

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФРАСТРУКТУРИ ТА ТЕХНОЛОГІЙ
Кафедра «Технологій транспорту та управління
процесами перевезень»**

В.І. Мацюк

**ІННОВАЦІЙНІ ТЕХНОЛОГІЇ УПРАВЛІННЯ
ПЕРЕВІЗНИМ ПРОЦЕСОМ**

МЕТОДИЧНІ ВКАЗІВКИ

до виконання курсового проекту, практичних занять та магістерських робіт
денної та заочної форм навчання напряму підготовки галузі знань:
27 «Транспорт», напряму підготовки: 275.02 «Транспортні технології (на
залізничному транспорті)»

Київ 2020

УДК 656.2

В.І. Мацюк

Інноваційні технології управління перевізним процесом: до виконання курсової роботи та практичних занять. – К.: ДУІТ, 2020. – 55 с.

Методичні вказівки до виконання курсового проекту, практичних занять та магістерських робіт для денної та заочної форм навчання напряму підготовки галузі знань: 27 «Транспорт», напряму підготовки: 275.02 «Транспортні технології (на залізничному транспорті)»

Методичні вказівки розглянуті та затверджені на засіданні кафедри «Управління процесами перевезень» (протокол №__ від «__» _____2020 року) та на засіданні методичної комісії Інституту залізничного транспорту (протокол №__ від «__» _____2020 року).

Укладач: Мацюк В.І., професор кафедри технологій транспорту та управління процесами перевезень, д.т.н., професор.

Рецензенти:

Зміст

Введення	4
1 Вибір індивідуального завдання	5
1.1 Індивідуальне завдання на курсовий проект	5
1.2 Варіанти індивідуального завдання.....	6
2 Розробка імітаційної моделі у середовищі AnyLogic	8
2.1 Загальні відомості про імітаційне моделювання транспортних процесів та систем	8
2.2 Огляд основних агрегатів <i>Process Modeling Library</i> середовища AnyLogic 8.5	10
2.3 Встановлення програми AnyLogic 8.5 та отримання тимчасової ліцензії	12
2.4 Огляд програмних можливостей AnyLogic при розробці імітаційних моделей залізничних транспортних процесів дискретно-подієвим способом	15
2.5 Розробка імітаційної моделі функціонування технологічної лінії з обробки транзитних вантажних поїздів дискретно-подієвим способом в середовищі AnyLogic.....	18
2.6 Збір статистичних даних моделі та їх аналіз	22
3 Експериментальне встановлення мінімально-можливої кількості приймально-відправних колій та оглядачів поїздів	25
4 Експериментальне встановлення потрібної граничної зміни обраного параметру при якому наявна імовірність технологічної відмови не буде меншою за потрібну	27
5 Структура та зміст пояснювальної записки	30
6 Практичні заняття	32
Список рекомендованої літератури	33

Введення

Управління ланцюгами поставок (УЛП, англ. SCM - Supply Chain Management) - це організація, контроль і виконання товарного потоку, від проектування і закупівлі через виробництво та розподіл до кінцевого споживача, відповідно до потреб ринку.

Очевидно, що рух (виконання) транспортного потоку, як одна з ключових складових УЛП, неможливий без транспорту. Тим більше неможливе без вдосконалення транспортних технологій кожного виду транспорту і взаємодії між ними ефективність управління ланцюгами постачань товарів у міжнародних торговельно-економічних і транспортних зв'язках, які все більше набувають глобальний і мультимодальний характер.

Мета дисципліни «Інноваційні технології управління перевізним процесом» є формування системних знань і практичних умінь щодо розробки й обґрунтування технологічних процесів пасажирських і вантажних перевезень, їх організації й управління в інтегрованих транспортних системах.

Завдання: розкриття сучасних наукових концепцій, понять, методів та технологій, направлених на формування у спеціалістів даного напрямку вмінь і навичок практичного використання принципів побудови і функціонування інтегрованих транспортних систем для прийняття управлінських та проектних рішень.

У результаті вивчення дисципліни студент повинен знати:

- принципи та методи розробки й обґрунтування технологічних процесів пасажирських і вантажних перевезень, їх організації й управління в інтегрованих транспортних системах.

вміти:

- використовувати методи розробки й обґрунтування технологічних процесів пасажирських і вантажних перевезень, їх організації й управління в інтегрованих транспортних системах.

Для успішного оволодіння цими навичками автори рекомендують слідувати даним методичним вказівкам, використовуючи наведені нормативно-довідкові документи додатків та рекомендовану літературу списку використаних джерел.

1 Вибір індивідуального завдання

Індивідуальні дані вибираються з таблиць відповідно до порядкового номеру у журналі академічної групи з таблиць 1.1 та 1.2.

Наприклад, для порядкового номеру **06** (остання цифра **6**, передостання **0**). Для цього варіанту нижче заповнено приклад завдання.

1.1 Індивідуальне завдання на курсовий проект

Міністерство освіти і науки України

Державний університет інфраструктури та технологій

Кафедра технологій транспорту та управління процесами перевезень

Завдання на курсовий проект з дисципліни

«Інноваційні технології управління перевізним процесом»

студент (ка) _____ гр. 1-ТТ__ (маг.).

1. Вихідні умови.

1.1 Характеристики процесу надходження поїздів до станції:

- a) середньодобовий потік – **120** поїздів на добу;
- b) функція розподілу інтервалів прибуття поїздів: експонентна;
- c) мінімальний (фактичний) інтервал попутного прибуття та відправлення поїздів: $7,5 \div 10$ із середнім 8 хв.;
- d) середній час займання горловини поїздом: $4 \div 7$ із середнім 5 хв.;
- e) середній час очікування поїздом відправлення: **5,0** хв.;
- f) функція розподілу часу очікування поїздом відправлення: експонента.

1.2 Характеристики процесу обробки поїздів на станції:

- a) середній час обробки поїзда оглядачами, хв.: **30** хв.
- b) функція розподілу часу обслуговування поїздів: експонентна.

1.3 Потрібні умови ефективності функціонування залізничної станції:

- a) гранична імовірність технологічної відмови $\xi_{п} \leq \underline{30}$.
- b) змінний параметр: середньодобова кількість поїздів.

2. Встановити:

2.1 Експериментально встановити мінімально-можливу потрібну кількість приймально-відправних колій та оглядачів поїздів, при яких будуть виконуватись умови:

- середньодобове завантаження оглядачів повинне бути не меншим за 0,5;
- рівень технологічної відмови не повинен перевищувати 15%;
- час знаходження поїздів, з врахуванням затримок, повинен бути мінімальним.

2.2 Експериментально встановити потрібну граничну зміну обраного параметру (*n. 1.3, b* завдання) при якому наявна імовірність технологічної відмови не буде меншою за потрібну (*n. 1.3, a* завдання).

Керівник проекту,

д.т.н., професор

В.І. Мацюк

1.2 Варіанти індивідуального завдання

Вибір індивідуальних даних завдання здійснюється відповідно до останньої та передостанньої цифр встановленого варіанту.

Таблиця 1.1 – Вибір вихідних даних за останньою цифрою варіанту

Параметр	Остання цифра варіанту									
	0	1	2	3	4	5	6	7	8	9
Середньодобовий потік поїздів на добу	60	70	80	90	100	110	120	130	140	150

Таблиця 1.2 – Вибір вихідних даних за передостанньою цифрою варіанту

Параметр	Передостання цифра варіанту		
	0	1	2
Гранична імовірність технологічної відмови $\xi_{п}$	30	40	50
Середній час обробки поїзда оглядачами	30	35	40
Середній час очікування поїздом відправлення	5,0	2,5	7,5
Змінний параметр	Середньодобова кількість поїздів	Середній час обробки поїзда оглядачами	Кількість приймально-відправних колій

2 Розробка імітаційної моделі у середовищі AnyLogic

2.1 Загальні відомості про імітаційне моделювання транспортних процесів та систем

Теорія основ математичного та комп'ютерного моделювання передбачає змістовне та формальне визначення категорій з метою побудови математичних моделей складних систем.

Основними методологічними категоріями теоретичних основ моделювання є поняття: об'єкт, клас, відношення (зв'язок), система, елемент, структура.

Визначення поняття об'єкта має різне тлумачення залежно від області науки та сфери діяльності. Якщо ми розглядаємо об'єкт імітаційного моделювання, то в стратегії парадигмі об'єктно-орієнтовного підходу об'єкт є першим важливим поняттям. Об'єкт – це деяка сутність у віртуальному просторі, що має певний набір станів і поведінок, має задані значення властивостей (атрибутів) та операцій над ними.

Наступним важливим поняттям об'єктно-орієнтовного підходу є клас. Схожі за певними характеристиками поведінці об'єктів об'єднуються в класи. В залежності від характеристик, одні й ті ж об'єкти можуть бути в різних класах.

В одній з розділів сучасної математики «теорії категорій» об'єкт використовується як термін для позначення елементів будь-якої категорії, що грають роль множин, груп, топологічних просторів і т. Д. п. Тут же вводиться поняття класу об'єктів і проводиться вивчення властивостей відносин між математичними об'єктами, що не залежать від внутрішньої структури об'єктів.

Поняття відносин визначає взаємне розташування об'єктів, зв'язок між об'єктами у вигляді асоціативних, алгоритмічних, табличних та інших структур.

Поняття система є базовим в теорії математичного моделювання. Існує кілька десятків різних визначень поняття «система», які використовуються в залежності від контексту, області знань і цілей дослідження. Вивченням систем займаються в рамках наукових дисциплін таких як системологія, кібернетика, системний аналіз, теорія систем, системна динаміка теорія транспортних процесів та систем та інші.

Система - це:

- 1) ціле, створене з частин і елементів цілеспрямованої діяльності, що отримує нові властивості, відсутні в елементах і частинах, що її утворюють;
- 2) об'єктивна частина світогляду, що включає схожі і сумісні елементи, що утворюють цілу, яка взаємодіє з зовнішнім середовищем;
- 3) об'єктивна єдність закономірно пов'язаних один з одним предметів, явлень, зведень, а також знань про природу, суспільство і т. д.

Допустимі багато інших визначень. Загальним в них є те, що система є деякою правильною комбінацією найважливіших, істотних властивостей досліджуваного об'єкта. Кожний об'єкт, для можливості розглядання його як системи, повинен мати чотири основні властивості або признаки (цілісність и подільність, наявність стійких зв'язків, організації та емерджентності).

Елемент – це найпростіша, неділима частина системи, а її властивості визначаються конкретним завданням. Елемент завжди пов'язаний з самою системою. Елемент складної системи може бути в свою чергу складною системою в іншому завданні.

Підсистема - компонент системи - об'єднання елементів, але по масштабу менше, ніж система в цілому.

Система може включати великий перелік елементів і її доцільно розділити на ряд підсистем.

Знаковими ознаками системи є безліч складових її елементів, єдність головної цілі для всіх елементів, наявність зв'язків між ними, цілісність та об'єднання елементів, наявність структури та ієрархічності, відносна самостійність та можливість управління цими елементами. Термін

«організація» в одній зі своїх лексичних значень означає також «систему», але не будь-яку систему, а в певному ступені упорядковану, організовану.

Поняття "підсистема" розроблено для аналізу складно-організованих систем, коли між елементами та системою є "проміжні" комплекси, більш складні, ніж елементи, але менш складні, ніж сама система. Вони об'єднують в собі різні частини (елементи) системи, у своїй сукупності здатні до виконання єдиної (приватної) програми системи. Будучи елементом системи, підсистема в свою чергу є системою по відношенню до елементів, її компонентів. Аналогічна обставина стосується відносин між поняттями "система" та "елемент": вони переходять одне в інше. Інакше кажучи, система і елемент ідентичні. З цієї точки зору вся матерія представляється як безкінечна система систем. "Системи" можуть бути системами відносин, детермінацій і т.п.

На додачу до представлення елементів в уяві про будь-якій системі входить і представлення про її структуру.

Структура - це сукупність стійких відносин і зв'язків між елементами. Сюди включається загальна організація елементів, їх просторове розташування, зв'язок між етапами розвитку і т.п.

2.2 Огляд основних агрегатів *Process Modeling Library* середовища AnyLogic 8.5

На сьогодні за допомогою AnyLogic стало можливим розробляти моделі в наступних областях:

- виробництво;
- логістика та ланцюги поставок;
- функціонування транспортних процесів та систем;
- оцінка та управління ризиками;
- ринок і конкуренція;
- бізнес-процеси та сфери обслуговування;

- охорона здоров'я та фармацевтика;
- управління активами та проектами;
- телекомунікації та інформаційні системи;
- соціальні та екологічні системи;
- пішохідна динаміка;
- оборона, безпека, надійність.

Версія AnyLogic 8.5 написана мовою програмування *Java* в популярному середовищі розробки *Eclipse* і є крос-платформним програмним забезпеченням, працює як під управлінням операційної системи Windows, так і під Mac OS і Linux.

AnyLogic включає в себе графічний режим моделювання, а також дозволяє користувачеві розширювати створені моделі за допомогою мови Java. Інтеграція Java-компілятора в AnyLogic забезпечує більш широкі можливості при створенні моделей, а також створення Java-апплетів, які можуть бути відкриті будь-яким браузером. Ці апплети дозволяють легко розміщувати моделі AnyLogic на веб-сайтах. В додаток до Java-апплетам, AnyLogic Professional підтримує створення Java-додатків, в цьому випадку користувач може запустити модель без встановлення AnyLogic.

Система AnyLogic – це комплексний інструмент, що охоплює основні напрямки моделювання: дискретні-події, системну динаміку, агентне та ін.

Використання AnyLogic дає можливість оцінити ефект конструктивних рішень у складних системах реального світу.

Інструмент імітаційного моделювання AnyLogic нового покоління розроблений на основі сучасних концепцій в області інформаційних технологій та результатів досліджень в галузі теорії гібридних систем та об'єктно-орієнтованого моделювання. Побудована на їх основі інструментальна система AnyLogic не обмежує користувача єдиною парадигмою моделювання, що характерна для існуючих на ринку інструментів моделювання. В AnyLogic розробник може гнучко використовувати різні рівні

абстрагування та різноманітні стилі і концепції та змішувати їх при створенні однієї і тієї ж моделі.

Програмний продукт AnyLogic побудовано на основі об'єктно-орієнтованої концепції. Об'єктивно-орієнтований підхід до представлення складних систем є найкращим на сьогоднішній день методом управління складністю інформації, ця концепція дозволяє просто і природним чином організувати і представити структуру складної системи. Таким чином, ідеї та методи, спрямовані на управління складністю, розроблені в останні десятиліття в області створення програмних систем, дозволяють розробникам моделей в середовищі AnyLogic організувати мислення, структурувати розробку і, в кінцевому рахунку, спростити і прискорити створення моделей.

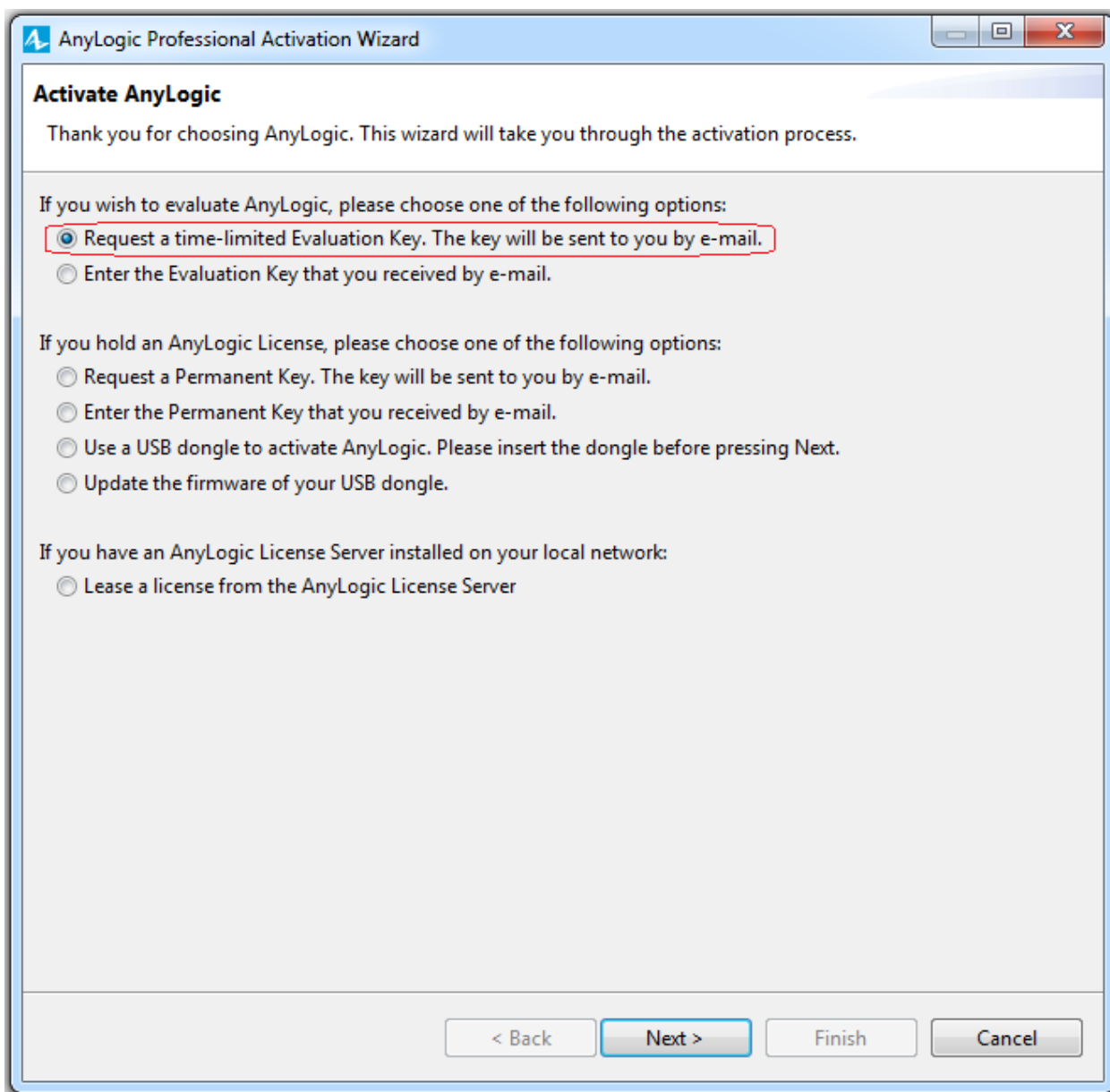
Іншою базовою концепцією AnyLogic є представлення моделі як набору взаємодіючих, паралельно функціонуючих активностей. Такий підхід до моделювання інтуїтивно дуже зрозумілий і природний в багатьох додатках, оскільки системи реального життя складаються з сукупності активностей, взаємодіючих з іншими об'єктами. Активний об'єкт AnyLogic – це об'єкт з власним функціонуванням, взаємодіючих з оточенням. Він може включати в себе будь-яку кількість екземплярів інших активних об'єктів. Активні об'єкти можуть динамічно породжуватися і зникати відповідно до законів функціонування системи. Так можуть моделюватися соціальні групи, холдинги компаній, транспортні системи і т. д.

2.3 Встановлення програми AnyLogic 8.5 та отримання тимчасової ліцензії

Версія для самостійного навчання AnyLogic Personal Learning Edition (PLE) встановлюється без активації. Найбільш же функціонально повна версія AnyLogic Professional передбачає проходження нескладної операції активації. Завантажте AnyLogic з адреси <https://www.anylogic.com/downloads/>, потім встановіть AnyLogic і дотримуйтесь цієї інструкції:

1. Запустіть AnyLogic. Якщо програма ще не була активована, з'явиться вікно *Майстра активації AnyLogic*.

2. На сторінці майстра *Активувати AnyLogic* виберіть опцію *Запросити обмежений за часом ознайомлювальний ключ*. Після заповнення нескладної форми, на вказану електронну адресу буде надіслано тимчасовий ключ для ознайомлення.



AnyLogic Professional Activation Wizard

AnyLogic License Request

Please complete your personal data and provide a valid e-mail address. You will receive your unlock key by e-mail.

*First Name:

Middle Name:

*Last Name:

*Company:

Department:

Web Site:

*Country:

*E-mail:

I want to receive monthly AnyLogic newsletter

Please specify a valid email address. The activation key will be sent to the email entered here.

*Phone:

*In what area will you use AnyLogic?

*How did you hear about AnyLogic?

Your license request will be sent to the AnyLogic Activation server.

[Proxy settings...](#)

< Back Next > Finish Cancel

AnyLogic Professional Activation Wizard

Enter the unlock key

Enter the evaluation key you received by e-mail

Please paste the key here:

```
DvI+uNyxJKaeGaKBgc4uQ1PzmtL/LaIwGv32fR0o
02yS620+kvT0KBV9MG8/Qt6h3RCOfCX4ZF7CY2/0
gvFq5reBKmT2AptPd6zLuVRiQ8YtpoXfb5P3xfsk
jTzYSOkEOK2bTHOVh+/IR+B2tLETcJjyER+VsvG/
UHh1X239sZTmiFVBw7ITv1By2mnyUZG+VEoJzv1g
oZdjHvaBgdA2CTXqEnDVIxrgThA1n9Dur0c8J66x
7wX/9yXAMaJzWnU4E9zrpx9uGWVhhterScXHVfpe
M9SbW5Kq4zo3u2PnS3tAtovpF9ji0dD6DrUF00Eo
Ey4wINI8qgI7wWgfpjLZqPkovSf9ktcGqYtK3WM
+msjwV3uDbakxvZj/JTj+bQeZRPPUoiQvB3/CLqX
LjWddUw1c1qsAfqOXhEtD3aEudxji3oMlWDmova4
gbNquhkpBuYwt8KGBDIJUNg10jYAUqqyDIYApPVC
bNB6vGM9pPwKwjuJ7AggM1XGktqfSrQ3XIv882yz
jDY/UvMuIueKL1ScMFwVLA5Zi4uNwgtCNKEmaSsq
5emE75FINvoip20sQw4xyq6cL1loFTqISqaTTjba
d+qLfbtvT/qT1+EIleqLs4wtq+akZ/Ea5BFw3xnE
k1X26JreJQZLh+F8c7989GKIfoa3A5/STskoyCim
53K6bu6Eg4cHj22Mb6z1TY4jcJDSidDdRIKS104S
CwmBXuKqMdyBEbsy4tfXsXyKD1zHHe0gfv1bg0E8
Ip65pJgGAF9hUTq6wYeMC2oQLfx0gCScAB84MSNF
hTLu5cnlo/4wIR4jL67vveDynpGkAbXhkoTuoK3l
EbRv3avyf736VoHF2kHrOKRz7YqZbouNnf00nqn
xi//scgfoUw1FhjvCDV2eYDsYCDricGe/KQ1Wih
fajwvkiIwukhBu5f3YgfpoLSjE780dIrCqTVP32W
FHDYSoCbxoAPaWewogj+emzWv34tQ06pctQDE686
Fwkz3F03rvHngT2Md10gGnhqtH14RC6fAkr+z0Gd
W4qAQAPd0Wk+N+gMoOXN/E=%$#
```

[Paste from clipboard](#)

< Back Next > Finish Cancel

2.4 Огляд програмних можливостей AnyLogic при розробці імітаційних моделей залізничних транспортних процесів дискретно-подієвим способом

Для дослідження функціонування технологічних ліній сортувальних та дільничних станцій найбільш зручним та ефективним є дискретно-подієвий спосіб, при якому процес імітації обробки поїздів буде являти поступовий перехід заявки через відповідні програмні блоки. Затримання заявок у блоках здійснюється на час, що відповідає тривалості технологічних операцій. Такий принцип дозволяє програмно фіксувати момент входу заявки у блок програми, що відповідатиме початку виконання технологічної операції, і момент виходу заявки з блоку, що є завершенням технологічної операції.

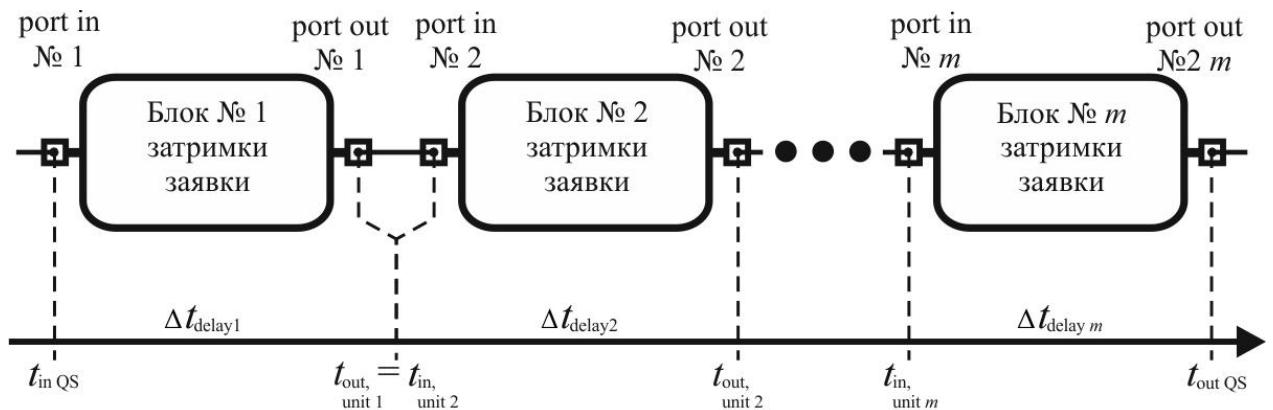


Рисунок 2.1 – Принцип дискретно-подієвого моделювання технологічного процесу ЗТС при послідовному виконанні операцій із поїздом

Інструментом розробки імітаційної моделі технологічної лінії із обробки транзитних поїздів сортувальних станцій обрано об'єктно-орієнтовну мову програмування *Java SE (Oracle)*, середовище розробки *Eclips* та *AnyLogic*. Для конструювання моделі, використовувалась наступні агрегати основних бібліотек *AnyLogic 8.5*, відповідно до специфікації [1] (табл. 2.1):

Таблиця 2.1 – Основні агрегати бібліотеки *Process Modeling Library* середі розробки *AnyLogic 8.5*, що використовуються при моделюванні

№ п.п	Вигляд агрегату у середовищі розробки	Функція та умови використання
1.		блок типу <i>Source</i> – генератор заявок. Генерує вхідний потік заявок за встановленими параметрами
2.		блок типу <i>Queue</i> – черга. Імітує очікування заявок у черзі. Заявки зорганізуються за принципом <i>FIFO</i> (first in – first out) – обслуговується у першу чергу та заявка, яка знаходиться у черзі найдовше
3.		блок типу <i>Delay</i> – блок затримки. Імітує затримку часу при виконанні операцій та обробці заявок
4.		блок типу <i>Service</i> – блок обслуговування. Імітує обслуговування заявки із застосуванням (захватом та використанням) певного ресурсу
5.		блок типу <i>ResourcePool</i> – ресурс. Імітує обсяг певного ресурсу, що використовується при імітуванні обслуговування заявки блоком <i>Service</i> .
6.		блок типу <i>Sink</i> . Імітує закінчення процесу
7.		блоки типу <i>restrictedAreaStart</i> та <i>restrictedAreaEnd</i> . Зазначенні елементи обмежують одночасне знаходження агентів відповідно до встановленої кількості
8.		елемент типу <i>Parameter</i> . Застосовується для задавання вихідних даних
9.		елемент типу <i>Variable</i> . Застосовується при задаванні логіки процесів, зборі результатів моделювання
10.		елемент типу <i>Histogram Data</i> . Застосовується для збору та обробці статистичних даних у вигляді інтервального ряду розподілу

Об'єктно-орієнтовне, на відміну від функціонального, програмування дозволяє створювати складні програмні продукти із розгалуженою архітектурою, використовуючи набагато менші ресурси. Сама об'єктно-орієнтовна парадигма передбачає створення класів, відповідно до яких при реалізації програми створюються об'єкти класів. Класи – це шаблони мікропрограм, що можуть містити як різні типи даних, так функції та процедури їх обробки. У процесі розробки складних програмних продуктів, до яких можна віднести імітаційні моделі технологічних процесів залізничних транспортних систем, при створенні однотипних блоків немає потреби розробляти (як при функціональному програмуванні) кожний блок окремо. Створюється необхідний клас, а при реалізації програми у віртуальній пам'яті створюються екземпляри класу стільки, скільки треба за установчими параметрами.

Приклад можна навести імітацію надходження заявки у залізничну транспортну систему, що за фізичним змістом відповідатиме, наприклад, прибуттю поїздів у обробку. Тільки один раз створюється клас «поїзд», який наділяється необхідними, загальними характеристиками всіх поїздів: номер, вага, категорія, час прибуття, час початку технічного огляду, час закінчення технічного огляду, час причеплення поїзного локомотива, час відправлення та інші необхідні параметри. При реалізації програми (імітаційної моделі) при генерації заявки буде створюватись екземпляр класу, який вже наділений вихідними параметрами, і ці параметри (за необхідності) можуть змінюватися у процесі моделювання. Наприклад, номер поїзда присвоюється при генерації заявки, а моменти надходження заявки у відповідні блоки присвоюються під час моделювання.

2.5 Розробка імітаційної моделі функціонування технологічної лінії з обробки транзитних вантажних поїздів дискретно-подієвим способом в середовищі AnyLogic

Архітектура моделі розробляється відповідно до «фізичного» процесу приймання поїздів та їх обробки. Основною фізичною ознакою процесу є модельний час та просторові бінарні координати (рис. 2.2).

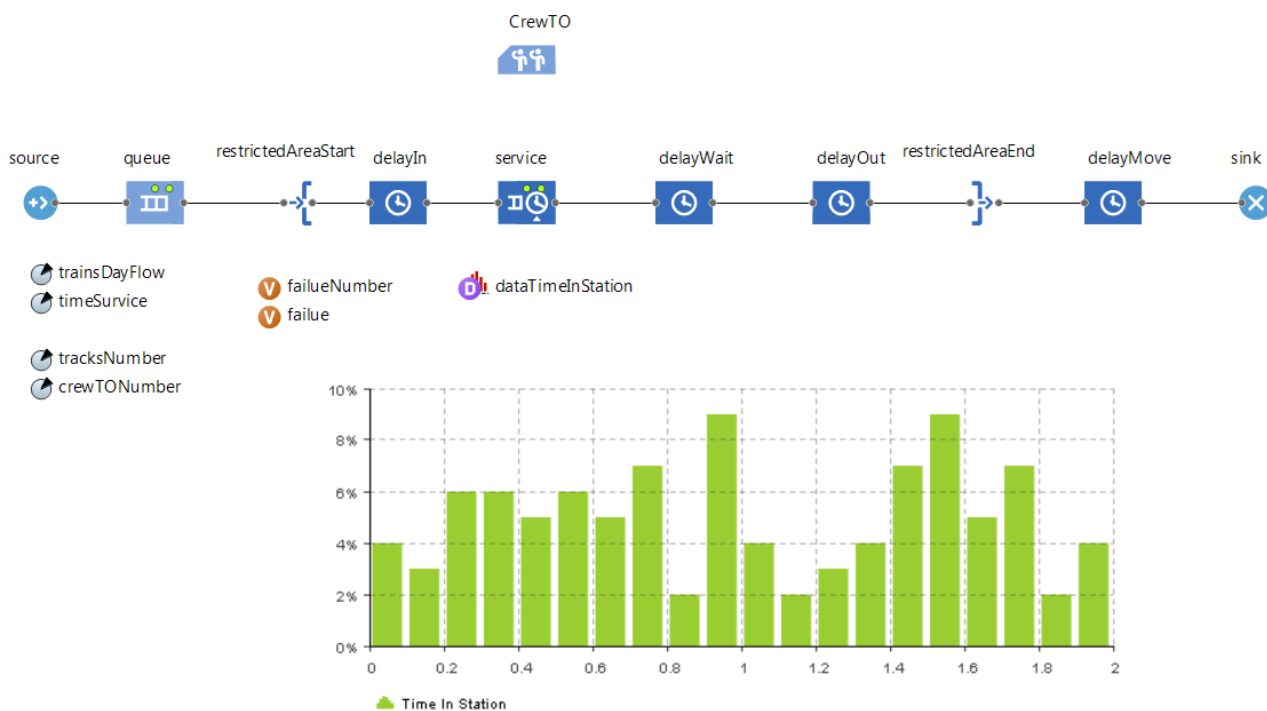


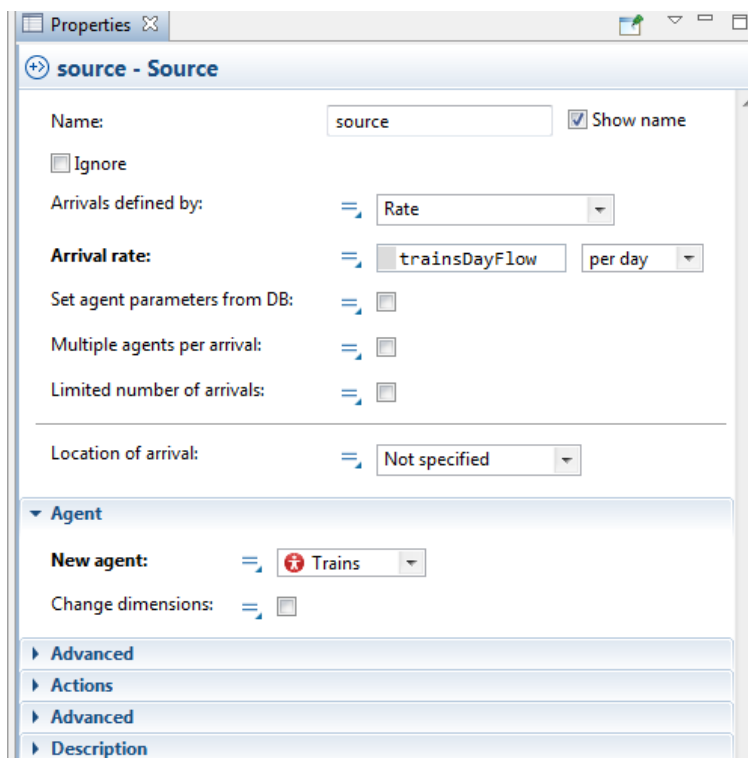
Рисунок 2.2 – Загальний вигляд імітаційної моделі функціонування приймально-відправного парку станції в середовищі розробки AnyLogic

Основні вихідні дані зазначаються у відповідних параметрах моделі (елементи типу *Parameter*):

- *trainsDayFlow* (тип *int*) – цілочисельний параметр, що визначає добовий поїздопотік;
- *timeService* (тип *double*) – параметр, що визначає середній час обробки поїздів оглядачами на коліях станції;
- *tracksNumber* (тип *int*) – цілочисельний параметр, що визначає розрахункову кількість приймально-відправних колій;

– *crewTONumber* (тип *int*) – цілочисельний параметр, що визначає розрахункову кількість груп оглядачів.

Блок *source* генерує заявки відповідно до встановлених особливостей процесу. У властивостях блоку *Arrivals defined by* треба зазначити *Rate*. Оскільки конкретне значення інтенсивності прибуття поїздів в моделі задається параметром *trainsDayFlow*, в графі *Arrival rate* вказується назва параметра *trainsDayFlow*, а період *per day*:

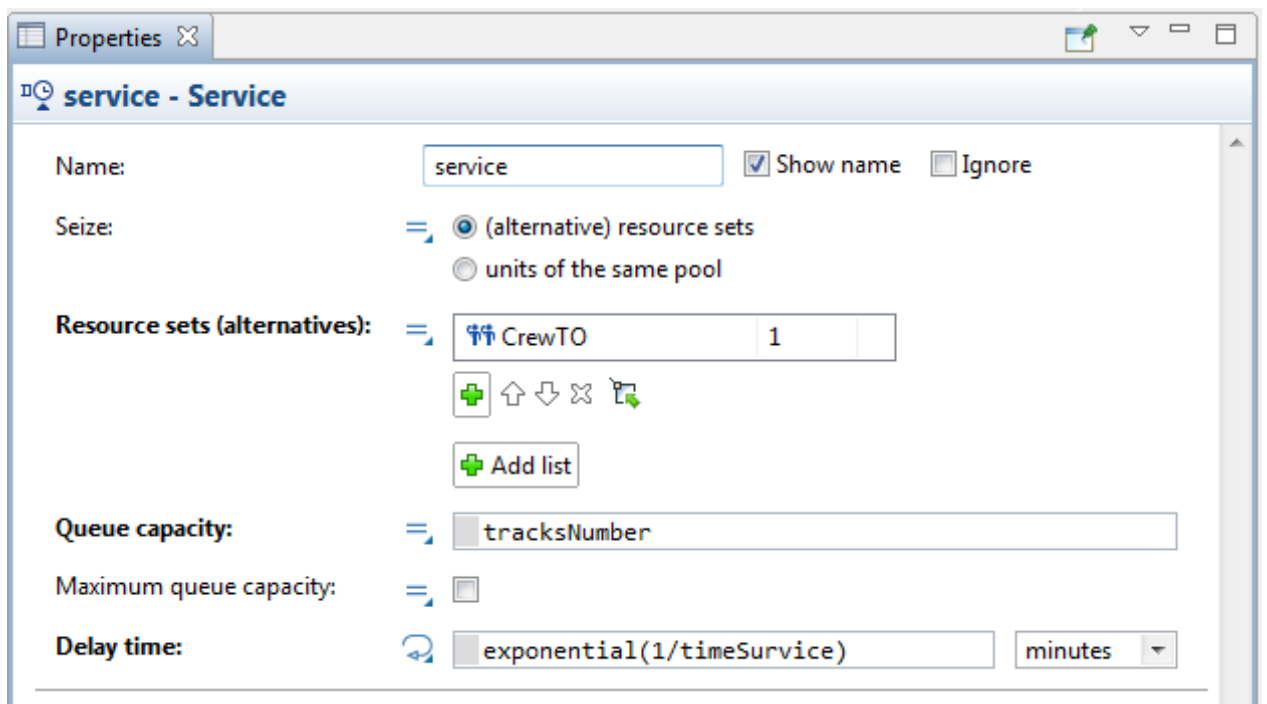


Далі у секції *Agent* створюється нова популяція агентів (графа *create a custom type*). У вікні створення нового агента задається назва *Trains*. У вікні *Step 2. Agent animation* вибирається пункт *None*, у вікні *Step 3. Agent parameters* створюються параметр типу *double*, із назвою *timeArriveInQueue*. Популяція нових створених агентів *Trains* буде імітувати проходження заявок (поїздів) по етапах технологічного процесу на станції. Створена змінна *timeArriveInQueue* необхідна для фіксації моменту модельного часу підходу поїзда до станції (*timeArriveInQueue*), із можливістю подальшого встановлення часу знаходження кожного окремого поїзда на станції.

При надходженні до вхідного світлофора станції, за умови вільності хоча б однієї приймально-відправної колії, поїзд приймається до станції. У випадках зайняття всіх колій, поїзд чекає приймання на перегоні. Так утворюються черги з поїздів. Формування черги та керування розміром черги здійснює блок типу *Queue* (табл. 2.1). Тому, одразу за блоком *source* слідує блок *queue*. У властивостях блоку треба активізувати опцію *Maximum capacity*. Таким чином блоки *source* та *queue* імітують надходження поїздів по перегону до вхідного світлофору станції.

Процес обробки поїздів на станції імітується блоками:

- 1) *delayIn* – займання поїздом вхідної горловини. У пункті властивостей *Delay time*: вказуються параметри часу. Відповідно до завдання *triangular(4, 5, 7)*. Одиниці виміру – хвилини. Місткість блоку (пункт *Capacity*:) – один, оскільки поїзди можуть прибувати з одного перегону тільки по черзі.
- 2) *service* – імітація процесу обробки поїздів на коліях парку. Обслуговування поїздів здійснюють групи оглядачів, що представлені у моделі у вигляді виробничого ресурсу – блок типу *ResourcePool* із назвою *CrewTO*. Розмір ресурсу в моделі задається параметром *crewTONumber*, тому у пункті *Capacity*:, властивості блоку *ResourcePool*, необхідно зазначити *crewTONumber*. В самому блоці *service*, у властивостях *Resource sets (alternatives)*: треба зазначити посилання на блок *CrewTO*. Місткість (графа *Queue capacity*:) відповідатиме кількості приймально-відправних колій, отже необхідно вказати параметр *tracksNumber*. Тривалість обробки поїзда задається завданням, отже в графі *Delay time*: зазначається Java-код *exponential(1/timeService)*, що вказує на експонентний розподіл параметра *timeService*. Одиниці виміру – хвилини (*minutes*).



3) *delayWait* – чекання поїзда відправки. Пункт *Delay time*: *exponential(1/5.0)*, де 5,0 – середній час затримки відповідно до завдання, хвилин. *Capacity*: – відповідатиме кількості приймально-відправних колій, а саме *tracksNumber*.

4) *delayOut* – займання поїздом вихідної горловини. Даний блок відповідає за своїми параметрами блоку *delayIn*.

Оскільки одночасно в парку станції не може знаходитись поїздів більше за кількість наявних приймально-відправних колій, блоки *delayIn service* *delayWait* та *delayOut* обмежуються блоками типу *restrictedAreaStart* та *restrictedAreaEnd* із зазначенням у графі *Capacity (max allowed)*: блоку *restrictedAreaStart* значенням *tracksNumber*.

Для імітації послідовного відправлення поїздів через інтервал попутного відправлення використовується блок *delayMove* із параметрами часу *triangular(7.5, 8, 10)* та місткості *Capacity 1*.

Завершенням процесу моделювання є блок *sink*.

2.6 Збір статистичних даних моделі та їх аналіз

Основними критеріями оцінки ефективності процесу, що моделюється, є середній час знаходження поїздів на станції, з урахуванням всіх затримок, та імовірність невчасного приймання поїздів – технологічної відмови.

Для визначення середнього часу знаходження поїздів на станції необхідно провести облік часу знаходження кожного поїзда. Для цього використовується елемент *dataTimeInStation* типу *data* та змінна агенту *Trains* – *timeArriveInQueue*:

- 1) у властивостях *Actions / On enter*: блока *queue* вписується Java-код:

```
agent.timeArriveInQueue = time();
```

- 2) у властивостях *On exit*: блока *delayOut* вписується інший Java-код:

```
dataTimeInStation.add(time() - agent.timeArriveInQueue);
```

- 3) для виводу результатів збору даних створюється графік типу *Histogram* з палітри *Analysis*. У властивостях *Data / Histogram*: графіка зазначається масив даних *dataTimeInStation*.

Фактична імовірність відмови визначається як:

$$\xi_{\text{п}} = \frac{N_{\text{відм}}}{N},$$

де $N_{\text{відм}}$ – кількість поїздів, що була прийнята із затримкою перед вхідним світлофором. Відповідно до діючих норм, затримкою вважається час, що перевищує 4 хв.

N - загальна кількість поїздів за звітній період.

Для встановлення кількості поїздів, що приймається із затримкою:

- 1) вводиться змінна *failueNumber* типу *int*; і змінна *failue* типу *double*;

2) у властивостях *Actions / On exit*: блоку *queue* вводиться Java-код:

```
if ((time() - agent.timeArriveInQueue) > 4.0){  
    failueNumber++;  
    failue = 100*failueNumber / (double)source.count();  
}
```

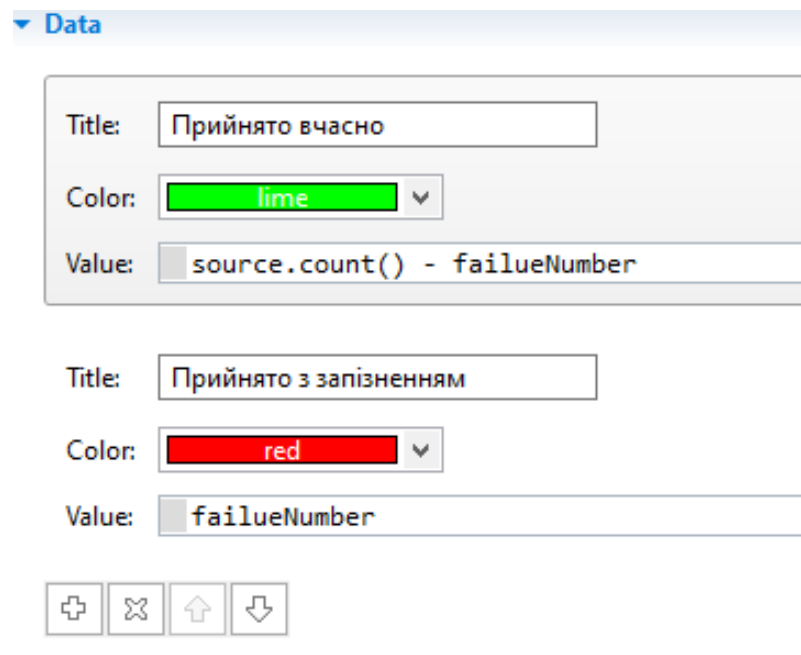
3) результати розрахунку вносяться до графіка типу *Stack Chart* з палітри *Analysis*. У властивостях *Data / Value*, двох груп даних, вводяться:

source.count() - failueNumber


та

failueNumber

відповідно:



На завершенні налаштування моделі задається модельний час один рік, від *Start data* до *Stop data*: вкладка *Projects*, агент *Simulation*, графа *Model time* властивості:

Properties 

Simulation - Simulation Experiment

Name: Ignore

Top-level agent:

Maximum available memory: Mb

Parameters

Model time

Execution mode: Virtual time (as fast as possible)
 Real time with scale

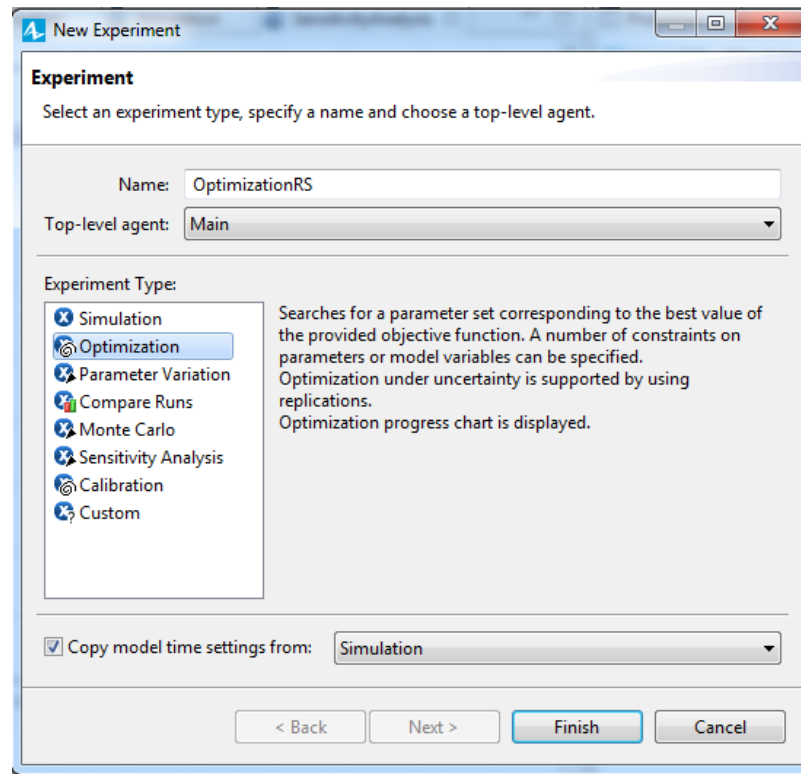
Stop:

Start time: Stop time:

Start date: Stop date:

3 Експериментальне встановлення мінімально-можливої кількості приймально-відправних колій та оглядачів поїздів

Для встановлення зазначених параметрів транспортно-технологічної системи доцільно скористатись інструментом *Experiment / Optimization*:



- 1) у властивостях *Objective*: вписується цільова функція *root.dataTimeInStation.mean()*, зазначається опція *minimize*;
- 2) у графі *Parameters* задаються діапазони зміни параметрів, що потрібно визначити.
- 3) у графі *Requirements* задаються обмеження, відповідно до завдання.
- 4) створюється інтерфейс вікна експерименту через натискання кнопки *Create default UI*.

Після запуску експерименту AnyLogic методом цілочисельного перебору знайде варіант, що буде відповідати заданим умовам.

Optimization - Optimization Experiment

Name: Ignore

Top-level agent:

Objective: minimize maximize

Number of iterations:

Automatic stop

Maximum available memory: Mb

Parameters

Parameters:

Parameter	Type	Value			
		Min	Max	Step	Suggested
tracksNumber	discrete	1	15	1	
trainsDayFlow	fixed	72			
timeService	fixed	30			
crewTONumber	discrete	1	10	1	

Model time

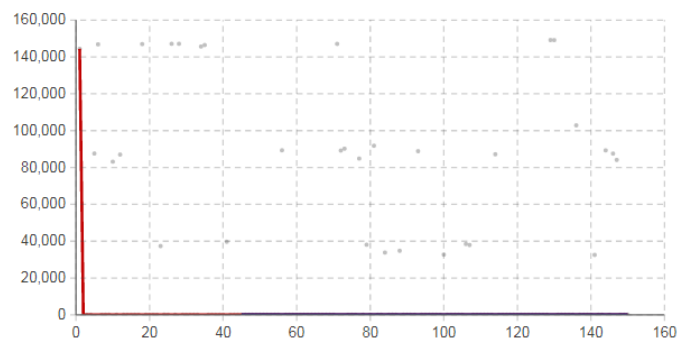
Constraints

Requirements

Requirements (are tested after a simulation run to determine whether the solution is feasible):

Enabled	Expression	Type	Bound
<input checked="" type="checkbox"/>	root.CrewTO.utilization()	>=	0.6
<input checked="" type="checkbox"/>	root.failue	<=	15.0

	Current	Best
Iterations completed:	150 <i>infeasible</i>	45
Objective: ↓	49.823	83.673
Parameters		<input type="button" value="Copy best"/>
tracksNumber	13	11
trainsDayFlow	72	72
timeService	30	30
crewTONumber	10	2

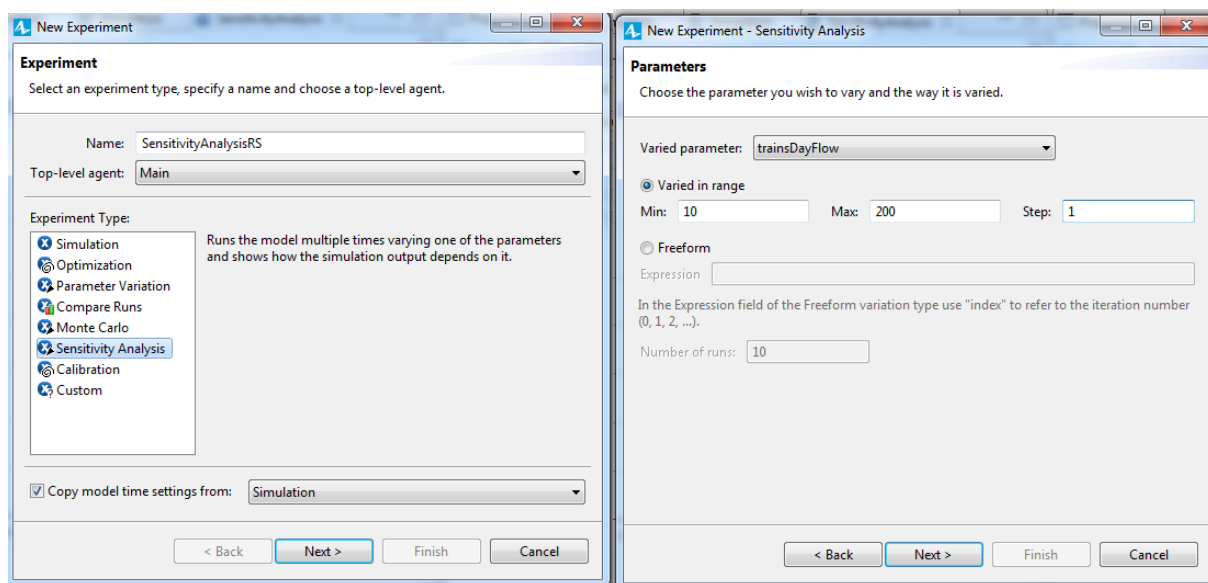


● Current ● Best infeasible ● Best feasible

4 Експериментальне встановлення потрібної граничної зміни обраного параметру при якому наявна імовірність технологічної відмови не буде меншою за потрібну

Встановлення граничного рівня навантаження на залізничну транспортну систему відноситься до комплексних (системних) оцінок відмовостійкого функціонування транспортних процесів та систем.

Задачею таких досліджень є визначення впливу одного або декількох взаємопов'язаних параметрів на функціонування одразу всієї системи. Для задач такого виду в AnyLogic є окремий вид експерименту – *Sensitivity Analysis*, або *Аналіз чутливості моделі*.

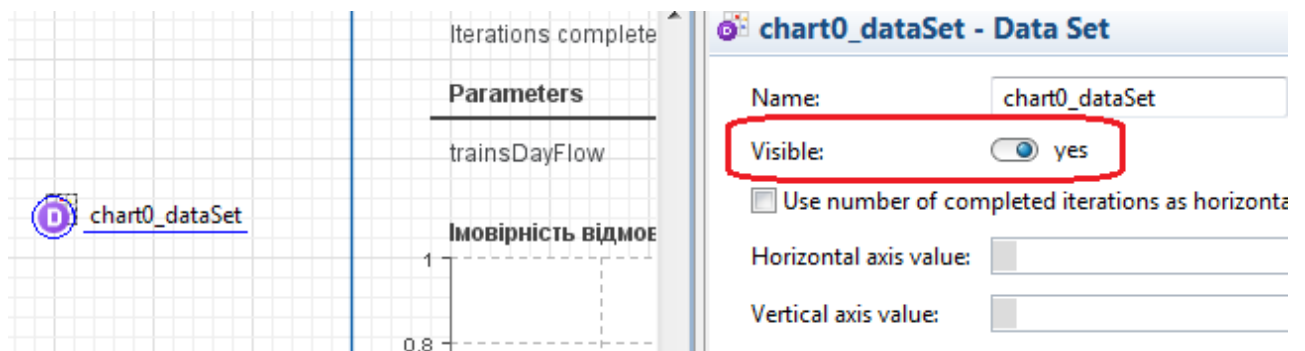
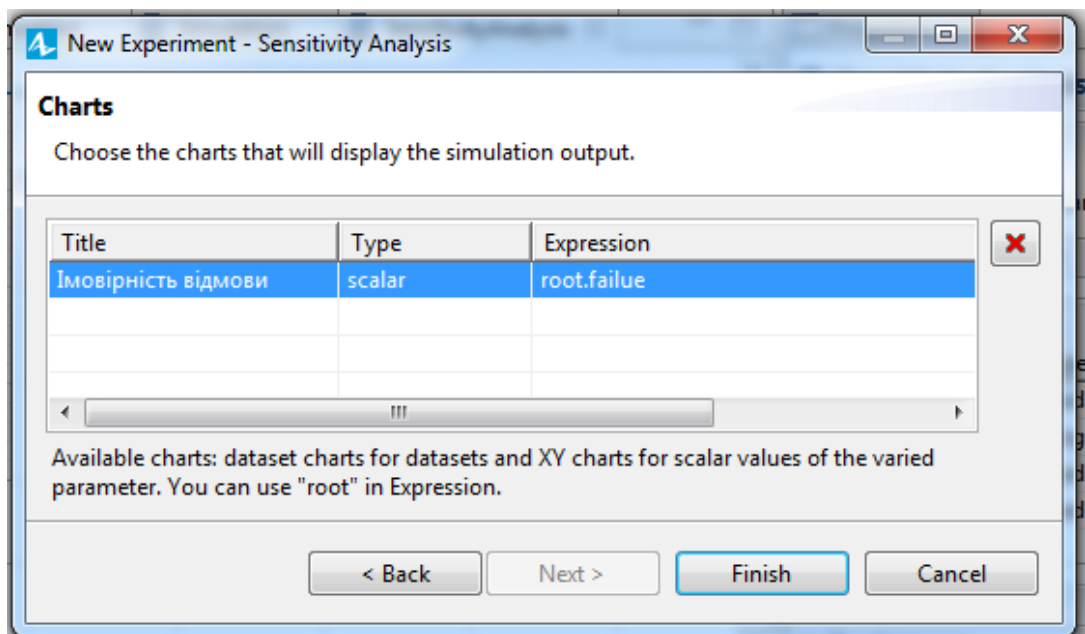


У другому вікні налаштування експерименту вибирається необхідний (відповідно до завдання, змінний параметр), далі діапазон та шаг його зміни. У третьому, останньому вікні налаштування, зазначається назва результату розрахунку (*Імовірність відмови*), варіант виводу результатів (необхідно вибрати *scalar*) та розрахункову змінну моделі, що відповідатиме обраному критерію оцінки функціонування моделі (змінна *failue* із правами доступу *root*, тобто *root.failue*).

При закінченні створення даного виду експерименту AnyLogic автоматично створить інтерфейс. В вікні інтерфейсу необхідно знайти елемент

формування бази результатів експериментів *chart0_dataSet*, та у властивостях до нього вибрати *Visible yes* (дивіться на рисунку нижче). Така процедура дозволить використовувати накопичувач *chart0_dataSet* для аналізу результатів даного експерименту.

При запуску експерименту *SensitivityAnalysis* отримаємо графік залежності рівня імовірності відмови від заданого (відповідно до завдання) параметру. У прикладі це потужність добового поїздопоток – *Середньодобова кількість поїздів*. З результатів експерименту видно, що граничний рівень імовірності відмови у 30% буде перевищений при добовій кількості поїздів 82 та більше.



Iterations completed: 95

Parameters

trainsDayFlow 104

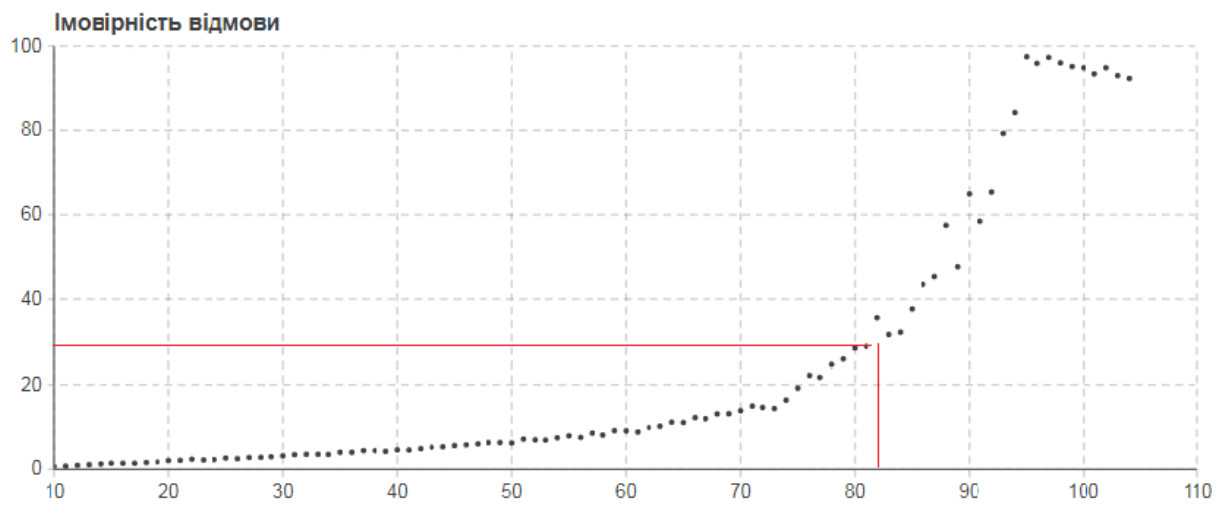


chart0_dataSet
95 samples (104, 92, 296)

x	y
79	25.863
80	28.463
81	28.798
82	35.613
83	31.62
84	32.2
85	37.678
86	43.499
87	45.35
88	57.487
89	47.695
90	64.935

5 Структура та зміст пояснювальної записки

Зміст

Вступ

У вступі повинно бути зазначено актуальність проекту, мета (загальна, та що планується зробити для досягнення мети). *Обсяг 1,5 – 2 арк.*

1. Постановка задачі та вибір інструментів дослідження

1.1 Постановка задачі

Після аналізу вихідних даних необхідно визначити конкретні задачі та алгоритм їх розв'язання відповідно до вихідних даних індивідуального завдання. Обґрунтувати критерії, обмеження та цільові функції. *Обсяг 3 – 4 арк.*

1.2 Обґрунтування вибору методів та методик проведення експериментів, збору експериментальних даних та їх аналізу

Обґрунтувати вибір інструментів (методів, методик) проведення досліджень: проведення експериментів, збору експериментальних даних та їх обробки. *Обсяг 3 – 4 арк.*

2. Розробка моделі технологічного процесу обробки наскрізних вантажних поїздів

2.1 Загальний опис технологічного процесу обробки наскрізних вантажних поїздів

Навести загальний опис нормативного технологічного процесу обробки наскрізних поїздів на технічних станціях. Провести аналіз технологічних елементів, що входять до даного процесу: канали обслуговування (приймально-відправні колії), прилади обслуговування (бригади ТО/КО, поїзні локомотиви) та ін. *Обсяг 2 – 3 арк.*

2.2 Розробка імітаційної моделі

Навести повний опис та покроковий порядок розробки імітаційної моделі. *Обсяг 10 – 15 арк.*

3. Експериментальне встановлення мінімально-можливої кількості приймально-відправних колій та оглядачів поїздів

Навести повний опис та покроковий порядок розробки експерименту *оптимізації*. Описати хід проведення експерименту. Навести результати експерименту у вигляді графіків і табличних даних та надати їм оцінку у вигляді висновків. *Обсяг 5 – 7 арк.*

4. Експериментальне встановлення потрібної граничної зміни обраного параметру при якому наявна імовірність технологічної відмови не буде меншою за потрібну

Навести повний опис та покроковий порядок розробки експерименту *оптимізації*. Описати хід проведення експерименту. Навести результати експерименту у вигляді графіків і табличних даних та надати їм оцінку у вигляді висновків. *Обсяг 5 – 7 арк.*

Висновок.

Привести загальні висновки щодо результатів дослідження у відповідність із поставленою метою у вступі проекту; їх аналізу, можливого практичного використання. *Обсяг 2 – 3 арк.*

Список використаних джерел

Навести актуальні посилання у обсязі не менш 20, з них: 5 – 10 нормативно інструктивні та методичні положення; 5 – 10 актуальні (не старше 5 років) наукові статті на тему моделювання складних транспортних процесів та систем залізничного транспорту.

Додатки (у разі необхідності)

6 Практичні заняття

1. Показники перевізної роботи. Показники транспортного забезпечення та доступності.
2. Сфери раціонального використання різних видів промислового транспорту, муніципального транспорту.
3. Собівартість перевезень, особливості визначення і різниця за видами транспорту. Капіталовкладення за видами транспорту.
4. Розподіли інтервалів прибуття поїздів.
5. Канальність СМО.
6. Міжопераційні простої та технологічні конфлікти.
7. Імітаційне моделювання.
8. Об'єктно-подієве моделювання.

Список рекомендованої літератури

1. AnyLogic Help System [Electronic resource]. – Mode of access: World Wide Web: – <http://www.anylogic.com/anylogic/help/> (viewed on April 21, 2017). – Title from the screen.
2. Matsiuk, V. A study of the technological reliability of railway stations by an example of transit trains processing [Text] / V. Matsiuk// Eastern – European Journal of Enterprise Technologies: Control processes. – 2017. – Vol. 1. P. 12-17.
3. ДБН-В.2.3-19-2008 Державні будівельні норми. Споруди транспорту. Залізниця колії 1520 мм. – Введений 2008-01-26. – К.: Мінрегіонбуд України, 2008. – 126 с.
4. Карпов Ю. Г. Изучение современных парадигм имитационного моделирования в среде AnyLogic. Компьютерные инструменты в образовании / Ю. Г. Карпов. – 2005. – № 4. – С. 3 – 14.
5. Карпов Ю. Имитационное моделирование систем. Введение и моделирование с AnyLogic 5 / Карпов Ю. – СПб.: БХВ-Петербург, 2005. – 400 с.
6. Мезенцев К. Н. Учебное пособие «Моделирование систем в среде AnyLogic 6.4.1». Часть 2 / Мезенцев К. Н.; Под редакцией Заслуженного деятеля науки РФ, д.т.н., профессора А.Б.Николаева. МАДИ. – М.: 2011. – 103 с.
7. Надійність техніки. Системи технологій. Терміни та визначення: ДСТУ 2740-94. – [Чинний від 01.01.1996]. – К.: Держстандарт України, 1994. – 33 с. – (Національний стандарт України).
8. Практичні рекомендації щодо складання технологічного процесу роботи дільничної станції, затверджені Наказом Укрзалізниці від 03.03.2010 р. № 163-Ц (ЦД-0082). – К.: ТОВ «Інпрес», 2010. – 204 с.
9. Практичні рекомендації щодо складання технологічного процесу роботи сортувальної станції, затверджені Наказом Укрзалізниці від 22.12.2009 р. № 715-Ц (ЦД-0081). – К.: ТОВ «НВП Поліграфсервіс», 2010. – 230 с.

10. Мацюк В. І. Дослідження стану та тенденцій розвитку технологічної надійності пасажирських перевезень [Текст] / В. І. Мацюк, В. О. Горбатюк, О. А. Горецький // Вісник східноукраїнського національного університету імені Володимира Даля. Науковий журнал. – 2017. – № 4 (234). – С. 150 – 154.

11. Мацюк В. І. Принципи забезпечення технологічної надійності залізничних транспортних систем [Текст] / В. І. Мацюк// Збірник наукових праць Державного економіко–технологічного університету транспорту. Серія: Транспортні системи і технології. – 2016, № 28. – С. 262 – 271.

12. Мацюк В. І. Дослідження технологічної надійності парків технічних станцій дискретно–подієвим моделюванням [Текст] / В. І. Мацюк// Збірник наукових праць Державного економіко–технологічного університету транспорту. Серія: Транспортні системи і технології. – 2015, № 26 – 27. – С. 268 – 272.

13. Мацюк В. І. Дослідження повної та систематичної технологічних відмов залізничних станцій [Текст] / В. І. Мацюк // Збірник наукових праць Державного економіко–технологічного університету транспорту. Серія: Транспортні системи і технології. – 2017. – Вип. 30. – С. 226-236.

Методичні вказівки

Мацюк Вячеслав Іванович

**Інноваційні технології управління
перевізним процесом**

Методичні вказівки до виконання курсового проекту, практичних занять та дипломного проектування для денної та заочної форм навчання напряму підготовки галузі знань: 27 «Транспорт», напряму підготовки: 275.02 «Транспортні технології (на залізничному транспорті)»