

UNIwersytet ŚLĄSKI W KATOWICACH
Instytut Fizyki
Zakład Fizyki Teoretycznej

Andrzej Więckowski
Nr albumu: 292715

Numeryczne metody analizy dynamiki nanodrutów kwantowych

PRACA INŻYNIERSKA

Promotor:
prof. dr hab. Marcin Mierzejewski

Katowice 2016

Słowa kluczowe:

niskowymiarowe układy oddziaływujących fermionów, metody numeryczne

Oświadczenie autora pracy

Ja, niżej podpisany Andrzej Więckowski, autor pracy dyplomowej pt. *Numeryczne metody analizy dynamiki nanodrutów kwantowych*, o numerze albumu: 292715, student Wydziału Matematyki, Fizyki i Chemii Uniwersytetu Śląskiego w Katowicach, kierunku studiów Fizyka Techniczna (Międzywydziałowe Indywidualne Studia Matematyczno-Przyrodnicze), specjalności *Modelowanie komputerowe* oświadczam, że ww. praca dyplomowa:

- została przygotowana przeze mnie samodzielnie¹,
- nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (tekst jednolity Dz. U. z 2006 r. Nr 90, poz. 631, z późn. zm.) oraz dóbr osobistych chronionych prawem cywilnym,
- nie zawiera danych i informacji, które uzyskałem w sposób niedozwolony,
- nie była podstawą nadania dyplomu uczelni wyższej lub tytułu zawodowego ani mnie, ani innej osobie.

Oświadczam również, że treść pracy dyplomowej zamieszczonej przeze mnie w Archiwum Prac Dyplomowych jest identyczna z treścią zawartą w wydrukowanej wersji pracy.

Jestem świadomy odpowiedzialności karnej za złożenie fałszywego oświadczenia.

.....
(miejsce i data)

.....
(podpis autora pracy)

¹uwzględniając merytoryczny wkład promotora (w ramach prowadzonego seminarium dyplomowego)

*Pracę dedykuję moim Rodzicom Monice i Markowi oraz
Ukochanej Karolinie za ogromną motywację podczas
zagłębiania wiedzy z dziedzin fizyki.*

*Za ogromny poświęcony czas, liczne konsultacje i wy-
trwałość składam serdeczne podziękowania mojemu
Mentorowi – Prof. dr hab. Marcinowi Mierzejewskie-
mu.*

Streszczenie

Rozwój współczesnych komputerów umożliwił badanie złożonych układów kwantowych z oddziaływaniami za pomocą zaawansowanych metod numerycznych. W niniejszej pracy inżynierskiej przedstawiono wybrane metody numeryczne stosowane w badaniu dynamiki nanodrutów kwantowych. Omówiono algorytm Lanczosa, który jest idealnym narzędziem do generowania stanów kwantowych i algorytm propagacji Czebyszewa służący np. do propagacji czasowej wektora stanu w równaniu Schrödingera. W pracy zamieszczono zarówno dowody, wyprowadzenia czy też schematy wybranych omawianych zagadnień. Testowanie działania opisanych algorytmów dokonywano na modelu nanodrutu opisanym hamiltonianem modelu ciasnego wiązania.

Spis treści

1	Model teoretyczny	1
1.1	Formalizm drugiej kwantyzacji	1
1.1.1	Przestrzeń Foka	1
1.1.2	Zakaz Pauliego	2
1.1.3	Operatory kreacji i anihilacji	2
1.2	Model ciasnego wiązania	2
1.2.1	Założenia modelu	3
1.2.2	Hamiltonian	4
1.2.3	Rozwiązanie analityczne	5
1.2.4	Obserwabela prądu cząstek	6
2	Metody numeryczne	9
2.1	Algorytm Lanczosa	9
2.1.1	Przestrzeń Kryłowa	9
2.1.2	Metoda potęgowa	9
2.1.3	Generowanie bazy Lanczosa	10
2.1.4	Implementacja algorytmu	13
2.1.5	Generowanie stanu podstawowego	14
2.1.6	Generowanie stanów dla dowolnej temperatury	14
2.2	Ewolucja czasowa Czebyszewa	16
2.2.1	Wielomiany Czebyszewa	16
2.2.2	Ortogonalność wielomianów Czebyszewa	17
2.2.3	Rozwinięcie funkcji w szereg Czebyszewa	18
2.2.4	Propagacja czasowa wektora stanu	19

3 Wyniki	22
3.1 Parametry	22
3.1.1 Sprzęt komputerowy	22
3.1.2 Oprogramowanie	22
3.2 Ewolucja czasowa	23
3.2.1 Porównanie z metodą Eulera	23
3.2.2 Relaksacja prądu	25
 Podsumowanie	 27
 Spis rysunków	 28
 Bibliografia	 29

Rozdział 1

Model teoretyczny

1.1 Formalizm drugiej kwantyzacji

Dany podrozdział został napisany na podstawie prac znajdujących się w bibliografii na pozycjach [1, 2].

1.1.1 Przestrzeń Foka

W celu opisu układu co najwyżej N cząstek musimy stworzyć przestrzeń Hilberta \mathcal{H}_N będącą produktem tensorowym pojedynczych przestrzeni Hilberta \mathcal{H} :

$$\mathcal{H}_N = \bigotimes_{i=1}^N \mathcal{H}$$

Oczywiście można pokazać, że tak utworzona przestrzeń będzie posiadać typowe cechy dla przestrzeni Hilberta (zupełność, liniowość). Fermiony to cząstki antysymetryczne, a zatem na naszą przestrzeń należy jeszcze podziałać operatorem antysymetryzacji¹ \mathcal{P}_F tworząc przestrzeń Hilberta \mathcal{F}_N Fermionów:

$$\mathcal{F}_N = \mathcal{P}_F \mathcal{H}_N$$

Taka przestrzeń opisuje układ N fermionów. Jeśli chcemy mieć przestrzeń, która opíše dowolną liczbę cząstek, wystarczy wykonać sumę prostą przestrzeni Fermionowych \mathcal{F}_N :

$$\mathcal{F} = \bigoplus_{i=0}^{\infty} \mathcal{F}_i$$

¹ Operator \mathcal{P}_F definiujemy za pomocą działania na funkcję falową Ψ :
 $\mathcal{P}_F \Psi(\vec{r}_1, \dots, \vec{r}_N) = \sum_P (-1)^P \Psi(\vec{r}_{P_1}, \dots, \vec{r}_{P_N})$, gdzie (P_1, \dots, P_N) reprezentuje permutację P zbioru $(1, \dots, N)$.

1.1.2 Zakaz Pauliego

W przestrzeni \mathcal{F}_N fermionów, zasada Pauliego jest spełniona bezpośrednio z definicji przestrzeni. Rozważmy stan:

$$|\alpha_1 \dots \alpha_N\rangle = \frac{1}{N!} \sum_P (-1)^P \bigotimes_{i=1}^N |\alpha_{P_i}\rangle$$

gdzie (P_1, \dots, P_N) reprezentuje permutację P zbioru $(1, \dots, N)$. Załóżmy, że dwie cząstki znajdują się w tym samym stanie $|\alpha_1\rangle = |\alpha_2\rangle = |\alpha\rangle$. Wtedy:

$$|\alpha_1 \alpha_2 \dots \alpha_N\rangle = -|\alpha_2 \alpha_1 \dots \alpha_N\rangle = \frac{1}{2} (-|\alpha_2 \alpha_1 \dots \alpha_N\rangle + |\alpha_1 \alpha_2 \dots \alpha_N\rangle) = 0$$

1.1.3 Operatory kreacji i anihilacji

Operator kreacji a_λ^\dagger zdefiniujemy w działaniu na stan z przestrzeni \mathcal{F}_N :

$$a_\lambda^\dagger |\lambda_1 \dots \lambda_N\rangle = \begin{cases} |\lambda \lambda_1 \dots \lambda_N\rangle & \text{jeśli } |\lambda\rangle \text{ nie ma w } |\lambda_1 \dots \lambda_N\rangle \\ 0 & \text{jeśli } |\lambda\rangle \text{ jest w } |\lambda_1 \dots \lambda_N\rangle \end{cases}$$

Operator anihilacji a_λ zdefiniujemy jako sprzężenie hermitowskie operatora a_λ^\dagger .

Operatory kreacji (lub anihilacji) nie działają w obrębie jednej przestrzeni. Działając na stan $|\psi\rangle$ z przestrzeni \mathcal{F}_n tworzą stan w przestrzeni \mathcal{F}_{n+1} (lub \mathcal{F}_{n-1}).

Podstawowe związki antykomutacyjne dla fermionów są następujące:

$$\{a_\lambda, a_\mu\} = \{a_\lambda^\dagger, a_\mu^\dagger\} = 0 \quad (1.1)$$

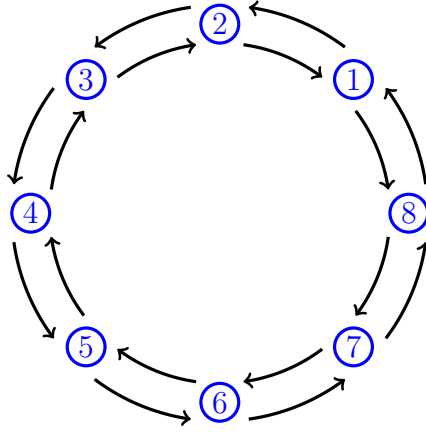
$$\{a_\lambda, a_\mu^\dagger\} = \delta_{\lambda\mu} \quad (1.2)$$

1.2 Model ciasnego wiązania

Dany podrozdział został napisany na podstawie prac znajdujących się w bibliografii na pozycjach [3–6].

1.2.1 Założenia modelu

W naszych rozważaniach przyjmiemy, że nasz układ to L węzłów (ponumerowanych $1, 2, \dots, L$), gdzie w każdym z nich może występować 1 bezspinowy fermion. Na tę jednowymiarową sieć narzucamy okresowe warunki brzegowe – po ostatnim węźle L następuje węzeł o numerze 1. Sieć wypełniamy N cząstkami. Poniżej zamieszczono rysunek wizualizujący badane zagadnienie dla $L = 8$.



Rysunek 1.1: Schemat jednowymiarowej sieci o $L = 8$ węzłach.

Kierunki zaznaczone na rysunku strzałkami związane są z przeskokami cząstek między węzłami.

Każdy węzeł sieci i posiada własną lokalną przestrzeń Hilberta \mathcal{H} , a zatem przestrzeń \mathcal{H}^L opisująca układ jest dana poprzez iloczyn tensorowy przestrzeni Hilberta:

$$\mathcal{H}^L = \bigotimes_{i=1}^L \mathcal{H}$$

W każdym węźle może znajdować się fermion ($|1\rangle$) lub nie ($|0\rangle$), a zatem wymiar naszej przestrzeni \mathcal{H}^L wynosi:

$$\dim \mathcal{H}^L = \dim(|n_1\rangle \otimes \dots \otimes |n_L\rangle) = \prod_{i=1}^L \dim |n_i\rangle = 2^L$$

Ustalając liczbę cząstek \hat{N} można przeprowadzić dekompozycję przestrzeni na podprzestrzenie \mathcal{H}_N^L z ustaloną liczbą cząstek N :

$$\mathcal{H}^L = \mathcal{H}_0^L \oplus \dots \oplus \mathcal{H}_N^L \oplus \dots \oplus \mathcal{H}_L^L$$

Każda taka podprzestrzeń \mathcal{H}_N^L będzie miała wymiar równy kombinacji $\binom{L}{N}$:

$$\dim \mathcal{H}^L = \dim \left(\bigoplus_{i=0}^L \mathcal{H}_N^L \right) = \sum_{i=0}^L \dim \mathcal{H}_N^L = \sum_{i=0}^L \binom{L}{i} = 2^L$$

otrzymaliśmy na końcu znaną tożsamość kombinatoryczną.

Podczas obliczeń wybierano podprzestrzeń z $N = L/2$ cząstkami. Ważne jest badanie układów o dużym L , ponieważ w granicy termodynamicznej ($L \rightarrow \infty$) układ jest pozbawiony artefaktów związanych ze skończonym rozmiarem układu:

$$\lim_{L \rightarrow \infty} \frac{\binom{L}{L/2}}{2^L} = 1$$

1.2.2 Hamiltonian

W tej pracy badany hamiltonian dla modelu ciasnego wiązania z uwzględnionymi oddziaływaniami ma następującą postać:

$$\hat{H} = \hat{H}_t + \hat{H}_V \quad (1.3)$$

$$\hat{H}_t = -t \sum_{i=1}^L \left(e^{i\varphi} a_{i+1}^\dagger a_i + \text{h.c.} \right) \quad (1.4)$$

$$\hat{H}_V = V \sum_{i=1}^L \tilde{n}_i \tilde{n}_{i+1} + W \sum_{i=1}^L \tilde{n}_i \tilde{n}_{i+2} \quad (1.5)$$

gdzie h.c. oznacza sprzężenie hermitowskie poprzedniego wyrazu.

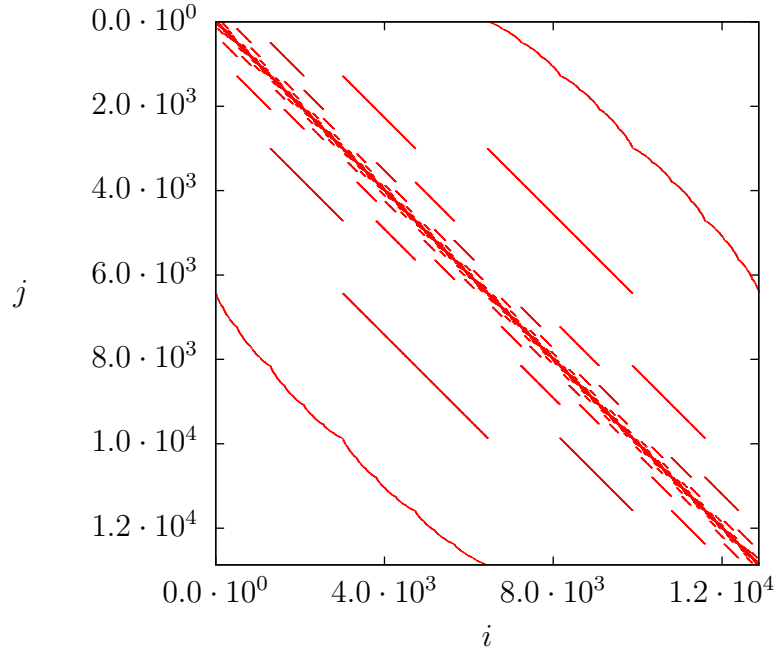
Część kinetyczna \hat{H}_t hamiltonianu stanowi sumę całek przeskoku cząstki między i -tym oraz $i + 1$ węzłem. Zakładamy, że całka przeskoku jest stała i wynosi t . Całka przeskoku t związana jest z energią tego przeskoku. Należy tutaj pamiętać o uwzględnionych okresowych warunkach brzegowych (PBC). Czynniki $e^{i\varphi}$ to fazy Peierlsa odpowiedzialna za przykładany strumień pola magnetycznego na układ.

Część potencjalną \hat{H}_V hamiltonianu zapisujemy w postaci oddziaływań ob-sadzonych sąsiednich węzłów. Operator \tilde{n}_i to operator liczby cząstek \hat{n}_i pomniejszony o wartość $\frac{1}{2}$:

$$\tilde{n}_i = \hat{n}_i - \frac{1}{2}$$

Operator liczby cząstek \hat{n}_i pomniejszono o $\frac{1}{2}$ w celu uzyskania symetrycznego widma hamiltonianu ze względu na transformację cząstka dziura. Uzasadnienie jest następujące: \hat{n}_i może przyjmować wartości ze zbioru $\{0, 1\}$, a po transformacji do \tilde{n}_i ze zbioru $\{-\frac{1}{2}, \frac{1}{2}\}$.

Dobłą własnością takiej macierzy, umożliwiającą łatwe dokonanie obliczeń numerycznych jest fakt, że ta macierz jest macierzą rzadką. Poniżej zamieszczono niezerowe wartości macierzy \hat{H} dla $L = 16$ węzłów z przestrzeni $N = L/2$ cząstek. Dla stosunkowo małej macierzy traktując ją jako macierz gęstą, wszystkich elementów jest $\sim 1.64 \cdot 10^8$, podczas gdy elementów różnych od 0 jest 122692, co stanowi około 0.07% wszystkich elementów.



Rysunek 1.2: Niezerowe elementy H_{ij} hamiltonianu dla modelu ciasnego wiązania: $L = 16$, $N = 8$.

1.2.3 Rozwiązanie analityczne

Dla zerowych potencjałów $V = W = 0$, dokonując transformacji operatorów, model przedstawiony w równaniach (1.3-1.5) można rozwiązać analitycznie. Niestety przy potencjałach różnych od zera $V \neq 0, W \neq 0$ model można rozwiązywać jedynie numerycznie za pomocą konstruowania pełnej macierzy hamiltonianu. Ten fakt związany jest z tym, że operatory typu $\hat{n}_i \hat{n}_j$ po transformowaniu do przestrzeni Fouriera tracą swoją diagonalną strukturę.

$$\hat{H} = -t \sum_i \left(e^{i\varphi} a_{i+1}^\dagger a_i + \text{h.c.} \right)$$

Poniżej przedstawiono operatory a_i, a_i^\dagger poddane dyskretnej transformacji Fouriera (DFT):

$$a_i = \frac{1}{\sqrt{L}} \sum_k e^{ikR_i} \tilde{a}_k$$

$$a_i^\dagger = \frac{1}{\sqrt{L}} \sum_k e^{-ikR_i} \tilde{a}_k^\dagger$$

gdzie $k = \frac{2\pi}{L}n$, a $n = 1, \dots, L$. Oczywiście prawdziwe są następujące własności $\tilde{a}_{k+2\pi} = \tilde{a}_k$, $\tilde{a}_\pi = \tilde{a}_{-\pi}$.

$$\begin{aligned} \hat{H}_t &= -t \sum_i \left(e^{i\varphi} \frac{1}{\sqrt{N}} \sum_k e^{-ik(R_i+1)} \tilde{a}_k^\dagger \frac{1}{\sqrt{N}} \sum_p e^{ipR_i} \tilde{a}_p + \text{h.c.} \right) = \\ &= -t \sum_k \sum_p \left(e^{i\varphi} \frac{1}{N} \underbrace{\sum_i e^{-i(k-p)R_i}}_{\delta_{kp}} e^{-ik} \tilde{a}_k^\dagger \tilde{a}_p + \text{h.c.} \right) = \\ &= -t \sum_k \sum_p \left(e^{i\varphi} \delta_{kp} e^{-ik} \tilde{a}_k^\dagger \tilde{a}_p + \text{h.c.} \right) = -t \sum_p \left(e^{i\varphi} e^{-ip} \tilde{a}_p^\dagger \tilde{a}_p + \text{h.c.} \right) = \\ &= -t \sum_p \left(e^{i(\varphi-p)} \tilde{a}_p^\dagger \tilde{a}_p + \text{h.c.} \right) = -t \sum_p \tilde{a}_p^\dagger \tilde{a}_p \left(e^{i(\varphi-p)} + e^{-i(\varphi-p)} \right) = \\ &= -2t \sum_p \cos(p - \varphi) \tilde{a}_p^\dagger \tilde{a}_p \end{aligned}$$

Rozwiązanie analityczne pozwoliło weryfikować poprawność znajdowanych wartości własnych Hamiltonianu podczas procesu pełnej diagonalizacji w fazie testowej przeprowadzonych obliczeń.

1.2.4 Obserwabla prądu cząstek

W badanych układach jedną z ważniejszych obserwabli jest prąd cząstek \hat{j} . Prąd można zdefiniować za pomocą wyrażenia $\hat{j} = -\frac{\partial \hat{H}}{\partial \varphi}$ lub też korzystając z równania ciągłości:

$$\dot{\rho} + \text{div} \vec{j} = 0$$

gdzie ρ to operator gęstości cząstek, \vec{j} to wektor prądu cząstek. Dla naszego jednowymiarowego układu równanie ciągłości możemy zapisać w następujący dyskretny sposób:

$$\frac{d}{dt} \rho_i + \hat{j}_{i+1} - \hat{j}_i = 0 \quad (1.6)$$

gdzie gęstość cząstek w węźle i wyrazimy poprzez $\rho_i = a_i^\dagger a_i$ i oczywiście $\hat{j} = \sum_{i=1}^L \hat{j}_i$. Pracując w obrazie Heisenberga ρ w funkcji czasu t przyjmie postać $\rho_i(t) = e^{i\hat{H}t/\hbar} a_i^\dagger a_i e^{-i\hat{H}t/\hbar}$. Korzystając z własności² $\frac{d}{dx} e^{\hat{A}x} = \hat{A} e^{\hat{A}x}$:

²można to pokazać poprzez rozwinięcie operatora w szereg Taylora

$$\begin{aligned} \frac{d}{dt}\rho &= \frac{d}{dt} \left(e^{i\hat{H}t/\hbar} a_i^\dagger a_i e^{-i\hat{H}t/\hbar} \right) = e^{i\hat{H}t/\hbar} i\hat{H} a_i^\dagger a_i e^{-i\hat{H}t/\hbar} - i e^{i\hat{H}t/\hbar} a_i^\dagger a_i \hat{H} e^{-i\hat{H}t/\hbar} \\ &= e^{i\hat{H}t/\hbar} i [\hat{H}, a_i^\dagger a_i] e^{-i\hat{H}t/\hbar} \end{aligned}$$

Wracając do równania (1.6) otrzymujemy:

$$\begin{aligned} e^{i\hat{H}t/\hbar} \underbrace{\left(i[\hat{H}, a_i^\dagger a_i] + \hat{j}_{i+1} - \hat{j}_i \right)}_0 e^{-i\hat{H}t/\hbar} &= 0 \\ i[\hat{H}, a_i^\dagger a_i] + \hat{j}_{i+1} - \hat{j}_i &= 0 \end{aligned} \quad (1.7)$$

Wyznaczmy teraz komutator $[H, a_i^\dagger a_i]$, pamiętając, że nasz hamiltonian \hat{H} jest postaci jak w równaniach (1.3-1.5)

$$\begin{aligned} [H, a_i^\dagger a_i] &= \left[-t \sum_{j=1}^L (e^{i\varphi} a_{j+1}^\dagger a_j + \text{h.c.}) + V \sum_{j=1}^L \tilde{n}_j \tilde{n}_{j+1} + W \sum_{j=1}^L \tilde{n}_j \tilde{n}_{j+2}, a_i^\dagger a_i \right] = \\ &= \left[-t \sum_{j=1}^L (e^{i\varphi} a_{j+1}^\dagger a_j + \text{h.c.}), a_i^\dagger a_i \right] = -t \sum_{j=1}^L (e^{i\varphi} [a_{j+1}^\dagger a_j, a_i^\dagger a_i] + \text{h.c.}) = \\ &= -t \sum_{j=1}^L (e^{i\varphi} (a_{j+1}^\dagger [a_j, a_i^\dagger a_i] + [a_{j+1}^\dagger, a_i^\dagger a_i] a_j) + \text{h.c.}) = \\ &= -t \sum_{j=1}^L \left(e^{i\varphi} (a_{j+1}^\dagger \underbrace{\{a_j, a_i^\dagger\}}_{\delta_{ij}} a_i - a_i^\dagger \underbrace{\{a_j, a_i\}}_0) + \right. \\ &\quad \left. + \left(\underbrace{\{a_{j+1}^\dagger, a_i^\dagger\}}_0 a_i - a_i^\dagger \underbrace{\{a_{j+1}^\dagger, a_i\}}_{\delta_{j+1,i}} \right) a_j \right) - \text{h.c.} \Big) = \\ &= -t (e^{i\varphi} (a_{i+1}^\dagger a_i - a_i^\dagger a_{i-1}) - \text{h.c.}) \end{aligned}$$

Wracając do równania (1.7):

$$\begin{aligned} i[\hat{H}, a_i^\dagger a_i] + j_{i+1} - j_i &= 0 \\ i \left(-t (e^{i\varphi} (a_{i+1}^\dagger a_i - a_i^\dagger a_{i-1}) - \text{h.c.}) \right) + \hat{j}_{i+1} - \hat{j}_i &= 0 \\ -t (i e^{i\varphi} (a_{i+1}^\dagger a_i - a_i^\dagger a_{i-1}) + \text{h.c.}) + \hat{j}_{i+1} - \hat{j}_i &= 0 \\ \underbrace{-t (i e^{i\varphi} a_{i+1}^\dagger a_i + \text{h.c.})}_{-\hat{j}_{i+1}} + \underbrace{t (i e^{i\varphi} a_i^\dagger a_{i-1} + \text{h.c.})}_{\hat{j}_i} + \hat{j}_{i+1} - \hat{j}_i &= 0 \end{aligned} \quad (1.8)$$

Następujące porównania oznaczone w poziomych klamrach z równania (1.8) mają sens ponieważ, jeśli zsumowalibyśmy po i równanie (1.8) otrzymalibyśmy zera po obu stronach równania:

Rozdział 2

Metody numeryczne

2.1 Algorytm Lanczosa

Dany podrozdział został napisany na podstawie prac znajdujących się w bibliografii na pozycjach: [7–11].

2.1.1 Przestrzeń Kryłowa

Przestrzeń Kryłowa \mathcal{K} to podprzestrzeń liniowa \mathbb{R}^n (lub ogólniej \mathbb{C}^n) generowana poprzez iteracyjne mnożenie wektora $h \in \mathbb{R}^n$ przez macierz kwadratową A :

$$\mathcal{K} = \text{lin}\{h, Ah, A^2h \dots, A^{n-1}h\}$$

Wiele współczesnych iteracyjnych metod rozwiązywania zagadnienia własnego macierzy opiera się na korzystaniu z przestrzeni (część z podprzestrzeni) Kryłowa.

2.1.2 Metoda potęgowa

Metoda potęgowa to trywialny algorytm znajdujący maksymalne wartości własne macierzy¹ A . Algorytm opiera się na iteracyjnym generowaniu wektorów z przestrzeni Kryłowa, z początkowego losowego wektora x_0 . Rekurencyjne generowanie wektorów przebiega następująco:

$$x_{n+1} = \frac{Ax_n}{\|x_n\|} \tag{2.1}$$

gdzie $\|\cdot\|$ oznacza normę wektora. Dla $n \rightarrow \infty$ norma wygenerowanego wektora x_n będzie równa wartości własnej λ_1 .

¹Jeśli jej wartości własne λ_i spełniają warunek: $|\lambda_1| > |\lambda_2| \geq \dots$

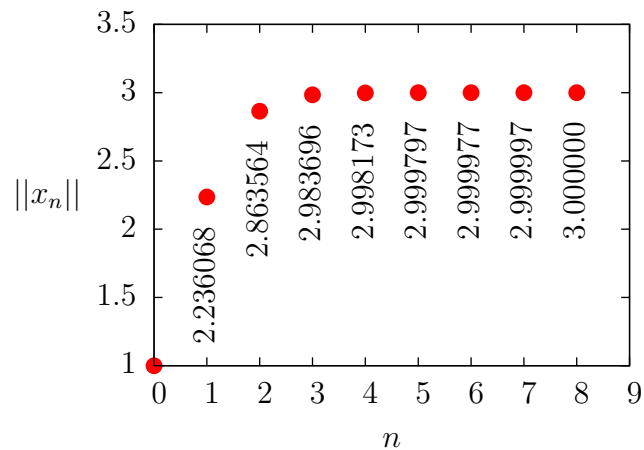
Dokonując transformacji macierzy A macierzą unitarną U można w skrócony sposób wyjaśnić działanie algorytmu:

$$A = U\tilde{A}U^{-1} = U\text{diag}(\lambda_i)U^{-1}$$

gdzie $\text{diag}(\lambda_i)$ to diagonalna macierz wartości własnych. Transformując do takiej postaci łatwo można pokazać, że potęgowanie macierzy sprowadza się do potęgowania elementów macierzy $\text{diag}(\lambda_i)$:

$$A^n = U\text{diag}(\lambda_i)\underbrace{U^{-1}U}_1\text{diag}(\lambda_i)U^{-1}\dots U\text{diag}(\lambda_i)U^{-1} = U\text{diag}(\lambda_i^n)U^{-1}$$

Macierz, która posiada wartość własną największą, dla $n \rightarrow \infty$ będzie podczas tego algorytmu dominować inne wartości własne. Jako przykład zaprezentowano zbieżność wartości własnej macierzy $\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$, która posiada wartości własne: $\lambda_1 = 3$, $\lambda_2 = 1$.



Rysunek 2.1: Zbieżność normy $\|x_n\|$ z równania (2.1) dla losowego unormowanego wektora x_0 .

2.1.3 Generowanie bazy Lanczosa

W metodzie Lanczosa generowana jest ortogonalna baza $\{|f_i\rangle\}_{i=0}^{k-1}$, będąca kombinacją liniową wektorów przestrzeni Kryłowa, w taki sposób, aby przetransformowana macierz H miała postać trójdiagonalną macierzy T_{kk} (stopień H : n , $H^\dagger = H$). Wektor początkowy $|f_0\rangle$ jest losowym² unormowanym wektorem z przestrzeni \mathbb{C}^n . Konstrukcja kolejnego stanu $|f_1\rangle$ odbywa się w następujący sposób:

$$|f_1\rangle = H|f_0\rangle - a_0|f_0\rangle \quad (2.2)$$

gdzie a_0 jest współczynnikiem na który narzucono warunek ortogonalności do poprzedniego stanu: $\langle f_0|f_1\rangle = 0$.

$$\langle f_0|f_1\rangle = \langle f_0|H|f_0\rangle - a_0\langle f_0|f_0\rangle = 0$$

$$a_0 = \frac{\langle f_0|H|f_0\rangle}{\langle f_0|f_0\rangle}$$

²Wymagane jest, aby nie był ortogonalny do stanu podstawowego $|\psi\rangle$: $\langle\psi|f_0\rangle \neq 0$

Kolejny stan $|f_2\rangle$ generowany jest poprzez uwzględnienie jeszcze jednego parametru b_0 , który zapewni nam trójdziagonalną postać:

$$|f_2\rangle = H|f_1\rangle - a_1|f_1\rangle - b_0|f_0\rangle \quad (2.3)$$

Wartość parametru a_1, b_0 znowu wyznaczymy narzucając warunki ortogonalności $\langle f_0|f_2\rangle = \langle f_1|f_2\rangle = 0$, pamiętając oczywiście o wcześniejszym warunku $\langle f_1|f_0\rangle = 0$. Równanie (2.3) mnożymy z lewej strony przez $\langle f_1|$:

$$\langle f_1|f_2\rangle = \langle f_1|H|f_1\rangle - a_1\underbrace{\langle f_1|f_1\rangle}_0 - b_0\underbrace{\langle f_1|f_0\rangle}_0 = 0$$

$$a_1 = \frac{\langle f_1|H|f_1\rangle}{\langle f_1|f_1\rangle}$$

Do wyznaczenia b_0 mnożymy równanie (2.3) z lewej strony przez $\langle f_0|$:

$$\langle f_0|f_2\rangle = \underbrace{\langle f_0|H|f_1\rangle}_X - a_1\underbrace{\langle f_0|f_1\rangle}_0 - b_0\underbrace{\langle f_0|f_0\rangle}_0 = 0$$

W celu wyznaczenia X sprzęgniemy po hermitowsku równanie (2.2):

$$\begin{aligned} \langle f_1| &= \langle f_0|H - a_0^*\langle f_0| \\ \langle f_0|H &= \langle f_1| + a_0^*\langle f_0| = X \end{aligned}$$

$$(\langle f_1| + a_0^*\langle f_0|)|f_1\rangle - b_0\underbrace{\langle f_0|f_0\rangle}_0 = 0$$

Ostatecznie otrzymujemy postać b_0 :

$$b_0 = \frac{\langle f_1|f_1\rangle}{\langle f_0|f_0\rangle}$$

Podobną procedurę przeprowadzamy dla ogólnego stanu $m + 1$:

$$\boxed{|f_{m+1}\rangle = H|f_m\rangle - a_m|f_m\rangle - b_{m-1}|f_{m-1}\rangle} \quad (2.4)$$

$$\langle f_m|f_{m+1}\rangle = \langle f_m|H|f_m\rangle - a_m\underbrace{\langle f_m|f_m\rangle}_0 - b_{m-1}\underbrace{\langle f_m|f_{m-1}\rangle}_0 = 0$$

$$\boxed{a_m = \frac{\langle f_m|H|f_m\rangle}{\langle f_m|f_m\rangle}}$$

$$\langle f_{m-1}|f_{m+1}\rangle = \langle f_{m-1}|H|f_m\rangle - a_m\underbrace{\langle f_{m-1}|f_m\rangle}_0 - b_{m-1}\langle f_{m-1}|f_{m-1}\rangle = 0$$

$$\langle f_{m+1}| = \langle f_m|H - a_m^*\langle f_m| - b_{m-1}^*\langle f_{m-1}|$$

$$\begin{aligned}\langle f_m | &= \langle f_{m-1} | H - a_{m-1}^* \langle f_{m-1} | - b_{m-2}^* \langle f_{m-2} | \\ \langle f_{m-1} | H &= \langle f_m | + a_{m-1}^* \langle f_{m-1} | + b_{m-2}^* \langle f_{m-2} |\end{aligned}$$

$$\langle f_{m-1} | f_{m+1} \rangle = (\langle f_m | + a_{m-1}^* \langle f_{m-1} | + b_{m-2}^* \langle f_{m-2} |) | f_m \rangle - b_{m-1} \langle f_{m-1} | f_{m-1} \rangle = 0$$

$$\boxed{b_{m-1} = \frac{\langle f_m | f_m \rangle}{\langle f_{m-1} | f_{m-1} \rangle}}$$

Aby przedstawić macierz H w tej bazie należy skorzystać z równania (2.4):

$$H|f_m\rangle = |f_{m+1}\rangle + a_m|f_m\rangle + b_{m-1}|f_{m-1}\rangle$$

Mnożąc to równanie lewostronnie przez wektory $\langle f_{m-1} |$, $\langle f_m |$, $\langle f_{m+1} |$ dostaniemy niezerowe elementy macierzy tródiagonalnej T_{kk} :

$$\begin{aligned}\langle f_{m-1} | H | f_m \rangle &= b_{m-1} \langle f_{m-1} | f_{m-1} \rangle = \langle f_m | f_m \rangle \\ \langle f_m | H | f_m \rangle &= a_m \langle f_m | f_m \rangle = \langle f_m | H | f_m \rangle \\ \langle f_{m+1} | H | f_m \rangle &= \langle f_{m+1} | f_{m+1} \rangle\end{aligned}$$

Rozważania te dotyczyły stanów $|f_m\rangle$ nieunormowanych. Dla bazy znormalizowanej $|\phi_m\rangle$ mamy:

$$|\phi_m\rangle = \frac{|f_m\rangle}{\sqrt{\langle f_m | f_m \rangle}}$$

W takiej bazie macierz T_{kk} będzie miała postać:

$$\sqrt{\langle f_{m-1} | f_{m-1} \rangle} \sqrt{\langle f_m | f_m \rangle} \langle \phi_{m-1} | H | \phi_m \rangle = \langle f_m | f_m \rangle$$

$$\langle \phi_{m-1} | H | \phi_m \rangle = \sqrt{\frac{\langle f_m | f_m \rangle}{\langle f_{m-1} | f_{m-1} \rangle}}$$

$$\boxed{\langle \phi_{m-1} | H | \phi_m \rangle = \sqrt{b_{m-1}}}$$

$$\cancel{\langle f_m | f_m \rangle} \langle \phi_m | H | \phi_m \rangle = a_m \cancel{\langle f_m | f_m \rangle}$$

$$\boxed{\langle \phi_m | H | \phi_m \rangle = a_m}$$

$$\sqrt{\langle f_{m+1} | f_{m+1} \rangle} \sqrt{\langle f_m | f_m \rangle} \langle \phi_{m+1} | H | \phi_m \rangle = \langle f_{m+1} | f_{m+1} \rangle$$

$$\langle \phi_{m+1} | H | \phi_m \rangle = \sqrt{\frac{\langle f_{m+1} | f_{m+1} \rangle}{\langle f_m | f_m \rangle}}$$

$$\langle \phi_{m+1} | H | \phi_m \rangle = \sqrt{b_m}$$

$$T_{kk} = \begin{pmatrix} a_0 & \sqrt{b_0} & & & 0 \\ \sqrt{b_0} & a_1 & \sqrt{b_1} & & \\ & \sqrt{b_1} & \ddots & \ddots & \\ & & \ddots & a_{k-1} & \sqrt{b_{k-1}} \\ 0 & & & \sqrt{b_{k-1}} & a_k \end{pmatrix}$$

Dzięki otrzymanej macierzy T_{kk} w bardzo prosty sposób można otrzymać k (wybrane z widma) wartości własne macierzy H , stosując znane algorytmy diagonalizacji macierzy gęstych (lub specjalne dla macierzy trójdiagonalnych). Należy podkreślić, że zbieżność wartości jest najszybsza na skrajach widma macierzy H . Największe korzyści z zastosowanego algorytmu pochodzą od wymiarów macierzy – stopień T_{kk} wynosi k . W przeprowadzonych obliczeniach wartość $k \ll n$. Dla przykładu, w pracy wykonywano obliczenia na macierzach o stopniu $n = 131\,072$, w tym przypadku wystarczający wymiar k dający poprawne wyniki to $k = 50$. Zakładając, że są to macierze gęste o elementach standardowych `complex<double>16B` to koszt pamięciowy macierzy H wynosi około 256 GB, a macierzy T_{kk} 39 kB.

2.1.4 Implementacja algorytmu

W poprzednim podrozdziale opisana została metoda generowania bazy Lanczosa, która różni się od metod zaimplementowanych bezpośrednio w komputerach. Problem może pojawić się przy metodach bezpośrednich przy normalizacji, kiedy norma wektora jest bardzo duża. Może to prowadzić do niestabilności algorytmu, dlatego proces normalizacji przeprowadza się podczas generowania stanów bazy Lanczosa:

$$|\phi_{m+2}\rangle = \frac{1}{\sqrt{\langle \phi_m | \phi_m \rangle}} (H|\phi_{m+1}\rangle - a_m|\phi_{m+1}\rangle - b_m|\phi_m\rangle)$$

W pracy korzystano z algorytmu w następującej postaci w pseudokodzie:

Algorytm 1 Algorytm lanczosa dla macierzy A , k kroków.

```

1: function LANCZOS( $A$ ,  $k$ )
2:    $|v_0\rangle = \text{random}()$ ;
3:    $|v_1\rangle = A \cdot |v_0\rangle$ ;
4:    $\alpha[0] = \langle v_0|v_1\rangle$ ;
5:    $|v_1\rangle = |v_1\rangle - \alpha[0] \cdot |v_0\rangle$ ;
6:    $\text{nrm} = \sqrt{\langle v_1|v_1\rangle}$ 
7:   for  $i = 1, i < k, i++$  do
8:      $|v_2\rangle = A \cdot |v_1\rangle$ ;
9:      $\alpha[i] = \langle v_1|v_2\rangle$ ;
10:     $|v_2\rangle = |v_2\rangle - \alpha[i] \cdot |v_1\rangle$ ;
11:     $\beta[i-1] = \text{nrm}$ ;
12:     $|v_2\rangle = |v_2\rangle - \text{nrm} \cdot |v_0\rangle$ ;
13:     $\text{nrm} = \sqrt{\langle v_2|v_2\rangle}$ ;
14:     $|v_2\rangle = |v_2\rangle / \text{nrm}$ ;
15:     $|v_0\rangle = |v_1\rangle$ ;
16:     $|v_1\rangle = |v_2\rangle$ ;
17:   end for
18: end function

```

Niestety taki algorytm dla bardzo dużych wartości k będzie generował stany, które będą traciły swoją własność ortogonalności – dokładność numeryczna. W celu zwiększenia dokładności należałoby po każdej generacji kolejnego wektora przeprowadzić procedurę reortogonalizującą bazę wygenerowanych wektorów (np. poprzez ortogonalizację Grama-Schmitda).

2.1.5 Generowanie stanu podstawowego

Stan podstawowy $|\psi\rangle$ generujemy sumując wektory bazy Lanczosa $|\phi_m\rangle$, a następnie domnażając je przez odpowiednią współrzędną wektora własnego w dla odpowiadającej mu najmniejszej wartości własnej macierzy T_{kk} :

$$|\psi\rangle = \sum_{m=0}^k w(m)|\phi_m\rangle.$$

W algorytmie nie przechowywano wszystkich wektorów $|\phi_m\rangle$ (jedynie 3 ostatnie dla każdej iteracji $m : |\phi_{m+1}\rangle, |\phi_m\rangle, |\phi_{m-1}\rangle$), a procedurę ich generacji rekurencyjnej uruchomiano po procesie diagonalizacji macierzy T_{kk} , co zdecydowanie oszczędziło pamięć.

2.1.6 Generowanie stanów dla dowolnej temperatury

Przedstawiony algorytm Lanczosa daje możliwość poprawnego wyznaczenia stanu jedynie na brzegach widma – tzw. *ground state*. Podczas obliczeń

zależy nam na obserwacji propagacji stanów o ustalonej temperaturze. W dalszej części posługujemy się odwrotną temperaturą³ β . W przybliżeniu β można powiązać z energią kinetyczną E_k następującą zależnością [12]:

$$E_k \approx -\frac{\beta L}{2}$$

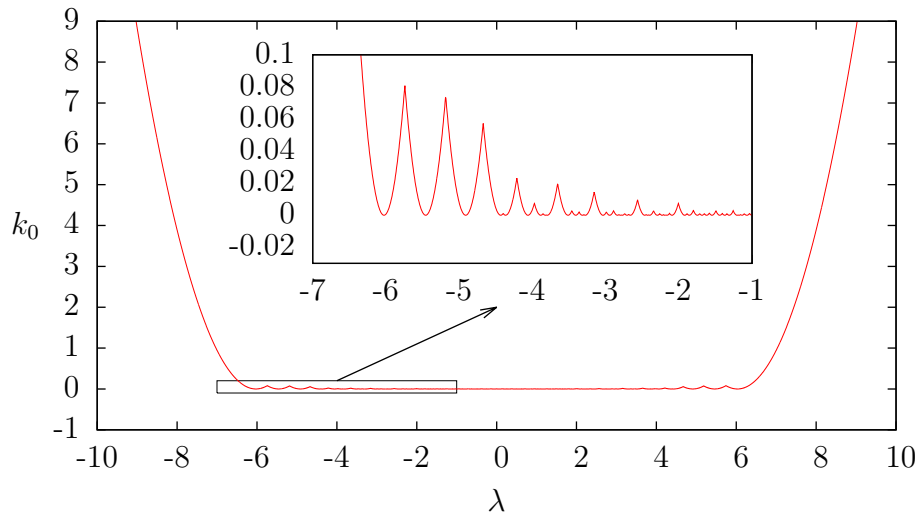
W celu znalezienia stanu o określonej energii wprowadza się specjalny operator K :

$$K = (H - \lambda \mathbb{1})^2$$

Można pokazać, że tak stworzony operator ma taką własność, że dla zadanej energii λ wygenerowany będzie stan najbliższy do energii λ . Przesuwając widmo operatora H o energię λ można generować stany ze „środka widma”. Rozwiązując zagadnienie własne operatora K przy otrzymanych wartościach własnych $k = 0$ otrzymujemy stany własne operatora H o energiach λ .

$$\begin{aligned} H|\psi\rangle &= \lambda|\psi\rangle \\ (H - \lambda\mathbb{1})|\psi\rangle &= 0 \\ (H - \lambda\mathbb{1})^2|\psi\rangle &= 0 \\ K|\psi\rangle &= 0 \\ \left\{ \begin{array}{l} K|\psi\rangle = k|\psi\rangle \\ k = 0 \end{array} \right. \end{aligned}$$

Na rysunku (2.2) pokazany został przykład działania algorytmu dla hamiltonianu modelu ciasnego wiązania (równania 1.3-1.5) o $L = 10$, $N = 5$ dla potencjałów $V = W = 0$.



Rysunek 2.2: Wykres energii własnej k (najmniejszej) operatora K od energii λ .

³gdzie β to odwrotność temperatury T : $\beta = \frac{1}{k_B T}$, a k_B to stała Boltzmana

2.2 Ewolucja czasowa Czebyszewa

Dany podrozdział został napisany na podstawie prac znajdujących się w bibliografii na pozycjach [13, 14].

2.2.1 Wielomiany Czebyszewa

Wielomiany Czebyszewa to układ wielomianów ortogonalnych. Ich definicja rekurencyjna jest następująca:

$$T_0(x) = 1$$

$$T_1(x) = x$$

$$T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) \quad (2.5)$$

gdzie T_k jest wielomianem Czebyszewa pierwszego rodzaju rzędu k , a $x \in [-1, 1]$.

Przyjmując taką definicję, można wykazać, że wielomiany, które spełniają równanie (2.5), można przedstawić w postaci:

$$T_k(x) = \cos(k \arccos(x)) \quad (2.6)$$

□

Aby pokazać ten związek skorzystano ze znanej tożsamości trygonometrycznej:

$$\cos \alpha \cos \beta = \frac{1}{2}(\underbrace{\cos(\alpha + \beta)}_{\psi} + \underbrace{\cos(\alpha - \beta)}_{\theta}) \quad (2.7)$$

$$\cos \psi + \cos \theta = 2 \cos \frac{\psi + \theta}{2} \cos \frac{\psi - \theta}{2}$$

Następnie dokonano podstawienia: $\psi = k\varphi$, $\theta = (k-2)\varphi$, gdzie $k \in \mathbb{N}$:

$$\cos k\varphi + \cos(k-2)\varphi = 2 \cos[(k-1)\varphi] \cos \varphi$$

$$\cos k\varphi = 2 \cos[(k-1)\varphi] \cos \varphi - \cos(k-2)\varphi$$

Podstawiając $x = \cos \varphi$ otrzymujemy definicję rekurencyjną z równania (2.5):

$$\underbrace{\cos k\varphi}_{T_n(x)} = 2x \underbrace{\cos[(k-1)\varphi]}_{T_{k-1}(x)} - \underbrace{\cos(k-2)\varphi}_{T_{k-2}}$$

■

2.2.2 Ortogonalność wielomianów Czebyszewa

Pracując w przestrzeni wielomianów Czebyszewa definiujemy następujący iloczyn skalarny:

$$\langle f|g \rangle = \int_{-1}^1 w(x)f(x)g(x)dx$$

gdzie $w(x)$ to wagowa funkcja $w(x) = \frac{1}{\pi} \frac{1}{\sqrt{1-x^2}}$.

Pokażemy teraz, że wielomiany dla dowolnego $n \neq m$ są ortogonalne przy tak zdefiniowanym iloczynie skalarnym, czyli:

$$\forall_{n,m} \langle T_n|T_m \rangle = \alpha_n \delta_{nm}$$

gdzie δ_{nm} to delta Kroneckera, a α_n to współczynnik, który wynosi 1 dla $n = 0$ i $\frac{1}{2}$ dla $n \neq 0$.

□

Rozważymy trzy przypadki:

1) $n \neq m$:

$$\begin{aligned} \langle T_n|T_m \rangle &= \int_{-1}^1 w(x)T_n(x)T_m(x)dx = \\ &= \int_{-1}^1 \frac{1}{\pi} \frac{1}{\sqrt{1-x^2}} \cos(n \arccos(x)) \cos(m \arccos(x))dx = \end{aligned}$$

Dokonano podstawienia zmiennych $t = \arccos x$, $\frac{dt}{dx} = -\frac{1}{\sqrt{1-x^2}}$, a następnie skorzystano z tożsamości (2.7):

$$\begin{aligned} &= - \int_{\pi}^0 \frac{1}{\pi} \cos(nt) \cos(mt)dt = \frac{1}{2\pi} \int_0^{\pi} (\cos((n-m)t) + \cos((n+m)t)) dt^{(*)} = \\ &= \frac{1}{2\pi} \frac{1}{n-m} \sin((n-m)t) \Big|_0^{\pi} + \frac{1}{2\pi} \frac{1}{n+m} \sin((n-m)t) \Big|_0^{\pi} = 0 \end{aligned}$$

2) $n = m \neq 0$ – wróćmy do całki oznaczonej (*):

$$\frac{1}{2\pi} \int_0^{\pi} (\cos(0) + \cos(2nt)) dt = \frac{1}{2} + \frac{1}{2\pi} \frac{1}{2n} \sin(2nt) \Big|_0^{\pi} = \frac{1}{2}$$

3) $n = m = 0$ – wróćmy do całki oznaczonej ^(*)

$$\frac{1}{2\pi} \int_0^\pi (\cos(0) + \cos(0)) dt = 1$$

■

Ostatecznie:

$$\langle T_n | T_k \rangle = \begin{cases} 0 & : n \neq m \\ 1 & : n = m = 0 \\ \frac{1}{2} & : n = m \neq 0 \end{cases} \quad (2.8)$$

2.2.3 Rozwinięcie funkcji w szereg Czebyszewa

Zbiór wielomianów Czebyszewa tworzy ortonormalną bazę, a zatem każdą funkcję kawałkami gładką i ciągłą na przedziale $-1 \leq x \leq 1$ można zapisać za pomocą rozwinięcia w następujący szereg:

$$f(x) = \sum_{k=0}^{\infty} c_k T_k(x)$$

Pomnożymy obie strony równania przez $|\cdot w(x)T_n(x)$:

$$\begin{aligned} w(x)f(x)T_n(x) &= \sum_{k=0}^{\infty} c_k w(x)T_k(x)T_n(x) \quad \left| \int_{-1}^1 dx \right. \\ \int_{-1}^1 w(x)f(x)T_n(x)dx &= \sum_{k=0}^{\infty} c_k \int_{-1}^1 w(x)T_k(x)T_n(x)dx \\ \langle f | T_n \rangle &= \sum_{k=0}^{\infty} c_k \langle T_k | T_n \rangle \quad \left| : \langle T_n | T_n \rangle \right. \\ \frac{\langle f | T_n \rangle}{\langle T_n | T_n \rangle} &= \sum_{k=0}^{\infty} c_k \underbrace{\frac{\langle T_k | T_n \rangle}{\langle T_n | T_n \rangle}}_{\delta_{kn}} = c_n \end{aligned}$$

Ostatecznie:

$$c_k = \frac{\langle f | T_k \rangle}{\langle T_k | T_k \rangle}$$

Współczynnik c_k dla $k = 0$ wynosi $c_0 = \langle T_0 | f \rangle$, a dla $k > 0$: $c_k = 2\langle T_k | f \rangle$. Definiując nowe współczynniki $\alpha_k = \langle T_k | f \rangle$ otrzymujemy następującą postać rozwinięcia funkcji $f(x)$:

$$f(x) = \langle T_0 | f \rangle \underbrace{T_0(x)}_1 + 2 \sum_{k=1}^{\infty} \langle T_k | f \rangle T_k(x) = \alpha_0 + 2 \sum_{k=1}^{\infty} \alpha_k T_k(x)$$

Taka postać jest najczęściej spotykana w literaturze.

2.2.4 Propagacja czasowa wektora stanu

Zależne od czasu równanie Schrodingera dane jest wzorem:

$$i\hbar|\dot{\psi}(t)\rangle = \hat{H}(t)|\psi(t)\rangle$$

Ogólne rozwiązanie tego równania ma postać:

$$|\psi(t)\rangle = U(0, t)|\psi(0)\rangle$$

gdzie $U(0, t)$ to operator propagacji czasu od czasu 0 do czasu t . Analitycznie taki propagator ma postać:

$$U(0, t) = T_\tau e^{-i \int_0^t \hat{H}(\tau) d\tau / \hbar}$$

gdzie T_τ to operator uporządkowania w czasie. Na potrzeby obliczeń numerycznych działanie takiego operatora można zastąpić rekurencyjnym działaniem operatora o małym kroku czasowym δt , zakładając wtedy pełną liniowość operatora \hat{H} na tak małym przedziale.

$$U(0, t) = \lim_{\delta t \rightarrow 0} U(t - \delta t) \cdots U(\delta t, 2\delta t) U(0, \delta t)$$

Korzystając z tych założeń operacja rekurencyjna ewolucji czasowej wektora stanu dla ciągu czasów $\{t_n\}_0^t$ ma następującą postać:

$$|\psi(t_n)\rangle = e^{-i\hat{H}(t_n)\delta t/\hbar} |\psi(t_{n-1})\rangle \quad (2.9)$$

Aproksymację operatora $e^{-i\hat{H}(t_n)\delta t/\hbar}$ można dokonać korzystając z wcześniej wyprowadzonego rozwinięcia funkcji w szereg Czebyszewa.

Analitycznie można obliczyć rozwinięcie funkcji postaci $f(x) = e^{-ixt}$ (pamiętając, że $x \in [-1, 1]$):

$$f(x) = \alpha_0 + 2 \sum_{k=1}^{\infty} \alpha_k T_k(x)$$

$$\alpha_k = \langle f | T_k \rangle = \int_{-1}^1 \frac{1}{\pi} \frac{1}{\sqrt{1-x^2}} T_k(x) e^{-ixt} dx =$$

Dokonano podstawienia zmiennych $x = -y$, $\frac{dy}{dx} = -1$ i skorzystano z własności parzystości wielomianów Czebyszewa $T_k(-x) = (-1)^k T_k(x)$:

$$= \int_{-1}^1 \frac{1}{\pi} \frac{1}{\sqrt{1-y^2}} T_k(-y) e^{iyt} dy = \int_{-1}^1 \frac{1}{\pi} \frac{1}{\sqrt{1-y^2}} (-1)^k T_k(y) e^{iyt} dy =$$

$$= \int_{-1}^1 \frac{1}{\pi} \frac{1}{\sqrt{1-y^2}} (-1)^k \cos(k \arccos y) e^{iyt} dy =$$

Dokonano podstawienia zmiennych $\theta = \arccos y$, $\frac{d\theta}{dy} = -\frac{1}{\sqrt{1-y^2}}$:

$$\begin{aligned} &= - \int_{\pi}^0 \frac{1}{\pi} (-1)^k \cos(k\theta) e^{i \cos(\theta) \cdot t} d\theta = (-1)^k \int_0^{\pi} \frac{1}{\pi} \cos(k\theta) e^{i \cos(\theta) \cdot t} d\theta = \\ &= (-1)^k i^k \underbrace{\frac{i^{-k}}{\pi} \int_0^{\pi} \cos(k\theta) e^{i \cos(\theta) \cdot t} d\theta}_{J_k(t)} = (-i)^k J_k(t) \end{aligned}$$

$$\boxed{\alpha_k(t) = (-i)^k J_k(t)}$$

gdzie J_n to funkcje Bessla pierwszego rodzaju rzędu k .

Dla badanego układu w tej pracy, czyli jednowymiarowej sieci bezspino-
wych fermionów w formalizmie drugiej kwantyzacji, hamiltonian \hat{H} posiada
skończone spektrum energetyczne zawierające się w przedziale $[E_{\min}, E_{\max}]$.
W pierwszym kroku dokonujemy transformacji liniowej $\hat{H} \rightarrow \tilde{H}$ tak aby nasze
widmo zmieściło się w przedziale $[-1,1]$:

$$\tilde{H} = \frac{1}{a}(\hat{H} - b\mathbb{1})$$

Współczynniki a , b wyznaczamy znając wartości E_{\min} , E_{\max} , które bardzo
łatwo można wyznaczyć korzystając z algorytmu Lanczosa.

$$\begin{aligned} a &= \frac{1}{2}(E_{\max} - E_{\min} + \varepsilon) \\ b &= \frac{1}{2}(E_{\max} + E_{\min}) \\ \varepsilon &= \alpha(E_{\max} - E_{\min}) \end{aligned}$$

wartość ε w wyznaczaniu wartości a uwzględnia błąd wyznaczenia przedzia-
łu energetycznego, tak żeby całe widmo zostało przeskalowane do przedziału
 $[-1,1]$. Niepewność na stałe ustalana jest na wartość $\alpha = 0.01$.

Transformując hamiltonian do \tilde{H} i rozwijając eksponentę z równania (2.9)
otrzymujemy:

$$|\psi(t_n)\rangle = e^{-ib\delta t/\hbar} \left(\alpha_0(a\delta t/\hbar) + 2 \sum_{k=1}^{\infty} \alpha_k(a\delta t/\hbar) T_k(\tilde{H}) \right) |\psi(t_{n-1})\rangle$$

Wracając do definicji rekurencyjnej wielomianów Czebyszewa (2.5) problem
aproxymacji propagatora sprowadza się do operacji mnożenia *macierz razy*
wektor z odpowiednimi współczynnikami α_k :

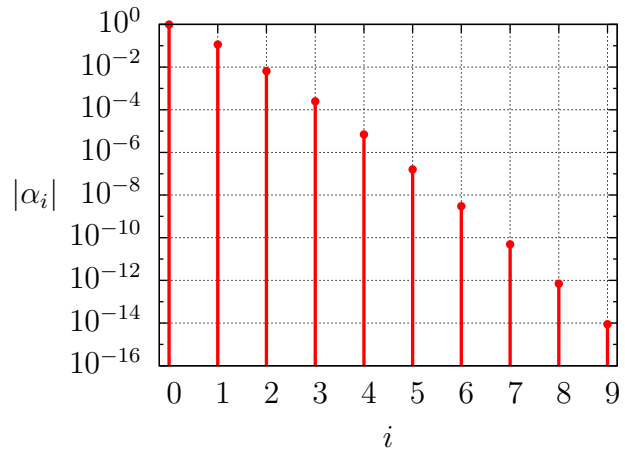
$$\begin{aligned}
T_0(\tilde{H}) &= \mathbb{1} \\
T_1(\tilde{H}) &= \tilde{H} \\
T_k(\tilde{H}) &= 2\tilde{H}T_{k-1}(\tilde{H}) - T_{k-2}(\tilde{H})
\end{aligned}$$

Wprowadzając wektory pomocnicze $|\nu_k\rangle$ zdefiniowane jako $|\nu_k\rangle = T_k(\tilde{H})|\psi(t_{n-1})\rangle$ otrzymujemy zależność rekurencyjną:

$$\begin{aligned}
|\nu_0\rangle &= |\psi(t_{n-1})\rangle \\
|\nu_1\rangle &= \tilde{H}|\psi(t_{n-1})\rangle \\
|\nu_k\rangle &= 2\tilde{H}|\nu_{k-1}\rangle - |\nu_{k-2}\rangle
\end{aligned}$$

$$|\psi(t_n)\rangle = e^{-ib\delta t/\hbar} \left(\alpha_0(a\delta t/\hbar)|\nu_0\rangle + 2 \sum_{k=1}^{\infty} \alpha_k(a\delta t/\hbar)|\nu_k\rangle \right)$$

Oczywiście przy obliczeniach numerycznych nie możemy obliczyć sumy nieskończonej liczby elementów szeregu, najczęściej wyznacza się sumę $M = 5 \div 15$ elementów, co w zupełności wystarcza do odtworzenia wyników analitycznych. Warto zwrócić uwagę, że współczynniki α_k dla ustalonego kroku δt nie zmieniają się, więc wystarczy je raz obliczyć i przechowywać w pamięci.



Rysunek 2.3: Wykres wartości bezwzględnej współczynników $|\alpha_i|$ rozwiniecia w szereg Czebyszewa funkcji e^{-ixt} .

Rozdział 3

Wyniki

3.1 Parametry

Dany podrozdział został napisany na podstawie pracy znajdującej się w bibliografii na pozycji [15].

3.1.1 Sprzęt komputerowy

Poniżej podano specyfikację sprzętową komputera PC (wraz z wybranym systemem operacyjnym), na którym prowadzone były obliczenia:

Procesor	Intel(R) Xeon(R) E5-1620 @ 3.50GHz (8 wątków)
Płyta główna	Dell Inc. 0K240Y
Pamięć RAM	Samsung, M393A5143DB0-CPB 4x4GB @ 2133MHz
System operacyjny	Ubuntu 14.04.4 LTS

3.1.2 Oprogramowanie

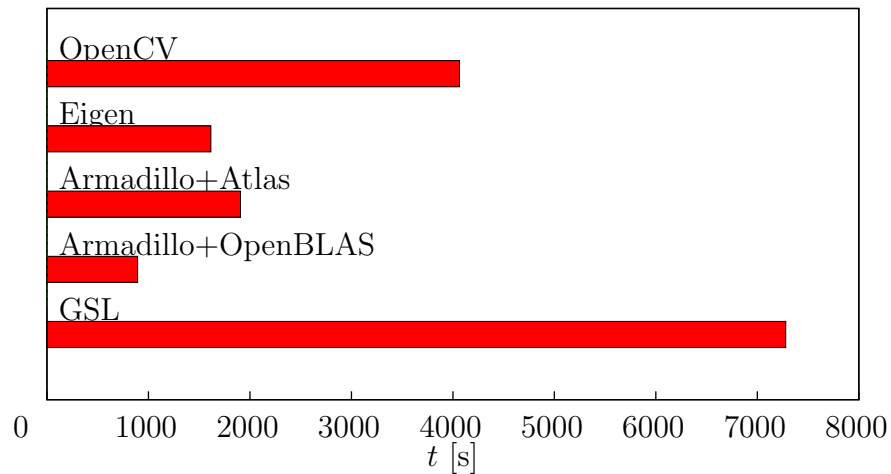
Programy wykonujące symulację zostały napisane w języku C++ z wykorzystaniem biblioteki `Armadillo`[15]. Wybrano tę bibliotekę ze względu na jej prostotę i ogromną wydajność. Domyślnie `Armadillo` używa biblioteki `BLAS` do mnożenia macierzy i innych podstawowych operacji, ale można dołączyć inne biblioteki np. `OpenBLAS`, która jest wielordzeniową wersją biblioteki `BLAS` (obliczenia równoległe). Można wykorzystać również bibliotekę `NVBLAS` czy `ACML`, w celu wykorzystania mocy obliczeniowej kart graficznych (GPGPU) do obliczeń numerycznych.

Poniżej zaprezentowano wykres, na którym porównano wydajności wybranych popularnych bibliotek¹. Pomiar wydajności odbywał się poprzez pomiar czasu rozwiązywania problemu układu liniowego:

¹Wykres na podstawie danych z postu dostępnego na witrynie internetowej: <http://nghiaho.com/?p=1726> [dostęp online 24.06.16]

$$AX = B$$

gdzie A jest macierzą o wymiarach $N \times M$, N jest dużo większe niż M , a B jest macierzą o wymiarach $N \times 1$. Dla przedstawionych danych $N = 1\,000\,000$, a $M = 16$.



Rysunek 3.1: Wykres porównujący działanie wybranych bibliotek.¹

Jak widać z wyżej wymienionych bibliotek podczas rozwiązywania wspomnianego wcześniej problemu biblioteka **Armadillo** wraz z **OpenBLAS** jest najbardziej wydajna pod względem czasowym.

3.2 Ewolucja czasowa

Dany podrozdział został napisany na podstawie prac znajdujących się w bibliografii na pozycjach [4, 16].

3.2.1 Porównanie z metodą Eulera

Pracując w obrazie Heisenberga można analitycznie znaleźć postać prądu cząstek w funkcji czasu t : $\langle \hat{j} \rangle(t)$ oraz energii układu w funkcji czasu t : $\langle \hat{H} \rangle(t)$ dla modelu ciasnego wiązania (równania 1.3-1.5) dla potencjałów $V = W = 0$ i przy zmieniającej się liniowo fazie Peierlsa $\varphi = -Ft$. F jest to stała odpowiedzialna za szybkość zmiany pola magnetycznego, w którym znajduje się badany układ. W obrazie Heisenberga ewolucja w czasie dowolnej obserwabli hermitowskiej \hat{A} dana jest następującym równaniem:

$$\frac{d}{dt} \langle \hat{A} \rangle = \frac{i}{\hbar} \langle \psi | [\hat{H}, \hat{A}] | \psi \rangle + \langle \psi | \frac{\partial \hat{A}}{\partial t} | \psi \rangle$$

Policzmy $\frac{d}{dt}\langle\hat{j}\rangle$ – obserwabla prądu \hat{j} komutuje z hamiltonianem \hat{H} , jeśli $V = W = 0$, a zatem:

$$\frac{d}{dt}\langle\hat{j}\rangle = \frac{i}{\hbar}\langle\psi|\underbrace{[\hat{H}, \hat{j}]}_0|\psi\rangle + \langle\psi|\frac{\partial\hat{j}}{\partial\varphi}|\psi\rangle \underbrace{\dot{\varphi}}_{-F} \quad (3.1)$$

Łatwo można zauważyć, że :

$$\frac{\partial\hat{j}}{\partial\varphi} = \hat{H}$$

Wracając do równania (3.1) otrzymujemy:

$$\frac{d}{dt}\langle\hat{j}\rangle = -F\langle\hat{H}\rangle$$

Analogicznie policzono $\frac{d}{dt}\langle\hat{H}\rangle$ otrzymując:

$$\frac{d}{dt}\langle\hat{H}\rangle = F\langle\hat{j}\rangle$$

Ostatecznie otrzymano układ równań różniczkowych:

$$\begin{cases} \frac{d}{dt}\langle\hat{j}\rangle &= -F\langle\hat{H}\rangle \\ \frac{d}{dt}\langle\hat{H}\rangle &= F\langle\hat{j}\rangle \end{cases}$$

Rozwiązaniem takiego układu (co do stałej a) jest:

$$\boxed{\begin{cases} \langle\hat{j}\rangle(t) &= a \cos\left(\frac{1}{F}t\right) \\ \langle\hat{H}\rangle(t) &= a \sin\left(\frac{1}{F}t\right) \end{cases}}$$

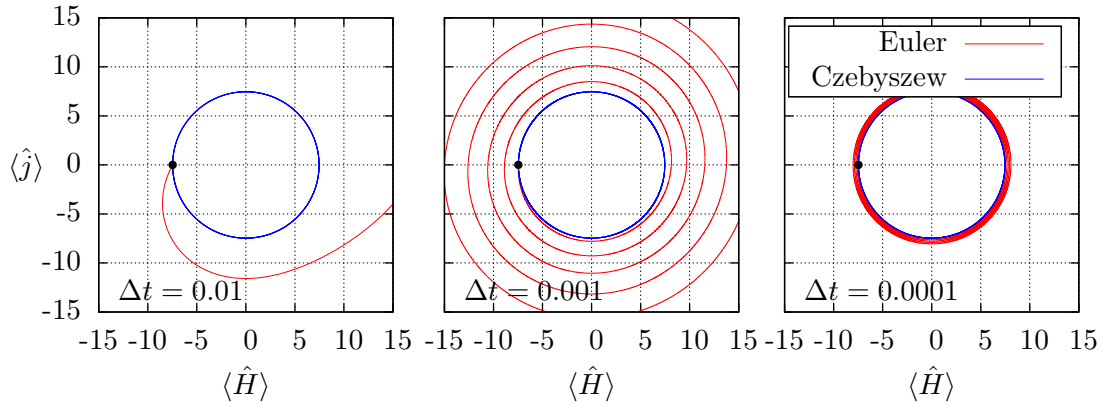
Badanie takiej ewolucji może posłużyć do weryfikacji poprawności implementacji algorytmów omawianych w rozdziale 2. Odłożenie tych funkcji na wykresie parametrycznym $(\langle\hat{j}\rangle, \langle\hat{H}\rangle)(t)$ powinno dać w wyniku okrąg.

Poniżej zaprezentowano (rysunek 3.2) porównanie działania metody propagacji Czebyszewa opisanej w sekcji 2.2 z klasyczną metodą Eulera² do rozwiązywania równań różniczkowych. Rozwiązywano równanie Schrödingera w zależności od czasu dla stanu podstawowego wygenerowanego dla układu o $L = 12$, $N = L/2$ i oczywiście z $V = W = 0$ i całą przeskoku $t = 1$, dla metody Czebyszewa brano $M = 10$ wielomianów Czebyszewa do rozwinięcia propagatora czasowego.

Metoda Eulera okazuje się całkowicie bezużyteczna przy takich zagadnieniach numerycznych, dopiero przy ekstremalnie małym kroku czasowym Δt

²Korzystano z metody Eulera wg. schematu: $|\psi(t_n)\rangle = |\psi(t_{n-1})\rangle - \Delta t \frac{i}{\hbar} \hat{H}(t) |\psi(t_{n-1})\rangle$.

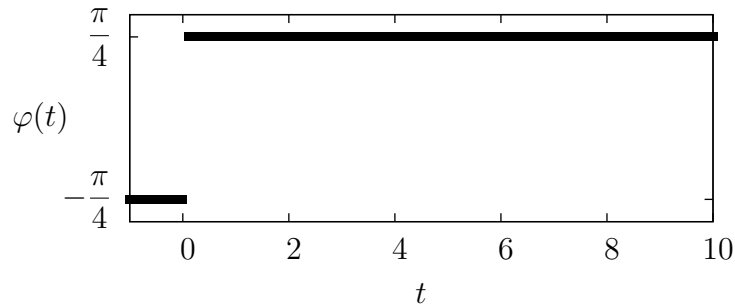
daje w przybliżeniu poprawne wyniki, za to algorytm Czebyszewa jest bardzo stabilny. Błędy pojawiają się dopiero przy dużych krokach czasowych ($\Delta t > 1$).



Rysunek 3.2: Porównanie efektywności metody Eulera z metodą Czebyszewa. Obliczenia wykonywano przy założeniu jednostek $\hbar = 1$, dla czasów $t \in [0, 30]$ (na wykresach zaznaczono kroki czasowe), punkt startowy ewolucji to: $(-a, 0)$.

3.2.2 Relaksacja prądu

Inną weryfikacją napisanych kodów było badanie ewolucji obserwabli prądu $\langle \hat{j} \rangle$ dla dwóch przypadków: $V = 1, W = 0$ oraz $V = W = 1$ przy skokowym³ przełączaniu fazy φ od wartości $-\frac{\pi}{4}$ do $\frac{\pi}{4}$ (pracowano w jednostkach $\hbar = 1$):



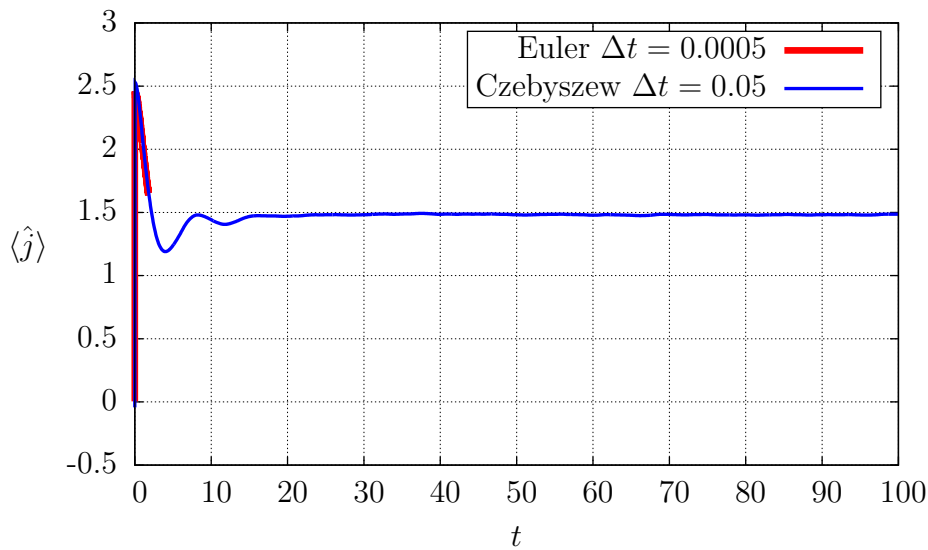
Rysunek 3.3: Wykres skokowej zmiany fazy φ .

Ewolucji poddawano stany wygenerowane o odwrotnej temperaturze $\beta = 0.2$ dla układu $L = 24, N = L/2$ hamiltonianu dla modelu ciasnego wiązania z oddziaływaniami (równania 1.3-1.5).

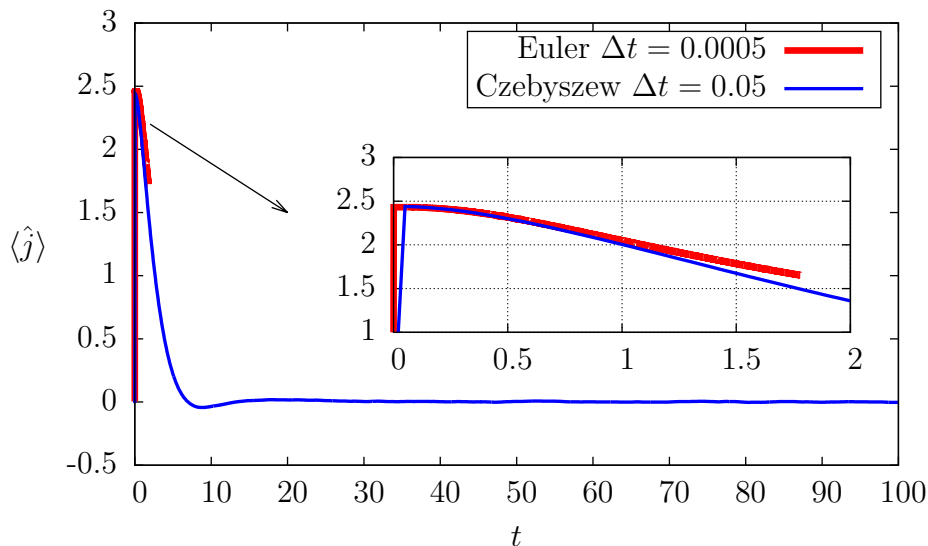
Przypadek, gdzie $V = 1, W = 0$ odpowiada układowi całkowalnemu, prąd w takim układzie pomimo nieprzykładanego pola poza krótkim impulsem nie

³Odpowiada to nieskończenie krótkiemu impulsowi przyłożonego pola elektrycznego \vec{E} nad układem.

zanika. W przypadku $V = W = 1$ prąd relaksuje – po pewnym czasie zanika. Na rysunkach (3.4, 3.5) przedstawiono otrzymane wyniki. Porównano tutaj działanie algorytmu Czebyszewa z metodą Eulera. Maksymalny krok, który nie dawał rozbieżnych wyników podczas ewolucji takiego układu dla metody Eulera wynosił $\Delta t = 0.0005$. Zaznaczony fragment na wykresach, odpowiadający ewolucji metodą Eulera, liczony był przez około 72h. Oszacowano, że obliczenia dla tej metody dla zakresu czasów $t \in [0, 100]$ trwałyby około pół roku. W tym zakresie czasów, dla metody Czebyszewa obliczenia trwały poniżej 24h.



Rysunek 3.4: Wykres $\langle \hat{j} \rangle(t)$ dla $V = 1, W = 0$.



Rysunek 3.5: Wykres $\langle \hat{j} \rangle(t)$ dla $V = 1, W = 1$.

Podsumowanie

Podsumowując, w pierwszym rozdziale czytelnik został wprowadzony w teorię i model nanodrutu opisywany hamiltonianem modelu ciasnego wiązania, w drugim rozdziale przedstawiono zasadę działania algorytmów: Lanczosa i Czebyszewa, a w ostatnim rozdziale zaprezentowano otrzymane wyniki wykorzystując w tym celu opisane wcześniej algorytmy.

W odróżnieniu od algorytmu Czebyszewa, który można stosować z względnie dużymi krokami czasowymi Δt unikając przy tym dużych błędów numerycznych, dość naiwny schemat Eulera całkowicie nie nadaje się do propagowania wektorów stanu przy badaniu modeli nanodrutów kwantowych. W drugim rozdziale pokazano dużą stabilność algorytmu Czebyszewa pomimo dużego kroku czasowego.

Wykorzystując algorytm Lanczosa, na współczesnych komputerach (standardowych) jesteśmy w stanie rozwiązywać zagadnienia dla układów do rozmiaru około $L = 30$. Stosując procedurę pełnej diagonalizacji limit rozmiaru układu jaki możemy rozwiązać jest sporo mniejszy i wynosi około $L = 18$. Oczywiście przewagą procedury pełnej diagonalizacji jest dokładność oraz fakt otrzymania pełnego zestawu stanów własnych układu. Ogromną przewagą algorytmu Lanczosa nad procedurami pełnej diagonalizacji jest możliwość badania układów o dużo większym L , co pozwala lepiej badać układ pod kierunkiem dążenia do granicy termodynamicznej ($L \rightarrow \infty$).

Spis rysunków

1.1	Schemat jednowymiarowej sieci	3
1.2	Niezerowe elementy hamiltonianu dla modelu ciasnego wiązania.	5
2.1	Zbieżność normy $\ x_n\ $ dla metody potęgowej.	10
2.2	Wykres energii własnej k (najmniejszej) operatora K od energii λ	15
2.3	Wykres wartości bezwzględnej współczynników $ \alpha_i $ rozwinięcia w szereg Czebyszewa funkcji e^{-ixt}	21
3.1	Wykres porównujący działanie wybranych bibliotek.	23
3.2	Porównanie efektywności metody Eulera z metodą Czebyszewa.	25
3.3	Wykres skokowej zmiany fazy φ	25
3.4	Wykres $\langle \hat{j} \rangle(t)$ dla $V = 1, W = 0$	26
3.5	Wykres $\langle \hat{j} \rangle(t)$ dla $V = 1, W = 1$	26

Bibliografia

- [1] H.Orland J.W. Negele. *Quantum Many-Particle Systems*.
- [2] John Dirk Walecka Alexander L. Fetter. *Quantum theory of many-particle systems*. Pure and Applied Physics. McGraw-Hill College, first edition edition, 1971.
- [3] Wikipedia. Binomial coefficient — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Binomial_coefficient, 2016. [Online; dostę: 23.06.2016].
- [4] Leonard I Schiff. *Quantum mechanics*. International series in pure and applied physics. McGraw-Hill Book Co, 1st edition, 1949.
- [5] Charles Kittel. *Quantum Theory of Solids*. 2nd revised edition, 1987.
- [6] A. A. Zvyagin. *Quantum Theory of One-Dimensional Spin Systems*, volume 2. Cambridge Scientific Publishers, 2010.
- [7] Anders W. Sandvik. Computational studies of quantum spin systems.
- [8] J. Bonca P. Prelovsek. Ground state and finite temperature lanczos methods. *Strongly Correlated Systems Springer Series in Solid-State Sciences Volume 176, pp 1-30*, 2013.
- [9] Wikipedia. Lanczos algorithm — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Lanczos_algorithm, 2016. [Online; dostę: 23.04.2016].
- [10] Wikipedia. Power iteration — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Power_iteration, 2016. [Online; dostę: 23.04.2016].
- [11] Wikipedia. Krylov subspace — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Krylov_subspace, 2016. [Online; dostę: 26.06.2016].
- [12] P. Prelovsek D. Crivelli, M. Mierzejewski. *Energy current and energy fluctuations in driven quantum wires*. Springer Netherlands. Nanotechnology in the Security Systems, 2015.

-
- [13] Gerald Schubert Gerhard Wellein Vladimir S. Filinov Alan R. Bishop Holger Fehske, Jens Schleede. Numerical approaches to time evolution of complex quantum systems. *Phys. Lett. A* 373, 2182, 2009.
- [14] Wikipedia. Chebyshev polynomials — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Chebyshev_polynomials, 2016. [Online; dostę: 23.04.2016].
- [15] Ryan Curtin Conrad Sanderson. Armadillo: a template-based c++ library for linear algebra. *Journal of Open Source Software*, Vol. 1, pp. 26., 2016.
- [16] Wikipedia. Euler method — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Euler_method, 2016. [Online; dostę: 23.04.2016].