

## **МАРШРУТИЗАЦІЯ**

# **АДАПТИВНА СТРАТЕГІЯ МАРШРУТИЗАЦІЇ БІЗНЕС-ОПЕРАЦІЙ В ОРГАНІЗАЦІЙНО-ТЕХНІЧНІЙ СИСТЕМІ**

## ЗМІСТ

Вступ.....	3
1. Аналіз проблемної області та формулювання задач дослідження.....	6
1.1. Аналіз проблем розподілення виконавчих ресурсів в організаційно-технічних системах .....	6
1.2. Огляд досліджень щодо керованих систем масового обслуговування.....	8
1.3. Висновки та формулювання задач дослідження .....	11
2. Розробка стратегії керування чергами та методу її адаптації.....	14
2.1. Задача динамічної маршрутизації.....	14
2.2. Основні положення стратегії управління чергами.....	16
2.3. Розробка методу і алгоритму оптимізації для оперативної адаптації стратегії управління чергами.....	20
2.4 Розробка структури імітаційної моделі.....	24
Висновки з розділу .....	26
3. Експериментальні дослідження.....	27
3.1. Короткий опис програмного засобу.....	27
3.2. Результати експериментів.....	27
Висновки з розділу .....	28
Висновки.....	29
Анотація.....	30
Список використаних джерел.....	31
Додаток А. Опис програмного засобу.....	33

## ВСТУП

Організаційно-технічна система (ОТС) являє собою сукупність технічних і людських ресурсів, призначену для виконання певної множини бізнес-процесів (БП). Кожен БП проходить ряд етапів, які представляють собою окремі бізнес-операції (БО), що виконуються на конкретних автоматизованих робочих місцях (АРМ). Будь-який БП створює потік заявок на виконання БО. Розподіл заявок між АРМ відбувається, з одного боку, відповідно до бізнес-логіки БП, а з іншого боку – з урахуванням компетенцій співробітників ОТС і їх поточного завантаження. Оскільки схожі компетенції зазвичай мають кілька співробітників, виникає можливість варіювати маршрути БП через мережу АРМ. Таким чином, ми приходимо до необхідності динамічного розподілу заявок або до динамічного формування виконавчих структур при виконанні множини БП. Потреба в динамічному формуванні виконавчих структур в рамках ОТС виникає з протиріччя між вимогою гарантованого виконання множини бізнес-операцій у встановлені терміни і обмеженнями на технологічні і організаційні можливості ОТС.

В процесі функціонування ОТС з динамічної маршрутизацією БО системі управління необхідно накопичувати завдання, які очікують обробки, і формувати черги заявок різних типів до кожного обслуговуючого вузлу (АРМ). Крім того, необхідно автоматично проводити періодичне оновлення черг і повідомляти виконавців про наявність в черзі нових, ще не переглянутих завдань, завдань з високим пріоритетом або завдань з встановленим граничним терміном виконання [1]. Це стане можливим в повній мірі, якщо буде розроблена адаптивна стратегія маршрутизації бізнес-процесів через множину автоматизованих робочих місць ОТС або, іншими словами, адаптивна стратегія управління чергами.

Як показує огляд літератури і практика, проблема оптимізації процесу динамічного розподілу навантаження на АРМ при реалізації БП як і раніше не вирішена. Вирішення цієї проблеми є актуальним завданням.

Метою роботи є розробка адаптивної стратегії управління чергами при виборі маршрутів для безлічі різнорідних бізнес-процесів і потоків заявок в рамках організаційно-технічної системи з урахуванням певного критерію оптимальності та деяких обмежень. Для досягнення поставленої мети в роботі вирішуються наступні завдання:

- аналіз проблем розподілу функцій і робіт в організаційних системах з точки зору теорії масового обслуговування;
- побудова стратегії управління чергами при виборі маршрутів бізнес-процесів через множину робочих місць, здатної до адаптації в умовах, що змінюються під час функціонування ОТС;
- розробка методу адаптації стратегії управління чергами при змінах у розподілі потоків заявок різного типу;
- розробка імітаційної моделі для реалізації та апробації методу оптимізації адаптивної стратегії.

**Об'єктом досліджень** є процес маршрутизації бізнес-операцій в організаційно-технічній системі.

**Предметом дослідження** є адаптивна стратегія керування чергами при виборі маршрутів для множини різнорідних бізнес-процесів.

**Методи дослідження.** Виконані дослідження базуються на положеннях методології системного аналізу, теорії систем масового обслуговування, теорії моделювання та комп'ютерних експериментах.

**Наукова новизна** полягає в наступному:

Отримала подальший розвиток адаптивна стратегія управління чергами в автоматизованих організаційно-технічних системах, за рахунок урахування в якості ознак ступеня інтенсивності потоку заявок, міри компетенції виконавців та ступеня завантаження виконавців, що дозволяє реалізувати процес структурної адаптації системи до змін у оперативній обстановці.

Удосконалено метод адаптації стратегії управління чергами в автоматизованих організаційно-технічних системах, заснований на еволюційному підході до підбору оптимальних граничних значень параметрів

відбору заявок за вартістю затримки і підбору компетенції співробітників, що дає можливість вибору найкращих умов багатофазного обслуговування і зниження втрат очікування.

**Практичне значення** одержаних результатів. Розроблена адаптивна стратегія управління чергами в ОТС апробована на імітаційній моделі. Результати статистичної обробки даних, отриманих в ході імітаційних експериментів, підтвердили можливість зниження втрат очікування при реалізації методу адаптації стратегії управління чергами. Оперативна зміна значень параметрів стратегії дозволяє утримувати квазістаціонарний режим обслуговування в широкому діапазоні інтенсивностей надходження заявок.

**Публікації.** За темою роботи опубліковано статтю у фаховому виданні та тези доповідей на двох міжнародних конференціях. Копії публікацій додаються.

# РОЗДІЛ 1

## АНАЛІЗ ПРОБЛЕМНОЇ ОБЛАСТІ ТА ФОРМУЛЮВАННЯ ЗАДАЧ ДОСЛІДЖЕННЯ

### 1.1 Аналіз проблем розподілення виконавчих ресурсів в організаційно-технічних системах

Організаційно-технічна система в даний час є обов'язковою складовою будь-якої компанії. Більшість організацій мають так звану функціонально-орієнтовану структуру. Це означає, що кожен підрозділ орієнтований на виконання цілком певного кола завдань (закупівлі, виробництво, фінанси і бухгалтерія, конструювання, технологія і т. ін.). Відповідно, один функціональний підрозділ несе відповідальність за все продукти і території в рамках своїх функцій. Перевага вузької спеціалізації службовців «компенсується» великими накладними витратами на комунікації і координацію функціональних підрозділів. Функціонально-орієнтовані організації мають ряд недоліків [2], основними з яких є:

- неможливість швидкої реакції на зміни;
- відірваність працюючих від кінцевого результату;
- головним споживачем результатів праці працівника є вищестоящий начальник;
- відсутність безпосередньої орієнтації на клієнта;
- ускладнено взаємодію та обмін інформацією між підрозділами організації.

Альтернативою строго функціональній структурі є процесно-орієнтована структура. Відповідно з новим поглядом на організацію робота повинна бути організована навколо процесів [3]. Метою організації повинно бути вдосконалення бізнес-процесів для подолання їх фрагментарності і для досягнення істотних поліпшень в ключових показниках результативності – витрати, якість, рівень обслуговування та оперативність.

Незважаючи на явні переваги процесно-орієнтованого устрою компанії, – домогтися створення такої структури в чистому вигляді не представляється можливим, тому що в будь-якій компанії переважає ієрархічне підпорядкування структур і спеціалізація підрозділів.

Організації, які мають складну і розгалужену структуру, характеризуються, як правило, такими особливостями [2]:

- досить велика кількість функцій, процесів, елементів даних і складні взаємозв'язки між ними, що вимагає ретельного моделювання і аналізу даних і процесів;
- наявність сукупності тісно взаємодіючих підсистем, що мають свої локальні завдання і цілі функціонування (наприклад, обробка транзакцій, рішення регламентних задач, підтримки прийняття рішень);
- роз'єднаність і різноманітність окремих груп виконавців за рівнем кваліфікації і традиціям, що склалися у роботі.

Аналогічна проблема виникає при наявності прямої (дисциплінарної) підпорядкованості одного організаційного елемента іншому і одночасно додаткової (функціональної) підпорядкованості. Найбільш яскравим прикладом може служити бухгалтерія великої компанії, що має кілька напрямків діяльності. Бухгалтери, які обслуговують певний напрямок діяльності такої компанії, входять до складу єдиної бухгалтерії і підпорядковуються (дисциплінарно) головному бухгалтеру (іноді фінансовому директору). Однак функціональна підпорядкованість (в рамках основних функціональних обов'язків бухгалтерів, які обслуговують напрямок) має на увазі їх підпорядкування керівнику функціонального блоку (напрямку).

Тим не менш, ключовим поняттям в сучасній теорії управління бізнесом є процес. Міжнародний стандарт ІСО 9000:2000 визначає процес як сукупність взаємозв'язаних і взаємодіючих видів діяльності, яка перетворює входи у виходи. Процес включає одну або більше пов'язаних між собою процедур або функцій, які спільно реалізують якусь задачу бізнесу – зазвичай в рамках

організаційної структури. Він може виконуватися в межах однієї організаційної одиниці, охоплювати кілька одиниць або навіть кілька різних організацій, наприклад, в системі «покупець-постачальник».

З урахуванням сказаного, першим кроком робіт по оптимізації діяльності повинно стати виділення основних продуктів компанії і вибудовування бізнес-процесів відповідно до продуктовими лініями. Це дозволяє отримати продуктові «зрізи» бізнес-процесів, що протікають в організації. Однак завжди існує кілька функціональних підрозділів, які приймають участь в обслуговуванні всіх продуктових ліній, наприклад, бухгалтерія, транспортний цех і т. ін. Перебудувати ці напрямки, розбивши їх на продуктові зрізи компанії, складно, тому що це призведе до додаткових проблем і витрат. Тім не менше, процесний підхід дає змогу збудувати відповідні ланки виконавців, яки безпосередньо задіяні у бізнес-процесі. Це, у свою чергу, дозволяє ефективно автоматизувати виконання бізнес-процесів не перебудовуючи функціональну структуру підприємства [4].

Отже, для організації в цілому, завдання оптимізації діяльності зводиться до виділення бізнес-процесів відповідно до продуктових ліній і функціоналу підрозділів, і ув'язці їх в наскрізні процеси організації. Це дозволяє автоматизувати диспетчеризацію процесів, що, у свою чергу, створює умови для підвищення продуктивності і якості діяльності організації, прискорення і підвищення узгодженості обробки бізнес-завдань.

## **1.2 Огляд досліджень щодо керованих систем масового обслуговування**

Моделі і методи теорії масового обслуговування знаходять широкі застосування в задачах організації виробництва, при побудові систем зв'язку і обчислювальних систем, у військовій справі і т. ін. Природно, що завдання оптимального управління, що постійно виникають в цих областях, привели до формування поняття керованої системи масового обслуговування та



постановці задач оптимального управління системами масового обслуговування. Дослідження керованих систем є одним з актуальних і найбільш перспективних напрямків розвитку теорії масового обслуговування.

Поняття керованої системи масового обслуговування було введено О. І. Бронштейном і В. В. Риковим в роботі [5]. Найбільш фундаментальний огляд УСМО міститься в роботі В. Рикова [6]. Як об'єкт математичного дослідження система масового обслуговування (СМО) характеризується наступними елементами:

- а) вхідними потоками заявок  $A$ ;
- б) механізмом і тривалістю обслуговування  $B$ ;
- у) структурою системи  $C$  (наприклад –  $m$  – кількість каналів);
- б) дисципліною обслуговування  $D$  (наприклад  $n$  – ємність накопичувачів).

Під абстрактної системою масового обслуговування (СМО) будемо розуміти математичний об'єкт  $S=(A,B,C,D)$  що задається цими чотирма елементами. Параметри  $A$  и  $B$  можуть приймати значення:  $D$  – детермінований розподіл;  $M$  – показовий;  $E_r$  – розподіл Ерланга;  $H_r$  – гіперпоказовий;  $G$  – розподіл загального виду.

Визначення. Систему, в якій будь-які з параметрів, що визначають той чи інший з її елементів, допускають застосування керуючих впливів, назвемо керованою системою масового обслуговування (КСМО).

При цьому завдання правила використання управляючих впливів у часі (*стратегії*) є необхідною умовою повного опису функціонування СМО. На клас стратегій можуть бути накладені обмеження. У таких випадках виникають завдання визначення стратегій з урахуванням цих обмежень, які є оптимальними в прийнятому сенсі. Рішенням завдань такого роду і займається теорія КСМО.

Критерієм якості управління в залежності від цілей дослідження можуть служити різні характеристики СМО: продуктивність системи, середня

кількість вимог в системі або середня довжина черги, середній час перебування вимог у системі або середній час очікування, ймовірності втрат, коефіцієнт завантаження системи або середній час простою приладів і т. ін., а також різні економічні показники, пов'язані з цими характеристиками.

Незважаючи на те, що існує безліч моделей, покликаних описувати системи з динамічною маршрутизацією, для всіх них можна виділити ряд загальних властивостей. В основному, у всіх моделях розглядається пуассоновський потік вхідних заявок і передбачається, що час обслуговування розподілено експоненціально. Узяті при цьому за основу припущення про однорідність часу обслуговування, не завжди відповідає дійсності, хоча і корисно, так як призводить до простих марковських систем, які наближено описуються системами диференціально-різницевих рівнянь [7, 8].

В роботі [9] для систем обслуговування з відмовами, довільним розподілом тривалості обслуговування і двома способами розподілу заявок по неоднорідним обслуговуючим приладам (рівноймовірним розподілом заявок по всіх приладах (вільних і зайнятих) і рівноймовірним розподілом заявок за вільними приладами) знайдені стаціонарні розподіли числа заявок і доведена їх інваріантність щодо функцій розподілу тривалості обслуговування.

В роботі [10] запропонований метод визначення характеристик системи обслуговування  $M/G/1/m$  з функцією випадкового відкидання заявок і розподілом часу обслуговування, що залежать від довжини черги. У реальних ОТС ні випадкове, ні детерміноване відкидання заявок не практикується, бізнес-процеси перериваються лише в особливих випадках. У роботах [11, 12] розглядається процес динамічної маршрутизації заявок, що надходять в систему з  $N$  серверів відповідно до розподілу Пуассона з інтенсивністю  $N\lambda$  і мають Вейбулівський розподіл часів обслуговування. Маршрутизація проводиться таким чином, що при вступі заявки до системи в момент часу  $t$ , випадково вибираються  $K$  серверів з  $N$ , і заявка стає в чергу з мінімальною довжиною. У реальних ОТС випадковий вибір підмножини вузлів неможливий

через обмеження по компетенції виконавців. Крім того, вибір черзі з мінімальною довжиною не завжди оптимальний, так як на стан черги впливає ціна втрат затримки, яка є індивідуальною по кожній заявці.

У роботах [13, 14] розглянуті алгоритми динамічної маршрутизації розподіляють надходять в систему запити, спираючись на поточні характеристики черг таким чином, що при надходженні запиту в систему він направляється на обслуговуючу станцію з чергою найменшої довжини або в чергу середньої довжини, якщо необхідно щоб в системі завжди залишалось кілька досить вільних черг, наприклад, для швидкого обслуговування пріоритетних запитів. Очевидно, що такі стратегії є емпіричними і не можуть бути рекомендовані для реальних ОТС з невизначеним розподілом потоку різнорідних заявок і мінливими інтенсивностями процесів обслуговування і відсутності відмов. Такі системи в кодуванні Д. Кендалла мають позначення  $G/G/m/n$ , причому, параметри  $m, n$  змінюються в часі.

Таким чином, незважаючи на те, що існує досить розвинений аналітичний апарат дослідження багатоканальних багатофазних систем масового обслуговування, ті обмеження, які дозволяють отримати аналітичні рішення, є неприйнятними і відповідні математичні схеми стають непридатні для опису реальних процесів з достатньою для практики точністю. Це говорить про актуальність проблеми розробки адаптивної стратегії управління чергами заявок в реальних ОТС.

### **1.3 Висновки та формулювання задач дослідження**

Розглянуті вище моделі керованих систем масового обслуговування по суті є математичними схемами, які можна використовувати для формалізації і математичного опису деяких реальних систем. Безумовно, важливим є те, що для згаданих схем при деяких обмеженнях щодо структури потоку заявок і процесу функціонування системи отримані аналітичні рішення задачі оцінки якості обслуговування. Одним з таких обмежень, яке для ряду завдань є

прийнятним, приймається гіпотеза про незалежність параметрів заявки від моментів надходження заявок як випадкових величин. Тоді потік моментів надходження  $t_j$ , з одного боку, і вектори параметрів заявок – з іншого, можуть бути описані окремо як незалежні випадкові об'єкти відповідними законами розподілу.

Подальші спрощення зводяться зазвичай до наступного. При опису потоків  $t_j$  використовують тільки потоки однорідних подій з обмеженою післядією (як стаціонарні, так і нестаціонарні). Це дозволяє розглядати інтервали часу між послідовними заявками як незалежні випадкові величини. Деякі з параметрів заявок можуть бути фіксованими або випадковими величинами, що не залежать від інших.

Однак у багатьох практичних випадках ті обмеження, які дозволяють отримати аналітичні рішення, є неприйнятними. Наприклад, в реальних ОТС на наступних фазах можуть з'явитися заявки, які не надходили на попередні фази, але мають певні ознаки або властивості. Може трапитися, що обслуговування, що стосується подальшої фази, починається ще до закінчення обслуговування на попередній і т. ін. Відповідно, класичні математичні схеми стають непридатні для вирішення тих проблем, які існують в організаційно-технічних системах.

З урахуванням сказаного, стає зрозуміло, що аналітичні вирази для опису і оптимізації режиму роботи СМО в традиційних постановках не можуть охопити різноманітність ситуацій, в яких потрібно застосувати оптимальний або раціональний динамічний розподіл ресурсів при виконанні множини бізнес-процесів. При цьому необхідно, щоб стратегія управління чергами заявок була адаптивною, тобто оптимізація повинна відбуватися практично в той момент, коли змінився розподіл потоків різнорідних заявок або інтенсивність інтегрального потоку.

Метод імітаційного моделювання дозволяє отримати рішення подібних завдань практично без згаданих обмежень.

Визначимо адаптивну стратегію, як таку стратегію, яка певним чином змінюється в процесі виконання завдання на основі накопичення даних про можливі результати того чи іншого варіанту рішення.

Метою роботи є розробка адаптивної стратегії управління чергами при виборі маршрутів для множини різнорідних бізнес-процесів і потоків заявок в рамках організаційно-технічної системи з урахуванням певного критерію оптимальності та деяких обмежень. Для досягнення поставленої мети в роботі вирішуються наступні завдання:

- побудова стратегії управління чергами при виборі маршрутів бізнес-процесів через множину робочих місць, здатної до адаптації в умовах, що змінюються під час функціонування ОТС;
- розробка методу оптимізації стратегії управління чергами при змінах у розподілі потоків заявок різного типу;
- розробка імітаційної моделі для реалізації і перевірки працездатності методу оптимізації адаптивної стратегії.

## РОЗДІЛ 2

### РОЗРОБКА СТРАТЕГІЇ КЕРУВАННЯ ЧЕРГАМИ ТА МЕТОДУ ЇЇ АДАПТАЦІЇ

#### 2.1 Задача динамічної маршрутизації

Розглядаючи ОТС як СМО, необхідно враховувати такі особливості системи:

1. Багатокритеріальність задачі оптимізації стратегії управління. Залежно від ситуації може виявитися необхідним оптимізувати загальні втрати часу проходження множини БП через ОТС, середнє число заявок в черзі, середній час перебування в черзі, будь-якої інший критеріальний параметр або сукупність параметрів в рамках інтегрального критерію.
2. Велика і невизначена кількість станів системи через наявність багатовимірною операційного простору на множині БП, БО, АРМ.
3. Неоднорідність заявок по вектору параметрів через різноманітність БП і БО, вартості затримки виконання, пріоритетам і т. ін.
4. Невизначеність розподілу потоків різнорідних заявок на вході ОТС.
5. Неоднорідність приладів обслуговування через обмеження в компетенціях співробітників.
6. Один і той же прилад (АРМ) обслуговує кілька типів заявок.
7. Режим обслуговування на будь-якому АРМ змінюється за довільним законом, від повної зупинки до перевантаження.

Таким чином, ОТС є СМО конфігурації  $GD\tilde{}/GD\tilde{}/m\tilde{}/n$ , де аббревіатура  $GD$  означає, що вхідні потоки СМО є послідовностями, члени яких можуть бути незалежні і, крім того, вони мають різні розподіли, вид яких заздалегідь невідомий.

Будемо вважати, що менеджер ОТС формує завдання на виконання БП. При цьому він використовує якусь нотацію, яка дозволяє здійснити

декомпозицію БП на БО, відносячи кожен БО до задалегідь певного класу, який, в свою чергу, поєднаний з певною компетенцією співробітника.

У автоматизованій ОТС виконання БП здійснюється багатоагентною системою. У складі системи є програмно реалізований агент-диспетчер, програмно реалізовані агенти-монітори, програмно реалізовані агенти-виконавці (боти або  $b$ -агенти) і співробітники –  $h$ -агенти. Менеджер ОТС контактує з агентом-диспетчером, який здійснює динамічну маршрутизацію БП. Агент-диспетчер (надалі – диспетчер) контактує з агентами-моніторами, кожен з яких контролює деяку підмножину АРМ. Стан усіх заявок (БО), всіх БП і всіх АРМ фіксується в динамічній таблиці, модель якої у вигляді сукупності кінцевих автоматів описана в роботі [15]. Фрагменти цієї таблиці для окремого АРМ формує агент-монітор (далі – монітор).

Метод, моделі і критерії динамічної маршрутизації заявок описані в роботі [16]. Формально задача динамічної маршрутизації записана наступним чином [17]. Необхідно визначити оптимальну структуру виконавчої мережі на період часу  $t$

$$S_{opt}(t) = \{E^*(t), A^*(t), X^*(t), Y^*(t)\}, \quad t \in T, \quad (2.1)$$

де  $E^*(t) \in E(t)$  – допустима множина елементів системи в період  $t$ ;  $A^*(t) \in A(t)$  – допустима множина взаємозв'язків елементів системи в період  $t$ ;  $X^*(t) \in X(t)$  – допустима множина власних функціональних характеристик елементів системи в період  $t$ ;  $Y^*(t) \in Y(t)$  – допустима множина системних функціональних характеристик в період  $t$  при виконанні ресурсних обмежень, в рамках яких здійснюється розвиток і функціонування системи:  $R[E(t), A(t), X(t), Y(t), t]$ .

Встановимо зв'язок, між імовірністю своєчасного виконання БО (компетенцією виконавця) і витратами на виконання БО. В роботі [16] запропонований показник витратності виконання роботизованою БО виду

$$C_{BO} = \frac{\alpha_H e_H T O_H + \alpha_B e_B T O_B + (\alpha_H + \alpha_B) R_M}{C_H^b}, \quad (2.2)$$

де  $\alpha_H=1+(1-P_{SH})$ ,  $\alpha_B=1+(1-P_{SB})$  – коефіцієнти, що враховують збільшення часу і витрат внаслідок ймовірності виникнення помилки h-агента і b-агента,  $P_{SH}$  і  $P_{SB}$  – відповідно ймовірності успішного (безпомилкового) завершення операцій h-агента і b-агента – ці величини будемо надалі використовувати як чисельні метрики компетенції агентів;  $C_H^b$  – базова величина собівартості бізнес-операції, виконуваної виключно людиною і без помилок,

$$C_H^b = e_H^b \cdot TO_H^b + R_M^b, \quad (2.3)$$

де  $e_H^b$ ,  $TO_H^b$ ,  $R_M^b$  – відповідно, базова ціна робочого часу h-агента, базовий нормативний час, відведений за результатами вартісного аналізу на виконання даної операції людиною, базові нормативні витрати на експлуатацію та обслуговування технічних засобів протягом базового нормативного часу  $TO_H^b$ .

Позначивши через  $CD$  ймовірну відносну вартість затримки виконання БО, показник витратності виконання БП, що містить  $I$  БО, запишемо у вигляді

$$C^{БП} = \sum_{i=1}^I (V_i + CD_i), \quad i = \overline{1, I} \quad (2.4)$$

Таким чином, формалізовано задачу динамічної маршрутизації, що включає функціональні обмеження, які враховують застосування агентів-ботів, і критерій витрат виконання БО.

## 2.2 Основні положення стратегії управління чергами

В роботі [16], викладений метод динамічної маршрутизації заявок в ОТС, на підставі якого і будемо будувати адаптивну стратегію управління чергами.

Виконання заявок в ОТС регламентовано режимом реального часу. Введемо часові обмеження  $T_{max}^v$  на обслуговування заявки  $v$ -ї категорії  $\eta_v$ . Відповідно визначимо інтенсивність обслуговування  $\mu_v$  (величина зворотна до середньої номінальної тривалості  $\mu_v = 1/E(\eta_v)$ , де  $E(\eta_v)$  – математичне очікування тривалості обслуговування заявки  $v$ -ї категорії. Очевидно, що



величина  $\mu_v$  знаходиться у функціональній залежності від компетенції конкретного виконавця  $Z_n$ , що визначає його продуктивність. З урахуванням часових обмежень можна записати  $\mu_v = f(Z_n) \geq 1/T_{max}^v$ .

Оскільки кожен виконавчий вузол обслуговує кілька категорій заявок, необхідно визначити стратегію вибору маршруту БП, у згоді з метою мінімізації тривалості або вартості затримки виконання, а також дотримання квазістаціонарного режиму, при якому довжина черг до всіх АРМ не збільшується.

Цілі стратегії:

1. Мінімізація критерію (4) для кожного БП.
2. Забезпечення квазістаціонарного режиму обслуговування заявок на всіх АРМ.

Оскільки функція розподілу тривалості обслуговування для кожного АРМ заздалегідь не визначена, але відомі часові обмеження  $T_{max}^v$ , то адаптивну стратегію призначення пріоритетів визначимо наступним чином:

1. Прийmemo, що критерієм для адаптації стратегії є  $CD_{kn}$  – вартість затримки виконання  $k$ -ї заявки (БО) на  $n$ -му АРМ.

2. Множина заявок класифікується за обмеженням  $CD_{max}$  на 2 групи ризику (urgency group): UG1, UG2. На будь-який поточний момент існують підмножини заявок в кожній групі.  $k$ -я заявка включається в групу UG1 при виконанні умови:  $CD_k^{max} > CD_{max}$ . Інтенсивність контролю в «старшій» групі вище, ніж в «молодшій», що дає економію машинного часу. Кратність періодичності циклів контролю і переформування черг в групах можна регулювати.

3. Вибір підмножини АРМ для маршрутизації за критерієм компетенції.

3.1 Для маршрутизації кожної заявки визначається підмножина АРМ по необхідній для даної БО компетенції. Згідно з методикою викладеною в роботі [11], відбирається робоча підмножина АРМ, у яких компетенція  $h$ -агента більше заданого порогу компетенції. Серед цих АРМ відбувається вибір

конкретного вузла обслуговування за мінімальним значенням вартості очікування.

3.2 Поріг необхідної компетенції для заявки залежить від відносної ціни затримки  $cd_k$ . Значення порога визначається виразом

$$P_k = \frac{a}{1 + \exp(-\alpha \cdot cd_k + b)} + c, \quad (2.5)$$

де параметри  $a$ ,  $b$ ,  $c$  визначають межі зміни порогу компетенції (0,3...0,9), а параметр  $\alpha$  – крутизну зміни порога, що визначається.

3.3 При підвищенні порога компетенції робоча підмножина теоретично може виявитися порожньою, тому необхідний контроль потужності сформованої підмножини.

На рис. 2.1 показані криві для визначення індивідуального порога необхідної компетенції за формулою (5) при значеннях параметра  $\alpha=14$  (крива а) і  $\alpha=8$  (крива б).

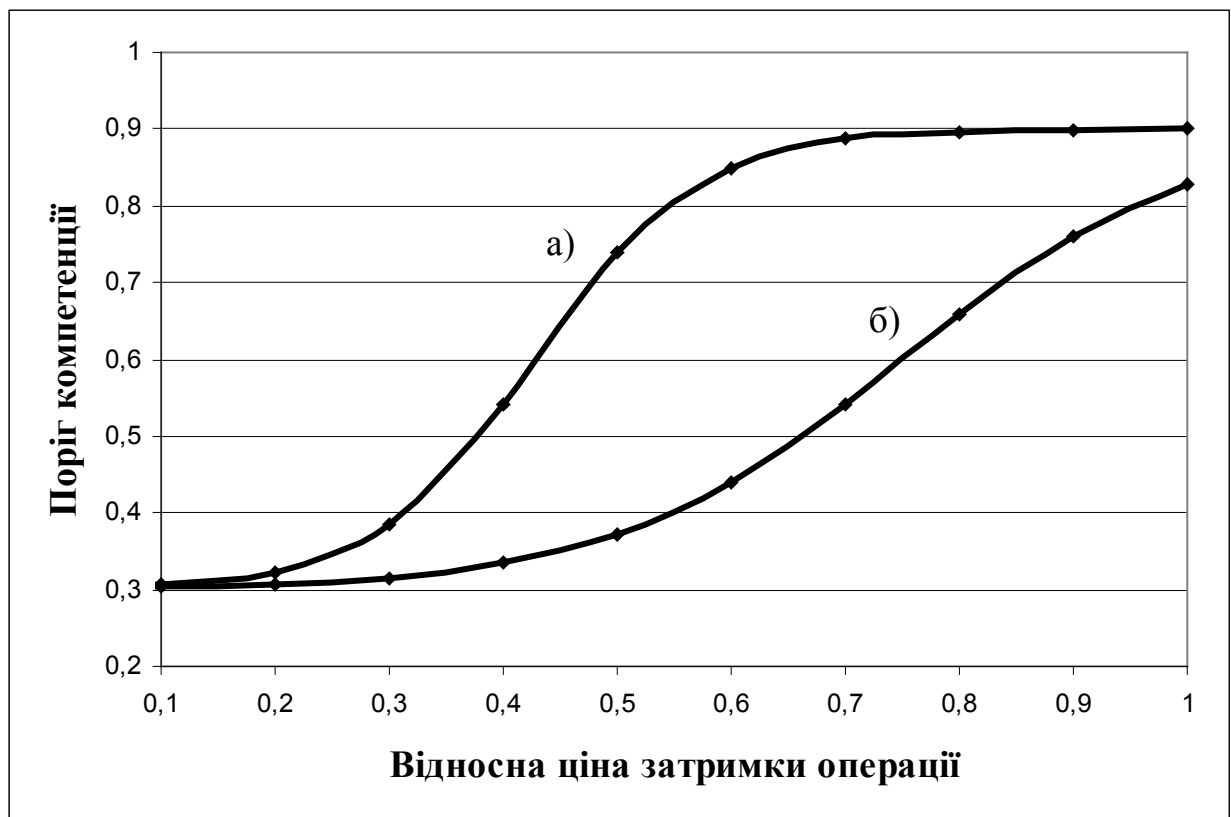


Рисунок 2.1 – Криві вибору порогу компетенції

Видно, що параметр  $\alpha$  істотно впливає на вибір робочої підмножини АРМ по компетенції виконавців.

4. Аналіз черг для вибору маршруту. Після розрахунку сумарного часу виконання  $TQ_n$  заявок, що стоять попереду в кожній  $n$ -ї черзі, з урахуванням ціни затримки  $cd_k$ , можна обчислити вартість затримки нової заявки  $CD_{kn} = CD_{kn} = cd_k \cdot TQ_n$  для кожної  $n$ -ї черги з робочої підмножини. Перебираючи чергі в межах робочої підмножини, знайдемо мінімальне значення  $CD^*_{kn}$  і поставимо заявку  $k$  в чергу  $n^*$ .

5. Сортування заявок в чергах по зменшенню ціни затримки  $c_{dk}$ .

6. Забезпечення квазістаціонарного режиму по кожному АРМ.

Як відомо, стаціонарний режим роботи СМО повинен бути незалежним від режиму роботи системи (перехідного чи усталеного) [13]. Для забезпечення стаціонарного режиму середні часові характеристики системи повинні бути приведені у відповідність з часовими обмеженнями, які визначені вихідними вимогами до системи. При динамічній маршрутизації заявок можна забезпечити так званий квазістаціонарний режим за рахунок зворотного зв'язку, коли диспетчер контролює тенденції зміни довжини черг і вживає заходів до того, щоб довжина кожної черги лишалася незмінною або зменшувалася. Тобто, для будь-якого каналу обслуговування повинна виконуватися умова  $\rho_n \leq \rho_T$ , де  $\rho_T$  – граничне значення завантаження, незалежно від поточного розподілу типів заявок.

6.1 Визначення вузьких місць за компетенціями. Для кожної підмножини заявок типу  $\nu$  (за компетенцією  $\nu$ ) обчислюється поточне значення ступеня завантаженості як відношення  $\rho_a$  середньої за контрольний період швидкості надходження заявок даного типу до середньої швидкості виходу обслужених заявок даного типу. Якщо  $\rho_a \geq \rho_{Th}$ , то по даній компетенції необхідно:

- ранжувати АРМ в робочій підмножині за рівнем компетенції типу  $\nu$ ;
- з урахуванням встановленого порога компетенції визначити цільове АРМ з меншою кількістю заявок з групи ризику UG1.

6.2. Якщо завантаження по компетенції нижче нижнього порога завантаження ( $\rho_a < \rho_{TL}$ ), тобто, АРМ часто простоює, здійснюється формування черги з заявок, що стоять в чергах на інші АРМ. З хвостових заявок в чергах до інших АРМ, утворюється і ранжується нова черга відповідно до компетенцій даного АРМ. Довжина черги вибирається за умовою:

$$L_{\text{нovoї черги}} \approx L_{\text{ср довжини черг що існують}}$$

6.3 Ранжувати черзі по спадаючій відносній ціні затримки.

Отже, отримала подальший розвиток адаптивна стратегія управління чергами в автоматизованих організаційно-технічних системах, за рахунок урахування в якості ознак ступеня інтенсивності потоку заявок, ступеня компетенції виконавців та ступеня завантаження агентів-виконавців, що дозволяє реалізувати процес структурної адаптації системи до змін у оперативній обстановці. Стратегія експериментально апробована на імітаційній моделі і показала свою працездатність.

### **2.3 Розробка методу і алгоритму оптимізації для оперативної адаптації стратегії управління чергами**

Для вибору параметрів стратегії використовується імітаційна модель, описана нижче. Робочими змінними в процесі адаптації є: поріг  $X1$  поділу заявок на групи ризику за вартістю  $CD_k$ ;  $X2$  – параметр  $\alpha$  виразу (2.5) для визначення порогу компетенції.

Метод оптимізації реалізується у вигляді наступних етапів:

1. Для імітаційного моделювання процесу багатофазного обслуговування і формування маршрутів заявок задаються імовірнісні розподіли потоку надходження БП, значення компетенцій, часові обмеження на виконання заявок різних типів (відповідних типів БО), відносні ціни затримок БО, характерні для даної ОТС.

2. За допомогою еволюційного алгоритму, викладеного нижче,

генерується популяція допустимих рішень у вигляді двійок  $(X1_r, X2_r)$ ,  $r=1...NP$ ., де  $NP$  – число членів популяції.

3. Кожна двійка (особина)  $(X1_r, X2_r)$  використовується в процесі імітаційного моделювання, для обчислення значення критерію оптимальності.

4. За допомогою еволюційних процедур, описаних нижче, відбирається найкраще рішення  $(X1^*_r, X2^*_r)$  за критерієм мінімізації сумарної вартості витрат і затримок (2.6)

$$Q = \sum_{j=1}^J \sum_{i=1}^{I_j} (C_{ij}^{BO} + CD_{ij}) \rightarrow \min, i = \overline{1, I}, j = \overline{1, J}, \quad (2.6)$$

де  $J$  – кількість запущених в роботу БП,  $I_j$  – кількість БО в  $j$ -м БП.

### **2.3.1 Розробка еволюційного алгоритму для вирішення задачі оптимізації**

Попередні дослідження показали, що критерій (2.5) є мультимодальною функцією, яка до того ж змінюється для різних ОТС і бізнес-процесів. З урахуванням цього факту для вирішення поставленого завдання доцільно використовувати адаптивні методи випадкового пошуку, розвитком яких є еволюційні алгоритми [18]. Основна продуктивна ідея еволюційних алгоритмів полягає в тому, що пошук ведеться по всій області допустимих значень і в процесі його виконання змінюються умови генерації вектора робочих параметрів. Зазвичай на початкових етапах розподіл точок є рівномірним, а потім кількість точок збільшується в тих областях, де ймовірно знаходяться локальні екстремуми цільової функції. Уточнення значень локальних екстремумів можна проводити за допомогою випадкового пошуку в вузькій області, яка, ймовірно, містить локальний екстремум. Імовірність швидкого сходження алгоритму при цьому значно зростає і, таким чином, прискорюється весь процес пошуку оптимального рішення. Отже, еволюційні алгоритми використовують інформацію про рельєф поверхні функції мети. З урахуванням викладеного сформулюємо основні правила, які повинен

використовувати алгоритм пошуку оптимального рішення:

1. Забезпечення рівномірного пошуку у всій області визначення функції мети на початковому етапі пошуку;
2. Забезпечення звуження меж випадкового пошуку в окремих областях, які імовірно містять локальні екстремуми;
3. Вузькі межі локального пошуку адаптивно зміщуються в міру уточнення значення локального екстремуму;
4. Вибір для локального аналізу не тільки елітарних точок, а й кількох випадкових точок;
5. З урахуванням того, що аналіз популяції ведеться за допомогою імітаційної моделі, необхідно вжити заходів щодо скорочення обсягу обчислень при аналізі популяції:
  - обмеження кількості членів популяції;
  - дискретизація рівнів робочих змінних в діапазоні  $[0,3 \dots 0,9]$  з кроком  $0,1$ ;
  - перевірка абсолютної ідентичності нової особини з екземплярами, наявними в глобальній таблиці особин.

Нижче наведені кроки алгоритму для реалізації розробленого методу:

Крок 1. Генерація вихідної популяції. Синтезується  $NP$  точок ( $k=1, \dots, NP$ ), в яких визначається значення цільової функції (6) (за допомогою імітаційної моделі і усереднення за  $NI$  реалізацій).

Крок 2. Сортування наборів (особин) по зростанню значень (оскільки в розв'язуваній задачі шукається мінімум цільової функції).

Крок 3. З усіх  $P$  точок відбирається  $\eta$  перших точок – претендентів на лідерство.

Крок 4. Обчислюються кордони пошуку уточненого значення екстремуму для кожної з  $\eta$  обраних точок:

$$a' = x^* - A(b-a)/2, \quad b' = x^* + A(b-a)/2, \quad (2.7)$$

де  $a', b'$  – звужені межі пошуку;  $a, b$  – початкові межі пошуку;  $x^*$  – координата обраного рішення;  $A$  – розмір, що обмежує кордони для уточненого пошуку

екстремуму.

Крок 6. Генерується  $M$  точок в кордонах  $a', b'$ .

Крок 7. Для кожної точки  $X_l$ ,  $l \in M$  обчислюється значення цільової функції.

Якщо нове значення менше попереднього, відповідно змінюються значення  $x^*$ .

Як результат в масиві популяції для обраних точок фіксуються нові значення координат і локальних екстремумів.

Крок 8. Координати і значення цільової функції для  $\eta$  лідерів заносяться в масив лідерів.

Крок 9. Якщо кількість заданих ітерацій не вичерпалося, то для всіх особин генеруються нові випадкові значення координат і відбувається перехід до кроку 3. Інакше перехід до кроку 10.

Крок 10. Масив лідерів сортується по зростанню цільової функції. Перший елемент в списку особин є знайденим оптимальним рішенням. Кінець.

Алгоритм має такі параметри:

- 1)  $NP$  – кількість членів популяції.
- 2)  $NI$  – кількість реалізацій в імітаційної моделі.
- 3)  $\eta$  – кількість претендентів в еліту.
- 4)  $A$  – коефіцієнт зменшення меж для уточненого пошуку екстремуму.
- 5)  $M$  – кількість пробних точок у вузьких межах уточненого пошуку.

Кожна епоха оптимізації режимів роботи СМО залишає в якості результату вихідні дані сеансу моделювання і кращий набір значень робочих параметрів. Збереження результатів в окремій таблиці надалі дозволяє відразу знаходити значення параметрів  $X1$ ,  $X1$  що близькі до оптимальних.

Таким чином, розроблений метод і алгоритм адаптації стратегії управління чергами в ОТС. При зміні відносного числа різнорідних БП і БО з'являється можливість адаптивного регулювання значень порогів  $X1$ ,  $X1$ , що дає можливість при зміні розподілу вхідних потоків вибрати найкращі умови

багатофазного обслуговування в процесі випадкового надходження вимог на виконання бізнес-процесів відомих класів.

## 2.4 Розробка структури імітаційної моделі

Оскільки аналітично вирішити сформульовану вище задачу оптимізації не представляється можливим, був обраний шлях імітаційного моделювання. Використання імітаційної моделі дозволяє провести цікаві в теоретичному і практичному відношенні дослідження СМО. В першу чергу це завдання аналізу системи. До них відносяться визначення показників ефективності та інших властивостей системи за відомими її параметрам: інтенсивності потоку вимог, кількості каналів і їх характеристикам, часу обслуговування і т. ін. Великий інтерес представляє дослідження впливу варіацій параметрів системи на показники, що характеризують її основні властивості. Зокрема, імітаційна модель необхідна для дослідження ефективності різних варіантів стратегій оперативного управління режимами роботи ОТС [19].

Структура імітаційної моделі показана на рис. 2.2. Всі операції програми диспетчує блок управління ходом моделювання. Користувач через блок інтерфейсу і за допомогою блоку формування вихідних даних фіксує бажані імовірнісні розподіли потоку надходження заявок, режими генерації потоків подій, матрицю значень компетенцій  $h$ -агентів, встановлює множину типів заявок і норми часу виконання в залежності від компетенції. Заповнюється також масив цін затримок виконання БО. Залежність часу виконання від рівня компетенції  $h$ -агента визначається коефіцієнтом збільшення часу  $\alpha_H = 1 + (1 - P_{SH})$ , якій враховано у формулі (2.2).

Після фіксації вихідних даних блок керування запускає генератор потоку подій і блок маршрутизації, який використовує поточні значення граничних параметрів стратегії управління чергами. Блок формування результатів проводить моніторинг процесу за параметрами завантаження вузлів мережі обслуговування, обчислює і фіксує поточні і підсумкові значення таких



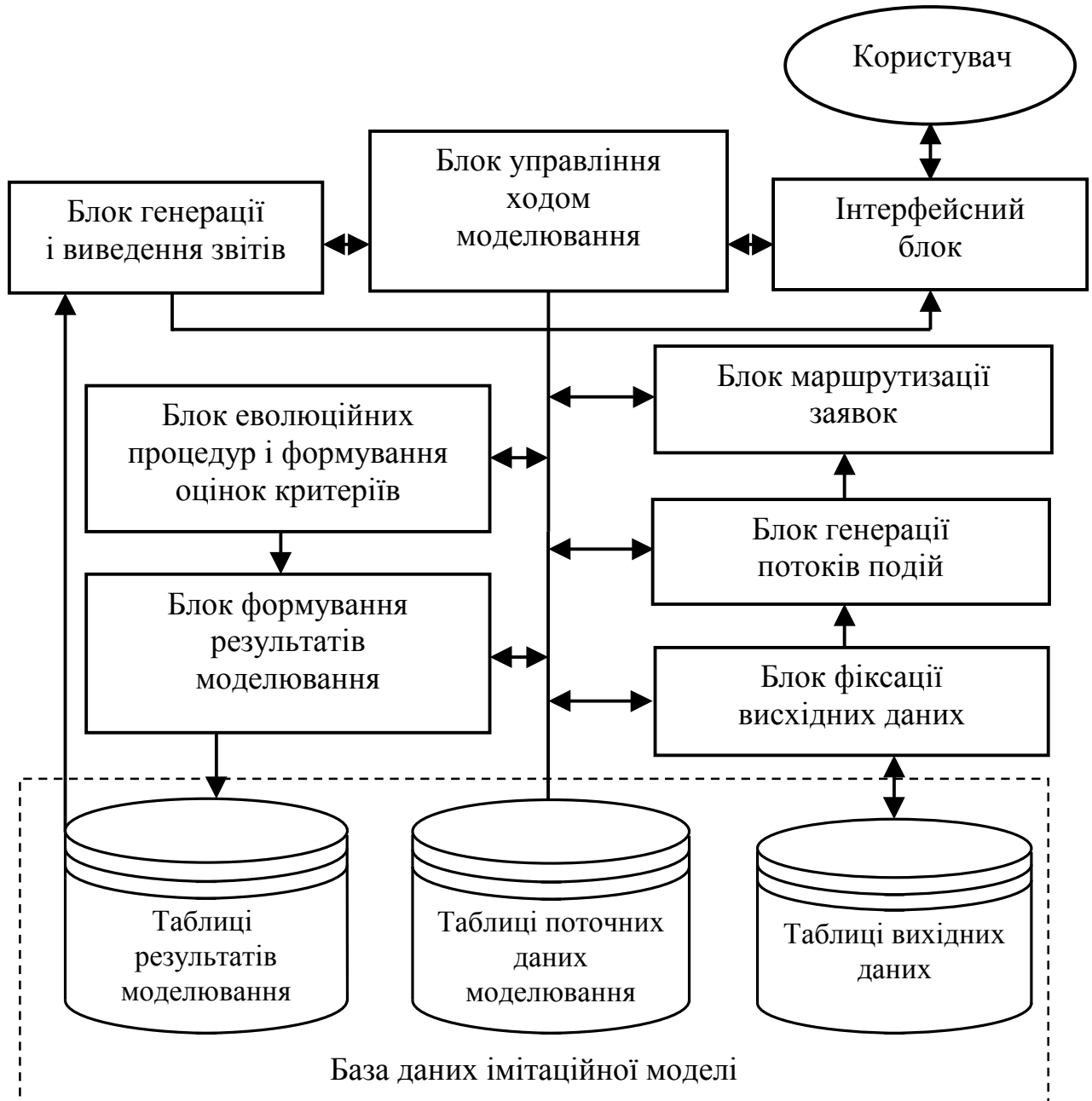


Рисунок 2.1 – Структура імітаційної моделі

параметрів: середня за контрольний період число заявок в чергах; середній час очікування; дисперсія часу очікування; втрати очікування; ступінь завантаженості АРМ.

Блок еволюційних процедур реалізує алгоритм відбору кращих варіантів пар значень параметрів стратегії і формує глобальну таблицю особин для запобігання повторення особин. Блок генерації і виведення звітів формує звіт за результатами моделювання для користувача. Користувач може запустити цикл автоматичної зміни значень основних вихідних даних (тип розподілу для вхідного потоку, крокові зміни інтенсивності надходження і обслуговування заявок, розподіл типів

заявок). Схема взаємозв'язку інформаційних процесів в інформаційній технології оптимізації стратегії управління чергами показано на рис. 2.2.

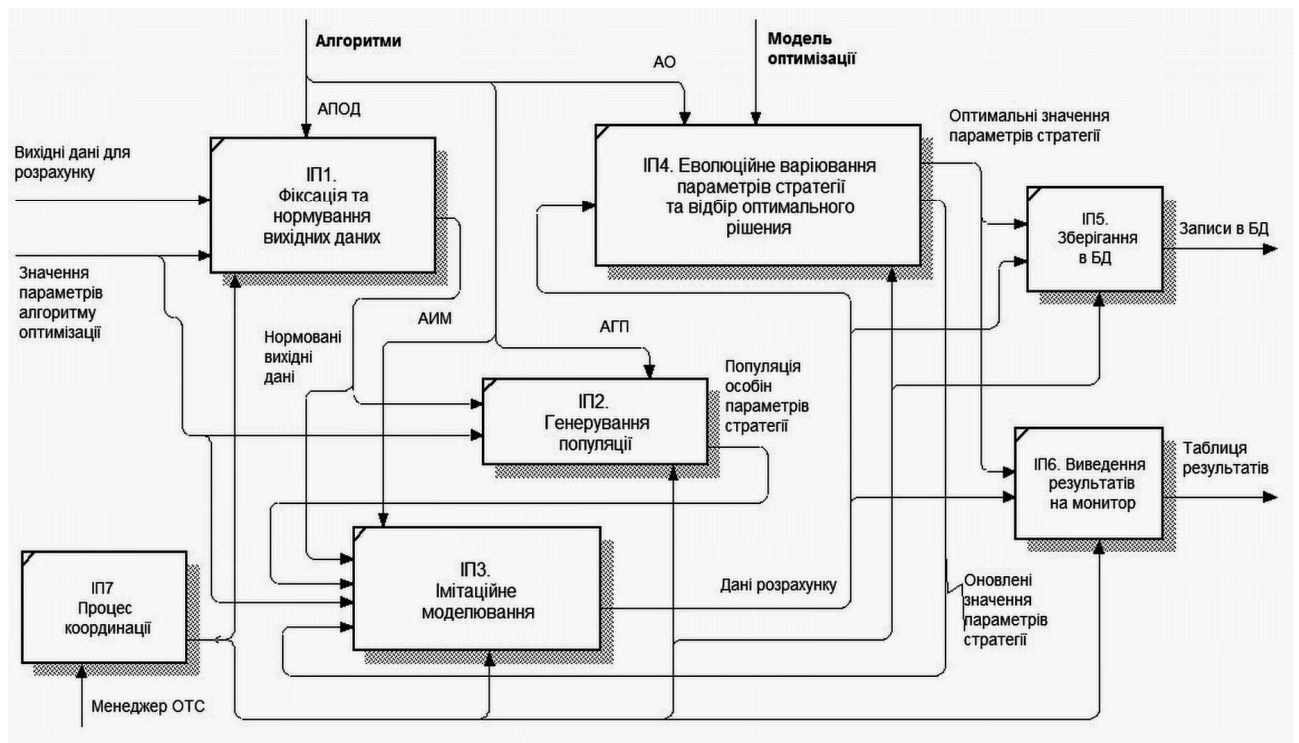


Рисунок 2.2 – Схема інформаційної технології оптимізації стратегії управління чергами

### Висновки з розділу

Поставлено та вирішено задачу розробки адаптивної стратегії управління чергами в автоматизованій організаційно-технічній системі. Стратегія враховує в якості ознак зміни розподілу потоку неоднорідних заявок, ступень компетенції виконавців та ступень завантаження агентів-виконавців, що дозволяє реалізувати процес структурної адаптації організаційно-технічній системі до мінливих оперативних обставин.

Розроблено метод і алгоритм адаптації стратегії управління чергами в організаційно-технічній системі, заснований на еволюційному підході до підбору оптимальних значень граничних параметрів відбору заявок за вартістю затримки і підбору компетенції співробітників.

Розроблено структуру імітаційної моделі процесу управління чергами в організаційно-технічній системі, яка слугує інструментом для експериментів у процесі оптимізації стратегії управління чергами заявок на виконання бізнес-операцій та дозволяє дослідити ефективність запропонованої стратегії.

## РОЗДІЛ 3

### ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

#### 3.1 Опис програмного засобу

Програмний засіб реалізовано як сукупність модулів на мові РНР з використанням хмарних технологій. Враховуючи обмеження на обсяг пояснювальної записки, опис програмного засобу винесено у Додаток А.

#### 3.2 Результати експериментів

Головний результат експериментів на імітаційній моделі – підтвердження гіпотези про те, що запропонована стратегія управління чергами працездатна і може бути адаптована до умов роботи ОТС, що змінюються, а саме – до змін інтенсивності потоків різнорідних заявок, ступеня компетенції виконавців, ступеня завантаження АРМ. На рис. 3.1 показана залежність дисперсії часу очікування від значень порога поділу заявок на групи ризику по вартості затримок. При збільшенні кратності періодичності циклів контролю і переформування черг дисперсія збільшується.

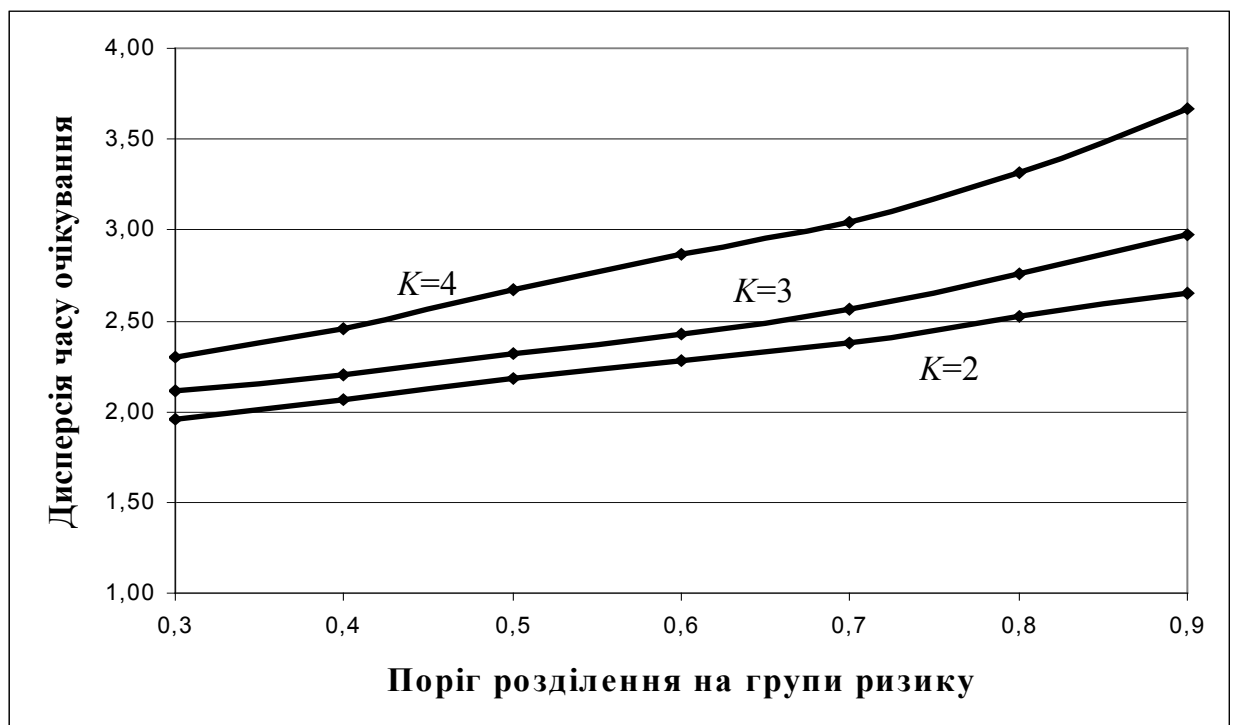


Рисунок 3.1 – Залежність дисперсії часу очікування від порога поділу заявок на групи і кратності  $K$  періодичності циклів контролю черг в групах

На рис. 3.2 показано залежність критерію втрат від середньої інтенсивності надходження заявок в систему. Крива а) відображає цю залежність при випадкових значеннях параметрів  $X1$  і  $X2$ . Крива б) показує ту ж залежність при оптимальних значеннях параметрів  $X1$  і  $X2$ . На кривій б) видно ділянку стабілізації втрат за рахунок оперативного зміни маршрутів заявок. У міру збільшення інтенсивності потоку заявок стабілізація стає практично неможливою і втрати зростають. Разом з тим видно, що адаптивна стратегія є ефективним інструментом зниження втрат очікування.

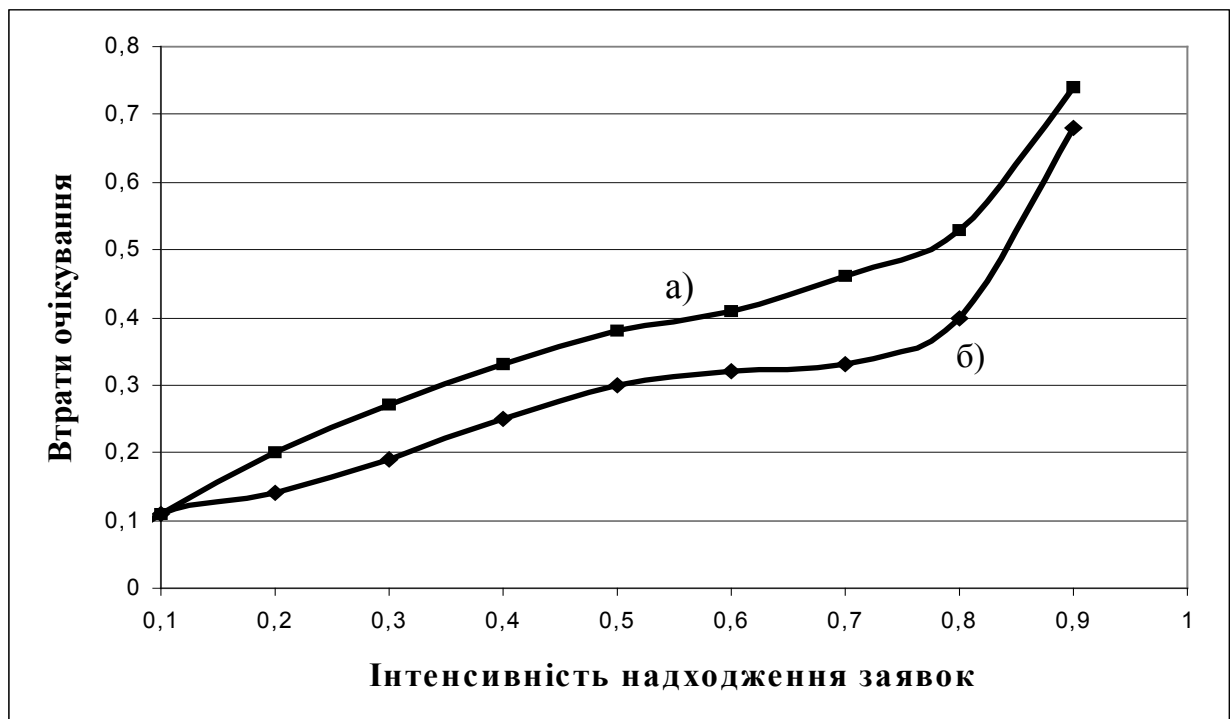


Рисунок 3.2 – Залежність витрат очікування від параметрів стратегії

Середній нормований час перебування в черзі для заявок з групи UG1 становив 0,63 при випадкових значеннях параметрів налаштування і 0,45 при оптимальних значеннях цих параметрів. Для групи UG2 ці значення відповідно склали 0,68 і 0,56.

### Висновки з розділу

Експерименти з імітаційною моделлю показали можливість зниження втрат очікування при реалізації методу адаптації стратегії управління чергами. Для адаптації стратегії необхідно мати таблицю рішень, яка побудована за результатами імітаційних експериментів з урахуванням особливостей ОТС.

## ВИСНОВКИ

На основі системних досліджень проблеми оперативного управління в автоматизованих організаційно-технічних системах формалізовано задачу динамічної маршрутизації, що включає функціональні обмеження, які враховують застосування агентів-ботів, і критерій витрат виконання бізнес-операцій для кожного бізнес-процесу, що реалізується в організаційно-технічній системі.

Розроблено основні положення адаптивної стратегії управління чергами в автоматизованій організаційно-технічній системі, яка відрізняється тим, що за рахунок обліку в якості ознак зміни розподілу потоку неоднорідних заявок, ступеня компетенції виконавців, ступеня завантаження агентів-виконавців, дозволяє реалізувати процес структурної адаптації організаційно-технічній системі до мінливих оперативних обставин.

Розроблено метод і алгоритм адаптації стратегії управління чергами в організаційно-технічній системі, заснований на еволюційному підході до підбору оптимальних граничних значень параметрів селекції заявок за вартістю затримки і рівня компетенції співробітників.

Розроблено структуру імітаційної моделі процесу управління чергами в організаційно-технічній системі, яка слугує інструментом для експериментів у процесі оптимізації стратегії управління чергами заявок на виконання бізнес-операцій та дозволяє дослідити ефективність запропонованої стратегії.

Адаптивна стратегія управління чергами в організаційно-технічній системі апробована на імітаційній моделі. Результати статистичної обробки даних, отриманих в ході імітаційних експериментів, підтвердили можливість зниження втрат очікування при реалізації методу адаптації стратегії управління чергами. Оперативна зміна значень параметрів стратегії дозволяє утримувати квазістаціонарний режим обслуговування в широкому діапазоні інтенсивностей надходження заявок.

## АНОТАЦІЯ

Організаційно-технічна система (ОТС) являє собою сукупність технічних і людських ресурсів, призначену для виконання певної множини бізнес-процесів (БП). Кожен БП проходить ряд етапів, які представляють собою окремі бізнес-операції (БО), що виконуються на конкретних автоматизованих робочих місцях (АРМ). Будь-який БП створює потік заявок на виконання БО. Потреба в динамічному формуванні виконавчих структур в рамках ОТС виникає з протиріччя між вимогою гарантованого виконання множини бізнес-операцій у встановлені терміни і обмеженнями на технологічні і організаційні можливості ОТС. Як показує огляд літератури і практика, проблема оптимізації процесу динамічного розподілу навантаження на АРМ при реалізації БП як і раніше не вирішена. Вирішення цієї проблеми є актуальним завданням.

Метою роботи є розробка адаптивної стратегії управління чергами при виборі маршрутів для множини різнорідних бізнес-процесів і потоків заявок в рамках автоматизованої організаційно-технічної системи з урахуванням певного критерію оптимальності та деяких обмежень. Для досягнення поставленої мети в роботі вирішуються наступні завдання:

- аналіз проблем розподілу функцій і робіт в організаційних системах з точки зору теорії масового обслуговування;
- побудова стратегії управління чергами при виборі маршрутів бізнес-процесів через множину робочих місць, здатної до адаптації в умовах, що змінюються під час функціонування ОТС;
- розробка методу адаптації стратегії управління чергами при змінах у розподілі потоків заявок різного типу;
- розробка імітаційної моделі для реалізації та апробації методу оптимізації адаптивної стратегії.

Отримала подальший розвиток адаптивна стратегія управління чергами в ОТС, за рахунок обліку змін розподілу потоку неоднорідних заявок, ступеня компетенції виконавців, ступеня завантаження агентів-виконавців, що дозволяє реалізувати процес структурної адаптації ОТС до мінливих оперативних обставин.

Розроблено метод і алгоритм адаптації стратегії управління чергами в ОТС, заснований на еволюційному підході до підбору оптимальних граничних значень параметрів селекції заявок за вартістю затримки і рівня компетенції співробітників.

Розроблена адаптивна стратегія управління чергами в ОТС апробована на імітаційній моделі. Результати статистичної обробки даних, отриманих в ході імітаційних експериментів, підтвердили можливість зниження втрат очікування при реалізації методу адаптації стратегії управління чергами. Оперативна зміна значень параметрів стратегії дозволяє утримувати квазістаціонарний режим обслуговування в широкому діапазоні інтенсивностей надходження заявок.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Аксенов К.А., Гончарова Н.В. Динамическое моделирование мультиагентных процессов преобразования ресурсов. Екатеринбург: ГОУ ВПО УГТУ-УПИ, 2006. 311 с.
2. Гибсон Дж. Л. Организации: поведение, структуры, процессы / Гибсон Дж. Л., Иванцевич Д. М., Донелли Д. Х. : пер. с англ. – 8-е изд. – М. : ИНФРА-М, 2000. – 662 с.
3. Петренко С. А. Порівняльний аналіз моделей організаційних структур підприємства / С. А. Петренко // Бюлетень Міжнародного Нобелівського економічного форуму, 2010. Том 2. № 1(3). С. 245-252.
4. Баронов В. В., Калянов Г. Н., Попов Ю. Н., Титовский И. Н. Информационные технологии и управление предприятием /. – М. : Компания АйТи, 2005. 328 с.
5. Бронштейн О. И., Рыков В.В. Об оптимальных дисциплинах обслуживания в управляющих системах. В сб. «Управление производством». Тр. III Всес. совещ. гр автомат, упр. (техн. кибернет.), 1967. М., Наука. С. 56-61.
6. Рыков В.В. Управляемые системы массового обслуживания, Итоги науки и техн. Сер. Теор. вероятн. Мат. стат. Теор. кибернет., 1975, том 12, 43–153.
7. Введенская Н.Д., Добрушин Р.Л., Карпелевич Ф.И. Система обслуживания с выбором наименьшей из двух очередей – асимптотический подход. *Проблемы передачи информации*. 1996. Т. 32. № 1. С. 20–34.
8. Suhov Yu.M., Vvedenskaya N.D. Fast Jackson Networks with Dynamic Routing. *Problems of Information Transmission*. 2002. Vol. 38. No. 2. PP. 136–153.
9. Жернов Ю.В. Многолинейные системы с отказами и равновероятным распределением заявок. *Информационные процессы*. 2013. Том 13. № 1. С. 19–34.
10. Жернов Ю.В., Жернов К.Ю. Метод потенциалов для системы M/G/1/m с активным управлением очередью. *Информационные процессы*. 2015. Том 15. № 1. С. 66–77.

11. Аленичев А.В. Система массового обслуживания с динамической маршрутизацией и распределением Вейбулла времени обслуживания. *Информационные процессы*. 2005. Том 5. № 5. С. 432–442.
12. Аленичев А.В., Лиханов Н.Б. Динамическая маршрутизация в системе с заявками, имеющими степенной закон распределения времени обслуживания. *Информационные процессы*. 2005. Т. 5. №3. С. 213–226.
13. Корячко В.П., Перепелкин Д.А. Математическая модель адаптивной маршрутизации в корпоративных сетях. *Информационные технологии в образовании*. 2012. С. 47–55.
14. Поздняк И. С. Формирование множества допустимых маршрутов с использованием алгоритма адаптивной маршрутизации. *Тез. докл. XIV российской научной конференции ПГАТИ* (г. Самара, Россия, 2007). С. 61–62.
15. Оксанич И.Г., Шевченко И.В., Краснопольская Ю.А. Отображение описания бизнес-процесса в операционное пространство организационно-технической системы. *Радиоэлектроника та інформатика*. 2019. №2. С. 54–60.
16. Шевченко И.В., Оксанич И.Г., Конох И.С. Модель и метод динамического формирования исполнительных структур в роботизированных организационно-технических системах. *Вісник КрНУ імені Михайла Остроградського*. 2019. Вип. 5 (118). С. 103–109.
17. Габагин А.В., Разбегин В.П. Анализ и синтез структуры Workflow-систем. *Международная конференция «Системы проектирования, технологической подготовки производства и управления этапами жизненного цикла промышленного продукта CAD/CAM/PDM»* (г. Москва, Россия, 26-28 октября 2011 г.) С. 34.
18. Курейчик В. М. Генетические алгоритмы и их применение. Таганрог: ТРТУ, 2002. 242 с.
19. Алиев Т.И. Основы моделирования дискретных систем. СПб: СПбГУ ИТМО, 2009. 363 с.



## ДОДАТОК А

Опис програмного модуля

Програмний модуль реалізовано на мові PHP. Мову PHP вибрано, оскільки проект, в який входить даний засіб, є веб-базованим. Функціонал програми відповідає схемі інформаційної технології, яка показана на рис. 2.2. Розглянемо послідовно відеограми вікон програми.

На рис. А1 показано вікно «Вихідні дані». На даному вікні розміщено поля вводу позначень класів БП, їх назв, та номерів екземплярів окремих класів БП. Оскільки мова іде про моделювання процесу обслуговування, для кожного БП вказується час запуску. Окрім того, необхідно задати початкові значення порогів стратегії керування чергами.

Клас БП	№	Назва	Час запуску, хв
A	1	Прийом на роботу	1
A	2	Прийом на роботу	13
A	3	Прийом на роботу	15
A	4	Прийом на роботу	16
B	1	Направлення на навчання	35
B	2	Направлення на навчання	50
B	3	Направлення на навчання	121
C	1	Переведення	122
C	2	Переведення	144
D	1	Звільнення	179

**Початкові значення порогів стратегії**

Поріг поділу на групи терміновості  
 0.2-0.9


Крутизна зміни порога компетенції  
 0.2-1.0

**Застосувати**

Рисунок А1 – Вікно введення початкових даних

На рис. А2 показано перше вікно модулю «Моделювання». Модуль забезпечує фіксацію загального часу моделювання в хвилинах, можливість запуску процесу, та відстеження і фіксацію ступеню завантаженості усіх АРМ. Модуль накопичує поточні дані про довжину та відносну вартість черг, сумарні втрати очікування по кожному АРМ та інші показники, повний перелік котрих можна бачити на рис. А3 – другому вікні модулю «Моделювання».

ОТС Вихідні дані Моделювання Оптимізація стратегії Налаштування

Задати час процесу, хв:   **Почати моделювання**

Поточний стан **Результати**

АРМ	Монітор	Завантаженість
АРМ 1	Монітор 1	0,9
АРМ 2	Монітор 1	1,1
АРМ 3	Монітор 2	0,3
АРМ 4	Монітор 2	0,93
АРМ 5	Монітор 2	1,0

Рисунок А2 – Перше вікно модулю Моделювання» – «Поточний стан»

ОТС Вихідні дані Моделювання Оптимізація стратегії Налаштування

Задати час процесу, хв:  **Почати моделювання**

Поточний стан **Результати**

**Втрати очікування** **Зберегти в Excel**

БП	Б0 1	Б0 2	Б0 3	Сума (ціна = 3)
А1	20	100	7	381
В2	5	0	4	27

**Сумарні втрати очікування по АРМ** **Зберегти в Excel**

АРМ	Сума (ціна = 3)
<b>АРМ 1</b>	
Середній час очікування, хв	15,87
Середнє число заявок в чергах	7,36
Середній ступінь завантаженості АРМ	0,68
Максимальний ступінь завантаженості АРМ	0,86
Мінімальний ступінь завантаженості АРМ	0,31
Дисперсія часу очікування (група 1)	28,34
Дисперсія часу очікування (група 2)	37,45

Рисунок А3 – Друге вікно модулю «Моделювання» – «Результати»

Результати, отримані у процесі однієї реалізації, можна зберегти у таблицях засобу Excel. На рис. А4 показане вікно модулю «Оптимізація стратегії – Налаштування». Користувач вводить розмір популяції значень параметрів стратегії, число циклів еволюційного алгоритму, число пробних точок при уточненні координат лідерів, число лідерів, коефіцієнт зменшення меж при уточненні координат лідерів. На рис. А5 показано вікно «Оптимізація стратегії – Результати». Після завершення процедури оптимізації з багаторазовим зверненням до модулю імітаційного моделювання модуль повертає оптимальні для даного варіанту розподілу інтенсивностей заявок значення параметрів  $X1$  та  $X2$ .

Рисунок А4 – Вікно Оптимізація стратегії – Налаштування

Рисунок А4 – Вікно Оптимізація стратегії – Результати

На рис. А6 показано вікно «Налаштування». Користувач завдає число АРМ, кількість класів БП та БО, розподіл класів БО по кожному БП, порядок слідування БО у кожному БП. Окремо задаються рівні компетенції співробітників по кожній БО та часові норми на виконання БО. Користувач також визначає закон розподілу інтенсивності появи заявок у потоках БП.

ОТС Вихідні дані Моделювання Оптимізація стратегії Налаштування

Число АРМ: 5 (5-20)

Кількість класів БП: 7 (5-10)

Кількість класів БО: 10 (8-20)

**Застосувати**

**Розподіл БО по БП**

БП	БО 1	БО 2	БО 3	БО 4	БО 5	БО 6	БО 7	БО 8	БО 9	БО 10
A	2	5	1	7	6	4	0	3	5	8
B	1	9	5	8	7	3	4	2	7	6
C	0	0	1	2	5	3	0	4	4	6
D	9	5	1	10	2	4	7	3	8	6
E	1	7	6	2	3	5	8	9	5	6
F	3	6	0	0	1	2	4	5	2	0
G	4	5	2	5	1	3	0	0	6	0

**Компетенції**

БО	АРМ 1	АРМ 2	АРМ 3	АРМ 4	АРМ 5
БО 1	0,54	0,31	0,77	0,56	0,53
БО 2	0,05	0,16	0,87	0,23	0,48

**Часові норми**

БО 1	БО 2	БО 3	БО 4	БО 5	БО 6	БО 7	БО 8	БО 9	БО 10
15	7	97	34	36	75	49	11	85	68

**Тип розподілу**

Показовий **λ**: 0.05 (0.05-0.9)

Рівномірний тип, інтервал часу: 5 (5-10)

**Застосувати**

Рисунок А6 – Вікно «Налаштування»

Нижче наведено вихідний код модулю оптимізації стратегії управління чергами та код контролера зв'язку. Вихідний код модулю імітаційного моделювання не наведено. Цей модуль, якій має величезний обсяг, розроблено іншим автором.

### **//Оптимізація параметрів стратегії**

```

namespace App\Console\Commands;
use App\Contracts\IProcessor;
use Illuminate\Console\Command;
class Optimize extends Command
{
    /**
     * The name and signature of the console command.
     *
     * @var string
     */
    protected $signature = 'processor:optimize';
    /**
     * The console command description.
     *
     * @var string
     */
    protected $description = 'Optimize process';
    /**
     * @var IProcessor
     */
    private $processor;
    /**
     * Optimize constructor.
     *
     * @param IProcessor $processor
     */
    public function __construct(IProcessor $processor)
    {
        parent::__construct();
        $this->processor = $processor;
    }
    /**
     * Генерування популяції
     *

```

```

* @param $minX1
* @param $maxX1
* @param $minX2
* @param $maxX2
*
* @return array
*/
public function generatePopulation($minX1, $maxX1, $minX2, $maxX2)
{
    $elCount = 100;
    $stepX1 = abs($maxX1 - $minX1) / $elCount;
    $stepX2 = abs($maxX2 - $minX2) / $elCount;
    $params = [];
    for ($i = 0; $i < $elCount; $i++) {
        $params[] = [
            'x1' => ($i + 1) * $stepX1,
            'x2' => ($i + 1) * $stepX2,
        ];
    }
    return $params;
}
/**
 * Отримання результатів для конкретної популяції
 *
 * @param array $bps
 * @param array $population
 *
 * @return array
 */
public function populationResult(array $bps, array $population)
{
    foreach ($population as $index => $item) {
        $population[$index]['result'] = $this->processor
            ->getResultFunction($bps, $item['x1'], $item['x2']);
    }
    return collect($population)
        ->sortBy('result')
        ->slice(0, 10)
        ->toArray();
}
/**
 * Execute the console command.
 *
 * @return mixed

```

```

*/
public function handle()
{
    $minX1 = 2;
    $maxX1 = 9;
    $minX2 = 2;
    $maxX2 = 10;
    // Отримання масиву вхідних даних про потоки
    $bps = $this->processor->getInputBps();
    $populations = [];
    $steps = 10;
    for ($i = 0; $i < $steps; $i++) {
        // Генерування популяції для заданих границь X1 і X2
        $population = $this->generatePopulation($minX1, $maxX1, $minX2,
            $maxX2);
    // Отримання результатів для масиву БП і генерованої популяції
        $populationResult = collect($this->populationResult($bps,
            $population))
            ->sortBy('result');
        $populations[] = $populationResult->toArray();
        // Оновлення мінімумів для звужених границь
        $minX1 += $minX1 / $steps;
        $minX2 += $minX2 / $steps;
    }
    // Вибір кращих значень параметрів
    $bestEntities = collect($populations)
        ->collapse()
        ->sortBy('result')
        ->slice(0, 10)
        ->map(function ($pair) {
            return implode('; ', $pair);
        })
        ->toArray();
    echo 'Кращі варіанти:' . PHP_EOL . implode(PHP_EOL,
        $bestEntities) . PHP_EOL;
}
}

```



**// Взаємодія клієнта, сервера і модулю оптимізації**

```

namespace App\Http\Controllers;
use App\Contracts\IProcessor;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Artisan;
class IndexController extends Controller
{
    /**
     * @var IProcessor
     */
    protected $processor;
    /**
     * IndexController constructor.
     */
    public function __construct()
    {
        $this->processor = app()->make(IProcessor::class);
    }
    /**
     * Запитання оптимізації
     *
     * @param Request $request
     *
     * @return \Illuminate\Http\JsonResponse
     */
    public function index(Request $request)
    {
        // Отримання вхідних параметрів для клієнтів
        $x1 = (float)$request->get('x1');
        $x2 = (float)$request->get('x2');
        // Отримання результатів генерування і обробки популяції
        // Повертає 10 кращих варіантів
        $results = Artisan::call('processor:optimize', [
            'minX1' => $x1 - $x1 / 2,
            'minX2' => $x2 - $x2 / 2,
            'maxX1' => $x1 + $x1 / 2,
            'maxX2' => $x2 + $x2 / 2,
        ]);
        // Передача результатів клієнту
        return response()->json($results);
    }
}

```