

VI. ЕКОНОМІКО МАТЕМАТИЧНІ-МЕТОДИ

УДК 004.9

В. М. Гужва, канд. екон. наук, доцент кафедри
інформаційних систем в економіці,
Київський національний економічний університет
імені Вадима Гетьмана

МОДЕЛЮВАННЯ ДІАЛогових ВЗАЄМОДІЙ МІЖ АГЕНТАМИ В ГОЛОНІСТИЧНИХ МУЛЬТИАГЕНТНИХ СИСТЕМАХ

Поява програмних агентів стала одним із найважливіших досягнень комп'ютерних наук у 90-х роках минулого століття. Технологія і теорія агентів продовжують розвиватися в рамках дослідницьких і комерційних проектів. Агенти застосовуються в додатках, де людина і комп'ютер тісно пов'язані між собою в процесі управління виробничими (технологічними) та невиробничими процесами. Термін «мультиагентні системи» (МАС) використовуються для позначення систем, до складу яких входить множина автономних модулів — програмних агентів. МАС можуть з успіхом використовуватися в економічній сфері, наприклад, для побудови інтелектуальних та віртуальних підприємств, мультиагентних систем підтримки прийняття рішень, систем електронної комерції, голоністичних виробничих систем тощо. Одне з ключових питань при проектуванні та реалізації МАС — це моделювання діалогових взаємодій окремих агентів, що входять до складу мультиагентної системи. Цій проблематиці і присвячена дана стаття.

КЛЮЧОВІ СЛОВА: програмний агент, голоністична мультиагентна система, комунікативна дія, протокол діалогу, розфарбовані мережі Петрі, атомарна схема діалогу, композитна схема діалогу.

Основні положення теорії мовних (комунікативних) взаємодій між агентами

Теорія мовних (комунікативних) дій традиційно використовується як основний механізм для моделювання, аналізу та проектування зв'язків між агентами. Але ще й досі немає засобів, які б дозволили агенту, який отримує повідомлення, повністю зрозуміти зміст повідомлення агента-відправника. В роботі [6] вказується на те, що основна проблема полягає в існуючих зв'язках агентів, які конкретизуються в термінах ментальних станів агента-відправника, і що шлях вирішення цієї проблеми — це створення специфікацій розширених зовнішніх протоколів, які харак-

теризують діалоги (розмови) агентів. Однак, перед тим, як розпочати розгляд взаємодії агентів, необхідно узгодити такі моменти: а) яким чином такі протоколи повинні бути конкретизовані; б) як вони повинні використовуватися при моделюванні та управлінні діалогами між агентами.

Насамперед необхідно виділити деякі ознаки, які можуть бути корисним формалізмом для представлення міжагентних діалогів. Наприклад, доцільним можна вважати такий підхід до побудови діалогів, який дозволяє:

- 1) визначати зразки діалогу (протоколи);
- 2) визначати моделі діалогів;
- 3) виділяти різні ролі, що мають місце в рамках мовної взаємодії;
- 4) виділяти загальні особливості діалогу (розмови);
- 5) виділяти стан комплексного діалогу (розмови) протягом часу виконання;
- 6) пряме відображення імплементації діалогу;
- 7) графічне представлення, яке підтримує візуалізацію, контроль та налагодження;
- 8) повторне багаторазове використання зразків діалогів.

Суть моделювання діалогових взаємодій агентів. Моделювання взаємодій агентів не є тривіальною задачею. Недостатньо мати відповідний формалізм моделювання з графічним представленням та формальною семантикою, яка підтримує процес аналізу. Важливо також розпізнати ключові аспекти взаємодій агентів та моделювати взаємодії на різних рівнях абстракції та різних рівнях деталізації. Для того, щоб вдосконалити процес моделювання, необхідно чіткіше визначитись у трьох основних поняттях, які традиційно використовуються при розгляді взаємодій агентів:

- 1) **протокол** (протокол взаємодії або діалогу) — це шаблон послідовності комунікативних дій;
- 2) **діалог (розмова)** — це зразок діалогу, специфічна послідовність комунікативних дій;
- 3) **політика** (політика взаємодії або діалогу) — стратегія, керівні принципи та обмеження, які визначають діалог (розмову).

Розглянемо детальніше ці поняття.

Насамперед слід зазначити, що найважливішим є термін **комунікативна дія**. **Комунікативна дія** є спеціальним типом дій у теорії мовних дій. Вона представляє основний «будівельний блок» діалогу між агентами і має чітку семантику, незалежну від змісту дії. Насьогодні є цілий ряд робіт, присвячених питанням стандартизації комунікативних дій, серед яких найбільшого поширення набули два стандарти, а саме: стандарт мови зв'язків агентів (МЗА), запропоно-

ваний Фондом інтелектуальних фізичних агентів (ФІФА), та стандарт мови запиту та маніпулювання знаннями (МЗМЗ). Обидва визначають набір комунікативних дій разом з зовнішнім (транспортним) шаром та внутрішнім (змістовним) шаром обміну повідомленнями.

Автором пропонується здійснювати моделювання діалогів між агентами у вигляді кількох розподілених шарів. Першим, найбільш фундаментальним, шаром є *протокольний шар*. **Протокол діалогу** або **протокол взаємодії** чи просто **протокол** є шаблоном послідовностей очікуваних комунікативних дій, організованих в ролі. Це визначення сумісне з визначенням протоколу, яке конкретизується в ФІФА наступним чином: «протокол — це загальний зразок діалогів, які використовуються для виконання деякого загального корисного завдання; протокол використовується з метою полегшення та спрощення роботи обчислювальних машин, необхідних для того, щоб підтримати задачі заданого діалогу між агентами; простіше: зразок діалогу» [3]. Протокол, окрім ідентифікації ролей та послідовностей комунікативних дій, також конкретизує відносини між **ролями**. **Роль** тотожна єдиній послідовності дій, які виконуються єдиною суттю; вона позначається ідентифікатором, тобто ім'ям такої послідовності.

Над цим шаром створюється ще один шар — *діалоговий шар*. **Діалог (розмова)** є специфічним зразком протоколу або набору протоколів; по суті це активізована послідовність повідомлень, обмін якими здійснюється між двома або більше агентами. При цьому слід виділити суттєву відмінність: протокол є тільки шаблоном або зразком, тоді як діалог (розмова) — це зразок даного шаблону або шаблонів. Таке визначення узгоджується з визначенням, наведеним у ФІФА: «діалог — це активізована послідовність комунікативних дій, яка має місце між агентами, що мають відношення до деякої активізованої теми діалогу (розмови)» [3].

Кінцевий, третій шар має назву шару *політики*. Це шар для координації; він не включається в явній формі в процес діалогового моделювання. Діалогова політика — це набір правил та специфікацій взаємодії, які визначають конкретний шлях або траєкторію в діалоговому просторі. *Політика* визначає деталі діалогу (розмови), який має місце між зацікавленими сторонами. Кожний *протокол* визначає простір можливих послідовностей комунікативних дій. Кожний *діалог (розмова)* слідує за однією траєкторією в цьому просторі. *Політика* веде специфічний діалог (розмову). Розглянемо, наприклад, протокол, який визначає дві ролі, *клієнта* і *торговця*; послідовність дій для клієнта є такою: *запит* (просить торговця продати конкретний товар), потім *прийняття* (приймає ціну і купує товари) або *відхилення* (визнає умови продажу неприйнятними, не купує товар).

А для торговця можливими відповідями на запити клієнта можуть бути наступні дії: *пропозиції* (пропонує ціни на товари), потім *продаж*, якщо пропозиції прийнято, або ж відсутність будь-яких дій, якщо пропозиції відхилено. Така проста специфікація є *протоколом*. Двом учасникам необхідно слідувати *за протоколом*, щоб сформулювати *діалог (розмову)*. Однією можливою *політикою* (стратегією) для клієнта може бути наступна — запитати товари у кількох інших агентів, що діють паралельно, прийняти найнижчу ціну та відхилити всі інші пропозиції. Таким чином можна динамічно створити відносно комплексний діалог (розмову), що об'єднує кількох агентів з продажу та одного агента-покупця. Простішою може бути така стратегія — *запитати* тільки одного агента з продажу, і *прийняти* або *відхилити* пропозицію, яка надається цим агентом, потім, якщо перша ітерація була закінчена без укладання угоди, розпочати діалог з іншим потенційним торговцем і повторювати вище наведені дії до тих пір, поки не буде досягнуто цілі.

Варіанти політики (стратегії) можуть бути здійснені простим набором правил, або, в комплексніших випадках, вони можуть мати свої власні комплексні протоколи, які існують і змінюють стан паралельно з визначеним контекстом активізованого діалогу (розмови). Під цими більшими комплексними обставинами можна розглядати «протокол взаємодії рівня політики» (інший протокол, але на рівні політики). При таких умовах, очевидно, можна мати користь від наявності іншого шару моделювання на рівні політики (над шаром звичайного моделювання діалогу) (рис. 1).

Таким чином, *діалог (розмова)* є комбінацією *протоколів*, які визначаються та маніпулюються специфічною *політикою*.

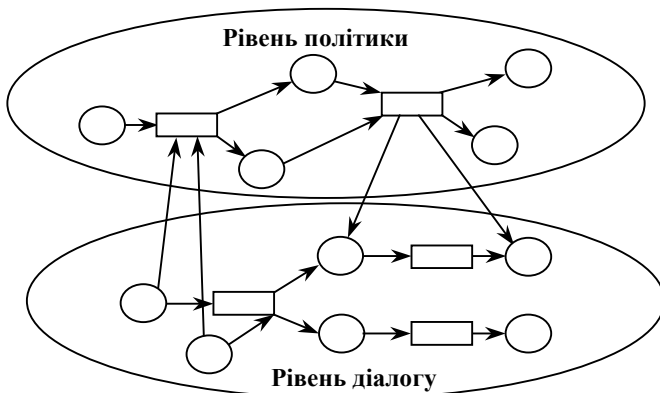


Рис. 1. Багатошарова архітектура діалогу між агентами

Аналіз існуючих підходів до моделювання взаємодій агентів в мультиагентних системах

Розглянемо деякі існуючі підходи до вирішення задач побудови діалогових взаємодій агентів. Зокрема, слід вказати такі методи, як метод детерміністичних скінчених автоматів (ДКА) [1], метод на основі удосконалених графів Дулі (Dooley) [2, 9] та метод на основі розширених UML-формалізмів [8].

Детерміністичні скінчені автомати представляють собою найпростіший формалізм моделювання, який використовується для побудови діалогів (розмов). Вони прийнятні для опису всіх станів, через які розмова повинна пройти, і можуть використовуватися для перевірки чинності розробленої моделі. Проте цей підхід страждає від двох головних недоліків: він практично не дозволяє розглядати паралелізм представлення діалогів і неспроможний явно зв'язати інформацію повідомлень з окремим учасниками розмови (тобто відсутня ідентифікація учасників діалогу). Моделювання на основі удосконалених графів Дулі — це метод, який базується на розширеннях до графів Дулі [2], які пропонуються в роботі [9]. Він забезпечує не тільки інформацію про стан, але також інформацію про індивідуальних учасників діалогу (розмови), що є значним позитивом у порівнянні з методикою на основі ДКА. Нещодавно було запропоновано ряд підходів, основою яких є уніфікована мова моделювання (UML) [7]. На відміну від методів на основі використання графів, методи на основі UML складаються з цілого ряду засобів та формалізмів моделювання, що базуються на схемах стану, діаграмах послідовності та діаграмах видів діяльності (активностей) [8]. Детальний порівняльний аналіз перерахованих та деяких інших методів наведено нижче.

Порівняльний аналіз існуючих та запропонованого методів для моделювання діалогових взаємодій агентів в мультиагентних системах. Для порівняння методів моделювання скористаємося такими властивостями (табл. 1):

1) *Можливість моделювання протоколу* — здатність формалізму моделювати єдиний зразок взаємодії, тобто протокол. Всі вказані в таблиці формалізми можуть справитися з цим завданням добре. Протокол повинен розподілятися на частини, а саме — на ініціатора діалогу та на інші частини (сторони), і в деяких із зазначених методів таке розподілення сприяє багаторазовому використанню (UML, мережі Петрі).

2) *Можливість моделювання діалогу (розмови), який складається з одного протоколу* — здатність вираження всієї діалогової

моделі у тому випадку, коли розмова складається з одного єдиного протоколу. Практично всі зазначені формалізми справляються з цим завданням добре, оскільки вони по суті використовуються здебільшого для цього.

3) *Можливість моделювання діалогу (розмови), який складається з багатьох протоколів* — здатність вираження всієї діалогової моделі, а саме у тому випадку, коли розмова складається більш, ніж з одного протоколу. Окрім ФІФА, всі інші формалізми можуть успішно використовуватися для об'єднання кількох протоколів в завершену модель діалогу.

Таблиця 1

ПОРІВНЯННЯ МЕТОДІВ МОДЕЛЮВАННЯ ДІАЛОГОВИХ ВЗАЄМОДІЙ МІЖ АГЕНТАМИ

Назва властивості	Протокол ФІФА-97	UML-діаграма	ДКА	Удосконалені графи Дулі	Розфарбовані мережі Петрі
Можливість моделювання протоколу	*	*	*	*	*
Можливість моделювання діалогу, який складається з одного протоколу	*	*	*	*	*
Можливість моделювання діалогу, який складається з багатьох протоколів		*	*	*	*
Можливість динамічної побудови діалогу			*	*	*
Можливість візуалізації протікання діалогу			*	*	*
Можливість установлювати відмінності між різними учасниками діалогу				*	*
Можливість виразити політику діалогу					*
Можливість підтримки паралелізму					*
Наявність засобів безпосередньої імплементації		*	*		*
Можливість багаторазового використання структури діалогів під час виконання					*

4) *Можливість динамічної побудови діалогу* — здатність формалізму створювати модель, динамічно побудовану в процесі виконання, тобто модель з'являється від протікання активізованого діалогу (розмови), і при цьому конкретний формалізм має охоп-

лювати деякі аспекти активізованої розмови та забезпечувати деякі кількісні вимірювання. Три формалізми, які базуються на моделюванні з використанням діаграм (ДКА, метод удосконалених графів Дулі та мережі Петрі), можуть в цілому легко справитися з цим завданням.

5) *Можливість візуалізації протікання діалогу* — здатність формалізму графічно предсталити процес протікання діалогу (розмови). Як і в попередньому випадку, всі методи, що базуються на графічних засобах моделювання, мають можливість забезпечити таку властивість. Що ж стосується UML, то при її використанні існує потреба в окремих нотаціях для моделювання діалогу та окремих нотаціях для моделювання процесів всередині моделі.

6) *Можливість установлювати відмінності між різними учасниками діалогу* — здатність моделі ідентифікувати різних учасників даного діалогу (розмови). Ця властивість у повному обсязі підтримується лише удосконаленими графами Дулі та мережами Петрі.

7) *Можливість виразити специфічну політику (стратегію) діалогу* — здатність діалогової моделі розмови включати специфічну політику для учасників діалогу. Тільки моделі на основі мереж Петрі з наявною можливістю формування написів та блокування можуть справитися з цим завданням.

8) *Можливість підтримки паралелізму* — здатність виразити паралелізм на рівні зразка діалогу. На сьогодні тільки дуги в мережах Петрі можуть підтримати цю ознаку повністю.

9) *Наявність засобів безпосередньої імплементації* — здатність автоматичного генерування протоколів підтримки імплементації та учасників для даної моделі діалогу. Ця можливість існує тільки в таких формалізмах, як UML, ДКА та мережі Петрі.

10) *Можливість багаторазового використання структури діалогів під час виконання* — здатність моделі забезпечити багаторазове використання сформованих структур під час виконання різними зразками даного діалогу (розмови). Тільки у випадку мереж Петрі існує можливість розповсюдження різних зразків повідомлень через різні місця однієї і тієї ж мережі, що використовується для моделювання діалогу. Слід зазначити, що в моделях на основі UML та ДКА є можливість повторного використання структур діалогів, але тільки у послідовному та синхронізованому вигляді. Як видно із проведеного аналізу, всім перерахованим вимогам задовольняє метод моделювання на основі розфарбованих мереж Петрі. Саме тому розглянемо його детальніше.

*Модельовання діалогових взаємодій агентів
на основі розфарбованих мереж Петрі*

Поняття мереж Петрі та їх основні характеристики.

Мережі типу «умова-подія» були запропоновані німецьким вченим Карлом Адамом Петрі в його дисертації «*Kommunikation mit Automaten*» в 1962 році [10]. Фактично в цьому дослідженні були сформульовані основні положення загальної теорії дискретних паралельних систем. Формальну модель мереж Петрі слід розглядати як узагальнення теорії автоматів, яке дозволяє представляти події, що здійснюються паралельно, в простій, але досить потужній для аналізу формі. На сьогодні існує досить значна кількість робіт, присвячених дослідженням мереж Петрі [4—5].

Мережі Петрі надають відповідний математичний формалізм для опису розподілених, діючих паралельно систем таким же чином, як і скінчені автомати, що виступають в якості інструментального засобу для опису послідовних систем. Зазвичай мережі Петрі описуються в термінах графічного представлення, проте вони мають і формальний математичний опис, який є незалежним від будь-якого графічного представлення. Базова (елементарна) структура мереж Петрі може бути формально визначена 5-ма змінними (**P**, **T**, **I**, **O**, **M0**),

де **P** — набір місць;

T — набір переходів;

I та **O** — вхідні та вихідні функції, які описуються як дуги та дозволяють рухатися від місць до переходів та від переходів до місць, відповідно;

M0 — *маркирування* — це вектор, що характеризує початковий стан системи за допомогою вказування кількості знаків у кожному місці в мережі.

Окрім структури мережі, існують ще й правила, які описують те, при яких умовах спрацьовують переходи, що приводить до породження нових станів системи. І хоча є ряд правил спрацювання переходів, пов'язаних з різними типами мереж Петрі, всі вони мають спільну властивість недетермінізму. Якщо дуги в мережі Петрі мають пов'язані з ними *ваги*, то перехід в мережі Петрі дозволяється (і може спрацювати), якщо:

— для кожного вхідного місця кількість знаків більша, ніж вага, яка пов'язана зі з'єднуючою дугою, та

— для кожного вихідного місця сума знаків, вже існуючих у місці, і вага, яка пов'язана із з'єднувальною дугою, рівна або менша, ніж місткість вихідного місця.

Коли перехід спрацьовує, то: 1) знаки в кожному вхідному місці зменшуються на кількість, рівну вазі його вхідної дуги, та 2) знаки в кожному вихідному місці збільшуються на кількість, рівну вазі вихідної дуги.

Після спрацювання переходу мережа Петрі має нове маркування, що характеризується новим розподілом знаків в різних місцях.

Розфарбовані мережі Петрі та переваги від їх використання при моделюванні динамічних систем. Розфарбовані мережі Петрі (РМП) були запропоновані Куртом Йенсенем у роботі [4] і по суті є графічно орієнтованою мовою загального призначення, що може використовуватися для опису, проектування та аналізу паралельних систем. РМП походять від мереж типу «предикат-перехід», які в свою чергу є узагальненням мереж «умова-подія» та мереж «місце-перехід».

Синтаксичними елементами РМП є: місця, переходи, дуги та написи (вирази). Тільки останній елемент є текстовим, решта — це графічні елементи (рис. 2). Місця подаються у вигляді кіл чи еліпсів та представляють щось на зразок контейнерів, які можуть містити мультинабори елементів (знаки) деякого конкретизованого типу (кольоровий набір). Набори знаків в місцях у РМП-моделі визначають у будь-який момент стан заданої системи.



Початкове маркування

Рис. 2. Елементи розфарбованих мереж Петрі

Переходи подаються у вигляді прямокутників і представляють можливі дії. Дуги представляють відношення між місцями і переходами, і визначають те, як змінюється стан тоді, коли відбувається дія. Дугові написи (вирази), які по суті є функціями, використовуються для того, щоб визначити кількість знаків, що переміщуються між станами. Роль блокуючих виразів — це фільтрація та обмеження можливих проявів дій. Можливі дії, що не конфліктують між собою, можуть відбуватися паралельно, і їх поява стається одночасно, тобто, як атомарні дії. Набори кольорів — це типи в місцях, і початкові маркування визначають початкову конфігурацію/стан РМП.

I, накінець, РМП включають підтримку великомасштабного моделювання за допомогою ієрархічної декомпозиції (переходів).

Таким чином розфарбована мережа Петрі має три основні компоненти:

- 1) структуру мережі із місць, переходів та дуг (як у звичайній мережі Петрі);
- 2) набір декларацій даних;
- 3) набір виразів у мережі (дугові вирази, вирази блокування та ініціалізації місць).

Розфарбовані мережі Петрі (РМП) довели ефективність свого використання в описі систем, де мають місце спільне використання ресурсів, синхронізація та паралелізм у роботі. По суті РМП є мовою, одночасно орієнтованою і на стани, і на події. Стан моделі представляються місцями, а події — переходами. Кожне місце має пов'язаний тип даних, що визначає зразок даних, які можуть знаходитися у конкретному місці. Двома основними причинами доречності використання РМП при моделюванні систем є:

1) РМП-модель представляє опис системи, що повинна бути промодельована, і може бути використана як специфікація або як презентація;

2) РМП можуть бути проаналізовані або шляхом симуляції, або шляхом використання формальних методів аналізу.

Ці переваги можуть бути підсилені іншими позитивними рисами розфарбованих мереж Петрі [4], а саме:

- 1) РМП мають графічне представлення;
- 2) РМП мають добре визначену семантику;
- 3) РМП є справді універсальними;
- 4) РМП мають дуже незначну кількість потужних базових елементів (примітивів);
- 5) РМП мають чіткий опис як станів, так і дій;
- 6) РМП мають семантику, яка будується на істинному паралелізмі, а не на розшаруванні;
- 7) РМП надають ієрархічні механізми для опису;
- 8) РМП інтегрують опис управління та синхронізації з описом маніпулювання даними;
- 9) РМП можуть бути розширені з метою включення параметра часу;
- 10) РМП стабільні в напрямі змін модельованої системи;
- 11) РМП можуть бути інтерактивно просимульовані з результатами, що подаються на РМП-діаграмі;
- 12) РМП можуть мати свої властивості, які доводяться кількома формальними методами аналізу;

13) РМП мають комп'ютерні інструментальні засоби, які підтримують їхнє графічне відображення, симуляцію та аналіз.

Доцільність використання звичайних мереж Петрі для опису систем на основі агентів. Мережі Петрі (МП) на сьогодні вже використовуються деякими авторами для моделювання систем на основі агентів. В роботі [5] виділено п'ять підходів щодо використання мереж Петрі для моделювання систем на основі агентів. Суть першого — це розгляд МП як єдиного агента. В цьому випадку МП використовується як діаграма станів, яка представляє внутрішні стани агента. Другий підхід зв'язує агента з переходом, що проявляється в системі, і вся мережа Петрі представляє мультиагентну систему цілому. В цьому випадку поведінка всієї системи задається за допомогою мережі поширення (діаграми досяжності). В рамках третього підходу поведінка кожного індивідуального агента описується за допомогою окремої МП і використовується механізм представлення мультиагентної системи у вигляді комбінації МП. Як правило, засобами, що використовуються для об'єднання мереж Петрі, є засоби злиття місць та злиття переходів, проте для цього можуть також використовуватися і ієрархічні мережі Петрі. В наступному підході агенти — це мережі Петрі і їхня взаємодія описується також за допомогою мережі Петрі, яка представляє наступний рівень в ієрархії. Останній, п'ятий, метод передбачає використання об'єктних МП. В цьому випадку знаки (маркери) розглядаються як окремі агенти. Однак, слід зазначити, що для цього підходу на сьогодні відсутнє визначення необхідної формальної семантики.

Мережі Петрі широко використовуються в багатьох аспектах проектування, аналізу та імплементації систем програмного забезпечення. Головна причина великого успіху мереж Петрі — це їхнє графічне представлення, чітка семантика і математичний формалізм, який дозволяє здійснювати формальний аналіз і перетворення [4]. Мережі Петрі успішно використовуються в моделюванні workflow, і на сьогодні існує багато вдалих спроб у цій сфері [11]. Виходячи з цього, можна сказати, що оскільки мережі Петрі відповідають вимогам для моделювання процесів workflow, вони можуть бути використані і для моделювання, симулювання, аналізу, контролю та налагодження діалогів між агентами в мультиагентних системах.

Практична реалізація процедур моделювання діалогових взаємодій агентів у голоністичних мультиагентних системах на основі розфарбованих мереж Петрі. Схеми діалогу моделюються через дослідження розмовних взаємодій між агентами для обміну

інформацією. Інформація, якою слід обмінюватися в діалоговій взаємодії, конкретизується через набір «діалогових тем». Схеми — це зразки діалогів, завдяки яким обмін та деталізація інформації здійснюється за допомогою діалогових взаємодій. Розфарбовані мережі Петрі використовуються як специфікація та інструментальний засіб для схеми діалогу на стадії проектування. Для полегшення координації та співробітництва в мультиагентних системах необхідно, щоб агенти обмінювались інформацією через канали зв'язку про цілі, наміри, результати та стан з іншими агентами. Складність такої задачі полягає в тому, що необхідно гарантувати надійний і гнучкий зв'язок між різнорідними агентами в тому сенсі, щоб:

- 1) не було будь-яких несумісностей та тупиків;
- 2) діалоги закінчувались очікуваними діями і очікуваними сподіваннями в пам'яті кожного агента;
- 3) зменшилась складність формування висновку стосовно того, які комунікативні дії стануть можливими в діалоговій послідовності.

Оскільки діалог є найбільш ефективним і прямим засобом для зв'язку, діалогове моделювання є фундаментом для генерації партнерських видів діяльності в мультиагентних системах. Діалоги — це послідовності повідомлень, якими обмінюються два або більше агентів, для досягнення специфічної мети. Для спрощення проектування модулів зв'язків агентів бажано отримати набір засобів, що називаються *засобами діалогу* (ЗД).

Пропонується розглядати *діалогову процедуру* (ДП) як набір специфікацій для інформаційного обміну між агентами, який включає відповідну зміну в станах агентів. Агенти можуть використовувати зразки для того, щоб зробити вирішення цієї задачі ефективнішим. При спілкуванні агенти можуть покладатися як на зразки діалогу, так і зразки групові поведінки. Виходячи з поняття зразків, для визначення діалогових процедур пропонується метод, що базується на схемах діалогу (в подальшому для стислості — схема).

Визначемо схему діалогу як зразок діалогових взаємодій, які конкретизують ДП та зосереджені на одній або кількох діалогових темах. Теми виділяються з прикладної області на стадії аналізу цієї області. В якості інструментального засобу для побудови схем на стадії розробки пропонується використовувати розфарбовані мережі Петрі завдяки перевагам формальних специфікацій, про що мова велась вище. Схема деталізується через мовні та діалогові взаємодії.

Схема діалогу між агентами. Проектування ДП є еволюційним процесом, який вимагає як абстрактного розмірковування та

експертизи прикладної області, так і проектування відповідного програмного забезпечення. Процес побудови схеми діалогу починається з ідентифікації агентів та визначення тем діалогу. Після цього окремі діалогові дії можуть бути агреговані з метою створення більш складних структур, які називаються «схемами» та які описують цільові колективні поведінки агентів. Схема може бути представлена за допомогою РМП, які дозволяють поряд з ншим здійснювати контроль можливих несумісностей та тупиків (рис. 3).

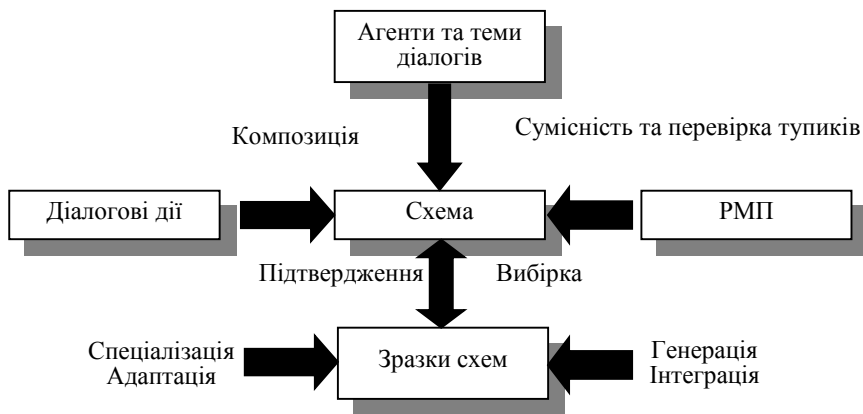


Рис. 3. Процедури побудови схеми діалогу між агентами

Діалогова взаємодія може розглядатися як набір *видів діяльності* двох або більше агентів, в яких передається інформація. Кінцевий результат взаємодії є зміна стану його учасників. Залежно від компонентів стану окремих агентів це може бути зміна в знаннях агента, його меті або інших атрибутах. Діалог завжди зосереджується на одній або кількох *темах*, кожна з яких пов'язана з набором атрибутів.

Таким чином, якщо знати мету або завдання агента, можна концептуально визначити діалогову поведінку за допомогою визначення тем для доречної діалогової взаємодії з іншими агентами.

Діалогова тема (ДТ) визначає одиницю інформації, обмін якою здійснюється за участі агентів (без посилання на інформацію щодо імплементації діалогу або на механізми контролю інформації, обмін якою здійснюється). ДТ може бути описана набором *змінних*, які мають конкретні значення і які повинні бути прийняті залученим агентом. Темі використовуються для того, щоб визначити суть стосунків всередині середовища агентів та

визначитись у відношенні до оточення. Наступна процедура дозволяє визначити діалогову тему:

Визначення 1. Діалогова тема (ДТ) може бути визначена через 3 змінні:

$$\text{ДТ: } (TP_ID, AGENTS, ARGUMENTS), \quad (1)$$

де TP_ID — ідентифікатор ДТ;

$AGENTS$ — перелік агентів, що приймають участь у темі;

$ARGUMENTS$ — перелік всіх аргументів у темі.

Приклад 1. У голоністичній виробничій системі, що працює «на замовлення», для досягнення згоди щодо таких параметрів виробничого замовлення, як строк поставки, ціна, місцезнаходження поставки, *логістичному агенту (LOA)*, *агенту виробничого планування (PAA)* та *заводському агенту (FAA)* необхідно провести кілька діалогів з різними темами. Нижче наводяться приклади доречних ДТ:

$$\begin{aligned} \text{ДТ1: } & (Prod_Tar, (LOA, PPA), \\ & (Cust_order, Cust, Loc, DD, Price)), \end{aligned} \quad (2)$$

$$\text{ДТ2: } (Mfg_Cost, (PPA, FAA), (Factory_order)(DD, Cost)), \quad (3)$$

де ДТ $Prod_Tar$ у (2) буде центром діалогової взаємодії між LOA і PPA про виробничу ціль: доставити товар $Prod$ споживачеві $Cust$ у місцезнаходження Loc , строк поставки DD , ціна товару $Price$.

Параметри продукції можуть ідентифікувати унікальні типи продукції або групу типів продукції. Значення параметрів розглядаються як агентом LOA , так і агентом PPA . Діалогова тема Mfg_cost в (3) зфокусована на діалоговій взаємодії між PPA і FAA щодо промислової вартості для заводського замовлення зі строком поставки DD . Ці *діалогові теми* абстрагують можливі переговори через набір діалогів для визначення можливих виробничих цілей та оптимальної дати поставки.

Таким чином, можна стверджувати, що діалогові теми визначають обмін інформацією на вищому рівні абстракції, проте не обговорюють визначену специфікацію і не описують *розклад взаємодій*. ДТ також не описують індивідуальні обмеження систем, але забезпечують зв'язки для схеми встановлення діалогів.

Побудова схеми діалогу в голоністичній мультиагентній системі. Схема може бути створена за допомогою ідентифікації діалогових дій, зміни внутрішнього стану та зовнішнього інформаційного обміну, що має відношення до ДТ.

Визначення 2. Схема діалогу в голоністичній мультиагентній системі (СД) складається з 4 змінних:

$$\text{СД} = (Agents, Acts, States, Arcs, TPs), \quad (4)$$

де $Agents = \{Agt_1, Agt_2, \dots, Agt_m\}$ ($m > 0$) — набір агентів, які беруть участь у діалозі; $Acts = \{Act_1, Act_2, \dots, Act_n\}$ ($n > 0$) — пакети діалогових дій, які виконуються кожним агентом із $Agents$ у деякій ситуації; $States = \{Sta_1, Sta_2, \dots, Sta_l\}$ ($l > 0$) — відповідні зміни станів кожного із агентів в $Agents$; $Arcs = Acts \times States \cup States \times Acts$ — це зв'язки між $Acts$ та $States$; $TPs = \{Tp_1, Tp_2, \dots, Tp_k\}$ ($k > 0$) — набір діалогових тем, доречних для агентів у процесі діалогу.

Приклад 2. В голоністичній виробничій системі, що базується на агентах, після отримання замовлення від споживача *логістичному агенту (LOA)* необхідно знайти інших агентів, які можуть асистувати (або допомогати); при цьому агент повинен звернутись з запитом до *локального зонного координатора (LAC)* на предмет отримання від нього інформації про те, «як співробітничати». Після цього LAC повинен звернутись до деякого *агента управління знаннями (KA)* з проханням допомоги знайти деяких підходяжих співучасників. Відповідні дії будуть проявлятися в зворотному порядку, наприклад, для LOA — це: $Agents = \{LOA, LAC1, KA1\}$, $TP = \langle 3 \text{ ким співробітничати} \rangle$. Ця схема також відповідає вимогам *агента виробничого планування (PPA)*, коли він отримує замовлення заводського агента (*FAA*) для формування віртуальної команди по виконанню виробничого замовлення, в якій $Agents = \{PPA, LAC2, KA2\}$, $TP = \langle \text{часткова інформація про продукцію} \rangle$ і $Agents = \{FAA, LAC3, KA3\}$, $TP = \langle \text{характерна інформація про частину} \rangle$, відповідно. Вони мають однакові $Acts = \{REQUES, REPLY\}$ (REQUES та REPLY в даному випадку є атомарними схемами).

Приклад 3. «Схема переговорів щодо дати поставки» у голоністичній виробничій системі, що базується на агентах, може бути утворена за допомогою ДТ₁ та ДТ₂ із *Прикладу 1* (рис. 4).

Схема:

$$Agents = \{LOA, PPA, FAA\}, TP_s = (TP_1, TP_2).$$

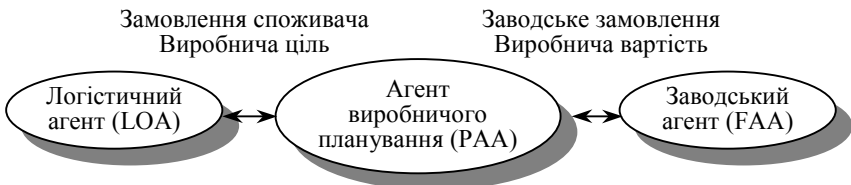


Схема переговорів щодо дати поставки

Рис. 4. Приклад орієнтованої на ціль схеми: переговори щодо дати поставки

Представлення схеми діалогу за допомогою розфарбованих мереж Петрі. Вибір РМП як методу формального опису був продиктований необхідністю природної підтримки паралельності та синхронізації діалогів, а також тим, що вони (тобто РМП) використовують «розфарбовані» маркери (знаки) як важливі схемні абстракції та засоби інтеграції. Маркери РМП можуть бути довільного кольору (типу) та складності. Завдяки тому, що нотації РМП базуються на функціональній мові програмування SML, дугові вирази дозволяють стисло подавати складні маніпуляції над маркерами. Для підтримки модульності РМП забезпечують такі ознаки, як змінні переходи та місця злиття. Змінні переходи (нотаційна сумісність, розроблена для модульного представлення великих і складних моделей) використовують концепцію портів. Коли змінний перехід оголошений, місця його входу і виходу розглядаються як порти. Змінний перехід потім поширюється на інші частини мережі і додаткові (внутрішні) місця і переходи можуть бути продекларовані там, де це необхідно. Однак всі «атомарні» переходи моделюють комунікативні дії на одному і тому ж рівні абстракції, хоча вони можуть з'являтися в різних та ієрархічно організованих частинах. Місця злиття використовуються для того, щоб уникнути групування дуже багатьох вхідних та вихідних дуг.

Конструкція атомарної схеми діалогу. Діалогові дії в схемі представляються як *переходи* в РМП. Стан схеми подається як комбінація *місць* РМП і *структурованого маркера*, що управляє *повідомленнями*. Конструкція схеми може бути реалізована через підтвердження, спеціалізацію, адаптацію, інтеграцію та переробку. Дугові вирази можуть бути двох типів: (1) вирази вхідних дуг визначають, які і як багато маркерів вилучається з місць вводу; (2) вирази вихідних дуг визначають, які і як багато маркерів створюється і вводиться у вихідних місцях.

Наведена нижче процедура описує кроки в рамках атомарної схеми діалогу:

1) для кожного агента кожна комунікативна дія в схемі зображається переходом в РМП. Переходи, які представляють дії, що виконуються цим же агентом, розміщуються горизонтально;

2) місця і дуги додаються між переходами і приєднуються до них. Таким чином моделюються не виражені явно інформаційні обміни, які мають місце між наступними комунікаційними діями, що виконуються кожним агентом;

3) додається інформаційний обмін, представлений колективними місцями, які з'являються між агентами в рамках ДТ.

Побудова композитної схеми діалогу. У випадку композитної схеми її побудову полегшують ієрархічний формалізм РМП

та об'єктивно-орієнтовані методології (ООМ). При цьому можна використовувати агрегацію, класифікацію, підтвердження та спадковість як головні засоби для побудови композитної схеми. Рис. 5 ілюструє приклад композитної схеми «Одержання інформації» за допомогою залучення атомарних схем REQUEST, REPLY та ACK. Три випадки, розглянуті в Прикладі 2, об'єднуються в РМП за допомогою розміщення кольорових наборів у місцях і переходах мережі. X, Y, Z є змінні агентів учасників.

Набір кольорів для переходів у мережі Петрі:
 $T1 = T2 = \text{REQUEST}(\text{"intent"}, \text{"data_info."})$
 $T3 = T4 = \text{REPLY}(\text{"intent"}, \text{"result_info."})$
 $T5 = \text{ACK}(\text{"result_info"})$
 Набір кольорів для місць у мережі Петрі:
 $C(p1) = \{<\text{Cus_Order}, \text{LOA}>, <\text{Fac_Order}, \text{PPA}>, <\text{Mfg_Order}, \text{FAA}>\}$
 $C(p2) = C(p1)$
 $C(p3) = \{<\text{LOA}, \text{PPA}>, <\text{PPA}, \text{Product Model}>, \text{PPA}, \text{Feature Model}>\}$
 $C(p4) = \{<\text{LAC1}>, <\text{LAC2}>, <\text{LAC3}>\}$
 $C(p5) = \{<\text{LAC1}, \text{Cus_Order}>, <\text{LAC2}, \text{Fac_Order}>, <\text{Mfg_Order}, \text{FAA}>\}$
 $C(p6) = \{<\text{KA1}>, <\text{KA2}>, <\text{KA3}>\}$
 $C(p7) = C(p1)$
 $C(p8) = \{<\text{PPA}>, <\text{Product Model}>, <\text{Feature Model}>\}$
 $C(p9) = C(p5)$
 $C(p10) = C(p8)$

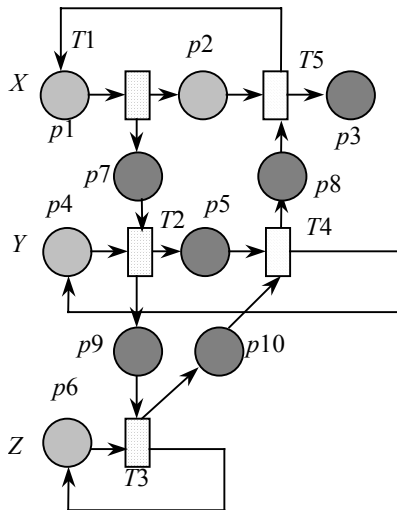


Рис. 5. Приклад композитної схеми, представлена за допомогою РМП

Для того, щоб зробити цю схему такою, яку можна повторно використовувати, обслуговувати та гнучкою до діалогу, синхро-

нізаційні обмеження відділені від внутрішньої частини кожної схеми. Саме цьому кожна схема має інтерфейс (INTERFACE) та тіло (BODY). Інтерфейси використовуються для з'єднання двох і більше схем під час виконання діалогу та для полегшення об'єднання кількох підмереж, що дозволяє перетворювати під-схеми в одну комплексну схему.

Клас схем використовується для того, щоб створити актуальну схему під час виконання. Наприклад, з класу схеми «*Одержання інформації*» клас з трьома змінними X , Y , Z та темою діалоговою темою TP , якою є замовлення O (Приклад 2, наведений вище), можна в разі потреби отримати три схеми: схему «*Пошуку агентів для співпраці*», схему «*Часткового пошуку інформації*» та схему «*Одержання інформації про характеристики*». При цьому можна вести мову про деяку абстрактну схему, з якої може успадковуватися більш конкретна схема.

Класи схем діалогу — це описи на основі РМП того, які групи агенти діють у конкретних ситуаціях. Клас схем діалогу визначає доступні процедури діалогу, механізм їхнього контролю, а також локальну базу знань, яка обслуговує стан діалогу. *Схема діалогу* може бути побудована шляхом об'єднання класів схем. Класи схеми використовуються для перетворення концептуальної схеми в модулі шляхом інкапсуляції.

Після завершення розгляду процедур побудови схем діалогу між агентами коротко зупинимося на її перевірці:

1) *Сумісність та перевірка тупиків*. Отримане РМП-представлення схеми дозволяє перевірити логічну сумісність, і, зокрема, відсутність тупикових ситуацій через симулювання та S-інваріантний аналіз. Після визначення початкового маркування, відповідно з яким кожному джерелу буде поставлено у відповідність один маркер, РМП може бути просимульована для перевірки логічної сумісності вимог та оцінки поведінки розробленої схеми.

2) *Механізм управління*. Діалог слід розглядати як набір *видів діяльності* двох або більше агентів, в яких передається інформація. Кінцевим результатом діалогу є зміна стану його учасників. Залежно від компонентів стану агентів-учасників це може бути зміна в знаннях агента, зміна цілей або інших атрибутів.

Схема діалогу на основі РМП може бути вибрана і адаптована як результат довіри, бажань та мети агентів та статусу ресурсу. Процес підбору конкретної схеми діалогу запускається тоді, коли агент отримує запит на виконання нової мети (з пов'язаним завданням). Такий запит може надійти від кількох джерел: користувача; через опис місії; неочікуваних подій, на які необхідно реагувати; інших агентів;

через запити або команди, а також з власного «міркування» агента (прояв субцілі). Коли з'являється нова мета, вона розміщується в списку «Що робити», який називається *порядком денним*.

Агент може мати кілька активних діалогів одночасно. Механізм управління діалогом існує для того, щоб дозволити агентам призупинити поточний діалог у той момент, коли необхідно почекати, щоб інші агенти досягли конкретних стадій. В результаті сценарії діалогу динамічно формують ієрархії діалогів. Якщо агент, який має ціль, ініціює діалог з іншим агентом шляхом вибору схеми, то після цього інші зразки відповіді агентів обмежені для цієї схеми. Механізм управління буде тільки виконувати «внутрішкємні рішення» шляхом виконання правил переходів у РМП.

В моделі діалогу на основі РМП — схеми цілі та стани агентів, що приймають участь у діалозі, представляються маркерами (знаками) з різними типами, що називаються *кольорами*. Процеси появи діалогу моделюються за допомогою зразка «розпізнання — переключення — дії» відповідно до поточних цілей та ситуації, в яких знаходяться агенти.

Якщо агент ініціює діалог за допомогою використання індивідуальної комунікативної дії, тоді отримуючий агент є «вільним» у плані того, як він має реагувати. В цьому випадку механізм зробить «*вибір активної схеми*» за допомогою пошуку в таблиці схем мети (*INTENT_SCHEMA_TABLE*) з відповідного класу схем. Після вибору класу схеми він створює зразок класу схеми діалогу. Фактичний діалог підтримує поточний стан спілкування, тобто набір діалогових змінних, чії значення управляються діалоговими правилами та різною історичною інформацією, яка нагромаджується протягом виконання діалогу. Кожний клас схеми діалогу описує спілкування з точки зору групи агентів, а не індивідуального агента.

Література

1. Bradshaw J. M., Dufield S., Benoit P., and Woolley J. D. *Kaos: Toward an industrial-strength open agent architecture* In J.M.Bradshaw, editor. *Software Agents* AAAI/MIT Press, 1998. — P. 11-87.
2. Dooley R. A. *Appendix B — Repartee as a graph, An Anatomy of Speech Notions*, 1976. — P. 348 — 358.
3. FIPA. FIPA Spec2-1999. Agent Communication Language Draft, Version 0.1, Available at FIPA <http://www.ipa.org/specifications/index.html>, 16 April 1999.
4. Jensen K. *Coloured Petri Nets — Basic Concepts, Analysis Methods and Practical Use. Volume 3, Practical Use*. Monographs in Theoretical Computer Science. Springer-Verlag, 1997. — 221 p.

5. Lomazova I. Multi-Agent Systems and Petri Nets. In *Int. Workshop on Distributed Artificial Intelligence and Multi-Agent Systems (DAUMAS'97)*, St.Petersburg, Russia, Jun-97. — P.147-152.

6. Nwana H. S. Software Agents: An Overview, Knowledge Engineering Review, Vol. 11, No 3, Cambridge University Press, Sept 1996. — pp. 1-40.

7. Object Management Group OMG Unified Modeling Language Specification, version 1.3 Available at [OMG ftp://ftp.omg.org/pub/docs/ad/99-06-09.pdf](http://ftp.omg.org/pub/docs/ad/99-06-09.pdf), June 1999.

8. Odell J., Parunak H.V.D., and Bauer B. Representing agent interaction protocols in UML In P Clancanni and M Wooldndge, editors, *Proceedings of the First International Workshop on Agent-Oriented Software Engineering 2000* Spnnger-Verlag, 2000 Available online at http://www.jamesodell.com/Rep_Agent_Protocols.pdf

9. Parunak, H.Van Dyke. Chapter 4 — Applications of Distributed Artificial Intelligence in Industry, *Foundations of Distributed Artificial Intelligence*, G. M. P. O'Hare and N. R. Jennings (eds.), John Wiley & Sons, 1996. — P. 139-163.

10. Petri C. A. *Kommunikation mit Automaten*. Bonn: Institut fur Instrumentelle Mathematik, Schriften des IIM Nr. 2, 1962. Also in English translation: New York: Griffiss Air Force Base, Technical Report RADC-TR-65-377, Vol.1, Suppl. 1, 1966. — 348 p.

11. W.M.P van der Aalst. Three good reasons for using a petri-net-based workflow management system. In S. Navathe and T. Wakayama, editors, *Proceedings of International Working Conference on Information and Process Integration in Enterprises (IPIC 96)*, November 1996. — P. 179—201.

Стаття надійшла до редакції 20.04.06

УДК 332.025.12

В. Б. Кириленко, доцент, канд. екон. наук,
Київський національний економічний університет
імені Вадима Гетьмана

НОВІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ОРГАНІЗАЦІЇ СИСТЕМИ ВНУТРІШНЬОГО КОНТРОЛЮ

У статті розглянуті питання пов'язані з впровадженням автоматизованих інформаційних технологій в процедури внутрішнього контролю. Розглянуті характерні риси і особливості побудови корпоративних інформаційних систем. Виділені найбільш розповсюджені погрози і злочини пов'язані з втручанням у роботу комп'ютерів. Зроблені відповідні висновки.

КЛЮЧОВІ СЛОВА: автоматизована інформаційна система, автоматизована інформаційна технологія.