

4. Glasserman P. Monte Carlo Methods in Financial Engineering. Springer, 2003. 596 p. URL: https://www.bauer.uh.edu/spirrong/Monte_Carlo_Methods_In_Financial_Enginee.pdf (дата звернення: 12.05.2025).

УДК 004.94:656.05 (043.2)

*Сапожнікова В. Є., здобувачка вищої освіти
3 курсу спеціальності 122 Комп'ютерні науки,
Ніколюк П. К., д-р фіз.-мат. наук, професор
кафедри інформаційних технологій*

ІМІТАЦІЙНА МОДЕЛЬ ІНТЕЛЕКТУАЛЬНОГО СВІТЛОФОРА З УРАХУВАННЯМ ТРАНСПОРТНОГО НАВАНТАЖЕННЯ В СЕРЕДОВИЩІ ANYLOGIC

Донецький національний університет імені Василя Стуса, м. Вінниця

У сучасному світі ефективне управління транспортними потоками є однією з ключових проблем, що безпосередньо впливає на мобільність, безпеку та екологічний стан міст. В Україні спостерігається постійне зростання кількості автомобілів, що спричиняє перевантаження на перехрестях, затори та зниження якості пересування містом. У цих умовах особливої актуальності набуває цифровізація транспортної інфраструктури, зокрема впровадження розумних світлофорних систем, здатних динамічно реагувати на поточну дорожню ситуацію.

Для перевірки ефективності таких підходів доцільно використовувати імітаційне моделювання, що дає змогу дослідити поведінку системи до її реального впровадження. Середовище AnyLogic надає інструменти для створення цифрових копій транспортних об'єктів, у тому числі світлофорів, зі врахуванням логіки перемикання фаз, структури дорожньої мережі та змінного потоку транспорту.

У сфері інтелектуального управління дорожнім рухом «розумні» світлофори розглядаються як ефективний засіб зниження заторів у міському середовищі. Головна ідея таких систем полягає в динамічній зміні тривалості фаз світлофора залежно від реального стану трафіка. Це дає змогу скорочувати середній час очікування на перехрестях та підвищувати пропускну здатність вузлів транспортної мережі [1]. У моделюванні транспортних процесів подібні підходи демонструють високу ефективність у сценаріях зі змінною інтенсивністю руху, зокрема у години пік.

Суттєвим компонентом таких систем є механізм виявлення кількості транспортних засобів, які наближаються до перехрестя. У моделюванні це реалізується за допомогою логічних сенсорів або спеціальних блоків, які виконують роль точок детекції. Під час перетину автомобілем цієї зони відбувається інкремент змінної-лічильника (наприклад, `queueWest++`), яка відображає довжину черги. Коли автомобіль перетинає стоп-лінію – значення зменшується. На основі поточного значення черги система приймає рішення: або залишити зелений сигнал активним, або перейти до наступної фази [2].

AnyLogic чудово справляється з моделюванням широкого спектра транспортних сценаріїв – від простих маршрутів до складних міських перехресть. Завдяки

гнучкості середовища можна вручну створювати логіку світлофорів, визначати правила руху агентів, керувати чергами автомобілів та змінювати сигнали на основі поточних умов. Такий підхід дає змогу точно контролювати поведінку моделі та випробовувати різні алгоритми оптимізації руху без використання шаблонних компонентів [3].

Перед початком розробки логіки адаптивного світлофора необхідно змодельовувати перехрестя. Для цього у середовищі AnyLogic було використано бібліотеку Road Traffic Library, яка дає змогу створювати дорожню інфраструктуру з точним контролем руху, смугами, напрямками і логікою перетинів. За допомогою елементів цієї бібліотеки створено хрестоподібне перехрестя, яке включає чотири підходи – з півночі, півдня, заходу та сходу.

Кожна дорога містить сегмент руху, блок генерації машин (CarSource), точку завершення (CarDispose) і стоп-лінію, яка допомагає регулювати проїзд світлофором. Також визначено можливі траєкторії руху – прямо, ліворуч, праворуч – через конфігурацію стрілок і контурних ліній у центрі перехрестя. Створена конфігурація дає змогу перевіряти сценарії руху з різних напрямків та оцінювати ефективність світлофорного регулювання.

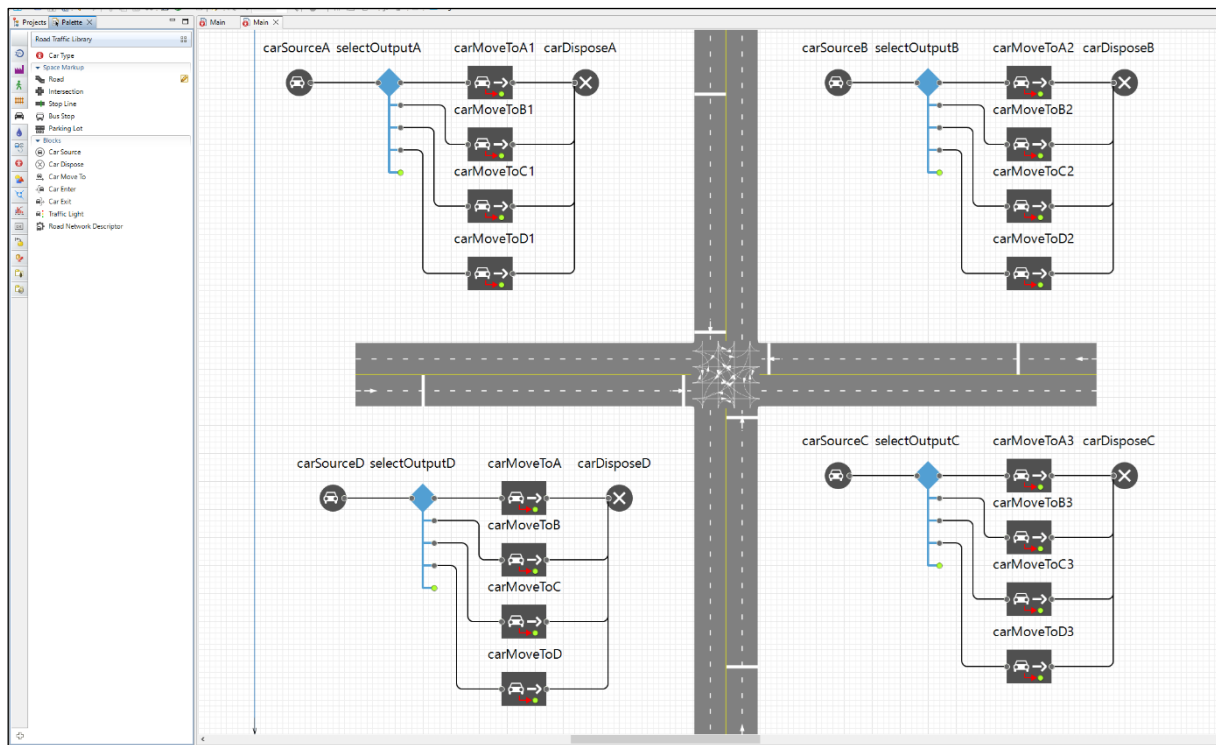


Рис. 1. Модель інтелектуального перехрестя в AnyLogic

Тепер доцільно приступити до регулювання руху. Загальна ідея полягає в імітації адаптивного, «розумного» світлофора, що змінює фази залежно від довжини черги автомобілів на кожному напрямку (захід, північ, схід, південь). Кожен напрямок має власний лічильник (queueWest, queueNorth тощо), який збільшується, коли машина в'їжджає в зону детекції, і зменшується після перетину перехрестя. Усі фази керуються подією (Event), яка циклічно перевіряє черги й перемикає фазу, якщо черга поточної фази дорівнює нулю. На кожному кроці модель активує одну з чотирьох фаз згідно зі змінною phaseIndex (0–3). Для активного напрямку встановлюється зелений сигнал, для решти – червоний. Після кож-

ного переключення `phaseIndex` змінюється на наступне значення, забезпечуючи ротацію між напрямками. Перед перемиканням фази подія оцінює довжину черг та визначає, наскільки довго має тривати зелений сигнал. Якщо максимальна черга ≥ 5 – 0 фаза триває 30 секунд, якщо ≤ 3 – 15 секунд, інакше – стандартні 20 секунд. Після цього подія перезапускається з оновленим інтервалом. Отже, світлофор динамічно реагує на ситуацію на дорозі без потреби у фіксованих таймерах.

```
// Обчислюємо тривалість фази залежно від найбільшої черги
int maxQueue = Math.max(Math.max(queueWest, queueNorth), Math.max(queueEast, queueSouth));
double phaseDuration;
if (maxQueue >= 5) { phaseDuration = 30; } // дуже довга черга - продовжуємо зелене
else if (maxQueue <= 3) { phaseDuration = 15; } // слабкий трафік - скорочуємо зелене
else { phaseDuration = 20; } // середній рівень - стандартна тривалість

switch (phaseIndex) { // Установка сигналів згідно з поточним індексом фази
case 0: // phaseIndex: 0=West
    stopLineWest.setSignal(SIGNAL_GREEN);
    stopLineNorth.setSignal(SIGNAL_RED);
    stopLineEast.setSignal(SIGNAL_RED);
    stopLineSouth.setSignal(SIGNAL_RED);
    break;
case 1: // phaseIndex: 1=North
    stopLineWest.setSignal(SIGNAL_RED);
    stopLineNorth.setSignal(SIGNAL_GREEN);
    stopLineEast.setSignal(SIGNAL_RED);
    stopLineSouth.setSignal(SIGNAL_RED);
    break;
case 2: // phaseIndex: 2=East
    stopLineWest.setSignal(SIGNAL_RED);
    stopLineNorth.setSignal(SIGNAL_RED);
    stopLineEast.setSignal(SIGNAL_GREEN);
    stopLineSouth.setSignal(SIGNAL_RED);
    break;
case 3: // phaseIndex: 3=South
    stopLineWest.setSignal(SIGNAL_RED);
    stopLineNorth.setSignal(SIGNAL_RED);
    stopLineEast.setSignal(SIGNAL_RED);
    stopLineSouth.setSignal(SIGNAL_GREEN);
    break;
}

phaseIndex = (phaseIndex + 1) % 4; // Готуємо індекс наступної фази
eventSmartTrafficLight.restart(phaseDuration, SECOND); // Рестарт з новим інтервалом
```

Рис. 2. Лістинг коду моделі «розумного» світлофора

Представлений фрагмент коду реалізує адаптивну логіку роботи світлофора на перехресті. Такий механізм дає змогу динамічно реагувати на зміну транспортного навантаження.

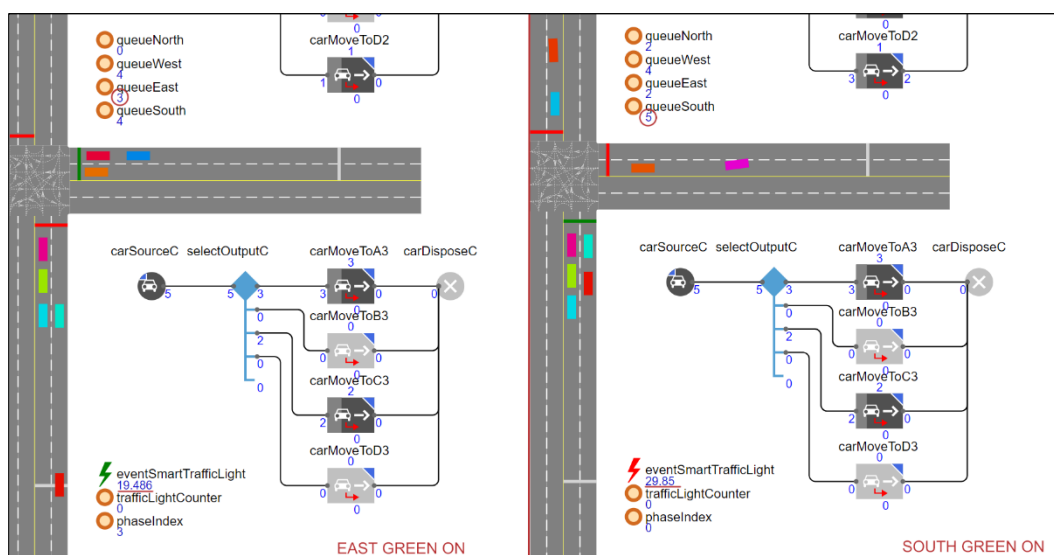


Рис. 3. Результат роботи «розумного» світлофора

Отже, реалізація адаптивного світлофора на основі довжини черги дасть змогу підвищити ефективність управління дорожнім рухом на перехресті. Це забезпечує динамічну реакцію на навантаження та зменшує затримки транспорту, а саме виконання динамічної зміни фаз не є проблемним.

Список використаних джерел

1. Comert G., Cetin M., Begashaw N. A Simple Traffic Signal Control Using Queue Length Information. *arXiv:2006.06337. Electrical Engineering and Systems Science*. 2020. URL: https://www.researchgate.net/publication/342120862_A_Simple_Traffic_Signal_Control_Using_Queue_Length_Information
2. Aljaafreh A. Adaptive Traffic-Signal Control using Discrete Event Simulation Model. Academia.edu, 2016. *International Journal of Computer Applications*. 2014. Vol. 101, № 12. P. 7–12. DOI: 10.5120/17737-8910. URL: <https://research.ijcaonline.org/volume101/number12/pxc3898910.pdf>
3. Liu Y., Song Y. Research on Simulation and Optimization of Road Traffic Flow Based on AnyLogic. *8th International Symposium on Vehicle Emission Supervision and Environment Protection (E3S Web of Conferences)*. 2022. Vol. 360. URL: https://www.researchgate.net/publication/365687632_Research_on_simulation_and_optimization_of_road_traffic_flow_based_on_Anylogic

УДК 004.414.23:336.71

*Тищенко О. М., здобувач вищої освіти 2 курсу спеціальності 122 Комп'ютерні науки,
Січко Т. В., канд. техн. наук, доцент, доцент
кафедри інформаційних технологій*

МОДЕЛЮВАННЯ ПРОЦЕСУ НАДАННЯ БАНКІВСЬКИХ ПОСЛУГ

Донецький національний університет імені Василя Стуса, м. Вінниця

У сучасних умовах цифрової трансформації банківська сфера потребує ефективних інструментів для моделювання, аналізу та оптимізації бізнес-процесів. BPWin дає змогу детально відображати та вдосконалювати процеси надання банківських послуг, що підвищує продуктивність, мінімізує ризики та покращує клієнтський досвід.

Метою роботи є дослідження використання методології IDEF0 для моделювання банківських процесів, оцінка ефективності її застосування у фінансових установах.

Автоматизація та оптимізація бізнес-процесів у банківській сфері є важливими складниками підвищення конкурентоспроможності та якості обслуговування клієнтів. Інструмент побудови блок-схем за методологією IDEF0 дає змогу чітко структурувати робочі процеси, ідентифікувати вузькі місця та впроваджувати вдосконалення. Завдяки цьому банки можуть ефективніше керувати операційними процесами, підвищувати їх прозорість і відповідність сучасним вимогам ринку.

Інформаційні технології бізнес-моделювання надають користувачам широкий набір функцій для аналізу та моделювання процесів, зокрема: