



КОМП'ЮТЕРНІ НАУКИ

DOI <https://doi.org/10.32782/2220-8674-2026-16-1-28>

УДК 004.42

А. І. Безверхий, доцент

ORCID: 0000-0002-0819-3690

І. А. Скрипник, канд. ф.-м. наук, доцент

ORCID: 0000-0002-9175-2683

О. О. Жуков, PhD

ORCID: 0009-0004-3683-2527

Запорізький національний університет

email: anatolbezv@gmail.com, sia@zsea.edu.ua

**ІНТЕГРАЦІЯ КВАНТОВИХ ОБЧИСЛЕНЬ В ЕКСПЕРТНІ СИСТЕМИ
З НЕЧІТКОЮ ЛОГІКОЮ**

Анотація. В роботі представлено концептуальну архітектуру та програмну реалізацію інтелектуальної системи, що об'єднує три парадигми: нечітку логіку, експертні системи та квантові алгоритми. За допомогою інструментарію scikit-fuzzy на мові Python реалізовано експертну систему з нечіткою логікою. Вона обробляє неточні вхідні дані і генерує тривалість поливу. Для вибору оптимальної стратегії в умовах невизначеності використовується симуляція квантового алгоритму за допомогою бібліотеки Qiskit від IBM. Інтеграція зазначених підходів дозволяє створити систему прийняття рішень, де нечітка логіка забезпечує гнучкість, експертна система – надійність, а квантові обчислення – оптимізацію стратегії. Представлений приклад демонструє можливості побудови гібридних систем для вирішення багатofакторних завдань у таких галузях, як точне землеробство, автоматизація виробничих процесів та управління ресурсами.

Ключові слова: гібридна інтелектуальна система, нечітка логіка, експертна система, квантові обчислення, оптимізація.

Постановка проблеми. Сучасні експертні системи, що базуються на нечіткій логіці, досягли значних успіхів у моделюванні людського мислення та прийнятті рішень в умовах невизначеності. Вони ефективно обробляють лінгвістичні змінні, імітуючи здатність експертів робити висновки на основі неповних або неточних даних. Однак їхня продуктивність стикається з фундаментальними обмеженнями класичних обчислень, особливо при зростанні складності предметної області.

Основні проблеми класичних експертних систем з нечіткою логікою включають:

1. Експоненційна складність: Кількість нечітких правил і обсяг обчислень для їх обробки зростають експоненційно зі збільшенням кількості вхідних змінних. Це призводить до значного сповільнення роботи системи та робить її непрактичною для вирішення задач у реальному часі.

2. Обмеження оптимізації: Процес налаштування функцій належності та оптимізації бази правил є NP-складною задачею.

3. Труднощі з паралелізмом: Хоча деякі аспекти нечіткого висновку можна розпаралелити, існуючі архітектури не дозволяють повною мірою використовувати паралелізм для одночасного аналізу можливих комбінацій правил і станів системи.

З іншого боку, квантові обчислення пропонують принципово нові підходи до обробки інформації, використовуючи принципи суперпозиції та квантової запутаності. Це відкриває



потенціал для подолання зазначених обмежень. Квантові алгоритми, такі як алгоритм Гровера, можуть забезпечити квадратичне прискорення для задач пошуку, а інші підходи пропонують нові методи для вирішення складних оптимізаційних проблем.

У зв'язку з цим виникає наукова проблема: розробка нових архітектур та алгоритмів для експертних систем, які б інтегрували переваги нечіткої логіки для роботи з невизначеністю та потужність квантових обчислень для експоненційного прискорення обробки інформації та виявлення синергетичного ефекту від такої інтеграції. Недостатньо досліджені питання представлення нечітких множин у вигляді кубітів, реалізації операцій нечіткої логіки на квантових вентиліях та створення гібридних квантово-класичних алгоритмів для нечіткого логічного висновку.

Аналіз останніх досліджень. Нечітка логіка в наш час залишається залишається актуальним інструментом для створення інтелектуальних систем. Класична стаття від засновника нечіткої логіки Лотфі Заде [1] пояснює, що суть нечіткої логіки – це обчислення зі словами, а не числами, що є основою для будь-якої експертної системи, яка намагається імітувати людське мислення. В роботі [2] Лотфі Заде із співавторами показані можливості нечіткої логіки для побудови інтелектуальних систем. Кількість публікацій з інтеграції квантових обчислень з машинним навчанням включаючи нечітку логіку стрімко зростає. Зокрема, у статтях [3] і [4] пропонуються моделі квантової нечіткої нейронної мережі. Автори досліджують, як принципи квантових обчислень (суперпозиція) можуть бути використані для більш ефективної обробки нечітких даних. Це прямий приклад інтеграції двох парадигм. Роботи [5] і [6] присвячені створенню інтелектуальних систем шляхом інтеграції алгоритмів кластеризації і квантових обчислень. Автори статті [7] використовують «квантово-натхненний» алгоритм рою частинок (PSO) для оптимізації параметрів нечіткої експертної системи. Це показує, як квантові ідеї навіть симульовані на класичних комп'ютерах можуть покращити роботу гібридних інтелектуальних систем.

Формулювання мети дослідження. Метою роботи є демонстрація синергетичного ефекту від поєднання нечіткої логіки, експертної системи та квантових алгоритмів для вирішення прикладної задачі – оптимізації процесу поливу рослин на основі аналізу зовнішніх умов.

Основна частина. Класична нечітка логіка використовує ступені належності від 0 до 1. Квантова механіка пропонує ідеальний аналог – кубіт [8]. Кубіт може перебувати в стані суперпозиції, тобто одночасно бути і 0, і 1 з певною амплітудою ймовірності. Цю амплітуду можна інтерпретувати як квантовий ступінь належності. Це дозволяє створювати набагато складніші та багатовимірні нечіткі множини. Замість простого «на 70 % належить до високої температури», ми можемо мати складний векторний стан, що описує належність до багатьох понять одночасно. Одночасно це можна трактувати, як нечіткі змінні.

Для побудови експертної системи з нечіткою логікою та використанням квантових обчислень треба в першу чергу була розроблена концептуальну архітектуру цієї системи. Ця архітектура складається з чотирьох основних блоків, що взаємодіють між собою.

Блок 1: Класичний інтерфейс та фазифікація (Classical Interface & Fuzzification).

Цей блок працює на класичному комп'ютері.

Функції:

1. Прийом вхідних даних: Отримує чіткі, вимірювані дані від користувача або сенсорів (наприклад, температура = 25.5 °C, вологість ґрунту = 20 %).

2. Фазифікація: Перетворює ці чіткі дані в нечіткі множини. Наприклад, «температура 25.5 °C» стає на 80 % Нормальна і на 20 % трохи підвищена.

3. Підготовка до квантової обробки: Результати фазифікації (ступені належності) кодуються в параметри для ініціалізації кубітів. Наприклад, ступінь належності μ може бути перетворена в кут обертання для квантового гейту $R_y(\theta)$, де $\theta = \arccos(\sqrt{\mu})$.

**Блок 2: Квантове ядро (Quantum Core).**

Це основа системи, що працює на квантовому процесорі (QPU) або на симуляторі. Тут відбуваються обчислення, які є надзвичайно повільними для класичних систем.

Компоненти:

1. Регістр даних (Data Register): Група кубітів, що кодує вхідні фазифіковані дані. Завдяки суперпозиції, один регістр може одночасно представляти безліч можливих вхідних станів.

2. Квантова база знань (Quantum Knowledge Base / QKB): Це головна інновація. Це не просто список, а єдиний, сильно заплутаний квантовий стан, що кодує всю базу правил (IF-THEN).

2.1. Кодування: Кожне правило IF {умова} THEN {висновок} не зберігається окремо. Замість цього, всі правила влітаються в єдиний стан за допомогою спеціалізованих квантових схем (наприклад, з використанням controlled-операторів). Це аналог квантової асоціативної пам'яті.

3. Квантова машина виведення (Quantum Inference Engine): Це не окремий компонент, а сам процес еволюції квантової системи.

Дія: регістр даних взаємодіє з квантовою базою знань (QKB). Завдяки квантовому паралелізму, вхідні дані одночасно перевіряються по відношенню до всіх правил, закодованих у QKB. Система природним чином еволюціонує до стану, що представляє результат логічного виведення. Цей процес є єдиною квантовою операцією, а не послідовним перебором правил.

Блок 3: Вимірювання та класична інтерпретація (Measurement & Classical Interpretation).

Цей блок є мостом від квантового світу назад до класичного.

Функції:

1. Квантове Вимірювання: Наприкінці роботи Квантового Ядра проводиться вимірювання стану кубітів. Результатом є набір класичних бітів (0 і 1) з певною ймовірністю.

2. Статистичний фналіз: Оскільки результат одного вимірювання є імовірнісним, квантова схема запускається багато разів (тисячі shots).

3. Формування Нечіткого Результату: Статистика результатів вимірювань інтерпретується як нечіткий висновок. Наприклад, якщо стан, що відповідає висновку «Полив інтенсивний», вимірюється у 75 % випадків, то ступінь істинності цього висновку дорівнює 0.75.

Блок 4: Дефазифікація та інтерфейс виведення (Defuzzification & Output Interface).

Цей блок знову працює на класичному комп'ютері.

Функції:

1. Агрегація Висновків: Збирає всі нечіткі висновки, отримані з попереднього блоці (наприклад, «Полив інтенсивний» на 20 хв., «Полив середній» на 15 хв.).

2. Дефазифікація: Перетворює агрегований нечіткий результат у чітке, зрозуміле для користувача рішення або команду. Наприклад, за методом центру ваги вираховується конкретне числове значення або обирається одна з кількох категорійних дій.

3. Виведення результату: Представляє фінальне рішення користувачеві або передає команду виконавчому механізму.

Ключові переваги такої архітектури:

1. Експоненційна ємність бази знань: Квантова база знань (QKB) може кодувати величезну кількість правил, що неможливо для класичних систем.

2. Миттєве логічне виведення: Завдяки квантовому паралелізму, процес виведення відбувається за один крок (еволюція системи), а не за тисячі кроків перебору правил.

3. Обробка складної невизначеності: Система може природно працювати зі складними кореляціями та заплутаними залежностями між даними, які важко описати в класичних нечітких системах.

Ця архітектура є концептуальною, але вона окреслює логічний шлях для поєднання потужності квантових обчислень із гнучкістю нечіткої логіки для створення експертних систем нового покоління.



Для прикладу застосування запропонованої архітектури розглянута наступна задача:

Оцінка доцільності поливу рослини.

- Вхідні дані: вологість ґрунту (%), температура повітря (°C).
- Висновок: тривалість поливу (хвилини).

Використані популярні бібліотеки Python:

- scikit-fuzzy для нечіткої логіки.
- qiskit для симуляції квантових обчислень.

Qiskit – це повноцінна програмна бібліотека (фреймворк) від IBM з відкритим вихідним кодом, написана на мові Python, яка призначена для роботи з квантовими комп'ютерами на рівні схем, імпульсів та алгоритмів.

Програмна реалізація складається з наступних кроків:

Крок 1. Нечітка експертна система (на scikit-fuzzy).

Нечітка експертна система для рекомендації поливу.

Програма створює повноцінну нечітку експертну систему, яка робить висновки за правилами, подібними до людських міркувань.

Крок 2. Додавання квантового компонента (Qiskit).

Квантовий компонент для пошуку оптимального правила (Grover Search).

У цьому прикладі:

- Кубіти представляють набір правил (наприклад, чотири нечіткі правила з бази знань).
- Оракул позначає одне правило як оптимальне.
- Алгоритм Гровера дозволяє знайти його з квадратичним прискоренням порівняно з класичним пошуком.

Крок 3. Концептуальне поєднання.

Конкретний результат роботи програми має вигляд:

– Система прийняття рішень для поливу рослин –

Вхідні дані:

- Вологість ґрунту: 35 %
- Температура повітря: 28 °C
- Прогноз дощу: False
- Рослина молода: False

Крок 1: Нечітка логіка

– Базова рекомендована тривалість поливу: 17.65 хв.

Крок 2: Експертна система

- Рішення базується на нечіткій логіці.
- Скоригована тривалість поливу: 17.65 хв.

Крок 3: Квантові обчислення

- Симуляція квантової схеми для вибору стратегії...
- Результат вимірювання: {'1': 1}
- Обрана стратегія: 'інтенсивний' полив.
- Застосовано 'інтенсивну' стратегію: тривалість збільшено.
- Квантово-оптимізована тривалість: 21.18 хв.
- ФІНАЛЬНИЙ РЕЗУЛЬТАТ –

Рекомендована тривалість поливу: 21.18 хвилин.

Висновки. Інтеграція нечіткої логіки, експертних систем та квантових обчи дозволяє створити багаторівневу систему прийняття рішень, де кожна технологія виконує свою унікальну функцію: нечітка логіка забезпечує гнучкість, експертна система – надійність та керованість, а квантові обчислення – оптимізацію стратегії. Представлений приклад є ілюстрацією потен-



ціалу гібридних систем для вирішення складних, багатофакторних завдань у таких галузях, як точне землеробство, автоматизація виробничих процесів та управління ресурсами.

Список використаних джерел

1. Zadeh L. A. Fuzzy Logic Computing with Words. *IEEE Transactions – Fuzzy Systems*. 1996. Vol. 4, No. 2. P. 103–111. DOI: <http://dx.doi.org/10.1109/91.493904>
2. Singh H., Gupta M. M., Meitzler T., Hou Z.-G., Garg K. K., Solo A. M. G., Zadeh L. A. Real-Life Applications of Fuzzy Logic. *Hindawi Publishing Corporation Advances in Fuzzy Systems*. 2013. Article ID 581879. 3 p. DOI: <http://dx.doi.org/10.1155/2013/581879>
3. Li Y., Tian M., Chen G., Du Y. Quantum fuzzy neural network: a new approach for fuzzy data processing. *International Journal of Fuzzy Systems*. 2020. Vol. 22, No. 1. P. 291–303. DOI: <https://doi.org/10.1007/s40815-019-00755-9>
4. Yang X. Quantum fuzzy neural network based on fuzzy number. *Frontiers in Computing and Intelligent Systems*. 2023. Vol. 3, No. 2. P. 99–105. DOI: [10.54097/fcis.v3i2.7524](https://doi.org/10.54097/fcis.v3i2.7524)
5. Ruan Y., Xue X., Tan X., Chen X. Quantum-inspired fuzzy clustering algorithm. *IEEE Access*. 2020. Vol. 8. P. 18585–18596. DOI: <https://doi.org/10.1109/ACCESS.2020.2968361>
6. Ouedrhiri O., Faghihi U., Toure F., Banouar O. Quantum Fidelity Based Fuzzy C-Means Clustering Algorithm. *Contribution to: QCE2*. 2024. P. 138–143. DOI: [10.1109/QCE60285.2024.10267](https://doi.org/10.1109/QCE60285.2024.10267)
7. Mishra R., Kumar S. Quantum-behaved particle swarm optimization for fuzzy expert system. *Journal of Intelligent & Fuzzy Systems*. 2021. Vol. 40, No. 4. P. 6291–6302. DOI: <https://doi.org/10.3233/JIFS-201530>
8. Ying M. Quantum computation, quantum theory and AI. *Artificial Intelligence*. 2010. Vol. 174. P. 162–176. DOI: [10.1016/j.artint.2009.11.009](https://doi.org/10.1016/j.artint.2009.11.009)

Дата першого надходження статті до видання: 19.02.2026

Дата прийняття статті до друку після рецензування: 12.03.2026

Дата публікації (оприлюднення) статті: 18.05.2026

Стаття поширюється на умовах ліцензії відкритого доступу (CC BY 4.0)



A. Bezverkhyi, I. Skrypnyk, O. Zhukov
Zaporizhzhia National University

INTEGRATING QUANTUM COMPUTING INTO FUZZY LOGIC EXPERT SYSTEMS

Summary

This paper presents the conceptual architecture and software implementation of a hybrid intelligent system that synthesizes three fundamental computing paradigms: fuzzy logic, expert systems, and quantum algorithms. The objective of this work is to demonstrate the synergistic effect of their combination in solving an applied problem – optimizing plant irrigation based on an analysis of external conditions. A fuzzy logic system is implemented in Python using the scikit-fuzzy toolkit. It processes imprecise input data (soil moisture, air temperature) and generates a baseline quantitative assessment – the irrigation duration – based on a set of linguistic rules. Based on the output from the fuzzy inference layer, a system of higher-order rules (an expert system) is applied. This layer introduces deterministic adjustments based on additional, contextual knowledge, thereby enhancing the reliability and adequacy of the final decision. To select the optimal course of action under conditions of uncertainty, a quantum algorithm simulation is employed using IBM's Qiskit library. This allows for the final optimization of the recommended duration.

Conclusions. The integration of the approaches described in this paper allows for the creation of a multi-level decision-making system where each technology performs a unique function: fuzzy logic provides flexibility, the expert system ensures reliability and control, and quantum computing delivers strategy optimization. The presented case study illustrates the potential of hybrid systems for solving complex, multi-factor problems in domains such as precision agriculture, industrial process automation, and resource management.

Keywords: hybrid intelligent system, fuzzy logic, expert system, quantum computing, optimization.