

Quanten Computing verstehen: Grovers Algorithmus

Sabrina Cielas,¹ Till Pilarczyk²

Abstract:

Keywords: Quantencomputer; Quantum Computing; Grovers-Algorithmen

1 Begriffserklärung (nur wenn Platz ist)

Der Text bezieht sich auf die Quelle... Kapitel 6.

Es Wird vorausgesetzt, das bekannt ist, was die folgende Begrifflichkeiten sind

- Superposition
- Amplitude
- Messen
- Quantenorakel
- Hadamar Gatter bzw Pauli-Gatter
- Lokalisitätsprinzip

Falls wird noch Platz haben können wir diese begriffe noch erklären

Habe eine Mail an den Schmidt geschrieben und gefragt, ob wir diese Sachen voraussetzen können

2 Einleitung

Fragestellung Motivation (Mit Aufwandserklärung) also das Aufrufe der Orkale = Laufzeit-entscheidend Aufbau des Papers

¹ Hochschule Düsseldorf, Gebäude 4, Münsterstraße 156, 40476 Düsseldorf, Deutschland sabrina.cielas@study.hs-duesseldorf.de

² Hochschule Düsseldorf, Gebäude 4, Münsterstraße 156, 40476 Düsseldorf, Deutschland till.pilarczyk@study.hs-duesseldorf.de

3 Grovers-Algorithmus

Um den Algorithmus besser verstehen zu können werden die Rahmenbedingungen wie folgt formalisiert. Die Datenbank, besitzt N Elemente, wobei $N = 2^n$ entspricht. Den Datensätzen werden die Elementen $\{0, 1\}^n$ zugeordnet. Wenn $n = 2$ entspräche, besäße die Datenbank folgende Elemente: **00, 01, 10, 11**. Das gesuchte Element wird als \hat{x} bezeichnet. Die Datenbank wird als eine Funktion $f: \{0, 1\}^n \rightarrow \{0, 1\}$ mit folgender Eigenschaft umgesetzt:

$$f(x) = \begin{cases} 1 & \text{für } x = \hat{x} \\ 0 & \text{sonst} \end{cases}$$

Ist die Eingabe in die Funktion das gesuchte Element, so gibt diese **1** zurück, in allen anderen Fällen **0**. Mithilfe dieser Funktion wird das Quantenorakel. $U_f : |x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$ erstellt. Das Quantenorakel, negiert das Vorzeichen des gesuchten Elements. Diese Funktionsweise ist aus dem Deutsch-Jozsa Algorithmus bekannt

3.1 Prinzip

Der Grover Algorithmus lässt sich in drei Schritte aufteilen.

1. **Superpositionen aufbauen**
Im ersten Schritt werden alle Quantenbits in die Superposition gebracht.
2. **Amplitudenveränderung durchführen (Grover Iteration G)**
Der zweite Schritt verändert die Amplituden der Elemente. Dabei wird die Amplitude des gesuchten Elementes erhöht und alle anderen verringert. Dieser Schritt wird auch *Grover Iteration (G)* genannt und abhängig von der Anzahl der Elementen öfters wiederholt. Wie oft G ausgeführt werden muss wird im Abschnitt 4.1.2 erläutert.
3. **Messen**
Im letzten Schritt werden die Quantenbits gemessen. Mit einer hohen Wahrscheinlichkeit kommt das gesuchte Element \hat{x} heraus.

3.1.1 Amplitudenveränderung

Die Amplitudenveränderung besteht aus zwei Schritten. Beim ersten Schritt wird das Vorzeichen der Amplitude von \hat{x} negiert. Im zweiten Schritt wird die negative Amplitude ausgenutzt um diese zu verstärken. Dies passiert in dem alle Amplituden am Mittelwert aller Amplituden gespiegelt werden.

Negieren der Amplitude

Um die Amplitude von \hat{x} zu negieren, wird ein Hilfsbit und das Quantenorakel benötigt. Das Hilfsbit wird in den Zustand $H|1\rangle$ mithilfe eines Hadamard Gatters gebracht. Dadurch

erhalten wird $|x\rangle \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Anschließend wird das Quantenorakel angewendet und damit das Vorzeichen der Amplitude von \hat{x} negiert. Das Hilfsbit wird nun nicht mehr benötigt und kann in folgenden Berechnungen weggelassen werden. Dies lässt sich auch an der Abbildung 1 erkennen. Dort ist Anwendung des Quantenorakel und graphisch die Amplituden aller Elemente der Datenbank mit $N = 4$ vor und nach der Anwendung des Quantenorakels zu sehen. Die negative Amplitude von \hat{x} hat keinen Einfluss auf das Messen,

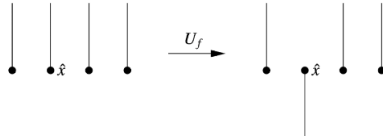
$$\begin{aligned}
 |x\rangle \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) &\xrightarrow{U_f} |x\rangle \frac{1}{\sqrt{2}}(|f(x)\rangle - |1 \oplus f(x)\rangle) \\
 &= |x\rangle (-1)^{f(x)} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)
 \end{aligned}$$


Abb. 1: Amplitudenveränderung
Quelle: [Ho17, S. 141]

um mit einer erhöhten Wahrscheinlichkeit das Element \hat{x} nach dem Messen zu erhalten wird Schritt zwei benötigt.

Spiegelung am Mittelwert

Um zu zeigen, dass die Spiegelung der Amplituden am Mittelwert den gewünschten Effekt hat, folgen nun ein paar Beispielrechnungen. Eine Spiegelung an einem Wert \mathbf{m} entspricht der Abbildung: $\alpha \rightarrow 2 \times \mathbf{m} - \alpha$.

Bei folgender Rechnung wird angenommen, dass die Datenbank vier Elemente enthält ($N = 4$). Der Mittelwert - nach der Negation von \hat{x} - hat den Wert: $\mathbf{m} = \frac{1}{4} \times (\frac{1}{2} - \frac{1}{2} + \frac{1}{2} + \frac{1}{2}) = \frac{1}{4}$. Die Spiegelung von \hat{x} entspricht $-\frac{1}{2} \times \frac{1}{4} - (-\frac{1}{2}) = 1$. Damit ist die Amplitude des gesuchten Elements gleich eins. Das Messen würde mit einer Wahrscheinlichkeit von 100% das gesuchte Element \hat{x} zurückgeben. Die Amplituden aller anderen Elemente entwickeln sich wie folgt: $\frac{1}{2} \times \frac{1}{4} - \frac{1}{2} = 0$

Sei $N = 8$, so würde die Amplitude von \hat{x} nach der ersten Grover Iteration $\frac{5}{5\sqrt{8}}$ und alle anderen Amplituden $\frac{1}{2\sqrt{8}}$ betragen. Nach einer Spiegelung am Mittelwert ist die Amplitude von \hat{x} wieder positiv, bevor ein erneutes Spiegeln möglich ist, um die Amplituden weiter zu verstärken/verringern, muss erneut das Quantenorakel angewandt werden. Nach einer erneuten Negation und Spiegelung, hätte das gesuchte Element \hat{x} eine Amplitude von **0,973**, alle anderen Elemente eine von **-0,088**. Würde nun gemessen werden, erhielte man mit einer Wahrscheinlichkeit von 93 % das gesuchte Element. Ein erneutes Spiegeln würde die Amplituden von \hat{x} im Gegensatz zur Erwartung wieder verringern und alle anderen erhöhen. Es ist daher besonders wichtig, dass nicht zu viele

Grover Iterationen ausgeführt werden. Wie die Genaue Anzahl an Iterationen berechnet werden kann folgt im Abschnitt 4.1.2

3.1.2 Graphische Darstellung des Grover Algorithmus

Der Grover Algorithmus sieht nach den bisherigen Erklärungen wie in Abbildung ?? aus. Alle QBits werden mithilfe der Hadamar Gatter in Superpositionen gebracht. Alles danach



Abb. 2: Graphische Darstellung des Grover Algorithmus
Quelle: Anlehnung an [Ho17, S. 146]

bis zum Messen Symbol am Ende der Abbildung ist die Grover Iteration. Der Äußere Kasten steht in der Abbildung für das mehrfache Wiederholen der Grover-Iteration. V_f steht für das Quantenorakel, jedoch wird hier das Hilfsbit nicht mit eingezeichnet. Anschließend folgt die Spiegelung am Mittelwert. Wie diese genau mithilfe von Gattern umgesetzt wird, folgt im nächsten Abschnitt (3.2). Nach dem ausführen der Grover Iterationen werden die QBits gemessen.

3.2 Realisierung der Spiegelung am Mittelwert

Die Abbildung $\alpha \rightarrow 2 \times \mathbf{m} - \alpha$ lässt sich mithilfe einer Matrixberechnung umsetzen.

$$\mathbf{D}_N \times \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{N-1} \end{pmatrix}, \text{ mit } \mathbf{D}_N = \begin{pmatrix} -1 + \frac{2}{N} & \frac{2}{N} & \cdots & \frac{2}{N} \\ \frac{2}{N} & -1 + \frac{2}{N} & \cdots & \frac{2}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{2}{N} & \frac{2}{N} & \cdots & -1 + \frac{2}{N} \end{pmatrix}$$

3.2.1 Beispielrechnung: Realisierung der Spiegelung am Mittelwert

Sei $N = 4$ so ergibt sich folgende Rechnung:

$$\mathbf{D}_4 \times \begin{pmatrix} 0,5 & -0,5 & 0,5 & 0,5 \end{pmatrix}^T.$$

$$\begin{pmatrix} -0,5 & 0,5 & 0,5 & 0,5 \\ 0,5 & -0,5 & 0,5 & 0,5 \\ 0,5 & 0,5 & -0,5 & 0,5 \\ 0,5 & 0,5 & 0,5 & -0,5 \end{pmatrix} \times \begin{pmatrix} 0,5 \\ -0,5 \\ 0,5 \\ 0,5 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

Dieses Ergebnis gleicht sich mit dem Ergebnis aus Abschnitt 3.1.1. In diesem wurden ebenfalls alle Elemente einer Datenbank mit $N = 4$ an dem Mittelwert der Amplituden gespiegelt. Folgende Abbildung 3 zeigt graphisch, wie sich die Amplituden verändert haben.

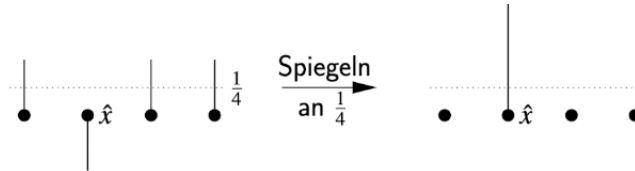


Abb. 3: Spiegelung am Mittelwert
Quelle: [Ho17, S. 142]

Wenn eine $N \times N$ Matrix verwendet wird, dann verstößt dies gegen das Lokalitätsprinzip. Daher muss die \mathbf{D}_N Matrix in verschiedene unitäre Matrizen zerlegt werden. \mathbf{D}_N kann in ein Produkt aus drei unitäre Matrizen zerlegt werden.

$$\mathbf{D}_N = -\mathbf{H}_N \times \mathbf{R}_N \times \mathbf{H}_N, \text{ mit } \mathbf{R} = \begin{pmatrix} -1 & 0 & \dots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{pmatrix}$$

3.2.2 Beispielrechnung: Realisierung der Spiegelung am Mittelwert mithilfe von unitären Matrizen

Um zu zeigen, dass die Matrix \mathbf{D}_N wie oben abgebildet als Produkt von drei Matrizen zerlegt werden kann, folgt ein Beispiel mit $N = 4$. Das Ergebnis der Berechnung ist, wie in der beschrifteten Rechnung 4 zu sehen, gleich mit der zu erwarteten Matrix \mathbf{D}_4 . Der Beweis, dass dies auch für beliebige N zutrifft, befindet sich in dem Buch vom M. Hoemeister [Ho17, S. 309].

$$\begin{array}{ccc}
\frac{1}{2} \begin{pmatrix} -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 \end{pmatrix} & \times & \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & -1 & -1 & -1 \\ 1 & 1 & -1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & 1 & -1 \end{pmatrix} \\
-H_2 & R_4 & \text{Zwischenergebnis} \\
\\
\frac{1}{2} \begin{pmatrix} 1 & -1 & -1 & -1 \\ 1 & 1 & -1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & 1 & -1 \end{pmatrix} & \times & \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} = \frac{1}{4} \begin{pmatrix} -2 & 2 & 2 & 2 \\ 2 & -2 & 2 & 2 \\ 2 & 2 & -2 & 2 \\ 2 & 2 & 2 & -2 \end{pmatrix} \\
\text{Zwischenergebnis} & H_2 & D_4
\end{array}$$

Abb. 4: Beispielzerlegung von D_4
Quelle: Eigene Darstellung

3.2.3 Matrix R als lokale Transformation

Es wurde gezeigt, dass die Matrix D_N in ein Produkt aus Matrizen zerlegt werden kann. Da die Hadamard-Matrix mit Hilfe eines Gatters als lokale Transformation umgesetzt werden kann, ist bekannt. Dies muss jedoch auch noch für R_N gezeigt werden.

Um ein Gatter zu entwickeln zu können, welches die Transformation R_N umsetzt, muss verstanden werden, wie sich R_N bei einer Multiplikation von Matrizen auswirkt. Alle Werte einer Matrix, die mit R_N multipliziert wird, bleiben gleich. Lediglich die erste Zeile oder Spalte wird negiert. Dies ist davon abhängig, ob die Matrix R_N auf der Linken oder Rechten Seite der Multiplikation steht. In der ersten Zeile der Abbildung 4 ist die Negation der ersten Spalte zu sehen.

Multipliziert man R_N mit Amplituden, bedeutet dies, dass lediglich die Amplitude des ersten Elementes ($|0\dots 0\rangle$) negiert wird. Die Abbildung 5 zeigt den Quantenschaltkreis, wenn $N = 4$ gilt.

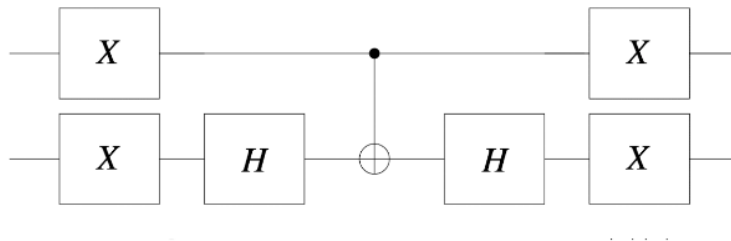


Abb. 5: R_4 Realisierung
Quelle: [Ho17, S. 145]

Es folgen weitere Beispielrechnungen, um zu verdeutlichen, dass der Quantenschaltkreis die Transformation R_4 ausführt.

3.2.4 Beispielrechnung: Matrix R als lokale Transformation

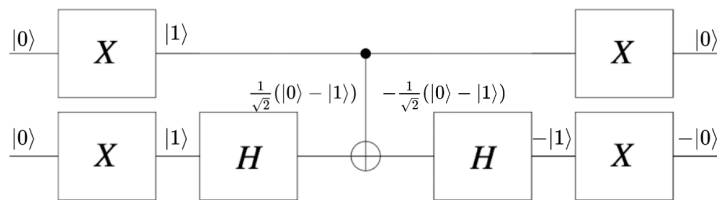


Abb. 6: $|00\rangle$ Werteveränderungen durch das R_4 -Gatter
Quelle: Anlehnung an [Ho17, S. 145]

In den Abbildungen 6 werden die Werte der QBits $|00\rangle$ und wie diese sich durch die einzelnen Gatter verändern dargestellt. Die Pauli X-Gatter invertieren den Wert eines QBits. Aus $|0\rangle$ wird $|1\rangle$ und andersherum. Das erste Bit wird bis zum letzten Pauli-X Gatter nicht mehr verändert. Es wird ausschließlich genutzt um zu schauen, ob das CNOT aktiviert wird. Das zweite QBit wird durch das erste Hadamar Gatter, in folgende Superposition $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ gebracht. Da das erste QBit den Wert $|1\rangle$ hat wird das zweite QBit durch das CNOT negiert, und hat den Wert: $-\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Anschließend durch läuft das zweite Bit wieder ein Hadamar Gatter. Das QBit hat anschließend folgenden Wert: $-|1\rangle$. Zum Schluss werden beide Bits ($|1\rangle$ und $-|1\rangle$) durch die Pauli-X Gatter invertiert und wir erhalten das gewünschte und erwartete Ergebnis $-|00\rangle$.

Die Abbildung 7 zeigt die QBits $|01\rangle$ und wie diese von den Gattern verändert

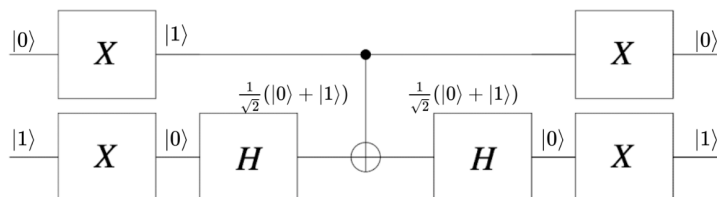


Abb. 7: $|00\rangle$ Werte Veränderungen durch das R_4 -Gatter
Quelle: Anlehnung an [Ho17, S. 145]

werden. Die ersten Pauli-X Gatter invertieren abermals die QBits, diese haben nun den Wert $|10\rangle$. Das erste QBit wird bis auf von dem letzten Pauli-X Gatter nicht verändert

und nur zur Aktivierung der Operation CNOT zuhelfe genommen. Das zweite QBit wird durch das Hadamar Gatter in die Superposition $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ gebracht. Diese verändert sich nicht durch die CNOT Operation. Nach der CNOT Operation wird das QBit durch das zweite Hadamar Gatter wieder in den Basiszustand $|0\rangle$ gebracht. Zuletzt durchlaufen die beiden QBits ein Pauli-X Gatter welches die QBits von den Basiszustände $|10\rangle$ in den erwarteten Zustand $|01\rangle$ überführt. Die QBits haben wie erwartet, nach dem durchlaufen des Quantenschaltkreises die selben Zustände wie vorher.

Bei den QBits $|10\rangle$ und $|11\rangle$, wird die CNOT Operation nicht ausgeführt, da das erste QBit in den Basiszustand $|0\rangle$ überführt werden. Die QBits werden durch die doppelte Ausführung der Gatter ebenfalls nicht verändert. Aus diesen Gründen ist für diese beiden Beispiel keine Abbildung vorhanden.

An den Beispielen kann gesehen werden, dass der Quantenschaltkreis die gewünschte Operation R_4 ausführen.

3.3 Graphische Darstellung des Grover Algorithmus

In der Abbildung 8 ist wie in der Abbildung 2 der Grover Algorithmus Graphisch dargestellt. Die Abbildung enthält alle Gatter die für den Grover Algorithmus benötigt werden. Hilfsbits sind hier ebenfalls nicht eingezeichnet.

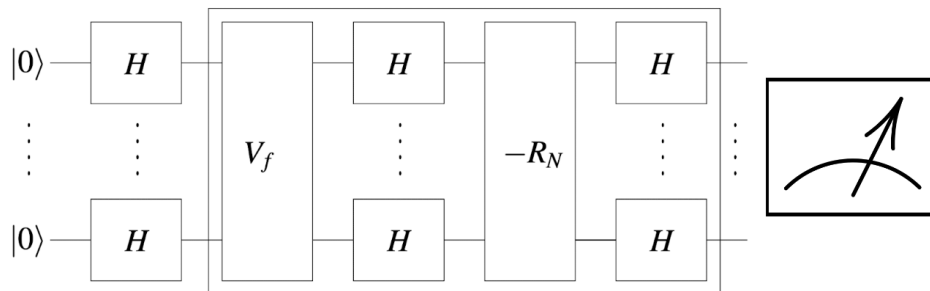


Abb. 8: Graphische Darstellung des Grover Algorithmus
Quelle: Anlehnung an [Ho17, S. 146]

4 Bestimmung der Anzahl an Grover-Iterationen

Nachdem der genaue Ablauf des Algorithmus erklärt wurde, bleibt die Frage: Wie oft muss die Grover Iteration durchgeführt werden, um mit einer hohen Wahrscheinlichkeit das gesuchte Element \hat{x} zu messen? noch offen.

4.1 Geometrische Veranschaulichung

Mithilfe der Geometrie kann die Anzahl der Iterationen bestimmt werden. Das Schrittweise erhöhen der Amplitude des gesuchten Elements kann auf der Bloch Kugel als Rotation aufgefasst werden.

In der Geometrie entspricht die Spiegelung eines Punktes an zwei Ebenen eine Drehung um den Winkel $2 \times \beta$, wobei β der Winkel zwischen den Ebenen ist. Dies ist in der Abbildung 9 verdeutlicht. Diese Eigenschaft kann sich zunutze gemacht werden. Ist der Winkel bekannt, kann ausgerechnet werden, um wie viel der Grad gedreht werden muss, um das gesuchte Element zu erreichen und damit wie oft die Grover-Iteration durchgeführt werden muss.

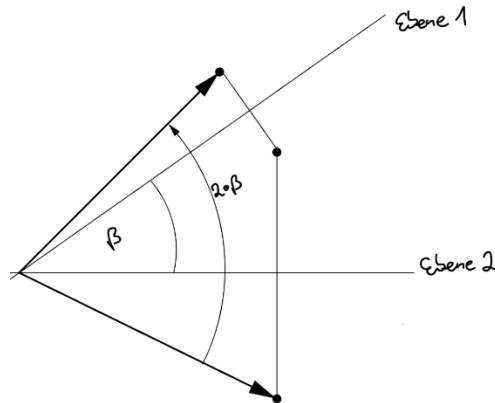


Abb. 9: R_4 Spiegelung an zwei Ebenen
Quelle: Anlehnung an [Ho17, S. 149]

Wie der Name schon sagt, handelt sich bei der Spiegelung am Mittelwert um eine Spiegelung. Die zweite Spiegelung ist das Negieren von dem gesuchten Element \hat{x} . Die Formel zur Spiegelung an einem Wert m lautet: $\alpha \rightarrow 2 \times m - \alpha$. Ist $m = 0$ erhalten wir $\alpha \rightarrow -\alpha$. Dies zeigt, dass die Negation eines Elementes, eine Spiegelung an dem Wert 0 ist.

Der Winkel lautet $\sin(\beta) = \langle s | \hat{x} \rangle$, mit $s = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$, welches die Allgemeine Superposition ist. Ausgerechnet ergibt dies $\sin(\beta) = \frac{1}{\sqrt{N}}$. Daran lässt sich erkennen, dass die Anzahl der Iterationen abhängig von der Anzahl der Datenbankelemente ist und nicht von \hat{x} ist. Wäre dies nicht so, hätte das zur Folge, dass für jedes gesuchte Element die Suche anpassen werden müsste.

4.1.1 Beispielrechnung: Geometrische Veranschaulichung

Sei $N = 4$, dann folgt daraus, dass $\sin(\beta) = \frac{1}{\sqrt{4}}$ ist. Nach Beta aufgelöst ergibt sich $\beta = \frac{\pi}{6}$. Wird nun eine Grover Iteration durchgeführt, ändert sich der Winkel wie folgt: $\beta = \frac{\pi}{6} + 2 \times \frac{\pi}{6} = \frac{\pi}{2}$. Wird nun $\sin(\frac{\pi}{2})$ ausgerechnet, erhalten wir **1**. Dies bedeutet, dass wir mit einer Drehung das gesuchte Element erreicht haben. Wird nun gemessen erhalten wir mit einer Wahrscheinlichkeit von 100 % das gesuchte Element. Dies wird auch wie im Abschnitt 3.1.1 beschrieben erwartet.

4.1.2 Anzahl an Grover Iterationen

Durch die Rechnung konnte gezeigt werden, dass der Startwinkel $\frac{1}{\sqrt{N}}$ beträgt und der Winkel nach T Grover Iterationen den Wert $(2 \times T + 1) \times \frac{1}{\sqrt{N}}$ hat. Falls für $T = \frac{\pi}{4} \times \sqrt{N}$ gewählt wird, wird immer sehr nah an das gesuchte Element rotiert. In der Praktischen Umsetzung, muss T immer abgerundet werden, da nur eine ganze Zahl an Iterationen durchgeführt werden kann. Wird T aufgerundet, werden zu viele Grover-Iterationen durchgeführt und es wird sich vom gesuchten Element wieder entfernt.

Dadurch dass die Anzahl der Grover-Iterationen $\frac{\pi}{4} \times \sqrt{N}$ beträgt und in jeder Iteration einmal das Quantenorakel aufgerufen wird, beträgt die Laufzeit des Grover Algorithmus $O(\sqrt{N})$

5 Varianten der Quantensuche

was für Varianten der Quantensuche gibt es ?

6 Anwendungen des Grovers Algorithmus

Wofür kann man den Algorithmus alles verwenden ?

7 Folgen für die Fähigkeit von Quantencomputern

Hier kommen die Folgen für die Fähigkeit von Quantencomputern hin

Literatur

[Ho17] Homeister, M.: Quantum Computing verstehen. Springer Vieweg Wiesbaden, 2017.