

СИМВОЛЬНЕ МОДЕЛЮВАННЯ КВАНТОВОГО АЛГОРИТМУ ПОШУКУ ГРОВЕРА НА ДВОХ КУБІТАХ: QUIRK + MAPLE

Г. П. Чуйко

Чорноморський національний університет імені Петра Могили,

кафедра комп'ютерної інженерії

E-mail: henadiy.chuyko@chmnu.edu.ua

© Чуйко Г. П., 2025

Запропонована символна модель квантового алгоритму пошуку Гровера для найпростіших систем з двох кубітів, яка паралельно використовує можливості системи комп'ютерної математики Maple і квантового браузерного симулятора Quirk. Модель створена з акцентом на застосування в інженерній освіті для сучасних курсів з квантових обчислень. Вона дозволяє узагальнення на системи з більшим числом кубітів, а також на випадки декількох цільових станів, і може бути базовою для розробки лабораторних робіт. Модель є однією з перших спроб вийти за межі орієнтованих на Python квантових симуляторів. Maple, мова програмування якої спирається на Java, чудово взаємодіє з квантовим симулятором Quirk, який базується на JavaScript.

Ключові слова: квантовий алгоритм Гровера, Maple, Quirk, символне моделювання, симуляція квантових алгоритмів.

Вступ

Алгоритм Гровера (також GSA від Grover Search Algorithm) є “детективом” серед родини відомих квантових алгоритмів, які демонструють так звану квантову перевагу порівняно з класичними алгоритмами. Він дає змогу знайти потрібний елемент у невідсортованому наборі даних, маючи квадратичне прискорення ($O(\sqrt{N})$) відносно класичного аналога ($O(N)$).

Практичне значення алгоритму Гровера охоплює різні сфери, серед яких виокремлюють такі:

- Пошук у базах даних: алгоритм можна адаптувати до задач типу “знайди x , який відповідає умові $f(x) = 1$ ”.
- Оптимізація та контроль обмежень: використовується для пошуку в графах, комбінаторній оптимізації – всюди, де є перевірка правильності рішення.
- Криптографія: зменшує стійкість симетричних шифрів, що змінює уявлення про безпеку в епоху квантових обчислень.
- Підготовка до постквантової доби: допомагає моделювати сценарії, в яких квантові атаки на лінії квантових комунікацій можуть бути реальністю, що особливо цікаво у сфері безпеки даних.

Навчально-методичне значення алгоритму Гровера визначається трьома головними факторами. По-перше, він досить простий у реалізації, має чітку геометричну інтерпретацію, доволі легко візуалізується, що робить його ідеальним прикладом для викладання квантових алгоритмів. По-друге, алгоритм добре працює в середовищах типу Quirk, Qiskit, Maple, отже, підходить для демонстрації квантових переваг студентам. Нарешті, він дає змогу змінювати кількість розв'язків, структуру квантового оракула, кількість ітерацій, що робить його платформою для самостійних експериментів.

Запропонована демонстраційна символна модель алгоритму Гровера здійснена на двох кубітах із синтетичним використанням середовища симуляції квантових схем Quirk [1] і відомої системи комп'ютерної математики Maple 2024 [2].

Огляд літературних джерел

Алгоритму Гровера [3] присвячено безліч публікацій різного типу: від академічних монографій до оглядових блогів. Найбільш детальна, нова і науково-технічне вивірена інформація щодо алгоритму, на думку автора, представлена у джерелі [4]. Тут читач може знайти формулювання проблеми пошуку, сам алгоритм крок за кроком, його геометричну інтерпретацію, аналіз складності, оптимальну кількість ітерацій, літературні посилання та релевантні ресурси. Менш детально, але в контексті квантового паралелізму, із прикладом роботи, алгоритм описаний у науковій статті [5]. Саме квантовий паралелізм на основі суперпозицій станів – ключовий момент для розуміння переваг алгоритму Гровера. Більш популярна і доступна нефахівцям версія з акцентом на прикладне значення алгоритму викладена у блозі [6]. Він пояснює алгоритм через приклад пошуку в телефонному довіднику.

Харківські автори [7], аналізуючи інструментарій сучасних квантових обчислень, в одному рядку наводять такі засоби квантових симуляцій, як Cirq, Quirk, Qutip, вважаючи їх дещо менш потужними ніж ProjectQ на базі мови Python [8]. Ця теза є щонайменше дискусійною, зважаючи на різні освітні цілі цих симуляторів. Проте домінування Python, наприклад у зв'язці з Qiskit у галузі квантових обчислень та квантових симуляцій, виглядає доволі виразним [9].

Накопичено достатній досвід символьного моделювання, зокрема і галузі квантових явищ, засобами системи комп'ютерної математики Maple [10], мова програмування якої базується на Java. Перевагою Maple є можливість символьного моделювання. Поширена думка, що всі квантові симулятори (наприклад, Quirk – браузерний симулятор, написаний на JavaScript) від початку “заточені” лише під Python, ймовірно, є упередженням, яке варто розвіяти.

Перевагами Quirk [1] є, по-перше, візуальна інтерактивність: користувач просто перетягує елементи схеми – без кодування (drag-and-drop інтерфейс), маючи миттєвий відгук у реальному часі, а також візуалізацію заплутаності та суперпозицій, що для квантових симуляторів є рідкістю. По-друге, цей симулятор не потребує інсталяції чи реєстрації, працюючи на будь-якому пристрої з JavaScript. Симулятор має бібліотеку алгоритмів та потужну Web-підтримку, яку забезпечує блог його автора [11].

Постановка задачі

Метою цього дослідження є символьне моделювання алгоритму пошуку Гровера для найпростішої квантової системи з двох кубітів за допомогою середовища Maple з одночасним використанням Quirk як візуального інструменту схемного представлення алгоритму. Для досягнення поставленої мети необхідно виконати такі завдання:

- Формалізацію алгоритму Гровера шляхом символьних операцій на матрицях Адамара операторів квантових фазових оракулів та дифузійного перетворення.
- Верифікацію символьної моделі візуальними засобами Quirk.
- Аналіз еволюції початкового квантового стану і перевірка коректності результатів пошуку відповідно до теоретичних передбачень.

Результати дослідження

1. Геометрія оператора Гровера

Як вхідні дані для алгоритму Гровера, припустимо, що маємо функцію $f: \{0, 1, \dots, j, \dots, N-1\} \rightarrow \{0, 1\}$. За аналогією з “неструктурованою базою даних” домен $\{0, 1, \dots, j, \dots, N-1\}$ представляє індекси елементів бази даних, а $f(x_j) = 1$ тоді і лише тоді, коли дані, на які вказує x_j , задовольняють критерій пошуку. Припустимо спочатку (пізніше це обмеження можна буде зняти), що лише один індекс j задовольняє умову: $f(j) = 1$ and $f(i) = 0$ if $i \neq j$, і цей індекс j відомий.

Сенс алгоритму Гровера полягає в ідеї “посилення амплітуди” цільового стану (j) завдяки амплітудам інших станів суперпозиції. Геометрично алгоритм передбачає обертання поточного вектора стану системи кубітів у напрямку точно до цільового стану (рух по найкоротшому шляху забезпечує оптимальність алгоритму Гровера). Процес, який реалізує схему алгоритму, веде до вибухового зростання спочатку малої амплітуди цільового стану, що здатне швидко довести її до реально спостережуваних величин.

На рис. 1 $|\psi\rangle$ – це вектор початкового стану, який представлений як рівноцінна (усі компоненти рівно імовірні) суперпозиція всіх базисних векторів. Зокрема, така суперпозиція охоплює також цільовий стан $|\beta\rangle$, а також

$|\alpha\rangle$ – суперпозицію всіх базисних станів, за винятком вектора розв’язку $|\beta\rangle$. Вектори $|\alpha\rangle$ та $|\beta\rangle$ разом утворюють повний набір базисних векторів, отже, вони є ортогональними один до одного (рис. 1).

Нехай $\theta/2$ – це кут між початковим станом $|\psi\rangle$ та станом $|\alpha\rangle$. Тоді можна застосувати операцію фазового оракула U_f до початкового стану $|\psi\rangle$. Ця операція змінює знак компоненти $|\beta\rangle$ на протилежний, водночас діючи на інші стани $|\alpha\rangle$ як тотожність, тобто не змінюючи їх, і так обертає стан $|\psi\rangle$ на кут θ . Початковий стан $|\psi\rangle$ переходить у стан $O|\psi\rangle$ (рис. 1). Далі інший оператор U_g , який називають дифузійним перетворенням, обертає вектор $O|\psi\rangle$ на кут 2θ , переводячи його у стан $G|\psi\rangle$. Добуток операторів $U_g U_f$ називають оператором Гровера.

Початковий стан $|\psi\rangle$ вважається рівноцінною суперпозицією базисних станів, отже, всі амплітуди його 2^n компонентів однакові та дорівнюють $1/\sqrt{2^n}$. Далі неважко переконавшись, що кут $\theta/2$ на рис. 1 дорівнює $\arcsin(\frac{1}{\sqrt{2^n}})$.

Враховуючи, що кожна ітерація (окрема операція) повертає вектор початкового стану в напрямку цільового стану на кут 2θ , неважко знайти потрібну кількість ітерацій. Ця величина дається виразом:

$$Q_{iter} = \left\lceil \frac{\pi}{4 \cdot \arcsin(\frac{1}{\sqrt{2^n}})} - \frac{1}{2} \right\rceil, \quad (1)$$

де квадратні дужки символізують цілу частину числа. Доволі часто у виразі (1) покладають $\arcsin(x) \approx x$ та нехтують дробом $1/2$, чого не варто робити, оскільки це може давати порівняно велику помилку, зокрема за малої кількості кубітів. Рис. 2 показує результати оцінок за виразом (1) для систем з малою кількістю кубітів.

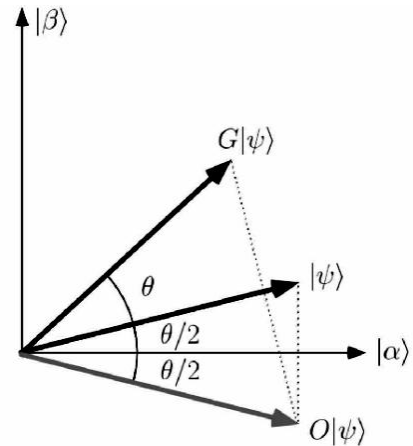


Рис. 1. Геометрична інтерпретація алгоритму Гровера [12]

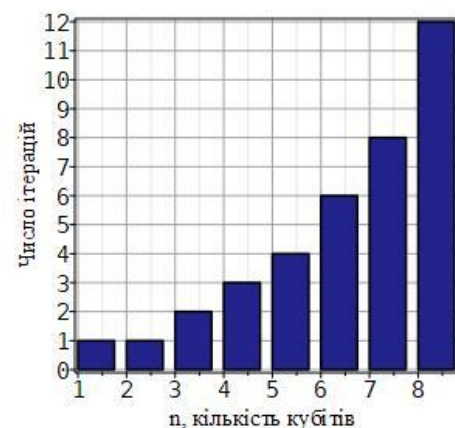


Рис. 2. Необхідне число ітерацій залежно від кількості кубітів в алгоритмі Гровера

Як показує рис. 2, у простій системі з двох кубітів вже перша ітерація є достатньою, тоді як для більших систем ($n > 2$) кількість необхідних ітерацій доволі швидко зростає.

2. Допоміжні квантові оператори (вентилі, гейти) в алгоритмі Гровера

Почнемо з опису базисних станів кубіту

$$|u\rangle, |d\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (2)$$

Далі наведемо перелік однокубітних вентилів: тотожної матриці, матриці Адамара та матриць Паулі

$$Id, H, X, Y, Z = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (3)$$

Далі перелік двокубітних вентилів (контрольовані NOT, Z та антиконтрольований Z):

$$CNOT, CZ, aCZ = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4)$$

3. Приклад алгоритму пошуку Гровера на системі з двох кубітів

Конкретизуємо параметри системи: кількість кубітів ($n = 2$), індекс цільового стану ($j = 3$) та кількість потрібних ітерацій ($Q(n) = 1$). Останній параметр розраховується з виразу (1) або береться з діаграми (рис. 2). Припустимо, що вхідні стани обох кубітів є однаковими $|u\rangle \equiv |0\rangle$. Тоді стан рівноцінної суперпозиції утворюється двома вхідними вентилями Адамара

$$|\psi\rangle = (H \cdot u) \otimes (H \cdot u), \quad (5)$$

де $\otimes$ – символ тензорного добутку Кронекера. Вираз (5) легко узагальнюється на складніші системи ($n > 2$), у правій частині (2) просто додається множників.

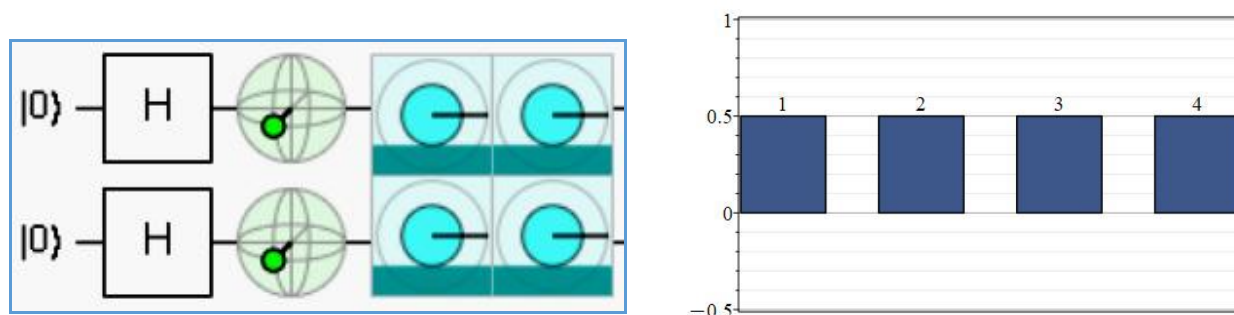


Рис. 3. Фрагмент квантової схеми Quirk з вхідними гейтами Адамара, сферами Блоха кубітів та дисплеєм амплітуд компонент суперпозиції (ліворуч) а також стовпчикова діаграма амплітуд компонент рівноцінної суперпозиції (праворуч), отримана в Maple з виразу (5)

Наступний крок – моделювання фазового оракула U_f . Його задача проста – змінити знак обраної цільової компоненти (нагадаємо, що $j = 3$) рівноцінної суперпозиції (рис. 3) на протилежний. Структура оракула доволі проста: це діагональна матриця з єдиним негативним елементом $u_{jj} = -1$. Подібну матрицю неважко створити, користуючись властивостями символу Кронекера $\delta_{j,j}$. Схемні рішення для чотирьох фазових оракулів ($j = 1, 2, 3, 4$) наведені на рис. 4.

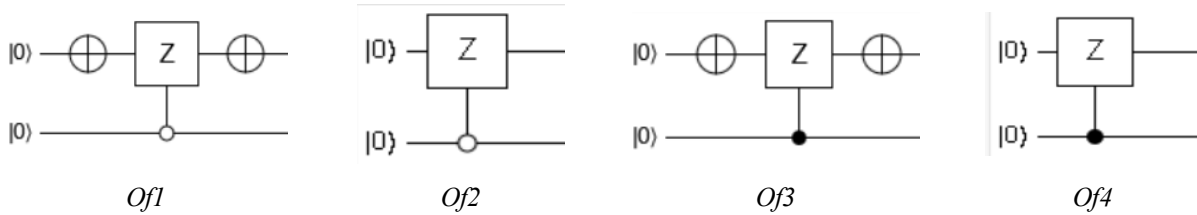


Рис. 4. Схемні рішення у Quirk для фазових оракулів Гровера у двокубітній системі. Кубіти на схемах Quirk нумеруються від нижнього до верхніх, як поверхи в будинках

Перевіримо коректність структури матриці оракула Of3. Відповідно до рис. 4 він задається таким добутком допоміжних операторів

$$Of3 = (Id2 \otimes X) \cdot CZ \cdot (Id2 \otimes X) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (6)$$

Далі моделюється так зване дифузійне перетворення, тобто оператор U_g , який на схемі з двома кубітами задається блоком, котрий показано на рис. 5.

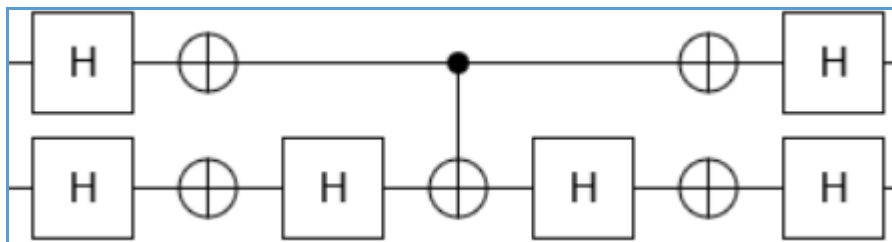


Рис. 5. Блок Quirk для дифузійного перетворення у системі з двох кубітів

Моделювати цей блок аналогічно фазовому оракулу, зважаючи на схему (рис. 5), в принципі можна, але незручно, через громіздкість такого виразу внаслідок розміру блока ($size = 11$) і його глибини ($depth = 7$). Простіше скористатися виразом [4] для дифузійного перетворення:

$$U_g = 2|\psi\rangle\langle\psi| - (Id2 \otimes Id2) = \frac{1}{2} \begin{pmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{pmatrix} \quad (7)$$

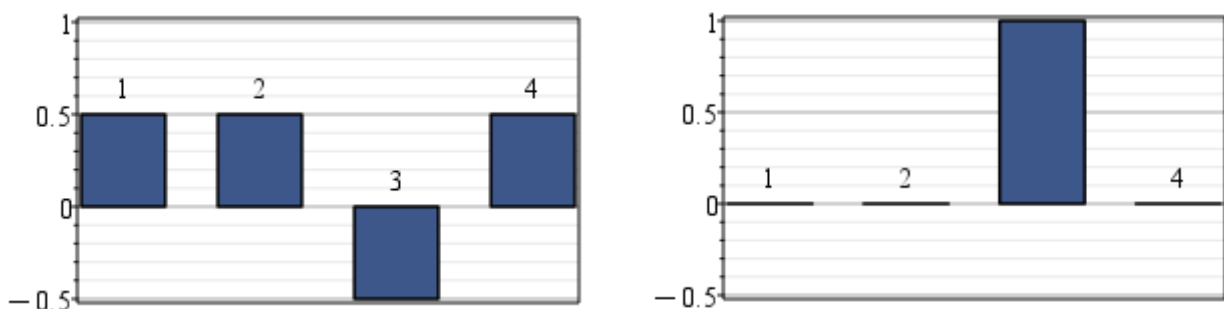


Рис. 6. Амплітуди компонентів суперпозиції станів після дії квантового фазового оракула (ліворуч) та після повної операції Гровера (праворуч)

На рис. 6 наведено амплітуди компонентів рівноцінної суперпозиції (рис. 3) після дії фазового оракула та після повної операції Гровера. Як видно з рис. 6, вже першої ітерації достатньо в системі двох кубітів для максимального збільшення амплітуди цільового стану завдяки іншим компонентам.

Рис. 7 також наочно переконує, що для двокубітної системи вже перша ітерація забезпечує значну перевагу амплітуди та ймовірності (яка визначається квадратом амплітуди) для обраного стану в рівноцінній суперпозиції.

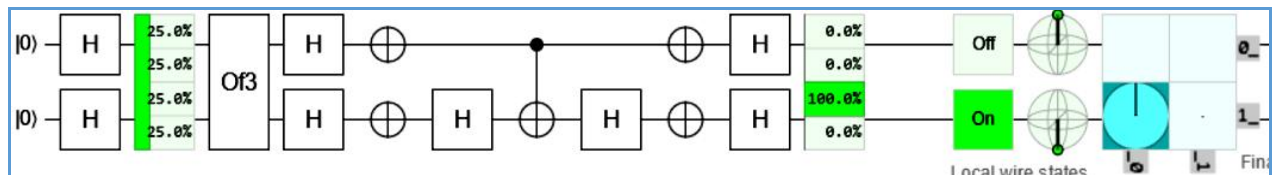


Рис. 7. Працездатна схема алгоритму Гровера для двокубітної системи у Quirk: фазовий оракул позначено блоком Of3 (рис. 4), цільовим є стан $|10\rangle$

Упроваджені в Україні останнім часом пілотні університетські курси зі сфери квантових обчислень, наприклад київський [13], львівський [14] або чернівецький [15] демонструють виразне лекційно-теоретичне спрямування. Певним винятком є УКУ (Український католицький університет), де нещодавно почали активно уживати PennyLane [16] – середовище з відкритим Python-кодом для квантових обчислень. Водночас, на думку автора, якісна і сучасна інженерна освіта потребує лабораторних практикумів, і наведене вище дослідження може і має бути базою для таких розробок.

Висновки

1. Символьне моделювання алгоритму пошуку Гровера для найпростішої квантової системи з двох кубітів засобами середовища Maple із паралельним використанням Quirk як візуального інструменту схемного представлення алгоритму реалізоване в цій публікації, отже, поставлена мета досягнута. Всі три завдання (формалізація алгоритму засобами Maple, верифікація символічної моделі Maple засобами Quirk, аналіз та візуалізація еволюції початкового квантового стану системи в ході алгоритму) виконані. Принагідно варто зауважити, що розроблена модель досить просто узагальнюється на мультикубітні системи та на випадок декількох цільових станів, що може бути здійснено у подальших дослідженнях.

2. Дослідження є одним з перших вдалих спроб вийти за межі Python-орієнтованих квантових моделей та симуляторів: Maple, з мовою програмування, заснований на Java, чудово взаємодіє з квантовим симулятором Quirk, який базується на JavaScript.

3. З практичного погляду наведена модель насамперед має вартість для сфери вищої освіти, зокрема для розробки вітчизняних лабораторних курсів та практикумів з квантових обчислень, незалежних від хмарних сервісів, доступ до яких ми не контролюємо.

Список літератури

1. *Algorithmic Assertion*. (n.d.). *Welcome to Quirk: A drag-and-drop quantum circuit simulator*. URL: <https://algassert.com/quirk>.
2. Maplesoft. (n.d.). *Maple Fundamentals Guide*. URL: <https://www.maplesoft.com/support/training/PDF/MapleFundamentalsGuide.pdf>.
3. Grover, L. K. (1996). *A fast quantum mechanical algorithm for database search*. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing* (pp. 212–219). ACM. DOI: <https://doi.org/10.1145/237814.237866>.
4. Microsoft. (2025). *Theory of Grover's search algorithm*. *Azure Quantum documentation*. URL: <https://learn.microsoft.com/en-us/azure/quantum/concepts-grovers>.

5. Кравець, П. (2025). Аналіз математичних моделей квантового паралелізму. *Information Systems and Networks*, 17, 248–268. DOI: <https://doi.org/10.23939/sisn2025.17.248>.
6. Все про квантові обчислення: історія, можливості та перспективи. (n.d.). URL: <https://gigacloud.ua/articles/vse-pro-kvantovi-obchyslennya-istoriya-mozhlyvosti-ta-perspektyvy/>.
7. Bozhko, I., Chetverykov, G., & Karatai, O. (2020). The investigation of modern quantum computing tools. *Бионика интеллекта*, 2(95), 15–20. DOI: [https://doi.org/10.30837/bi.2020.2\(95\).02](https://doi.org/10.30837/bi.2020.2(95).02).
8. ProjectQ Framework. (n.d.). ProjectQ: An open source software framework for quantum computing. URL: <https://github.com/ProjectQ-Framework/ProjectQ>.
9. Kawai, R. (n.d.). A first step to quantum computation with Qiskit. URL: <https://kawaihome.link/jbooks/qcomp-short/q1gates/r.html>.
10. Maplesoft. (n.d.). Maple Application Center: Quantum Mechanics. URL: <https://www.maplesoft.com/applications/Search.aspx?q=Quantum%20Mechanics>.
11. Gidney, C. (n.d.). Algorithmic Assertion. URL: <https://algassert.com/about.html>.
12. Maji, R., Panigrahi, P. K., & Behera, B. K. (2018). Solving linear systems of equations by using the concept of Grover's search algorithm: An IBM Quantum Experience. *arXiv preprint arXiv:1801.00778*. DOI: <https://doi.org/10.48550/arXiv.1801.00778>.
13. Карлаш, Г. Ю. (2018). Квантові інформаційні системи: Навчальний посібник для спеціальності “Прикладна фізика та наноматеріали”. Київ: КНУ імені Тараса Шевченка, факультет радіофізики, електроніки та комп’ютерних систем.
14. Крохмальський, Т. Є. (2018). Вступ до квантових обчислень: Навчальний посібник. Львів: ЛНУ імені Івана Франка.
15. Остапов, С. Е., & Добровольський, Ю. (2021). Квантова інформатика та квантові обчислення: Навчальний посібник. Чернівці: Чернівецький національний університет.
16. Xanadu Quantum Technologies. (2025). PennyLane documentation. URL: <https://docs.pennylane.ai/en/stable/index.html>.

SYMBOLIC MODELING OF THE QUANTUM GROVER SEARCH ALGORITHM ON TWO QUBITS: QUIRK + MAPLE

G. P. Chuiko

Petro Mohyla Black Sea National University,
Department of Computer Engineering

© Chuiko G. P., 2025

A symbolic model of the quantum Grover search algorithm for the simplest two-qubit system is proposed, which leverages the capabilities of the Maple computer algebra system and the Quirk quantum browser simulator. The model is developed with a focus on its application in modern quantum computing courses in engineering education. It enables generalization to systems with more qubits and cases involving multiple target states. It can serve as a foundation for developing laboratory work. This model is among the first attempts to go beyond the limitations of Python-based quantum simulators. Maple, a Java-based programming language, interacts effectively with the Quirk quantum simulator, which is based on JavaScript.

Keywords: Grover’s quantum algorithm, Maple, Quirk, simulation of quantum algorithms, symbolic modeling.