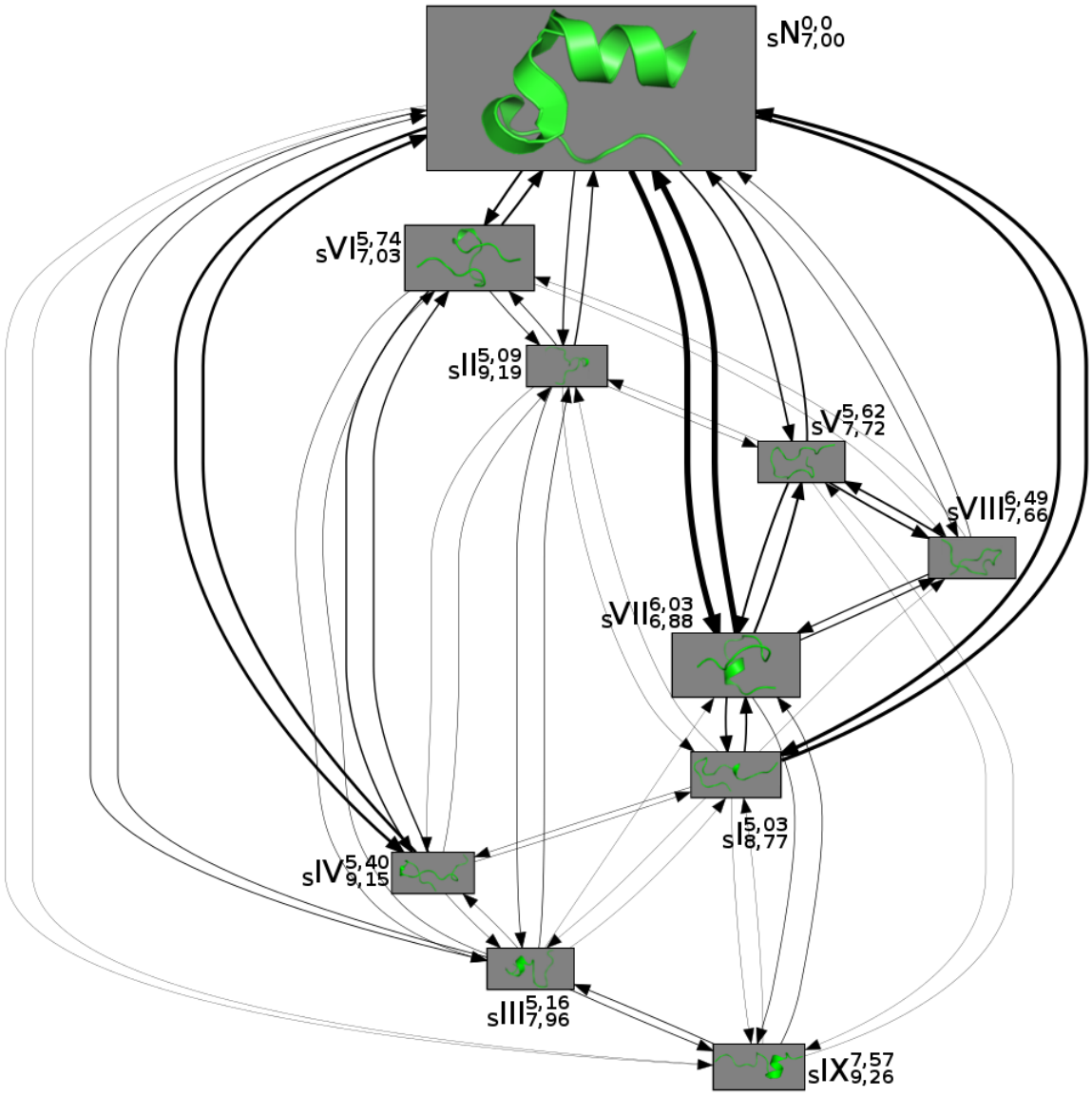
9.3.2. 1L2Y

Na rysunku 9.18 znajduje się graf przejścia dla symulacji białka o ID 1L2Y w programie AMBER. Do jego przygotowania wykorzystałem tylko przejścia pomiędzy strukturami w temperaturze 310K. Jego macierz przejścia znajduje się poniżej. Jest ona bliska symetrycznej, mimo istnienia paru wartości znacząco różniących się od swoich odpowiedników. Wskazuje to na raczej prawidłowe przygotowanie modelu Markova. Na rysunku 9.19 znajdują się wielkości populacji struktur w poszczególnych temperaturach w grupach wykorzystanych do stworzenia Modelu Markova. Na rysunku 9.20 znajdują się modele struktury natywnej oraz centrów największych uzyskanych grup. Tabela 9.21 przedstawia liczbę replik które osiągnęły poszczególne stany modelu Markova jako pierwsze w czasie tej symulacji.

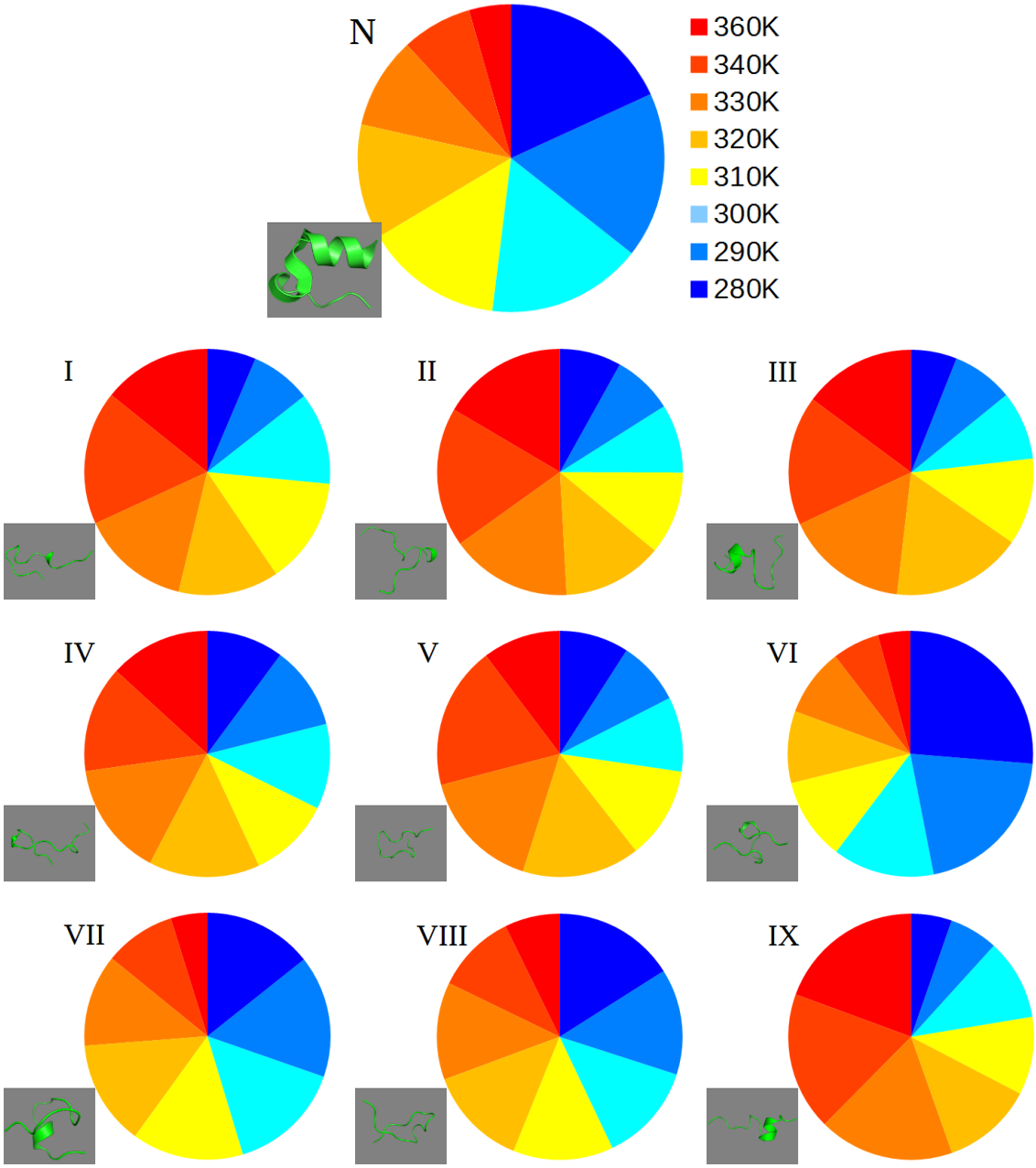
Z\Do	**N**	**VI**	**VII**	**IX**	**I**	**III**	**VIII**	**IV**	**V**	**II**
N	**31614**	168	477	1	280	58	38	223	130	97
VI	167	**5529**	0	0	0	26	4	111	0	53
VII	476	0	**6834**	39	129	0	93	0	168	0
IX	1	0	40	**2037**	4	57	0	0	3	0
I	278	0	128	6	**2405**	1	1	17	0	8
III	55	27	1	52	1	**1490**	0	15	0	43
VIII	38	6	88	0	0	0	**1699**	0	137	0
IV	220	101	0	0	16	17	0	**1114**	0	21
V	145	0	156	1	0	0	139	0	**1232**	9
II	106	53	0	0	8	42	0	13	12	**1163**

W tej symulacji grupa utworzona przez strukturę natywną zawiera 30,5% struktur użytych w analizie i jest największą uzyskaną grupą. Struktura natywna zawiera 2 helisy α, które dalej będą oznaczane α_1 i α_2. 66% jej struktur znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Sugeruje to, że dolina energetyczna związana z tym stanem jest dość głęboka i wskazuje na dużą stabilność tej struktury, co jest oczekiwane dla struktury natywnej. Struktura ta przechodzi najczęściej do i ze stanu VII, istotne są też przejścia do stanów I i IV.

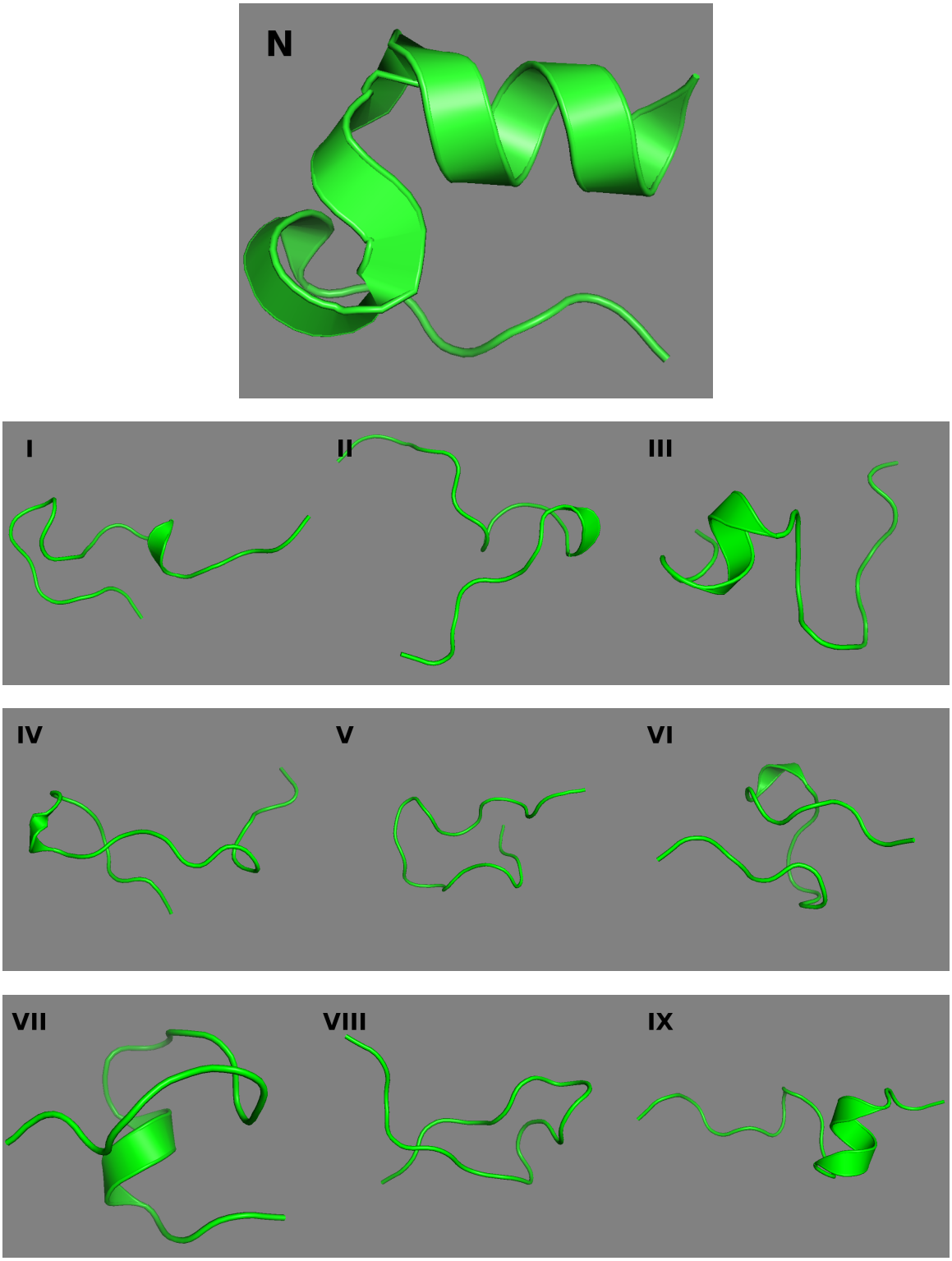
W stanie VII znajduje się 7,2% struktur użytych w analizie i jest to trzeci najliczniejszy stan. Struktura go reprezentująca posiada jedną helisę α odpowiadającą drugiej



Rysunek 9.18. Graf wynikowy dla białka o ID 1L2Y symulowanego w programie AMBER dla przejść w temperaturze 310K. Obok każdego numeru struktury w indeksie górnym znajduje się jej RMSD względem struktury natywnej, a w indeksie dolnym jej współczynnik żyroskopowy. Stany osiągane przez repliki jako pierwsze są poprzedzone literą S w indeksie dolnym, a liczba tych replik znajduje się w tabeli 9.21. Powiększone modele struktur znajdują się na rysunku 9.20.



Rysunek 9.19. Wykresy kołowe populacji największych grup w poszczególnych temperaturach dla symulacji struktury o ID 1L2Y przeprowadzonej w pakiecie AMBER. W lewym dolnym rogu każdego wykresu znajduje się model struktury której dany wykres dotyczy. Powiększone modele struktur znajdują się na rysunku 9.20.



Rysunek 9.20. Modele struktury natywnej oraz struktur centralnych największych grup dla symulacji struktury o ID 1L2Y przeprowadzonej w pakiecie AMBER. Pochodzą one z grafu na rysunku 9.18.

Stan	N	I	II	III	IV	V	VI	VII	VIII	IX
Liczba replik	3	7	14	7	7	2	1	3	1	19

Rysunek 9.21. Tabela przedstawiająca pierwsze osiągnięte przez poszczególne repliki stany wchodzące w skład modelu Markova pochodzące z symulacji struktury o ID 1L2Y przeprowadzonej w pakiecie AMBER.

połowie α_1. Jej struktura trzeciorzędowa jest bardzo kompaktowa, ale kształtem raczej nie przypomina struktury natywnej. Reszty aminokwasowe tworzące strukturę C-końca znajdują się blisko siebie, ale ich łańcuchy boczne są inaczej ułożone. 66% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co pokazuje że ten stan jest tylko trochę mniej stabilny niż natywny.

Stan I obejmuje około 2,8% struktur użytych w obliczeniach. Struktura go reprezentująca posiada jedną bardzo krótką helisę α, odpowiadającą środkowi α_1. Struktura C-końca jest nieobecna, jej reszty aminokwasowe są oddalone od siebie. Strukturą trzeciorzędową przypomina natywną, ale jej N-koniec jest bardziej rozciągnięty. 40,5% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co pokazuje że ten stan jest dość mało stabilny.

Stan IV obejmuje około 1,9% struktur użytych w obliczeniach. Struktura go reprezentująca posiada jedną krótką helisę α będącą fragmentem α_2. Struktura C-końca jest nieobecna, reszty ją tworzące są w dużym oddaleniu a ich łańcuchy boczne skierowane są w przeciwne strony. Jej struktura trzeciorzędowa przypomina stan I ze zgiętym pod kątem 90° N-końcem. 43% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co pokazuje że ten stan jest dość mało stabilny.

Stan VI obejmuje około 7,4% struktur i jest to drugi najliczniejszy stan. Przechodzi dość często do i ze stanu natywnego. Struktura go reprezentująca posiada jedną krótką helisę α będącą fragmentem α_2. Struktura C-końca jest nieobecna, reszty ją tworzące są w dużym oddaleniu od siebie. Jej struktura trzeciorzędowa jest bardzo kompaktowa, ale kształtem raczej nie przypomina struktury natywnej. 71% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej) co wskazuje, że dolina energetyczna z nim związana jest głęboka i sprawia, że jest to najbardziej stabilny uzyskany stan.

Stan II jest stanem pośrednim łączącym wiele innych stanów. Zawiera on około 1,8%

struktur. Struktura go reprezentująca, podobnie jak poprzednie, posiada jedną krótką helisę α będącą fragmentem α_2. Struktura C-końca jest nieobecna, reszty ją tworzące są w dużym oddaleniu a ich łańcuchy boczne skierowane są na zewnątrz. Jej struktura trzeciorzędowa przypomina literę 'L' i zawiera mały, złożony z reszt aminokwasowych od 7 do 16, ciasno zwinięty rdzeń na zgięciu. 36% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co pokazuje że ten stan jest mało stabilny.

Stany V i VIII są do siebie dość podobne i razem ze stanem VII tworzą trójkę często przechodzącą w siebie nawzajem. Zawierają odpowiednio 1,9% i 2% struktur użytych w analizie skupień. Nie posiadają one ani helis α ani łańcuchów β. Nie posiadają też zwiniętej struktury na C-końcu. W strukturze trzeciorzędowej posiadają zwrot o 180° i dwa ramiona, które początkowo biegną równolegle do siebie, częściowo skręcając się i oddalając na końcach cząsteczki. Sprawia to, że przypominają nieco strukturę spinki do włosów. Odpowiednio 39% i 56% struktur w tych grupach znajduje się w temperaturze fizjologicznej i niższej (310K i mniej). Pokazuje to, że stan V jest mało stabilny, a stan VIII zauważalnie stabilniejszy.

Stan III jest kolejnym stanem pośrednim. Obejmuje on około 2% struktur. Struktura go reprezentująca posiada jedną krótką helisę α, odpowiadająca drugiej połowie α_1. Posiada częściowo zwinięta strukturę na C-końcu. Jej struktura trzeciorzędowa posiada dość zwarty rdzeń złożony z reszt aminokwasowych 6-12 i ogólnym kształtem przypomina liczbę '3'. 35% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co pokazuje że ten stan jest mało stabilny.

Stan IX jest najbardziej rozwiniętym stanem i zawiera około 3% struktur. Struktura go reprezentująca posiada jedną krótką helisę α, odpowiadająca drugiej połowie α_1. Nie posiada struktury na C-końcu. Jego struktura trzeciorzędowa jest liniowa z jednym bardziej zwartym fragmentem złożonym z reszt aminokwasowych 6-11. 33% struktur w tej grupie znajduje się w temperaturze fizjologicznej i niższej (310K i mniej), co pokazuje że ten stan jest mało stabilny.

Tabela pierwszych osiąganych stanów (9.21) pokazuje, że wszystkie stany były osiągane jako pierwsze przez repliki. Jest to istotne zwłaszcza dla struktury natywnej, którą

osiągnęły jako pierwszą 3 repliki. Wskazuje to na istnienie innych możliwych ścieżek zwijania, których nie obejmuje niniejszy model Markova. Pozostałe stany pokazują, że istnieje rozległa sieć dodatkowych mechanizmów które doprowadzają rozwiniętą strukturę do poszczególnych stanów modelu Markova. Liczbą replik wyróżniają się stany II i IX, które jako pierwsze osiągnęła około połowa replik.

Uzyskany model sieci stanów struktury białka pokazuje zarówno różne możliwe ścieżki zwijania, jak i częściowe rozwijanie struktury natywnej w sieć form pośrednich, zgodnie z przyjętą hipotezą badawczą. Uzyskane wyniki z pola sił AMBER nie wskazują jednak tak jasno możliwych dróg zwijania tego białka jak uzyskane w programie UNRES. Stany I, V i VIII sugerują, że cząsteczka najpierw przyjmuje kształt przypominający strukturę trzeciorzędową, a tworzenie helis α i struktury na C-końcu zachodzi jednocześnie i bardzo szybko pod koniec procesu zwijania. Ten proces mógłby być jedną ze ścieżek zwijania białka. W różnych stanach istnieją różne fragmenty struktur α, ale jest ich niewiele i ich zawartość nie zwiększa się w stanach bliżej związanych ze strukturą natywną. Z drugiej strony w kilku bardziej rozwiniętych strukturach zauważyć można było mocniej zwinięty rdzeń obejmujący reszty aminokwasowe od około 6 do około 12. Być może odgrywa on rolę w procesie zwijania stabilizując cząsteczkę i promując dalsze jej zwijanie. Wymagałoby to dalszej analizy. Może też reprezentować kolejny mechanizm zwijania. Struktury reprezentujące stany VI i VII są najbardziej zwarte i mają współczynnik żyroskopowy podobny do struktury natywnej. Pozostałe struktury mają wyższe wartości tego współczynnika.

9.3.3. 2MQ8

Na rysunku 9.22 znajduje się graf przejścia dla symulacji białka o ID 2MQ8 w programie AMBER. Do jego przygotowania wykorzystałem tylko przejścia pomiędzy strukturami w temperaturze 310K. Jego macierz przejścia znajduje się poniżej. Jest ona bliska symetrycznej, co wskazuje na prawidłowe przygotowanie modelu Markova. Na rysunku 9.23 znajdują się wielkości populacji struktur w poszczególnych temperaturach w grupach wykorzystanych do stworzenia Modelu Markova. Na rysunku 9.24 znajdują się modele struktury natywnej oraz centrów największych uzyskanych grup. Tabela

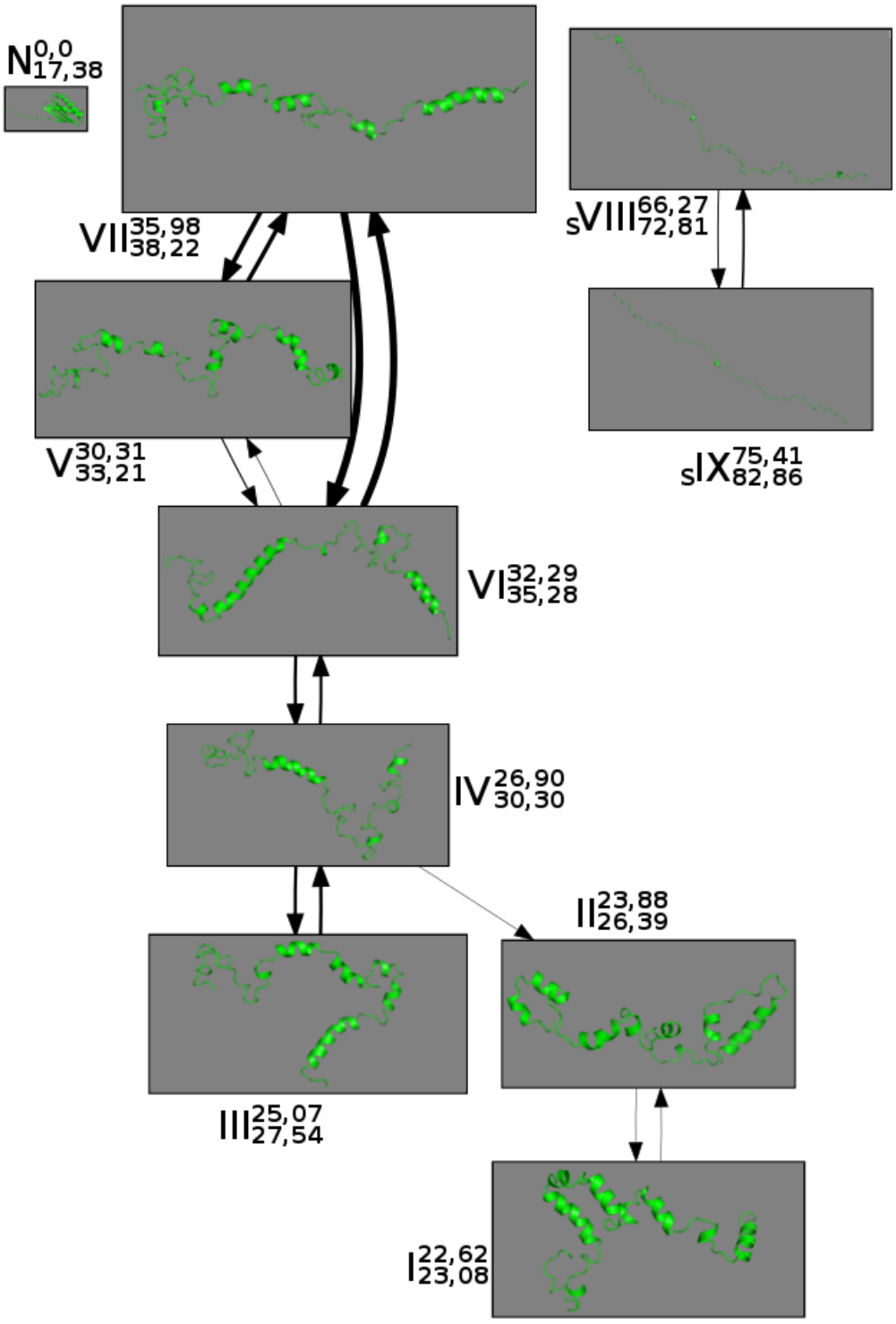
9.25 przedstawia liczbę replik które osiągnęły poszczególne stany modelu Markova jako pierwsze w czasie tej symulacji.

Z\Do	**N**	**VII**	**VIII**	**III**	**V**	**I**	**VI**	**II**	**IX**	**IV**
N	**0**	0	0	0	0	0	0	0	0	0
VII	0	**1987**	0	0	19	0	32	0	0	0
VIII	0	0	**1309**	0	0	0	0	0	4	0
III	0	0	0	**1039**	0	0	0	0	0	13
V	0	17	0	0	**1906**	0	5	0	0	0
I	0	0	0	0	0	**345**	0	1	0	0
VI	0	30	0	0	2	0	**1717**	0	0	12
II	0	0	0	0	0	1	0	**439**	0	0
IX	0	0	8	0	0	0	0	0	**1068**	0
IV	0	0	0	12	0	0	11	1	0	**1316**

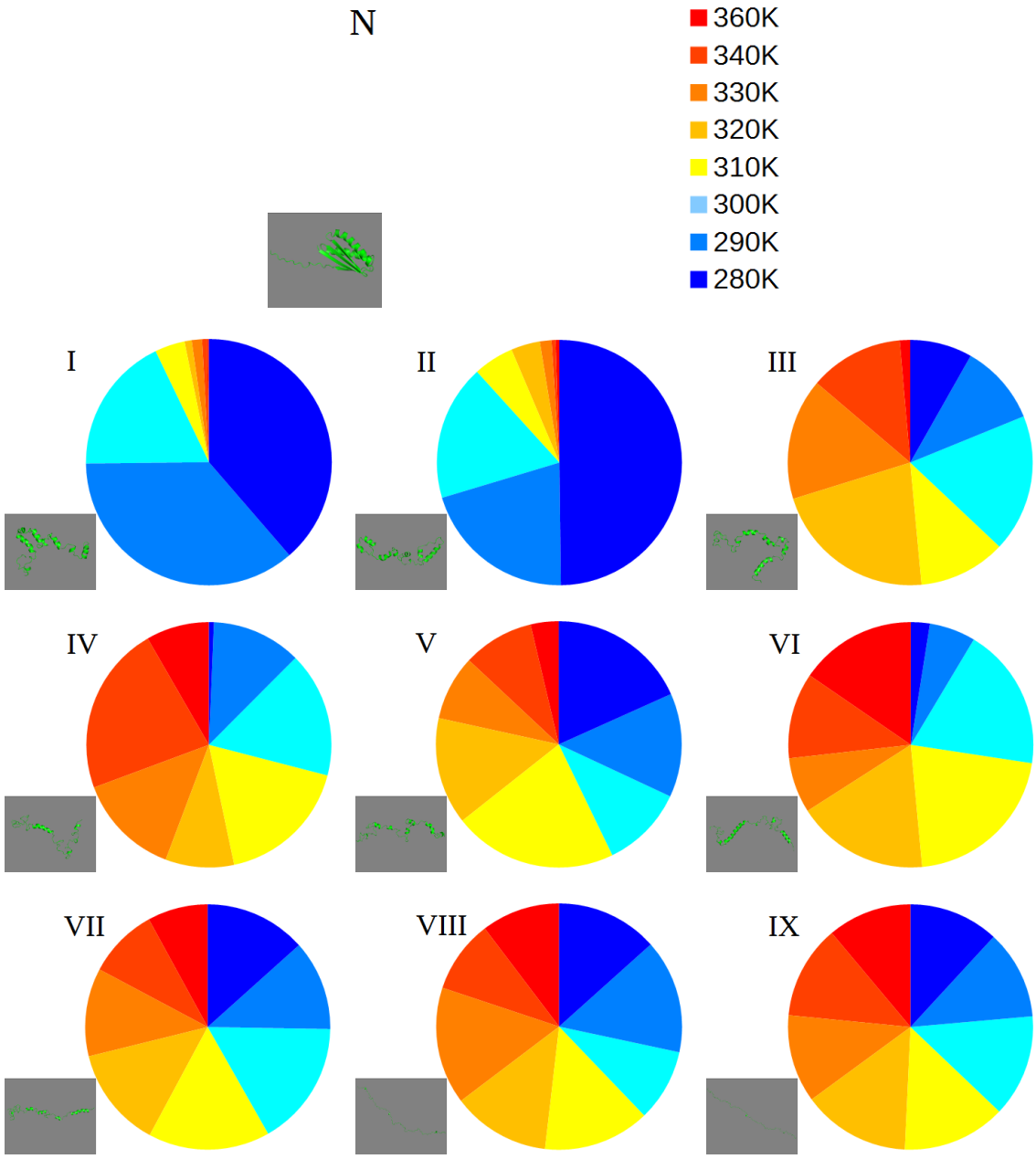
Sam graf przejścia nie jest spójny, co dyskwalifikuje uzyskany model. Można na podstawie jego największego fragmentu próbować wyciągać wnioski na temat początku procesu zwijania, ale jest to ryzykowne. Grupa zawierająca strukturę natywną jest pusta, tak jak oczekujemy na podstawie analizy RMSD opisanej powyżej. Jest to największa analizowana struktura i najprawdopodobniej czas symulacji był zbyt krótki, żeby układ osiągnął stan równowagi i zaszło całkowite zwinięcie w strukturę bliską natywnej. Przeprowadzenie analizy przejść we wszystkich temperaturach nie zmieniło tego obrazu. Graf wciąż był niespójny.

Stan	N	I	II	III	IV	V	VI	VII	VIII	IX
Liczba replik	0	0	0	0	0	0	0	0	5	59

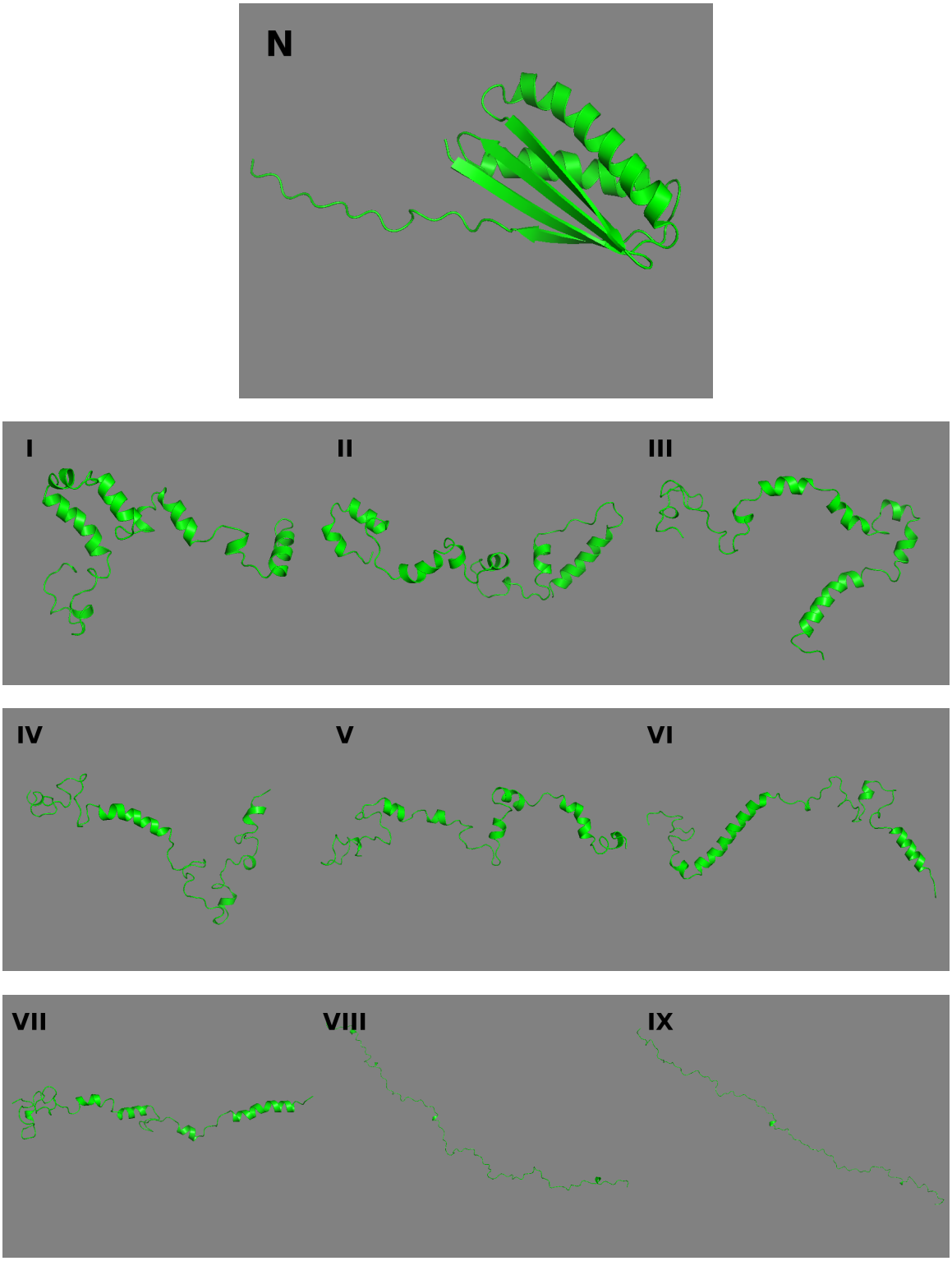
Rysunek 9.25. Tabela przedstawiająca pierwsze osiągnięte przez poszczególne repliki stany wchodzące w skład modelu Markova pochodzące z symulacji struktury o ID 2MQ8 przeprowadzonej w pakiecie AMBER.



Rysunek 9.22. Graf wynikowy dla białka o ID 2MQ8 symulowanego w programie AMBER dla przejść w temperaturze 310K. Obok każdego numeru struktury w indeksie górnym znajduje się jej RMSD względem struktury natywnej, a w indeksie dolnym jej współczynnik żyroskopowy. Stany osiągane przez repliki jako pierwsze są poprzedzone literą S w indeksie dolnym, a liczba tych replik znajduje się w tabeli 9.25. Powiększone modele struktur znajdują się na rysunku 9.24.



Rysunek 9.23. Wykresy kołowe populacji największych grup w poszczególnych temperaturach dla symulacji struktury o ID 2MQ8 przeprowadzonej w pakiecie AMBER. W lewym dolnym rogu każdego wykresu znajduje się model struktury której dany wykres dotyczy. Grupa struktury natywnej jest pusta, stąd brak dla niej wykresu. Powiększone modele struktur znajdują się na rysunku 9.24.



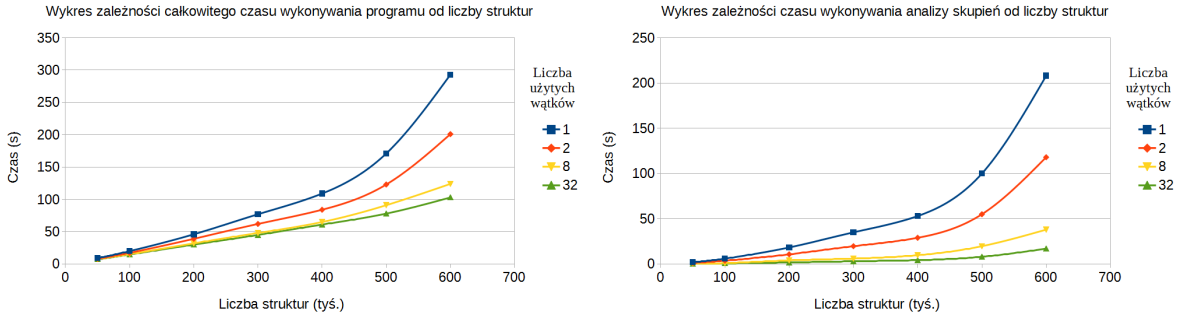
Rysunek 9.24. Modele struktury natywnej oraz struktur centralnych największych grup dla symulacji struktury o ID 2MQ8 przeprowadzonej w pakiecie AMBER. Pochodzą one z grafu na rysunku 9.22.

10. Analiza wydajności programu pdbclust

Analizę wydajności programu pdbclust przeprowadziłem na komputerze dysponującym procesorem AMD Ryzen Threadripper 1950X. Posiada on 16 fizycznych rdzeni, które dzięki technologii SMT (Simultaneous Multithreading) traktowane są jako 32 rdzenie przez system operacyjny. Technologia ta pozwala uzyskać nieco większą wydajność przy tej samej liczbie rdzeni[106]. Każdy pomiar wykonałem trzykrotnie i obliczyłem z nich średnią.

Na rysunku 10.1 znajdują się wykresy zależności czasu wykonywania całego programu (po lewej) i samej analizy skupień (po prawej) od liczby struktur dla różnej liczby użytych w obliczeniach rdzeni. Analizę tą przeprowadziłem przy użyciu 50, 100, 200, 300, 400, 500, 600 tysięcy struktur oraz 1, 2, 8 i 32 wątków obliczeniowych. Użyte dane wejściowe pochodziły z opisywanej w niniejszej pracy symulacji struktury o kodzie ID 2MQ8 w programie UNRES. Na wykresach nie umieściłem czasu przygotowania modelu Markova, gdyż proces ten jest bardzo szybki i dla opisanych przypadków zajmował co najwyżej sekundę. W przebiegu programu prawie cały czas nie przeznaczony na analizę skupień zajmowało wczytywanie danych. Na wykresach tych widać, że złożoność obliczeniowa algorytmu grupowania znajduje się pomiędzy $O(n)$ a $O(n^2)$. Jest to zgodne z naszymi oczekiwaniami. W zastosowanym w programie algorytmie nie obliczamy RMSD wszystkich możliwych par struktur, a tylko te, które mogą należeć do badanej grupy. Wartości graniczne pokazują dwa zdegenerowane przypadki. $O(n)$ oznacza że wszystkie struktury znalazły się w jednej grupie, a $O(n^2)$ że każda struktura znalazła się w osobnej grupie.

Efektywność zrównoleglenia E_p jest procentową miarą oznaczającą zysk na czasie wynikający z przeprowadzenia obliczeń równolegle zamiast sekwencyjnie. 100% oznacza perfekcyjne zrównoleglenie, w którym podwojenie ilości użytych rdzeni zmniejsza



Rysunek 10.1. Wykresy zależności czasu wykonywania całego programu (po lewej) i samej analizy skupień (po prawej) od liczby struktur dla różnej liczby użytych rdzeni.

o połowę czas potrzebny na obliczenia. Wartości mniejsze niż 100% oznaczają mniejszą wydajność. Oblicza się ją ze wzoru:

$$E_p = \frac{S(p)}{p} \qquad S(p) = \frac{T_s}{T_p(p)}$$

Gdzie

E_p - efektywność zrównoleglenia na p rdzeniach

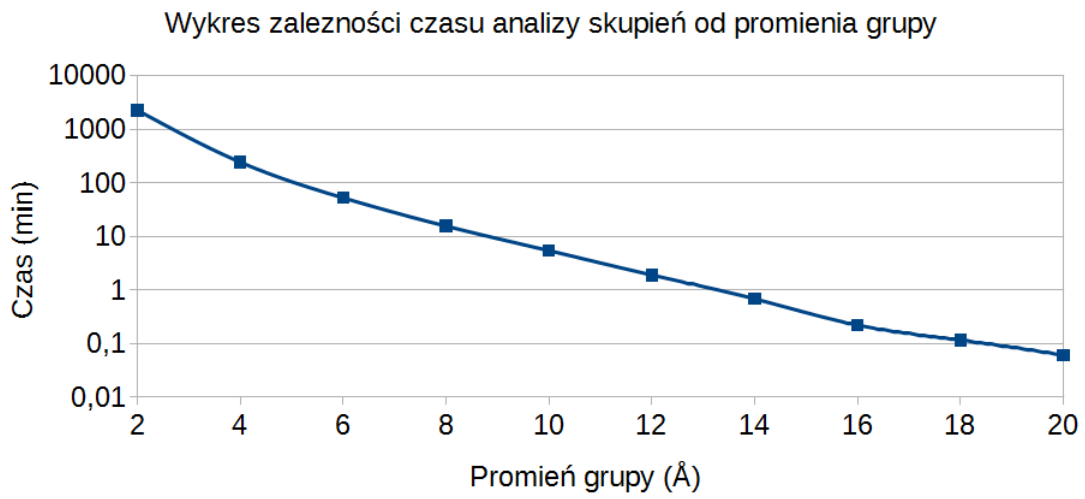
$S(p)$ - przyspieszenie obliczeń

p - liczba procesorów użytych do obliczeń

T_s - czas potrzebny na wykonanie obliczeń sekwencyjnie

$T_p(p)$ - czas potrzebny na wykonanie obliczeń równolegle na p rdzeniach[107]

W programie pdbclust dla analizy skupień średnia efektywność zrównoleglenia z użytych punktów danych wynosi 86% dla $p = 2$ i 63% dla $p = 8$. Dla 32 wątków wartość ta wynosi 67% jeśli przyjmiemy $p = 16$ i 33% jeśli przyjmiemy $p = 32$. Otrzymane wartości są zgodne z oczekiwaniami. Mimo, że dzielimy struktury pomiędzy wątki po równo nie gwarantujemy w ten sposób, że każdy z nich będzie musiał wykonać tyle samo obliczeń RMSD, gdyż część struktur może już znajdować się w innych grupach i mieć na tyle mały RMSD względem ich centrów, że jego obliczanie będzie pomijane. W związku z tym niektóre wątki skończą swoją pracę szybciej i będą musiały czekać na pozostałe. dla pełnego czasu wykonywania programu efektywność zrównoleglania wynosi 63% dla $p = 2$, 20% dla $p = 8$, 11% dla $p = 16$ i 6% dla $p = 32$. Tak duże zmniejsze-

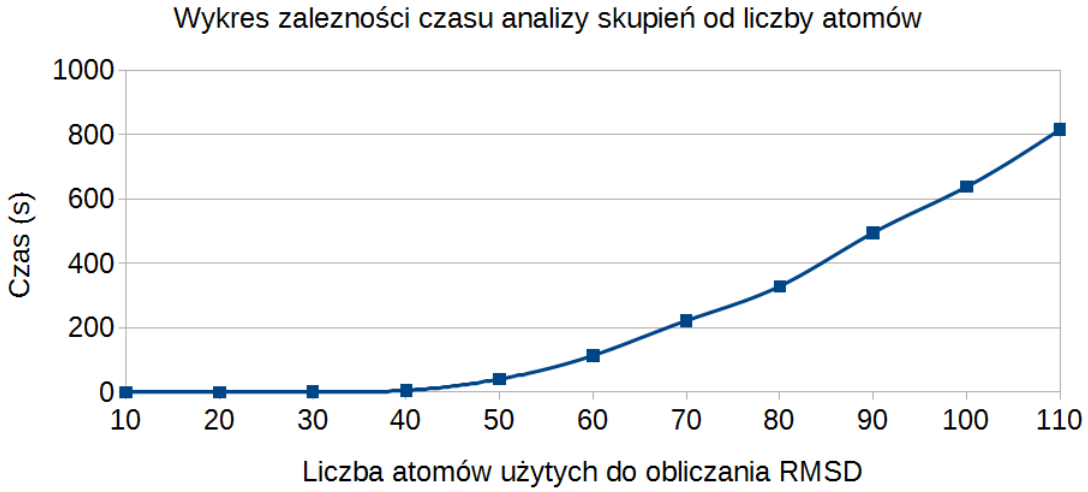


Rysunek 10.2. Wykres zależności czasu wykonywania analizy skupień od promienia grupy.

nie efektywności zrównoleglania wynika ze wspomnianego dużego czasu potrzebnego na wczytanie danych, który jest niezależny od liczba wykorzystanych procesorów.

Na rysunku 10.2 znajduje się wykres zależności czasu wykonywania analizy skupień od promienia grupy. Analizę tą przeprowadziłem używając 600 tyś. struktur i 4 wątków obliczeniowych. Użyte dane wejściowe pochodziły ponownie z opisywanej w niniejszej pracy symulacji struktury o kodzie ID 2MQ8 w programie UNRES. Widać na nim tendencję wzrostową czasu potrzebnego na wykonanie analizy skupień wraz ze spadkiem użytego promienia grupy, co wskazuje na wzrastająca liczbę obliczanych RMSD wraz ze spadkiem promienia grupy. Zjawisko to jest zgodne z oczekiwaniami, gdyż w przypadku większych promieni więcej struktur może mieć na tyle mały RMSD względem centrum grupy, że jego obliczanie będzie pomijane dla innych kandydatów na centra grup.

Na rysunku 10.3 znajduje się wykres zależności czasu wykonywania analizy skupień od liczby atomów użytych do obliczania RMSD. Analizę tę przeprowadziłem używając 600 tyś. struktur, promienia grupy równego 8Å i 4 wątków obliczeniowych. Użyte dane wejściowe pochodziły ponownie z opisywanej w niniejszej pracy symulacji struktury o kodzie ID 2MQ8 w programie UNRES. Widać na nim tendencję wzrostową, szybszą niż liniowa, czasu potrzebnego na wykonanie analizy skupień wraz ze wzrostem liczby atomów użytych do obliczania RMSD. Zjawisko to jest zgodne z oczekiwaniami.



Rysunek 10.3. Wykres zależności czasu wykonywania analizy skupień od liczby atomów użytych do obliczania RMSD.

Większa liczba atomów sprawia, że obliczanie każdego RMSD pary struktur staje się dłuższe. Dodatkowo mniejsze (pod względem liczby atomów) struktury są do siebie bardziej podobne i mogą częściej znajdować się bliżej centrów grup, co zmniejsza liczbę obliczanych RMSD.

Część V

Dyskusja

Uzyskane wyniki w symulacjach które doprowadziły do struktur podobnych natywnym potwierdzają przyjęta hipotezę badawczą. W każdym z nich uzyskany graf przejścia nie jest liniowy, ale tworzy skomplikowaną sieć złożoną z różnych struktur. Na ich podstawie możliwe jest niekiedy podanie pojedynczej, bardzo ogólnej ścieżki zwijania, jak w przypadku struktury o kodzie ID w polu sił UNRES. Jednocześnie widać w nich, że stadiów pośrednich procesu zwijania białek jest wiele i nie istnieje jedna, ściśle określona ścieżka prowadząca do struktury natywnej.

Zaproponowana przeze mnie metoda analizy skupień różni się od standardowych, opisanych w rozdziale 3.3. Celem opisanych w mojej pracy badań było dostosowanie jej do charakterystyki analizowanych danych, w tym przypadku struktur białek i innych biomolekuł. Z tego powodu w analizie skupień przyjąłem jako centra grup struktury o najniższych energiach oraz strukturę natywną. Struktura natywna reprezentuje oczekiwaną, stabilną konformację układu. Struktury o najniższych energiach znajdują się w pobliżu minimów energetycznych i przedstawiają pośrednie, częściowo stabilne stadia fałdowania struktury biomolekuły. Taki sposób wyboru centrów grupy sprawia, że zostają nimi struktury istotne z biochemicznego punktu widzenia. W przeciwieństwie do niej w standardowej implementacji algorytmu najbliższego sąsiada jedynym kryterium wyboru centrum grupy jest liczba sąsiadów. Nie ma gwarancji, że wybrana w ten sposób struktura będzie miała istotne znaczenie biologiczne. W algorytmie k-średnich centrum grupy zostaje struktura będąca średnią ze wszystkich struktur grupy. Taka struktura może nawet nie mieć sensu fizycznego ze względu na, m.in. zawady steryczne[57].

Obliczanie RMSD jest bardzo popularnym sposobem porównywania struktur molekularnych, między innymi ze względu na swoja prostotę i intuicyjność koncepcji. Metoda ta nie jest jednak doskonała, gdyż sprowadza porównywane struktury jedynie do zbioru punktów, tracąc informacje o fizycznych oddziaływaniach i wiązaniach obecnych w cząsteczce. Sprawia to, że istnieją sytuacje w których para struktur o niskim RMSD znacząco różni się od siebie z biologicznego punktu widzenia. Najprostszym przykładem mogłaby być para struktur złożonych z dwóch skrzyżowanych helis α połączonych pętlą, z których jedna jest "na wierzchu" w pierwszej strukturze, a druga

w drugiej. Taka para struktur będzie miała bardzo niski RMSD mimo, że przejście jednej struktury w drugą będzie wymagało znaczących zmian konformacyjnych, ponieważ fragmenty cząsteczki nie mogą przeniknąć przez siebie. Mimo tych problemów metryka ta jest uważana za rozsądny sposób porównywania struktur molekularnych, zwłaszcza między strukturami o niskim RMSD względem siebie[57].

Przyjęcie definicji grupy jako określonej maksymalnej wartości RMSD pomiędzy strukturą centralną a jej członkami jest jedną z prostszych możliwych definicji. Jej wadą jest konieczność ustalenia właściwego promienia grupy, co niestety należy zrobić metodą prób i błędów dla każdej badanej struktury. Przyjęcie zbyt małej wartości spowoduje powstanie dużej liczby małych grup, a powstały na ich podstawie model Markova będzie miał albo zbyt mało przejść albo zbyt dużo stanów, aby być użyteczny w analizie jakościowej. Zbyt duża wartość promienia spowoduje tworzenie grup w których znajdą się konformacje niepowiązane bezpośrednio ze sobą. Alternatywą dla tej metody mogłyby być inne algorytmy, na przykład K-medoidów, które jednak wprowadziłyby charakterystyczne dla siebie problemy, na przykład konieczność ustalenia liczby grup przed przeprowadzeniem analizy skupień[54, 57].

Kolejnym problemem przy tej metodzie jest założenie, że wszystkie struktury o określonym RMSD względem struktury centralnej są z nią blisko powiązane w rzeczywistości. Jak wspomniałem wyżej mogą istnieć sytuacje, w których dwie struktury o niskim RMSD względem siebie mają konformacje znacząco się różniące pod względem biochemicznym. Alternatywą mogłoby być obliczanie RMSD z wykorzystaniem kątów torsyjnych, co może pozwolić uzyskać lepszy podział[108]. Podobnie przyjęcie tego samego promienia dla wszystkich grup jest dużym uproszczeniem wobec skomplikowanej struktury krajobrazu energetycznego cząsteczki[57]. Problemu tego nie da się rozwiązać w jednoznaczny sposób, gdyż pełna analiza krajobrazu energetycznego cząsteczek składających się z więcej niż kilku atomów jest obecnie niemożliwa i każdy zaprojektowany podział będzie uproszczeniem.

W wielu publikacjach opisany jest dwustopniowy podział na grupy. Najpierw tworzone jest wiele (tysiące) mikrostanów które następnie są grupowane przy pomocy analizy przejść pomiędzy nimi do mniejszej liczby makrostanów, zazwyczaj kilkunastu

dla modeli mających przedstawiać układ jakościowo. Metoda ta rozwiązuje częściowo opisane w poprzednim akapicie problemy, ale konieczność przypisania wszystkich struktur do niewielkiej liczby makrostanów może spowodować powstanie obszernych stanów, w których znajduje się wiele słabo związanych ze sobą konformacji[54, 57]. Użycie przeze mnie tylko części struktur, zawierających się w największych grupach i w grupie tworzonej przez strukturę natywną, pozwala pozbyć się tego problemu. Przy założeniu, że użyte grupy zawierają znaczącą większość struktur wykorzystanych w obliczeniach ryzyko pominięcia istotnych grup i przejść jest niewielkie. Czasem stosowana jest również modyfikacja dwustopniowego podziału na grupy, w której tworzy się większą liczbę makrostanów, ale prezentuje się model Markova utworzony tylko z najistotniejszych przejść pomiędzy strukturą rozwiniętą a natywną[58].

W zaproponowanej przeze mnie metodzie po przeprowadzeniu analizy skupień należy wskazać liczbę grup, które zostaną wykorzystane jako stany w konstruowaniu modelu Markova. Z moich obserwacji wynika, że 10 jest w wielu przypadkach rozsądną wartością. Posiadając gotowy podział na grupy tworzenie kolejnych modeli Markova przy użyciu różnych parametrów nie wymaga dużej ilości obliczeń w związku z czym jest bardzo szybkie i można niewielkim nakładem mocy obliczeniowej uzyskać wiele modeli.

Skupiłem się w niniejszej pracy na analizie symulacji typu REMD których charakterystyczną cechą jest użycie kilku różnych temperatur. Aby zwiększyć możliwości analizy uzyskanych danych dla każdej uzyskanej grupy pokazałem rozkład temperatur, w których występują należące do niej struktury. Pozwala to na formułowanie dodatkowych hipotez na temat ich stabilności i roli w symulowanych procesach.

Zdecydowałem się na zdefiniowanie w obliczeniach struktury natywnej, która będzie traktowana w specjalny sposób. Zawsze będzie tworzyła grupę i znajdzie się wśród grup wykorzystanych do budowy łańcucha Markova. Struktura ta nie musi być wprost strukturą natywną badanej molekuły, ale może być dowolną konformacją którą uważamy za interesującą i chcemy zbadać jej obecność, zachowanie i możliwą ścieżkę zwijania w symulacji. To specjalne traktowanie pozwala też na badanie i formułowanie hipotez o użytym polu sił pod względem jego zdolności zwinięcia znanej struktury.

Istotnym zagadnieniem jest czas opóźnienia (ang. *lag time*) użyty w budowie modelu Markova, czyli odstęp pomiędzy parami struktur używanymi do zliczania przejść pomiędzy stanami. Najczęściej stosowany jest stały czas pomiędzy 1 ns a 100 ns. Jako alternatywę dla tej metody zaproponowałem czas krótszy i zmienny w określonych granicach. Zliczane są tylko przejścia pomiędzy grupami użytymi do budowy łańcucha Markova co umożliwia nam zwiększenie liczby tych przejść poprzez ignorowanie mniej istotnych, mało spopulowanych grup które mogły znajdować się w przejściach pomiędzy nimi.

W toku dalszych badań związanych z zaproponowanymi przeze mnie rozwiązaniami można rozważyć przeprowadzenie kolejnej symulacji dynamiki molekularnej używając centrów największych grup jako struktur startowych, co powinno doprowadzić do polepszenia jakości uzyskanego modelu Markova. Innym, potencjalnie interesującym, kierunkiem jest zmiana sposobu liczenia czasu opóźnienia na bliższy standardowemu, przez szukanie przejść nie wprost w kolejnych kilku strukturach, ale po określonym odstępie.

Analiza skupień mogłaby zostać ulepszona przez dodanie stosowanych niekiedy "stref niczyich" znajdujących się blisko granic grup. Przejścia do i z nich nie są zliczane w trakcie tworzenia modelu Markova. Pozwoliłoby to na zredukowanie problemów powodowanych przez skomplikowaną strukturę krajobrazu energetycznego cząsteczki oraz na wyeliminowanie szybkich, płytkich i powtarzalnych przejść przez granice pomiędzy grupami, co mogłoby zwiększyć jakość uzyskanego modelu Markova[57].

Istotną rzeczą, w związku z wzrastającą ilością danych uzyskiwanych z symulacji, jest wydajność zaproponowanej metody. Wyniki zaprezentowane w rozdziale 10 pokazują, że przy pomocy mojej metody możliwa jest analiza setek tysięcy i więcej struktur w czasie rzędu godzin, co jest dobrym wynikiem.

Wyniki uzyskane z obydwu symulacji białka o kodzie ID 2MQ8 sugerują, że czas symulacji mógł być za krótki, żeby większa liczba replik osiągnęła konformacje bliskie natywnej. Białko to było największą symulowaną strukturą. Być może kilkukrotne zwiększenie długości symulacji pozwoliłoby uzyskać lepsze wyniki.

W wyniku niektórych analiz (1L2Y w obu programach) powstały pojedyncze stany w których znajduje się wyraźnie największy procent struktur użytych w grupowaniu.

Kolejnym krokiem badań tego układu mogłoby być rozbicie takiego stanu na mniejsze, na przykład poprzez wyizolowanie należących do niego struktur i przeprowadzenie na nich analizy skupień przy użyciu mniejszego promienia grupy.

Przeprowadzone analizy skłaniają mnie ku stwierdzeniu, że pomyślnie zweryfikowałem postawioną hipotezę badawczą. Z sześciu przebadanych symulacji pięć potwierdza założoną hipotezę, a symulacja która jej nie wspiera (opisałem ją w rozdziale 9.3.3) najprawdopodobniej była zbyt krótka.

Część VI

Dodatki

11. *pdbclust* manual

11.1. Introduction

pdbclust is a computer program designed to perform energy-based clustering of biomolecular structures and transition path analysis of biomolecular simulations. It works on individual structures which form chronological trajectories. After the clustering procedure the Markov chain model is used to analyze transitions between those clusters. In the end a transition matrix of clusters and a simple graphic representation of it are produced. The program accepts input of many trajectories at one time, obtained, for example, from the Replica Exchange Molecular Dynamics simulation, but it can also analyze single trajectories.

This program does not create Markov model from which kinetic data can be established, but rather simple, coarse-grained representation of transitions in a system and their relation with native structure.

11.2. Description

In *pdbclust* the clustering is performed by a simple nearest neighbor algorithm. It is based on RMSD calculation between structures and user can define which atom types should be taken into consideration in those calculations. In its default mode the program uses structures with lowest energies as centers (the so-called kernels) of clusters. Each structure can belong to one cluster only. If information about energy is unavaliable the random clustering may be performed. If structures in trajectory vary by their temperature it is possible to either use all of them or only ones with one selected temperature for clustering. After each structure is assigned to a cluster a user-defined number of largest clusters is used for further calculations.

Markov chain model of transitions between those clusters is created based on trajectory data. Structures are analyzed chronologically. Transition between clusters is recorded when a structure belonging to one cluster transitions into another one within user-defined number of steps (one structure in trajectory is one step). It is possible to create the Markov model only from structures with given temperature or all of them.

Next a transition matrix is prepared based on this Markov model and in the end a *graphviz* input script is created which can be later customized by user and used to create a simple graphic representation of a graph depicting the largest clusters and transitions between them.

It is possible to define and load a native structure. It will always get the chance to create cluster first and it will be used in Markov Model creation.

This program uses the *fitsq* subroutine written by Dr. Kenneth D. Gibson, Cornell University, taken from UNRES software package.

11.3. Installation

pdbclust is being distributed in the form of a source code package. It was written and compiled under *Linux* operating system and requires OpenMP library (for example GCC OpenMP support library). The Makefile is provided. To compile program extract its archive and use the terminal to input *make* command. The executable file, called *pdbclust* is created inside the *bin* folder.

A short, simple test packet is provided. To run it enter the command *pdbclusttester* from the *bin* folder (after successful compilation of a program).

11.4. Usage

pdbclust should be run from terminal and requires one argument - a name of the input script file. A second argument can be provided which represents a name of a file where the program log will be written. If second argument is not provided default value of *log.txt* is used. For example:

```
pdbclust InputScript.txt my_log.txt
```

11.5. The input and output file formats

pdbclust makes use of several different file types, both as its inputs and outputs. Generally they belong to two categories. Output files from MD simulations:

— UNRES output files

— multiPDB files

— native structure PDB files

— AMBER output files

— custom data files

And files generated by pdbclust itself:

— kernels information files

— structure information files

— transition matrix files

— graphviz script files

— log files

They are all described in detail below.

11.5.1. The accepted output files from MD simulations

UNRES output files - UNRES output is written as the PDB formatted files containing many structures each and the special REMARK record which contains energy and temperature information for every structure. The individual structures are separated with ENDMDL record line.

PDB files - A lot of simulation software writes its structure output in its own file formats (e.g. AMBER mdcrd trajectory file) which can be later converted to a more standard form (PDB files). *pdbclust* can read structures from those standard PDB

files. The coordinates are read from ATOM records and individual structures are separated with ENDMDL record lines.

native structure PDB file - The native structure has to be provided as a plain PDB file containing a single model of a structure with no alternative atom locations. It can be ommited if it is not avaliable or user does not want to use it. It is recommended to use minimized or even minimized and heated structure, because model obtained directly from e.g. crystallography may be significantly different from structures in simulation trajectory and fail to create a native structure cluster of any noticeable size.

AMBER output files - AMBER is one of the most widely used computer simulation program. To help with analysis of its data *pdbclust* can read energy and temperature of structures directly from AMBER mdout files. The program expects that each record in mdout file has a corresponding structure in trajectory file and that overall numbers of records are identical. Both kinds of files are read sequentially and it is up to a user to make sure they correspond to each other.

custom data files - Like with structure files a lot of simulation software writes its energy and temperature output in its own file formats. *pdbclust* also offers possibility to read such data. It needs to be converted to a simple format which contains energy (as a floating point number) and temperature (as integer, if avaliable) data of each structure in a single line. The format of those files is as follows:

```
E: {value of energy} T: {value of temperature}
```

If temperature information is unavaliable it can be ommited as follows:

```
E: {value of energy}
```

For example:

```
E: 1.25 T: 300
E: 5.65 T: 270
E: 3.845 T: 270
E: -10.235 T: 270
```

11.5.2. Files produced by *pdbclust*

kernels information file - Contains information about clusters created by a program after performing clustering. It contains all clusters equal or larger than the value of **clustersizefilter** parameter. Clusters are written in descending order by their populations. If two (or more) have identical populations then ones with lower energy kernel structure are written first. For each cluster the following data is given:

— kernel structure ID,

— number of structures in cluster,

— energy of kernel structure,

— temperature of kernel structure,

— name of file containing kernel structure and its number in it,

— how many structures in each temperature belongs to this cluster (temperature populations),

— types and coordinates of atoms used during clustering.

structure information file - Contains information about every structure read by program after performing clustering: ID of the structure and an ID of a cluster it belongs to (ID of a cluster is equal to an ID of its kernel), the trajectory number, RMS value relative to a center of cluster, energy and temperature (if avaliable). One line for each structure. Structures are written in an order they were read from trajectory files.

transition matrix file - Contains transition matrix generated from Markov chain in the form of a table. The maximum size (number of clusters used to create it) of this matrix is defined by the **markovnumberofclusters** option. Less clusters will be taken if there are not enough clusters that satisfy value of **clustersizefilter** option. The first column contains IDs of clusters we are transitioning from, while first row contains IDs of clusters we are transitioning to. Both are written in descending order by their populations. The rest are number of transitions between those kernels in given direction. For example value in third column and second row represents the number of transitions from the first most populous cluster to the second most

populous cluster. Note that this is a modification of standard transition matrix: values in it represent number of transitions instead of probabilities.

graphviz script file - Contains input script for Graphviz, which allows user to quickly create a simple graphical representation of the results of the program. The size of nodes in dependent on cluster population while thickness of edges is dependent on number of transitions between clusters in a given direction. There are some customization options avaliable (see below).

log file - Contains program log, it includes the more detailed information than what is being written to console window (stdout).

11.6. Input script

11.6.1. Line format

Input script is a text file. Each line of input script can contain either single command, comment, or be empty. Most command consist of command name, followed by equals sign ('='), followed by parameter(s). For example:

```
inputmode = unres
```

or

```
clustersizefilter = 10
```

Note space characters surrounding '='. The empty lines and lines starting with '#' character are treated as comments and ignored. The order of commands is not relevant except when defining trajectories (see below) and data files. In the list below commands are bolded while value type or list of supported values is in italics inside the parentheses.

11.6.2. The main commands

command [*fullrun, cluster, markov, visualize*] default: none

Defines what will be done during a given program run. Each run creates a log

file. The required input and created output files are listed in descriptions of each command.

fullrun means that the full program run, from clustering, through Markov model creation to writing a visualization script will be executed. This command requires a *trajectory*, *data* (unless using UNRES trajectory), and *native structure* (if present) files as inputs and it outputs the *kernels information*, *structure information*, *transition matrix* and *graphviz script* files.

cluster means that only the clustering will be performed. This command requires a *trajectory*, *data* (unless using UNRES trajectory) and *native structure* (if present) files as inputs and it outputs the *kernels information* and *structure information* files.

markov means that only creation of Markov model will be performed. This command requires a *kernels information* and *structure information* files as inputs and it outputs the *transition matrix* file.

visualize means that only creation of Graphviz script will be performed. This command requires a *kernels information* and *transition matrix* files as inputs and it outputs the *graphviz script* file.

inputmode [*amber, unres, custom*] default: custom

Defines format of the input. Files coming from the UNRES run can be read directly. Files from AMBER run can be used after converting them into the multiPDB file format and making sure the data in AMBER out files is appropriate as described above and **temperaturelist** is set correctly. The files from other programs should be converted into the multiPDB formatted files while the energy and temperature information should be provided in a custom data file (described above).

runmode [*sequential, parallel*] default: sequential

Defines whether program will be run sequentially or in parallel.

threadnumber [*positive integer*] default: 0

Defines how many threads should parallel portions of the program use. Default value, '0' lets OpenMP library decide.

11.6.3. File names and options

datafile [*text*] default: none

Defines the name of a file containing temperature and energy data (either *AMBER output files* or *custom information files*). It can appear multiple times but needs to be in correct order sequentially, so that each structure in trajectories will get correct values. It is used only as input file for **fullrun** and **cluster** program runs.

graphvizscript [*text*] default: script.gv

Defines the name of a *graphviz script file*. It is used only as output file for **fullrun** and **visualize** program runs.

kernelsfile [*text*] default: kernels.txt

Defines the name of a *kernels information file*. It is used as output file for **fullrun** and **cluster** and as input for **markov** and **visualize** program runs.

nativefile [*text*] default: none

Defines the name of a PDB file containing the *native structure*. It can be ommited if it is not avaliable or user does not want to use it. If present it will always be used as a kernel of cluster and is guaranteed to appear in transition matrix. Used only as input file for **fullrun** and **cluster** program run.

structureinfofile [*text*] default: structureinfo.txt

Defines the name of a *structure information file*. It is used as output file for **fullrun** and **cluster** and as input for **markov** program runs.

trajfile [*text*] default: none

Defines the name of a file containing structures from simulations (either *UNRES output files* or *multi PDB files*). It can appear multiple times. Each file is treated as a separate trajectory by default. If a single molecular dynamics trajectory is

contained in multiple files then use the **trajstart** and **trajend** options described below. It is used only as input files for **fullrun** and **cluster** program runs.

trajstart, trajend

Defines beginning and end of definition of molecular dynamics trajectory contained in several files. The files between those two commands are read sequentially as a single trajectory. Those files are defined using **trajfile** commands. It can appear multiple times. It is used only for **fullrun** and **cluster** program runs.

transitionmatrixfile [*text*] default: transitionmatrix.txt

Defines the name of a *transition matrix file*. It is used as an output file for **fullrun** and **markov** and as input for **visualize** program runs.

11.6.4. The clustering options

clustermode [*energy, random*] default: energy

Defines how clustering will be performed. If set to *energy* then the kernels of clusters will be chosen based on their energy. If set to *random* then they will be determined randomly. In this mode the Energy and Temperature are not read at all and the data files for AMBER and Custom *inputmode* are not read.

clusterradius [*positive integer*] default: 4

Defines the radius of a cluster in Angstroms. Only structures with lower or equal RMSD to a kernel of a cluster belongs to this cluster. In case two kernels are within this radius given structure belongs to closer cluster (with the „distance" defined by the RMSD value).

clustersizefilter [*positive integer*] default: 20

Defines the smallest size of a cluster which will be considered for Markov model and transition matrix creation. The size of a cluster is defined by the number of structures it consists of.

clustertemperature [*positive integer*] default: 300

Defines a single temperature (in K) of structures which will be used for clustering.

Other structures will be ignored and will not even be loaded by the *pdbclust*. See **temperaturelist** option to see how to make it work with data from AMBER or other sources, where temperature can fluctuate.

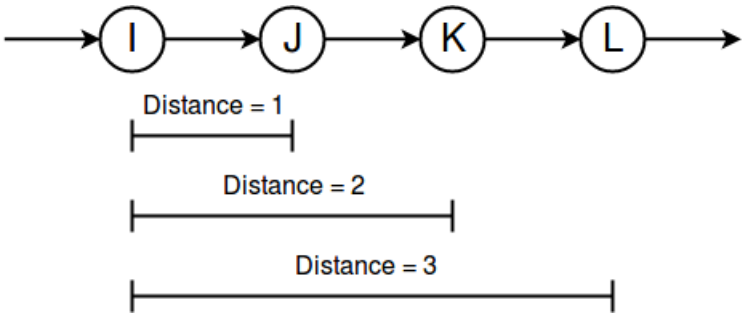
clusterusealltemperatures [*true, false*] default: false

If set to *true* the program will use structures with any temperature to perform clustering. If set to *false* then the value set by **clustertemperature** option will be used.

11.6.5. Markov model options

markovlookuprange [*positive integer*] default: 1

Defines a maximum distance: number of structures, located further in trajectory from currently analyzed structure. The program will check that many structures to determine whether current structure transitions into any cluster used for Markov model building. First transition to appropriate cluster will be counted. How distance is counted is shown on a figure below.



markovnumberofclusters [*positive integer*] default: 10

Defines maximum number of clusters which will be used for creating Markov model. Less clusters will be taken if there are not enough clusters that satisfy the value of **clustersizefilter** option.

markovtemperature [*positive integer*] default: 300

Defines a single temperature (in K) of structures which will be used for Markov model creation. Other structures will be ignored. Together with *clusterusealltemperatures* option it allows user to perform clustering on all structures and then

Markov model creation only on subset of them. Note that this does not change clusters which will be used, just how transitions between them will be counted. See **temperaturelist** option to see how to make it work with data from AMBER or other sources, where temperature can fluctuate.

markovusealltemperatures [*true, false*] default: false

If set to *true* program will use structures with any temperature to perform Markov model creation. If set to *false* then the value set by **markovtemperature** option will be used.

11.6.6. Graphviz script options

graphvizignoreloops [*true, false*] default: false

Defines whether self-loop edges should be written to Graphviz input script. They tend to dominate other values and ignoring them may produce visually clearer output.

graphvizmaxcutoff [*positive integer*] default: 0

Defines the lowest value from transition matrix that will be written to Graphviz input script. Lower values will be ignored and their edges will not appear.

graphvizmincutoff [*positive integer*] default: 0

Defines the highest value from transition matrix that will be written to Graphviz input script. Higher values will be reduced to it.

graphvizscalevalues [*true, false*] default: false

Defines whether values of node box size and edge width in Graphviz script should be scaled. The program uses inverse power scale. Quadratic for node box size and cubic for edge width. This helps to mitigate domination of largest values.

11.6.7. Other options

atomtypes [*list of space separated identifiers*] default: CA

Defines list of atom names (types) which will be read from trajectory structures by *pdbclust* and used for calculating RMSD. For example:

```
atomtypes = CA CB N O
```

readtemperature [*true, false*] default: true

Defines whether the program should attempt to read temperatures at all. Used when data about temperatures is unavaliable.

temperaturelist [*list of space separated integers*] default: none

Defines list of temperature points to use with AMBER trajectories. AMBER allows temperatures to fluctuate in certain range. Temperature of each structure will be changed to the closest one defined in this list for all actions performed by the program. It will be used by custom input if defined and will be ignored in UNRES mode. For example:

```
temperaturelist = 270 285 300 320 340
```

11.7. Example input scripts

1. Full program run using mostly default values, with native structure and 10 different trajectories from UNRES, each in separate file:

```
command = fullrun
runmode = parallel
inputmode = unres
nativefile = 1DIV.pdb
trajfile = trajectory_0.pdb
trajfile = trajectory_1.pdb
trajfile = trajectory_2.pdb
trajfile = trajectory_3.pdb
trajfile = trajectory_4.pdb
trajfile = trajectory_5.pdb
trajfile = trajectory_6.pdb
trajfile = trajectory_7.pdb
```

```
trajfile = trajectory_8.pdb
trajfile = trajectory_9.pdb
```

2. Full program run setting many parameters, without native structure and 5 different trajectories each consisting of 1-3 files. Trajectories were obtained from AMBER so data files are necessary. Structures from all temperatures will be used for both clustering and Markov model creation.

```
command = fullrun
runmode = parallel
inputmode = amber
atomtypes = CA CB
clusterradius = 6.0
clusterusealltemperatures = true
temperaturelist = 270 300 330 360 400
graphvizscript = my_script.gv
graphvizignoreloops = true
graphvizscalevalues = true
kernelsfile = my_kernels.txt
structureinfofile = my_info.txt
threadnumber = 4
clustersizefilter = 20
markovlookuprange = 5
markovusealltemperatures = true
transitionmatrixfile = my_matrix.txt
# trajectory in a single file
trajfile = trajectory_0.pdb
# trajectory contained in 3 files
trajstart
trajfile = trajectory_1_1.pdb
trajfile = trajectory_1_2.pdb
trajfile = trajectory_1_3.pdb
trajend
# 3 trajectories, each in separate file
trajfile = trajectory_2.pdb
trajfile = trajectory_3.pdb
trajfile = trajectory_4.pdb
# trajectory contained in two files
trajstart
trajfile = trajectory_5_1.pdb
trajfile = trajectory_5_2.pdb
trajend
datafile = Amberdata1.mdout
datafile = Amberdata2.mdout
datafile = Amberdata3.mdout
datafile = Amberdata4.mdout
datafile = Amberdata5.mdout
```

3. Clustering run setting most options, using custom energy and temperature information from one file.

```
command = cluster
runmode = parallel
inputmode = custom
nativefile = 5ABC.pdb
clusterradius = 5.0
clustertemperature = true
kernelsfile = my_kernels.txt
structureinfofile = my_info.txt
trajfile = trajectory_0.pdb
trajfile = trajectory_1.pdb
trajfile = trajectory_2.pdb
trajfile = trajectory_3.pdb
trajfile = trajectory_4.pdb
threadnumber = 2
datafile = datafile.txt
clustersizefilter = 20
```

4. Markov model run setting most of its options.

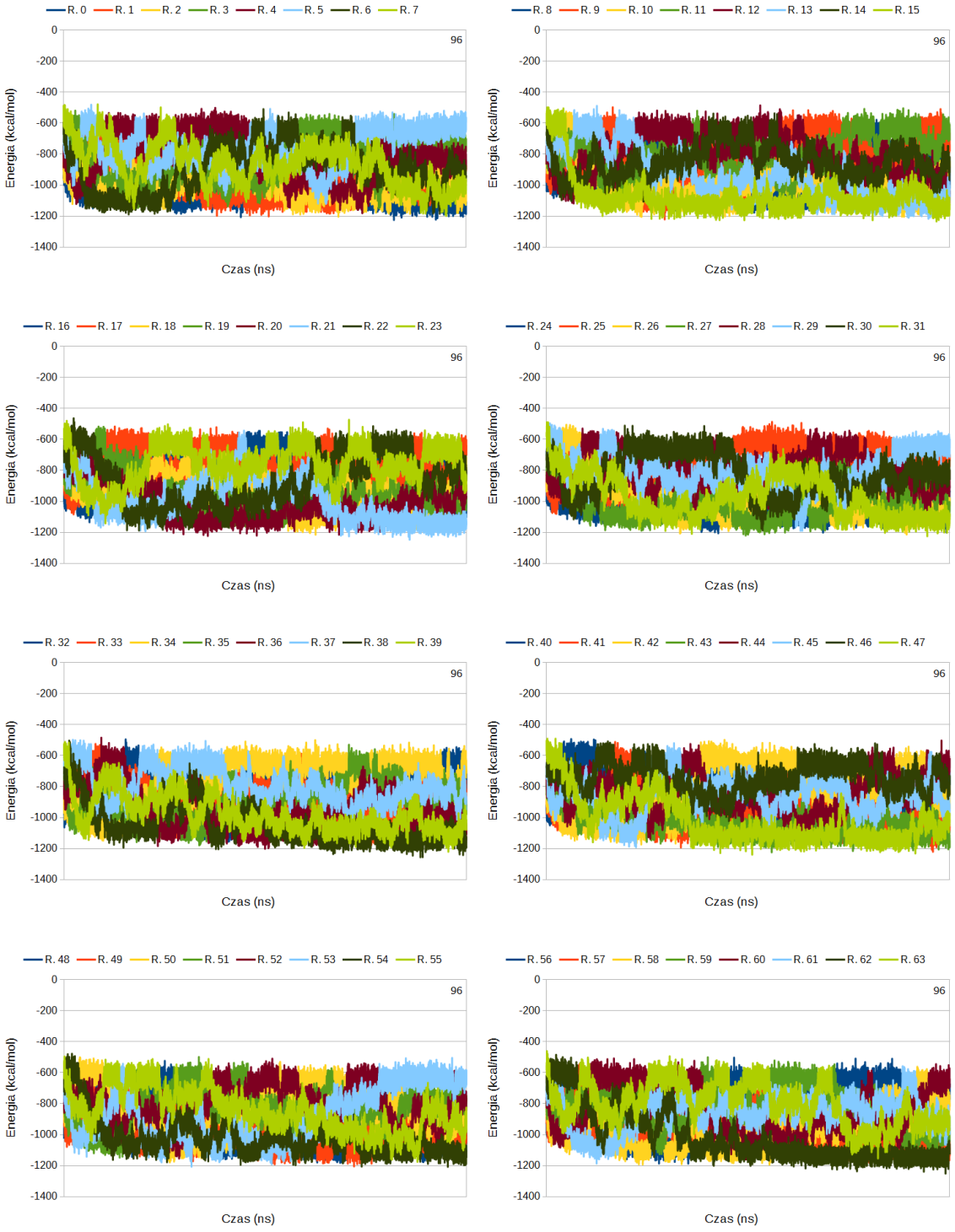
```
command = markov
runmode = sequential
kernelsfile = my_kernels.txt
structureinfofile = my_info.txt
clustersizefilter = 20
markovlookuprange = 3
markovnumberofclusters = 20
transitionmatrixfile = my_matrix.txt
```

5. Visualization run setting most of its options.

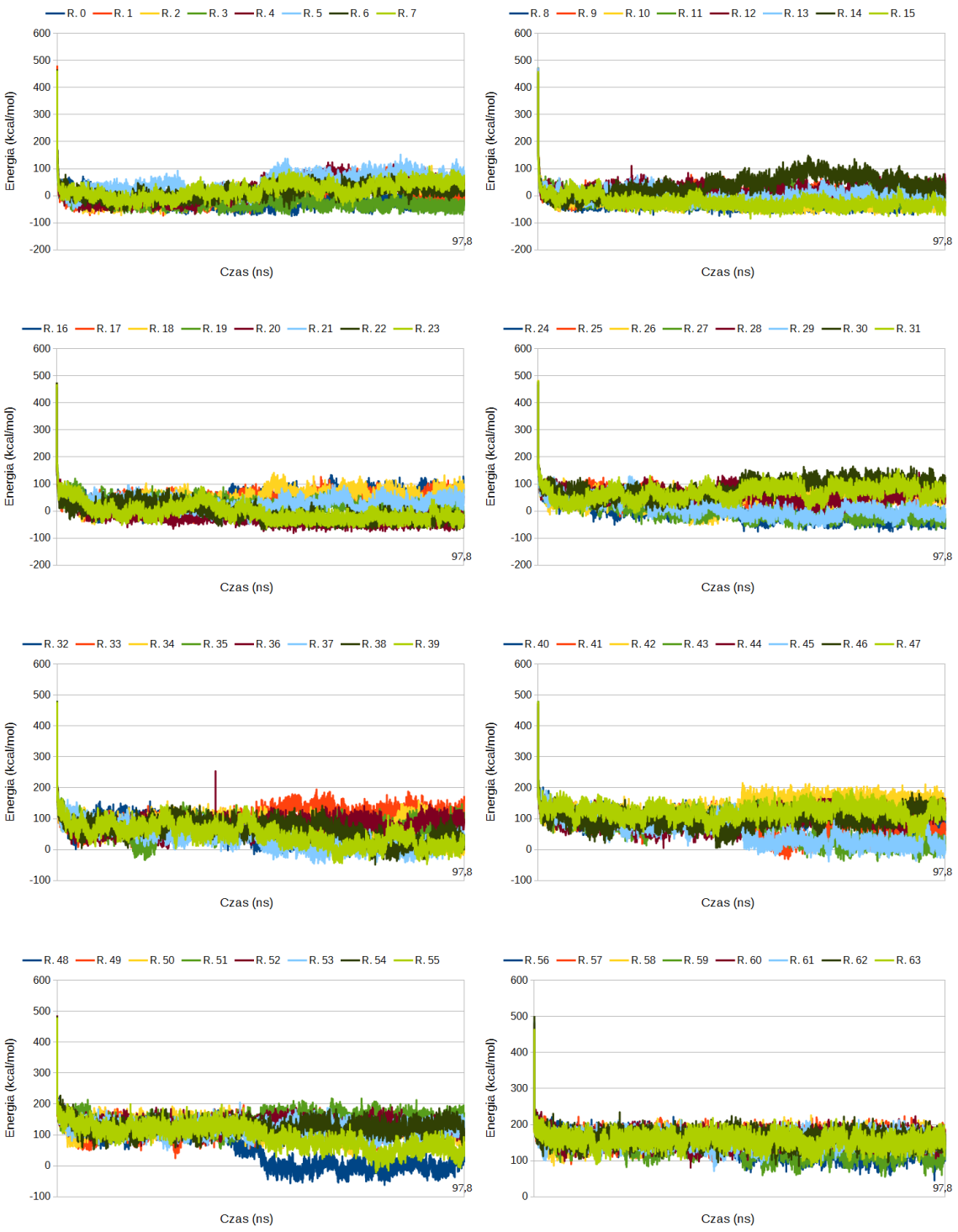
```
command = visualize
kernelsfile = my_kernels.txt
transitionmatrixfile = my_matrix.txt
graphvizignoreloops = true
graphvizscalevalues = true
graphvizmincutoff = 20
graphvizmaxcutoff = 800
graphvizscript = script.gv
```

12. Szczegółowe wykresy energii, RMSD względem struktury natywnej i współczynnika żyroskopowego w przeprowadzonych symulacjach

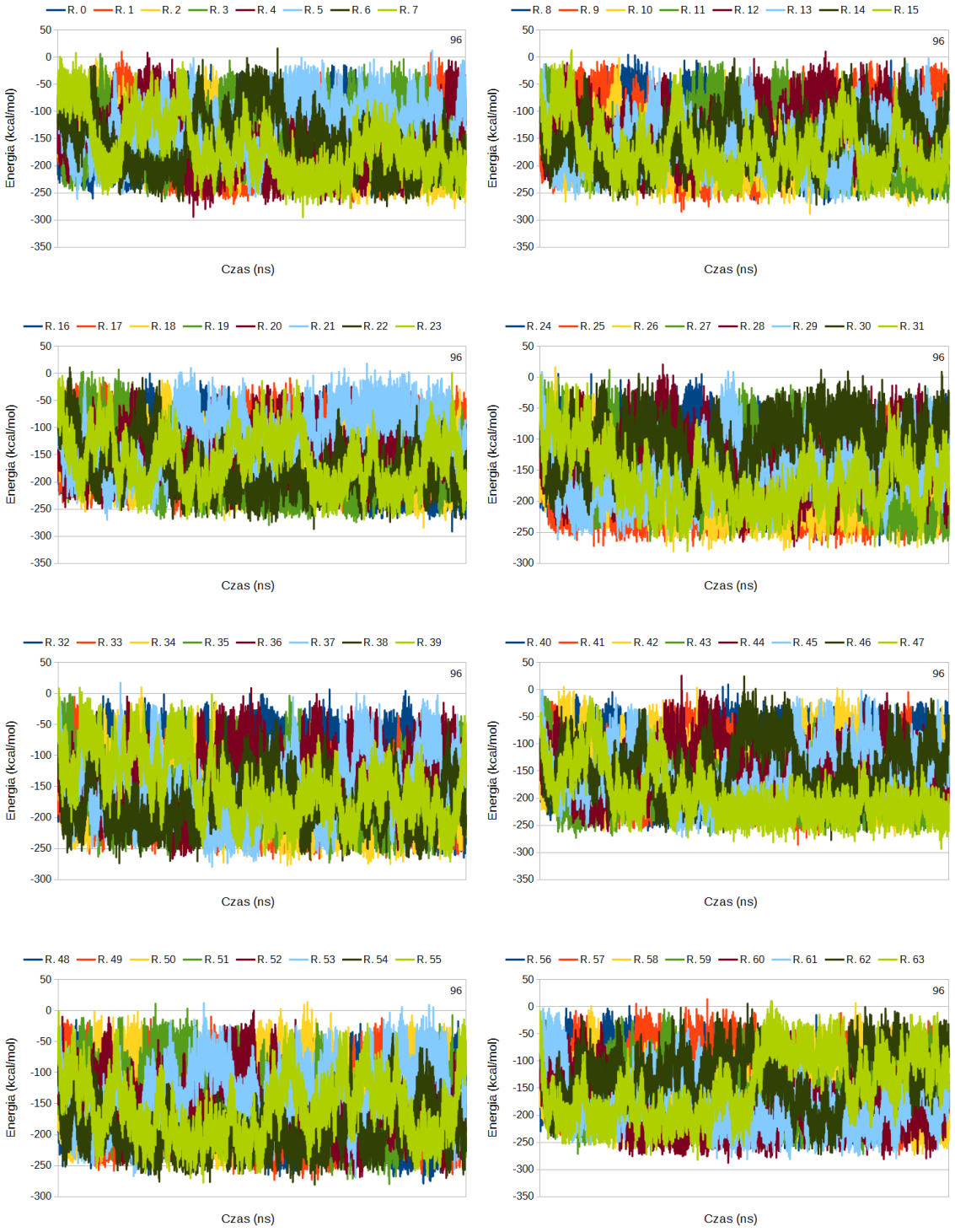
Na zaprezentowanych w dalszej części tego rozdziału wykresach pokazałem, wspomniane w rozdziale 8, wykresy energii całkowitej, współczynnika żyroskopowego i RMSD względem struktury natywnej we wszystkich przeprowadzonych symulacjach. Umieściłem je głównie w celu pokazania ogólnej poprawności tych symulacji i z tego powodu nie omawiam ich szczegółowo.



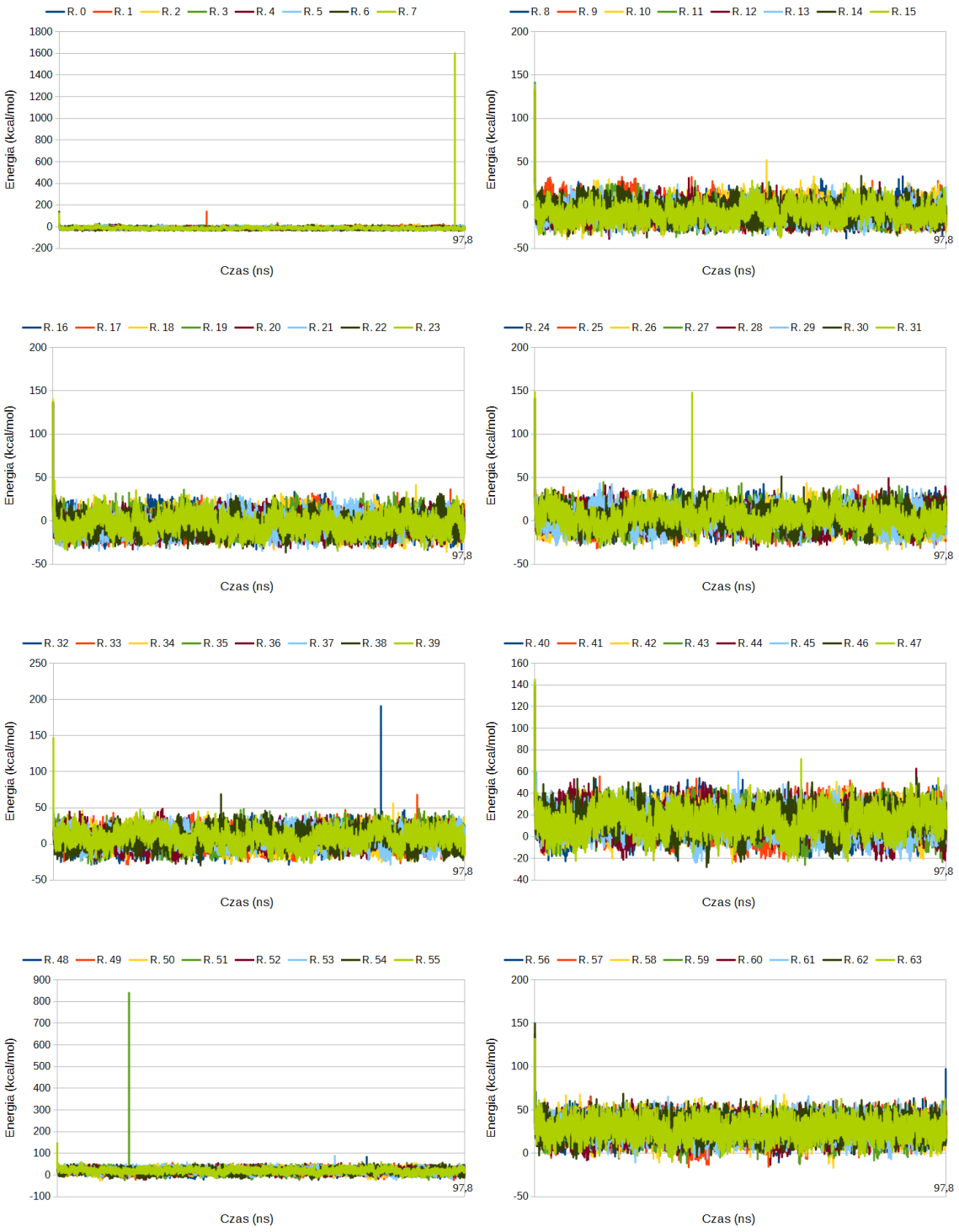
Rysunek 12.1. Wykresy zależności całkowitej energii od czasu dla symulacji struktury o kodzie ID 1BDD przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



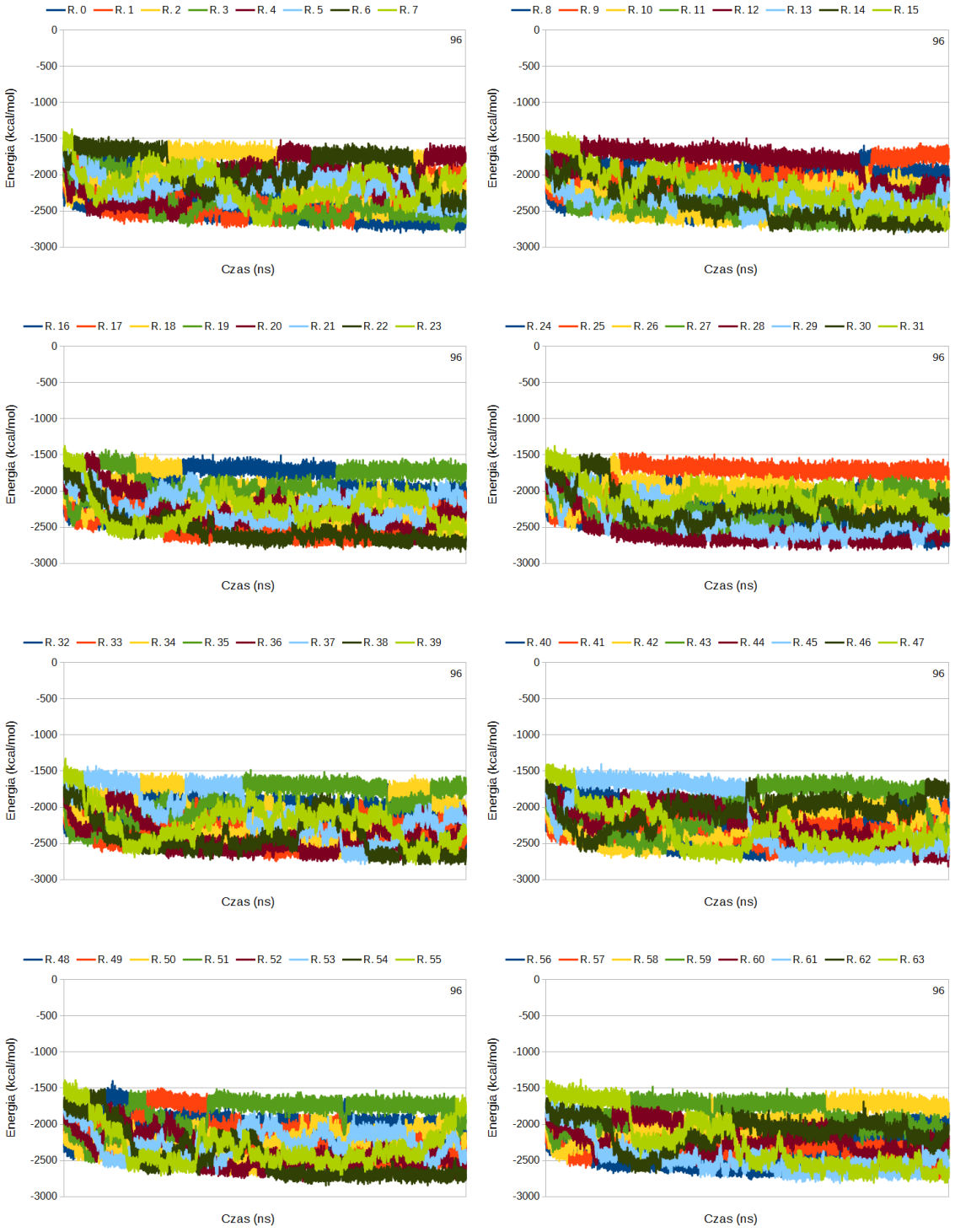
Rysunek 12.2. Wykresy zależności całkowitej energii od czasu dla symulacji struktury o kodzie ID 1BDD przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



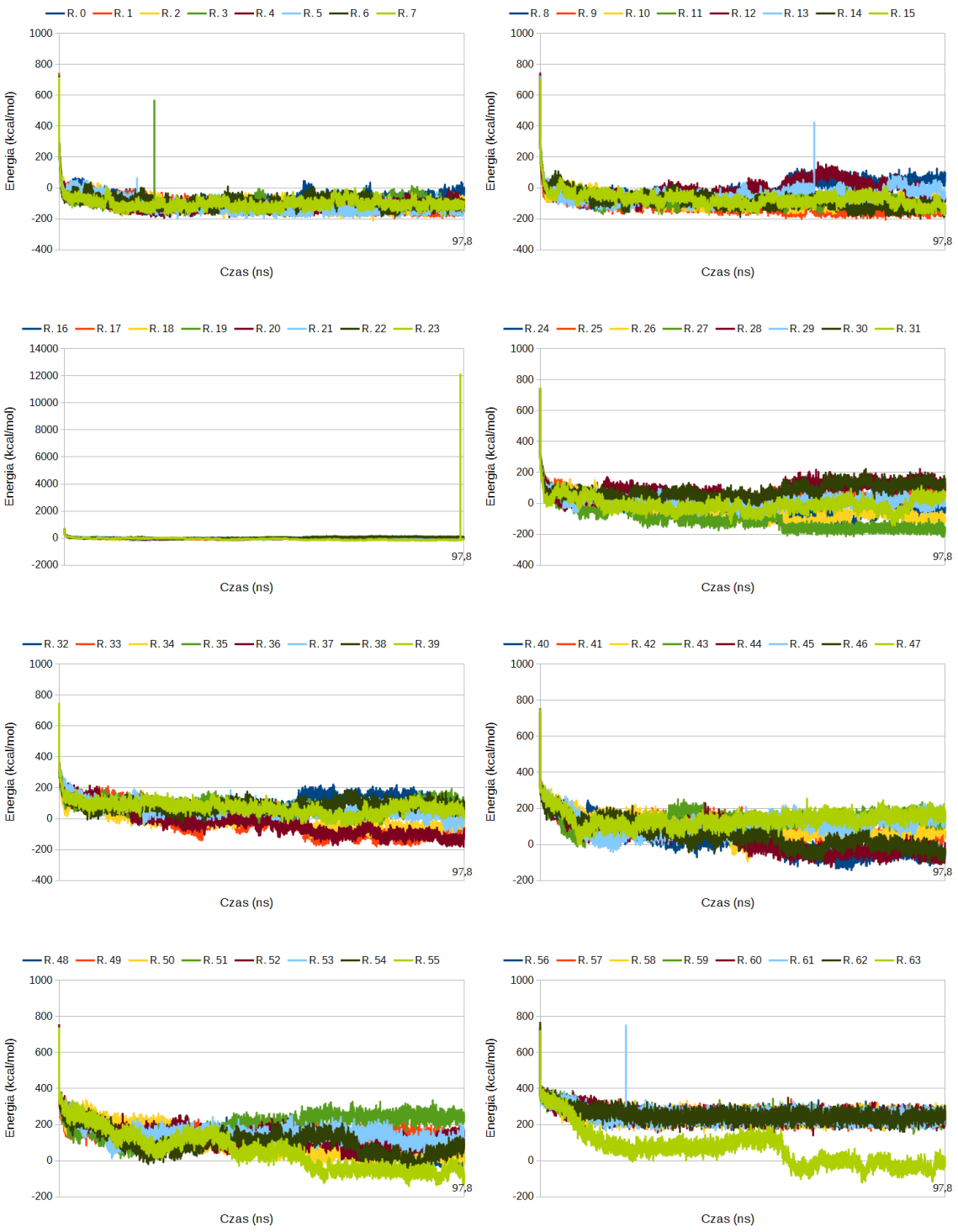
Rysunek 12.3. Wykresy zależności całkowitej energii od czasu dla symulacji struktury o kodzie ID 1L2Y przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



Rysunek 12.4. Wykresy zależności całkowitej energii od czasu dla symulacji struktury o kodzie ID 1L2Y przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)

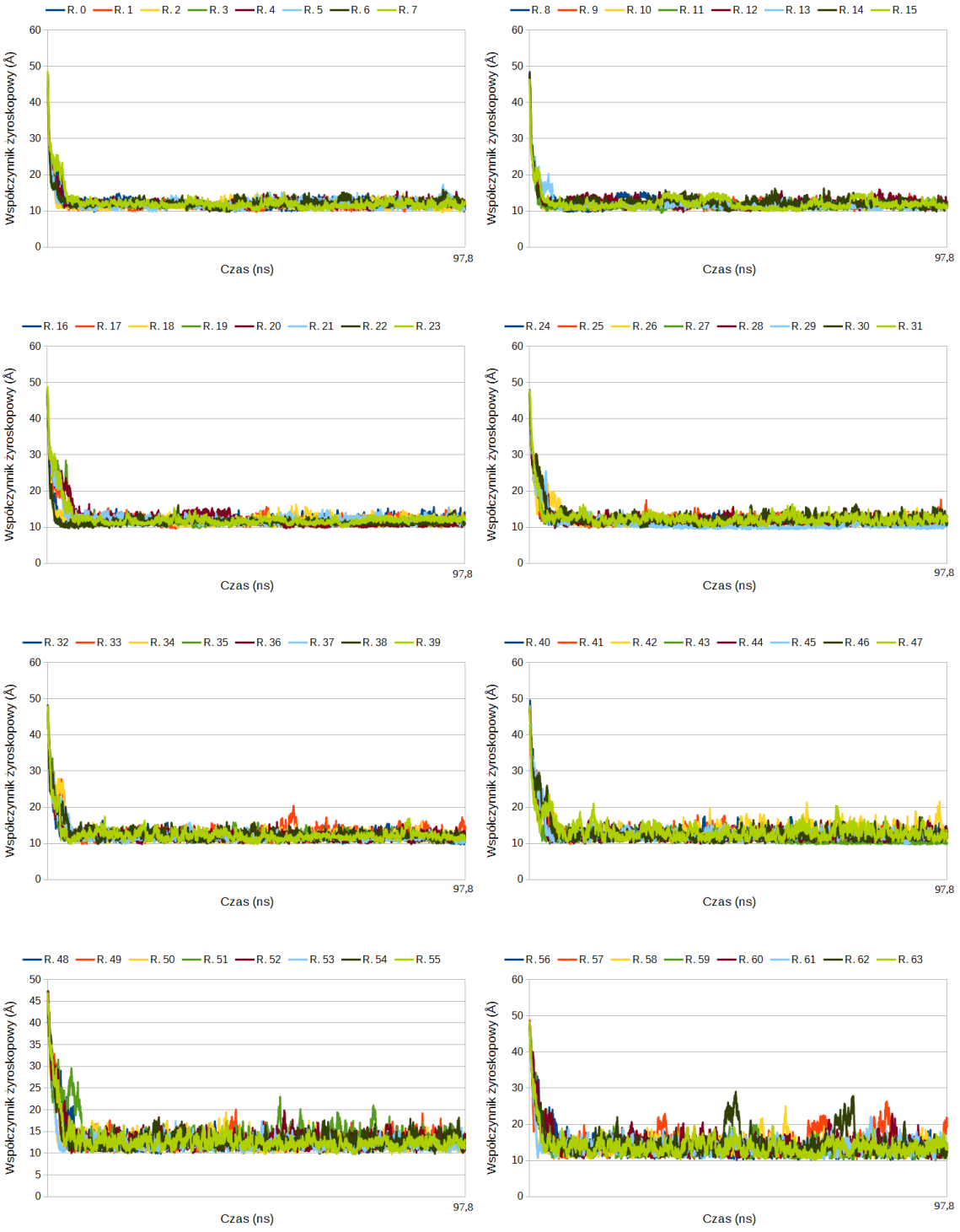


Rysunek 12.5. Wykresy zależności całkowitej energii od czasu dla symulacji struktury o kodzie ID 2MQ8 przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



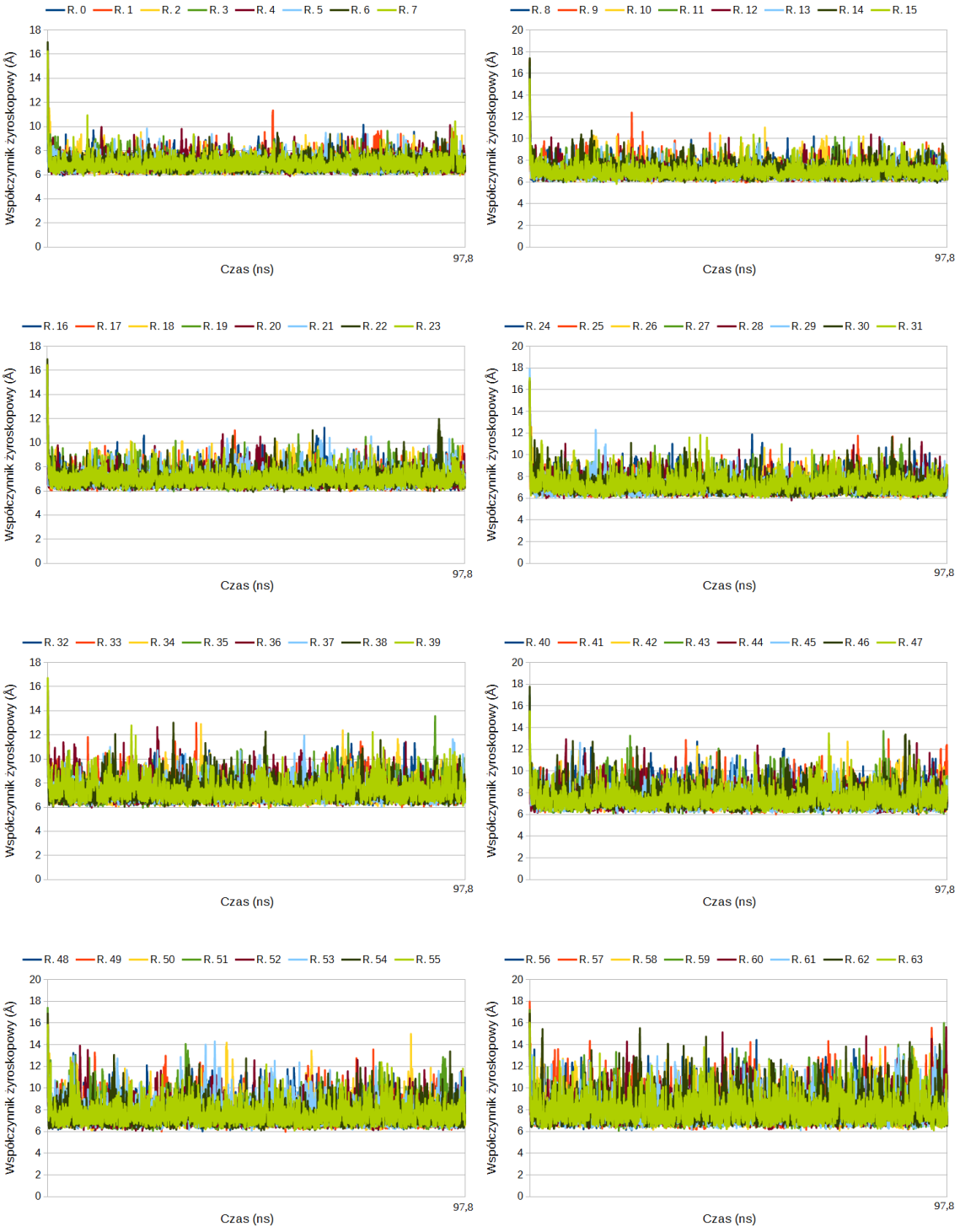
Rysunek 12.6. Wykresy zależności całkowitej energii od czasu dla symulacji struktury o kodzie ID 2MQ8 przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)

Rysunek 12.7. Wykresy zależności współczynnika żyroskopowego od czasu dla symulacji struktury o kodzie ID 1BDD przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)

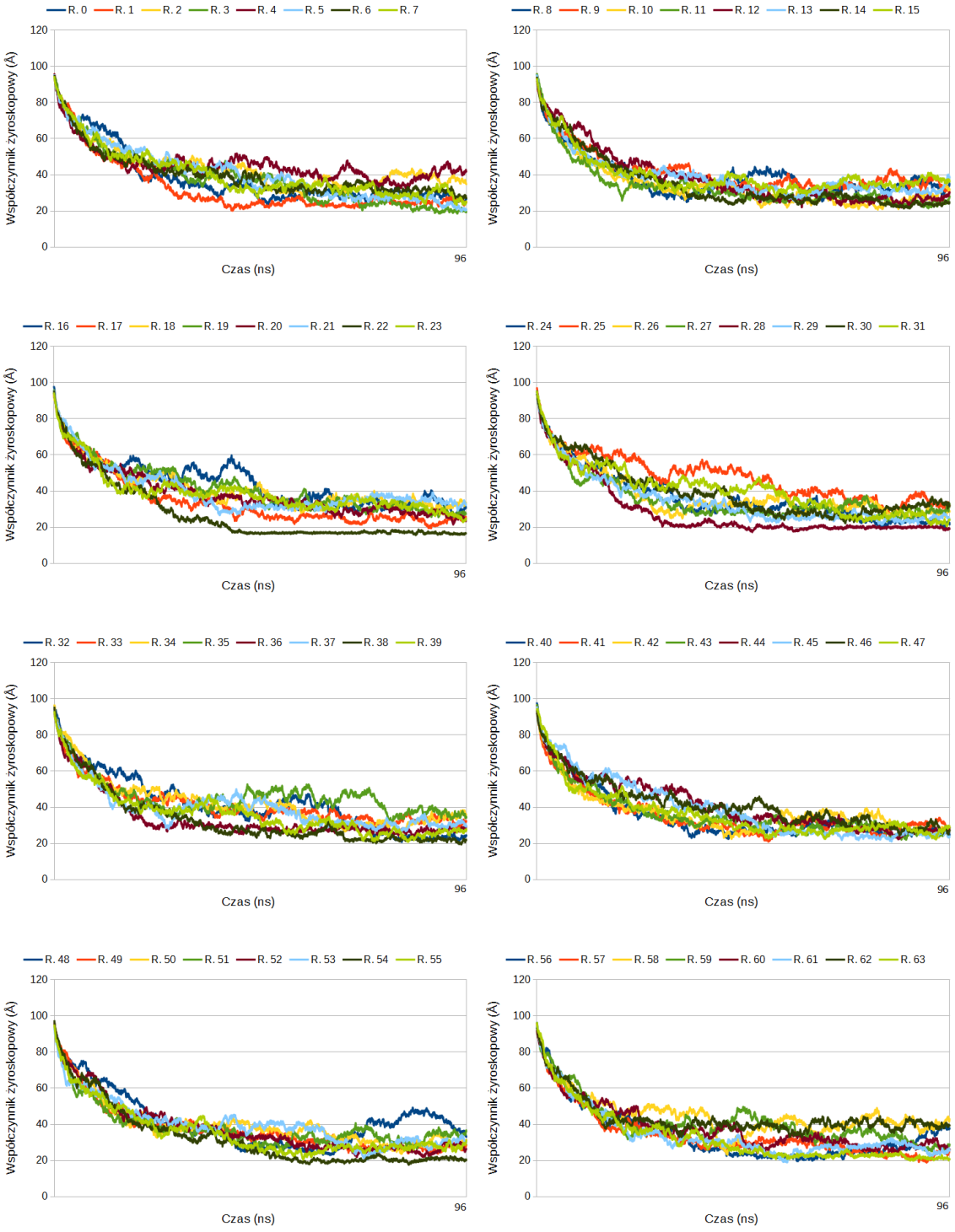


Rysunek 12.8. Wykresy zależności współczynnika żyroskopowego od czasu dla symulacji struktury o kodzie ID 1BDD przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)

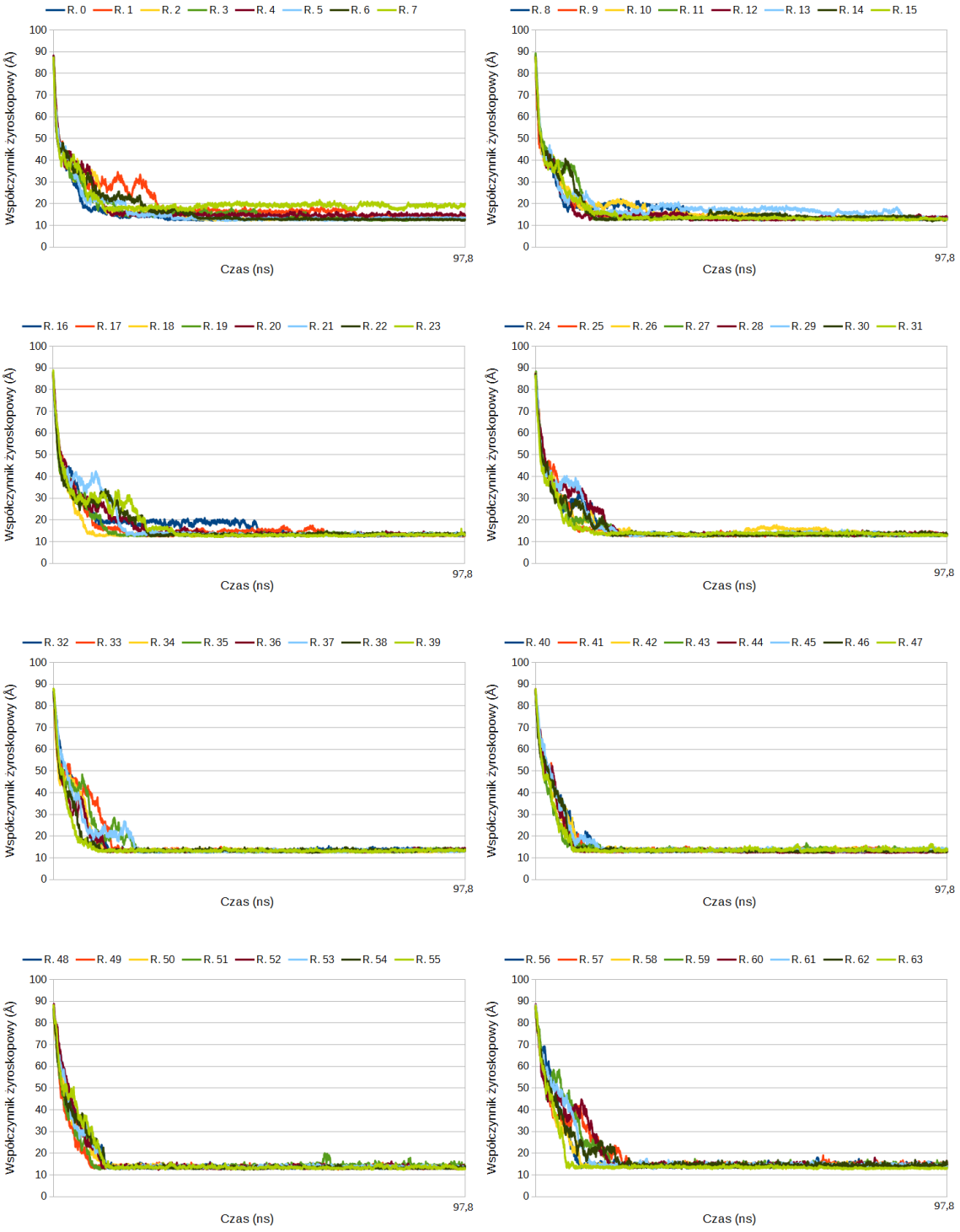
Rysunek 12.9. Wykresy zależności współczynnika żyroskopowego od czasu dla symulacji struktury o kodzie ID 1L2Y przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



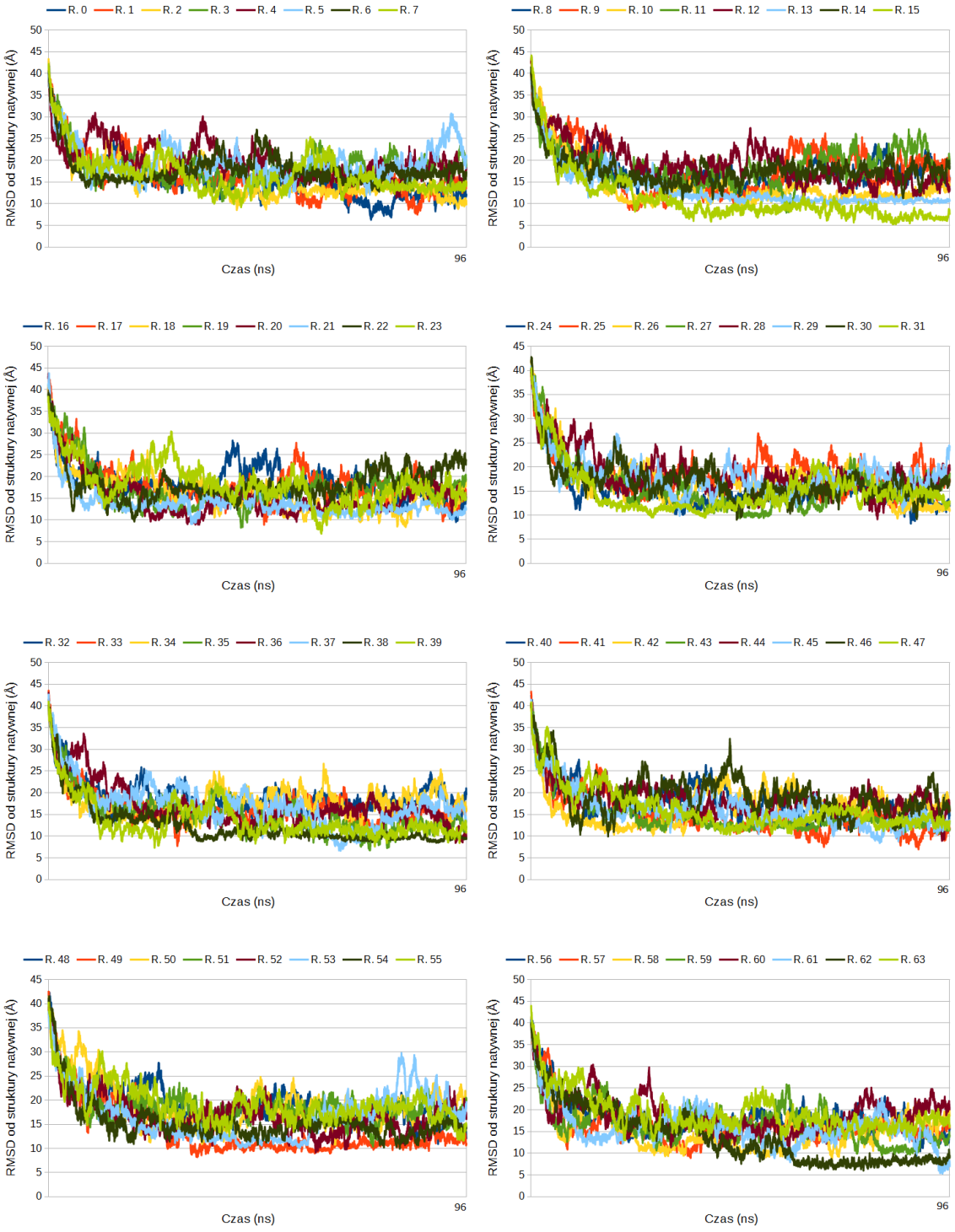
Rysunek 12.10. Wykresy zależności współczynnika żyroskopowego od czasu dla symulacji struktury o kodzie ID 1L2Y przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



Rysunek 12.11. Wykresy zależności współczynnika żyroskopowego od czasu dla symulacji struktury o kodzie ID 2MQ8 przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



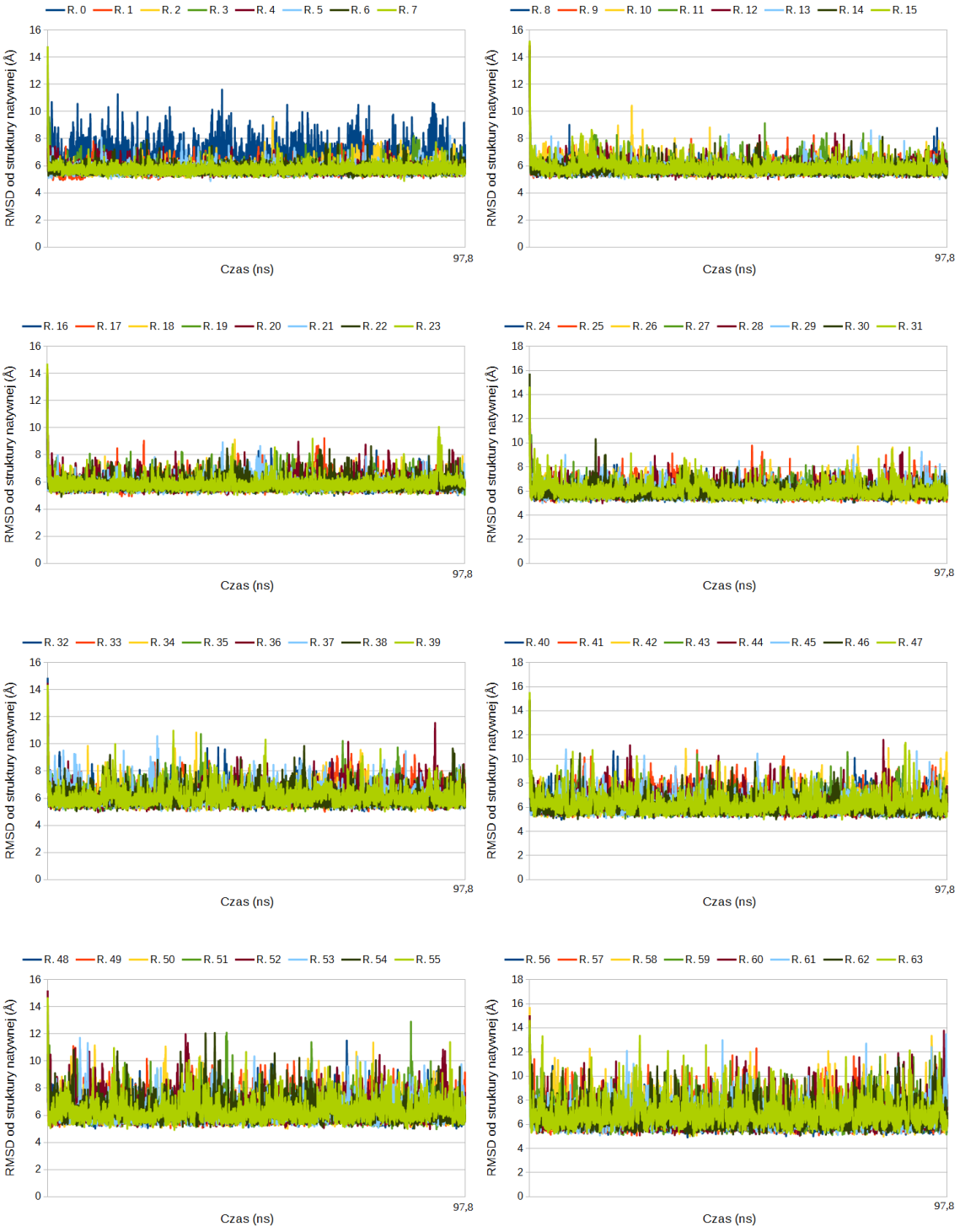
Rysunek 12.12. Wykresy zależności współczynnika żyroskopowego od czasu dla symulacji struktury o kodzie ID 2MQ8 przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



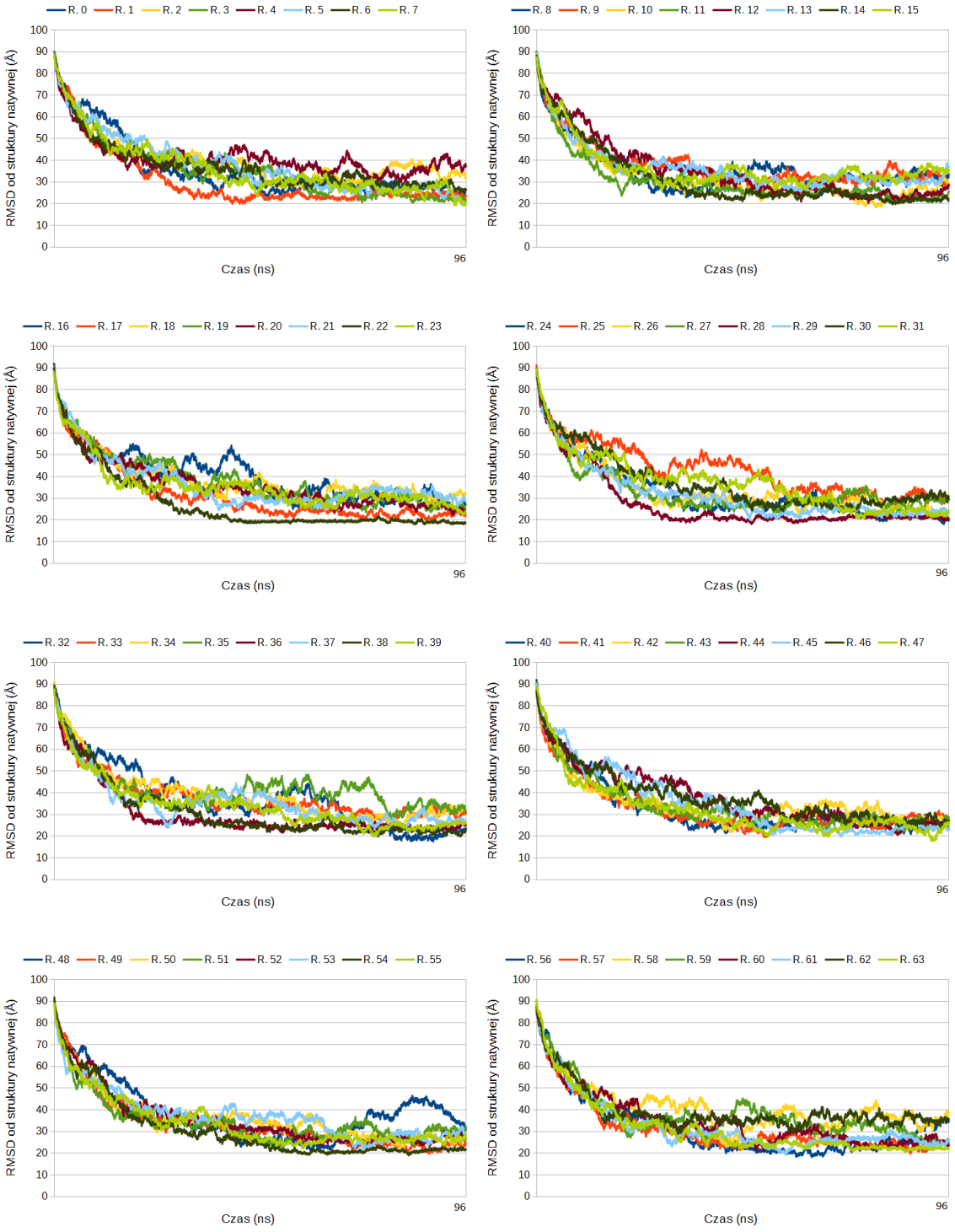
Rysunek 12.13. Wykresy zależności RMSD względem struktury natywnej od czasu od czasu dla symulacji struktury o kodzie ID 1BDD przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)

Rysunek 12.14. Wykresy zależności RMSD względem struktury natywnej od czasu dla symulacji struktury o kodzie ID 1BDD przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)

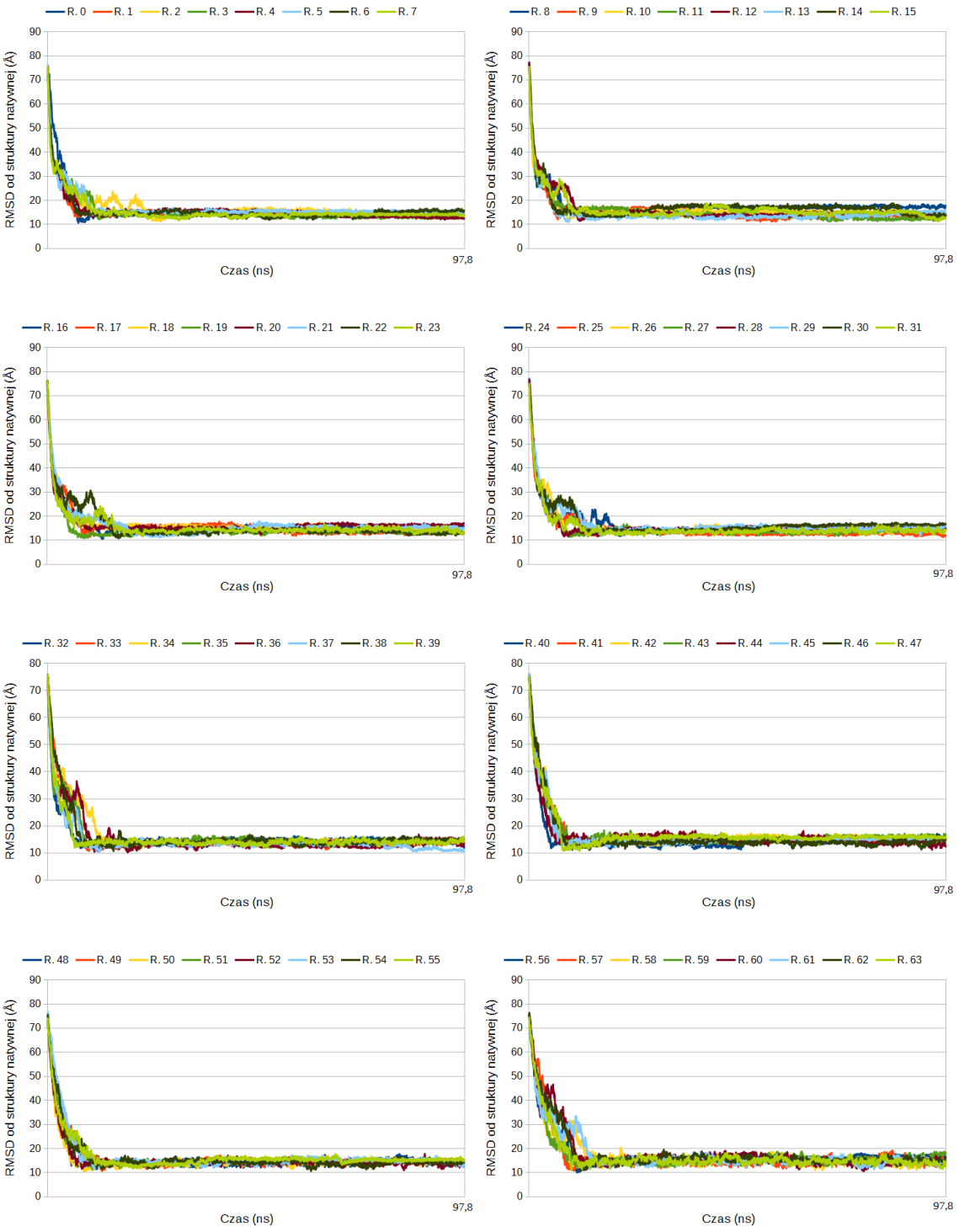
Rysunek 12.15. Wykresy zależności RMSD względem struktury natywnej od czasu dla symulacji struktury o kodzie ID 1L2Y przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



Rysunek 12.16. Wykresy zależności RMSD względem struktury natywnej od czasu dla symulacji struktury o kodzie ID 1L2Y przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



Rysunek 12.17. Wykresy zależności RMSD względem struktury natywnej od czasu dla symulacji struktury o kodzie ID 2MQ8 przeprowadzonej w pakiecie AMBER (tylko production run, pominięto minimalizację i podgrzewanie). Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)



Rysunek 12.18. Wykresy zależności RMSD względem struktury natywnej od czasu dla symulacji struktury o kodzie ID 2MQ8 przeprowadzonej w pakiecie UNRES. Kolorami oznaczono poszczególne repliki. Ich kolejne numery znajdują się w legendzie nad każdym wykresem (R. 1 to Replika 1 itd.)

Bibliografia

[1] Stryer L., Berg J.M. i Tymoczko J.L. *Biochemia*. Warszawa: Wydawnictwo Naukowe PWN, 2009.

[2] Murray R.K., Granner D.K. i Rodwell V.W. *Biochemia Harpera*. Warszawa: Wydawnictwo Lekarskie PZWL, 2008.

[3] McMurry J. *Organic Chemistry*. Pacific Grove: Brooks/Cole, 2000.

[4] Rother M. i Krzycki J.A. „Selenocysteine, Pyrrolysine, and the Unique Energy Metabolism of Methanogenic Archaea". W: *Archaea* 2010 (2010 2010).

[5] Knorre D.G., Kudryashova N.V. i Godovikova T.S. „Chemical and Functional Aspects of Posttranslational Modification of Proteins." W: *Acta Naturae* 1 (3 2009), s. 29–51.

[6] www.umass.edu/microbio/rasmol/seleccmd.htm. dostęp 09.07.18.

[7] Ramachandran G.N., Ramakrishnan C. i Sasisekharan V. „Stereochemistry of polypeptide chain configurations". W: *Journal of Molecular Biology* 7 (1 1963), s. 95–99.

[8] Fidler A.L., Vanacore R.M., Chetyrkin S.V. i in. „A unique covalent bond in basement membrane is a primordial innovation for tissue evolution". W: *Proceedings of the National Academy of Sciences of the USA* 111 (1 2014), s. 331–336.

[9] Chou K.C. „Prediction of tight turns and their types in proteins". W: *Analytical Biochemistry* 286 (1 2000), s. 1–16.

[10] Toniolo C. i Benedetti E. „The polypeptide 310-helix". W: *Trends in Biochemical Sciences* 16 (9 1991), s. 350–353.

[11] Fodje M.N. i Al-Karadaghi S. „Occurrence, conformational features and amino acid propensities for the pi-helix". W: *Protein Engineering* 15 (5 2002), s. 353–358.

[12] Adzhubei A.A., Sternberg M.J. i Makarov A.A. „Polyproline-II helix in proteins: structure and function". W: *Journal of Molecular Biology* 425 (12 2013), s. 2100–2132.

[13] Branden C. i Tooze J. *Introduction to Protein Structure*. Nowy Jork: Garland Publishing, 1999.

[14] Anfinsen C.B. „Principles that govern the folding of protein chains." W: *Science* 181 (4096 1973), s. 223–230.

[15] Zwanzig R. „Two-state models of protein folding kinetics". W: *Proceedings of the National Academy of Sciences of the USA* 94 (1 1997), s. 148–150.

[16] Baryshnikova E.N., Melnik B.S., Finkelstein A.V. i in. „Three-state protein folding: Experimental determination of free-energy profile". W: *Protein Science* 14 (10 2005), s. 2658–2667.

[17] Khorasanizadeh S., Peters I.D. i Roder H. „Evidence for a three-state model of protein folding from kinetic analysis of ubiquitin variants with altered core residues". W: *Nature Structural & Molecular Biology* 3 (2 1996), s. 193–205.

[18] Berman H.M., Westbrook J., Feng Z. i in. „The Protein Data Bank." W: *Nucleic Acids Research* 28 (1 2000), s. 235–242.

[19] Kendrew J.C., Bodo G., Dintzis H.M. i in. „A three-dimensional model of the myoglobin molecule obtained by x-ray analysis." W: *Nature* 181 (4610 1958), s. 662–666.

[20] McPherson A. i Gavira J.A. „Introduction to protein crystallization." W: *Acta Crystallographica Section F* 70 (2014), s. 2–20.

[21] Wlodawer A., Minor W., Dauter Z. i in. „Protein crystallography for non-crystallographers, or how to get the best (but not more) from published macromolecular structures." W: *The FEBS Journal* 275 (1 2008), s. 1–21.

[22] Cavanagh J., Fairbrother W.J., Palmer A.G. i in. *Protein NMR Spectroscopy: Principles and Practice*. Londyn: Elsevier Academic Press, 2007.

[23] Braun W., Bösch C., Brown L.R. i in. „Combined use of proton-proton overhauser enhancements and a distance geometry algorithm for determination of poly-

peptide conformations. Application to micelle-bound glucagon." W: *Biochimica et Biophysica Acta* 667 (2 1981), s. 377–396.

[24] https://www.rcsb.org/stats. dostęp 8.10.18.

[25] Milne J.L., Borgnia M.J., Bartesaghi A. i in. „Cryo-electron microscopy - a primer for the non-microscopist." W: *FEBS Journal* 280 (1 2013), s. 28–45.

[26] Cheng Y., Grigorieff N., Penczek P.A. i in. „A primer to single-particle cryo-electron microscopy." W: *Cell* 161 (3 2015), s. 438–449.

[27] Henderson R., Baldwin J.M., Ceska T.A. i in. „Model for the structure of bacteriorhodopsin based on high-resolution electron cryo-microscopy." W: *Journal of Molecular Biology* 213 (4 1990), s. 899–929.

[28] Leach A.R. *Molecular Modelling: Principles and Applications*. Harlow: Pearson Education Limited, 2001.

[29] Floudas C.A., Fung H.K., McAllister S.R. i in. „Advances in protein structure prediction and de novo protein design: A review." W: *Chemical Engineering Science* 61 (3 2006), s. 966–988.

[30] Field M.J. *A Practial Introduction to the Simulation of Molecular Systems*. Cambridge: Cambridge University Press, 2007.

[31] Holtje H.D., Sippl W., Rognan D. i in. *Molecular Modeling Basic Principles and Applications*. Weinheim: Wiley-VCH, 1997.

[32] Kaźmierkiewicz R. *Introduction to molecular modeling*. Gdańsk: Intercollegiate Faculty of Biotechnology UG-MUG, 2011.

[33] Yu Z. i Lau D. „Development of a coarse-grained α-chitin model on the basis of MARTINI forcefield." W: *Journal of Molecular Modeling* 21 (5 2015), s. 128.

[34] *Amber 2017 Reference Manual*.

[35] Haile J.M. *Molecular Dynamics Simulation: Elementary Methods*. John Wiley & Sons, 1997.

[36] Mahoney M.W. i Jorgensen W.L. „A five-site model for liquid water and the reproduction of the density anomaly by rigid, nonpolarizable potential functions." W: *The Journal of Chemical Physics* 112 (20 2000), s. 8910–8922.

[37] Jorgensen W.L., Chandrasekhar J., Madura J.D. i in. „Comparison of simple potential functions for simulating liquid water." W: *The Journal of Chemical Physics* 79 (2 1983), s. 926–935.

[38] Allen M.P. i Tildesley D.J. *Computer Simulation of Liquids*. Oxford: Oxford Science Publications, 2009.

[39] Ryckaert J.P., Ciccotti G. i Berendsen H.J.C. „Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes". W: *Biophysical Journal* 23 (3 1977), s. 327–341.

[40] Hansen J.P. i McDonald I.R. *Theory of Simple Liquids*. Academic Press, 2006.

[41] Sugita Y. i Okamoto Y. „Replica-exchange molecular dynamics method for protein folding." W: *Chemical Physics Letters* 314 (1999), s. 141–151.

[42] Bernardi R.C., Melo M.C.R. i Schulten K. „Enhanced Sampling Techniques in Molecular Dynamics Simulations of Biological Systems." W: *Biochimica et Biophysica Acta* 1850 (5 2015), s. 872–877.

[43] Rhee Y.M. i Pande V.S. „Multiplexed-Replica Exchange Molecular Dynamics Method for Protein Folding Simulation". W: *Biophysical Journal* 84 (2 2003), s. 775–786.

[44] Jain A.K., Murty M.N. i Flynn P.J. „Data Clustering: A Review". W: *ACM Computing Surveys* 31 (3 1999), s. 264–323.

[45] Keller B., Daura X. i van Gunsteren W.F. „Comparing geometric and kinetic cluster algorithms for molecular simulation data". W: *The Journal of Chemical Physics* (132 2010).

[46] https://encyklopedia.pwn.pl. dostęp 20.07.18.

[47] Gagniuc P.A. *Markov Chains: From Theory to Implementation and Experimentation*. John Wiley & Sons, 2017.

[48] Noé F. i Fischer S. „Transition networks for modeling the kinetics of conformational change in macromolecules." W: *Current Opinion in Structural Biology* 18 (2 2008), s. 154–162.

[49] Razmara J., Deris S.B., Illias R.B. i in. „Artificial signal peptide prediction by a hidden markov model to improve protein secretion via Lactococcus lactis bacteria." W: *Bioinformation* 9 (7 2013), s. 345–348.

[50] Zhao X.Y., Zhang J., Chen Y.Y. i in. „Promoter recognition based on the maximum entropy hidden Markov model." W: *Computers in Biology and Medicine* 51 (2014), s. 73–81.

[51] joshmillard.com/garkov/. dostęp 15.04.18.

[52] Chatterjee A. i Bhattacharya S. „Uncertainty in a Markov state model with missing states and rates: Application to a room temperature kinetic model obtained using high temperature molecular dynamics." W: *The Journal of Chemical Physics* 143 (11 2015), s. 114109.

[53] Shukla D., Hernández C.X., Weber J.K. i in. „Markov State Models Provide Insights into Dynamic Modulation of Protein Function." W: *Accounts of Chemical Research* 480 (2 2015), s. 414–22.

[54] Malmstrom R.D., Lee C.T., Van Wart A. i in. „On the Application of Molecular Dynamics Based Markov State Models to Functional Proteins." W: *Journal of Chemical Theory and Computation* 10 (7 2014), s. 2648–2657.

[55] Noé F., Schütte C., Vanden-Eijnden E. i in. „Constructing the equilibrium ensemble of folding pathways from short off-equilibrium simulations." W: *Proceedings of the National Academy of Sciences of the USA* 106 (45 2009), s. 19011–6.

[56] Feng H., Costaouec R., Darve E. i in. „A comparison of weighted ensemble and Markov state model methodologies." W: *The Journal of Chemical Physics* 142 (21 2015), s. 214113.

[57] Bowman G.R., Pande V.S. i Noé F. *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*. Dordrecht: Springer Science+Business Media, 2014.

[58] Voelz V.A., Bowman G.R., Beauchamp K. i in. „Molecular simulation of ab initio protein folding for a millisecond folder NTL9(1-39)." W: *Nature Structural & Molecular Biology* 132 (5 2010), s. 1526–8.

[59] Lane T.J., Bowman G.R., Beauchamp K. i in. „Markov state model reveals folding and functional dynamics in ultra-long MD trajectories." W: *Journal of the American Chemical Society* 133 (45 2011), s. 18413–19.

[60] Da L.T., Pardo Avila F., Wang D. i in. „A two-state model for the dynamics of the pyrophosphate ion release in bacterial RNA polymerase." W: *PLOS Computational Biology* 9 (4 2013), e1003020.

[61] Asciutto E.K., Gedeon P.C., General I.J. i in. „Structure and Dynamics Study of LeuT Using the Markov State Model and Perturbation Response Scanning Reveals Distinct Ion Induced Conformational States." W: *The Journal of Physical Chemistry B* 120 (33 2016), s. 8361–8368.

[62] Zeng X., Zhang L, Xiao X. i in. „Unfolding mechanism of thrombin-binding aptamer revealed by molecular dynamics simulation and Markov State Model." W: *Scientific Reports* 6 (2016), s. 24065.

[63] Kernighan B. i Ritchie D. *Język ANSI C*. Warszawa: Wydawnictwa Naukowo - Techniczne, 2000.

[64] www.open-std.org/jtc1/sc22/wg14. dostęp 28.11.17.

[65] www.tiobe.com/tiobe-index. dostęp 28.11.17.

[66] pypl.github.io/PYPL.html. dostęp 28.11.17.

[67] Porter C.T. i Martin A.C.R. „BiopLib and BiopTools—a C programming library and toolset for manipulating protein structure." W: *Bioinformatics* 31 (24 2015), s. 4017–4019.

[68] Dagum L. i Menon R. „OpenMP: an industry standard API for shared-memory programming." W: *IEEE Computational Science and Engineering* 5 (1 1998), s. 46–55.

[69] www.openmp.org. dostęp 28.11.17.

[70] The MPI Forum. „MPI: A Message Passing Interface." W: *CSETech* 324 (1994).

[71] mpi-forum.org/docs. dostęp 28.11.17.

[72] www.graphviz.org. dostęp 28.11.17.

[73] linuxmint.com. dostęp 28.11.17.

[74] www.codeblocks.org. dostęp 28.11.17.

[75] www.geany.org. dostęp 28.11.17.

[76] gcc.gnu.org. dostęp 28.11.17.

[77] www.gnu.org/software/make. dostęp 28.11.17.

[78] valgrind.org. dostęp 28.11.17.

[79] www.mercurial-scm.org. dostęp 28.11.17.

[80] bitbucket.org. dostęp 28.11.17.

[81] Kabsch W. „A solution for the best rotation to relate two sets of vectors". W: *Acta Crystallographica Section A* (A32 1976), s. 922–923.

[82] www.unres.pl. dostęp 09.03.18.

[83] www.rcsb.org/structure/1bdd. dostęp 13.11.17.

[84] Schrödinger, LLC. „The PyMOL Molecular Graphics System, Version 2.0". 2017.

[85] Gouda H., Torigoe H.., Saito A. i in. „Three-Dimensional Solution Structure of the B Domain of Staphylococcal Protein A: Comparisons of the Solution and Crystal Structures". W: *Biochemistry* (31 1992), s. 9665–9672.

[86] www.rcsb.org/structure/1l2y. dostęp 13.11.17.

[87] Neidigh J.W., Fesinmeyer R.M. i Andersen N.H. „Designing a 20-residue protein". W: *Nature Structural Biology* (9 2002), s. 425–430.

[88] www.rcsb.org/structure/2MQ8. dostęp 13.11.17.

[89] Koga N., Tatsumi-Koga R., Liu G. i in. „Principles for designing ideal protein structures." W: *Nature* (491 2012), s. 222–7.

[90] Lin Y., Koga N., Tatsumi-Koga R. i in. „Control over overall shape and size in de novo designed proteins". W: *Proceedings of the National Academy of Sciences of the USA* (112 2015), E5478–85.

[91] ambermd.org. dostęp 09.03.18.

[92] Case D.A., Betz R.M., Cerutti D.S. i in. *AMBER 2016*. 2016.

[93] Salomon-Ferrer R., Case D.A. i Walker R.C. „An overview of the Amber biomolecular simulation package". W: *WIREs Computational Molecular Science* (3 2013), s. 198–210.

[94] Ponder J.W. i Case D.A. „Force fields for protein simulations." W: *Journal of Molecular Modeling* 66 (2003), s. 27–85.

[95] Roe D.R. i Cheatham T.E. III. „PTRAJ and CPPTRAJ: Software for Processing and Analysis of Molecular Dynamics Trajectory Data." W: *Journal of Chemical Theory and Computation* 9 (7 2013), s. 3084–3095.

[96] Weiner S.J., Kollman P.A., Case D.A. i in. „A New Force Field for Molecular Mechanical Simulation of Nucleic Acids and Proteins." W: *Journal of the American Chemical Society* 106 (3 1984), s. 765–784.

[97] Voth G.A. *Coarse-graining of Condensed Phase and Biomolecular Systems*. Boca Raton: CRC Press, 2009.

[98] Liwo A., Khalili M. i Scheraga H.A. „Ab initio simulations of protein-folding pathways by molecular dynamics with the united-residue model of polypeptide chains." W: *Proceedings of the National Academy of Sciences of the USA* 102 (7 2005), s. 2362–2367.

[99] Liwo A., Baranowski M., Czaplewski C. i in. „A unified coarse-grained model of biological macromolecules based on mean-field multipole–multipole interactions." W: *Journal of Molecular Modeling* 20 (8 2014), s. 2306.

[100] Czaplewski C., Kalinowski S., Liwo A. i in. „Application of Multiplexed Replica Exchange Molecular Dynamics to the UNRES Force Field: Tests with α and $\alpha+\beta$ Proteins." W: *Journal of Chemical Theory and Computation* 5 (3 2009), s. 627–640.

[101] Maier J.A., Martinez C., Kasavajhala K. i in. „ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB." W: *Journal of Chemical Theory and Computation* 11 (8 2015), s. 3696–3713.

[102] Onufriev A., Bashford D. i Case D.A. „Modification of the Generalized Born Model Suitable for Macromolecules." W: *Journal of Physical Chemistry B* 104 (15 2000), s. 3712–3720.

[103] Onufriev A., Bashford D. i Case D.A. „Exploring Protein Native States and Large-Scale Conformational Changes With a Modified Generalized Born Model." W: *Proteins: Structure, Function, and Bioinformatics* 55 (2 2004), s. 383–394.

[104] Czaplewski C., Karczynska A., Sieradzan A.K. i in. „UNRES server for physics-based coarse-grained simulations and prediction of protein structure, dynamics and thermodynamics." W: *Nucleic Acid Research* 46 (W1 2018), W304–W309.

[105] Rotkiewicz P. i Skolnick J. „Fast procedure for reconstruction of full-atom protein models from reduced representations." W: *Journal of Computational Chemistry* 29 (9 2008), s. 1460–5.

[106] *Software Optimization Guide for AMD Family 17h Processors*. Advanced Micro Devices, 2017.

[107] Czech Z.J. *Wprowadzenie do obliczeń równoległych*. Warszawa: Wydawnictwo Naukowe PWN, 2013.

[108] Jung S., Bae S.E. i Son H.S. „Validity of Protein Structure Alignment Method Based on Backbone Torsion Angles." W: *Journal of Proteomics & Bioinformatics* 4 (10 2011), s. 218–226.