**Seminar: Quanten-Computer**
**Referent: M.Tarik**
**Vortrag: Der Grover-Algorithmus (Ausarbeitung)**

## I) Einleitung:

Mit Hilfe des Grover-Algorithmus ist es möglich, klassische Suchvorgänge zu beschleunigen. Er ist einer von wenigen grundlegenden Algorithmen, die die quantenmechanischen Konzepte benutzen. Der Suchverfahren ist schneller als jeder klassische Suchalgorithmus. Dadurch wird der Algorithmus wichtiger als zum Beispiel der Deutsch-Josza-Algorithmus, der kaum praktische Anwendungen hat.

L.Grover hat in 1996 seinen neuen Algorithmus für die Suche in Datenbanken vorgestellt. Seitdem ist der Algorithmus kaum erweitert worden, weil man beweisen kann, dass es keinen schnelleren Suchalgorithmus für Quantencomputer geben kann.

Will man z.B. in einem Telefonbuch mit 1.000.000 Telefonnummern den zu einer bestimmten Telefonnummer Name finden, muss man klassisch einen Eintrag nach dem anderen durchprobieren, bis man den Richtigen findet. Im Mittel braucht man 500 000 Versuche.

Im Allgemeinen für eine Datenbank mit N Elemente braucht man also klassisch durchschnittlich $\frac{N}{2}$ Schritte, um das gesuchte Element zu finden, d.h. die Geschwindigkeit dieses Algorithmus beträgt O(N). Der Grover-Algorithmus benötigt für die Suche in solchen Datenbanken, die keine Struktur oder Sortierung aufweisen, nur O($\sqrt{N}$) Schritte, was eine quadratische Verbesserung im Vergleich zum klassischen Algorithmus darstellt. In der Laufzeit unterscheidet sich der Grover-Algorithmus ein wenig von anderen Quantenalgorithmen, die oft eine exponentielle Beschleunigung gegenüber klassischen Algorithmen bieten, wie z.B. Shor-Algorithmus. Aber wenn N groß ist, dann bedeutet diese quadratische Verbesserung jedoch einen enormen Geschwindigkeitsvorteil.

## II) Problemdarstellung:

Der Suchraum, der durchsucht werden soll, habe N Elemente:

S = {$x_0$, $x_1$,…$x_{N-1}$}. Wir werden bestimmte Elemente $x_i$ aus S durchsuchen. Sei L die Lösungsmenge, und M die Lösungsanzahl. Gegeben sei die Funktion f:

f(x)=1 wenn x ϵ L, f(x)= 0 sonst.

Das Ergebnis von f zeigt uns an, ob wir eine Lösung gefunden haben. Erfüllt x unser Suchkriterium, ist f(x) = 1, andernfalls ist f(x) = 0.

Klassisch werden alle Datensätze durchprobiert, die Funktion f wird im Mittel $\frac{N}{2}$ aufgerufen, da der Algorithmus O(N) benötigt. Die Behauptung ist nun, dass dieses Problem mit einem Quantenschaltkreis mit der Komplexität O($\sqrt{N}$) gelöst werden kann. Z.B. das Problem mit 1.000.000 Telefonnummern wird eine Lösung in 1000 Schritten gefunden, statt 500 000 klassisch.

**III) Das Orakel:**

Das Herz des Algorithmus ist das Orakel. Ein Orakel ist ein Black Box, d.h. die interne Implementierung bleibt im Voraus unbekannt, d.h. auch es ist nicht von Algorithmus vorgegeben. Das Orakel ist der unitäre Operator $U_f$:

$$U_f |x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$$

Sei $|y\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, wenn x eine Lösung ist (f(x)=1), wird ihre Phase um π verschoben (Abb.1): $U_f |x\rangle|y\rangle = \left(\frac{1}{\sqrt{2}}\right)(-1)^{f(x)}|x\rangle(|0\rangle - |1\rangle)$

Das Orakel kann die Lösung in einem Schritt erkennen. Aber eine Messung nach der Anwendung des Orakels ergibt das gesuchte Ergebnis nur mit einer Wahrscheinlichkeit $\frac{1}{N}$. Also wie kann man diese Wahrscheinlichkeit deutlich erhöhen?

**IV) die Grover-Idee:**

Das Ziel ist es, aus dem Suchraum ein bestimmtes Element zu finden, das die Lösung des Suchproblems darstellt. Dabei sollen möglichst wenige Orakelaufrufe erfolgen. Die Idee ist: Man verwendet die sog. „Grover-Iteration“ in einem Quantenregister, um die Wahrscheinlichkeit des richtigen Ergebnisses zu erhöhen und die der Nichtlösungen zu verringern. Das Ziel des Grover-Algorithmus ist, ein n-Qubit-Register so zu präparieren so, dass

die Wahrscheinlichkeitsamplitude einer Lösungskomponente hinreichend groß ist. Das Verfahren lautet:

**1-** Das Register wird im Anfangszustand $|0\rangle$ präpariert.

**2-** Eine gleichmäßige Superposition $|\Psi\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$ über alle Elemente von S durch die Hadamard-Transformation $\mathrm{H}^{\otimes n}$ wird erzeugt, jede Komponente von $|\psi\rangle$ hat eine Amplitude $\frac{1}{\sqrt{N}}$.

**3-** Das Orakel wird angewandt, und dadurch die Amplituden der gefundenen Lösungen geflippt. Alle N-M Nichtlösungskomponenten haben unveränderte Amplitude $\frac{1}{\sqrt{N}}$ und für die Lösungen $-\frac{1}{\sqrt{N}}$.

**4-** Wir wenden die Hadamard-Transformation an.

**5-** Wir wenden den Phasenverschiebungsoperator P an. Er ist definiert durch (P ist unitär): $\mathrm{P} = 2|0\rangle\langle 0| - \mathrm{I}$. P verändert nicht den Zustand 0, und für die anderen Zustände wird die Phase um π verschoben.

**6-** Dann wieder die Hadamard-Transformation anwenden.

Die drei letzten Schritte werden die Amplitude der Lösungen erhöhen, d.h. die Wahrscheinlichkeit - bei einer Messung - wird größer. Und die vier letzten Schritte sind die Grover-Iteration oder Grover-Operator genannt.

Der Grover-Operator lautet:

$$G = H^{\otimes n}.(2|0\rangle\langle 0| - \mathrm{I})H^{\otimes n}.U_f$$
$$= (2|\Psi\rangle\langle\Psi| - \mathrm{I}).U_f$$

**V) Inversion um den Mittelwert:**

Sei der Operator: $D = 2|\psi\rangle\langle\psi| - \mathrm{I}$. Die Wirkung von D auf einen allgemeinen Zustand $|\phi\rangle = \sum_{x=0}^{N-1} \alpha_x |x\rangle$ ist: $D|\phi\rangle = \sum (2A - \alpha_x)|x\rangle$. D.h. die

Koeffizienten von |x> werden um den Mittelwert A gespiegelt, man nennt D die „Inversion um den Mittelwert“ (Abb.1):

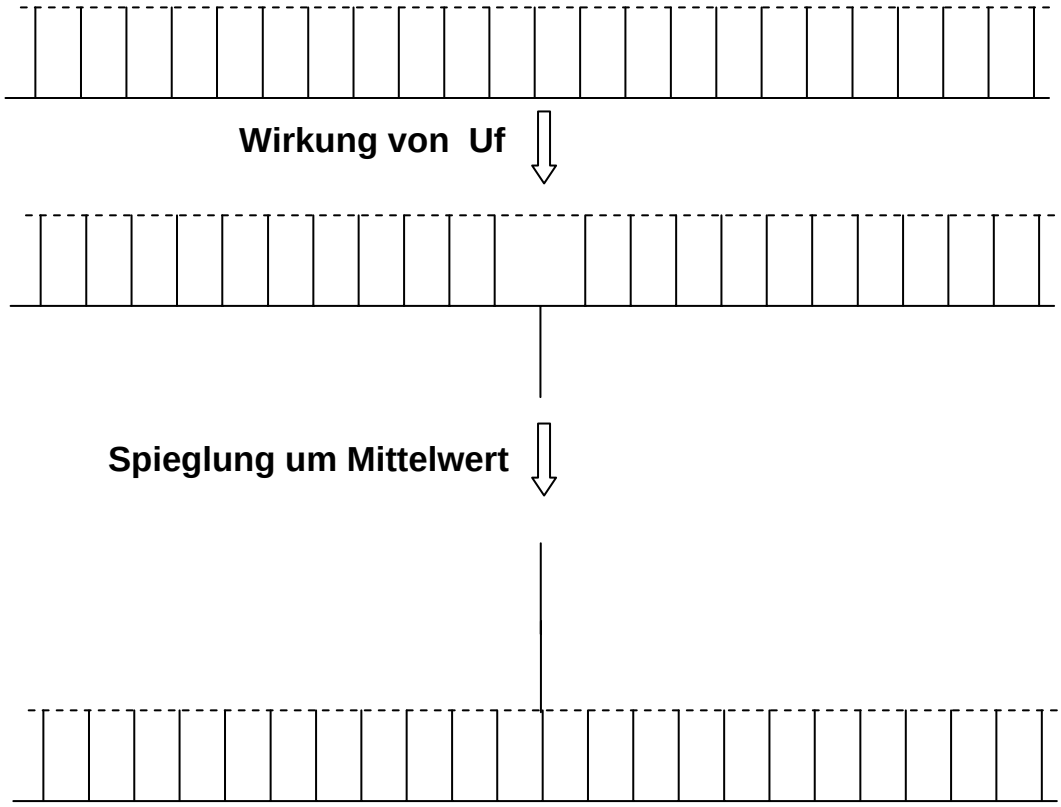


**Abb.1 Wirkung der Grover-Iteration.**

## VI) Der Grover-Algorithmus:

Die Wahrscheinlichkeitsamplitude wird nach einer Anwendung der Grover-Iteration nur ca. drei Mal erhöht. Dann wird die Grover-Iteration „oft“ angewandet, um mit Sicherheit bei einer Messung eine oder mehrere Lösungen zu finden. Das Ziel der Grover- Iteration ist es, die Amplitude des Lösungszustandes mit jedem Iterationsschritt zu erhöhen. Nach dem Aufruf des Orakels hat der Lösungszustand das Vorzeichen gewechselt. Dadurch sinkt der Amplituden-Mittelwert ein wenig. Je größer der Suchraum ist, desto kleiner ist diese Änderung. Danach werden alle Amplituden am Mittelwert gespiegelt. So erhöht sich die Amplitude der Lösung stark, während allen anderen Amplituden kleiner werden. Auf dieser Weise steigt die Amplitude des gesuchten Zustandes mit jeder Grover-Iteration so, dass am Ende eine Messung mit hoher Wahrscheinlichkeit das gewünschte Ergebnis liefert.

- **Grover-Algorithmus: (Der Fall M=1):**

**1-** Input: $|0\rangle$

**2-** Anwendung von $H^{\otimes n}$ :dann wird eine gleichmäßige Superposition erzeugt.

**3-** G k-Mal anwenden: (mit k = O($\sqrt{N}$ ) die Laufzeit des Algorithmus):

$$(G)^{\otimes k}|\Psi\rangle[(\frac{1}{\sqrt{2}})(|0\rangle-|1\rangle)] = |x_0\rangle[(\frac{1}{\sqrt{2}})(|0\rangle-|1\rangle)]$$

**4-** Output: $|x_0\rangle$ (bei einer Messung ergibt sich die gesuchte Lösung $x_0$)

**VII) Die geometrische Interpretation und die Laufzeit:**

Man kann sich die Grover-Iteration anschaulich als Rotation eines Vektors in einem zweidimensionalen Vektorraum vorstellen. Dieser Raum wird durch zwei Basisvektoren aufgespannt. Aus alle Lösungszuständen bildet man den Basisvektor $|\beta\rangle = \frac{1}{\sqrt{M}} \sum_{x\in L} |x\rangle$ und aus Nicht-Lösungen den Vektor:

$|\alpha\rangle = \frac{1}{\sqrt{N-M}} \sum_{x\in S,\, x\notin L} |x\rangle$. Also der allgemeine zustand lautet:

$$|\psi\rangle = \frac{1}{\sqrt{N}}\Sigma|x\rangle = \sqrt{\frac{N-M}{N}}|\alpha\rangle + \sqrt{\frac{M}{N}}|\beta\rangle = \cos(\frac{\theta}{2})|\alpha\rangle + \sin(\frac{\theta}{2})|\beta\rangle$$

Der erste Schritt der Iteration, das Orakel $U_f$ stellt eine Spiegelung um den Basisvektor $|\alpha\rangle$ dar. Der zweite Schritt ist die Inversion um den Mittelwert. Diese Operation entspricht wieder einer Spiegelung um den Vektor $|\Psi\rangle$. Zusammen mit der ersten Spiegelung ergibt sich insgesamt eine Rotation. Insgesamt wird also um den Winkel $\theta$ rotiert (Abb.2).

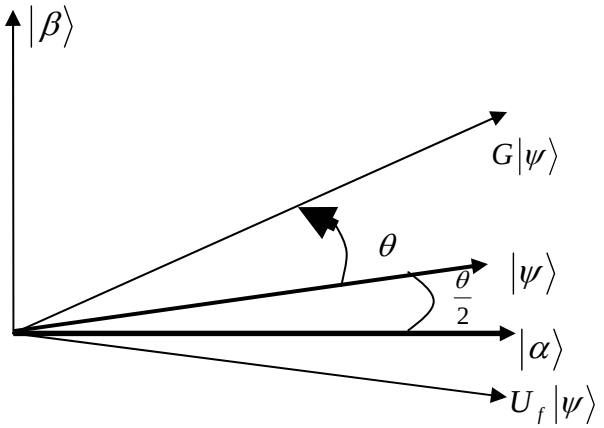


**Abb.2: Geometrische Veranschaulichung der Grover-Iteration.**

Das Ziel ist, $|\Psi\rangle$ durch mehrere Grover-Iterationen möglichst in $|\beta\rangle$-Richtung zu drehen. Der Operator G dreht $|\Psi\rangle$ also um den Winkel $\theta$ in Richtung des Lösungsvektors $|\beta\rangle$. Da nach einer Iteration man den Zustand $G|\psi\rangle = \cos(\frac{3\theta}{2})|\alpha\rangle + \sin(\frac{3\theta}{2})|\beta\rangle$ erhält. Nach genügend Iterationen wird man am Ende mit hoher Wahrscheinlichkeit den gesuchten Zustand Bzw. die gesuchten Zustände messen. Bei weiterer Iterationen entfernt sich der Vektor $|\Psi\rangle$ wieder von $|\beta\rangle$ und die Wahrscheinlichkeit, einen Lösungszustand zu messen, sinkt. Anders als man erwarten würde, steigt die Wahrscheinlichkeit nicht mit der Anzahl der Iterationen, sondern sie oszilliert zwischen 0 und 1 mit einer gewissen Periodizität. Das bedeutet, man muss die günstigste Zahl an Iterationen bestimmen, bevor man messen kann. Nach k-Iterationen folgt der Zustand: $G^{\otimes k}|\psi\rangle = \cos(\frac{2k+1}{2}\theta)|\alpha\rangle + \sin(\frac{2k+1}{2}\theta)|\beta\rangle$

Die Wahrscheinlichkeit, das gewünschte Element am Ende durch eine Messung zu finden, ist: $f(k) = \sin^2(\frac{2k+1}{2}\theta)$. Um mit Sicherheit, eine Lösung zu finden, muss also: $\sin(\frac{2k+1}{2}\theta) = 1$. Der Wert k gibt dabei die Anzahl der benötigten Grover-Iterationen an. Dann folgt: $k = \left[\frac{\pi}{4}\sqrt{\frac{N}{M}}\right]$. Hier ist angenommen, dass $\theta$ klein ist, d.h. die Lösungsanzahl M<<N ist. Somit haben wir eine obere Grenze für die Zahl k der Grover-Iterationen gefunden. Die Laufzeit des Grover-Algorithmus ist demnach O($\sqrt{N}$) und das ist eine quadratische Beschleunigung im Vergleich um klassischen Algorithmus, der O(N) Schritte braucht.