digital whisper
Technologic Papers

בין 0 ל-1: מבוא למחשבים שלא חושבים כמונו

מאת עומר גולן וגיל בוכבינדר

הקדמה

רובנו שמענו הרבה על מחשבים קוונטים בשנים האחרונות אבל על מה כל הדיבור? מה ההבדל בין המחשב הסטנדרטי שכולנו מכירים ואוהבים למחשב הקוונטי? מהם המחשבים הקוונטים האלה ולמה הם כל כך טובים? על כל השאלות האלה נענה, או לפחות ננסה לענות, במאמר זה. המאמר מתמקד בצד המתמטי והתיאורטי של המחשב הקוונטי.

אך, רגע לפני שמתחילים לקרוא חשוב לציין שבמאמר נשתמש במושגים בסיסיים באלגברה לינארית, בקומבינטוריקה ובהסתברות.

כמה מושגים לפני שנתחיל

קיוביטים

ההבדל העיקרי בין מחשבים רגילים למחשבים קוונטים הוא הדרך בה הם מאחסנים את המידע. בעוד שמחשבים רגילים שומרים את המידע ביחידות שנקראות ביטים שמצבם הוא 0 או 1, מחשבים קוונטים משתמשים בביטים מיוחדים הנקראים קיוביטים.

קיוביטים הם לא רק 0 או 1, הם שומרים את ההסתברות שהם יהיו 0 או 1 ובכך הם מין שילוב של שניהם. דוגמה טובה לזה היא הייצוג של ביט דלוק רגיל - ההסתברות שהוא יהיה 0 היא 0 וההסתברות שהוא יהיה 1 היא 1. למצב שבו קיוביט יכול להיות בין 0 ל 1 נקרא סופרפוזיציה.

סופרפוזיציה

בדרך כלל הסופרפוזיציה תיוצג ע"י הסימון: $\alpha|0\rangle + \beta|1\rangle$ כאשר α יקרא המקדם של המצב "0" וβ יקרא המקדם של המצב "1". עוד סימון לסופרפוזיציה של קיוביט בודד הוא וקטור בגודל 2 כאשר הערך בכניסה הראשונה הוא המקדם של המצב "0" והערך בכניסה השניה הוא המקדם של המצב "1". הנורמה בריבוע של מקדם של מצב מסוים מייצגת את ההסתברות שהקיוביט יהיה במצב הזה.

מהחוק הזה, סופרפוזיציה יכולה להתקיים אם הנורמה של הוקטור המייצג אותה, שהיא אורך הוקטור, שווה ל-1. חשוב לציין שהמקדמים של מצבים יכולים להיות חיוביים, שליליים ואפילו מדומים כל עוד הנורמה היא 1. נשים לב שאם אחד המקדמים הוא 0 ניתן להשמיט את המצב מהביטוי, ואם אחד המקדמים הוא אחד אפשר לכתוב רק את המצב. לדוגמא:

$$0|0\rangle + 1|1\rangle = |1\rangle$$

אבל, אם הקיוביט מורכב מהסתברויות של המצבים 0 ו-1 מה נעשה כאשר נרצה לקרוא אותו? איך נדע מה הוא כאשר הוא מוגדר רק לפי הסתברות? במצב זה נצטרך לבצע "קריסה" של הקיוביט. מה זה אומר? בדומה לבעית החתול של שרדינגר, החתול נמצא בסופרפוזיציה של שני המצבים - חי ומת. אבל, כאשר נרצה לבדוק מה המצב שלו ונפתח את הקופסה החתול יהיה באחד משני המצבים, הוא יהיה חי או מת ולא שניהם ויפסיק להיות בסופרפוזיציה, פעולה זאת תקרא מדידה של המערכת ובעזרתה נדע מה המצב של הקיוביט - 0 או 1. לאחר פעולה זו הקיוביט יהפוך לביט רגיל - 0 או 1.

רגיסטר של קיוביטים

עד עכשיו עסקנו במצב שבו יש רק קיוביט בודד, אבל איך תראה מערכת של כמה קיוביטים? בשונה מביטים רגילים, קיוביטים יכולים להיות תלויים אחד בשני, כלומר יכול להיווצר מצב שבו אם נדע את המצב של קיוביט אחד נוכל לדעת מה המצב של השני ולכן במצב זה ניתן רק לחשוב עליהם כסופרפוזיציה יחידה. למערכת כזאת נקרא רגיסטר קוונטי.

סופרפוזיציה של רגיסטר קוונטי המכיל n קיוביטים תיוצג על ידי וקטור באורך 2^n כאשר כל כניסה מייצגת את המקדם של קומבינציה אחרת של n ביטים רגילים. עוד דרך לייצג רגיסטר קוונטי כזה היא בעזרת הסימון השני שהצגנו בהתחלה, לדוגמא הייצוג של שני קיוביטים:

$$\alpha_1|00\rangle + \alpha_2|01\rangle + \alpha_3|10\rangle + \alpha_4|11\rangle$$

נשים לב שניתן לכתוב את הייצוג העשרוני של המצבים במקום את ייצוגם הבינארי:

$$\alpha_1|0\rangle + \alpha_2|1\rangle + \alpha_3|2\rangle + \alpha_4|3\rangle$$

כשמודדים רגיסטר, אפשר לבחור קבוצה של קיוביטים שאותה רוצים למדוד. המדידה מקריסה את החלק הזה של הרגיסטר, ומשאירה את החלק השני. נשים לב שתוצאות המדידה משפיעות על הרגיסטר שנותר. אם לדוגמא במדידה של קיוביט אחד בתוך רגיסטר עם שני קיוביטים, מקבלים את הערך "1", הקיוביט שנשאר יכיל רק את המצבים שבהם לפני המדידה ערך הקיוביט השני היה 1. לדוגמה, ברגיסטר הבא:

$$\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

כאשר נמדוד את הקיוביט הראשון, אם התוצאה של המדידה היא 0 נשאר עם הקיוביט $|0\rangle$, ואם התוצאה היא 1 נשאר עם הקיוביט $|1\rangle$.

טנזור של קיוביטים

טנזור של שני קיוביטים הוא דרך לאחד את שניהם לרגיסטר אחד. טנזור יוצר רגיסטר שמדידתו שקולה למדידת שני הקיוביטים בנפרד (לפני איחודם לרגיסטר). נסמן טנזור כך: $\otimes$. הנוסחה לטנזור של שני קיוביטים היא:

$$(\alpha_1|0\rangle + \alpha_2|1\rangle) \otimes (\beta_1|0\rangle + \beta_2|1\rangle) = \alpha_1\beta_1|00\rangle + \alpha_1\beta_2|01\rangle + \alpha_2\beta_1|10\rangle + \alpha_2\beta_2|11\rangle$$

בתוצאה, ההסתברות שמצב xy יתקיים שווה להסתברות שהמצב x יתקיים בקיוביט הראשון וגם שהמצב y מתקיים בקיוביט השני, כלומר למכפלה שלהם.

נשים לב שאפשר לעשות טנזור גם על רגיסטרים בדרך דומה.

איך עובד מחשב קוונטי - שערים ומעגלים

שערים קוונטים

כמו ביטים, לקיוביטים יש שערים לוגיים שניתן להפעיל עליהם. לשערים האלה קוראים שערים קוונטים. כל שער קוונטי פועל על רגיסטר קוונטי אחד ומשנה את המקדמים וכתוצאה מכך את ההסתברויות של המצבים השונים בו. שער קוונטי הפועל על רגיסטר המכיל n קיוביטים מיוצג על ידי מטריצה בגודל 2^n על 2^n. הפעלה של שער קוונטי על רגיסטר שקולה להכפלה של הוקטור המייצג אותו במטריצה המייצגת את השער. המטריצה היא בגודל מתאים להכפלה בוקטור המתאר רגיסטר המורכב מ-n ביטים.

שערים חשובים על קיוביט בודד

לרגיסטר של קיוביט בודד נגדיר שער **NOT** כמטריצה:

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

כדי להפעיל את השער על הקיוביט $\alpha|0\rangle + \beta|1\rangle$ נחשב:

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}$$

קוראים לשער הזה NOT כי הוא הופך את הקיוביט $|0\rangle$ ל-$|1\rangle$ ולהיפך, כמו NOT רגיל. לשער זה מקובל לקרוא גם שער X. עוד דוגמאות לשערים שעובדים על קיוביט בודד שבהם נשתמש במאמר הם:

$$Y: \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} Z: \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

השערים X, Y, Z נקראים שערי פאולי והם בסיס לכל המטריצות האוניטריות המהוות שערים, לכן ניתן ליצור מהפעלת שלושתם כל שער (בדומה ל-NOT ו-OR בבריטים רגילים).

עוד שער שבו נשתמש במאמר הוא שער **Hadamard** או בקיצור שער $\boldsymbol{H}$. את השער תסמן המטריצה הבאה:

$$H:\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}$$

כאשר השער $\boldsymbol{H}$ יקבל קיוביט מהצורה $|0\rangle$ או $|1\rangle$ הוא יהפוך אותו לקיוביט שבהסתברות חצי יהיה 0 ובהסתברות חצי יהיה 1.

שערים חשובים על 2 קיוביטים או יותר

שערים יכולים גם לעבוד על רגיסטרים גדולים יותר, לדוגמא על 2 קיוביטים אפשר להפעיל שער **SWAP** המוגדר על ידי המטריצה:

$$\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

השער SWAP מחליף את הסדר של שני הקיוביטים ברגיסטר. אפשר לראות זאת על ידי הפעלה של השער:

$$\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\alpha_1\\ \alpha_2\\ \alpha_3\\ \alpha_4\end{bmatrix}=\begin{bmatrix}\alpha_1\\ \alpha_3\\ \alpha_2\\ \alpha_4\end{bmatrix}$$

התוצאה היא $\alpha_1|00\rangle+\alpha_3|01\rangle+\alpha_2|10\rangle+\alpha_4|11\rangle$. נשים לב שהשער החליף את המקדם של $|01\rangle$ עם המקדם של $|10\rangle$.

עוד שערים שבהם נשתמש במאמר הם שערים מבוקרים (**Controlled Gates**) שהם שערים העובדים על $n>1$ קיוביטים. שערים כאלה מיועדים להפעיל שער קוונטי קטן יותר על $n-1$ הקיוביטים האחרונים רק אם הקיוביט הראשון (הקיוביט המבקר, **control qubit),** הוא $|1\rangle$, ואחרת (אם הוא $|0\rangle$) - לא עושים כלום.

דוגמאות לשערי בקרה לשערים **NOT** ו-**Z** הם:

$$CZ:\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & -1\end{bmatrix}CNOT/CX:\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{bmatrix}$$

אם נכפיל אותם בסופרפוזיציה כלשהי נוכל לראות שהשערים המקוריים מופעלים רק אם הקיוביט המבקר הוא $|1\rangle$. לשער כללי A הפועל על n קיוביטים נגדיר את מטריצת CA כמטריצה הפועלת על $n+1$ קיוביטים מהצורה:

$$CA=\begin{bmatrix}I_{2^n} & 0\\ 0 & A\end{bmatrix}$$

אבל מה אם הקיוביט המבקר הוא לא $|1\rangle$ או $|0\rangle$ אלא שילוב שלהם? במקרה כזה השער מופעל רק על המקדמים של המצבים שבהם הקיוביט המבקר הוא $|1\rangle$.

לדוגמא אם ניקח את הרגיסטר:

$$\alpha_1|00\rangle + \alpha_2|01\rangle + \alpha_3|10\rangle + \alpha_4|11\rangle$$

נפעיל את **CNOT** עליו ויצא:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_4 \\ \alpha_3 \end{bmatrix}$$

כלומר יצא $\alpha_1|00\rangle + \alpha_2|01\rangle + \alpha_4|10\rangle + \alpha_3|11\rangle$ אז השער **NOT** הופעל רק על המצבים שבהם הקיוביט המבקר (הראשון) ברגיסטר הוא אחד.

טנזור של שערים

לשני שערים $A \in M_{nxn}, B \in M_{mxm}$ נגדיר את שער הטנזור שלהם כמטריצה $A \otimes B \in M_{n \cdot mxn \cdot m}$ שמקיימת $A|v\rangle \otimes B|u\rangle = (A \otimes B)|v \otimes u\rangle$ לכל v, u לדוגמא הפעלה של השער X על קיוביט אחד והפעלה של Y על הקיוביט השני שקולה להפעלת $X \otimes Y$ עליהם יחד. מחשבים את הייצוג של השער כך:

$$A \otimes B = \begin{bmatrix} A_{1\,1}B & A_{1\,2}B & \dots & A_{1\,n}B \\ A_{2\,1}B & A_{2\,2}B & & A_{2\,n}B \\ \dots & & \dots & \dots \\ A_{n\,1}B & A_{n\,2}B & \dots & A_{n\,n}B \end{bmatrix}$$

לדוגמא בשערים על קיוביט בודד:

$$X \otimes Y = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} = \begin{bmatrix} 0\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} & 1\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \\ 1\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} & 0\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & i & 0 \\ 0 & -i & 0 & 0 \\ i & 0 & 0 & 0 \end{bmatrix}$$

נשים לב שאם ניקח את הקיוביטים $a = \alpha_1|0\rangle + \alpha_2|1\rangle, b = \beta_1|0\rangle + \beta_2|1\rangle$ ונפעיל את X על a ואת Y על b ואז נחבר אותם לרגיסטר אחד נקבל:

$$X|a\rangle \otimes Y|b\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \otimes \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \cdot \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} = \begin{bmatrix} \alpha_2 \\ \alpha_1 \end{bmatrix} \otimes \begin{bmatrix} -i\beta_2 \\ i\beta_1 \end{bmatrix} = \begin{bmatrix} -i\alpha_2\beta_2 \\ i\alpha_2\beta_1 \\ -i\alpha_1\beta_2 \\ i\alpha_1\beta_1 \end{bmatrix}$$

נשים לב שאם נפעיל את $X \otimes Y$ על $a \otimes b$ נקבל:

$$(X \otimes Y)(|a\rangle \otimes |b\rangle) = \begin{bmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & i & 0 \\ 0 & -i & 0 & 0 \\ i & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha_1\beta_1 \\ \alpha_1\beta_2 \\ \alpha_2\beta_1 \\ \alpha_2\beta_2 \end{bmatrix} = \begin{bmatrix} -i\alpha_2\beta_2 \\ i\alpha_2\beta_1 \\ -i\alpha_1\beta_2 \\ i\alpha_1\beta_1 \end{bmatrix}$$

כלומר קיבלנו את מה שרצינו. בנוסף לשער A נסמן את A בחזקת:

$$A^{\otimes n} = A \otimes \cdots \otimes A \text{ (} n \text{ פעמים } A \otimes A\text{)} \text{ כ:} \ (\otimes n)$$

דוגמא לחזקת טנזור:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{\otimes 2} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \otimes \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a^2 & ab & ba & b^2 \\ ac & ad & bc & bd \\ ca & cb & da & db \\ c^2 & cd & dc & d^2 \end{bmatrix}$$

נשים לב שכאשר השער $H^{\otimes n}$ מקבל רגיסטר מהצורה e_i (כלומר רגיסטר קלאסי) הוא מחזיר רגיסטר קוונטי שבהסתברות שווה יתן את כל האפשרויות למחרוזות בינאריות בגודל n. בפרט, כאשר השער יקבל את הרגיסטר 0 הוא יחזיר כך שכל המצבים שווים ולא רק ההסתברויות. לדוגמא על רגיסטר של שני קיוביטים:

$$H^{\otimes 2} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}^{\otimes 2} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

כלומר כפי שראינו, כאשר נפעיל את השער $H^{\otimes 2}$ על רגיסטר קלאסי של שני ביטים נקבל הסתברויות שוות לכל האפשרויות למחרוזות בינאריות באורך n.

הכפלת שערים

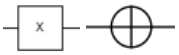
כאשר מפעילים על קיוביט שני שערים אחד אחרי השני, במקום להכפיל את הסופרפוזיציה במטריצה הראשונה ואז במטריצה השניה ניתן להכפיל את השערים ואז להכפיל את הסופרפוזיציה בהם, דבר זה נובע ישירות מאסוציאטיביות כפל מטריצות. אז למשל במקום להפעיל שער A על קיוביט ואז להפעיל שער B עליו ניתן להפעיל עליו את $A \cdot B$. לדוגמא במקום להפעיל את X ואז את Y נפעיל את:

$$Y \cdot X = \begin{bmatrix} -i & 0 \\ 0 & i \end{bmatrix}$$

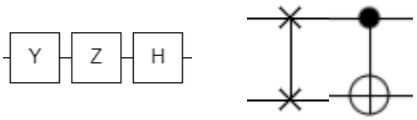
מעגל קוונטי

למערכת של כמה שערים קוונטיים נקרא מעגל קוונטי. נצייר מעגלים קוונטים בעזרת קווים וסמלים עליהם. כל קו בשרטוט מסמן קיוביט אחד ברגיסטר. למשל לרגיסטר המכיל שני קיוביטים המעגל הריק יסומן:

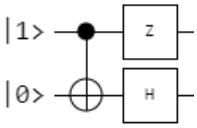
על הקווים נסמן את השערים שבהם משתמשים למעגל, מסודרים משמאל לימין כך שבצד שמאל של הקווים נשים את השער הפועל ראשון, ובצד ימין נשים את השער שפועל אחרון. את המצב הראשוני של הקיוביטים נשים משמאל לקווים. למשל להפעלה של **NOT** על קיוביט בודד נסמן באחת משתי הדרכים הבאות:



להפעלה של **CNOT,SWAP, H, Y,Z** נסמן בהתאמה:



כך שב-**CNOT** העיגול המלא ממוקם על הקו שמסמל את הקיוביט המבקר. דוגמה למעגל קוונטי:



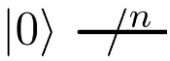
בדוגמה הרגיסטר מאותחל כ-$|10\rangle$. בהתחלה מופעל שער CNOT על שני הקיוביטים כך שהקיוביט המבקר הוא הראשון. אחריו יש שער $\boldsymbol{Z}$ שפועל על הקיוביט הראשון ושער $\boldsymbol{H}$ שפועל על השני. בדוגמה זו הפלט יהיה הרגיסטר:

$$\frac{-\sqrt{2}}{2}|10\rangle + \frac{\sqrt{2}}{2}|11\rangle$$

סימונים חשובים נוספים

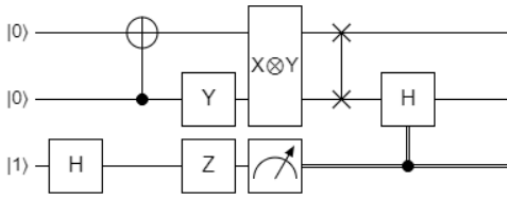
הסימון:

מסמן מדידה של קיוביט שגורמת לקריסתו והפיכתו לביט רגיל שיסומן כקו כפול במקום קו רגיל. הסימון:



מסמן שורה שבה מאתחלים n קיוביטים (במקרה שלנו את $0^{\otimes n}$), כלומר הפעלה של שער A על השורה הזאת תפעיל את השער $A^{\otimes n}$.

דוגמא נוספת לסימון מעגל:



כדי לבנות ולהתנסות עם מעגלים קוונטים ניתן להיעזר באתר: https://algassert.com/quirk

האלגוריתם של סימון

רקע

עד עכשיו הסברנו על מבנה המחשב הקוונטי, אבל מה השימושים שלו? בפרק זה נענה על שאלה זו באמצעות בעיה פשוטה שהפתרון הקוונטי שלה יעיל בהרבה מהפתרון הקלאסי.
בעיה זו נקראת הבעיה של סימון על שם דניאל סימון.

הבעיה

נתונה פונקציה $f:\{0,1\}^n \to \{0,1\}^n$ כלומר f פונקציה שמקבלת מחרוזת בינארית ומחזירה מחרוזת בינארית, כך שלכל s ($x,y: f(x) = f(y) \Leftrightarrow x \oplus y = s$ קבוע). המשימה - למצוא את s.

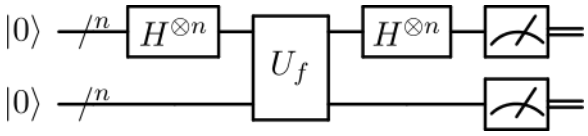
הפתרון הקלאסי

הפתרון הקלאסי הנאיבי הוא לעבור על כל המחרוזות הבינאריות עד מציאת שתי מחרוזות שמובילות לאותה מחרוזת, אם מצאנו שתיים כאלה x,y אז נחשב את $x \oplus y$ והרי זה s. אם לא מצאנו שתיים כאלה אז $s = 0^{\otimes n}$ (מחרוזת האפסים).

פתרון זה עובד בסיבוכיות אקספוננציאלית: $O(2^n)$ שכן במקרה הגרוע יש לבדוק את כל המחרוזות הבינאריות (2^n מחרוזות). שיפור לאלגוריתם הנאיבי הוא בדיקה של $\sqrt{2^n}$ מחרוזות בלבד (לפי פרדוקס יום ההולדת בהסתברות טובה מאוד נמצא את ההתנגשות, אם יש כזאת) אז הסיבוכיות של השיפור היא $O(\sqrt{2^n})$ כלומר $O(2^{n/2})$ פתרון זה כמובן לא טוב במיוחד ולכן נרצה למצוא פתרון טוב יותר.

הפתרון הקוונטי

ההצגה של החלק הקוונטי של האלגוריתם היא:



נסביר חלק חלק את כל החלקים באלגוריתם: בהתחלה ניקח שני רגיסטרים של אפסים בגודל n קיוביטים ($|0^{\otimes n}\rangle$). אחר כך על הרגיסטר הראשון נפעיל את השער $H^{\otimes n}$. לכל קיוביט $|k\rangle$, השער עושה:

$$H^{\otimes n}|k\rangle = \frac{1}{\sqrt{2^n}}\sum_{j=0}^{2^n-1}(-1)^{j\cdot k}|j\rangle$$

כשלכל j,k מחרוזות בינאריות נגדיר: $j\cdot k = j_1k_1 + j_2k_2 + \cdots + j_nk_n$ (j_i הוא האיבר במקום ה-i ב- j). לפי זה, כאשר מפעילים את השער על הרגיסטר $|0^{\otimes n}\rangle$ לכל המצבים יש מקדם זהה של $\frac{1}{\sqrt{2^n}}$ כלומר ההסתברויות שמצב זה יקרה שוות. (נשים לב שאם היינו מפעילים את השער על רגיסטר אחר שהוא לא $|0^{\otimes n}\rangle$ היו יוצאות הסתברויות שונות, למשל על $|1^{\otimes n}\rangle$ המקדמים היו יוצאים $\frac{\pm 1}{\sqrt{2^n}}$) אחרי זה מפעילים את השער U_f על שני הרגיסטרים. השער U_f מפעיל את f על הרגיסטר הראשון ושומר את התוצאה ברגיסטר השני. כלומר למקרה שלנו:

$$U_f\left(\left(\frac{1}{\sqrt{2^n}}\sum_{j=0}^{2^n-1}|j\rangle\right)\otimes|0^{\otimes n}\rangle\right) = \frac{1}{\sqrt{2^n}}\sum_{j=0}^{2^n-1}|j\rangle\otimes|f(j)\rangle$$

לאחר מכן, נמדוד את הרגיסטר השני. נקבל מספר המתקבל מהפעלתו של f על מחרוזת, אז נמדוד את הרגיסטר כך שתוצאת המדידה היא $f(r)$. שימו לב שאנחנו יודעים רק את ערך $f(r)$ ולא את ערך r. לאחר המדידה, כל המצבים שלא תואמים את המדידה "נמחקים". לכן נשארים רק עם המצבים שבהם הרגיסטר השני הוא $f(r)$. המצבים האלה הם רק $|r\rangle|f(r)\rangle$ ו-$|r\oplus s\rangle|f(r)\rangle$. לפי זה הרגיסטר הראשון הופך ל: $\frac{1}{\sqrt{2}}|r\rangle + \frac{1}{\sqrt{2}}|r\oplus s\rangle$. נפעיל על הרגיסטר הראשון את $H^{\otimes n}$, ונחשב את התוצאה:

$$H^{\otimes n}\left(\frac{1}{\sqrt{2}}|r\rangle + \frac{1}{\sqrt{2}}|r\oplus s\rangle\right) = H^{\otimes n}\left(\frac{1}{\sqrt{2}}|r\rangle\right) + H^{\otimes n}\left(\frac{1}{\sqrt{2}}|r\oplus s\rangle\right) =$$

$$\frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2^n}}\sum_{j=0}^{2^n-1}(-1)^{j\cdot r}|j\rangle\right) + \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2^n}}\sum_{j=0}^{2^n-1}(-1)^{j\cdot(r\oplus s)}|j\rangle\right) =$$

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{j=0}^{2^n-1}\left((-1)^{j\cdot r} + (-1)^{j\cdot(r\oplus s)}\right)|j\rangle = \frac{1}{\sqrt{2^{n+1}}}\sum_{j=0}^{2^n-1}(-1)^{j\cdot r}\left(1 + (-1)^{j\cdot s}\right)|j\rangle$$

שימו לב שאתם מבינים את כל המעברים, הם חשובים.

לכל j שבו $j \cdot s$ אי זוגי, $1 + (-1)^{j \cdot s} = 0$ ולכן המקדם של $|j\rangle$ הוא 0. לפי זה, נשארים רק עם מצבים $|j\rangle$ שבהם $j \cdot s$ זוגי, ולכן כשמודדים את הרגיסטר הראשון, המחרוזת j שמקבלים בהכרח מקיימת ש- $j \cdot s = 0(mod\ 2)$. נחזור על התהליך הזה $n-1$ פעמים ונקבל $n-1$ מחרוזות. $y_1, y_2 \cdots y_{n-1}$ כך שלכל $1 \le i < n$ מתקיים $y_i \cdot s = 0(mod\ 2)$. נסמן y_i^j המחרוזת y_i במקום j. לפי ההגדרה שלנו של כפל, מתקיים שלכל i: $y_i^1 \cdot s^1 + \cdots + y_i^n s^n = 0(mod\ 2)$. לפי זה, אפשר ליצור מערכת של $n-1$ משוואות עם n מעל $\mathbb{Z}_2$:

$$\begin{bmatrix} y_1^1 & y_1^2 & \cdots & y_1^n \\ y_2^1 & y_2^2 & & y_2^n \\ \cdots & & \cdots & \cdots \\ y_{n-1}^1 & y_{n-1}^1 & \cdots & y_{n-1}^n \end{bmatrix} \begin{bmatrix} s^1 \\ s^2 \\ \cdots \\ s^n \end{bmatrix} = 0$$

נשים לב שב-$\mathbb{Z}_2$ שני וקטורים הם תלויים לינארית רק אם הם שווים, ולכן $y_1, y_2 \cdots y_{n-1}$ תלויים לינארית רק אם בשתי חזרות שונות על התהליך במדידה האחרונה יוצא אותו מצב. יש 2^{n-1} אפשרויות למחרוזות בינאריות שיכולות לצאת מהמדידה ו$\frac{n-1}{2}$ אפשרויות לוקטורים שיכולים להיות שווים אז ההסתברות ששני וקטורים יהיו שווים היא: $\frac{\frac{n-1}{2}}{2^{n-1}}$ כלומר $\frac{(n-1)(n-2)}{2^n}$. נשים לב שההסתברות שהוקטורים $y_1, y_2 \cdots y_{n-1}$ הם תלויים לינארית דומה להסתברות שקיימים שני וקטורים כך שהפעלת f על שניהם תוציא את אותה תוצאה. כלומר בהסתברות טובה המחרוזות בלתי תלויות לינארית ובמקרה שהן תלויות אפשר לבצע את האלגוריתם שוב, בממוצע ייקח פעם אחת. נחזור על האלגוריתם עד שהמחרוזות יהיו בלתי תלויות לינארית ואז מרחב הפתרונות למערכת יהיה ממימד 1. במקרה שלנו בגלל שאנחנו ב$\mathbb{Z}_2$ יהיו שני פתרונות.

נפתור את מערכת המשוואות ונקבל שני פתרונות כפי שציפינו. הפתרונות יהיו 0 (כי המערכת הומוגנית) ועוד מחרוזת s. s' חייב להיות פתרון למערכת אז הוא חייב להיות שווה לאחד הפתרונות: 0 או s'. כדי לוודא האם $s' = s$ נבדוק האם $f(s') = f(0)$. אם כן - אז $s = s'$. אם לא - אז $s = 0^{\otimes n}$.

ההוכחה שאין פתרון קלאסי מסיבוכיות פולינומיאלית

נתחיל עם אלגוריתם קלאסי שקרא לפונקציה k פעמים על הערכים $x_1, \cdots, x_k$. נניח שהאלגוריתם יודע לבדוק אם הוא כבר מצא את s בסיבוכיות $O(1)$. כלומר, נסתכל על המקרה הטוב ביותר שבו הוא יכול לעבור על כל i, j ולבדוק אם $f(x_i) = f(x_j)$ ב $O(1)$. נחשב את הסיכוי שהאלגוריתם יצליח למצוא את s בקריאה ה$k+1$ לפונקציה בהנחה שהוא לא הצליח עד אז.

בשלב זה ידוע שלכל i, j לא מתקיים $f(x_i) = f(x_j)$. זה פוסל עד $\frac{k}{2}$ אפשרויות ל-s. בגלל זה נשארות $2^n - 1 - \frac{k}{2}$ אפשרויות. נקרא לפונקציה עם המחרוזת x_{k+1}. הקריאה החדשה לפונקציה מספקת לנו מספיק מידע בשביל לגלות את s אם ורק אם קיים i שמקיים $x_i \oplus x_{k+1} = s$.

בגלל זה, יש עד k אפשרויות ל-$x_1 \oplus x_{k+1},\cdots,x_k \oplus x_{k+1}$: s מתוך $2^n - 1 - \frac{k}{2}$ שעבורן יש לנו מספיק מידע בשביל לקבוע את s. כלומר ההסתברות שלאחר הוספת x_{k+1} יש לנו מספיק מידע כדי לגלות את s (כלומר מצאנו התנגשות) היא $\frac{k}{2^n-1-\frac{k}{2}}$. בגלל זה, בעזרת חסם האיחוד, עבור עד m קריאות לפונקציה, הסיכוי למצוא את s גדול או שווה ל:

$$\sum_{k=1}^{m} \frac{k}{2^n - 1 - \frac{k}{2}} \leq \sum_{k=1}^{m} \frac{k}{2^n - k^2} \leq \frac{m^2}{2^n - m^2}$$

כדי להשיג הסתברות קבועה לפחות צריך ש $\frac{m^2}{2^n-m^2} \geq c$ עבור c קבוע כלשהו אז:

$$\frac{m^2}{2^n - m^2} \geq c$$
$$m^2 \geq c \cdot 2^n - c \cdot m^2$$
$$(c+1)m^2 \geq c \cdot 2^n$$
$$m^2 \geq 2^n$$
$$m \geq 2^{n/2}$$

אז חייבים לקרוא לפונקציה לפחות מספר אקספוננציאלי של פעמים, כלומר כל פתרון קלאסי הוא מסיבוכיות $\Omega(2^n)$.

אלגוריתם גרובר

הבעיה והפתרון הקלאסי

נתונה פוקציה $f:\{1,\cdots,N\} \to \{1,0\}$, וידוע לנו שרק אינדקס אחד מקיים $f(i) = 1$. המטרה של האלגוריתם היא למצוא את האינדקס הזה. ברור שכל פתרון של מחשב קלאסי חייב לעבור על $O(N)$ אינדקסים בשביל למצוא את האינדקס הנכון בהסתברות קבועה לפחות.

אינטואיציה לפתרון הקוונטי

האלגוריתם פועל על $n = log_2(N)$ קיוביטים, ולכן הוקטורים המייצגים אותם הם בגודל N, אבל הם תמיד נשארים בתוך תת מרחב במימד 2, אז לאינטואיציה הגיאומטרית מספיק להתייחס לשני מימדים. נתחיל בלסמן את האינדקס הרצוי - w ואת הוקטור המאונך ל$|w\rangle$ במרחב - $|w'\rangle$.

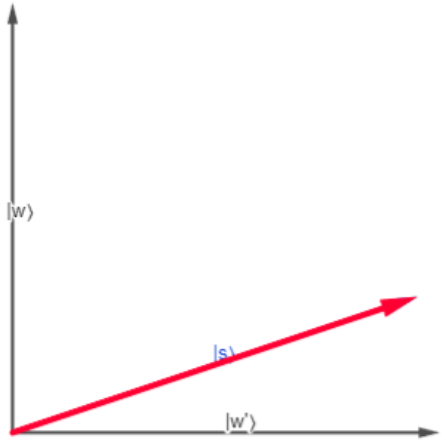
נגדיר את השער $U_{|k\rangle}$ לכל רגיסטר $|k\rangle$:

$$U_{|k\rangle} = 2|k\rangle\langle k| - I$$

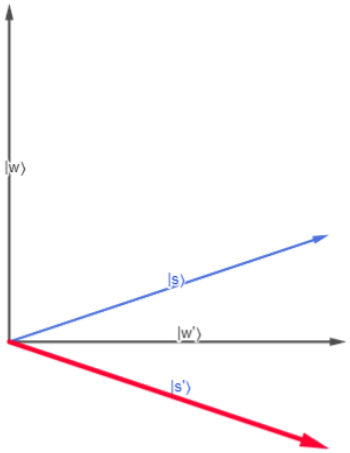
נשים לב ש$U_{|k\rangle}$ מגדיר שיקוף סביב הוקטור $|k\rangle$.

נאתחל את הרגיסטר שלנו כוקטור $|s\rangle$ רנדומלי במרחב הדו מימדי. האלגוריתם מפעיל על הרגיסטר את השער $U_{|w'\rangle}$ ואז $U_{|s\rangle}$. אחרי הפעולות האלו, הרגיסטר נשאר במרחב, ומתקרב לוקטור $|w\rangle$.

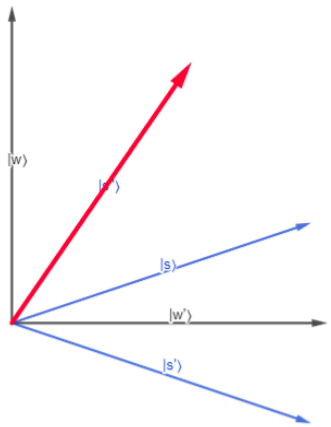
השרטוטים מראים את המצב של הרגיסטר (החץ האדום) במהלך כל חזרה. $|s\rangle$ הוא הקיוביט ההתחלתי:



לאחר שיקוף $|s\rangle$ סביב $|w\rangle$ נקבל:



כך ש$|s'\rangle$ הוא הוקטור החדש. נשקף את $|s'\rangle$ סביב $|s\rangle$ ונקבל:



הפתרון הקוונטי

האלגוריתם פועל על ידי שער אורקל. השער נותן לאלגוריתם גישה לפונקציה f ומוגדר כך:

$U_f|x\rangle = |x\rangle$ אמ"מ $f(|x\rangle) = 0$, ואחרת $U_f|x\rangle = -|x\rangle$. (אפשר להוכיח ש $U_f = U_{|w'\rangle}$ עבור וקטורים במרחב הדו מימדי, אבל לא נראה את זה פה).

האלגוריתם פועל על $n = log_2(N)$ קיוביטים המאותחלים במצב שבו לכל המצבים יש מקדם שווה (הם מאותחלים כך בעזרת הפעלת השער $H^{\otimes n}$ על הרגיסטר $0^{\otimes n}$). נגדיר את המצב ההתחלתי הזה $|s\rangle$. נשים לב שהרגיסטר תמיד נשאר במרחב $sp\{|s\rangle, |w\rangle\}$, ולכן נייצג את השערים U_f ו-$U_{|s\rangle}$ מעל הבסיס $B = \{|s\rangle, |w\rangle\}$:

$$[U_{|s\rangle}]_B = \begin{bmatrix} -1 & 0 \\ 2/\sqrt{N} & 1 \end{bmatrix}$$

$$[U_f]_B = \begin{bmatrix} -1 & -2/\sqrt{N} \\ 0 & 1 \end{bmatrix}$$

באלגוריתם מפעילים את השערים אחד אחרי השני הרבה פעמים. נסמן את מספר החזרות k, אז כפי שראינו הפעלת השערים שקולה להכפלה במטריצה:

$$\left([U_{|s\rangle}]_B [U_f]_B\right)^k = \begin{bmatrix} 1 & 2/\sqrt{N} \\ -2/\sqrt{N} & 1-4/N \end{bmatrix}^k$$

כדי להקל על החישוב של החזקה, נלכסן את המטריצה. הערכים העצמיים הם: $e^{2i\cdot arcsin(\frac{1}{\sqrt{N}})}$ ו-$e^{-2i\cdot arcsin(\frac{1}{\sqrt{N}})}$. הוקטורים העצמיים המתאימים להם הם:

$$\begin{bmatrix} -i \\ e^{i\cdot arcsin(1/N)} \end{bmatrix} \quad \begin{bmatrix} i \\ e^{-i\cdot arcsin(1/N)} \end{bmatrix}$$

כלומר נלכסן בעזרת הבסיס:

$$P = \begin{bmatrix} -i & i \\ e^{i\arcsin\left(1/\sqrt{N}\right)} & e^{-i\arcsin\left(1/\sqrt{N}\right)} \end{bmatrix}$$

אז:

$$[U_{|s\rangle}]_B [U_f]_B = P \begin{bmatrix} e^{2i\arcsin\left(1/\sqrt{N}\right)} & 0 \\ 0 & e^{-2i\arcsin\left(1/\sqrt{N}\right)} \end{bmatrix} P^{-1}$$

כלומר:

$$\left([U_{|s\rangle}]_B [U_f]_B\right)^k = P \begin{bmatrix} e^{2ik\arcsin\left(1/\sqrt{N}\right)} & 0 \\ 0 & e^{-2ik\arcsin\left(1/\sqrt{N}\right)} \end{bmatrix} P^{-1}$$

לפי זה המצב הסופי של הרגיסטר (אחרי הפעלת השער) הוא:

$$P\begin{bmatrix} e^{2ik\arcsin\left(1/\sqrt{N}\right)} & 0 \\ 0 & e^{-2ik\arcsin\left(1/\sqrt{N}\right)} \end{bmatrix} P^{-1\prime}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \sqrt{\frac{N}{N-1}}\begin{bmatrix} \sin\left(2k\arcsin\left(1/\sqrt{N}\right)\right) \\ \cos\left((2k+1)\arcsin\left(1/\sqrt{N}\right)\right) \end{bmatrix}$$

נחשב את אורך ההיטל של הרגיסטר על $|w\rangle$ כדי למצוא את המקדם של w ברגיסטר:

$$\langle |w\rangle, \sqrt{\frac{N}{N-1}} sin(2k\ arcsin(\frac{1}{\sqrt{N}})) \cdot |w\rangle + \sqrt{\frac{N}{N-1}} cos((2k+1)\ arcsin(\frac{1}{\sqrt{N}})) \cdot |s\rangle\rangle$$

נפשט ויצא:

$$\sqrt{\frac{N}{N-1}} sin(2k\ arcsin(\frac{1}{\sqrt{N}})) + \sqrt{\frac{1}{N-1}} cos((2k+1)\ arcsin(\frac{1}{\sqrt{N}}))$$

נשים לב שהביטוי תמיד גדול מ-$sin(2k\ arcsin(\frac{1}{\sqrt{N}}))$, ולכן במדידה ההסתברות שמה שיצא הוא w היא לפחות $sin^2(2k\ arcsin(\frac{1}{\sqrt{N}}))$.

נשים לב שעבור k קרוב ל-$\frac{\pi}{4arcsin(1/\sqrt{N})}$ ההסתברות קרובה ל $sin^2(\frac{\pi}{2}) = 1$ ולכן האלגוריתם עובד בהסתברות טובה.

סיכום

אז מה ראינו במאמר? התחלנו מלגלות מהם המחשבים הקוונטים, איך הם עובדים ומה הם עושים, התעמקנו במרכיבי היסוד שלהם - קיוביטים בסופרפוזיציה וראינו כיצד ניתן לחבר קיוביטים. המשכנו עם שערים קוונטים, שערים הפועלים על קיוביטים, ראינו כיצד ניתן לשרשר אותם ולהפעיל אותם בו זמנית, דיברנו על מבנה המעגל הקוונטי והראנו סימולים רבים שעוזרים להצגה ויזואלית של אלגוריתמים.

אחר כך צללנו לשתי בעיות מוכרות - בעיית סימון ובעיית החיפוש וראינו כיצד מחשבים קוונטים יכולים לעזור לנו לפתור את הבעיות הללו ביעילות טובה משמעותית מהיעילות הטובה ביותר כאשר משתמשים במחשבים קלאסיים.

ראינו שלמרות שהמחשבים הקוונטים הם כלי חזק מאוד והם ויכולים לפצח בעיות שלא אפשריות עם מחשבים קלאסיים, צריך לזכור שאלגוריתמים קוונטים הם מאוד מסובכים ולא טריוויאלים, ואי אפשר ליעל כל בעיה בעזרת מחשבים קוונטים.

יש בעיות רבות שמחשבים קוונטים, מעצם היותם לא דטרמיניסטים, יקרים וקשים לתחזוקה, לא פותרים ולכן מחשבים קוונטים הם לא יהוו תחליף למחשבים קלאסיים, גם לא בעתיד הרחוק.

על המחברים

עומר גולן, בן 17 מכפר הס, לומד לתואר ראשון בתוכנית אודיסאה באוניברסיטת תל אביב, מפתח תוכנה בחברת XM Cyber.

גיל בוכבינדר, בן 16 מהרצליה, לומד בתוכנית אודיסאה באוניברסיטת תל אביב לקראת תואר ראשון.

ברצוננו להודות לד"ר **שלומי בוטנרו** ו**אורן רנרד** על הליווי המקצועי והעצות המועילות במהלך כתיבת המאמר.

מקורות ולקריאה נוספת

- https://batistalab.com/classes/v572/Mosca.pdf
- https://gadial.net/2019/02/11/what_is_quantum_computer/
- https://sites.math.washington.edu/~morrow/336_20/papers20/chris.pdf
- https://medium.com/@russfein/the-quantum-leaps-beginner-guide-to-entanglement-f4282dc73d04
- https://medium.com/quantum-untangled/simons-algorithm-quantum-algorithms-untangled-62c8f81ed27a
- https://courses.cs.washington.edu/courses/cse599d/06wi/lecturenotes8.pdf
- https://learning.quantum.ibm.com/tutorial/grovers-algorithm
- https://www.qi.damtp.cam.ac.uk/files/QIC-9.pdf
- https://courses.cs.washington.edu/courses/cse599d/06wi/lecturenotes11.pdf
- https://wybiral.github.io/quantum/
- https://algassert.com/quirk
- https://www.youtube.com/watch?v=bYvfGNiIcmc