Algorytm Grovera

Kwantowe przeszukiwanie zbiorów

Robert Nowotniak

Wydział Fizyki Technicznej, Informatyki i Matematyki Stosowanej
Politechnika Łódzka

13 listopada 2007

Plan wystąpienia

1. Informatyka Kwantowa – podstawy
2. Opis problemu (przeszukiwanie zbioru)
3. Intuicyjna interpretacja geometryczna
4. Uzasadnienie matematyczne
5. Wynik symulacji w Numerical Python

Plan wystąpienia

1. Informatyka Kwantowa – podstawy
2. Opis problemu (przeszukiwanie zbioru)
3. Intuicyjna interpretacja geometryczna
4. Uzasadnienie matematyczne
5. Wynik symulacji w Numerical Python

Informatyka Kwantowa – podstawy

Podstawową jednostką informacji w informatyce kwantowej są **kubity** (ang. *qubits*) i rejestry kwantowe (ang. *quantum registers*).

$$|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$\alpha, \beta \in \mathbb{C} \qquad |\alpha|^2 + |\beta|^2 = 1$$

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Liczby α i β nazywamy **amplitudami prawdopodobieństwa**.

W wyniku pomiaru wartości rejestru kwantowego, stan rejestru przyjmuje wartość $|0\rangle$ (*ket 0*) z prawdopodobieństwem $|\alpha|^2$ lub wartość $|1\rangle$ (*ket 1*) z prawdopodobieństwem $|\beta|^2$.

Informatyka Kwantowa – podstawy

Układ wielu kubitów tworzy **rejestr kwantowy**.

Pojedynczy kubit $|\psi\rangle \in \mathbb{C}^2$ reprezentowany jest jako para liczb zespolonych, układ dwukubitowy reprezentowany jest wektorem z przestrzeni $\mathbb{C}^4$.

Informatyka Kwantowa – podstawy

Układ wielu kubitów tworzy **rejestr kwantowy**.

Pojedynczy kubit $|\psi\rangle \in \mathbb{C}^2$ reprezentowany jest jako para liczb zespolonych, układ dwukubitowy reprezentowany jest wektorem z przestrzeni $\mathbb{C}^4$.

Stan trzykubitowego rejestru jest opisany przez wektor z przestrzeni $\mathbb{C}^8$ itd

$$|\phi\rangle = \alpha_1|000\rangle + \alpha_2|001\rangle + \ldots \alpha_8|111\rangle$$

Informatyka Kwantowa – podstawy

Układ wielu kubitów tworzy **rejestr kwantowy**.

Pojedynczy kubit $|\psi\rangle \in \mathbb{C}^2$ reprezentowany jest jako para liczb zespolonych, układ dwukubitowy reprezentowany jest wektorem z przestrzeni $\mathbb{C}^4$.

Stan trzykubitowego rejestru jest opisany przez wektor z przestrzeni $\mathbb{C}^8$ itd

$$|\phi\rangle = \alpha_1|000\rangle + \alpha_2|001\rangle + \ldots \alpha_8|111\rangle$$

Dla uproszczenia używa się także zapisu dziesiętnego $|101\rangle = |5\rangle$, $|110\rangle = |6\rangle$ itd.

Informatyka Kwantowa – podstawy

Ewolucja układu kwantowego opisywana jest przez **operatory unitarne** – reprezentowane przez macierze liczb zespolonych i spełniające warunek $UU^{\dagger} = I$.
Przykładowa bramka Hadamarda, to jedna z podstawowych bramek kwantowych:

$$H_1 = \frac{\sqrt{2}}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Informatyka Kwantowa – podstawy

Rezultat działania pewnej bramki kwantowej (reprezentowanej przez macierz unitarną) na rejestr kwantowy, obliczany jest przez **pomnożenie macierzy przez wektor stanu** rejestru.

Przykład

$$H_1|0\rangle = \frac{\sqrt{2}}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} = \frac{\sqrt{2}}{2}|0\rangle + \frac{\sqrt{2}}{2}|1\rangle$$

$$H_1|1\rangle = \frac{\sqrt{2}}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} \end{bmatrix} = \frac{\sqrt{2}}{2}|0\rangle - \frac{\sqrt{2}}{2}|1\rangle$$

Plan wystąpienia

1. Informatyka Kwantowa – podstawy
2. Opis problemu (przeszukiwanie zbioru)
3. Intuicyjna interpretacja geometryczna
4. Uzasadnienie matematyczne
5. Wynik symulacji w Numerical Python

Opis problemu — przeszukiwanie zbioru

1. Przeszukiwany jest pewien ***N*-elementowy zbiór**
2. Dokładnie **jeden obiekt** w tym zbiorze posiada **poszukiwaną cechę**
3. Elementy w zbiorze nie są w żaden sposób uporządkowane

Opis problemu — przeszukiwanie zbioru

1. Przeszukiwany jest pewien ***N*-elementowy zbiór**
2. Dokładnie **jeden obiekt** w tym zbiorze posiada **poszukiwaną cechę**
3. Elementy w zbiorze nie są w żaden sposób uporządkowane
4. Przy takich założeniach, najlepszy klasyczny algorytm, znajdujący poszukiwany element, ma złożoność $O(N)$
5. ...ponieważ w najgorszym przypadku musi „sprawdzić” $N - 1$ elementów, a w przeciętnym przypadku $\frac{N}{2}$ elementów

Opis problemu — przeszukiwanie zbioru

1. Przeszukiwany jest pewien ***N*-elementowy zbiór**
2. Dokładnie **jeden obiekt** w tym zbiorze posiada **poszukiwaną cechę**
3. Elementy w zbiorze nie są w żaden sposób uporządkowane
4. Przy takich założeniach, najlepszy klasyczny algorytm, znajdujący poszukiwany element, ma złożoność $O(N)$
5. ...ponieważ w najgorszym przypadku musi „sprawdzić” $N-1$ elementów, a w przeciętnym przypadku $\frac{N}{2}$ elementów
6. **Kwantowy algorytm Grovera realizuje to zadanie ze złożonością $O(\sqrt{N})$.**

Algorytm Grovera – podstawowe informacje

Lov K. Grover.
A fast quantum-mechanical algorithm for database search. Proceedings of the 28th Annual ACM Symposium on the Theory of Computing STOC, 212 - 219 (1996)

1. Wyszukiwanie elementu w zbiorze
   np. przeszukiwanie **nieuporządkowanej** bazy danych
2. Złożoność obliczeniowa: $O(\sqrt{N})$
3. Klasa złożoności **BQP** (ang. *bounded-error, quantum, polynomial*) — „ograniczone prawdopodobieństwo błędu w czasie wielomianowym”
4. W rejestrze kwantowym przetwarzana jest jednocześnie ***superpozycja*** wszystkich rozwiązań
5. Wykorzystuje iteracyjne „**wzmacnianie amplitudy prawdopodobieństwa**” (ang. *amplitude amplification*) poszukiwanego rozwiązania

Algorytm Grovera – podstawowe informacje II

1. Algorytm używa jednego **rejestru kwantowego** $|\phi\rangle$

Algorytm Grovera – podstawowe informacje II

1. Algorytm używa jednego **rejestru kwantowego** $|\phi\rangle$
2. **Wymiar przestrzeni stanów** rejestru $|\phi\rangle$ jest określony przez **rozmiar N przeszukiwanego zbioru**.

Algorytm Grovera – podstawowe informacje II

1. Algorytm używa jednego **rejestru kwantowego** $|\phi\rangle$
2. **Wymiar przestrzeni stanów** rejestru $|\phi\rangle$ jest określony przez **rozmiar N przeszukiwanego zbioru**.
3. Każdemu elementowi zbioru odpowiada jeden **wektor bazy** przestrzeni stanów rejestru: $|0\rangle$, $|1\rangle$, $|2\rangle$, ..., $|N-1\rangle$.

Algorytm Grovera – podstawowe informacje II

1. Algorytm używa jednego **rejestru kwantowego** $|\phi\rangle$
2. **Wymiar przestrzeni stanów** rejestru $|\phi\rangle$ jest określony przez **rozmiar N przeszukiwanego zbioru**.
3. Każdemu elementowi zbioru odpowiada jeden **wektor bazy** przestrzeni stanów rejestru: $|0\rangle$, $|1\rangle$, $|2\rangle$, ..., $|N-1\rangle$.
4. Początkowy stan rejestru $|\phi_0\rangle$ jest ustawiany na „**równomierną superpozycję**” wszystkich możliwych rozwiązań

Algorytm Grovera – podstawowe informacje II

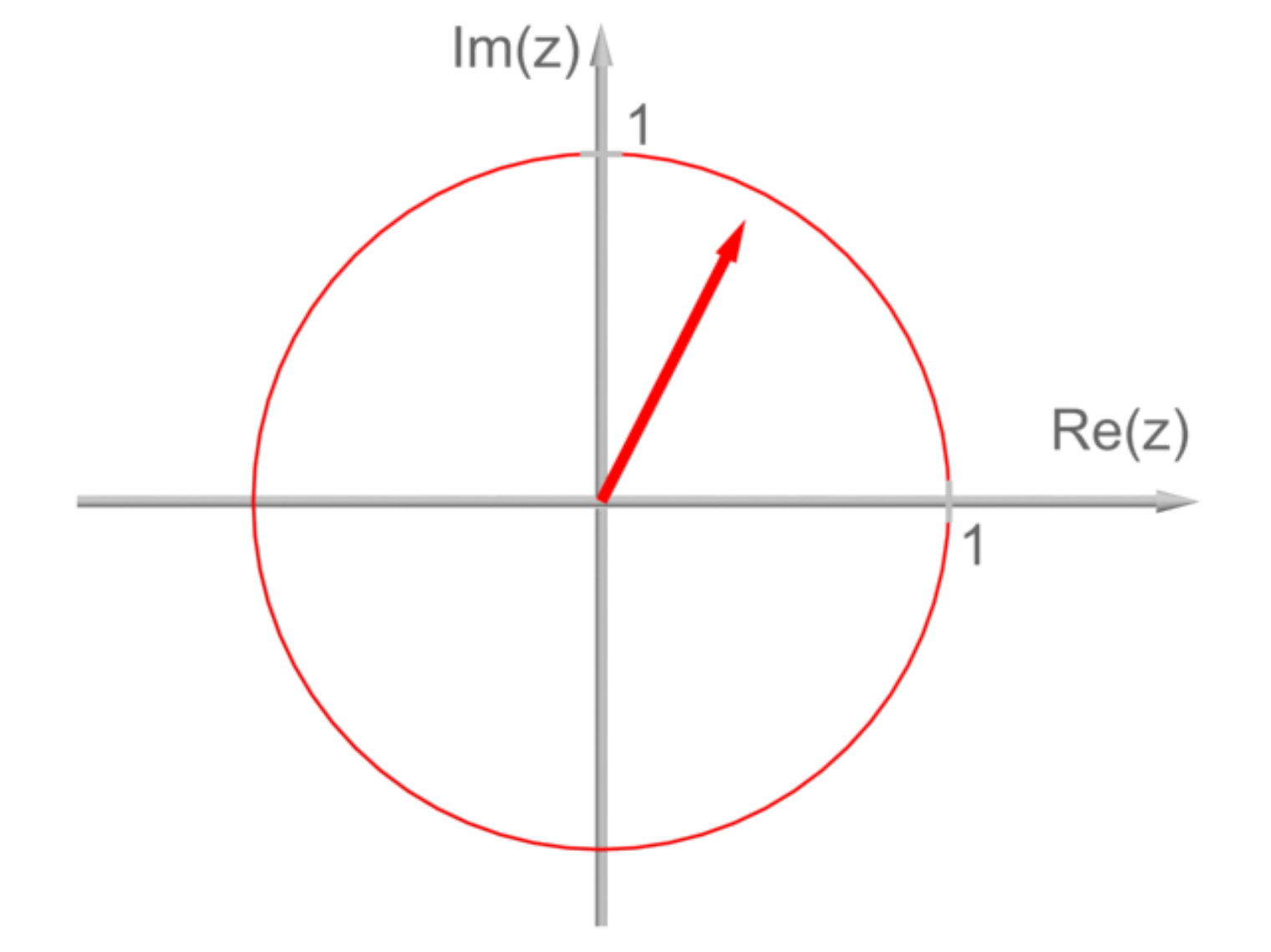
1. Algorytm używa jednego **rejestru kwantowego** $|\phi\rangle$
2. **Wymiar przestrzeni stanów** rejestru $|\phi\rangle$ jest określony przez **rozmiar N przeszukiwanego zbioru**.
3. Każdemu elementowi zbioru odpowiada jeden **wektor bazy** przestrzeni stanów rejestru: $|0\rangle$, $|1\rangle$, $|2\rangle$, ..., $|N-1\rangle$.
4. Początkowy stan rejestru $|\phi_0\rangle$ jest ustawiany na „**równomierną superpozycję**” wszystkich możliwych rozwiązań
5. W kolejnych iteracjach algorytmu wykonywane są cyklicznie dwie „procedury kwantowe”, reprezentowane przez operatory $\mathcal{A}$ i $\mathcal{B}$.

$$|\phi_{n+1}\rangle = \mathcal{B}\mathcal{A}|\phi_n\rangle$$

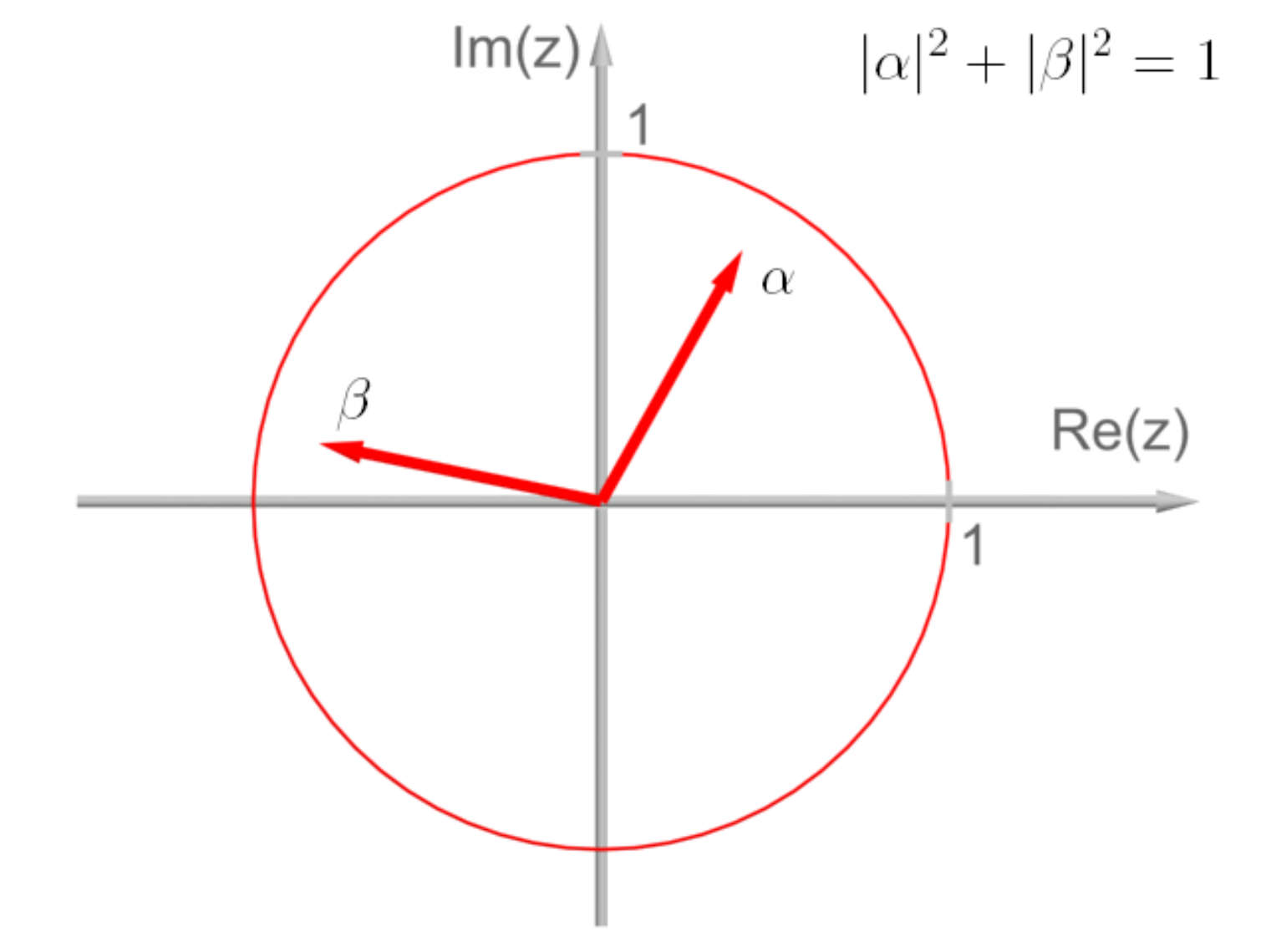
Plan wystąpienia

1. Informatyka Kwantowa – podstawy
2. Opis problemu (przeszukiwanie zbioru)
3. Intuicyjna interpretacja geometryczna
4. Uzasadnienie matematyczne
5. Wynik symulacji w Numerical Python

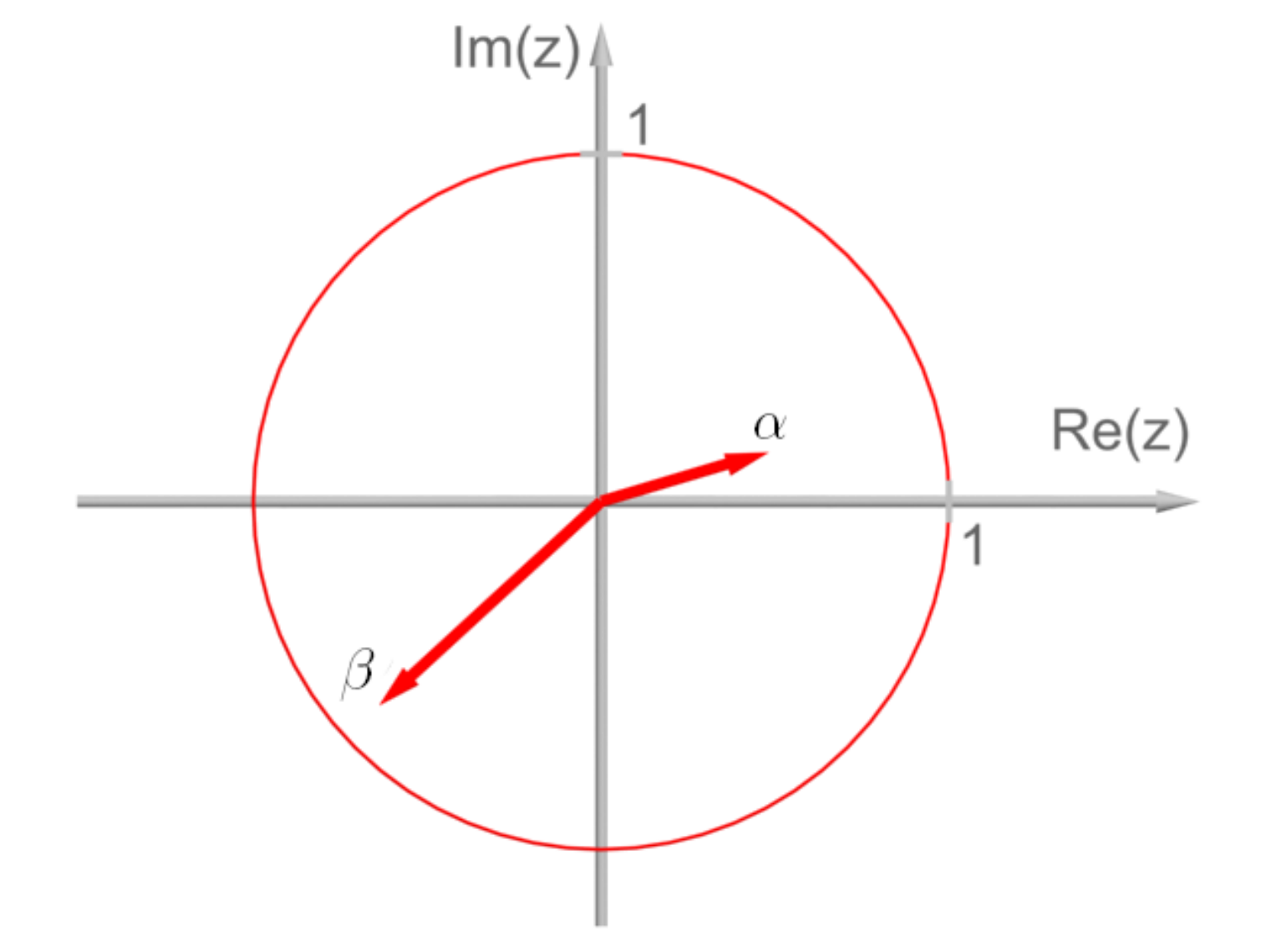
Amplituda prawdopodobieństwa



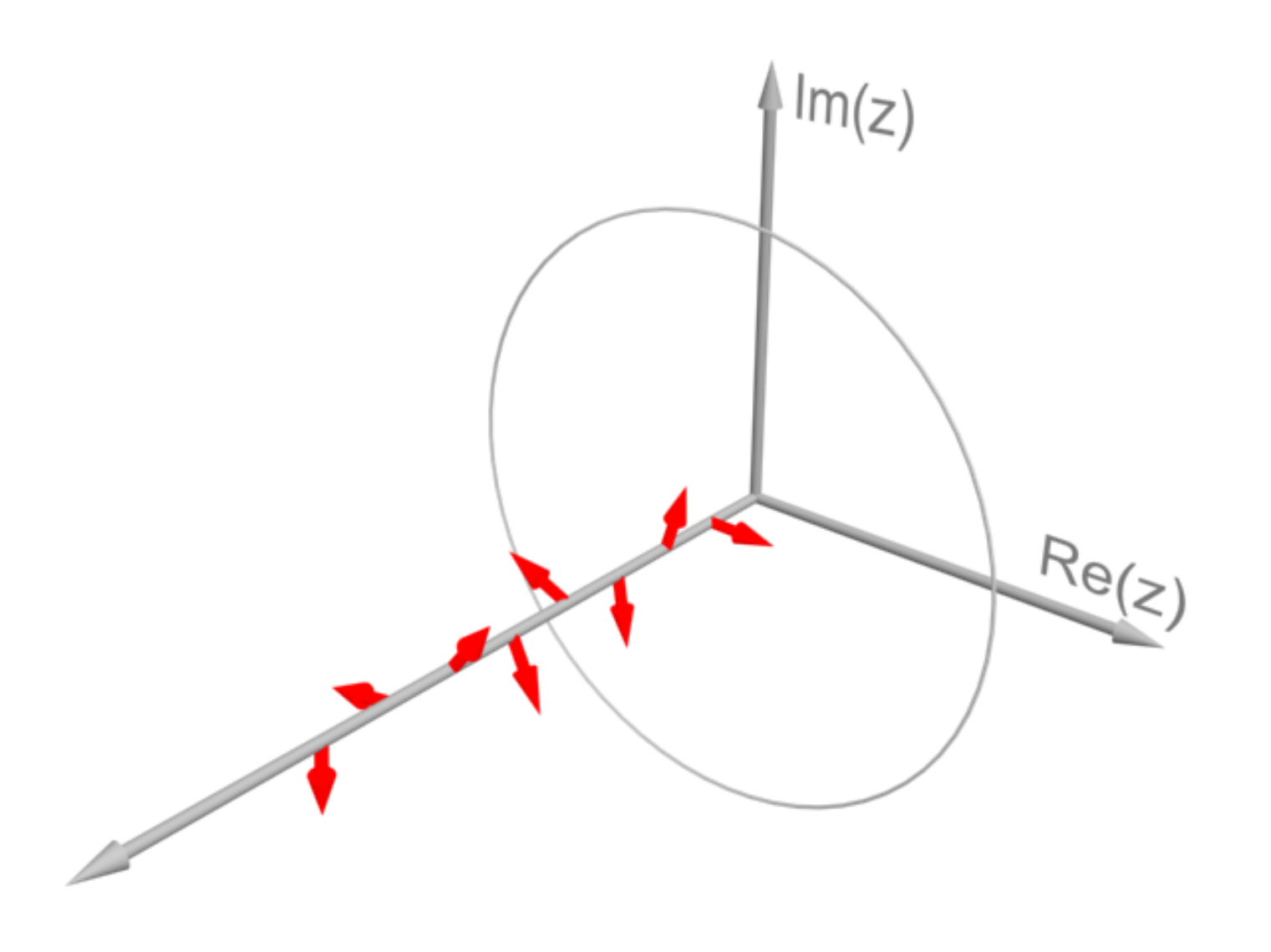
Reprezentacja kubitu na płaszczyźnie zespolonej



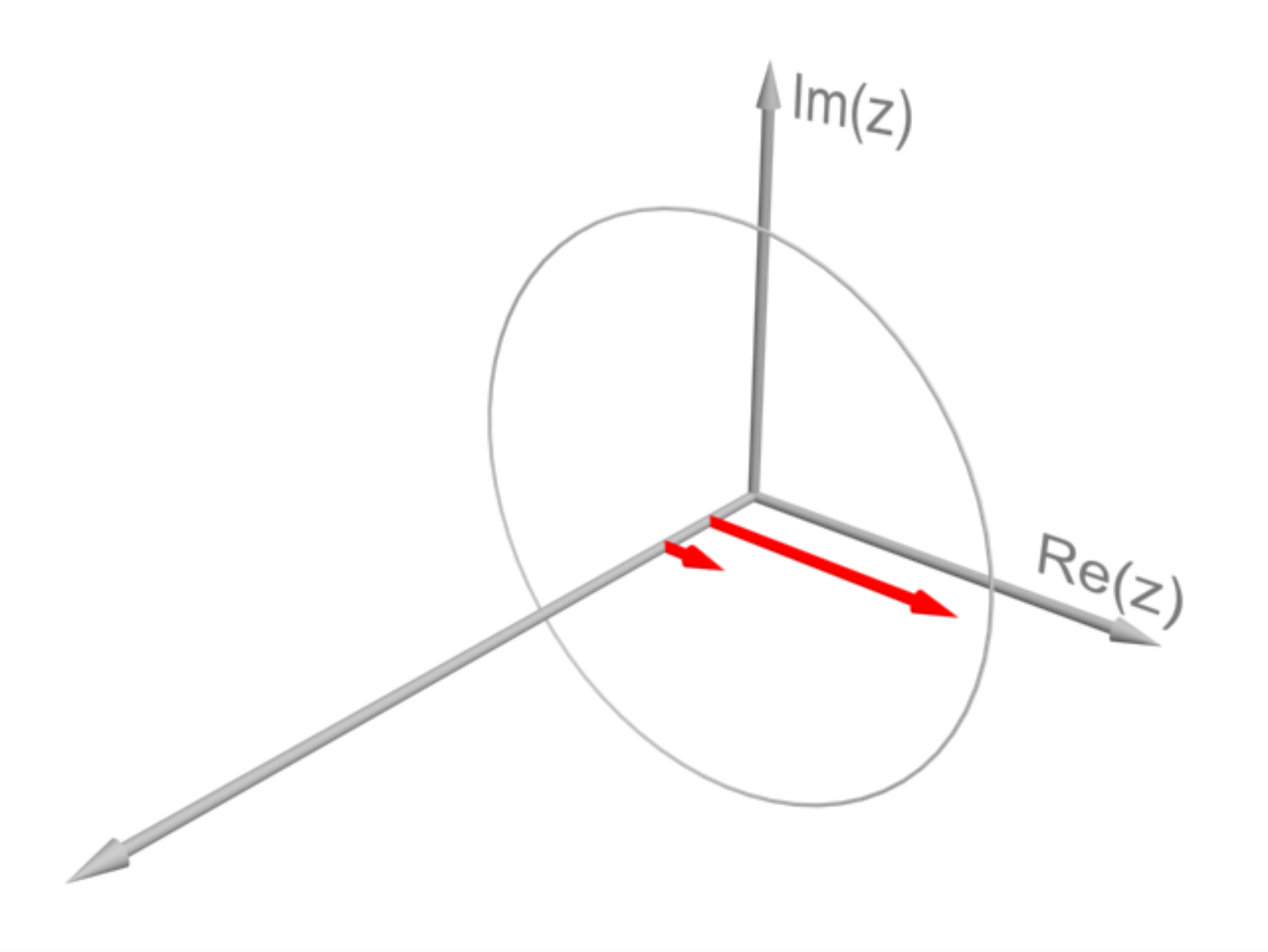
Reprezentacja kubitu na płaszczyźnie zespolonej



Reprezentacja 3-kubitowego rejestru

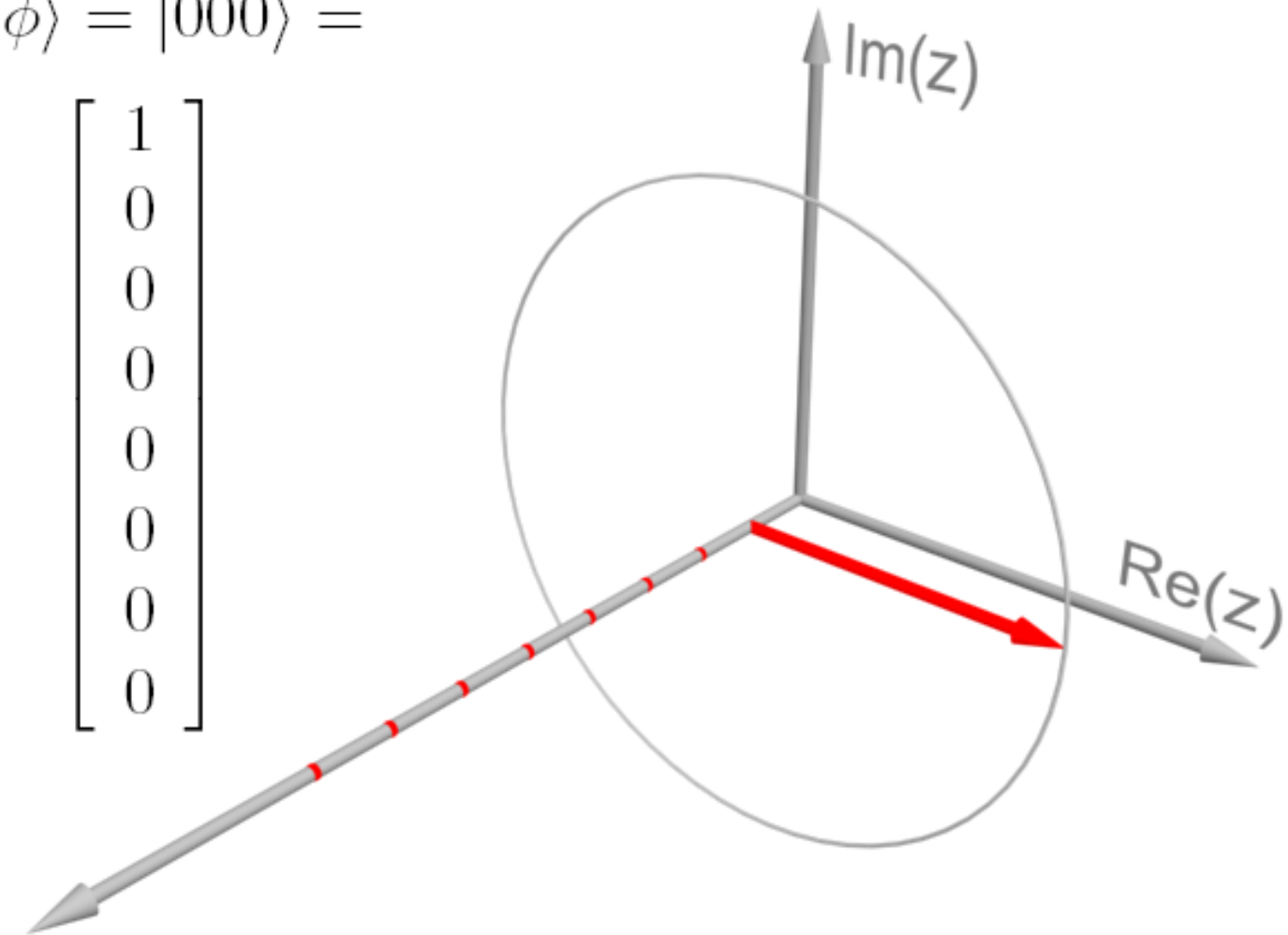


Reprezentacja rejestru kwantowego

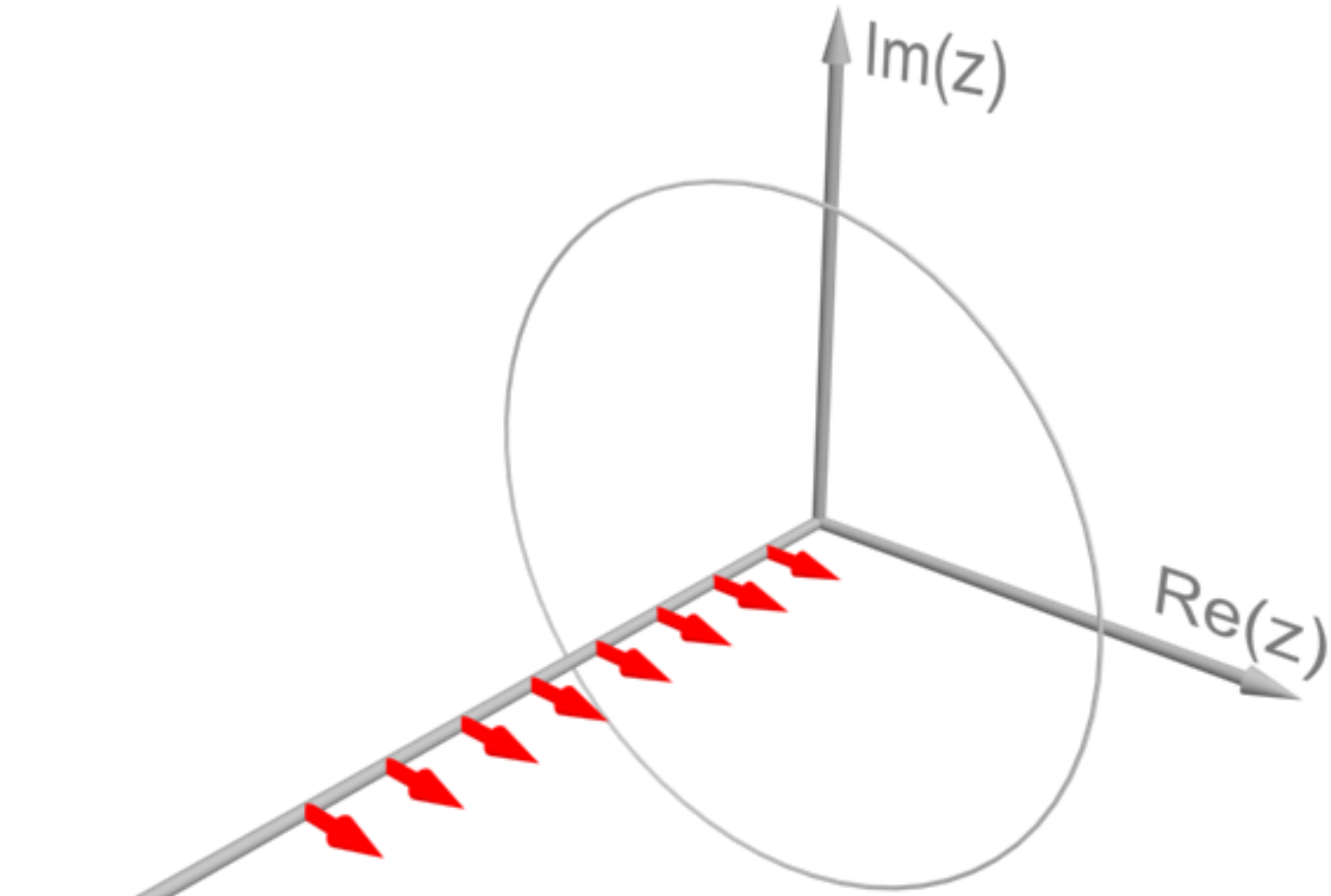


Reprezentacja stanu $|000\rangle$

$$|\phi\rangle = |000\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

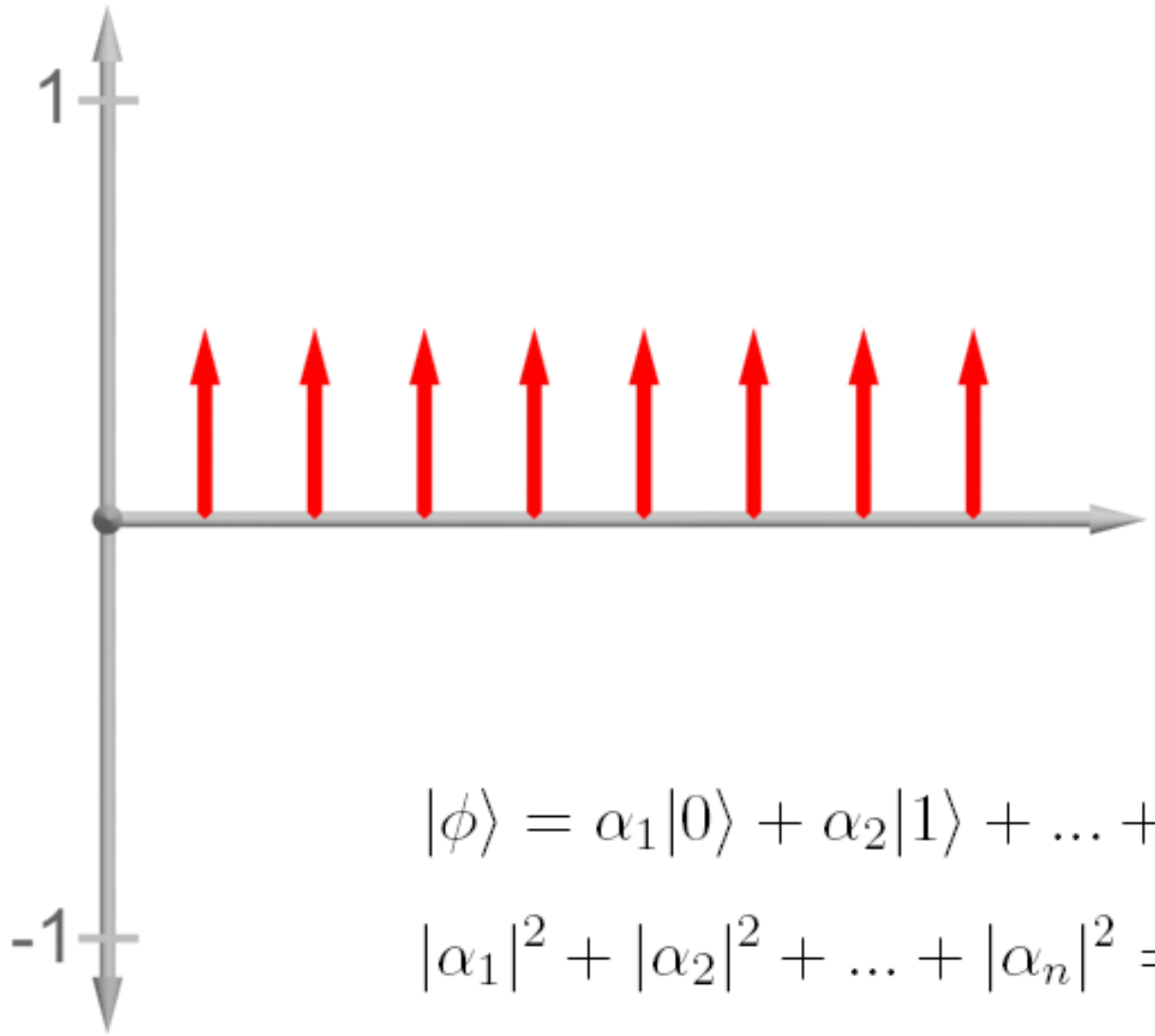


Rejestr w stanie superpozycji



$$H|000\rangle = \frac{1}{\sqrt{8}}|0\rangle + \frac{1}{\sqrt{8}}|1\rangle + \ldots + \frac{1}{\sqrt{8}}|7\rangle$$

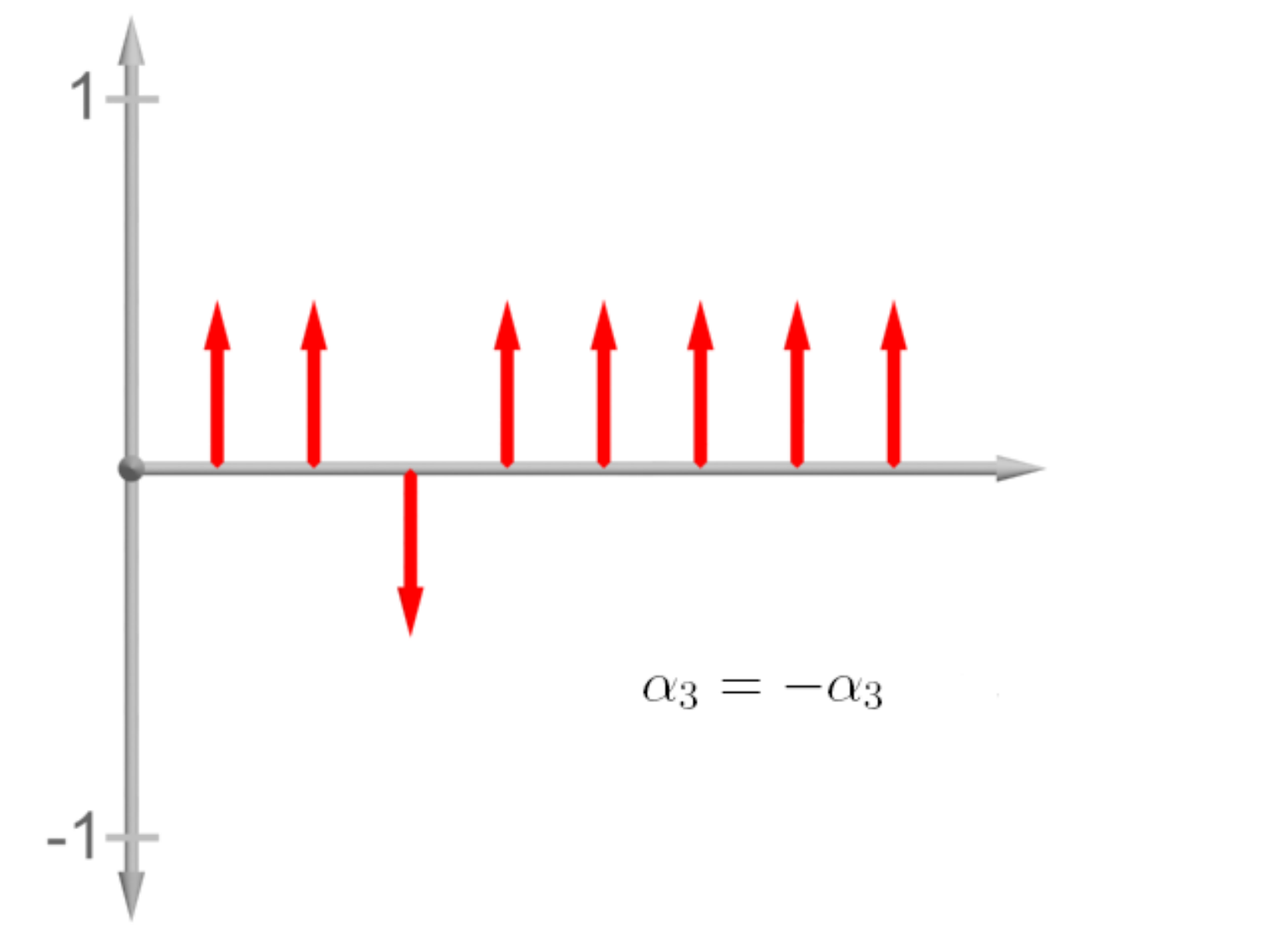
Początkowa superpozycja stanów



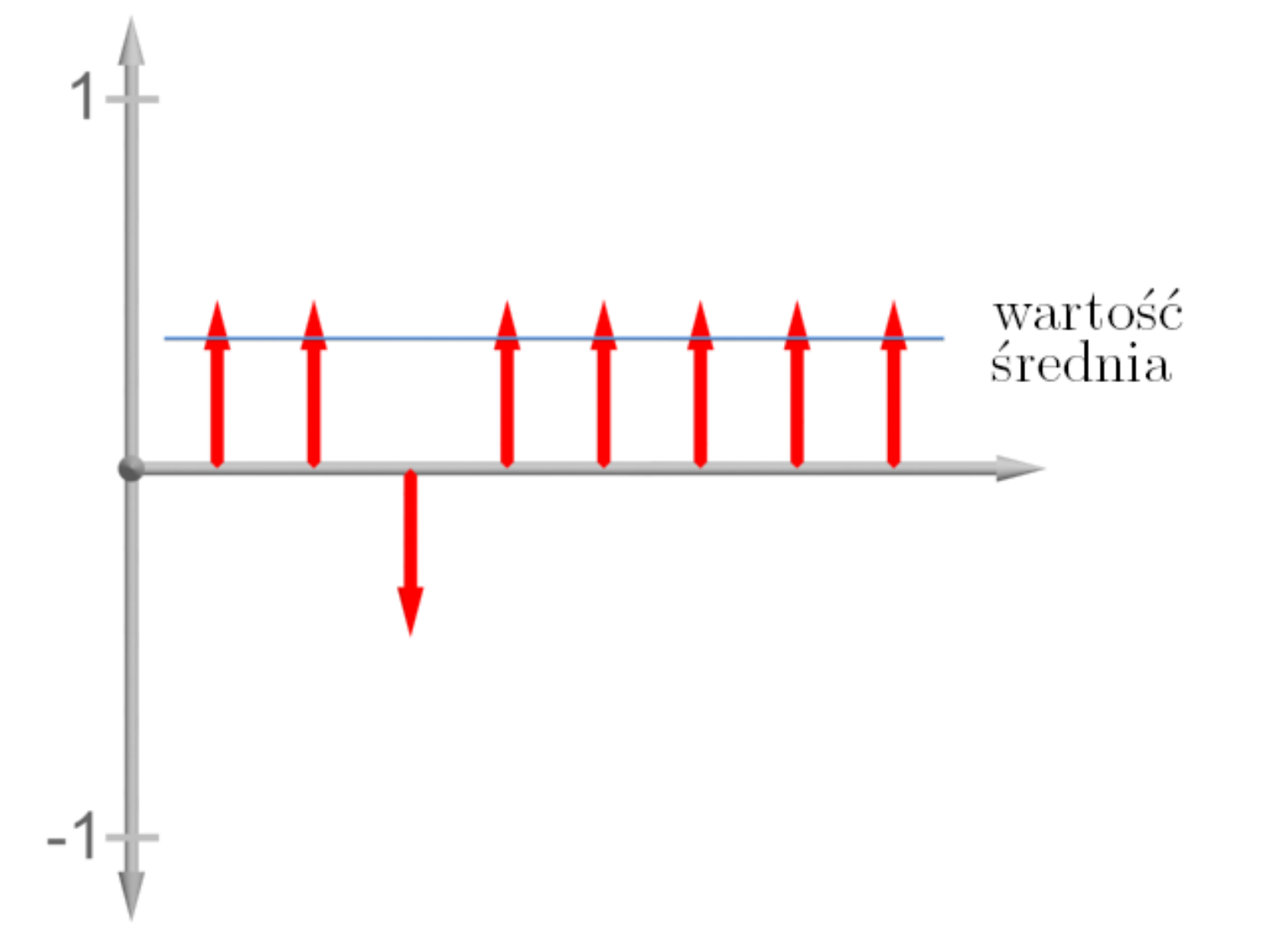
$$|\phi\rangle = \alpha_1|0\rangle + \alpha_2|1\rangle + ... + \alpha_8|7\rangle$$

$$|\alpha_1|^2 + |\alpha_2|^2 + ... + |\alpha_n|^2 = 1$$

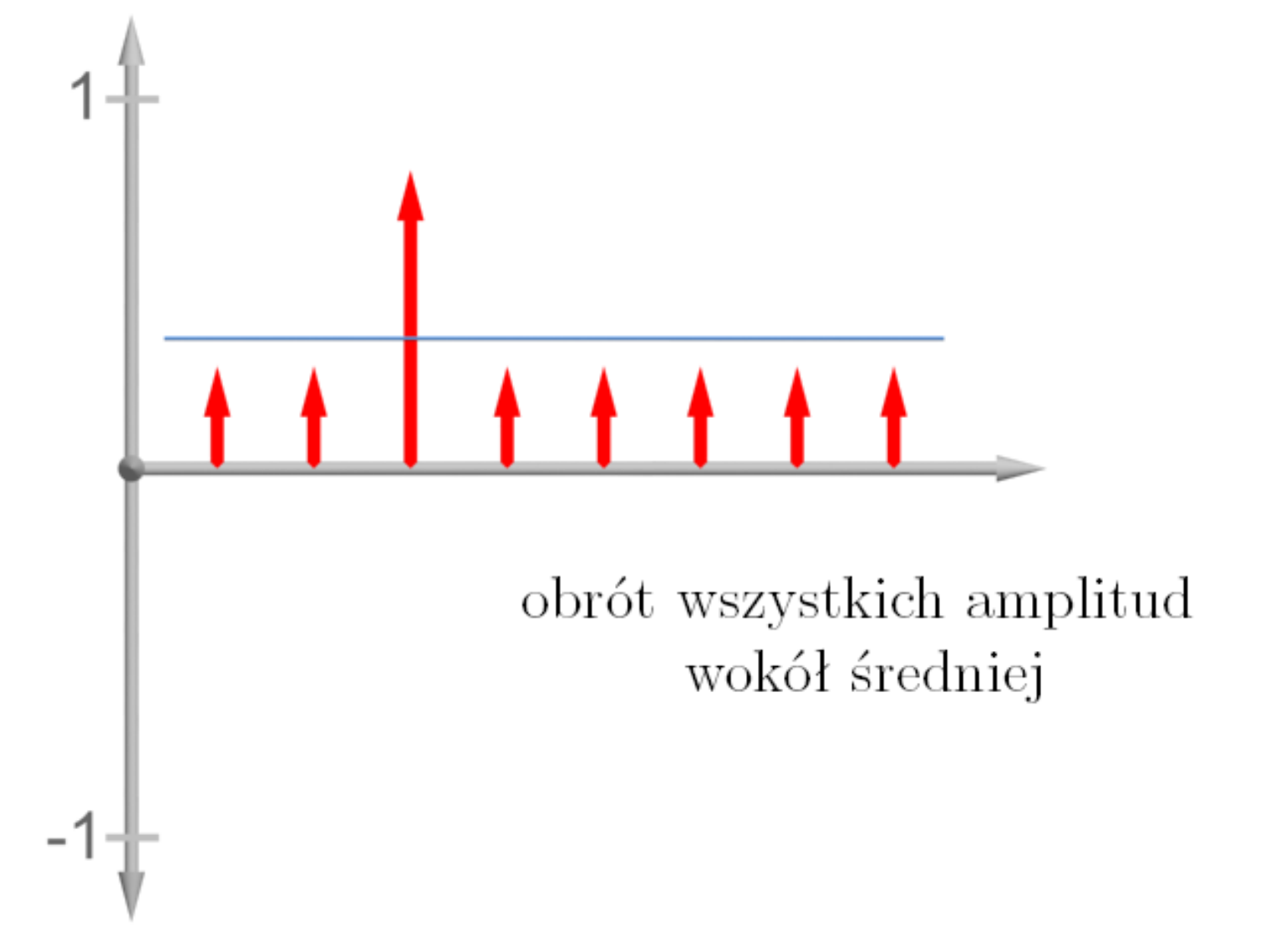
Operator $\mathbb{A}$: selektywny obrót amplitudy



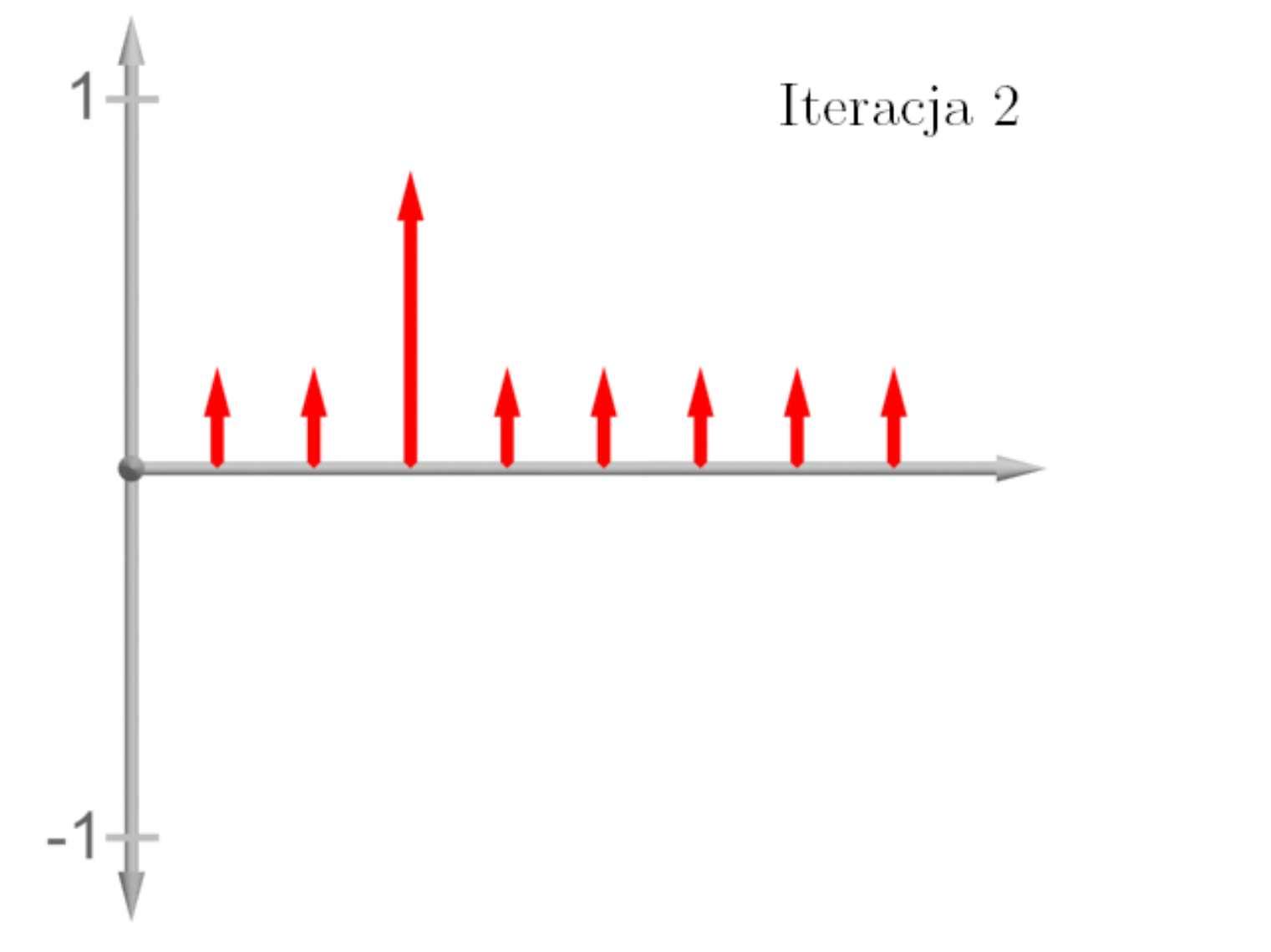
Operator $\mathbb{B}$: obrót wokół średniej



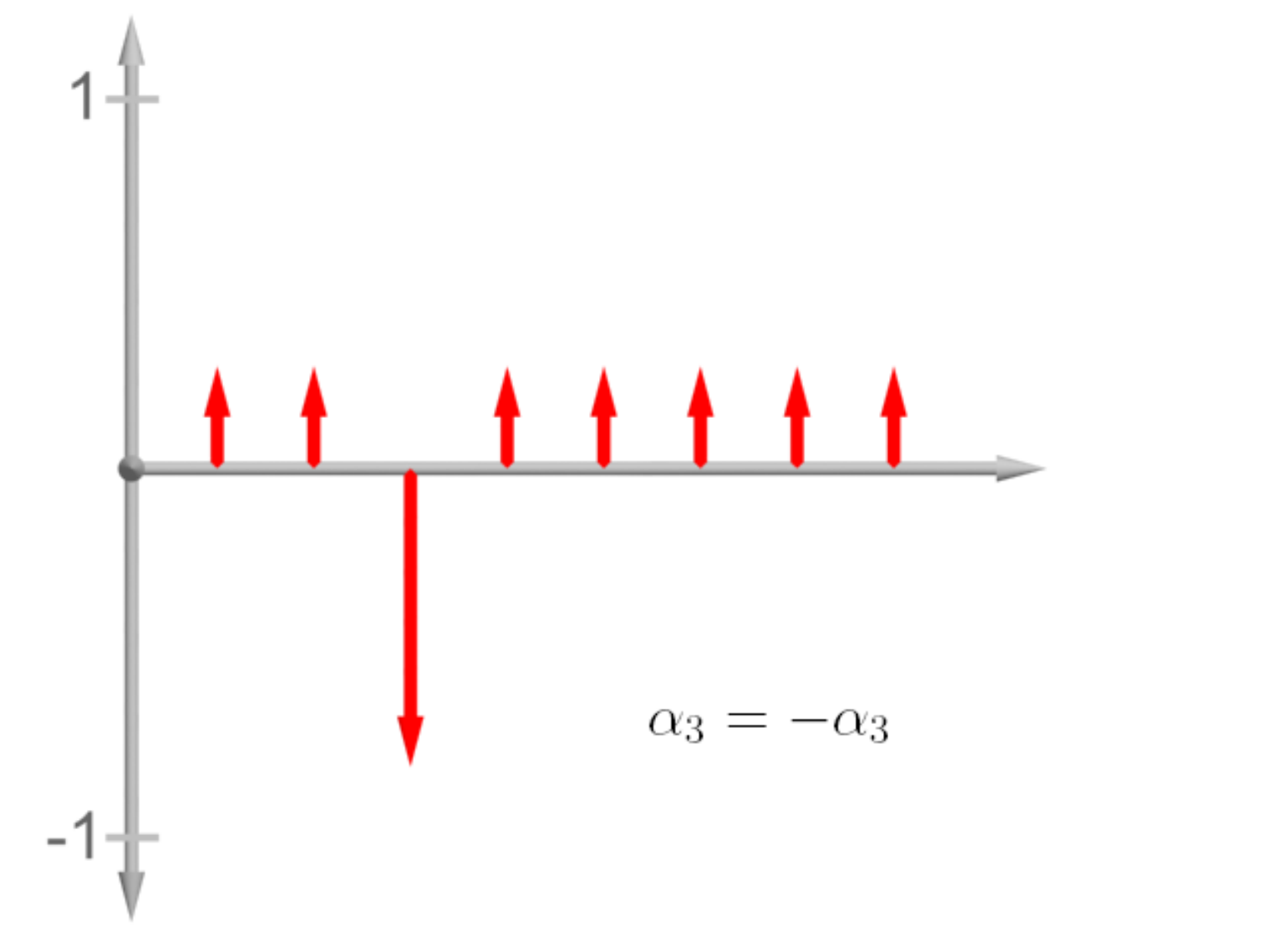
Operator $\mathbb{B}$: obrót wokół średniej



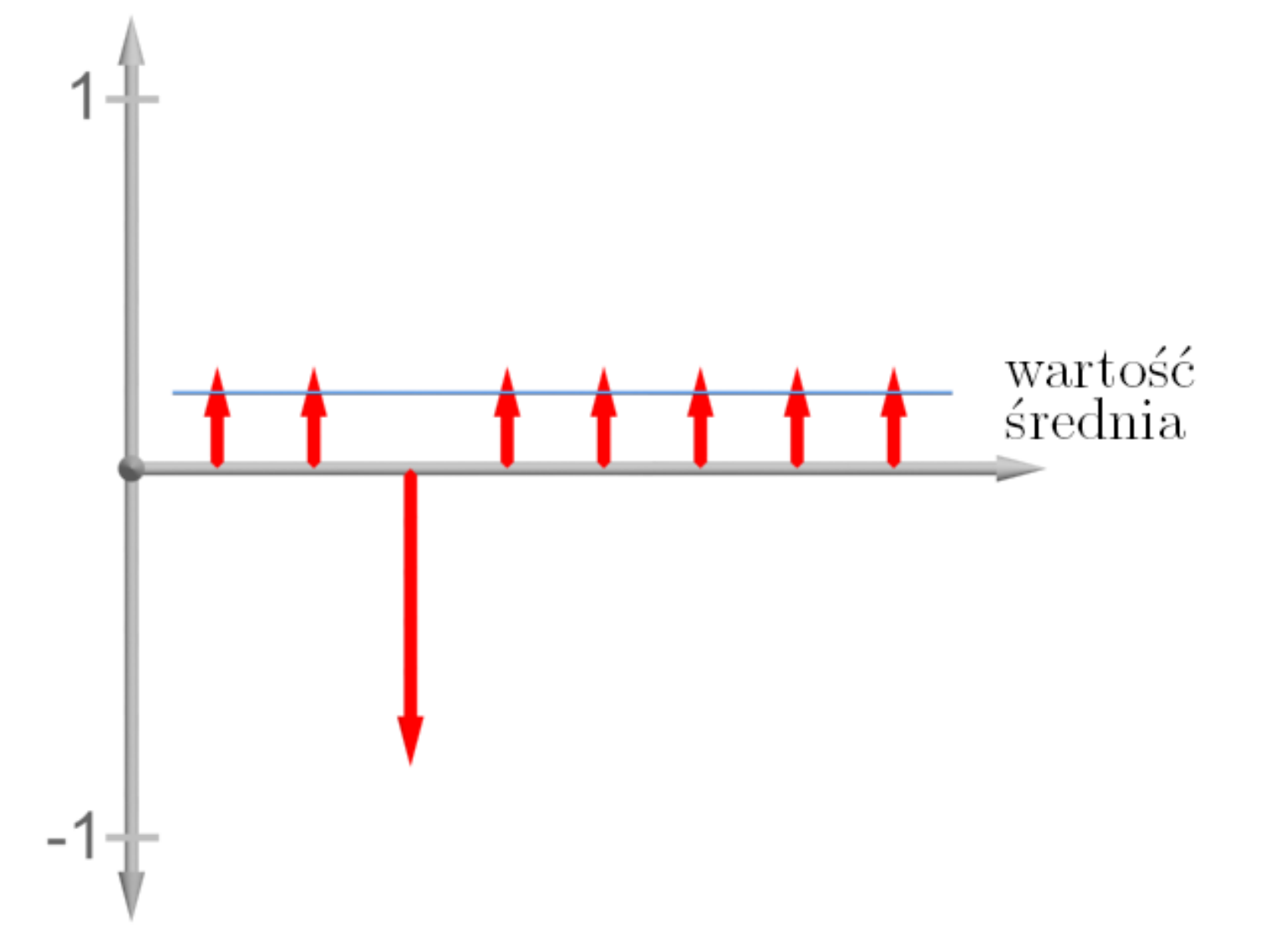
Stan po pierwszej iteracji



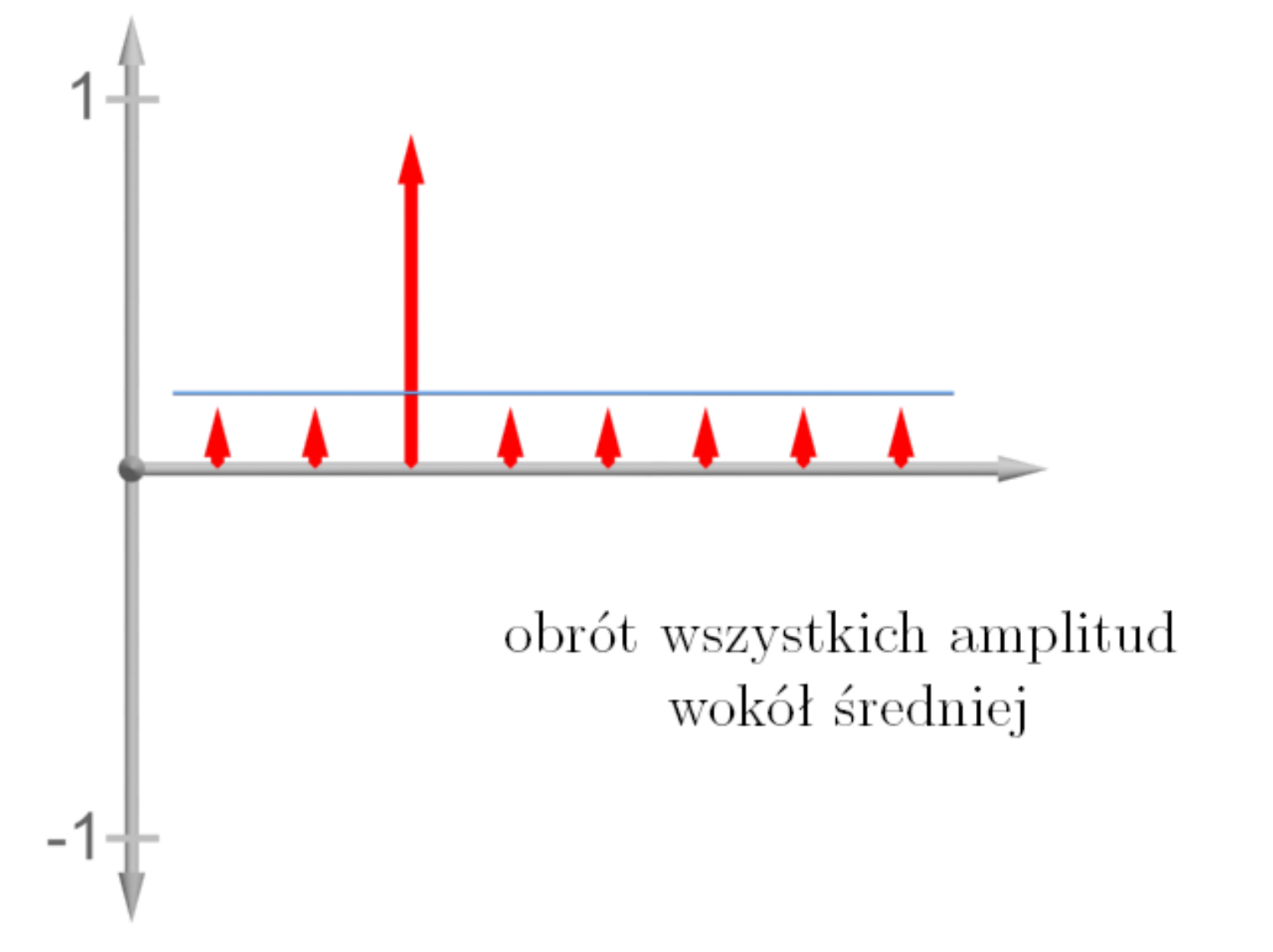
Operator $\mathbb{A}$: selektywny obrót amplitudy



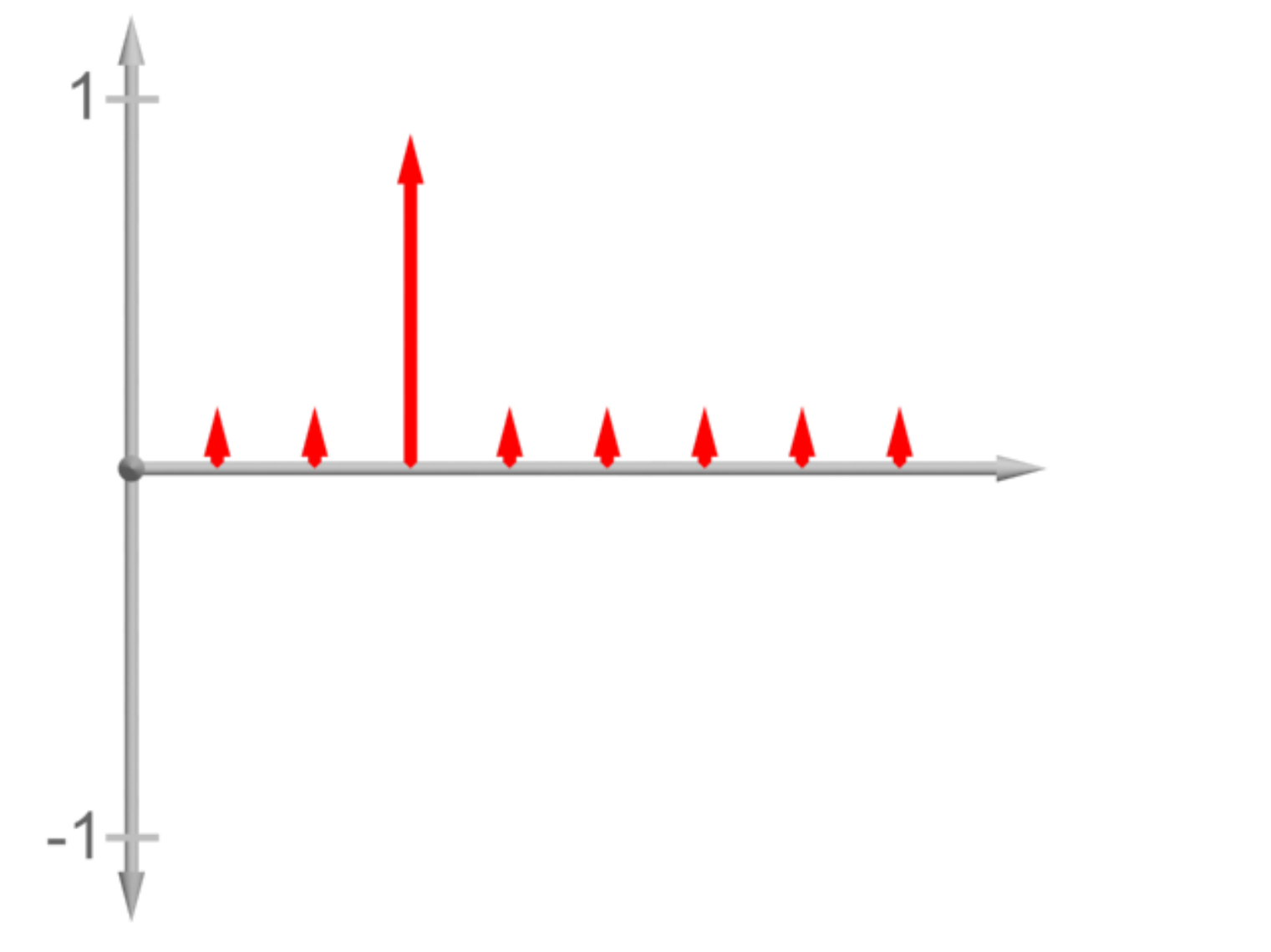
Operator $\mathbb{B}$: obrót wokół średniej



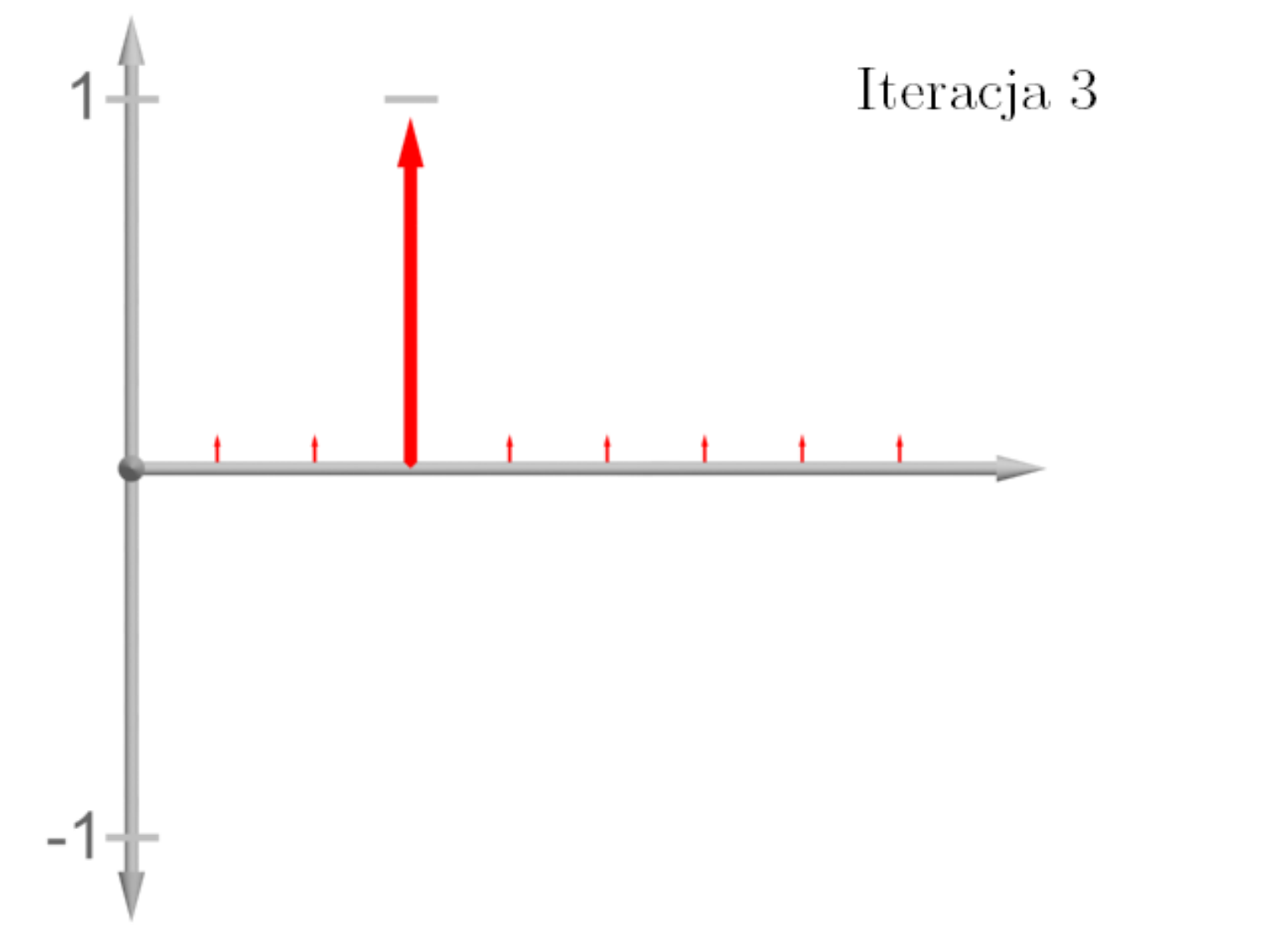
Operator $\mathbb{B}$: obrót wokół średniej



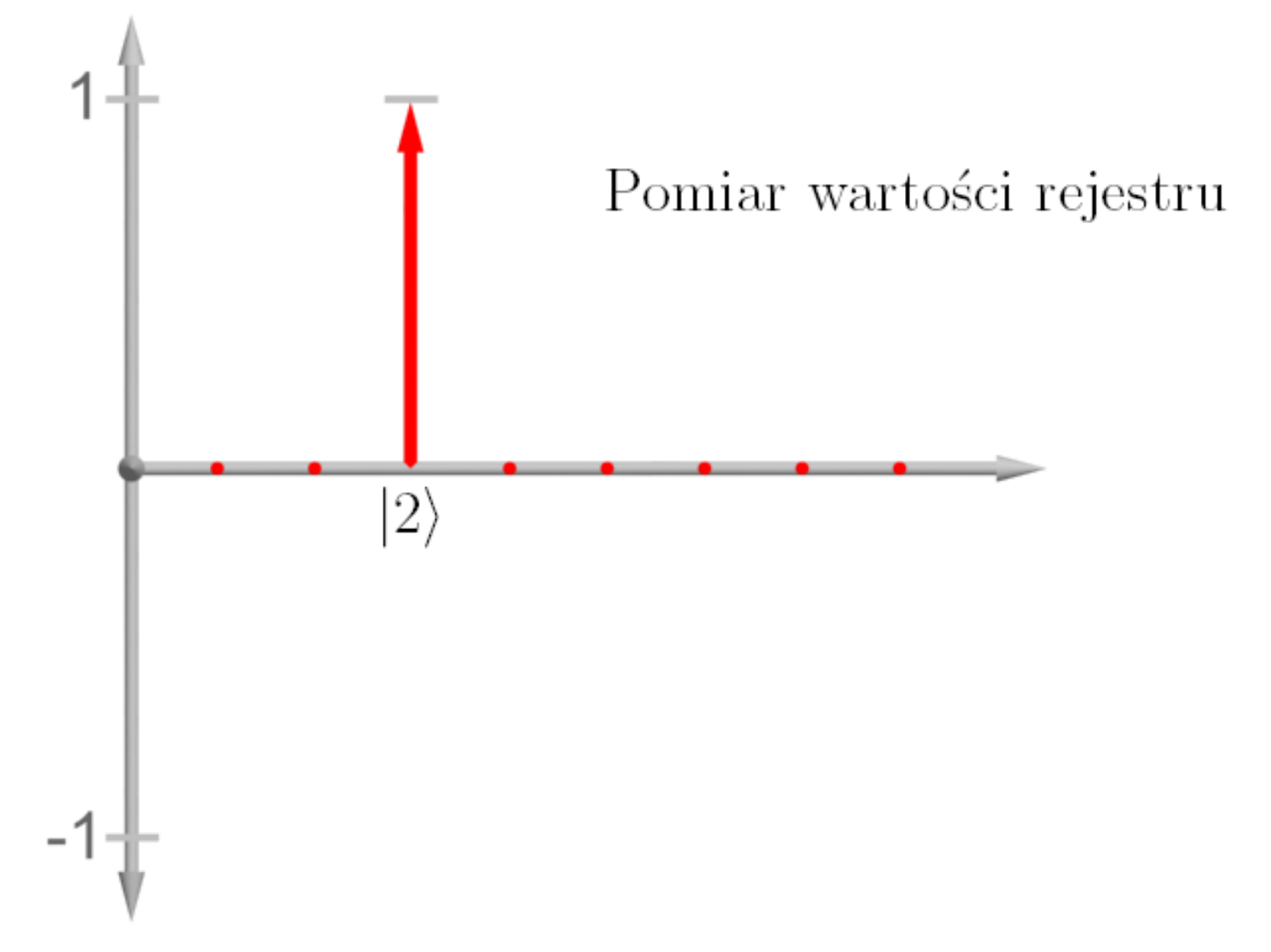
Stan po dwóch iteracjach



Stan po trzech iteracjach



Stan po operacji *pomiaru*



Prawdopodobieństwo sukcesu

Po wykonaniu k iteracji prawdopodobieństwo odczytu wartości związanej z **szukanym elementem** wyraża się wzorem:

$$|\alpha_k|^2 = sin^2\left((2k+1)arcsin\left(\sqrt{\frac{1}{N}}\right)\right)$$

Prawdopodobieństwo sukcesu

Po wykonaniu k iteracji prawdopodobieństwo odczytu wartości związanej z **szukanym elementem** wyraża się wzorem:

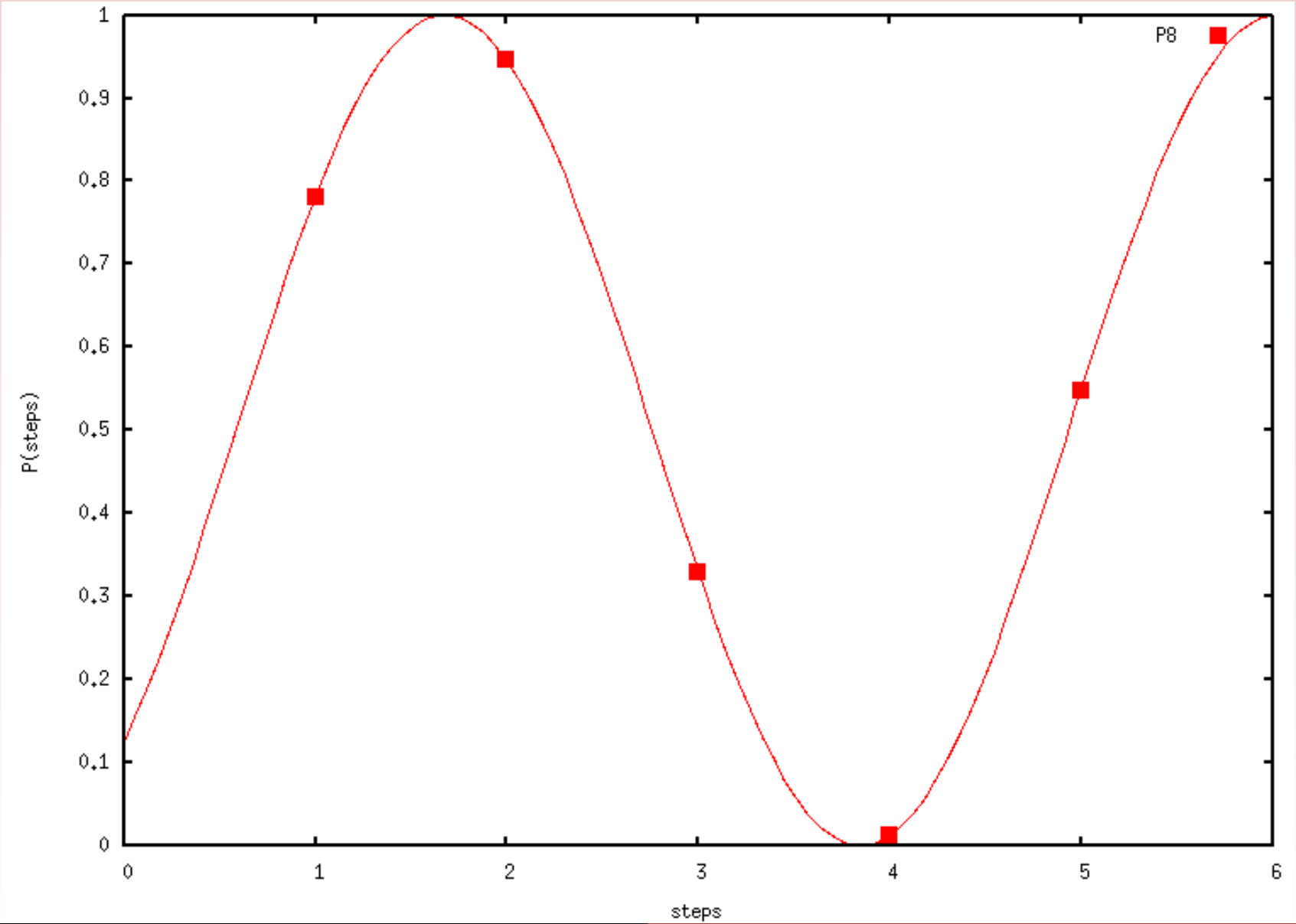
$$|\alpha_k|^2 = sin^2\left((2k+1)arcsin\left(\sqrt{\frac{1}{N}}\right)\right)$$

Maksymalne prawdopodobieństwo otrzymania właściwego stanu $|\omega_0\rangle$ jest dla dla liczby kroków $\bar{k}$ równej:

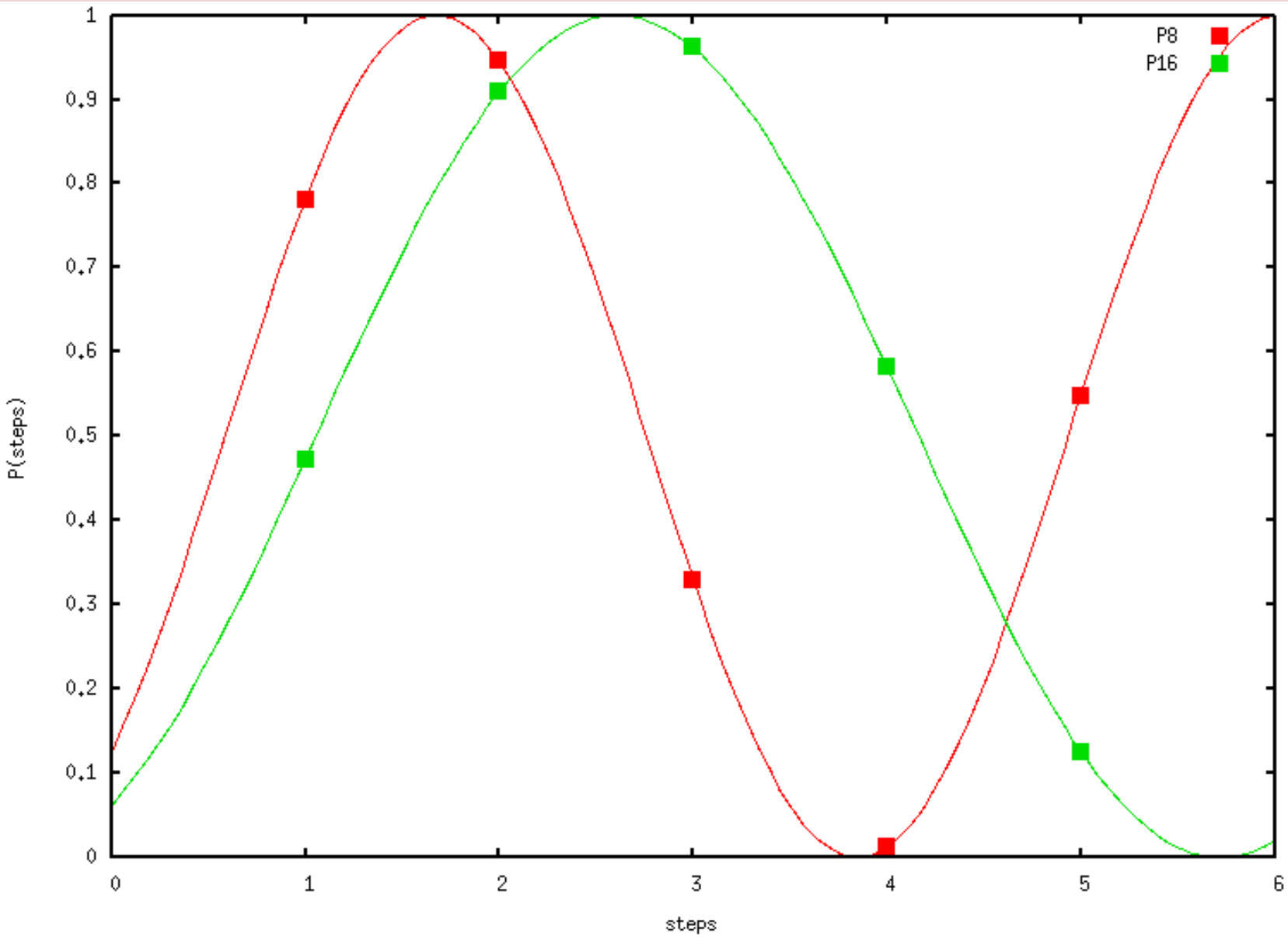
$$\bar{k} = \left\lfloor \frac{\pi}{4}\left(arcsin\left(\sqrt{\frac{1}{N}}\right)\right)^{-1} \right\rfloor$$

Prawdopodobieństwo to jest wówczas większe od $1 - \frac{1}{N}$

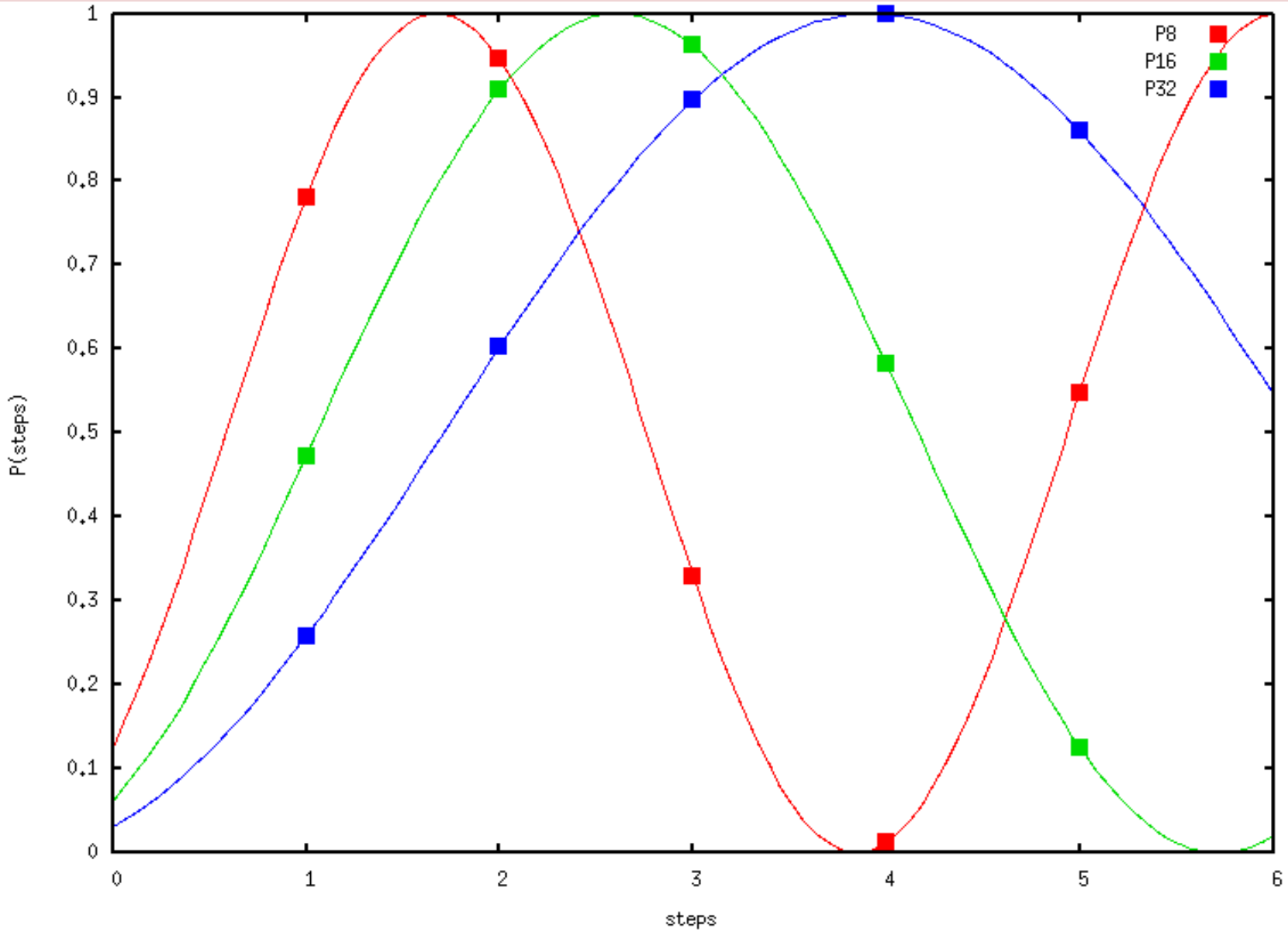
Prawdopodobieństwo w kolejnych krokach



Prawdopodobieństwo w kolejnych krokach



Prawdopodobieństwo w kolejnych krokach



Prawdopodobieństwo w kolejnych krokach

Przeszukiwanie **czterokrotnie** większego zbioru wymaga jedynie **dwukrotnie** większej liczby operacji.

Prawdopodobieństwo w kolejnych krokach

Przeszukiwanie **czterokrotnie** większego zbioru wymaga jedynie **dwukrotnie** większej liczby operacji.
(złożoność obliczeniowa $O(\sqrt{N})$)

Plan wystąpienia

1. Informatyka Kwantowa – podstawy
2. Opis problemu (przeszukiwanie zbioru)
3. Intuicyjna interpretacja geometryczna
4. Uzasadnienie matematyczne
5. Wynik symulacji w Numerical Python

Algorytm Grovera

Algorytm polega na iteracyjnym wykonywaniu operacji:

$$|\phi_{n+1}\rangle = \mathcal{B}\mathcal{A}|\phi_n\rangle$$

Algorytm Grovera

Algorytm polega na iteracyjnym wykonywaniu operacji:

$$|\phi_{n+1}\rangle = \mathcal{B}\mathcal{A}|\phi_n\rangle$$

Operatory $\mathcal{A}$ i $\mathcal{B}$ mają postać:

$$\mathcal{A} = \mathbb{I} - 2|\omega_0\rangle\langle\omega_0|$$

$$\mathcal{B} = 2|\phi_0\rangle\langle\phi_0| - \mathbb{I}$$

Operator $\mathcal{A}$ pełni rolę „procedury kwantowej”, sprawdzającej – jednocześnie – wszystkie elementy i zmieniającej znak amplitudy poszukiwanego elementu.

Uzasadnienie algorytmu

Jak działa operator $\mathcal{A}$ na dowolny stan **bazowy** $|\omega\rangle$?

Uzasadnienie algorytmu

Jak działa operator $\mathcal{A}$ na dowolny stan **bazowy** $|\omega\rangle$?

Działanie operatora $\mathcal{A}$

$$\mathcal{A}|\omega\rangle = (\mathbb{I} - 2|\omega_0\rangle\langle\omega_0|) \cdot |\omega\rangle$$

Uzasadnienie algorytmu

Jak działa operator $\mathcal{A}$ na dowolny stan **bazowy** $|\omega\rangle$?

Działanie operatora $\mathcal{A}$

$$\begin{aligned} \mathcal{A}|\omega\rangle &= (\mathbb{I} - 2|\omega_0\rangle\langle\omega_0|) \cdot |\omega\rangle \\ &= |\omega\rangle - 2|\omega_0\rangle \cdot \begin{cases} 1 & \text{gdy } \omega = \omega_0 \\ 0 & \text{gdy } \omega \neq \omega_0 \end{cases} \end{aligned}$$

Uzasadnienie algorytmu

Jak działa operator $\mathcal{A}$ na dowolny stan **bazowy** $|\omega\rangle$?

Działanie operatora $\mathcal{A}$

$$\begin{aligned}
\mathcal{A}|\omega\rangle &= (\mathbb{I} - 2|\omega_0\rangle\langle\omega_0|) \cdot |\omega\rangle \\
&= |\omega\rangle - 2|\omega_0\rangle \cdot \begin{cases} 1 & \text{gdy } \omega = \omega_0 \\ 0 & \text{gdy } \omega \neq \omega_0 \end{cases} \\
&= \begin{cases} -|\omega\rangle & \text{gdy } \omega = \omega_0 \\ |\omega\rangle & \text{gdy } \omega \neq \omega_0 \end{cases}
\end{aligned}$$

Uzasadnienie algorytmu

Jak działa operator $\mathcal{B}$ na **dowolny** stan $|\phi\rangle$?

$$\begin{aligned} \mathcal{B}|\phi\rangle &= \dots \\ &= \sum_{\omega=0}^{N-1} \left[\alpha + (\alpha - \alpha_\omega)\right] |\omega\rangle \end{aligned}$$

Uzasadnienie algorytmu

Jak działa operator $\mathcal{B}$ na **dowolny** stan $|\phi\rangle$?

$$\begin{aligned}
\mathcal{B}|\phi\rangle &= \mathcal{B}\left(\sum_{\omega=0}^{N-1}\alpha_\omega|\omega\rangle\right) = \sum_{\omega=0}^{N-1}\alpha_\omega\mathcal{B}|\omega\rangle = \sum_{\omega=0}^{N-1}\alpha_\omega\left(\frac{2}{\sqrt{N}}|\phi_0\rangle - |\omega\rangle\right) \\
&= \frac{2}{\sqrt{N}}\sum_{\omega=0}^{N-1}\alpha_\omega|\phi_0\rangle - \sum_{\omega=0}^{N-1}\alpha_\omega|\omega\rangle \\
&= \frac{2}{\sqrt{N}}\sum_{\omega=0}^{N-1}\alpha_\omega\left(\frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}|k\rangle\right) - \sum_{\omega=0}^{N-1}\alpha_\omega|\omega\rangle \\
&= 2\cdot\underbrace{\frac{1}{N}\sum_{\omega=0}^{N-1}\alpha_\omega}_{\alpha}\sum_{k=0}^{N-1}|k\rangle - \sum_{\omega=0}^{N-1}\alpha_\omega|\omega\rangle \\
&= \sum_{\omega=0}^{N-1}(2\alpha-\alpha_\omega)|\omega\rangle = \sum_{\omega=0}^{N-1}\left[\alpha+(\alpha-\alpha_\omega)\right]|\omega\rangle
\end{aligned}$$

Uzasadnienie algorytmu

Podsumowując:

- **Operator** $\mathbb{A}$ — obrót amplitudy prawdopodobieństwa związanej z poszukiwanym elementem
- **Operator** $\mathbb{B}$ — obrót wszystkich amplitud wokół wartości średniej

Uzasadnienie algorytmu

Podsumowując:

- **Operator** $\mathbb{A}$ — obrót amplitudy prawdopodobieństwa związanej z poszukiwanym elementem
- **Operator** $\mathbb{B}$ — obrót wszystkich amplitud wokół wartości średniej
- **Rezultat** — Wzmacnianie amplitudy prawdopodobieństwa związanej z właściwym rozwiązaniem

Plan wystąpienia

1. Informatyka Kwantowa – podstawy
2. Opis problemu (przeszukiwanie zbioru)
3. Intuicyjna interpretacja geometryczna
4. Uzasadnienie matematyczne
5. Wynik symulacji w Numerical Python

Symulacja

```
Total number of steps: 2

Step number: 0
----------------
Probability of search success:  0.1250

Step number: 1
----------------
Probability of search success:  0.7812

Step number: 2
----------------
Probability of search success:  0.9453

Final quantum register |phi> state:
[[-0.0884]
 [-0.0884]
 [-0.0884]
 [-0.0884]
 [-0.0884]
 [ 0.9723]
 [-0.0884]
 [-0.0884]]

Probability of search success:  0.9453
MEASUREMENT! ->  success.
```

Koniec

Dziękuję.
Pytania lub wątpliwości?