

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ПРИВАТНЕ АКЦІОНЕРНЕ ТОВАРИСТВО «ПРИВАТНИЙ ВИЩИЙ
НАВЧАЛЬНИЙ ЗАКЛАД «ЗАПОРІЗЬКИЙ ІНСТИТУТ ЕКОНОМІКИ ТА
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ»

Кафедра інформаційних технологій

ДО ЗАХИСТУ ДОПУЩЕНА

Зав. кафедрою _____

д.е.н., доц. Левицький С.І.

КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА

МОДЕЛЮВАННЯ ПРОДУКТИВНОСТІ ПРОМІЖНИХ ВУЗЛІВ
КОМП'ЮТЕРНОЇ МЕРЕЖІ

Виконав
ст. гр. КІ-129

Р.С. Гармаш

Керівник
професор

Н.Р. Полуектова

Запоріжжя

2023

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ БАКАЛАВРСЬКУ РОБОТУ
студенту гр. КІ-129, спеціальності 121 - «Комп'ютерна інженерія»

Гармашу Роману Станіславовичу

1. Тема: «Моделювання продуктивності проміжних вузлів комп'ютерної мережі»

затверджена наказом № 02-02 від 11 січня 2023 р.

2. Термін здачі студентом закінченої роботи: 16 червня 2023 р.

3. Перелік питань, що підлягають розробці:

1. Провести огляд літератури, що присвячена тематиці досліджень.

2. Провести аналіз існуючих методів та інструментів дослідження продуктивності проміжних вузлів комп'ютерної мережі.

3. Обґрунтувати вибір методів та програмного забезпечення для моделі простого проміжного вузла мережі

4. Розробити імітаційну модель роботи проміжного вузла.

5. Провести експерименти з моделлю.

6. Зробити висновки за результатами експериментів

7. Оформити звіт за результатами роботи

КАЛЕНДАРНИЙ ГРАФІК
 підготовки кваліфікаційної бакалаврської роботи
 здобувачем освіти інституту ЗІЕІТ денної форми навчання
 гр. КІ-129 Гармашем Романом Станіславовичем
 2022-2023 навчальний рік

№ етапу	Зміст	Терміни виконання	Готовність по графіку %, підпис керівника	Підпис керівника про повну готовність етапу, дата
1	Збір практичного матеріалу за темою кваліфікаційної бакалаврської роботи	16.01.23-11.02.23		
2	I атестація I розділ кваліфікаційної бакалаврської роботи	27.03.23-31.03.23		
3	II атестація II розділ кваліфікаційної бакалаврської роботи	24.04.23-28.04.23		
4	III атестація III розділ кваліфікаційної бакалаврської роботи, висновки та рекомендації, додатки, реферат	22.05.23-26.05.23		
5	Перевірка кваліфікаційної бакалаврської роботи на оригінальність	15.05.23-12.06.23		
6	Доопрацювання кваліфікаційної бакалаврської роботи, підготовка презентації, отримання відгуку керівника і рецензії	29.05.23-12.06.23		
7	Попередній захист кваліфікаційної бакалаврської роботи	12.06.23-18.06.23		
8	Подача кваліфікаційної бакалаврської роботи на кафедру	за 3 дні до захисту		
9	Захист кваліфікаційної бакалаврської роботи	19.06.23-24.06.23		

Керівник _____ Н.Р. Полуктова

Здобувач освіти _____ Р.С. Гармаш

РЕФЕРАТ

Кваліфікаційна бакалаврська робота містить 73 сторінки, 44 рисунки, 32 бібліографічних посилання.

Метою роботи є розробка імітаційної моделі проміжного вузла комп'ютерної мережі для дослідження його продуктивності.

Об'єктом дослідження є проміжні вузли комп'ютерної мережі.

Предметом дослідження є моделювання проміжних вузлів.

В роботі описані принципи побудови сучасних комп'ютерних мереж, методи оцінки продуктивності вузлів мережі та огляд сучасних методів дослідження їх параметрів, ділі розглянуто призначення та основні принципи, методи та моделі імітаційного моделювання. В результаті розроблено дискретно-подієву імітаційну модель проміжного вузла мережі яка дозволяє дослідити основні параметри продуктивності та спрогнозувати поведінку роботи вузла в різних умовах навантаження.

ANYLOGIC, ПРОМІЖНИЙ ВУЗОЛ, ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ,
КОМП'ЮТЕРНА МЕРЕЖА

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	7
ВСТУП	8
РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМИ ДОСЛІДЖЕННЯ ПРОДУКТИВНОСТІ ПРОМІЖНИХ ВУЗЛІВ КОМП'ЮТЕРНИХ МЕРЕЖ.....	11
1.1. Принципи побудови сучасних комп'ютерних мереж	11
1.1.1 Поняття та види вузлів	12
1.1.2 Топології мережі.....	14
1.2 Методи оцінки продуктивності вузлів мережі.....	21
1.3 Огляд сучасних методів дослідження параметрів вузлів комп'ютерних мереж.	25
РОЗДІЛ 2. ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЯК МЕТОДОЛОГІЯ ДОСЛІДЖЕННЯ ХАРАКТЕРИСТИК КОМП'ЮТЕРНИХ МЕРЕЖ.....	30
2.1. Призначення та основні принципи імітаційного моделювання.....	30
2.2. Види імітаційних моделей	33
2.3. Системи масового обслуговування.....	37
2.4. Інструментальні засоби для побудови імітаційних моделей ..	42
2.5 Обґрунтування вибору методів та інструментів імітаційного моделювання для оцінки продуктивності проміжних вузлів комп'ютерних мереж	45
РОЗДІЛ 3. ПОБУДОВА ТА ВИКОРИСТАННЯ ІМІТАЦІЙНОЇ МОДЕЛІ ДЛЯ ОЦІНКИ ПРОДУКТИВНОСТІ ПРОМІЖНОГО ВУЗЛА МЕРЕЖІ	47
3.1. Опис математичної моделі.....	47

3.2.	Побудова імітаційної моделі в середовищі AnyLogic	51
3.3	Проведення та аналіз експериментів	63
ВИСНОВКИ.....		68
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ		70

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

Скорочення	Повна назва	Пояснення/переклад
NIC	Network Interface Controller	Мережевий інтерфейс карти
Wi-Fi	Wireless Fidelity	Технологія бездротової мережі
IP	Internet Protocol	Інтернет протокол
WLAN	wireless local area network	бездротова локальна мережа
WAN	Wide Area Network	Глобальна мережа
M&S	Modeling and Simulation	Моделювання та імітація
GPSS	General Purpose Simulation System	система моделювання загального призначення
СМО	Система масового обслуговування	Система масового обслуговування
FIFO	First In, First Out	Перший зайшов - перший вийшов
LIFO	Last In, First Out	Останній зайшов – перший вийшов
MAC-адреса	Media Access Control	Управління доступом до посередників
OSI	Open Systems Interconnection	Базова еталонна модель взаємодії відкритих систем

ВСТУП

В останні десятиріччя активно розвиваються мережеві технології, створюються все більш продуктивні технології розподіленого збереження та розподіленої обробки даних. Але всі сучасні технології побудовані на класичних принципах архітектури мереж, які включають сукупність вузлів, які пов'язані між собою. Серед цих вузлів виділяють модеми, комутатори, концентратори, мости, сервери та принтери, це все є вузлами комп'ютерної мережі, так само як і будь-які інші пристрої, які підключаються через Wi-Fi або Ethernet. Наприклад, якщо в мережі з'єднані три комп'ютери та один принтер, а також два бездротових пристрої, то загалом мережа складається з шести вузлів.

Продуктивність роботи мереж значним чином залежить від продуктивності окремих вузлів. Для оцінки продуктивності вузлів розроблено багато показників, які можуть вимірюватись на реальних працюючих системах, але, більш ефективно вимірювати та прогнозувати продуктивність вузлів при різних рівнях навантаження в мережі за допомогою математичних та імітаційних моделей.

Актуальність моделювання роботи вузлів комп'ютерної мережі полягає в кількох ключових аспектах:

Оптимізація та планування ресурсів. Моделювання дозволяє попередньо оцінити навантаження на вузли мережі та оптимізувати їх конфігурацію та розподіл ресурсів. Це допомагає уникнути перевантажень і забезпечити ефективне використання доступних ресурсів.

Прогнозування продуктивності. Моделювання дозволяє оцінити продуктивність вузлів мережі в різних умовах навантаження та топології. Це допомагає виявити вузькі місця та потенційні проблеми продукування, такі як затримки та втрата пакетів, і вжити заходів для їх усунення.

Аналіз відмовостійкості. Моделювання дозволяє досліджувати

поведінку мережі у разі відмов вузлів або мережевих зв'язків. Це допомагає визначити вразливість і розробити стратегії резервування ресурсів і відновлення роботи мережі у випадку збоїв.

Тестування нових технологій і сценаріїв. Моделювання дозволяє проводити віртуальні експерименти з новими технологіями, протоколами і сценаріями роботи мережі. Це допомагає оцінити їх ефективність, прийняти рішення щодо впровадження та знизити ризики, пов'язані з реальними випробуваннями на живих системах.

Навчання та освіта: Моделювання роботи вузлів комп'ютерної мережі є важливим інструментом для навчання та освіти спеціалістів у сфері мережевих технологій. За допомогою моделей можна візуалізувати і продемонструвати різні аспекти роботи мережі, пояснити принципи функціонування і розробити навчальні матеріали.

Всі ці аспекти підкреслюють актуальність моделювання роботи вузлів комп'ютерної мережі як інструмента для оптимізації, аналізу та розробки ефективних і надійних мережевих рішень.

В цій роботі будується імітаційна модель для аналізу продуктивності простого проміжного вузла комп'ютерної мережі на основі дискретно-подієвого підходу.

Імітаційне моделювання відкриває широкі можливості для дослідження систем та процесів, які не існують на даний момент, але можуть бути створені в майбутньому. Це стосується розробки нових технологій, продуктів і методів управління та оптимізації процесів, які ще не випущені на ринок. Завдяки імітаційному моделюванню можна провести аналіз, експерименти та прогнозування поведінки цих систем, що допомагає зменшити ризики та забезпечити більш впевнене впровадження нових рішень.

У першому розділі описані принципи побудови сучасних комп'ютерних мереж, методи оцінки продуктивності вузлів мережі та огляд сучасних методів дослідження їх параметрів.

У другому розділі розглянуто призначення та основні принципи, методи та моделі імітаційного моделювання.

Третій розділ присвячено побудові та використанню імітаційної моделі для оцінки продуктивності проміжного вузла мережі.

РОЗДІЛ 1

АНАЛІЗ ПРОБЛЕМИ ДОСЛІДЖЕННЯ ПРОДУКТИВНОСТІ ПРОМІЖНИХ ВУЗЛІВ КОМП'ЮТЕРНИХ МЕРЕЖ

1.1. Принципи побудови сучасних комп'ютерних мереж

Мережу можна визначити як взаємозв'язок між різними пристроями, які з'єднані за допомогою різних комунікаційних середовищ. Вони використовуються для обміну даними, а також ресурсами шляхом з'єднання двох або більше елементів на обмеженій території, яка відома як мережа.

Вузол мережі - це точка з'єднання, яка використовується для передачі, прийому, створення або зберігання інформації за допомогою розподілених мережевих маршрутів [1]. Кожен вузол мережі є точкою перерозподілу, яка використовується для розпізнавання процесів, передачі даних з однієї мережі в іншу. Поняття мережевих вузлів використовується для розподілу мережі, а також комутації пакетів.



Рис. 1.1 – Вузол мережі

1.1.1 Поняття ти види вузлів

Вузол - це фізичний елемент в комп'ютерній мережі, який має можливість передавати, отримувати та / або передавати інформацію між іншими пристроями. Пристрої, що беруть участь у функціонуванні комп'ютерної мережі, називаються вузлами мережі. Найпоширенішим вузлом прийнято вважати персональний комп'ютер – це називається комп'ютерним вузлом або інтернет вузлом.

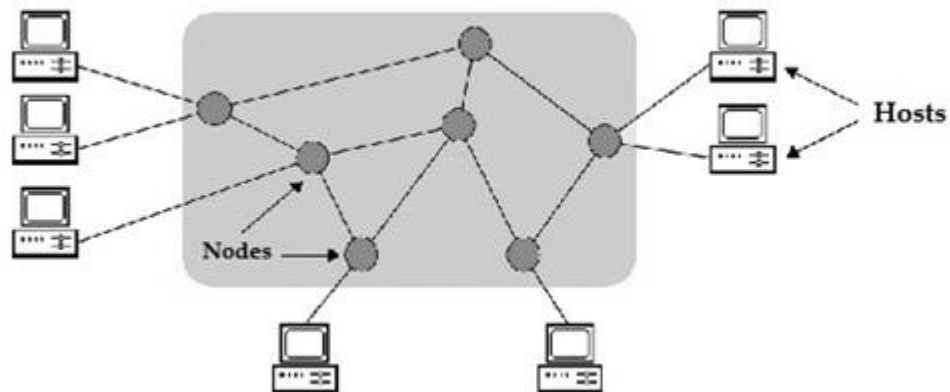


Рис. 1.2 – Вузли в комп'ютерній мережі [1]

Модеми, комутатори, концентратори, мости, сервери та принтери є вузлами комп'ютерної мережі, так само як і будь-які інші пристрої, які підключаються через Wi-Fi або Ethernet. Наприклад, якщо в мережі з'єднані три комп'ютери та один принтер, а також два бездротових пристрої, то загалом мережа складається з шести вузлів.

У комп'ютерній мережі кожен вузол повинен мати свій ідентифікатор, такий як IP-адреса або MAC-адреса, щоб його можна було розпізнати іншими пристроями в мережі. Якщо вузол не має цієї інформації або був вилучений в автономному режимі, то він вже не може функціонувати як вузол мережі [2].

Виділяють різні типи мережевих вузлів в залежності від типу мережі.

Інтернет-мережі. У мережах хост-комп'ютери - це фізичні мережеві вузли, які розпізнаються за допомогою IP-адреси (інтернет-протоколу). Деяке обладнання для передачі даних, таке як точки доступу WLAN, не включають адреси IP-хостів. Вони розглядаються як вузли локальної мережі або фізична мережа, а не хости чи інтернет-вузли.

Передача даних. Фізичні мережеві вузли в передачі даних в основному включають пристрої передачі даних або обладнання. Вони розташовані між DTE (кінцевим обладнанням даних), а також ланцюгами передачі даних. До таких пристроїв відносяться мости, комутатори, концентратори, інакше модеми. Основна функція цих пристроїв полягає у виконанні кодування та перетворення сигналу.

Мережеві вузли в межах передачі даних в основному включають DTE, такі як принтери, цифрові телефони, в іншому випадку розміщують комп'ютери, такі як сервери, маршрутизатори, інші робочі станції.

Вузли мережі залежно від підключення бувають проміжними, суміжними та кінцевими. Кінцевий вузол може розташовуватися тільки в кінці мережі, після нього не може продовжуватися гілка. Проміжний вузол розміщується на з'єднанні кількох гілок мережі. Суміжний вузол з'єднується хоча б одним шляхом, що не містить жодних інших вузлів.

Телекомунікаційні системи. У постійних телефонних мережах мережевими вузлами можуть бути приватні або громадські телефонні станції або послуги смарт-мережі на комп'ютері. Вузли стільникового зв'язку в основному складаються з контролерів базових станцій, і основною функцією цих контролерів є управління декількома базовими станціями. Але базові станції в стільникових мережах не розглядаються як вузли.

Локальні мережі та глобальні мережі. Вузол мережі в локальних мережах і глобальних мережах - це пристрій, який використовується для виконання точної функції. Для кожного вузла потрібна MAC-адреса, яка

використовується для кожного NIC (мережевої інтерфейсної карти). Прикладами цього в основному є комп'ютери, точки доступу бездротової локальної мережі та модеми, що використовують інтерфейси Ethernet тощо [1].

Система кабельного телебачення. В якості передаючого середовища використовується кабельна проводка, у цій мережі вузли – це будинки або підприємства, які підключаються до одного оптоволоконного приймача.

1.1.2 Топології мережі

Топологія мережі відображає характеристики мережі, які не залежать від її розміру, показує структуру, що складається з вузлів мережі та багатьох каналів, що їх об'єднує [6]. Проте ця характеристика не враховує продуктивність та принцип роботи вузлів, типів та довжин каналів. Можна виділити чотири базові топології мережі з точки зору фізичного розташування її функціональних компонентів та методів доступу до середовища передачі даних.

Топологія шини (рисунок 1.3). Це тип мережі, при якому кожен комп'ютер і мережевий пристрій підключається до одного кабелю, відомого як магістральний кабель. Ця топологія має багато-точковий зв'язок і двонаправлений характер. Однак, вона є ненадійною, оскільки, якщо кабель вийде з ладу, то топологія буде недієздатною.



Рис. 1.3 – Схема топології шини [6]

Шинна топологія мережі має кілька переваг, зокрема:

- вона є надійною і простою у використанні для невеликих мереж;
- для з'єднання комп'ютерів потрібно менше кабелю, що знижує витрати на побудову мережі порівняно з іншими схемами кабельних з'єднань;
- ця топологія може бути легко розширена;
- завдяки меншій довжині кабелів та більш високій надійності, мережа не втрачає працездатність у разі виходу з ладу одного комп'ютера.

Проте є й кілька недоліків:

- якщо основний кабель перерветься, то всі комп'ютери в мережі втратять зв'язок;
- велика кількість мережевого трафіку знижує продуктивність шинної мережі;
- інформація в мережі на фізичному рівні може бути не дуже добре захищеною, оскільки повідомлення від одного комп'ютера до іншого можуть бути перехоплені будь-яким комп'ютером в мережі.

Топологія зірки. Топологія зірки передбачає підключення кожного пристрою до одного центрального пристрою через кабель. В ролі центральної частини може виступати мультиплексор або концентратор, що дозволяє підключати більшу кількість пристроїв за допомогою меншої кількості каналів передачі даних. Цей центральний пристрій повністю управляє всіма пристроями, що до нього підключені. При цьому мультиплексор зберігає можливість відновлення всіх сигналів введення, що надійно захищає від втрати даних.

Мережа складається з одного центрального вузла, з якого випромінюються промені, що закінчуються периферійними пристроями (рисунок 1.4). У такій топології всі станції мають пряме з'єднання з центральним комп'ютером, який контролює потік повідомлень у мережі, тому що передача повідомлень від однієї станції до іншої можлива лише через

центральний комп'ютер.

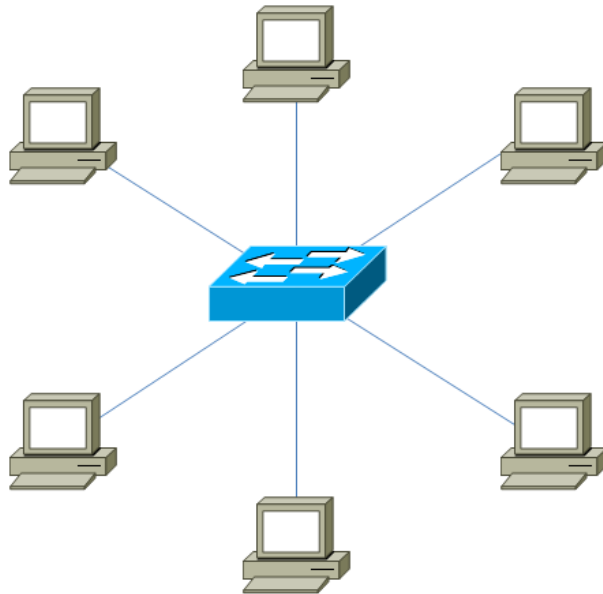


Рис. 1.4 – Схема мережі з топологією зірка [7]

Зіркову топологію можна розширити, додавши ще один концентратор замість одного з комп'ютерів, та підключивши до нього додаткові машини. Це дозволяє створити гібридну зіркову мережу (рисунок 1.5).

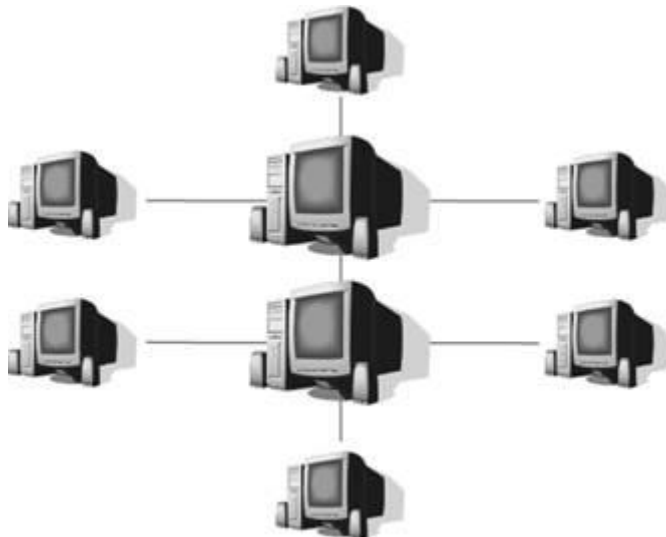


Рис. 1.5 – Схема гібридної зіркоподібною мережі [6]

Переваги мережі зіркової топології полягають в наступному:

- зручність модифікації та додавання комп'ютерів до мережі без порушення роботи інших її частин;
- легкість діагностики завдяки наявності центрального комп'ютера;
- відсутність повної зупинки мережі при відмові одного з комп'ютерів;
- можливість використання різних типів кабелів в одній мережі.

Недоліки мережі зіркової топології полягають в наступному:

- при відмові центрального комп'ютера всі комп'ютери в мережі стають недоступними;
- велика протяжність кабелів (залежно від розташування центрального комп'ютера) збільшує вартість мережі порівняно з іншими топологіями;
- продуктивність мережі залежить від роботи єдиного центрального пристрою.

Топологія кільце. Мережа складається з кільця, в якому кожен вузол зв'язаний з двома іншими вузлами. У цій мережі кожна станція виконує функцію центрального комп'ютера і безпосередньо зв'язана з двома сусідніми станціями (рисунк 1.6).

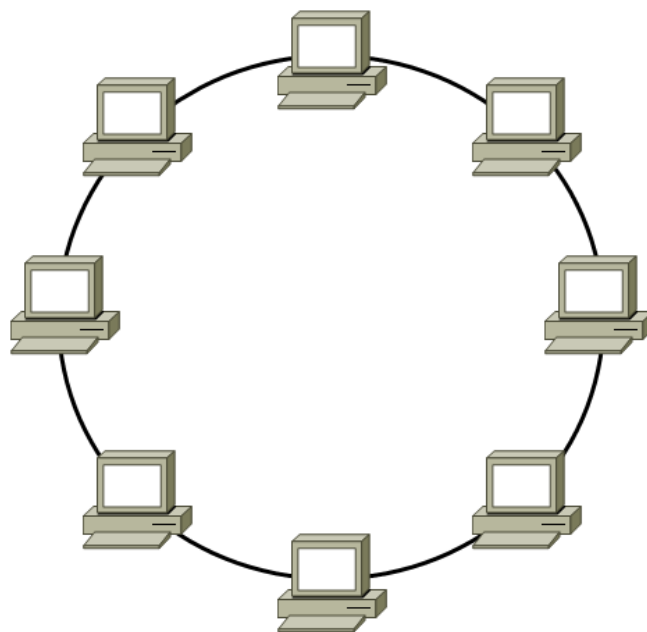


Рис. 1.6 – Схема мережі з топологією кільце [7]

У цій топології дані течуть в одному напрямку, тобто ця топологія односпрямована. Проте її можна зробити двонаправленою, додавши другий рівень зв'язку між кожним вузлом мережі – це називається топологією подвійного кільця. Усі дані у кільцевій топології течуть за годинниковою стрілкою, також він не має кільцевих кінців, тобто кожен вузол з'єднаний з іншим вузлом і не має точки припинення.

Найпоширенішим методом доступу до топології кільця є проходження токенів. Проходження токена – це метод доступу до мережі, при якому токен передається з одного вузла на інший вузол. Маркер – він являє собою рамку, яка циркулює по мережі.

Робота з проходження токенів:

- токен переміщається по мережі, і він передається з комп'ютера на комп'ютер, поки не досягне пункту призначення;
- відправник змінює маркер, вставляючи адресу разом із даними;
- дані передаються з одного пристрою на інший, поки адреса призначення не збігається. Після того, як приймач отримує токен, він надсилає підтвердження відправнику;
- у кільцевій топології в якості носія використовується токен.

Переваги мережі з кільцевою топологією:

- так як всі комп'ютери отримують рівний доступ до токена, немає можливості, що хтось із них монополізує мережу;
- система має більш високу надійність при обриві кабелів, оскільки до кожного комп'ютера є два шляхи доступу.

Недоліки мережі з кільцевою топологією:

- велика протяжність кабелю;
- вихід з ладу одного вузла може привести до збою всієї мережі;
- слабка захищеність інформації;
- складне усунення несправностей;

– при збільшенні кількості користувачів і навантаження на мережу, продуктивність її роботи поступово зменшується через спільне використання.

Сітчаста топологія (рисунок 1.7). У сітчастій топології кожен пристрій підключається до іншого пристрою через певний канал. Топологія в основному використовується для реалізацій WAN, де збої зв'язку є критичною проблемою. Також сітчаста топологія в основному використовується для бездротових мереж.

Сітчаста топологія ділиться на дві категорії:

- Повністю пов'язана сітчаста топологія;
- Частково пов'язана сітчаста топологія.

У топології повної сітки кожен комп'ютер підключений до всіх комп'ютерів, доступних в мережі.

У частковій сітчастій топології не всі, але певні комп'ютери підключені до тих комп'ютерів, з якими вони часто взаємодіють.

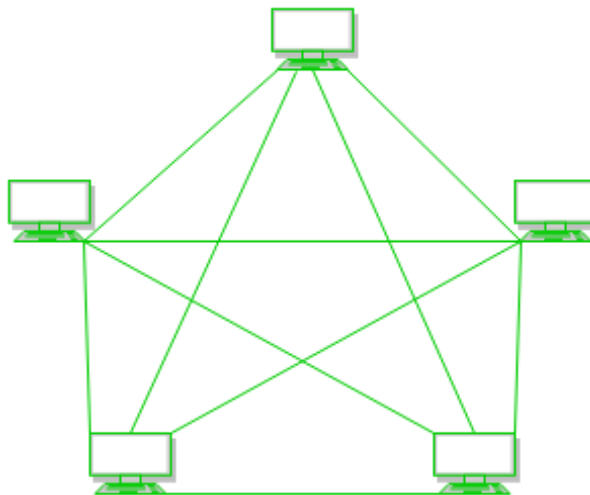


Рис. 1.7 – Сітчаста топологія [9]

Переваги цієї топології:

- швидкий та надійний зв'язок між вузлами;
- легке виявлення несправностей. Дані передаються через виділені канали або послання, що забезпечує їх надійність;

- забезпечення безпеки та конфіденційності.

Недоліки цієї топології:

- складна установка та настройка;
- висока вартість обслуговування;
- підходить для невеликої кількості пристроїв (адже для великої кількості пристроїв потрібна об'ємна проводка, а вартість кабелів висока).

Змішана топологія (рисунок 1.8). Найпопулярніша топологія, яка об'єднала всі топології вище у собі. Є деревоподібною структурою, яка поєднує всі топології. Одна з найбільш стійких до відмов топологія, оскільки якщо у двох майданчиків відбудеться обрив, то паралізований буде зв'язок тільки між ними, а всі інші об'єднані майданчики працюватимуть безвідмовно. На сьогоднішній день, ця топологія використовується у всіх середніх та великих компаніях.

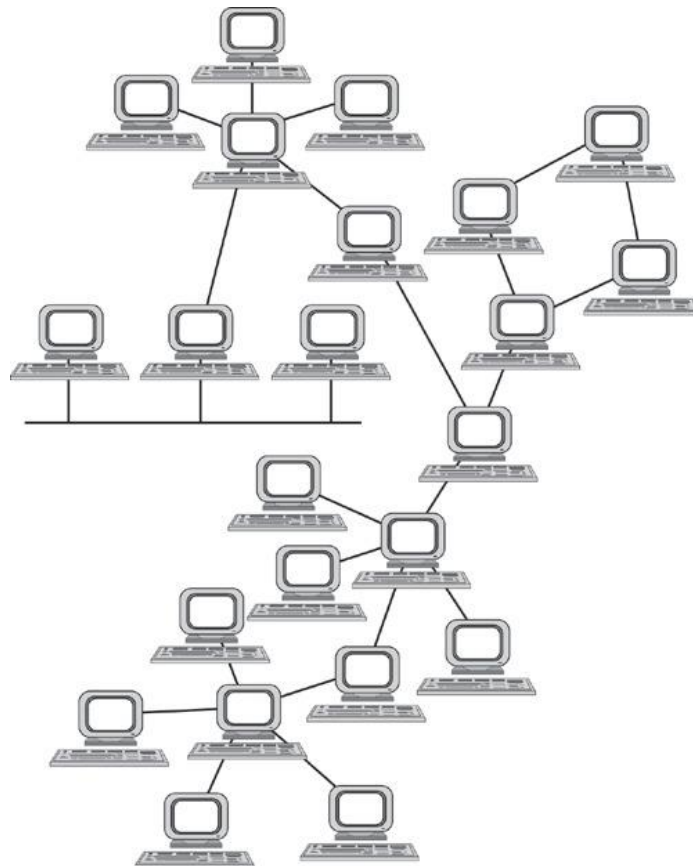


Рис. 1.8 – Змішана топологія [7]

1.2 Методи оцінки продуктивності вузлів мережі

Характеристики функціонування комп'ютерної мережі охоплюють її властивості, які визначаються під час аналізу та є функцією параметрів. Параметри мережі описують основні властивості мережі та є вихідними даними під час її аналізу. Характеристики комп'ютерних мереж - це набір показників ефективності (якості) мережі.

До основних характеристик комп'ютерних мереж відносяться наступні [11].

Безпека. Це одна з найважливіших характеристик комп'ютерної мережі. Більшість підприємств сьогодні покладаються на комп'ютери, доступ до яких здійснюється через мережу. Як наслідок, якщо технологія комп'ютерної мережі не є надійною та безпечною, несанкціонований доступ до критичних даних компанії може бути можливим. Однак у наш час засоби комп'ютерних мереж в першу чергу забезпечують найвищий рівень безпеки і запобігають будь-якому несанкціонованому доступу.

Надійність. Комп'ютерні мережі є дуже надійними інструментами, і користувачі можуть легко з'єднати між собою свої пристрої з їх допомогою. Комп'ютерні мережі мають альтернативні джерела постачання для забезпечення високої надійності. Їхня робота в мережі залишається незмінною, навіть якщо користувачам потрібно друкувати, перевіряти повідомлення, відвідувати нараду або отримувати доступ до даних з іншого комп'ютера.

Масштабованість. Масштабованість відноситься до здатності масштабуватися у відповідь на мінливі потреби при збереженні високої продуктивності. Інтернет є найкращим прикладом масштабованості; Багато нових користувачів підключаються через Інтернет і спілкуються з іншими пристроями, але наша мережа все ще працює належним чином.

Потік даних. Користувачі можуть отримувати доступ і передавати такі

дані, як файли, документи та інші типи інформації, за допомогою комп'ютерних мереж. Це важлива особливість комп'ютерної мережі, оскільки дозволяє передавати дані з одного пристрою на інший.

Продуктивність. Продуктивність оцінюється за допомогою часу, якщо для передачі даних потрібно менше часу, а відповідь швидка, для користувачів є значною перевагою передавати дані та використовувати кілька ресурсів. Продуктивність можна збільшити за допомогою декількох процесорів.

Відмовостійкість. Це також є функцією, яку пропонують комп'ютерні мережі. Припустимо, два пристрої підключені як з дротовими, так і з бездротовими носіями. Якщо пристрій відправника надсилає інформацію, а пристрій одержувача має блокування на своєму бездротовому носії. Він знайде інший найкращий альтернативний носій для надсилання інформації на пристрій приймача, який у цьому випадку є провідним носієм. Використовуючи це, вони можуть продовжувати працювати, навіть якщо мережа вийшла з ладу або пошкоджена. Так працює відмовостійкість.

Якість обслуговування (QoS). Це означає, що користувачі можуть визначати пріоритети та налаштовувати передачу даних, а також мінімізувати будь-які затримки передачі. Це також дозволяє втратити будь-які дані, якщо це відбувається регулярно. В результаті ще однією особливістю комп'ютерної мережі є те, що вона забезпечує високий рівень сервісу для своїх користувачів.

Сумісність з апаратними та програмними компонентами. Ще однією чудовою особливістю комп'ютерних мереж є надання доступу кільком підключеним пристроям до одного програмного забезпечення. Це тягне за собою те, що одне і те ж програмне забезпечення може використовуватися на різних пристроях. Він покращує сумісність, дозволяючи завершити цю роботу. В результаті комп'ютерні мережі полегшують використання програмного забезпечення і найкращим чином використовують фізичні компоненти.

Продуктивність комп'ютерної мережі залежить від продуктивності окремих її компонентів, таких як швидкість передачі даних каналами зв'язку,

що вимірюється обсягом даних, що передаються за одиницю часу, та швидкодія процесора, що вимірюється числом команд, що виконуються в одиницю часу. Для оцінки продуктивності комп'ютерної мережі використовуються показники, що включають у себе ці параметри [11]:

- продуктивність мережі передачі, вимірювана кількістю повідомлень, пакетів, кадрів або бітів, які передаються за одиницю часу;
- продуктивність засобів обробки даних, яка є сумарною продуктивністю всіх компонентів комп'ютерної мережі, таких як ЕОМ та системи.

Продуктивність СТК (комунікаційна потужність) може бути задана такими показниками:

- максимальна або гранична продуктивність, звана пропускною здатністю мережі передачі даних та вимірювана кількістю пакетів (кадрів), що передаються в мережі за одиницю часу;
- реальна або фактична продуктивність мережі передачі даних, яка може бути визначена середнім значенням за певний інтервал часу або миттєвим значенням в конкретний момент часу.

Для оцінки продуктивності мережі в цілому використовуються наступні показники:

- час реакції, що визначається як час між виникненням запиту до якогось мережевого сервісу / пристрою та отриманням відповіді на нього;
- пропускна здатність, яка відображає обсяг даних, переданих мережею в одиницю часу, та затримка передачі, яка дорівнює інтервалу між моментом надходження пакета на вхід будь-якого мережного пристрою та моментом його появи на виході цього пристрою.

Для оцінки продуктивності існує ціла низка мережевих систем управління, призначених для отримання статистичної інформації про будь-який з вузлів мережі.

Найлегшим способом аналізу є поділ системи на рівні, використовуючи

модель OSI, так-як на будь-якому рівні використовуються різні параметри визначення продуктивності системи.

У спрощеній моделі OSI представляють зазвичай 3 рівні:

- Фізичний рівень
- Мережний рівень
- Канальний рівень

Першим кроком при аналізі продуктивності мережі є вимірювання параметрів роботи мережі, таких як щоденні навантаження (безперервні та пікові), зміна навантажень за часом, різні типи помилок низького рівня.

Другим кроком – інтерпретація зібраної інформації.

Треба визначити та запровадити показники продуктивності. До них відносяться наступні.

1. Коефіцієнт використання мережі. Визначається як фактична кількість біт, переданих по мережі в будь-який момент часу, поділена на максимально можливе число біт, яке можна передати за той самий проміжок часу. Таким чином, 100% коефіцієнт використання мережі визначається як передача максимально можливої кількості біт за секунду без порушення технічних умов мережі. Система аналізу продуктивності локальної мережі фірми Hewlett Packard надає засоби для збору інформації про коефіцієнт використання мережі. Інформація може бути представлена як у графічному вигляді, так і роздрукована та записана на диск для періодичного порівняння.

Коефіцієнт використання або навантаження мережі може різко змінюватись в залежності від активності її вузлів. Навантаження мережі може залежати від звичок користувача. Завжди треба знати найвищий миттєвий пік у межах хвилини протягом дня.

2. Вимірювання помилок на нижніх рівнях. Кількість помилок, що виникають при роботі мережі, також визначає її продуктивність. Це неправильні контрольні послідовності кадрів у мережах IEEE 802 на рівнях 1-

2, неузгодженість кадрів, передача даних, що затягнулася, укорочені кадри, конфлікти.

В результаті буде надане зведення роботи мережі, тобто таблиця, що містить інформацію про коефіцієнт використання мережі та її пропускну здатність, про помилки та конфлікти, розмір кадрів і так далі. Для збірки цієї інформації є можливість вибору із трьох вимірювань:

- час вибірки
- тривалість виміру
- межі відображення

1.3 Огляд сучасних методів дослідження параметрів вузлів комп'ютерних мереж.

Сучасні засоби аналізу продуктивності комп'ютерних мереж або їх окремих елементів базуються на декількох типах моделей, серед яких потрібно розрізняти детерміновані та стохастичні, дискретні та безперервні. Далі розглянути відмінності цих підходів [12].

– Детермінована проти стохастичної: детермінована модель передбачає певний вихід із заданого набору вхідні дані, що не містять ані випадковості, ані ймовірнісних компонентів. Даний вхід завжди буде створювати той самий вихід за однакових початкових умов. На відміну від цього, стохастична модель має деякі вхідні дані з випадковістю, отже, модель передбачає набір можливих результатів, зважених за їх ймовірності або ймовірності [18].

– Стійкий стан проти динамічного: Стаціонарна модель намагається встановити результати відповідно до заданого набору вхідних даних, коли система досягла стаціонарної рівноваги. Навпаки, динамічна модель

забезпечує реакції системи на змінні входи. Стаціонарні підходи часто використовуються для створення спрощеної попередньої моделі [14].

– Дискретні та безперервні значення: дискретна модель представлена скінченним набором даних, отже, змінні стану приймають свої значення з рахункового набору значень. Навпаки, безперервна модель відповідає нескінченному набору даних. Таким чином, змінні стану можуть приймати будь-які значення в межах діапазону двох значень [15]. Однак є деякі системи, які повинні бути змодельовані, що показує аспекти обох підходів, які призводять до комбінованого дискретно-неперервного моделі [16].

– Дискретний проти безперервного часу: у дискретній моделі зміни стану можуть відбуватися лише в певний час. миттєво в часі. Ці моменти відповідають значущим подіям, які впливають на вихід або внутрішні дані стан системи. Навпаки, у безперервній моделі змінні стану змінюються безперервно шляхом, а не різко з одного стану в інший [17]. Тому безперервні моделі охоплюють нескінченну кількість станів.

Найчастіше використовуються дискретні моделі мережі M&S. Моделювання можна проводити за двома підходами: локальним і розподіленим. Розподілене моделювання є так, що кілька систем взаємопов'язані, щоб працювати разом, взаємодіяти одна з одною, проводити моделювання. Натомість локальне моделювання виконується на одному комп'ютері. Історично останнє, підхід був найбільш широко використовуваним для моделювання комп'ютерних мереж, але все більшої складності моделювання сприяє важливості першого підходу [14].

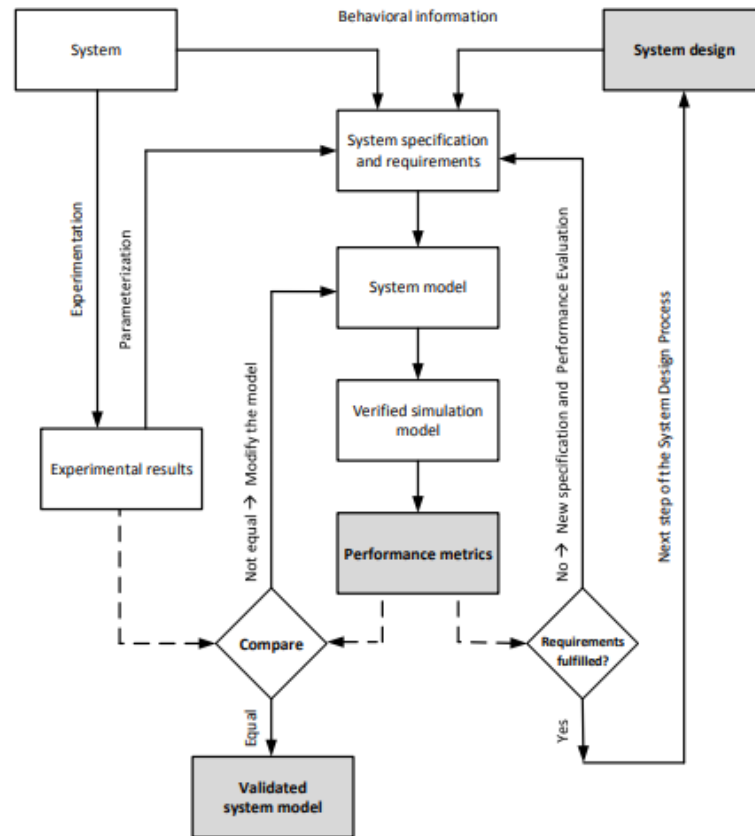


Рис. 1.9 – Схема моделювання та імітації системи. [12]

На рисунку 1.9 показаний процес M&S (моделювання та імітації). Поведінкова інформація, отримана з реальної системи, використовується разом із системними нормами та вимогами. На основі цих входів створюється системна модель, яка потім перевіряється симуляцією. Як правило, у процесі моделювання визначається кілька показників продуктивності, які можна порівняти з експериментальними результатами реальної системи. Якщо вони подібні, модель вважається ефективною; якщо вони не схожі, потрібно виправити системну модель. Аналогічним чином, показники продуктивності використовуються для визначення того, чи відповідає системна модель потребам, тому проектувана система може бути уточнена та розширена контрольованим чином.

Математичне моделювання з використанням аналітичного методу передбачає використання апарату теорії ймовірностей і теорії масового

обслуговування. Більшість моделей реальних систем будуються на базі моделей масового обслуговування, таких як базові СМО та мережеві моделі СеМО [13].

При розробці аналітичної моделі мережі необхідно визначити склад та перелік параметрів та характеристик моделі в термінах теорії масового обслуговування, і встановити їх зв'язок з параметрами та характеристиками системи, що досліджується. Це означає, що необхідно здійснити параметризацію моделі. Для аналізу характеристик протоколів другого, третього та четвертого рівнів застосовуються моделі теорії мереж масового обслуговування. На каналному рівні ці моделі використовуються визначення ефективної швидкості передачі. При аналізі наскрізних протоколів моделі теорії мереж масового обслуговування дозволяють знаходити затримку повідомлень джерело-адресат, визначати параметри управління потоками і т.д. Мережеві моделі окремих частин комп'ютерної мережі адекватно відображають процес обробки повідомлень у цих пристроях та дозволяють розраховувати характеристики та здійснювати вибір різних параметрів, таких як обсяг буферної пам'яті вузлів комутації. При обчисленні передачі даних в цілому, потрібно також враховувати складну структуру окремих вузлів..

Необхідність вирішення завдань вибору топології та пропускних здібностей каналів зв'язку, відшукування оптимальних маршрутів і т. д. вимагає застосування спрощених моделей мереж масового обслуговування, що дозволяють знаходити явний вид цільової функції, в якості якої використовується час затримки повідомлень. Цей метод потребує подальшої деталізації характеристик мережі за допомогою більш реалістичних моделей, які враховують особливості мережевих протоколів. На характеристики мережі пакетної комутації значно впливають способи доставки пакетів і методи маршрутизації. Моделювання методів маршрутизації та різних способів доставки пакетів здійснюється шляхом вибору відповідного типу мережі масового обслуговування та матриці маршрутів $P=\{p_{ij}\}$, структура якої

враховує топологію мережі.

Комп'ютерні мережі найчастіше моделюються з допомогою дискретно-подієвого моделювання. Основна причина такого широкого використання полягає в тому, що дискретно-подієве моделювання краще адаптується для представлення поведінки комп'ютерних мереж, оскільки протоколи комп'ютерних мереж можуть бути змодельовані як кінцеві стани. У комп'ютерній мережі існує стійкий стан між двома послідовними подіями, а дискретно-подієве моделювання дозволяє переходити від одного стійкого до іншого, що призводить до більш швидкого моделювання. Іншими цікавими аспектами дискретно-подійного моделювання є гнучкість та менші обчислювальні витрати.

РОЗДІЛ 2

ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЯК МЕТОДОЛОГІЯ ДОСЛІДЖЕННЯ ХАРАКТЕРИСТИК КОМП'ЮТЕРНИХ МЕРЕЖ

2.1. Призначення та основні принципи імітаційного моделювання

Імітаційне моделювання – це метод дослідження різних систем та процесів, який ґрунтується на створенні віртуальної моделі цих систем з використанням комп'ютерних інструментів. Основна мета імітаційного моделювання – вивчення поведінки системи в різних умовах, що максимально наближені до реальних умов її функціонування [19].

Імітаційне моделювання також дозволяє проводити дослідження систем та процесів, які не існують на даний момент, але можуть бути створені в майбутньому. Зокрема, це може стосуватися розробки нових технологій та продуктів, які ще не були випущені на ринок, або нових методів управління та оптимізації процесів.

Крім того, імітаційне моделювання може використовуватись для розробки та тестування нових алгоритмів та стратегій управління системами, що дозволяє зменшити ризики при їх практичному впровадженні. Наприклад, в області фінансів та інвестицій імітаційне моделювання дозволяє вивчати поведінку ринків та прогнозувати можливі ризики та прибутки при різних умовах функціонування [20].

Також широко використовується у наукових дослідженнях, зокрема у фізиці, хімії, біології та інших галузях науки. Наприклад, в області фізики імітаційне моделювання дозволяє вивчати поведінку складних систем, таких як атоми, молекули та кристали, що дозволяє зрозуміти їх властивості та взаємодії.

Імітаційне моделювання є корисним інструментом для вивчення та

оптимізації різних систем, включаючи комп'ютерні мережі. Цей метод дозволяє досліджувати поведінку системи та її характеристики в різних умовах, зокрема в екстремальних ситуаціях, які в реальному житті можуть бути дуже складними або навіть небезпечними.

Загальною метою імітаційного моделювання є підвищення ефективності та надійності систем, а також зниження витрат на їх впровадження та експлуатацію. Дослідження з використанням імітаційного моделювання дозволяють виявити можливі проблеми та ризики, які можуть виникнути у реальних умовах функціонування систем.

Основні принципи такого моделювання включають побудову віртуальної моделі системи, що відображає всі її основні характеристики та процеси, та проведення експериментів на основі цієї моделі з метою вивчення поведінки системи та її характеристик в різних умовах функціонування.

Імітаційне моделювання має кілька переваг, зокрема [19]:

- можливість вивчення поведінки системи в різних умовах, що з реальних умов є складні або неможливі до відтворення;
- економічність та безпека проведення досліджень, що дозволяє уникнути фінансових та матеріальних втрат у разі провалу експерименту;
- можливість моделювання системи з використанням різних параметрів та умов, що дозволяє підібрати оптимальні налаштування для максимальної ефективності системи.

Крім того, імітаційне моделювання може бути використано для прогнозування та планування роботи системи в майбутньому. Це дає змогу прогнозувати можливі проблеми та вирішувати їх до того, як вони виникнуть у реальній системі. Також, імітаційне моделювання може бути використане для вивчення впливу різних факторів на систему та визначення найбільш ефективних стратегій її управління.

На наступній схемі представлений загальний процес імітаційного моделювання.

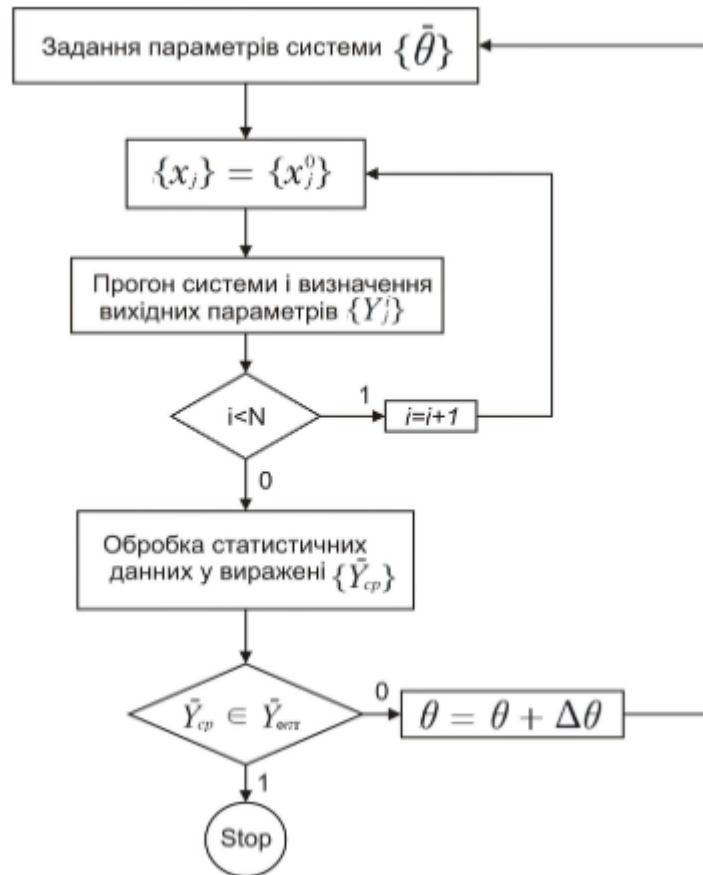


Рис. 2.1 – Блок-схема імітаційного моделювання [19]

На рисунку прийняті такі позначення: X_j – вхідні змінні системи, Y_j – вихідні змінні, θ – параметри системи, i – номер прогону, N – кількість прогонів.

Взагалі, імітаційне моделювання є потужним інструментом для дослідження та вивчення різних систем та процесів, який дозволяє збільшити ефективність та економічність проведення досліджень, а також зменшити ризики та витрати при впровадженні нових технологій та методів управління.

Використання різних видів імітаційних моделей залежить від конкретної задачі, яку необхідно вирішити, та властивостей системи, яка досліджується.

2.2. Види імітаційних моделей

Основні види імітаційних моделей, які можна використовувати для вивчення характеристик комп'ютерних мереж, включають такі:

1. Дискретно-подієві моделі (Discrete Event Models) – це тип імітаційних моделей, які моделюють дискретні події у системі. Ці моделі використовуються в багатьох галузях, включаючи комп'ютерні мережі, транспортні системи, виробничі процеси та інші. Вони дозволяють досліджувати різні аспекти системи, такі як ефективність, пропускна здатність, час очікування та інші.

Цим терміном ми позначимо підхід, в основі якого лежить концепція заявок (транзактів, entities), ресурсів та поточкових діаграм (flowcharts), що визначають потоки заявок та використання ресурсів. Цей підхід сягає своїм корінням до Джефрі Гордона, який у 1960-х роках придумав та розвинув систему імітаційного моделювання GPSS та реалізував її, працюючи в ІВМ. Заявки (транзакти в GPSS) – це пасивні об'єкти, що представляють людей, деталі, документи, завдання, повідомлення тощо. Вони подорожують через flowchart, стоячи в чергах, обробляючись, захоплюючи та звільняючи ресурси, розділяючись, з'єднуючись тощо. Дискретно-подієву модель можна розглядати як глобальну схему обробки заявок, події відбуваються в дискретні моменти часу, і кожна подія може вплинути на подальшу поведінку системи. Для моделювання цих подій використовуються стохастичні процеси, такі як випадкові змінні [20].

Дискретно-подієві моделі використовуються для дослідження поведінки системи в часовому проміжку. Вони дозволяють моделювати багато подій, що відбуваються в системі одночасно, і визначати їх вплив на стан системи. Ці моделі дають гарний спосіб для дослідження різних проблем, пов'язаних з комп'ютерними мережами, таких як загрузка мережі, пропускна здатність, задержки та інші характеристики.

У дискретно-подієвих моделях час подій зазвичай є дискретним і відрізняється від часу моделювання. Кожна подія може змінити стан системи, і зазвичай залежить від попередніх подій в системі. Для моделювання дискретно-подієвих систем часто використовують такі інструменти, як мережеві симулятори та мови програмування, що дозволяють створювати дискретні події і зберігати стан системи.

Щоб створити ефективну дискретно-подієву модель комп'ютерної мережі, необхідно уважно відбирати відповідні події, які впливають на систему, та визначати їх частоту виникнення. Також важливо враховувати взаємодію між подіями та можливість збоїв в системі.

Ці моделі можуть бути досить складними, оскільки вони містять велику кількість подій, що відбуваються випадковим чином. Також вони можуть потребувати значної кількості обчислювальних ресурсів для їхньої реалізації. Однак, завдяки використанню комп'ютерних технологій, дискретно-подійні моделі можуть бути створені та використані для дослідження різноманітних характеристик комп'ютерних мереж.

2. Моделі на основі агентів (Agent-based Models) – це моделі, в яких агенти взаємодіють між собою та з оточенням. Кожен агент у моделі має свої власні характеристики, поведінку та правила взаємодії з іншими агентами та середовищем. Моделі на основі агентів можуть бути використані для дослідження різних аспектів комп'ютерних мереж, таких як протоколи маршрутизації, взаємодія мережевих пристроїв та споживачів послуг мережі [20].

У моделях на основі агентів агенти можуть мати різні рівні складності, від простих агентів зі статусом on/off до складних агентів, які можуть мати власні бази даних та логіку прийняття рішень. Моделювання на основі агентів дає змогу аналізувати ефективність різних стратегій взаємодії між агентами та їх вплив на характеристики мережі в цілому.

Однією з переваг моделей на основі агентів є те, що вони можуть

відображати нелінійні та складні взаємодії між агентами та середовищем, що є важливим для дослідження різних сценаріїв та стратегій в мережах. Також моделі на основі агентів можуть використовуватись для прогнозування майбутнього розвитку мереж та оцінки впливу змін на різні аспекти мережі.

3. Системно-динамічні моделі (System Dynamics Models) – є ще одним видом імітаційних моделей, які використовуються для дослідження характеристик комп'ютерних мереж. Вони дозволяють моделювати поведінку системи з часовою динамікою і зв'язками між її компонентами.

Основним елементом системно-динамічної моделі є «балансові рівняння», які відображають динаміку системи з часом. Ці рівняння описують взаємодію між різними компонентами системи і зміни їх стану в часі.

Системно-динамічні моделі зазвичай розробляються з використанням спеціальних інструментів, таких як Vensim, AnyLogic, Simulink и інших. Вони можуть бути використані для дослідження різних характеристик мережі, таких як пропускна здатність, задержки передачі даних, стійкість до збоїв то

Системно-динамічні моделі є особливо корисними при дослідженні довготривалих динамічних процесів, таких як зміни в трафіку мережі або вплив нових протоколів на її характеристики. Вони також можуть бути використані для моделювання взаємодії між різними компонентами мережі, такими як маршрутизатори, комутатори, сервери тощо.

В цілому, системно-динамічні моделі є потужним інструментом для дослідження характеристик комп'ютерних мереж, оскільки вони дозволяють зрозуміти, як система працює в часі і як зміни в одному компоненті можуть вплинути на інші частини системи.

4. Моделі на основі графів (Graph-based Models) – це імітаційні моделі, які використовують графи для зображення взаємодії між компонентами системи. У таких моделях компоненти системи відображаються в вигляді вузлів, а взаємодії між ними - у вигляді ребер графу.

Одним з основних застосувань графових моделей є аналіз топології

мережі. Такі моделі дозволяють візуалізувати мережу та з'ясувати, які компоненти мережі взаємодіють між собою та які є найбільш важливими для її функціонування.

Застосування графових моделей включає також використання алгоритмів теорії графів для вирішення різних задач, таких як задачі маршрутизації.

Важливою перевагою графових моделей є їх можливість легко масштабуватись для моделювання великих та складних систем, що робить їх популярним інструментом для дослідження комп'ютерних мереж та інших складних систем.

Загалом, графові моделі є досить корисним інструментом для аналізу та моделювання характеристик комп'ютерних мереж. Графові моделі можуть бути використані для візуалізації структури мережі, аналізу прохідності та шляхів, виявлення аномальних поведінок та пошуку найкоротших шляхів між вузлами мережі. Графові моделі можуть бути побудовані для різних рівнів абстракції мережі, від фізичного до логічного рівня, тому вони можуть бути корисними для різних типів досліджень, включаючи аналіз безпеки мережі, планування розширення мережі та визначення оптимальних конфігурацій мережі.

5. Моделі на основі мереж Петрі (Petri Net Models) – це інша форма імітаційних моделей, що використовує математичні об'єкти, відомі як мережі Петрі. Мережі Петрі використовуються для опису імовірнісних процесів, що відбуваються в системах, і складаються з двох основних компонентів: позицій і переходів.

Позиції є станами системи, в якій можуть перебувати об'єкти, а переходи – це події, які переводять систему з одного стану в інший. Ці компоненти зв'язуються між собою, утворюючи мережу, яка відображає залежності між різними станами системи.

Моделі на основі мереж Петрі зазвичай використовуються для

моделювання розподілених систем, таких як комп'ютерні мережі, де важливо визначати, які процеси можуть відбуватися паралельно та які обмеження на ресурси можуть впливати на продуктивність системи. Моделювання на основі мереж Петрі може допомогти інженерам виявити проблемні зони в мережі та розробити ефективні стратегії оптимізації роботи системи.

2.3. Системи масового обслуговування

У багатьох випадках дослідження за допомогою дискретно-подієвого моделювання пов'язано з моделюванням реальної системи масового обслуговування (СМО) або, якого-небудь компонента обслуговування, який сам є системою масового обслуговування.

Функціонування будь-якої СМО полягає в обслуговуванні замовлень, які надходять до неї у деякі випадкові моменти часу. Замовлення обробляється протягом випадкового терміну часу і залишають систему.

Тому у СМО передбачена наявність таких блоків [19]:

1. Генератор замовлень, який генерує прихід замовлення в систему у випадкові моменти часу, які підпорядковані заданому закону розподілу ймовірності.

2. Пристрої обслуговування, які обслуговують замовлення протягом випадкового терміну часу.

3. Черги до пристроїв обслуговування з різними дисциплінами та пріоритетами.

4. Знищувач, який імітує вихід замовлення з системи.

Найчастіше ефективність функціонування будь-якої СМО визначається такими показниками:

1. Середня кількість обслугованих замовлень за одиницю часу.

2. Середній відсоток замовлень, які не були обслуговувані.

3. Ймовірність того, що замовлення буде обслуговуване.
4. Середній час очікування у черзі.
5. Закон розподілу часу очікування.
6. Середня кількість замовлень у черзі.
7. Закон розподілу замовлень у черзі.
8. Коефіцієнт завантаження пристроїв обслуговування.
9. Середня кількість завантажених пристроїв.

Вхідні характеристики СМО:

1. Вхідний потік замовлень.
2. Дисципліни постановки замовлень у чергу та вибору замовлень з неї.
3. Правила обслуговування (структурна схема).
4. Вихідний потік замовлень.
5. Режим роботи.

Система масового обслуговування описується як потік заявок, які надходять в систему випадковим чином. Для простіших потоків заявок частота надходження заявок у систему підпорядковується закону Пуассона. Ймовірність надходження точно k заявок за час t обчислюється за формулою:

$$P_k(t) = \frac{(\lambda t)^k}{k!} e^{-\lambda t}, \quad (2.1)$$

В формулі символом λ позначена кількість запитів, що надходять до системи за одиницю часу (інтенсивність потоку). Розподіл Пуассона описує виникнення рідкісних подій з незмінною частотою. Максимум цього розподілу припадає на область навколо λ , і з віддаленням в обидві сторони він швидко знижується.

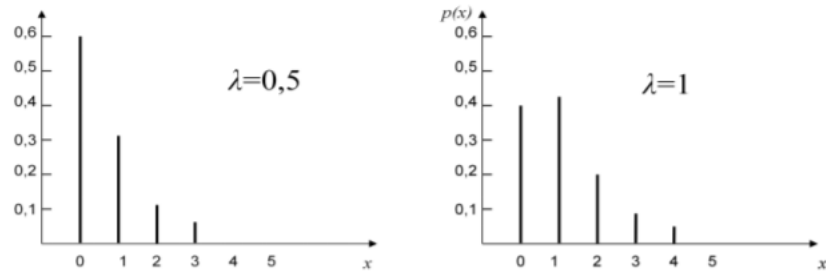


Рис.2.2 – Розподіл Пуасона

Простий потік має три основні властивості: одиночність, стаціонарність та відсутність наслідків. Одиночність потоку означає практичну неможливість одночасного надходження двох або більше заявок. Ймовірність надходження до системи певної кількості заявок протягом заданого проміжку часу Δt залежить від його величини і не залежить від початку його відліку на вісі часу.

Експоненційний розподіл часто використовується для представлення інтервалів часу між випадковими подіями, наприклад, часу між надходженнями заявок в моделі СМО або часу між відмовами в моделях надійності.

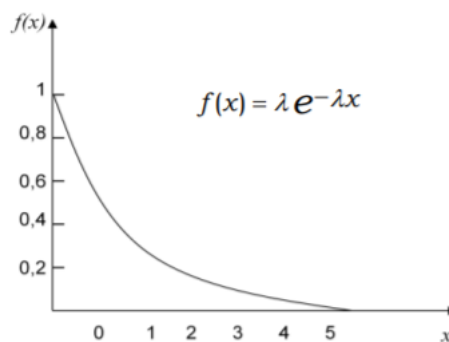


Рис.2.3 – Експоненційний розподіл

Процес обслуговування спрямований «навстріч» потоку заявок і також може характеризуватися інтенсивністю (часто позначається як μ) або часом обслуговування. Час обслуговування також є випадковою величиною. В простих моделях (наприклад, при відсутності інформації про роботу каналу

обслуговування) для її моделювання застосовують трикутний розподіл.

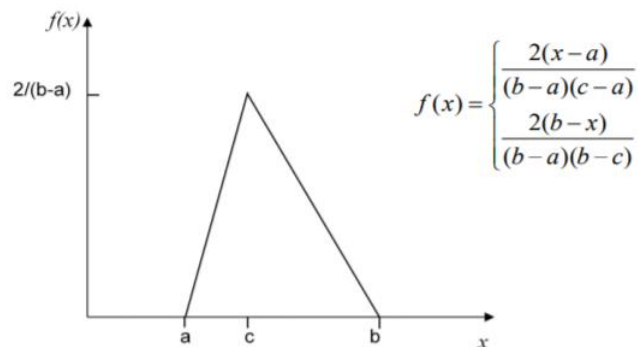


Рис. 2.4 – Трикутний розподіл

Формування черг залежить від виду дисципліни обслуговування. В межах черги заявки можуть обслуговуватися або в порядку FIFO (першим прийшов - першим обслужений), або LIFO (останнім прийшов - першим обслужений). Дисципліною обслуговування також може бути SJF (найкоротше за часом виконання завдання обробляється першим) та обслуговування за пріоритетом. Пріоритет обслуговування має три види:

- відносний пріоритет (заявка високого пріоритету очікує завершення обслуговування заявки з меншим пріоритетом);
- абсолютний пріоритет (заявка високого пріоритету при надходженні негайно витісняє заявку з меншим пріоритетом);
- змішаний пріоритет (якщо заявка з найнижчим пріоритетом обслуговувалась протягом часу, меншого за критичний, то використовується відносний пріоритет, в іншому випадку виконується абсолютний пріоритет).

Значення (призначення) пріоритету заявки може бути статичним або обчислюватися та змінюватися з часом.

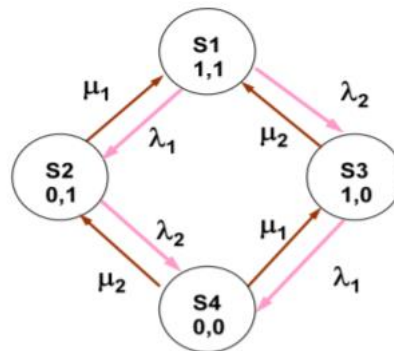
Системи масового обслуговування – це динамічні неперервні системи. В прикладі розглядається система S (наприклад, монтажний цех), яка має два пристрої (наприклад, сервери). Кожен пристрій може мати два стани: (1,0). Наприклад, «працює» і «не працює», «зайнятий - вільний» та інше. Тоді S може перебувати в одному з чотирьох станів:

- S1 - обидва пристрої працюють;
- S2 - перший пристрій не працює, другий працює;
- S3 - перший пристрій працює, другий не працює;
- S4 - обидва пристрої не працюють.

Введемо позначення для потоків подій:

- λ_i - інтенсивності переходу i -го пристрою в стан 0;
- μ_i - інтенсивності переходу i -го пристрою в стан 1.

Дану систему можна розглядати як двоканальну СМО з потоками запитів λ_i та потоками обслуговування μ_i . Побудуємо граф станів системи.



Допустим, p_i - це імовірність (перехідна імовірність), що система перебуває в стані S_i . Тоді для стану S1 вираз

$$p_1(\lambda_1 + \lambda_2) = p_2 \mu_1 + p_3 \mu_2 \quad (2.2)$$

є ємністю переходу системи зі стану S1, віднесеною до одиниці часу. Розглядаючи баланс приходу та відходу для кожного з чотирьох станів, ми отримуємо систему рівнянь Колмогорова:

$$\begin{cases} p_1(\lambda_1 + \lambda_2) = p_2 \mu_1 + p_3 \mu_2, \\ p_2(\lambda_2 + \mu_1) = p_1 \lambda_1 + p_4 \mu_2, \\ p_3(\lambda_1 + \mu_2) = p_1 \lambda_2 + p_4 \mu_1, \\ p_4(\mu_1 + \mu_2) = p_2 \lambda_2 + p_3 \mu_1. \end{cases} \quad (2.3)$$

Прийняв до уваги те, що система з ймовірністю 1 перебуває в одному з чотирьох станів, то

$$p_1 + p_2 + p_3 + p_4 = 1. \quad (2.4)$$

Моделювання складних систем масового обслуговування за допомогою систем рівнянь часто є праце-витратно і складно піддається аналізу. При цьому виявляється природним розглядати такі системи з точки зору дискретно-подієвого підходу.

2.4. Інструментальні засоби для побудови імітаційних моделей

Інструментальні засоби для побудови імітаційних моделей дозволяють інженерам і науковцям створювати складні імітаційні моделі для дослідження та аналізу різноманітних систем. Деякі з найбільш популярних інструментів для побудови імітаційних моделей включають в себе:

1. AnyLogic (рисунок 2.) – це програмне забезпечення для побудови імітаційних моделей, яке дозволяє моделювати системи будь-якої складності. AnyLogic підтримує різні типи моделювання, такі як дискретно-подієві моделі, моделі на основі агентів, системно-динамічні моделі та інші [21].

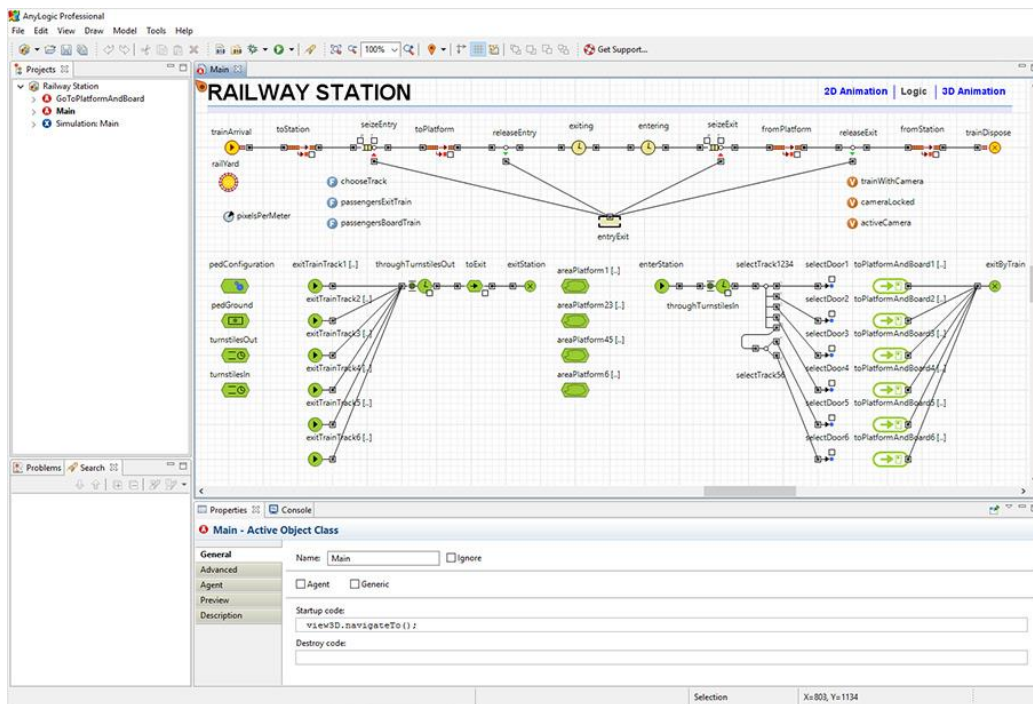


Рис. 2.5 – Anylogic

2. Simulink (рисунок 2.) – це інструмент для побудови імітаційних моделей, що розробляється компанією MathWorks. Simulink дозволяє моделювати різні системи, включаючи комп'ютерні мережі, за допомогою блоків, які відповідають різним функціональним елементам системи і розгортання без написання коду [22].

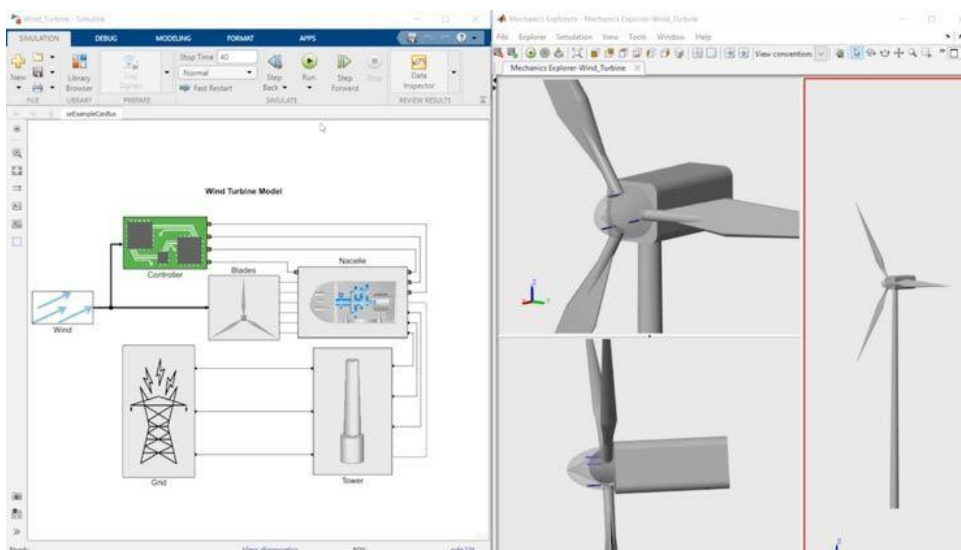


Рис 2.6 – Simulink

3. Arena (рисунок 2.) – це програмне забезпечення для моделювання та аналізу складних систем. Arena дозволяє створювати дискретно-подієві моделі та моделі на основі системної динаміки, які можуть бути використані для дослідження різних аспектів комп'ютерних мереж [23].

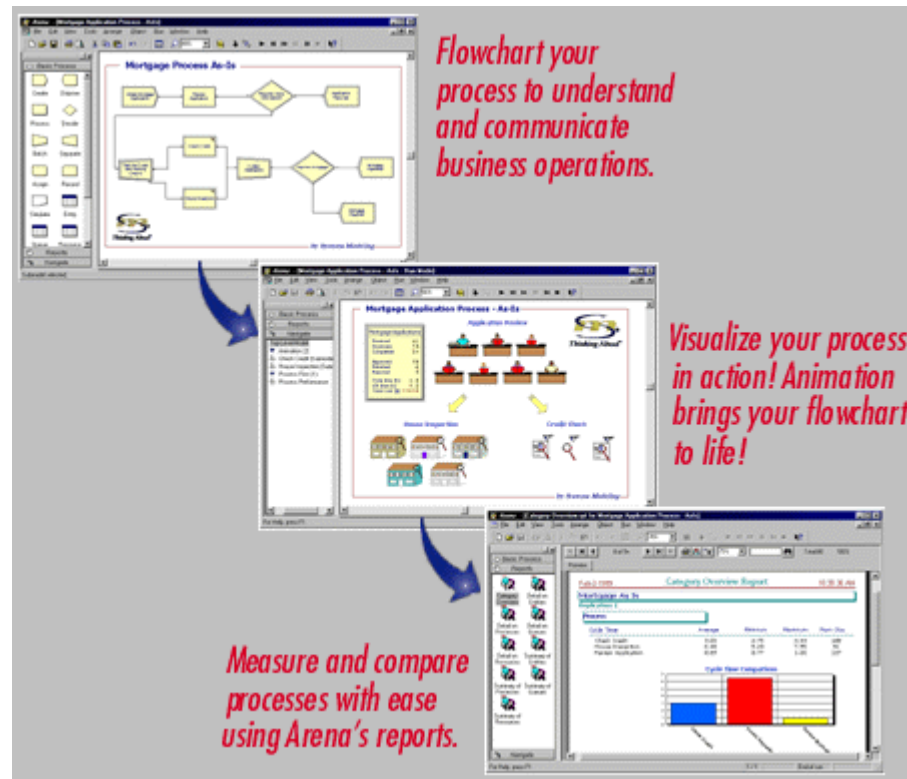


Рис 2.7 – Arena [23]

4. NetLogo (рисунок 2.) – це інструмент для моделювання на основі агентів, який дозволяє створювати моделі з великою кількістю взаємодіючих агентів. NetLogo забезпечує зручний інтерфейс користувача та різноманітні засоби візуалізації даних, що робить його популярним для дослідження комп'ютерних мереж, його використовують багато сотень тисяч студентів, викладачів і дослідників по всьому світу [24].

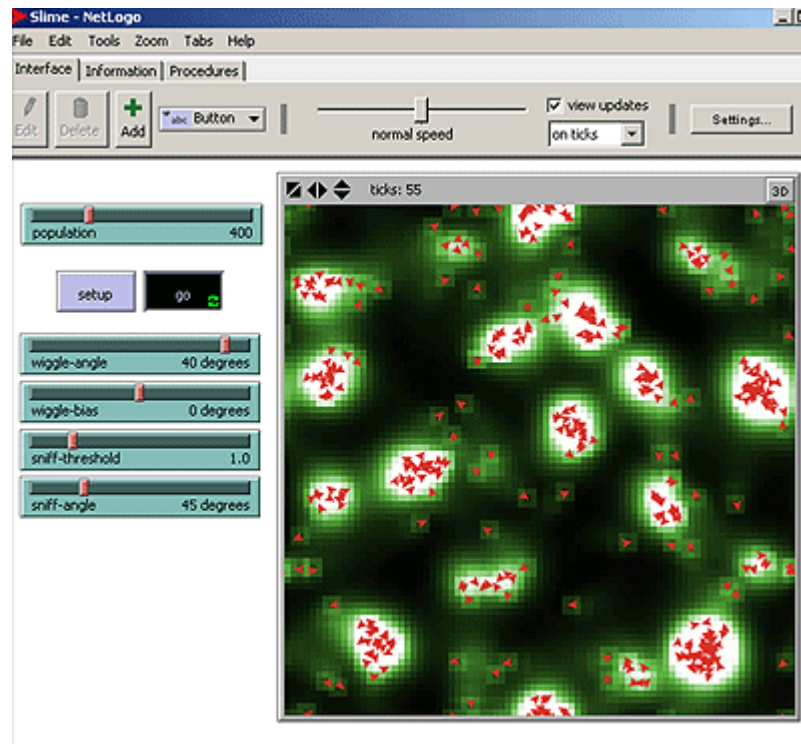


Рис 2.8 – NetLogo [25]

2.5 Обґрунтування вибору методів та інструментів імітаційного моделювання для оцінки продуктивності проміжних вузлів комп'ютерних мереж

Наведений аналіз показав, що використання дискретно-подієвих моделей та програми AnyLogic є обґрунтованим вибором для оцінки продуктивності проміжних вузлів комп'ютерних мереж. Дискретно-подієві моделі дозволяють моделювати події, що відбуваються в системі в конкретний момент часу, та їх взаємодію. Такі моделі дозволяють реалістично відтворити поведінку системи та оцінити продуктивність її складових.

AnyLogic є зручним інструментом для побудови імітаційних моделей, що дозволяє моделювати системи різного рівня складності. Програма підтримує різні типи моделей, включаючи дискретно-подієві моделі, а також інші типи моделей, такі як моделі на основі агентів, системно-динамічні

моделі, моделі на основі графів та моделі на основі мереж Петрі.

РОЗДІЛ 3

ПОБУДОВА ТА ВИКОРИСТАННЯ ІМІТАЦІЙНОЇ МОДЕЛІ ДЛЯ
ОЦІНКИ ПРОДУКТИВНОСТІ ПРОМІЖНОГО ВУЗЛА МЕРЕЖІ

3.1. Опис математичної моделі

М/М/Н/К - це модель черги, де надходження заявок відповідає процесу Пуассона, а час обслуговування заявок - експоненціальному розподілу. Літера «М» позначає «марковський» процес, що означає, що момент надходження наступної заявки не залежить від історії попередніх заявок.

"N" вказує на кількість каналів обслуговування (обслуговувальних приладів), доступних для обслуговування заявок.

"K" позначає максимальну ємність черги, тобто максимальну кількість заявок, які можуть бути в черзі одночасно.

Ця модель використовується для аналізу ефективності систем масового обслуговування, де надходження та обслуговування заявок відбуваються випадковим чином. Вона дозволяє визначити такі показники, як інтенсивність надходження, інтенсивність обслуговування, середню кількість заявок у черзі, середній час очікування та інші параметри, що допомагають оцінити ефективність системи обслуговування [26].

В цій роботі буде моделюватись система типу М/М/2/1 (рисунок 3.1)

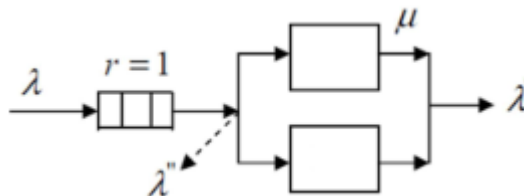


Рис. 3.1 – Двоканальна система обслуговування

$k=0$ - в системі немає додатків;

$k=1$ - в системі є один додаток;

$k=2$ - в системі є два додатки;

$k=3$ - в системі є три додатки.

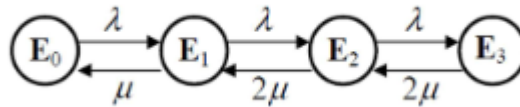


Рис. 3.2 – Граф переходів з позначками [28]

Таким чином можна змоделювати роботу простого вузла обробки в комп'ютерній мережі, який одночасно може обробляти дві операції замовлення, замовлення надходять з черги без пріоритетів.

Вхідними параметрами моделі є частота надходження замовлень λ а також частота обробки замовлень, яка характеризує виробничу потужність вузла мережі μ .

На основі цих вхідних параметрів можуть бути розраховані теоретичні вихідні параметри моделі, які представлені в таблиці 3.1.

Таблиця 3.1.

Математичні характеристики системи M/M/2/1 [26, 27].

Назва	Формула
Навантаження (Load)	$y = \lambda / \mu = \lambda b$
Використання (Utilization)	$\rho = (p_1 + 2 * p_2 + 2 * p_3) / 2$
Середня кількість робочих інструментів (Average number of working instruments)	$k = (p_1 + 2 * p_2 + 2 * p_3) = 2\rho$
Співвідношення простою системи (System idle ratio)	$\eta = 1 - \rho$

Середня кількість запитів у черзі (Average number of requests in the queue)	$l = p_3$
Середня кількість додатків у системі (Average number of applications in the system)	$m = p_1 + 2 * p_2 + 3 * p_3 = l + k'$
Ймовірність втрати запитів (Probability of losing requests)	$\pi = p_3$
Потік обслугованих запитів (The flow of served requests)	$\lambda' = \lambda(1 - \pi)$
Потік запитів, які були відхилені (The flow of requests denied service)	$\lambda'' = \lambda\pi$
Середній час очікування додатків (Average waiting time of applications)	$w = l/\lambda'$
Середній час додатків (Average time of applications)	$u = m/\lambda' = w + b$

Завантаження (Load) - це кількість вхідних запитів, що потрапляють до системи. Використання (Utilization) - це співвідношення виконаних запитів до вхідного потоку. Загальна кількість пакетів, які надходять до системи, визначає навантаження. Використання вказує, як система обробляє ці пакети.

Співвідношення простою системи (System idle ratio) - це обернений показник навантаження або використання. Воно відображає ймовірність того, що система буде порожньою [31].

Середня кількість запитів у черзі (Average number of requests in the queue) – це кількість пакетів, які очікують обслуговування і розташовані у буфері.

Середня кількість додатків у системі (Average number of applications in

the system) - це сума двох показників. Один пакет очікує обслуговування, а інший знаходиться в процесі обробки. Максимальна завантаженість дорівнює одиниці в системі M/M/2/1.

Втрата пакетів виникає через обмежений розмір буфера. Ймовірність втрати запитів визначається ймовірністю втрати пакетів.

Середній час очікування додатків (Average waiting time of applications) - це час, який абонент проводить у черзі, очікуючи обслуговування.

Середній час додатків (Average time of applications) - це сума двох часових інтервалів. Перший - це час очікування абонента в черзі, а другий - це час обслуговування абонента на сервер [32].

За допомогою формул, вказаних у таблиці 1, отримуються теоретичні математичні результати.

Використання (Utilization) $\rho < 1$ для m -сервера в інтервалі від 0.1 до 0.9. Щоб обчислити значення використання, відомі значення p_1, p_2, p_3 , які визначаються за допомогою графа переходів на рисунку 3.2. Початкові параметри рівняються кінцевим, і формули записуються з урахуванням умов E_0, E_1, E_2, E_3 [29].

Формулу, отриману за допомогою p_0 , застосовують до p_1, p_2, p_3 , і отримане значення підставляють у наступну формулу: $p_0 + p_1 + p_2 + p_3 = 1$ [30]. Значення p_0 знаходиться за допомогою формули перевантаження, і також відомі значення p_1, p_2, p_3 , які використовуються при обчисленні p_0 .

$$p_0 = \frac{1}{1 + \frac{\lambda}{\mu} + \frac{\lambda^2}{2\mu^2} + \frac{\lambda^3}{4\mu^3}} \quad (3.1)$$

$$p_1 = p_0 \frac{\lambda}{\mu}; p_2 = p_0 \frac{\lambda^2}{2\mu^2}; p_3 = p_0 \frac{\lambda^3}{4\mu^3}; \quad (3.2)$$

Значення p_0, p_1, p_2, p_3 , підставляються в формулу використання, і стан

зайнятості λ_1 , λ_2 , λ_3 , λ_4 обчислюється чотири рази. Інші показники обчислюються за формулами, показаними в таблиці 3.1.

3.2. Побудова імітаційної моделі в середовищі AnyLogic

Першим кроком буде створення основного файлу моделі. Для моделювання логіки моделі буде використана бібліотека моделювання процесів:

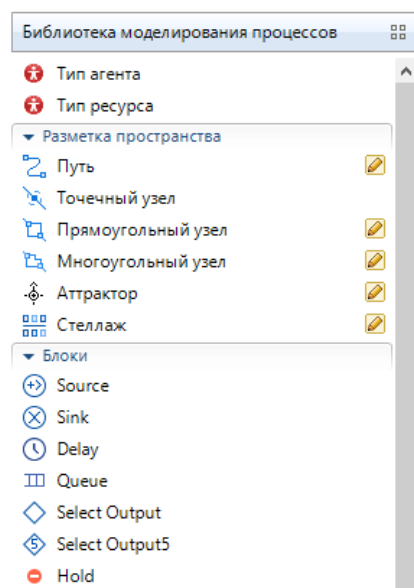


Рис 3.3 – Бібліотека моделювання процесів

Наступним кроком буде створення нового типу агенту (рисунок 3.4), змінимо його назву «arr» та продовжимо налаштування. При запрошенні надати йому анімацію, натискаємо ні та йдемо далі.

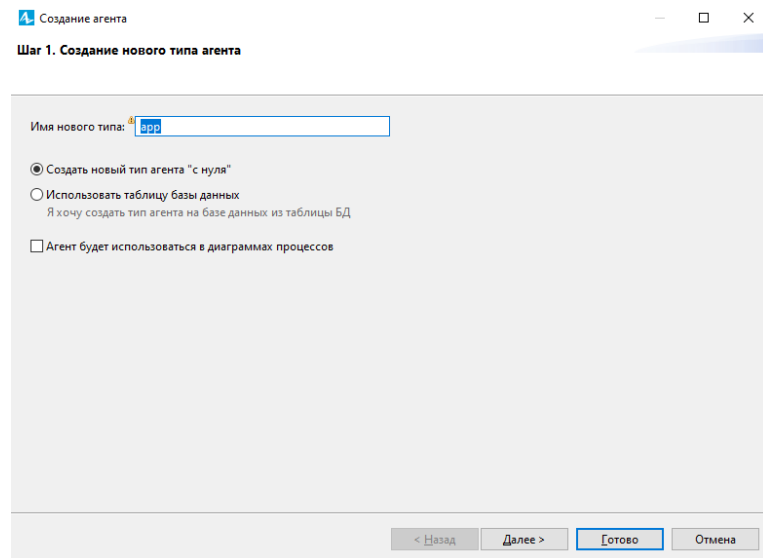


Рис. 3.4 – Присвоєння ім'я для типу агенту

Після створення агенту наступним кроком буде перенесення його до основного поля в Main. Після цього створюємо елемент Rectangle Node (рисунок 3.5), який дозволяє генерувати замовлення типу App через створення аттракторів:

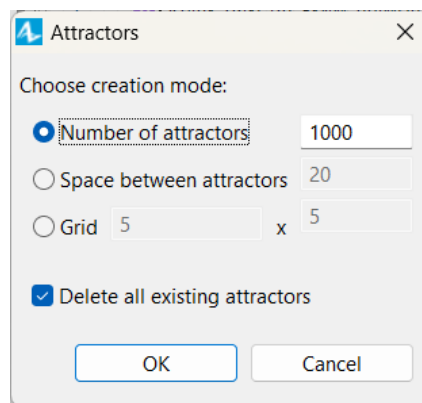


Рис. 3.5 – Створення елементу Rectangle Node

До поля із бібліотеки додаємо джерело заявок, цей блок має назву «source». В його налаштуваннях (рисунок 3.6) у полі агенту міняємо значення на «app». У полі часу між прибуттям даємо значення нашої змінної λ , яка в наступному буде змінюватись від потреби.

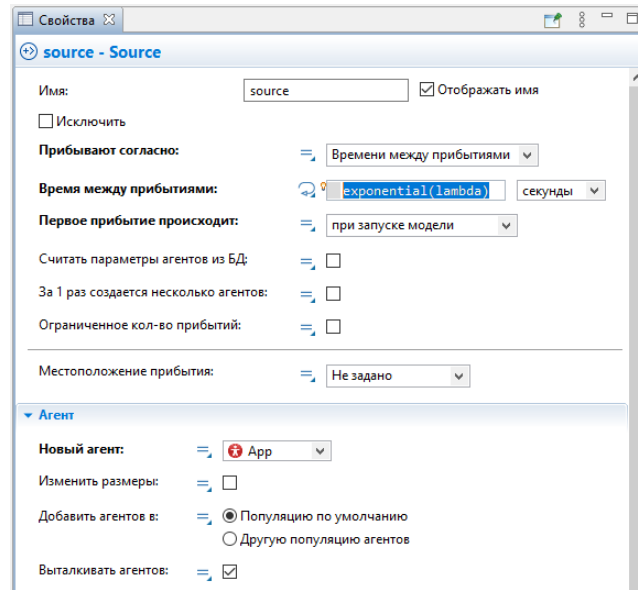


Рис. 3.6 – Налаштування блоку source

Наступним кроком є додавання блоку «queue» до джерела заявок та налаштування його параметрів (рисунок 3.7), щоб забезпечити моделювання черги до обробки замовлення.

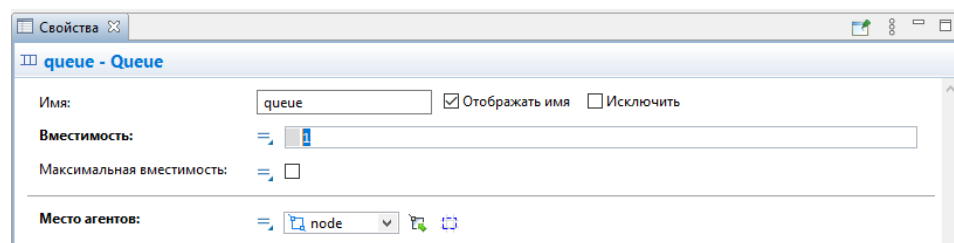


Рис.3.7 – Параметри налаштування блоку «queue»

Для розширення функціональності було додано до «queue» блоки «sink» (для моделювання кількості замовлень, яким було відмовлено в обслуговуванні) та «timeMeasureStart» (для відслідковування кількості замовлень, які поступають на обробку до вузла) в налаштуванні яких все залишилось без змін.

На даний момент модель виглядає наступним чином:

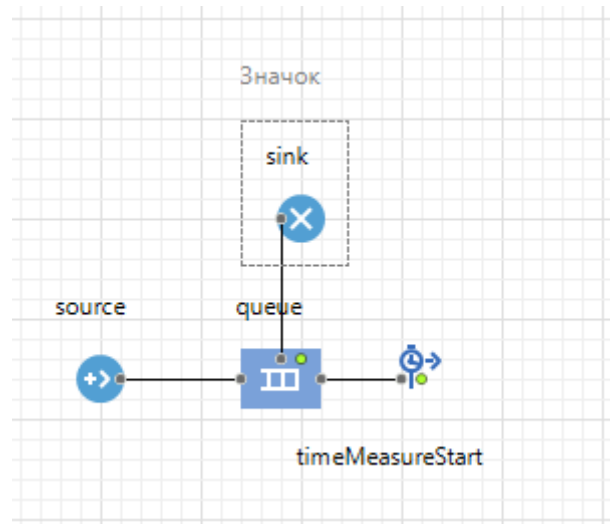


Рис. 3.8 – Результат підключення початкових блоків

Наступним кроком було додавання до схеми затримки з своїми показниками (рисунок 3.9), вона має імітувати процес оброблення замовлення. Час затримки буде братись з змінної « μ », яка в майбутньому буде мати вид $1/b$.

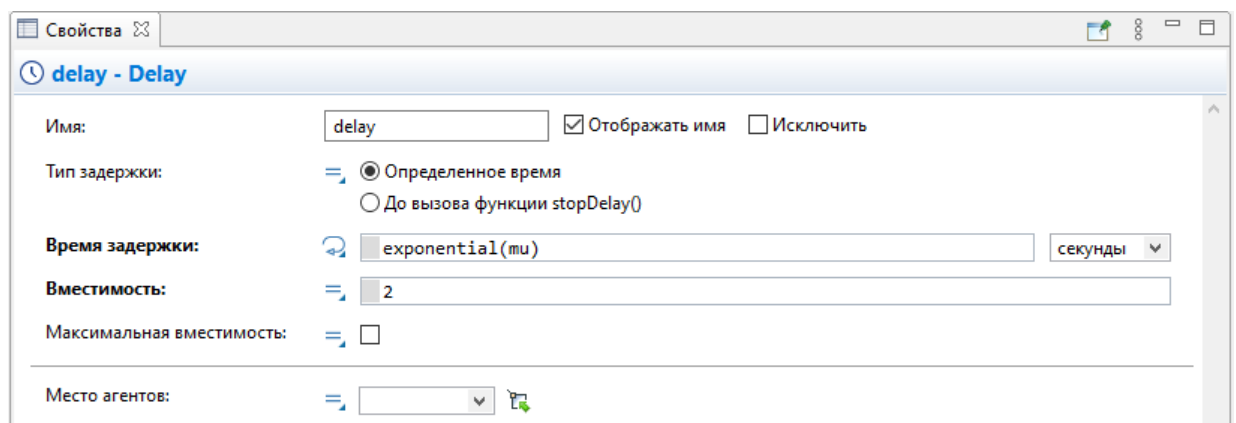


Рис. 3.9 – Налаштування «delay»

Блоки «timeMeasureStart» та «timeMeasureEnd» дозволяють виміряти час, проведений агентами в обмеженій цими блоками ділянці, отже до «delay» було підключено другий блок «timeMeasureEnd», який замикає процес вимірювання часу. Після блоку «timeMeasureEnd» був підключений кінцевий блок «sink» без внутрішніх змін. Отже, кінцевий результат схеми має такий

ВИД:

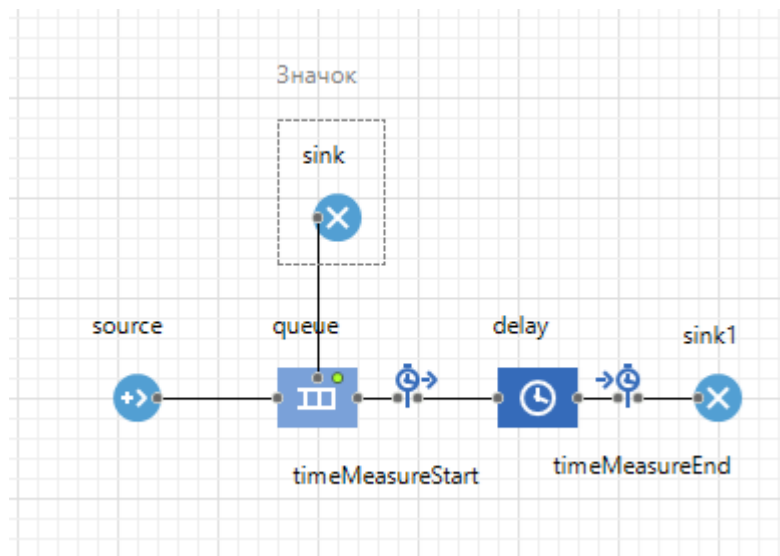


Рис. 3.10 – Кінцевий результат

Наступним кроком буде створення змінних (рисунок 3.11) та додавання до них формул, значення яких будуть взяті з Таблиці 3.1, що відображає математичні характеристики системи $M/M/2/1$. Ці змінні дозволять нам враховувати різні параметри системи, такі як середній час обслуговування, середнє число заявок у черзі, ймовірність відмови та інші. Використовуючи ці змінні в формулі, ми зможемо отримати більш детальну та точну оцінку продуктивності системи $M/M/2/1$.

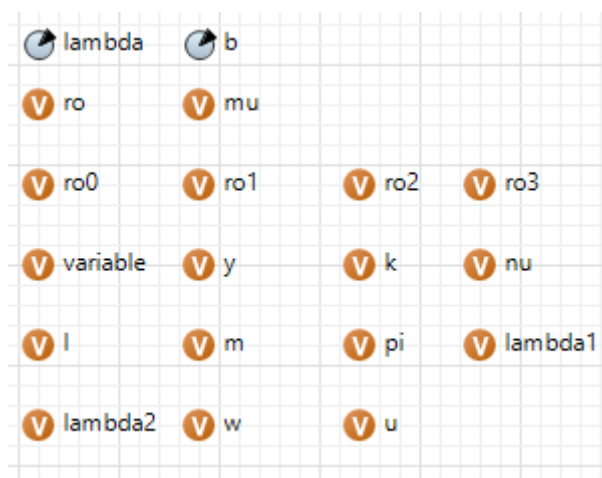


Рис. 3.11 – Усі змінні для схеми

Для збору даних з роботи схеми було взято 3 однакові елементи статистики. Кожному з них були надані імена та значення, які дозволяють виміряти певні характеристики процесу. Перший елемент статистики вимірює довжину черги (рисунок 3.12) і дає нам інформацію про кількість заявок, які очікують на обробку.

Рис. 3.12 – Довжина черги

Другий елемент статистики вимірює кількість пакетів, які успішно оброблені (рисунок 3.13) і дозволяє нам оцінити продуктивність процесу.

Рис. 3.13 – Кількість пакетів які були оброблені

Третій елемент статистики вимірює кількість пакетів, яким було відмовлено (рисунок 3.14) і надає нам інформацію про недоліки або проблеми в процесі обробки. Збираючи дані за допомогою цих елементів статистики, ми можемо аналізувати та вдосконалювати роботу схеми.

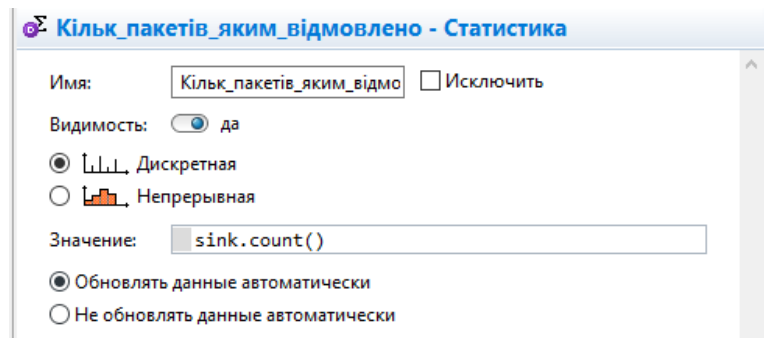


Рис 3.14 – Кількість пакетів яким було відмовлено

Додаймо до створеної схеми ще 4 діаграми, які допоможуть нам краще проаналізувати продуктивність системи. Кожна діаграма має свою важливу роль.

Першою буде гістограма (рисунок 3.15) її ціль відображати час обробки. Це дозволить нам візуально спостерігати, як змінюється час, необхідний для обробки заявок у системі.

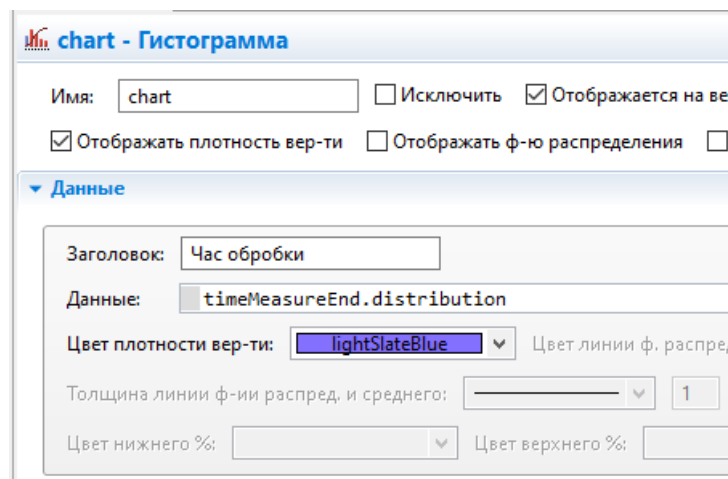
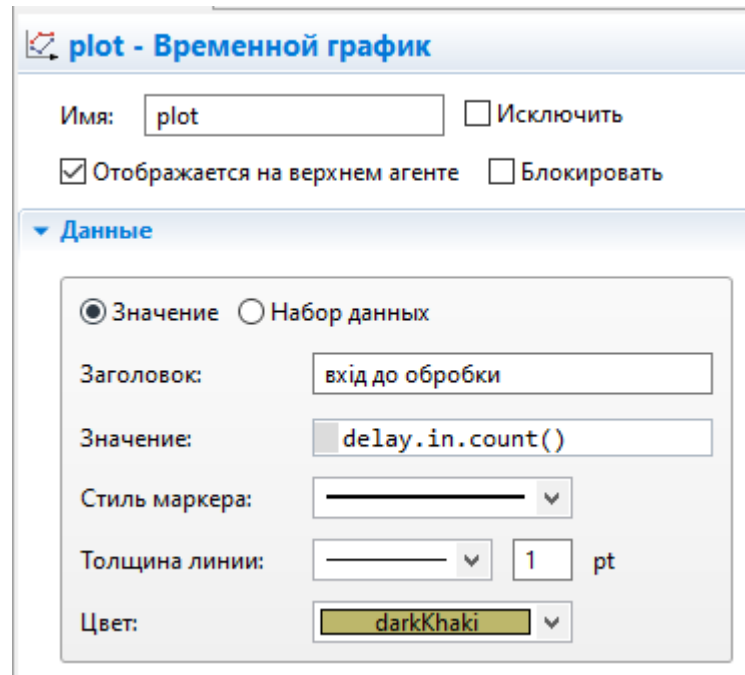


Рис. 3.15 – Налаштування гістограми

Другою буде часовий графік (рисунок 3.16) він покаже вхід та вихід з обробки. Це допоможе нам оцінити навантаження на систему і зрозуміти, як швидко вона здатна обробляти заявки.



plot - Временной график

Имя: Исключить

Отображается на верхнем агенте Блокировать

▼ Данные

Значение Набор данных

Заголовок:

Значение:

Стиль маркера: ▼

Толщина линии: ▼ pt

Цвет: ▼

Рис 3.16 – Налаштування часового графіку

Третя діаграма буде відображати середні значення оброблених заявок. Це дасть нам уявлення про середню продуктивність системи і дозволить порівняти різні періоди часу.

Рис 3.17 – Налаштування діаграми обробки

Нарешті, четвертий графік (рисунок 3.18) показуватиме вхід та вихід з черги. Це допоможе нам зрозуміти, як ефективно система керує чергою та розподіляє навантаження.

plot1 - Временной график

Имя: Исключить

Отображается на верхнем агенте Блокировать

▼ Данные

Значение Набор данных

Заголовок:

Значение:

Стиль маркера:

Толщина линии: pt

Цвет:

Значение Набор данных

Заголовок:

Значение:

Стиль маркера:

Толщина линии: pt

Цвет:

Рис 3.18 – Налаштування часового графіку черги

Таким чином, після додавання цих 4 діаграм, кінцевий вигляд нашої моделі буде наступним:

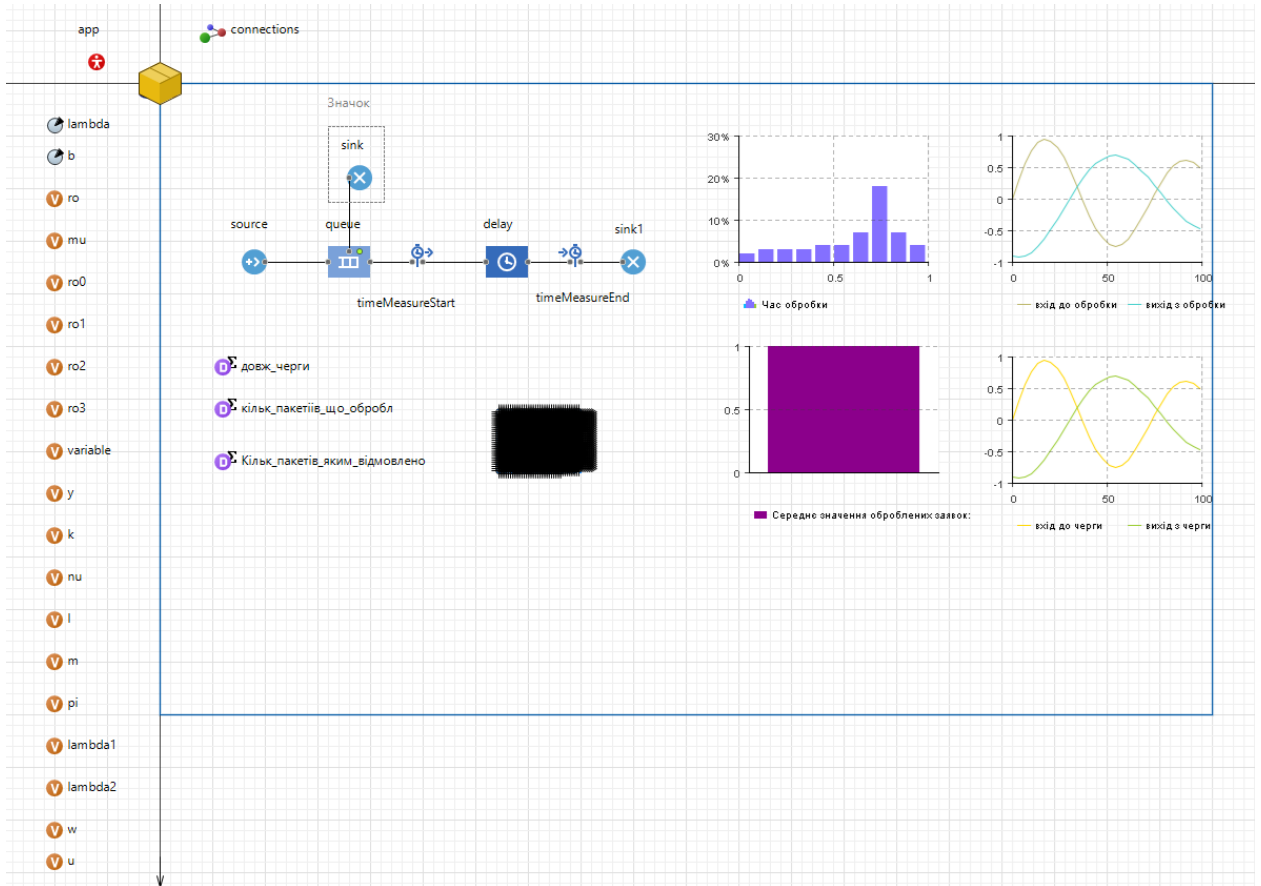


Рис 3.19 – Кінцевий результат

Після завершення роботи з імітаційною моделлю було прийнято рішення розширити модель шляхом додавання додаткових змінних для оцінки відсотку оброблених запитів і відсотку відхилених запитів (рисунок 3.20).

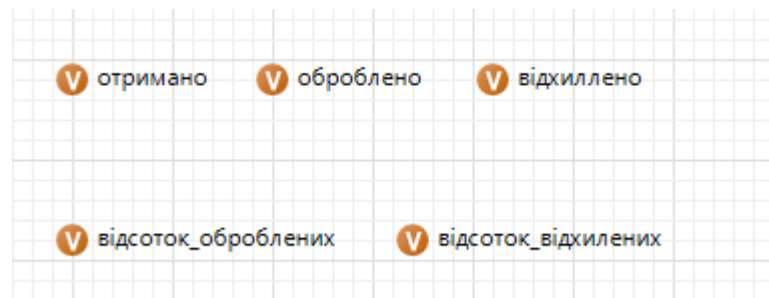


Рис. 3.20 – Нові змінні

Для оновлення отриманих заявок у блоку «source» (рисунок 3.8) було додано «отримано++» (рисунок 3.21), що вказує на кількість нових заявок, які

надійшли. Ця зміна дозволяє більш детально відстежувати потік заявок і аналізувати його динаміку.

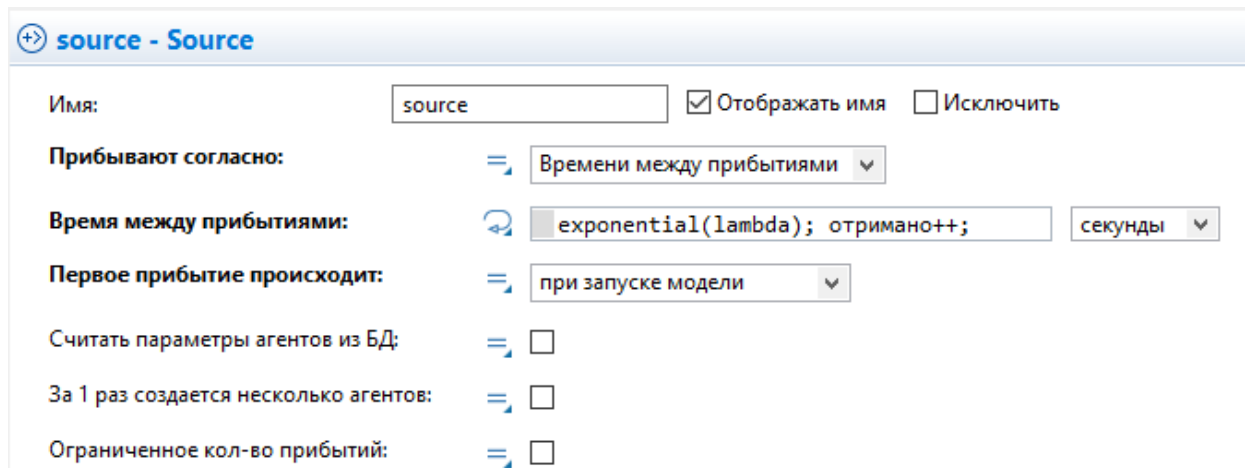


Рис. 3.21 – Змінення налаштувань блоку «source»

Для оновлення заявок які були відхилені, до блоку «sink» (рисунок 3.8) було додано «відхилено++» (рисунок 3.22) та одразу ж було прораховано відсоток відхилених заявок. Для розрахунку відсотку відхилених заявок було використано формулу (відхилено / отримано) *100.

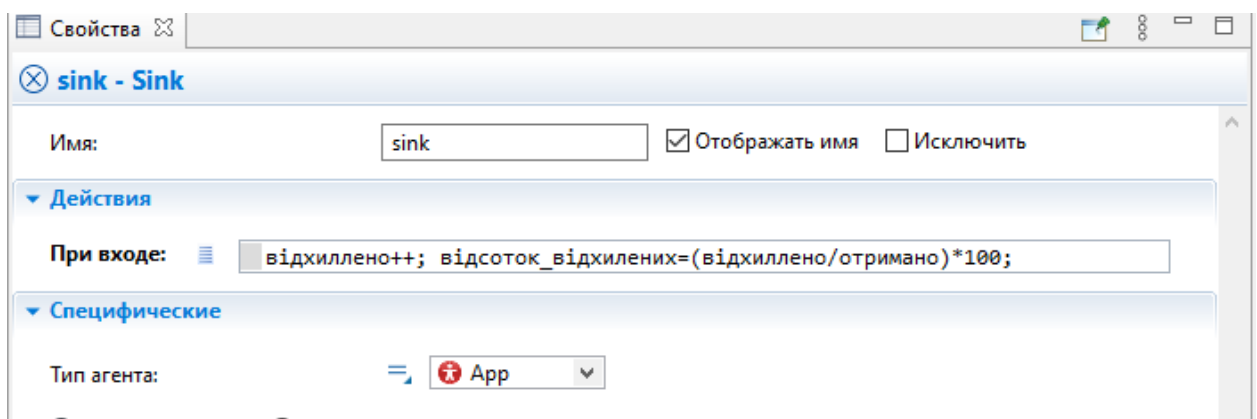


Рис. 3.22 – Змінення налаштувань блоку «sink»

Для оновлення оброблених заявок до блоку «sink1» було додано «оброблено++» (рисунок 3.23) та одразу ж було прораховано відсоток оброблених заявок. Для розрахунку було використано формулу

відсоток_оброблених = (оброблено/отримано)*100;

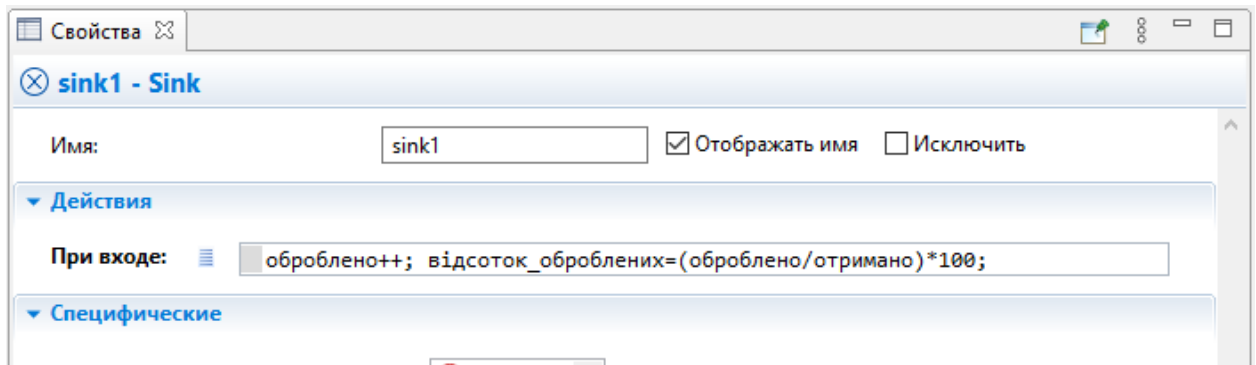


Рис. 3.23 – Змінення налаштувань блоку «sink1»

3.3 Проведення та аналіз експериментів

Для визначення навантаження у необхідно знати значення середньої кількості пакетів і середнього часу обслуговування запитів. Ці значення обираються добровільно. Значення b обирається одним постійним значенням. Значення λ були промодельовані чотирма значеннями:

$$\lambda_1 = 0,21, \lambda_2 = 0,45, \lambda_3 = 0,64, \lambda_4 = 0,81$$

Результати роботи моделі представлені в таблиці 3.2.

Таблиця 3.2.

Результати експериментів з моделлю

	$\lambda_1 = 0,21$	$\lambda_2 = 0,45$	$\lambda_3 = 0,64$	$\lambda_4 = 0,81$
Використання (ρ)	0.207	0.417	0.547	0.638
Середня кількість робочих інструментів (k)	0.415	0.834	1.095	1.275
Співвідношення простою системи (η)	0.793	0.583	0.453	0.362
Середня кількість запитів у черзі (l)	0.012	0.073	0.145	0.213

Середня кількість додатків у системі (m)	0.472	0.907	1.239	1.488
Ймовірність втрати запитів (π)	0.012	0.073	0.145	0.213
Потік обслугованих запитів (λ')	0.207	0.417	0.547	0.784
Потік запитів, які були відхилені (λ'')	0.003	0.033	0.093	0.172
Відсоток оброблених	99 %	92 %	88,5 %	78 %
Відсоток відхилених	1 %	8 %	11,5 %	22 %

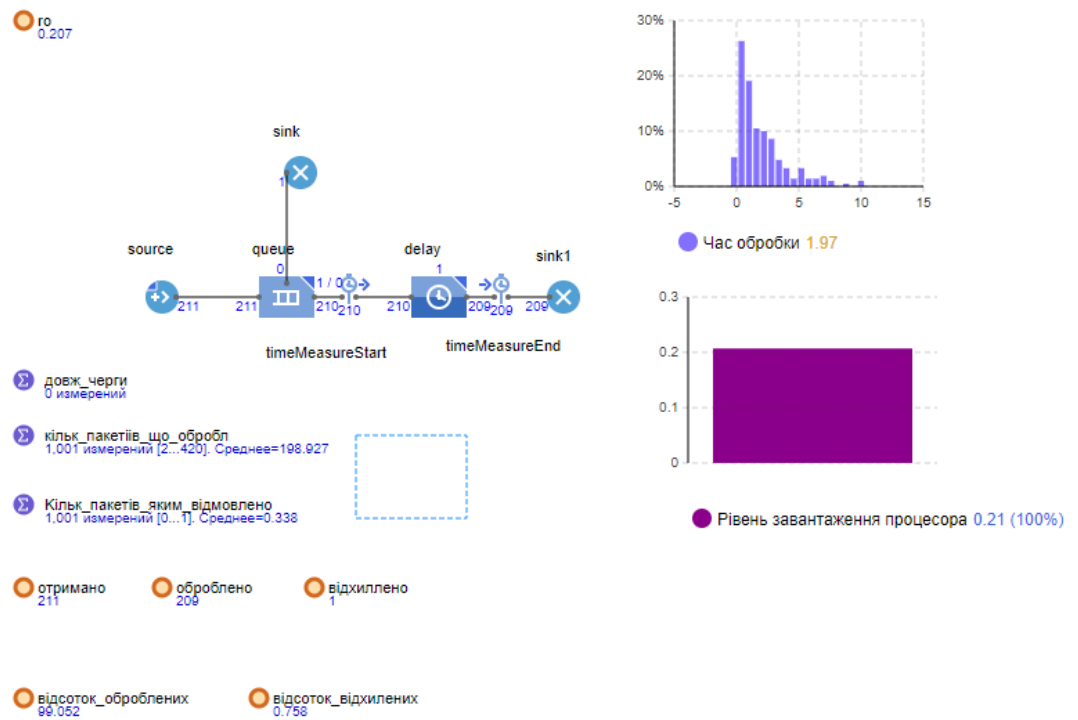


Рис. 3.24 - Результат імітації при $\lambda = 0.21$

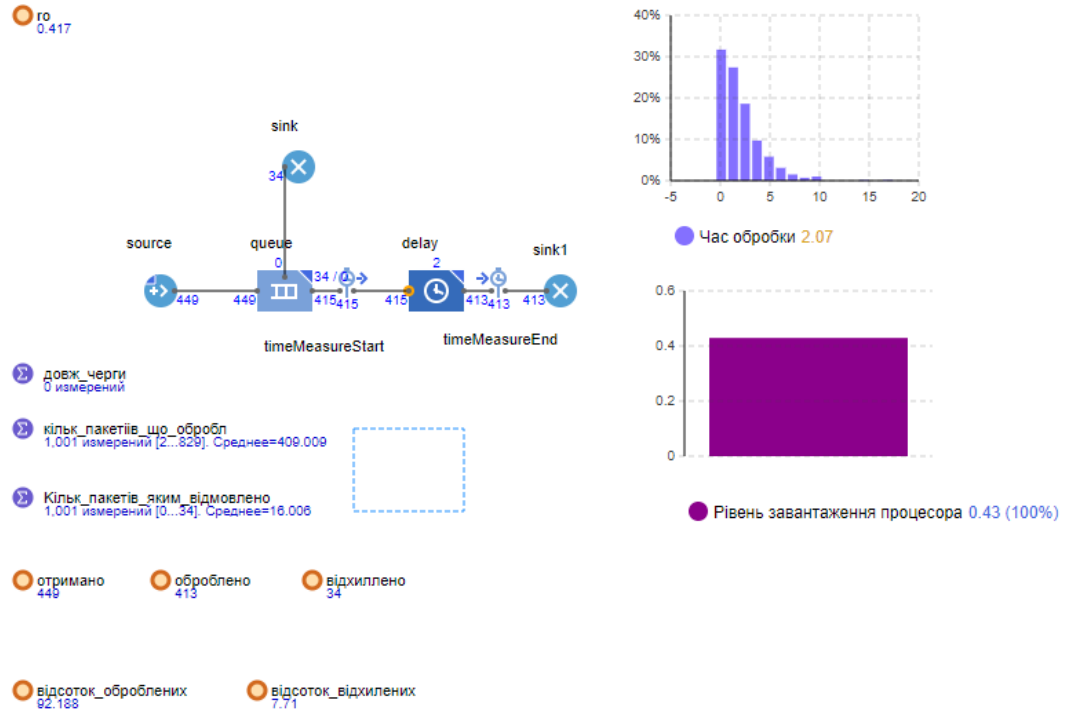


Рис. 3.25 - Результат імітації при $\lambda = 0.45$

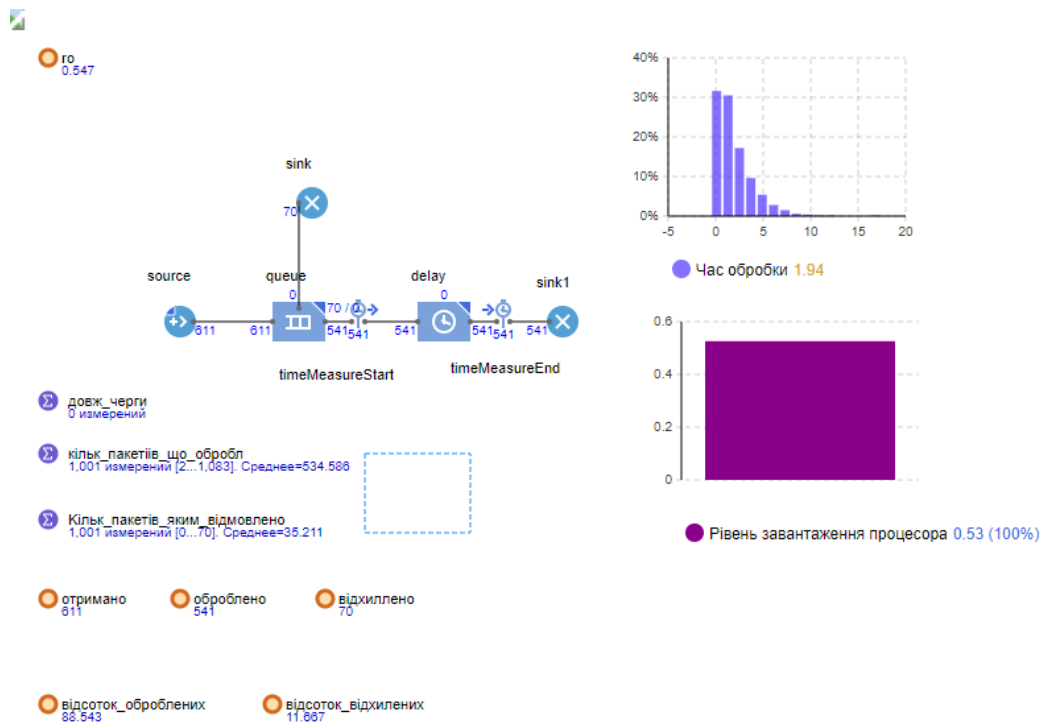


Рис. 3.26 - Результат імітації при $\lambda = 0.64$

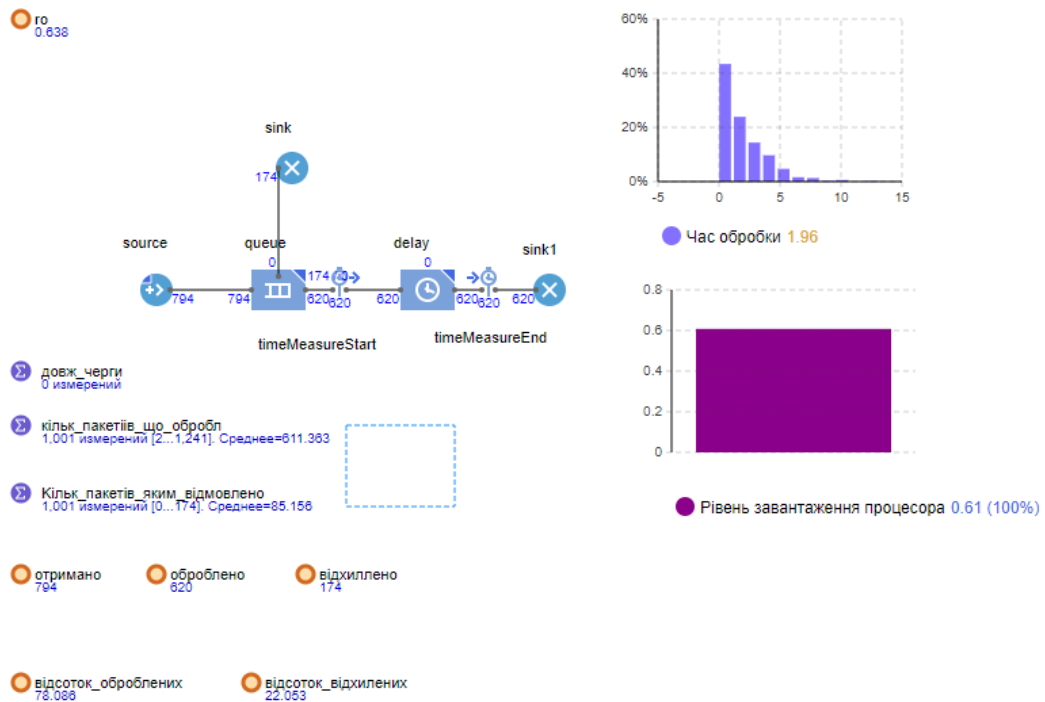


Рис. 3.27 - Результат імітації при $\lambda = 0.81$

Використання (ρ) та співвідношення простою системи (η). Зі збільшенням інтенсивності надходження запитів (λ), значення використання (ρ) також зростають. Це означає, що система інтенсивніше використовується для обробки запитів, одночасно співвідношення простою системи (η) зменшується, що вказує на скорочення часу простою системи.

Середня кількість робочих інструментів (k) та середня кількість додатків у системі (m). Зі збільшенням інтенсивності надходження запитів (λ), значення k та m також збільшуються. Це свідчить про більшу кількість запитів, які знаходяться в системі та потребують обробки.

Середня кількість запитів у черзі (l). Зі зростанням інтенсивності надходження запитів (λ), значення l також зростає. Це вказує на збільшення кількості запитів, які очікують своєї черги в черзі перед завершальним блоком системи.

Потік обслугованих запитів (λ') та потік запитів, які були відхилені (λ''). Зі збільшенням інтенсивності надходження запитів (λ), значення λ' та значення

λ " зростає. Це свідчить про те, що зі збільшенням навантаження на систему більше запитів відхиляються та не обробляються.

Отже, результати з таблиці показують взаємозв'язок між інтенсивністю надходження запитів, використанням системи, завантаженням процесора, часом обробки та кількістю відхилених запитів.

Зміна показників системи також може бути пов'язана зі зміною її структури або параметрів. Наприклад, додавання нових ресурсів або зміна часу обслуговування можуть вплинути на швидкість обробки заявок і загальну продуктивність системи. Отже, аналіз і розуміння змін показників системи допомагають вдосконалити її роботу та приймати обґрунтовані рішення для підвищення ефективності і якості обслуговування.

Важливо підкреслити, що це спрощена модель, яка не враховує багато факторів реальної системи. Проте, вона дозволяє зрозуміти основні залежності між швидкістю приходу, кількістю заявок і часом обробки в Марківській моделі чергування.

ВИСНОВКИ

Дискретно-подієві моделі широко використовуються при аналізі комп'ютерних мереж з кількох причин:

Моделювання часу: Дискретно-подієві моделі дозволяють враховувати час і послідовність подій в мережі. Вони представляють мережеві події у вигляді дискретних моментів часу та моделюють їх у хронологічному порядку. Це дозволяє аналізувати динаміку мережі та прогнозувати поведінку вузлів і зв'язків у різних ситуаціях.

Моделювання конкурентних процесів: В комп'ютерних мережах одночасно відбувається безліч паралельних подій і взаємодій. Дискретно-побутові моделі дозволяють моделювати конкурентні процеси, такі як передача пакетів даних між вузлами або обробка запитів мережевими пристроями. Це допомагає оцінити пропускну здатність, захист і ефективність роботи мережі.

Дослідження стохастичних процесів: У комп'ютерних мережах багато факторів, які схильні до випадковості, такі як час затримки, втрати пакетів і зміни трафіку. Дискретно-побутові моделі можуть враховувати випадковість і моделювати стохастичні процеси, дозволяючи проводити аналіз і оцінювати ймовірність різних подій і стану мережі.

Оптимізація ресурсів: Дискретно-подієві моделі дозволяють оцінити використання ресурсів у комп'ютерних мережах і оптимізувати їх розподіл. Моделі можуть представляти вузли, зв'язки та протоколи мережі, а також ресурси, такі як пропускну здатність і пам'ять. Це допомагає ідентифікувати вузькі місця, прогнозувати продуктивність і приймати рішення з оптимізації ресурсів.

Тестування сценаріїв і алгоритмів: дискретно-побутові моделі дозволяють проводити віртуальні експерименти з різними сценаріями роботи мережі і алгоритмами маршрутизації, контролю навантаження і управління

трафіком. Це допомагає оцінити ефективність нових алгоритмів, протоколів і стратегій в різних умовах і прийняти рішення щодо їх застосування в реальних мережевих системах.

У цій роботі була розроблена система оброблення замовлень проміжним вузлом мережі, яка базується на дискретно-подієвому підході та має обмеження щодо кількості оброблювальних центрів (процесорів) та кількості замовлень в черзі. Втрата пакетів виникає внаслідок обмеженого розміру буфера.

З моделлю проведено низку експериментів, які дозволяють оцінити зміни параметрів продуктивності вузла мережі в залежності від частоти замовлень, виробничої потужності вузла, довжини черги.

Проте ця модель є спрощенням реальної проблеми і не враховує всі аспекти проблеми, а також може працювати лише в певних ситуаціях. Якщо ситуація змінюється, модель потрібно перераховувати з самого початку, що займає багато часу. Імітаційні моделі можуть бути корисними навіть при зміні стану. Ця модель також економить час і кошти. Проте, для того, щоб визначити, чи працює імітаційна модель належним чином, потрібно розробити математичну модель.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. What are Network Nodes in Computer Network and Their Types [Електронний ресурс] / Режим доступу: WWW. URL: <https://www.elprocus.com/what-are-network-nodes-in-computer-network-and-their-types/>
2. Вузол в комп'ютерній мережі [Електронний ресурс] / Режим доступу: WWW. URL: <https://uk.go-travels.com/39429-what-is-a-node-4155598-4160332>
3. КОМП'ЮТЕРНІ МЕРЕЖІ [Електронний ресурс] / Режим доступу: WWW. URL: <http://nkker.com/wp-content/uploads/2020/11/P-21-Form-04.11.pdf>
4. Що таке вузол в комп'ютерній мережі? [Електронний ресурс] / Режим доступу: WWW. URL: <https://www.lifewire.com/what-is-a-node-4155598>
5. What are the Components of Computer Network? [Електронний ресурс] / Режим доступу: WWW. URL: <https://www.interviewbit.com/blog/components-of-computer-network/>
6. Принципы построения локальных сетей [Електронний ресурс] / Режим доступу: WWW. URL: https://studme.org/50138/informatika/printsipy_postroeniya_lokalnyh_setey_о_snovnye_komponenty_naznachenie_funktsii
7. Основи комп'ютерних мереж [Електронний ресурс] / Режим доступу: WWW. URL: <https://habr.com/ru/post/307252/>
8. What is Network Topology? [Електронний ресурс] / Режим доступу: WWW. URL: <https://www.javatpoint.com/computer-network-topologies>
9. Types of Network Topology [Електронний ресурс] / Режим доступу: WWW. URL: <https://www.geeksforgeeks.org/types-of-network-topology/>
10. Анализ продуктивности [Електронний ресурс] / Режим доступу: WWW. URL: <https://studfile.net/preview/4240/>

11. Characteristics of Computer networks [Електронний ресурс] / Режим доступу: WWW. URL: <https://www.codingninjas.com/codestudio/library/basic-characteristics-of-computer-networks>
12. Computer Networks Performance Modeling and Simulation [Електронний ресурс] / Режим доступу: WWW. URL: https://www.researchgate.net/publication/282076891_Computer_Networks_Performance_Modeling_and_Simulation
13. Методи расчёта показателей производительности сетей ЭВМ с неоднородным трафиком [Електронний ресурс] / Режим доступу: WWW. URL: <http://elib.osu.ru/bitstream/123456789/10372/1/%D0%9A%D0%BE%D0%BD%D0%BD%D0%BE%D0%B2%20%D0%9C%D0%B5%D1%82%D0%BE%D0%B4%D1%8B%20%D1%80%D0%B0%D1%81%D1%87%D0%B5%D1%82%D1%8B.pdf>
14. Burbank, J., Kasch, W., Ward, J., 2011. An Introduction to Network Modeling and Simulation for the Practicing Engineer. The ComSoc Guides to Communications Technologies. Wiley.
15. Wainer, G. A., 2009. Discrete-Event Modeling and Simulation: A Practitioner's Approach, 1st Edition. CRC Press, Inc., Boca Raton, FL, USA
16. Law, A.M. Kelton D.M., 1999. Simulation Modeling and Analysis. McGraw-Hill Higher Education.
17. Pidd, M., 1988. Computer simulation in management science. Wiley.
18. Taylor, H. E., Karlin, S., Feb. 1998. An Introduction to Stochastic Modeling, 3rd Edition. Academic Press.
19. Імітаційні моделі [Електронний ресурс] / Режим доступу: WWW. URL: <https://lib.chmnu.edu.ua/pdf/posibnuku/281/3.pdf>
20. Неруш. В. Б. Імітаційне моделювання систем та процесів: Електронне навчальне видання. Конспект лекцій / В. Б. Неруш, В. В. Курдеча. – К.: НН ІТС НТУУ «КПІ», 2012. [Електронний ресурс] / Режим доступу: WWW.

URL:

http://ela.kpi.ua/bitstream/123456789/15598/1/Konspekt_lekciy_Imit_modelyr_syst_process%28CHANGED%29.pdf

21. AnyLogic [Электронный ресурс] / Режим доступа: WWW. URL: <https://www.anylogic.ru/features/>
22. Simulink [Электронный ресурс] / Режим доступа: WWW. URL: <https://www.mathworks.com/products/simulink.html>
23. Arena [Электронный ресурс] / Режим доступа: WWW. URL: <http://www.interface.ru/sysmod/arena.htm>
24. NetLogo [Электронный ресурс] / Режим доступа: WWW. URL: <https://ccl.northwestern.edu/netlogo/>
25. NetLogo [Электронный ресурс] / Режим доступа: WWW. URL: <https://ru.wikipedia.org/wiki/NetLogo>
26. Fundamentals of modeling discrete systems. T. I. Aliyev, St. Petersburg-2009.
27. M. Guizani, A. Rayes Network Modeling and Simulation.- John Wiley & Sons Ltd, 2010.
28. Michel C. Jeruchim Simulation of Communication Systems. – Neju York, Kluwer Academic Publishers, 2002.
29. K. Wehrle, M. Gunes Modeling and Tools for Network Simulation.- Springer-Verlog Berlin Heidelberg, 2010.
30. Kamilov M. M. Ergashev A. K. Mathematical modeling.- Tashkent, TUIT, 2007.- 176 p
31. Jan Beran. Statistics for long-memory processes, chapter 12. Monographs on statistics and applied probability, No. 61. Chapman & Hall, New York, NY, USA, 1994.
32. Brian D. Bunday, An Introduction to Queueing Theory- 1996.