

Uniwersytet im. Adama Mickiewicza
w Poznaniu

Wydział Fizyki

Praca doktorska

Symulacje numeryczne wpływu dynamiki białka
na pewne biomolekularne procesy
przeniesienia elektronu

Mieczysław Torchała

Promotor

prof. dr hab. Michał Kurzyński

Uniwersytet im. Adama Mickiewicza, Wydział Fizyki

Poznań 2010

Dziękuję prof. dr. hab. Michałowi Kurzyńskiemu oraz dr. Przemysławowi Chełminia-
kowi za możliwość dyskusji problemów i wyników.

Dziękuję Fundacji na rzecz Nauki Polskiej za finansowanie pracy naukowej w ramach
stypendium FOCUS.

Dziękuję dr. hab. Marcinowi Hoffmannowi oraz dr. Leszkowi Rychlewskiemu za możli-
wość realizacji ciekawych projektów.

Spis treści

Wstęp	5
1 Dynamika białka	7
1.1 Dynamika wibracyjna a dynamika przejść konformacyjnych	8
1.2 Procesy stochastyczne – przypadkowe błędzenie	10
2 Metodologia badań numerycznych	13
2.1 Przebieg eksperymentu	14
2.1.1 Błąd wyznaczenia wartości średniej	15
2.1.2 Badane sieci	16
2.1.3 Problem generowania liczb pseudolosowych	22
2.2 Przetwarzanie równoległe	26
2.2.1 MPI i MPICH	29
2.2.2 Przygotowanie komputera do obliczeń równoległych	31
2.2.3 Przyspieszenie i wydajność	34
2.2.4 Problemy obliczeniowe i wyniki	35
3 Efekty dynamiki wibracyjnej: anomalna zależność temperaturowa	41
3.1 Teoretyczne sformułowanie problemu	42
3.2 Szczegóły symulacji i wyniki	48
3.2.1 Zależność średniego czasu pierwszego przejścia od temperatury i wysokości bariery energetycznej	51
3.2.2 Zależność średniego czasu pierwszego przejścia od temperatury i wyboru stanu początkowego	52
3.2.3 Wyjaśnienie anomalnej zależności temperaturowej czasu przenie- sienia elektronu w fotosyntetycznym centrum reakcji	54
4 Efekty dynamiki przejść konformacyjnych: przetwarzanie energii swobod- nej	57
4.1 Teoretyczne sformułowanie problemu	58

4.2	Szczegóły symulacji i wyniki	63
4.2.1	Wyznaczanie średnich czasów pierwszego przejścia	73
4.2.2	Jeden proces dostarczający i jeden odbierający energię swobodną	77
4.2.3	Eksperymentalne potwierdzenie teorii	81
4.2.4	Rozszerzenie modelu na większą liczbę procesów dostarczających i odbierających energię swobodną	86
4.2.5	Rozszerzenie modelu na sieci innego typu	90
Podsumowanie		99
Bibliografia		102
Spis tabel		107
Spis rysunków		109

Wstęp

Procesy przeniesienia elektronu zachodzą powszechnie w przyrodzie ożywionej, m.in. w fotosyntezie i oddychaniu komórkowym, i są kluczowe dla istnienia życia na różnych poziomach organizacji. Pierwsza teoria przeniesienia elektronu została stworzona ponad 50 lat temu przez Rudolpha A. Marcusa, który doczekał się następnie nagrody Nobla z chemii „for his contributions to the theory of electron transfer reactions in chemical systems” [1]. Mimo, że teoria ta wywarła ogromny wpływ na rozumienie kinetyki procesów przeniesienia elektronu, jej zakres stosowalności jest, jak każdej teorii, ograniczony, a w kolejnych latach pojawiały się różne uzupełnienia i nowe podejścia do problemu. Jednym z motywów do dalszych poszukiwań były wyniki badań eksperymentalnych, które pokazały, że dynamika otoczenia białkowego jest istotnym czynnikiem, który musi być brany pod uwagę przy opisie procesów przeniesienia elektronu zachodzących w układach biologicznych. Na uwagę zasługuje fakt, że rozwój teorii opisujących układy biologiczne jest znacznie wolniejszy niż postępy w stosowaniu metod eksperymentalnych. Układy biologiczne znajdują się w nierównowadze termodynamicznej, stąd np. próby opisu reakcji przy pomocy teorii stanu przejściowego są ogromnym przybliżeniem, a poszukiwanie nowych sposobów opisu i weryfikacja ich przydatności jest jednym z ważnych nurtów biofizyki teoretycznej.

W pracy tej stawiam tezę, że przy pomocy symulacji Monte Carlo prostych modeli dynamiki biomolekuł i używając stochastycznej teorii szybkości reakcji można badać efekty wpływu dynamiki, zarówno wibracyjnej jak i przejść konformacyjnych, otoczenia białkowego na procesy przeniesienia elektronu oraz wyjaśniać i próbować przewidywać fakty doświadczalne.

Sam opis przeniesienia elektronu jest tak szerokim i interdyscyplinarnym problemem naukowym (np. rysunek 1 na stronie 3 w „Electron transfer past and future” [2]), że niemożliwe jest ujęcie w ramach jednej pracy jego wszelkich aspektów. Od strony chemicznej i fizycznej tylko dwa tomy z serii *Advances in Chemical Physics* [3, 4] pod tytułem „Electron transfer – from isolated molecules to biomolecules” to razem ponad 1400 stron, a nie wyczerpują one ogromu tematu. Tym bardziej, że problemem należało zająć się od różnych stron (matematyki, informatyki, fizyki, biologii i chemii). Nie jest prosto wobec znacznej liczby publikacji i problemów wybrać te najważniejsze. W tej pracy skupiam się

głównie na podejściu numerycznym, a drogą do realizacji tematu był wybór pewnej liczby elementów z każdej z dziedzin. Narzędziem do badań jest oprogramowanie głównie w języku Fortran stworzone przez autora pracy.

Praca doktorska składa się z następujących części:

- **Rozdział 1** – zawiera wprowadzenie do zagadnień związanych z dynamiką białka i do ogólnego modelu procesów przeniesienia elektronu wykorzystującego teorię procesów stochastycznych.
- **Rozdział 2** – zawiera metodologię badań numerycznych, szczegóły na temat przebiegu eksperymentu Monte Carlo wraz z napotykanymi problemami, informacje o generatorach liczb pseudolosowych i sieciach, napisanych programach i możliwości zastosowania przetwarzania równoległego wraz z dyskusją problemów obliczeniowych.
- **Rozdział 3** – poświęcony jest próbie modelowania dynamiki wibracyjnej na przykładzie bakteryjnego centrum fotosyntetycznego i dotyczy problemu pierwszego etapu procesu przeniesienia elektronu.
- **Rozdział 4** – poświęcony jest próbie modelowania dynamiki przejść konformacyjnych na przykładzie kompleksu cytochromu bc_1 i dotyczy ogólnego problemu przetwarzania energii swobodnej.

Zakończenie zawiera podsumowanie najważniejszych wyników przedstawionych w tej pracy, która jest numerycznym uzupełnieniem i weryfikacją wieloletnich badań prof. Michała Kurzyńskiego i współpracowników [5–11]. Nowe wyniki przedstawione w pracy doktorskiej zostały opublikowane w postaci dwóch prac [A, B], trzecia praca jest przygotowywana do publikacji [C].

Rozdział 1

Dynamika białka

W najprostszy sposób białko definiuje się jako polimer składający się z aminokwasów połączonych wiązaniami peptydowymi. Sama kolejność ułożenia aminokwasów dana jest przez strukturę pierwszorzędową, która narzuca określone możliwości tworzenia się struktur wyższych rzędów. Przyjęcie wybranej konformacji umożliwia spełnianie różnorodnych funkcji, jak np. wiązanie cząsteczek, katalizowanie reakcji chemicznych, kontrolowanie przepływu materii przez membrany, regulowanie stężenia metabolitów, działanie jako sensory i przekładnie molekularne [12].

Motywacją do badań były złożone układy takie jak fotosyntetyczne centrum reakcji czy elementy łańcucha oddechowego, w których otoczenie białkowe uczestniczy w procesach przeniesienia elektronu pomiędzy kolejnymi centrami. W ogólności proces przeniesienia elektronu może być rozpatrywany jako proces utleniania/redukcji, w którym jeden kofaktor traci elektron (ulega utlenieniu), a drugi akceptuje elektron (ulega redukcji). Zmianom ulegają nie tylko centra, pomiędzy którymi następuje przeniesienie elektronu, ale i otoczenie, którego dynamika wpływa na przebieg procesu.

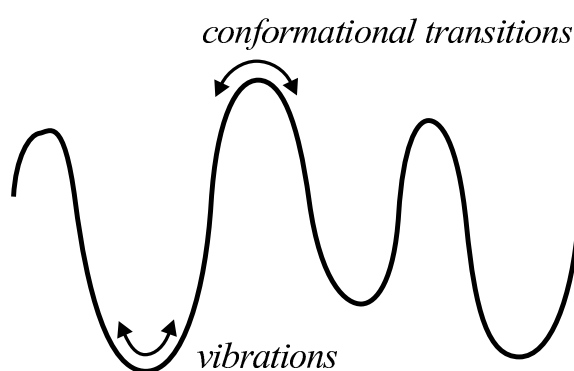
Każda makrocząsteczka białka posiada $3N$ stopni swobody, gdzie N to liczba atomów. Stopnie swobody określają organizację przestrzenną i dynamikę cząsteczki. Trzy z nich odpowiadają za ruch translacyjny cząsteczki jako całości, trzy kolejne za rotację, a pozostałe $3N-6$ nazywane są wewnętrznymi stopniami swobody. W ramach wewnętrznych stopni swobody wyróżniamy takie zmienne jak długości wiązań, kąty pomiędzy wiązaniami, czy też kąty dwuścienne przy rotacjach wokół wiązania pojedynczego. Wszystkie one tworzą złożoną strukturę wraz z odpowiednim wysokowymiarowym pejzażem energetycznym. Ogromna liczba atomów, a zatem i stopni swobody powoduje, że stworzenie jakiegokolwiek teorii (lub wykonanie obliczeń) dla układów biologicznych jest bardzo trudne. Sytuację komplikuje dodatkowo fakt, że obiektem zainteresowania są układy mezoskopowe. Patrząc na teoretyczne metody badawcze w zależności od skali układu, to na jednym końcu dominuje mechanika kwantowa, a na drugim fizyka statystyczna, jednakże pośrodku nie ma jeszcze dobrze rozwiniętej metodologii. Warto wspomnieć, że najprost-

sze białko, glicyna, składa się z 10 atomów, a to ma się nijak do obliczeń kwantowo-chemicznych wysokiej precyzji gdzie granicą jest też liczba około 10, ale elektronów.

W podejściu stosowanym w tej pracy nie wnikam w strukturę konkretnego białka, celem jest próba znalezienia najbardziej ogólnych modeli, które znajdą następnie zastosowanie do opisu różnych procesów, a następnie ich weryfikacja. Białko jest rozumiane jako „skrzynka”, która realizuje konkretną funkcję. Organizacja przestrzenna cząsteczki może być reprezentowana w komputerze jako ciąg liczb w pewnym formacie. Stąd ogólnie białko można rozumieć jako „procesor”, który przetwarza dane wejściowe na dane wyjściowe. Największym problemem jest sposób reprezentacji struktury „skrzynki”, natomiast sposób „przetworzenia” danych (program) to właśnie funkcja, co zresztą jest intuicyjnie zgodne z definicją funkcji znaną z języków programowania.

1.1 Dynamika vibracyjna a dynamika przejść konformacyjnych

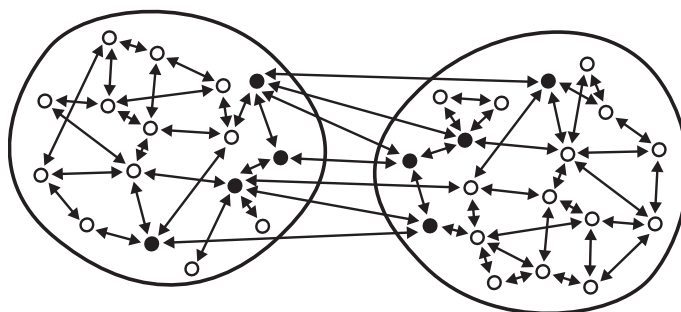
Hiperpowierzchnia energii potencjalnej składa się z ogromnej liczby lokalnych minimów rozdzielonych przez bariery energetyczne. Jako stany konformacyjne rozumiemy obszary przestrzeni konfiguracyjnej otaczające lokalne minima. Przy założeniu, że bariery pomiędzy stanami konformacyjnymi są wystarczająco wysokie, wewnętrzną dynamikę można rozdzielić na tłumione drgania (ang. *vibrations*) w ramach konkretnego podstanu konformacyjnego oraz przejścia konformacyjne (ang. *conformational transitions*), co przedstawia rysunek 1.1 [10].



Rysunek 1.1. Wibracje a przejścia konformacyjne – schemat. Za pracą [10].

Dynamika vibracyjna i konformacyjna charakteryzują się innymi skalami czasowymi procesów. Dynamika vibracyjna to okresy rzędu od 10^{-14} s (słabo tłumione zlokalizowane drgania rozciągające w grupach N-H oraz C-H) do 10^{-11} s (przetłumione drgania

kolektywne całej makromolekuły). Natomiast dla przejść konformacyjnych w warunkach fizjologicznych najkrótsze czasy relaksacji zaczynają się od 10^{-11} s [10].



Rysunek 1.2. Przykład modelu odwracalnej jednocząsteczkowej reakcji chemicznej pomiędzy stanami R (po lewej) i P (po prawej). Za pracę [8].

Na rysunku 1.2 przedstawiono ogólny model odwracalnej jednocząsteczkowej reakcji chemicznej pomiędzy stanami substratu R (po lewej) i produktu P (po prawej) w obrazie stochastycznej teorii szybkości reakcji. Każdy ze stanów chemicznych (substratu i produktu) składa się z wielu podstanów (białe i czarne kółka), reprezentujących podstany konformacyjne lub skwantowane stany vibracyjne. Pomiedzy podstanami zachodzą czysto stochastyczne przejścia (strzałki). Wyjście ze stanu R traktowane jest jako przejście pomiędzy wybranym podstanem nazwanym stanem przejściowym $R^\ddagger$ znajdującym się w R (czarne kółka), a stanem przejściowym $P^\ddagger$ w P. Stanów przejściowych może być wiele (w ogólności mogą różnić się energią – model fluktuujących barier) lub mogą być zredukowane do konkretnej pojedynczej bramy.

W związku z tym, że mówimy o zespole statystycznym podstanów tworzących pewien stan chemiczny makromolekuły, przy założeniu stałości temperatury, objętości (ciśnienia) i liczby cząstek używamy pojęcia energii swobodnej układu:

$$F = U - TS, \quad (1.1)$$

gdzie U oznacza energię, T temperaturę, a S entropię. W temperaturach powyżej 0 K na przebieg procesów wpływa wzajemny stosunek wkładów energetycznych i entropowych, który dodatkowo komplikuje dynamikę układu.

Ogniwem łączącym poruszane w tej pracy problemy jest kinetyka procesów biomolekularnych i próba stosowania teorii procesów stochastycznych do opisu dynamiki układów biologicznych. Szczególny nacisk położony jest na problem kinetyki przeniesienia elektronu jak i przetwarzania energii swobodnej. Standardowa kinetyka chemiczna zakłada szybką dynamikę wewnętrzną biomolekuł, tzn. zakłada, że przed każdym krokiem kinetycznym nastąpi dojście do równowagi częściowej. Wyniki eksperymentalne wskazują jednak na fakt, że dynamika wewnątrzcząsteczkowa może być tak wolna jak reakcje

biochemiczne lub nawet wolniejsza, a samo przeniesienie elektronu może być szybkie albo wolne. Typowe skale czasowe procesów biochemicznych zawierają się w granicach od mikrosekund do sekund [10]. Zwykle zatem dynamika wibracyjna jest za szybka by wpłynąć na reakcje i tylko dynamika przejść konformacyjnych wpływa na procesy zachodzące w układach biomolekularnych. Od tej reguły istnieją jednak wyjątki, np. bardzo szybkie procesy nieadiabaticznego przeniesienia elektronu (rzędu pikosekund), o których między innymi będzie mowa w tej pracy.

Mimo, że różne skale czasowe procesów powodują, że makroukłady mają złożoną dynamikę, to zasadne może być uproszczenie, że dynamika układu jest praktycznie kontrolowana przez najwolniejszy z procesów (wąskie gardło).

1.2 Procesy stochastyczne – przypadkowe błędzenie

Proces stochastyczny jest określony przez dyskretne czasy, które oznaczmy jako $t_1, t_2, t_3, \dots$, i przebiega na układzie składającym się ze skończonej liczby dyskretnych stanów $S_1, S_2, S_3, \dots$. Fakt, że w chwili t układ znajduje się w stanie X oznaczmy jako X_t , a warunkowe prawdopodobieństwo, że $X_{t_n} = S_{i_n}$ zapiszmy jako [13]

$$p(X_{t_n} = S_{i_n} | X_{t_{n-1}} = S_{i_{n-1}}, X_{t_{n-2}} = S_{i_{n-2}}, \dots, X_{t_1} = S_{i_1}). \quad (1.2)$$

Naszym obiektem zainteresowania jest szczególna grupa procesów stochastycznych, w których warunkowe prawdopodobieństwo jest niezależne od wszystkich stanów z wyjątkiem poprzedzającego (brak efektu pamięci):

$$p = p(X_{t_n} = S_{i_n} | X_{t_{n-1}} = S_{i_{n-1}}). \quad (1.3)$$

Takie procesy nazywane są procesami Markowa, a sekwencja stanów X_t nazywana jest łańcuchem Markowa [13]. Tak zredukowane prawdopodobieństwo warunkowe może być rozumiane jako prawdopodobieństwo w_{ji} przejścia ze stanu i do stanu j [13]:

$$w_{ji} = w(S_i \rightarrow S_j) = p(X_{t_n} = S_j | X_{t_{n-1}} = S_i). \quad (1.4)$$

Prawdopodobieństwa przejść powinny być nieujemne oraz sumować się do jedności, co zapisujemy jako $w_{ji} \geq 0$ oraz $\sum_j w_{ji} = 1$.

Prawdopodobieństwo, że układ w czasie t_n znajdzie się w stanie S_j zapisujemy jako

$$p(X_{t_n} = S_j) = p(X_{t_n} = S_j | X_{t_{n-1}} = S_i) p(X_{t_{n-1}} = S_i) = w_{ji} p(X_{t_{n-1}} = S_i), \quad (1.5)$$

natomiast zmianę prawdopodobieństwa w czasie dyskretnym zapisujemy przy pomocy równania różnicowego

$$p(X_{t_{n+1}} = S_j) = p(X_{t_n} = S_j) - \sum_i w_{ij} p(X_{t_n} = S_j) + \sum_i w_{ji} p(X_{t_n} = S_i), \quad (1.6)$$

które po uciągleniu czasu prowadzi do równania różniczkowego nazywanego fundamentalnym (ang. *master equation*) [13]:

$$\dot{p}(S_j, t) = \sum_i (w_{ji}p(S_i, t) - w_{ij}p(S_j, t)) . \quad (1.7)$$

Zauważmy, że obojętne jest czy uwzględniamy możliwość pozostawania w tym samym stanie w_{ii} , gdyż i tak tego typu człony ulegną redukcji.

Prawdopodobieństwo obsadzenia stanu S_j jest stałe w czasie zarówno dla stanu równowagi jak i stanu stacjonarnego. Z równania (1.7) otrzymujemy wtedy

$$\dot{p}(S_j, t) = 0 . \quad (1.8)$$

W stanie stacjonarnym cała suma w równaniu (1.7) jest równa 0, natomiast w stanie równowagi dodatkowo każdy ze składników sumy jest równy 0, czyli spełniony jest warunek równowagi szczegółowej (ang. *detailed balance condition*) z prawdopodobieństwem równowagowym $p^{\text{eq}}(S_j)$:

$$w_{ji}p^{\text{eq}}(S_i) = w_{ij}p^{\text{eq}}(S_j) , \quad (1.9)$$

który narzuca związki pomiędzy prawdopodobieństwami przejść.

Proces Markowa z pewnymi prawdopodobieństwami przejść, które określają ewolucję „wędrowca” (przypadkowe błędzenie) na sieci może być realizowany jako proces Monte Carlo. Czas jest dyskretny i wyrażony w krokach wędrowca na sieci składającej się z pewnej liczby stanów. Od definicji sieci zależy sposób dostania się do konkretnego stanu. Struktura geometryczna sieci określa model przestrzeni stanów wibracyjnych lub konformacyjnych, a błędzenie przypadkowe może być utożsamiane z dyfuzją w tej przestrzeni [10].

Jeżeli j numeruje stany chemiczne, to w przypadku najprostszej jednocząsteczkowej reakcji chemicznej $R \leftrightarrow P$, S_j przyjmuje dwie wartości, a równanie (1.7) można zapisać jako

$$\dot{p}(S_0, t) = -k_+p(S_0, t) + k_-p(S_1, t) \quad (1.10)$$

oraz

$$\dot{p}(S_1, t) = -k_-p(S_1, t) + k_+p(S_0, t) . \quad (1.11)$$

Po nałożeniu warunku normującego

$$p(S_0, t) + p(S_1, t) = 1 \quad (1.12)$$

równania (1.10) i (1.11) stają się równaniami kinetycznymi zależnymi od siebie, a k_+ i k_- stałymi szybkości reakcji „do przodu” i „do tyłu”. Z warunku danego równaniem (1.9) otrzymujemy

$$\frac{p^{\text{eq}}(S_1, t)}{p^{\text{eq}}(S_0, t)} = \frac{k_+}{k_-} \equiv K , \quad (1.13)$$

gdzie K jest stałą równowagi chemicznej. Odwrotność sumy stałych szybkości

$$\tau = (k_+ + k_-)^{-1} \quad (1.14)$$

jest czasem dochodzenie do pełnej równowagi chemicznej. Oczywiście dla bardziej złożonych schematów kinetycznych jak choćby reakcja enzymatyczna, odpowiednie równania fundamentalne i stałe równowagi mają bardziej złożoną postać [10]. W dalszej części dla uproszczenia będziemy posługiwali się oznaczeniami R i P odpowiednio dla substratu i produktu reakcji.

W stochastycznej teorii szybkości reakcji wykorzystujemy pojęcie średniego czasu pierwszego przejścia (ang. *mean first passage time*). W ogólności można stosować diagramy kinetyczne Hilla do wyznaczania średniego czasu pierwszego przejścia dla konkretnych schematów kinetycznych [14]. Nie zawsze jednak da się w prosty sposób takie wyrażenia wyprowadzić. Użyteczne okazują się tu symulacje Monte Carlo, gdyż czasy pierwszego przejścia to jedne z wielkości, które można dzięki nim policzyć.

Warto zauważyć, że dla przypadku reakcji nieodwracalnej równanie (1.14) zyskuje jeszcze prostszą postać, tzn. czas relaksacji jest odwrotnością jednej stałej szybkości. Czas relaksacji utożsamiamy z czasem pierwszego przejścia z pewnego stanu początkowego układu do stanu przejściowego.

Warto nadmienić, że w symulacjach typu Monte Carlo stan modelu zmienia się w sposób stochastyczny w zależności od użytej sekwencji liczb pseudolosowych. Nowy zestaw liczb spowoduje, że ewolucja układu będzie inna, jednakże pewna wartość średnia będąca ostatecznym wynikiem symulacji powinna być zgodna w zakresie błędu statystycznego.

Rozdział 2

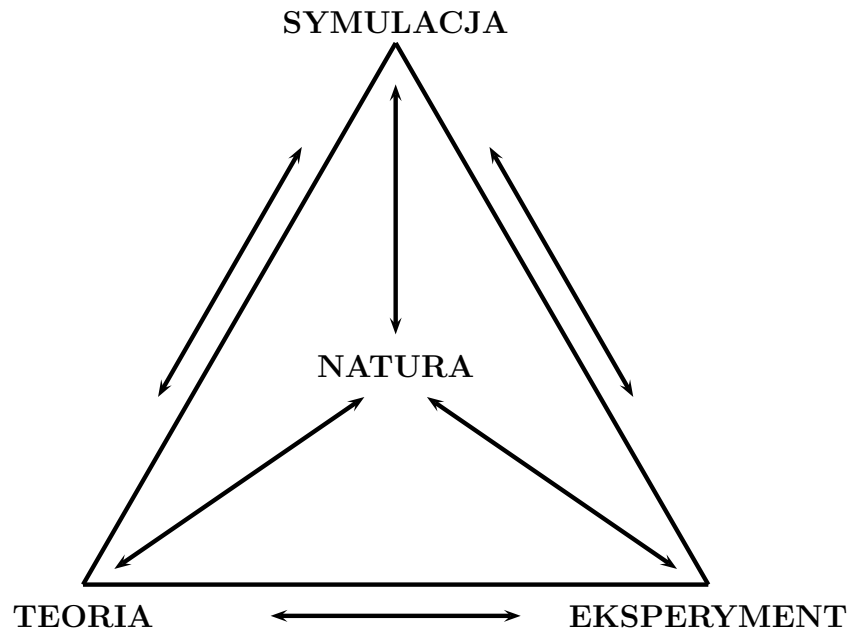
Metodologia badań numerycznych

Symulacje komputerowe wraz z postępem metod obliczeniowych i zasobów sprzętowych pozwalają na rozwiązywanie coraz większej gamy problemów. Obok teorii i eksperymentu są jakby osobną metodą badawczą nauk przyrodniczych. Uzupełniają one słaby postęp w teorii układów biologicznych, dla których metody doświadczalne są dość dobrze rozwinięte.

Ideą symulacji komputerowych jest zrozumienie właściwości fizycznych procesów tak dobrze jak to tylko możliwe, najlepiej przenosząc warunki eksperymentu na ich grunt. Nie zawsze jest jednak możliwy wystarczająco ścisły opis rzeczywistości, a zaproponowanie sensownych przybliżeń jest podstawą dobrej symulacji komputerowej (tak samo zresztą jak i dobrej teorii).

Schemat pokazujący związek pomiędzy symulacją, teorią a eksperymentem przedstawiono na rysunku 2.1 [13]. Granica rozdzielająca symulację od teorii i eksperymentu potrafi być płynna, a konkretne miejsce może znajdować się w środku trójkąta zamiast w jego wierzchołkach. Czasami symulacja zastępuje eksperyment (kiedy nie jest możliwe jego przeprowadzenie) będąc dodatkiem do teorii, innym znów razem zastępuje ścisłą teorię będąc krokiem w wyjaśnieniu faktów doświadczalnych (np. jeśli badane układy są bardzo złożone). W moich badaniach obecne były obie sytuacje, czasami wyjaśniano wyniki doświadczalne, czasami symulacja była używana w celu weryfikacji wyników analitycznych. Ponadto dzięki pewnemu wyborowi parametrów symulacje pozwalają na włączanie konkretnych efektów, które w rzeczywistych układach nie dają się rozdzielić.

Przed każdym eksperymentem badacz powinien „skalibrować” przyrząd pomiarowy. W moich symulacjach narzędziem są własnoręcznie napisane programy głównie w języku Fortran 77 i 90/95 (elementy w C) [15–17], a metoda „kalibracji” to sprawdzenie programów w znanych problemach. Po skalibrowaniu narzędzia można mieć większą pewność, że wyniki otrzymane dla nowych problemów (często bez szans na porównanie z wyrażeniami analitycznymi) będą poprawne.



Rysunek 2.1. Symulacja, teoria a eksperyment – schemat. Według książki [13].

2.1 Przebieg eksperymentu

Przypadkowe błędzenie może być symulowane na dowolnej sieci, która może być określona deterministycznie lub statystycznie, a także może ewoluować w czasie. W moich symulacjach sieć jest stała w całym przebiegu eksperymentu, a sąsiedzi ściśle określani. W ogólności w moich symulacjach sieć jest grafem nieskierowanym, czyli uporządkowaną parą $G = (V, E)$, gdzie V to zbiór wierzchołków, a E to rodzina dwuelementowych podzbiorów zbioru wierzchołków V , które nazywamy krawędziami. Wierzchołki to stany, a krawędzie to ścieżki ewolucji wędrowca. Eksperyment składa się z następujących podstawowych etapów:

1. Należy zdefiniować sieć oraz stan początkowy.
2. Ze stanu początkowego startuje wędrowiec, a ewolucja jego ruchu na sieci dana jest odpowiednimi prawdopodobieństwami przejść (zależnymi od parametrów związanych z konkretną symulacją).
3. Ewolucja układu zachodzi dzięki użyciu generatora liczb pseudolosowych.
4. Należy ustalić warunek końca symulacji – zależnie od typu symulacji i wyników, które chcemy otrzymać definiuje się punkt końcowy (np. obliczenia czasu pierwszego przejścia do pewnego stanu) albo czas ewolucji wyrażony w liczbie kroków (np. obliczanie strumieni czy też średniej odległości od stanu początkowego).

2.1.1 Błąd wyznaczenia wartości średniej

Należy pamiętać, że obliczenia typu Monte Carlo w związku z użyciem liczb pseudolosowych charakteryzują się pewnym błędem, który zależy od eksperymentu (liczba wędrówców lub czas ewolucji pojedynczego wędrowca). Aby stwierdzić czy mamy odpowiednią liczbę wędrówców można przeprowadzić kilka symulacji i policzyć błąd wyznaczenia wartości średniej. Wartość średnia $\bar{x}$ z n wartości x_i dana jest jako

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i. \quad (2.1)$$

Nie znamy rzeczywistej wartości wielkości mierzonej, stąd nie możemy określić prawdziwego błędu pojedynczego pomiaru. Zamiast tego używamy odchylenia δx_i (tzw. błąd pozorny pojedynczego pomiaru) danego wyrażeniem [18]

$$\delta x_i = x_i - \bar{x}. \quad (2.2)$$

W związku z tym, że średnie odchylenie jest równe zeru, sens ma dopiero średni kwadrat odchylenia nazywany wariancją [13, 19]:

$$\overline{\delta x^2} = \frac{1}{n} \sum_{i=1}^n (\delta x_i)^2 = \overline{x^2} - (\bar{x})^2. \quad (2.3)$$

W praktyce używa się wielkości zwanej odchyleniem standardowym średniej [13]:

$$\Delta = \sqrt{\frac{1}{n-1} \left(\overline{x^2} - (\bar{x})^2 \right)}, \quad (2.4)$$

gdyż obliczanie średniej kwadratu oraz kwadratu średniej danej wielkości bez problemu można włączyć w algorytm. Przy założeniu, że błąd pomiaru podlega rozkładowi normalnemu, prawdopodobieństwo, że prawdziwy błąd nie przekracza wartości $\pm\Delta$ wynosi 68,27 %, $\pm 2\Delta$ wynosi 95,45 %, a $\pm 3\Delta$ wynosi 99,73 % wyników [18, 19].

Wartość błędu bez wartości średniej nie dostarcza żadnej informacji o wielkości błędu, stąd użyteczny jest względny błąd procentowy definiowany jako [18]

$$\Delta_{\%} = \frac{\Delta}{\bar{x}} \cdot 100\%, \quad (2.5)$$

gdzie Δ jest błędem danym równaniem (2.4).

Przydatne dla nas będzie też obliczanie błędu wielkości wyrażającej się przez iloraz dwóch innych wielkości obarczonych pewnymi błędami. Jeśli $a = b/c$, a wielkość b i c dane są z błędami Δb i Δc , to błąd Δa obliczamy korzystając z wyrażenia

$$|\Delta a| = \left| \frac{\partial a}{\partial c} \right| |\Delta c| + \left| \frac{\partial a}{\partial b} \right| |\Delta b| = |a| \frac{|\Delta c|}{|c|} + \frac{|\Delta b|}{|c|}. \quad (2.6)$$

Oczywiście zmniejszanie błędów w symulacjach jest ograniczone zasobami sprzętowymi, jednakże wykonanie symulacji z dużym błędem nie ma sensu. Czas wyrażany w liczbie kroków jest związany z czasem rzeczywistym i zależnie od symulacji (dla różnych parametrów czy też dużej liczby węzłowców) może być bardzo długi, stąd potrzeba użycia obliczeń równoległych (o czym będzie mowa w dalszej części rozdziału) by sprostać wymaganiom statystyki. Osobnym tematem są np. błędy obciążenia wynikające bezpośrednio z architektury komputerów, na których prowadzimy obliczenia i które są błędami związanymi z błędnym napisaniem programu.

W wykonywanych symulacjach zachowana jest zasada ergodyczności, stąd średnia po zespole statystycznym jest równa średniej po wystarczająco długim czasie. W praktyce trudno określić co oznacza „wystarczająco długi czas”, tym bardziej, że może zmieniać się w szerokim zakresie zależnie od parametrów eksperymentu, określających czas dojścia do stanu równowagi lub stacjonarnego. Ponadto każdy generator ma swój okres (zwykle długi lecz skończony). Ważne jest też by rozkład liczb był jednorodny oraz by pomiędzy liczbami nie występowały korelacje. Z mojego doświadczenia wynika, że dobrym rozwiązaniem jest stosowanie podejścia mieszanego, np. pewna liczba symulacji z pewną liczbą węzłowców, co jednocześnie zapewnia częstszą zmianę zestawu liczb pseudolosowych.

Jeśli stany na sieci stanowią model stanów wibracyjnych lub konformacyjnych układu biologicznego, to można wyróżnić specjalne stany, które pozwalają na zachodzenie określonych reakcji chemicznych. W zależności czy taki stan jest jeden czy wiele można mówić o jednej bramie lub o całym zbiorze bram, które dodatkowo mogą różnić się energią [10].

2.1.2 Badane sieci

W tym podrozdziale przedstawiam trzy rodzaje sieci – sieć liniową, hipersześcian o różnej wymiarowości oraz trójkąt Sierpińskiego o różnym rzędzie. Wybór konkretnej sieci dyktowany jest często specyfiką problemu, jednakże później np. ustalenie liczby dostępnych stanów czy liczby sąsiadów nie jest już oczywiste i znów często zależy od możliwości obliczeniowych. Jeśli problem charakteryzuje się hierarchicznością, to można zastanowić się nad wykorzystaniem struktur fraktalnych. Z technicznego punktu widzenia najważniejsze jest zdefiniowanie listy lub macierzy sąsiadów jako metody reprezentacji powiązań pomiędzy stanami na dowolnej sieci wraz z prawdopodobieństwami przejść. Odległości pomiędzy stanami definiujemy jako tzw. odległości chemiczne, czyli odległości liczone jako najmniejsza liczba krawędzi, które należy przejść aby przemieścić się pomiędzy stanami.

Z siecią związany jest jej wymiar, a najważniejsze rodzaje wymiarów to [20]:

- euklidesowy: np. kwadrat $d = 2$, sześcian $d = 3$,
- fraktalny (Hausdorffa-Besicovitcha) – związek pomiędzy parametrem redukcji s , a liczbą części n , na które można podzielić układ:

$$n = s^{d_f}, \quad (2.7)$$

stąd

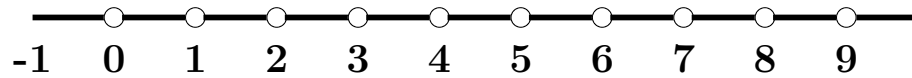
$$d_f = \frac{\ln(d+1)}{\ln s}, \quad (2.8)$$

- spektralny (fraktonowy):

$$d_s = 2 \frac{\ln(d+1)}{\ln(d+3)}. \quad (2.9)$$

Sieć liniowa

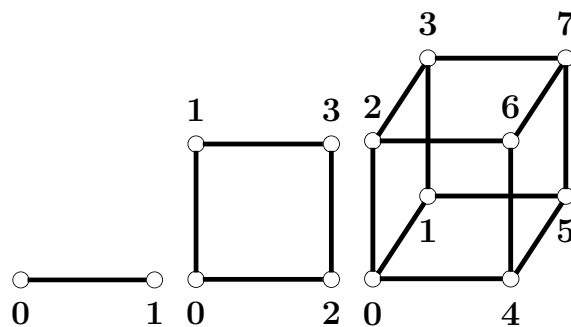
Sieć liniowa jest siecią jednowymiarową, a każdy stan posiada dwóch sąsiadów. Zakładamy, że stany są numerowane liczbami całkowitymi od 0 do $N - 1$, gdzie N oznacza liczbę stanów. Zasadniczo nie trzeba generować listy sąsiadów, gdyż wiadomo, że sąsiadami stanu m , są stany $m - 1$ oraz $m + 1$. Dodatkowe własności sieci są ustalane w zależności od typu symulacji (np. odbijający warunek brzegowy na jednym końcu). Dla prostoty można założyć, że odległość pomiędzy sąsiadami (koszt przejścia) jest stała i wynosi 1. Sposób generowania jest zatem prosty, bo obejmuje tylko określenie ile stanów sieć ma posiadać, a liczba ta może być dowolna. Odległość pomiędzy dwoma stanami można prosto określić odejmując mniejszy numer stanu od większego. Przykład sieci liniowej przedstawiono na rysunku 2.2.



Rysunek 2.2. Przykład sieci liniowej z zaznaczonymi stanami. Skoro stany numerujemy od 0, to można zdefiniować wirtualny stan -1 z warunkiem odbijającym. Natomiast w prawo sieć można rozbudowywać w nieskończoność.

Hipersześcian

Przykłady sieci typu hipersześcianu dla wymiarów od 1 do 3 przedstawiono na rysunku 2.3. Różne metody wizualizacji dla wyższego wymiaru można odnaleźć w literaturze [21, 22].



Rysunek 2.3. Sieć typu hipersześcianu o wymiarach od 1 do 3 wraz z zaznaczonymi numerami stanów (numeracja od 0).

Zakładamy, że stany numerowane są liczbami całkowitymi od 0 do $N - 1$, gdzie N oznacza liczbę stanów. Liczba stanów (wierzchołków) oraz liczba sąsiadów każdego stanu zależy od wymiaru – liczba sąsiadów jest dokładnie równa wymiarowi d , natomiast liczba stanów dana jest wyrażeniem

$$L = 2^d. \quad (2.10)$$

Warto zaznaczyć, że w przypadku hipersześcianu nie potrzebujemy warunków brzegowych i jest to w porównaniu do sieci liniowej sieć składająca się ze skończonej liczby stanów.

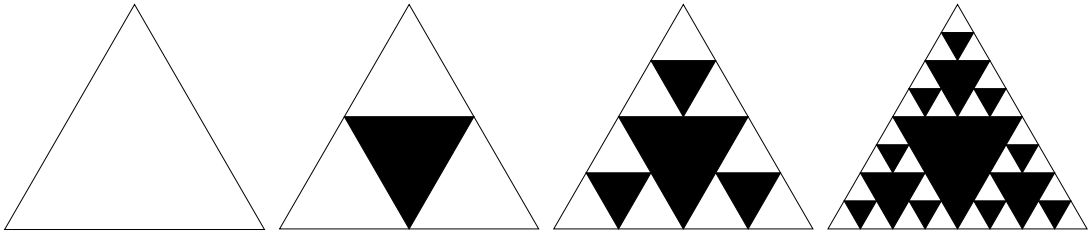
Osobnym problemem jest stworzenie listy sąsiadów, szczególnie przy numerowaniu stanów liczbami całkowitymi. Warto zauważyć, że używając systemu dwójkowego, każdy stan można zapisać na d bitach, gdzie d to wymiar hipersześcianu. Sąsiadem stanu jest stan różniący się o 1 bit, ponadto odległość dwóch stanów od siebie (najkrótsza ścieżka) jest równa sumie różnic w bitach, czyli np. dla 8-wymiarowego hipersześcianu dwa stany odległe o 8 będą różniły się na wszystkich ośmiu bitach. Stąd też wniosek, że maksymalna odległość stanów jest równa wymiarowi hipersześcianu. Przeliczając liczby z systemu dwójkowego na dziesiętny otrzymamy stany numerowane liczbami całkowitymi, które używane są w programach.

Trójkąt Sierpińskiego

Trójkąt Sierpińskiego jest fraktalem, tzn. obiektem, którego wymiar zazwyczaj nie jest liczbą całkowitą, a jego wygląd nie zależy od skali i przenosi się na następną generację (samopodobieństwo). Jest on przykładem fraktala deterministycznego (ściśle samopodobnego), w przeciwieństwie do tzw. fraktali losowych (statystycznie samopodobnych), których przykładem są np. klastery perkolacyjne [20].

Trójkąt Sierpińskiego generujemy rekurencyjnie w następujący sposób (patrz rysunek 2.4):

1. Należy stworzyć trójkąt równoboczny na płaszczyźnie (rzęd 0).
2. Należy wyznaczyć punkty będące środkami trzech boków i usunąć trójkąt równoboczny, którego wierzchołkami są wyznaczone punkty.
3. Otrzymujemy trzy nowe trójkąty równoboczne, których długość boku wynosi połowę długości poprzedniego trójkąta.
4. Powyższą procedurę należy zastosować dla nowych trójkątów.



Rysunek 2.4. Trójkąt Sierpińskiego rzędu 0, 1, 2 oraz 3. W wierzchołkach nie zaznaczono numerów stanów. Sposób generowania opisano w treści rozdziału.

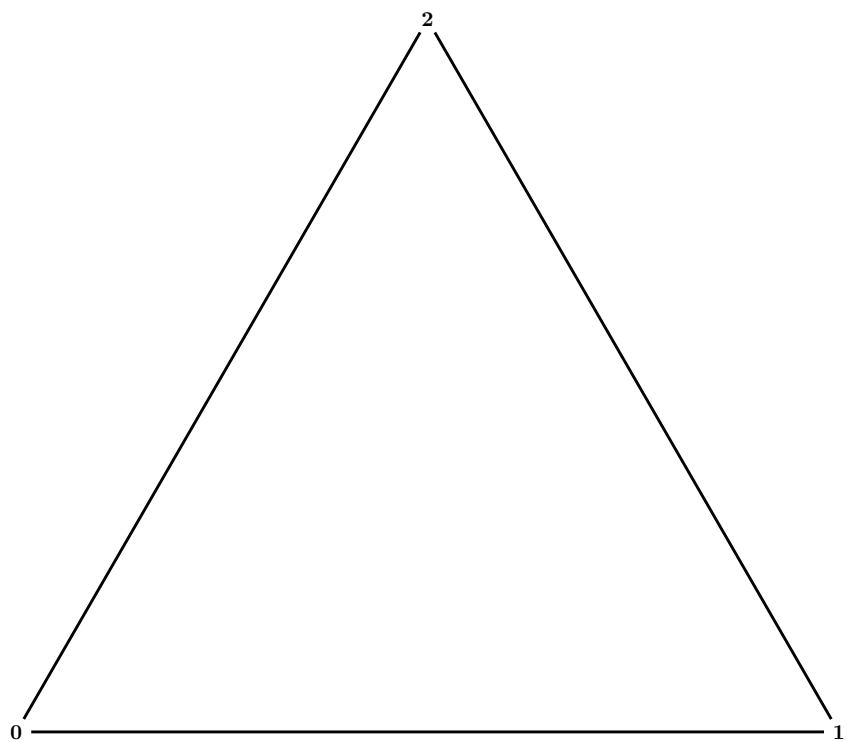
Zakładamy, że stany numerowane są liczbami całkowitymi od 0 do $N - 1$, gdzie N oznacza liczbę stanów. Warto zwrócić uwagę na wybrany sposób numerowania stanów zgodny z rekurencyjnym generowaniem sieci (rysunki 2.5, 2.6, 2.7, 2.8). W przypadku trójkąta Sierpińskiego konieczne jest zdefiniowanie warunków brzegowych dla wierzchołków znajdujących się w największym trójkącie (stany 0, 1, 2), gdyż posiadają one dwóch sąsiadów, podczas gdy wszystkie inne stany posiadają czterech sąsiadów. Wybrano periodyczne warunki brzegowe, tzn. gdyby wziąć trójkąt Sierpińskiego trzeciego rzędu (rysunek 2.8), to stany 0, 1 i 2 mają następujących sąsiadów: 0: {9,10,1,2}, 1: {24,26,0,2}, 2: {40,41,0,1}.

Liczba sąsiadów nie zależy od rzędu trójkąta Sierpińskiego i wynosi 4, natomiast liczba stanów (wierzchołków) zależy od rzędu i dana jest wyrażeniem

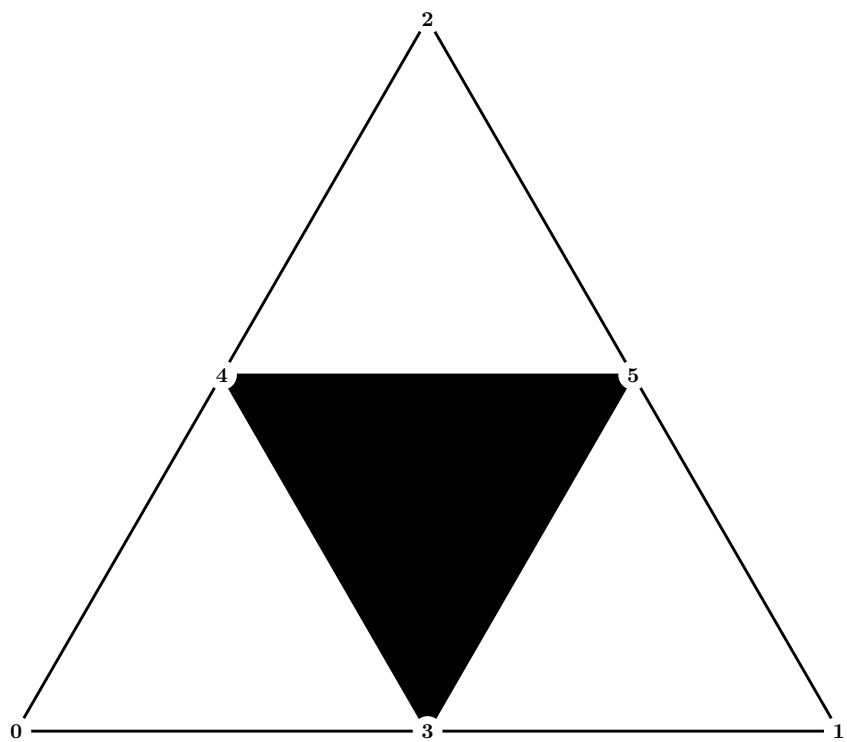
$$L = 3 + 3^1 + 3^2 + \dots + 3^R = \frac{3}{2} (3^R + 1) , \quad (2.11)$$

gdzie R to rząd trójkąta Sierpińskiego.

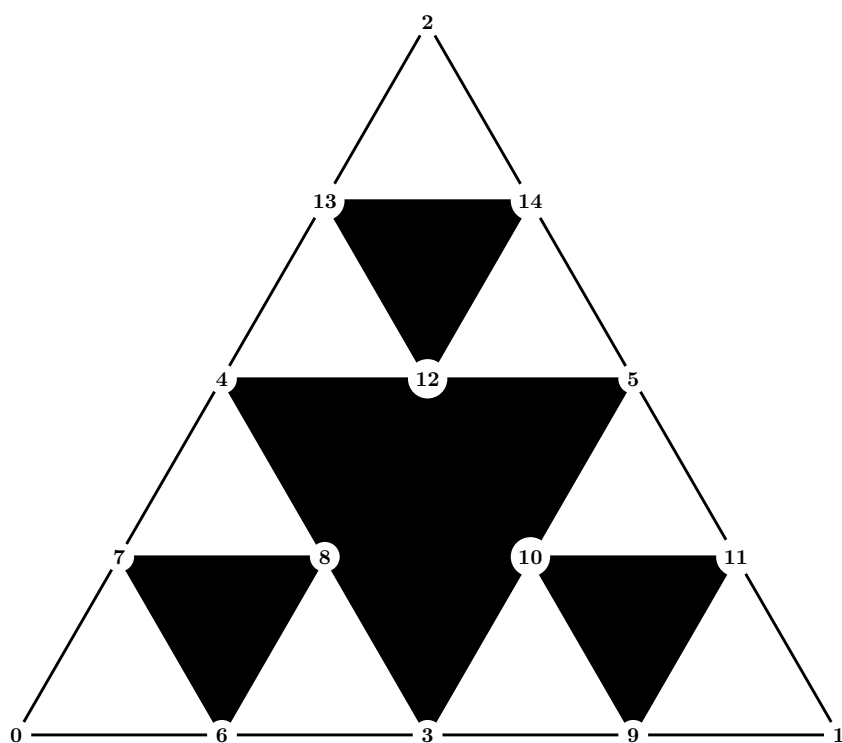
Wymiar euklidesowy trójkąta Sierpińskiego jest stały (równy 2). Zakładam, że odległość pomiędzy wierzchołkami (waga krawędzi) wynosi zawsze 1. Własnoręcznie napisany program rekurencyjnie generuje trójkąt Sierpińskiego dowolnego rzędu, tworzy rysunki zamieszczone w tej pracy (kod źródłowy dla PSTricks [23]) z numeracją stanów lub bez, generuje listę odległości pomiędzy stanami (zdefiniowanie najkrótszej ścieżki nie



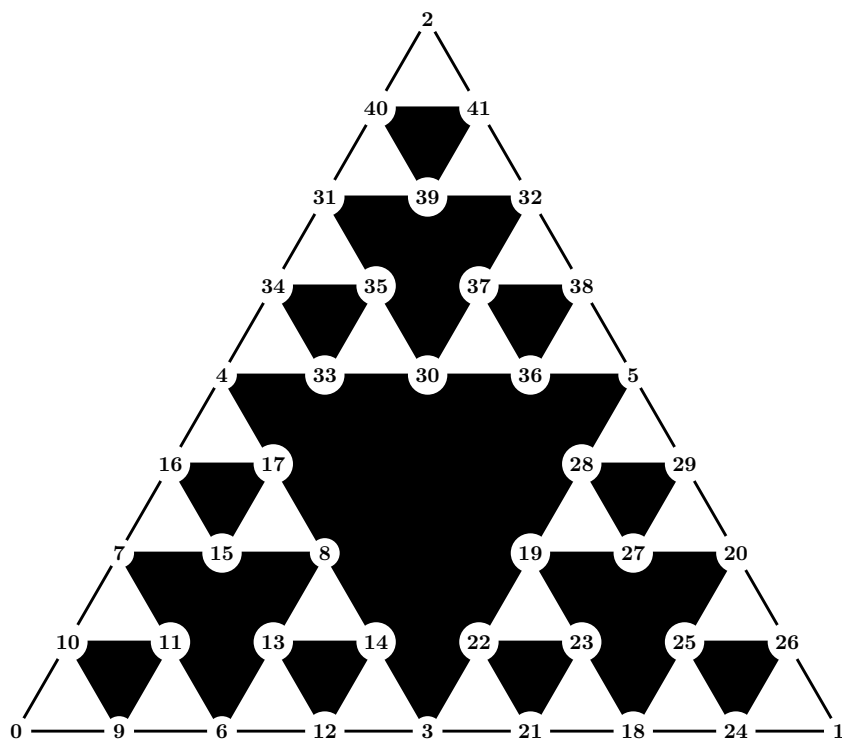
Rysunek 2.5. Trójkąt Sierpińskiego rzędu 0 z zaznaczonymi stanami w wierzchołkach. Stany numerowane są od 0.



Rysunek 2.6. Trójkąt Sierpińskiego rzędu 1 z zaznaczonymi stanami w wierzchołkach. Stany numerowane są od 0.



Rysunek 2.7. Trójkąt Sierpińskiego rzędu 2 z zaznaczonymi stanami w wierzchołkach. Stany numerowane są od 0.



Rysunek 2.8. Trójkąt Sierpińskiego rzędu 3 z zaznaczonymi stanami w wierzchołkach. Stany numerowane są od 0.

jest tak łatwe jak w sieciach regularnych), a także listę sąsiadów gotową do włączenia do programów do symulacji (aby z każdą symulacją nie generować od nowa sieci).

Wymiar fraktalny (równanie (2.8)) i spektralny (równanie (2.9)) wynoszą dla trójkąta Sierpińskiego odpowiednio (współczynnik podziału wynosi 2)

$$d_f = \frac{\ln(d+1)}{\ln s} = \frac{\ln(3)}{\ln 2} = 1,58496 \quad (2.12)$$

oraz

$$d_s = 2 \frac{\ln(d+1)}{\ln(d+3)} = 2 \frac{\ln(3)}{\ln(5)} = 1,36521. \quad (2.13)$$

Metod na wyznaczanie najkrótszej ścieżki pomiędzy wierzchołkami grafu jest wiele. Wybrano algorytm Dijkstry [24], który generuje następujące informacje:

- najmniejszy koszt dojścia od wybranego stanu początkowego do pozostałych stanów (założono, że waga wszystkich krawędzi jest równa 1) wyrażony w liczbie kroków,
- poprzednika na trasie (stąd można odtworzyć, przez które stany przebiega najkrótsza ścieżka).

Oczywiście jeśli chcemy obliczyć odległość tylko pomiędzy dwoma wybranymi stanami, to warunkiem końca wykonywania algorytmu jest osiągnięcie odpowiedniego stanu końcowego, przez co można skrócić jego działanie. Używając algorytmu w pętli można wygenerować odległości dla wszystkich par, jednakże należy pamiętać, że np. dla trójkąta Sierpińskiego wysokiego rzędu dane te zabierają ogromne ilości miejsca w pamięci, zatem należy ostrożnie podchodzić do generowania takich danych.

2.1.3 Problem generowania liczb pseudolosowych

Warto na początek zauważyć, że liczby generowane przy użyciu komputera nie są liczbami całkowicie losowymi, gdyż są tworzone wg określonego algorytmu zależącego od zarodka generatora i po pewnym czasie zwanym okresem będą powtarzały się. Istnieje wiele generatorów liczb pseudolosowych stosujących różne podejścia, także mieszane [13,25]. W moich programach użyłem generatorów dostępnych w kompilatorach. W Fortranie 77 (i w następnych wersjach by zapewnić kompatybilność) dostępna jest funkcja `rand`, a jako zarodka można użyć aktualnego czasu zwracanego dzięki funkcji `itime` [17]. Odpowiednie przykłady kodu (generator **G1**) przedstawiono poniżej:

1. Pobieranie aktualnego czasu (`integer, dimension(3) :: tArr`):

```
call itime(tArr)
```

2. Inicjalizacja generatora (integer :: czas):

```
czas = rand(tArr(1)+tArr(2)+tArr(3))
```

3. Generowanie liczby pseudolosowej:

```
random_number = rand()
```

Jednakże zaleca się przejście na nowszy wbudowany generator używając funkcji o nazwie RANDOM_NUMBER, która używa generatora o okresie przekraczającym [17]

$$2^{123} = 10633823966279326983230456482242756608.$$

Odpowiednie przykłady kodu (generator G2) przedstawiono poniżej:

1. Generowanie zarodka można zrealizować następująco (także używając aktualnego czasu) [17]:

```
SUBROUTINE init_random_seed()  
  INTEGER :: i, n, clock  
  INTEGER, DIMENSION(:), ALLOCATABLE :: seed  
  CALL RANDOM_SEED(size = n)  
  ALLOCATE(seed(n))  
  CALL SYSTEM_CLOCK(COUNT=clock)  
  seed = clock + 37 * (/ (i - 1, i = 1, n) /)  
  CALL RANDOM_SEED(PUT = seed)  
  DEALLOCATE(seed)  
END SUBROUTINE
```

2. Inicjalizacja generatora:

```
CALL init_random_seed()
```

3. Generowanie liczb:

```
CALL RANDOM_NUMBER(random_numbers)
```

Dużą zaletą nowego sposobu generowania jest fakt, że random_numbers może być zarówno skalarą jak i macierzą, co umożliwia wygenerowanie zestawu liczb przy pomocy jednego wywołania.

Wspomniane generatory losują liczby z przedziału [0; 1). Wiadomo, że lista sąsiadów

indeksowana jest liczbami całkowitymi (sąsiad 1, 2, 3, 4 itp.). Dla niektórych symulacji (szczególnie gdy prawdopodobieństwa przejść są identyczne) warto przeskalować losowane liczby rzeczywiste na liczby całkowite, dodatkowo często na inne przedziały. Dla przykładu jeśli chcielibyśmy losować sąsiada z przedziału $[1; d]$ (np. hipersześcian d -wymiarowy z d -sąsiadami), to korzystamy z wyrażenia

```
sasiad = int(1.0 + d*random_number)
```

gdzie `random_number` to wylosowana pojedyncza liczba rzeczywista.

Sprawdzenie poprawności generatora

Metod sprawdzania poprawności generatora jest wiele [13], jednakże fakt, że generator przejdzie pewną grupę testów nie oznacza, że przejdzie wszystkie. Ponadto każdy generator ma swój okres, a w niektórych przypadkach wiele zależy od wyboru zarodka. Standardowe metody obejmują: generowanie par liczb i szukanie regularności, generowanie liczb i wrzucanie do małych przedziałów następnie sprawdzanie jednorodności rozkładu, testowanie korelacji pomiędzy liczbami w ciągu, czy też sprawdzanie generatora w znanych problemach fizycznych, np. w przypadkowym błędzeniu. Sprawdzono użyte generatory w teście pierwszym i ostatnim. Na potrzeby testów zakłada się, że prawdopodobieństwa przejść na sieciach są identyczne, należy jednak pamiętać, że w realnych problemach często będą różne i zależne od parametrów symulacji.

1. Szukanie regularności w generowanych parach liczb.

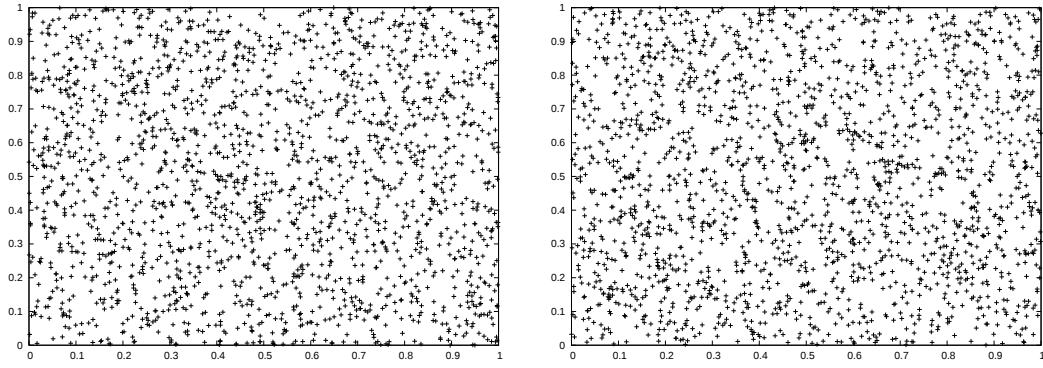
Wygenerowano 2000 par liczb i przedstawiono na rysunku 2.9. Przykładowy kod źródłowy może wyglądać następująco:

```
program pary_liczb_f90
implicit none
integer :: i
integer, parameter :: ile_liczb = 4000
real, dimension(ile_liczb) :: random_numbers

CALL init_random_seed()
CALL RANDOM_NUMBER(random_numbers)

do i=1, ile_liczb, 2
    print *, random_numbers(i), random_numbers(i+1)
enddo

end program pary_liczb_f90
```



Rysunek 2.9. Test generatora liczb pseudolosowych. Pary wylosowanych liczb przedstawiono na wykresie. Po lewej stronie generator **G1**, po prawej generator **G2** (patrz tekst).

Brak regularności w strukturze powoduje, że oba generatory przeszły ten test. Struktury są różne, gdyż generatory są różne. Jednakże w przypadku pojedynczego generatora zmiana zarodka daje ten sam efekt.

2. Średni kwadrat odległości w zależności od czasu przypadkowego błędzenia na trójkącie Sierpińskiego.

Wędrowiec na fraktalu spełnia zależność [26]

$$\langle r^2(t) \rangle \sim t^{2/(2+\theta)}, \quad (2.14)$$

gdzie parametr θ jest zwykle niezerowy. Zależność pomiędzy wymiarem fraktalnym i spektralnym przedstawia się następująco [26]:

$$d_s = \frac{2d_f}{2 + \theta}. \quad (2.15)$$

Zatem

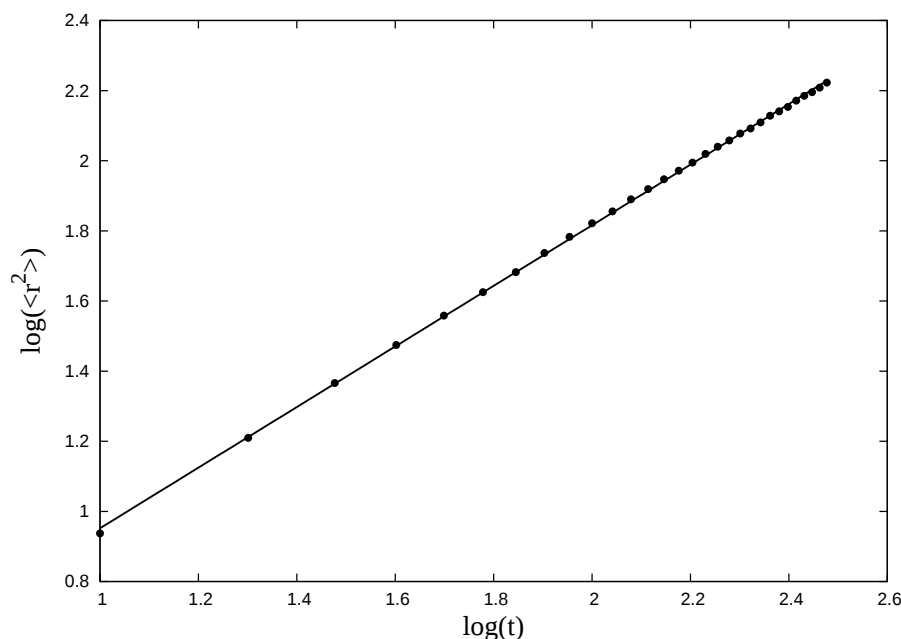
$$\langle r^2(t) \rangle \sim t^{d_s/d_f}. \quad (2.16)$$

Wcześniej obliczono dla trójkąta Sierpińskiego wartość wymiaru spektralnego i fraktalnego (równania (2.12) i (2.13)). Po prostym podstawieniu otrzymujemy

$$\langle r^2(t) \rangle \sim t^{d_s/d_f} = t^{0,86135}, \quad (2.17)$$

co odpowiada $\theta = 0,32193$ (tzw. subdyfuzja).

Wykonano symulację przypadkowego błędzenia na trójkącie Sierpińskiego rzędu 10 z 10^7 wędrowców, start ze stanu 9000, czas symulacji w liczbie kroków od 10 do 300 z krokiem co 10. Prawdopodobieństwa przejść były identyczne we wszystkich kierunkach (równe 0,25 do każdego z czterech sąsiadów). Dla każdego punktu przeprowadzono nową symulację z nowym zestawem liczb pseudolosowych. Wykres w skali logarytmicznej zależności średniego kwadratu odległości od czasu symulacji przedstawiono na rysunku



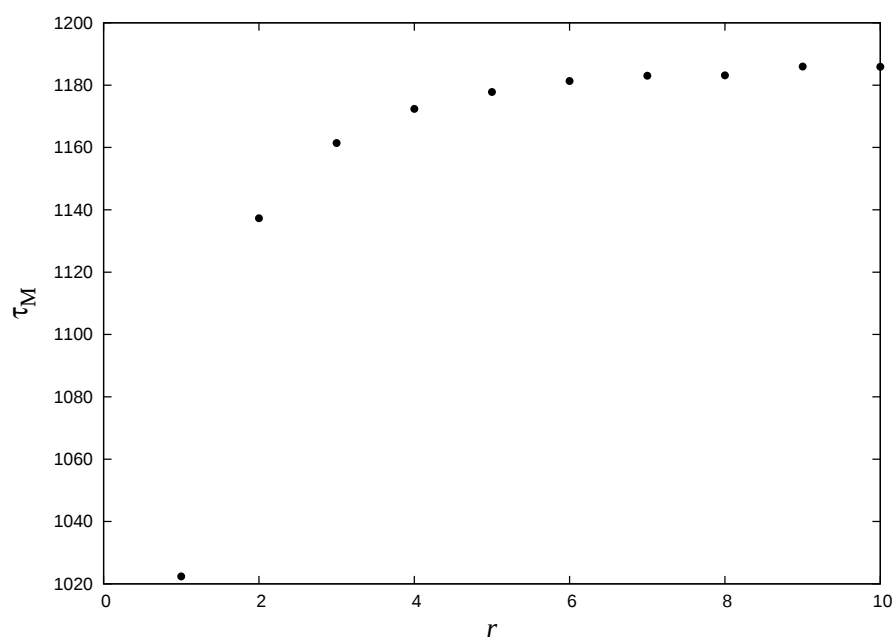
Rysunek 2.10. Test generatora liczb pseudolosowych. Zależność średniego kwadratu odległości od czasu przedstawiona w skali logarytmicznej. Kropkami zaznaczono wyniki symulacji, a linią ciągłą najlepsze dopasowanie do wyników symulacji.

2.10. Dopasowanie prostej do wyników symulacji zrealizowano przy użyciu GNUPLOT 4.2 [27], a dopasowana prosta dana jest wzorem $y = 0,863x + 0,089$. Otrzymana wartość wykładnika $0,863 \pm 0,002$ (błąd wyznaczony także przez GNUPLOT) jest zgodna z wynikiem analitycznym ($0,861$), co powoduje, że generator **G2** przeszedł także ten test. Generator **G1** także przeszedł ten test jednak nie umieszczono już podobnego wykresu po raz drugi (wykładnik równy $0,863 \pm 0,002$, co pozostaje w zgodzie z wynikiem analitycznym oraz wynikiem z użyciem drugiego generatora, który dodatkowo wygenerował inny zestaw liczb).

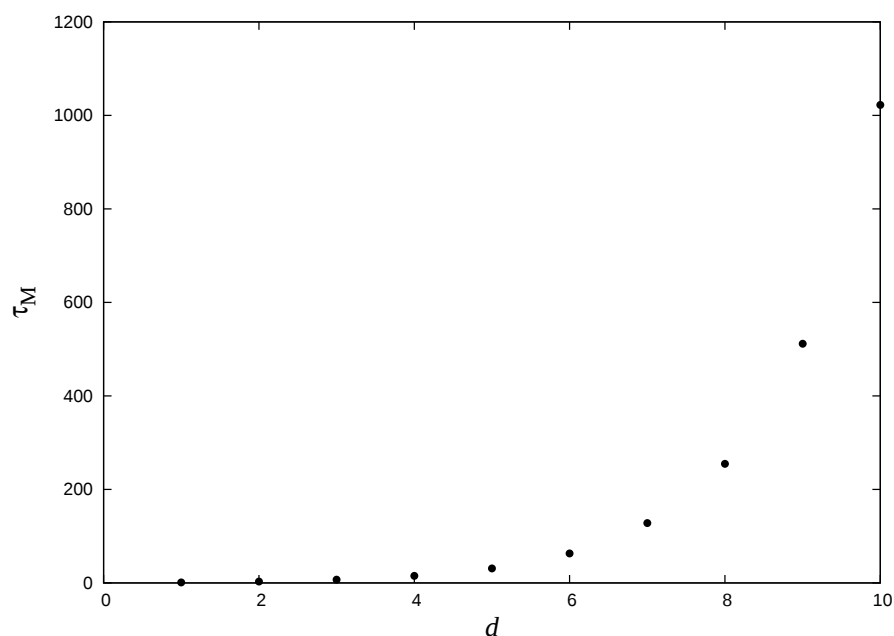
2.2 Przetwarzanie równoległe

Przetwarzanie równoległe wykorzystywane jest w wielu problemach obliczeniowych. Podstawowe założenie to posiadanie przynajmniej dwóch procesorów, które potrafią ze sobą współpracować przy rozwiązywaniu zadania. W symulacjach przypadkowego błędzenia zapotrzebowanie na moc obliczeniową może drastycznie rosnąć w zależności od typu symulacji, rozmiaru sieci, liczby węzłów, a także definicji prawdopodobieństw przejść. W pewnym momencie może okazać się, że użycie przetwarzania równoległego jest niezbędne aby w realnym czasie otrzymać wyniki obliczeń z żadaną dokładnością.

Na rysunkach 2.11 i 2.12 pokazano jak średni czas pierwszego przejścia τ_M wyra-



Rysunek 2.11. Zależność średniego czasu pierwszego przejścia τ_M wyrażonego w liczbie kroków na hipersześcianie o wymiarze $d = 10$ od odległości r pomiędzy stanem początkowym i końcowym. Warto zauważyć, że skala dla wartości τ_M nie zaczyna się od 0 oraz, że dla $r = 8$ wystąpiły niewielkie fluktuacje, które można zmniejszyć przez powiększenie liczby węzłów.



Rysunek 2.12. Zależność średniego czasu pierwszego przejścia τ_M wyrażonego w liczbie kroków od wymiaru d hipersześcianu dla odległości pomiędzy stanem początkowym i końcowym $r = 1$.

Tabela 2.1. Rzeczywisty czas obliczeń (w sekundach) T_o oraz błąd procentowy wyznaczenia $\Delta_{\%}\tau_M$ średniego czasu pierwszego przejścia τ_M w zależności od efektywnej liczby wędrówców (rozumianej jako iloczyn liczby wędrówców i liczby symulacji z ich udziałem) N_w na hipersześcianie o wymiarze $d = 5$.

N_w	$\Delta_{\%}\tau_M$	T_o
10^6	$9 \cdot 10^{-2}$	4
10^7	$3 \cdot 10^{-2}$	36
10^8	$9 \cdot 10^{-3}$	359
10^9	$3 \cdot 10^{-3}$	3580

żony w liczbie kroków zmienia się zależnie od wymiaru hipersześcianu (dla zachowanej odległości pomiędzy stanami) oraz dla różnych odległości między stanem początkowym i końcowym (dla zachowanego wymiaru). Warto zwrócić uwagę, że jest to średni czas (kolejni wędrówcy mogą kończyć symulację wcześniej lub później), a wyniki prezentowane na wykresach wskazują jedynie na zmianę rzędu czasu oraz dodatkowo pokazują, że dla odpowiednio dużych odległości następuje pewnego rodzaju nasycenie. Jednakże rzeczywisty czas obliczeń zależy także od żądanej dokładności wyznaczenia wartości średniej – przykładowe wartości dokładności i rzeczywistego czasu obliczeń przedstawiono w tabeli 2.1 (prawdopodobieństwa przejść były równe $1/d$, gdzie d jest wymiarem hipersześcianu, w tym przypadku $d = 5$). Rzeczywisty czas zależy od parametrów komputera wykorzystanego do obliczeń i zainstalowanego systemu (obliczenia na procesorze o architekturze 64-bitowej z 32-bitowym systemem będą trwały dłużej niż z 64-bitowym systemem, co zostanie pokazane w dalszej części rozdziału) oraz bieżącego obciążenia i ilości potrzebnej pamięci. Czas obliczeń był liczony dzięki funkcji `etime`, która zwraca czas w sekundach od wywołania programu (do obliczania czasów wykonania fragmentu kodu można użyć funkcji `cpu_time`) [17].

Testy wykonano na komputerze z procesorem Intel® Pentium® D 820 2.8 GHz i 1024 MB RAM z systemem Debian 5.0 (architektura i686). Błąd był liczony względem liczby symulacji z pewną liczbą wędrówców korzystając z równania (2.4). Symulacja była rozpoczynana w stanie początkowym 0, a kończona w stanie końcowym 31 (odległość $r = 5$). Średni czas pierwszego przejścia wyrażony w liczbie kroków dla tego typu symulacji wynosi około 43 kroki (można porównać z rysunkiem 2.12).

W tabeli 2.1 zamieszczono względny błąd procentowy wyznaczenia τ_M dla efektywnej liczby wędrówców rozumianej jako iloczyn liczby wędrówców i liczby symulacji z

ich udziałem. Warto zauważyć, że prawie godzinne obliczenia dały wartość średnią z błędem na poziomie 0,003 % (wartość 42,665 z błędem 0,001), co dla niektórych obliczeń może być wartością błędu wciąż za dużą (np. jeśli obliczamy wartość wyrażenia, w których występują sumy i różnice czasów pierwszego przejścia). Z danych w tabeli 2.1 można też wyciągnąć pewne wnioski co do skalowania się błędu obliczeń i w ten sposób próbować oszacować potrzebny czas, co jest szczególnie istotne w przypadku gdy z komputerów do obliczeń korzysta większa grupa osób. Dodatkową trudność stanowi fakt, że błędy wyznaczania średniego czasu pierwszego przejścia zależą będą także od typu i rozmiaru sieci (liczba sąsiadów, istnienie wąskich gardeł i tzw. martwych końców).

2.2.1 MPI i MPICH

Z technicznego punktu widzenia wyróżniamy dwa typy architektur komputerów, są to tzw. *shared memory* (procesory dzielą wspólną pamięć) oraz *interconnection-network* (oddzielne komputery tworzą sieć komputerową, a komunikacja realizowana jest przez sieć) [28]. Można wyobrazić sobie też połączenie obu podejść, np. komputery wieloprocessorowe (lub również powszechne wielordzeniowe) są połączone siecią komputerową. Oczywiście każde podejście ma swoje zalety i wady, np. w klastrach komputerowych wadliwy komputer można łatwo wymienić, lecz sieć komputerowa ma niższą przepustowość niż gdy procesory dzielą wspólną pamięć.

Komputery i system operacyjny to nie wszystko aby przeprowadzić obliczenia równoległe. Potrzebujemy jeszcze jakiegoś mechanizmu komunikacji pomiędzy procesorami. Języki programowania równoległego można podzielić na następujące klasy [29]: paradygmaty nowe lub nietradycyjne (np. Strand), rozszerzenia do już istniejących języków (np. Compositional C++), języki oparte na równoległości danych (np. Fortran 90, High Performance Fortran) oraz biblioteki umożliwiające urównoleglanie (np. Message Passing Interface).

W odróżnieniu od specjalnych języków programowania (których kompilatory są często płatne) bardzo ciekawy jest standard Message Passing Interface [30] w implementacji MPICH [31, 32]. MPI nie jest osobnym językiem programowania, gdyż programy pisane są w standardowych językach (np. C lub Fortran), a komunikacja między procesami realizowana jest dzięki wywołaniom specjalnych funkcji dostarczanych ze standardem MPI (ang. *library-based approach*). Zbiór procesów, które mogą komunikować się ze sobą, to tzw. *komunikator*, a nazwa najprostszego to `MPI_COMM_WORLD` (wystarcza do większości programów). Tworzenie nowych komunikatorów stosowane jest m.in. gdy procesy mają być podzielone na grupy. Każdy proces w komunikatorze ma unikalny identyfikator, który jest liczbą całkowitą od 0 (proces macierzysty) do $N - 1$, gdzie N to liczba procesów w komunikatorze. MPICH jest darmowy, może współpracować z róż-

nymi kompilatorami, zaczynając od bezpłatnych, dostępnych np. w systemie Debian [33], aż do płatnych rozwiązań.

Druga wersja standardu MPI [30] zawiera ponad 200 funkcji, wiele z nich ma dodatkowe opcje. Jednakże do napisania prostych i wydajnych programów równoległych wystarczy 6 podstawowych funkcji. Nazwy tych funkcji wraz ze znaczeniem zostały przedstawione w tabeli 2.2.

Tabela 2.2. Podstawowe funkcje MPI.

Nazwa funkcji	Znaczenie
<code>MPI_Init</code>	rozpoczyna równoległą część kodu
<code>MPI_Comm_size</code>	zwraca liczbę procesów w komunikatorze
<code>MPI_Comm_rank</code>	zwraca identyfikator każdego procesu
<code>MPI_Send</code>	wysyła wiadomość do innego procesu
<code>MPI_Recv</code>	odbiera wiadomość od innego procesu
<code>MPI_Finalize</code>	kończy równoległą część kodu

Program równoległy może zawierać także części wykonywane sekwencyjnie, jednakże jakakolwiek równoległa część kodu musi znaleźć się pomiędzy funkcjami rozpoczynającymi i kończącymi obliczenia równoległe (tabela 2.2), a w preambule programów w Fortranie musi znaleźć się `include 'mpif.h'`. Składnię poszczególnych funkcji można odnaleźć w literaturze [30, 34, 35].

W przypadku komunikacji point-to-point standardem jest tzw. *blocking communication*, tzn. proces wysyłający wiadomość jest blokowany dopóki proces odbierający nie odbierze wiadomości, co potencjalnie prowadzi do gorszej synchronizacji procesów. Warto jednak zauważyć, że oprócz prostej komunikacji point-to-point procesy mogą komunikować się z wieloma innymi przy pomocy funkcji kolektywnych. Dwie przykładowe to `MPI_Broadcast` (rozsyła wiadomość do wszystkich procesów w komunikatorze) oraz `MPI_Barrier` (używana do synchronizacji procesów). W moich programach wykorzystuję też `MPI_Reduce` z opcją `MPI_SUM` (zbieranie wyników i sumowanie).

Podstawowe pytanie jakie należy sobie zadać brzmi: które procesy powinny się komunikować i jak często? Większa wydajność obliczeń związana jest z minimalną liczbą zdarzeń komunikacyjnych oraz bardzo obciążającymi pojedynczymi zadaniami obliczeniowymi.

2.2.2 Przygotowanie komputera do obliczeń równoległych

Poniżej w punktach wymieniono zadania, które wg moich doświadczeń należy wykonać zakładając, że chcemy skonfigurować pojedynczy komputer wielordzeniowy. Rozszerzenie na inne komputery wiąże się z zainstalowaniem systemu, a następnie zdalnym udostępnieniem odpowiednich katalogów z MPICH i równoległymi programami.

1. Zainstaluj system operacyjny (np. Linux Debian [33]), zainstaluj kompilatory, ssh i inne potrzebne pakiety.
2. Ustaw zmienną systemową RSHCOMMAND (zakładamy, że komunikacja będzie odbywać się przez ssh):

```
export RSHCOMMAND='ssh'
```

3. Zadbaj o zawartość plików hosts, hosts.allow, hosts.deny oraz stwórz klucze ssh by zapewnić komunikację bezhasłową:

```
ssh-keygen -t rsa
```

Do `authorized_keys2` (w katalogu `.ssh` konta użytkownika, stwórz jeśli nie istnieje) dodaj zawartość pliku `id_rsa.pub`. Od tej chwili powinno być możliwe łączenie się przy użyciu ssh bez hasła, co można sprawdzić (jako `hostname` należy wpisać adres IP lub nazwę komputera):

```
ssh hostname
```

4. Pobierz MPICH1 [31]. Rozpakuj i skopiuj np. do `/usr/local`, wejdź do katalogu `mpich-1.2.7p1`, uruchom skrypt konfiguracyjny (`prefix` wskazuje miejsce instalacji), następnie skompiluj i zainstaluj:

```
./configure --with-device=ch_p4 --with-comm=shared  
--prefix=/usr/local/mpich127p1
```

```
make
```

```
make install
```

5. Do `.bashrc` dodaj ścieżki do skryptów (postać zależna od katalogu z MPICH dodatkowo skompilowanego z wybranym kompilatorem Fortranu lub C) oraz wskaż używany kompilator:

```
export PATH=/usr/local/mpich127p1/bin:$PATH
```

6. Skonfiguruj `/usr/local/mpich127p1/share/machines.LINUX` stosując format:

```
nazwa_komputera:liczba_procesorów
```

7. Poprawność konfiguracji można sprawdzić skryptem

```
/usr/local/mpich-1.2.7p1/util/tstmachines
```

wywołanym z opcją `-v` (w katalogu z tym skryptem znajduje się kolejny podkatalog `machines` z plikiem `machines.LINUX`, który należy skonfigurować do testów jak plik w poprzednim punkcie). Jeśli wszystko działa poprawnie będziemy używali tylko katalogu `/usr/local/mpich127p1`.

8. Skompiluj plik z programem `walker.f` (stworzony zostanie plik wykonywalny o nazwie `walker`):

```
mpif77 -o walker walker.f
```

9. Uruchom program o nazwie `walker` na zadanej liczbie procesorów (za zmienną `liczba_procesorów` wstaw liczbę całkowitą):

```
mpirun -np liczba_procesorów walker
```

Wraz z pojawieniem się stabilnej dystrybucji Debian 5.0 Lenny nastąpiła migracja z `g77` na `gfortran`. Niestety MPICH w wersji 1 nie jest do końca kompatybilny z `gfortran` i prostszym rozwiązaniem jest przejście na MPICH2 [32], którego konfiguracja i sposób użycia są trochę inne. Migracja na `gfortran` i ogólnie na standard Fortran 90/95 ma wiele zalet, choć wymusza między innymi zmiany w składni kodu. Dla uproszczenia przedstawię konfigurację na jednym komputerze, np. na wielordzeniowym komputerze, który wiele osób ma na biurku, a nie wykorzystuje w pełni mocy obliczeniowych. Punkty od 1 do 3 w konfiguracji dla MPICH1 pozostają bez zmian. W punkcie 4 wystarczy w wielu przypadkach użyć (należy oczywiście rozpakować pobraną wersję MPICH2):

```
./configure --prefix=/usr/local/mpich2  
make  
make install
```

W punkcie 5 zmianie ulegnie ścieżka, np:

```
PATH=/usr/local/mpich2/bin:$PATH; export PATH
```

Zamiast punktów 6-9 wykonujemy następujące czynności:

1. W katalogu domowym tworzymy plik `.mpd.conf` z hasłem (plik powinien mieć prawa do zapisu i odczytu tylko przez właściciela):

```
secretword=nasze_hasło
```

2. Uruchamiamy w tle demona `mpd`:

```
mpd &
```

3. Kompilujemy program w Fortranie 90 używając `mpif90` (`mpif77` nadal jest dostępny):

```
mpif90 -o walker walker.f
```

4. Uruchamiamy program przy pomocy `mpiexec` (`mpirun` nadal jest dostępny lecz zawiera mniej opcji, warto zauważyć zmianę opcji `-np` na `-n`):

```
mpiexec -n liczba_procesorow walker
```

Po zakończeniu obliczeń można zatrzymać demona `mpd` poleceniem `mpdallexit`, natomiast przed i w trakcie obliczeń można używać `mpdtrace` w celu sprawdzenia, na których węzłach klastra można wykonywać obliczenia.

Dalsze szczegóły dotyczące używania i konfiguracji MPICH2 (w szczególności rozszerzanie obliczeń na kolejne komputery i nowa definicja komputerów w klastrze) można odnaleźć w dokumentacji (*User's Guide* oraz *Installer's Guide*) [36, 37].

Oczywiście MPICH może współpracować z różnymi kompilatorami. W przypadku takich, które dodatkowo ustawiają zmienne systemowe (np. Lahey Fortran) należy zadbać o to, by odpowiednio ustawić odpowiednie zmienne w `environment` dla `ssh`, gdyż standardowe zmienne systemowe (np. te w `.bashrc`) nie są przekazywane jeśli łączymy się zdalnie z innymi komputerami przez `ssh` (zakładamy, że kompilatory zainstalowane są tylko na węźle dostępowym, potrzebne katalogi montowane są w przypadku większej liczby komputerów zdalnie przez sieć przy użyciu np. NFS). Warto wspomnieć, że MPICH można zainstalować i używać w swoim katalogu domowym – wtedy nie jest konieczne posiadanie uprawnień administratora.

Najprostszy program równoległy w Fortranie, który może posłużyć do testów (wypisuje informację o identyfikatorze i liczbie procesów w komunikatorze) wygląda następująco:

```

program hello
implicit none
include 'mpif.h'
integer :: kim_jestem, ilu_nas_jest, blad
call MPI_Init(blad)
call MPI_Comm_rank(MPI_COMM_WORLD, kim_jestem, blad)
call MPI_Comm_size(MPI_COMM_WORLD, ilu_nas_jest, blad)
print *, "Jestem ", kim_jestem, " z ", ilu_nas_jest, "!"
call MPI_Finalize(blad)
end program hello

```

2.2.3 Przyspieszenie i wydajność

Czasy obliczeń zależą od wielu czynników. Aby móc porównywać jakość przetwarzania równoległego na różnych komputerach wprowadzono dwa pojęcia, które są niezależne od rzeczywistego czasu wykonania programu: przyspieszenie i wydajność. Przyspieszenie definiuje się jako

$$S_N(R) = \frac{T_1(R)}{T_N(R)}, \quad (2.18)$$

gdzie $T_N(R)$ oznacza czas potrzebny by uruchomić zadanie o rozmiarze R na N procesorach. Jeśli podzielimy przyspieszenie przez liczbę procesorów otrzymamy wydajność

$$W_N(R) = \frac{S_N(R)}{N}. \quad (2.19)$$

Teoretyczne limit dla przyspieszenia to liczba procesorów, a dla wydajności to 1 (100 %). Skalowalność obliczeń to osobny problem – dodanie nowych procesorów wcale nie musi oznaczać znaczącego wzrostu wydajności (może być wprost przeciwnie). Każdy program składa się z części szeregowej i równoległej, a wydajność spada gdy część szeregową zaczyna dominować (prawo Amdahla). Zasadniczo efektywność programu przy zwiększaniu liczby procesorów będzie maleć do zera [22]. Pozostaje kwestia obliczania czasu wykonania kodu równoległego – może on być mierzony przy użyciu funkcji `MPI_WTime`.

W obliczeniach zarówno sekwencyjnych jak i równoległych pierwsze zadanie to stworzenie algorytmu i użycie odpowiednich struktur danych. Dodatkowo w programowaniu równoległym pojawia się problem podziału zadania na podzadania, zapewnienia komunikacji i zebrania wyników z poszczególnych procesów. W obliczeniach Monte Carlo należy zadbać dodatkowo o to, aby zarodek do generowania liczb pseudolosowych był różny dla każdego procesu. Oczywiście jest, że istnieją zadania, których zrównoleglić się nie da, gdyż niemożliwy jest podział na podzadania. Przykładem w obliczeniach Monte

Carlo może być pojedyncza symulacja przypadkowego błędzenia, gdzie każdy krok zależy od poprzedniego. W następnym podrozdziale pokażę jak problem ten można rozwiązać. Przykłady przedstawione w dalszej części pracy dotyczą sieci homogenicznych, tzn. składających się z takich samych procesorów, natomiast pojawienie się heterogeniczności znacząco komplikuje obliczenia i przyczynia się do spadku wydajności.

Podstawowe etapy programowania równoległego można w uproszczeniu przedstawić następująco [34]:

1. Wymyślenie algorytmu oraz wybór struktur danych.
2. Podział algorytmu na podzadania.
3. Ustalenie, które procesy muszą komunikować się ze sobą.
4. Mapowanie, czyli przyporządkowanie podzadań do procesorów.

2.2.4 Problemy obliczeniowe i wyniki

Zrównoleglanie przypadkowego błędzenia jednego wędrowca jest niemożliwe, gdyż kolejny krok zależy od poprzedniego, jednakże liczenie średnich czasów pierwszego przejścia można bez problemu zrównoleglić. Podstawą przeprowadzenia obliczeń jest tzw. mapowanie podzadań na procesory. Idealne zrównoważenie obciążeń (ang. *load balance*) zachodzi wtedy, gdy każdy proces otrzymał identyczne zadanie pod względem czasu wykonania. Jednakże występuje to niezwykle rzadko, chociażby dlatego, że często proces macierzysty dodatkowo zbiera wyniki i wykonuje operacje wejścia/wyjścia. Przedstawienie podstawowych typów mapowania takich jak blokowe, cykliczne i blokowo-cykliczne można odnaleźć w literaturze [34]. W rozważanym przypadku średniego czasu pierwszego przejścia można założyć, że na każdy procesor przydzielamy taką samą liczbę wędrowców. Jednakże jeśli chcielibyśmy uzyskać w tym względzie dowolność, to warto skorzystać z mapowania z użyciem funkcji modulo, które charakteryzuje się nierównoważeniem obciążenia (ang. *load imbalance*) równym maksymalnie jednemu wędrowcowi (pierwotnie to wyrażenie wprowadzono dla elementów macierzowych) [A].

Jeśli zdarzenia komunikacyjne są rzadkie i do tego zaniedbywalnie krótkie w stosunku do problemu obliczeniowego, to można zaniedbać komunikację kolektywną i pozostawić komunikację typu point-to-point. W podejściu master-slave proces macierzysty (z którego zostały rozdzielone zadania na resztę procesów) także wykonuje obliczenia, a dodatkowo obsługuje inne procesy – zbiera dane, wykonuje operacje wejścia/wyjścia itd.

Zakładamy, że proces macierzysty nie musi znać wszystkich pojedynczych czasów pierwszego przejścia, a interesuje nas tylko średnia z każdego procesu z osobna. W wersji

point-to-point proces zbierający dane wywołuje poniższą procedurę w pętli po reszcie procesów (zmienna `zrodlo`)

```
call MPI_Recv(srednia, 1, MPI_DOUBLE_PRECISION, zrodlo, &
              1, MPI_COMM_WORLD, stat, blad)
```

natomiast każdy proces wysyłający wywołuje

```
call MPI_Send(srednia, 1, MPI_DOUBLE_PRECISION, 0, 1, &
              MPI_COMM_WORLD, blad)
```

Konieczne jest zatem przesłanie/odebranie jednej zmiennej o nazwie `srednia` typu podwójnej precyzji. Ważny jest też piąty argument tzw. `tag`, który rozróżnia komunikaty gdyby przyszły w odwrotnej kolejności niż zostały wysłane. Proces macierzysty sumuje średnie otrzymane od pozostałych procesów, dodaje własną i dzieli przez liczbę procesów.

Natomiast w przypadku komunikacji kolektywnej każdy proces wywołuje

```
call MPI_Reduce(srednia, suma, 1, MPI_DOUBLE_PRECISION, &
                MPI_SUM, 0, MPI_COMM_WORLD, blad)
```

Do procesu 0 przesyłana jest jedna zmienna podwójnej precyzji o nazwie `srednia`, a proces 0 sumuje zmienne odebrane z pozostałych procesów (opcja `MPI_SUM`) i zapisuje je w zmiennej `suma`. Oczywiście ostateczną sumę średnich należy następnie podzielić przez liczbę procesów.

Równoległe liczenie średniego czasu pierwszego przejścia i innych wielkości można schematycznie zapisać w następujący sposób [A]:

```
call MPI_Init(blad)
call MPI_Comm_rank(MPI_COMM_WORLD, id, blad)
call MPI_Comm_size(MPI_COMM_WORLD, N, blad)
{GENEROWANIE ZARODKA W SPOSÓB RÓWNOLEGŁY}
do i = 1, liczba_symulacji
  do j = 1, liczba_wędrówców_w_każdej_symulacji
    {EWOLUCJA POJEDYNCZEGO UKŁADU-WĘDROWCA}
  enddo
enddo
{SUMOWANIE ŚREDNICH, MPI_Reduce}
call MPI_Finalize(blad)
```

Jak wynika z powyższego kodu podstawowym elementem są pętle, które rozdzielają zadania. W przypadku stosowania mapowania każdy proces musi dodatkowo sprawdzić, czy powinien dane podzadanie wykonać. Powyższy schemat tyczy się także i innych symulacji – w pętli może znajdować się procedura z dowolnym warunkiem końca (osiągnięcie stanu całkowicie absorbującego, wykonanie określonej liczby kroków itd.).

Warto zauważyć, że w powyższym schemacie skorzystano z funkcji kolektywnej `MPI_Reduce` z opcją `MPI_SUM`, która sumuje średnie czasy pierwszego przejścia z każdego procesu (liczone przez każdy proces dla wielu pojedynczych czasów pierwszego przejścia). Gdyby zależało nam na komunikacji point-to-point, to powyższy schemat zyskałby kolejną pętlę:

```
call MPI_Init(blad)
call MPI_Comm_rank(MPI_COMM_WORLD, id, blad)
call MPI_Comm_size(MPI_COMM_WORLD, N, blad)
{GENEROWANIE ZARODKA W SPOSÓB RÓWNOLEGŁY}
do i = 1, liczba_symulacji
  do j = 1, liczba_wędrowców_w_każdej_symulacji
    {EWOLUCJA POJEDYNCZEGO UKŁADU-WĘDROWCA}
  enddo
enddo
if (id .eq. 0) then
  {ZBIERANIE W PĘTLI WYNIKÓW Z INNYCH PROCESÓW, MPI_Recv}
else
  {WYSYŁANIE WYNIKU DO PROCESU 0, MPI_Send}
endif
call MPI_Finalize(blad)
```

Liczbę wędrowców można wybrać dowolnie lecz powinna wystarczyć do przeprowadzenia symulacji. Dobrze jest wybrać taką liczbę, by liczba wędrowców była podzielna bez reszty przez liczbę procesorów, choć w przypadku obliczeń średniego czasu pierwszego przejścia pojawia się dodatkowo problem niezrównoważenia obciążeń wynikającego z różnicy w czasach ewolucji. Warto zaznaczyć, że w przypadku gdy dodatkowo liczba kroków jest ustalona i potrzebujemy np. średnią odległość po pewnej liczbie kroków albo wartość strumienia, to czas obliczeń staje się identyczny, co dodatkowo zrównoważa obciążenia.

Warto podkreślić, że zarodek w obliczeniach równoległych także musi być przygotowany równolegle (nie trzeba stosować specjalnych algorytmów równoległych). Możemy użyć np. aktualnego czasu, ale należy go zmodyfikować, bo inaczej kolejne procesory miałyby te same zestawy liczb pseudolosowych (może zdarzyć się, że uruchomią się w tej

samej godzinie, minucie i sekundzie albo rzeczywisty czas uruchomienia będzie inny ale dwa różne komputery będą miały nieznacznie przesunięte względem siebie zegary systemowe). Najprostszym rozwiązaniem jest dodanie iloczynu id (identyfikator procesu) i N (liczba procesów) do sumy aktualnej godziny, minuty i sekundy, np. dla generatora **G1**

```
rand(tArr(1)+tArr(2)+tArr(3)+id*N)
```

a dla **G2**

```
seed = (clock + 37 * (/ (i - 1, i = 1, n) /))+id*N
```

Problem złego generowania można pokazać na przykładzie, gdzie w pierwszej kolumnie umieszczono numer procesu, a w drugiej ciąg liczb pseudolosowych. W pierwszym przypadku nie zastosowano dodatkowo iloczynu id i N – widać, że zestawy liczb wylosowane przez oba procesy są identyczne:

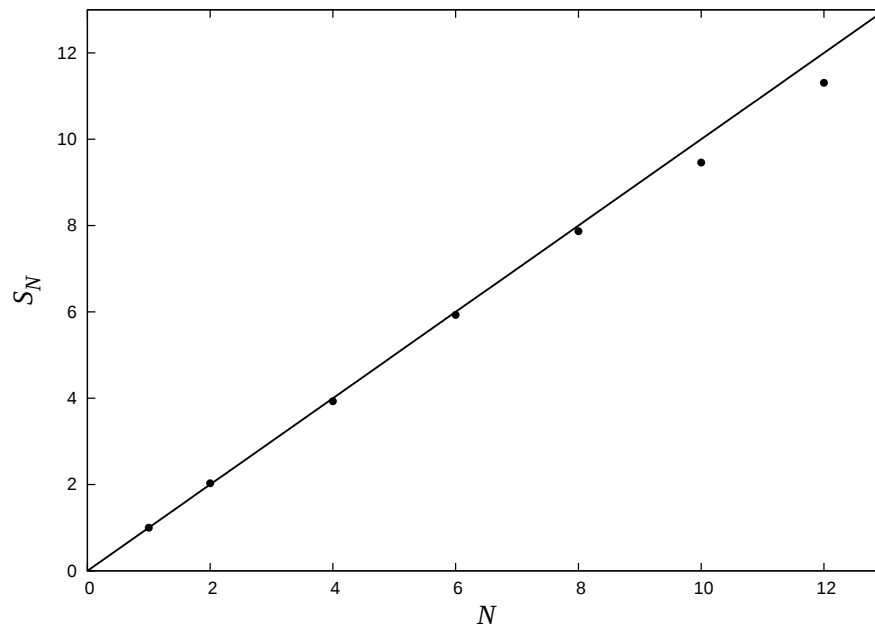
```
0  0.2646522493742661893
0  0.2286739764524849643
0  6.4152144415453450914E-0002
1  0.2646522493742661893
1  0.2286739764524849643
1  6.4152144415453450914E-0002
```

w drugim przypadku został wprowadzony iloczyn id i N :

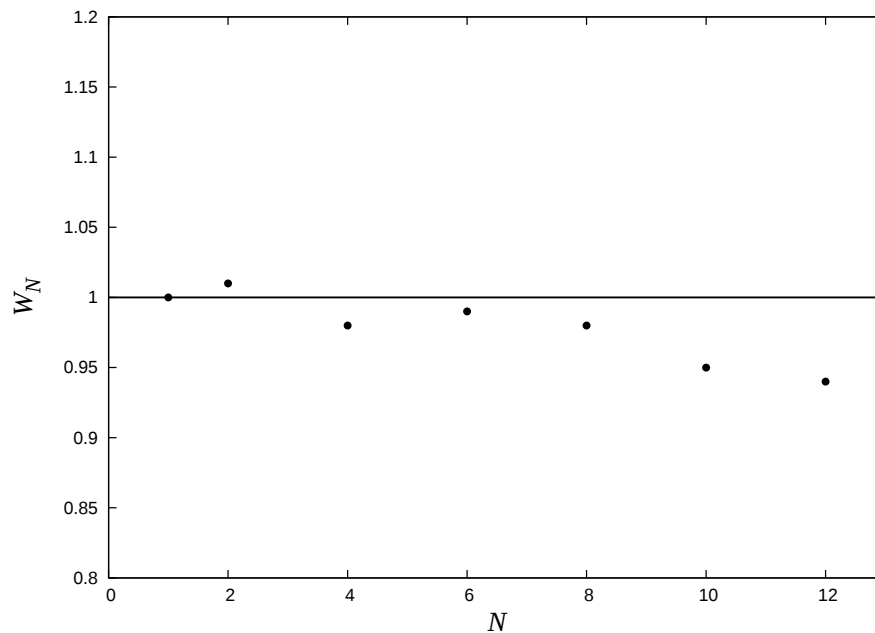
```
0  0.8711254414013441503
0  0.1368578684750689371
0  5.6794255077164005198E-0002
1  0.4198970205806118750
1  0.2456028461827439034
1  0.8796190128707528952
```

Na rysunkach 2.13 i 2.14 przedstawiono przyspieszenie i wydajność dla obliczeń równoległych średniego czasu pierwszego przejścia dodatkowo wzbogaconych o użycie technologii Hyper-Threading (HTT, hiperwątkowość) [38]. Symulacja przypadkowego błędzenia była wykonana na sieci liniowej z całkowicie absorbującym stanem (prawdopodobieństwa przejść w obu kierunkach nie były równe, a szczegóły ich ustalania można odnaleźć w rozdziale dotyczącym dynamiki wibracyjnej).

Wydajność obliczeń jest dość dobra, a odchylenie od liniowości nawet dla 12 procesorów jest niewielkie. Jeśli zauważymy, że pojedynczy czas pierwszego przejścia może być elementem macierzy indeksowanym numerem wędrowca i numerem symulacji, to w przeciwieństwie do równoległych obliczeń zwykłych elementów macierzowych (patrz



Rysunek 2.13. Przyspieszenie S_N obliczania średniego czasu pierwszego przejścia w zależności od liczby procesorów N . Linia ciągłą zaznaczono najlepszy możliwy wynik teoretyczny. Procesory Intel® Xeon® 2.8 GHz z technologią Hyper-Threading. Za pracą [A].



Rysunek 2.14. Wydajność W_N obliczania średniego czasu pierwszego przejścia w zależności od liczby procesorów N . Linia ciągłą zaznaczono najlepszy możliwy wynik teoretyczny. Procesory Intel® Xeon® 2.8 GHz z technologią Hyper-Threading. Za pracą [A].

praca [A]), po włączeniu technologii Hyper-Threading superliniowość nie była obserwowana dla równoległych obliczeń średniego czasu pierwszego przejścia, tzn. wydajność nie przekroczyła 100 %. Im większa liczba wędrówców na każdym procesorze, tym niezrównoważenie obciążeń wynikające z losowej ewolucji każdego wędrówca będzie coraz mniejsze.

Warto wrócić też do problemu pojedynczych komputerów wielordzeniowych, które wiele osób posiada na swoich biurkach. Wykonano obliczenia równoległe na dwóch rdzeniach dla trzech problemów obliczeniowych: liczenie elementów macierzowych danych analitycznym wyrażeniem, średniego czasu pierwszego przejścia (tym razem prawdopodobieństwa równe we wszystkich kierunkach, wersja point-to-point oraz kolektywna) oraz strumieni (wersja point-to-point oraz kolektywna, więcej informacji o tego typu symulacjach w rozdziale o dynamice konformacyjnej). Zbadano też wpływ wyboru procesora (mniejsza częstotliwość taktowania i więcej pamięci cache oraz nowsza generacja, a większa częstotliwość taktowania a mniej pamięci cache oraz starsza generacja) oraz użycia systemu 32-bitowego oraz 64-bitowego z procesorem 64-bitowym (który to wpływ występuje też dla obliczeń sekwencyjnych). Ponadto porównano wydajność programów z komunikacją point-to-point i kolektywną. Każdy eksperyment wykonano trzy razy, a mierzone czasy (a stąd wydajności) dotyczyły wykonania części równoległej (funkcja `MPI_WTime`). Komputer **K1** posiadał dwurdzeniowy procesor Intel® Core™2 Duo E6550 2.33 GHz (4096 KB pamięci cache), 2048 MB RAM. Komputer **K2** posiadał dwurdzeniowy procesor Intel® Pentium® D 2.80 GHz (1024 KB pamięci cache), 1024 MB RAM. Testowano wydajność obliczeń na jednym i dwóch rdzeniach dla systemów 32 i 64-bitowych. Z wykonanych obliczeń wynikają następujące wnioski:

- We wszystkich problemach obliczeniowych wydajność była większa od 93 %.
- Dla tak małej liczby procesów nie było różnicy pomiędzy zastosowaniem komunikacji kolektywnej i point-to-point.
- Czas obliczeń w systemie 32-bitowym był nawet do trzech razy dłuższy od tego w systemie 64-bitowym na tym samym komputerze z praktycznie identyczną konfiguracją.
- Komputer **K1** z procesorem nowszej generacji, mimo wolniejszego taktowania procesora, okazał się bardziej wydajny od komputera **K2** (nawet dwukrotnie).

Jak wynika z tego podrozdziału, korzyści z zastosowania przetwarzania równoległego w problemie obliczeń Monte Carlo są ogromne, gdyż pozwalają na udokładnienie wyników przy podobnym rzeczywistym czasie obliczeń oraz na zbadanie większej przestrzeni parametrów.

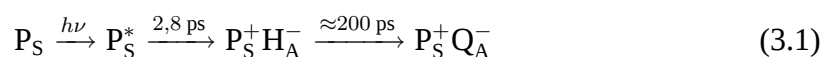
Rozdział 3

Efekty dynamiki wibracyjnej: anomalna zależność temperaturowa

Fotosynteza jest procesem, w którym zachodzi zamiana energii świetlnej w chemiczną energię swobodną. Na proces ten składa się transfer energii wzbudzenia (70 fs do pikosekund) w systemach antenowych, a następnie przeniesienie elektronu w centrach reakcji (pikosekundy do milisekund). Odpowiednie ustawienie chromoforów w macierzy białkowej wraz ze zoptymalizowanymi strukturalnie przenośnikami elektronów pozwalają na wysokowydajny rozdział ładunku i zapewniają odpowiedni kierunek biegu reakcji [39]. W tej pracy obiektem zainteresowania jest bakteryjne centrum fotosyntetyczne bakterii purpurowych *Rhodospseudomonas viridis* oraz *Rhodobacter sphaeroides*, w którym można rozdzielić systemy antenowe od centrów reakcji [39]. Struktury takiego centrum oraz schematy zachodzących reakcji można odnaleźć w literaturze [40].

Centrum reakcji składa się z czterech cząsteczek bakteriochlorofilu, dwóch cząsteczek bakteriofeofityny oraz dwóch cząsteczek chinonu. Pomędzy tymi ostatnimi znajduje się jon żelaza. Są one wbudowane w białko błonowe i działają jako fotoreceptory i przenośniki elektronów. Dwie cząsteczki bakteriochlorofilu są położone blisko siebie i tworzą dimer nazywany parą specjalną P_S . Para ta jest pułapką dla energii dostarczanej z systemów antenowych i pełni funkcję pierwotnego donora elektronu. Od pierwotnego donora odchodzą dwie gałęzie (A oraz B), jednakże tylko gałąź A jest aktywna. Elektron jest przekazywany przez bakteriochlorofil B_A i bakteriofeofitynę H_A do chinonu Q_A . Dalej elektron przekazywany jest do chinonu Q_B , a w następnych etapach zachodzi sprzężenie z transportem protonów.

Eksperymenty Martina i Bretona [41,42] dostarczyły informacji o czasach przeniesienia elektronu przy założeniu przejścia w jednym kroku z pierwotnego donora na feofitynę:



Osobnym tematem jest istnienie wirtualnego przenośnika w postaci bakteriochlorofilu B_A , którego uwzględnienie prowadzi do bardziej szczegółowej kinetyki (należy pamiętać, że

centra reakcji obu bakterii są dodatkowo trochę heterogeniczne w związku z występowaniem dwóch typów bakteriochlorofilu) [39]:



Niezależnie od przyjętego schematu, na uwagę zasługuje bardzo krótka skala czasowa pierwotnych procesów przeniesienia elektronu. Kolejne przejście od chinonu A do chinonu B trwa już znacznie dłużej (czasy rzędu mikrosekund).

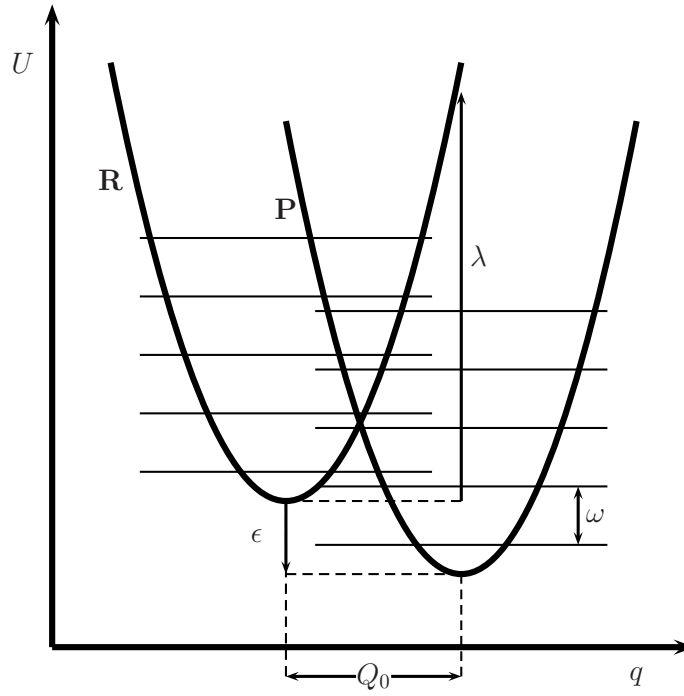
Należy zwrócić uwagę, że w przypadku pierwotnego procesu przeniesienia elektronu obserwowana jest doświadczalnie tzw. anomalna zależność temperaturowa, tzn. szybkość procesu wzrasta wraz z obniżaniem temperatury. Od eksperymentów Martina i Bretona minęło już ponad 20 lat, w czasie których próbowano tłumaczyć anomalną zależność na różne sposoby [43]. W tym rozdziale zostanie zaprezentowane nowe podejście do rozwiązania tej zagadki wychodzące poza teorię Marcusa, a wyniki otrzymane z symulacji dość dobrze zgadzają się z wynikami doświadczalnymi.

3.1 Teoretyczne sformułowanie problemu

Obiektem naszego zainteresowania jest proces przeniesienia elektronu pomiędzy pojedynczo wzbudzonym dimerem chlorofilu (pierwotny donor), a akceptorem, którym może być pośredniczący chlorofil lub feofityna (stany z przeniesieniem ładunku).

Zlokalizowane stany elektronowe reprezentowane są przez hiperpowierzchnie energii potencjalnej. Marcus zaproponował by problem wielowymiarowy zredukować do pojedynczej współrzędnej reakcji [44, 45]. W następnych latach rozszerzono model Marcusa m.in. wprowadzając dodatkową współrzędną reprezentującą szybkie zmiany otoczenia [40].

W standardowej teorii szybkości reakcji przeniesienia elektronu zakłada się, że relaksacja wibracyjna zachodzi dużo szybciej niż przeniesienie elektronu (równowaga wibracyjna musi ustalić się szybciej zanim zajdzie reakcja chemiczna) [46]. Dodatkowym założeniem jest istnienie statycznej bariery energetycznej, której przekroczenie jest konieczne do zajścia procesu. W rzeczywistości okazuje się, że relaksacja wibracyjna może zachodzić wolniej (a przynajmniej równie szybko) jak przeniesienie ładunku [10]. Ponadto rozdział ładunku może być ograniczony dynamiką otoczenia białkowego, a nie tylko statyczną barierą [47, 48]. Głównym założeniem proponowanego modelu jest włączenie poprawki związanej z dochodzeniem do równowagi wibracyjnej, tzn. założenie, że przeniesienie elektronu zachodzi z niezrelaksowanych wibracyjnie stanów, a przejścia pomiędzy tymi stanami odbywają się w przestrzeni energetycznej wybranego słabo tłumionego modu wibracyjnego otoczenia białkowego.



Rysunek 3.1. Redukcja problemu wielowymiarowego do pojedynczej współrzędnej reakcji q . Dwie parabole reprezentują energie wibracyjne stanów elektronowych substratu i produktu. Zaznaczono stany wibracyjne oraz charakterystyczne wielkości Q_0 , ϵ , λ , ω .

Ograniczymy się do rozważenia dwóch wzbudzonych zlokalizowanych stanów elektronowych $|R\rangle$ i $|P\rangle$ (substratu i produktu), sprzężonych ze stanami wibracyjnymi jednego oscylatora, w przybliżeniu harmonicznym numerowanymi przez liczby kwantowe m i n . W rezultacie otrzymujemy stany wibronowe (wibracyjno-elektronowe), odpowiednio $|Rm\rangle$ i $|Pn\rangle$.

Na rysunku 3.1 przedstawiono dwie parabole obrazujące adiabatyczne potencjały dla substratu i produktu reakcji w przybliżeniu harmonicznym wraz z istotnymi parametrami. Q_0 oznacza przesunięcie minimów parabol, ϵ oznacza różnicę potencjału pomiędzy minimami. Częstota ω ma sens energii wibracyjnej wyrażonej w jednostkach $\hbar \equiv 1$. Ważnym parametrem jest *energia reorganizacji* λ , która może być rozumiana jako energia jakiej należy dostarczyć substratom, aby przyjęły konfigurację równowagową produktów (lub odwrotnie) [45]. Jej związek z przesunięciem Q_0 jest następujący [10]:

$$\lambda = \frac{\omega}{2} Q_0^2. \quad (3.3)$$

Energie λ i ϵ ustalają wysokość bariery energetycznej dla reakcji:

$$\epsilon^\ddagger = \frac{(\epsilon + \lambda)^2}{4\lambda}. \quad (3.4)$$

W zależności od wzajemnego stosunku energii reorganizacji λ do odległości minimów energetycznych ϵ wyróżnia się kilka szczególnych przypadków teorii Marcusa [45]. Dla

$\epsilon = 0$ (minima obu parabol odpowiadają tej samej energii) bariera dla reakcji wynosi $\lambda/4$. Gdy $|\epsilon| < \lambda$, to bariera energetyczna zmniejsza się i reakcja przebiega szybciej. Szczególny przypadek stanowi $|\epsilon| = \lambda$ – wtedy reakcja przebiega najłatwiej. Istnieje także tzw. *obszar odwrotny* ($|\epsilon| > \lambda$), w którym szybkość reakcji maleje mimo dalszego wzrostu energii ϵ .

Oprócz wspomnianych wcześniej parametrów na uwagę zasługują jeszcze dwa: parametr związany z energią sprzężenia zlokalizowanych stanów elektronowych V oraz parametr tłumienia drgań κ [10]. Dla przeniesienia elektronu dużo wolniejszego od relaksacji wibracyjnej, $V \ll \kappa$, poprawne jest podejście nieadiabatyczne. Odwrotnie, jeśli przeniesienie elektronu jest o wiele szybsze od relaksacji wibracyjnej, $V \gg \kappa$, wtedy właściwe jest podejście adiabatyczne. W przypadku adiabatycznym prawdopodobieństwo przejścia jest bliskie jedności, w nieadiabatycznym jest ono małe, a elektron wiele razy zbliży się do obszaru przejściowego zanim zajdzie przeniesienie. W omawianym problemie oddziaływania V i κ mają niestety porównywalną siłę. Zdecydowano się na podejście nieadiabatyczne. Jeśli do problemu podejź się na gruncie rachunku zaburzeń, to zaburzeniem jest człon hamiltonianu związany z przeniesieniem elektronu, a prawdopodobieństwo przejść pomiędzy niezaburzonymi stanami R i P może być opisane przez złotą regułę Fermiego [10].

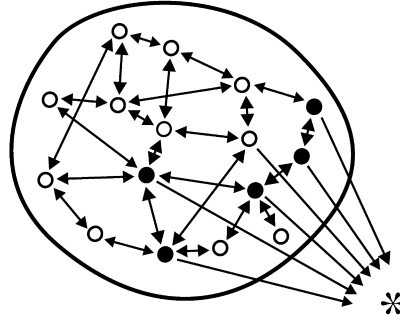
Zakładamy, że sprzężenie pomiędzy stanami elektronowymi a modami jest silne ($\lambda \gg \omega$). Jeśli weźmiemy pod uwagę tłumienie wibracyjne, to w przypadku dużego tłumienia wibracyjnego ($\kappa \gg \omega$) relaksacja zachodzi wzdłuż jednowymiarowego łańcucha podstanów określającego położenie oscylatora (ruch przetłumiony). W naszym przypadku istotniejszy jest przypadek małego tłumienia wibracyjnego ($\kappa \ll \omega$). Wtedy relaksacja zachodzi wzdłuż jednowymiarowej współrzędnej określającej energię układu [10].

Ogólny model reakcji jednocząsteczkowej z dynamiką stochastyczną pokazano wcześniej na rysunku 1.2. Warto przypomnieć, że stan substratu i produktu składa się z podstanów, a dopiero dojście do stanu przejściowego pozwala na zajście reakcji chemicznej. Dla reakcji jednocząsteczkowej $R \leftrightarrow P$ odwrotność stałej szybkości w każdą stronę, w szczególności od R do P, k_{PR}^{-1} , składa się z trzech członów [8, 10]:

$$k_{PR}^{-1} = (k_{PR}^{\text{eq}})^{-1} + \tau_R + K^{-1}\tau_P, \quad (3.5)$$

gdzie pierwszy człon reprezentuje czas potrzebny do zajścia reakcji chemicznej przy założeniu, że stan przejściowy $R^\ddagger$ znajduje się w lokalnej równowadze z innymi podstanami w R. W wyniku zajścia reakcji chemicznej równowaga zostaje zakłócona, stąd pojawiają się człony drugi i trzeci, które odpowiadają za dochodzenie do równowagowego obsadzenia stanu przejściowego odpowiednio od strony R i od strony P.

Z przybliżenia klasycznego stosowanego w wysokich temperaturach $T \equiv (k_B\beta)^{-1}$, czyli



Rysunek 3.2. Ogólny model nieodwracalnej jednocząsteczkowej reakcji chemicznej. Kółkami zaznaczono podstany (w tym przypadku wibracyjne) stanu substratu R, a gwiazdką stan *limbo*. Za pracę [10].

dla ($\beta\omega \ll 1$), wynika równowagowa stała szybkości Marcusa [10, 46]:

$$k_{PR}^{\text{eq}} = 2\pi|V|^2(4\pi\lambda k_B T)^{-1/2} \exp(-\beta\epsilon^\ddagger), \quad (3.6)$$

gdzie $\epsilon^\ddagger$ jest energią aktywacji (równanie (3.4)), a $\beta = (k_B T)^{-1}$.

Aby uprościć problem zakładamy, że reakcja $R \rightarrow P$ jest nieodwracalna i redukujemy problem umieszczając całkowicie absorbujący stan zaznaczony na rysunku 3.2 gwiazdką, który w teorii procesów stochastycznych nazywamy *limbo* („otchłań”; prawdopodobieństwo powrotu $w_{m,*} = 0$). W takim przypadku odwrotność stałej szybkości przyjmuje postać [10]:

$$k_{PR}^{-1} = (k_{PR}^{\text{eq}})^{-1} + \tau_R. \quad (3.7)$$

Zarówno składnik równowagowy jak i nierównowagowy w równaniu (3.7) mają identyczny czynnik wykładniczy, związany z prawdopodobieństwem obsadzenia stanu przejściowego [10]:

$$k_{PR}^{\text{eq}} = \nu e^{-\epsilon^\ddagger/k_B T}, \quad \tau_R^{-1} = \gamma e^{-\epsilon^\ddagger/k_B T}. \quad (3.8)$$

Różnicą są tu czynniki przedeksponencjalne, które mają inny sens fizyczny, gdyż ν oznacza uśrednione prawdopodobieństwo przejścia na jednostkę czasu ze stanu przejściowego substratu do stanu przejściowego produktu, natomiast czynnik γ związany jest z czasem dochodzenia do równowagi wibracyjnej.

Warto zatem zwrócić uwagę na trzy charakterystyczne czasy: k_{PR}^{-1} , ν^{-1} i γ^{-1} . Jeśli ν^{-1} i γ^{-1} są porównywalne (np. reakcje małych cząsteczek w fazie gazowej), to stałe szybkości reakcji mogą być z powodzeniem opisane przez teorię stanu przejściowego. Jeśli jednak γ^{-1} jest dużo większe od ν^{-1} , to człon τ_R w równaniu (3.7) przeważa i reakcja chemiczna jest kontrolowana przez dynamikę wewnątrzcząsteczkową.

Przypomnijmy, że rozważane przeniesienie elektronu składa się z dwóch głównych elementów – relaksacji do równowagowego obsadzenia stanu przejściowego, a następnie samej reakcji. Ogólne równanie fundamentalne dla ewolucji prawdopodobieństw dla

reakcji nieodwracalnej ma następującą postać [10]:

$$\dot{p}_m(t) = \sum_{m'} \left[w_{mm'} p_{m'}(t) - w_{m'm} p_m(t) \right] - v_m p_m(t), \quad (3.9)$$

gdzie w naszym przypadku indeksy m i m' numerują stany wibracyjne w R.

Fizycznie uzasadniony jest wybór sieci liniowej oraz dopuszczenie przejść tylko pomiędzy sąsiednimi stanami wibracyjnymi. Stąd równanie (3.9) przyjmie postać [10]:

$$\begin{aligned} \dot{p}_m(t) &= w_{m,m-1} p_{m-1}(t) + w_{m,m+1} p_{m+1}(t) \\ &- (w_{m+1,m} + w_{m-1,m}) p_m(t) - v_m p_m(t). \end{aligned} \quad (3.10)$$

Pominięto indeks R oznaczający, że chodzi o stan substratu, a także w porównaniu do równania (1.7) ogólne stany S_i zastąpiono konkretnymi oznaczeniami dotyczącymi sieci. Prawdopodobieństwa przejść na jednostkę czasu dane są wyrażeniami, które spełniają warunek równowagi szczegółowej (równanie (1.9)):

$$w_{m-1,m} = 2m\kappa [1 + e^{-\beta\omega}]^{-1}, \quad w_{m,m-1} = 2m\kappa [1 + e^{\beta\omega}]^{-1}. \quad (3.11)$$

Kurzyński zaproponował przejście do granicy ciągłej ($\beta\omega \ll 1$) i wprowadzenie gęstości rozkładu prawdopodobieństwa:

$$p(e, t) = \omega^{-1} p_m(t), \quad e \equiv \omega m. \quad (3.12)$$

Stąd otrzymał następujące równanie dyfuzji w energii [10]:

$$\begin{aligned} \frac{\partial}{\partial t} p(e, t) &= \frac{\partial}{\partial e} D(e) \left[\beta \frac{\partial U(e)}{\partial e} + \frac{\partial}{\partial e} \right] p(e, t) \\ &- v(e) p(e, t), \end{aligned} \quad (3.13)$$

gdzie parametr dyfuzji $D(e)$ oraz potencjał $U(e)$ zależą liniowo od energii e

$$D(e) = \kappa\omega e, \quad U(e) = e \quad (3.14)$$

dla $e > 0$, natomiast dla $e \leq 0$: $D(e) = 0$, $U(e) \rightarrow \infty$. W ogólności prawdopodobieństwo przejścia v_m może być rozmyte na kilka stanów, ale rozmycie jest niewielkie. Wprowadźmy dodatkowy parametr ζ wyrażający stosunek energii reorganizacji λ do częstości modu ω , który określa siłę sprzężenia elektron-wibracja. Przy założeniu silnego sprzężenia:

$$\zeta = \frac{\lambda}{\omega} \gg 1, \quad (3.15)$$

prawdopodobieństwo zajścia reakcji chemicznej $v(e)$ może być przybliżone rozkładem Gaussa [43]

$$v(e) = 2|V|^2 (4\lambda\beta)^{-1/2} \zeta e^{-\zeta^2(e-\epsilon^\ddagger)^2}, \quad (3.16)$$

co w granicy $\zeta \rightarrow \infty$ przechodzi w deltę Diraca.

Jeśli relaksacja wibracyjna zachodzi szybciej niż sama reakcja przeniesienia elektronu, to układ osiąga najpierw równowagowy rozkład prawdopodobieństwa, co determinuje równowagową stałą szybkości $k_{\text{PR}}^{\text{eq}}$ (równanie (3.6)). Człon nierównowagowy τ_{R} (równanie 3.7) mający sens czasu dochodzenia do równowagowego obsadzenia stanu przejściowego może być przybliżony (przypadek jednowymiarowej dyfuzji w przestrzeni energetycznej) jako średni czas pierwszego przejścia ze stanu podstawowego 0 do stanu przejściowego $\epsilon^\ddagger$ (bramy) [49]:

$$\tau(0 \rightarrow \epsilon^\ddagger) = \frac{1}{\kappa\omega} \int_0^{\epsilon^\ddagger} dx e^{\beta U(x)} D(x)^{-1} \int_0^x dy e^{-\beta U(y)}. \quad (3.17)$$

Korzystając z postaci potencjału $U(e)$ i parametru dyfuzji $D(e)$ (równanie (3.14)) otrzymujemy [10]:

$$\begin{aligned} \tau(0 \rightarrow \epsilon^\ddagger) &= \frac{1}{\kappa\omega\beta} \int_0^{\beta\epsilon^\ddagger} d(\beta e) \frac{e^{\beta e} - 1}{\beta e} \\ &= \frac{1}{\kappa\omega\beta} \left[\text{Ei}(\beta\epsilon^\ddagger) - \ln(\beta\epsilon^\ddagger) - \text{Eu} \right], \end{aligned} \quad (3.18)$$

gdzie Ei oznacza eksponens całkowity, a Eu jest stałą Eulera [50]. Dla stanu początkowego znajdującego się poniżej rozsądnej bariery energetycznej jego wybór nie ma większego wpływu na średni czas pierwszego przejścia i może być przybliżony przez ewolucję od stanu podstawowego.

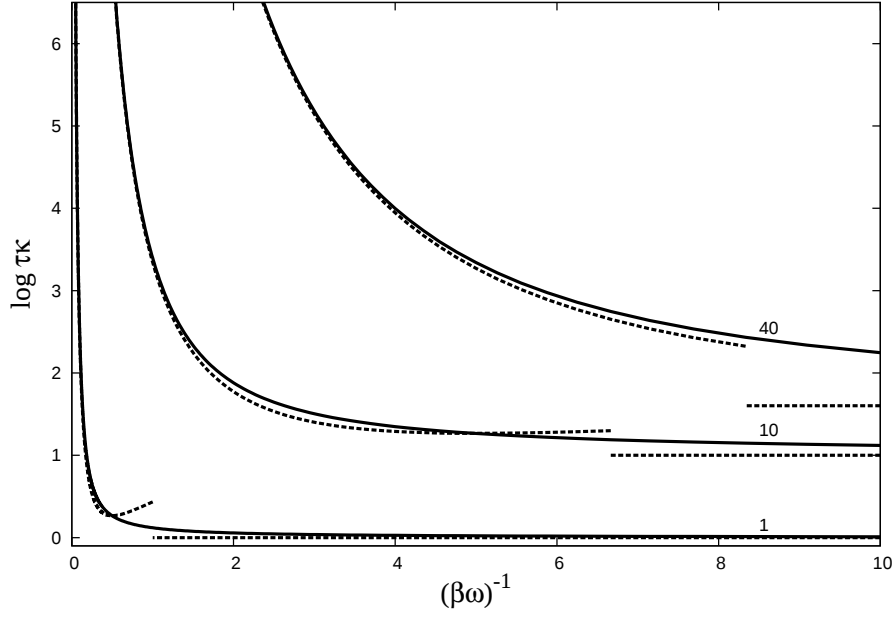
Kurzyński zaproponował przybliżenia wyrażenia (3.18) ze względu na wartość $\beta\epsilon^\ddagger$ [10]. Dla $\beta\epsilon^\ddagger \gg 1$ przybliżono eksponens całkowity wyrażeniem

$$\tau(0 \rightarrow \epsilon^\ddagger) \approx \frac{(k_{\text{B}}T)^2}{\kappa\omega\epsilon^\ddagger} \exp(\beta\epsilon^\ddagger), \quad (3.19)$$

natomiast dla $\beta\epsilon^\ddagger \ll 1$ otrzymano

$$\tau(0 \rightarrow \epsilon^\ddagger) \approx \frac{\epsilon^\ddagger}{\kappa\omega}. \quad (3.20)$$

W celu sprawdzenia przydatności przybliżeń wykonałem wykres zależności średniego czasu pierwszego przejścia od temperatury dla różnych wysokości barier energetycznych ($\epsilon^\ddagger\omega^{-1} = 1, 10, 40$). Na rysunku 3.3 linią ciągłą wyrysowano krzywą daną równaniem (3.18), a liniami przerywanymi przybliżone krzywe dane równaniami (3.19) i (3.20) w zakresie ich potencjalnej stosowności. Oczywistym było, że żadne z przybliżeń nie będzie działało w całym zakresie temperatur. Jednakże z przedstawionego wykresu wynika, że zaproponowane przybliżenia analityczne (3.19) i (3.20) są dość kiepskie i mało użyteczne, a wynikający z zależności (3.19) wzrost szybkości procesu w pewnym zakresie niższych temperatur okazał się przypadkowy. Dla przypadku niskich barier energetycznych $\epsilon^\ddagger\omega^{-1} \simeq 1$ przybliżenie eksponencjalne (równanie (3.19)) wykazuje niepoprawną



Rysunek 3.3. Zakres stosowalności przybliżenia (3.19) dla $\beta\epsilon^\ddagger \gg 1$ oraz przybliżenia (3.20) dla $\beta\epsilon^\ddagger \ll 1$ (linie przerywane) równania (3.18) dla wybranych wysokości bariery energetycznej $\epsilon^\ddagger\omega^{-1}$. Za pracę [B].

zależność temperaturową już dla $(\beta\omega)^{-1} > 0.5$. Natomiast przybliżenie (3.20) źle działa dla wyższych barier, czego można było się spodziewać, gdyż rozwinięcia były tworzone w oparciu o $\beta\epsilon^\ddagger$, zatem występuje tu połączenie temperatury z wysokością bariery.

Podstawowym celem symulacji dynamiki wibracyjnej było wyjaśnienie anomalnej zależności stałej szybkości przeniesienia elektronu od temperatury, tzn. dlaczego ze spadkiem temperatury szybkość reakcji wzrasta. Weryfikacja zastosowanego przez Kurzyńskiego przybliżenia ciągłego (3.18) pokazała, że nie nadaje się ono do opisu anomalnej zależności temperaturowej. Na drodze do osiągnięcia tego celu stało kilka kroków weryfikujących teorię i model, a mianowicie zbadanie zależności średnich czasów pierwszego przejścia do bramy od temperatury, wysokości bariery energetycznej i wyboru stanu początkowego. Wyniki tych „wstępnych” eksperymentów numerycznych również przedstawiono, gdyż prezentują drogę do głównego celu dostarczając wielu niezwykle ważnych wskazówek i pokazując gdzie przybliżenie ciągłe zaproponowane przez Kurzyńskiego ma sens.

3.2 Szczegóły symulacji i wyniki

Zespół statystyczny układów (wędrówców) umieszczono na jednowymiarowym łańcuchu węzłów (sieć liniowa), które numerowano liczbami całkowitymi od $m = 0$ do $m = L - 1$

Tabela 3.1. Parametry wejściowe symulacji numerycznych dynamiki wibracyjnej.

Parametr	Znaczenie
N_w	Liczba układów (wędrówców), których ewolucja była badana, dobierana tak by błąd wyznaczenia wartości średniej nie przekraczał 5%.
$m_o = -1$	Położenie bariery odbijającej.
$e_0\omega^{-1}$	Położenie stanu początkowego.
$e^\dagger\omega^{-1}$	Położenie stanu końcowego.
$f = \exp(-\beta\omega)$	Parametr związany z temperaturą (tabela 3.2).
L	Długość sieci, czyli liczba stanów dostępna w symulacji.
v	Prawdopodobieństwo wpadnięcia w bramkę (zajścia procesu); wartość 1 oznacza całkowitą absorpcję.

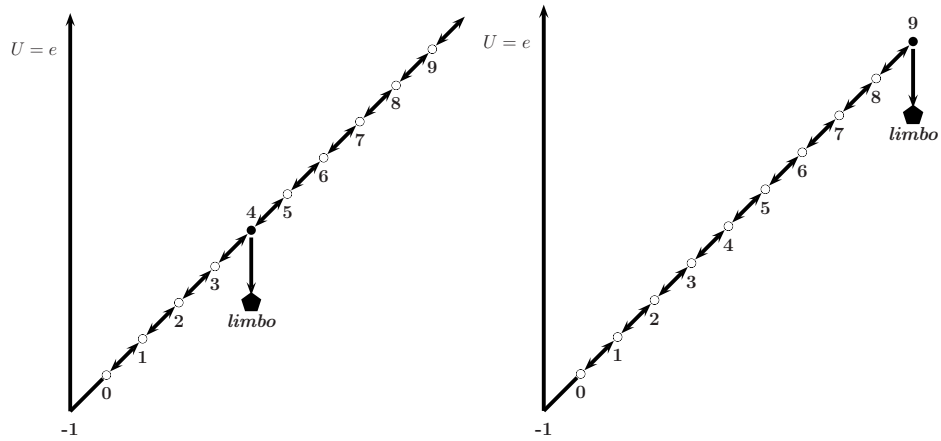
(L oznacza liczbę węzłów). Przejścia dozwolone były tylko pomiędzy sąsiednimi stanami modelującymi stany wibracyjne. Obliczaną wielkością był średni czas pierwszego przejścia, a ewolucja zachodziła w zadanym potencjale liniowym (dyfuzja w energii). Na sieci znajduje się stan z warunkiem odbijającym (stan podstawowy) oraz absorbującym (stan przejściowy, przez który zachodzi reakcja nieodwracalna). Odległości pomiędzy węzłami (stanami) są stałe i wynoszą ω . Parametry wejściowe symulacji przedstawiono w tabeli 3.1. W programie umieszczono kod sprawdzający czy dany zestaw parametrów jest dozwolony. Prawdopodobieństwa przejść zależą od parametrów i zależnie od stanu mogą przyjmować różne wartości.

Używana sieć liniowa jest szczególnym przypadkiem ogólnego modelu przedstawionego na rysunku 3.2. Na sieci znajduje się jeden stan przejściowy (zgodnie z założeniem silnego sprzężenia (3.15)). W zależności od prawdopodobieństwa v można wyróżnić dwa przypadki sieci liniowej – przedstawiono je na rysunku 3.4. Jeśli $v = 1$ to bramka jest stanem całkowicie absorbującym czyli właśnie *limbo*.

Bardzo ważne jest określenie stanu początkowego i weryfikacja czy jego wybór ma znaczący wpływ na wyniki symulacji. Jeśli stan końcowy jest całkowicie absorbujący, to nie ma większych kłopotów z ewolucją, jednakże jeśli stan końcowy nie jest całkowicie absorbujący, wtedy wędrowiec może uciec poza skończoną sieć, a stąd mogą pojawić się problemy z powrotem w okolice stanu przejściowego oraz z definicją skończonej sieci. Ważny jest też wybór rodzaju warunku odbijającego – w tych symulacjach wybrano cze-

Tabela 3.2. Związek pomiędzy wybranymi wartościami f a $\beta\omega$ i $(\beta\omega)^{-1}$.

f	$\beta\omega$	$(\beta\omega)^{-1}$	f	$\beta\omega$	$(\beta\omega)^{-1}$
0,01	4,61	0,22	0,10	2,30	0,43
0,20	1,61	0,62	0,30	1,20	0,83
0,40	0,92	1,09	0,50	0,69	1,44
0,60	0,51	1,96	0,70	0,36	2,80
0,80	0,22	4,48	0,90	0,11	9,49
0,99	0,01	99,50			



Rysunek 3.4. Schemat sieci liniowej (zaznaczono 10 stanów), na której zachodzi przypadkowe błędzenie w energii e . Po lewej stronie sieć ze stanem–bramką, obok której układ może przejść nie wpadając w *limbo*. Po prawej stronie sieć z całkowicie absorbującym stanem końcowym, którego osiągnięcie powoduje wpadnięcie w *limbo*.

kanie przez jeden krok.

Istotnym elementem programu jest obliczenie zależnych od położenia prawdopodobieństw przejść. Wyróżniono trzy takie prawdopodobieństwa dla każdego stanu m : krok w lewo $p_{-}(m)$, w prawo $p_{+}(m)$ oraz postój $p_0(m)$ przez jedną jednostkę czasu (równą jednemu krokowi ewolucji). Wyrażenia na prawdopodobieństwa przejść pomiędzy sąsiednimi stanami wynikające z równań (3.11) przyjmują następującą postać, odpowiednio dla ruchu w lewo, w prawo i postoju:

$$\begin{aligned} p_{-}(m) &= \frac{m}{(f+1)L}, \\ p_{+}(m) &= \frac{(m+1)f}{(f+1)L}, \\ p_0(m) &= 1 - p_{-}(m) - p_{+}(m). \end{aligned} \quad (3.21)$$

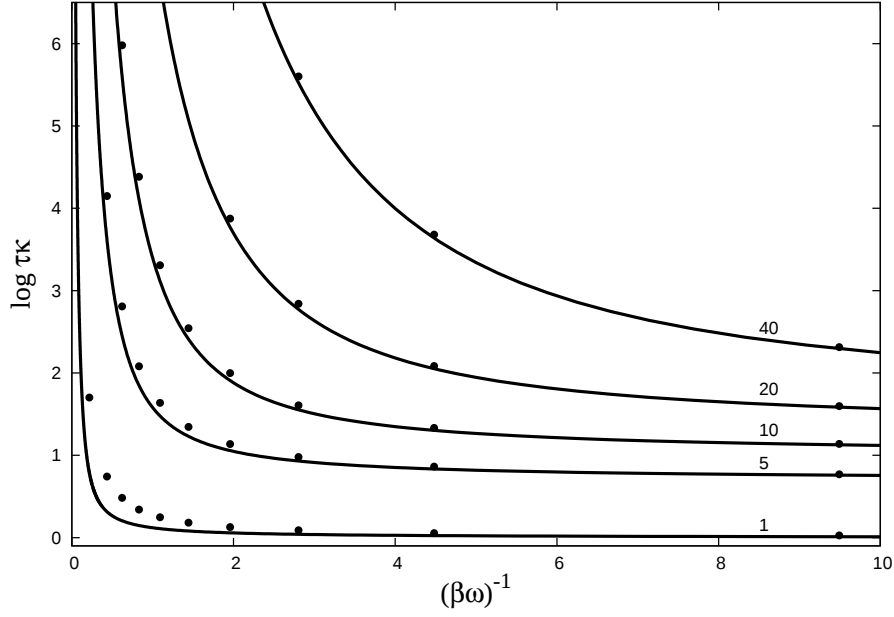
Ponadto jeśli bramka nie jest całkowicie absorbująca, to dla stanu będącego bramką prawdopodobieństwa dane równaniami (3.21) muszą zostać pomnożone przez wyrażenie $(1 - v)$, gdzie v jest prawdopodobieństwem wejścia w bramkę. Możliwość czekania została wprowadzona ze względu na istniejącą na sieci bramkę, tak aby w każdym stanie całkowite prawdopodobieństwo sumowało się do jedności.

Ze względu na postać wyrażeń danych równaniami (3.21) (pojawia się długość sieci), średnie czasy pierwszego przejścia także zależą od długości sieci L . Sposobem na uniezależnienie wyników od dowolnie wybieranej długości sieci jest przeskalowanie każdego wyniku przez czynnik $(2L)^{-1}$. Średni czas pierwszego przejścia liczony w ten sposób odpowiada bezwymiarowemu czasowi w jednostkach κ^{-1} (równania (3.13) i (3.17)).

3.2.1 Zależność średniego czasu pierwszego przejścia od temperatury i wysokości bariery energetycznej

Celem pierwszej symulacji było zbadanie wpływu temperatury oraz wysokości bariery energetycznej ($\epsilon^{\dagger}\omega^{-1} = 1, 5, 10, 20, 40$) na średni czas pierwszego przejścia od stanu podstawowego do stanu przejściowego (tożsamy z τ_R w równaniu (3.7)). Wyniki symulacji przedstawiono na rysunku 3.5. Krzywe ciągłe dane są równaniem (3.18), a kropki oznaczają wyniki symulacji.

Patrząc na wykres od razu rzuca się w oczy fakt, że przyczynek pochodzący od relaksacji wibracyjnej jest praktycznie niezależny od temperatury dla rozsądnych barier energetycznych poniżej $\epsilon^{\dagger}\omega^{-1} = 10$ i wzrasta szybko eksponencjalnie dopiero bardzo blisko temperatury 0 K. Ponadto wyniki symulacji lepiej odpowiadają krzywom teoretycznym w wyższych temperaturach (większe $(\beta\omega)^{-1}$), co odpowiada przechodzeniu do granicy ciągłej (klasycznej). Ostatnia własność jest efektem założeń teoretycznych przy wyprowadzaniu równania (3.13). W ogólności wyniki symulacji komputerowych będą lepiej



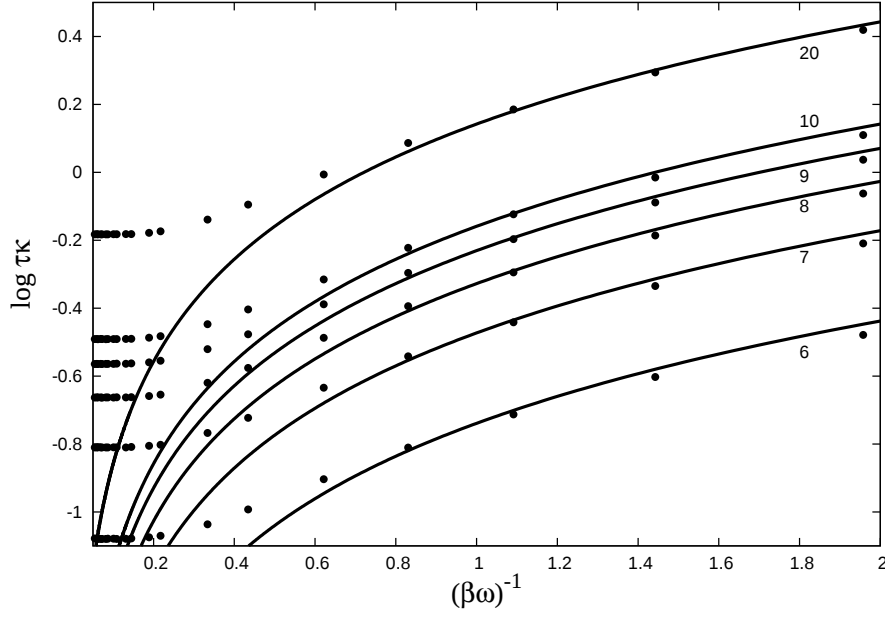
Rysunek 3.5. Zależność logarytmu średniego czasu pierwszego przejścia $\tau(0 \rightarrow \epsilon^\ddagger)$ (w jednostkach κ^{-1}) od temperatury (w jednostkach ω) dla różnych wysokości bariery energetycznej $\epsilon^\ddagger\omega^{-1}$. Kropki oznaczają wyniki symulacji, a krzywe oznaczone liniami ciągłymi dane są równaniem (3.18). Za pracą [B].

zgadzały się z wynikami teoretycznymi dla wyższych temperatur i większych wysokości bariery energetycznej. W niższych temperaturach ($(\beta\omega)^{-1} < 2$) i dla niższych barier energetycznych ($\epsilon^\ddagger\omega^{-1} < 5$) efekty kwantowe stają się bardzo istotne i równanie (3.18) nie nadaje się do stosowania.

3.2.2 Zależność średniego czasu pierwszego przejścia od temperatury i wyboru stanu początkowego

Ani postać stałej szybkości Marcusa $k_{\text{PR}}^{\text{eq}}$ (równanie (3.6)), ani postać członu odpowiedzialnego za dochodzenie do równowagi wibracyjnej τ_{R} (równanie (3.18)) nie tłumaczą anomalnej zależności stałej szybkości reakcji przeniesienia elektronu od temperatury. Ponadto wynik symulacji dla stanu początkowego znajdującego się poniżej stanu przejściowego (rysunek 3.5) także wykazuje zależność inną od tej, której szukamy. W przypadku wolnej relaksacji okazuje się, że istotny jest etap początkowy reakcji, nie opisany jeszcze przez równanie (3.7). Efekt ten staje się tym wyraźniejszy im bardziej stan początkowy odległy jest od stanu podstawowego.

Rozważmy różne drogi dyfuzji. Czas dyfuzji z dowolnego punktu e_0 w górę poten-



Rysunek 3.6. Zależność logarytmu średniego czasu pierwszego przejścia $\tau(e_0 \rightarrow \epsilon^\ddagger)$ (w jednostkach κ^{-1}) od temperatury (w jednostkach ω) dla różnych początkowych poziomów wibracyjnych $e_0\omega^{-1}$ znajdujących się ponad stałą barierę energetyczną $\epsilon^\ddagger\omega^{-1} = 5$. Kropki oznaczają wyniki symulacji, a krzywe oznaczone liniami ciągłymi dane są równaniem (3.23). Za pracą [B].

cjału ($e_0 < \epsilon^\ddagger$) dany jest wyrażeniem

$$\tau(e_0 \rightarrow \epsilon^\ddagger) = \tau(0 \rightarrow \epsilon^\ddagger) - \tau(0 \rightarrow e_0), \quad (3.22)$$

natomiast dla dyfuzji w dół potencjału ($e_0 > \epsilon^\ddagger$) otrzymujemy [10]

$$\tau(e_0 \rightarrow \epsilon^\ddagger) = \frac{k_B T}{\kappa \omega} \ln \frac{e_0}{\epsilon^\ddagger}. \quad (3.23)$$

Wobec tego w kolejnej symulacji zbadano wpływ wyboru stanu początkowego i w ogólności zależność średniego czasu pierwszego przejścia od temperatury jeśli stan początkowy jest zlokalizowany powyżej stanu przejściowego ($e_0\omega^{-1} > \epsilon^\ddagger\omega^{-1}$).

Wyniki symulacji przedstawiono na rysunku 3.6. Kropkami zaznaczono wyniki pochodzące z symulacji, natomiast krzywa ciągła dana jest równaniem (3.23). Wybrano różne stany początkowe ($e_0\omega^{-1} = 6, 7, 8, 9, 10, 20$) powyżej stanu przejściowego ustalonego jako $\epsilon^\ddagger\omega^{-1} = 5$.

Podkreślmy fakt, że na rysunku 3.6 skala temperatur wyrażonych w $(\beta\omega)^{-1}$ zawiera się pomiędzy 0 i 2, zatem gdyby porównywać ten wykres z rysunkiem 3.5, to prezentuje on zależność w wąskim przedziale ale najbardziej dla nas interesującym. Jak wspomniano wcześniej, dla stanu początkowego znajdującego się poniżej rozsądnej bariery energetycznej jego wybór nie ma większego wpływu na średni czas pierwszego przejścia. Jednakże wybór stanu początkowego powyżej stanu przejściowego ma już wpływ

(czas dochodzenia do równowagi wibracyjnej jest wyraźnie różny), co w modelu można tłumaczyć większą przestrzenią dostępną do przypadkowego błędzenia. Najważniejszym faktem widocznym na wykresie jest jeszcze to, że średni czas pierwszego przejścia maleje z obniżaniem temperatury (odwrotnie niż dla stanów początkowych poniżej stanu przejściowego), czyli szybkość relaksacji wibracyjnej zwiększa się. Właśnie takie zachowanie dla procesu przeniesienia elektronu było obserwowane doświadczalnie [41]. Spadek średniego czasu pierwszego przejścia nie jest tak szybki jak w przybliżeniu danym równaniem (3.23). Dążenie średniego czasu pierwszego przejścia do stałej wartości dla $T = 0$ jest zgodne z równaniem (3.11), gdyż będą zachodziły tylko przejścia w dół potencjału.

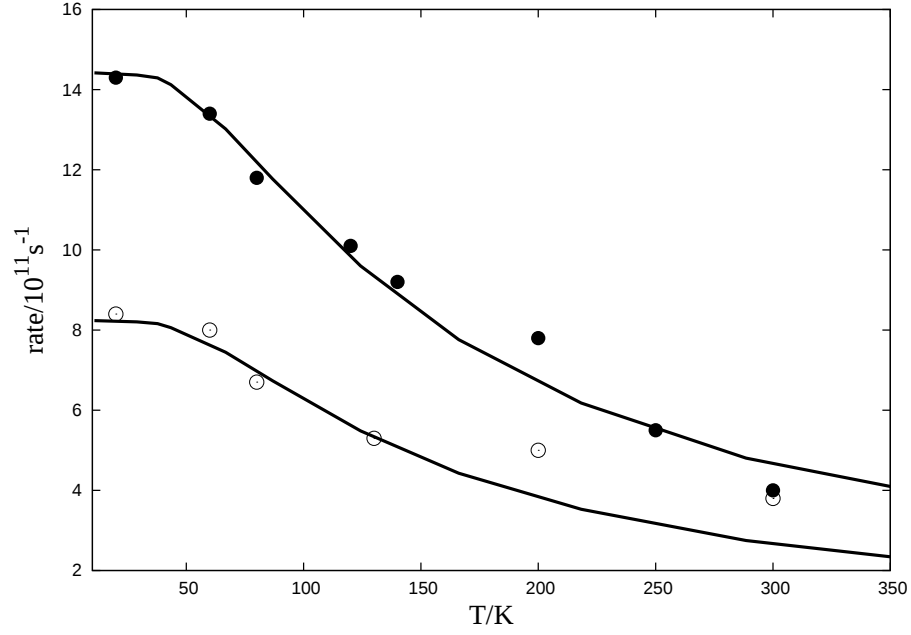
3.2.3 Wyjaśnienie anomalnej zależności temperaturowej czasu przeniesienia elektronu w fotosyntetycznym centrum reakcji

Poprzednie symulacje stanowiły wstęp do zadania jakim było zastosowanie modelu do wyjaśnienia anomalnej zależności temperaturowej zaobserwowanej przez Fleminga, Martina i Bretona [41]. Zajmowali się oni centrami fotosyntetycznymi bakterii *Rhodopseudomonas viridis* oraz *Rhodobacter sphaeroides* i eksperymentalnie wyznaczyli zależność stałej szybkości przeniesienia elektronu od temperatury. Autorzy wprawdzie dokonali próby dopasowania pewnych krzywych teoretycznych do wyników eksperymentalnych lecz szczególnie w przypadku *Rhodopseudomonas viridis* próby te, jak sami przyznali, zawiodły. Anomalna zależność temperaturowa skłania do zwrócenia uwagi na problem wyboru wibracyjnego stanu początkowego. Aby otrzymać zależność obserwowaną eksperymentalnie, należy wybrać stan początkowy powyżej stanu przejściowego.

Dwa wspomniane bakteryjne centra reakcji różnią się nieznacznie, jednakże prezentują znaczną różnicę w szybkości przeniesienia elektronu w niskich temperaturach [41, 43]. Ważne jest stwierdzenie, które parametry rzeczywiście wpływają na te różnice. Zakładamy, że centra reakcji nie mogą się znacznie różnić w częstości ω , parametrze tłumienia κ i energii stanu początkowego e_0 wyznaczonego przez kwant wzbudzenia. Zatem fizycznie możliwe zmiany mogą dotyczyć energii reorganizacji λ , a zatem energii stanu przejściowego $\epsilon^\ddagger$.

Założono, że w obu przypadkach stan początkowy m jest identyczny (przyjęto, że wzbudzenie jest niekoherentne, tzn. możliwe jest wzbudzenie pojedynczego poziomu wibracyjnego), lecz centra te różnią się położeniem stanu przejściowego n o jeden kwant ω . Odpowiednie stałe szybkości można ogólnie zapisać dla *Rhodopseudomonas viridis* (krótszy czas) jako

$$k_{vir}^{-1} = \tau_{vir} = \tau(e_0 \rightarrow \epsilon_{vir}^\ddagger) = \tau(m \rightarrow n), \quad (3.24)$$



Rysunek 3.7. Zależność temperaturowa szybkości pierwotnego przeniesienia elektronu w bakteryjnych fotosyntetycznych centrach reakcji. Dopasowano wyniki symulacji komputerowych (linie ciągłe; za pracą [B]) do danych doświadczalnych (otwarte kółka *Rhodobacter sphaeroides*, zamknięte *Rhodospseudomonas viridis*; za pracą [41]).

a dla *Rhodobacter sphaeroides* (dłuższy czas) jako

$$k_{sph}^{-1} = \tau_{sph} = \tau(e_0 \rightarrow \epsilon_{sph}^{\dagger}) = \tau(m \rightarrow n - 1). \quad (3.25)$$

Jak już wspomniano wcześniej, w temperaturze 0 K będą zachodzić tylko przejścia pomiędzy najbliższymi sąsiadami w dół, stąd średnie czasy pierwszego przejścia z równań (3.24) i (3.25) będą sumą kolejnych średnich czasów pierwszego przejścia dla poszczególnych pojedynczych kroków. W temperaturze 0 K z równania (3.11) otrzymujemy

$$\tau(n \rightarrow n - 1) = \frac{1}{2n\kappa}, \quad (3.26)$$

stąd

$$k(0)_{sph}^{-1} = k(0)_{vir}^{-1} + \frac{1}{2n\kappa}. \quad (3.27)$$

Porównując wartości $k(0)_{vir}^{-1}$ i $k(0)_{sph}^{-1}$ w temperaturze 0 K dopasowano wartości $m = 5$, $n = 3$ dla *Rhodospseudomonas viridis* oraz $m = 5$, $n - 1 = 2$ dla *Rhodobacter sphaeroides*. Stąd

$$\tau_{vir} = \tau(5 \rightarrow 4) + \tau(4 \rightarrow 3) \quad (3.28)$$

oraz

$$\tau_{sph} = \tau(5 \rightarrow 4) + \tau(4 \rightarrow 3) + \tau(3 \rightarrow 2). \quad (3.29)$$

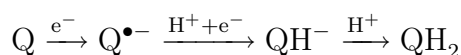
Korzystając z eksperymentalnych wartości $k(0)_{vir}^{-1}$ i $k(0)_{sph}^{-1}$ oraz równania (3.26) otrzymujemy przybliżoną wartość parametru tłumienia $\kappa^{-1} = 3,25$ ps. Założono, że częstość ω wynosi 200 K, stąd okres drgań wynosi 0,25 ps. Porównując ten okres drgań z wartością κ^{-1} widać, że rzeczywiście mamy do czynienia ze słabym tłumieniem wibracyjnym (okres tłumienia ponad 10 razy dłuższy od okresu drgań).

Rezultat dopasowania wyników symulacji komputerowych i wyników eksperymentalnych przedstawiono na rysunku 3.7. Mimo, że dopasowanie szczególnie w niskich temperaturach jest dość dobre, to trzeba pamiętać, że model ten jest grubym przybliżeniem ze względu na szereg wprowadzonych założeń, takich jak choćby nieodwracalność reakcji czy też założenie, że czas przejścia ze stanu przejściowego do *limbo* jest zanedbywalnie krótki [B]. Jednakże model ten może być podstawą do dalszych badań. Warto też zauważyć, że kompletny opis kolejnych etapów przeniesienia elektronu w fotosyntetycznym centrum reakcji wymaga ze względu na inne skale czasowe uwzględnienia dodatkowych czynników, np. wpływu dynamiki przejść konformacyjnych, a także innych modów drgań.

Rozdział 4

Efekty dynamiki przejść konformacyjnych: przetwarzanie energii swobodnej

W poprzednim rozdziale dotyczącym dynamiki wibracyjnej obiektem zainteresowania były pierwotne procesy przeniesienia elektronu zachodzące w bakteryjnych fotosyntetycznych centrach reakcji. Jednym z kolejnych etapów fotosyntezy jest redukcja chinonu Q do hydrochinonu (chinolu) QH_2 , której poszczególne fazy

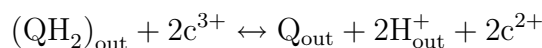


związane są z przejściami konformacyjnymi [51]. Podobna reakcja przeniesienia elektronu, ale w odwrotną stronę, zachodzi w kompleksie cytochromów bc_1 , który jest białkiem błonowym katalizującym utlenianie hydrochinonu (chinolu) i redukcję cytochromu c w bakteryjnych łańcuchach oddechowych i fotosyntetycznych, a także w mitochondrialnych łańcuchach oddechowych [52, 53].

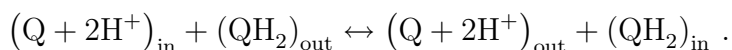
W kompleksie cytochromów bc_1 , chinol jest utleniany, a chinon redukowany w dwóch odrębnych miejscach oznaczanych jako Q_o oraz Q_i . Utlenianie cząsteczki chinolu dostarcza dwóch elektronów, które przemieszczają się dwiema drogami. Uważa się, że jeden elektron jest przekazywany do centrum żelazowo-siarkowego i następnie na cytochrom c_1 . Powstaje wtedy semichinon $\text{Q}^{\bullet-}$, a następnie kolejny elektron jest przekazywany do hemu b_L (b_{566}) i dalej przez hem b_H (b_{562}) do Q_i . Osobną kwestią jest problem stabilności semichinonu i mechanizmy zabezpieczające przed przeniesieniem dwóch elektronów korzystniejszą energetycznie drogą na cytochrom c_1 [53]. Struktury takiego kompleksu oraz schematy zachodzących reakcji można odnaleźć w literaturze [51]. Dynamika procesu jest bardzo złożona ale powolna, stąd też główną rolę odgrywa dynamika przejść konformacyjnych, a przyjmowanie określonej konformacji przez białko wpływa na proces przetwarzania energii swobodnej.

Najważniejszym faktem dla tworzonego modelu dynamiki przejść konformacyjnych

jest sprzężenie procesu dostarczającego energię swobodną (transport elektronów)



i odbierającego energię swobodną (transport protonów w poprzek błony)



Indeksy in i out oznaczają cząsteczkę przyłączoną w centrum Q_i i Q_o , odpowiednio po wewnętrznej i zewnętrznej stronie błony. Transport cząsteczki chinolu z pozycji $(\text{QH}_2)_{\text{in}}$ do $(\text{QH}_2)_{\text{out}}$ połączony z transportem chinonu z pozycji Q_{out} do Q_{in} oznacza transport dwóch atomów wodoru w poprzek błony. Jest to równoważne z transportem dwóch protonów przy założeniu, że dwa elektrony są przenoszone w kierunku przeciwnym [52, 53].

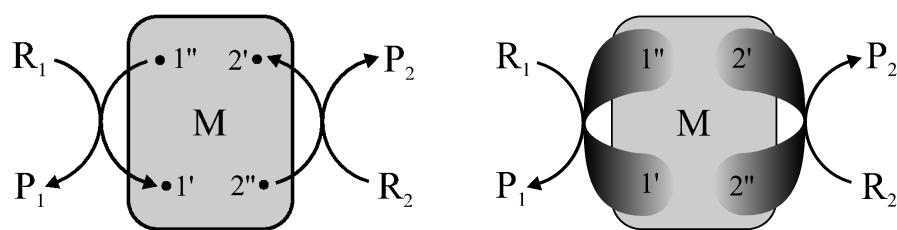
Powszechnie przyjmuje się, że sprzężenie transportu elektronów z transportem protonów zachodzi w stosunku 1:1, tzn. w centrum Q_o musi nastąpić bifurkacja i dokładnie jeden elektron przemieszcza się na cytochrom c_1 i jeden na centrum Q_i [52, 53]. Jest to tylko hipoteza; Kurzyński przedstawił argumenty, że wcale tak być nie musi [54]. Znane są inne przykłady zmiany współczynników stechiometrycznych w zależności od warunków otoczenia. Grupa Yanagidy zaobserwowała, że głowa miozynowa może wykonywać wiele kroków na jedną hydrolizowaną cząsteczkę ATP [55]. Innym przykładem mogą być wyniki eksperymentalne Kettnera i współpracowników, którzy obserwowali różną liczbę przepompowywanych protonów na jedną hydrolizowaną cząsteczkę ATP w V-ATPazie [56].

Głównym celem rozważań przedstawionych w tym rozdziale jest próba odpowiedzi na pytanie, czy możliwe jest przetwarzanie energii swobodnej ze zmieniającym się stopniem sprzężenia pomiędzy procesem dostarczającym i odbierającym energię swobodną i przy jakich warunkach stopień sprzężenia może być większy od 1.

4.1 Teoretyczne sformułowanie problemu

Maszyna jest dowolnym układem fizycznym, który umożliwia innym układom wykonanie pracy jednemu na drugim. W warunkach izotermicznych wykonanie pracy jest tożsame z przetwarzaniem energii swobodnej, stąd takie maszyny nazywane są przetwornikami energii swobodnej (ang. *free energy transducers*) [10].

Wspomniany białkowy kompleks enzymatyczny jest przykładem chemochemicznej maszyny molekularnej. Ponieważ zachodzące reakcje są wysoce złożone dla uproszczenia założymy, że rozważana tu maszyna katalizuje i sprzęga dwie jednocząsteczkowe reakcje: dostarczającą energię swobodną $\text{R}_1 \leftrightarrow \text{P}_1$ i odbierającą energię swobodną $\text{R}_2 \leftrightarrow \text{P}_2$.



Rysunek 4.1. Schemat maszyny chemochemicznej sprzęgającej dwa procesy: dostarczający i odbierający energię swobodną. Po lewej stronie we wnętrzu szarej skrzynki wyróżnione są pojedyncze bramy, po prawej stronie przedstawiono rozszerzenie na wiele bram. Za pracę [C].

Z osobna każda reakcja zachodzi w kierunku określonym przez drugą zasadę termodynamiki, tzn. w takim, że ilość dysypowanej energii chemicznej jest dodatnia. Jednakże jeśli obie reakcje zachodzą w tym samym cyklu (używają tego samego enzymu), to mogą zachodzić w tą samą stronę, tzn. jedna z reakcji zachodząca w zgodzie z drugą zasadą termodynamiki może „zmusić” drugą reakcję do zachodzenia przeciw drugiej zasadzie termodynamiki, czyli zmieniać jej kierunek na przeciwny do ustalonego przez stacjonarne wartości stężeń [14]. W ten sposób pierwsza reakcja przekazuje część swojej energii swobodnej odzyskanej z dysypacji wykonując pracę na rzecz drugiej reakcji.

Standardowa kinetyka chemiczna zakłada, że przed każdym krokiem kinetycznym szybka dynamika wewnątrzcząsteczkowa powoduje ustalenie się stanu równowagi częściowej. Jednakże wewnętrzna stochastyczna dynamika białka tworzącego maszynę molekularną jest bardzo skomplikowana i co więcej powolna, a sprzężone reakcje enzymatyczne są przez nią kontrolowane, co prowadzi do konieczności zastąpienia rozkładu równowagowego stanów konformacyjnych rozkładem stacjonarnym [10].

Elektron może znajdować się na donorze lub akceptorze przyłączonym do enzymu tworzącego kompleks, którego stany zarówno elektronowe jak i konformacyjne opisuje wnętrze szarej skrzynki z rysunku 4.1. W ogólności kompleks może znajdować się w ogromnej liczbie stanów konformacyjnych. Jak już wspomniano, między nimi ustala się stan stacjonarny, a nie stan równowagi częściowej. Tylko niektóre stany konformacyjne enzymu umożliwiają przyłączanie substratów i odczepianie produktów. Najprostszy schemat maszyny chemochemicznej przedstawiono na rysunku 4.1 po lewej stronie. Stany konformacyjne umożliwiające zachodzenie procesów zredukowane są do pojedynczej bramy wejściowej (indeks bis) i wyjściowej (indeks prim) dla procesu dostarczającego (oznaczonego jako 1) i odbierającego (oznaczonego jako 2) energię swobodną. Schemat ten można rozszerzyć na większą liczbę bram co przedstawiono na rysunku 4.1 po prawej stronie.

Stężenia substratów mogą być stałe, wtedy $[R_i] = [R_i]_0 = \text{const}$, albo mogą być

związane ze stężeniem produktów, $[R_i] + [P_i] = [R_i]_0 = \text{const.}$ Zmiennymi określającymi strumień reakcji są stężenia produktów $[P_i]$ ($i = 1, 2$). Dokładniej, strumień J_i ($i = 1, 2$) określony jest jako liczba cząsteczek produktu wytwarzana przez jedną cząsteczkę enzymu na jednostkę czasu:

$$J_i = \frac{1}{[E]_0} \frac{d[P_i]}{dt}, \quad (4.1)$$

gdzie $[E]_0$ oznacza całkowite stężenie enzymu. Sprzężone ze strumieniami J_i siły A_i ($i = 1, 2$) są z definicji różnicami pomiędzy potencjałami chemicznymi produktu i substratu:

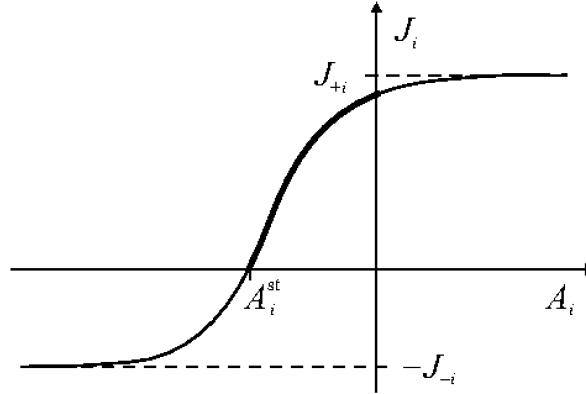
$$\beta A_i = \ln K_i \frac{[R_i]}{[P_i]}, \quad K_i \equiv \frac{[P_i]^{\text{eq}}}{[R_i]^{\text{eq}}}, \quad (4.2)$$

gdzie K_i jest stałą równowagi, a indeks eq oznacza stężenie równowagowe. Siły A_i są miarą odległości od stanu równowagi, w którym znikają [C, 8].

Dla schematu z pojedynczymi bramami udało się otrzymać wyrażenia analityczne. Zależność strumień-siła dla dwóch sprzężonych reakcji ma ogólną postać [8, 10]

$$J_i = \frac{1 - e^{-\beta(A_i - A_i^{\text{st}})}}{J_{+i}^{-1} + J_{-i}^{-1} e^{-\beta(A_i - A_i^{\text{st}})} + J_{0i}^{-1} (K_i + e^{\beta A_i})^{-1}}, \quad (4.3)$$

gdzie $\beta = (k_B T)^{-1}$. Parametry J_{+i} , J_{-i} , J_{0i} i A_i^{st} są zależne od wnętrza szarej skrzynki. A_i^{st} ma sens siły blokującej (ang. *stalling force*), dla której strumień J_i znika ($J_i(A_i^{\text{st}}) = 0$).



Rysunek 4.2. Schematyczna zależność strumienia J_i od siły A_i . Zaznaczono charakterystyczne parametry J_{+i} , J_{-i} oraz A_i^{st} . Pogrubiona linia oznacza obszar przetwarzania energii swobodnej. Za pracę [10].

Zależność strumienia J_i od siły A_i przedstawiono schematycznie na rysunku 4.2. Zależność $J_i(A_i)$ jest rosnąca z punktem przegięcia zależącym od J_{0i} i dwiema asymptotami J_{+i} i J_{-i} [8, 10].

Bardzo ważny jest iloczyn $J_2 A_2$, który ma sens mocy wyjściowej maszyny. Jeśli jest on ujemny (strumień i odpowiadająca mu siła mają przeciwny znak), to w tym obszarze zachodzi przetwarzanie energii swobodnej. Z równania (4.3) wynika, że dzieje się tak kiedy odpowiadająca siła blokująca A_i^{st} jest ujemna.

W ogólności wydajność maszyny określona jest jako stosunek mocy wyjściowej do mocy wejściowej:

$$\eta = -J_2 A_2 / J_1 A_1, \quad (4.4)$$

a sam iloraz strumieni

$$\epsilon = J_2 / J_1 \quad (4.5)$$

nazywany jest stopniem sprzężenia [10]. Jeśli $\epsilon < 1$, to mówimy, że maszyna ma poślizg.

Dla reakcji z pojedynczymi bramami udało się znaleźć wyrażenia analityczne na zależność stopnia sprzężenia od A_1 i A_2 [8, 10]:

$$\epsilon = \frac{1 - e^{-\beta(A_1 + A_2)} + W_1(A_1)(1 - e^{-\beta A_2})}{1 - e^{-\beta(A_1 + A_2)} + W_2(A_2)(1 - e^{-\beta A_1})}, \quad (4.6)$$

a siła blokująca A_2^{st} wyraża się poprzez

$$\beta A_2^{\text{st}} = \ln \frac{e^{-\beta A_1} + W_1(A_1)}{1 + W_1(A_1)}. \quad (4.7)$$

Wyrażenie na A_1^{st} można uzyskać z A_2^{st} przez zamianę indeksu 1 z 2 i na odwrót. Wielkości $W_i(A_i)$ są miarą poślizgu – im mniejszy stopień poślizgu tym mniejsza wartość odpowiadającej wielkości W_i . Stopień poślizgu odpowiada za liczbę „beziproduktywnych” cykli, czyli takich, w których proces dostarczający energię swobodną nie napędza procesu odbierającego energię swobodną. Dla przypadku braku poślizgu otrzymujemy: $A_2^{\text{st}} = -A_1$ i $A_1^{\text{st}} = -A_2$. Wtedy stopień sprzężenia osiąga maksymalną możliwą wartość dla schematu z jednym procesem dostarczającym i jednym odbierającym energię swobodną, $\epsilon = 1$.

Stopień poślizgu $W_1(A_1)$ dany jest przez wyrażenie

$$W_1(A_1) = \frac{\tau_M(1'' \leftrightarrow \{1', 2'\}) + \tau_M(1' \leftrightarrow \{1'', 2''\})e^{-\beta A_1} + \tau_1(A_1)}{\tau_M(1' \leftrightarrow \{1'', 2'\}) - \tau_M(1' \leftrightarrow \{1'', 2''\})}, \quad (4.8)$$

natomiast po zastąpieniu indeksów 1 z 2 i na odwrót otrzymujemy wyrażenie dla stopnia poślizgu $W_2(A_2)$ [8]:

$$W_2(A_2) = \frac{\tau_M(2'' \leftrightarrow \{1', 2'\}) + \tau_M(2' \leftrightarrow \{1'', 2''\})e^{-\beta A_2} + \tau_2(A_2)}{\tau_M(2' \leftrightarrow \{1', 2''\}) - \tau_M(2' \leftrightarrow \{1'', 2''\})}. \quad (4.9)$$

Jak pokażemy później, kolejność stanów w nawiasach klamrowych nie ma znaczenia, stąd przy zamianie indeksów zachowywano kolejność z $W_1(A_1)$ jeśli było to możliwe.

W równaniach (4.8) i (4.9) odpowiednie czasy $\tau_i(A_i)$ związane są z odwrotnościami stałych szybkości przejść na zewnątrz szarej skrzynki reprezentującej kompleks przejściowy (rysunek 4.1), natomiast τ_M są średnimi czasami pierwszego przejścia wewnątrz. Oba równania zawierają ten sam mianownik, co wymaga osobnego dowodu analitycznego [C], lecz w dalszej części pracy zostanie to udowodnione także numerycznie.

Obie wartości W_i mają ten sam znak, dodatni lub ujemny. Stąd wynikają ważne wnioski z wcześniejszych równań. W równaniu (4.7) wyrażenie pod logarytmem jest zawsze dodatnie. Dodatkowo jeśli $\beta A_1 \geq 0$ i $\beta A_2 \leq 0$ i oba W_i są dodatnie, to znaki βA_i^{st} z równania (4.7) są takie same jak powiązanych z nimi sił βA_i (obszar przetwarzania energii swobodnej).

Przy założeniu, że $\beta A_1 \rightarrow \infty$, równanie (4.7) redukuje się do postaci

$$\beta A_2^{\text{st}}(\infty) = \ln \frac{W_1(\infty)}{1 + W_1(\infty)}, \quad (4.10)$$

podczas gdy równanie (4.8) przyjmuje postać

$$W_1(\infty) = \frac{\tau_M(1'' \leftrightarrow \{1', 2'\}) + \tau_1(A_1)}{\tau_M(1' \leftrightarrow \{1'', 2'\}) - \tau_M(1' \leftrightarrow \{1'', 2''\})}. \quad (4.11)$$

Przy założeniu, że $\beta A_1 \rightarrow \infty$ i $\beta A_2 = 0$, równanie (4.6) redukuje się do postaci

$$\epsilon(0) = \frac{1}{1 + W_2(0)}, \quad (4.12)$$

gdzie $\epsilon(0)$ jest największym możliwym stopniem sprzężenia w obszarze przetwarzania energii swobodnej. Natomiast równanie (4.9) przyjmuje postać

$$W_2(0) = \frac{\tau_M(2'' \leftrightarrow \{1', 2'\}) + \tau_M(2' \leftrightarrow \{1'', 2''\}) + \tau_2(A_2)}{\tau_M(2' \leftrightarrow \{1', 2''\}) - \tau_M(2' \leftrightarrow \{1'', 2''\})}. \quad (4.13)$$

Dla $W_2(0)$ równego zeru, stopień sprzężenia $\epsilon(0)$ wynosiłby 1.

W równaniach (4.8) oraz (4.9) występują średnie czasy pierwszego przejścia „do przodu” i „do tyłu” zdefiniowane jako [8]

$$\tau_M(a \leftrightarrow \{b, c\}) = \tau_M(a \rightarrow \{b, c\}) + \tau_M(\{b, c\} \rightarrow a). \quad (4.14)$$

O ile człon $\tau_M(a \rightarrow \{b, c\})$ można łatwo zinterpretować jako średni czas pierwszego przejścia od stanu a do stanu b lub c , to już interpretacja czasu $\tau_M(\{b, c\} \rightarrow a)$ jest problematyczna nie tylko koncepcyjnie ale i obliczeniowo (jeden wędrowiec nie może jednocześnie rozpocząć ewolucji z dwóch stanów). Można próbować śledzić ewolucję dwóch wędrowców z tym samym zestawem liczb pseudolosowych i sprawdzać, z którego stanu doszedł jako pierwszy, ale znów nie we wszystkich ewolucjach dojdzie pierwszy startując z tego stanu. Problem ten można znacznie prościej rozwiązać analitycznie korzystając z wyrażenia [8]

$$\tau_M(b \rightarrow a) = \tau_M(\{b, c\} \rightarrow a) + \tau_M(b \rightarrow \{a, c\}), \quad (4.15)$$

co prowadzi nas do równania

$$\tau_{\mathbf{M}}(\{b, c\} \rightarrow a) = \tau_{\mathbf{M}}(b \rightarrow a) - \tau_{\mathbf{M}}(b \rightarrow \{a, c\}). \quad (4.16)$$

Ostatecznie zatem równanie (4.14) można zapisać jako

$$\tau_{\mathbf{M}}(a \leftrightarrow \{b, c\}) = \tau_{\mathbf{M}}(a \rightarrow \{b, c\}) + \tau_{\mathbf{M}}(b \rightarrow a) - \tau_{\mathbf{M}}(b \rightarrow \{a, c\}). \quad (4.17)$$

Założenie, że mamy tylko jeden proces dostarczający i jeden odbierający energię swobodną jest bardzo ograniczające. Niestety próba wyprowadzenia wyrażeń analitycznych dla większej liczby procesów prowadzi do niezwykle nieprzejrzystych wzorów. Wobec tego jedyną rozsądną metodą badań są symulacje Monte Carlo na konkretnych sieciach. Pierwszym krokiem jest porównanie wyników symulacji z wyrażeniami analitycznymi dla pojedynczych procesów. Zgodność wyników upewni nas, że po rozszerzeniu modelu na większą liczbę procesów, wyniki będą nadal wiarygodne. Ze względu na postać wyrażenia (4.17) dyskusję analityczną kiedy $\epsilon(0)$ jest największe można przeprowadzić na dowolnej sieci, jednakże należy obliczyć występujące wartości średnich czasów pierwszego przejścia. Stąd też wybór konkretnych stanów na danej sieci i ich geometria wpływają na średnie czasy pierwszego przejścia, a stąd na stopień sprzężenia. Najważniejszym celem jest poszukiwanie takiego modelu, w którym stopień sprzężenia może przekroczyć wartość 1. Jest to próba realizacji idei tzw. „przekładni molekularnej”, czyli układu, w którym na jeden proces dostarczający energię swobodną może realizować się kilka procesów odbierających energię swobodną [54].

Należy zaznaczyć, że w tym rozdziale przedstawione zostaną wyniki dwóch niezależnych eksperymentów numerycznych, dla których napisano różne programy. W pierwszym z nich wyznaczano średnie czasy pierwszego przejścia, w drugim strumienie. Oba eksperymenty razem wzięte pozwolą na porównanie wyników symulacji z przewidywaniami teoretycznymi.

4.2 Szczegóły symulacji i wyniki

W symulacjach dynamiki przejść konformacyjnych zazwyczaj potrzebujemy sieci, na której każdy stan będzie miał więcej niż jednego sąsiada. Najprostszym i najbardziej symetrycznym typem sieci, dla której nie trzeba definiować warunków brzegowych, jest wielowymiarowy hipersześcian (patrz rozdział na temat metodologii badań). Dyskusję przeprowadzimy dla przypadku $d = 5$, dla którego każdy stan ma pięciu najbliższych sąsiadów. Rzeczywiste sieci mają różne energie stanów konformacyjnych. W związku z tym, że dyskusja byłaby zbyt złożona ograniczymy się do stanów izoenergetycznych.

Zespół statystyczny układów (wędrowców) umieszczono na wielowymiarowym hipersześcianie, którego węzły (stany konformacyjne kompleksu przejściowego) ponumerowano liczbami całkowitymi od $m = 0$ do $m = L - 1$ (L oznacza liczbę węzłów). Na sieci zachodzą dwa rodzaje przejść pomiędzy stanami: wewnętrzne (wewnątrz szarej skrzynki) oraz zewnętrzne (pomiędzy bramami). Przejścia wewnętrzne dozwolone są tylko pomiędzy sąsiednimi stanami, a odległość pomiędzy wszystkimi węzłami sieci jest stała. Wybrane stany będące bramami pozwalają na zachodzenie z określonym prawdopodobieństwem przejść zewnętrznych – procesów dostarczających i odbierających energię swobodną. W tych przejściach zaburzany jest warunek równowagi szczegółowej. Z bram mogą zachodzić też przejścia wewnętrzne. Symulacja składa się z ewolucji pewnego zespołu statystycznego wędrowców przez określony czas mierzony liczbą kroków, a głównymi obliczanymi tu wielkościami są strumienie procesów dostarczających i odbierających energię swobodną.

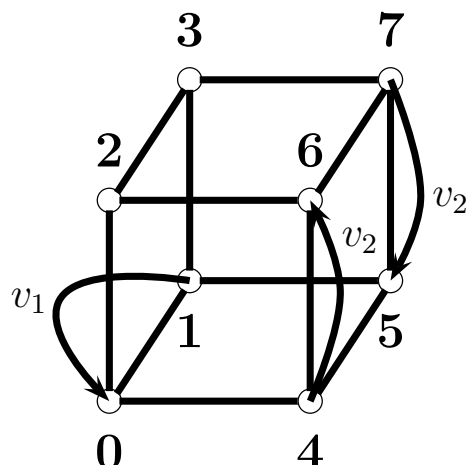
Warto wspomnieć, że wybór hipersześcianu w początkowych etapach tworzenia i testowania modelu nie jest wyborem przypadkowym. W odróżnieniu od sieci liniowej, na hipersześcianie nie jest potrzebna bariera odbijająca (rozumiana jako rodzaj warunku brzegowego). Jednakże już na trójkącie Sierpińskiego warunki brzegowe muszą być zdefiniowane i przy zbyt małym rzędzie trójkąta wybór konkretnego typu warunków brzegowych ma wpływ na przebieg symulacji. Ponadto ze względu na hierarchiczność sieci fraktalnych (system „wąskich gardeł”) czas pierwszego przejścia nie musi być identyczny dla różnych par stanów będących w tej samej odległości od siebie. Co więcej zamieniając stan początkowy z końcowym często otrzymuje się różne średnie czasy pierwszego przejścia, choć prawdopodobieństwo równowagowego obsadzenia stanów nie zmienia się. Szukając możliwie najlepszego modelu dla procesu należy unikać, przynajmniej w początkowych etapach jego tworzenia, potencjalnie niebezpiecznych zachowań układu na sieci, stąd też wybór sieci regularnej w postaci hipersześcianu, która ma jednoznacznie określoną zależność średniego czasu pierwszego przejścia od odległości (rysunek 2.11). Wybór $d = 5$ także nie jest przypadkowy. Z rysunku 2.12 wynika jak szybko potrafi rosnąć średni czas pierwszego przejścia przy zmianie wymiaru hipersześcianu nawet przy odległości 1 pomiędzy stanem początkowym i końcowym. Stąd też wybór $d = 5$ warunkuje rozsądne pod względem czasu obliczeń średnie czasy pierwszego przejścia i zapewnia dostateczną liczbę stanów.

Przykładową realizację na sieci o wymiarze $d = 3$ z jednym procesem dostarczającym energię swobodną i dwoma odbierającymi przedstawiono na rysunku 4.3 (w tym przypadku przejście zewnętrzne powoduje przeskok do stanu odległego o 1, ale w ogólności tak być nie musi).

Parametry wejściowe symulacji przedstawiono w tabeli 4.1. W programie umiesz-

Tabela 4.1. Parametry wejściowe symulacji numerycznych dynamiki przejść konformacyjnych.

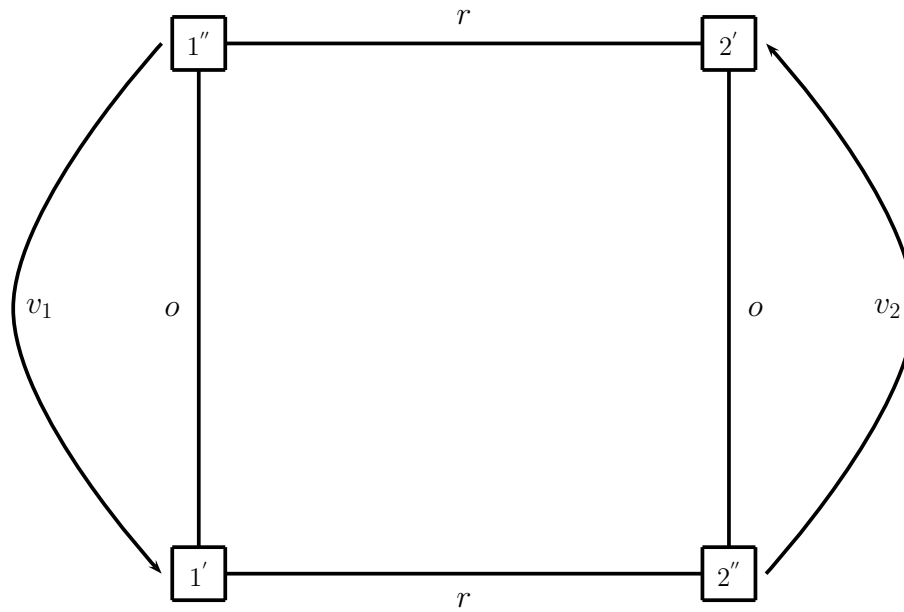
Parametr	Znaczenie
N_w	Liczba układów (wędrówców), których ewolucja była badana.
N_k	Czas ewolucji każdego wędrowca wyrażony w liczbie kroków symulacji (z uwzględnieniem możliwości czekania).
N_{P1}	Liczba bram (wejściowa plus wyjściowa) typu 1 (dostarczanie energii swobodnej).
N_{P2}	Liczba bram (wejściowa plus wyjściowa) typu 2 (odbieranie energii swobodnej).
$a_1 = \beta A_1$	Siła napędzająca przejście zewnętrzne typu 1.
$a_2 = \beta A_2$	Siła napędzająca przejście zewnętrzne typu 2.
v_1	Prawdopodobieństwo przejścia zewnętrznego typu 1 „do przodu”.
v_2	Prawdopodobieństwo przejścia zewnętrznego typu 2 „do przodu”.
$v_1 \exp(-a_1)$	Prawdopodobieństwo przejścia zewnętrznego typu 1 „do tyłu” (wyznaczane na podstawie v_1 i a_1).
$v_2 \exp(-a_2)$	Prawdopodobieństwo przejścia zewnętrznego typu 2 „do tyłu” (wyznaczane na podstawie v_2 i a_2).
p	Prawdopodobieństwo przejść wewnętrznych pomiędzy stanami konformacyjnymi.



Rysunek 4.3. Przykład realizacji modelu przejść konformacyjnych na sześcianie z jednym procesem dostarczającym energię swobodną (z prawdopodobieństwem v_1) i dwoma odbierającymi (z prawdopodobieństwem v_2). Strzałkami zaznaczono kierunek przejść zewnętrznych „do przodu”. Nie zostały zaznaczone przejścia zewnętrzne „do tyłu”.

czono kod sprawdzający czy dany zestaw parametrów jest dozwolony i czy sieć jest prawidłowo zdefiniowana. Prawdopodobieństwa przejść wewnętrznych p pomiędzy wszystkimi stanami są identyczne (wszystkie stany mają taką samą równowagową energię swobodną) – zależą one od parametrów ale nie zmieniają się zależnie od stanu (różnica w stosunku do poprzednich symulacji dynamiki wibracyjnej na sieci liniowej). Konieczność wprowadzenia dodatkowych prawdopodobieństw przejść ze stanów będących bramami zrekompensoвано niezerowym prawdopodobieństwem czekania przez jeden krok iteracji. Wprowadzenie czekania było konieczne, gdyż prawdopodobieństwa przejść wewnętrznych do i od stanu będącego bramą muszą być identyczne, a całkowite prawdopodobieństwo wszystkich możliwych przejść z danego stanu musi być równe jedności. Z drugiej strony długie czasy czekania wymuszają użycie dłuższych czasów symulacji by zminimalizować fluktuacje (układy-wędrowcy muszą zdążyć zwiedzić całą sieć przynajmniej raz). Mój program umożliwia użycie dowolnej sieci (definiowanej przez listę sąsiadów wczytywaną z zewnętrznego pliku) oraz definicję dowolnej liczby procesów dostarczających i odbierających energię swobodną. Warto jednak zauważyć, że zdefiniowanie zbyt rozległej sieci przy zachowaniu tej samej liczby wędrowców i liczby kroków każdego z nich może spowodować duże fluktuacje, a na pewno mniejszą efektywność przetwarzania energii swobodnej (większe problemy z dotarciem do stanów będących bramami). Dobór zestawu parametrów nie musi być prosty – może okazać się, że dany zestaw jest niedopuszczalny, gdyż odpowiednie całkowite prawdopodobieństwa przejść w którymś ze stanów nie będą zawierały się pomiędzy 0 a 1.

Warto wskazać kolejne różnice w porównaniu z poprzednimi symulacjami dynamiki

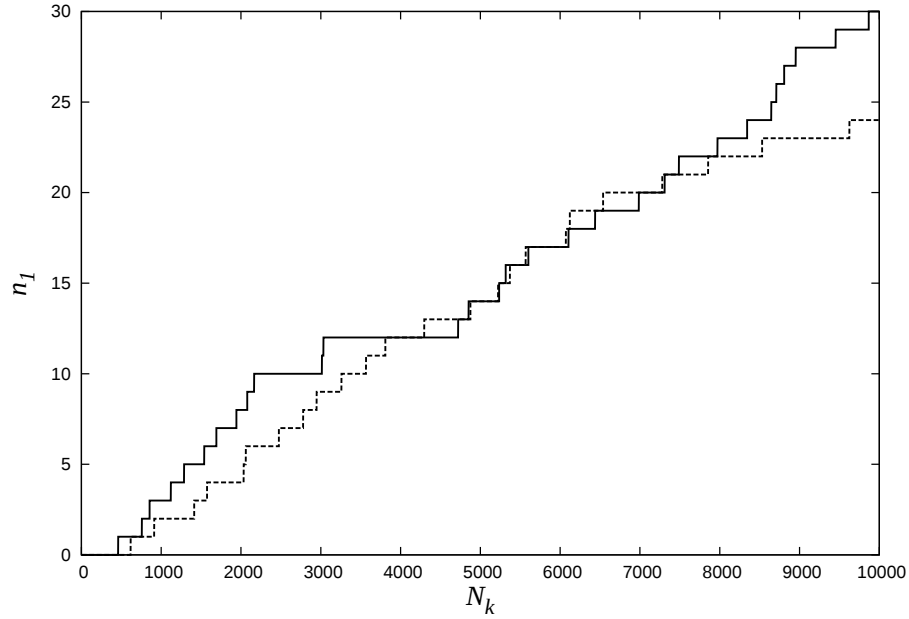


Rysunek 4.4. Ogólna geometria bram dla modelu z jednym procesem dostarczającym i jednym odbierającym energię swobodną. Zaznaczono charakterystyczne odległości r i o oraz ustalony kierunek przejść zewnętrznych „do przodu”. Nie zostały zaznaczone przejścia zewnętrzne „do tyłu” określone przez czynniki $\exp(-a_1)$ oraz $\exp(-a_2)$.

wibracyjnej (poprzedni rozdział). Składa się na nie większa liczba bram, prawdopodobieństwo zajścia procesu mniejsze od jedności, odwracalność procesu (choć można to kontrolować), a także wprowadzenie odległości pomiędzy bramami, która na sieci może być większa od jednej krawędzi.

Ogólny schemat geometrii bram praktycznie na dowolnej sieci dla przypadku jednego procesu dostarczającego i jednego odbierającego energię swobodną wraz z zaznaczonymi parametrami związanymi z odległościami przedstawiono na rysunku 4.4. Aby nie dodawać dodatkowych parametrów przyjęto, że jest jedna odległość typu o i jedna typu r (wykorzystaliśmy symetrię obrotu o 180 stopni, patrz rysunek 4.1 po lewej stronie). Zależnie od odległości o i r , pomiędzy bramami mogą znajdować się inne stany. Dla o mniejszego od maksymalnego, parametrem stają się także odległości pomiędzy stanami $1'$ i $2'$ oraz $1''$ i $2''$, które w ogólności mogą być różne. W dalszych podrozdziałach zostaną pokazane wybrane sposoby rozszerzenia modelu z rysunku 4.4 na więcej bram oraz inne typy sieci.

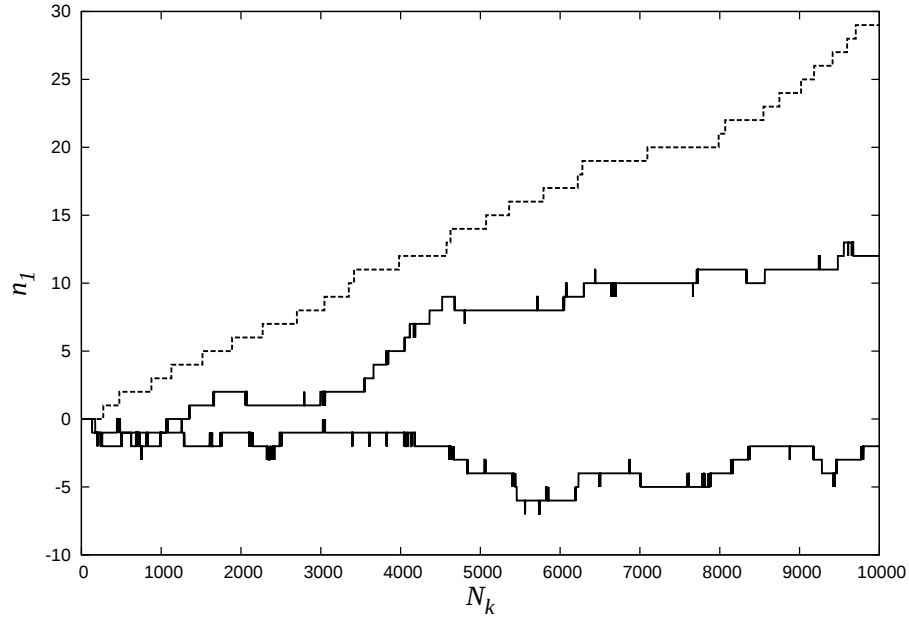
Mimo, że ostatecznie znaczenie będą miały wartości strumieni będące ilorazem liczby przejść i czasu ewolucji, to warto pokazać też ewolucję pojedynczego wędrowca. Na rysunku 4.5 przedstawiono zależność liczby przejść typu 1 (związana z procesem dostarczającym energię swobodną) od czasu wyrażonego w liczbie kroków wędrowca (od 0 do 10000). Jak widać każda ewolucja przebiega trochę inaczej i przy krótkich czasach sens



Rysunek 4.5. Dwie przykładowe zależności liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ od czasu wyrażonego w liczbie kroków N_k dla dwóch ewolucji pojedynczego wędrowca na pięciowymiarowym hipersześcianie. Parametry użyte w symulacji: $N_{P1} = 1$, $N_{P2} = 1$, $v_1/p = 50$, $v_2/p = 30$, $p = 1,82 \cdot 10^{-2}$, $a_1 = 10$, $a_2 = 5,5$, $r = 1$, $o = 5$, $v_1 \exp(-a_1) = 4,13 \cdot 10^{-5}$, $v_2 \exp(-a_2) = 2,23 \cdot 10^{-3}$.

ma dopiero wartość średnia strumienia. Natomiast dla wystarczająco długich czasów (co w praktyce trudno określić) strumienie w obu symulacjach powinny dążyć do tej samej wartości. Warto zauważyć, że gdyby zatrzymać symulację na ok. 6000 krokach, to można wyciągnąć błędny wniosek, że ta stała wartość strumienia została już osiągnięta. Prawdopodobieństwa będące liczbami rzeczywistymi będą podawane w tej pracy zazwyczaj z dokładnością do trzech miejsc znaczących. Jednakże w programach reprezentowane są jako liczby podwójnej precyzji.

Warto jeszcze zwrócić uwagę na odwracalność procesów, tzn. z parametrów a_1 , a_2 , v_1 , v_2 wynikają również prawdopodobieństwa przejść zewnętrznych „do tyłu” – są to $v_1 \exp(-a_1)$ oraz $v_2 \exp(-a_2)$. Z parametrów przedstawionych w podpisie do rysunku 4.5 wynika, że prawdopodobieństwa zachodzenia przejść „do tyłu” były dużo mniejsze od prawdopodobieństw zachodzenia przejść „do przodu”. Odpowiedni dobór prawdopodobieństw przejść „do przodu” i „do tyłu” pozwala kontrolować odwracalność procesów, co przedstawiono na rysunku 4.6. Schodkowość zależności wynika z przejść wewnętrznych pomiędzy stanami konformacyjnymi oraz z dopuszczenia możliwości czekania na jeden krok czasowy. Warto zaznaczyć, że czekanie w żaden sposób nie wpływa na dynamikę opisywaną równaniem fundamentalnym (1.7), gdyż stan początkowy pokrywa się



Rysunek 4.6. Przykładowe zależności liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ od czasu wyrażonego w liczbie kroków N_k dla ewolucji pojedynczego wędrowca na pięciowymiarowym hipersześcianie. Poprzez zmianę siły a_1 uzyskano efekt odwracalności procesu. Linia przerywaną zaznaczono symulację z parametrami identycznymi jak na rysunku 4.5. W kolejnych dwóch symulacjach (linia ciągła) zmieniono wartość siły napędzającej na $a_1 = 1$ ($v_1 \exp(-a_1) = 0,334$) oraz $a_1 = 0,1$ ($v_1 \exp(-a_1) = 0,823$). Już dla $a_1 = 0,1$ widoczny jest efekt głównie cofania się procesu (dolna krzywa), chociaż w dłuższym czasie proces średnio przebiega do przodu.

z końcowym, $S_i = S_j$. Wtedy $w_{ij} = w_{ji}$, a $p(S_j, t) = p(S_i, t)$ i oba składniki w sumie znoszą się.

Bardzo ważnym elementem symulacji jest definicja prawdopodobieństwa przejść wewnętrznych p . Wybieramy stan, z którego prawdopodobieństwo przejść zewnętrznych jest największe i przyjmujemy, że w tym stanie prawdopodobieństwo czekania jest równe zero. W związku z tym, że mamy dwa prawdopodobieństwa przejść zewnętrznych v_1 i v_2 (definiujemy tylko typ reakcji, a liczba reakcji danego typu na sieci może być dowolna), istnieją cztery możliwości zdefiniowania p . Możliwe definicje prawdopodobieństw przejść wewnętrznych p przedstawiono równaniami od (4.18) do (4.21) (n jest liczbą sąsiadów, czyli w tym przypadku wymiarem hipersześcianu):

$$np + v_1 = 1, \quad p = \frac{1 - v_1}{n} \quad (4.18)$$

$$np + v_1 \exp(-a_1) = 1, \quad p = \frac{1 - v_1 \exp(-a_1)}{n} \quad (4.19)$$

$$np + v_2 = 1, \quad p = \frac{1 - v_2}{n} \quad (4.20)$$

$$np + v_2 \exp(-a_2) = 1, \quad p = \frac{1 - v_2 \exp(-a_2)}{n} \quad (4.21)$$

odpowiednio dla stanów typu $1''$, $1'$, $2''$, $2'$ (porównaj rysunek 4.1).

W symulacjach przyjęto, że $v_1 > v_2$, stąd najrozsądniejszy jest wybór pierwszej definicji p (równanie (4.18)), szczególnie jeśli interesuje nas głównie zakres sił odpowiadający przetwarzaniu energii swobodnej. Prawdopodobieństwo czekania we wszystkich stanach oprócz wyróżnionego zdefiniowane jest jako

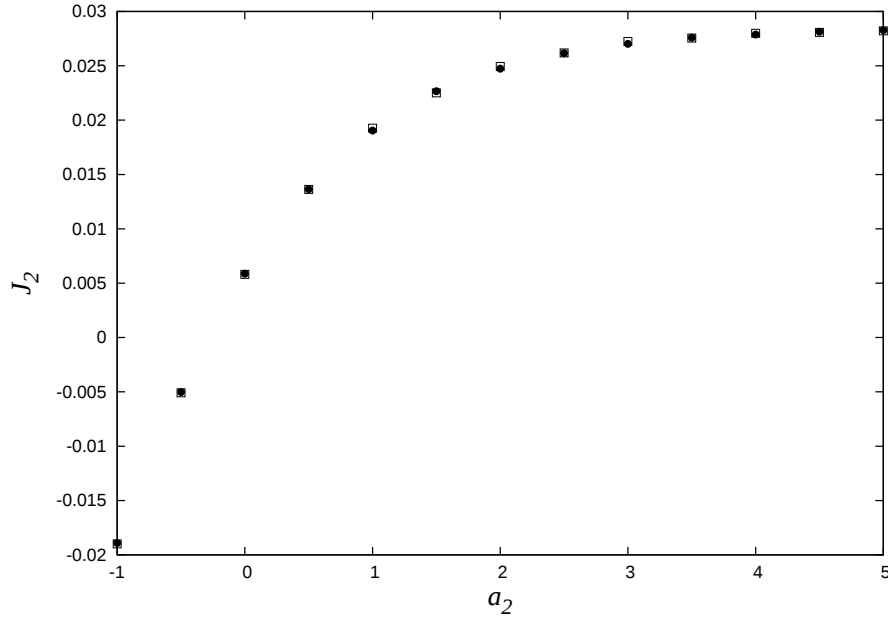
$$p_c(i) = 1 - v(i) - np, \quad (4.22)$$

gdzie $v(i)$ oznacza prawdopodobieństwo przejścia zewnętrznego ze stanu i (odpowiednio v_1 , v_2 , $v_1 \exp(-a_1)$, $v_2 \exp(-a_2)$ dla wyróżnionych stanów będących bramami, a 0 dla pozostałych stanów), n oznacza liczbę sąsiadów, p jest prawdopodobieństwem przejścia wewnętrznego.

Aby uzyskać ostateczną wartość strumienia, liczbę przejść każdego wędrowca (równą różnicy liczby przejść danego typu „do przodu” i „do tyłu”) należy podzielić przez liczbę kroków N_k oraz wartość p . Następnie tak otrzymane wartości są sumowane i dzielone przez liczbę wędrowców N_w . Rzeczywisty czas ewolucji z pominięciem czekania można utożsamiać z efektywną długością ścieżki rozumianą jako $N_w \cdot N_k \cdot p$. Nie bez znaczenia jest zatem wybór prawdopodobieństwa przejść wewnętrznych p , który w powiązaniu z liczbą kroków określa nam rzeczywisty czas wędrówek, czyli stopień zwiedzenia sieci. Wielkość $N_w \cdot N_k \cdot p$ powinna być dużo większa od największej możliwej odległości pomiędzy dwoma stanami na sieci danego typu. Przyjęta jednostka czasu to p^{-1} . Występujące w równaniu fundamentalnym prawdopodobieństwa w_{ij} także muszą być odniesione do używanej w symulacji jednostki czasu. Przy tym wyborze jednostki czasu prawdopodobieństwo przejścia wewnętrznego na jednostkę czasu wynosi $p/p = 1$.

W praktyce mamy dodatkową piątą możliwość definicji p – dla bardzo ujemnych wartości a_2 należy ręcznie ustawić wartość p tak, aby były spełnione różne warunki poprawności definicji sieci i prawdopodobieństw (w takim przypadku może się okazać, że we wszystkich stanach będzie konieczność czekania, a czekanie będzie bardzo długie).

Jak wspomniano wcześniej, wartości średnie strumieni dzielone są przez p , dzięki czemu uzyskuje się wielkości niezależne od czasu symulacji. Używanymi bezpośrednio w symulacji wartościami prawdopodobieństw przejść zewnętrznych są v_1 i v_2 . Jednakże prawdopodobieństwami przejść na przyjętą jednostkę czasu są ilorazy v_1/p oraz v_2/p . Z wielkości tych, korzystając z równań od (4.18) do (4.21), można wyznaczyć p , a następnie v_1 i v_2 .



Rysunek 4.7. Zależność strumienia J_2 od a_2 dla dwóch symulacji **S1** (czarne kropki) i **S2** (puste kwadraciki). Parametry użyte w symulacji: $N_w = 10^2$, $N_k = 10^6$, $N_{P1} = 1$, $N_{P2} = 1$, $a_1 = 10$, a_2 w przedziale od -1 do 5 , $r = 1$, $o = 5$. W obu symulacjach stosunek $v_1/p = 5$, a $v_2/p = 1$.

W programie można zdefiniować prawdopodobieństwa przejść zewnętrznych za pomocą v_1 i v_2 albo za pomocą v_1/p i v_2/p (program oblicza wtedy p , v_1 i v_2). W dalszej części konsekwentnie będę posługiwał się wielkościami v_1/p i v_2/p . Powód jest dość ważny – możliwy jest wybór wielu równoważnych par v_1 i v_2 dających taką samą zależność strumień-siła, dla których zachowany jest właśnie stosunek v_1/p oraz v_2/p . W rezultacie konieczność czekania i wpływ tego faktu na ewolucję nie wpływa na końcową wartość strumieni dzięki dzieleniu przez p (oczywiście w przypadku obliczania stopnia sprężenia problem znika w sposób naturalny).

W celu wykazania prawdziwości stwierdzenia na temat istotności wyłącznie stosunku v_1/p oraz v_2/p przeprowadzono dwie symulacje **S1** i **S2**, których wynik przedstawiono na rysunku 4.7. Dla obu symulacji stosunek $v_1/p = 5$, a $v_2/p = 1$, natomiast różniły się wartościami v_1 , v_2 oraz p . W symulacji **S1** użyto $v_1 = 0,5$, $v_2 = 0,1$, stąd korzystając z równania (4.18) $p = 0,1$. Natomiast w symulacji **S2** użyto $v_1 = 0,25$, $v_2 = 0,05$ i ręcznie ustawiono $p = 0,05$. Na rysunku 4.7 widać, że obie zależności pokrywają się, stąd prawdziwość wniosku o istotności tylko stosunków v_1/p oraz v_2/p . Świadomie wybrano takie parametry symulacji aby drobne fluktuacje spowodowały niedokładne pokrywanie się krzywych.

Warto podkreślić fakt, że zwykle powodem fluktuacji wyznaczania wartości mierzonej w symulacjach Monte Carlo jest niewystarczająco liczny zbiór liczb pseudolosowych

Tabela 4.2. Procentowe błędy $\Delta_{\%}$ wyznaczenia wartości strumienia J_1 i J_2 oraz stopnia sprzężenia ϵ w zależności od iloczynu liczby wędrówców N_w i liczby kroków N_k . Wyniki przedstawione w dwóch ostatnich wierszach pokazują, że w przypadku zbyt krótkiej ścieżki mogą pojawić się efekty wyboru stanu początkowego.

$N_w \cdot N_k$	$\Delta_{\%} J_1$	$\Delta_{\%} J_2$	$\Delta_{\%} \epsilon$
10^9	0,06	0,3	0,3
10^8	0,2	1	1
10^7 (gdy $N_k = 10^5$)	0,6	3	3
10^7 (gdy $N_k = 10^3$)	0,6	4	4

użytych w eksperymencie (zbyt krótki efektywny czas ewolucji). Jednakże rozważany tu układ będzie jeszcze bardziej podatny na fluktuacje, gdyż występują dość znaczne prawdopodobieństwa czekania p_c w stosunku do prawdopodobieństw przejść wewnętrznych p (jeśli liczba kroków będzie zbyt mała to układ tym bardziej nie zdąży zwiedzić sieci). Niezwykle istotny jest dobór odpowiedniej liczby kroków i wędrówców do liczby węzłów sieci.

W tabeli 4.2 przedstawiono procentowe błędy (równanie (2.5)) wyznaczenia strumienia J_1 i J_2 oraz stopnia sprzężenia ϵ w zależności od iloczynu liczby wędrówców N_w i liczby kroków N_k . Wartość stopnia sprzężenia ϵ obarczona jest też pewnym błędem zależnym od błędów wyznaczania wartości strumieni. Jeśli wartości błędów uzyskanych z równania (2.4) oznaczmy jako ΔJ_1 i ΔJ_2 odpowiednio dla strumienia pierwszego i drugiego, to błąd wyznaczenia stopnia sprzężenia $\Delta \epsilon$ można obliczyć korzystając z równania (2.6), a na tej podstawie błąd procentowy z równania (2.5). Dla naszego przypadku będzie to odpowiednio

$$|\Delta \epsilon| = |\epsilon| \left(\frac{|\Delta J_1|}{|J_1|} + \frac{|\Delta J_2|}{|J_2|} \right) \quad (4.23)$$

i

$$\Delta_{\%} \epsilon = \frac{|\Delta \epsilon|}{|\epsilon|} \cdot 100\% . \quad (4.24)$$

Wykonano dwa typy symulacji: **S3**, w której każdy kolejny wędrowiec rozpoczynał ewolucję w tym samym stanie, oraz **S4**, w której stan początkowy kolejnego wędrowca był stanem końcowym poprzedniego (co odpowiada wykonaniu odpowiednio dłuższej symulacji z jednym wędrowcem). Z otrzymanych wyników można wyciągnąć dwa bardzo istotne wnioski: oba strumienie wyznaczane są z różnymi błędami oraz przy zbyt krótkim efektywnym czasie ewolucji pojedynczego wędrowca ($N_k \cdot p$) ujawnia się efekt

wyboru stanu początkowego i nieosiągnięcia stanu stacjonarnego (większy błąd wyznaczenia wielkości). Pokazano to w dwóch ostatnich wierszach tabeli 4.2. Mimo takiej samej wartości iloczynu $N_w \cdot N_k = 10^7$, błędy wyznaczania wielkości są różne, gdyż w ostatnim wierszu $N_k = 10^3$, a $p = 0,018$ stąd efektywny czas ewolucji dla jednego wędrowca wynosił 18 kroków, co jest wielkością zbyt małą. Efektu tego nie obserwowano gdy każdy kolejny wędrowiec rozpoczynał ewolucję ze stanu, w którym skończył swą ewolucję poprzedni wędrowiec (symulacja S4).

Przedstawione błędy wyznaczania strumieni i stopnia sprzężenia dotyczą zmieniającej się liczby wędrowców i kroków ale ustalonych pozostałych parametrów. Niestety błędy zmieniają się także w zależności od parametrów. Główny wpływ ma tu relacja pomiędzy liczbą kroków a p . Jest to całkowicie uzasadnione, gdyż jeśli wędrowiec spędzi więcej czasu ze swojej ewolucji na czekaniu niż na zwiedzaniu sieci, to różnice pomiędzy ewolucją wędrowców zwiększają się przy zachowaniu określonej liczby kroków. W dalszych symulacjach każdy wędrowiec będzie rozpoczynał ewolucję z tego samego stanu, gdyż efektywny czas ewolucji pojedynczego wędrowca będzie wystarczająco długi aby cała sieć została zwiedzona przynajmniej jeden raz.

4.2.1 Wyznaczanie średnich czasów pierwszego przejścia

Teoria jest użyteczna pod warunkiem, że znamy średnie czasy pierwszego przejścia występujące w równaniach (4.8) i (4.9). Warto przyjrzeć się jak średnie czasy pierwszego przejścia zależą od odległości dla przypadku hipersześcianu o wymiarze $d = 5$ dla dwóch typów warunków końca wędrówki: $\tau_M(a \rightarrow b)$ (od stanu a do b) oraz $\tau_M(a \rightarrow \{b, c\})$ (od stanu a do stanu b lub c). Technicznie przejście pomiędzy dwoma podejściami wymaga modyfikacji jednej linii w programie w warunku kontynuacji ewolucji wędrowca z

```
do while (a /= b)
```

```
na
```

```
do while ((a /= b) .and. (a /= c))
```

Efektywna liczba wędrowców wynosiła 10^9 , a odchylenie standardowe średniej (równanie 2.4) wynosiło 0,001. W związku z tym, że na hipersześcianie o wymiarze $d = 5$ każdy stan ma pięciu sąsiadów, to prawdopodobieństwo przejścia do sąsiedniego stanu było równe 0,20. Dla diskutowanego problemu nie ma oczywiście przejść zewnętrznych, zatem nie ma też problemu czekania.

W tabeli 4.3 przedstawiono wyniki dla przypadku $\tau_M(a \rightarrow b)$. Na sieci regularnej typu hipersześcianu bez warunków brzegowych średni czas pierwszego przejścia będzie identyczny dla różnych par stanów w danej odległości. Jednakże już dla sieci typu trójkąta

Tabela 4.3. Zależność średniego czasu pierwszego przejścia $\tau_M(a \rightarrow b)$ od odległości pomiędzy stanami a i b dla hipersześcianu o wymiarze $d = 5$. Błąd wyznaczenia $\Delta\tau_M = 0,001$.

Odległość $r_{a \rightarrow b}$	$\tau_M(a \rightarrow b)$
1	31,000
2	37,498
3	40,168
4	41,666
5	42,667

Tabela 4.4. Możliwe stany końcowe w odległościach $r_{a \rightarrow b}$ od 1 do 5 dla hipersześcianu o wymiarze $d = 5$ gdy stanem początkowym jest 0. Po zapisaniu numerów stanów w systemie dwójkowym stanie się widoczne, że odległość oznacza na ilu miejscach bity są różne.

$r_{a \rightarrow b}$	$a \rightarrow b$
1	$0 \rightarrow \{1, 2, 4, 8, 16\}$
2	$0 \rightarrow \{3, 5, 6, 9, 10, 12, 17, 18, 20, 24\}$
3	$0 \rightarrow \{7, 11, 13, 14, 19, 21, 22, 25, 26, 28\}$
4	$0 \rightarrow \{15, 23, 27, 29, 30\}$
5	$0 \rightarrow \{31\}$

Sierpińskiego wybór pary stanów ma znaczenie (system wąskich gardeł i warunki brzegowe), a stąd dyskusja zależności od odległości byłaby utrudniona. Obliczenia tego typu są bezpośrednim dowodem na przydatność przetwarzania równoległego w celu uzyskania kilkudziesięciu czasów z żadaną dokładnością w rozsądnym czasie rzeczywistym.

Niestety przypadek $\tau_M(a \rightarrow \{b, c\})$ komplikuje sytuację, gdyż istotne są nie tylko odległości $r_{a \rightarrow b}$ i $r_{a \rightarrow c}$, ale i odległość $r_{b \rightarrow c}$. Tak więc na hipersześcianie jeszcze zwykły czas pierwszego przejścia $\tau_M(a \rightarrow b)$ jest jednoznacznie zależny od odległości niezależnie od wyboru stanów, natomiast czas pierwszego przejścia typu $\tau_M(a \rightarrow \{b, c\})$ należy już dodatkowo określić wprowadzając dodatkowy parametr mierzący odległość stanu b od c .

Tabela 4.5. Zależność średniego czasu pierwszego przejścia $\tau_M(a \rightarrow \{b, c\})$ dla różnych odległości $r_{a \rightarrow b}$ i $r_{a \rightarrow c}$ przy różnych możliwych odległościach $r_{b \rightarrow c}$ (podanych w nawiasach kwadratowych w odpowiednich komórkach) dla hipersześcianu o wymiarze $d = 5$. a , b i c są różnymi stanami. Błąd wyznaczenia $\Delta\tau_M = 0,001$.

	$r_{a \rightarrow c} = 1$	$r_{a \rightarrow c} = 2$	$r_{a \rightarrow c} = 3$	$r_{a \rightarrow c} = 4$	$r_{a \rightarrow c} = 5$
$r_{a \rightarrow b} = 1$	[2] : 12, 250	[1] : 18, 750 [3] : 14, 167	[2] : 16, 834 [4] : 14, 750	[3] : 16, 249 [5] : 15, 000	[4] : 16, 000
$r_{a \rightarrow b} = 2$	[1] : 18, 748 [3] : 14, 168	[2] : 18, 750 [4] : 16, 666	[1] : 23, 333 [3] : 18, 750 [5] : 17, 500	[2] : 20, 834 [4] : 18, 751	[3] : 20, 000
$r_{a \rightarrow b} = 3$	[2] : 16, 833 [4] : 14, 750	[1] : 23, 334 [3] : 18, 750 [5] : 17, 500	[2] : 21, 416 [4] : 19, 333	[1] : 25, 418 [3] : 20, 833	[2] : 22, 666
$r_{a \rightarrow b} = 4$	[3] : 16, 249 [5] : 15, 000	[2] : 20, 834 [4] : 18, 750	[1] : 25, 417 [3] : 20, 834	[2] : 22, 917	[1] : 26, 666
$r_{a \rightarrow b} = 5$	[4] : 16, 000	[3] : 20, 001	[2] : 22, 667	[1] : 26, 668	—

Założmy, że stan początkowy jest zawsze równy 0. Możliwe odległości pomiędzy stanami na hipersześcianie pięciowymiarowym wynoszą od 1 do 5, a w tabeli 4.4 wypisano możliwe stany końcowe w danych odległościach od stanu 0.

Warto zaznaczyć, że celowo odległości przedstawiane są w postaci $r_{a \rightarrow b}$, tzn. ze strzałką skierowaną w jedną stronę. W ogólności używana w całej pracy odległość chemiczna nie jest odległością w sensie matematycznym, gdyż może zależeć od kierunku, np. gdy wprowadzimy krawędzie umożliwiające przejście tylko w jednym kierunku, albo w obu ale z różnymi wagami. Ponadto już dla sieci typu trójkąta Sierpińskiego kierunek będzie miał wpływ na średni czas pierwszego przejścia, czego nie obserwuje się jeszcze na sieci typu hipersześcianu.

Dokonajmy analizy tabeli 4.4 pod kątem dodatkowej odległości $r_{b \rightarrow c}$. Wyniki obliczeń $\tau_M(a \rightarrow \{b, c\})$ dla wszystkich możliwych kombinacji odległości $r_{a \rightarrow b}$ i $r_{a \rightarrow c}$ wraz z możliwymi odległościami $r_{b \rightarrow c}$ przedstawiono w tabeli 4.5. Warto porównać szczególnie przypadki $r_{a \rightarrow b} = r_{a \rightarrow c}$ (stany b i c wybierano różne choć znajdowały się w tej samej odległości od stanu początkowego) z danymi w tabeli 4.3. Czasy typu $\tau_M(a \rightarrow \{b, c\})$ są znacząco mniejsze od czasów typu $\tau_M(a \rightarrow \{b, b\})$ tożsamy z $\tau_M(a \rightarrow b)$. Wyjątkiem będzie przypadek $r_{a \rightarrow b} = r_{a \rightarrow c} = 5$, gdyż na hipersześcianie pięciowymiarowym nie ma dwóch innych stanów w odległości 5 od stanu początkowego, stąd taki czas pierwszego przejścia jest równy czasowi dla $r_{a \rightarrow b} = 5$. Celowo pokazano, że tabela 4.5 jest symetryczna, co dodatkowo udowadnia dobre oszacowanie błędu oraz fakt, że $\tau_M(a \rightarrow \{b, c\}) = \tau_M(a \rightarrow \{c, b\})$.

Wartości średnich czasów pierwszego przejścia umieszczone w tabelach 4.3 i 4.5 sugerują, że powinno istnieć analityczne wyrażenie na średni czas pierwszego przejścia w zależności od odległości pomiędzy stanami i wymiaru hipersześcianu, dane przez proste ułamki. Niestety w dostępnej mi literaturze nie udało się takiego wyrażenia odnaleźć, ani sam nie potrafiłem do tej pory go wyprowadzić.

Warto wspomnieć też o rzeczywistych czasach obliczeń. Przyjęto liczbę węzłów równą 10^9 i zmierzono rzeczywisty czas obliczeń na komputerze z procesorem Intel® Core™2 Duo E6550 2.33 GHz. Dla przypadku $\tau_M(0 \rightarrow 1)$ było to 1512 sekund (można porównać z 3580 dla tego samego problemu i z tym samym systemem operacyjnym tylko z innym procesorem – tabela 2.1), a dla przypadku $\tau_M(0 \rightarrow \{1, 2\})$ (oba stany w odległości 1 od stanu początkowego) było to 618 sekund. Zatem przy tej liczbie węzłów jeden krok całego zespołu statystycznego zajmuje około 50 sekund. Wartość tą można skutecznie zmniejszyć stosując metody przetwarzania równoległego, o których była mowa w rozdziale dotyczącym metodologii symulacji numerycznych.

4.2.2 Jeden proces dostarczający i jeden odbierający energię swobodną

Najprostszy rozważany przypadek zawiera jeden proces dostarczający i jeden odbierający energię swobodną. Dla tego typu diagramu kinetycznego wprowadzono już w tym rozdziale wyrażenia analityczne na strumienie, stopnie poślizgu i stopień sprzężenia. Przypadek ten posłuży do przedyskutowania roli parametrów z tabeli 4.1, tzn. v_1 , v_2 , a_1 , a_2 oraz parametrów o i r związanych ze wzajemnym rozmieszczeniem stanów będących bramami. Celem jest uzyskanie możliwie największego stopnia sprzężenia ϵ rozumianego jako iloraz strumienia procesu odbierającego energię swobodną i strumienia procesu dostarczającego energię swobodną. Po wybraniu możliwie najlepszego zestawu parametrów zostaną przeprowadzone symulacje dające w wyniku zależności strumienia J_2 od siły a_2 , zależność stopnia sprzężenia od siły a_2 oraz zależność siły blokującej a_2^{st} od a_1 .

Rola parametrów o i r

Wyniki 24 symulacji dla różnych wartości parametrów o i r przedstawiono w tabeli 4.6. Dla o mniejszego od maksymalnego na hipersześcianie o danym wymiarze ważne jest dodatkowo ustalenie odległości po przekątnej pomiędzy stanami $1'$ i $2'$ oraz $1''$ i $2''$ (rysunek 4.4). W przypadku kilku możliwych geometrii dla danej pary o i r różniących się odległościami po przekątnych, wybierano taką geometrię aby obie odległości po przekątnej były takie same i jak największe, co też dawało dla danej pary o i r największe wartości stopnia sprzężenia $\epsilon(0)$.

Do dalszych symulacji wybrano konfigurację bram dla $o = 5$ i $r = 1$, która daje jedną z najwyższych wartości stopnia sprzężenia. Wybór ten był uzasadniony jednoznacznością geometrii – odległość po przekątnej pomiędzy $1'$ i $2'$ oraz $1''$ i $2''$ może wynosić tylko 4 niezależnie od wyboru bram, zatem uniknąłem konieczności definiowania dodatkowych parametrów. Będzie to korzystne przy rozszerzaniu modelu na więcej procesów dostarczających i odbierających energię swobodną.

Warto zwrócić uwagę na symetrię w tabeli 4.6, która wynika z faktu, że do otrzymania geometrii z inną parą o i r może prowadzić także zamiana bram $2''$ z $2'$ (dla widocznej symetrii ważne jest zachowanie maksymalnej możliwej odległości po przekątnej). W takim przypadku stopień sprzężenia zmienia znak na przeciwny (przejścia „do przodu” dla strumienia J_2 liczone są w przeciwną stronę). Symetria w tabeli 4.6 dotyczy następujących geometrii $\{o, r\}$: $\{5, 1\}$ i $\{5, 4\}$, $\{5, 2\}$ i $\{5, 3\}$, $\{4, 1\}$ i $\{4, 5\}$, $\{4, 2\}$ i $\{4, 4\}$, $\{3, 2\}$ i $\{3, 5\}$, $\{3, 3\}$ i $\{3, 4\}$, $\{2, 3\}$ i $\{2, 5\}$, $\{1, 4\}$ i $\{1, 5\}$. Pary te różnią się tym, że w wyniku zamiany bramy $2''$ z $2'$, odległość po przekątnej została zamieniona z odległością r . Dla innego wyboru odległości po przekątnych pojawiłyby się inne symetryczne konfiguracje.

Tabela 4.6. Wpływ parametrów o i r na stopień sprzężenia $\epsilon(0)$ przy założeniu maksymalnej możliwej odległości po przekątnej między stanami $1'$ i $2'$ oraz $1''$ i $2''$. Parametry symulacji: $N_w = 10^2$, $N_k = 10^7$, $a_1 = 10$, $a_2 = 0$, $v_1/p = 50$, $v_2/p = 30$, $p = 1,82 \cdot 10^{-2}$, $v_1 \exp(-a_1) = 4,13 \cdot 10^{-5}$, $v_2 \exp(-a_2) = 0,545$. Błąd wyznaczenia $\Delta\epsilon(0) = 0,001$.

	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$
$o = 1$	0,194	0,080	0,045	0,031	-0,030
$o = 2$	0,228	0,105	0,061	-0,001	-0,062
$o = 3$	0,249	0,121	0,036	-0,034	-0,121
$o = 4$	0,263	0,093	-0,001	-0,094	-0,263
$o = 5$	0,235	0,059	-0,059	-0,235	-

Warto zauważyć, że z samych równań (4.12) i (4.13) można wyciągnąć pewne wnioski na temat geometrii zapewniającej najwyższą wartość stopnia sprzężenia w obszarze przetwarzania energii swobodnej.

Rozważmy przypadek, gdy $W_2(0)$ jest dodatnie, a co za tym idzie wartość $\epsilon(0)$ zawiera się pomiędzy 0 a 1. Z równania (4.12) można wyciągnąć wniosek, że aby $\epsilon(0)$ było jak największe, to $W_2(0)$ musi być jak najmniejsze. Gdyby $W_2(0)$ było zerem, to stopień sprzężenia $\epsilon(0)$ wynosiłby 1. Przyjrzyjmy się równaniu (4.13) w celu odpowiedzenia na pytanie kiedy $W_2(0)$ jest najmniejsze. Będzie tak gdy wyrażenie w liczniku będzie jak najmniejsze, a to w mianowniku jak największe. Dokonajmy zatem analizy wyrażenia w mianowniku. Po drodze skorzystamy z równania (4.17). Zarówno czas $\tau_M(2' \leftrightarrow \{1', 2''\})$ jak i czas $\tau_M(2' \leftrightarrow \{1'', 2''\})$ zawierają ten sam składnik $-\tau_M(1' \rightarrow \{2', 2''\}) = -\tau_M(1'' \rightarrow \{2', 2''\})$ (własność ta wynika z faktu, że zamiana wszystkich bram prim na bis i na odwrót nie wpływa na średni czas pierwszego przejścia, co jednak nie musi być spełnione dla wszystkich sieci), a w związku z tym, że w mianowniku występuje różnica czasów, to oba składniki ulegną redukcji. Zatem zależy nam aby suma

$$\tau_M(2' \rightarrow \{1', 2''\}) + \tau_M(1' \rightarrow 2')$$

była jak największa, a suma

$$\tau_M(2' \rightarrow \{1'', 2''\}) + \tau_M(1'' \rightarrow 2')$$

jak najmniejsza. Posłużymy się dodatkowo średnimi czasami pierwszego przejścia zebranymi w tabelach 4.3 i 4.5, pomocny okaże się też rysunek 4.4. $\tau_M(1' \rightarrow 2')$ będzie największe gdy odległość $r_{1' \rightarrow 2'}$ będzie wynosiła 5 (przekątna na rysunku 4.4). $\tau_M(1'' \rightarrow 2')$

będzie jak najmniejsze gdy odległość $r_{1'' \rightarrow 2'}$ będzie wynosiła 1. Następnie $\tau_M(2' \rightarrow \{1', 2''\})$ ma być jak największe, a wiemy już że $r_{1' \rightarrow 2'} = r_{2' \rightarrow 1'} = 5$ (należy pamiętać, że już dla sieci typu trójkąta Sierpińskiego zamiana stanu początkowego z końcowym może spowodować zmianę wartości średniego czasu pierwszego przejścia). Zatem z tabeli 4.5 można odczytać, że najdłuższy czas odpowiadałby $r_{2' \rightarrow 2''} = 5$, jednakże tylko jeden stan może być w maksymalnej odległości od $2'$ (różnica na wszystkich bitach) i jest nim już $1'$. Stąd też musimy zadowolić się $r_{2' \rightarrow 2''} = o = 4$. W ten sposób dobrano wszystkie parametry, gdyż odległość $r_{2' \rightarrow 2''}$ występuje też w drugim członie i mimo, że w nim lepiej gdyby ta odległość wynosiła 1, to zmniejszenie wartości drugiego członu byłoby niewielkie, a człon pierwszy zmniejszyłby się drastycznie, co byłoby niekorzystne. Ze względu na symetrię obrotu o 180 stopni, $r_{1' \rightarrow 2'} = r_{1'' \rightarrow 2''}$, $r_{2' \rightarrow 2''} = r_{1' \rightarrow 1''} = o$ i $r_{1'' \rightarrow 2'} = r_{1' \rightarrow 2''} = r$.

Rozważania te pozostają w zgodzie z wynikami symulacji przedstawionymi w tabeli 4.6, gdzie największą wartość $\epsilon(0)$ otrzymano właśnie dla $o = 4$, $r = 1$ i odległości po przekątnej maksymalnej, tu równej 5. Warto przypomnieć, że występujące w wyrażeniu na stopień poślizgu średnie czasy pierwszego przejścia zostały uzyskane przy pomocy symulacji komputerowych odrębnych od tych, w których otrzymano dane przedstawione w tabeli 4.6. Bez tych czasów przeprowadzona tu dyskusja nie byłaby możliwa.

Rola parametrów v_1 i v_2

Wpływ prawdopodobieństwa przejść zewnętrznych v_1 na siłę blokującą a_2^{st} oraz wpływ prawdopodobieństwa przejść zewnętrznych v_2 na stopień sprzężenia $\epsilon(0)$ obrazują wyniki symulacji przedstawione w tabeli 4.7. Zwiększanie prawdopodobieństwa v_2 powoduje wzrost stopnia sprzężenia. Tego typu efektu nie obserwuje się dla zmiany v_1 przy ustalonym v_2 . Ponadto im większe v_1 tym bardziej ujemna jest siła blokująca. Warto pamiętać, że na wielkość obszaru przetwarzania energii swobodnej ma wpływ zarówno wartość a_2^{st} jak i $\epsilon(0)$. Do dalszych symulacji wybrano $v_1/p = 50$ i $v_2/p = 30$.

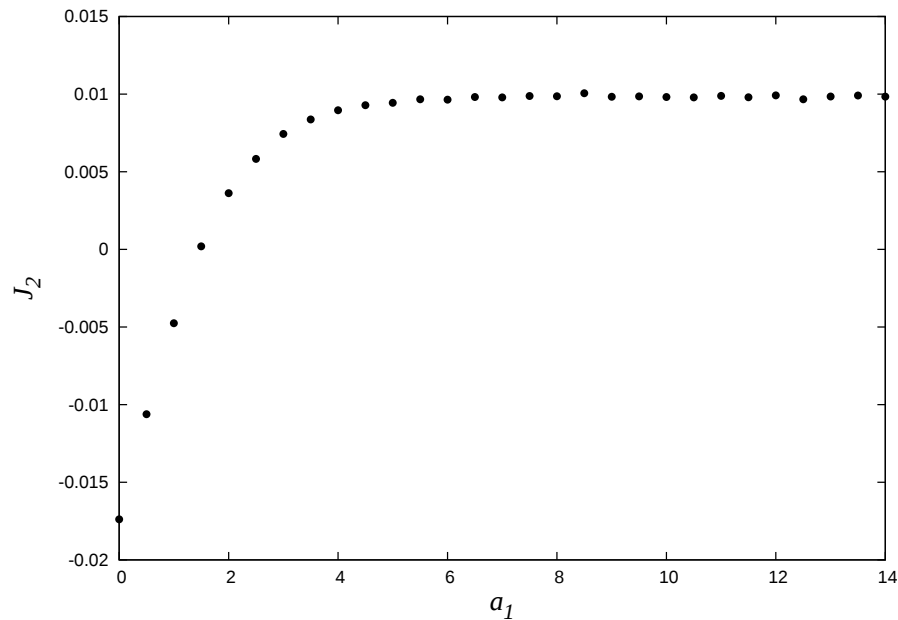
Rola parametru a_1 i a_2

Oczywistą rolą parametrów a_1 i a_2 jest kontrola odwracalności procesów dostarczających i odbierających energię swobodną. Duży wpływ ma znak sił napędzających – eksponens argumentu dodatniego jest większy od 1, a ujemnego mniejszy od 1. Dodatkowo prawdopodobieństwa v_1 czy v_2 muszą zawierać się między 0 a 1 – mnożąc przez eksponens musimy nadal otrzymać wielkość pomiędzy 0 a 1, stąd znów ograniczenie na siły.

Dalej założymy, że układ znajduje się daleko od równowagi, a zatem $a_1 \rightarrow \infty$. Należy sprawdzić jaka wartość a_1 oznacza, że siła ta jest wystarczająco duża, tzn. występuje nasycenie strumienia $J_2(a_1)$. Przeprowadzono symulację, której wyniki przedstawiono na

Tabela 4.7. Wpływ prawdopodobieństwa przejścia zewnętrznego na jednostkę czasu v_1/p na siłę blokującą a_2^{st} przy ustalonym $v_2/p = 30$ oraz wpływ prawdopodobieństwa przejścia zewnętrznego na jednostkę czasu v_2/p na stopień sprzężenia $\epsilon(0)$ przy ustalonym $v_1/p = 50$. Wartości a_2^{st} zostały wyznaczone z błędem 0,01, a wartości $\epsilon(0)$ z błędem 0,001.

$v_2/p = 30$		$v_1/p = 50$	
v_1/p	a_2^{st}	v_2/p	$\epsilon(0)$
10	-0,37	10	0,211
30	-0,45	30	0,235
50	-0,48	50	0,241



Rysunek 4.8. Zależność strumienia J_2 od siły napędzającej a_1 . Widoczne są drobne fluktuacje występujące w obszarze nasycenia.

rysunku 4.8 (stopień sprzężenia ϵ będzie zachowywał się podobnie gdyż J_1 praktycznie nie zmienia się ze względu na i tak o wiele większe prawdopodobieństwo biegu procesu „do przodu”). Widać, że wartość $a_1 = 10$ może z powodzeniem przybliżać przejście do nieskończoności.

Zależność strumień-siła $J_2(a_2)$ dla procesu odbierającego energię swobodną

Zatem wybrano już wartości wszystkich potrzebnych parametrów: $v_1/p = 50$, $v_2/p = 30$, $r = 1$, $o = 5$, $a_1 = 10$ (wartość stała chyba, że będziemy wyznaczać zależność pewnych wielkości od a_1). Parametr a_2 może zmieniać się w dowolnym zakresie od wartości ujemnych do dodatnich. Wartość jest ograniczana koniecznością uzgodnienia wszystkich prawdopodobieństw na sieci. Jednakże najbardziej interesujący jest obszar przetwarzania energii swobodnej.

Wykonano symulację, której wynik zaznaczono kropkami na rysunku 4.9. Założyliśmy, że siła a_2 zmienia się w zakresie od -6 do 6 z krokiem $0,5$. Do wyznaczania wartości p należało użyć różnych wyrażeń, a następnie wykres „zszyć” z fragmentów dla różnych przedziałów wartości a_2 (dla wartości siły $a_2 < -0,5$ użyto $p = 0,0001$ – wartość ta mogłaby służyć do stworzenia całego wykresu jednakże by pokazać efekt „zszycia” dla $a_2 \geq -0,5$ użyto wyrażenia danego równaniem (4.18)). Przypomnijmy, że im mniejsze p , tym błędy wyznaczania wielkości typu siły blokującej a_2^{st} będą większe przy zachowanym iloczynie liczby wędrówców i kroków każdego z nich. Stąd też należy starać się wybierać największe możliwe p aby uniknąć konieczności zmniejszania błędów przez dodawanie dodatkowych kroków lub wędrówców, bo to już przekłada się na zwiększenie rzeczywistego czasu obliczeń. W obszarze bardzo ujemnych wartości siły a_2 występują duże fluktuacje, które są wynikiem bardzo dużych prawdopodobieństw czekania.

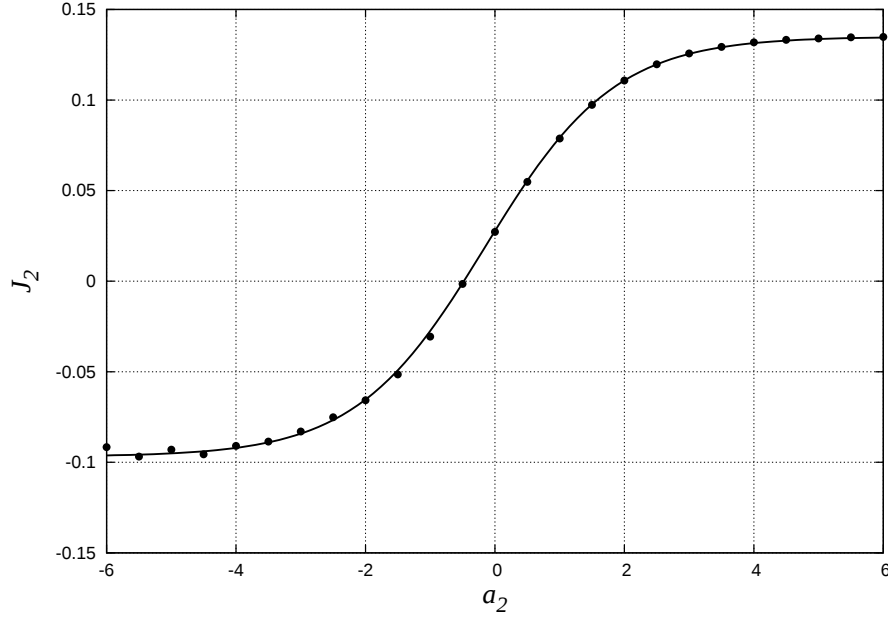
Mimo, że obszar przetwarzania energii swobodnej jest niewielki (od $a_2 = 0$ do $a_2 = a_2^{st} = -0,48$), to symulowana maszyna molekularna działa. Dla dodatnich i ujemnych sił wartość strumienia dąży asymptotycznie do dwóch różnych wartości. W naszym przypadku J_{+2} to około $0,135$, a J_{-2} to około $-0,097$. Dla czasu τ_2 niezależnego od siły A_2 , $J_{02}^{-1} = 0$ [C]. Wstawiając te wartości do równania (4.3) otrzymujemy

$$J_2(a_2) = \frac{1 - 0,618 e^{-a_2}}{7,407 + 6,379 e^{-a_2}}. \quad (4.25)$$

Zależność tą zaznaczono linią ciągłą na rysunku 4.9. Warto jednak pamiętać, że wyrażenie analityczne dostarcza tylko części informacji o kształcie krzywej, a reszta niezbędnych wartości (J_{+2} , J_{-2} i a_2^{st}) pochodzi z dopasowania do wyników. Jak widać zgodność wyrażenia dopasowanego przy użyciu trzech parametrów jest bardzo dobra.

4.2.3 Eksperymentalne potwierdzenie teorii

Wyrażenia (4.6) i (4.7) na stopień sprzężenia i siłę blokującą zależą od stopni poślizgu W_1 i W_2 , a te z kolei od średnich czasów pierwszego przejścia. Niestety te ostatnie zależą od konkretnej sieci i nie są wielkościami, które można łatwo określić. Czasy te musimy wyznaczyć w osobnym eksperymencie numerycznym.



Rysunek 4.9. Zależność strumienia J_2 od sprzężonej z nim siły a_2 dla przypadku jednego procesu odbierającego i jednego dostarczającego energię swobodną. Wyniki symulacji zaznaczono kropkami, natomiast linia ciągła dana jest równaniem (4.25).

Dla $o = 5$ i $r = 1$ dokonano następującego wyboru stanów będących bramami: $1' = 31, 1'' = 0, 2' = 4, 2'' = 27$. We wszystkich obliczeniach w tym i następnym podrozdziale posługujemy się średnimi czasami pierwszego przejścia umieszczonymi w tabelach 4.3 i 4.5.

Wyznaczanie stopnia poślizgu $W_1(a_1)$ (równanie (4.8))

Potrzebne średnie czasy pierwszego przejścia (korzystamy z równania (4.17)) to:

$$\begin{aligned}
 \tau_M(1'' \leftrightarrow \{1', 2'\}) &= \tau_M(1'' \rightarrow \{1', 2'\}) + \tau_M(1' \rightarrow 1'') - \tau_M(1' \rightarrow \{1'', 2'\}) \\
 &= \tau_M(0 \rightarrow \{31, 4\}) + \tau_M(31 \rightarrow 0) - \tau_M(31 \rightarrow \{0, 4\}) \\
 &= 16,000 + 42,667 - 26,668 = 31,999 \pm 0,003. \quad (4.26)
 \end{aligned}$$

$$\begin{aligned}
 \tau_M(1' \leftrightarrow \{1'', 2'\}) &= \tau_M(1' \rightarrow \{1'', 2'\}) + \tau_M(1'' \rightarrow 1') - \tau_M(1'' \rightarrow \{1', 2'\}) \\
 &= \tau_M(31 \rightarrow \{0, 4\}) + \tau_M(0 \rightarrow 31) - \tau_M(0 \rightarrow \{31, 4\}) \\
 &= 26,668 + 42,667 - 16,000 = 53,335 \pm 0,003. \quad (4.27)
 \end{aligned}$$

$$\begin{aligned}
 \tau_M(1' \leftrightarrow \{1'', 2''\}) &= \tau_M(1' \rightarrow \{1'', 2''\}) + \tau_M(1'' \rightarrow 1') - \tau_M(1'' \rightarrow \{1', 2''\}) \\
 &= \tau_M(31 \rightarrow \{0, 27\}) + \tau_M(0 \rightarrow 31) - \tau_M(0 \rightarrow \{31, 27\}) \\
 &= 16,000 + 42,667 - 26,668 = 31,999 \pm 0,003. \quad (4.28)
 \end{aligned}$$

Odwrotność czasu τ_1 jest iloczynem prawdopodobieństwa przejścia zewnętrznego na jednostkę czasu oraz prawdopodobieństwa równowagowego obsadzenia pojedynczego stanu (standardowa teoria stanu przejściowego) [C]:

$$\tau_1^{-1} = \frac{v_1}{p} p^{\text{eq}}. \quad (4.29)$$

Umieszczone w tabelach 4.3 i 4.5 średnie czasy pierwszego przejścia powinny być pomnożone przez odwrotność przyjętej jednostki czasu $p_M = 0,2$ (prawdopodobieństwo przejścia na pięciowymiarowym hipersześcianie). Podobny efekt można uzyskać mnożąc wyrażenie (4.29) przez p_M . Stąd dla pięciowymiarowego hipersześcianu i $v_1/p = 50$ otrzymujemy

$$\tau_1 = \frac{1}{50} \frac{1}{0,2} 32 = 3,200. \quad (4.30)$$

Jak widać przyczynek τ_1 jest znacząco mniejszy od τ_M ale nie jest zaniedbywalny, tzn. proces dostarczający energię swobodną kontrolowany jest zarówno przez wewnętrzną jak i zewnętrzną dynamikę (standardowa teoria stanu przejściowego zupełnie pomija dynamikę wewnętrzną).

W granicy siły napędzającej $a_1 \rightarrow \infty$ (zweryfikowano wcześniej, że wystarczy przyjąć $a_1 = 10$), równanie (4.7) redukuje się do (4.10), a równanie (4.8) do (4.11). Podstawiając wartości średnich czasów pierwszego przejścia oraz τ_1 otrzymujemy

$$W_1(\infty) = 1,650, \quad a_2^{\text{st}}(\infty) = \beta A_2^{\text{st}}(\infty) = -0,474 \quad (4.31)$$

z błędami wyznaczenia na miejscach poza podaną powyżej liczbą cyfr znaczących.

Gdyby przyjąć, że $v_1/p \rightarrow \infty$, wtedy τ_1 stanowić będzie znikomą poprawkę do W_1 , a proces kontrolowany będzie całkowicie przez dynamikę wewnętrzną. W takim przypadku

$$W_1(\infty) = 1,500, \quad a_2^{\text{st}}(\infty) = \beta A_2^{\text{st}}(\infty) = -0,511. \quad (4.32)$$

Wyznaczanie stopnia poślizgu $W_2(a_2)$ (równanie (4.9))

Potrzebne tym razem średnie czasy pierwszego przejścia (znów korzystamy z równania (4.17)) to:

$$\begin{aligned} \tau_M(2'' \leftrightarrow \{1', 2'\}) &= \tau_M(2'' \rightarrow \{1', 2'\}) + \tau_M(1' \rightarrow 2'') - \tau_M(1' \rightarrow \{2'', 2'\}) \\ &= \tau_M(27 \rightarrow \{31, 4\}) + \tau_M(31 \rightarrow 27) - \tau_M(31 \rightarrow \{27, 4\}) \\ &= 16,000 + 31,000 - 15,000 = 32,000 \pm 0,003. \end{aligned} \quad (4.33)$$

$$\begin{aligned} \tau_M(2' \leftrightarrow \{1'', 2''\}) &= \tau_M(2' \rightarrow \{1'', 2''\}) + \tau_M(1'' \rightarrow 2') - \tau_M(1'' \rightarrow \{2', 2''\}) \\ &= \tau_M(4 \rightarrow \{0, 27\}) + \tau_M(0 \rightarrow 4) - \tau_M(0 \rightarrow \{4, 27\}) \\ &= 16,000 + 31,000 - 15,000 = 32,000 \pm 0,003. \end{aligned} \quad (4.34)$$

$$\begin{aligned}
\tau_M(2' \leftrightarrow \{1', 2''\}) &= \tau_M(2' \rightarrow \{1', 2''\}) + \tau_M(1' \rightarrow 2') - \tau_M(1' \rightarrow \{2', 2''\}) \\
&= \tau_M(4 \rightarrow \{31, 27\}) + \tau_M(31 \rightarrow 4) - \tau_M(31 \rightarrow \{4, 27\}) \\
&= 26,668 + 41,666 - 15,000 = 53,334 \pm 0,003. \quad (4.35)
\end{aligned}$$

Podobnie jak w przypadku czasu τ_1 , odwrotność czasu τ_2 jest iloczynem prawdopodobieństwa przejścia zewnętrznego na jednostkę czasu oraz prawdopodobieństwa równowagowego obsadzenia pojedynczego stanu

$$\tau_2^{-2} = \frac{v_2}{p} p^{\text{eq}}, \quad (4.36)$$

stąd dla pięciowymiarowego hipersześcianu i $v_2/p = 30$, po uwzględnieniu p_M , otrzymujemy

$$\tau_2 = \frac{1}{30} \frac{1}{0,2} 32 = 5,333. \quad (4.37)$$

Również w tym przypadku przyczynę τ_2 jest znacząco mniejszy od τ_M ale nie jest zaniedbywalny, tzn. proces odbierający energię swobodną kontrolowany jest zarówno przez wewnętrzną jak i zewnętrzną dynamikę.

W granicy siły napędzającej $a_1 \rightarrow \infty$ (zweryfikowano wcześniej, że wystarczy przyjąć $a_1 = 10$) i dla $a_2 = 0$, równanie (4.6) zredukowało się do (4.12), a równanie (4.9) do (4.13). Podstawiając wartości średnich czasów pierwszego przejścia oraz τ_2 otrzymujemy

$$W_2(0) = 3,250, \quad \epsilon(0) = 0,235 \quad (4.38)$$

z błędami wyznaczenia na miejscach poza podaną powyżej liczbą cyfr znaczących.

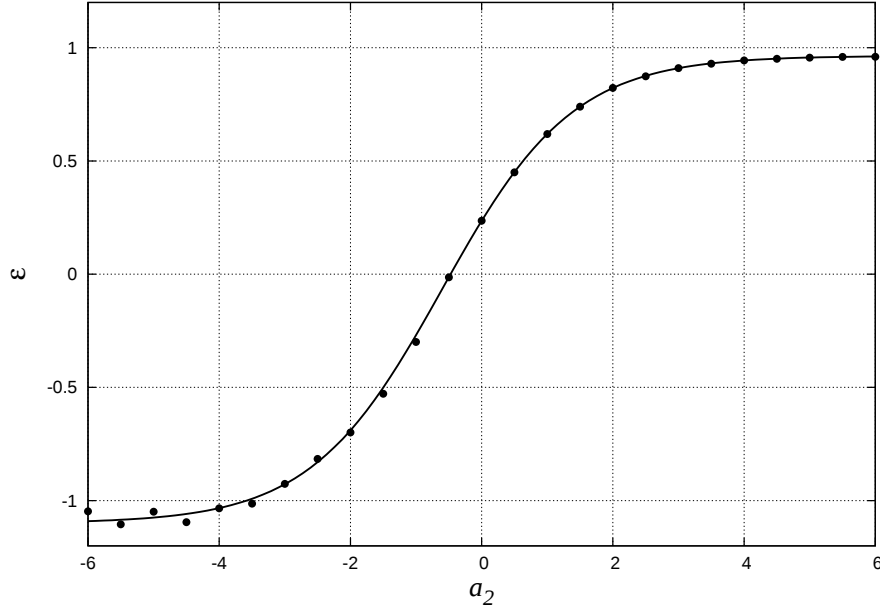
Gdyby przyjąć, że $v_2/p \rightarrow \infty$, wtedy τ_2 stanowiłby znikomą poprawkę do W_2 , a proces kontrolowany byłby całkowicie przez dynamikę wewnętrzną. W takim przypadku

$$W_2(0) = 3,000, \quad \epsilon(0) = 0,250. \quad (4.39)$$

Wcześniej wspomniano, że mianowniki w wyrażeniach na $W_1(A_1)$ (równanie (4.8)) i $W_2(A_2)$ (równanie (4.9)) są identyczne. Oprócz ścisłego dowodu (praca [C]) można ten fakt udowodnić także numerycznie, gdyż mianownik w wyrażeniu na $W_1(\infty)$ jest równy 21,336, a ten w $W_2(0)$ jest równy 21,334, co jest zgodne w zakresie błędu pomiaru.

Stopień sprzężenia ϵ w zależności od a_2

W symulacji przedstawionej na rysunku 4.9 zapisywano też wartości strumienia J_1 i obliczano stopień sprzężenia. Zależność stopnia sprzężenia ϵ od a_2 przedstawiono na rysunku 4.10. Wyniki symulacji zaznaczono kropkami, a linia ciągła dana jest równaniem (4.6). Warto zauważyć, że w równaniu (4.6) występują stopnie poślizgu $W_1(a_1)$ oraz $W_2(a_2)$, które składają się z odpowiednich średnich czasów pierwszego przejścia wyznaczonych



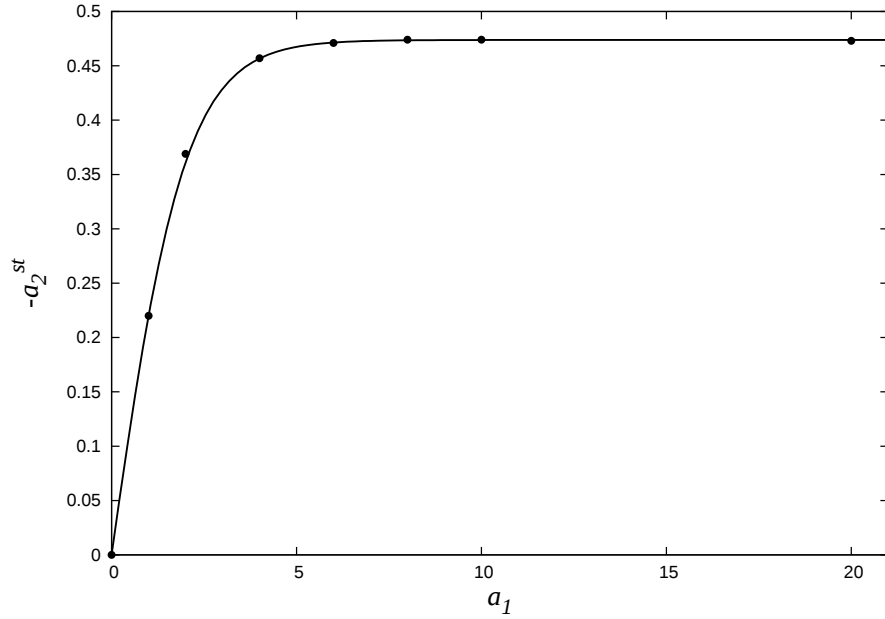
Rysunek 4.10. Zależność stopnia sprzężenia ϵ od siły a_2 dla przypadku jednego procesu dostarczającego i jednego odbierającego energię swobodną. Wyniki symulacji zaznaczono kropkami, natomiast linia ciągła dana jest równaniem (4.6).

tak jak opisano wcześniej. Zgodność wyników symulacji z wyrażeniem analitycznym jest bardzo dobra. W obszarze bardzo ujemnych wartości siły a_2 występują duże fluktuacje, które są wynikiem bardzo dużych prawdopodobieństw czekania. Strumień J_1 zmieniał się w zakresie od 0,087 dla $a_2 = -6$ do 0,140 dla $a_2 = 6$, a zależność stopnia sprzężenia od siły a_2 jest bardzo podobna do zależności przedstawionej na rysunku 4.9.

Podkreślmy, że rysunek 4.10 przedstawia dokładny wynik teoretyczny na tle eksperymentu numerycznego i nie jest dopasowaniem teorii do doświadczenia.

Zależność siły blokującej a_2^{st} od a_1

Aby dokładnie wyznaczyć wartość siły a_2 , dla której strumień $J_2 = 0$, wyznaczono pewną liczbę wartości $J_2(a_2)$ w okolicy $J_2 = 0$. W tym obszarze zależność strumień-siła jest liniowa, stąd przy użyciu GNUPLOT 4.2 [27] dopasowano prostą, której współczynniki pozwoliły na wyznaczenie wartości a_2^{st} . Powyższą procedurę wykonano dla kilku wartości sił a_1 . Na rysunku 4.11 wyniki symulacji zaznaczono kropkami, natomiast linia ciągła dana jest równaniem (4.7). Zgodność wyników symulacji z wyrażeniem analitycznym jest bardzo dobra. Warto zauważyć, że w równaniu (4.7) występuje stopień poślizgu $W_1(A_1)$, stąd podobnie jak w przypadku analitycznego wyrażenia na stopień sprzężenia, wyrażenie to wymaga obliczenia wartości średnich czasów pierwszego przejścia dla konkretnej sieci. Przy długości ścieżki wynoszącej 10^9 i p liczonym z równania typu (4.18)



Rysunek 4.11. Zależność siły blokującej a_2^{st} od siły a_1 dla przypadku jednego procesu dostarczającego i jednego procesu odbierającego energię swobodną. Wyniki symulacji zaznaczono kropkami, natomiast linia ciągła dana jest równaniem (4.7).

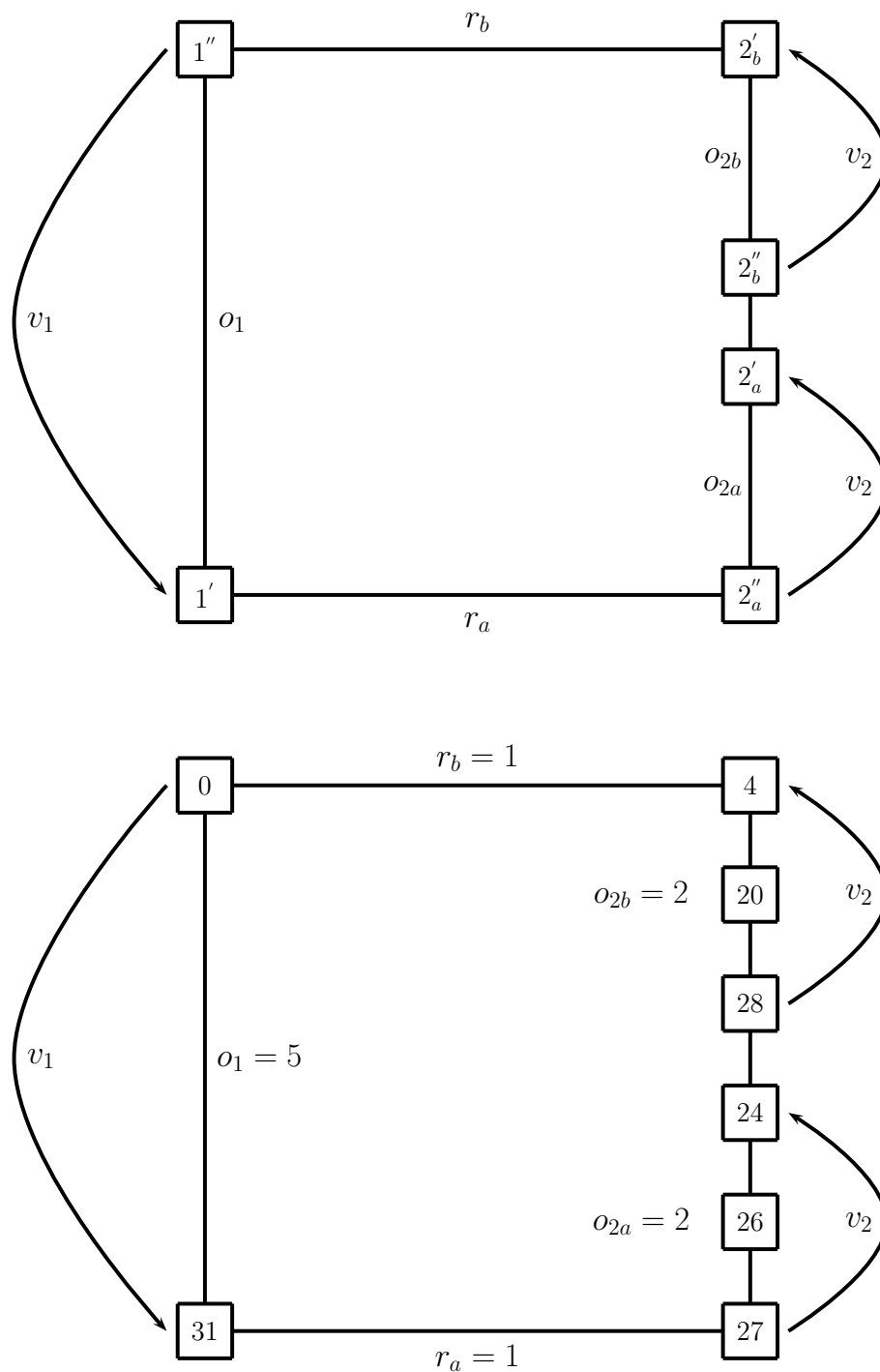
błąd wyznaczenia a_2^{st} wynosił około 0,01 (przydatne jest równanie (2.6)). Jak wynika z wykresu, założenie $a_1 \rightarrow \infty$ przy $a_1 = 10$ jest w pełni uzasadnione.

Podkreślimy, że rysunek 4.11 przedstawia dokładny wynik teoretyczny na tle eksperymentu numerycznego i nie jest dopasowaniem teorii do doświadczenia.

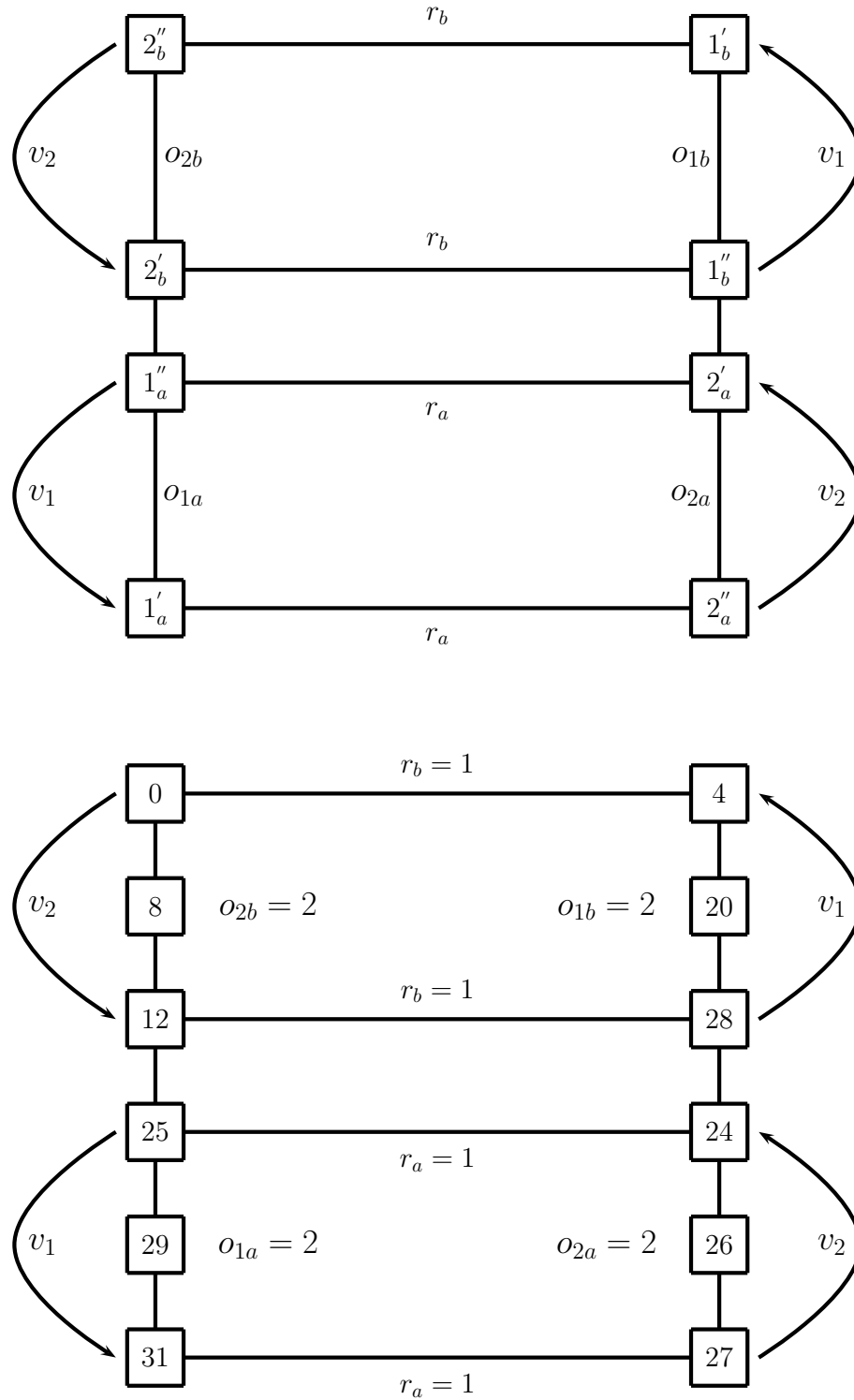
4.2.4 Rozszerzenie modelu na większą liczbę procesów dostarczających i odbierających energię swobodną

Dotychczasowe symulacje dla pojedynczych procesów dostarczających i odbierających energię swobodną można było porównać z wyrażeniami teoretycznymi na strumienie, stopnie poślizgu, stopień sprzężenia i siłę blokującą, co pokazuje jakość narzędzia badawczego. Takich wyrażeń analitycznych niestety nie ma do tej pory dla schematów kinetycznych z wieloma procesami dostarczającymi i odbierającymi energię swobodną. Stąd też symulacje są jedynym źródłem danych dla wyznaczanych zależności strumień-siła i stopnia sprzężenia w zależności od siły, a poprawność metody, która została potwierdzona we wcześniejszych rozważaniach jest bardzo ważna.

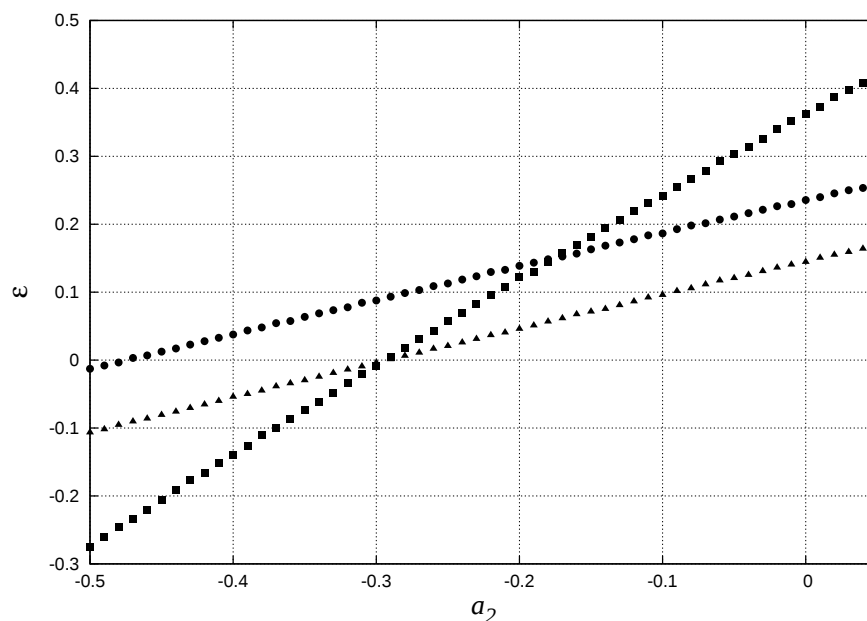
Na rysunku 4.12 przedstawiono wybraną realizację modelu z jednym procesem dostarczającym energię swobodną i dwoma procesami odbierającymi ($N_{P1} = 1, N_{P2} = 2$), a na rysunku 4.13 przedstawiono wybraną realizację modelu z dwoma procesami dostarczającymi i dwoma procesami odbierającymi energię swobodną ($N_{P1} = 2, N_{P2} = 2$). W



Rysunek 4.12. Wybrany sposób rozszerzenia geometrii z rysunku 4.4 na model z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Zaznaczono charakterystyczne odległości oraz ustalony kierunek przejść zewnętrznych „do przodu”. Na górze ogólny schemat, na dole używany przykład realizacji z numerami stanów i wartościami odległości. Nie zostały zaznaczone przejścia zewnętrzne „do tyłu” określone przez czynniki $\exp(-a_1)$ oraz $\exp(-a_2)$. Odległość $r_{2'_a \rightarrow 2''_b} = r_{2''_b \rightarrow 2'_a} = 1$.



Rysunek 4.13. Wybrany sposób rozszerzenia geometrii z rysunku 4.4 na model z dwoma procesami dostarczającymi i dwoma odbierającymi energię swobodną. Zaznaczono charakterystyczne odległości oraz ustalony kierunek przejść zewnętrznych „do przodu”. Na górze ogólny schemat, na dole używany przykład realizacji z numerami stanów i wartościami odległości. Nie zostały zaznaczone przejścia zewnętrzne „do tyłu” określone przez czynniki $\exp(-a_1)$ oraz $\exp(-a_2)$. Odległość $r_{2'_a \rightarrow 1''_b} = r_{1''_b \rightarrow 2'_a} = r_{1''_a \rightarrow 2'_b} = r_{2'_b \rightarrow 1''_a} = 1$.



Rysunek 4.14. Zależność stopnia sprężenia ϵ od siły a_2 dla obszaru obejmującego przetwarzanie energii swobodnej dla modelu typu $N_{P1} = 1, N_{P2} = 1$ (kółka), $N_{P1} = 1, N_{P2} = 2$ (kwadraty) oraz $N_{P1} = 2, N_{P2} = 2$ (trójkąty).

poprzednich rozważaniach zostało pokazane jak wiele parametrów wpływa na dynamikę modelu, zarówno te związane z prawdopodobieństwami przejść jak i z geometrią bram. W tym podrozdziale dodatkowym parametrem będzie liczba procesów (par bram) dwóch typów. Interesować nas będzie zależność stopnia sprężenia ϵ od a_2 bezpośrednio w obszarze przetwarzania energii swobodnej dla trzech przypadków z różną liczbą procesów dostarczających i odbierających energię swobodną. Wyniki przeprowadzonych symulacji zamieszczono na rysunku 4.14.

Wyznaczone wartości siły blokującej $a_2^{st}(\infty)$ i stopnia sprężenia $\epsilon(0)$ dla trzech typów modeli dynamiki przejść konformacyjnych z różną liczbą procesów dwóch typów przedstawiono w tabeli 4.8. Niestety rozbudowanie modelu o dodatkowy proces odbierający energię swobodną nie zwiększyło w sposób znaczący wartości stopnia sprężenia $\epsilon(0)$. Pomimo, że symulowana maszyna molekularna przetwarza energię swobodną, to jej wydajność jest dość słaba. Zmiany konformacyjne, które nie prowadziły do przetwarzania energii swobodnej okazały się w rozważanym modelu bardziej częste od samych procesów dostarczania i odbierania energii swobodnej, stąd też niski stopień sprężenia. Ponadto dodanie nowych procesów mimo wzrostu $\epsilon(0)$ powoduje ogólne zmniejszenie się obszaru przetwarzania energii swobodnej, który może być liczony jako pole trójkąta, ze względu na coraz bliższą zeru wartość siły blokującej a_2^{st} . W tabeli 4.8 umieszczono w formie uzupełnienia także przypadek dwóch procesów dostarczających, a jednego od-

Tabela 4.8. Wyznaczone wartości siły blokującej $a_2^{st}(\infty)$ i stopnia sprężenia $\epsilon(0)$ dla modelu z jednym procesem dostarczającym i jednym odbierającym energię swobodną oraz dla modeli rozszerzonych o dodatkowe procesy. Błędy wyznaczania wielkości identyczne jak dla wcześniej przedstawianych obliczeń $a_2^{st}(\infty)$ i $\epsilon(0)$.

Model	$a_2^{st}(\infty)$	$\epsilon(0)$
$N_{P1} = 1, N_{P2} = 1$	-0,48	0,235
$N_{P1} = 1, N_{P2} = 2$	-0,29	0,362
$N_{P1} = 2, N_{P2} = 2$	-0,29	0,144
$N_{P1} = 2, N_{P2} = 1$	-0,70	0,147

bierającego energię swobodną ($N_{P1} = 2, N_{P2} = 1$), który otrzymujemy poprzez zamianę stanów w modelu z rysunku 4.12.

Wcześniej pokazano, że dla przypadku jednego procesu dostarczającego i jednego odbierającego energię swobodną, stopień sprężenia nie może być większy od 1. Niestety mimo wprowadzenia większej liczby przejść dostarczających i odbierających energię swobodną nie udało się otrzymać na sieci typu hipersześcianu stopnia sprężenia większego od 1. Symulacje na dziewięciowymiarowym hipersześcianie (512 stanów) także zakończyły się niepowodzeniem. Wysuwam hipotezę, że na izoenergetycznej sieci regularnej typu hipersześcianu nie można otrzymać stopnia sprężenia większego od jedności.

4.2.5 Rozszerzenie modelu na sieci innego typu

Wszystkie dotychczasowe obliczenia w ramach badania efektów dynamiki przejść konformacyjnych przeprowadzono na sieci regularnej typu hipersześcianu, której wybór był podyktowany względną prostotą sieci. Niestety nie udało się uzyskać w obszarze przetwarzania energii swobodnej stopnia sprężenia ϵ większego od jedności nawet dla większej liczby sprzężonych procesów dostarczających i odbierających energię swobodną.

W tym podrozdziale będziemy podążać drogą od struktur regularnych do hierarchicznych, co jest w pełni uzasadnione, gdyż hierarchiczność jest cechą dynamiki białek (patrz model białko-szkło w książce [10]). Najpierw dokonamy rozszerzenia dyskutowanego modelu na sieć typu trójkąta Sierpińskiego, a obliczenia przeprowadzimy dla trójkąta Sierpińskiego rzędu 3 z okresowymi warunkami brzegowymi (patrz rozdział na temat metodologii) i numeracją stanów przedstawioną na rysunku 2.8.

Rozszerzenie tego typu możliwe było dopiero po poznaniu pewnych własności modelu na prostszej sieci, gdyż już na trójkącie Sierpińskiego system wąskich gardeł i wybrane warunki brzegowe powodują, że sama zależność średniego czasu pierwszego przejścia od odległości chemicznej nie jest jednoznaczna, co więcej w wielu przypadkach nawet po zamianie stanu początkowego z końcowym otrzymuje się inny średni czas pierwszego przejścia. Dla maksymalnej odległości pomiędzy dwoma stanami równej $r = 7$ (ma to związek z narzuconymi periodycznymi warunkami brzegowymi), znaleziono sześć wartości średnich czasów pierwszego przejścia charakteryzujących grupy par stanów. W ramach każdej grupy zawarte są pary stanów, których wybór jest równoważny ze względu na symetrię trójkąta Sierpińskiego. Czasy te zamieszczono w tabeli 4.9.

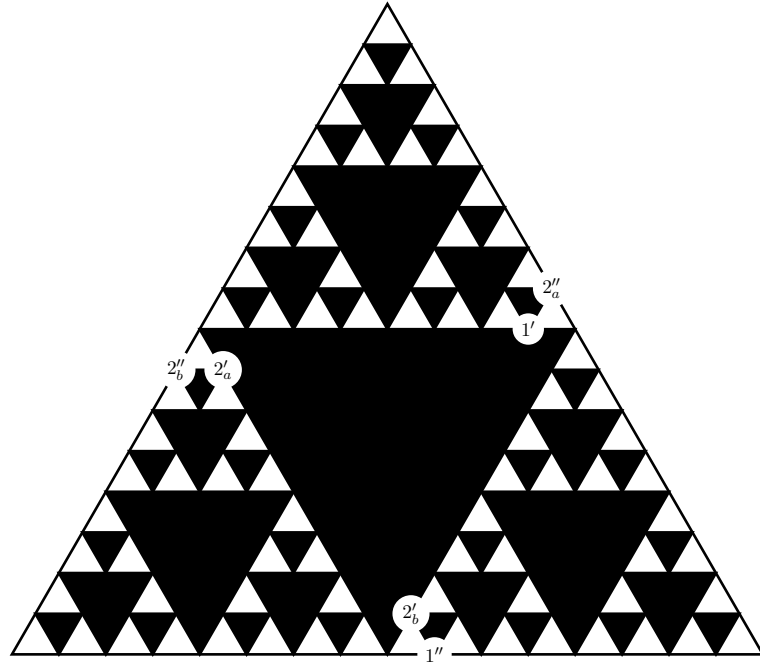
Tabela 4.9. Niejednoznaczność zależności średniego czasu pierwszego przejścia τ_M od odległości chemicznej i wyboru stanu początkowego na trójkącie Sierpińskiego. Badano trójkąt trzeciego rzędu (rysunek 2.8) i maksymalną odległość pomiędzy stanami $r = 7$.

τ_M	$a \rightarrow b$
106, 3	$39 \rightarrow 3, 25 \rightarrow 4, 11 \rightarrow 5$
114, 8	$3 \rightarrow 39, 4 \rightarrow 25, 5 \rightarrow 11$
111, 6	$22 \rightarrow 31, 33 \rightarrow 18, 17 \rightarrow 20, 14 \rightarrow 32, 36 \rightarrow 6, 28 \rightarrow 7$
114, 2	$31 \rightarrow 22, 18 \rightarrow 33, 20 \rightarrow 17, 32 \rightarrow 14, 6 \rightarrow 36, 7 \rightarrow 28$
115, 2	$23 \rightarrow 34, 37 \rightarrow 12, 27 \rightarrow 16, 35 \rightarrow 21, 13 \rightarrow 38, 15 \rightarrow 29$
117, 4	$34 \rightarrow 23, 12 \rightarrow 37, 16 \rightarrow 27, 21 \rightarrow 35, 38 \rightarrow 13, 29 \rightarrow 15$

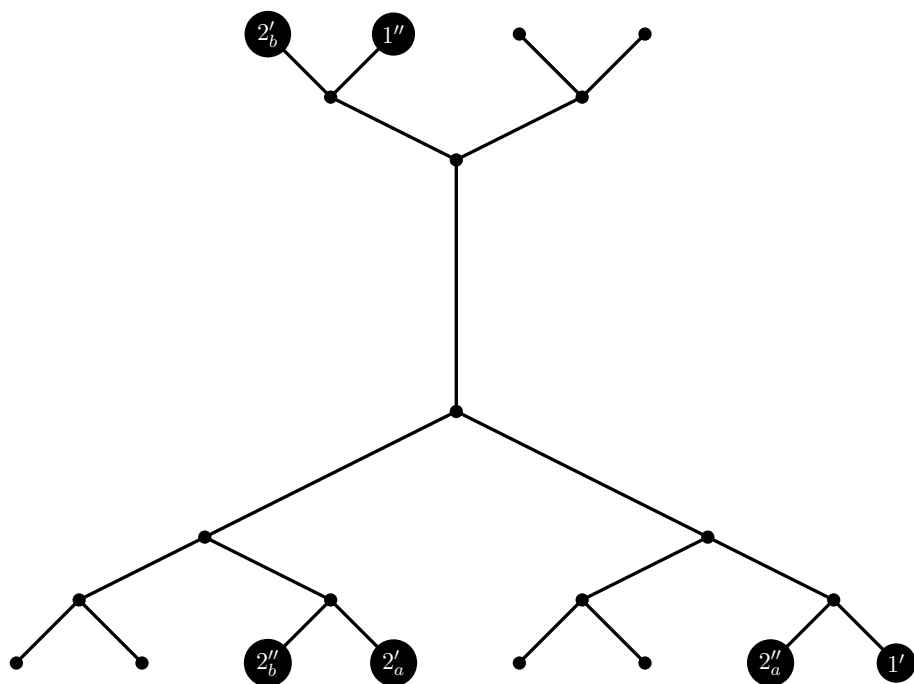
Jak wynika z powyższych rozważań, wybór stanów będących bramami jest jeszcze bardziej skomplikowany niż dla sieci typu hipersześcianu. W tabeli 4.10 zamieszczono wartości stopnia sprzężenia $\epsilon(0)$ w zależności od wyboru i liczby stanów będących bramami. Warto przypomnieć, że ważny w tym przypadku jest wybór bramy początkowej (bis) i końcowej procesu (prim), gdyż w przypadku trójkąta Sierpińskiego zamiana bram może wpływać na średni czas pierwszego przejścia, ale nie musi (porównaj wiersz pierwszy i trzeci w tabeli 4.10). Istotna jest też zamiana procesów ze sobą (porównaj wiersz pierwszy i drugi w tabeli 4.10). Mimo, że na trójkącie Sierpińskiego trzeciego rzędu stopnie sprzężenia są znacząco wyższe od tych jakie uzyskiwano na hipersześcianie, to nie udało odnaleźć się takiej geometrii bram, aby $\epsilon(0)$ było większe od jedności. Dla najlepszej znalezionej geometrii (ostatni wiersz w tabeli 4.10) otrzymano wartość $\epsilon(0) = 0,954$. Model rozszerzono na trójkąt Sierpińskiego czwartego rzędu. W tym przypadku dla geo-

Tabela 4.10. Wartość stopnia sprężenia $\epsilon(0)$ w zależności od wyboru geometrii stanów i liczby bram na trójkącie Sierpińskiego trzeciego rzędu. W tabeli podano numery stanów będące bramami zgodne z rysunkiem 2.8. Z tabeli wynika też liczba procesów dostarczających (cyfra 1) i odbierających (cyfra 2) energię swobodną. Wartości podano z błędem wyznaczenia $\Delta\epsilon(0) = 0,001$.

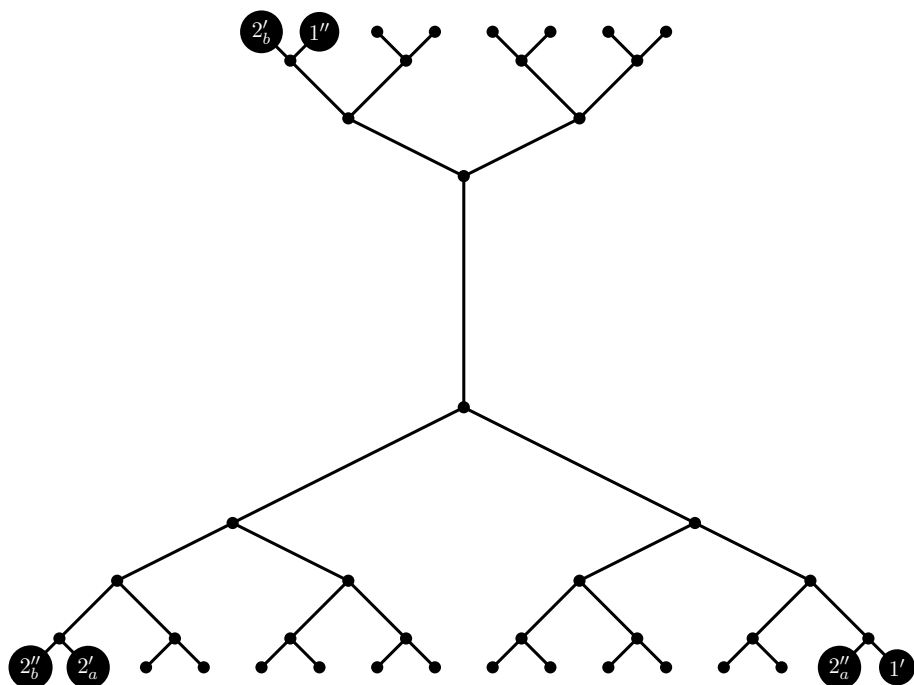
$1''_a$	$1'_a$	$1''_b$	$1'_b$	$1''_c$	$1'_c$	$2''_a$	$2'_a$	$2''_b$	$2'_b$	$2''_c$	$2'_c$	$\epsilon(0)$
21	35	—	—	—	—	33	18	—	—	—	—	0,644
33	18	—	—	—	—	21	35	—	—	—	—	0,626
35	21	—	—	—	—	18	33	—	—	—	—	0,644
21	35	12	37	—	—	33	18	36	6	—	—	0,539
21	35	—	—	—	—	33	18	14	32	28	7	0,310
21	36	—	—	—	—	38	17	16	22	—	—	0,954



Rysunek 4.15. Trójkąt Sierpińskiego czwartego rzędu, na którym zrealizowano symulację z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Dla przedstawionej geometrii bram otrzymano stopień sprężenia $\epsilon(0) = 1,271$.



Rysunek 4.16. Sieć Bethego z trzema powłokami, na której zrealizowano symulację z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Odległość pomiędzy parą bram dla danego procesu wynosi $o = 6$, a odległość pomiędzy najbliższymi bramami $r = 2$.



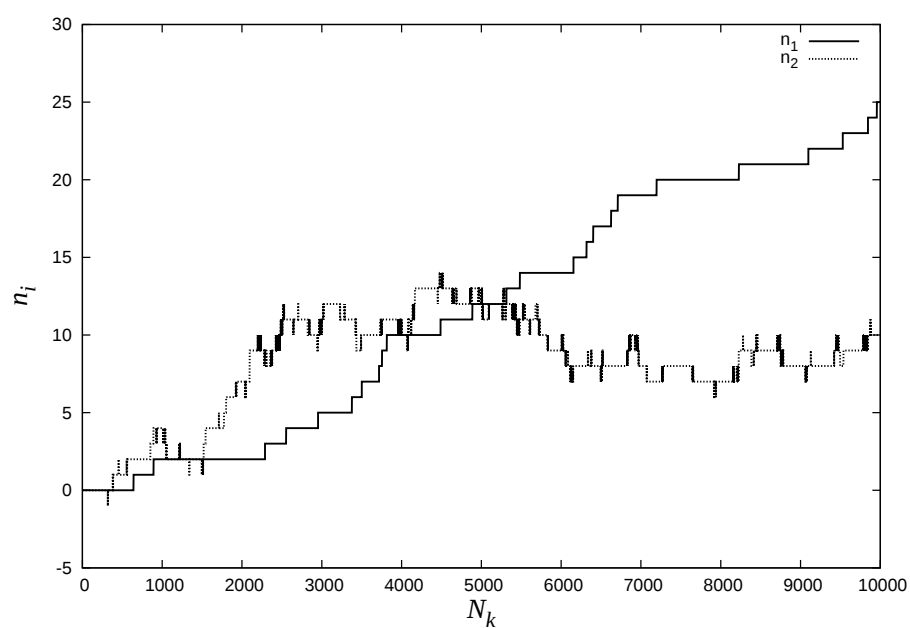
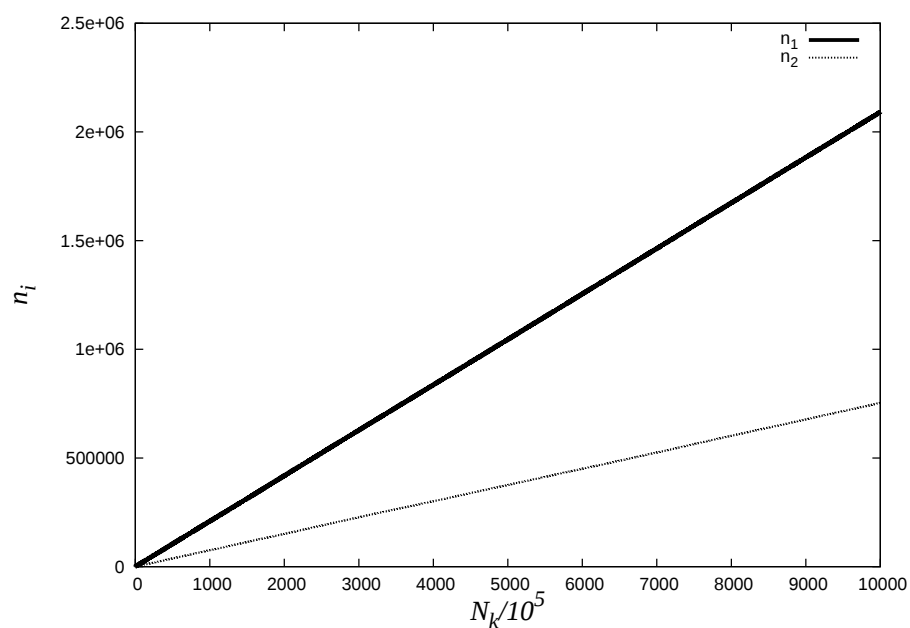
Rysunek 4.17. Sieć Bethego z czterema powłokami, na której zrealizowano symulację z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Odległość pomiędzy parą bram dla danego procesu wynosi $o = 8$, a odległość pomiędzy najbliższymi bramami $r = 2$.

metrii bram przedstawionej na rysunku 4.15 otrzymano już stopień sprężenia większy od jedności, $\epsilon(0) = 1,271 \pm 0,002$.

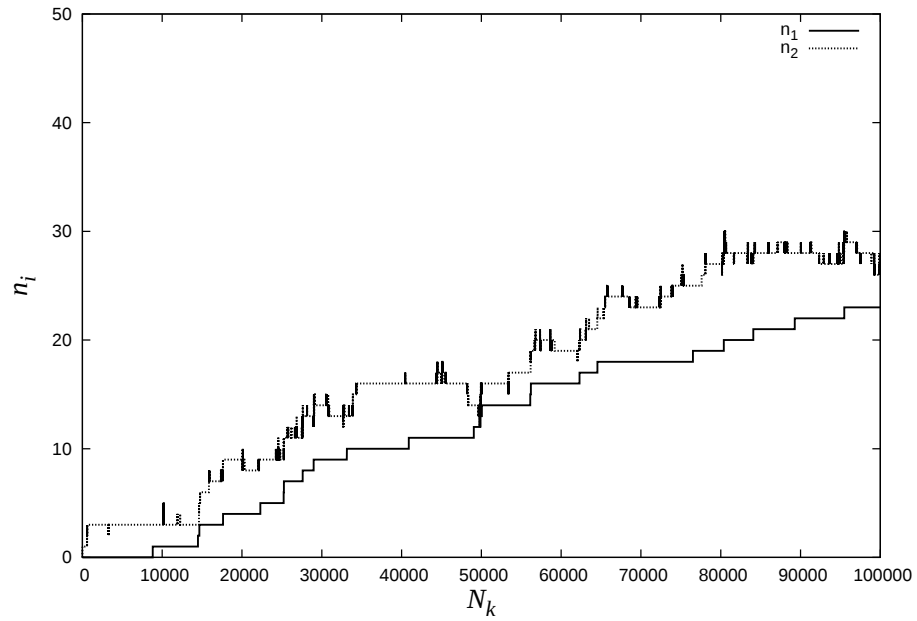
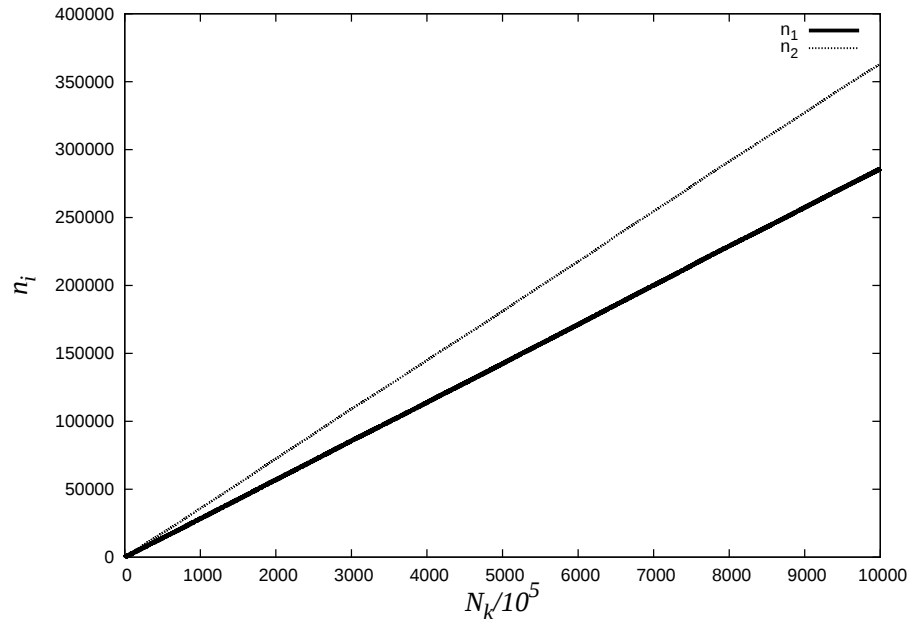
Na koniec sprawdzimy efekt rozszerzenia modelu na sieć typu sieci Bethego (rysunek 4.16 i 4.17) [57]. W ten sposób zamiast systemu wąskich gardeł pojawiają się tzw. martwe końce. Wszystkie odległości pomiędzy sąsiednimi stanami są identyczne. Skrajne stany mają tylko jednego sąsiada zamiast trzech, zatem nałożono warunki brzegowe w postaci czekania na jeden krok czasowy jeśli wędrowiec wybierze jednego z dwóch nieistniejących sąsiadów. To inny sposób oprócz warunków periodycznych aby zachować prawdopodobieństwa przejść wewnętrznych w obie strony takie same. Celowo nie wprowadzono warunków periodycznych aby zachować wyłącznie jedną ścieżkę pomiędzy parą stanów, czyli nie zniszczyć ważnej cechy tej sieci jaką jest brak cykli. W symulacji użyto 10 wędrowców, z których każdy przechodził 10^9 kroków, a ponadto, inaczej niż w poprzednich symulacjach, każdy wędrowiec rozpoczynał swoją ewolucję z losowo wybranego stanu na sieci. Symulację tego typu wykonywano kilka razy aby upewnić się, że mimo występowania martwych końców i czekania jako warunku brzegowego możliwe jest dostateczne zwiedzenie sieci.

Dla sieci i rozkładu bram z rysunku 4.16 (3 powłoki, 22 stany), przy $v_1/p = 50$, $v_2/p = 30$, $a_1 = 10$, $a_2 = 0$, otrzymano $\epsilon(0) = 0,992 \pm 0,001$, natomiast po dodaniu czwartej powłoki jak na rysunku 4.17 (46 stanów, odległość $o = 8$) otrzymano stopień sprężenia większy od jedności, $\epsilon(0) = 1,193 \pm 0,002$.

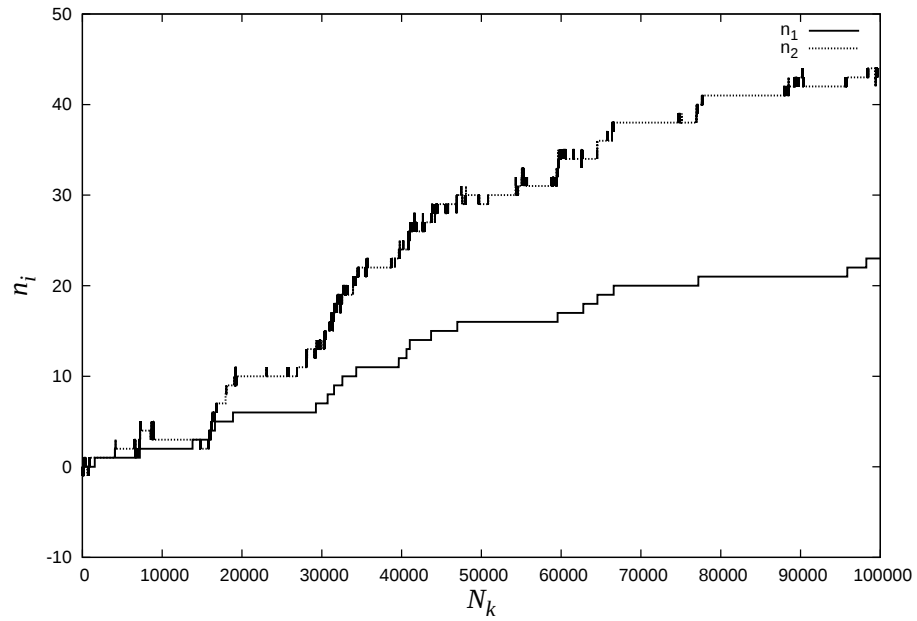
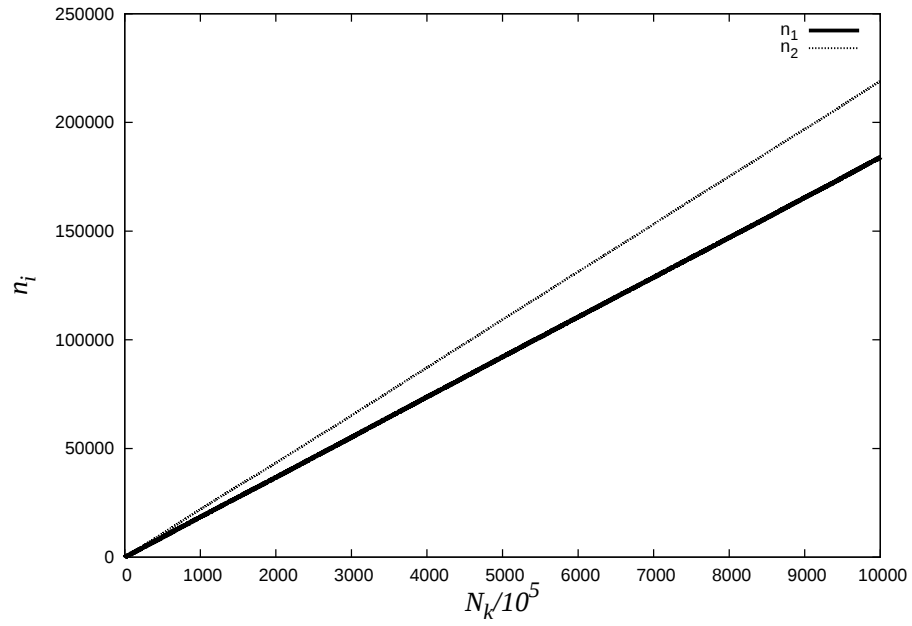
Uzyskanie wartości stopnia sprężenia większego od 1 było jednym z celów pracy. Przypomnijmy, że $\epsilon(0) > 1$ pojawił się dopiero dla trójkąta Sierpińskiego czwartego rzędu i sieci Bethego z czterema powłokami. Pozostaje próba odpowiedzenia na pytanie jakie własności sieci powodują, że stopień sprężenia $\epsilon(0)$ jest większy od jedności. Zarówno sieć typu hipersześcianu jak i trójkąt Sierpińskiego miała cykle, podczas gdy sieć typu sieci Bethego jest bezcyklowa. Z drugiej strony trójkąt Sierpińskiego i sieć Bethego są sieciami fraktalnymi i cechują się hierarchicznością, co nie jest cechą sieci typu hipersześcianu. W rezultacie można postawić hipotezę, że pojawienie się $\epsilon(0) > 1$ związane jest zarówno z odpowiednim rzędem wąskich gardeł (trójkąt Sierpińskiego), jak i z brakiem cykli i istnieniem martwych końców (sieć Bethego). Wzrost stopnia sprężenia przy zwiększeniu rzędu trójkąta Sierpińskiego i dodaniu powłoki do sieci Bethego można powiązać z ogólnym zwiększeniem się średnich czasów pierwszego przejścia. Ważne jest zatem aby przejście wewnętrzne było o wiele trudniejsze do realizacji od przejścia zewnętrznego, a ponadto aby po zajęciu procesu dostarczającego energię swobodną kolejne dojście do bramy wejściowej procesu dostarczającego energię swobodną było o wiele trudniejsze od dojścia do bramy procesu odbierającego energię swobodną.



Rysunek 4.18. Zależność liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ (linia ciągła) i liczby n_2 przejść zewnętrznych $R_2 \leftrightarrow P_2$ (linia przerywana) od czasu wyrażonego w liczbie kroków N_k dla ewolucji pojedynczego wędrowca na pięciowymiarowym hipersześcianie. Na górnym rysunku przedstawiono 10^9 kroków „fotografowanych” co 10^5 kroków. Na dolnym rysunku przedstawiono pierwsze 10000 kroków. Parametry użyte w symulacji: $N_{P1} = 1$, $N_{P2} = 2$, $a_1 = 10$, $a_2 = 0$, $v_1/p = 50$, $v_2/p = 30$, $p = 1,82 \cdot 10^{-2}$. Dla tego przypadku $\epsilon(0) = 0,362$.



Rysunek 4.19. Zależność liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ (linia ciągła) i liczby n_2 przejść zewnętrznych $R_2 \leftrightarrow P_2$ (linia przerywana) od czasu wyrażonego w liczbie kroków N_k dla ewolucji pojedynczego wędrowca na trójkącie Sierpińskiego czwartego rzędu. Na górnym rysunku przedstawiono 10^9 kroków „fotografowanych” co 10^5 kroków. Na dolnym rysunku przedstawiono pierwsze 100000 kroków. Zatem dolny rysunek reprezentowany jest na górnym rysunku tylko przez dwa punkty. Parametry użyte w symulacji: $N_{P1} = 1$, $N_{P2} = 2$, $a_1 = 10$, $a_2 = 0$, $v_1/p = 50$, $v_2/p = 30$, $p = 1,85 \cdot 10^{-2}$. Dla tego przypadku $\epsilon(0) = 1,271$.



Rysunek 4.20. Zależność liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ (linia ciągła) i liczby n_2 przejść zewnętrznych $R_2 \leftrightarrow P_2$ (linia przerywana) od czasu wyrażonego w liczbie kroków N_k dla ewolucji pojedynczego wędrowca na sieci Bethego z 46 stanami. Na górnym rysunku przedstawiono 10^9 kroków „fotografowanych” co 10^5 kroków. Na dolnym rysunku przedstawiono pierwsze 100000 kroków. Zatem dolny rysunek reprezentowany jest na górnym rysunku tylko przez dwa punkty. Parametry użyte w symulacji: $N_{P_1} = 1$, $N_{P_2} = 2$, $a_1 = 10$, $a_2 = 0$, $v_1/p = 50$, $v_2/p = 30, 1, 89 \cdot 10^{-2}$. Dla tego przypadku $\epsilon(0) = 1, 193$.

Na zakończenie warto pokazać zależność liczby przejść dostarczających $R_1 \leftrightarrow P_1$ i odbierających $R_2 \leftrightarrow P_2$ energii swobodną od czasu dla ewolucji jednego wędrowca dla przypadku gdy $\epsilon(0)$ jest mniejsze i większe od 1. W obu przypadkach rozważano modele z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Na rysunku 4.18 przedstawiono ewolucję na pięciowymiarowym hipersześcianie (geometria z rysunku 4.12), na rysunku 4.19 ewolucję na trójkącie Sierpińskiego (geometria z rysunku 4.15), a na rysunku 4.20 ewolucję na sieci Bethego z 46 stanami (geometria z rysunku 4.17). W przeciwieństwie do pięciowymiarowego hipersześcianu, na trójkącie Sierpińskiego i sieci Bethego wędrowiec musi przejść więcej kroków aby doszło do stacjonarnego obsadzenia stanów. Dla wszystkich sieci pokazano najpierw ewolucję przez 10^9 kroków, a dane na wykresie były „fotografiami” co 10^5 kroków, a następnie pokazano ewolucję przez pierwsze 10000 kroków dla hipersześcianu i 100000 kroków dla trójkąta Sierpińskiego i sieci Bethego. O ile pierwszy typ wykresu przy powtórzeniu symulacji będzie wyglądał tak samo, to już ewolucje przez mniejszą liczbą kroków będą inne w każdej symulacji.

Podsumowanie

Po krótkim wprowadzeniu do najważniejszych zagadnień związanych z dynamiką białka i teorią procesów stochastycznych, w pracy przedstawiono podstawy sztuki przeprowadzania symulacji Monte Carlo, którą autor pracy samodzielnie od podstaw zgłębiał i rozwijał w trakcie pisania oprogramowania w języku Fortran.

W rozdziale dotyczącym metodologii badań zwrócono uwagę na takie zagadnienia jak przebieg eksperymentu Monte Carlo, obliczanie błędu wyznaczania wartości średniej, wybór sieci (pokazano trzy typy: sieć liniową, hipersześcian oraz trójkąt Sierpińskiego) oraz problem generowania liczb pseudolosowych wraz z testami sprawdzającymi używane przez autora generatory. Pokazano jak średnie czasy pierwszego przejścia zależą na hipersześcianie od wymiaru i odległości chemicznej oraz jak skalują się błędy i rzeczywiste czasy obliczeń w zależności od liczby wędrówców. Fakty te były wprowadzeniem do przetwarzania równoległego, podania szczegółów na temat Message Passing Interface i implementacji MPICH wraz z przykładami kodów równoległych. Pokazano jak w prosty sposób można przygotować komputer do obliczeń równoległych, a następnie zdefiniowano dwa ważne pojęcia - przyspieszenie i wydajność. Pokazano też, że obliczanie średniego czasu pierwszego przejścia może być zrównoleglone z wysoką wydajnością przy niewielkich modyfikacjach programu sekwencyjnego. Osobnym tematem było wykorzystanie powszechnie używanych komputerów z procesorami wielordzeniowymi. Nawet na nich wydajność może być większa od 93% dla trzech problemów o różnym zrównoważeniu obciążeń, co powinno być zachętą dla szerokiej rzeszy naukowców aby w pełni wykorzystywać moce obliczeniowe komputerów.

Podstawy teoretyczne i praktyczne były wstępem do wykonania symulacji dynamiki wibracyjnej i przejść konformacyjnych (wyłącznie przy pomocy autorskiego oprogramowania napisanego w języku Fortran), które udowadniają tezę, że przy pomocy symulacji Monte Carlo prostych modeli dynamiki biomolekuł i używając stochastycznej teorii szybkości reakcji można badać efekty wpływu dynamiki otoczenia białkowego w szczególności na procesy przeniesienia elektronu oraz wyjaśniać i próbować przewidywać fakty doświadczalne.

W rozdziale dotyczącym dynamiki wibracyjnej zajęto się pierwotnym pikosekundowym procesem przeniesienia elektronu w fotosyntetycznym centrum reakcji. Rozważono

wpływ sprzężenia ze słabo tłumionym szybkim modem wibracyjnym otoczenia białkowego. Sformułowano problem w języku stochastycznej teorii szybkości reakcji jako dyfuzję w przestrzeni energetycznej i opisano szczegóły symulacji. Zbadano analityczne wyrażenie na średni czas pierwszego przejścia w przybliżeniu ciągłym i wykazano niedostateczną jakość rozwinięć wynikających z tego wyrażenia. Pokazano też, że przybliżenie ciągle słabo działa w fizjologicznych temperaturach i wysokościach barier potencjału. Dla przypadku gdy stan początkowy znajdował się powyżej stanu przejściowego pojawił się efekt wzrostu szybkości procesu wraz z obniżaniem temperatury. Było to cenną wskazówką na drodze do wyjaśnienia takiej anomalnej zależności obserwowanej eksperymentalnie. W rezultacie udało się dość dobrze dopasować wyniki symulacji komputerowych do wyników eksperymentalnych. Okazało się, że nie wystarczy wprowadzić poprawki do stałej szybkości związanej z relaksacją wibracyjną. Dodatkowo należy uwzględnić etap początkowy reakcji rozpoczynający się od wzbudzenia energetycznie powyżej stanu przejściowego.

W rozdziale dotyczącym dynamiki przejść konformacyjnych zajęto się przetwarzaniem energii swobodnej w kompleksie cytochromów bc_1 i zauważono związek z podobnymi reakcjami zachodzącymi w centrach fotosyntetycznych. Rozważono zastosowanie ogólnego modelu dwóch sprzężonych reakcji jednocząsteczkowych w języku stochastycznej teorii stanu stacjonarnego i opisano szczegóły symulacji. Badane sieci podstawów konformacyjnych były izoenergetyczne, stąd ważne były przyczynki entropowe a nie energetyczne do stopnia sprzężenia obu reakcji. Również brak równowagi częściowej okazał się bardzo ważny. Symulacje rozpoczęto dla modelu pięciowymiarowego hiper-sześcianu, ze względu na jego regularność i jednoznaczność zależności średniego czasu pierwszego przejścia od odległości chemicznej. Policzono średnie czasy pierwszego przejścia dla przypadku jednego i dwóch stanów końcowych dla wszystkich możliwych odległości stanów od siebie. W tym drugim przypadku dodatkowym parametrem była odległość pomiędzy alternatywnymi stanami końcowymi. Badania rozpoczęto od poszukiwania takiego zestawu parametrów aby stopień sprzężenia dla braku obciążenia zewnętrznego (kraniec obszaru przetwarzania energii swobodnej) był jak największy. Sprawdzono zarówno parametry związane z geometrią, jak i z prawdopodobieństwami przejść i siłami. Dzięki istniejącym wyrażeniom analitycznym dla modelu z jednym procesem dostarczającym i jednym odbierającym energię swobodną pokazano zgodność wyników symulacji numerycznych z teorią. Niestety dla większej liczby procesów wyrażenia teoretyczne nie są znane, a zatem symulacje są jedynym źródłem wyników. Pokazano zależność strumienia procesu odbierającego energię swobodną oraz stopnia sprzężenia od siły związanej z tym strumieniem, a także zależność siły blokującej związanej ze strumieniem procesu odbierającego energię swobodną od siły związanej ze strumieniem procesu dostarczają-

cego energię swobodną. Modelowana maszyna molekularna działała, gdyż występowało przetwarzanie energii swobodnej, jednakże obszar sił był stosunkowo niewielki.

Z teorii wynika, że dla jednego procesu dostarczającego i jednego odbierającego energię swobodną nie można otrzymać stopnia sprzężenia (ilorazu strumienia procesu odbierającego energię swobodną i strumienia procesu dostarczającego energię swobodną) większego od 1. Niestety po rozszerzeniu modelu na większą liczbę procesów na pięciowymiarowym hipersześcianie nie otrzymano wartości większej od 1. Rozważono zatem sieci fraktalne, lecz tam sytuacja skomplikowała się z powodu systemu wąskich gardeł, martwych końców i wpływu nałożonych warunków brzegowych. Na trójkącie Sierpińskiego trzeciego rzędu stopnie sprzężenia były znacząco wyższe, ale dopiero na trójkącie Sierpińskiego czwartego rzędu otrzymano stopień sprzężenia większy od jedności. Zbadano także sieć Bethego, w której nie ma cykli (nie licząc tych utworzonych przez wprowadzenie przejść zewnętrznych) i otrzymano stopień sprzężenia większy od jedności dopiero dla sieci z czterema powłokami. Stopnie sprzężenia większe od jedności otrzymano już dla modelu z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną.

Bardzo ważnym wnioskiem z przedstawionych rozważań jest to, że stopień sprzężenia może być zarówno mniejszy jak i większy od jedności. Wysunięto hipotezę, że wartość stopnia sprzężenia zależy zarówno od rzędu wąskich gardeł (trójkąt Sierpińskiego) jak i od braku cykli i istnienia martwych końców (sieć Bethego). Stwierdzono, że ważne jest aby przejście wewnętrzne było o wiele trudniejsze do realizacji od przejścia zewnętrznego, a ponadto aby po zejściu procesu dostarczającego energię swobodną kolejne dojście do bramy wejściowej procesu dostarczającego energię swobodną było o wiele trudniejsze od dojścia do bramy procesu odbierającego energię swobodną. Przypuszczalnie nowe efekty powinny pojawić się gdy odrzuci się założenie o izoenergetyczności sieci, ale wykracza to już poza zakres tej pracy.

Proponowane modele są dość ogólne i mogą znaleźć zastosowanie przy badaniu innych problemów. Tworzenie i weryfikacja tego typu modeli przybliży nas do coraz lepszego zrozumienia zjawisk, a często pozwala badać wpływ efektów, których nie można badać doświadczalnie.

Bibliografia

Prace własne

- [A] M. Torchała, „Efficient parallel computations using MPI in chemistry and physics”, *Adv. ChemInfo.* **1**, 21-27 (2007).
- [B] M. Torchała, M. Kurzyński, „Underdamped vibrations control the primary electron transfer in photosynthesis at low temperatures”, *J. Phys. Chem. B* **112**, 6508-6512 (2008).
- [C] M. Kurzyński, M. Torchała, „Free energy transduction in conformationally fluctuating proteins”, w przygotowaniu.

Pozostałe źródła

- [1] *The Nobel Prize in chemistry 1992* (odwiedzono 1.03.2010);
http://nobelprize.org/nobel_prizes/chemistry/laureates/1992/index.html.
- [2] R. A. Marcus, „Electron transfer past and future”, *Adv. Chem. Phys.* **106**, 1-6 (1999).
- [3] J. Jortner, M. Bixon (edytorzy), *Electron transfer – from isolated molecules to biomolecules*, Part 1, Advances in Chemical Physics, Volume 106, John Wiley & Sons, Inc., 1999.
- [4] J. Jortner, M. Bixon (edytorzy), *Electron transfer – from isolated molecules to biomolecules*, Part 2, Advances in Chemical Physics, Volume 107, John Wiley & Sons, Inc., 1999.
- [5] M. Kurzyński, „Enzymatic catalysis as a process controlled by protein conformational relaxation”, *FEBS Lett.* **328**, 221-224 (1993).
- [6] M. Kurzyński, „A synthetic picture of intramolecular dynamics of proteins. Towards a contemporary statistical theory of biochemical processes”, *Progr. Biophys. Mol. Biol.* **69**, 23-82 (1998).

- [7] M. Kurzyński, K. Palacz, P. Chełminiak, „Time course of reactions controlled and gated by intramolecular dynamics of proteins: Predictions of the model of random walk on fractal lattices”, *Proc. Natl. Acad. Sci. USA* **95**, 11685-11690 (1998).
- [8] M. Kurzyński, P. Chełminiak, „Mean first-passage time in the stochastic theory of biochemical processes. Application to actomyosin molecular motor”, *J. Statist. Phys.* **110**, 137-181 (2003).
- [9] P. Chełminiak, M. Kurzyński, „Mean first-passage time for diffusion on fractal lattices with imposed boundary conditions”, *Physica A* **342**, 507-515 (2004).
- [10] M. Kurzyński, *The thermodynamic machinery of life*, Springer, 2006.
- [11] M. Kurzyński, „Statistical properties of the dichotomous noise generated in biochemical processes”, *Cell. Mol. Biol. Lett.* **13**, 502-513 (2008).
- [12] H. Lodish, A. Berk, L. S. Zipursky, P. Matsudaira, D. Baltimore, J. Darnell, *Molecular cell biology*, Fourth Edition, W. H. Freeman, 2000.
- [13] D. P. Landau, K. Binder, *A guide to Monte Carlo simulations in statistical physics*, Cambridge University Press, 2000.
- [14] T. L. Hill, *Free energy transduction and biochemical cycle kinetics*, Springer-Verlag, 1989.
- [15] W. Sobieski, *GNU Fortran z elementami wizualizacji danych*, Wydawnictwo Uniwersytetu Warmińsko-Mazurskiego w Olsztynie, 2008.
- [16] A. Trykozko, *Ćwiczenia z języka Fortran*, MIKOM, 1999.
- [17] *Using GNU Fortran. For GCC version 4.4.2.* (odwiedzono 1.03.2010); <http://gcc.gnu.org/onlinedocs/>.
- [18] I. N. Bronsztejn, K. A. Siemiendajew, G. Musiol, H. Mühlig, *Nowoczesne kompendium matematyki*, Wydawnictwo Naukowe PWN, 2004.
- [19] E. Steiner, *Matematyka dla chemików*, Wydawnictwo Naukowe PWN, 2001.
- [20] J. C. Angles d’Auriac, A. Benoit, R. Rammal, „Random walk on fractals: numerical studies in two dimensions”, *J. Phys. A: Math. Gen.* **16**, 4039-4051 (1983).
- [21] D. Loshin, *Superkomputery bez tajemnic*, MIKOM, 1997.
- [22] B. E. Borowik, *Programowanie równoległe w zastosowaniach*, MIKOM, 2001.

- [23] *PSTricks* (odwiedzono 1.03.2010); <http://tug.org/PSTricks/main.cgi>.
- [24] E. W. Dijkstra, „A note on two problems in connexion with graphs”, *Numer. Math.* **1**, 269-271 (1959).
- [25] W. P. Petersen, P. Arbenz, *Introduction to parallel computing*, Oxford University Press, 2004.
- [26] Ben O’Shaughnessy, I. Procaccia, „Diffusion on fractals”, *Phys. Rev. A* **32**, 3073-3083 (1985).
- [27] *GNUPLOT* (odwiedzono 1.03.2010); <http://www.gnuplot.info/>.
- [28] S. G. Akl, „The design of efficient parallel algorithms”, w: *Handbook on parallel and distributed processing*, red. J. Błażewicz, K. Ecker, B. Plateau, D. Trystram, Springer, 2000.
- [29] I. Foster, „Languages for parallel processing”, w: *Handbook on parallel and distributed processing*, red. J. Błażewicz, K. Ecker, B. Plateau, D. Trystram, Springer, 2000.
- [30] *The Message Passing Interface (MPI) standard* (odwiedzono 1.03.2010); <http://www.mcs.anl.gov/research/projects/mpi/>.
- [31] *MPICH – A portable implementation of MPI* (odwiedzono 1.03.2010); <http://www.mcs.anl.gov/research/projects/mpi/mpich1/>.
- [32] *MPICH 2 – High-performance and widely portable MPI* (odwiedzono 1.03.2010); <http://www.mcs.anl.gov/research/projects/mpich2/>.
- [33] *Linux Debian* (odwiedzono 1.03.2010); <http://www.debian.org>.
- [34] P. S. Pacheco, *Parallel programming with MPI*, Morgan Kaufmann Publishers, Inc., 1997.
- [35] M. J. Quinn, *Parallel programming in C with MPI and OpenMPI*, McGraw-Hill, 2004.
- [36] W. Gropp, E. Lusk, D. Ashton, P. Balaji, D. Buntinas, R. Butler, A. Chan, D. Goodell, J. Krishna, G. Mercier, R. Ross, R. Thakur, B. Toonen, *MPICH2 user’s guide version 1.2.1* (odwiedzono 1.03.2010); <http://www.mcs.anl.gov/research/projects/mpich2/>.

- [37] W. Gropp, E. Lusk, D. Ashton, P. Balaji, D. Buntinas, R. Butler, A. Chan, D. Goodell, J. Krishna, G. Mercier, R. Ross, R. Thakur, B. Toonen, *MPICH2 installer's guide version 1.2.1* (odwiedzono 1.03.2010);
<http://www.mcs.anl.gov/research/projects/mpich2/>.
- [38] *Hyper-Threading Technology* (odwiedzono 1.03.2010);
<http://www.intel.com/technology/platform-technology/hyper-threading/index.htm>.
- [39] W. Zinth, J. Wachtveitl, „The first picoseconds in bacterial photosynthesis – ultrafast electron transfer for the efficient conversion of light energy”, *ChemPhysChem* **6**, 871-880 (2005).
- [40] D. A. Cherepanov, L. I. Krishtalik, A. Y. Mulkidjanian, „Photosynthetic electron transfer controlled by protein relaxation: analysis by Langevin stochastic approach”, *Biophys. J.* **80**, 1033-1049 (2001).
- [41] G. R. Fleming, J. L. Martin, J. Breton, „Rates of primary electron transfer in photosynthetic reaction centres and their mechanistic implications”, *Nature* **333**, 190-192 (1988).
- [42] J. L. Martin, J. Breton, J. C. Lambry, G. Fleming, „The primary electron transfer in photosynthetic purple bacteria: long range electron transfer in the femtosecond domain at low temperature”, w: *The photosynthetic bacterial reaction center*, red. J. Breton, A. Vermeglio, 195-203, Plenum Publishing Corporation, 1988.
- [43] M. Bixon, J. Jortner, „Electron transfer – from isolated molecules to biomolecules”, *Adv. Chem. Phys.* **106**, 35-201 (1999).
- [44] R. A. Marcus, „Tutorial on rate constants and reorganization energies”, *J. Electroanal. Chem.* **483**, 2-6 (2000).
- [45] L. Piel, *Idee chemii kwantowej*, Wydawnictwo Naukowe PWN, 2003.
- [46] A. J. Hoff, J. Deisenhofer, „Photophysics of photosynthesis. Structure and spectroscopy of reaction centers of purple bacteria”, *Phys. Reports* **287**, 1-247 (1997).
- [47] H. Wang, S. Lin, J. P. Allen, J. C. Williams, S. Blankert, C. Laser, N. W. Woodbury, „Protein dynamics control the kinetics of initial electron transfer in photosynthesis”, *Science* **316**, 747-750 (2007).
- [48] H. Wang, S. Lin, E. Katilius, C. Laser, J. P. Allen, J. C. Williams, N. W. Woodbury, „Unusual temperature dependence of photosynthetic electron transfer due to protein dynamics”, *J. Phys. Chem. B* **113**, 818-824 (2009).

- [49] R. Zwanzig, „Diffusion in a rough potential”, *Proc. Natl. Acad. Sci. USA* **85**, 2029-2030 (1988).
- [50] M. Abramowitz, I. A. Stegun, *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, Dover Publications, Inc., 1972.
- [51] A. Y. Mulikidjanian, „Ubiquinol oxidation in the cytochrome bc_1 complex: Reaction mechanism and prevention of short-circuiting”, *Biochim. Biophys. Acta* **1709**, 5-34 (2005).
- [52] A. R. Crofts, „The cytochrome bc_1 complex: Function in the context of structure”, *Annu. Rev. Physiol.* **66**, 689-733 (2004).
- [53] A. Osyczka, C. C. Moser, F. Daldal, P. L. Dutton „Reversible redox energy coupling in electron transfer chains”, *Nature* **427**, 607-612 (2004).
- [54] M. Kurzyński, praca w przygotowaniu.
- [55] D. Cyranoski, „Swimming against the tide”, *Nature* **408**, 764-766 (2000).
- [56] C. Kettner, A. Bertl, G. Obermeyer, C. Slayman, H. Bihler, „Electrophysiological analysis of the yeast V-type proton pump: variable coupling ratio and proton shunt”, *Biophys. J.* **85**, 3730-3738 (2003).
- [57] R. J. Baxter, *Exactly solved models in statistical mechanics*, Academic Press, 1982.

Spis tabel

2.1	Rzeczywisty czas obliczeń (w sekundach) T_o oraz błąd procentowy wyznaczenia $\Delta_{\%}\tau_M$ średniego czasu pierwszego przejścia τ_M w zależności od efektywnej liczby wędrówców (rozumianej jako iloczyn liczby wędrówców i liczby symulacji z ich udziałem) N_w na hipersześcianie o wymiarze $d = 5$	28
2.2	Podstawowe funkcje MPI.	30
3.1	Parametry wejściowe symulacji numerycznych dynamiki wibracyjnej. . .	49
3.2	Związek pomiędzy wybranymi wartościami f a $\beta\omega$ i $(\beta\omega)^{-1}$	50
4.1	Parametry wejściowe symulacji numerycznych dynamiki przejść konformacyjnych.	65
4.2	Procentowe błędy $\Delta_{\%}$ wyznaczenia wartości strumienia J_1 i J_2 oraz stopnia sprzężenia ϵ w zależności od iloczynu liczby wędrówców N_w i liczby kroków N_k . Wyniki przedstawione w dwóch ostatnich wierszach pokazują, że w przypadku zbyt krótkiej ścieżki mogą pojawić się efekty wyboru stanu początkowego.	72
4.3	Zależność średniego czasu pierwszego przejścia $\tau_M(a \rightarrow b)$ od odległości pomiędzy stanami a i b dla hipersześcianu o wymiarze $d = 5$. Błąd wyznaczenia $\Delta\tau_M = 0,001$	74
4.4	Możliwe stany końcowe w odległościach $r_{a \rightarrow b}$ od 1 do 5 dla hipersześcianu o wymiarze $d = 5$ gdy stanem początkowym jest 0. Po zapisaniu numerów stanów w systemie dwójkowym stanie się widoczne, że odległość oznacza na ilu miejscach bity są różne.	74
4.5	Zależność średniego czasu pierwszego przejścia $\tau_M(a \rightarrow \{b, c\})$ dla różnych odległości $r_{a \rightarrow b}$ i $r_{a \rightarrow c}$ przy różnych możliwych odległościach $r_{b \rightarrow c}$ (podanych w nawiasach kwadratowych w odpowiednich komórkach) dla hipersześcianu o wymiarze $d = 5$. a , b i c są różnymi stanami. Błąd wyznaczenia $\Delta\tau_M = 0,001$	75

4.6	Wpływ parametrów o i r na stopień sprzężenia $\epsilon(0)$ przy założeniu maksymalnej możliwej odległości po przekątnej między stanami $1'$ i $2'$ oraz $1''$ i $2''$. Parametry symulacji: $N_w = 10^2$, $N_k = 10^7$, $a_1 = 10$, $a_2 = 0$, $v_1/p = 50$, $v_2/p = 30$, $p = 1,82 \cdot 10^{-2}$, $v_1 \exp(-a_1) = 4,13 \cdot 10^{-5}$, $v_2 \exp(-a_2) = 0,545$. Błąd wyznaczenia $\Delta\epsilon(0) = 0,001$	78
4.7	Wpływ prawdopodobieństwa przejścia zewnętrznego na jednostkę czasu v_1/p na siłę blokującą a_2^{st} przy ustalonym $v_2/p = 30$ oraz wpływ prawdopodobieństwa przejścia zewnętrznego na jednostkę czasu v_2/p na stopień sprzężenia $\epsilon(0)$ przy ustalonym $v_1/p = 50$. Wartości a_2^{st} zostały wyznaczone z błędem $0,01$, a wartości $\epsilon(0)$ z błędem $0,001$	80
4.8	Wyznaczone wartości siły blokującej $a_2^{st}(\infty)$ i stopnia sprzężenia $\epsilon(0)$ dla modelu z jednym procesem dostarczającym i jednym odbierającym energię swobodną oraz dla modeli rozszerzonych o dodatkowe procesy. Błędy wyznaczania wielkości identyczne jak dla wcześniej przedstawianych obliczeń $a_2^{st}(\infty)$ i $\epsilon(0)$	90
4.9	Niejednoznaczność zależności średniego czasu pierwszego przejścia τ_M od odległości chemicznej i wyboru stanu początkowego na trójkącie Sierpińskiego. Badano trójkąt trzeciego rzędu (rysunek 2.8) i maksymalną odległość pomiędzy stanami $r = 7$	91
4.10	Wartość stopnia sprzężenia $\epsilon(0)$ w zależności od wyboru geometrii stanów i liczby bram na trójkącie Sierpińskiego trzeciego rzędu. W tabeli podano numery stanów będące bramami zgodne z rysunkiem 2.8. Z tabeli wynika też liczba procesów dostarczających (cyfra 1) i odbierających (cyfra 2) energię swobodną. Wartości podano z błędem wyznaczenia $\Delta\epsilon(0) = 0,001$	92

Spis rysunków

1.1	Wibracje a przejścia konformacyjne – schemat. Za pracą [10].	8
1.2	Przykład modelu odwracalnej jednocząsteczkowej reakcji chemicznej pomiędzy stanami R (po lewej) i P (po prawej). Za pracą [8].	9
2.1	Symulacja, teoria a eksperyment – schemat. Według książki [13].	14
2.2	Przykład sieci liniowej z zaznaczonymi stanami. Skoro stany numerujemy od 0, to można zdefiniować wirtualny stan -1 z warunkiem odbijającym. Natomiast w prawo sieć można rozbudowywać w nieskończoność.	17
2.3	Sieć typu hipersześcianu o wymiarach od 1 do 3 wraz z zaznaczonymi numerami stanów (numeracja od 0).	18
2.4	Trójkąt Sierpińskiego rzędu 0, 1, 2 oraz 3. W wierzchołkach nie zaznaczono numerów stanów. Sposób generowania opisano w treści rozdziału.	19
2.5	Trójkąt Sierpińskiego rzędu 0 z zaznaczonymi stanami w wierzchołkach. Stany numerowane są od 0.	20
2.6	Trójkąt Sierpińskiego rzędu 1 z zaznaczonymi stanami w wierzchołkach. Stany numerowane są od 0.	20
2.7	Trójkąt Sierpińskiego rzędu 2 z zaznaczonymi stanami w wierzchołkach. Stany numerowane są od 0.	21
2.8	Trójkąt Sierpińskiego rzędu 3 z zaznaczonymi stanami w wierzchołkach. Stany numerowane są od 0.	21
2.9	Test generatora liczb pseudolosowych. Pary wylosowanych liczb przedstawiono na wykresie. Po lewej stronie generator G1 , po prawej generator G2 (patrz tekst).	25
2.10	Test generatora liczb pseudolosowych. Zależność średniego kwadratu odległości od czasu przedstawiona w skali logarytmicznej. Kropkami zaznaczono wyniki symulacji, a linią ciągłą najlepsze dopasowanie do wyników symulacji.	26

2.11	Zależność średniego czasu pierwszego przejścia τ_M wyrażonego w liczbie kroków na hipersześcianie o wymiarze $d = 10$ od odległości r pomiędzy stanem początkowym i końcowym. Warto zauważyć, że skala dla wartości τ_M nie zaczyna się od 0 oraz, że dla $r = 8$ wystąpiły niewielkie fluktuacje, które można zmniejszyć przez powiększenie liczby wędrówców.	27
2.12	Zależność średniego czasu pierwszego przejścia τ_M wyrażonego w liczbie kroków od wymiaru d hipersześcianu dla odległości pomiędzy stanem początkowym i końcowym $r = 1$.	27
2.13	Przyspieszenie S_N obliczania średniego czasu pierwszego przejścia w zależności od liczby procesorów N . Linia ciągłą zaznaczono najlepszy możliwy wynik teoretyczny. Procesory Intel® Xeon® 2.8 GHz z technologią Hyper-Threading. Za pracą [A].	39
2.14	Wydajność W_N obliczania średniego czasu pierwszego przejścia w zależności od liczby procesorów N . Linia ciągłą zaznaczono najlepszy możliwy wynik teoretyczny. Procesory Intel® Xeon® 2.8 GHz z technologią Hyper-Threading. Za pracą [A].	39
3.1	Redukcja problemu wielowymiarowego do pojedynczej współrzędnej reakcji q . Dwie parabole reprezentują energie wibracyjne stanów elektronowych substratu i produktu. Zaznaczono stany wibracyjne oraz charakterystyczne wielkości $Q_0, \epsilon, \lambda, \omega$.	43
3.2	Ogólny model nieodwracalnej jednocząsteczkowej reakcji chemicznej. Kółkami zaznaczono podstany (w tym przypadku wibracyjne) stanu substratu R, a gwiazdką stan <i>limbo</i> . Za pracą [10].	45
3.3	Zakres stosowalności przybliżenia (3.19) dla $\beta\epsilon^\ddagger \gg 1$ oraz przybliżenia (3.20) dla $\beta\epsilon^\ddagger \ll 1$ (linie przerywane) równania (3.18) dla wybranych wysokości bariery energetycznej $\epsilon^\ddagger\omega^{-1}$. Za pracą [B].	48
3.4	Schemat sieci liniowej (zaznaczono 10 stanów), na której zachodzi przypadkowe błędzenie w energii e . Po lewej stronie sieć ze stanem-bramką, obok której układ może przejść nie wpadając w <i>limbo</i> . Po prawej stronie sieć z całkowicie absorbującym stanem końcowym, którego osiągnięcie powoduje wpadnięcie w <i>limbo</i> .	50
3.5	Zależność logarytmu średniego czasu pierwszego przejścia $\tau(0 \rightarrow \epsilon^\ddagger)$ (w jednostkach κ^{-1}) od temperatury (w jednostkach ω) dla różnych wysokości bariery energetycznej $\epsilon^\ddagger\omega^{-1}$. Kropki oznaczają wyniki symulacji, a krzywe oznaczone liniami ciągłymi dane są równaniem (3.18). Za pracą [B].	52

3.6	Zależność logarytmu średniego czasu pierwszego przejścia $\tau(e_0 \rightarrow \epsilon^\dagger)$ (w jednostkach κ^{-1}) od temperatury (w jednostkach ω) dla różnych początkowych poziomów wibracyjnych $e_0\omega^{-1}$ znajdujących się ponad stałą barierą energetyczną $\epsilon^\dagger\omega^{-1} = 5$. Kropki oznaczają wyniki symulacji, a krzywe oznaczone liniami ciągłymi dane są równaniem (3.23). Za pracą [B].	53
3.7	Zależność temperaturowa szybkości pierwotnego przeniesienia elektronu w bakteryjnych fotosyntetycznych centrach reakcji. Dopasowano wyniki symulacji komputerowych (linie ciągłe; za pracą [B]) do danych doświadczalnych (otwarte kółka <i>Rhodobacter sphaeroides</i> , zamknięte <i>Rhodopseudomonas viridis</i> ; za pracą [41]).	55
4.1	Schemat maszyny chemochemicznej sprzęgającej dwa procesy: dostarczający i odbierający energię swobodną. Po lewej stronie we wnętrzu szarej skrzynki wyróżnione są pojedyncze bramy, po prawej stronie przedstawiono rozszerzenie na wiele bram. Za pracą [C].	59
4.2	Schematyczna zależność strumienia J_i od siły A_i . Zaznaczono charakterystyczne parametry J_{+i} , J_{-i} oraz A_i^{st} . Pogrubiona linia oznacza obszar przetwarzania energii swobodnej. Za pracą [10].	60
4.3	Przykład realizacji modelu przejść konformacyjnych na sześciianie z jednym procesem dostarczającym energię swobodną (z prawdopodobieństwem v_1) i dwoma odbierającymi (z prawdopodobieństwem v_2). Strzałkami zaznaczono kierunek przejść zewnętrznych „do przodu”. Nie zostały zaznaczone przejścia zewnętrzne „do tyłu”.	66
4.4	Ogólna geometria bram dla modelu z jednym procesem dostarczającym i jednym odbierającym energię swobodną. Zaznaczono charakterystyczne odległości r i o oraz ustalony kierunek przejść zewnętrznych „do przodu”. Nie zostały zaznaczone przejścia zewnętrzne „do tyłu” określone przez czynniki $\exp(-a_1)$ oraz $\exp(-a_2)$.	67
4.5	Dwie przykładowe zależności liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ od czasu wyrażonego w liczbie kroków N_k dla dwóch ewolucji pojedynczego wędrowca na pięciowymiarowym hipersześciianie. Parametry użyte w symulacji: $N_{P1} = 1$, $N_{P2} = 1$, $v_1/p = 50$, $v_2/p = 30$, $p = 1,82 \cdot 10^{-2}$, $a_1 = 10$, $a_2 = 5,5$, $r = 1$, $o = 5$, $v_1 \exp(-a_1) = 4,13 \cdot 10^{-5}$, $v_2 \exp(-a_2) = 2,23 \cdot 10^{-3}$.	68

- 4.6 Przykładowe zależności liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ od czasu wyrażonego w liczbie kroków N_k dla ewolucji pojedynczego wędrówca na pięciowymiarowym hipersześcianie. Poprzez zmianę siły a_1 uzyskano efekt odwracalności procesu. Linia przerywaną zaznaczono symulację z parametrami identycznymi jak na rysunku 4.5. W kolejnych dwóch symulacjach (linia ciągła) zmieniono wartość siły napędzającej na $a_1 = 1$ ($v_1 \exp(-a_1) = 0,334$) oraz $a_1 = 0,1$ ($v_1 \exp(-a_1) = 0,823$). Już dla $a_1 = 0,1$ widoczny jest efekt głównie cofania się procesu (dolna krzywa), chociaż w dłuższym czasie proces średnio przebiega do przodu. 69
- 4.7 Zależność strumienia J_2 od a_2 dla dwóch symulacji **S1** (czarne kropki) i **S2** (puste kwadraciki). Parametry użyte w symulacji: $N_w = 10^2$, $N_k = 10^6$, $N_{P1} = 1$, $N_{P2} = 1$, $a_1 = 10$, a_2 w przedziale od -1 do 5 , $r = 1$, $o = 5$. W obu symulacjach stosunek $v_1/p = 5$, a $v_2/p = 1$ 71
- 4.8 Zależność strumienia J_2 od siły napędzającej a_1 . Widoczne są drobne fluktuacje występujące w obszarze nasycenia. 80
- 4.9 Zależność strumienia J_2 od sprzężonej z nim siły a_2 dla przypadku jednego procesu odbierającego i jednego dostarczającego energię swobodną. Wyniki symulacji zaznaczono kropkami, natomiast linia ciągła dana jest równaniem (4.25). 82
- 4.10 Zależność stopnia sprzężenia ϵ od siły a_2 dla przypadku jednego procesu dostarczającego i jednego odbierającego energię swobodną. Wyniki symulacji zaznaczono kropkami, natomiast linia ciągła dana jest równaniem (4.6). 85
- 4.11 Zależność siły blokującej a_2^{st} od siły a_1 dla przypadku jednego procesu dostarczającego i jednego procesu odbierającego energię swobodną. Wyniki symulacji zaznaczono kropkami, natomiast linia ciągła dana jest równaniem (4.7). 86
- 4.12 Wybrany sposób rozszerzenia geometrii z rysunku 4.4 na model z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Zaznaczono charakterystyczne odległości oraz ustalony kierunek przejść zewnętrznych „do przodu”. Na górze ogólny schemat, na dole używany przykład realizacji z numerami stanów i wartościami odległości. Nie zostały zaznaczone przejścia zewnętrzne „do tyłu” określone przez czynniki $\exp(-a_1)$ oraz $\exp(-a_2)$. Odległość $r_{2'_a \rightarrow 2''_b} = r_{2''_b \rightarrow 2'_a} = 1$ 87

- 4.13 Wybrany sposób rozszerzenia geometrii z rysunku 4.4 na model z dwoma procesami dostarczającymi i dwoma odbierającymi energię swobodną. Zaznaczono charakterystyczne odległości oraz ustalony kierunek przejść zewnętrznych „do przodu”. Na górze ogólny schemat, na dole używany przykład realizacji z numerami stanów i wartościami odległości. Nie zostały zaznaczone przejścia zewnętrzne „do tyłu” określone przez czynniki $\exp(-a_1)$ oraz $\exp(-a_2)$. Odległość $r_{2'_a \rightarrow 1''_b} = r_{1''_b \rightarrow 2'_a} = r_{1''_a \rightarrow 2'_b} = r_{2'_b \rightarrow 1''_a} = 1$ 88
- 4.14 Zależność stopnia sprzężenia ϵ od siły a_2 dla obszaru obejmującego przetwarzanie energii swobodnej dla modelu typu $N_{P1} = 1, N_{P2} = 1$ (kółka), $N_{P1} = 1, N_{P2} = 2$ (kwadraty) oraz $N_{P1} = 2, N_{P2} = 2$ (trójkąty). 89
- 4.15 Trójkąt Sierpińskiego czwartego rzędu, na którym zrealizowano symulację z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Dla przedstawionej geometrii bram otrzymano stopień sprzężenia $\epsilon(0) = 1, 271$ 92
- 4.16 Sieć Bethgo z trzema powłokami, na której zrealizowano symulację z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Odległość pomiędzy parą bram dla danego procesu wynosi $o = 6$, a odległość pomiędzy najbliższymi bramami $r = 2$ 93
- 4.17 Sieć Bethgo z czterema powłokami, na której zrealizowano symulację z jednym procesem dostarczającym i dwoma odbierającymi energię swobodną. Odległość pomiędzy parą bram dla danego procesu wynosi $o = 8$, a odległość pomiędzy najbliższymi bramami $r = 2$ 93
- 4.18 Zależność liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ (linia ciągła) i liczby n_2 przejść zewnętrznych $R_2 \leftrightarrow P_2$ (linia przerywana) od czasu wyrażonego w liczbie kroków N_k dla ewolucji pojedynczego wędrowca na pięciowymiarowym hipersześcianie. Na górnym rysunku przedstawiono 10^9 kroków „fotografowanych” co 10^5 kroków. Na dolnym rysunku przedstawiono pierwsze 10000 kroków. Parametry użyte w symulacji: $N_{P1} = 1, N_{P2} = 2, a_1 = 10, a_2 = 0, v_1/p = 50, v_2/p = 30, p = 1, 82 \cdot 10^{-2}$. Dla tego przypadku $\epsilon(0) = 0, 362$ 95

- 4.19 Zależność liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ (linia ciągła) i liczby n_2 przejść zewnętrznych $R_2 \leftrightarrow P_2$ (linia przerywana) od czasu wyrażonego w liczbie kroków N_k dla ewolucji pojedynczego wędrowca na trójkącie Sierpińskiego czwartego rzędu. Na górnym rysunku przedstawiono 10^9 kroków „fotografowanych” co 10^5 kroków. Na dolnym rysunku przedstawiono pierwsze 100000 kroków. Zatem dolny rysunek reprezentowany jest na górnym rysunku tylko przez dwa punkty. Parametry użyte w symulacji: $N_{P1} = 1$, $N_{P2} = 2$, $a_1 = 10$, $a_2 = 0$, $v_1/p = 50$, $v_2/p = 30$, $p = 1,85 \cdot 10^{-2}$. Dla tego przypadku $\epsilon(0) = 1,271$ 96
- 4.20 Zależność liczby n_1 przejść zewnętrznych $R_1 \leftrightarrow P_1$ (linia ciągła) i liczby n_2 przejść zewnętrznych $R_2 \leftrightarrow P_2$ (linia przerywana) od czasu wyrażonego w liczbie kroków N_k dla ewolucji pojedynczego wędrowca na sieci Bethgo z 46 stanami. Na górnym rysunku przedstawiono 10^9 kroków „fotografowanych” co 10^5 kroków. Na dolnym rysunku przedstawiono pierwsze 100000 kroków. Zatem dolny rysunek reprezentowany jest na górnym rysunku tylko przez dwa punkty. Parametry użyte w symulacji: $N_{P1} = 1$, $N_{P2} = 2$, $a_1 = 10$, $a_2 = 0$, $v_1/p = 50$, $v_2/p = 30$, $1,89 \cdot 10^{-2}$. Dla tego przypadku $\epsilon(0) = 1,193$ 97