

УДК 004.05

DOI: 10.31673/2412-9070.2024.043746

А. С. ШАНТИР, канд. техн. наук, доцент,  
ORCID: 0000-0002-0466-3659;

О. В. ЗІНЧЕНКО, доктор техн. наук, доцент,  
ORCID: 0000-0002-3973-7814;

Т. М. КИСІЛЬ, ст. викладач,  
ORCID: 0000-0002-5123-0768;

О. В. СЕНЬКОВ, канд. техн. наук,  
ORCID: 0009-0008-0811-7306,

Державний університет інформаційно-комунікаційних технологій, Київ

## НЕВИЗНАЧЕНІСТЬ КІЛЬКІСНОЇ ОЦІНКИ ХАРАКТЕРИСТИК ЯКОСТІ В РОЗПОДІЛЕНИХ СИСТЕМАХ ТА МІКРОСЕРВІСАХ

У сучасному світі розробки програмного забезпечення — розподілені системи та мікросервіси набувають все більшої популярності. Вони дозволяють розробникам створювати гнучкі, масштабовані та надійні програми, які можуть швидко адаптуватися до змін в бізнес-потребах та технологічному середовищі. Натомість сучасне застосування мікросервісів в якості архітектурного підходу при розробці програмних систем, сприяє розділенню монолітних додатків на менші, незалежні сервіси, що можуть розгортатися та оновлюватися окремо. Це, в свою чергу, значно підвищує швидкість розробки та впровадження нових функцій. Однак, попри численні переваги, розподілені системи та мікросервіси створюють нові виклики, особливо у сфері оцінки їхньої якості.

Одним із головних аспектів є невизначеність, яка виникає через складність системи, варіабельність навантаження, різноманітність конфігурацій та залежність від зовнішніх компонентів. Суттєвий внесок у розробку теоретичних й практичних аспектів, щодо узагальнення проблематики питання невизначеності оцінювання кількісних характеристик якості внесли такі вчені, як С. Ageces, R. Ferrari, A. Saravia, F. Velázquez-Quesada, S. Guaman, J. Alamo, J. Caiza, M. Nakamura та ін. Метою даної статті є вирішення проблеми пов'язаної із невизначеністю кількісної оцінки характеристик якості в розподілених системах та мікросервісах.

Для реалізації мети в статті поставлені і вирішені такі завдання: проведено теоретичний огляд проблеми невизначеності при кількісній оцінці характеристик якості в розподілених системах та мікросервісах; розглянуто методологічні особливості врахування невизначеності при кількісній оцінці характеристик якості в розподілених системах та мікросервісах; досліджено практичний приклад реалізації кількісної оцінки характеристик якості в розподілених системах та мікросервісах в умовах невизначеності. В процесі вирішення піднятих завдань було використано методи аналізу, синтезу, узагальнення, порівняння.

**Ключові слова:** якість програмних систем; невизначеність кількісних характеристик якості; розподілені системи; мікросервіси; метод Монте-Карло; невизначеність; аналіз надійності мережі.

### ВСТУП

Якість програмного забезпечення визначається багатьма параметрами такими, як продуктивність, надійність, масштабованість, безпека та зручність використання. У розподілених системах та мікросервісах ці параметри набувають ще більшого значення, оскільки кожен компонент системи може впливати на загальну якість. Наприклад, затримки у відповіді одного мікросервісу можуть негативно позначитися на продуктивності всієї системи. Тому важливо мати ефективні методи для оцінки та забезпечення якості на всіх рівнях.

Однією з головних проблем при оцінці якості розподілених систем є невизначеність. У відповідності до [9] невизначеність у розподілених системах та мікросервісах відноситься до непередбачуваних змін та коливань в поведінці програмних систем, що можуть впливати на їх якість, продуктивність та надійність. В основному зазначена невизначеність виникає через наступні чинники.

◆ **Змінність навантаження:** кількість запитів, які надходять до системи, може значно коливатися, викликаючи труднощі в прогнозуванні продуктивності та відповідності. Відповідно до [5] непередбачувані зміни у кількості користувачів або запитів можуть впливати на продуктивність та масштабованість системи. Натомість в роботі [19] показано, що кількість запитів, які надходять до системи, може значно коливатися, викликаючи труднощі в прогнозуванні продуктивності та відповідності. В узагальненому випадку невизначеність у навантаженні відноситься до непередбачуваних змін у кількості користувачів або запитів, які можуть значно впливати на продуктивність та масштабованість системи. Це може проявлятися в формах, поданих далі:

• **різкі піки навантаження.** Наприклад, у випадку інтернет-магазинів кількість запитів може різко зрости під час розпродажів або святкових акцій;

© А. С. Шантир, О. В. Зінченко, Т. М. Кисіль, О. В. Сеньков, 2024

- *добові коливання*. В деяких системах навантаження може значно змінюватися протягом дня, наприклад, корпоративні системи можуть бути завантажені в робочі години та майже не завантажені вночі;

- *сезонні зміни*. Для деяких сервісів характерні сезонні коливання попиту, як-от туристичні портали, які можуть мати більший трафік у певні пори року.

Ці непередбачувані зміни у навантаженні можуть призвести до наступних проблем:

- *зниження продуктивності програмних систем*. При високому навантаженні система може працювати повільніше або навіть не відповідати на запити;

- *збої в роботі*. Різкі піки навантаження можуть призвести до збоїв або краху системи, якщо вона не здатна обробити таку кількість запитів;

- *проблеми з масштабованістю*. Масштабування системи для обробки змінного навантаження може бути складним і витратним процесом.

- ♦ *Складність інфраструктури*: системи можуть складатися з багатьох взаємодіючих компонентів, кожен з яких має власні параметри та характеристики. Згідно з [7] доцільно виокремлювати невизначеність, яка пов'язана із конфігураційними особливостями програмних систем. Різні конфігурації мікросервісів та інфраструктури програмних систем можуть по-різному впливати на їхню ефективність.

- ♦ *Збої та відмови*: розподілені системи піддаються ризику відмов на різних рівнях (наприклад, мережеві збої, відмови серверів або баз даних).

- ♦ *Нестабільність мережі*: затримки, втрата пакетів і змінна пропускна здатність можуть впливати на комунікацію між компонентами системи.

- ♦ *Невизначеність даних*: дані, що обробляються системою, можуть бути неповними, неточними або застарілими.

- ♦ *Невизначеність у зовнішніх залежностях*: інтеграції з іншими програмними системами або сервісами можуть додавати непередбачуваність через зміни в їх роботі або доступності.

- ♦ *Невизначеність у вимірюваннях*: складність точного вимірювання параметрів таких, як час відповіді або пропускна здатність, через варіабельність навантаження та умови виконання.

Отже, нині постає нагальна потреба у пошуку методичних спрямувань щодо вирішення піднятої проблематики.

## ОСНОВНА ЧАСТИНА

### *Теоретичний огляд проблеми невизначеності при кількісній оцінці характеристик якості в розподілених системах та мікросервісах*

Значна кількість науковців акцентують увагу на тому, що на практиці оцінка якості розподілених систем та мікросервісів із врахуванням невизначеності стикається з кількома ключовими проблемами.

Зокрема в [9] розглянуто проблематику вимірювання продуктивності, яка зводиться до того, що в умовах невизначеності важко точно виміряти та передбачити продуктивність програмної системи під різними умовами навантаження та конфігурації інфраструктури.

У роботі [22] порушується питання масштабованості, яке полягає в тому, що системи повинні ефективно масштабуватися для обробки зростаючої кількості запитів, зберігаючи при цьому якість обслуговування.

У [19] піднято проблему щодо забезпечення точного оцінювання надійності та відмовостійкості програмних систем в умовах невизначеності, яке полягає в тому, що забезпечення високої доступності та відмовостійкості є критично важливим при розробці та експлуатації програмних систем, але досягнення цього є складним завданням через непередбачувані збої.

У праці [8] розглянуто важливість оцінки якості синхронізації даних в умовах невизначеності при експлуатації програмних систем: дана оцінка має враховувати рівень підтримки консистентності даних в розподілених системах і це є складним завданням через можливі затримки та конфлікти при оновленні даних.

Робота [7] торкається теми, пов'язаної із безпекою програмних систем в умовах невизначеності. Проблематика вищезазначеного питання зводиться до того, що захист системи від зловмисних атак та несанкціонованого доступу ускладнюється через розподілену природу системи.

У [15] піднято питання щодо врахування моніторингу та логуювання в умовах невизначеності, Дану тенденційну потребу пов'язують із тим, що постійний моніторинг та аналіз логів є необхідним для виявлення проблем та відхилень, але це вимагає великих ресурсів та ефективних методів обробки даних.

Стаття [12] розглядає важливість врахування автоматизації процесів управління програмними системами та реакції на інциденти, що в свою чергу допомагає знизити ризики, але її налаштування та підтримка можуть бути досить складними, через наступні чинники.

- **Складність впровадження.** Процес налаштування автоматизації може вимагати значних витрат часу і ресурсів. Це включає розробку сценаріїв автоматизації, інтеграцію з наявними системами, та тестування для забезпечення коректної роботи.

- **Необхідність висококваліфікованих фахівців.** Для налаштування та підтримки автоматизованих систем потрібні спеціалісти з відповідними знаннями і досвідом. Це може створити додаткові труднощі, особливо якщо на ринку праці є дефіцит таких фахівців.

- **Постійна підтримка та оновлення.** Автоматизовані системи потребують регулярного оновлення та підтримки для адаптації до нових загроз і технологій. Це включає моніторинг, виправлення помилок, оновлення програмного забезпечення та адаптацію до змін у бізнес-процесах.

- **Початкові витрати.** Впровадження автоматизації може вимагати значних початкових інвестицій в обладнання, програмне забезпечення та навчання персоналу.

Важливість урахування автоматизації процесів управління програмними системами та реакції на інциденти зумовлена кількома факторами:

- **зниження ризиків.** Автоматизація допомагає швидко і ефективно реагувати на інциденти, дозволяючи зменшити час простою системи та мінімізувати негативний вплив на бізнес. Наприклад, автоматизовані системи можуть швидко виявити і нейтралізувати загрози, зменшуючи ймовірність витоку даних або інших серйозних проблем;

- **підвищення ефективності.** Автоматизація дозволяє виконувати рутинні та повторювані завдання без участі людини, що знижує ймовірність людських помилок та підвищує загальну ефективність роботи. Це також звільняє співробітників для виконання більш складних та творчих завдань.

- **швидкість реакції.** Автоматизовані системи можуть реагувати на інциденти в режимі реального часу, що значно швидше, ніж ручна реакція. Це особливо важливо в критичних ситуаціях, де кожна секунда може бути вирішальною.

Отже, не заважаючи на те, що автоматизація процесів управління програмними системами і реакції на інциденти в умовах невизначеності є важливою і корисною, її налаштування та підтримка пов'язані з певними складнощами, які необхідно враховувати при прийнятті рішення про впровадження таких систем.

У межах піднятого питання доцільно відзначити, що можливими підходами до вирішення проблем невизначеності у навантаженні є:

- **автоматичне масштабування.** Використання технологій автоматичного масштабування (auto-scaling) дозволяє системі автоматично додавати, або зменшувати ресурси у відповідь на зміни у навантаженні. Це знижує ризик відмов через пікові навантаження та покращує ефективність використання ресурсів;

- **балансування навантаження.** Балансувальники навантаження (load balancers) допомагають рівномірно розподіляти запити між різними серверами, що дає можливість уникати перевантаження окремих компонентів системи;

- **кешування.** Використання кешів для зберігання часто запитуваних даних може зменшити навантаження на основні сервери та знизити затримки відповіді;

- **мікросервісна архітектура.** Розподіл системи на окремі мікросервіси дозволяє масштабувати лише ті частини системи, які найбільше навантажені, що знижує загальні витрати на масштабування.

- **прогнозування навантаження.** Використання алгоритмів машинного навчання та аналізу історичних даних для прогнозування майбутнього навантаження може допомогти у більш точному плануванні ресурсів;

- **резервування ресурсів.** Підготовка резервних ресурсів, які можна швидко активувати у разі підвищеного навантаження, допомагає забезпечити безперебійну роботу системи.

У результаті невизначеність у навантаженні є критичним фактором, що впливає на продуктивність та масштабованість розподілених систем та мікросервісів. Використання ефективних методів автоматичного масштабування, балансування навантаження, кешування та прогнозування допомагає зменшити вплив невизначеності та забезпечити стабільну роботу системи у будь-яких умовах.

Таким чином, проблема невизначеності кількісної оцінки характеристик якості в розподілених системах та мікросервісах пов'язана з їх складністю, динамічністю та взаємозалежностями, що створює труднощі у точному вимірюванні та прогнозуванні їх характеристик.

### ***Методологічні особливості врахування невизначеності при кількісній оцінці характеристик якості в розподілених системах та мікросервісах***

Для ефективного оцінювання якості розподілених систем та мікросервісів, відповідно до [19] необхідно використовувати комплексний підхід, що включає різні методи та інструменти.

Результати аналізу переваг та недоліків основних сучасних методів оцінки якості розподілених систем та мікросервісів із урахуванням невизначеності наведено в табл. 1.

Таблиця 1

Переваги та недоліки основних сучасних методів оцінки якості розподілених систем та мікросервісів з урахуванням невизначеності

Імовірнісні моделі		
Метод	Переваги	Недоліки
Байєсові мережі	Враховують залежності між змінними	Вимагають великої кількості даних для точної оцінки ймовірностей
	Можуть оновлювати ймовірності на основі нових даних	Складність побудови та інтерпретації мережі при великій кількості змінних
Марковські процеси	Добре підходять для моделювання систем із чіткими станами та переходами	Можуть бути складними для систем з великим числом станів
	Простота математичного апарату при малих системах	Вимагають знання ймовірностей переходів між станами
Симуляційне моделювання		
Метод	Переваги	Недоліки
Монте-Карло	Гнучкий метод, може застосовуватись для різних задач	Висока обчислювальна складність при великій кількості симуляцій
	Враховує випадковість та невизначеність в системі	Вимогливість до ресурсів
Дискретно-подійне моделювання	Реалістичне моделювання подій та їхнього впливу на систему	Може бути складним в реалізації для великих систем
	Детальне відстеження динаміки системи	Вимогливість до точного опису всіх подій та їхньої взаємодії
Аналіз чутливості		
Метод	Переваги	Недоліки
Однофакторний аналіз	Простота реалізації та інтерпретації	Не враховує взаємодії між параметрами
	Дозволяє швидко оцінити вплив окремих параметрів	Обмежений аналізом одного параметра за раз
Метод кількісної оцінки	Забезпечує відносні показники стабільності системи	Може не враховувати всі аспекти невизначеності
	Легко обчислюється	Не дає детальної картини впливу різних факторів
Постійний моніторинг та логування		
Метод	Переваги	Недоліки
Метрики надійності	Дає чіткі показники надійності та доступності	Може потребувати складного налаштування та підтримки
	Легко інтерпретуються керівництвом та технічними спеціалістами	Вимагає постійного збору та аналізу даних
Автоматизація процесів		
Метод	Переваги	Недоліки
Контрольні межі	Дозволяє автоматично відстежувати та реагувати на відхилення в процесах	Може вимагати точного налаштування меж для коректної роботи
	Знижує необхідність ручного втручання	Може не враховувати всі можливі сценарії відхилень

У роботі [1] відзначається, що імовірнісні моделі, симуляція, аналіз чутливості, постійний моніторинг та логування є лише деякими з методів, які можуть допомогти у виявленні та мінімізації невизначеності.

Відповідно до [3] оцінка якості розподілених програмних систем та мікросервісів в умовах невизначеності має певні методологічні особливості, котрі зумовлені складністю, варіабельністю та динамічністю цих систем. У межах піднятого питання на методологічному рівні наведемо ключові методологічні аспекти, що враховують невизначеність у кількісній оцінці якості.

**1. Імовірнісні моделі** — використовуються для врахування невизначеності у вимірюваннях і прогнозуванні. Основні підходи включають у себе:

- **марковські ланцюги** — використовуються для моделювання ймовірностей переходів між станами програмної системи. Дозволяють оцінювати час простою, відмови та відновлення [11];

• **басові мережі** — використовуються для моделювання причинно-наслідкових зв'язків між компонентами програмної системи та їх параметрами. Допомагають враховувати невизначеність у залежностях між компонентами [22].

2. **Симуляційне моделювання** — відповідно до [12] симуляція дає можливість відтворювати поведінку системи під різними умовами та навантаженням. Основні методи:

• **симуляція із застосуванням методу Монте-Карло** — використовується для оцінки невизначеності у результатах за допомогою численних повторень експериментів з випадковими входами;

• **системна динаміка** — моделює взаємодії між компонентами системи з урахуванням часу, що дає можливість оцінювати динамічну поведінку системи.

3. **Аналіз чутливості** відповідно до [6] допомагає визначити, які параметри найбільше впливають на якість системи. Основні підходи:

• **моделювання сценаріїв** — дає можливість створювати різні сценарії для оцінки впливу змін у параметрах на результати. Допомагає визначити критичні параметри;

• **факторний аналіз** — дозволяє виявляти та оцінювати вплив окремих факторів на результати, а також визначити найважливіші фактори, що впливають на якість.

4. **Застосування засад моніторингу та логуювання** — згідно з працею [10] постійний моніторинг та логуювання забезпечують збір даних про роботу системи в режимі реального часу. Для даного підходу характерні наступні основні аспекти:

• **моніторинг продуктивності** — це відстеження ключових метрик продуктивності — часу відповіді, пропускну здатності, використання ресурсів;

• **аналіз журналів** — збір та аналіз журналів подій для виявлення відхилень, відмов та інших аномалій. Допомагає виявити причини проблем та оцінити їх вплив на систему.

5. **Аспекти, які включають застосування засад автоматизації та спеціальних автоматизованих інструментів** — автоматизація процесів моніторингу, тестування та аналіз є ключовим аспектом для зниження рівня невизначеності. Основні інструменти та підходи в даному разі зводяться до застосування принципових засад, таких як:

• **контейнеризація** — це використання контейнерів для ізоляції мікросервісів, що спрощує їх розгортання та тестування;

• **CI/CD (Continuous Integration/Continuous Deployment)** — автоматизація процесів інтеграції та розгортання, яка дає можливість швидко виявляти та виправляти помилки.

• **тестування в реальних умовах** — використання хмарних платформ для тестування системи під реальними умовами навантаження та конфігурацій.

У межах розробки методології на методологічному рівні допустимо, що якість мікросервісів залежить від надійності кожного мікросервісу  $Q_i$  та їх взаємодії. Загальна якість системи  $Q_{total}$  може бути представлена так

$$Q_{total} = f(Q_1, Q_2, \dots, Q_n), \quad (1)$$

де  $f$  — функція, що описує взаємодію мікросервісів.

Тоді вирішення висунутої проблеми зводиться до реалізації засад комплексного моделювання та відповідної реалізації аналізу якості розподілених систем.

Розглянемо два шляхи вирішення даної реалізації.

1. Застосування Методичного спрямування на базі методу Монте-Карло — використання симуляцій для оцінки загальної якості системи на основі ймовірнісних моделей кожного компонента подамо таким чином:

$$\hat{Q}_{total} = \frac{1}{N} \sum_{i=1}^N f(Q_1^i, Q_2^i, \dots, Q_n^i), \quad (2)$$

де  $Q_1^i, Q_2^i, \dots, Q_n^i$  — ймовірнісні оцінки якості окремих мікросервісів.

2. Аналіз надійності мережі — використання теорії графів для аналізу надійності мережевих взаємодій запишемо як

$$R(G) = \prod_{i=1}^n R(Q_i), \quad (3)$$

де  $R(G)$  — надійність системи;  $R(Q_i)$  — надійність окремих мікросервісів.

Таким чином, методологічні особливості оцінки якості розподілених систем та мікросервісів з урахуванням невизначеності включають використання ймовірнісних моделей, симуляційного моделювання, аналізу чутливості, постійного моніторингу та логуювання, а також автоматизації процесів. Ці підходи дають можливість ефективно враховувати та мінімізувати вплив невизначеності, що забезпечує високу якість і надійність програмних рішень.

**Практичний приклад реалізації кількісної оцінки характеристик якості в розподілених системах та мікросервісах в умовах невизначеності**

У розподілених системах та мікросервісах якість системи залежить від надійності кожного мікросервісу та їх взаємодії. Загальна якість системи може бути представлена, як функція взаємодії мікросервісів.

**1-й методологічний підхід** (застосування методу Монте-Карло).

Нехай у нас є мікросервіси  $n = 3$ , і ми використовуємо Метод Монте-Карло з  $N = 1000$  симуляцій. Висувається практична задача в межах невизначеності надати кількісну оцінку характеристик якості в розподілених системах та мікросервісах.

*Розв'язання:*

1) Створимо імовірнісні моделі надійності кожного мікросервісу  $Q_i$ :

$$Q_1 = 0,95;$$

$$Q_2 = 0,90;$$

$$Q_3 = 0,96.$$

2) Виконаємо  $N = 1000$  симуляцій, результати якої подано в табл. 2:

- генеруємо для кожної симуляції випадкові значення надійності кожного мікросервісу;
- розрахуємо загальну якість системи  $Q_{total}$  для кожної симуляції за допомогою функції взаємодії мікросервісів.

3) Обчислимо середнє значення загальної якості системи як середнє значення результатів усіх симуляцій.

Після опрацювання результатів маємо середнє значення загальної якості системи  $Q_{total}$  після 1000 симуляцій — 0,938.

Таблиця 2

Початкові дані для кількісної оцінки надійності системи та виявлення можливих точок вразливості за допомогою аналізу графа при застосуванні методу Монте-Карло

Симуляція	$Q_1$	$Q_2$	$Q_3$	$Q_{total}$
1	0,95	0,90	0,96	0,939
2	0,94	0,91	0,95	0,934
...	...	...	...	...
1000	0,96	0,92	0,94	0,940

**2-й методологічний підхід** (аналіз надійності мережі на базі застосування теорії графів для аналізу надійності мережевих взаємодій).

Нехай у нас є мережа мікросервісів з  $n = 4n = 4$  вузлів.

1) Створимо граф взаємодії мікросервісів:

- вершини графа представляють мікросервіси;
- ребра графа відображають взаємодію між мікросервісами.

2) Розрахуємо надійність кожного мікросервісу:

$$R(Q_1) = 0,95;$$

$$R(Q_2) = 0,90;$$

$$R(Q_3) = 0,96;$$

$$R(Q_4) = 0,92;$$

- визначимо загальну надійність системи  $R(G)$ , як добуток надійностей окремих мікросервісів;
- підставимо дані надійності мікросервісів в формулу (3) та отримаємо:

$$R(G) = 0,813.$$

• оцінимо надійність системи та виявимо можливі точки вразливості за допомогою аналізу графа. Результати розрахунків подано в табл. 3.

Таблиця 3

Початкові дані для практичного прикладу аналізу надійності мережі

Мікросервіс	Надійність $R(Q_i)$
1	0,95
2	0,90
3	0,96
4	0,92

Для виявлення вразливих точок у системі, необхідно оцінити внесок кожного мікросервісу в загальну надійність та зрозуміти, які з них є найкритичнішими для роботи системи.

Якщо надійність одного з мікросервісів суттєво знизиться, це може значно вплинути на загальну надійність системи. Для цього можна розглянути, як зміниться загальна надійність при різних сценаріях.

*Сценарій 1.* Зниження надійності мікросервісу  $R(Q_2)$ . Якщо надійність  $R(Q_2)$  знизиться до 0,8, то в даному разі маємо  $R(G) = 0,6696$ .

*Сценарій 2.* Зниження надійності мікросервісу  $R(Q_4)$  до 0,80 дасть  $R(G) = 0,65664$ .

Отже, зниження надійності одного мікросервісу суттєво знижує загальну надійність системи.

### Обговорення

Відповідно до [7] структурний аналіз графа взаємодій може допомогти ідентифікувати вузли та ребра, які є критичними для загальної роботи системи. Згідно з [12], відсутність зв'язків або збої у критичних мікросервісах можуть призвести до відмови всієї системи.

Щоб підвищити надійність системи, можна застосувати різні стратегії резервування та відмовостійкості. Згідно [22] доцільно застосовувати резервні мікросервіси — впроваджувати резервні копії критичних мікросервісів, що можуть автоматично замінити основні у випадку відмови. В роботі [3] наголошується доцільність застосування балансування навантаження, де розподіл навантаження між кількома інстанціями одного мікросервісу застосовується для зменшення ризику відмови. Натомість в праці [21] показана перспектива застосування моніторингу та швидке реагування — регулярний моніторинг стану мікросервісів та автоматичне виявлення і виправлення проблем.

Окрім методу Монте-Карло та аналізу надійності мережі за допомогою теорії графів, які були задіяні в нашій статті, існує кілька інших методів, що використовуються для оцінки надійності систем мікросервісів. Розглянемо деякі з них та порівняємо їх з обраними методами.

Аналіз відмов та наслідків (FMEA) (Failure Mode and Effects Analysis) є систематичним підходом до виявлення можливих відмов у системі та оцінки їхніх наслідків на роботу системи. Цей метод включає в себе ідентифікацію можливих відмов, їх причин та наслідків для системи, а також розробку плану дій для мінімізації ризиків.

Перевагами даного методу є те що він:

- надає детальний аналіз кожної можливої відмови та її впливу;
- допомагає у виявленні критичних компонентів, що потребують підвищеної уваги.

Проте у даного методичного підходу є суттєві недоліки, а саме:

- він досить трудомісткий і вимагає значних ресурсів;
- даний метод залежить від суб'єктивних оцінок експертів, що може вплинути на точність результатів.

Тому в межах порівняльного аналізу запропонованого нами підходу та можливістю застосування FMEA, який пропонують використовувати в [18], зазначимо, що на відміну від аналізу надійності за допомогою теорії графів, FMEA надає більш детальний аналіз кожного елемента системи. Проте в нашому випадку застосування методу Монте-Карло надає точніші кількісні оцінки, тоді як FMEA більше зосереджений на якісному аналізі.

У статті [1] автори пропонують застосовувати методику «Дерева відмов» (FTA). Згідно з [20], FTA (Fault Tree Analysis) використовує ієрархічну модель для аналізу причин потенційних відмов у системі. В праці [19] наголошується, що «Дерево відмов» починається з основної події (відмови системи) і далі розбивається на підподії, які можуть спричинити основну подію.

Перевагами такого методичного спрямування є:

- наочне представлення взаємозв'язків між відмовами.
- ідентифікаційна допомога в з'ясуванні причини відмов на різних рівнях системи.

Проте згідно з [3] у даного метода також є суттєві недоліки, а саме:

- FTA вимагає детальної інформації про систему і може бути складним для великих систем.
- при застосуванні FTA не завжди легко врахувати всі можливі взаємодії та випадкові варіанти.

У межах порівняльного обговорення зауважимо — FTA надає структурований підхід до аналізу відмов, що є корисним для візуалізації причинно-наслідкових зв'язків. Проте в нашому випадку аналіз надійності за допомогою теорії графів є більш простим і менш ресурсомістким. Натомість наша пропозиція щодо застосування методу Монте-Карло є гнучкішою і краще моделює випадкові варіанти взаємодій.

У праці [17] пропонується застосовувати статистичний аналіз даних про відмови. Згідно з [16], використання статистичних методів для аналізу історичних даних про відмови та прогнозування надійності системи. Дане спрямування носить комплексний методологічний характер та має цілий ряд методів (регресійний аналіз, аналіз виживання та статистичні моделі відмов). Перевагами такого підходу є те, що він:

- надає кількісні оцінки надійності на основі фактичних даних;
- дає можливість моделювати ймовірності відмов та прогнозувати майбутню надійність.

Проте у даного методу на відміну від запропонованого нами підходу є цілий ряд недоліків, а саме:

- 1) він вимагає наявності великого обсягу історичних даних;
- 2) дане спрямування досить складне для систем з малою кількістю даних або для нових систем.

У межах порівняльного обговорення відмітимо, що хоч статистичний аналіз і надає точні кількісні оцінки, але він вимагає наявності даних, що не завжди можливо для нових або унікальних систем. У той же час, на відміну від [14], у нашому випадку аналіз надійності за допомогою теорії графів можна застосовувати навіть за відсутності великого обсягу даних. Проте застосування методу Монте-Карло також потребує значних обчислювальних ресурсів, але може працювати з імітованими даними.

У [13] пропонується застосовувати аналіз надійності на основі агентного моделювання. Згідно з [9], використання агентного моделювання досить перспективне для симуляції взаємодій між компонентами системи і аналізу їх впливу на загальну надійність. Відповідно до [4] подібні моделі включають агентів, які діють відповідно до заданих правил і взаємодіють між собою.

Перевагами такого підходу є те, що він дає можливість моделювати складні взаємодії та поведінку системи в динаміці та може враховувати різноманітні сценарії та варіанти розвитку подій.

Проте дане методологічне спрямування має також недоліки, пов'язані із тим, що даний метод вимагає значних обчислювальних ресурсів і часу для розробки моделей. А також даний методологічний підхід згідно з [3] досить складний у реалізації і вимагає спеціальних знань.

Таким чином, хоч агентне моделювання є дуже гнучким і потужним, але його складність і обчислювальні вимоги можуть бути значними. Проте застосування запропонованого нами підходу на базі аналізу надійності за допомогою теорії графів є простішим у реалізації та менш ресурсомістким. Тоді як метод Монте-Карло може враховувати різні сценарії, але більш спрямований на випадкові варіанти взаємодій.

Отже, при порівнянні різних методів оцінки надійності систем мікросервісів впливає, що аналіз надійності мережі за допомогою теорії графів є простим і ефективним для швидкої оцінки та виявлення критичних точок, але менш точним для складних динамічних систем. Відмітимо, що метод Монте-Карло надає більш точні оцінки, враховуючи випадкові варіанти взаємодій, але потребує значних обчислювальних ресурсів. FMEA і FTA надають детальний якісний аналіз і структуроване представлення відмов, але можуть бути трудомісткими та залежними від суб'єктивних оцінок. Статистичний аналіз надає точні кількісні оцінки на основі даних, але вимагає наявності великого обсягу даних. Агентне моделювання є потужним для моделювання складних взаємодій, але вимагає значних ресурсів і складних моделей.

Вибір методу залежить від конкретних потреб, ресурсів та вимог до точності оцінки надійності. У багатьох випадках комбіноване використання різних методів може забезпечити найбільш повну та точну оцінку надійності системи.

У нашому випадку загальна надійність системи мікросервісів залежить від надійності кожного окремого мікросервісу. Найменш надійний мікросервіс (в практичному випадку  $R(Q_2)$ ) визначає потенційну вразливу точку системи. Зниження надійності одного мікросервісу значно впливає на загальну надійність системи. Натомість використання стратегій резервування та відмовостійкості може підвищити загальну надійність системи. Також відповідно до [2] доцільно застосувати графічний аналіз, який допомагає візуалізувати взаємодії та ідентифікувати критичні вузли і ребра.

У результаті оцінка та підвищення надійності системи мікросервісів є безперервним процесом, який включає моніторинг, аналіз та впровадження заходів для забезпечення стабільної, а також безперервної роботи додатків.

Використання методу Монте-Карло дає можливість оцінити якість системи шляхом імітації випадкових варіантів взаємодії мікросервісів. Тоді як аналіз надійності мережі дозволяє ідентифікувати точки недоліків та можливі вразливості в системі, що дає змогу приймати ефективні заходи щодо їх вирішення та підвищення загальної якості системи.

Таким чином, на відміну від методичних спрямувань, запропонованих в [19], аналіз надійності мережі з використанням теорії графів дає можливість оцінити загальну надійність системи, враховуючи взаємодію між мікросервісами. Такий аналіз допомагає виявити можливі точки вразливості та підвищити надійність системи в цілому.

Отже, метод Монте-Карло, на відміну від методичних спрямувань, викладених в праці [1], дає більш точну оцінку, оскільки він враховує взаємодію між мікросервісами, тоді як аналіз надійності мережі надає загальне уявлення про стан системи.



### Висновки

У цілому, проблема невизначеності кількісної оцінки характеристик якості в розподілених системах та мікросервісах пов'язана з їх складністю, динамічністю та взаємозалежностями, що створює труднощі у точному вимірюванні та прогнозуванні їх характеристик.

У загальному випадку невизначеності кількісної оцінки характеристик якості в розподілених системах та мікросервісах пов'язана, в основному, із непередбачуваністю поведінки програмних систем, котра виникає через велику кількість компонентів та різні умови виконання, поведінка системи може бути непередбачуваною. Наприклад, затримки в мережі, зміни навантаження на різні сервіси та відмови компонентів можуть сильно впливати на якість обслуговування.

Проведене дослідження виявило, що невизначеність кількісної оцінки характеристик якості в розподілених системах та мікросервісах є значним викликом через динамічність та складність таких систем.

У ході дослідження було встановлено, що на практиці використання імовірнісних моделей, симуляційного моделювання, аналізу чутливості, постійного моніторингу та автоматизації процесів допомагає зменшити невизначеність та забезпечити високу якість роботи системи.

Аналіз надійності мережі з використанням теорії графів дає змогу оцінити загальну надійність системи, враховуючи взаємодію між мікросервісами. Такий аналіз допомагає виявити можливі точки вразливості та підвищити надійність системи в цілому.

### Список використаної літератури

1. **Uncertainty-based knowing how logic** / C. Areces, R. Fervari, A. R. Saravia, F. R. Velázquez-Quesada // *Journal of Logic and Computation*. 2023. URL: <https://doi.org/10.1093/logcom/exad056>
2. **Behera D.** *Alternative methodology for epistemic uncertainty-based linear programming problem* // *Soft Computing*. 2023. URL: <https://doi.org/10.1007/s00500-023-08725-5>
3. **Bougeret M., Pessoa A., Poss M.** *Single machine robust scheduling with budgeted uncertainty* // *Operations Research Letters*. 2023. URL: <https://doi.org/10.1016/j.orl.2023.01.007>
4. **Boukhelifa N., Johnson C. R., Potter K.** *Visualization and decision making design under uncertainty* // *IEEE Computer Graphics and Applications*. 2023. Vol. 43(5). P. 23–25. URL: <https://doi.org/10.1109/mcg.2023.3302172>
5. **Sources of uncertainty and subjective prices** / V. Cappelli, S. Cerreia-Vioglio, F. Maccheroni [et al.] // *Journal of the European Economic Association*. 2020. URL: <https://doi.org/10.1093/jeea/jvaa013>
6. **Computerized software quality evaluation with novel artificial intelligence approach** / D. Chandra Yadav, Y. Singh, A. Kumar Pandey, A. Kannagi // *Proceedings on Engineering Sciences*. 2024. Vol. 5(4). P. 363–372. URL: <https://doi.org/10.24874/pes.si.24.02.019>
7. **Theoretical characterization of uncertainty in high-dimensional linear classification** / L. Clarté, B. Loureiro, F. Krzakala, L. Zdeborova // *Machine Learning: Science and Technology*. 2023. URL: <https://doi.org/10.1088/2632-2153/acd749>
8. **Guaman D. S., Alamo J. M. D., Caiza J. C.** *A systematic mapping study on software quality control techniques for assessing privacy in information systems* // *IEEE Access*, 2020. Vol. 8. P. 74808–74833. URL: <https://doi.org/10.1109/access.2020.2988408>
9. **Hahn S.-J., Lee B.-H.** *Quality evaluation to small scaled software implementation result* // *Journal of Korean Institute of Information Technology*. 2023. Vol. 21(1). P. 1–10. URL: <https://doi.org/10.14801/jkiit.2023.21.1.1>
10. **Significance of quantitative evaluation and assessment using automated volumetric breast density measurements software** / S. Hayashi, M. Kubo, H. Mori, M. Nakamura // *Nihon Nyugan Kenshin Gakkaishi (Journal of Japan Association of Breast Cancer Screening)*. 2023. Vol. 32(1). P. 63–65. URL: <https://doi.org/10.3804/jjabcs.32.63>
11. **Humphrey W. S., Singpurwalla N. D.** *A bayesian approach for assessing software quality and productivity* // *International Journal of Reliability, Quality and Safety Engineering*. 1998. Vol. 05(02). P. 195–209. URL: <https://doi.org/10.1142/s0218539398000194>
12. **Evaluation of the effectiveness of technology-based project management systems for software development** / M. M. Indi, D. M. Priyangan, F. D. Herdiani [et al.] // *Global International Journal of Innovative Research*. 2023. Vol. 1(2). P. 175–181. URL:

<https://doi.org/10.59613/global.v1i2.30>

13. **Evaluation of data quality based on Bayesian networks in railway rolling stock monitoring systems** / S. Kamaletdinov, N. Aripov, S. Khudayberganov [et al.] // *E3S Web of Conferences*. 2023. Vol. 460. P. 04014. URL: <https://doi.org/10.1051/e3sconf/202346004014>

14. **Signal quality indices evaluation for robust ECG signal quality assessment systems** / F. Kuetche, A. Noura, P. E. Ntsama [et al.] // *Biomedical Physics & Engineering Express*. 2023. URL: <https://doi.org/10.1088/2057-1976/ace9e0>

15. **Software defect prediction for quality evaluation using learning techniques ensemble stacking** / M. R. Kusuma, Windu Gata, Sigit Kurniawan [et al.] // *Jurnal Teknologi Informasi Dan Komunikasi*. 2023. Vol. 13(2). P. 1–13. URL:

<https://doi.org/10.35585/inspir.v13i2.58>

16. **Liu C. Quantitative evaluation of software component behavior discovery approach** // *IEICE Transactions on Information and Systems*. 2021. E104.D(1). P. 117–120. URL:

<https://doi.org/10.1587/transinf.2020mpl0001>

17. **Merzlyakova E. Y., Yanchenko E. V. Review of software quality verification and evaluation methods** // *The Herald of the Siberian State University of Telecommunications and Informatics*. 2023. Vol. 17(1). P. 92–106. URL:

<https://doi.org/10.55648/1998-6920-2023-17-1-92-106>

18. **Pankov P. S., Tagaeva S. B. Systems of differential equations and computer phenomena** // *Herald of Institute Mathematics of the National Academy of Sciences of the Kyrgyz Republic*. 2020. Vol. (2). P. 86–93. URL: [https://doi.org/10.52448/16948173\\_2020\\_2\\_86](https://doi.org/10.52448/16948173_2020_2_86)

19. **Yoon H. A quantitative evaluation for usability under software quality models** // *International Journal on Recent and Innovation Trends in Computing and Communication*. 2023. Vol. 11(3). P. 24–29. URL:

<https://doi.org/10.17762/ijritcc.v11i3.6194>

20. **An entropy-based group decision-making approach for software quality evaluation** / C. Yue, R. Huang, D. Towey [et al.] // *Expert Systems With Applications*. 2023. P. 121979. URL:

<https://doi.org/10.1016/j.eswa.2023.121979>

21. **Informative data selection with uncertainty for multimodal object detection** / X. Zhang, Z. Li, Z. Zou [et al.] // *IEEE Transactions on Neural Networks and Learning Systems*. 2023. P. 1–13. URL:

<https://doi.org/10.1109/tnnls.2023.3270159>

22. **Evaluating the impact of uncertainty visualization on model reliance** / J. Zhao, Y. Wang, M. V. Manceido [et al.] // *IEEE Transactions on Visualization and Computer Graphics*. 2023. P. 1–15. URL:

<https://doi.org/10.1109/tvcg.2023.3251950>

A. Shantyr, O. Zinchenko, T. Kysil, O. Senkov

### UNCERTAINTY IN THE QUANTITATIVE EVALUATION OF QUALITY CHARACTERISTICS IN DISTRIBUTED SYSTEMS AND MICROSERVICES

In the modern world of software development, distributed systems and microservices are becoming increasingly popular. They enable developers to create flexible, scalable, and reliable applications that can quickly adapt to changes in business needs and the technological environment. The contemporary use of microservices as an architectural approach in software system development promotes the division of monolithic applications into smaller, independent services that can be deployed and updated separately. This, in turn, significantly increases the speed of development and implementation of new features. However, despite numerous advantages, distributed systems and microservices present new challenges, particularly in the evaluation of their quality.

One of the main aspects is the uncertainty that arises from the complexity of the system, variability of the load, diversity of configurations, and dependence on external components. Significant contributions to the development of theoretical and practical aspects related to the generalization of the problem of uncertainty in the evaluation of quantitative quality characteristics have been made by researchers such as C. Areces, R. Ferrari, A. Saravia, F. Velázquez-Quesada, S. Guaman, J. Alamo, J. Caiza, M. Nakamura, and others. The purpose of this article is to address the issue related to the uncertainty in the quantitative evaluation of quality characteristics in distributed systems and microservices.

To achieve this goal, the article sets and resolves the following tasks: conducting a theoretical review of the problem of uncertainty in the quantitative evaluation of quality characteristics in distributed systems and microservices; considering methodological features of accounting for uncertainty in the quantitative evaluation of quality characteristics in distributed systems and microservices; exploring a practical example of implementing the quantitative evaluation of quality characteristics in distributed systems and microservices under conditions of uncertainty. In the process of addressing the tasks, methods of analysis, synthesis, generalization, and comparison were used.

**Keywords:** software system quality; uncertainty of quantitative quality characteristics; distributed systems; microservices; Monte Carlo method; uncertainty; network reliability analysis.