

# Programowanie i metody numeryczne

## Projekt *Kwantowy chaos*

Krystian Jabłonowski  
Krystian.Jablonowski@fuw.edu.pl

## 1. Cel projektu

Celem projektu jest numeryczne zbadanie widma cząstki kwantowej zamkniętej w dwuwymiarowym obszarze o zadanym kształcie. Taki układ nazywa się *kwantowym bilardem*. Twoim zadaniem jest napisanie programu, który dla zadanej geometrii buduje dyskretny Hamiltonian, wyznacza jego wartości i wektory własne, rysuje wybrane funkcje falowe oraz analizuje statystykę odstępów między sąsiednimi poziomami energii.

Główne pytanie fizyczne brzmi: czy zmiana kształtu obszaru, po którym porusza się cząstka, wpływa na strukturę widma i funkcji falowych? W szczególności interesuje nas porównanie geometrii regularnej, takiej jak prostokąt, z geometriami, które w mechanice klasycznej prowadzą do ruchu chaotycznego, takimi jak bilard Sinai oraz stadion Bunimovicha.

Projekt ma charakter numeryczny. Odpowiedzi na pytania postawione w treści projektu powinny wynikać bezpośrednio z działania programu. Program powinien wypisywać krótkie komentarze na ekran, zapisywać dane liczbowe do plików tekstowych lub CSV oraz generować rysunki w katalogu z wynikami. Wybrane wyniki mogą być następnie omówione podczas odpowiedzi ustnej.

## 2. Model fizyczny

Rozważamy cząstkę poruszającą się wewnątrz dwuwymiarowego obszaru  $\Omega$ . Cząstka nie może opuścić tego obszaru, co odpowiada nieskończonej wysokości potencjału na brzegu. Stacjonarne równanie Schrödingera ma postać

$$-\frac{\hbar^2}{2m}\nabla^2\psi(x,y) = E\psi(x,y), \quad (1)$$

gdzie  $\psi(x,y)$  jest funkcją falową, a  $E$  jest energią. Na brzegu obszaru obowiązuje warunek Dirichleta

$$\psi(x,y) = 0, \quad (x,y) \in \partial\Omega. \quad (2)$$

W dalszej części można przyjąć jednostki  $\hbar^2/(2m) = 1$ . Wtedy Hamiltonian ma postać

$$H = -\nabla^2. \quad (3)$$

W klasycznym bilardzie cząstka porusza się po liniach prostych i odbija sprężysto od brzegu. Dla niektórych kształtów ruch klasyczny jest regularny, a dla innych chaotyczny. W mechanice

kwantowej nie analizujemy pojedynczej trajektorii. Zamiast tego badamy energie własne oraz funkcje falowe. Jednym z najprostszych śladów chaosu kwantowego jest *odpychanie poziomów*: w układach chaotycznych sąsiednie poziomy energii rzadko leżą bardzo blisko siebie.

### 3. Uwaga o zakresie i interpretacji wyników

Porównanie statystyki poziomów z przewidywaniami teorii macierzy losowych ma w tym projekcie charakter jakościowy. Nie powinniśmy oczekiwać idealnej zgodności z rozkładami granicznymi. Wynik może zależeć od rozmiaru siatki, liczby obliczonych poziomów, zakresu analizowanego widma oraz obecności symetrii geometrycznych.

Szczególnie ważna jest obecność symetrii odbicia. Jeżeli geometria ma symetrie, całe widmo może być sumą kilku niezależnych sektorów symetrii. Mieszanie takich sektorów może zniekształcić statystykę odstępów. Dlatego w analizie bilardu Sinai zaleca się, oprócz przypadku centralnego, rozważyć także przeszkodę przesuniętą względem środka prostokąta, na przykład  $c_x = 0.63L_x$ ,  $c_y = 0.47L_y$ . Taka geometria łamie symetrię odbicia i zwykle daje czytelniejszy wynik numeryczny.

### 4. Dyskretyzacja równania Schrödingera

Obszar  $\Omega$  należy umieścić na prostokątnej siatce punktów. Następnie trzeba zbudować maskę logiczną, która określa, które punkty siatki należą do obszaru dostępnego dla cząstki, a które znajdują się poza nim.

Dla równomiernej siatki o krokach  $h_x$  i  $h_y$  operator  $-\nabla^2$  można przybliżyć schematem różnicowym

$$(H\psi)_{i,j} = 2 \left( \frac{1}{h_x^2} + \frac{1}{h_y^2} \right) \psi_{i,j} - \frac{1}{h_x^2} (\psi_{i+1,j} + \psi_{i-1,j}) - \frac{1}{h_y^2} (\psi_{i,j+1} + \psi_{i,j-1}). \quad (4)$$

Jeżeli sąsiedni punkt znajduje się poza maską, należy potraktować go jako punkt brzegowy, na którym funkcja falowa jest równa zero. W praktyce najwygodniej jest numerować wyłącznie punkty należące do obszaru  $\Omega$  i budować rzadką macierz Hamiltonianu.

Program powinien rozwiązywać problem własny

$$H\phi_n = E_n\phi_n, \quad (5)$$

gdzie  $E_n$  są energiami własnymi, a  $\phi_n$  odpowiadającymi im funkcjami własnymi. Dla większych siatek należy używać metod dla macierzy rzadkich, na przykład funkcji `scipy.sparse.linalg.eigsh`. Nie należy budować pełnej macierzy gęstej dla dużych siatek.

### 5. Geometrie badanych obszarów

Program powinien pozwalać na zmianę geometrii obszaru przez podanie odpowiedniej maski. W projekcie należy uwzględnić co najmniej prostokąt oraz bilard Sinai. Stadion Bunimovicha można potraktować jako drugą geometrię chaotyczną albo jako rozszerzenie projektu.

## 5.1. Prostokąt

Najprostszym przypadkiem testowym jest prostokąt

$$\Omega_{\text{rect}} = [0, L_x] \times [0, L_y]. \quad (6)$$

Dla tego kształtu znane jest rozwiązanie analityczne. W jednostkach  $\hbar^2/(2m) = 1$  energie wynoszą

$$E_{n,m} = \pi^2 \left( \frac{n^2}{L_x^2} + \frac{m^2}{L_y^2} \right), \quad n, m = 1, 2, 3, \dots \quad (7)$$

Prostokąt służy do sprawdzenia poprawności programu. Warto wybrać taki stosunek boków  $L_x/L_y$ , który nie prowadzi do wielu degeneracji poziomów.

## 5.2. Bilard Sinai

Bilard Sinai jest prostokątem z wyciętą kołową przeszkodą. Można go opisać jako

$$\Omega_{\text{Sinai}} = [0, L_x] \times [0, L_y] \setminus \left\{ (x, y) : (x - c_x)^2 + (y - c_y)^2 < R^2 \right\}, \quad (8)$$

gdzie  $(c_x, c_y)$  jest środkiem przeszkody, a  $R$  jej promieniem. Warunek Dirichleta obowiązuje zarówno na zewnętrznym brzegu prostokąta, jak i na brzegu kołowej przeszkody.

W podstawowej wersji można przyjąć  $c_x = L_x/2$ ,  $c_y = L_y/2$ . Do analizy statystyki poziomów lepszy jest jednak wariant z przesuniętą przeszkodą, ponieważ łamie symetrię odbicia.

## 5.3. Stadion Bunimovicha

Stadion można zbudować z prostokąta oraz dwóch półkoli. Wygodny opis analityczny, dla stadionu ustawionego wzdłuż osi  $x$ , ma postać

$$\begin{aligned} \Omega_{\text{stad}}(a, R) = & \{ (x, y) : |x| \leq a, |y| \leq R \} \\ & \cup \left\{ (x, y) : (x - a)^2 + y^2 \leq R^2 \right\} \\ & \cup \left\{ (x, y) : (x + a)^2 + y^2 \leq R^2 \right\}. \end{aligned} \quad (9)$$

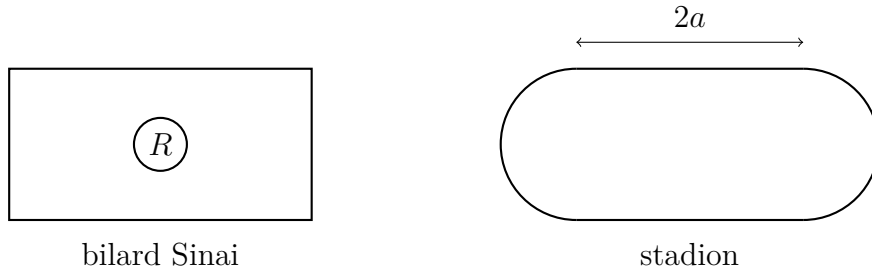
Parametr  $R$  jest promieniem półkoli, a  $2a$  jest długością prostokątnej części stadionu. Dla  $a = 0$  stadion redukuje się do koła.

## 6. Przykładowe dane wejściowe

Program może przyjmować dane wejściowe w dowolnej czytelnej formie: jako argumenty funkcji, plik konfiguracyjny albo słownik parametrów. Ważne jest, aby jednoznacznie określić rozmiar siatki, fizyczny rozmiar obszaru oraz typ geometrii.

Przykładowy słownik parametrów dla bilardu Sinai może wyglądać następująco:

```
params = {  
    "geometry": "sinai",  
    "Nx": 120,  
    "Ny": 80,
```



Rysunek 1: Dwa przykłady geometrii prowadzących w mechanice klasycznej do ruchu chaotycznego: bilard Sinai oraz stadion Bunimovicha. Rysunki są narysowane w jednym układzie współrzędnych, dzięki czemu są wyrównane względem wspólnej osi poziomej.

```
"Lx": 1.5,
"Ly": 1.0,
"cx": 0.95,    # przeszkoda przesunięta względem środka
"cy": 0.47,
"R": 0.18,
"num_eigenvalues": 120
}
```

Przykładowa funkcja tworząca maskę dla bilardu Sinai może mieć postać:

```
def mask_sinai(X, Y, Lx, Ly, cx, cy, R):
    inside_rectangle = (0 <= X) & (X <= Lx) & (0 <= Y) & (Y <= Ly)
    outside_obstacle = (X - cx)**2 + (Y - cy)**2 >= R**2
    return inside_rectangle & outside_obstacle
```

Analogicznie, maska stadionu może być zapisana jako:

```
def mask_stadium(X, Y, a, R):
    rectangle = (abs(X) <= a) & (abs(Y) <= R)
    right_cap = (X - a)**2 + Y**2 <= R**2
    left_cap = (X + a)**2 + Y**2 <= R**2
    return rectangle | right_cap | left_cap
```

## 7. Etap 1: test programu na prostokącie

Pierwszym krokiem powinno być sprawdzenie, czy dla prostokąta program odtwarza wzór analityczny (7). Należy policzyć kilka lub kilkanaście najniższych energii numerycznych i porównać je z odpowiednimi energiami analitycznymi.

Program powinien wypisać na ekran oraz zapisać do pliku tabelę zawierającą co najmniej następujące kolumny:

| numer poziomu | $E_{\text{num}}$   | $E_{\text{an}}$     | błąd względny                                    |
|---------------|--------------------|---------------------|--------------------------------------------------|
| $n$           | wartość numeryczna | wartość analityczna | $ E_{\text{num}} - E_{\text{an}} /E_{\text{an}}$ |

Należy także narysować wykres błędu kilku najniższych poziomów w funkcji kroku siatki  $h$  albo liczby punktów siatki  $N_x, N_y$ .

Na tym etapie program powinien umożliwiać odpowiedź na następujące pytania:

- Czy numeryczne wartości własne zbiegają do wartości analitycznych przy zagęszczaniu siatki?
- Jak zmienia się błąd najniższych poziomów energetycznych wraz z liczbą punktów siatki?
- Czy metoda gorzej odtwarza wyższe energie niż niższe? Odpowiedź uzasadnij.

## 8. Etap 2: funkcje falowe

Po sprawdzeniu poprawności programu należy policzyć funkcje własne dla prostokąta, bilar-du Sinai oraz, opcjonalnie, stadionu. Dla kilku wybranych poziomów energetycznych należy narysować mapy gęstości prawdopodobieństwa

$$|\phi_n(x, y)|^2. \quad (10)$$

Rysunki powinny zostać zapisane do plików, na przykład w formacie PDF lub PNG.

Program powinien wypisać na ekran krótkie informacje o każdym narysowanym stanie: numer poziomu, energię, typ geometrii oraz rozmiar siatki. Celem tego etapu jest wizualne porównanie regularnych i bardziej nieregularnych struktur funkcji falowych.

Na tym etapie program powinien umożliwiać odpowiedź na następujące pytania:

- Jak wyglądają najniższe stany w układzie prostokątnym?
- Czy w układzie z przeszkodą funkcje falowe mają bardziej nieregularną strukturę?
- Czy wynik wizualny zależy od położenia i promienia przeszkody?

## 9. Etap 3: statystyka poziomów energetycznych

Najważniejszą ilościową częścią projektu jest obliczenie średniego wskaźnika stosunku sąsiednich odstępów poziomów energetycznych  $\langle r \rangle$ . Dla jednego ustalonego kształtu należy uporządkować energie rosnąco:

$$E_1 < E_2 < \dots < E_n. \quad (11)$$

Następnie należy policzyć odstęp między kolejnymi poziomami

$$\delta_n = E_{n+1} - E_n. \quad (12)$$

Dla każdej pary sąsiednich odstępów definiujemy

$$r_n = \frac{\min(\delta_n, \delta_{n+1})}{\max(\delta_n, \delta_{n+1})}. \quad (13)$$

Jeżeli mamy  $n$  energii, to mamy  $n - 1$  odstępów oraz  $n - 2$  wartości  $r_n$ . Średnia jest równa

$$\langle r \rangle = \frac{1}{n-2} \sum_{i=1}^{n-2} r_i. \quad (14)$$

W praktyce nie należy używać bardzo małej liczby poziomów. Jeżeli pozwala na to rozmiar macierzy, warto policzyć co najmniej kilkadziesiąt, a najlepiej około stu lub więcej poziomów. Należy też sprawdzić, czy wynik nie zmienia się istotnie po odrzuceniu kilku najniższych energii. Najniższe poziomy nie zawsze dobrze reprezentują własności statystyczne widma.

## 9.1. Rozkłady referencyjne

Dla poziomów nieskorelowanych, typowych dla układów regularnych, odstępów po odpowiednim przeskalowaniu są opisywane rozkładem Poissona

$$P_P(s) = e^{-s}. \quad (15)$$

W takim przypadku średnia wartość wskaźnika  $r$  wynosi

$$\langle r \rangle_P = 2 \ln 2 - 1 \approx 0.386. \quad (16)$$

Dla układów chaotycznych z symetrią odwrócenia czasu oczekuje się statystyki zbliżonej do zespołu macierzy losowych GOE. Przybliżony rozkład odstępów, znany jako surmiza Wignera, ma postać

$$P_{WD}(s) = \frac{\pi}{2} s \exp\left(-\frac{\pi s^2}{4}\right). \quad (17)$$

Czynnik  $s$  oznacza odpychanie poziomów: prawdopodobieństwo znalezienia dwóch bardzo bliskich poziomów jest małe. Dla GOE średnia wartość wskaźnika  $r$  wynosi w przybliżeniu

$$\langle r \rangle_{GOE} \approx 0.5307. \quad (18)$$

Zaletą wskaźnika  $\langle r \rangle$  jest to, że nie wymaga pełnego rozwijania widma, czyli tak zwanego unfoldingu. Dzięki temu jest wygodny w projekcie numerycznym.

## 9.2. Zadanie numeryczne

Należy zbadać statystykę poziomów dla następujących przypadków:

- prostokąt,
- bilard Sinai z przeszkodą centralną,
- bilard Sinai z przeszkodą przesuniętą względem środka,
- opcjonalnie: stadion dla kilku wartości parametru  $a/R$ .

Dla każdego przypadku program powinien zapisać do pliku energie  $E_n$ , odstępów  $\delta_n$ , wartości  $r_n$  oraz średnią  $\langle r \rangle$ . Należy także narysować wykres  $\langle r \rangle$  w funkcji promienia przeszkody  $R$  dla bilardu Sinai. Dla stadionu można analogicznie narysować  $\langle r \rangle$  w funkcji parametru  $a/R$ .

Program powinien umożliwiać odpowiedź na następujące pytania:

- Czy prostokąt daje wartość  $\langle r \rangle$  bliższą statystyce Poissona?
- Czy bilard Sinai z przesuniętą przeszkodą daje wartość  $\langle r \rangle$  bliższą statystyce GOE niż prostokąt?
- Jak zmienia się  $\langle r \rangle$  wraz ze wzrostem promienia przeszkody w bilardzie Sinai?
- Czy wynik zależy istotnie od liczby punktów siatki i liczby uwzględnionych poziomów energetycznych?
- Czy przesunięcie przeszkody, czyli złamanie symetrii odbicia, zmienia wynik statystyki poziomów?

## 10. Etap dodatkowy: ewolucja w czasie

Ewolucja w czasie nie jest konieczna do wykonania podstawowej części projektu, ale może stanowić dobre rozszerzenie. Należy wybrać stan początkowy  $\psi(x, y; 0)$ , na przykład pakiet falowy zlokalizowany w małym obszarze, i obliczyć jego ewolucję w czasie. Jeżeli znamy energie i funkcje własne, ewolucję można zapisać jako

$$\psi(x, y; t) = \sum_n c_n e^{-iE_n t} \phi_n(x, y), \quad c_n = \int_{\Omega} \phi_n^*(x, y) \psi(x, y; 0) dx dy. \quad (19)$$

W wersji dyskretnej całkę należy zastąpić sumą po punktach siatki.

Dla kilku wybranych czasów należy narysować mapy gęstości

$$|\psi(x, y; t)|^2. \quad (20)$$

Program powinien zapisać te mapy do plików oraz wypisać krótką informację o użytych parametrach: geometrii, liczbie punktów siatki, liczbie stanów własnych użytych do ewolucji oraz czasach, dla których wykonano rysunki.

## 11. Minimalne wymagania programu

Końcowa wersja programu powinna zawierać następujące elementy:

1. funkcję tworzącą siatkę dwuwymiarową;
2. funkcję tworzącą maski dla prostokąta i bilardu Sinai;
3. opcjonalnie funkcję tworzącą maskę stadionu Bunimovicha;
4. funkcję budującą rzadką macierz Hamiltonianu metodą różnic skończonych;
5. procedurę diagonalizacji Hamiltonianu dla macierzy rzadkiej;
6. porównanie numerycznych energii prostokąta z wynikiem analitycznym;
7. rysowanie map  $|\phi_n(x, y)|^2$  dla wybranych stanów własnych;
8. obliczanie odstępów  $\delta_n$ , wartości  $r_n$  oraz średniej  $\langle r \rangle$ ;
9. zapisywanie wyników liczbowych i rysunków do plików;
10. krótkie komunikaty tekstowe wypisywane przez program, wyjaśniające, jakie parametry zostały użyte i jakie wartości  $\langle r \rangle$  otrzymano.

## 12. Sugerowany wariant minimalny i rozszerzony

Aby projekt był wykonalny, można rozdzielić go na część podstawową i rozszerzoną.

### 12.1. Część podstawowa

W części podstawowej należy wykonać następujące zadania:

1. Zbudować Hamiltonian dla prostokąta i sprawdzić zgodność z rozwiązaniem analitycznym.
2. Zbudować Hamiltonian dla bilardu Sinai.
3. Narysować kilka funkcji własnych dla prostokąta i bilardu Sinai.

4. Policzyc średnie  $\langle r \rangle$  dla prostokąta oraz dla bilardu Sinai z przesuniętą przeszkodą.
5. Sprawdzić stabilność wyniku dla co najmniej dwóch rozdzielczości siatki.

## 12.2. Część rozszerzona

W części rozszerzonej można wykonać jedno lub kilka zadań dodatkowych:

- dodać stadion Bunimovicha i zbadać zależność  $\langle r \rangle$  od  $a/R$ ;
- narysować histogram odstępów poziomów i porównać go z rozkładem Poissona oraz surmizą Wignera;
- zbadać prawo Weyla dla liczby poziomów poniżej energii  $E$ ;
- obliczyć ewolucję pakietu falowego w czasie;
- sprawdzić wpływ przesunięcia przeszkody w bilardzie Sinai na statystykę poziomów.

## 13. Uwagi numeryczne

Przy zbyt małej siatce brzeg kołowej przeszkody oraz brzeg stadionu są reprezentowane schodkowo. Może to wpływać na energie oraz statystykę poziomów. Dlatego należy sprawdzić, czy główne wnioski nie zmieniają się po zwiększeniu liczby punktów siatki.

Dla siatek rzędu  $80 \times 80$  albo  $100 \times 100$  macierz Hamiltonianu ma rozmiar rzędu kilku tysięcy, ale jest bardzo rzadka. Taki problem jest odpowiedni dla metod rzadkich. Pełna diagonalizacja macierzy gęstej byłaby niepotrzebnie kosztowna i może szybko stać się niewykonalna pamięciowo.

W analizie statystyki poziomów należy unikać przypadkowych degeneracji oraz bardzo małej liczby poziomów. Jeżeli wynik  $\langle r \rangle$  nie jest bliski ani wartości Poissona, ani wartości GOE, nie musi to oznaczać błędu programu. Może to wynikać z małej liczby poziomów, zbyt niskich energii, obecności symetrii albo zbyt grubej siatki.

## Literatura

- [1] M. V. Berry, M. Tabor, *Level clustering in the regular spectrum*, Proceedings of the Royal Society of London A **356**, 375–394 (1977).
- [2] O. Bohigas, M. J. Giannoni, C. Schmit, *Characterization of chaotic quantum spectra and universality of level fluctuation laws*, Physical Review Letters **52**, 1–4 (1984).
- [3] Y. Y. Atas, E. Bogomolny, O. Giraud, G. Roux, *Distribution of the ratio of consecutive level spacings in random matrix ensembles*, Physical Review Letters **110**, 084101 (2013).