

SOFTWARE ENGINEERING

ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

УДК 004.942+004.415:530.145

DOI:10.15330/itee.2025.3.03

В. М. ТКАЧУК, канд. фіз.-мат. наук

ГІБРИДНИЙ КЛАСИЧНО-КВАНТОВИЙ АЛГОРИТМ ТА ЙОГО РЕАЛІЗАЦІЯ НА РЕАЛЬНОМУ КВАНТОВОМУ КОМП'ЮТЕРІ

Вступ

Еволюційні алгоритми відносяться до класу евристичних методів оптимізації, в основу роботи яких покладено принципи еволюції природних біологічних систем. Вони працюють із множиною потенційних розв'язків, еволюціонуючи їх за допомогою операторів відбору, схрещування та мутації [1]. У їх класичній реалізації мінімальною одиницею інформації є біт, який може приймати тільки два значення: 0 та 1. Особливо ефективними вони є до задач оптимізації, де необхідно знайти глобальний екстремум функції пристосованості у складному пошуковому просторі. Класичні підходи оптимізації з великою імовірністю зупиняються у локальних мінімумах [2].

Квантові генетичні алгоритми ґрунтуються на принципах квантових обчислень у поєднанні з ідеями класичних генетичних алгоритмів [3, 4]. При цьому мінімальною одиницею інформації є кубіт, що перебуває в стані суперпозиції двох базових станів ($|0\rangle$ та $|1\rangle$):

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle, \text{ де } |\alpha|^2 + |\beta|^2 = 1 \quad (1)$$

Тут α та β визначають амплітуду ймовірності перебування кубіта у відповідному стані. Завдяки цьому забезпечується потенційна перевага паралелізму обчислень та одночасної оцінки всіх можливих станів кубіта. Ймовірнісний механізм квантового вимірювання забезпечує ефективний глобальний пошук, що в поєднанні з традиційними еволюційними технологіями дає швидку збіжність навіть при невеликих розмірах популяції [5].

Розробка та дослідження такого класу алгоритмів орієнтована на задачі оптимізації без принципової можливості їх практичної реалізації на квантовому комп'ютері. Їх виконання, в тому числі і базові підходи та квантові оператори, орієнтовані під реалізацію чисто засобами класичних обчислювальних систем. Фізична неможливість клонування квантових станів взагалі унеможливорює організацію традиційних еволюційних процесів. Актуальність розробки такого алгоритму стає особливо актуальною на фоні успіхів у розробці як доступних ефективних квантових стимуляторів, так і робочих вільно доступних багатокубітних квантових комп'ютерів [6].

Для доступу до квантових комп'ютерів розроблено проект із відкритим кодом Qiskit, написаний на Python, та включає в себе набір інструментів для написання програм на рівні квантових схем. На сьогодні з використанням хмарних технологій доступними є багатокубітні квантові процесори компанії IBM (наприклад, `ibm_fez` (156 кубітів), `ibm_marrakesh` (156 кубітів), `ibm_torino` (133 кубіти)), так і ряд квантових ідеальних та реальних стимуляторів (модуль Qiskit Aer) [6].

Зважаючи на обмеженість доступу до таких квантових комп'ютерів важливу роль у аналізі роботи квантових генетичних алгоритмів відіграють симулятори, які дозволяють проводити масштабні експерименти без необхідності доступу до реальних квантових процесорів та впливу апаратного шуму. Ідеальний симулятор AerSimulator (метод `statevector` або `unitary`) забезпечує точне відтворення квантової динаміки без втрат, що є критично важливим для перевірки теоретичної коректності роботи алгоритму та аналізу впливу окремих квантових операторів на ефективність збіжності еволюційного процесу. Водночас

рад бекендів (FakeMarrakesh, FakeBrisbane, FakeKyoto, тощо) дозволяють моделювати реалістичний шум, помилки гейтів, помилки зчитування та топологію з'єднань реальних комп'ютерних пристроях IBM, що дає змогу оцінити стійкість алгоритму до апаратних обмежень ще до запуску на реальному обладнанні [6].

Основними параметрами роботи таких симуляторів у Qiskit є: кількість запусків (shots) - статистична точність вимірювань (зазвичай 1000–100000); метод симуляції (statevector, density_matrix, tensor_network тощо); рівень оптимізації (optimization_level від 0 до 3); включення шуму (noise_model); та максимальна кількість кубітів (обмежується обсягом оперативної пам'яті та обчислювальними можливостями комп'ютера). Правильний вибір цих параметрів дозволяє балансувати між точністю, швидкістю обчислень та реалістичністю моделювання, що є ключовим для розробки та надійного тестування гібридних квантово-класичних алгоритмів.

Для реалізації квантового генетичного алгоритму (GIGQ) на квантовому комп'ютері необхідно адаптувати як його структуру в цілому, так і реалізацію базових квантових операторів. В першу чергу це зумовлено неможливістю клонування невідомих квантових станів кубітів для організації еволюції популяції в часі [7].

Основні ідеї та підходи до реалізації такого гібридного алгоритму були запропоновані автором в [8], а нижче вони отримають подальший розвиток та аналіз із врахуванням додаткових можливостей, зумовлених появою нових ефективних квантових симуляторів, використання сучасних GPU-бібліотек для прискорення роботи основних квантових операцій [9].

Аналіз літературних публікацій, проведених вище, дозволяє сформулювати наступну порівняльну таблицю (табл. 1).

Таблиця 1

Порівняльний аналіз різних підходів до реалізації генетичних алгоритмів

	Класичний GA (CGA)	Квантово-інспірований GA (QIGA)	Гібридний квантовий GA (HQGA)
Спосіб кодування особин	Бінарний	Кубіти, моделювання на класичному комп'ютері	Справжнє квантове представлення кубітів та виконання операторів на квантовому комп'ютері
Еволюційні оператори	Мутація, кросовер, селекція (класичні)	Обертання кутів квантово-інспіроване, але класичне)	Квантова томографія, R_y - гейти, квантовий шум як мутація
Переваги	Простота реалізації, ефективний для локального пошуку	Швидша збіжність для багатопараметричних задач	Експоненційний паралелізм, стійкість до шуму
Обмеження	Повільна збіжність, чутливість до розміру популяції	Обмежено класичними обчисленнями, немає справжньої суперпозиції	Обмеження кількості кубітів, квантовий шум та помилки у роботі квантових комп'ютерах
Збіжність	Залежить від розміру поколінь (~100–1000)	Швидша збіжність (~50–500 поколінь)	Найшвидша збіжність (~20–100 поколінь) [8]

Гібридний квантовий генетичний алгоритм для організації його роботи на квантовому комп'ютері IBM-Q

Алгоритм повністю відповідає принципам гібридної архітектури, запропонованим у попередніх роботах [8, 10, 11], але адаптований під сучасні можливості Qiskit Aer (ідеальний та реальний із шумовим режимом).

Використовується гібридна класично-квантова архітектура реалізації генетичного алгоритму: ключові еволюційні оператори, такі як генерація початкової популяції та її еволюція в часі, повністю реалізовано з використанням квантового комп'ютера. Інші, допоміжні обчислення, такі як оцінка функції пристосованості та пошук найкращої особини, проводяться на класичному комп'ютері. При цьому один кубіт у суперпозиції квантових

станів з використанням квантової томографії визначає одну особину популяції, а квантовий реєстр реалізує всю квантову популяцію. Алгоритм буде апробовано на прикладі задач функціональної оптимізації, де квантова суперпозиція дозволяє ефективно досліджувати багатомірний простір пошуку (рис. 1).

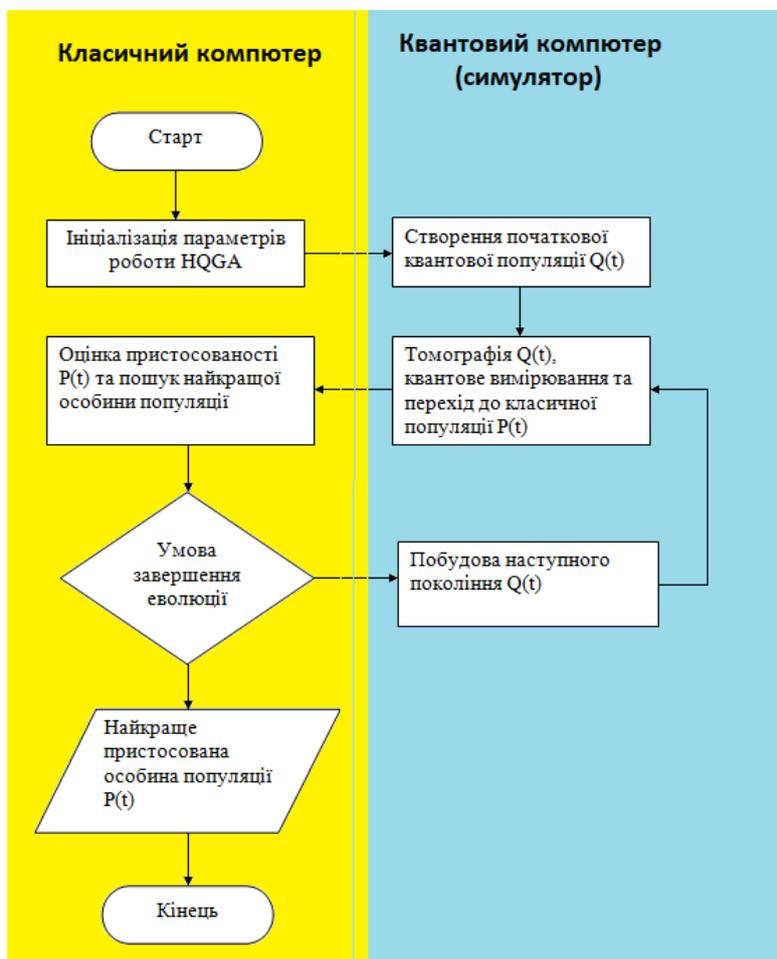


Рис. 1. Гібридний класично-квантовий генетичний алгоритм

Ключовими особливостями запропонованої архітектури є:

- одна особина = один кубіт (невеликий квантовий реєстр – квантова популяція);
- квантова популяція являє собою суперпозицією станів кубітів ($Q(t)$);
- відновлення квантової популяції здійснюється за допомогою квантової томографії;
- класична популяція $P(t)$ отримується в результаті квантового вимірювання;
- основні квантові еволюційні оператори:
 - мутація - природна через імовірнісний характер томографії та квантовий шум;
 - адаптивне керування кутами обертання (квантовий оператор R_y) на основі результатів томографії попереднього покоління.

Початкова квантова популяція $Q(0)$ генерується відповідно наступного квантового протоколу (рис. 2).

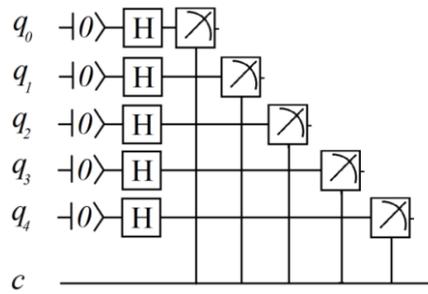


Рис. 2. Квантовий протокол початкової квантової популяції $Q(0)$

Томографія стану кубіта виконана відповідно до квантового протоколу (рис. 3).

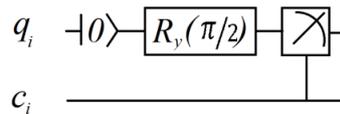
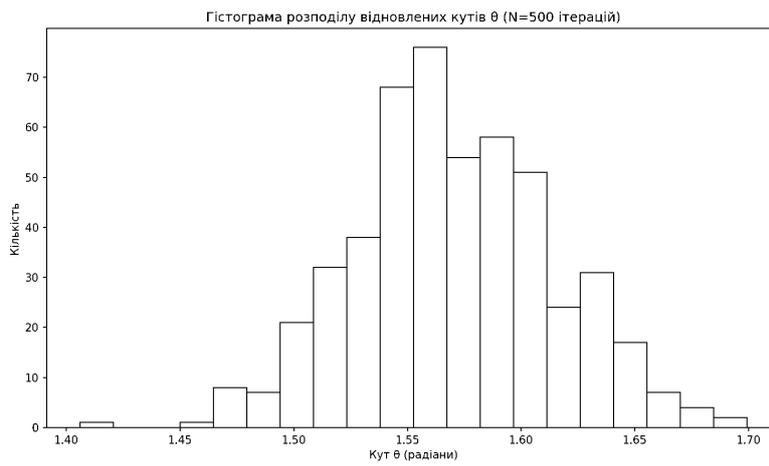
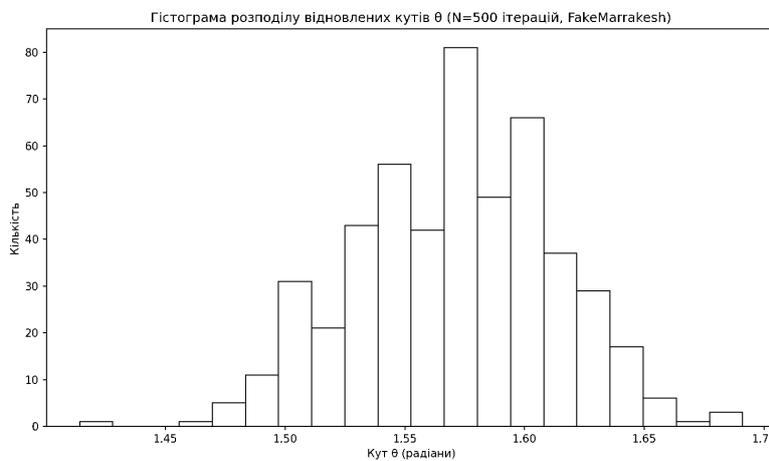


Рис. 3. Томографія стану кубіта

Точність відновлення стану визначається як типом симулятора, так і розміром квантового ансамблю (рис. 4 та табл. 2).



а)



б)

Рис. 4. Томографія стану кубіта при розмірі квантового ансамблю у 500 кубітів:
а – AerSimulator; б- FakeMarrakesh

Точність відновлення стану кубіта в залежності від розміру квантового ансамблю

Розмір квантового ансамблю	AerSimulator	FakeMarrakesh	FakeBrisben
50	1.566 ± 0.0494	1.563 ± 0.0472	1.575 ± 0.0471
100	1.572 ± 0.0485	1.570 ± 0.0464	1.568 ± 0.0466
200	1.570 ± 0.0453	1.570 ± 0.0453	1.568 ± 0.0448
500	1.571 ± 0.0424	1.571 ± 0.0453	1.570 ± 0.0449

Для організації еволюційного процесу відновлення стану популяції проведена згідно наступного протоколу (рис. 5).

$$q_i \rightarrow |0\rangle \rightarrow R_y(\theta_i)$$

Рис. 5. Протокол еволюційного процесу відновлення стану популяції

Кут θ_i в операторі квантового гейту $R_y(\theta_i)$ визначається результатами квантової томографії попереднього покоління та значенням кута θ_{opt} для найкращої особини популяції:

$$\theta_i = \begin{cases} \theta_i + \Delta\theta, & \text{при } \theta_i < \theta_{opt}, \\ \theta_i - \Delta\theta, & \text{при } \theta_i > \theta_{opt}. \end{cases} \quad (2)$$

Величина приросту кута $\Delta\theta$ є параметром роботи алгоритму. Для визначення його оптимального значення було проведено додаткові дослідження на прикладі функції:

$$f(x) = \sin(x) + \sin\left(\frac{10}{3}x\right), x \in [-2.5, 7], \quad (3)$$

із глобальним максимумом $f(2.297) = 1.728$.

Динаміка еволюції найкращої особини квантової популяції розміром 5 кубітів представлено на рис. 6.

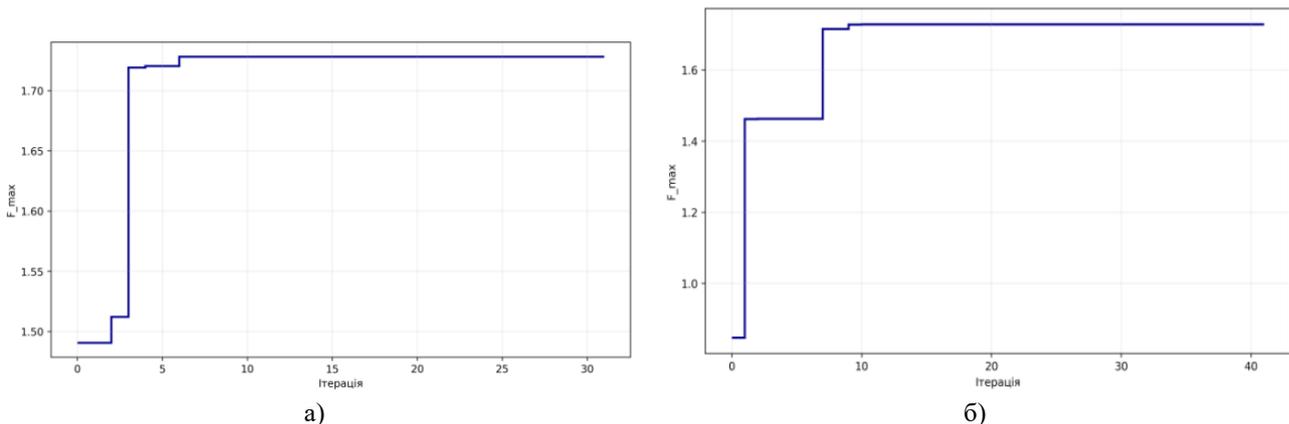


Рис. 6. Еволюція в часі найкращої особини квантової популяції: а – AerSimulator; б- FakeMarrakesh

Для ефективного глобального пошуку, як показали проведені числові експерименти, значення $\Delta\theta$ у випадку нескладних функцій повинно орієнтовно бути рівним $2 \cdot \Delta\theta_{avr}$ - отриманих в результаті квантової томографії середньоквадратичних відхилень. Аналіз результатів моделювань із різними типами симуляторів показує, що не залежно від згенерованої випадковим чином початкової популяції вона завжди, на протязі не більше 20

циклів еволюції, попадає в глобальний максимум. Остаточні уточнені значення для даних параметрів можна отримати при оптимізації функцій декількох змінних зі складнішою топологією.

Результати числового експерименту

Зважаючи на обмежені обчислювальні можливості (розмір квантової системи не може перевищувати 20 кубітів) для апробації пропонованого алгоритму було використано наступні тестові функції, розмірність яких не перевищує 4 [2]:

- Rastrigin Function:

$$F_1(x) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cdot \cos(2\pi x_i)], \quad x_i \in [-5.12, 5.12]$$

з глобальним мінімумом $F_1 = 0$ при $x = (0, \dots, 0)$.

- Ackley Function:

$$F_2(x) = -20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)} + 22.718, \quad x_i \in [-5.0, 5.0]$$

з глобальним мінімумом $F_2 = 0$ при $x = (0, \dots, 0)$.

- Sphere Function:

$$F_3(x) = \sum_{i=1}^n x_i^2, \quad x_i \in [0.0, 0.0]$$

з глобальним мінімумом $F_3 = 0$ при $x = (0, \dots, 0)$.

Функції є складними для традиційних градієнтних методів через велику кількість локальних мінімумів і плато. Тому вони підходять для перевірки ефективності гібридних квантово-генетичних алгоритмів. Числові експерименти проведено з використанням ідеального симулятора AerSimulator (метод statevector) та шумових бекендів (FakeMarrakesh, FakeManila), що дозволяють при моделюванні врахувати реальні квантові помилки та шум. Параметри експерименту були наступними: розмір популяції: 5 кубітів на один вимір; максимальна кількість поколінь - до 100; кількість спроб на один етап еволюції: 100–200.

Експеримент проведено на комп'ютері з процесором Intel Core i7, 32 ГБ ОЗУ та симулятором AerSimulator без GPU-прискорення. Результати моделювання приведені в табл. 3.

Таблиця 3

Експериментальні результати оптимізації тестових функцій

Функція	Симулятор	Середня кількість поколінь до збіжності	Середнє найкраще значення функції	Стійкість до початкових умов
Rastrigin Function	AerSimulator	18–32	0.004 ± 0.002	висока (з 30 запусків 28 досягли значення < 0.01)
	FakeMarrakesh	25–38	0.12 ± 0.08	розкид результатів, але зберігається збіжність у 85 % випадків
Ackley Function	AerSimulator	22–41	0.008 ± 0.004	висока (з 30 запусків 28 досягли значення < 0.01)
	FakeMarrakesh	26–51	0.35 ± 0.15	розкид результатів, але зберігається збіжність у 91 % випадків
Sphere Function	AerSimulator	12–25	0.001 ± 0.0005	Найшвидша збіжність серед усіх функцій завдяки гладкості поверхні
	FakeMarrakesh	28–35	0.09 ± 0.039	Найменший вплив квантового шуму

Як видно із приведених результатів для стійкості розв'язку бажано додати оператор квантової катастрофи [5]. Для зменшення залежності оптимального значення кута $\Delta\theta$ від вибраного стимулятора (чи квантового комп'ютера) краще застосувати його адаптивну поведінку в процесі еволюції. Імовірнісний характер вимірювання та шум реального комп'ютера чи шумового бекенду вводять стохастичність, одна вона не завжди приводить до покращення ефективності роботи алгоритму. Гібридний квантовий алгоритм демонструє в 3–5 разів швидшу збіжність у ідеальному режимі та зберігає прийнятну ефективність при квантовому шумі.

Порівняння ефективності роботи HQGA зі стандартним CGA (популяція 50 особин, бінарне кодування, мутація 0.01, кросовер 0.7) приведено в таб.4.

Таблиця 4

Ефективність роботи HQGA у порівнянні з класичним CGA

	Алгоритм	Середня кількість поколінь до збіжності
Rastrigin Function	CGA	85 - 150
	HQGA	18 - 32
Ackley Function	CGA	110 - 200
	HQGA	22 - 41
Sphere Function	CGA	60 - 120
	HQGA	12 - 25

Із приведених даних числового експерименту можна зробити наступні висновки:

- результати підтверджують перспективність гібридних класично-квантових підходів для задач функціональної оптимізації в умовах обмеженої кількості кубітів та наявності квантового шуму;

- HQGA показує високу швидкість збіжності до глобального екстремуму навіть при дуже малій популяції (5 кубітів), що є перевагою над класичними підходами;

- шумовий режим (шумові бекенди чи симулятори) дещо погіршує збіжність розв'язку, але сам алгоритм залишається стійким;

- в ідеальному режимі HQGA перевершує класичний GA в 3–5 разів за швидкістю збіжності завдяки квантовій суперпозиції та паралельному пошуку.

Висновки

Отримані результати дозволяють зробити наступні висновки:

- сучасні інструменти симуляції (Qiskit Aer, CUDA-Q) дозволяють ефективно моделювати квантові алгоритми (до 100+ кубітів у спеціальних випадках), але для повноцінного їх використання необхідне апаратне прискорення операцій на GPU та подальша оптимізація самих алгоритмів. Класична симуляція квантових алгоритмів не лише не втрачає актуальності, а навпаки – стає все більш потужним і необхідним компонентом квантових досліджень у перехідний період до реальних квантових комп'ютерів;

- класична симуляція квантових алгоритмів залишається критично важливим інструментом на сучасному, коли реальні квантові процесори обмежені кількістю кубітів (до 100–127 на 2025 рік), високим рівнем шуму та помилок. Фундаментальне обмеження – експоненційне зростання розмірності простору Гілберта $O(2^N)$ – робить повну симуляцію на основі вектора стану (statevector) практично неможливою для $N > 60$ –65 кубітів навіть на найпотужніших комп'ютерах;

- відсутність апаратного прискорення на GPU є суттєвим обмеженням поточної реалізації HQGA у Qiskit Aer.

Список літератури:

1. Holland J. H. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Ann Arbor, MI : *University of Michigan Press*, 1975. 211 p.
2. Surjanovic S., Bingham D. *Virtual Library of Simulation Experiments: Test Functions and Datasets*. Simon Fraser University, 2023. URL: <https://www.sfu.ca/~ssurjano/optimization.html>.
3. A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *IEEE Int. Conf. Evol. Computation*, Nagoya, Japan. IEEE, 1996. doi: <https://doi.org/10.1109/icc.1996.542334>
4. G. Zhang, "Quantum-inspired evolutionary algorithms: a survey and empirical study," *J. Heuristics*, vol. 17, no. 3, pp. 303–351, Jun. 2010. doi: <https://doi.org/10.1007/s10732-010-9136-0>.
5. H. Wang, J. Liu, J. Zhi, and C. Fu, "The Improvement of Quantum Genetic Algorithm and Its Application on Function Optimization," *Math. Problems Eng.*, vol. 2013, pp. 1–10, 2013. doi: <https://doi.org/10.1155/2013/730749>.
6. IBM Quantum Experience Dashboard. URL: <https://quantum.ibm.com/>.
7. W. K. Wootters and W. H. Zurek, "A single quantum cannot be cloned," *Nature*, vol. 299, no. 5886, pp. 802–803, Oct. 1982. doi: <https://doi.org/10.1038/299802a0>
8. В. Ткачук, "Квантовий генетичний алгоритм та його реалізація на квантовому комп'ютері," *Model. Control Inf. Technol.*, no. 5, pp. 137–139, Nov. 2021. doi: <https://doi.org/10.31713/mcit.2021.45>
9. Yovchev D. "Simulations of quantum algorithms using the Qiskit Aer library and the matrix product state method on CPU and GPU," *Telecommunication inf. technol.*, vol. 88, no. 3, pp. 72–79, 2025. doi: <https://doi.org/10.31673/2412-4338.2025.038707>
10. Ткачук В.М. *Можливості реалізації генетичного алгоритму на реальному квантовому комп'ютері // Кібернетичні моделювання та системи штучного інтелекту : матеріали конф. (КМОСС-2021)*. Черкаси : ЧДТУ, 2021. С. 74.
11. E. Ballinas and O. Montiel, "Hybrid Quantum Genetic Algorithm for the 0-1 Knapsack Problem in the IBM Qiskit Simulator," *Computación y Sist.*, vol. 26, no. 2, Jun. 2022. doi: <https://doi.org/10.13053/cys-26-2-4253>.

Надійшла до редколегії 18.11.2025

Відомості про автора:

Ткачук Валерій Михайлович – кандидат фіз.-мат. наук, доцент кафедри інформаційних технологій, Карпатський національний університет імені Василя Стефаника / Vasyl Stefanyuk Carpathian National University, Україна; email: valerij.tkachuk@cnu.edu.ua; ORCID: <https://orcid.org/0000-0001-7366-1676>