

Uniwersytet Warszawski
Wydział Matematyki, Informatyki i Mechaniki

Jan Wróblewski

Nr albumu: 277632

**Porównanie symulacji automatu
komórkowego z dyskretyzowanym
równaniem reakcji-dyfuzji**

**Praca magisterska
na kierunku MATEMATYKA
w zakresie MATEMATYKI STOSOWANEJ**

Praca wykonana pod kierunkiem
prof. Piotra Rybki
Instytut Matematyki Stosowanej i Mechaniki

5 czerwca 2013

Oświadczenie kierującego pracą

Potwierdzam, że niniejsza praca została przygotowana pod moim kierunkiem i kwalifikuje się do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data

Podpis kierującego pracą

Oświadczenie autora (autorów) pracy

Świadom odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.

Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w wyższej uczelni.

Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

Data

Podpis autora (autorów) pracy

Streszczenie

Praca zawiera analizę i porównanie dwóch metod symulacji komputerowych równań reakcji-dyfuzji. Pierwszą metodą jest dyskretyzacja po czasie zamkniętym schematem Eulera i po przestrzeni metodą elementu skończonego, a drugą jest użycie automatu komórkowego. Metody te posiadają zasadniczo różny sposób obliczeń, ale w wielu sytuacjach dają podobne wyniki symulacji. W pracy dokonano obserwacji zjawisk zachodzących w obu przypadkach (m. in. potoku średniokrzywiznowego), podano przypadki odróżniające jakościowo te dwie metody oraz dokonano próby sprowadzenia rozwiązań do wspólnej przestrzeni i porównania ich za pomocą normy.

Słowa kluczowe

automaty komórkowe, symulacje numeryczne, porównanie metod symulacji, równanie reakcji-dyfuzji

Dziedzina pracy (kody wg programu Socrates-Erasmus)

11.1 Matematyka

Klasyfikacja tematyczna

35K57. Reaction-diffusion equations

35Q99. Equations of mathematical physics and other areas of application

65M60. Finite elements, Rayleigh-Ritz and Galerkin methods, finite methods

68Q80. Cellular automata

Tytuł pracy w języku angielskim

Comparing a cellular automaton with a discretized reaction-diffusion equation

Spis treści

| | |
|---|----|
| Wprowadzenie | 7 |
| 1. Równania stechiometryczne | 11 |
| 1.1. Model 3-cząsteczkowy | 12 |
| 1.2. Model 2,5-cząsteczkowy | 12 |
| 1.3. Model 2-cząsteczkowy | 13 |
| 2. Modele matematyczne reakcji-dyfuzji | 15 |
| 2.1. Równanie reakcji | 15 |
| 2.1.1. Tworzenie na podstawie równań stechiometrycznych | 15 |
| 2.1.2. Model 3-cząsteczkowy | 16 |
| 2.1.3. Model 2,5-cząsteczkowy | 19 |
| 2.1.4. Model 2-cząsteczkowy | 19 |
| 2.2. Równanie reakcji-dyfuzji | 19 |
| 2.2.1. Model 2-cząsteczkowy | 20 |
| 2.2.2. Zasada zachowania masy | 20 |
| 2.2.3. Równanie dyfuzji dla sumy mas | 20 |
| 2.2.4. Nieujemność rozwiązania | 20 |
| 2.3. Skalowanie | 22 |
| 2.4. Istnienie i jednoznaczność rozwiązania | 22 |
| 2.5. Zbiory niezmiennicze i implikacje | 22 |
| 2.5.1. Teoria | 22 |
| 2.5.2. Zastosowanie do modelu 3-cząsteczkowego | 23 |
| 3. Metody numeryczne | 25 |
| 3.1. Opis schematu | 25 |
| 3.1.1. Aproksymacja metodą predictor-corrector | 26 |
| 3.1.2. Aproksymacja metodą Newtona | 26 |
| 3.1.3. Oszacowanie błędu metody Newtona | 27 |
| 3.2. Użyte narzędzia | 28 |
| 3.3. Przestrzeń funkcji próbnych | 28 |
| 3.3.1. Siatka | 28 |
| 3.3.2. Funkcje kawałkami wielomianowe drugiego stopnia | 28 |
| 3.4. Rozwiązywanie układu równań liniowych | 29 |
| 3.5. Warunki stopu metod iteracyjnych | 29 |
| 3.6. Oszacowanie błędu | 29 |

| | |
|--|----|
| 4. Automaty komórkowe | 31 |
| 4.1. Wstęp | 31 |
| 4.2. Definicja modelu | 33 |
| 4.3. Analiza modelu | 33 |
| 4.3.1. Model 3-cząsteczkowy | 34 |
| 4.3.2. Model 2,5-cząsteczkowy | 34 |
| 4.3.3. Model 2-cząsteczkowy | 34 |
| 4.4. Możliwe modyfikacje | 34 |
| 4.4.1. Warunki brzegowe | 34 |
| 4.4.2. Reakcja | 35 |
| 4.4.3. Dyfuzja | 35 |
| 5. Wyniki symulacji i porównanie | 37 |
| 5.1. Porównanie w przypadku reakcji Bielousowa-Żabotyńskiego | 37 |
| 5.2. Symulacje numeryczne | 37 |
| 5.2.1. Parametry | 38 |
| 5.2.2. Zbierane dane | 40 |
| 5.3. Symulacje automatami komórkowymi | 41 |
| 5.3.1. Parametry | 42 |
| 5.3.2. Warunki początkowe | 42 |
| 5.3.3. Warunek stopu | 42 |
| 5.3.4. Animacja kolorów cząsteczek | 42 |
| 5.3.5. Animacja rozkładu masy | 42 |
| 5.4. Symulacje porównawcze | 43 |
| 5.4.1. Sprowadzanie do wspólnej przestrzeni | 43 |
| 5.4.2. Sposób porównania | 44 |
| 5.4.3. Stan początkowy dla automatu komórkowego | 45 |
| 5.4.4. Parametry | 45 |
| 5.5. Obserwacje | 46 |
| 5.5.1. Fronty | 46 |
| 5.5.2. Zachowanie asymptotyczne | 47 |
| 5.5.3. Potok średniokrzywiznowy | 48 |
| 5.5.4. Norma odległości | 50 |
| 5.5.5. Przypadki odróżniające modele | 57 |
| 5.5.6. Grubość frontu | 59 |
| 5.5.7. Rozkład masy | 59 |
| 5.5.8. Dokładność obliczeń numerycznych | 60 |
| 5.5.9. Zmodyfikowana dyfuzja | 64 |
| 6. Szczegóły techniczne | 65 |
| 6.1. Wymagane oprogramowanie | 65 |
| 6.2. Uruchamianie symulacji | 65 |
| 6.3. Komentarz do kodu symulacji numerycznych | 66 |
| 6.4. Komentarz do kodu automatów komórkowych | 67 |
| 6.5. Komentarz do kodu porównującego symulacje numeryczne i automaty komórkowe | 68 |
| 6.6. Generator liczb pseudolosowych i determinizacja wyników | 69 |
| 6.7. Tworzenie animacji z pojedynczych klatek | 69 |
| 6.8. Zawartość dołączonej płyty DVD | 70 |

| | |
|----------------------------------|----|
| 7. Podsumowanie | 75 |
| Bibliografia | 79 |

Wprowadzenie

W pracy porównujemy ze sobą dwa różne sposoby symulacji modelu interakcji cząstek aktywnych optycznie. Pierwszą metodą jest numeryczne rozwiązywanie układu równań różniczkowych cząstkowych reakcji-dyfuzji. Drugą są symulacje za pomocą opisanych w dalszej części wstępu automatów komórkowych. Niniejsza praca magisterska jest w pewnym sensie kontynuacją prac magisterskich Pawła Koniecznego [7] i Magdaleny Włodarczyk [12], stworzonych również na wydziale MIMUW (Matematyki, Informatyki i Mechaniki Uniwersytetu Warszawskiego). Praca [7] jest na temat porównania automatów komórkowych (opisanych pokrótce dalej we wstępie i szerzej w rozdziale 4) i równań różniczkowych cząstkowych dla reakcji Biełousowa-Żabotyńskiego. Wynikiem tej pracy jest stwierdzenie podobnych wizualnie wyników, lecz najprawdopodobniej brak jakiegokolwiek równoważności tych modeli. Podobne wyniki możemy zaobserwować w tej pracy. Praca [12] zawiera implementację trójwymiarowego automatu komórkowego dla konkretnego modelu reakcji-dyfuzji, występującego również w tej pracy jako model 3-cząsteczkowy (opisany w sekcji 1.1 i w rozdziale 4). Implementacja wykorzystywana w tej pracy jest jednak różna - dwuwymiarowa i dostosowana do badania podobieństwa omawianych dwóch metod symulacji.

Interakcje cząstek aktywnych optycznie, o których mowa w pracy, są antagonistyczne, tzn. obecność cząstek obu typów powoduje zmniejszenie ilości cząstek co najmniej jednego z rodzajów. Ich badanie może pomóc w odpowiedzi na pytanie dlaczego w przyrodzie mieszaniny substancji organicznych z reguły nie są racemiczne (tzn. posiadające równe ilości molowe prawo- i lewoskrętnych cząstek danego związku chemicznego). Na przykład w żywych organizmach występują najczęściej lewoskrętne cukry i wyłącznie prawoskrętne aminokwasy. Jak się okazuje, cząsteczki danej substancji chemicznej o innej skrętności niż występujące w naturze są często nieprzyswajalne albo nawet toksyczne dla organizmów. Jak można przeczytać w [10] bądź [3], odpowiedź na pytanie dlaczego organiczne mieszaniny związków aktywnych optycznie nie są racemiczne mogłaby dostarczyć informacji na temat procesu powstania życia na Ziemi.

Interakcje pomiędzy cząsteczkami aktywnymi optycznie modelujemy w tej pracy za pomocą równań stechiometrycznych. W pracy zakładamy, że równania te są symetryczne ze względu na cząsteczki prawo- i lewoskrętne. Użyte przez nas modele zostały opisane w rozdziale 1. Przypisujemy cząsteczkom prawo- i lewoskrętnym różne kolory - niebieski (N) i czerwony (C). Jeżeli w modelu występują cząsteczki neutralne, to przypisujemy im kolor biały (B). Używając kolorów B , N i C , określamy reakcje zachodzące pomiędzy cząsteczkami oraz sposób ich zachodzenia. W pracy przyjmujemy, że wszystkie cząsteczki mają taką samą masę i zakładamy zasadę zachowania masy. Reakcje pomiędzy substancjami symulujemy na dwa sposoby: za pomocą numerycznego rozwiązywania równań różniczkowych cząstkowych (opisanego w rozdziale 3) oraz za pomocą automatu komórkowego (opisanego w rozdziale 4). Źródłem podstawowych informacji na temat numerycznego rozwiązywania równań różniczkowych cząstkowych jest na przykład [2]. Automaty komórkowe zostały omówione w [5]. W pracy, jeżeli nie podamy inaczej, równania różniczkowe oznaczają używane w naszych

modelach układu równań różniczkowych cząstkowych, a automaty komórkowe dotyczą tylko rozważanych przez nas w rozdziale 4 automatów komórkowych.

W przypadku równań różniczkowych tworzymy następujący układ równań reakcji-dyfuzji:

$$\begin{cases} u_t - D\Delta u = f(u), & \text{w } \Omega, \\ \frac{\partial u}{\partial n} = 0, & \text{na } \partial\Omega, \\ u(0, x) = u_0(x), & x \in \Omega. \end{cases}$$

$$u = \begin{pmatrix} b \\ n \\ c \end{pmatrix}, \quad D = \text{diag}(d_b, d_n, d_c), \quad u_0 = \begin{pmatrix} b_0 \\ n_0 \\ c_0 \end{pmatrix},$$

gdzie b , n , c to gęstości cząstek odpowiednich kolorów, d_x odpowiednimi współczynnikami dyfuzji. Przez laplasjan wektora rozumiemy wektor laplasjanów poszczególnych jego składowych. Cząsteczki każdego rodzaju dyfundują tylko wewnątrz siebie (macierz dyfuzji jest diagonalna). f jest prawą stroną równania reakcji, określającą sposób i szybkość zamiany poszczególnych cząstek na inne. Dokładna postać funkcji f odpowiada użytemu modelowi. W rozdziale 2 podane są funkcje f dla występujących w pracy modeli oraz bardziej szczegółowy opis i analiza dla jednego z nich. O ile nie zostało podane inaczej, w dalszej części zakładamy warunki brzegowe Neumanna $\frac{\partial u}{\partial n} = 0$ dla omawianych równań różniczkowych.

W przypadku automatu komórkowego określamy lokalnie w przestrzeni sposób zmiany stanów komórek (czyli wektory ilości cząstek poszczególnych rodzajów) w poszczególnych krokach czasowych i symulujemy w ten sposób przebieg procesu reakcji-dyfuzji. Automat komórkowy posiada przestrzeń w postaci siatki złożonej z pojedynczych komórek. W każdej komórce może przebywać całkowita nieujemna ilość cząstek poszczególnych kolorów. Definiujemy funkcję przejścia określającą stan komórki wyłącznie na podstawie stanów komórek z bezpośredniego sąsiedztwa. W tej pracy używamy definicji sąsiedztwa o kształcie plusa zawartego w kwadracie 3×3 komórek. Do symulacji dyfuzji używamy losowego przemieszczania się cząstek do jednej z sąsiednich komórek albo pozostawiania w miejscu. Do symulacji reakcji wybieramy losowo cząsteczki-substraty i zamieniamy je na cząsteczki-produkty. Po wygenerowaniu początkowego stanu automatu komórkowego jesteśmy w stanie użyć funkcji przejścia do wygenerowania jego stanu dla kolejnych kroków czasowych. Warunki brzegowe można pominąć tworząc przestrzeń w postaci torusa, tzn. sklejając przeciwległe brzegi siatki. Można również stworzyć odpowiedniki warunków brzegowych Dirichleta i Neumanna. Wszystko to opisujemy dokładniej w rozdziale 4. Automaty komórkowe są efektywną metodą symulacji, ale trudną do analizy teoretycznej, w szczególności dotyczącej globalnego zachowania układu. Zdefiniowane w ten sposób są tak naprawdę złożonymi procesami stochastycznymi.

W tej pracy porównujemy z sobą te dwa typy symulacji. Ze względu na ich różne własności, osiągnięte wyniki na ogół będą się różniły. Ważnymi różnicami pomiędzy równaniami różniczkowymi a automatami komórkowymi są między innymi:

- Różne przestrzenie - przestrzeń Euklidesowa \mathbb{R}^n w przypadku równań różniczkowych i skończona przestrzeń z metryką miejską w przypadku automatów komórkowych;
- Różna lokalność oddziaływań - zasięg oddziaływania cząstek w przypadku dyfuzji (laplasjanu) jest nieskończony, podczas gdy zasięg oddziaływania cząstek w automacie komórkowym jest ściśle ograniczony do sąsiedztwa pojedynczej komórki. Możemy to wyraźnie zaobserwować również w wynikach symulacji. W przypadku automatu komórkowego stan danego punktu w przestrzeni w t -tym kroku czasowym zależy wyłącznie

od punktów odległych o nie więcej niż t . W przypadku równań różniczkowych wartość $u(t, x)$ zależy od wartości $u_0(x)$ dla x należących do całej przestrzeni;

- Różne mechanizmy - symulacje numeryczne równań różniczkowych z jednoznacznym rozwiązaniem są deterministyczne, a naszych automatów komórkowych niedeterministyczne. Automaty komórkowe są w tym względzie bardziej podobne do dyskretnych procesów stochastycznych. Losowość ma w nich istotny wpływ na przebieg symulacji, tym większy im bardziej lokalne własności są rozpatrywane.

Ze względu na różny charakter przestrzeni oraz różne właściwości tych metod, zadanie ich porównania wymaga stworzenia odpowiednich metod. Stosowanym u nas podejściem jest stworzenie pewnej dyskretnej przestrzeni Ξ , sprowadzenie do niej stanów automatu komórkowego i wyników symulacji numerycznej, a następnie porównanie. Dzięki temu zabiegowi możliwe jest stosowanie norm w przestrzeni Ξ i otrzymywanie liczbowych danych na temat bliskości rozwiązań. Głównym problemem takiego rozwiązania jest konieczność skalibrowania licznych parametrów tak, aby rozwiązania jak najbardziej sobie odpowiadały. Szczegóły przedstawiamy w rozdziale 5.

Przy porównywaniu symulacji numerycznych z symulacjami automatu komórkowego najważniejszą rolę odgrywają wyniki jakościowe. W tym celu w rozdziale 2 została wykonana analiza teoretyczna układu równań reakcji-dyfuzji z zerowymi warunkami brzegowymi Neumanna dla jednego z modeli. Przede wszystkim zostały udowodnione istnienie i jednoznaczność rozwiązania. Poza tym zostały znalezione następujące niezmienniki (oznaczymy przez S sumę gęstości cząstek, $S = b + n + c$ i $S_0 = b_0 + n_0 + c_0$):

- Nieujemność rozwiązania: $b_0, n_0, c_0 \geq 0 \Rightarrow b, n, c \geq 0$;
- Zasada zachowania masy: $\int_{\Omega} S \equiv \int_{\Omega} S_0 = \text{const}$;
- Równanie ciepła dla S , pod warunkiem $d_b = d_n = d_c$: $S_t - \Delta S = 0$. Implikuje to zasadę maksimum i, w połączeniu z nieujemnością rozwiązania, oszacowanie $0 \leq b, n, c \leq \|S_0\|_{L^\infty}$.

Ponadto dla przypadku równych współczynników dyfuzji $d_b = d_n = d_c$, przy użyciu prac [11] i [1], zostały udowodnione twierdzenia na temat zachowania się rozwiązania dla długich czasów. Wyniki symulacji numerycznych rzeczywiście zachowywały wymienione niezmienniki i zachowywały się zgodnie z teorią.

Do dyskretyzacji równań różniczkowych cząstkowych reakcji-dyfuzji została użyta metoda elementu skończonego dla przestrzeni oraz zamknięty schemat Eulera dla czasu. W pracy zbadaliśmy trzy metody dyskretyzacji czasu: otwarty schemat Eulera, zamknięty schemat Eulera i metoda Cranka-Nicolson. Został wybrany zamknięty schemat Eulera ze względu na jego dobre zachowywanie teoretycznych własności modelu. Ze względu na nieliniowość funkcji f , trudno jest użyć wprost zamkniętego schematu Eulera bądź schematu Cranka-Nicolson. Problem dla każdej warstwy czasowej został rozwiązany iteracyjnie. W tym celu zostały użyte metody predictor-corrector (przybliżania funkcji f przez jej wartość z poprzedniej iteracji) oraz metody Newtona (przybliżania funkcji f przez wartość jej i jej gradientu z poprzedniej iteracji). Ze względu na rząd zbieżności została wybrana metoda Newtona. Do metody elementu skończonego została użyta przestrzeń funkcji ciągłych kawałkami wielomianowych drugiego stopnia. Zostały podane oszacowania błędów dla badanych schematów numerycznych. Przy użyciu przestrzeni w normie $L^\infty(0, T; H^1(\Omega))$, kroku czasowym τ i grubości siatki h wynosiły one $O(\tau + h^2)$ dla schematów Eulera i $O(\tau^2 + h^2)$ dla schematu Cranka-Nicolson.

Symulacje numeryczne były przeprowadzane za pomocą programu FreeFem++, a automat komórkowy zasymulowany przez program napisany w C++. Symulacje numeryczne były zwi-
zualizowane za pomocą poziomicy gęstości poszczególnych cząstek w przestrzeni. W przypadku
automatu komórkowego naturalną wizualizacją było przyporządkowanie komórkom pól kwa-
dratowej siatki i użycie kolorów zgodnych z kolorami cząstek wewnątrz do wskazania ich stanu.
Zostały stworzone animacje prezentujące ewolucję układu pod wieloma kątami dla różnych
długości czasów i parametrów. Szczegóły techniczne zostały zawarte w rozdziale 6. Znajdują
się tam opisane kody źródłowe użytych skryptów, programu w FreeFem++ i programu auto-
matu komórkowego. Został tam również opisany sposób obsługi skryptów do automatycznej
generacji i wizualizacji wyników.

Jedną z najważniejszych hipotez badanych eksperymentalnie w tej pracy jest istnienie
potoku średniokrzywiznowego. Potok średniokrzywiznowy to własność zmiennych w czasie
poziomic opisująca się równaniem $\bar{v} = -\kappa\bar{n}$, gdzie \bar{v} jest prędkością poziomicy w danym
punkcie, κ krzywizną, a \bar{n} wektorem normalnym. Posiada on własności zmniejszania miary
brzegu, powodowania zanikania kulistych obszarów z kwadratowo szybko zmniejszającym się
ich promieniem oraz niezmienności dla obszarów płaskich. W dwóch z trzech rozpatrywanych
modeli możemy zaobserwować potok średniokrzywiznowy, zarówno w symulacjach numerycz-
nych jak i w automatach komórkowych. W przypadku równań różniczkowych występuje on
dla poszczególnych poziomicy gęstości każdego z rodzajów cząstek. W przypadku automatów
komórkowych możemy już we wczesnych etapach symulacji zaobserwować grupowanie się czą-
stek w podzbiory ograniczone krzywymi zawierające cząsteczki tylko jednego typu. Krzywe te
zachowują się zgodnie z potokiem średniokrzywiznowym. Wynik ten został dokładniej opisany
w sekcji 5.5.3.

Związany z potokiem średniokrzywiznowym wynikiem jest również brak równoważno-
ści tych dwóch metod symulacji w przypadku modelu 3-cząsteczkowego. Dane początkowe w
postaci pasa na torusie o powierzchni mniejszej niż połowa powierzchni całej przestrzeni w
przypadku symulacji numerycznych równań różniczkowych dały dla długich czasów wynik w
postaci zbliżonego do jednorodnego rozkładu gęstości jednej cząsteczki. W wyniku symulacji
na automacie komórkowym ten pas się utrzymywał. Zachowanie automatu komórkowego w
tym przypadku (biorąc pod uwagę jego budowę opisaną w rozdziale 4) przypominało zagad-
nienie ruiny dwóch graczy o skończonych kapitałach. Dokładniejsze informacje na ten temat
znajdują się w sekcji 5.5.5.

W stworzeniu tej pracy pomogli pracownicy wydziału MIMUW, prof. Maksymilian Dryja
(przy metodach numerycznych), dr hab. Leszek Marcinkowski (również przy metodach nume-
rycznych), prof. Piotr Rybka (jako promotor przy każdej części pracy) i prof. Jerzy Tyszkiewicz
(przy automatach komórkowych). Autor pracy serdecznie dziękuje im za pomoc.

Rozdział 1

Równania stechiometryczne

W pracy będziemy rozpatrywać poniższe trzy modele interakcji pomiędzy cząsteczkami o kolorze niebieskim (N), czerwonym (C), a w przypadku modelu 3-cząsteczkowego również białym (B). Kolory niebieski i czerwony reprezentują cząsteczki chiralne prawo- i lewoskrętne. Kolor biały reprezentuje cząsteczkę achiralną. Modelujemy sytuację, w której pewna ilość cząsteczek porusza się w losowy sposób w przestrzeni czasem zderzając się z sobą i wchodząc w ten sposób w reakcje. Reakcje są definiowane za pomocą odpowiedniego układu równań stechiometrycznych, innego dla każdego modelu. W każdym modelu jest to układ zawierający równania posiadające dwie cząsteczki-substraty dla każdej możliwej pary kolorów i dwie cząsteczki-produkty będące cząsteczkami jednego z kolorów. Zakładamy, że wszystkie cząsteczki mają tę samą masę. Zatem z samej postaci reakcji mamy zasadę zachowania masy oraz zasadę zachowania ilości cząsteczek.

Obok produktów i substratów, kolejnym istotnym czynnikiem w równaniach jest ich szybkość. Wyprowadzamy ją obliczając prawdopodobieństwo losowego wybrania ze zwracaniem dwóch cząstek danych kolorów ze zbioru wszystkich cząstek. Ze względu na sposób wyprowadzania szybkości reakcji, ich suma wynosi 1 dla każdego modelu.

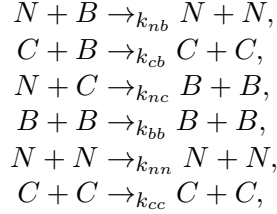
Na tym etapie modelowania zakładamy, że reakcja zachodzi w punkcie i nie uwzględniamy czynników przestrzennych. Zostaną one dodane później, przy okazji tworzenia odpowiednich równań różniczkowych i automatów komórkowych. Mają one znaczny wpływ na długoterminowy przebieg reakcji. Możemy to zaobserwować w eksperymentach dla modeli 3-cząsteczkowego i 2,5-cząsteczkowego, w których tworzy się potok średniokrzywiznowy.

Uwaga 1. W literaturze spotyka się oznaczenia D i L dla cząsteczek kolejno lewo- i prawoskrętnych. W naszym przypadku nie wyróżniamy *à priori* żadnej z chiralnych cząsteczek i przypisujemy im w dowolny sposób kolory N i C .

Uwaga 2. Jeden ze sposobów doboru szybkości reakcji chemicznych (opisany w rozdziale 2 książki [4], [9] bądź w [14] i [13]) zależy od ich rzędu, czyli ilości substratów, których gęstości są brane pod uwagę w doświadczeniu. Przy użyciu tej metody szybkość reakcji jest wprost proporcjonalna do iloczynu gęstości substratów. Rząd reakcji jest jednak możliwy do określenia tylko doświadczalnie. W naszym przypadku używamy do tego modelu probabilistycznego, który skaluje szybkości reakcji odpowiednio do proporcji zagęszczenia cząstek. W jakościowej analizie naszych modeli ważną rolę odgrywają proporcje tych statych do siebie, więc stosujemy model probabilistyczny.

1.1. Model 3-cząsteczkowy

W tym modelu występują 3 kolory cząstek: B , N i C . Ze względu na jego ciekawe własności, model ten będzie głównym modelem używanym i analizowanym w tej pracy. Występuje on także w innych pracach na temat chiralności, np. w artykule [10] Układ równań stechiometrycznych reprezentujący go jest następujący:



gdzie k_{xy} są szybkościami poszczególnych reakcji. W tym modelu zakładamy, że są to reakcje drugiego rzędu, tzn. szybkość reakcji jest proporcjonalna do iloczynu gęstości dwóch substratów. Musimy jednak podać odpowiednie stałe przy każdej reakcji. W takim przypadku, jeżeli b , n i c oznaczają gęstość cząsteczek B , N i C , to wzory na szybkości reakcji są następujące:

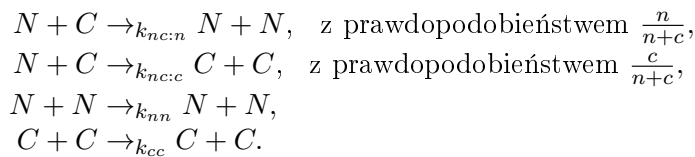
$$\begin{aligned} k_{nb} &= \frac{2nb}{(n+b+c)^2}, \\ k_{cb} &= \frac{2cb}{(n+b+c)^2}, \\ k_{nc} &= \frac{2nc}{(n+b+c)^2}, \\ k_{bb} &= \frac{b^2}{(n+b+c)^2}, \\ k_{nn} &= \frac{n^2}{(n+b+c)^2}, \\ k_{cc} &= \frac{c^2}{(n+b+c)^2}. \end{aligned}$$

Reakcje cząstek tego samego typu niczego nie zmieniają, więc będą dalej pomijane. Przejawiają się wyłącznie w opóźnianiu właściwych reakcji pomiędzy cząsteczkami różnych kolorów.

Po przyjrzeniu się modelowi można zauważyć, że dla długich czasów trwania reakcji prawdopodobieństwo zdominowania przez cząsteczki białe przestrzeni jest znikome, a prawdopodobieństwo dominacji cząsteczek niebieskich i czerwonych zależy od złamania symetrii pomiędzy ich gęstościami. Innymi słowy, stan równowagi ilościowej cząsteczek N i C jest niestabilny, a stany dominacji jednej cząsteczek N i C są stabilne. Będzie to w dalszej części potwierdzone przy badaniu układu równań różniczkowych zwyczajnych reakcji.

1.2. Model 2,5-cząsteczkowy

W tym modelu występują 2 kolory cząstek: N i C . Układ równań stechiometrycznych reprezentujący go jest następujący:



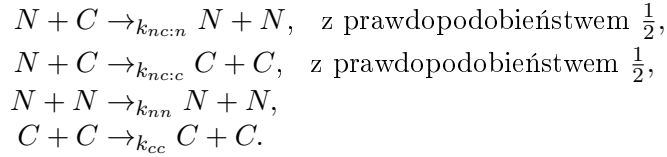
Szybkości reakcji są następujące:

$$\begin{aligned}k_{nc:n} &= \frac{2n^2c}{(n+c)^3}, \\k_{nc:c} &= \frac{2nc^2}{(n+c)^3}, \\k_{nn} &= \frac{n^2}{(n+c)^2}, \\k_{cc} &= \frac{c^2}{(n+c)^2}.\end{aligned}$$

Model ten charakteryzuje się losowością wyniku reakcji cząstek N i C . Z prawdopodobieństwem proporcjonalnym do gęstości względnej cząsteczki niebieskiej ($\frac{n}{n+c}$) zachodzi reakcja zamiany obu cząsteczek na cząsteczki niebieskie. Analogicznie w wypadku reakcji w cząsteczki czerwone. Podobnie do przypadku modelu 3-cząsteczkowego, widać że układ ten posiada niestabilny stan równowagi ilościowej cząsteczek N i C . Złamanie tej symetrii z dużym prawdopodobieństwem doprowadzi do dominacji cząsteczki jednego typu.

1.3. Model 2-cząsteczkowy

Model ten jest podobny do modelu 2,5-cząsteczkowego - występują w nim 2 kolory cząstek (N i C) i podobne reakcje. Jednak różni się prawdopodobieństwo wyboru koloru przy reakcji cząstek N i C :



Szybkości reakcji są następujące:

$$\begin{aligned}k_{nc:n} &= k_{nc:c} = \frac{nc}{(n+c)^2}, \\k_{nn} &= \frac{n^2}{(n+c)^2}, \\k_{cc} &= \frac{c^2}{(n+c)^2}.\end{aligned}$$

Zauważmy, że w tym modelu wartość oczekiwana ilości każdej z cząstek jest stała. Jest to zadanie równoważne z ruiną dwóch graczy o skończonych kapitałach, zatem dynamika jest bardzo powolna i przewagę posiada cząsteczka o większym początkowym zagęszczeniu.

Model ten możemy odnieść do teorii opisywanej w artykule [10], w której dominacja cząstek danego typu w naturze o jednej, określonej chiralności powstała na skutek losowych symetrycznych bądź nieznacznie niesymetrycznych reakcji zmieniających chiralność cząstek w sposób losowy. Jest ona uznawana w tym artykule za mało prawdopodobną do uzyskania bliskiej 100% czystości chiralnej substancji, nawet w ciągu kilkunastu miliardów lat istnienia Wszechświata. Znacznie bardziej wiarygodna wydaje się teoria absorpcyjna, w której prawdopodobieństwo zmiany chiralności cząstek zależy od proporcji zagęszczenia cząstek chiralnych. Tę teorię, wraz z znacznie szybszą dynamiką, przedstawiają modele 3-cząsteczkowy i 2,5-cząsteczkowy.

Rozdział 2

Modele matematyczne reakcji-dyfuzji

2.1. Równanie reakcji

Jako wstęp do równań reakcji-dyfuzji omówmy najpierw równania reakcji. Są to układy równań różniczkowych zwyczajnych opisujące przebieg reakcji chemicznej w punkcie lub doskonale wymieszanych substancji. Następnie stwórzmy odpowiednie równania reakcji dla poszczególnych modeli i zanalizujmy je. Równania te są tworzone na podstawie układu równań stechiometrycznych według określonych zasad, opisanych w rozdziale 2 książki [4] bądź w [14]. Szereg informacji o sposobie ich analizy możemy znaleźć w rozdziale 6 książki [9].

2.1.1. Tworzenie na podstawie równań stechiometrycznych

Rozpatrzmy ogólny układ l równań stechiometrycznych, w którym występują substancje X_i , $i = 1..m$, szybkości reakcji k_j , $j = 1..l$, współczynniki s_{ij} przy substratach i r_{ij} przy produktach. Każde z równań dla $j = 1..l$ przedstawia się następująco:

$$\sum_{i=1}^m s_{ij} X_i \rightarrow_{k_j} \sum_{i=1}^m r_{ij} X_i, \quad j = 1..l.$$

Jeżeli w j -tym równaniu substancja X_i nie występuje jako substrat bądź produkt, to odpowiednio współczynniki (s_{ij} bądź r_{ij}) są równe 0.

Ogólną formą układu równań różniczkowych zwyczajnych opisujących reakcje jest:

$$\dot{u} = f(u), \quad (2.1)$$

gdzie:

- $u = (x_1, \dots, x_m): [0, T) \mapsto \mathbb{R}^k$ jest wektorem gęstości substancji dla czasu od 0 do T , odpowiednio x_i będącymi gęstościami substancji X_i . Spodziewamy się, że wartości te będą nieujemne;
- $f: \mathbb{R}^m \mapsto \mathbb{R}^m$ jest funkcją zmiany gęstości substancji w czasie, odpowiadającą układowi równań stechiometrycznych.

Funkcję f zapiszmy w postaci wektora $f = (f^i)_{i=1}^m$, gdzie f^i oznacza zmianę gęstości substancji X_i w czasie. Wtedy funkcja f_i jest równa

$$f_i(u) = \dot{x}_i = \sum_{j=1}^l (r_{ij} - s_{ij}) k_j(u) \prod_{i=1}^m x_i^{s_{ij}}.$$

Powyższy sposób jest pokrótce opisany w [14] i rozwinięty w rozdziale 2 książki [4].

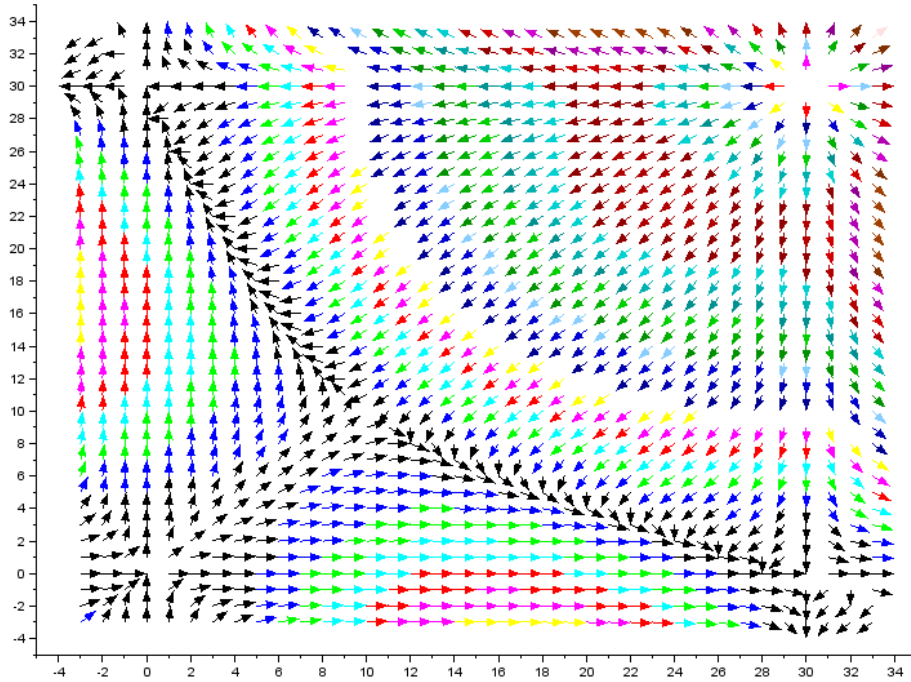
2.1.2. Model 3-cząsteczkowy

Podstawmy do wzoru (2.1.1) reakcje zachodzące w modelu 3-cząsteczkowym opisane w sekcji 1.1:

$$\begin{pmatrix} \dot{b} \\ \dot{n} \\ \dot{c} \end{pmatrix} = f \begin{pmatrix} b \\ n \\ c \end{pmatrix} = \begin{pmatrix} -b^2n^2 - b^2c^2 + 2n^2c^2 \\ b^2n^2 - n^2c^2 \\ b^2c^2 - n^2c^2 \end{pmatrix} \cdot \frac{2}{(b+n+c)^2}. \quad (2.2)$$

Zanalizujemy równania różniczkowe zwyczajne opisujące ten model nieco dokładniej, jako że będzie on w tej pracy głównym przedmiotem badań.

Oznaczmy sumę gęstości wszystkich cząsteczek przez $S = b + n + c$. Sumując wszystkie równania zobaczymy, że $\dot{S} = 0$, czyli całkowita gęstość cząstek w układzie jest stała. Jest to zasada zachowania masy w naszym układzie. Podstawmy $S - n - c$ za b i narysujmy pole wektorowe \dot{u} dla $S = 30$.



Rysunek 2.1: Pole wektorowe modelu 3-cząsteczkowego

Osie odpowiadają gęstościom n i c . Obserwujemy symetrię ze względu na n i c .

2.1.2.1. Nieujemność rozwiązania

Ustalmy czas t . Zauważmy, że jeżeli dwie z wartości b, n, c są zerowe, to $\dot{b} = \dot{n} = \dot{c} = 0$, zatem rozwiązanie jest stałe. Jeżeli $b = 0$, a $n, c > 0$, to $\dot{b} > 0$. Z ciągłości pola wektorowego, w pewnym otoczeniu punktu $(0, n, c)$ mamy $\dot{b} > 0$. Zatem jeżeli $n, c > 0$, to $b = 0$ może zajść wyłącznie w warunkach początkowych (dla $t = 0$). Z kolei jeżeli $n = 0$, a $b, c > 0$, to $\dot{n} = 0$. Podobnie, jeżeli $c = 0$, a $b, n > 0$, to $\dot{c} = 0$. Zatem dla nieujemnych warunków początkowych

mamy nieujemność rozwiązania oraz dla warunków początkowych $n_0, c_0 > 0$, $b_0 \geq 0$ mamy dodatniość rozwiązania dla dowolnego $t > 0$.

Zauważmy, że jeżeli ilość substratów jest dodatnia, to współczynniki szybkości reakcji są również dodatnie. W przeciwnym wypadku szybkość odpowiednich reakcji wynosi zero.

2.1.2.2. Oszacowania funkcji f i ∇f

Zbadajmy jakiej klasy jest funkcja f i spróbujmy ją oszacować. Te informacje będą przydatne w kolejnych częściach pracy. Na początek obliczmy:

$$\begin{aligned} \nabla f|_{(b,n,c)} = & \begin{pmatrix} -bn^2 - bc^2 & -b^2n + 2nc^2 & -b^2c + 2n^2c \\ bn^2 & b^2n - nc^2 & -n^2c \\ bc^2 & -nc^2 & b^2c - n^2c \end{pmatrix} \cdot \frac{4}{(b+n+c)^2} + \\ & \begin{pmatrix} -b^2n^2 - b^2c^2 + 2n^2c^2 \\ b^2n^2 - n^2c^2 \\ b^2c^2 - n^2c^2 \end{pmatrix} \cdot \frac{-4}{(n+b+c)^3} \cdot \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}. \end{aligned} \quad (2.3)$$

Widać, że zarówno f jak i ∇f są ograniczone dla $n+b+c > 0$. Dla dużych wartości możemy wtedy oszacować f przez funkcje kwadratowe, a ∇f przez liniowe. Zatem z oszacowania $0 \leq b, n, c \leq S$ otrzymujemy oszacowania na f i ∇f .

Jeżeli $0 \leq b, n, c \rightarrow 0$, to szacujemy poszczególne czynniki w następujący sposób:

$$\frac{b^2n^2}{(b+n+c)^2} \leq \frac{b^2n^2}{\max(b^2, n^2, c^2)} \rightarrow 0.$$

Przy odpowiednich modyfikacjach wyrazów i potęg w powyższym otrzymujemy oszacowania prowadzące do zbieżności $f \rightarrow 0$ i $\nabla f \rightarrow 0$, otrzymując tym sposobem jednostronną ciągłość tych funkcji. Zatem rozszerzona w zerze funkcja f jest klasy $C^1[0, +\infty)$.

2.1.2.3. Punkty stacjonarne

Przepiszmy układ równań (2.2) podstawiając $b = S - n - c$:

$$\begin{pmatrix} \dot{n} \\ \dot{c} \end{pmatrix} = g \begin{pmatrix} n \\ c \end{pmatrix} = \begin{pmatrix} (S-n-c)^2n^2 - n^2c^2 \\ (S-n-c)^2c^2 - n^2c^2 \end{pmatrix} \cdot \frac{2}{S^2}, \quad (2.4)$$

Znajdźmy punkty stacjonarne przy założeniu, że $S > 0$:

$$\begin{cases} (S-n-c)^2n^2 = n^2c^2, \\ (S-n-c)^2c^2 = n^2c^2. \end{cases}$$

Od razu widać mniej ciekawe punkty stacjonarne (b, n, c) równe $(S, 0, 0)$, $(0, S, 0)$ i $(0, 0, S)$. Będziemy te punkty nazywać skrajnymi. Nas jednak interesuje ciekawy przypadek, w którym $n > 0$ i $c > 0$:

$$(S-n-c)^2 = n^2 = c^2.$$

To równanie ma dwa rozwiązania. Pierwsze to $n = c = S$, które odrzucamy, ponieważ $n+c \leq b+n+c \leq S$. Drugie, najbardziej interesujące to $b = n = c = \frac{1}{3}S$. Nazwiemy ten przypadek środkowym.

2.1.2.4. Stabilność

Zlinearyzujemy nasz układ równań (2.4) w punktach stacjonarnych.

W momencie gdy dwie spośród wartości b , n i c są równe zero, część liniowa funkcji g jest również równa zero. Spróbujmy innej metody niż linearyzacja. Stwórzmy odpowiednią funkcję Lapunowa dla każdego z przypadków. To znaczy, taką funkcję $V: \mathbb{R}^2 \mapsto \mathbb{R}$, by dla badanej funkcji φ :

$$\begin{cases} V(0) = 0, \\ V(x) > 0, \quad x \neq 0, \\ (\forall x \neq 0 \quad (DV|_x, \varphi(x)) < 0) \vee (\forall x \neq 0 \quad (DV|_x, \varphi(x)) > 0). \end{cases}$$

Ograniczmy się do zbioru $(n, c) \in [0, S]^2$, jako że udowodniliśmy, że jest to zbiór niezmienniczy. Ponadto, z szkicu pola wektorowego wynika, że stabilność tych punktów w całej przestrzeni jest różna od stabilności we właściwej podprzestrzeni.

Dla punktu stacjonarnego $(b, n, c) = (S, 0, 0)$ weźmy funkcję

$$V(x, y) = \frac{1}{2}x^2 + \frac{1}{2}y^2.$$

Wtedy dla pewnego ε , $0 \leq n, c < \varepsilon$, $(n, c) \neq (0, 0)$:

$$(DV|_{(n,c)}, g(n, c)) = \frac{2}{S^2} ((S - n - c)^2 n^3 - n^3 c^2 + (S - n - c)^2 c^3 - n^2 c^3) > 0.$$

Powyższa nierówność zachodzi w pewnym otoczeniu tego punktu, jako że czynnik S dominuje nad pozostałymi. Udowodniliśmy tym sposobem, że jest on niestabilny w badanej podprzestrzeni.

Ze względu na symetrię równań ze względu na n i c , wystarczy zbadać stabilność punktu $(b, n, c) = (0, S, 0)$. Przypadki te są takie same z dokładnością do zamiany liter c i n we wzorach, zatem rozważymy tylko pierwszy. Dokonajmy najpierw zamiany zmiennych $z = S - n$, by punkt stacjonarny znajdował się w zerze, a wartości (z, c) w małym, nieujemnym otoczeniu zera. Wtedy operujemy na układzie równań:

$$\begin{pmatrix} \dot{z} \\ \dot{c} \end{pmatrix} = h \begin{pmatrix} z \\ c \end{pmatrix} = \begin{pmatrix} -(z - c)^2 (S - z)^2 + (S - z)^2 c^2 \\ (z - c)^2 c^2 - (S - z)^2 c^2 \end{pmatrix} \cdot \frac{2}{S^2}.$$

Weźmy funkcję

$$V(x, y) = e^{xy} - 1.$$

Spełnia ona wymagania w naszej podprzestrzeni. Wtedy dla pewnego ε , $0 \leq z, c < \varepsilon$, $(z, c) \neq (0, 0)$:

$$(DV|_{(z,c)}, h(z, c)) = \frac{2}{S^2} e^{xy} (z - c)^2 (c^2 - (S - z)^2) < 0.$$

$c^2 - (S - z)^2 < 0$, ponieważ dla dostatecznie małego ε zachodzi $S - z > c \geq 0$. Zatem powyższa nierówność zachodzi w pewnym otoczeniu badanego punktu. Udowodniliśmy tym sposobem, że jest on asymptotycznie stabilny w badanej podprzestrzeni.

W przypadku środkowym ($b = n = c = \frac{1}{3}S$) wystarczy zlinearyzować układ (2.4), otrzymując macierz linearyzacji

$$\begin{pmatrix} -\frac{4}{27}S & -\frac{8}{27}S \\ -\frac{8}{27}S & -\frac{4}{27}S \end{pmatrix},$$

której wartości własne to $\frac{4S}{27}$ oraz $-\frac{12S}{27}$. Jej podprzestrzenie własne to odpowiednio $\text{lin}(1, -1)$ i $\text{lin}(1, 1)$. Zatem przypadek środkowy jest niestabilny, typu siodło.

2.1.3. Model 2,5-cząsteczkowy

Podstawmy do wzoru (2.1.1) reakcje zachodzące w modelu 2,5-cząsteczkowym opisane w sekcji 1.2:

$$\begin{pmatrix} \dot{n} \\ \dot{c} \end{pmatrix} = f \begin{pmatrix} n \\ c \end{pmatrix} = \begin{pmatrix} n^3 c^2 - n^2 c^3 \\ n^2 c^3 - n^3 c^2 \end{pmatrix} \cdot \frac{2}{(n+c)^2}.$$

Na początek zauważmy, że suma gęstości cząstek $S = n + c$ jest stała. Widać również, że układ ten posiada 3 stany stacjonarne: $n = 0, c = 0$ i $n = c$. Zauważmy, że dla $n = 0$ bądź $c = 0$ mamy $\dot{n} = \dot{c} = 0$, zatem rozwiązania tego układu posiadają ograniczenie $0 \leq n, c \leq S$. Podstawiając $S - n$ w miejsce c (bądź $S - c$ w miejsce n) i obserwując znaki pochodnych w okolicy stanów stacjonarnych możemy wykazać, że $n = 0$ i $c = 0$ są stabilnymi stanami stacjonarnymi, a $n = c$ niestabilnym. Jest to zgodne z wcześniejszymi rozważaniami probabilistycznymi, w których zachwianie symetrii układu prowadzi do dominacji cząsteczek jednego koloru.

2.1.4. Model 2-cząsteczkowy

Podstawiając do wzoru (2.1.1) reakcje zachodzące w modelu 2-cząsteczkowym opisane w sekcji 1.3, otrzymujemy trywialny układ równań różniczkowych:

$$\dot{u} = f(u) = 0.$$

Posiada on zerową funkcję zmienności f i w związku z tym wyłącznie stałe rozwiązania. Jest to zgodne z wcześniejszą analizą probabilistyczną.

2.2. Równanie reakcji-dyfuzji

W dalszej części pracy będziemy rozważać układy równań reakcji-dyfuzji opisujące przebieg reakcji zachodzących pomiędzy m substancjami, które jednocześnie dyfundują. Ogólne równanie różniczkowe cząstkowe opisujące to zjawisko ma postać:

$$u_t - D \cdot \Delta u = f(u), \quad (2.5)$$

gdzie:

- $u: [0, T) \times \Omega \mapsto \mathbb{R}^m$ jest gęstością jednej z m substancji chemicznych w przestrzeni Ω dla czasu od 0 do T ,
- $f: \mathbb{R}^m \mapsto \mathbb{R}^m$ jest prawą stroną równania różniczkowego zwyczajnego reakcji (2.1), opisującą zmianę gęstości substancji chemicznych w wyniku reakcji pomiędzy nimi,
- $D: \mathbb{R}^{m \times m}$ jest macierzą opisującą współczynniki dyfuzji reakcji chemicznej.

Przez laplasjan wektora rozumiemy wektor laplasjanów jego składowych. Aby model był poprawny fizycznie, to dla nieujemnych danych początkowych u_0 rozwiązanie u musi pozostać nieujemne (dla każdej ze współrzędnych wektora).

W dalszej części będziemy zajmować się tym równaniem zastosowanym do naszych modeli. Zatem $m = 3$ i $u = (b, n, c)$ bądź $m = 2$ i $u = (n, c)$. f będzie równe prawym stronom równań reakcji z sekcji 2.1. Będziemy zakładać, że $\Omega \subset \subset \mathbb{R}^d$, $d = 2, 3$. Będziemy zakładać zerowe warunki brzegowe Neumanna. Jeżeli nie zaznaczymy inaczej, będziemy rozważać diagonalne macierze dyfuzji D : $D = \text{diag}(d_b, d_n, d_c)$ bądź $D = \text{diag}(d_n, d_c)$. Skoncentrujemy się na własnościach modelu 3-cząsteczkowego. Wiele z nich będzie ponadto wymagało, by współczynniki

dyfuzji były równe $d_b = d_n = d_c$. Dodając warunki brzegowe i początkowe, nasz układ równań będzie następujący:

$$\begin{cases} u_t - D \Delta u = f(u), & \text{w } \Omega, \\ \frac{\partial u}{\partial n} = 0, & \text{na } \partial\Omega, \\ u(0, \cdot) = u_0, \end{cases} \quad (2.6)$$

Dla wygody będziemy zamiennie używać indeksów b, n i c oraz liczbowych. Zbadajmy teraz kilka podstawowych własności naszych modeli.

2.2.1. Model 2-cząsteczkowy

Jako że w modelu 2-cząsteczkowym funkcja f jest zerowa, to równanie (2.6) jest wtedy zwykłym równaniem dyfuzji. Zostało ono dobrze zbadane i w związku z tym nie będziemy się tym modelem zajmować w dalszej części.

2.2.2. Zasada zachowania masy

Nadajmy oznaczenie sumie gęstości $S = \sum_{i=1}^m u^i$. Z zasady zachowania masy dla równań reakcji obecnych w każdym z trzech rozpatrywanych modeli $\sum_{i=1}^m f_i(u) \equiv 0$. Zatem po zsumowaniu i scałkowaniu wszystkich równań otrzymamy:

$$\int_{\Omega} S_t = \int_{\Omega} \sum_{i=1}^m u_t^i = \sum_{i=1}^m \int_{\Omega} d_i \Delta u^i + \int_{\Omega} \sum_{i=1}^m f_i(u) = 0.$$

2.2.3. Równanie dyfuzji dla sumy mas

Założmy, że zachodzi zasada zachowania masy oraz ponadto współczynniki dyfuzji są równe: $d_i = d$ dla każdego $i = 1..m$. Wtedy suma gęstości substancji S spełnia równanie dyfuzji z zerowymi warunkami brzegowymi Neumanna:

$$\begin{cases} S_t - d \Delta S = 0, & \text{w } \Omega, \\ \frac{\partial S}{\partial n} = 0, & \text{na } \partial\Omega, \\ S(t, \cdot) = S_0, \end{cases}$$

gdzie $S_0 = \sum_{i=1}^m u_0^i$.

2.2.4. Nieujemność rozwiązania

Lemat 1. *Niech u spełnia równanie reakcji-dyfuzji (2.6) dla modelu 3-cząsteczkowego. Załóżmy, że f jest dane wzorem (2.2) oraz $u_0 \geq 0$. Ponadto założmy à priori, że $u \in C^2(\Omega)$. Wtedy rozwiązania $u = (b, n, c)$ układu (2.6) są nieujemne dla dowolnego $t \geq 0$ i $x \in \Omega$:*

$$0 \leq b, n, c.$$

Ponadto, jeżeli współczynniki dyfuzji są równe ($d_b = d_n = d_c$), to mamy również ograniczenie górne:

$$b, n, c \leq S \leq \|S_0\|_{L^\infty}.$$

Dowód. Udowodnimy nieujemność rozwiązania po kolei dla każdej z współrzędnych b , n , c wektora $u(t, x)$. Zaczniemy od współrzędnej b .

Weźmy dowolne $t > 0$, dla którego istnieje minimum lokalne na domknięciu wykresu b (oznaczanym po prostu jako b) w punkcie $x_0 \in \bar{\Omega}$ oraz

$$b(t, x_0) = 0.$$

Nazwijmy (t, x_0) punktem krytycznym. Chcemy udowodnić, że $b_t(t, x_0) \geq 0$.

$$b_t = \Delta b + f_b. \quad (2.7)$$

W punkcie (t, x_0) :

$$f_b(b, n, c) = 2 \frac{-b^2(n^2 + c^2) + 2n^2c^2}{(n + b + c)^2} = \frac{2n^2c^2}{(n + b + c)^2} \geq 0.$$

W szczególności dla $n > 0$ i dla $c > 0$ otrzymujemy nierówność ostrą. Pozostaje udowodnić, że $\Delta b(t, x_0) \geq 0$.

Weźmy pewne dostatecznie małe, kuliste otoczenie U punktu x_0 . Ponieważ x_0 jest minimum lokalnym, to pole wektorowe ∇b wpływa do wnętrza U :

$$\frac{\partial b}{\partial n}|_{\partial U} \leq 0.$$

W szczególności, jeżeli $x_0 \in \partial\Omega$, to z warunków brzegowych ta wielkość jest dobrze określona na $\partial U \cap \partial\Omega$:

$$\frac{\partial b}{\partial n}|_{\partial U \cap \partial\Omega} = 0.$$

Scałkujmy równanie (2.7) i podzielmy przez miarę U :

$$\int_U b_t = \int_U \Delta b + \int_U f_b = -\frac{1}{\mu(U)} \int_{\partial U} \frac{\partial b}{\partial n} + \int_U f_b = g_U + \int_U f_b.$$

Z wcześniejszej analizy wiemy, że $g_U \geq 0$. Weźmy zstępujący ciąg takich otoczeń (U_i) . Wtedy:

$$b_t(t, x_0) \leftarrow \int_{U_i} b_t = -\frac{1}{\mu(U_i)} \int_{\partial U_i} \frac{\partial b}{\partial n} + \int_{U_i} f_b \rightarrow g(t, x_0) + f_b(t, x_0) \geq 0.$$

$g = \Delta b$ dla $x_0 \in \Omega$. W każdym wypadku $g \geq 0$ jako granica nieujemnych funkcji. $f_b(t, x_0) \geq 0$ zgodnie z wcześniejszą analizą. Kończy to dowód, że $b_t(t, x_0) \geq 0$.

Ponieważ rozważane punkty krytyczne (t, x_0) były jedynymi, w których wartości funkcji b mogły stać się ujemne, a pochodna b jest w nich nieujemna, to b jest nieujemna dla dowolnego $t \geq 0$.

Pozostają przypadki współrzędnych n i c . Rozwiązuje się je analogicznie. Najpierw analizujemy wartość funkcji f w punktach krytycznych

$$f(b, 0, c) = f(b, n, 0) = 0.$$

Wiedząc to, postępujemy analogicznie do przypadku z b otrzymując nieujemność n_t i c_t w punktach krytycznych. Z tego mamy nieujemność n , c dla dowolnego $t > 0$.

Jeżeli wszystkie współczynniki dyfuzji są równe, to mamy równanie dyfuzji na sumę gęstości substancji. W związku z nieujemnością rozwiązania mamy dla dowolnego czasu t :

$$0 \leq b, n, c \leq b + n + c = S.$$

W związku z zasadą maksimum równania dyfuzji oraz nieujemnością S mamy $S \leq \|S_0\|_{L^\infty}$, co kończy ciąg nierówności z lematu. \square

Lemat 2. Niech u spełnia równanie reakcji-dyfuzji (2.6) dla modelu 2,5-cząsteczkowego. Załóżmy, że f jest dane wzorem (2.1.3) oraz $u_0 \geq 0$. Ponadto załóżmy à priori, że $u \in C^2(\Omega)$. Wtedy rozwiązania $u = (n, c)$ układu (2.6) są nieujemne dla dowolnego $t \geq 0$ i $x \in \Omega$:

$$0 \leq n, c.$$

Ponadto, jeżeli współczynniki dyfuzji są równe ($d_n = d_c$), to mamy również ograniczenie górne:

$$n, c \leq S \leq \|S_0\|_{L^\infty}.$$

Dowód. Dowód przebiega identycznie do przypadku modelu 3-cząsteczkowego. Wykorzystywaną własnością funkcji f będzie w tym wypadku:

$$f(0, c) = f(n, 0) = 0.$$

□

2.3. Skalowanie

Weźmy równanie reakcji-dyfuzji (2.6), pomnożmy współrzędne przestrzenne przez α , współrzędną czasową przez β , a szybkości reakcji przez γ . Wtedy, po zamianie nazw zmiennych na oryginalne, otrzymujemy:

$$\beta u_t - \alpha^2 D \cdot \Delta u = \gamma f(u).$$

Oznacza to, że nie zmieniamy oryginalnego równania, jeżeli

$$\beta = \alpha^2 = \gamma.$$

Przyspieszenie reakcji γ razy jest równoważne przyspieszeniu czasu γ razy, i pomnożeniu współrzędnych przestrzennych przez $\sqrt{\gamma}$.

2.4. Istnienie i jednoznaczność rozwiązania

Istnienie i jednoznaczność gładkich rozwiązań lokalnych dla układu równań (2.6) - równania z gładkimi współczynnikami i warunkami brzegowymi Neumanna - jest wnioskiem z twierdzenia z rozdziału VII, paragrafu 10 książki [8].

2.5. Zbiory niezmiennicze i implikacje

W tej sekcji wykorzystamy teorię zbiorów niezmienniczych dla równań reakcji-dyfuzji obecną w rozdziale 14 „Systems of Reaction-Diffusion Equations” książki Smollera [11] i w pracy Chueha, Conleya i Smollera [1] do udowodnienia kilku faktów dla modelu 3-cząsteczkowego z równymi współczynnikami dyfuzji $d_b = d_n = d_c = d$. Najpierw przybliżmy kilka definicji i uproszczonych wersji twierdzeń z tych prac.

2.5.1. Teoria

Na początku podajmy potrzebne definicje.

Definicja 1. Zbiór B nazwiemy **dodatnio niezmienniczym**, jeżeli rozwiązanie $u(x, t)$ układu (2.5) spełnia $u(x, t) \in \Sigma$ dla $0 \leq t < T$, jeżeli $u(x, 0) \in \Sigma$.

Określmy zbiór B za pomocą skończonego zbioru gładkich funkcji $G_i: \Omega \mapsto \mathbb{R}$, $i = 1..n$ (gdzie Ω jest przestrzenią wartości rozwiązania (2.5)) w następujący sposób:

$$B = \{x \in \Omega: G_i(x) \leq 0, i = 1..n\}.$$

Definicja 2. *Lewy wektor własny macierzy D to taki wektor v , że dla pewnego $\lambda \in \mathbb{R}$ zachodzi $vD = \lambda v$.*

Definicja 3. *Funkcję $G: \mathbb{R}^n \mapsto \mathbb{R}$ nazywamy **quasi-wypukłą** w punkcie x , jeżeli dla dowolnego $\eta \in \mathbb{R}^n$ zachodzi implikacja $dG|_x(\eta) = 0 \Rightarrow d^2G|_x(\eta) \leq 0$.*

Podajmy teraz uproszczone na nasze potrzeby twierdzenia z podanych wcześniej prac.

Twierdzenie 1 (Twierdzenie 4.3 z pracy [1] ze strony 378). *Niech B będzie zbiorem określonym przez funkcje G_i jak powyżej. Niech rozwiązania (2.5) zależą w sposób ciągły od funkcji f . Niech D będzie dodatnio określoną macierzą. Wtedy B jest zbiorem dodatnio niezmienniczym wtedy i tylko wtedy, gdy dla $x \in \partial B$ (tak, że $G_i(x) = 0$ dla pewnego i) zachodzi:*

- $dG_i(x)$ jest lewym wektorem własnym macierzy D ,
- G_i jest quasi-wypukłe w x ,
- $dG_i(f) \leq 0$.

Twierdzenie 2 (Twierdzenie 14.17 z książki [11] ze strony 223). *Rozważmy układ (2.6) w przestrzeni Ω . Załóżmy, że D jest macierzą dodatnio określoną, a f funkcją klasy C^1 . Załóżmy, że posiada on ograniczony zbiór niezmienniczy Σ . Niech λ jest główną wartością własną operatora $-\Delta$ na Ω . Niech $M = \max\{|df_u|: u \in \Sigma\}$. Niech d jest najmniejszą wartością własną macierzy D . Zdefiniujmy $\sigma = \lambda d - M$. Jeżeli $\sigma > 0$ i $u_0(\Omega) \subset \Sigma$, to istnieją takie stałe dodatnie c_i , że dla $t > 0$ zachodzi:*

$$\|\nabla_x u(\cdot, t)\|_{L^2(\Omega)} \leq c_1 e^{-\sigma t}, \quad (2.8)$$

$$\|u(\cdot, t) - \bar{u}(t)\|_{L^2(\Omega)} \leq c_2 e^{-\sigma t}, \quad (2.9)$$

$$|g(t)| \leq c_3 e^{-\sigma t}, \quad (2.10)$$

gdzie:

$$\begin{aligned} \bar{u}(t) &= \oint_{\Omega} u(x, t) dt, \\ \frac{d\bar{u}}{dt} &= f(\bar{u}) + g(t). \end{aligned}$$

Ponadto, jeżeli D jest macierzą diagonalną, to zachodzi również

$$\|u(\cdot, t) - \bar{u}(t)\|_{L^\infty(\Omega)} \leq c_4 e^{-\sigma t}. \quad (2.11)$$

2.5.2. Zastosowanie do modelu 3-cząsteczkowego

Rozpatrzmy (2.6) dla modelu 3-cząsteczkowego w kontekście twierdzeń 1 i 2. Załóżmy, że $D = dI$ dla pewnego $d > 0$. Wzór funkcji f znajduje się w (2.2), a ∇f w (2.3). Przedłużmy wartość 0 w sposób ciągły f i ∇f w zerze.

Zastosujmy twierdzenie 1, by udowodnić istnienie zbioru dodatnio niezmienniczego dla rozwiązań (2.6) w inny sposób niż w poprzednich sekcjach. Jeżeli $S_0 = b_0 + n_0 + c_0$, to możemy określić zbiór B za pomocą funkcji G_i , $i = 1..4$ w następujący sposób:

$$G_1(b, n, c) = -b, \quad G_2(b, n, c) = -n, \quad G_3(b, n, c) = -c, \quad G_4(b, n, c) = b + n + c - \|S_0\|_{L^\infty}.$$

Niech zbiór B będzie określony za pomocą tych funkcji G_i tak, jak w poprzedniej sekcji. Wtedy B zawiera wartości (b, n, c) określone za pomocą nierówności

$$0 \leq b, n, c, \quad b + n + c \leq \|S_0\|_{L^\infty}.$$

Rozwiązania u zależą od funkcji f w sposób ciągły. D jest dodatnio określoną macierzą. Wartości dG_i są następujące:

$$dG_1 = (-1, 0, 0), \quad dG_2 = (0, -1, 0), \quad dG_3 = (0, 0, -1), \quad dG_4 = (1, 1, 1).$$

Wszystkie te wektory są lewymi wektorami własnymi macierzy D . Ponadto, $d^2G_i \equiv 0$, zatem funkcje G_i są quasi-wypukłe w całej przestrzeni. Obliczmy $dG_i(f)$ zakładając, że $G_i(b, n, c) = 0$ dla poszczególnych i :

$$\begin{aligned} dG_1(f) &= -1 \cdot f_b(0, n, c) = -\frac{4n^2c^2}{(b+n+c)^2} \leq 0, \\ dG_2(f) &= -1 \cdot f_n(b, 0, c) = 0 \leq 0, \\ dG_3(f) &= -1 \cdot f_c(b, n, 0) = 0 \leq 0, \\ dG_4(f) &= 1 \cdot f_b(b, n, c) + 1 \cdot f_n(b, n, c) + 1 \cdot f_c(b, n, c) = 0 \leq 0. \end{aligned}$$

Wszystkie założenia twierdzenia 1 są spełnione, zatem B jest zbiorem dodatnio niezmienniczym dla (2.6).

D jest macierzą dodatnio określoną o jedynej wartości własnej $d > 0$, $f \in C^1$, a (2.6) posiada zbiór niezmienniczy B określony powyżej. Jest on ograniczony, zatem $|\nabla f|$ jest również ograniczone, a wartość M z twierdzenia 2 jest również dobrze określona. Zatem, jeżeli główna wartość własna λ operatora $-\Delta$ na Ω spełnia $\lambda d - M > 0$, to zachodzi twierdzenie 2. Oznacza to, że dla pewnych przestrzeni Ω rozwiązania dążą do rozwiązań stałych w przestrzeni. Możemy się spodziewać, że są to rozwiązania odpowiadające stanom stacjonarnym równania reakcji (2.2).

Rozdział 3

Metody numeryczne

Chcemy rozwiązać numerycznie równania reakcji-dyfuzji (2.6) dla czasu $t \in [0, T]$. W tym celu użyjemy metody elementu skończonego do dyskretyzacji przestrzeni i zamkniętego schematu Eulera do dyskretyzacji czasu.

3.1. Opis schematu

Ustalmy długość kroku czasowego na τ . Oznaczmy przez $u_n: \Omega \mapsto \mathbb{R}^k$ obliczoną numerycznie aproksymację (zamkniętym schematem Eulera) rozwiązania dla $t = n\tau$. Wewnątrz każdego kroku czasowego będziemy rozwiązywać za pomocą metody elementu skończonego następujący układ równań:

$$\begin{cases} \frac{u_{n+1} - u_n}{\tau} - \Delta u_{n+1} - f(u_{n+1}) = 0, \\ \frac{\partial u}{\partial n} = 0, \end{cases} \quad (3.1)$$

przy czym u_0 jest warunkiem początkowym. Ponieważ funkcja f nie jest liniowa, będziemy obliczać u_{n+1} iteracyjnie. Zbadamy dwa rodzaje aproksymacji nieliniowości: metodę predictor-corrector oraz metodę Newtona.

Metoda elementu skończonego nie będzie w tej pracy omówiona. Informacje na jej temat znajdują się na przykład w rozdziale 10.3 książki [2] lub (dla zagadnień parabolicznych) w rozdziale 8 książki [6]. Podczas analizy obu użytych schematów, przez Φ oznaczmy przestrzeń funkcji próbnych metody elementu skończonego z bazą $(\varphi_i)_{i=1}^{kN}$, gdzie $\varphi_i: \mathbb{R}^n \mapsto \mathbb{R}^k$ będzie funkcją przyjmującą wartość zero na wszystkich współrzędnych poza $(i \bmod k) + 1$. Niech \tilde{u}_{n+1}^m będzie funkcją z tej przestrzeni przybliżającą funkcję u_{n+1}^m , a \bar{u} wektorem liczb rzeczywistych odpowiadającym jej w następujący sposób:

$$\tilde{u} = \sum_{i=1}^N \bar{u}^i \varphi_i.$$

Oznaczmy przez \tilde{u}_n funkcję \tilde{u}_n^m z ostatniej iteracji. Będziemy dążyć do utworzenia układu równań liniowych na \bar{u} .

3.1.1. Aproksymacja metodą predictor-corrector

Będziemy aproksymować nieliniowość przez iteracyjne obliczanie za pomocą metody elementu skończonego funkcji u_{n+1}^m w następującym układzie równań:

$$\begin{cases} \frac{u_{n+1}^{m+1} - u_n}{\tau} - \Delta u_{n+1}^{m+1} - f(u_{n+1}^m) = 0, \\ \frac{\partial u_{n+1}^{m+1}}{\partial n} = 0. \\ u_{n+1}^0 = u_n, \end{cases} \quad (3.2)$$

Napiszmy słabą postać równania (3.2) na \tilde{u} biorąc funkcje próbne φ_j z przestrzeni Φ :

$$\begin{aligned} 0 &= \int_{\Omega} \langle D \nabla \tilde{u}_{n+1}^{m+1}, \nabla \varphi_j \rangle + \int_{\Omega} \left\langle \frac{\tilde{u}_{n+1}^{m+1}}{\tau}, \varphi_j \right\rangle + \int_{\Omega} \left\langle \frac{-\tilde{u}_n}{\tau} - f(\tilde{u}_{n+1}^m), \varphi_j \right\rangle \\ &= \sum_{i=1}^{kN} \bar{u}^i \left(\int_{\Omega} \langle D \nabla \varphi_i, \nabla \varphi_j \rangle + \int_{\Omega} \left\langle \frac{\varphi_i}{\tau}, \varphi_j \right\rangle \right) + \int_{\Omega} \left\langle \frac{-\tilde{u}_n}{\tau} - f(\tilde{u}_{n+1}^m), \varphi_j \right\rangle. \end{aligned}$$

Otrzymujemy zatem następujący układ równań liniowych:

$$\begin{aligned} M \bar{u} &= \bar{f}, \\ M &= (a_{ji})_{i,j=1..kN}, \quad \bar{f} = (f_j)_{j=1..kN}, \\ a_{ji} &= \int_{\Omega} \langle D \nabla \varphi_i, \nabla \varphi_j \rangle + \int_{\Omega} \left\langle \frac{\varphi_i}{\tau}, \varphi_j \right\rangle, \\ \bar{f}_j &= \int_{\Omega} \left\langle \frac{-\tilde{u}_n}{\tau} - f(\tilde{u}_{n+1}^m), \varphi_j \right\rangle. \end{aligned}$$

Zauważmy, że jeżeli D będzie symetryczna i dodatnio określona, to macierz M również będzie symetryczna i dodatnio określona. Ponadto, zarówno M jak i \bar{f} zależą od obu indeksów n i m , zatem nie pozostają stałe w żadnej z iteracji.

3.1.2. Aproksymacja metodą Newtona

Kolejną metodą aproksymacji nieliniowości jest użycie metody Newtona. Opiszmy ją dla ogólnego przypadku. Niech F będzie operatorem, którego zero chcemy znaleźć, a $L|_{x_n}$ będzie częścią liniową tego operatora w punkcie x_n . Wtedy metoda Newtona opisuje się wzorem:

$$F(x_n) + L|_{x_n}(x_{n+1} - x_n) = 0,$$

gdzie x_0 jest początkowym przybliżeniem. Dla F będących funkcjami klasy C^1 częścią liniową jest gradient tej funkcji oraz można udowodnić kwadratową zbieżność tej metody, jeżeli tylko startujemy z x_0 dostatecznie bliskiego szukanemu zeru.

W naszym przypadku:

$$\begin{aligned} F(u_{n+1}^{m+1}) &= \frac{u_{n+1}^{m+1} - u_n}{\tau} - D \Delta u_{n+1}^{m+1} - f(u_{n+1}^m) - \nabla f|_{u_{n+1}^m}(u_{n+1}^{m+1} - u_{n+1}^m), \\ L|_{u_{n+1}^m}(v) &= \frac{1}{\tau} v - D \Delta v - \nabla f|_{u_{n+1}^m} v. \end{aligned}$$

Na etapie rozwiązywania pojedynczej iteracji metody Newtona korzystamy z metody elementu skończonego, więc operator F jest zwykłą funkcją, a L jej gradientem. Zatem, traktując (3.1)

jako funkcję nieliniową zmiennej u_{n+1} , znajdujemy jej zero metodą Newtona. Otrzymany schemat jest następujący:

$$\begin{cases} \frac{u_{n+1}^{m+1} - u_n}{\tau} - D\Delta u_{n+1}^{m+1} - f(u_{n+1}^m) - \nabla f|_{u_{n+1}^m}(u_{n+1}^{m+1} - u_{n+1}^m) = 0, \\ \frac{\partial u_{n+1}^{m+1}}{\partial n} = 0. \\ u_{n+1}^0 = u_n, \end{cases} \quad (3.3)$$

Napiszmy słabą postać równania (3.3) na \tilde{u} biorąc funkcje próbne φ_j z przestrzeni Φ :

$$\begin{aligned} 0 &= \int_{\Omega} \langle D\nabla \tilde{u}_{n+1}^{m+1}, \nabla \varphi_j \rangle + \int_{\Omega} \left\langle \left(\frac{I}{\tau} - \nabla f|_{\tilde{u}_{n+1}^m} \right) \tilde{u}_{n+1}^{m+1}, \varphi_j \right\rangle \\ &\quad + \int_{\Omega} \left\langle \frac{-\tilde{u}_n}{\tau} - f(\tilde{u}_{n+1}^m) + \nabla f|_{\tilde{u}_{n+1}^m} \tilde{u}_{n+1}^m, \varphi_j \right\rangle \\ &= \sum_{i=1}^{kN} \bar{u}^i \left(\int_{\Omega} \langle D\nabla \varphi_i, \nabla \varphi_j \rangle + \int_{\Omega} \left\langle \left(\frac{I}{\tau} - \nabla f|_{\tilde{u}_{n+1}^m} \right) \varphi_i, \varphi_j \right\rangle \right) \\ &\quad + \int_{\Omega} \left\langle \frac{-\tilde{u}_n}{\tau} - f(\tilde{u}_{n+1}^m) + \nabla f|_{\tilde{u}_{n+1}^m} \tilde{u}_{n+1}^m, \varphi_j \right\rangle. \end{aligned}$$

Otrzymujemy zatem następujący układ równań liniowych:

$$\begin{aligned} M\bar{u} &= \bar{f}, \\ M &= (a_{ji})_{i,j=1..kN}, \quad \bar{f} = (f_j)_{j=1..kN}, \\ a_{ji} &= \int_{\Omega} \langle D\nabla \varphi_i, \nabla \varphi_j \rangle + \int_{\Omega} \left\langle \left(\frac{I}{\tau} - \nabla f|_{\tilde{u}_{n+1}^m} \right) \varphi_i, \varphi_j \right\rangle, \\ \bar{f}_j &= \int_{\Omega} \left\langle \frac{\tilde{u}_n}{\tau} + f(\tilde{u}_{n+1}^m) - \nabla f|_{\tilde{u}_{n+1}^m} \tilde{u}_{n+1}^m, \varphi_j \right\rangle. \end{aligned}$$

Zauważmy, że w ogólności jeżeli D bądź ∇f nie są symetryczne, to macierz M również nie będzie symetryczna. Zauważmy również, że jeżeli macierz D jest dodatnio określona, τ jest dostatecznie małe, a ∇f ograniczona w punktach naszego rozwiązania, to macierz M będzie dodatnio określona. Ponadto, zarówno M jak i \bar{f} zależą od obu indeksów n i m , zatem nie pozostają stałe w żadnej z iteracji.

3.1.3. Oszacowanie błędu metody Newtona

Metoda Newtona dla przypadku wielowymiarowego ma zbieżność kwadratową. W naszym przypadku, po rozpisaniu na układy równań liniowych, metodę Newtona wykonujemy na przestrzeni kN -wymiarowej wartości rzeczywistych (obliczając wektor \bar{u}). Podajmy twierdzenie zaczerpnięte z [2] o zbieżności metody Newtona.

Twierdzenie 3 (Twierdzenie 7.5 z książki [2] ze strony 93). *Jeżeli pierwsza pochodna f jest Lipschitzowska, początkowe przybliżenie x_0 należy do kuli S o środku w zerze funkcji f oznaczonym α i promieniu R , takim, że dla pewnego $0 < q < 1$*

$$\sup_{x,y \in S} \frac{R|f'(x) - f'(y)|}{2|f'(\alpha)||x - y|} \leq \frac{q}{1 + 2q},$$

to metoda Newtona z rozwiązaniami k -tej iteracji x_k jest dobrze określona oraz:

- $|x_{k+1} - \alpha| \leq q|x_k - \alpha|;$

- $|x_{k+1} - \alpha| \leq C|x_k - \alpha|^2$ dla pewnej stałej C ;
- $\lim_{k \rightarrow +\infty} x_k = \alpha$.

Istotnym warunkiem jest tutaj bliskość początkowego przybliżenia do rozwiązania. Globalna zbieżność metody nie jest gwarantowana. Jednak przy dostatecznie małym kroku czasowym τ rozwiązania \bar{u} będą się od siebie mało różniły, dlatego ten warunek metody Newtona powinien być spełniony.

3.2. Użyte narzędzia

Użyjemy programu **FreeFem++** (www.freefem.org/ff++) do symulacji numerycznych równań różniczkowych cząstkowych. Wiele zagadnień nie omówionych dokładnie w tej pracy, a zastosowanych podczas obliczeń numerycznych, jest rozwiązanych w tym programie. Są to między innymi:

- tworzenie siatki dla przestrzeni Ω ,
- całkowanie funkcji z wskazanej przestrzeni Φ ,
- obliczanie współczynników macierzy M oraz wektora \bar{f} ,
- rozwiązywanie układu równań liniowych korzystając z istniejących pakietów i wskazanej metody,
- wizualizacja $u(t, \cdot)$ za pomocą poziomu.

Do tworzenia wykresów został użyty program **gnuplot**.

3.3. Przestrzeń funkcji próbnych

3.3.1. Siatka

Aby określić przestrzeń funkcji próbnych, potrzeba najpierw stworzyć siatkę dla przestrzeni Ω . Jest ona automatycznie generowana przez **FreeFem++** w następujący sposób:

- określamy przestrzeń Ω za pomocą parametryzacji jej brzegu,
- określamy gęstość węzłów siatki na brzegu,
- siatka jest automatycznie generowana korzystając z algorytmu Delaunay-Voronoi, zapewniając proporcjonalność gęstości jej węzłów do gęstości węzłów brzegowych.

Siatka zawiera elementy wielokątne o kątach wspólnie ograniczonych od dołu przez pewną stałą $\gamma > 0$, co oznacza, że nie jest zdegenerowana.

3.3.2. Funkcje kawałkami wielomianowe drugiego stopnia

Za przestrzeń funkcji próbnych weźmy funkcje ciągle kawałkami wielomianowe drugiego stopnia i oznaczmy ją P_2 . Macierz M będzie miała wartość 0 w polu a_{ji} , jeżeli przecięcie nośników funkcji φ_i i φ_j będzie miary zero. Należy zatem zadbać o odpowiedni dobór bazy przestrzeni Φ . Rozwiązanie tego problemu jest wewnętrzną częścią **FreeFem++**. Natomiast wymaganiem a priori stawianym rozwiązaniu u jest $u(t, \cdot) \in H^1(\Omega)$. Zauważmy, że funkcje z $H^1(\Omega)$ mogą być aproksymowane przez funkcje z P_2 z dokładnością $O(h^3)$ w normie L^2 i $O(h^2)$ w normie H^1 , gdzie h jest grubością siatki. Przez grubość siatki h rozumiemy minimalny promień kuli, w której można zawrzeć każdy z elementów siatki.

3.4. Rozwiązywanie układu równań liniowych

Dobieramy odpowiedni algorytm rozwiązujący otrzymany układ równań liniowych w zależności od użytego schematu. Założymy symetryczność macierzy D oraz niesymetryczność macierzy ∇f .

- Macierz M dla schematu prostego jest symetryczna i dodatnio określona, zatem w tym przypadku użyjemy metody CG.
- Macierz M dla schematu z metodą Newtona jest niesymetryczna, zatem w tym przypadku użyjemy metody GMRES.

3.5. Warunki stopu metod iteracyjnych

Użyliśmy dwóch metod iteracyjnych: rozwiązującej układ równań liniowych CG bądź GMRES oraz aproksymującej zamknięty schemat Eulera. Musimy ustalić dla nich warunki stopu.

W przypadku metody rozwiązującej układ równań liniowych $M\bar{u} = \bar{f}$ zatrzymujemy się, gdy

$$\|M\bar{u} - \bar{f}\| < \varepsilon_1.$$

W przypadku metody aproksymującej zamknięty schemat Eulera zatrzymujemy się, gdy

$$\left| \int_{\Omega} \left\langle \frac{u_{n+1}^m - u_n}{\tau}, u_{n+1}^m \right\rangle + \int_{\Omega} \langle D \nabla u_{n+1}^m, \nabla u_{n+1}^m \rangle - \int_{\Omega} \langle f(u_{n+1}^m), u_{n+1}^m \rangle \right| \leq \varepsilon_2,$$

bądź gdy

$$\|u_{n+1}^m - u_{n+1}^{m-1}\|_{L^2} \leq \varepsilon_3.$$

Zatrzymujemy się również, jeżeli liczba iteracji przekroczy pewną (małą) stałą K . Nie jest to zachowanie pożądane i nie powinno występować przy odpowiednio dobranych ε_i . Pierwsza z wartości powinna być równa 0 po rozwiązaniu metodą elementu skończonego (3.1), a druga mierzy postęp algorytmu.

Dobór odpowiednich stałych ε_i zależy od konkretnego zagadnienia, grubości siatki i oczekiwanej dokładności. Oczekujemy jednak, że ε_1 i ε_3 będą małymi wartościami w porównaniu z ε_2 . Dobranie zbyt małego ε_1 bądź ε_2 może powodować brak praktycznej zbieżności metody ze względu na możliwą precyzję metody CG bądź GMRES. ε_3 powinno być możliwie małe, ale nie przekraczające granicy dokładności metody użytej do całkowania różnic rozwiązań.

3.6. Oszacowanie błędu

Oszacujemy całkowity błąd metody zakładając odpowiednią regularność u . Odpowiednie oszacowania możemy znaleźć w [2] w rozdziale 10.3.5. na stronie 248. Należy zwrócić uwagę, że założenie w odpowiednich twierdzeniach, że $u(t, \cdot) \in H_0^1(\Omega)$ można zamienić na $u(t, \cdot) \in H^1(\Omega)$, jako że mamy jednoznaczność oraz zerowe warunki brzegowe Neumanna.

W schemacie wykonujemy $\frac{T}{\tau}$ iteracji po czasie, a wewnątrz pewną ograniczoną z góry przez stałą ilość iteracji aproksymowanego zamkniętego schematu Eulera, gdzie w każdym kroku iteracji rozwiązujemy równanie różniczkowe cząstkowe metodą elementu skończonego. Dodajemy do siebie błędy z schematu różnicowego dla czasu oraz metody elementu skończonego. Błąd schematu różnicowego dla czasu jest w naszym przypadku równy $O(\tau)$. Jak wynika z błędów aproksymacji podanego w sekcji 3.3.2, błąd metody elementu skończonego z przestrzenią funkcji kawałkami wielomianowych drugiego stopnia i siatce o grubości h wynosi $O(h^3)$ w normie L^2 i $O(h^2)$ w normie H^1 . Całość możemy zawrzeć w postaci poniższego twierdzenia.

Twierdzenie 4 (Twierdzenie oparte na rozdziale 10.3.5 książki [2] ze stron 244-249). *Niech u_h będzie rozwiązaniem dyskretyzacji równania parabolicznego (2.6) dla $t \in [0, T]$ po czasie zamkniętym schematem Eulera i po przestrzeni metodą elementu skończonego z przestrzenią funkcji kawałkami wielomianowych drugiego stopnia. Niech rozwiązanie u będzie istniało w przestrzeni $H^1(\Omega)$ i będzie jednoznaczne. Niech f jest funkcją ciągłą. Niech h jest grubością siatki (definicja w 3.3.2), niech siatka nie będzie zdegenerowana (definicja w 3.3.1) i niech krok czasowy jest równy τ . Wtedy:*

- $\|u - u_h\|_{L^\infty(0,T;L^2(\Omega))} = O(\tau + h^3);$
- $\|u - u_h\|_{L^\infty(0,T;H^1(\Omega))} = O(\tau + h^2).$

Rozdział 4

Automaty komórkowe

4.1. Wstęp

Jak można przeczytać w rozdziale pierwszym książki Ilachinskiego, [5], automaty komórkowe są klasą matematycznych układów istniejących w dyskretnej przestrzeni, charakteryzujące się lokalnymi, odbywającymi się równolegle w całej przestrzeni oddziaływaniami. Za twórców tego modelu uznaje się Johna von Neumanna i Stanisława Marcina Ulama. Automaty komórkowe posiadają proste do analizy mikroskopowe zmiany, ale makroskopowe zachowanie układu jest już zwykle bardzo skomplikowane. Do rozwoju i zwiększenia popularności automatów komórkowych przyczynił się John Horton Conway tworząc reguły dla automatu komórkowego „Gra w życie”. Posiada on bardzo zróżnicowaną dynamikę. W szczególności można w nim zasymulować Maszynę Turinga (o czym Ilachinski wspomina w rozdziale 1.2.1 książki [5]), czyli wykonanie dowolnego algorytmu. Zatem wiele problemów związanych z automatami komórkowymi jest nierozstrzygalnych (w ujęciu informatycznym, tzn. niemożliwych do automatycznego rozwiązania za pomocą maszyny Turinga).

Automaty komórkowe są tworzone dla określonych potrzeb, dlatego zarówno przestrzeń jak i reguły oddziaływań pomiędzy jej punktami są różne. W automatach komórkowych przeważnie można spotkać następujące własności:

- Przestrzeń w postaci jedno-, dwu- lub trójwymiarowej siatki komórek - najczęściej siatki złożonej z kostek, ale czasem np. z sześciokątów foremnych;
- Jednorodność komórek - wszystkie komórki są identyczne i rządzą nimi te same reguły. W przypadku istnienia warunków brzegowych, ta własność może być zaburzona;
- Dyskretną dynamikę - dyskretny czas oraz oddziaływania pomiędzy komórkami mające miejsce pomiędzy krokami czasowymi.
- Dyskretna ilość stanów, która charakteryzuje każdą komórkę - każdą komórkę charakteryzuje wyłącznie jej położenie w przestrzeni oraz przyjmowany przez nią stan (w określonym kroku czasowym);
- Lokalność zmian - oddziaływanie pomiędzy komórkami jest zmianą stanu komórki w zależności od stanu jej i sąsiednich komórek. Sąsiedztwo danej komórki definiujemy zwykle poprzez komórki posiadające z nią wspólny brzeg.

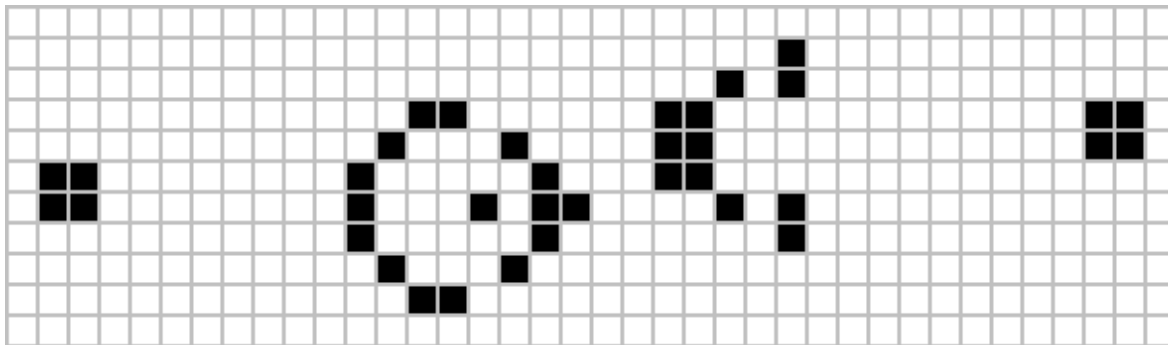
Funkcję definiującą stan komórki w następnym kroku czasowym w zależności od stanów sąsiednich komórek nazywamy regułą przejścia.

Automaty komórkowe posiadają szereg pożądanych cech, które sprawiają, że są one bardzo dobre do symulacji różnych zjawisk. Programy komputerowe symulujące automaty komórkowe są bardzo proste w implementacji. Dzięki lokalności oddziaływań i braku zależności stanu komórki od stanu innych komórek w obecnym kroku czasowym, automaty komórkowe są proste do masywnego zrównoleglenia. Siła wyrazu tych modeli jest wręcz nieograniczona (czego dowodzi możliwość stworzenia Maszyny Turinga w „Grze w życie”). Są bardzo dobre do symulacji modeli, którego mikroskopowe własności (odpowiadające regułom przejścia) dobrze znamy, a jesteśmy zainteresowani zachowaniami makroskopowymi układu. W szczególności wiele automatów komórkowych pozwala obserwować samoorganizujące się struktury. Przykładem może być automat komórkowy symulujący reakcję Biełousowa-Żabotyńskiego (omówiony na przykład w [7] i [5]). Szczegóły na temat porównania tego automatu komórkowego z modelem matematycznym reakcji Biełousowa-Żabotyńskiego znajdują się w sekcji 5.1. Inną dobrą własnością automatów komórkowych jest łatwość ich wizualizacji poprzez pokolorowanie komórek według ich stanów w danym kroku czasowym.

Wspomniany model „Gry w życie” jest następujący:

- Przestrzeń jest nieskończoną siatką kwadratową;
- Komórki posiadają dwa stany: żywa (czarna) i martwa (biała);
- Reguły przejścia są następujące: martwa komórka staje się żywa, jeżeli w poprzednim kroku czasowym dokładnie 3 z 8 otaczających ją komórek jest żywych, a żywa komórka umiera, jeżeli ilość otaczających ją komórek nie jest równa 2 lub 3.

Poniżej znajduje się wizualizacja części przestrzeni automatu komórkowego „Gry w życie” z strukturą o nazwie „Gosper Glider Gun”, emitującej poruszające się w przestrzeni struktury „Glider” (szybowce):



Rysunek 4.1: „Gosper Glider Gun” w automacie komórkowym „Gra w życie” (źródło: Wikipedia)

Jak wspomnieliśmy wyżej, mimo takich prostych reguł jest możliwe stworzenie w nim stanu początkowego, przy którym symuluje on działanie Maszyny Turinga. Przytaczamy ten model, by pokazać siłę automatów komórkowych oraz sposób ich definiowania.

W książce [9] na stronach 411-418 możemy znaleźć inne, biologiczne zastosowanie automatów komórkowych. Jest nim analiza modelu strategii szczepień priorytetowych (PVP) na zbiorze krów i borsuków. W przeciwieństwie do równań różniczkowych zwyczajnych opisujących ten model, w tym wypadku otrzymujemy również informacje o postępie choroby w przestrzeni.

W dalszej części stworzymy nieco bardziej skomplikowany automat komórkowy dla reakcji-dyfuzji określonej poprzez równania stechiometryczne z rozdziału 1. Podamy jego dyskretną przestrzeń odpowiadającą torusowi $[0, 1]^2$, opiszemy interpretację jego stanów oraz podamy regułę przejścia w sposób dwufazowy odpowiadającą połączeniu dyfuzji (losowego przemieszczania się cząsteczek w przestrzeni) oraz reakcji (zachodzącej w wyniku spotkania się dwóch cząsteczek w jednym miejscu). Podamy definicję reguły przejścia dla analizowanych przez nas modeli 2-, 2,5- i 3-cząsteczkowych.

4.2. Definicja modelu

Zdefiniujmy model automatu komórkowego dla reakcji-dyfuzji, który używamy w doświadczeniach. Przestrzenią automatu komórkowego jest kwadratowa siatka $N \times N$ komórek na torusie (w której przeciwległe brzegi są z sobą skleione). Dzięki temu unikamy rozpatrywania warunków brzegowych i operujemy na skończonej przestrzeni. Przyjmujemy sąsiedztwo pojedynczej komórki typu von Neumanna - sąsiedztwo w kształcie „+”, w którym sąsiadami komórki są 4 komórki posiadające z nią wspólny bok (komórka na górze, dole, po lewej i po prawej). Przy definicji sąsiedztwa komórek na brzegu pamiętamy o sklejeniu boków siatki. Dla wygody zdefiniujmy otoczenie komórki jako ją oraz jej sąsiedztwo. Stanem pojedynczej komórki jest trójka lub para liczb naturalnych oznaczająca ilość cząstek danego koloru znajdujących się w komórce. Standardowo czas w modelu jest dyskretny.

Regułę przejścia określmy za pomocą dwóch następujących po sobie faz:

- Faza dyfuzji - każda cząsteczka znajdująca się w danej komórce przemieszcza się do losowej komórki z otoczenia z równym prawdopodobieństwem (po $\frac{1}{5}$);
- Faza reakcji - jeżeli w komórce istnieją co najmniej dwie cząsteczki, to dwie losowo wybrane cząsteczki reagują z sobą zgodnie z równaniami stechiometrycznymi odpowiedniego modelu z rozdziału 1, tworząc dwie cząsteczki.

Formalnie, możemy przyporządkować każdej komórce ciąg zmiennych losowych indeksowanych czasem, od których będzie zależeć wybór kierunku dyfuzji cząsteczek i cząsteczek do reakcji (w sposób niezależny od siebie). Regułę przejścia dla danej komórki możemy wtedy określić jako funkcję stanu komórek z jej otoczenia w poprzednim kroku czasowym i wartości przyporządkowanych im zmiennych losowych w poprzednim kroku czasowym.

Ostatnim elementem jest określenie stanu początkowego komórek. Wraz z liczbą N zależy on od konkretnego modelu (3-, 2,5- bądź 2-cząsteczkowego) i eksperymentu.

4.3. Analiza modelu

Podobnie jak reakcje, model ten jest symetryczny ze względu na cząsteczki niebieskie i czerwone. Mówiąc więc o ich własnościach możemy skupić się na jednym kolorze.

Zauważmy, że ponieważ każda z rozpatrywanych reakcji posiada dwa substraty i dwa produkty, ilość cząstek w automacie pozostaje stała. Nazwijmy rozkładem masy stan przestrzeni $N \times N$ stworzony przez dodanie do siebie ilości cząstek poszczególnych kolorów w każdej z komórek. Ponieważ faza reakcji nie zmienia ilości cząstek, to rozkład masy w czasie jest taki sam, jak w automacie komórkowym z regułą przejścia w postaci samej fazy dyfuzji, czyli automatu komórkowego modelującego samą dyfuzję.

Kolejną własnością tego modelu jest prawdopodobieństwo zajścia w reakcję danej cząsteczki odwrotnie proporcjonalne do ilości cząsteczek w komórce (konkretniej równe $\frac{1}{n-1}$ zakładając, że w komórce znajduje się $n \geq 2$ cząsteczek).

4.3.1. Model 3-cząsteczkowy

Zanalizujmy sytuację, w której wytworzył się front zawierający niebieskie i białe cząsteczki po jednej stronie oraz czerwone i białe po drugiej. Białe cząsteczki w grupie cząsteczek kolorowych zanikają przy reakcji z cząsteczką kolorową, dlatego prawdopodobieństwo przetrwania białej cząsteczki w jednokolorowym morzu cząsteczek niebieskich jest równe prawdopodobieństwu nie zajścia reakcji z żadną inną cząsteczką i maleje wykładniczo z czasem. Dowód tego opiera się na fakcie, że (asymptotycznie lub przy losowych danych początkowych) prawdopodobieństwa trafienia cząsteczki na pustą komórkę jest ograniczone z dołu przez stałą oraz prawdopodobieństwo nie zajścia w reakcję w przypadku trafienia na komórkę niepustą jest również ograniczone z dołu przez stałą. Podobnie prawdopodobieństwo przetrwania cząsteczki czerwonej w morzu cząsteczek niebieskich również maleje wykładniczo z czasem, ponieważ spotkanie dwóch cząsteczek o różnych kolorach nie produkuje cząsteczek kolorowych, tylko białe. Dowód przebiega identycznie.

Spodziewamy się zatem, że szybko zostaną wytworzone obszary posiadające wewnątrz prawie wyłącznie cząsteczki kolorowe jednego typu. Obszary takie byłyby oddzielone od siebie frontami posiadającymi mieszaninę cząsteczek kolorowych oraz białych (wytworzonych w wyniku spotkań frontów). Ze względu na wykładniczo szybki zanik cząsteczek w morzu cząsteczek innego koloru spodziewamy się, że grubość tego frontu nie będzie duża.

4.3.2. Model 2,5-cząsteczkowy

Podobnie jak w przypadku modelu 3-cząsteczkowego, ilość pojedynczych cząsteczek czerwonych w morzu cząsteczek niebieskich z rosnącym prawdopodobieństwem powinna dążyć do zera, szczególnie, gdy średnia ilość cząsteczek niebieskich w komórce przekracza 1. Jednak w przeciwieństwie do modelu 3-cząsteczkowego, istnieje niezerowe prawdopodobieństwo zwiększania się ilości cząsteczek w kolejnych krokach czasowych. Dlatego również możemy spodziewać się pogrupowania przestrzeni w obszary o dominacji cząsteczek jednego koloru, ale z grubszym frontem o płynniejszym przejściu.

4.3.3. Model 2-cząsteczkowy

Przebieg reakcji w tym modelu nie jest w żaden sposób skorelowany z proporcją gęstości cząsteczek do siebie. Zatem mimo istnienia dyfuzji, wartość oczekiwana sumarycznej ilości cząsteczek danego typu pozostaje niezmienna. Zatem przebieg reakcji-dyfuzji jest z jakościowego punktu widzenia niezależną dyfuzją i problemem ruiny dla dwóch graczy o skończonych kapitałach.

4.4. Możliwe modyfikacje

4.4.1. Warunki brzegowe

Możemy sobie zadać pytanie w jaki sposób stworzyć warunki brzegowe dla naszego automatu komórkowego. Odpowiednik zerowych warunków brzegowych Neumanna można by stworzyć zmieniając przestrzeń z torusa w kwadrat (lub inną przestrzeń) i odpowiednio modyfikując definicję sąsiedztwa komórek brzegowych (obcinając go do przestrzeni).

Niezerowe warunki Neumanna byłyby trudniejsze do analizy, ale możemy zawsze określić, że pewna ilość cząsteczek danego koloru (być może losowa) dostaje się do brzegowych komórek albo ubywa z nich. W drugim przypadku jest problem, ponieważ nie możemy bez istotnej modyfikacji modelu zapewnić, że komórki brzegowe zawierają cząsteczki.

Warunki brzegowe Dirichleta możemy osiągnąć dodając dodatkową warstwę komórek na brzegu, która po każdej fazie dyfuzji jest przywracana do stanu właściwego warunku brzegowego.

4.4.2. Reakcja

Szybkość reakcji w tym modelu znacznie różni się od szybkości reakcji w równaniach stechiometrycznych (rozdział 1). W przypadku automatu komórkowego centralną rolę przyjmuje komórka. W rzeczywistości obliczane szybkości reakcji są oparte na prawdopodobieństwie zderzeń cząsteczek w przestrzeni. Możemy zmodyfikować odpowiednio nasz automat na przykład wybierając w losowej kolejności cząsteczki z komórki i z pewnym prawdopodobieństwem reagując je z innymi obecnymi w komórce.

4.4.3. Dyfuzja

W tym wypadku możemy za centralny obiekt przyjąć również komórkę, a nie cząsteczkę. Można wybierać losowo jedną cząsteczkę z komórki i dyfundować ją do jednej z komórek z otoczenia. Należy zauważyć jednak, że jeżeli każda z komórek z otoczenia zawiera przynajmniej 1 cząsteczkę, to wartość oczekiwana zmiany ilości cząsteczek w rozpatrywanej komórce wynosi zero. Wtedy dyfuzja w rozkładzie masy w automacie w ogóle by nie zachodziła.

Rozdział 5

Wyniki symulacji i porównanie

Przez symulacje numeryczne będziemy rozumieć wykonanie programu dla 3-cząsteczkowego modelu reakcji-dyfuzji (z reakcjami zdefiniowanymi w sekcji 1.1) rozwiązującego zdyskretyzowany układ równań różniczkowych cząstkowych (omówiony w rozdziale 2). Przez symulacje automatów komórkowych będziemy rozumieć wykonanie programu obliczającego kolejne stany automatów komórkowych (omówionych w rozdziale 4) dla modeli 2-, 2,5- i 3-cząsteczkowych. Porównamy wyniki jakościowe tych symulacji oraz podejmiemy próbę sprowadzenia ich do wspólnej przestrzeni i porównania za pomocą norm l_1 i l_2 .

Wykonywane były trzy rodzaje eksperymentów: symulacje numeryczne równań różniczkowych cząstkowych, symulacje automatów komórkowych oraz obliczenia porównujące te dwie metody (zwane dalej symulacjami porównawczymi). Obliczenia porównujące zawierają w sobie symulację automatu komórkowego połączone z zbieraniem odpowiednich statystyk. Omówimy sposób przeprowadzania poszczególnych symulacji oraz opiszemy znaczenie uzyskiwanych danych, a później przejdziemy do interpretacji wyników eksperymentów. Pełna lista wykonanych eksperymentów znajduje się w sekcji 6.8. Na początek jednak omówimy próbę porównania innego modelu automaty komórkowej, dla reakcji Biełousowa-Żabotyńskiego, z odpowiadającym mu układem równań różniczkowych cząstkowych.

5.1. Porównanie w przypadku reakcji Biełousowa-Żabotyńskiego

Automat komórkowy reakcji Biełousowa-Żabotyńskiego został porównany z układem równań różniczkowych cząstkowych reakcji Biełousowa-Żabotyńskiego w pracy [7]. Wizualnie wyniki obu symulacji były podobne, jednak różniły się od badanego modelu matematycznego. Brak było w automatach komórkowych pewnych charakterystycznych cech reakcji Biełousowa-Żabotyńskiego takich, jak występujące w modelu matematycznym spiralne struktury. Autor pracy skłonił się do hipotezy, że modele te nie są równoważne.

5.2. Symulacje numeryczne

Były to symulacje wykonywane za pomocą programu FreeFem++ zgodnie z metodami opisanymi w rozdziale 3. Symulacje przede wszystkim obejmowały dyskretyzację metodą elementu skończonego po przestrzeni (z przestrzenią funkcji ciągłych kawałkami wielomianowych drugiego stopnia) oraz zamkniętym schematem Eulera po czasie. Ze względu na nieliniowość równania, zamknięty schemat Eulera został zaimplementowany za pomocą metody Newtona.

W miarę rozwijania się skryptu, zbierane były różne dane i powiększała się ilość parametrów. Omówimy używane parametry oraz ich wpływ na wyniki. Później scharakteryzujemy

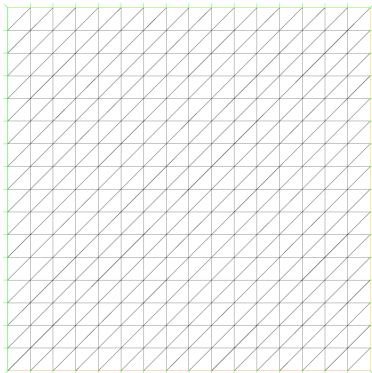
zbierane dane.

5.2.1. Parametry

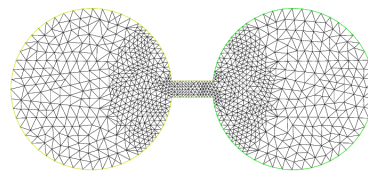
5.2.1.1. Przestrzeń

Używane były dwa kształty przestrzeni: torus $[0, 1]^2$ i hantla (dwie kule połączone paskiem). Gęstość siatki w przypadku torusa była jednorodna, ale różniła się pomiędzy doświadczeniami (była równa 16×16 lub 32×32). W Przypadku dwóch kul połączonych paskiem mogliśmy obserwować ciekawe efekty związane z potokiem średniokrzywiznowym na środku paska. W związku z występującymi często dużymi gradientami wewnątrz paska (dla dużych czasów) oraz nieróżniczkowalnością brzegu w miejscach połączenia kul i paska, siatka została podzielona na mniejsze trójkąty. Rozdrobnienie było tym większe, im bliżej trójkątne elementy znajdowały się środka figury. Rozdrobnienie polegało na podziale wnętrza trójkąta na trójkąty podobne o boku $\frac{1}{3}$, $\frac{1}{2}$ bądź 1 długości oryginalnego boku.

Dalsze rozdrobnienie siatki powoduje zwiększenie czasu obliczeń proporcjonalne do ilości trójkątów. Jest to najistotniejszy czynnik wpływający na szybkość obliczeń.



Rysunek 5.1.1: Torus



Rysunek 5.1.2: Hantla

5.2.1.2. Dane początkowe

Dane początkowe były różnych typów, w zależności od doświadczenia. Najczęściej były to dane losowe, z losowym wielomianem drugiego stopnia na każdym elemencie. Losowanie następowało przed ewentualnym podziałem siatki, zatem dane były losowane na elementach podobnej wielkości.

Im większe gradienty danych początkowych, tym dłużej trwa pierwsze kilka iteracji i tym mniejsza jest dokładność rozwiązania. Gradienty rozwiązań jednak szybko maleją, zatem nie ma to dużego wpływu na całkowity czas obliczeń.

5.2.1.3. Długość doświadczenia T

Doświadczenia były wykonywane dla czasów rzędu od 1 do 100 (co odpowiadało ok. 100-10000 krokom czasowym). Długość doświadczenia zależała od obserwowanych własności. Zwykle dobierana była tak, że po upływie tego czasu rozwiązanie było bliskie stałemu.

Czas trwania obliczeń jest proporcjonalny do T , zatem jeżeli nie znamy T potrzebnego do pewnego doświadczenia, możemy najpierw uruchomić symulację dla bardzo dużego T z niską dokładnością i grubszą siatką i zanotować interesującą nas wielkość. Możemy również nie ustalać konkretnego T , ale na przykład wyznaczyć koniec symulacji na podstawie normy

różnicy rozwiązania dla dwóch kolejnych kroków czasowych. W opisanych doświadczeniach zastosowano pierwszą z tych metod.

5.2.1.4. Macierz dyfuzji D

W doświadczeniach została użyta macierz dyfuzji $D = dI$, czyli wszystkie współczynniki dyfuzji były równe. Zachodzi zatem założenie większości twierdzeń z rozdziału 2, w których wiele własności miało miejsce wyłącznie przy równych współczynnikach dyfuzji.

5.2.1.5. Szybkość reakcji k

W równaniach numerycznych szybkość reakcji była skalowana przez stałą k . Na potrzeby zwykłych symulacji numerycznych była ona równa $k = 1$, ale na potrzeby porównania wyników numerycznych z automatem komórkowym została ustalona na wartość $k = 200$. W przypadku automatów komórkowych można zaobserwować wyraźnie istnienie obszarów zawierających prawie wyłącznie cząsteczki jednego koloru, zatem aby doświadczenia te można było porównywać, szybkość reakcji została znacznie zwiększona. W celu uzyskania lepszych porównań z automatami komórkowymi, wartość ta wymaga jeszcze dokładnej kalibracji. Przypomnijmy, że jak wynika z rozważań w sekcji 2.3, zwiększenie szybkości reakcji k razy jest równoważne przyspieszeniu czasu k razy i zamianę przestrzeni na \sqrt{k} razy większą w każdym wymiarze.

Musimy uważać ustawiając duży parametr k , ponieważ, jak omówiliśmy w sekcji 3.1.2, spowoduje to, że macierz rozwiązywanych układów liniowych (metodą GMRES) nie będzie dodatnio określona. Brak tej własności może spowodować znacznie wolniejsze zbieganie tej metody. Przy $k = 1000$ metoda Newtona używana w rozwiązaniu nie zbiegała w ciągu 10 iteracji, podczas gdy dla $k = 1$ zbiegała zwykle w ciągu około trzech iteracji, a przy dużych gradientach około siedmiu. W przypadku konieczności użycia dużych wartości k lepsze może być zastosowanie metody predictor-corrector (omówionej w sekcji 3.1.1).

5.2.1.6. Dokładność obliczeń

Użyte zostały cztery parametry kontrolujące dokładność obliczeń. Pierwszym z nich jest krok czasowy τ . W opisywanych doświadczeniach było użyte $\tau = 0.01$ bądź $\tau = 0.02$. Długość obliczeń jest odwrotnie proporcjonalna do tej wielkości.

Kolejnymi parametrami są liczby określające warunki stopu dla różnych metod iteracyjnych. Są to wartości ε_i opisane w sekcji 3.5. Użyte wartości różniły się w zależności od zastosowania i były równe $\varepsilon_1 = \gamma_1 10^{-11}$, $\varepsilon_2 = \gamma_2 10^{-10}$ i $\varepsilon_3 = \gamma_3 10^{-13}$, gdzie $\gamma_i \in [10^{-2}, 10^2]$ w zależności od doświadczenia.

Ostatnim parametrem jest maksymalna ilość iteracji metody Newtona. Była ona ustalona na 10 i ze względu na kwadratową zbieżność tej metody oczekiwane było, że nigdy nie zostanie ona przekroczona. Przekroczenie jej oznaczało zwykle bardzo złe uwarunkowanie problemu. Wystąpiło na przykład przy ustawieniu parametru szybkości reakcji na $k = 1000$.

5.2.1.7. Animacja

Pozostałe parametry doświadczenia były już techniczne i dotyczyły głównie sposobu generowania danych i zostały omówione w dziale 6. Do poprawnej interpretacji wyników warto przyrzeć się parametrowi szybkości animacji. Ze względu na bardzo dużą ilość kroków czasowych, nie dla każdego kroku czasowego obraz przestrzeni został w niej zawarty. Dla małych czasów $t < T_1$ (kiedy to rozwiązanie posiada zwykle bogatą dynamikę) każda klatka jest rejestrowana w animacji, po czym czas jest skalowany kwadratowo aż do czasu $t = T_2$, tzn. rośnie

ilość klatek odrzucanych aż do osiągnięcia pewnej granicy w $t = T_2$ (w której rozwiązanie jest już najczęściej praktycznie ustabilizowane), od której czas jest już skalowany liniowo. Ze względu na to, oglądając animacje należy wziąć poprawkę na skalowanie czasu przy ocenie szybkości zachodzenia zjawisk. W niektórych starszych wersjach skryptu skalowanie czasu może być inne (również nieliniowe).

5.2.2. Zbierane dane

Zbierane dane dzielą się na trzy kategorie: wykresy, animacje oraz dane wejściowe do symulacji porównawczych.

5.2.2.1. Animacje

Pierwszym rodzajem zbieranych danych są animacje wykresów funkcji b , n , c i $S = b + n + c$ w czasie. Czas jest reprezentowany w sposób naturalny, a wartości funkcji w przestrzeni za pomocą poziomic. Znaczenie kolorów różni się w czasie i jest skalowane w zależności od wartości obecnych na wykresie. Oprócz tego na górze wykresu znajdują się informacje (\mathbf{x} oznacza cząsteczki \mathbf{b} , \mathbf{n} , \mathbf{c}):

- której cząsteczki to wykres (pierwsza litera),
- jakie są współczynniki dyfuzji ($\mathbf{D}\mathbf{x}$),
- ile wynosi obecny czas (\mathbf{t} , przy czym należy zwrócić uwagę na nieliniowość czasu, jak opisaliśmy wyżej),
- jaka jest sumaryczna masa poszczególnych cząstek ($\mathbf{xL1}$, czyli norma $L^1(\Omega)$ poszczególnych funkcji),
- ile wynosi różnica masy względem masy początkowej ($S - S_0$ oznaczone jako \mathbf{SDiff} ; teoretycznie wartość ta powinna być stale równa 0, ale w rzeczywistości jest małą liczbą określającą w pewien sposób poprawność i dokładność obliczeń).

5.2.2.2. Wykresy

Wykresy są pogrupowane tematycznie. Poniżej opisujemy ich znaczenie:

- $\mathbf{diffL2}$ - $\|u_n - u_{n-1}\|_{L^2(\Omega)}$ - norma L^2 różnicy rozwiązania względem poprzedniego kroku czasowego;
- $\mathbf{diffSup}$ - $\|u_n - u_{n-1}\|_{L^\infty(\Omega)}$ - jak wyżej;
- \mathbf{bInf} , \mathbf{nInf} , \mathbf{cInf} - $\inf_{x \in \Omega} b(x, t)$ dla czasu t , analogicznie dla pozostałych kolorów, wartości te muszą pozostać nieujemne;
- \mathbf{bSup} , \mathbf{nSup} , \mathbf{cSup} - $\sup_{x \in \Omega} b(x, t)$ dla czasu t , analogicznie dla pozostałych kolorów;
- $\mathbf{bL1}$, $\mathbf{nL1}$, $\mathbf{cL1}$ - $\|b(\cdot, t)\|_{L^1(\Omega)}$ dla czasu t , analogicznie dla pozostałych kolorów;
- $\mathbf{bL2}$, $\mathbf{nL2}$, $\mathbf{cL2}$ - $\|b(\cdot, t)\|_{L^2(\Omega)}$ dla czasu t , analogicznie dla pozostałych kolorów;
- \mathbf{Savg} , \mathbf{Smin} , \mathbf{Smax} - odpowiednio wartość średnia, infimum i supremum funkcji S w czasie t , w przypadku równych współczynników dyfuzji oczekujemy, że \mathbf{Savg} będzie funkcją stałą, \mathbf{Smin} niemalejącą, a \mathbf{Smax} nierosnącą, co wynika z zasady maksimum;

- **Sdiff** (w starszych wersjach skryptu `totalDiff`) - wartość $S - \|S_0\|_{L^\infty(\Omega)}$, powinna być możliwie bliska zero;
- **rdIntWeak**, **rdIntWeakLog** - jest to wartość mówiąca o lokalnej dokładności obliczeń, będąca residuum zamkniętego schematu Eulera w postaci słabej:

$$\int_{\Omega} \left\langle \frac{u_n - u_{n-1}}{\tau}, u_n \right\rangle + \int_{\Omega} \langle D \nabla u_n, \nabla u_n \rangle - \int_{\Omega} \langle f(u_n), u_n \rangle.$$

Wartość ta powinna być możliwie mała podczas obliczeń. W przypadku dokładnego rozwiązania nieliniowego zdyskretyzowanego układu równań (3.1) wartość ta powinna być równa zero;

- **rdIntStrong** (obecna w starszych wersjach skryptu) - wartość mówiąca o lokalnej dokładności obliczeń, będąca residuum zamkniętego schematu Eulera w postaci mocnej z już obliczoną całką z Laplasjanu równą zero:

$$\int_{\Omega} \sum_{i=b,n,c} \frac{u_n^i - u_{n-1}^i}{\tau} - f_i(u_n).$$

Wartość ta jednak nie dostarczała bardziej interesujących danych niż **rdIntWeak**, jako że suma f^i jest zawsze równa zero (z dokładnością do zaokrągleń numerycznych), a wartość sumy składników różnicy dzielonej jest tak naprawdę pochodną skończoną wykresu `total`;

- **total** (obecna w starszych wersjach skryptu) - równa $\|S\|_{L^1(\Omega)}$ dla czasu t , powinna być to funkcja stała, zastąpiona przez `Savg`.

Oprócz tego jest jeszcze wykres przestrzeni elementu skończonego, na którym zaznaczone są kontury elementów i całej przestrzeni.

5.2.2.3. Dane do symulacji porównawczych

Generowane były również dane wejściowe do symulacji porównawczych zawierające siatkę $N \times N$ punktów wysokościowych wykresu rozwiązania. Omówimy je dokładniej dalej w sekcji 5.4.

5.3. Symulacje automatami komórkowymi

Symulacje modelu reakcji-dyfuzji automatem komórkowym zostały wykonane za pomocą napisania programu w C++. Był to automat komórkowy dla modeli 2-, 2.5- i 3-cząsteczkowych stworzony według zasad z rozdziału 4. Zostało przeprowadzonych też kilka doświadczeń w modelu ze zmodyfikowaną, osłabioną dyfuzją, który został opisany w sekcji 4.4.3. W zaprogramowaniu tych symulacji oraz doborze właściwego modelu pomógł profesor Jerzy Tyszkiewicz (MIMUW).

Jedyne dane, które były zbierane w tych doświadczeniach to animacje jednego z dwóch typów - dominacji kolorów cząstek i rozkładu masy. Różne były natomiast parametry i warunki początkowe. Omówmy je.

5.3.1. Parametry

Parametry automatu komórkowego były następujące:

- N - rozmiar automatu komórkowego równy $N = 256$ lub $N = 512$ w zależności od doświadczenia. Automat komórkowy był osadzony w przestrzeni siatki kwadratowej $N \times N$ na torusie;
- T - maksymalna ilość iteracji automatu komórkowego, zwykle równa $T = 250000$;
- **skip** - liczba całkowita oznaczająca co ile iteracji stan automatu jest wizualizowany jako kolejna klatka w animacji, równa 50;
- p - ciśnienie - ilość cząsteczek przypadających na 1 komórkę.

Złożoność obliczeń była równa $O(TN^2p)$. Algorytm jednak nie zawsze wykonywał T iteracji, co omówimy niżej.

5.3.2. Warunki początkowe

Warunki początkowe były dobierane w zależności od doświadczenia. Nie miały one wpływu na szybkość działania algorytmu. W przeciwieństwie do równań różniczkowych cząstkowych, nie mamy tutaj żadnych ograniczeń ze względu na warunki początkowe. Dobierane były one tak, aby początkowa ilość cząsteczek przypadających na 1 komórkę była równa p lub zero w pewnych przypadkach.

5.3.3. Warunek stopu

Automat komórkowy kończył symulację, gdy upłynęła maksymalna ilość iteracji T bądź ilość cząsteczek czerwonych lub niebieskich wynosiła zero. Automat przeważnie albo ustabilizował się w pewnej klasie stanów dla dużych czasów albo zakończył działanie znacznie wcześniej, gdy jedna z cząsteczek kolorowych przestała istnieć.

5.3.4. Animacja kolorów cząsteczek

W pierwszym rodzaju animacji stany automatu komórkowego zostały zwizualizowane poprzez klatki $N \times N$ pikseli, w których kolor piksela odpowiadał zawartości odpowiedniej komórki. Gdy komórka zawierała wyłącznie cząsteczki niebieskie, piksel był niebieski. Gdy czerwone - czerwony. Gdy zawierała cząsteczki obu typów - zielony. W przeciwnym wypadku biały. Jako że cząsteczki białe występowały praktycznie wyłącznie w miejscach, w których cząsteczki czerwone i niebieskie były zmieszane, zostały one pominięte w wizualizacji.

5.3.5. Animacja rozkładu masy

W drugim rodzaju animacji piksele odpowiadały ilości cząsteczek w komórce. Kolor biały odpowiadał braku cząsteczek, a kolor czarny maksymalnej ilości cząsteczek w danej chwili. Animacja ta odpowiadała stanom automatu komórkowego dla dyfuzji, co zostało dokładniej opisane w rozdziale 4.

5.4. Symulacje porównawcze

Symulacje porównawcze mają na celu sprowadzenie wyników symulacji numerycznych i automatów komórkowych do wspólnej przestrzeni, a następnie ich porównanie. Przedstawimy odpowiednie transformacje, a następnie omówimy parametry, które możemy kontrolować starając się jak najlepiej przybliżyć te rozwiązania. W dalszej części wymienimy zbierane dane.

5.4.1. Sprowadzanie do wspólnej przestrzeni

Kulminacją prac nad dwoma powyższymi modelami symulacji komputerowych modelu reakcji-dyfuzji była próba ich porównania. Na początek zauważmy, że przestrzenie tych modeli się zasadniczo różnią. Aby móc porównać ze sobą wyniki tych symulacji w możliwie ścisły sposób, dobrze jest sprowadzić te obiekty do wspólnej przestrzeni z dobrze określoną metryką. Dla ustalenia uwagi przyjmijmy, że pracujemy na przestrzeni odpowiadającej torusowi $\Omega = [0, 1]^2$ w obu przypadkach.

W przypadku równań różniczkowych używana przestrzeń jest równa przestrzeni Ω . Z kolei w przypadku automatów komórkowych naturalne wydaje się być przyporządkowanie stanów jego komórek kwadratowi rozmiaru $\frac{1}{N} \times \frac{1}{N}$, tworząc funkcję kawałkami (kwadratami) stałą. Ze względu na niską ilość makroskopowych informacji uzyskiwanych z małego obszaru dla automatów komórkowych, ich przestrzeń jest podzielona na o wiele więcej (rzędu dziesiątek razy) elementów niż przestrzeń elementu skończonego. Ponadto, ze względu na sposób interpretacji jego stanów w przestrzeni Ω , uzyskujemy funkcje nie tylko nieciągłe, ale także o dużej „wariancji”.

Jedną z dróg sprowadzenia tych dwóch obiektów do jednej przestrzeni jest próba wygładzenia funkcji odpowiadającej stanowi automatu komórkowego. Zastanówmy się nad znaczeniem tego stanu. Składa się on z komórek wypełnionych cząsteczkami. Ruch cząsteczek jest losowy, a dopiero ruch większej ilości cząsteczek na raz staje się dla nas interesujący. Ponadto tak naprawdę nie interesuje nas pozycja cząsteczki w konkretnym momencie czasu, ale raczej w pewnym przedziale czasu. Rozpatrzmy więc równanie ciepła (równoważne równaniu dyfuzji z równymi współczynnikami) z warunkiem początkowym równym delcie Diraca. Rozwiązaniem jest rozkład normalny z wariancją proporcjonalną do współczynnika dyfuzji i czasu. Spróbujmy zatem najpierw zdyfundować rozwiązanie odpowiadające automatu komórkowemu, a dopiero później porównać je z rozwiązaniem równań różniczkowych w przestrzeni Ω .

Ze względu na bardzo mały rozmiar pojedynczej komórki automatu komórkowego, do zasympulowania właściwej dyfuzji musielibyśmy użyć przestrzeni elementu skończonego o bardzo dużej ilości elementów. Porównanie serii tysięcy kroków czasowych automatu komórkowego wymagałoby bardzo dużej mocy obliczeniowej. Możemy zrobić to jednak w prostszy sposób.

Stwórzmy przestrzeń Ξ składającą się z funkcji określonych na siatce $N \times N$ z wartościami w zbiorze nieujemnych liczb rzeczywistych ($\Xi \ni \xi: \{1..N\}^2 \mapsto [0, +\infty)$). Będziemy do niej sprowadzać funkcje z przestrzeni Ω oraz funkcje z przestrzeni automatu komórkowego dla ustalonych czasów. Funkcję $v: [0, 1]^2 \mapsto [0, +\infty)$ odpowiadającą gęstości jednej z cząsteczek dla ustalonego czasu t zamienimy na funkcję ξ_r biorąc jej punkty wysokościowe w środkach „komórek”:

$$\xi_r(i, j) = v\left(\frac{i - 0.5}{N}, \frac{j - 0.5}{N}\right).$$

Z kolei chcąc sprowadzić stan automatu komórkowego w kroku czasowym obcięty do rozkładu cząsteczki jednego koloru $v: \{1..N\}^2 \mapsto \mathbb{N}$ będziemy używać gęstości dwuwymiarowego

rozkładu normalnego o wariancji σ^2 :

$$\xi_a(i, j) = \sum_{k, l=1..N} v(k, l) C e^{-\frac{|(k-i, l-j)|^2}{2\sigma^2}},$$

gdzie C jest odpowiednią stałą normującą, a $|\cdot|$ metryką euklidesową dwuwymiarową. Ze względu na dyskretną przestrzeń C nie jest równe $2\pi\sigma$, chociaż jest bliskie tej wartości. Obliczenie powyższej sumy dla N^2 punktów wymagało by $O(N^4)$ operacji, co jest zdecydowanie zbyt dużą ilością. Jako że gęstość rozkładu normalnego maleje wykładniczo szybko możemy ograniczyć się tylko do punktów leżących w promieniu r od badanego punktu otrzymując złożoność $O(N^2 r^2)$. Dla $r = 4$ bądź $r = 8$ taka wartość jest już akceptowalna. Zauważmy, że dla przypadku $r = 0$ otrzymamy transformację identycznościową. Ostateczna wersja jest następująca:

$$\begin{aligned} \xi(i, j) &= \sum_{k, l=1..N} v(k, l) C_r e^{-\frac{|(k-i, l-j)|^2}{2\sigma^2}} 1_{|(k-i, l-j)| \leq r}, \\ C_r &= \left(\sum_{k, l=1..N} e^{-\frac{|(k-i, l-j)|^2}{2\sigma^2}} 1_{|(k-i, l-j)| \leq r} \right)^{-1}. \end{aligned}$$

Na przestrzeni Ξ mamy w naturalny sposób określone normy l_p , $1 \leq p \leq \infty$. Możemy zatem porównywać w niej funkcje odpowiadające rozwiązaniom numerycznym równań różniczkowych i stanom automatów komórkowych.

5.4.2. Sposób porównania

Aby porównać rozwiązania, najpierw rozwiążemy numerycznie zagadnienie metodą elementu skończonego. Następnie wybierzemy pewną chwilę $0 \leq t_0 < T$ i przekształcimy rozwiązanie $u(t_0, \cdot)$ na odpowiedni stan początkowy automatu komórkowego (proces ten opiszemy poniżej w 5.4.3). Następnie uruchomimy symulację automatu komórkowego. W dalszej części będziemy przekształcać rozwiązania w opisany wyżej w 5.4.1 sposób do przestrzeni $[t_0, T] \times \Xi$. Ponieważ w rzeczywistości w obu przypadkach operujemy na dyskretnym czasie, założymy, że ilość iteracji automatu komórkowego odpowiadających jednemu krokowi czasowemu metody elementu skończonego jest pewną liczbą naturalną Q .

Ustalmy, że polem powierzchni w przestrzeni Ξ będziemy nazywać ilość punktów podzielonych przez N^2 . Za miarę odległości przyjmijmy odległość euklidesową z przestrzeni torusa $[0, 1]^2$, któremu odpowiada przestrzeń Ξ . Funkcje $\xi \in \Xi$ dla odpowiednich cząsteczek i ustalonego czasu nazywamy zagęszczeniami cząsteczek w przestrzeni Ξ .

W programie symulującym porównanie, oprócz samych symulacji numerycznych i automatu komórkowego, zbierane są następujące dane:

- **area cell|FEM B|N|C** - pole powierzchni dominacji cząsteczek - rozumiemy przez to pole powierzchni na którym cząsteczka danego koloru posiada zagęszczenie większe niż 50% sumy zagęszczeń wszystkich cząsteczek;
- **avg cell|FEM B|N|C** - średnią wartość zagęszczenia poszczególnych cząsteczek;
- **11, 12** - normę l_1 i l_2 różnicy funkcji ξ_r (odpowiadającej rozwiązaniu równań różniczkowych) i ξ_a (odpowiadającej stanom automatu komórkowego).

Zauważmy, że badanie infimum bądź supremum różnicy $\xi_a - \xi_r$ nie ma sensu, ponieważ sam proces przekształcania jest zbyt daleki od dokładnego.

5.4.3. Stan początkowy dla automatu komórkowego

Aby powyższe porównanie mogło nastąpić, musimy w pewien sposób uzgodnić warunki początkowe symulacji numerycznych i automatu komórkowego. Korzystając z tego, że rozwiązania dla automatu komórkowego i tak będą później dyfundowane oraz siatka automatu komórkowego jest dużo gęstsza od siatki metody elementu skończonego proponujemy następujące przekształcenie funkcji $u: \Omega \mapsto [0, +\infty)$ w stan automatu komórkowego $v: \{1..N\}^2 \mapsto \mathbb{N}$:

$$v(i, j) = [u(x, y)] + X_{ij}(\{u(x, y)\}), \\ X_{ij} \in \{0, 1\}, \quad X_{ij} \text{ iid}, \quad \mathbb{P}(X_{ij}(x) = 1) = 1 - \mathbb{P}(X_{ij}(x) = 0) = x.$$

$[\cdot]$ i $\{\cdot\}$ oznaczają odpowiednio część całkowitą i ułamkową. Dodajmy jeszcze do tego parametr p oznaczający ciśnienie w automacie komórkowym. Aby je zastosować wystarczy pomnożyć wartości funkcji u przez p przed zastosowaniem powyższej transformacji.

5.4.4. Parametry

Omówmy podane wcześniej w tym rozdziale parametry mające istotny wpływ na opisywane porównanie:

- t_0 - czas przeprowadzania symulacji numerycznej, zanim zostanie ona zamieniona w dane początkowe dla automatu komórkowego, a wyniki zaczną być porównywane. W doświadczeniach ustawione na $t_0 = 0$ dla ręcznie podanych danych i na $t_0 = 30$ dla danych losowych;
- k - skalowanie szybkości równania reakcji-dyfuzji w symulacjach numerycznych. Można zaobserwować, że automaty komórkowe posiadają istotnie szybszą reakcję (istotnie wolniejszą dyfuzję) niż równania różniczkowe, zatem aby zachowanie tych dwóch modeli było zbliżone należy dobrać dosyć duże k . W naszych doświadczeniach używamy $k = 200$;
- τ - długość kroku czasowego w metodzie numerycznej. Zwiększenie tej wartości c razy bez zmiany innych wartości spowoduje zwiększenie faktycznej szybkości rozwiązania numerycznego względem automatu komórkowego c razy. W doświadczeniach używamy $\tau = 0.01$ i $\tau = 0.02$;
- N - równe boku siatki porównywanego automatu). Ustalając N pamiętajmy, że im większe N tym wolniejsza dyfuzja dla automatu komórkowego w sensie szybkości rozchodzenia się cząsteczek w Ξ ;
- σ^2 oraz r - opisane w 5.4.1. Ustalenie odpowiedniego σ^2 oraz możliwie dużego r może poprawić normę odległości porównywanych rozwiązań. W doświadczeniach używamy zwykle $\sigma^2 = 1$ i $r = 4$;
- p - ciśnienie porównywanego automatu komórkowego. W naturalny sposób norma odległości porównywanych rozwiązań powinna się zwiększyć około p razy względem wersji $p = 1$. W praktyce jednak można zaobserwować mniejszą zmianę, co może oznaczać, że warto analizować również normę odległości podzieloną przez p . W doświadczeniach używane jest zwykle $p = 1$;
- Q - szybkość względna symulacji numerycznych względem automatów komórkowych. Tą wartość musimy dobrać tak, by wyniki symulacji numerycznych i automatu komórkowego były możliwie dobrze zsynchronizowane w czasie. Eksperymenty dla pozostałych

wartości jak powyżej wykazały, że wartość ta jest równa w przybliżeniu $Q = 270$. Należy tę wartość zmieniać wraz z zmianami τ . Jest ona trudna do precyzyjnego szacowania, w szczególności gdy zachowanie automatu komórkowego i symulacji numerycznych zaczyna rozbiegać.

Możemy zauważyć, że odpowiedni dobór tych parametrów zależy od siebie nawzajem, w szczególności w przypadku τ , N i Q . Ponieważ zasięg dyfuzji jest równy jednej komórce w przypadku automatu komórkowego, N nie może być dobrane jak największe, lecz musi być dobrane tak, by całość miała odpowiednio dużą szybkość dyfuzji. Modyfikacje tego automatu komórkowego mogłyby pomóc rozwiązać ten problem.

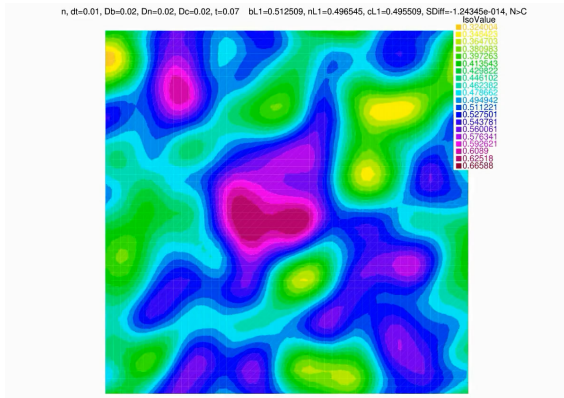
5.5. Obserwacje

W tej sekcji omówimy wyniki obserwacji symulacji numerycznych, automatów komórkowych i ich porównań.

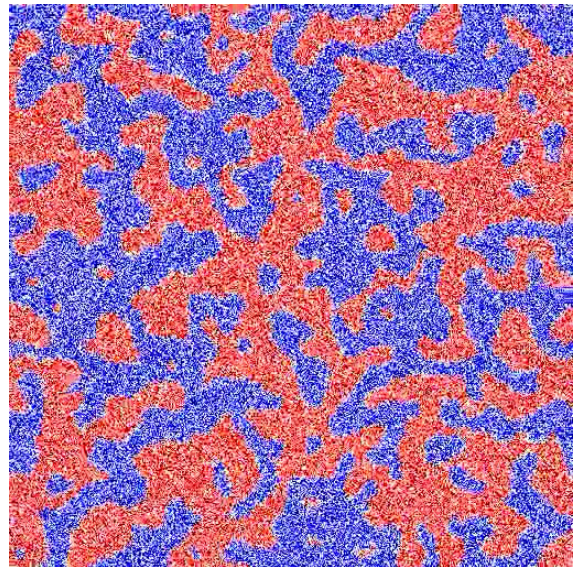
5.5.1. Fronty

Obserwując animacje o losowych danych początkowych można zauważyć bardzo szybkie wygładzanie się rozwiązania oraz wytworzenie się obszarów o wyraźnej dominacji cząsteczek jednego z kolorów. Dotyczy to zarówno symulacji numerycznych (dla praktycznie każdych parametrów), automatów komórkowych 3-cząsteczkowych i 2,5-cząsteczkowych. W przypadku automatów komórkowych 2-cząsteczkowych nie widać tworzenia się jakichkolwiek struktur.

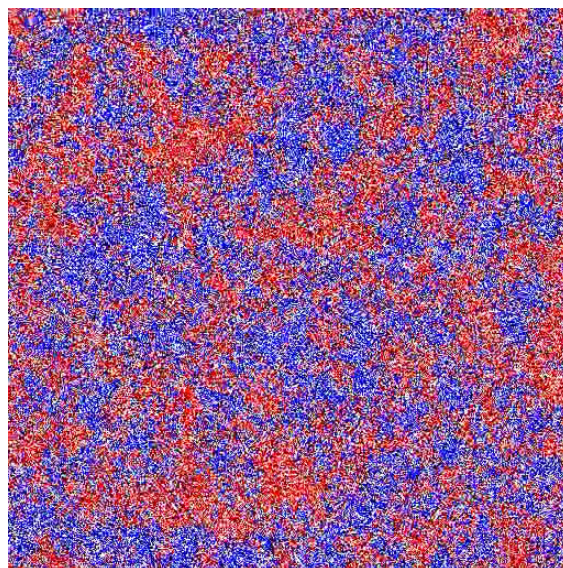
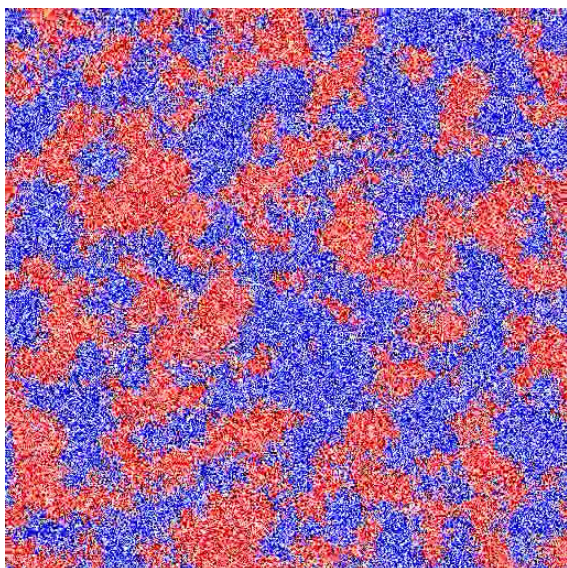
Poniżej znajdują się obrazy przestrzeni dla stosunkowo niewielkich czasów, w których widać potwierdzenie tych obserwacji. W przypadku automatów komórkowych można zauważyć, że najszybciej fronty tworzą się w automacie 3-cząsteczkowym i są przy tym bardzo gładkie. Fronty w automacie 2,5-komórkowym są stosunkowo mało wyraźne i mniej stabilne. W przypadku symulacji numerycznych każda poziomicca zachowuje się jak osobny front i jest wygładzana w miarę upływu czasu.



Rysunek 5.2.1: Poziomicce w symulacjach numerycznych

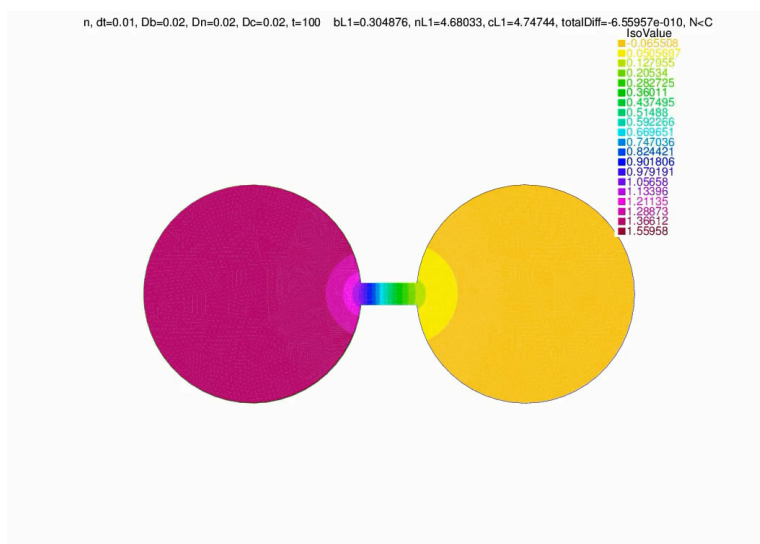


Rysunek 5.2.2: Front dla automatu 3-cząsteczkowego



5.5.2. Zachowanie asymptotyczne

W przypadku symulacji numerycznych możemy zaobserwować jeden z dwóch rodzajów zachowań asymptotycznych. W stabilnym stanie albo przestrzeń jest wypełniona jednym z dwóch kolorów (niebieskim albo czerwonym) albo istnieje stabilny front na wąskim przesmyku, a dominacja cząsteczek różnych kolorów po obu jego stronach. W przypadku przestrzeni torusa nie zaobserwowano utrzymywania się frontów dla dużych czasów.



Rysunek 5.4: Front utworzony w wąskim przesmyku

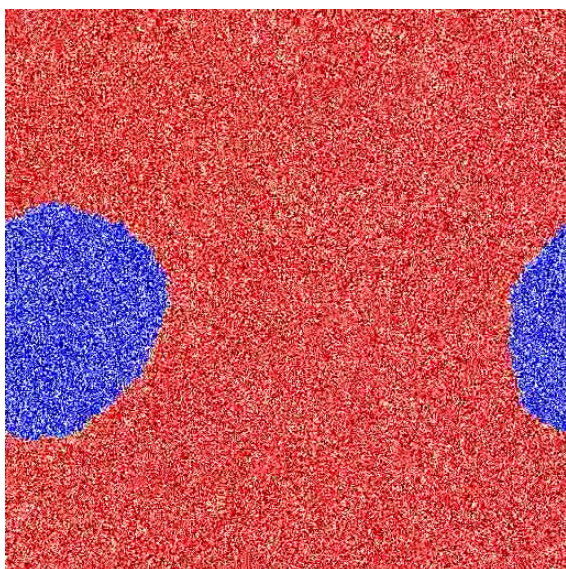
Ze względu na skończoną przestrzeń automatu komórkowego, istnieje niezerowe (aczkolwiek bardzo małe) prawdopodobieństwo, że w $N + 3S$ krokach (gdzie N to rozmiar torusa, a S ilość wszystkich cząsteczek) wszystkie cząsteczki przejdą do jednego punktu, a następnie przereagują na cząsteczki jednego koloru. W związku z tym dla nieskończonego czasu zdarzenie to zajdzie kiedyś prawie na pewno, więc przestrzeń torusa zaczęła wypełniać cząsteczki

tylko jednego koloru. Będziemy się jednak zajmować ewolucją automatów komórkowych dla dużych, ale skończonych czasów. W praktyce automat 2-cząsteczkowy nie zmienia swojego zachowania przez cały czas symulacji, a przy automatach 2.5- i 3-cząsteczkowych obserwujemy dwa rodzaje „asymptotycznych” zachowań:

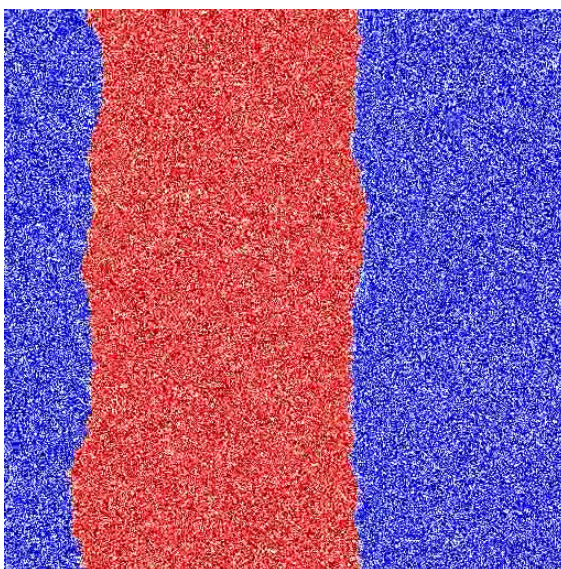
- Cząsteczki jednego koloru utworzą jednospójne obszary, które powoli zaczną zanikać, aż przestrzeń będzie zawierać tylko cząsteczki drugiego koloru;
- Cząsteczki jednego koloru utworzą pas (potencjalnie możliwe że wiele pasów) i ten stan się ustabilizuje. Od tego momentu zadanie zamieni się w symulację ruiny dwóch graczy o skończonych kapitałach, gdzie kapitałem kolorowych cząsteczek (graczy) jest grubość pasa. Można zaobserwować, że pasy te są wyłącznie poziome albo pionowe, nigdy skośne. Prawdopodobnie jest to efektem siatki i reguł, które nie są anizotropowe. Poza tym poruszają się bardzo wolno w losowych kierunkach (co jest spodziewanym zjawiskiem ze względu na losowy charakter reguł automatu komórkowego).

Przeprowadzono symulacje porównawcze dla danych początkowych w postaci wąskiego pionowego pasa. W przypadku symulacji numerycznych zaniknął on, a w przypadku automatu komórkowego ten stan się ustabilizował. Wynik ten, pozwalający odróżnić od siebie te dwa rodzaje symulacji, omawiamy dokładniej w sekcji 5.5.5.

Kolejne symulacje dotyczyły anizotropii. Danymi początkowymi był skośny (obrócony w kierunku $(1,1)$) pas o szerokości $\frac{1}{2}$ wypełniający dokładnie połowę przestrzeni. Zarówno w przypadku symulacji numerycznych (które z natury badanych równań są anizotropowe), jak i automatów komórkowych stan ten się utrzymywał. Może to oznaczać, że izotropowe reguły automatu komórkowego mają znacznie większe znaczenie w procesie formowania się tego pasa, niż jego stabilizacji.



Rysunek 5.5.1: Jednospójny zanikający obszar w automacie 3-cząsteczkowym



Rysunek 5.5.2: Pas w automacie 3-cząsteczkowym

5.5.3. Potok średniokrzywiznowy

W każdej z animacji dla symulacji numerycznych albo automatów komórkowych 2,5- lub 3-cząsteczkowych możemy zaobserwować potok średniokrzywiznowy. W przypadku symulacji

numerycznych poziomice rozwiązania zachowują się zgodnie z zasadami potoku średniokrzywiznowego, a w przypadku automatów komórkowych frontu utworzonego pomiędzy cząsteczkami czerwonymi i niebieskimi zachowuje się w ten sposób. Omówmy dokładniej potok średniokrzywiznowy i wykonajmy doświadczenie, aby go zaobserwować.

Potok średniokrzywiznowy (opisany we wprowadzeniu) definiuje się za pomocą wzoru

$$\bar{v} = -\kappa \bar{n},$$

gdzie \bar{v} jest prędkością krzywej w danym punkcie, κ jej krzywizną, a \bar{n} wektorem normalnym. Potok średniokrzywiznowy najłatwiej zanalizować na przykładzie okręgu. Jeżeli model ma własność potoku średniokrzywiznowego dla brzegu okręgu, to powinien się on kurczyć z odpowiednią szybkością. Krzywizna okręgu o promieniu r wynosi $\frac{1}{r}$, a wektor normalny jest zwrócony na zewnątrz. Zatem równanie różniczkowe opisujące zmianę promienia okręgu w czasie jest następujące:

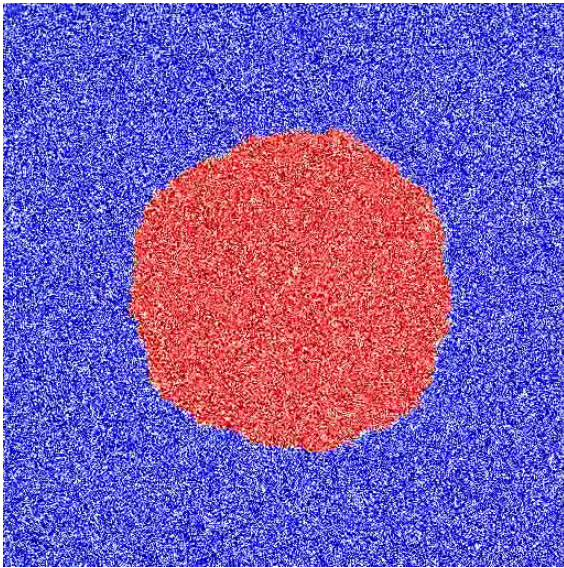
$$\dot{r} = -\frac{1}{r}, \quad r(0) = r_0.$$

Rozwiązując to równanie otrzymujemy:

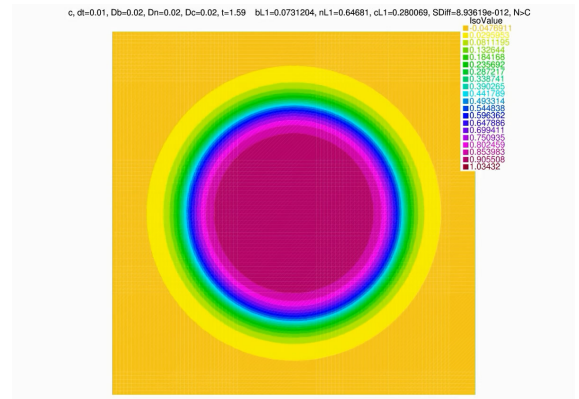
$$r(t) = \sqrt{r_0^2 - 2t}, \quad \pi r^2 = \pi(r_0^2 - 2t).$$

Zatem tworząc w eksperymentach koło z cząsteczek czerwonych w morzu cząsteczek niebieskich spodziewamy się liniowego zaniku powierzchni koła. Po dodaniu ewentualnych zaburzeń zauważmy, że kształt elipsy zbliża się do kształtu koła (ponieważ elipsa ma tym większą krzywiznę w punkcie, im bardziej jest on oddalony od jej środka ciężkości). Możemy więc wysnuć hipotezę, że mimo wprowadzenia zaburzeń, kształt koła powinien być zachowany. Jest to szczególnie istotne w przypadku automatów komórkowych, w których zaburzenia są nieuniknione.

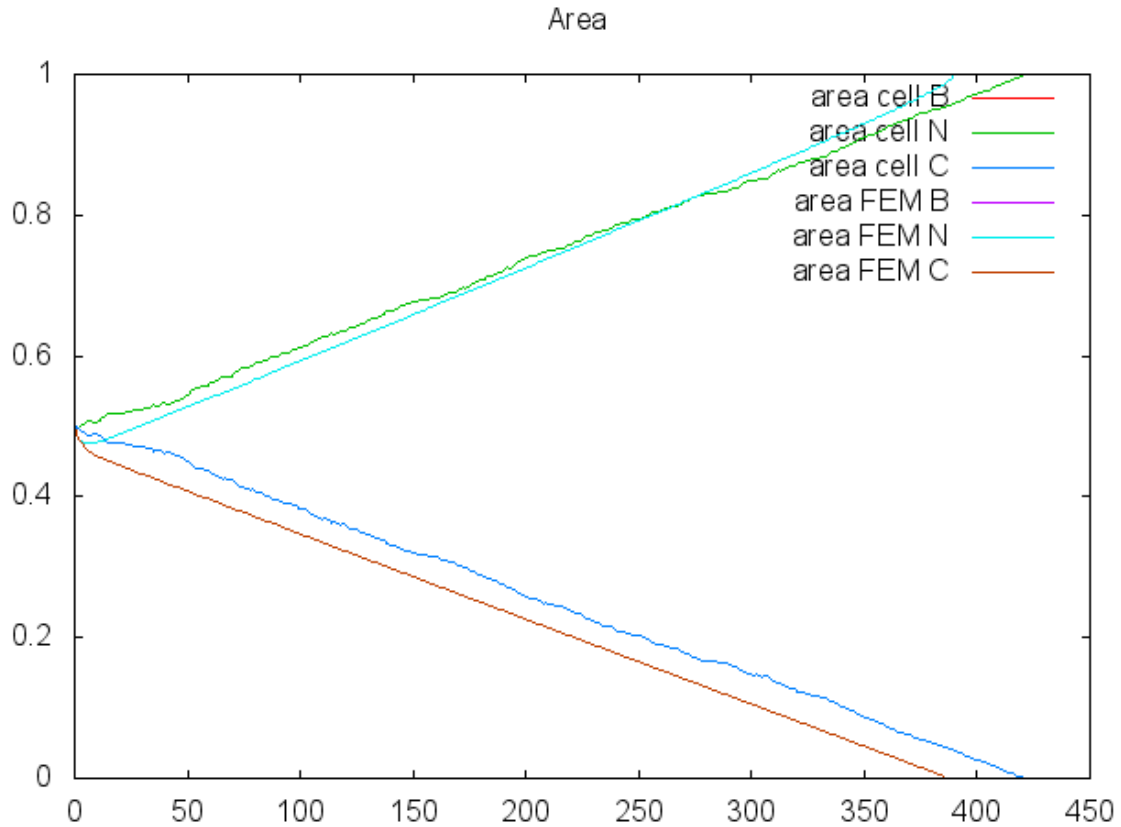
Do eksperymentów musimy jeszcze zdefiniować granicę badanego obszaru. Niech będzie to pole powierzchni dominacji cząsteczek określonego koloru, zgodnie z wcześniejszą definicją. Przyjrzyjmy się otrzymanym wynikom.



Rysunek 5.6.1: Kula w automacie komórkowym



Rysunek 5.6.2: Kula w symulacji numerycznej



Rysunek 5.7: Wykres pola powierzchni cząsteczek przy porównaniu

Możemy zauważyć, że zarówno w przypadku wyników symulacji 3-cząsteczkowego automatu komórkowego, jak i w przypadku symulacji numerycznych, funkcja pola powierzchni cząsteczki czerwonej od czasu jest w dobrym przybliżeniu liniowa. Początkowe zaburzenie w przypadku symulacji numerycznych wynika z wytworzenia się cząsteczek białych, a przypadku automatu komórkowego obserwujemy losowe zaburzenia związane z jego budową. W symulacjach numerycznych kształt kulisty jest doskonale zachowany, a w automatach komórkowych zachowany z dokładnością do małych, losowych zaburzeń.

5.5.4. Norma odległości

5.5.4.1. Cel

Normy odległości rozwiązań z symulacji numerycznych i automatów komórkowe były porównywane w przestrzeni Ξ zgodnie z metodą opisaną w 5.4. Rozważane były trzy typy danych początkowych: losowe, kula cząsteczek czerwonych w morzu niebieskich i pas ustawiony pod kątem 0° , 45° lub 25.5° . Zachowanie normy w czasie zależało od wszystkich parametrów wymienionych w 5.4.4.

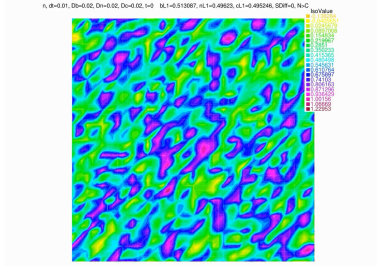
Zadanie porównania za pomocą normy w przestrzeni Ξ wymaga jak najlepszego skalibrowania parametrów, tzn. takiego ich dobrania, by norma różnicy dla danego zjawiska miała najmniejszą wartość oczekiwaną. Zaprezentujemy wyniki eksperymentów dla różnych parametrów i omówimy je.

5.5.4.2. Szybkość reakcji k

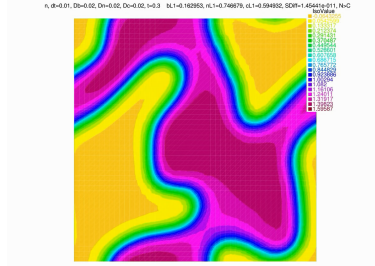
Przedstawione wyniki dotyczą porównania symulacji numerycznych z przeskalowaną szybkością reakcji $k = 200$ razy z automatem komórkowym. W symulacjach numerycznych dla małych wartości k obserwujemy stosunkowo małe różnice w zagęszczeniu cząstek poszczególnych rodzajów w punktach. Ponieważ komórki posiadają zwykle cząsteczki wyłącznie jednego typu, porównanie rozwiązań przy małych k nie daje dobrych rezultatów.

5.5.4.3. Czas początkowy t_0

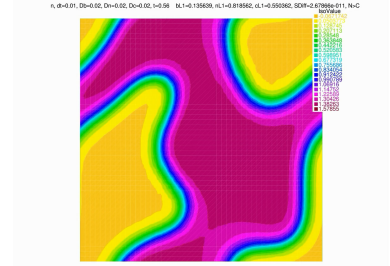
Jak wynikało z doświadczeń, aby osiągnąć dobre wyniki, dane początkowe dla automatu komórkowego pobierane z rozwiązania numerycznego muszą przypominać stan automatu komórkowego po pierwszej fazie stabilizacji, tzn. muszą już istnieć wyraźne fronty. Dane początkowe dla równania różniczkowego zawierające gładkie obszary, na których zagęszczenia cząsteczek są funkcjami stałymi niezerowymi tylko na jednej współrzędnej dobrze się do tego celu nadawały. Z kolei złe rezultaty dawały dane losowe. Badane zachowanie równań różniczkowych w czasie przedstawiało się jak poniżej.



Rysunek 5.8.1: Losowe dane początkowe ($t = 0$)

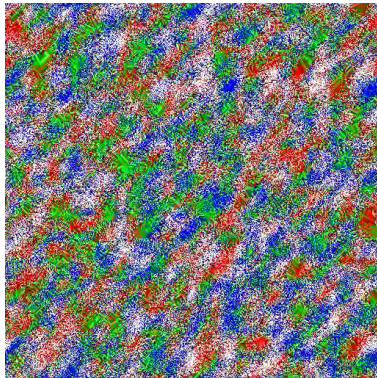


Rysunek 5.8.2: Stan układu po $t = 3$

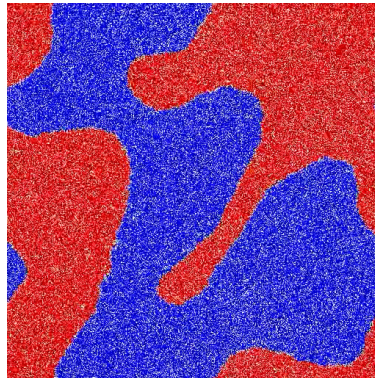


Rysunek 5.8.3: Dalsza ewolucja układu

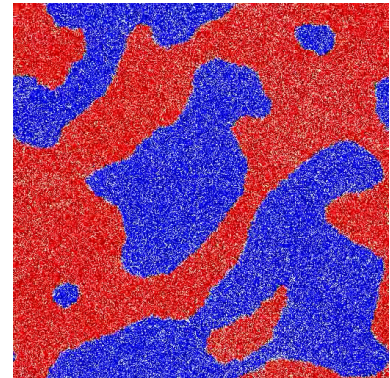
Następnie wykonywane były testy na automatach komórkowych, w których za chwilę początkową do pobrania danych użyto $t_0 = 0$ (brak stabilizacji - losowe dane) i $t_0 = 3$ (układ po wstępnej fazie). Wyniki dla automatu komórkowego z $t_0 = 0$ nie dość, że odbiegały od wyników symulacji numerycznych, to jeszcze znacząco się od siebie różniły za sprawą losowej ewolucji automatów komórkowych oraz losowej zamiany danych symulacji numerycznych na dane początkowe automatu komórkowego.



Rysunek 5.9.1: Stan początkowy automatu komórkowego dla $t_0 = 0$ w jednym z przypadków

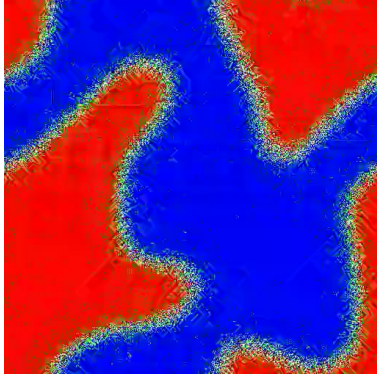


Rysunek 5.9.2: Stan automatu komórkowego dla $t_0 = 0$ po krótkim czasie ewolucji - pierwsza próbka

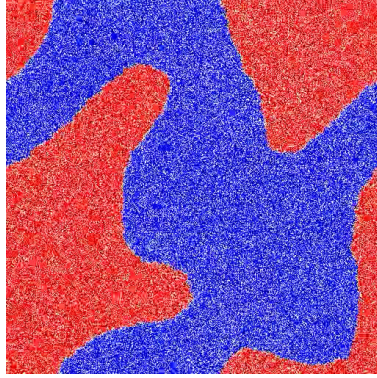


Rysunek 5.9.3: Stan automatu komórkowego dla $t_0 = 0$ po krótkim czasie ewolucji - druga próbka

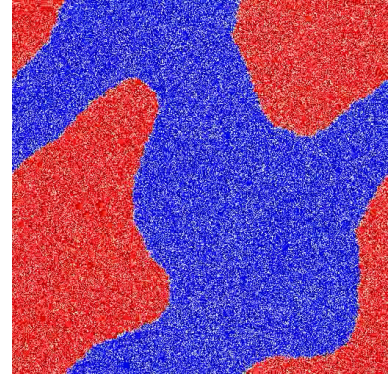
Symulacje w automacie komórkowym dla $t_0 = 3$ były już dużo bardziej stabilne i mniej wpływały na nie zaburzenia losowe. Ewolucja wszystkich automatów komórkowych z $t_0 = 3$ przebiegała podobnie. Poniżej możemy zaobserwować dane początkowe dla automatu komórkowego z $t_0 = 3$, następnie jego stan po kilkuset iteracjach o podobnym kształcie i stan po pewnym czasie ewolucji, w którym kształt frontu jest podobny, jednak bardziej płaski tam, gdzie wcześniej był wklęsły oraz bardziej zaokrąglony tam, gdzie wcześniej miał dużą krzywiznę. Zauważmy podobieństwo ewolucji porównując dwa ostatnie rysunki dla przypadku symulacji numerycznych powyżej oraz automatów komórkowych poniżej.



Rysunek 5.10.1: Stan początkowy automatu komórkowego dla $t_0 = 3$



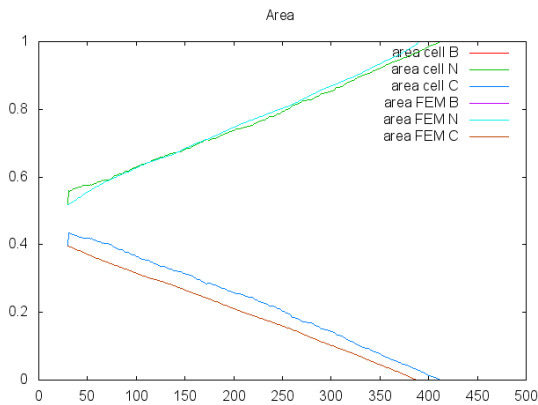
Rysunek 5.10.2: Stan początkowy automatu komórkowego dla $t_0 = 3$ po kilkuset iteracjach



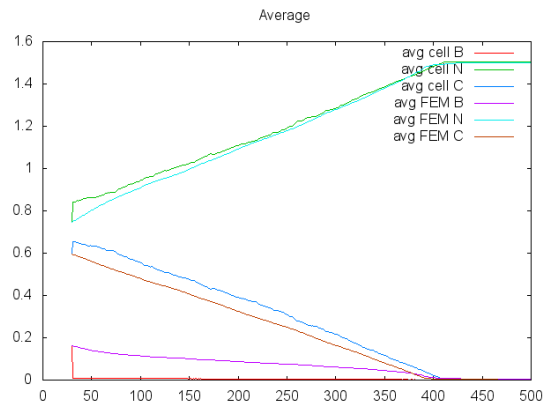
Rysunek 5.10.3: Stan automatu komórkowego dla $t_0 = 3$ po krótkim czasie ewolucji

5.5.4.4. Rozmiar siatki N , krok czasowy τ i szybkość względna Q

Wartości N , τ i Q mają bezpośredni wpływ na obserwowaną ilość kroków automatu komórkowego przypadających na krok czasowy symulacji numerycznej. Widać, że tak naprawdę istotna jest względna szybkość czasów proporcjonalna do $\frac{Q}{\tau}$ i skalowanie przestrzeni proporcjonalne do $\frac{N}{\sqrt{k}}$. Chcemy osiągnąć sytuację, w której szybkość rozchodzenia się informacji podzielona przez rozmiar przestrzeni jest taka sama. Poniżej znajdują się wykresy, na podstawie których można ocenić trafność doboru tych parametrów.



Rysunek 5.11.1: Wykres powierzchni dominacji cząsteczek dla $Q = 270$ i $p = 1$



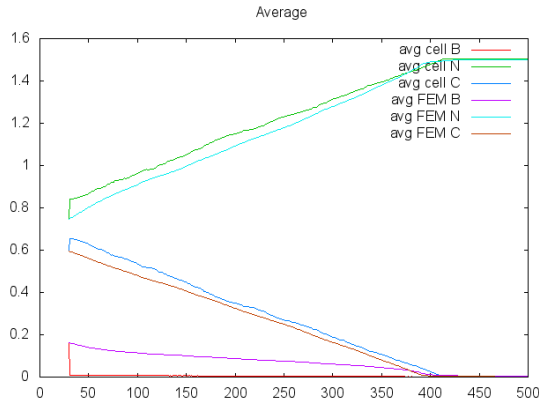
Rysunek 5.11.2: Wykres średnich gęstości cząsteczek dla $Q = 270$ i $p = 1$

Rozmiar siatki wykorzystywany w porównaniach był równy zawsze $N = 512$. Przy takim rozmiarze zaburzenia losowe odgrywały małą rolę na ewolucję układu (zaczynając od ustabiliz-

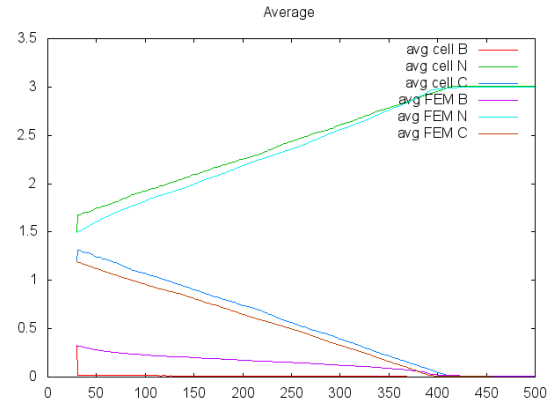
zowanych stanów). τ było równe $\tau = 0.01$ bądź $\tau = 0.02$. Wykonywane były eksperymenty dla różnych wartości Q - $Q = 200, 250, 270, 300$ dla $\tau = 0.01$ i $Q = 540$ dla $\tau = 0.02$. Zmiana Q rzeczywiście powodowała liniową zmianę szybkości działania układu, a najlepsze wyniki były osiągane dla $Q = 270$. Wartość Q można było najłatwiej dobrać obserwując kąty nachylenia wykresów pól powierzchni dominacji cząsteczek i ich średniej gęstości. Ewolucje automatów komórkowych dla różnych próbek były bardzo podobne - różne mikroskopijne zaburzenia losowe nie powodowały jakościowej zmiany wyników.

5.5.4.5. Ciśnienie p

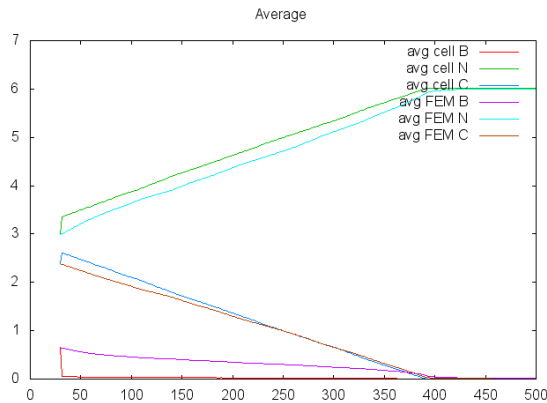
Można było zauważyć, że względna szybkość układów delikatnie zwiększała się wraz ze wzrostem p z $p = 2$ na $p = 4$, lecz nie różni się dla pozostałych przypadków. Może to mieć związek z tym, że do reakcji są potrzebne dokładnie 2 cząsteczki w jednej komórce. Dokładna analiza tego zjawiska wymagałaby dużo większej ilości próbek. Natomiast wyraźnie widać, że wykresy te są dużo gładkie, dla większych p , co jest zgodne z przewidywaniami. Większa ilość cząsteczek powoduje mniejszy wpływ zaburzeń dla pojedynczych cząsteczek. Poniżej znajdują się wykresy średnich gęstości cząsteczek, na których można zaobserwować te zjawiska.



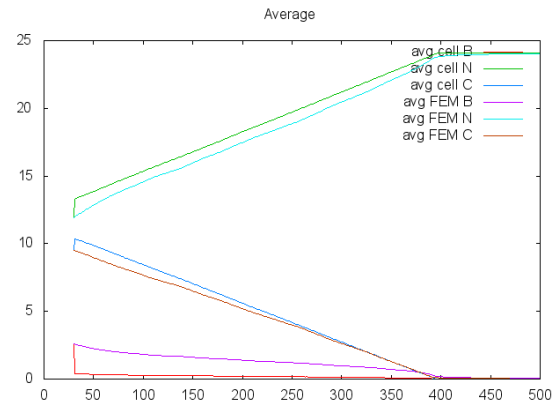
Rysunek 5.12.1: Wykres średnich gęstości cząsteczek dla losowych danych początkowych, $Q = 250$ i $p = 1$



Rysunek 5.12.2: Wykres średnich gęstości cząsteczek dla losowych danych początkowych, $Q = 250$ i $p = 2$



Rysunek 5.12.3: Wykres średnich gęstości cząsteczek dla losowych danych początkowych, $Q = 250$ i $p = 4$

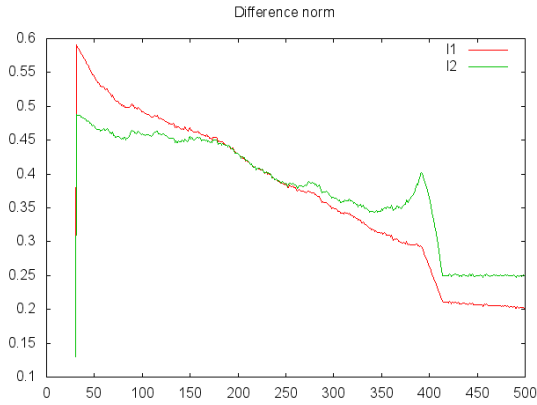


Rysunek 5.12.4: Wykres średnich gęstości cząsteczek dla losowych danych początkowych, $Q = 250$ i $p = 16$

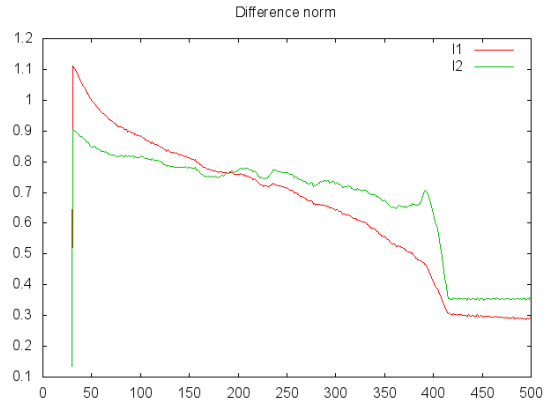
Zauważmy, że dla niskich ciśnień cząsteczki czerwone w automacie komórkowym nieznacz-

nie później dochodzą do wartości równej w przybliżeniu zero niż w przypadku symulacji numerycznych, co jest związane z wcześniejszą obserwacją o różnicy prędkości.

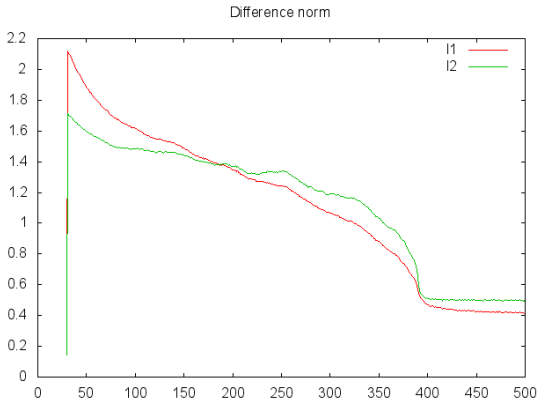
Wzrost ciśnienia powinien w naturalny sposób mieć wpływ na normę odległości, skalując ją przez wartość równą w przybliżeniu p . W rzeczywistości wraz ze wzrostem ciśnienia norma podzielona przez ciśnienie nieznacznie malała. Zachowanie to można przypisać szybkości reakcji na cząsteczkę odwrotnie proporcjonalnej do ciśnienia w przypadku rozpatrywanego przez nas modelu reakcji. Z kolei osłabienie reakcji względem dyfuzji przybliża automat komórkowy do symulacji numerycznych. Poniżej znajdują się wykresy norm l_1 i l_2 odległości rozwiązań w przestrzeni Ξ dla różnych p .



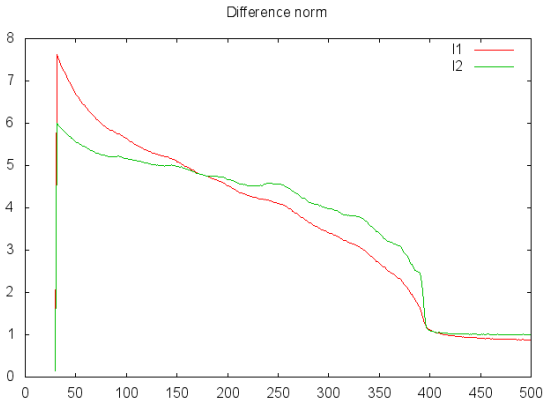
Rysunek 5.13.1: Wykres norm dla $Q = 250$ i $p = 1$ ($\sigma^2 = 1$ i $r = 4$)



Rysunek 5.13.2: Wykres norm dla $Q = 250$ i $p = 2$ ($\sigma^2 = 1$ i $r = 4$)

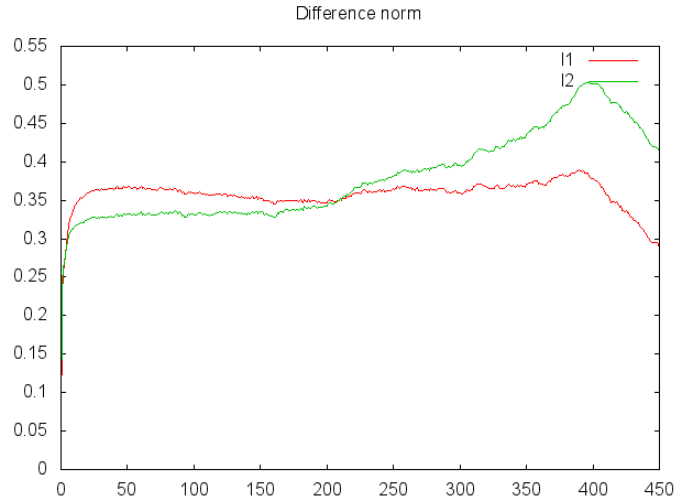


Rysunek 5.13.3: Wykres norm dla $Q = 250$ i $p = 4$ ($\sigma^2 = 1$ i $r = 4$)



Rysunek 5.13.4: Wykres norm dla $Q = 250$ i $p = 16$ ($\sigma^2 = 1$ i $r = 4$)

Zauważmy, że oprócz początkowego skoku, dla kroku czasowego równego około $i = 400$ i niskich ciśnień $p = 1, 2$ występuje kolejny skok w normie l_2 i szybszy spadek w normie l_1 . Gdy porównamy wykresy norm z wykresami średnich gęstości, zauważymy, że ma to miejsce w ostatnich momentach zanikania kuli. Szybszy spadek w normie l_1 można wytłumaczyć tym, że rozwiązanie symulacji numerycznej jest w tym czasie w przybliżeniu równe zero (i już prawie nie maleje), a dla automatu komórkowego jeszcze dąży do tej wartości. Zatem rozwiązania te zbliżają się do siebie. W doświadczeniu użyte były wartości $\sigma^2 = 1$ i $r = 4$. Zwiększenie tych wartości w przypadku $p = 1$ do $\sigma^2 = 2$ i $r = 16$ dało rezultaty jak na rysunku poniżej. Skok nadal występuje, ale funkcja jest nieco gładzsza.

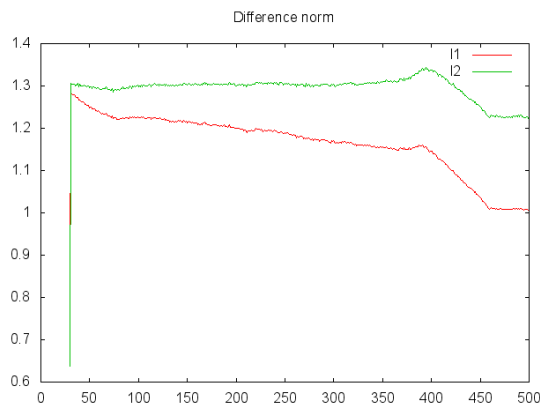


Rysunek 5.14: Wykres norm dla $Q = 250$ i $p = 1$ ($\sigma^2 = 2$ i $r = 16$)

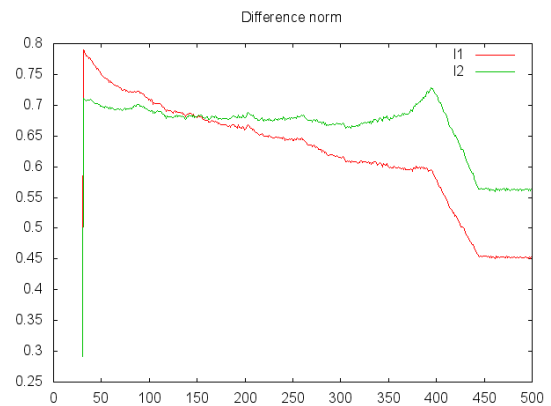
5.5.4.6. Wartości σ^2 i r

Wartości σ^2 i r opisane w sekcji 5.4.1 miały istotny wpływ na normę odległości. Odpowiedni dobór tych wartości powodował zmniejszenie normy odległości rozwiązań. Było to istotne w szczególności przy niskim ciśnieniu $p = 1$, ze względu na mniejszą regularność przestrzeni automatu komórkowego. Doświadczenia wykazały, że większe wartości r dają lepsze efekty (jednak znacznie zwiększając czas obliczeń). Ze względu na charakterystykę rozkładu normalnego istotne jest również, by dla większych wartości σ^2 dobrać większe wartości r . Doświadczenia pokazały, że zwiększanie σ^2 zmniejsza badane normy.

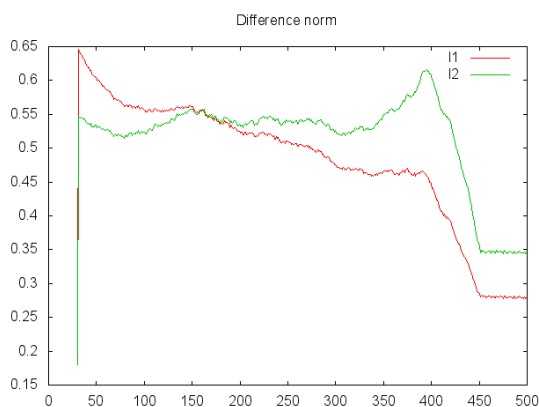
Największą zbadaną konfiguracją było $\sigma^2 = 4$ i $r = 16$. Ze względu na wykładniczy spadek gęstości rozkładu normalnego w zależności od odległości od średniej, wynikła norma była równie dobra, co w przypadku $\sigma^2 = 4$ i $r = 8$. Najmniejszą zbadaną konfiguracją było $r = 0$ (i dowolna wartość σ^2), czyli brak wygładzania. Poniżej znajdują się wykresy badanych norm dla losowych danych początkowych z $t_0 = 3$ omówionych w sekcji 5.5.4.3, różnych wartości σ^2 i r oraz różnych ciągów losowych w badanych automatach (dających jednak bardzo podobne wyniki).



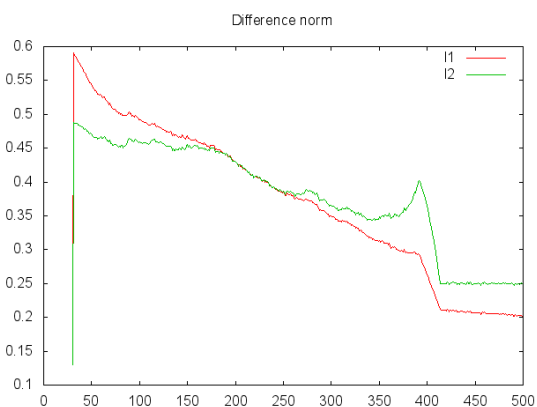
Rysunek 5.15.1: Wykres norm dla $\sigma^2 = 1$ i $r = 0$



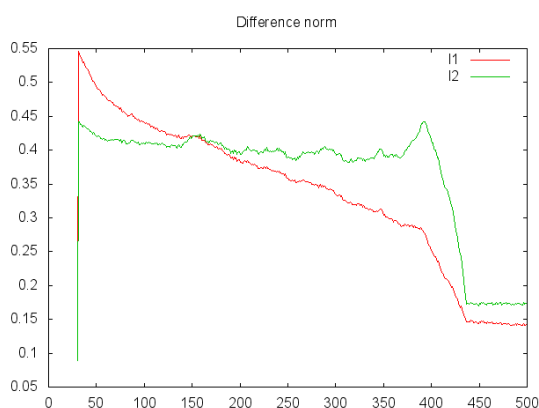
Rysunek 5.15.2: Wykres norm dla $\sigma^2 = 1$ i $r = 1$



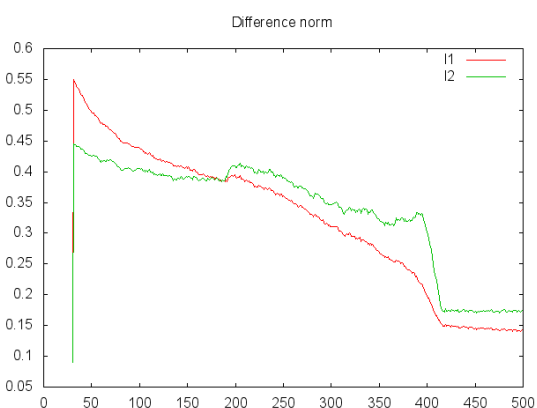
Rysunek 5.16.1: Wykres norm dla $\sigma^2 = 1$ i $r = 4$



Rysunek 5.16.2: Wykres norm dla $\sigma^2 = 2$ i $r = 4$



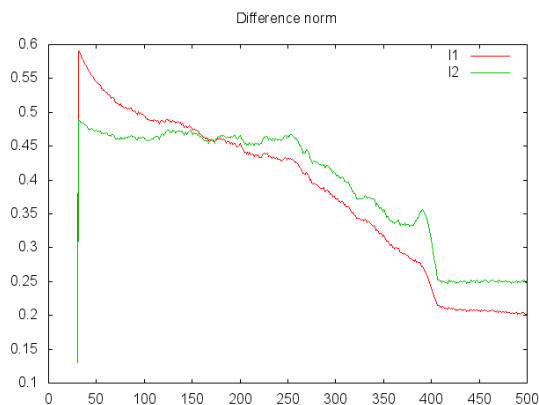
Rysunek 5.16.3: Wykres norm dla $\sigma^2 = 4$ i $r = 8$



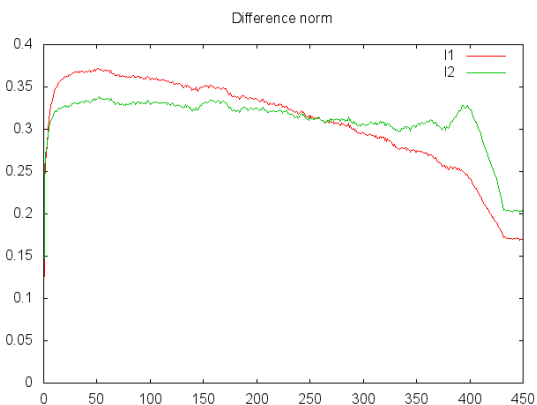
Rysunek 5.16.4: Wykres norm dla $\sigma^2 = 4$ i $r = 16$

5.5.4.7. Skok norm różnic rozwiązań dla małych czasów i dalszy spadek

Można zauważyć, że w chwili początkowej normy różnic są niewielkie, po czym następuje gwałtowny ich wzrost i następnie powolny spadek w miarę ewolucji układu. Może być to spowodowane niedoskonałościami w przybliżaniu warunków początkowych dla automatów komórkowych, które miały odpowiadać warunkom początkowym dla odpowiedniego t_0 z symulacji numerycznych.



Rysunek 5.17.1: Wykres normy różnicy dla losowych danych początkowych



Rysunek 5.17.2: Wykres normy różnicy dla danych początkowych w kształcie kuli

Na podanych wykresach możemy zaobserwować początkowy skok normy i dalszy jej spadek na przykładzie z danymi początkowymi losowymi ($t_0 = 3$) i w kształcie kuli. Możemy przypuszczać, że powodem zmniejszania się normy odległości jest zmniejszenie skomplikowania się struktury oraz często wyeliminowania wszystkich kolorów poza jednym w obu przypadkach. Mimo wszystko, możemy zauważyć, że normy różnic są niezerowe nawet w granicznym przypadku, kiedy w obu rodzajach symulacji przestrzeń zdominował jeden kolor. Oznacza to, że sposób porównania jest jeszcze daleki od doskonałego.

Oprócz początkowego skoku możemy zaobserwować kolejny (szczególnie w normie l_2) pod koniec spadku jej wartości. Skok ten został omówiony w sekcji 5.5.4.5 i zanika on dla dużych wartości ciśnienia p .

5.5.4.8. Wnioski

Przekształcanie rozwiązań numerycznych i automatami komórkowych do wspólnej przestrzeni Ξ i porównywanie ich za pomocą normy dostarcza dobrych, liczbowych informacji na temat bliskości rozwiązań i jakości porównania, jeżeli odpowiednie parametry przekształcenia zostały dobrze dobrane. Mimo wszystko, przedstawiony sposób przekształcania stanu automatu komórkowego do przestrzeni Ξ wymaga dopracowania.

Prawdopodobnie należy stworzyć sposób tworzenia danych początkowych dla automatu komórkowego lepiej odwzorowujący jego naturę. Na przykład, przy ciśnieniu $p = 1$ automat ten często posiada puste komórki, a przedstawiony odpowiednik numerycznych danych początkowych o sumie mas w punkcie równej 1 nie zawiera ich.

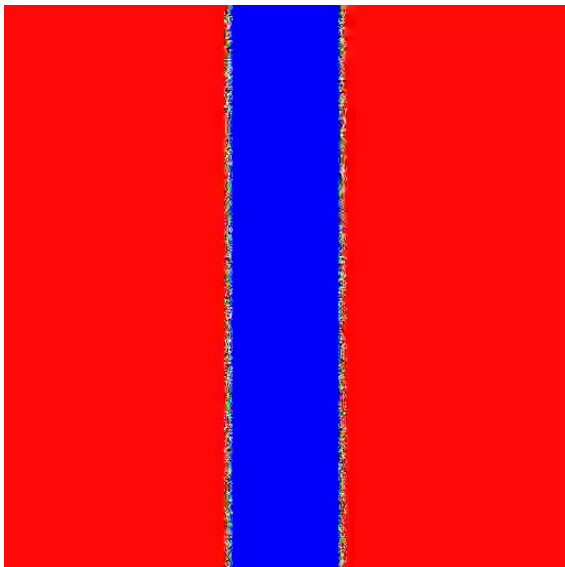
W wykonanych doświadczeniach parametry nie były kalibrowane w żaden usystematyzowany sposób. Należałoby wykonać wiele eksperymentów dla różnych kombinacji w celu próby osiągnięcia minimalnych wartości norm. Podobnym problemem jest graniczna norma odległości. Można zaobserwować, że jest ona stosunkowo duża mimo osiągnięcia stanu bliskiego stacjonarnemu (dominacji cząsteczki kolorowej) przez oba modele. W tym wypadku możliwe że ustalenie dużo większych parametrów σ^2 i r częściowo rozwiązałoby ten problem, ale nadal odpowiedni dobór pozostałych parametrów jest konieczny, by uzyskać normę możliwie bliską zeru przy jakościowo identycznych rozwiązaniach (dominacja przestrzeni przez jeden typ cząsteczek).

5.5.5. Przypadki odróżniające modele

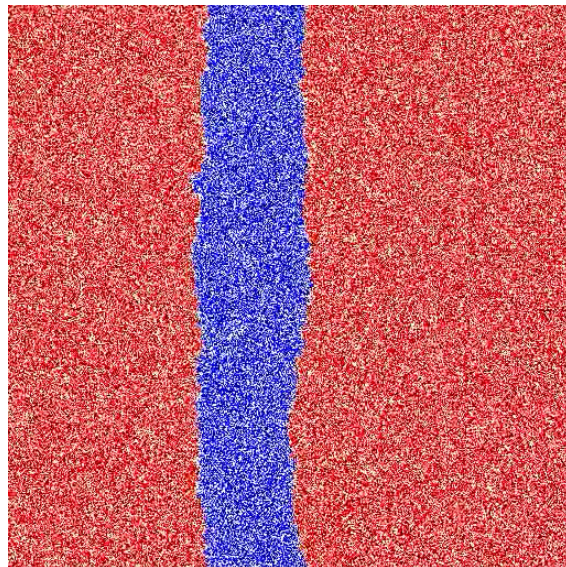
Jak wspomnieliśmy w sekcji 5.5.2, w automacie komórkowym dla losowych danych często tworzą się stabilne pasy. Można zaproponować więc następujące doświadczenie: weźmy dane początkowe w postaci wąskiego pasa i porównajmy wyniki symulacji numerycznych i automatu komórkowego.

Po przeprowadzeniu takiego doświadczenia okazało się, że w symulacjach numerycznych pas powoli zanikał, a w przypadku automatów komórkowych utrzymywał się. Wcześniej wysnuiliśmy hipotezę, że w przypadku automatów komórkowych zadanie w przypadku pasa jest podobne do ruiny dwóch graczy o skończonych kapitałach. Jednak w przypadku równań różniczkowych, ze względu na nieograniczony zasięg dyfuzji, można obserwować powolne znikanie pasa.

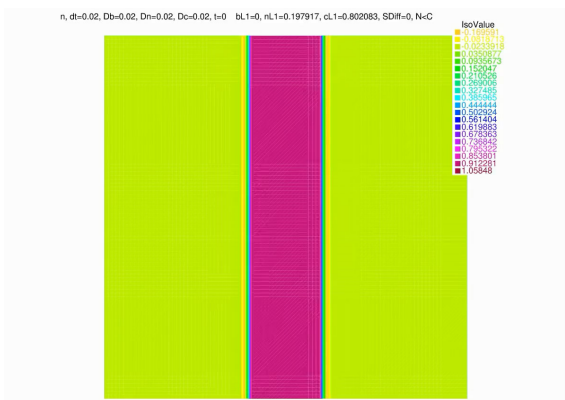
Można zaryzykować hipotezę, że symulacje numeryczne i automaty komórkowe nie są równoważne, niezależnie od doboru podanych wcześniej parametrów. Omawiane doświadczenie zostało wykonane korzystając z symulacji porównawczej. Przyjrzyjmy się wynikom obserwując rozwiązania na obu przestrzeniach oraz wykresy powierzchni dominacji i średnich gęstości cząsteczek.



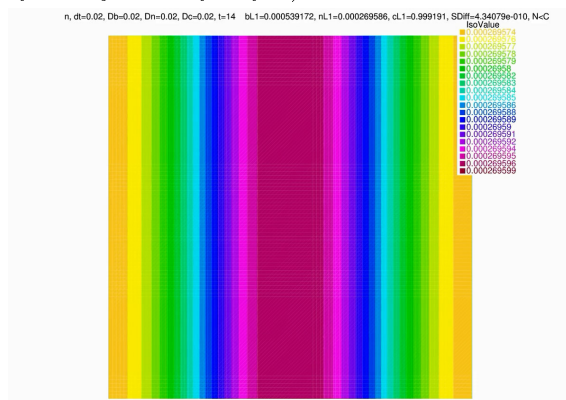
Rysunek 5.18.1: Dane początkowe automatu komórkowego



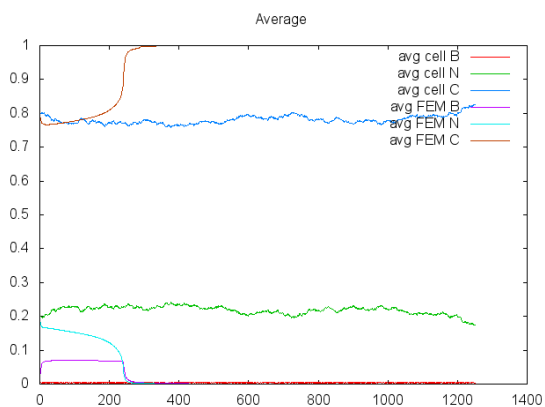
Rysunek 5.18.2: Utrzymujący się pas (na rysunku iteracja 675000 - odpowiadająca ok. $t = 25$ dla symulacji numerycznych)



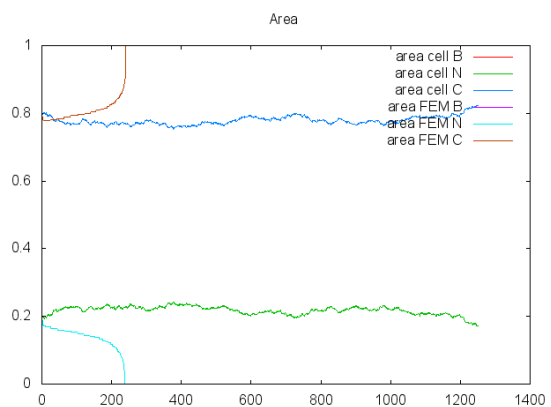
Rysunek 5.18.3: Dane początkowe symulacji numerycznej



Rysunek 5.18.4: Osiągnięcie stanu symulacji numerycznej bliskiego stacjonarnemu (dominacja cząsteczek czerwonych) w $t = 14$



Rysunek 5.18.5: Wykresy średnich gęstości cząsteczek



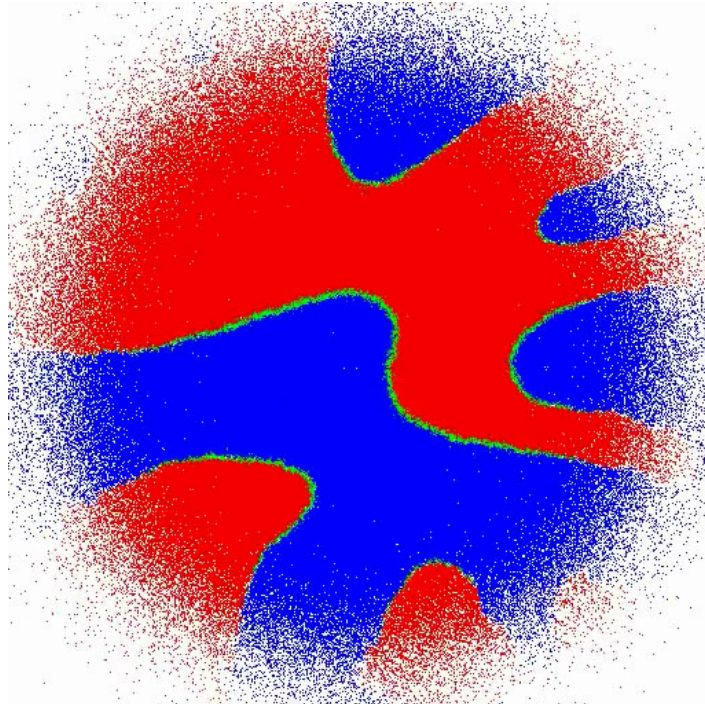
Rysunek 5.18.6: Wykresy powierzchni dominacji cząsteczek

5.5.6. Grubość frontu

Zostało wykonane doświadczenie dla automatu komórkowego, w którym danymi początkowymi była kula zawierająca po 8 cząsteczek koloru czerwonego lub niebieskiego (losowo). Po pewnym czasie ewolucji układu, w wyniku dyfuzji, gęstość cząsteczek stopniowo malała z odległością od środka kuli, a cząsteczki utworzyły wyraźny potok średniokrzywiznowy.

Można było zaobserwować w nim, że front (brzeg obszaru dominacji cząsteczek jednego koloru zawierający mieszaninę wielu kolorów) był najcieńszy dla obszaru o ciśnieniu ok. $p = 2$ (oceniając ciśnienie na podstawie gęstości komórek pustych), a grubszy w miarę zbliżania się do środka lub na zewnątrz.

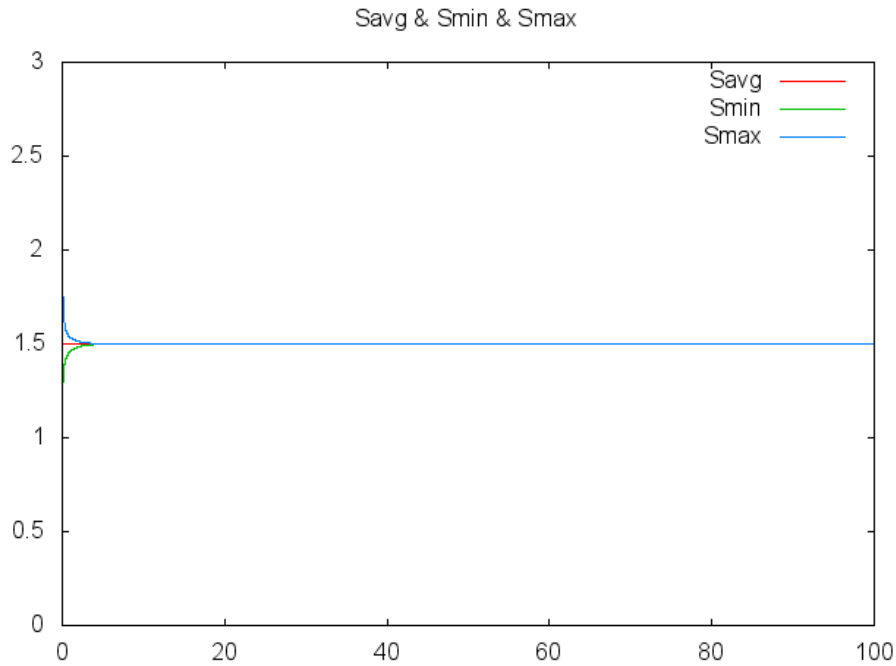
Zjawisko to wynika bezpośrednio z prostej analizy probabilistycznej dla poszczególnych cząsteczek. Dla dużego ciśnienia prawdopodobieństwo reakcji jest niższe, więc cząsteczka może pokonać większą odległość w morzu cząsteczek innego koloru. Z kolei dla małego ciśnienia prawdopodobieństwo napotkania cząsteczki do reakcji jest tym mniejsze, im mniejsze jest ciśnienie.



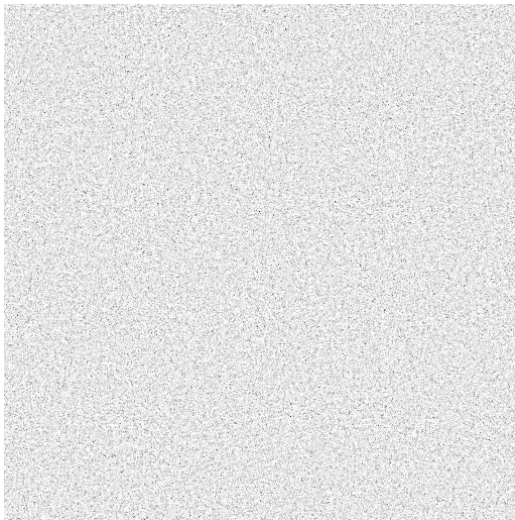
Rysunek 5.19: Zdyfundowana kula losowych cząsteczek w próżni dla $p = 8$ i po ok. 4000 iteracji

5.5.7. Rozkład masy

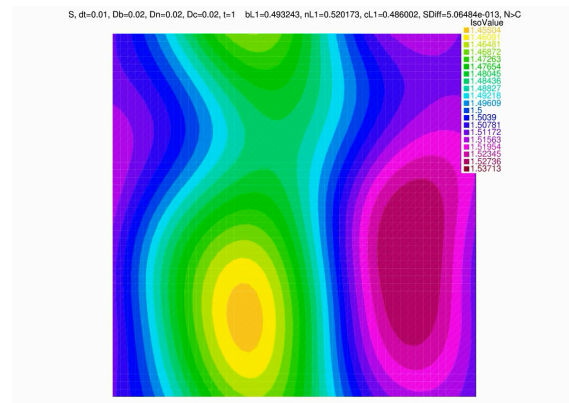
Jak wcześniej udowodniliśmy, równanie na rozkład masy jest równaniem ciepła w przypadku równań różniczkowych oraz symulacją samej dyfuzji w przypadku automatów komórkowych. Obserwacje potwierdziły własności teoretyczne w obu przypadkach. W przypadku symulacji numerycznych dodatkowo były zbierane dane na temat infimum, wartości średniej i supremum sumy gęstości cząsteczek w czasie, zatem możemy zaobserwować na wykresie szybkość zbieżność rozkładu masy do jednorodnego.



Rysunek 5.20: Wykres $\inf S$, \bar{S} i $\sup S$



Rysunek 5.21.1: Rozkład masy w automacie komórkowym



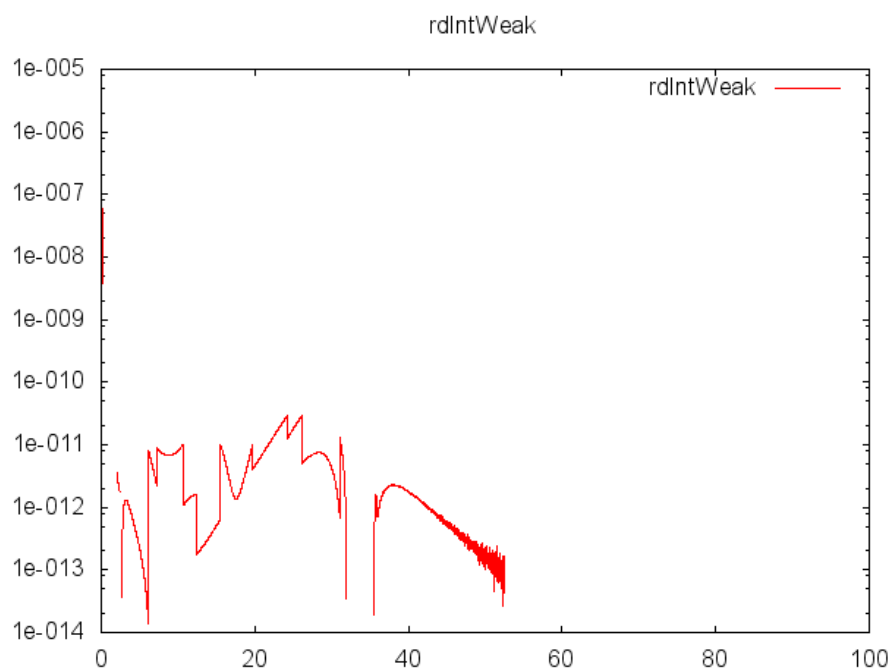
Rysunek 5.21.2: Rozkład masy bliski jednorodnemu dla $t = 1$

5.5.8. Dokładność obliczeń numerycznych

5.5.8.1. Sposoby pomiaru

Ponieważ nie posiadamy wzoru na rozwiązanie układu równań różniczkowych reakcji-dyfuzji dla modeli 2,5- i 3-cząsteczkowych, nie jesteśmy w stanie zbadać dokładności obliczeń porównując normy rozwiązania numerycznego i dokładnego. W celu sprawdzenia tej dokładności zbadaliśmy wcześniej teoretyczne właściwości układu równań różniczkowych i zbieraliśmy odpowiadające im dane. Sprawdzamy następujące własności:

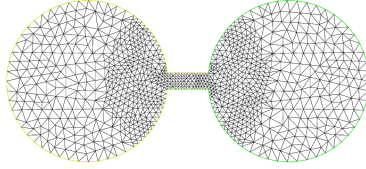
- Nieujemność rozwiązań - badając infimum funkcji b , n i c . Oczekujemy spełnienia tej własności w każdym przypadku i rzeczywiście jest ona spełniona;
- Stałość sumy masy - badamy wartość $S - \|S_0\|_{L^\infty(\Omega)}$. Powinna ona być możliwie bliska zeru. W wykonanych doświadczeniach wartości te były rzędu 10^{-10} , zatem można uznać tą właściwość za spełnioną. W procesie tworzenia symulacji numerycznych praktycznie każdy błąd został wykryty za pomocą obserwacji tej wartości;
- Rozkład masy - można było obserwować spełnienie zasady maksimum obserwując wykresy S_{min} , S_{avg} i S_{max} omówione powyżej w sekcji 5.5.7. Ponadto, obserwacja animacji rozkładu masy potwierdziła zachowanie się układu zgodnie z równaniem ciepła;
- Wykresy `rdIntWeak` i `rdIntWeakLog` - dostarczają one informacji na temat dokładności obliczeń w pojedynczym kroku czasowym. Spodziewamy się występowania w nim małych wartości. Rzeczywiście, wartości te oscylowały na poziomie 10^{-8} , przy czym w początkowej fazie przy losowych danych były dużo wyższe (czasem większe od jedności). W miarę malenia gradientów rozwiązania można było zaobserwować spadek tej wartości. Poniżej znajduje się szczytkowy wykres `rdIntWeakLog`. Zbyt małe wartości nie zostały na nim zaznaczone.



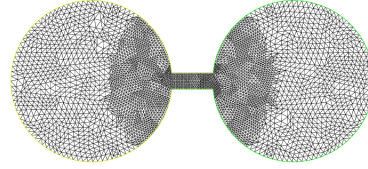
Rysunek 5.22: Wykres błędów lokalnych metody numerycznej

5.5.8.2. Porównanie metod predictor-corrector i Newtona

Powyższe własności posłużyły również do oceny dokładności aproksymacji zamkniętego schematu Eulera przez metodę predictor-corrector oraz metodę Newtona. Oprócz porównywania wyników między sobą porównywane były przyrosty dokładności w zależności od parametrów sterujących dokładnością. Dla każdej metody porównano z sobą dwie symulacje - jedna bazowa i druga dokładniejsza.

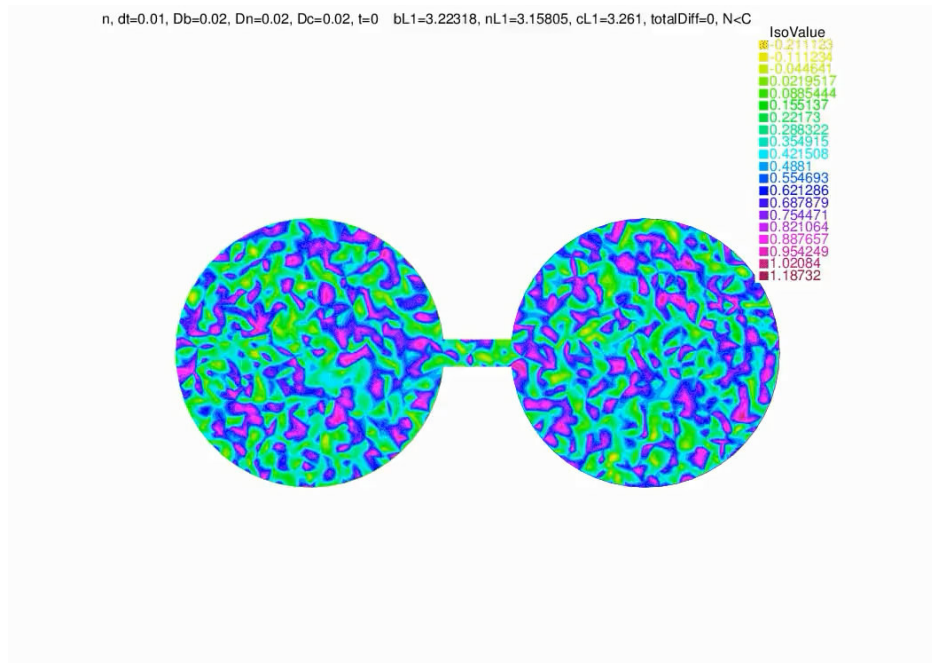


Rysunek 5.23.1: Siatka bazowa



Rysunek 5.23.2: Siatka z trójkątami o 2 razy mniejszym boku (4 razy więcej elementów)

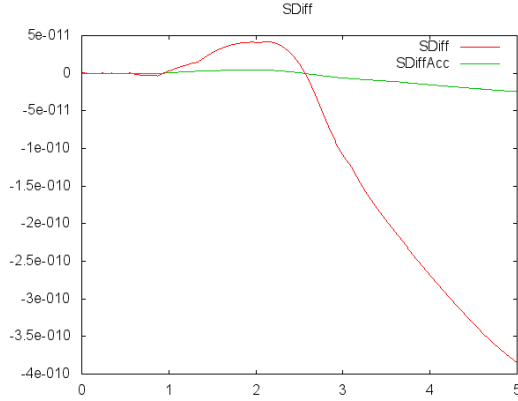
Dane początkowe w obu przypadkach były identyczne i losowo wygenerowane. Przestrzenią była przedstawiona wcześniej hantla. Siatki były użyte dwie, przy czym jedna z nich była pierwszą z trójkątami podzielonymi na trójkąty podobne o mniejszym boku. Wartości kroku czasowego były również mniejsze od bazowego $\tau = 0.01$. Przetestowane były następujące kombinacje mniejszego kroku czasowego/mniejszego boku trójkątów siatki: 9/3, 4/2. Dla samej metody Newtona ponadto 10/2 i 3/3. Ponadto w każdym przypadku zmienne sterujące dokładnością obliczeń ε_i opisane w sekcji 5.2.1.6 były 100 razy mniejsze. Przebieg symulacji obserwowaliśmy na przedziale o długości od 1 do 5 w zależności od symulacji (dobranego tak, aby czas trwania obliczeń był podobny), w którym to gradienty były duże.



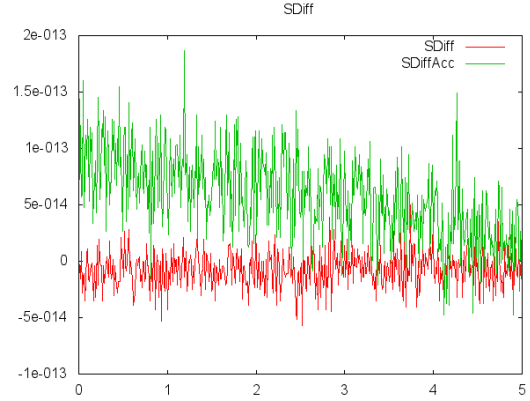
Rysunek 5.24: Losowe dane początkowe na hantli

W każdym przypadku wykresy danych i animacje były praktycznie takie same. Zatem nawet przy tak nieregularnych danych jak losowe metody dawały jakościowo identyczne rozwiązania. Można zauważyć, że w obu metodach wykres **rdIntWeak** posiadał wysoki skok dla

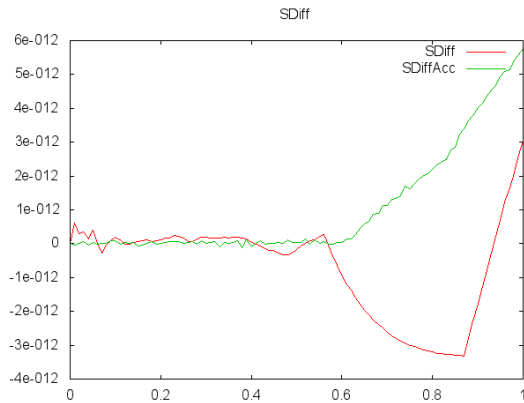
bardzo małych czasów w przypadku drobniejszej siatki. Porównajmy z sobą wykresy \mathbf{SDiff} różnic $S - \|S_0\|_{L^\infty(\Omega)}$, które możemy uznać za główny liczbowy wyznacznik dokładności metod.



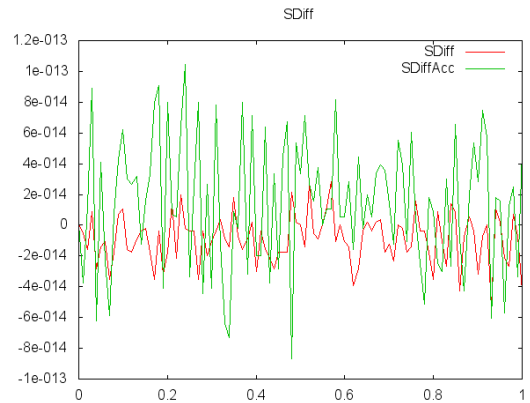
Rysunek 5.25.1: Wykres dla metody Newtona, wariantu podziału 4/2



Rysunek 5.25.2: Wykres dla metody predictor-corrector, wariantu podziału 4/2



Rysunek 5.25.3: Wykres dla metody Newtona, wariantu podziału 9/3

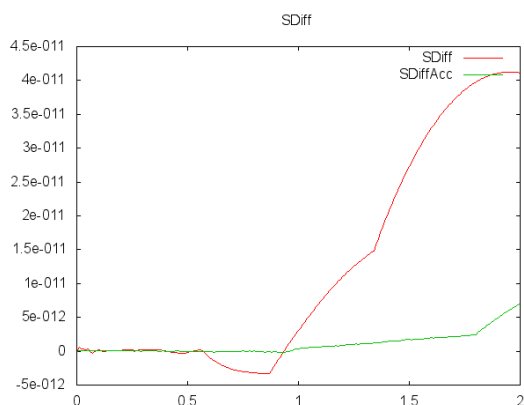


Rysunek 5.25.4: Wykres dla metody predictor-corrector, wariantu podziału 9/3

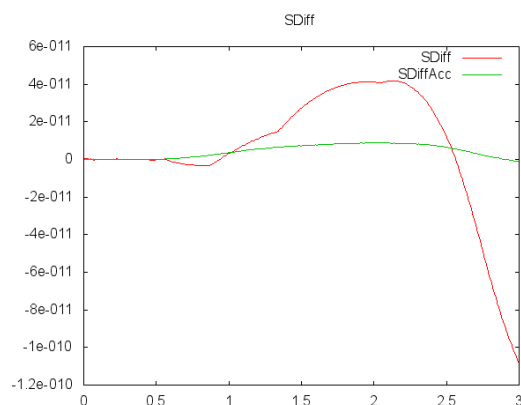
Widać, że błędy dla metody predictor-corrector są o rząd wielkości mniejsze, niż błędy metody Newtona. Jednak obserwujemy dziwne zjawisko, w którym błędy dla parametrów z większą teoretyczną dokładnością są większe w przypadku metody predictor-corrector. Z kolei w przypadku metody Newtona obserwujemy wzrost całkowitej masy od pewnego momentu. W innych doświadczeniach również metoda Newtona posiadała tendencję do zmiany całkowitej masy układu, lecz nawet dla długich czasów nie była to wartość duża.

Ostatecznie została wybrana metoda Newtona ze względu na jej lepsze teoretyczne właściwości, w szczególności dla większych czasów (czyli mniejszych gradientów i odległości rozwiązań od siebie w kolejnych krokach czasowych). Ponadto metoda Newtona posiadała mniejszą ilość wykonywanych iteracji (zwykle ok. 3, podczas gdy metoda predictor-corrector potrzebuje ok. 7 iteracji dla tych samych warunków stopu iteracji), w efekcie skutkując krótszym o 10%-20% czasem obliczeń mimo bardziej skomplikowanego algorytmu.

Przyjrzyjmy się jeszcze wykresom \mathbf{SDiff} dla pozostałych wariantów podziału dla metody Newtona.



Rysunek 5.26.1: Wykres dla metody Newtona, wariantu podziału 10/2



Rysunek 5.26.2: Wykres dla metody Newtona, wariantu podziału 3/3

Widać, że wariant 3/3 jest lepszy od 10/2. Tego typu wykresy mogą pomóc dobrać optymalne parametry wielkości siatki i długości kroku czasowego przy ustalonej ilości mocy obliczeniowej do wykorzystania.

Uwaga 3. *Chcąc wykonywać obliczenia na mało regularnych danych początkowych i o stosunkowo małej dokładności, dobrym pomysłem może być wykorzystanie metody predictor-corrector.*

5.5.9. Zmodyfikowana dyfuzja

Przeprowadzone zostały doświadczenia z automatem komórkowym 3-cząsteczkowym posiadającym zmodyfikowaną dyfuzję opisaną w sekcji 4.4.3. Mimo zmian, wciąż możemy zaobserwować potok średniokrzywiznowy i podobne jakościowe zachowanie (np. wytwarzanie się poziomych bądź pionowych pasów). Problemem tego rozwiązania jest omówiona w 4.4.3 zerowa wartość oczekiwana różnicy całkowitej ilości cząsteczek w komórkach nie posiadających pustych komórek w ich sąsiedztwie. Zostało to eksperymentalnie potwierdzone. Zjawisko to występuje już przy ciśnieniu $p = 1$, a dla $p = 8$ jest jeszcze lepiej widoczne.

Rozdział 6

Szczegóły techniczne

W tym rozdziale opiszemy środowisko wymagane do skompilowania i uruchomienia programów dla symulacji wymienionych w pracy. Wyniki wszystkich symulacji, kody źródłowe programów oraz inne materiały zostały umieszczone na dołączonej do pracy płycie DVD.

6.1. Wymagane oprogramowanie

Następujące oprogramowanie jest wykorzystywane w pracy:

- **FreeFem++** - oprogramowanie do symulacji numerycznych równań różniczkowych cząstkowych metodą elementu skończonego. Jest ono dostępne na stronie www.freefem.org/ff++;
- **ffmpeg** - oprogramowanie do konwersji filmów pomiędzy różnymi formatami, wykorzystywane w celu tworzenia animacji z pojedynczych klatek. Jest ono dostępne na stronie www.ffmpeg.org;
- **ImageMagick** - oprogramowanie do konwersji obrazków pomiędzy różnymi formatami. Jest ono dostępne na stronie www.imagemagick.org;
- **gnuplot** - oprogramowanie do tworzenia wykresów. Jest ono dostępne na stronie www.gnuplot.info;
- Uniksowe środowisko programistyczne zawierające m. in. programy **g++**, **make** i **bash**.

Wymienione oprogramowanie jest dostępne za darmo dla systemów operacyjnych Linux, OS X i Windows.

Symulacje były wykonywane na systemie operacyjnym Microsoft Windows 7 z zainstalowanym środowiskiem GNUstep (www.gnustep.org) zapewniającym uniksowe środowisko programistyczne. Niektóre symulacje były również testowane pod systemem operacyjnym PLD Linux.

6.2. Uruchamianie symulacji

Do uruchomienia symulacji, stworzenia odpowiednich animacji i wykresów zostały przygotowane skrypty napisane w bashu. Powodują one stworzenie w bieżącym folderu nowego folderu o nazwie według wzoru `<nazwa symulacji>-<losowy numer>` z wynikami. Dzięki losowości numeru, z bardzo dużym prawdopodobieństwem nie nadpiszemy poprzednich symulacji o tej samej nazwie. Skrypty są następujące:

- `automaton/make_animation.sh` - kompiluje program automatu komórkowego napisany w C++, uruchamia go i tworzy z klatek animację. Wymaga 1 lub 2 argumentów. Pierwszym jest nazwa symulacji, drugim (opcjonalnym) słowo `grain` dodające opcję do kodowania animacji dającą lepszą jakość filmu o wysokiej ziarnistości, przy większej objętości pliku;
- `numeric/make_animation.sh` - uruchamia program w FreeFem++, tworzy animacje zagęszczenia cząsteczek B , N , C , ich sumy S oraz odpowiednie wykresy. Wymaga 1 lub 2 argumentów. Pierwszym jest nazwa symulacji, drugim (opcjonalnym) słowo `saveAutomata` powodujące dodatkowo stworzenie podfolderu z danymi dla porównania (`make_comparison.sh`);
- `comparison/make_comparison.sh` - kompiluje program konwertujący dane z symulacji numerycznych na dane początkowe automatu komórkowego, symulującego go i porównującego z dalszymi danymi z symulacji numerycznych. Wymaga on podania 8 argumentów. Są to kolejno nazwa symulacji, folder z danymi z symulacji numerycznej, numer pierwszej iteracji (czyli takie $I \in \mathbb{N}$, że $t_0 = I\tau$), numer ostatniej iteracji (takie $I \in \mathbb{N}$, że $T = I\tau$), szybkość względna automatu komórkowego ($Q \in \mathbb{R}$), ciśnienie ($p \in \mathbb{N}$), wartość σ^2 i r . Wartości te zostały opisane wcześniej w sekcji 5.5.4. Folder z danymi symulacji numerycznej to folder `fem2cell` znajdujący się w katalogu wynikowym symulacji numerycznej uruchomionej z argumentem `saveAutomata`.

W trakcie tworzenia pracy powstawały ciągle nowe wersje skryptów, zatem część z nich uruchamia się z mniejszą liczbą opcji. Większość starszych skryptów wymaga modyfikacji zawartych w nich ścieżek do uruchomienia.

W nowych wersjach skryptu symulacje wykonują się zapisując dane w systemowym katalogu tymczasowym (tzn. utworzonym poleceniem `mktemp -d`), a po ukończeniu potrzebne pliki są przenoszone do katalogu wynikowego. Reszta plików (m. in. kolejne ramki animacji) są usuwane. Pliki tymczasowe w zależności od długości symulacji mogą zająć do kilku gigabajtów na każde tysiąc ramek wynikowej animacji. W przypadku przerwania obliczeń pliki tymczasowe nie są usuwane.

6.3. Komentarz do kodu symulacji numerycznych

Kod ten został napisany w języku programu FreeFem++, podobnym do C++. Dokładniejszy opis składni języka znajduje się na stronie programu, www.freefem.org/ff++.

Pierwszą częścią programu jest inicjalizacja wartości. Oprócz wymienionych w 5.2.1 stałych, są to także zmienne logiczne oznaczające istnienie argumentów `saveAutomata` i `savePlots`. `saveAutomata` powoduje zapisywanie danych wejściowych do symulacji porównawczych do katalogu `fem2cell`, a `savePlots` powoduje zapisywanie kolejnych klatek rozwiązania w formacie `eps`. Na potrzeby losowych danych początkowych inicjalizowany jest też na stałą generator liczb pseudolosowych funkcją `randinit`.

Dalej inicjalizowane są przestrzenie. Zmienne `Vh0` i `Vh` odpowiadają przestrzeniom elementu skończonego, a `Th` i `Th0` ich siatkom. Dane początkowe są tworzone w przestrzeni `Vh0`, a następnie aproksymowane w `Vh`. Przestrzeń `Vh0` zawiera siatkę z elementami o podobnej wielkości i służy do generowania losowych danych o równomiernym stopniu nieregularności. Przestrzeń `Vh` jest przestrzenią o drobniejszej (lub identycznej) siatce i służy do wykonywania na niej właściwych obliczeń. Początkowy fragment kodu zawiera szereg wykorzystywanych przestrzeni, z których tylko aktualnie używana nie jest zakomentowana.

Dalej odbywa się inicjalizacja danych początkowych zmiennych `b0`, `n0` i `c0` odpowiadających warunkowi początkowemu u_0 . Zmienne te są typu funkcyjnego $\Omega \mapsto \mathbb{R}$ (w przestrzeni elementu skończonego). Możliwe wartościowania są albo losowe albo ustalone takie jak kula.

W kolejnej części znajduje się inicjalizacja zmiennych potrzebnych do obliczeń numerycznych oraz definicja słabej postaci liniowego operatora różniczkowego odpowiadającego pojedynczej iteracji metody numerycznej. Następnie jest wykonywana podwójna pętla: na zewnątrz iterowana czasem, wewnątrz iteracjami metody Newtona. Wewnętrzne iteracje rozwiązywały metodą elementu skończonego zadany w postaci słabej problem i są przerywane w momencie, gdy błąd numeryczny jest odpowiednio mały bądź maksymalna ilość iteracji została przekroczona.

W zewnętrznej pętli są ponadto zbierane są ponadto dane do wykresów, zapisywane rozwiązania w postaci obrazka z poziomiami do plików `eps` (jeżeli włączono `savePlots`) i zapisywane dane do symulacji porównawczych (jeżeli włączono `saveAutomata`). Warto zauważyć, że przy zbieraniu danych zapisujemy obrazki do pliku o numerze $100000 + i$, gdzie i to numer iteracji. Dzięki temu pliki uporządkowane leksykograficznie zawierają kolejne ramki (zapobiega to problemowi, w którym po ramce `frame1.eps` pojawiałyby się `frame10.eps` zamiast `frame2.eps`).

W przypadku symulacji numerycznych porównujących dokładność metod numerycznych dla różnych podziałów siatki i różnego kroku czasowego, jednocześnie były wykonywane dwie symulacje. Kod składał się wtedy praktycznie z dwóch kopii oryginalnego kodu oraz zmodyfikowanej części zbierającej statystyki.

6.4. Komentarz do kodu automatów komórkowych

Kod symulacji automatu komórkowego został napisany w C++. Używa on wyłącznie standardowych bibliotek.

Na początku kodu źródłowego mamy definicję stałych określających automat komórkowy, opisanych w sekcji 5.3.1. Kod związany z automaty komórkowymi jest podzielony na funkcje. Przyjmowany przez nie argument `particles**` oznacza tablicę (indeksowaną kolorami) trzech tablic (indeksowaną pozycjami), czyli stan automatu komórkowego w danej chwili. Pozycją jest liczba całkowita równa $y * \text{size} + x$, gdzie x i y to współrzędne komórki, a `size` rozmiarem boku automatu. Opiszymy po kolei występujące w programie funkcje:

- `zero` - zeruje indeksowaną pozycjami tablicę;
- `wallNC` - ustawia `particles` na warunki początkowe z pasem niebieskim o grubości `width` ustawionym pod kątem 22.5° znajdującym się w morzu czerwonych cząsteczek, z `p` (`pressure` w kodzie) cząsteczkami na komórkę;
- `randomNC` - ustawia `particles` na dane losowe po `p` cząsteczek na komórkę losowego koloru czerwonego albo niebieskiego;
- `randomBallNCInVoid` - ustawia `particles` na losowe dane początkowe (jak w `randomNC`), nie zawierające jednak cząsteczek poza kulą o promieniu `r`;
- `ballNC` - ustawia `particles` na dane początkowe w postaci czerwonej kuli o promieniu `r` w morzu niebieskich cząsteczek;
- `diffusion` - wykonuje jedną fazę dyfuzji automatu na cząsteczkach jednego koloru - każda cząsteczka dyfunduje losowo do jednej z 5 komórek z otoczenia;

- `reaction3Particles`, `reaction2_5Particles`, `reaction2Particles` - wykonują fazę reakcji dla automatów kolejno 3-, 2.5- i 2-cząsteczkowego;
- `writeBNC` - zapisuje wizualizację dominacji cząsteczek w poszczególnych komórkach dla stanu automatu komórkowego `particles` do pliku `ppm` o nazwie `filename`;
- `writeMass` - zapisuje wizualizację rozkładu masy dla stanu automatu komórkowego `particles` do pliku w formacie `ppm` o nazwie `filename`;
- `main` - główna funkcja programu, którą omówimy poniżej.

W funkcji `main` znajduje się właściwy program. Na początku alokowana jest pamięć na stan automatu (`particles`) oraz tymczasową tablicę na stan automatu obcięty do cząsteczek jednego koloru (`tempParticles`). Następnie inicjalizujemy generator liczb pseudolosowych funkcją (`srand`) wartością zależną od czasu (zatem inną przy prawie każdym uruchomieniu programu). Kolejnym krokiem jest inicjalizacja warunku początkowego automatu za pomocą jednej z funkcji `wallNC`, `randomNC`, `randomBallNCInVoid` lub `ballNC`. W dalszej części mamy pętlę wizualizującą interesujące nas dane za pomocą `writeBNC` lub `writeMass`, a następnie wykonującą fazę dyfuzji dla każdego koloru cząsteczek i fazę reakcji. Na koniec zwalniana jest pamięć.

Jeżeli interesuje nas dominacja kolorów, czyli używamy funkcji wizualizującej `writeBNC`, to sprawdza ona jakie kolory cząsteczek jeszcze istnieją w automacie. Program zatrzymuje wtedy symulację albo w momencie upłynięcia zadanej ilości iteracji `iter` albo `endFrames` iteracji po zniknięciu cząsteczek czerwonych bądź niebieskich z automatu. Jeżeli używamy wizualizacji rozkładu masy `writeMass`, to program kontynuuje działanie aż do upłynięcia `iter` iteracji. Dzieje się to za pomocą zwrócenia wartości logicznej przez funkcje wizualizujące, która oznacza czy automat ma zakończyć symulację.

Aby zmienić model, wystarczy podmienić funkcję inicjalizującą warunek początkowy i funkcję reakcji.

6.5. Komentarz do kodu porównującego symulacje numeryczne i automaty komórkowe

Kod ten jest oparty na kodzie symulacji automatu komórkowego (również napisany w C++ i używający wyłącznie standardowych bibliotek).

Na początku kodu mamy określony rozmiar automatu komórkowego. W dalszej części napotykamy podział na funkcje, z których `zero`, `diffusion`, `reaction3Particles`, `writeBNC` i `writeMass` mają taką samą semantykę, jak w symulacji automatu komórkowego. Pozostałe funkcje to:

- `loadDataFromFile` - ładuje dane wejściowe z pliku `<prefix><100000+fileNumber>.dat` do tablicy tablic liczb zmiennoprzecinkowych dla każdego koloru odpowiadających rozkładowi kolorów cząsteczek w symulacji numerycznej;
- `initWithData` - przetwarza w podany w sekcji 5.4.3 sposób dane wejściowe otrzymane z `loadDataFromFile` na stan automatu komórkowego;
- `dataDiffusionDensity` - zwraca gęstość proporcjonalną do rozkładu normalnego obciętego do kuli $B(0, r)$ w punkcie (dx, dy) i wariancji σ^2 (`dataDiffusionVariance`);
- `gatherStatistics` - dla stanu automatu komórkowego `particles` i danych wejściowych z symulacji numerycznej `data` zwraca szereg statystyk omówionych w 5.4.2;

- **printUsage** - wypisuje pomoc w postaci oczekiwanych argumentów programu;
- **main** - główna funkcja programu, którą omówimy poniżej.

Funkcja **main** wygląda podobnie do głównej funkcji symulacji automatu komórkowego. Na początku interpretowane są argumenty jako liczby całkowite/zmiennoprzecinkowe. Następnie alokowana jest pamięć oraz inicjalizowany generator liczb pseudolosowych. Potem inicjalizowany jest automat komórkowy przez dane z symulacji numerycznej (z czasu t_0 , czyli iloczynu τ i drugiego argumentu programu, **minFrameNumber**). Następnie wykonywana jest pętla po krokach czasowych symulacji numerycznej od **minFrameNumber** do **maxFrameNumber**. Wewnątrz tej pętli na początku zbierane i zapisywane są statystyki, a następnie wykonywanych jest ok. **femSpeed** (Q) iteracji 3-cząsteczkowego automatu komórkowego, aż do osiągnięcia synchronizacji tych dwóch metod (wykonywanych jest dokładnie Q iteracji, gdy jest to liczba całkowita). Do wizualizacji danych mamy do wyboru funkcje **writeBNC** i **writeMass**, lecz zwracana przez nie wartość dotycząca warunku stopu dla automatu komórkowego jest ignorowana.

W przypadku posiadania danych z symulacji numerycznych dla modeli innych niż 3-cząsteczkowy, można skopiować odpowiednie reakcje z kodu symulacji automatów komórkowych i zadbać o wyzerowanie danych dla białych cząsteczek (bądź kompletnego ich usunięcia z wykresów).

6.6. Generator liczb pseudolosowych i determinizacja wyników

We wszystkich programach używamy generatora liczb pseudolosowych. Jego zasada działania jest następująca: inicjowany jest pewną wartością zwaną ziarnem, a następnie przy każdym wywołaniu funkcji **rand** bądź podobnej zwracającej pseudolosową liczbę zmienia się stan generatora liczb pseudolosowych na nowy, zależny wyłącznie od ziarna i historii wywołań funkcji pseudolosowych. Zatem inicjując ziarno na ustaloną liczbę całkowitą otrzymamy zawsze takie same wyniki przy kolejnych uruchomieniach programu. Chcąc uzyskać większy stopień losowości, zwykle inicjalizujemy ziarno na pewną funkcję zależną od czasu systemowego. Często jednak nie zależy nam na różnych wynikach, lecz na nieregularności danych. Przy tego typu badaniach warto ustawić ziarno na stałą.

Uwaga 4. *Implementacja generatora liczb pseudolosowych może zależeć od dostępnych bibliotek i architektury komputera, zatem dla ustalonego ziarna wyniki na różnych komputerach mogą być różne.*

6.7. Tworzenie animacji z pojedynczych klatek

Animacje były wykonywane z pojedynczych klatek z plików **eps** w przypadku symulacji numerycznych i **ppm** w przypadku automatów komórkowych. Ponadto, plik **eps** z rysunkiem siatki był konwertowany na plik **png**. Kod funkcji bashowych konwertujących pomiędzy formatami znajduje się w plikach **encoding.sh**. Zawierają one funkcje **ppm2gif**, **ppm2mp4**, **eps2gif**, **eps2mp4**, **gif2mp4** i **eps2png** i przyjmują 1 argument: albo prefiks plików do zamiany na animację, albo ścieżkę pliku źródłowego w przypadku **gif2mp4** i **eps2png**.

Do konwersji obrazków używany jest zestaw programów ImageMagick (konkretnie programy **convert** i **mogrify**). **ppm** (wersja P3) jest prostym formatem tekstowym do zapisu kolorów poszczególnych pikseli wykorzystywanym do wizualizacji automatów komórkowych. **eps** jest formatem grafiki wektorowej produkowanej przez wizualizację w **FreeFem++**. W celu zamiany na animację **mp4** oba te formaty są najpierw konwertowane na format **jpeg** obsługiwany przez

program **ffmpeg**. Następnie, program ten przetwarza podany ciąg obrazków na animację z 10 klatkami wejściowymi na sekundę za pomocą komendy

```
ffmpeg -y -r 10 -i folder/prefiks%d.jpg -c:v libx264 -preset veryslow  
-crf 30 animacja.mp4
```

Animacja jest tworzona używając kodeka H.264, który bardzo dobrze zachowuje jakość filmów przy dużej kompresji w stosunku do innych kodeków (np. MPEG-2). Początkowo animacje były tworzone w postaci plików **gif**, ale animacje posiadające około tysiąc klatek miały rozmiar równy 100-250MB w przypadku automatów komórkowych i 100-150MB w przypadku symulacji numerycznych. Oprócz dużego rozmiaru plików, problem związany z animacjami **gif** to inne przeznaczenie niż oglądanie filmów - popularne programy nie potrafiły np. przewijać filmu do odpowiedniego momentu albo nie działały poprawnie przez zbyt duży rozmiar plików.

Zmiany pomiędzy kolejnymi klatkami były stosunkowo małe oraz nie była potrzeba duża jakość, więc format **gif** został zastąpiony filmami kodowanymi H.264. Zwróćmy uwagę na parametry w powyższej komendzie. Są to kolejno **-y** powodujący ignorowanie zapytań potwierdzających akcje skierowanych do użytkownika. Drugim jest **-r 10** ustawiającym ilość wejściowych klatek na sekundę na 10. Następnym jest **-i folder/prefiks%d.jpg**, który powoduje, że wejściowymi obrazkami będą pliki o ścieżce **folder/prefiks0.jpg**, **folder/prefiks1.jpg** itd. Kolejnym argumentem jest **-c:v libx264** oznaczającym, że kodekiem wideo jest H.264. Parametr **-preset veryslow** ustawia ilość mocy obliczeniowej poświęconej na zwiększenie stosunku jakości do czasu kodowania na największy (poza **placebo**, który znacznie wydłuża czas kodowania praktycznie nie dając efektów). Kolejny parametr, **-crf 30**, ustawia jakość (skorelowaną z objętością pliku) na 30. Jakość 0 oznacza bezstratną, a 51 najgorszą. Wprawdzie do filmów zalecana jest jakość 18-28, ale w naszym przypadku taka jakość nie wnosi zbyt dużo. Eksperymenty wykazały, że obrazy z symulacji numerycznych (zawierające tekst) są dobrze widoczne przy jakości ok. 28-30, a automaty komórkowe już przy jakości ok. 42, dzięki wyraźnemu oddzielaniu się obszarów dominacji cząsteczek. Jednak w animacjach, w których to nie występuje (model 2-cząsteczkowy) albo istotne są pojedyncze piksele (rozkład masy) używamy jakości ok. 30 i dodatkowej opcji **-tune grain** wstawianej po **-crf 30**, która powoduje lepszą widoczność na bardzo ziarnistych filmach. Zabiegi te powodują to wzrost objętości pliku. Ostatecznie filmy dla automatów komórkowych zaczęły mieć rozmiar do ok. 80-120MB (a w wypadku niektórych filmów np. z dużym ciśnieniem, rozkładem masy do kilku MB). W przypadku symulacji numerycznych oszczędność miejsca była znacznie większa - filmy posiadały rozmiar 2.5-5MB.

6.8. Zawartość dołączonej płyty DVD

Najważniejszą i najobszerniejszą część dołączonej do pracy płyty DVD zajmują wyniki symulacji (przez ***** oznaczamy dowolny ciąg znaków w nazwie pliku):

1. **automaton/2.5particles-p*-*** - automat komórkowy 2,5-cząsteczkowy z losowymi danymi początkowymi. Obserwujemy dla dużych czasów raz pas i dwa razy zanikające obszary jednospójne;
2. **automaton/2.5particles-wall225-p1-*** - automat komórkowy 2,5-cząsteczkowy z danymi początkowymi w postaci pasa odchylonego od pionu 22.5° przeciwnie do ruchu wskazówek zegara. Obserwujemy tutaj w dwóch symulacjach przerwanie pasa w pewnym momencie i jego kurczenie się do zera;

3. `automaton/2particles-p*-*` - automat komórkowy 2-cząsteczkowy z losowymi danymi początkowymi. Nie obserwujemy powstawania żadnych struktur;
4. `automaton/3particles-ball-in-void-p*-*` - automat komórkowy 3-cząsteczkowy z losowymi danymi początkowymi w kształcie kuli w próżni. Obserwujemy rozpraszanie się kuli, potok średniokrzywiznowy i różnice grubości frontów;
5. `automaton/3particles-ball-p*-*` - automat komórkowy 3-cząsteczkowy z danymi początkowymi w kształcie kuli C w morzu N. Obserwujemy zmniejszanie się promienia kuli i jej zanik;
6. `automaton/3particles-massball-p1-*` - rozkład masy automatu 3-cząsteczkowego z danymi początkowymi w kształcie kuli. Obserwujemy rozpraszanie się kuli;
7. `automaton/3particles-ball-p*-*` - rozkład masy automatu 3-cząsteczkowego z losowymi danymi początkowymi. Obserwujemy dyfuzję;
8. `automaton/3particles-p*-*` - automat komórkowy 3-cząsteczkowy z losowymi danymi początkowymi. Obserwujemy dla dużych czasów sześć razy pas i raz zanikający obszar jednospójny;
9. `automaton/3particles-wall225-p*-*` - automat komórkowy 2,5-cząsteczkowy z danymi początkowymi w postaci pasa odchylonego od pionu 22.5° przeciwnie do wskazówek zegara z różnymi grubościami. Obserwujemy stabilizację pasa w dla długich czasów;
10. `automaton/3particles-sameInit-p1-*` - automaty komórkowe 3-cząsteczkowe z losowymi, ale identycznymi danymi początkowymi. Obserwujemy bardzo różne kierunki ewolucji - jeden prowadzący do pasa, a drugi do zanikania obszaru jednospójnego;
11. `automaton/diff1PerCell/3particles-p*-*` - automat komórkowy 3-cząsteczkowy z losowymi danymi początkowymi i zmodyfikowaną według 4.4.3 dyfuzją. Dla długich czasów obserwujemy powstawanie pasów, zanikanie obszarów jednospójnych i dwa razy małą część powolnej ewolucji. Obserwujemy również potok średniokrzywiznowy;
12. `automaton/diff1PerCell/3particles-mass-p*-*` - rozkład masy automatu komórkowego 3-cząsteczkowego z losowymi danymi początkowymi i zmodyfikowaną według 4.4.3 dyfuzją. Obserwujemy bardzo małą efektywność dyfuzji, szczególnie dla większego ciśnienia;
13. `numeric/ball-k200-*` - symulacja numeryczna z danymi początkowymi w postaci kuli C w morzu N o powierzchni większej niż połowa powierzchni przestrzeni, na torusie, $k = 200$;
14. `numeric/global_accuracy_newton/output-t*-s*-*` - symulacja numeryczna porównująca wyniki bazowe z krokiem czasowym t razy mniejszym i przestrzenią s razy mniejszą, z losowymi danymi początkowymi, na hantli. Opis wyników w 5.5.8.2;
15. `numeric/global_accuracy_predictor-corrector/output-t*-s*-*` - symulacja numeryczna porównująca wyniki bazowe z krokiem czasowym t razy mniejszym i przestrzenią s razy mniejszą, z losowymi danymi początkowymi, na hantli. Opis wyników w 5.5.8.2;
16. `numeric/newton_test/v2-based-*` - test zbieżności metody Newtona, gdzie dla 2^k -tych iteracji ($k \in \mathbb{N}$) było wykonywanych 1000 iteracji metody Newtona, na hantli.

Odpowiada to wykresom `time iter XXXXX`. Zaobserwowano szybką zbieżność i brak anomalii;

17. `numeric/old_predictor-corrector/P*-*` - starsze symulacje dla metody predictor-corrector z losowymi danymi początkowymi, na hantli. Wykonywane dla przestrzeni funkcji kawałkami wielomianowych pierwszego (P1) i drugiego (P2) stopnia i różnej ilości iteracji metody Newtona;
18. `numeric/predictor-corrector_test/v2-based_*` - test zbieżności metody predictor-corrector, gdzie dla 2^k -tych iteracji ($k \in \mathbb{N}$) było wykonywanych 1000 iteracji metody predictor-corrector, na hantli. Odpowiada to wykresom `time iter XXXXX`. Zaobserwowano trochę wolniejszą zbieżność niż w przypadku metody Newtona i brak anomalii;
19. `numeric/square-k*-*` - symulacje numeryczne z losowymi danymi początkowymi i podaną stałą k , na torusie. Zawsze dominowały cząsteczki jednego typu (we wszystkich pięciu przypadkach dominowała N, lecz dwie pary wyników są ze sobą powiązane danymi początkowymi);
20. `numeric/v*_newton_long` - symulacje numeryczne dla wersji 1 i 3 skryptu z metodą Newtona, z losowymi danymi początkowymi, na hantli. Obserwujemy podział hantli na dwie połówki z różnymi dominacjami oraz front pośrodku;
21. `numeric/v*_newton_no_front_long` - symulacje numeryczne dla wersji 1 i 3 skryptu, z losowymi danymi początkowymi, na hantli. Obserwujemy dominację cząsteczki czerwonej po pewnym czasie;
22. `numeric/v*_predictor-corrector_long` - symulacje numeryczne dla wersji 3 skryptu z metodą predictor-corrector, z losowymi danymi początkowymi, na hantli. Obserwujemy podział hantli na dwie połówki z różnymi dominacjami oraz front pośrodku;
23. `numeric/wall45-*` - symulacja numeryczna z danymi początkowymi w postaci pasów N i C pod kątem 45° i identycznej grubości, na torusie, $k = 200$. Obserwujemy utrzymywanie się tego stanu;
24. `numeric/wall-*` - symulacja numeryczna z danymi początkowymi w postaci cienkiego pionowego pasa N w morzu C, na torusie, $k = 200$. Obserwujemy szybki zanik pasa i dominację C;
25. `comparison/cmp-ball-*` - symulacje porównawcze symulacji numerycznych z automatem komórkowym z danymi początkowymi w postaci zanikającej kuli, wykorzystane do eksperymentów z potokiem średniokrzywiznowym w sekcji 5.5.3 oraz normą opisaną w sekcji 5.5.4. Kolejne liczby oznaczają argumenty liczbowe skryptu `make_comparison.sh` wymienione w 6.2. Dane do symulacji zostały pobrane z symulacji `numeric/ball-k200-12536`;
26. `comparison/cmp-rand-*` - symulacje porównawcze symulacji numerycznych z automatem komórkowym z losowymi danymi początkowymi, wykorzystane do eksperymentów z normą opisaną w sekcji 5.5.4. Kolejne liczby oznaczają argumenty liczbowe skryptu `make_comparison.sh` wymienione w 6.2. Dane do symulacji zostały pobrane z symulacji `numeric/square-k200-2286`;
27. `comparison/cmp-wall*-*` - symulacje porównawcze symulacji numerycznych z automatem komórkowym z danymi początkowymi w postaci pasa pionowego i obróconego o

45°, wykorzystane do eksperymentów z różnicą pomiędzy symulacjami numerycznymi i automatami komórkowymi, opisanych w sekcji 5.5.5. Kolejne liczby oznaczają argumenty liczbowe skryptu `make_comparison.sh` wymienione w 6.2. Dane do symulacji zostały pobrane z symulacji `numeric/wall-8908` i `numeric/wall45-19020`.

Przez losowe dane początkowe rozumiemy rozkład p cząsteczek na komórkę z każdą cząsteczką losowo typu N albo C. Wartość p jest podana w nazwie folderu. We wszystkich przykładach można zaobserwować potok średniokrzywiznowy dla cząsteczek N i C. Jeżeli nie podano inaczej dla symulacji numerycznych to $k = 1$. Macierz dyfuzji jest zawsze diagonalna o równych współczynnikach. W skryptach metoda predictor-corrector jest nazywana również „simple” (prosta). W niektórych starszych skryptach symulacji numerycznych ilość pomijanych ramek w czasie może mieć inną charakterystykę niż opisana w sekcji 5.2.1.7.

Poza wynikami symulacji na płycie znajdują się:

28. `automaton/automaton.cpp`, `automaton/encoding.sh`, `automaton/make_animation.sh`, `automaton/Makefile` - najnowsze wersje skryptów i programów do symulacji automatów komórkowych;
29. `numeric/encoding.sh`, `numeric/make_animation.sh`, `react-diff.edp`, `numeric/react-diff.plt` - najnowsze wersje skryptów do symulacji numerycznych;
30. `comparison/comparison.plt`, `comparison/encoding.sh`, `comparison/fem2cellComparison.cpp`, `comparison/make_comparison.sh`, `comparison/Makefile` - najnowsze wersje skryptów i programów do symulacji porównawczych;
31. `numeric/old_edp` - starsze wersje skryptów do symulacji numerycznych w programie FreeFem++;
32. `react-diff.pdf` - niniejsza praca.

Rozdział 7

Podsumowanie

W tej pracy zbadaliśmy dwa różne sposoby symulacji modelu reakcji-dyfuzji. Zostały wykonane eksperymenty pokazujące zarówno podobieństwa jak i różnice tych metod. W obu przypadkach mogliśmy zaobserwować potok średniokrzywiznowy oraz podobne ewolucje układu dla danych posiadających niezerową krzywiznę. Charakterystyczną cechą w obu przypadkach była tendencja do zanikania obszarów jednospójnych, na których dominowały cząsteczki jednego z kolorów (co jest zgodne z potokiem średniokrzywiznowym). Został skonstruowany przypadek odróżniający obie metody w postaci pasa na torusie o powierzchni mniejszej niż połowa przestrzeni. W przypadku symulacji numerycznych równań różniczkowych cząstkowych zaobserwowaliśmy zanik pasa (w wyniku działania dyfuzji), a w przypadku automatu komórkowego zaobserwowaliśmy ustabilizowanie się pasa podobnego do danych początkowych. Możemy na tym przypadku zaobserwować zasadniczą różnicę pomiędzy tymi modelami - zasięg oddziaływania dyfuzji. Wyłącznie lokalne oddziaływania w automacie komórkowym powodowały brak przesuwania się frontu reakcji, podczas gdy w przypadku równań różniczkowych cząstkowych front przesuwał się w kierunku substancji o mniejszej sumarycznej masie.

Przedstawiliśmy próbę sprowadzenia do wspólnej przestrzeni rozwiązań symulacji numerycznych i automatów komórkowych i tam porównania ich za pomocą norm. Takie podejście dostarcza cennych, liczbowych informacji na temat podobieństwa tych modeli i otwiera drogę do automatycznego doboru optymalnych parametrów. Jednak przedstawione podejście było dalece niedoskonałe (uzyskane normy różnic były stosunkowo duże) i wymaga znacznych ulepszeń.

W pracy przedstawiliśmy szereg własności teoretycznych równań różniczkowych cząstkowych reakcji-dyfuzji, które zostały później potwierdzone w eksperymentach, jednocześnie dostarczając informacji o ich poprawności i błędzie numerycznym. W przypadku automatów komórkowych pobieżnie zanalizowaliśmy kilka podstawowych własności (tworzenie się frontów, zadanie ruiny dla dwóch graczy o skończonych kapitałach pojawiające się dla szczególnych danych początkowych i inne), które również znalazły swoje potwierdzenie w eksperymentach. W przypadku symulacji numerycznych nie zaobserwowaliśmy żadnych artefaktów numerycznych, podczas gdy w automatach komórkowych widać było preferencję tworzenia się pionowych lub poziomych pasów dla niejednostajnych danych. Zaobserwowaliśmy również dużą wrażliwość automatów komórkowych na mało jednorodny rozkład danych początkowych, gdzie losowość reguł działających na mikroskopowym poziomie zaczynała odgrywać rolę tworząc odmienne ewolucje układu. W przypadku symulacji numerycznych zaobserwowaliśmy wzrost błędów dla losowych danych o dużych gradientach, ale jakościowe zachowanie układu pozostało takie samo.

Zarówno symulacje numeryczne i automaty komórkowe mają swoje wady i zalety. W przy-

padku symulacji numerycznych analiza teoretyczna była prostsza i byliśmy w stanie przewidzieć wiele własności układu. Przykładem jest na przykład zasada maksimum dla równania sumarycznego zagęszczenia cząsteczek - prosta do eksperymentalnego sprawdzenia w przypadku symulacji numerycznych, a trudna przy automatach komórkowych (już na etapie zbierania danych). Technicznie do wykonania dużo bardziej były skomplikowane symulacje numeryczne od automatów komórkowych. Czas obliczeń w obu przypadkach, by osiągnąć podobne rezultaty był podobny. Ilość opcji skalowania czasu obliczeń była jednak dużo większą w przypadku symulacji numerycznych (dobór kroku czasowego, rozmiaru siatki, błędów numerycznych, ilości wewnętrznych iteracji, całkowitego czasu) niż w przypadku automatu komórkowego (gdzie głównymi czynnikami wpływającym na długość obliczeń były jego rozmiar i całkowity czas symulacji (ilość iteracji)). W przypadku automatu komórkowego mieliśmy za to większą swobodę w modyfikowaniu własności modelu przy zachowaniu podobnych właściwości - można łatwo stworzyć i sprawdzić wiele modeli symulujących dyfuzję czy reakcję (przykłady znajdują się w sekcji 4.4), podczas gdy w równaniach różniczkowych możemy tylko operować na współczynnikach. Z drugiej strony w przypadku symulacji numerycznych mieliśmy dużo lepsze numeryczne odwzorowanie teoretycznego modelu matematycznego.

Przedstawiliśmy dużą ilość małych eksperymentów wykonywanych jednowątkowo w kilkakilkanaście godzin na procesorze o częstotliwości 2.9GHz na rdzeń. Ze względu na charakter obliczeń, możliwe jest masowe ich zrównoleglenie w każdym z przypadków na potrzeby większych eksperymentów. W używanym programie do obliczeń numerycznych **FreeFem++** jest możliwość zastosowania pakietów wykorzystujących bibliotekę MPI do pracy na klastrach komputerowych. W przypadku automatów komórkowych, dzięki lokalności oddziaływań, nie-trudne byłoby przepisanie kodu na taki, w którym przestrzeń dzielona jest na wiele równoległych obliczanych fragmentów. Można efektywnie zaimplementować te symulacje wykonując te obliczenia na kartach graficznych (w technologii OpenCL bądź CUDA) lub też wykorzystując bibliotekę MPI i klastry komputerowe.

Pozostawiliśmy wiele niedokończonych problemów zarówno teoretycznych, jak i eksperymentalnych w tej pracy. Większość zbadanych teoretycznie własności dotyczyła modelu 3-cząsteczkowego oraz przypadku, w którym macierz dyfuzji to $D = dI$, $d > 0$. Nie wykazaliśmy między innymi istnienia zbioru niezmienniczego dla pozostałych macierzy dyfuzji (a w związku z tym również oszacowań na asymptotyczne zachowanie się układu). Za główną metodę dyskretyzacji czasu wybraliśmy zamknięty schemat Eulera implementowany za pomocą metody Newtona. Więcej uwagi wymagałoby zbadanie metody predictor-corrector, która mogłaby dawać lepsze wyniki. Małym kosztem mógłby również znaleźć zastosowanie schemat Cranka-Nicolson. Z kolei w przypadku dyskretyzacji przestrzeni użyliśmy przestrzeni funkcji kawałkami wielomianowych drugiego stopnia. Uzasadnieniem jest większy rząd aproksymacji oraz naturalna właściwość wygładzania danych w czasie, ale być może inna przestrzeń dałaby równie dobre rezultaty mniejszym kosztem. Nie zostało również zbadane symulacjami numerycznymi zachowanie układu przy różnych współczynnikach dyfuzji (czy też dla bardziej dowolnej dodatnio określonej macierzy dyfuzji). Nie zostały wykonane również symulacje numeryczne dla modelu 2,5-cząsteczkowego. Dalszy rozwój mógłby również obejmować symulacje w trójwymiarowej przestrzeni, przy czym już prawdopodobnie byłoby potrzebne zrównoleglenie programów w sposób opisany w poprzednim akapicie.

Zaobserwowaliśmy już na przykładzie zmodyfikowanej dyfuzji, że ogólne własności układu (na przykład potok średniokrzywiznowy) pozostają podobne przy zmianach modelu automatu komórkowego. Zmiany te są w tym przypadku również proste do wykonania. Prawdopodobnie do osiągnięcia bliższych wyników symulacji numerycznych i automatów komórkowych potrzebna jest modyfikacja fazy reakcji w automatach komórkowych, w której wzrost gęstości substancji powoduje wzrost szybkości reakcji, a nie jej spadek. Poza tym, przydatne byłoby

dodanie kontroli nad szybkością reakcji. W obecnym modelu przy niskim zagęszczeniu cząsteczek szybkość reakcji jest bardzo duża względem szybkości dyfuzji - po przebyciu jednej jednostki odległości cząsteczka ma dużą szansę na wejście w reakcję. Ponadto, ze względu na większe znaczenie dyfuzji i grubszy, bardziej płynny front w automatach 2,5-cząsteczkowych, model ten mógłby lepiej nadawać się do osiągnięcia bliższych rozwiązań symulacji numerycznych i automatów komórkowych od modelu 3-cząsteczkowego. Istnieje zatem wiele ścieżek rozwoju dla dalszego porównania tych dwóch typów symulacji.

Bibliografia

- [1] K. N. Chueh, C. C. Conley, J. A. Smoller, Positively Invariant Regions for Systems of Nonlinear Diffusion Equations, *Indiana University Mathematics Journal*, nr 26 (2), strony 373-392, 1976.
- [2] Maksymilian Dryja, Janina i Michał Jankowscy, Przegląd metod i algorytmów numerycznych, część 2, Warszawa, WNT, 1982.
- [3] M. H. Engel, S. A. Macko, Isotopic evidence for extraterrestrial non-racemic amino acids in the Murchison meteorite, *Nature*, nr 389, strony 265-268, 1997.
- [4] Paul Houston, Chemical Kinetics and Reaction Dynamics, New York, McGraw-Hill, 2001.
- [5] Andrew Ilachinski, Cellular Automata, Singapur, World Scientific, 2002.
- [6] Claes Johnson, Numerical solutions of partial differential equations by the finite element method, Cambridge, 1987.
- [7] Paweł Konieczny, Związki między automatami komórkowymi a równaniami różniczkowymi cząstkowymi na przykładzie reakcji Biełousowa-Żabotyńskiego, Warszawa, praca magisterska na wydziale MIMUW, 2005.
- [8] O. A. Ladyzenskaja, V.A. Solonnikov, N. N. Ural'ceva, Lineinye i kvazilineinye uravneniya parabolicheskogo tipa, Moskwa, Nauka, 1967.
- [9] J. D. Murray, Wprowadzenie do biomatematyki, Warszawa, PWN, 2006.
- [10] Marek Orlik, O pochodzeniu „czystości chiralnej” biomolekuł, Warszawa, *Dwumiesięcznik Ministerstwa Edukacji Narodowej*, nr 4 (174), 1988.
- [11] Joel Smoller, Shock Waves and Reaction-Diffusion Equations, Springer-Verlag New York Heidelberg Berlin, 1983.
- [12] Magdalena Włodarczyk, Narzędzie do symulacji ewolucji układu wielu cząstek o różnych chiralnościach, Warszawa, praca magisterska na wydziale MIMUW, 2009.
- [13] Wikipedia, Elementary reaction, http://en.wikipedia.org/w/index.php?title=Elementary_reaction&oldid=544574684 (dostęp 5 maja 2013).
- [14] Wikipedia, Rate equation, http://en.wikipedia.org/w/index.php?title=Rate_equation&oldid=549880639 (dostęp 5 maja 2013).