

АЛГЕБРАЇЧНЕ МОДЕЛЮВАННЯ В СИСТЕМАХ МІЖНАРОДНОЇ ТА МІСЦЕВОЇ ОБСЛУГОВУЮЧОЇ ЛОГІСТИКИ

Міжнародна та внутрішня обслуговуюча логістика розвивається шаленими темпами у сучасному житті, а прогнози на майбутнє для цієї галузі оптимістичні. З цих причин ми постали перед задачею контролю, оптимізації, безпечної та надійної перевірки комплексних логістичних систем з багатьма внутрішніми агентами, які працюють у середовищі, що змінюється. Дана робота має на меті показати як математичне моделювання, зокрема, алгебра поведінок, надає можливість передбачувати поведінку, стабільність логістичних середовищ та перевіряти їх властивості безпеки, та надійності. Ці процедури є модельно-орієнтованими розробками комплексних систем програмного забезпечення, таких як логістичні системи.

Ключові слова: алгебра поведінок, інсерційне моделювання, формальна верифікація, ланцюг постачання, модельно-орієнтовані розробки, властивості безпеки.

1. Загальна інформація

Стабільність функціонування, стійкість до зовнішніх загроз, виявлення та усунення вразливостей є необхідними властивостями комплексних систем міжнародної та внутрішньої обслуговуючої логістики. Такі системи програмного забезпечення є критично важливими до безпеки та вимагають розробок з модельно-орієнтованим підходом для надійності. Модельний підхід до розробки передбачає створення моделей як інструментів на кожному етапі розробки програмного забезпечення для застосування методів верифікації, тестування та валідації.

Методи математичного моделювання з такими інструментами, як алгебра поведінок, дають можливість керувати логістичними системами та забезпечувати їх безпеку та захищеність за допомогою формальної верифікації та формалізації за допомогою тестування на основі моделей. Алгебраїчні моделі логістичних систем можуть бути використані для аналізу поведінки всіх залучених агентів і доведення їх здатності виконувати свої цілі та здатності всієї системи постійно існувати та залишатися стабільною.

Формальна верифікація використовується в аналізі та верифікації бізнес-логістики вже декілька десятиріч. Формальні мови такі як: BPMN [1], UML [2], SysML [3] використовуються для опису

логістичних бізнес процесів, а досить велика кількість методів такі як: VDM [4], SPIN [5] та інші застосовується для перевірки властивостей безпеки. Ускладнення специфіки та збільшення інформаційного навантаження обумовило використання більш новітні методики та відповідні математичних теорії.

У цьому дослідженні ми демонструємо, як алгебраїчний підхід, оснований на алгебрі поведінок, може бути ефективно використаний для перевірки властивостей безпеки складних систем. Як приклади ми застосовуємо її до реальної закритої логістичної системи місцевого обслуговування (фермерське господарство) та відкритої міжнародної логістичної системи (експортні поставки виробництва вищезгаданого фермерського домогосподарства).

2. Алгебра поведінок

На початку 1970-х, В.М. Глушков, директор Інституту кібернетики Національної академії наук України, заснував комп'ютерну наукову школу, яка сфокусувалась на алгебраїчних методах. Тематика дослідження варіювалась від автоматизованих доведень теорем до алгебраїчного моделювання та від наукових концепцій до промислових прикладних програм. Пізніше в 2000 Система Алгебраїчного Програмування (APS) [1] як складова частина цієї

програми досліджень розширилась за допомогою включення в себе концепції взаємодії перехідних систем у деяких алгебраїчних середовищах. Ключова ідея – інсерційне функціонування, яке визначає поведінку взаємодії перехідних систем в алгебраїчному середовищі. Ця концепція розроблена в контексті IMS [2]. Алгебра поведінок та основні специфікації протоколу використовується для формального опису моделі. Такі алгебраїчні інструменти, як теореми та абстрактні алгоритми, служать основою для формальних методів верифікації. Алгебраїчний підхід та інсерційне моделювання дозволяють довести або спростувати властивості, подавши контр-приклади для системи з довільною кількістю агентів. Використовуючи формальну модель програми на якомусь рівні абстракції, ми можемо генерувати різні сценарії поведінки агентів або груп агентів. Ці сценарії є символічними і можуть бути проілюстровані конкретними прикладами за допомогою формальних методів. Генерація символічних сценаріїв забезпечує гарне висвітлення поведінки, а отже, отримані конкретні сценарії можуть розглядатися як тестовий набір для створеного програмного забезпечення.

Інсерційне моделювання фокусується на моделях побудови та вивченні взаємодії агентів та середовищ у складних розподілених мультиагентних системах. Загальні концепції інерційного моделювання – це ієрархія середовищ та агентів, занурених у ці середовища, взаємодія цих агентів, середовищ та середовищ вищого рівня, двосторонній вплив агентів та середовищ один на одного, зміна поведінки набору агентів під час занурення в нові середовища. Середовище – це агент, який має функцію занурення. Агенти розглядаються як системи переходу атрибутів. У таких системах стани визначаються значеннями атрибутів. Агенти мають набір атрибутів, які визначають тип агента. Атрибути середовища пов'язані з глобальними атрибутами, які відомі всім агентам.

В 1997 Гілберт та Летичевський ввели поняття алгебри поведінок [3] як інструмент та невід'ємна частина інсерційного моделювання. Алгебра поведінок – це

універсальна алгебра двох видів. Основний різновид – це набір поведінок, а другий – сукупність дій. Ця алгебра має дві операції, три термінальні константи та відношення наближення. Операції позначаються префіксами $a.u$ (де « a » – дія, а « u » – поведінка) та недетермінований вибір поведінки $u + v$ (асоціативні, комутативні та ідемпотентні операції на множині поведінок). Кінцевими константами є успішно визначене Δ , заключення 0 , та дивергентна поведінка \perp . Знаходження наближеного зв'язку \sqsubseteq є частковим порядком у наборі поведінок з мінімальним елементом \perp . Алгебра поведінок також наповнена двома операціями: паралельними (\parallel) та послідовними ($;$) композиціями поведінок.

Одним з прикладів вираження поведінок є:

$$B_0 = a_1.a_2.B_1 + a_3.B_2,$$

$$B_1 = a_4, B_2 = \dots$$

Мається на увазі, що поведінка B_0 може бути представлена, як послідовність дій a_1 та a_2 та поведінки B_1 , або як послідовність дії a_3 та поведінки B_2 . Поведінка B_1 закінчується після дії a_4 [4].

3. Приклад формалізації системи взаємодії агентів закритої логістичної системи

Розглянемо застосування на практиці алгебри поведінок на прикладі діючої закритої логістичної системи фермерського господарства. Закрите середовище містить у собі набір агентів, що мають певні атрибути, властивості та список дій, які вони виконують за певних передумов. Контроль та взаємодія агентів даної системи здійснюється за допомогою Централізованої бази даних (сервера), яка збирає інформацію про стан кожного агента та координує їх взаємодію за допомогою команд. Централізована база даних є одним із агентів системи, функція якого є безперервне отримання, накопичення та обробка інформації від агентів. За умови виконання певної передумови Централізована база даних надсилає агенту сповіщення-команду (коротке сервісне повідомлення) про початок виконання певної дії. Активність

кожного агенту описана математичною моделлю поведінок, яка передбачає хід усіх можливих дій такого агенту. Моделювання поведінок усіх агентів дає змогу

проаналізувати цілісну комплексну логістичну систему в роботі та виявити її вразливості. Список основних агентів та їх дій наведено в табл. 1.

Таблиця 1. Список агентів та їх дій

Агент	Список дій
Агроном	– дає команду про готовність сої для збору урожаю
Комбайн (один або більше)	– простій (або зупинка) – збір урожаю та вивантаження на ходу у вантажівку – технічне обслуговування – заправка – заміна водія при цілодобовій роботі позмінно
Бензовоз	– простій (або зупинка) – забір пального на базі – заправка транспортних засобів та механізмів у разі отримання сигналу від централізованої бази даних про низькі об'єми палива, отримані від датчиків палива цих засобів та механізмів – заміна водія при цілодобовій роботі позмінно – технічне обслуговування – заправка
Вантажівка (одна або більше)	– простій (або зупинка) – супровід комбайна під час збору урожаю та приймання сої на ходу – транспортування на склад та вивантаження – технічне обслуговування – заправка – заміна водія при цілодобовій роботі позмінно
Склад та лінія переробки	– простій (або зупинка) – приймання сої та зважування – контроль показників якості – контроль вологості та досушка, провіювання – переробка сої на соєву олію – зважування готової продукції – відвантаження готової продукції – заміна працівників при цілодобовій роботі позмінно – передача отриманої інформації на всіх етапах до ЦБД
Фура	– простій (або зупинка) – очікування завантажування на складі – завантажування та зважування – доставка кінцевому споживачеві (по Україні або за кордон)
Сервісна служба (механік)	– простій (або зупинка) – виконання технічного обслуговування – заправка – заміна водія при цілодобовій роботі позмінно
Централізована база даних (ЦБД)	– безперервний збір та обробка інформації від GPS та інших датчиків від всіх агентів на всіх етапах – передача команд та інформації відповідним агентам – формування бази даних – опрацювання отриманої інформації на всіх етапах від агента Склад та лінія переробки, внесення до операційної бази даних – надання інформації зацікавленим сторонам та агентам

Приклади запису моделей поведінок агентів за допомогою алгебри поведінок, розглянемо поведінки декількох базових агентів:

1. Комбайн

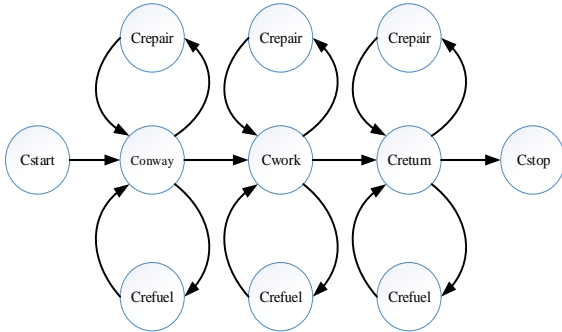


Рис. 1. Схема математичної моделі комбайну

Математична поведінкова модель:

$$C = Cstart.Conway.Cwork.Creturn.Cstop$$

$$Conway = Cmove.Conway + Cmove.Crepair.Conway + Cmove.Crefuel.Conway$$

$$Creturn = Cmove.Creturn + Cmove.Crepair.Creturn + Cmove.Crefuel.Creturn$$

$$Cmove = (PdX.Cmove + PdY.Cmove) + Conplace.$$

Поведінка переміщення комбайна до робочого місця Conway та назад до бази Creturn складається з поведінки циклу Cmove, що закінчується Conplace та поведінок ремонту Crepair та заправки Crefuel.

$$Cwork = Cgath.Cwork + Cgath.Crepair.Cwork + Cgath.Crefuel.Cwork$$

$$Cgath = SdX.Cgath + Sdy.Cchdir.Cgath) + Send.$$

Поведінка роботи комбайна Cwork складається з поведінки циклу Cgath, що закінчується Send та поведінок ремонту Crepair та заправки Crefuel.

$$Crepair = Cstop.Cdamage,$$

Поведінка ремонту комбайна Crepair є дії Cstop та Cdamage.

$$Crefuel = Cstop.Cfuel.$$

Поведінка заправки комбайна Crefuel є дії Cstop та Cfuel.

2. Бензовоз

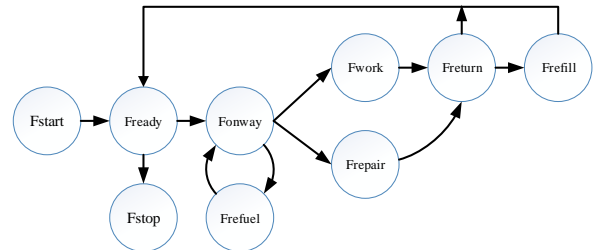


Рис. 2. Схема математичної моделі бензовозу

Математична модель поведінки:

$$F = Fstart.Fcycle.Fstop,$$

$$Fcycle = Fready.Fcycle + Fready.Fonway.Fwork.Freturn.Frefill.Fcycle + Fready.Fonway.Freturn.Fcycle.$$

Поведінка робочого циклу бензовоза Fcycle – цикл зміни дій:

$$Fonway = Fmove.Fonway + Fmove.Frepair + Fmove.Frefuel.Fonway,$$

$$Freturn = Fmove.Freturn + Fmove.Frepair.Freturn + Fmove.Frefuel.Freturn.$$

Поведінка переміщення бензовоза до робочого місця Fmove є циклом зміни положення, що закінчується Fonplace або поведінками ремонту Frepair та заправки Frefuel.

$$Fmove = (SdX.Fmove + SdY.Fmove) + Fonplace.$$

Поведінка переміщення бензовоза до робочого місця Fmove є циклом зміни положення, що закінчується Fonplace або поведінками ремонту Frepair та заправки Frefuel.

$$Frepair = Fstop.Fdamage,$$

Поведінка ремонту бензовоза Frepair є дії Fstop та Fdamage.

$$Frefuel = Fstop.Ffuel.Fwork.$$

Поведінка заправки бензовоза Frefuel є дії Fstop, Ffuel та Fwork.

3. Механік

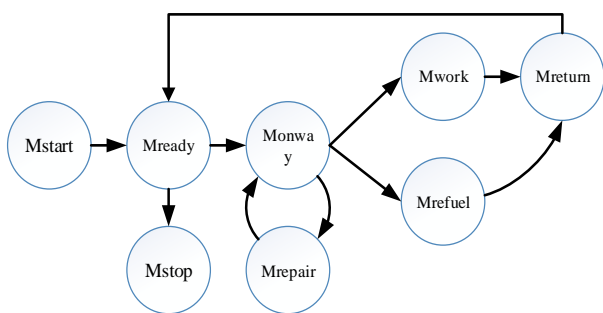


Рис. 3. Схема математичної моделі механіка.

Математична модель:

$$M = Mstart.Mcycle.Mstop$$

$$Mcycle = Mready.Mcycle + Mready.Monway.Mwork.Mreturn.Mcycle + Mready.Monway.Mreturn.Mcycle$$

Поведінка робочого циклу механіка Mcycle – цикл зміни дій

$$Monway = Mmove.Monway + Mmove.Mrefuel + Mmove.Mrepair.Monway$$

$$Mreturn = Mmove.Mreturn + Mmove.Mrepair.Mreturn + Mmove.Mrefuel.Mreturn$$

Поведінка переміщення механіка до робочого місця Monway та назад до бази Mreturn складається з поведінок ремонту Mrepair, заправки Mrefuel та поведінки циклу Mmove

$$Mmove = (SdX.Mmove + SdY.Mmove) + Monplace$$

Поведінка переміщення механіка до робочого місця Mmove – цикл зміни положення, що закінчується Monplace або

поведінками ремонту Mrepair та заправки Mrefuel

$$Mrepair = Mstop.Mdamage.Mwork$$

Поведінка ремонту механіка Mrepair – дія Mstop та Mdamage

$$Mrefuel = Mstop.Mfuel$$

Поведінка заправки механіка Mrefuel – дія Mstop, Mfuel та Mwork.

4. Вантажівка

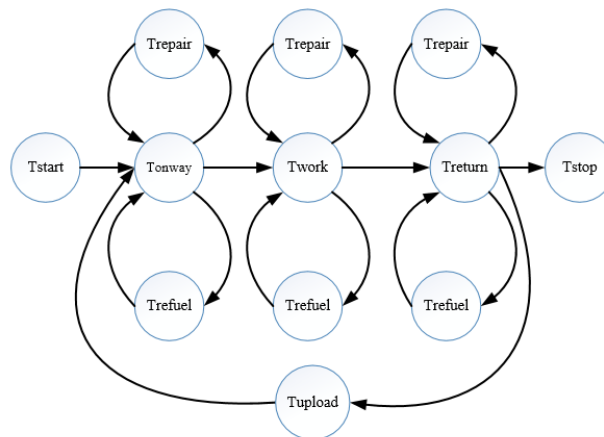


Рис. 4. Схема математичної моделі вантажівки.

Математична модель:

$$T = Tstart.Tonway.Twork.Treturn.Tstop$$

$$Tonway =$$

$$Tmove.Tonway + Tmove.Trepair.Tonway + Tmove.Trefuel.Tonway$$

$$Treturn = Tmove.Treturn + Tmove.Trepair.Treturn + Tmove.Trefuel.Treturn$$

$$Tmove = (PdX.Tmove + PdY.Tmove) + Tonplace$$

Поведінка переміщення вантажівки до робочого місця Tonway та назад до бази Treturn складається з поведінки циклу Tmove, що закінчується Tonplace та поведінок ремонту Trepair та заправки Trefuel

$$Twork = Tgath.Twork + Tgath.Trepair.Twork +$$

```
+ Tgath.Trefuel.Twork +
+ Tgath.Tunload.Twork
Tgath = (SdX.Tgath +
Sdy.Tchdir.Tgath) + Send
```

Поведінка роботи вантажівки Twork складається з поведінки циклу Tgath, що закінчується Send та поведінок ремонту Trepair та заправки Trefuel

```
Trepair = Tstop.Tdamage
```

Поведінка ремонту вантажівки

Trepair є дії Tstop та Tdamage

```
Trefuel = Tstop.Tfuel
```

Поведінка заправки вантажівки Trefuel є дії Tstop та Tfuel

```
Tunload = Treturn.Treload.Tonway
```

Поведінка вивантаження вантажівки Tunload є дії Treturn, Treload та Tonway.

Крім поведінок ми маємо також список дій, що можуть бути представленими перед- та післяумовою. Ми наводимо як приклад дії комбайну з описом відповідної семантики. Ми не наводимо поведінку інших агентів таких як склад, фура, механіка, вантажівки, централізована база даних у зв'язку з великою кількістю формалізації (табл. 2).

Таблиця 2. Список дій та їх передумов та післяумов

Назва дії	Передумова	Післяумова	І-ція переходу	Коментар
Cstart	1 (безумовна дія)	(X==Xs) && (Y==Ys)		Початок роботи
Cdamage	1<=i<=combineQuantity	combine(i).state=DAMAGE	send help(i)	Якщо трапилась поломка відіслати сигнал про допомогу та перейти в стадію DAMAGE
Cstop	1<=i <= combineQuantity	combine(i).state=STOP		Зупинитися та перейти в стадію STOP
Cfuel	1<=i<=combineQuantity) && (combine(i).fuelLevel <= 0.2 * combine(i).fuelVolume)	combine(i).state = FUEL	send help(i)	Якщо рівень пального менше норми відіслати сигнал про допомогу та перейти в стадію FUEL
PdX	X < Xp	X = X + deltaX		Якщо не на місці призначення, то рухатись на якусь deltaX
PdY	Y < Yp	Y = Y + deltaY		Якщо не на місці призначення, то рухатись на якусь deltaY
SdX	((dir > 0) && (X < XLe)) ((dir < 0) && (X > XLe))	X = X + deltaX * dir		Якщо не кінець довжини ділянки, то рухатись на якусь deltaX
SdY	((dir > 0) && (X > XLe)) ((dir < 0) && (X < XLe)) && (Y < YWe)	Y = Y + deltaY		Якщо кінець довжини ділянки, але не кінець ширини поля, то починаємо нову смугу на полі
Cchdir	((dir > 0) && (X > XLe)) ((dir < 0) && (X < XLe)) && (Y < YWe)	dir = -dir		Якщо кінець довжини ділянки, але не кінець ширини поля, то змінюємо напрям
Conplace	(X == Xp) &&(Y == Yp)	1		Якщо комбайн на місці призначення

4. Формальна верифікація моделей

Дана методика формалізації дозволяє застосувати далі формальні методи верифікації, таких як перевірка властивостей безпеки та випадки загрози для безперервної життєдіяльності системи. Надзвичайно важливо перевіряти деякі ключові фактори, які впливають на можливість виконання злагоджених дій усіх агентів системи, що в результаті і відображається в безперервній та продуктивній діяльності системи. Розглянемо основні властивості безпеки, що перевіряються в логістичній системі фермерського господарства.

1. Терміни збору врожаю. Дана властивість може бути виражена формулою у базовій логічній мові, що вибрана для формалізації дій ($T \leq 90$ днів). При розгляді довільної кількості комбайнів та обслуговуючого транспорту така ситуація може бути при нераціональній кількості транспортних та стану техніки. Для перевірки властивості використовуються методи моделювання такі як символічне, конкретне та змішане моделювання системи. На початку враховуються такі умови, як кількість допустимих несправностей техніки (як арифметична нерівність), конкретна кількість одиниць техніки та відстані до місця роботи. Для визначених початкових даних проводиться конкретне моделювання та визначається властивість порушення термінів збору врожаю. При використанні зворотного символічного моделювання від стану, що відповідає дотримання термінів та гіпотетичних можливих кількостей одиниць, несправностей та відстаней до місця роботи, можливо визначити початкові умови при яких ця властивість не буде порушена. Слід відмітити, що дана технологія не вирішує задачу оптимізації, а лише перевіряє можливість її реалізації. Методи символічного моделювання розглянуті в [9].

2. Достатність кількості обслуговуючих агентів (типу бензовоз чи механік) для обслуговування життєдіяльності головних агентів системи, здатність вчасно прибути до заданої точки призначення для виконання заданих Центральною базою даних дій. Дані властивості перевіряються аналогічно при висуванні деяких гіпотез

при проектуванні задачі або реальної наявності техніки. Такі властивості також представляються за допомогою нерівностей (або рівностей).

3. Перевірка злагодженості та узгодженості алгоритмів дій агентів (наприклад, дії бензовоза при одночасному виклику обслуговування декількох агентів). Дана властивість може бути виражена за допомогою алгебри поведінок. Наприклад, чи встигне бензовоз заправити всі комбайни, якщо вони одночасно зупиняться, за встановлений термін. Маємо властивість безпеки ($T \leq 2$ год) та саму поведінку:

```
Cdamage.Cdamage.Cdamage.Cdamage.X; (Trepair.Trepair.Trepair.Trepair).
```

Розглянутий приклад ілюструє поведінку чотирьох комбайнів, які мали несправність та були полагоджені за деякий час. Чи відповідає цей час встановленому визначає конкретне або змішане моделювання. Зауважимо, що при символічному моделюванні можливо оцінити властивість для довільної кількості комбайнів.

4. Вплив непередбачуваних факторів типу погоди, рельєфу та інше. Дана властивість вивчається за допомогою встановлення можливих дій, що моделюють дані фактори та відповідними видами моделювання.

5. Дотримання умов перевезення вантажу (в нашому випадку зібраного урожаю) при передачі від одного агента іншому. При перевезенні розглядаються додаткові фактори середовища, які моделюються в процесі верифікації. Це час перевезення, вологість, температура, які беруться до уваги в моделі, як додаткові атрибути системи.

5. Верифікація і формалізації системи міжнародної логістики

Відкрита система міжнародної логістики є продовженням описаної закритої логістичної обслуговуючої системи фермерського господарства. Вихідний продукт фермерського господарства (наприклад, соя) є вхідним продуктом для відкритої міжнародної логістичної системи (переробка сої на соєву олію та її експорт). Істотною відмінністю цієї системи є теоре-

тично нескінченна кількість не пов'язаних між собою незалежних агентів кожного типу, і, як наслідок, постійна зміна середовищ існування. Аналогічно з закритою логістичною системою, де функцію збору інформації та контролю виконує Централізована база даних, в даній відкритій системі існує подібний агент – Система. Однак на відміну від закритої системи, де агенти діють злагоджено та в спільних

інтересах, вимоги надійності та безпеки до відкритої системи значно вищі, так як кожен з незалежних агентів діє в своїх особистих інтересах. Тому для даних цілей доцільно застосовувати технологію розподілених децентралізованих систем (блокчейн) замість стаціонарного серверу. Список типових основних агентів, їх функцій та інформацію, яку вони надають розподіленій системі, наведено в табл. 3.

Таблиця 3. Список агентів, їх функцій та інформації, яку вони надають централізованій системі

Agent	Функції	Інформація, яку надає для системи
Виробник (він же експортер) товару	<ul style="list-style-type: none"> – виробництво товару – пошук покупця – аналіз цін (пропозицій) та вибір оптимального покупця – відвантаження товару – пошук оптимальних логістичних агентів – портових операторів та морських перевізників 	<ul style="list-style-type: none"> – наявний об'єм товару до відвантаження – базові характеристики та якість товару (замовляється експортером) – узгоджена ціна – узгоджений покупець – узгоджені з Покупцем умови відвантаження та оплати всіх дій (хто оплачує послуги посередників та на яких етапах здійснення) – дата готовності до відвантаження – узгоджені логістичні агенти – портовий оператор та морський перевізник
Портовий оператор	<ul style="list-style-type: none"> – здійснення внутрішньої обслуговуючої логістики від виробника до завантаження на морське судно 	<ul style="list-style-type: none"> – дата подачі контейнера виробнику згідно дати готовності до відвантаження, надану виробником (пізніше або в день його дати) – підтвердження здійснення та дата проміжних етапів: відвантаження у виробника, прибуття в порт, завантаження на судно та інші можливі етапи
Морський перевізник	<ul style="list-style-type: none"> – приймає на борт судна контейнер від Портового оператора та здійснення перевезення 	<ul style="list-style-type: none"> – факт та дата прийняття контейнера на борт судна
Почувець товару	<ul style="list-style-type: none"> – пошук продавця – аналіз цін (пропозицій) та вибір оптимального продавця – оплата за товар 	<ul style="list-style-type: none"> – бажаний об'єм товару для відвантаження – мінімальні характеристики та якість товару (підтверджується продавцем) – узгоджена ціна – узгоджений продавець – узгоджені з продавцем умови відвантаження та оплати всіх дій (хто оплачує послуги посередників та на яких етапах здійснення) – підтвердження наданих продавцем логістичних агентів – портовий оператор та морський перевізник
Банк кожного з агентів	<ul style="list-style-type: none"> – здійснює оплату за товар чи послуги згідно інформації, отриманої від свого агента 	<ul style="list-style-type: none"> – інформація про суму та дату здійснення оплати, кому з агентів та за яку роботу

Поведінка та дії агентів описуються аналогічно моделі фермерського господарства. До моделі включаються відстані та географічна інформація про точки в яких діють агенти. Як властивості які можуть бути перевірені такі:

- своєчасна доставка;
- дотримання температурного режиму;
- дотримання режиму вологості;
- здатність протистояти зовнішнім факторам.

Властивості перевіряються такими методами алгебри поведінок, як алгебраїчне зіставлення, символічне моделювання системи із довільною кількістю агентів та статичні методи доведення. Специфіка системи – використання децентралізованих систем, які є однотипними, але зберігають всі атрибути, які стосуються всього ланцюга постачання, дані про перевозку, транзакції з оплатою тощо. Кожен агент зберігає файл з інформацією про ланцюг постачання або транзакції. Ми розглядаємо два типи використання формальних методів в даній моделі.

1. Моделювання ланцюга постачання для перевірки властивостей. В даній процедурі ми перевіряємо чи відповідає вибраний ланцюг постачання вимогам перевезення. Перевіряються також властивості супротиву до зовнішніх небажаних дій – погоди, шахрайства, кібератаки.

2. Моніторинг перевезення згідно вибраної моделі безпеки. Дана процедура призначена для виявлення порушення властивостей безпеки заздалегідь при вивченні сценарію поведінки. При моніторингу разом із моделлю безпеки може бути використана модель класифікації, що створена за допомогою машинного навчання.

Висновки

У цій роботі ми дослідили можливі підходи до застосування математичного моделювання та застосування алгебри поведінок для прогнозування безпеки та надійності логістичних систем відкритого та закритого типів. Наші експериментальні результати підтвердили велику значущість

використання математичного моделювання та використання алгебри поведінок для застосування в реальних логістичних системах.

Для складних комплексних систем, такі як логістичні, вкрай важливе значення має перевірка властивостей безпеки та здатності функціонувати тривалий час без втрати результативності. Дана методика є одним з вагомих інструментів при виконанні цих задач. Моделювання поведінки кожного агента системи дає змогу проаналізувати властивості всієї системи загалом.

На даний час перспектива цієї роботи вбачається у поєднанні описаної методики з розподіленими системами (зокрема блокчейн). Вже розроблена платформа-клієнт, що виступає незалежним агентом, що контролює дії усіх учасників, та найближчим часом планується експеримент з використання даного проекту при реальних експортних відправках.

Література

1. BPMN (Business Process Model and Notation). www.bpmn.org
2. UML (Unified Modelling Language). www.uml.org
3. SysML (Systems Modelling Language). www.sysml.org
4. VDM (Vienna Development Method). www.vienna.cc/e/evdm.htm
5. SPIN. <http://spinroot.com/spin/whatispin.html>
6. APS (Algebraic Programming System). www.apsystem.org.ua
7. Letichevsky A., Letychevskiy O., and Peschanenko V. Insertion Modeling and Its Applications. *Computer Science Journal of Moldova*. 2016. vol. 24. N. 3. P. 357–370.
8. Letichevsky A. and Gilbert D. Interaction of agents and environments. In *Recent Trends in Algebraic Development Technique*. LNCS 1827. Springer-Verlag. 1999.
9. Letychevskiy O., Letichevsky A., Peschanenko V., Weigert T., Insertion modeling and symbolic verification of large systems,"I LNCS 9369 SDL 2015: Model-Driven Engineering for Smart Cities. Springer. 2015. P. 3–18.

References

1. BPMN (Business Process Model and Notation). www.bpmn.org
2. UML (Unified Modelling Language). www.uml.org
3. SysML (Systems Modelling Language). www.sysml.org
4. VDM (Vienna Development Method). www.vienna.cc/e/evdm.htm
5. SPIN. <http://spinroot.com/spin/whatispin.html>
6. APS (Algebraic Programming System). www.apsystem.org.ua
7. Letichevsky A., Letychevskyi O., and Peschanenko V. Insertion Modeling and Its Applications. *Computer Science Journal of Moldova*. 2016. vol. 24. N. 3. P. 357–370.
8. Letichevsky A. and Gilbert D. Interaction of agents and environments. In *Recent Trends in Algebraic Development Technique*. LNCS 1827. Springer-Verlag. 1999.
9. Letychevskyi O., Letichevsky A., Peschanenko V., Weigert T., Insertion modeling and symbolic verification of large systems,”I LNCS 9369 *SDL 2015: Model-Driven Engineering for Smart Cities*. Springer. 2015. P. 3–18.

Одержано 23.10.2020

Про авторів:

Летичевський Олександр Олександрович, доктор фізико-математичних наук, завідувач відділу теорії цифрових автоматів
Кількість наукових публікацій в українських виданнях – 32.
Кількість наукових публікацій в зарубіжних виданнях – 37.
Індекс Гірша – 4,

Горбатюк Сергій Олександрович, аспірант кафедри Комп’ютерних наук та інформаційних технологій.
<https://orcid.org/0000-0001-6834-4211>,

Горбатюк Віктор Олександрович, аспірант кафедри Комп’ютерних наук та інформаційних технологій.
<https://orcid.org/0000-0001-7544-0260>.

Місце роботи авторів:

Інститут кібернетики імені В.М. Глушкова
Національної академії наук України.
03187, м. Київ
проспект Академіка Глушкова, 40.

Тел.: (044) 526-20-08.

E-mails: gorbatiuk_sergiy@i.ua,
lit@issukraine.com,
viktor.gorbatiuk@gmail.com