

УДК 004.942

DOI: 10.15587/1729-4061.2021.239212

Розробка архітектури комбінованого підходу до імітаційного моделювання систем з паралелізмом

О. О. Супруненко

Описані парадигми і графоаналітичний інструментарій для побудови засобів імітаційного моделювання та формування архітектури комбінованого підходу вивчення динамічних властивостей систем з паралелізмом. Представлене розширення формальної мови мереж Петрі, яке має більшу потужність моделювання ніж WF-мережі. Властивості ієрархічних мереж Петрі використовуються для синтезу цілісної моделі.

Підґрунтям для комбінованого підходу до імітаційного моделювання систем з паралелізмом виступили дискретно-подієве моделювання та моделювання динамічних систем, які дозволяють відображати кількісні та якісні характеристики елементів в одній моделі. На їх основі запропоновано графоаналітичні інструментальні засоби, які надають можливість опису модельованої системи, дотримуючись принципу структурної подібності. Вони мають засоби динамічної імітації, що спрощує візуальний аналіз та коригування моделі. Також запропонований інструментарій передбачає аналіз динамічних властивостей моделі, який дозволяє виявляти накопичені явища, що можуть призвести до непередбачуваного функціонування системи.

Запропоновано концептуальну модель синтезу та аналізу систем з паралелізмом, яка передбачає побудову складових моделі на основі архітектурного рішення. Їх поетапний аналіз та формування цілісної моделі програмної системи здійснюється з використанням мережевого представлення, за матричним описом якого розраховуються інваріанти. Аналіз інваріантів дозволяє отримати динамічні властивості моделі та визначити локалізацію конструкцій, що приводять до критичних ситуацій при їх виявленні.

Побудовано архітектуру комбінованого підходу до імітаційного моделювання систем з паралелізмом, яка забезпечує вивчення їх динамічних властивостей для підвищення надійності функціонування програмних систем.

Ключові слова: архітектура комбінованого підходу, формальні мови мереж Петрі, засоби динамічного моделювання.

1. Вступ

В умовах зростаючої складності програмних систем [1] постає задача прийняття обґрунтованих проектних рішень на етапах проектування, відлагодження та супроводу програмних проектів. Складність програмного забезпечення, зокрема структурна та функціональна складність, обумовлена наявністю численних паралельних та конкуруючих процесів, які взаємодіють при функціонуванні програмного продукту. Для вивчення статичних та динамічних властивостей програмних систем з паралелізмом одним з найпотужніших інструме-

нтів дослідження є імітаційне моделювання. Його засоби дозволяють описати досліджувані системи на різних рівнях абстракції, враховуючи їх кількісні та якісні характеристики [2].

При описі та аналізі програмних систем часто будують кілька моделей [3], які відображають різні аспекти досліджуваної системи. Коли висновки, зроблені на основі цих моделей, є протирічними або неповними виникає необхідність будувати комбіновану модель, що сполучує засоби різних методологій імітаційного моделювання [3, 4]. Інструментарій, на основі якого можливо будувати комбіновані моделі, що сполучують кілька методологій, слабо розвинутий [5, 6]. Так, при проектуванні програмних систем застосовуються моделі, які описують різні аспекти їх побудови та функціонування. Наприклад, діаграми UML [7], які застосовуються для представлення та аналізу проектних рішень, дозволяють описати структурні аспекти на діаграмах класів, пакетів, компонентів та ін. Поведінкові аспекти описують діаграмами діяльності та автоматів, а також діаграмами сценаріїв використання. Динамічних аспекти взаємодії представляються на діаграмах послідовності, комунікації, огляду взаємодії та синхронізації. Але відсутність засобів динамічної імітації, а також неузгодженість рівнів деталізації надають не дозволяють проаналізувати динамічні властивості компонентів та всієї системи, що є проблемою при проектуванні програмних засобів [6]. Тому актуальною є задача розробки єдиного інструментально-методологічного базису комбінованих імітаційних моделей для дослідження складних систем з паралельними та конкуруючими процесами.

2. Аналіз літературних даних та постановка проблеми

При проектуванні програмних засобів (ПЗ) використовують набір схем та діаграм [7], наприклад діаграми UML, кожна з яких відображає певні аспекти розроблюваного ПЗ. Але специфіка діаграм, яка не передбачає візуальної динамічної імітації при аналізі функціонування єдиної моделі ПЗ, обумовлює складності при оцінці та корекції програмного проекту. Це може привести до не виявлення певної частини помилок чи конструктивних рішень, які у подальшому можуть обумовити виникнення непередбачуваного функціонування програмного продукту. Тому невирішеними є питання, пов'язані з засобами представлення та аналізу загальної моделі програмної системи, що насамперед обумовлено великою розмірністю даної задачі та різноманітністю діаграм в рамках одного підходу, а іноді й використанням діаграм, які розроблені у різних підходах. Натомість у роботах Хоара та Дейкстри зазначається, що програмні засоби повинні мати «точно визначену поведінку» [7] і їх можна аналізувати, як математичні об'єкти. Ця ідея використовується у роботі [8] для моделювання керування потоками робіт на основі Workflow-інтерпретації мереж Петрі (WF-мереж). Але WF-мережі мають суттєві обмеження, які не дозволяють у повній мірі використовувати їх при моделюванні та аналізі моделей програмних систем. Тому доцільним є проведення дослідження щодо розробки підходу та інструментальних засобів для моделювання та аналізу систем з паралелізмом, до яких належать і програмні системи.

Для визначення підґрунтя для розробки підходу до моделювання програмних систем розглянемо імітаційне моделювання, до якого належить і інструментарій на основі WF-мереж. Імітаційне моделювання дозволяє на порядок, а інколи й більше, підвищити ефективність розв'язання прикладних задач [5]. **Ошибка! Источник ссылки не найден.** За сучасними дослідженнями імітаційне моделювання має чотири парадигми [6]: моделювання динамічних систем, дискретно-подієве (процесно-орієнтоване) моделювання, системну динаміку та агентне моделювання. У моделюванні динамічних систем досліджується система описується множиною змінних стану та алгебраїчними аналогами диференціальних рівнянь різного виду, які задані для змінних і описують їх перетворення в часі. Змінні частіше за все мають фізичний сенс і є неперервними. Моделювання динамічних систем належить до концепції якісного моделювання [4].

Дискретно-подієве моделювання дозволяє описати основні стани системи та умови переходу від поточних до наступних станів, враховуючи складні причинно-наслідкові зв'язки [8]. Ця парадигма імітаційного моделювання також надає можливість відтворити можливі дії системи в середині та в оточуючому середовищі у послідовні моменти часу. Вона застосовується у прикладних галузях виробництва, транспортних системах та ін. [4].

З допомогою системно-динамічного підходу описують акумулювання характеристик досліджуваних процесів, які характеризуються нелінійністю, що обумовлено ефектами запізнювання [4]. У моделях системної динаміки в якості змінних виступають узагальнені кількісні характеристики потоків, що обумовлює належність системної динаміки до концепції кількісного моделювання. Системно-динамічні моделі застосовуються для розв'язання задач стратегічного управління великими організаційними системами, вони застосовуються лише на макрорівні і не мають засобів для детального опису складних систем. На противагу системній динаміці [4] засоби дискретно-подієвого моделювання (рис. 1) дозволяють описати причинно-наслідкові зв'язки між елементами моделі як на метарівні, так і на макро- та мікрорівнях моделювання [4, 9]. Ці засоби відображають кількісні та якісні характеристики елементів системи, переводячи їх у асоціативно-числову форму.

Агентне моделювання призначене для дослідження децентралізованих систем, його відмінністю є розгляд об'єктів систем як активних сутностей з причетною їм поведінкою. Також агентні моделі є адаптивними, що надає можливість відображення пристосування моделі до зміни зовнішніх умов [4]. Моделі у цій парадигмі імітаційного моделювання будуються за принципом «знизу-вверх», тому при їх побудові складніше врахувати загальні характеристики модельованої системи, зокрема при проектуванні програмної системи.

З розповсюджених інструментів імітаційного моделювання особливу групу утворюють графові інструменти [2], які дозволяють описувати систему у певний момент часу та моделювати її поведінку під впливом зовнішніх факторів. Дані інструменти належать до парадигми дискретно-подієвого моделювання і дозволяють описати стани, в які переходить система під впливом певних факторів, характеристики цих станів та умови переходу. Інша парадигма імітацій-

ного моделювання [4] – моделювання динамічних систем – дозволяє моделювати обмежену кількість станів (рис. 1), що є обмежуючим фактором моделювання систем з паралелізмом. Натомість сполучення дискретно-подієвого моделювання та моделювання динамічних систем надає можливість поєднати у інструментальних засобах графоаналітичні властивості. Вони поєднують графічне відображення динаміки функціонування модельованої системи з однозначним математичним описом її структури та динамічних ознак, який використовується для визначення та локалізації непередбачуваної поведінки моделі. Таким чином, інструментарій, створений на базі двох даних парадигм дозволяє відтворювати поведінку моделі у часі та розраховувати динамічні характеристики моделі. Такий підхід допомагає виявляти накопичені явища, що проявляються лише за певних умов, але можуть призвести до непередбачуваної поведінки, виявлення якої є складною проблемою при аналізі моделей паралельних і розподілених програмних систем [9].

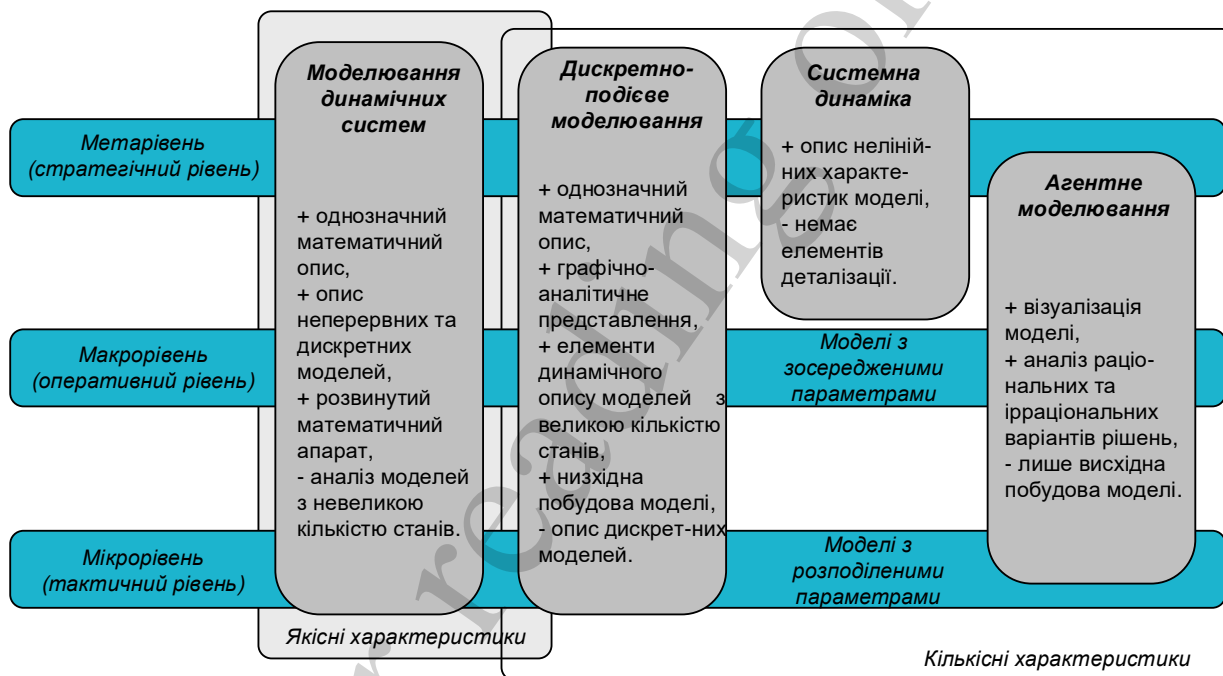


Рис. 1. Переваги та недоліки парадигм імітаційного моделювання для задач моделювання систем з паралелізмом

Одним з представників, побудованих на сполученні дискретно-подієвого моделювання та моделювання динамічних систем, є теорія мереж Петрі (Petri Nets, PN) [10]. Вона дозволяє не тільки описати причинно-наслідкові зв'язки у моделях з паралельними та конкуруючими процесами, а і проводити дослідження моделей [6, 11]. Мережі Петрі сполучують зручне графічне представлення з математично строгим описом моделі [12, 13], що обумовлює їх використання при автоматизованому аналізі побудованої моделі (критичні властивості, виключні ситуації).

Важливими характеристиками програмної системи є передбачуваність поведінки та працездатність всіх її елементів. Для дослідження цих характеристик

застосовують обмежені мережі Петрі, до яких належать безпечні, оціночні мережі Петрі, WF-мережі (Work Flow Petri Nets, *WF*) та часові мережі Петрі [14]. Для даних інтерпретацій та модифікацій мереж Петрі доведені властивості живості (досяжності), збережуваності та несуперечливості [8]. Для компонування складових систем з паралелізмом в єдину модель запропоновані вкладені [15] та взаємодіючі [16] мережі Петрі. Але механізми імітації динамічних характеристик у вкладених мережах Петрі досить складні. Так, при імітації моделі в цілому складно контролювати активні потоки подій, оскільки приходится переходити з одного рівня деталізації на інший. Для взаємодіючих мереж Петрі, зокрема для їх найбільш дослідженої WF-інтерпретації, не доведена умова живості [16].

У роботі [17] запропоноване наближене доведення бісимуляції ресурсів для однолічильникових мереж Петрі, яке знайшло розвиток у роботі [18] для мереж Петрі з λ -переходами. Дану ідею можливо використовувати при зменшенні розмірності моделі з паралелізмом для її дослідження аналітичними методами. Але використання λ -переходів у інструментальних засобах графічного відображення динаміки функціонування моделі нівелюють переваги візуального аналізу. Тому у розроблюваному підході та засобах моделювання потрібно передбачити опис допоміжних функцій моделі, що зазвичай описуються λ -переходами, у явній формі.

У роботі [19] до недоліків моделювання корпоративних додатків мережами Петрі відносять відсутність врахування часу, натомість час має велике значення при імітації функціонування даних систем. Також авторами [19] зазначається як недолік застосування правил мереж Петрі вільного вибору, коли активованими одночасно можуть бути кілька переходів. Для вирішення даних проблемних моментів автори [19] пропонують сполучувати мережі Петрі з ланцюгами Маркова та теорією масового обслуговування. Але потенціал аналітичного дослідження імітаційних моделей теорії мереж Петрі не розглядається.

Деякі модифікації обмежених мереж Петрі добре досліджені, але потребують адаптації для опису програмних систем. Так, найбільш розповсюджені WF-мережі [11] застосовують для опису потоків робіт, але вони не дозволяють вирішити усіх задач, які стоять при аналізі моделей програмних систем. Зокрема, вони не підтримують конструкції типу «контроль доступу до критичної секції» та «передача ресурсу з обмеженим буфером» [14] у задачі виробника/споживача. Це обумовлено відсутністю контролюючих вершин місця, які мають бути розміченими у початковій розмітці, що суперечить визначенню та основним властивостям WF-мереж [8].

При роботі з моделлю в процесі її аналізу потрібно виявляти локалізацію та причину виникнення конфліктної ситуації, що потребує відповідних засобів. При проектуванні програмних систем [6, 7] опис та візуальний аналіз проектних рішень проводять на діаграмах та схемах. Але відсутність в них динамічної складової, яка дозволяє відтворювати динамічні особливості функціонування при аналізі, ускладнює ситуацію. Тому при формуванні інструментальних засобів динамічного моделювання систем з паралелізмом є необхідність у вдосконаленні інструментів імітаційного моделювання.

Таким чином, потребують подальшого розвитку інструментальні засоби для опису та аналізу моделей програмних систем, які дозволяють опрацьовувати моделі великої розмірності. Потрібно сформувавши підхід, який об'єднує засоби графічної побудови моделей систем з паралелізмом, їх аналізу та корегування на єдиному представленні. Для імітації та візуального аналізу потребують розширення WF-мережі та вдосконалення механізми візуального аналізу моделей.

3. Мета та задачі дослідження

Метою дослідження є розробка архітектури комбінованого підходу до імітаційного моделювання систем з паралелізмом. Це дасть можливість визначати динамічні характеристики складових та системи з паралелізмом в цілому для підвищення надійності функціонування програмних систем.

Для досягнення мети поставлені наступні завдання:

- описати особливості та обмеження формальних мов мереж Петрі для побудови інструментальних засобів комбінованого підходу імітаційного моделювання систем з паралелізмом;
- проаналізувати властивості обмежених мереж Петрі для аналізу динамічних властивостей імітаційних моделей з паралелізмом;
- сформувавши принципи та концептуальну модель для розробки архітектури комбінованого підходу до імітаційного моделювання систем з паралелізмом.

4. Матеріали та методи дослідження

Формування підходу та інструментальних засобів для моделювання систем з паралелізмом потребує визначення потенціалу та обмежень існуючих засобів імітаційного моделювання. На їх основі формується комбінований підхід до моделювання програмних систем з паралелізмом, який в єдиному представленні враховує особливості опису структури та аналізу функціонування досліджуваних систем.

У теоретичному дослідженні використані парадигми імітаційного моделювання, які дозволили сформувавши базис для комбінованого підходу до моделювання систем з паралелізмом на основі моделювання динамічних систем та дискретно-подієвого моделювання. Теорія формальних мов мереж Петрі використовується для визначення потенціалу та обмежень інтерпретацій і модифікацій мереж Петрі, що застосовуються при розробці інструментальних засобів імітаційного моделювання систем з паралелізмом. Теорія обчислювальних систем дозволила формалізувати основні етапи дослідження моделей програмних систем з паралелізмом. Алгебра процесів та компонентно-орієнтований підхід до проектування програмних систем надав можливість побудувати архітектуру комбінованого підходу до імітаційного моделювання систем з паралелізмом.

Гіпотеза дослідження: розширення WF-мереж властивостями обмежених інтерпретацій та модифікацій мереж Петрі дозволить будувати структурно-подібні моделі систем з паралелізмом, враховуючи якісні та кількісні параметри. Матричне представлення складових систем з паралелізмом обмеженими мережами Петрі можливо використовувати для визначення динамічних властиво-

стей мережевих моделей. Аналіз розмітки при імітації дозволить підвищити ефективність аналізу динаміки функціонування мережевих моделей.

5. Результати дослідження підґрунтя та розробки архітектури комбінованого підходу для імітаційного моделювання систем з паралелізмом

5.1. Формальні мови мереж Петрі, їх особливості та обмеження

Мережі Петрі (PN) породжують клас формальних мов мереж Петрі. Мова мереж Петрі $\gamma=(PN, \Sigma, \sigma, s_0, M_F)$ представляється множиною рядків над абеткою Σ символів, що позначають кінцеву множину вершин переходів функцією помічення $\sigma:T\rightarrow\Sigma$, а також початковий стан s_0 та множину кінцевих станів M_F . Таку мережу Петрі називають поміченою.

Абетка мов мереж Петрі пов'язана з вершинами переходів, таким чином функція помічення ставить у відповідність вершини переходів символам абетки формальної мови: $\sigma:T\rightarrow\Sigma$. Послідовність запусків переходів мережі породжує рядок мови мереж Петрі. В залежності від множини підсумкових маркувань розрізняють формальні мови мереж Петрі L, G, T та P-типу [14].

Мови мереж Петрі базуються на принципі строгої монотонності [20], оскільки якщо вершина переходу є дозволеною для розмітки μ_i , та вона буде дозволеною і для розмітки $\mu_j > \mu_i$, причому якщо $\mu_i \xrightarrow{t} \mu'_i$ і $\mu_j \xrightarrow{t} \mu'_j$, то $\mu'_j \geq \mu'_i$. Це говорить про скінченність покриваючого дерева для мережі Петрі, а також про розв'язуваність проблем покриття (живості), підтримки управляючого стану та обмеженості.

Модифікація WF-мереж описується класом формальних мов мереж Петрі L-типу (мовою цілком закінчених мереж Петрі) [14], який є найбільш обмеженим. Клас мов L-типу замкнений по відношенню до таких важливих для моделювання видів композиції, як конкатенація, об'єднання, перетин, обернення, паралельна композиція та регулярна підстановка [20]. Тому мережі Петрі, що описуються формальними мовами L-типу є перспективними для створення інструментальних засобів моделювання програмних систем.

Але у мережах Петрі, що описуються мовами L-типу, не передбачений запуск моделі з довільної розмітки, яка відповідає проміжному стану цільової системи, а також безпечний доступ до критичної секції та подібні конструкції. Класи мов мереж Петрі тісно пов'язані між собою, зокрема будь-яка мова G-типу входить у клас мов L-типу $G \subseteq L$. Таким чином, обмеження мов L-типу дозволяє подолати застосування до них частини властивостей формальних мов G-типу, які надають можливість гнучко працювати з проміжними розмітками. Мови G-типу (клас вільних мов мереж Петрі) передбачають роботу з набором будь-яких початкових розміток, а також не мають обмежень на кінцеві розмітки. Також клас мов G-типу описує ієрархічні мережі Петрі [10], які дозволяють отримувати модель шляхом злиття окремих її фрагментів. Це важлива властивість, яка використовується при проектуванні програмних систем.

Розширенням мови мереж Петрі L-типу елементами мови G-типу є L_G -мова, що описує мережу Петрі $PN=(P, T, K, Q, R)$ з поміченням вершин переходів

$\sigma: T \rightarrow \Sigma$, початковим маркуванням μ_0 , проміжним маркуванням μ' та множиною підсумкових маркувань M_F , таких, що

$$L_{L_G} = \left\{ \sigma(t_j) \in \Sigma \mid t_j \in T \wedge \left(\delta(\mu_0, t_j) \vee \delta(\mu', t_j) \right) \in M_F \right\}. \quad (1)$$

Таким чином, мова мереж Петрі L_G -типу дозволяє проводити моделювання, починаючи з будь-якої розмітки, яка належить до кінцевої множини підсумкових маркувань мережі M_F . Також мережі Петрі, які описуються мовами L_G -типу, можуть мати скінченну множину контролюючих вершин, розмітка яких має відновлюватись по закінченні сеансу імітації моделі. Подібні моделі можуть бути перетворені до моделей, що описуються формальними мовами L -типу, але такі моделі будуть порушувати структурну подібність по відношенню до модельованих систем, що ускладнить їх аналіз. Також перетворені моделі будуть мати значно більшу кількість елементів, що збільшить складність розрахунку їх динамічних властивостей.

5. 2. Властивості обмежених мереж Петрі для аналізу динаміки систем з паралелізмом

До обмежених мереж Петрі належать [14] безпечні, оціночні та часові мережі Петрі, WF-мережі (Work Flow Petri Nets, *WF*). Властивості мереж Петрі поділяють на статичні та динамічні [13]. Статичні властивості мереж Петрі пов'язані зі структурними особливостями їх побудови. При аналізі статичних властивостей підкласу оціночних мереж Петрі – управляючих мереж Петрі – розглядається структура мережі Петрі разом з управляючими функціями [13].

При перевірці динамічних властивостей мереж Петрі розглядається функціонування моделі та проводиться аналіз повних і часткових послідовностей переходів. До динамічних характеристик належать [8, 13] живість, досяжність, безпечність (обмеженість), збережуваність, безконфліктність (бездефектність для WF-мереж).

У роботах Хека [21] розглядалися властивості живості та безпечності, зокрема поняття живості пов'язувалось з відсутністю тупикових ситуацій, а поняття безпечності – з відсутністю непередбачуваного функціонування моделі. Проблеми тупиків та непередбачуваної поведінки у мережевих моделях виробничих систем, пов'язувались з конструкціями вільного вибору. Тобто коли вершина місця є входною до кількох вершин переходів, виникає конфліктна ситуація, яка вирішувалась [21] шляхом надання вибору будь-якої з вершин переходів для подальшого запуску.

Визначення 1. Мережа Петрі є живою тоді, коли вона при будь-яких можливих послідовностях спрацьовування верши переходів σ_i не має неживих розміток, тобто $\mu_i \notin M_{\max}$.

Задача доведення властивості досяжності може бути зведене до доведення живості [21]. Доведення властивості досяжності є важливим, оскільки до нього може бути зведено більшість задач аналізу мереж Петрі [14].

Визначення 2. Задача досяжності для мережі Петрі PN з розміткою μ полягає у визначенні досяжності розмітки μ' , тобто $\mu' \in M(PN, \mu)$.

Досяжність розмітки μ_n з деякої початкової розмітки μ_0 забезпечується існуванням послідовно спрацьовуючих вершин переходів $\sigma = t_1, t_2, t_3, \dots, t_n$, в результаті роботи яких досягається розмітка μ_n , що можна записати як $\mu_0[\sigma] > \mu_n$. Множина всіх можливих досяжних розміток з μ_0 $M(PN, \mu_0)$ відповідає множині всіх можливих послідовностей $\Omega(PN, \mu_0) = \{\sigma_1, \sigma_2, \dots, \sigma_s\}$.

Властивість безпечності мереж Петрі стосується обмеження кількості міток, що можуть перебувати у вершині місця одночасно. Умова безпечності для безпечних мереж Петрі вимагає, щоб у вершині місця не перебувало більше однієї мітки, тобто $m(p_\varepsilon) = n_\varepsilon, n_\varepsilon \in \{0, 1\}$. Безпечні мережі Петрі застосовуються для детального опису паралельних процесів, наприклад, у випадку, коли треба контролювати, щоб наступний набір процесів міг відбутися тільки по закінченню попереднього. Безпечність у класі безпечних мереж Петрі є частинним випадком властивості обмеженості, яка передбачає встановлення певного обмеження на максимальну кількість міток, що одночасно перебувають у досліджуваній вершині місця.

Визначення 3. Вершина місця $p_\varepsilon \in P$ мережі Петрі з початковою розміткою $\mu_0 \in k$ -обмеженою (k -безпечною), якщо $\forall \mu' \in M(PN, \mu_0) : \mu'(p_\varepsilon) \leq k$.

У безпечних мережах Петрі всі вершини місця є 1-обмеженими. В оціночних мережах Петрі допускаються вершини місць що мають різні обмеження, таку мережу називають k -обмеженою [13] за вершиною місця, яка має найбільше обмеження $k = \max(k_1, k_2, \dots, k_n)$. Властивість обмеженості мереж Петрі дозволяє моделювати з їх допомогою процеси, які залежать від кількісних показників, наприклад, накопичення певних ресурсів.

Задача досяжності може бути поставлена, як задача досяжності тупикової розмітки, продемонструємо це на прикладі моделі, побудованої оціночною мережею Петрі (рис. 2, а). При початковій розмітці $\mu_0 = [1, 0, 0, 0]^T$ послідовно спрацюють переходи за послідовністю $\sigma_1 = t_1, t_2, t_3, t_2, t_4$, що дозволить відновити початкову розмітку. При початковій розмітці $\mu_{01} = [5, 2, 1, 1]^T$ послідовність спрацьовування переходів:

$$\begin{aligned} \sigma_{01} &= t_0, t_2, t_2, t_3, t_2, t_1, t_0, t_2, t_2, t_3, t_0, t_0, t_2, t_0, t_1, t_0, t_0, \\ &t_1, t_3, t_0, t_1, t_2, t_2, t_2, t_1, t_2, t_1, t_1, t_2, t_1, t_3, t_2, t_1, t_2, t_1, t_2, \\ &t_0, t_3, t_2, t_0, t_2, t_1, t_1, t_1, t_2, t_2, t_2, t_2, t_2, t_2, t_2, t_2. \end{aligned}$$

приведе до розмітки $\mu_{11} = [0, 0, 0, 31]^T$, яка відповідає стану тупику, оскільки за такої розмітки жодна вершина переходу у модель не може бути активізована. Таким чином, важливо проаналізувати варіанти функціонування моделі та виявити ознаки тупикової ситуації, особливо у випадках неявного тупику (пастки). Вирішити цю задачу дозволяє розрахунок T- та P-інваріантів на основі основного рівняння мереж Петрі [22], а також аналіз рангу матриці інцидентності моделі.

Розв'язком системи лінійних однорідних діофантових рівнянь [22]: $W^T X=0$ є вектор X який називають Т-інваріантом. Він забезпечує існування послідовностей переходів $\sigma=t_1, t_2, \dots, t_n$ з розмітки μ_0 , яке приводить до цієї ж початкової розмітки. Т-інваріант, який має покриття всіх елементів ненульовими значеннями, відповідає властивостям живості (повторюваності) та досяжності. Р-інваріанти знаходять з розв'язку системи лінійних діофантових рівнянь: $WP=0$, при покритті всіх елементів Р-інваріантів ненульовими значеннями встановлюється відповідність властивостям обмеженості та збережуваності моделі.

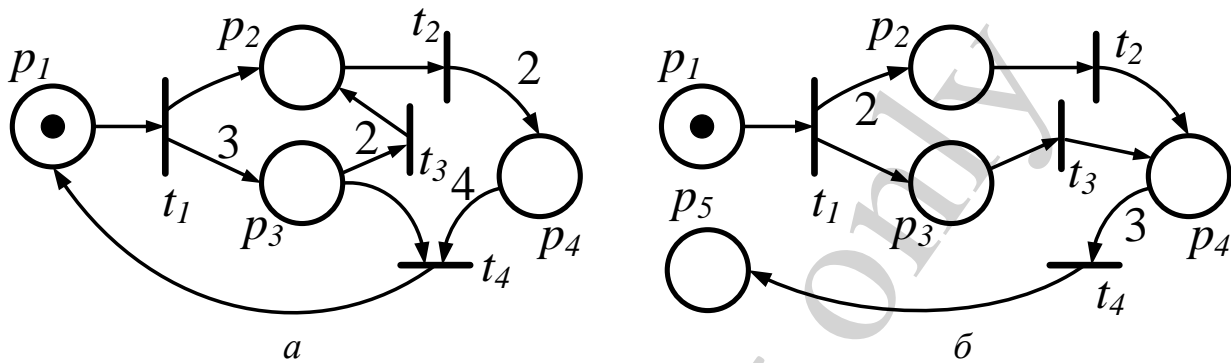


Рис. 2. Модель на основі оціночної мережі Петрі

Для опису структури мережі використовується матриця інцидентності W , що складається зі стовпців, які відповідають вершинам переходів, та рядків, які відповідають вершинам місць. При підрахунку рангу матриці W визначають повну чи неповну керованість моделі (за умови повного покриття ненульовими значеннями Т- та Р-інваріантів). Якщо ранг матриці інцидентності менше найменшої з потужностей множин вершин переходів $|T|$ чи вершин місць $|P|$ $rang(W) < \min(|T|, |P|)$, то модель не повністю керована, інакше, при рівності даних показників, модель буде повністю керованою.

Розрахунок інваріантів для моделі (рис. 2, а) мав наступні результати: $T_{2a}=[1, 2, 1, 1]^T$; $P_{2a}=[5, 2, 1, 1]^T$; ранг матриці інцидентності: $rang(W_{2a})=3 < \min(|T_{2a}|, |P_{2a}|)=4$. Таким чином, хоч Т- та Р-інваріанти покриті повністю ненульовими елементами, але ранг матриці інцидентності менший за величину $|T|$, тому можемо передбачити наявність конфліктної ситуації, яка може бути викликана конструкцією вільного вибору від вершини місця p_3 . Так, за відповідної розмітки мітки з вершини місця p_3 при вільному виборі подальшого шляху можуть активізувати вершину переходу t_4 і не запустити жодного разу вершину переходу t_3 . Для усунення непередбачуваного функціонування потрібно перебудувати модель так, щоб зберегти кількість спрацьовування вершин переходів (активні дії), тобто значення Т-інваріанту $T_{2b}=[1, 2, 1, 1]^T$. Приклад перебудованої моделі представлено на рис. 2, б, де уточнено структуру моделі для забезпечення подвійної активізації вершини переходу t_2 при одинарному спрацьовуванні вершини переходу t_3 та усунуто конструкцію вільного вибору, яка може привести до тупику. При підрахунку інваріантів та рангу матриці інцидентності отримані наступні результати:

$$P_{2b} = [3 \ 1 \ 1 \ 1 \ 3]^T;$$

$$W_{2b} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 2 & 0 & -1 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{rang}(W_{2b}) = 4.$$

Таким чином, інваріанти повністю покриті ненульовими значеннями, а також $\text{rang}(W_{2b})=4=\min(|T_{2b}|, |P_{2b}|)$, що вказує на повну керованість та дотримання властивостей живості, досяжності, обмеженості та збережуваності.

Однією з важливих характеристик обмежених мереж Петрі, що забезпечує її надійне функціонування, є збережуваність. Кожна вершина місця мереж Петрі при моделюванні прикладних систем є k -обмеженою, тобто має певне максимальне значення розмітки, яка в ній може бути досягнута, що враховує кількість міток, яка залишилась від попередніх кроків та кількість міток, яка додається на поточному кроці: $\#(p_{\eta}^0, I_R(t_h)) + \#(p'_{\eta}, I_R(t_h)) \leq \xi$.

Природнім чином абсолютна збережуваність досягається у автоматних мережах Петрі, у яких кожен перехід має одну вхідну і одну вихідну дугу [10], кількість міток у такій мережі не збільшується і не зменшується під час її функціонування. Також абсолютна збережуваність може бути досягнута за умов забезпечення однакової кількості вхідних та вихідних дуг для кожної вершини переходу мережі Петрі.

Визначення 4. Мережа Петрі PN з початковою розміткою μ_0 називається строго збережуваною, якщо для всіх її розміток μ' виконується умова:

$$\sum_{p_i \in P} \mu'(p_i) = \sum_{p_i \in P} \mu(p_i). \quad (2)$$

Дана умова накладає дуже жорсткі обмеження, але при моделюванні більшості прикладних задач кількість міток у моделі може збільшуватись та зменшуватись при її функціонуванні. Тому властивість збережуваності таких моделей часто оцінюють не за строгим поняттям збережуваності, а за зваженим вектором $\bar{\omega}$ цілих додатних чисел ($\omega_i \in Z^+$), визначеним для кожної розмітки. Елементи вектора дозволяють відобразити кількість міток, які одночасно переміщуються у мережі Петрі. Наприклад, на рис. 3, а. представлена мережа Петрі, в якій можливо збалансувати кількість міток у кожній розмітці, таку мережу Петрі можна назвати нестрого збережуваною або збережуваною по відношенню до вектора зважування $\omega = \{\omega_1, \omega_2, \dots, \omega_n\}$.

Визначення 5. Мережа Петрі PN з початковою розміткою μ_0 називається збережуваною по відношенню до вектора зважування $\omega = \{\omega_1, \omega_2, \dots, \omega_n\}$, де $n = |P|$ та $\omega_i \in \mathbb{Z}^+$, якщо для всіх розміток μ виконується умова:

$$\sum_{i=\{1,2,\dots,n\}} \omega_i \mu'(p_i) = \sum_{i=\{1,2,\dots,n\}} \omega_i \mu(p_i). \quad (3)$$

Для збережуваної по відношенню до вектора зважування $\bar{\omega}$ мережі Петрі важливий збіг початкової та кінцевої розміток [21], що наочно можливо продемонструвати при її перетворенні у замкнуту мережу Петрі.

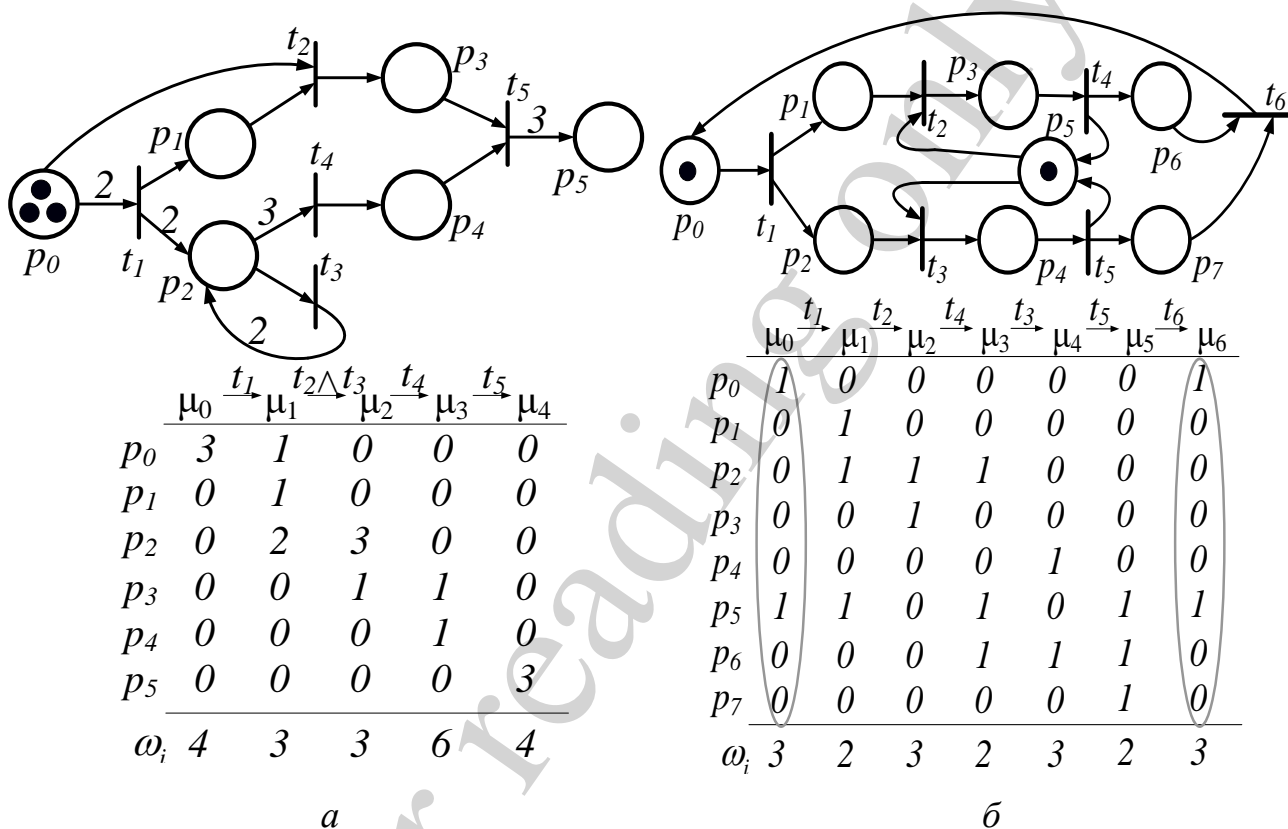


Рис. 3. Приклад нестрого збережуваної мережі Петрі

Прикладом збережуваної по відношенню до вектора зважування $\bar{\omega}$ мережі Петрі, яка здатна відновити початкову розмітку незліченну кількість разів, є замкнута мережа Петрі, приклад якої представлений на рис. 3, б. Наведена замкнута нестрого збережувана мережа Петрі, відновлює початкову розмітку через шість кроків, тобто $\mu_0 = \mu_6$.

Вектори зважування, що представлені на рис. 3, а, б для варіантів нестрого збережуваних мереж Петрі, дозволяють підтримувати постійним добуток $\omega_i \mu_i$, який визначає властивість збережуваності наведених моделей. Для автоматизованої перевірки даної властивості варіант замкнутої мережевої моделі (рис. 3, б) має переваги, оскільки реалізується простіше.

Мови мереж Петрі замкнуті по відношенню до кінцевої підстановки [13, 14], що дозволяє використовувати проаналізовані та відлагоджені локальні моделі та конструкційні примітиви для моделювання складних систем, в моделях яких певна вершина місця може бути замінена на відповідну локальну модель чи певний шаблон [14]. Заміна відбувається шляхом злиття початкової вершини місця вбудовуваної ділянки, наприклад, вершини p'_1 з вершиною місця p_3 , в яку здійснюється кінцева підстановка, та кінцевої вершини місця ділянки p'_3 з дублікатом цієї ж вершини p_{31} (рис. 4).

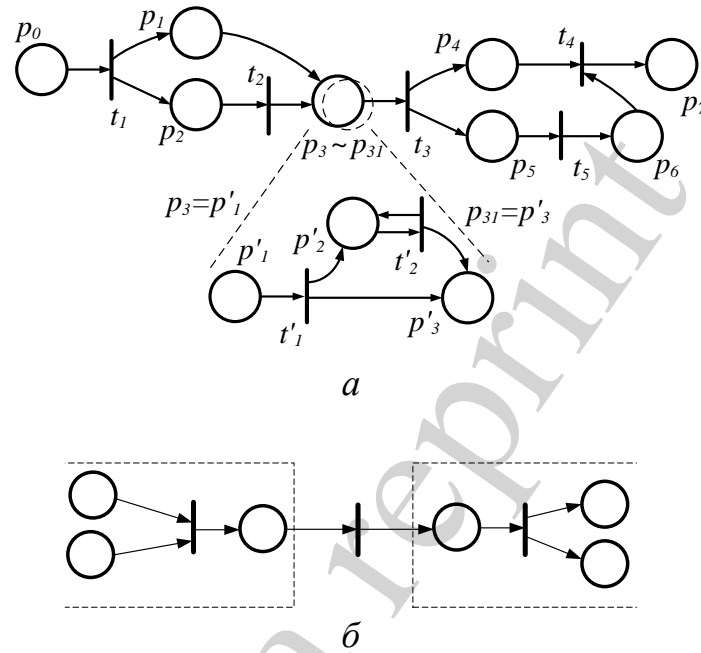


Рис. 4. Варіанти збірки локальних моделей: *a* – кінцева підстановка у вершину місця p_3 ; *б* – послідовне з'єднання

Також при збірці відлагоджених моделей може застосовуватися конструкція, представлена на рис. 4, *б*, якщо це відповідає логіці побудови моделі. Коли вбудовувана підмодель має початковою вершиною вершину місця, а кінцевою вершиною – вершину переходу (як розімкнута модель p_0-t_6 на рис. 3, *б*, тоді її вбудовують за прийнятою методикою побудови мереж Петрі на лінійній ділянці між відповідною вершиною переходу та наступною суміжною вершиною місця.

5. 3. Концептуальна модель та комбінований підхід до імітаційного моделювання систем з паралелізмом

При формуванні архітектури комбінованого підходу до імітаційного моделювання систем з паралелізмом (рис. 4) теоретичними основами виступили дві парадигми імітаційного моделювання – дискретно-подієве моделювання та моделювання динамічних систем.

Понятійний апарат підходу складають паралельні та конкуруючі процеси, формальні мови мереж Петрі L-типу та G-типу, які описують обмежені мережі Петрі. Також до понять належать PN-патерни та інструментальні за-

соби, що є комбінацією безпечних, оціночних та управляючих мереж Петрі для побудови конструктивних особливостей моделі та міжкомпонентних зв'язків. Будемо називати кожен підпорядковану модель субмоделлю, а модель, у яку поступово збираються субмоделі – частковою моделлю системи. Послідовне збирання субмоделей у часткову модель зі згортокою елементів субмоделей будемо називати редукцією.

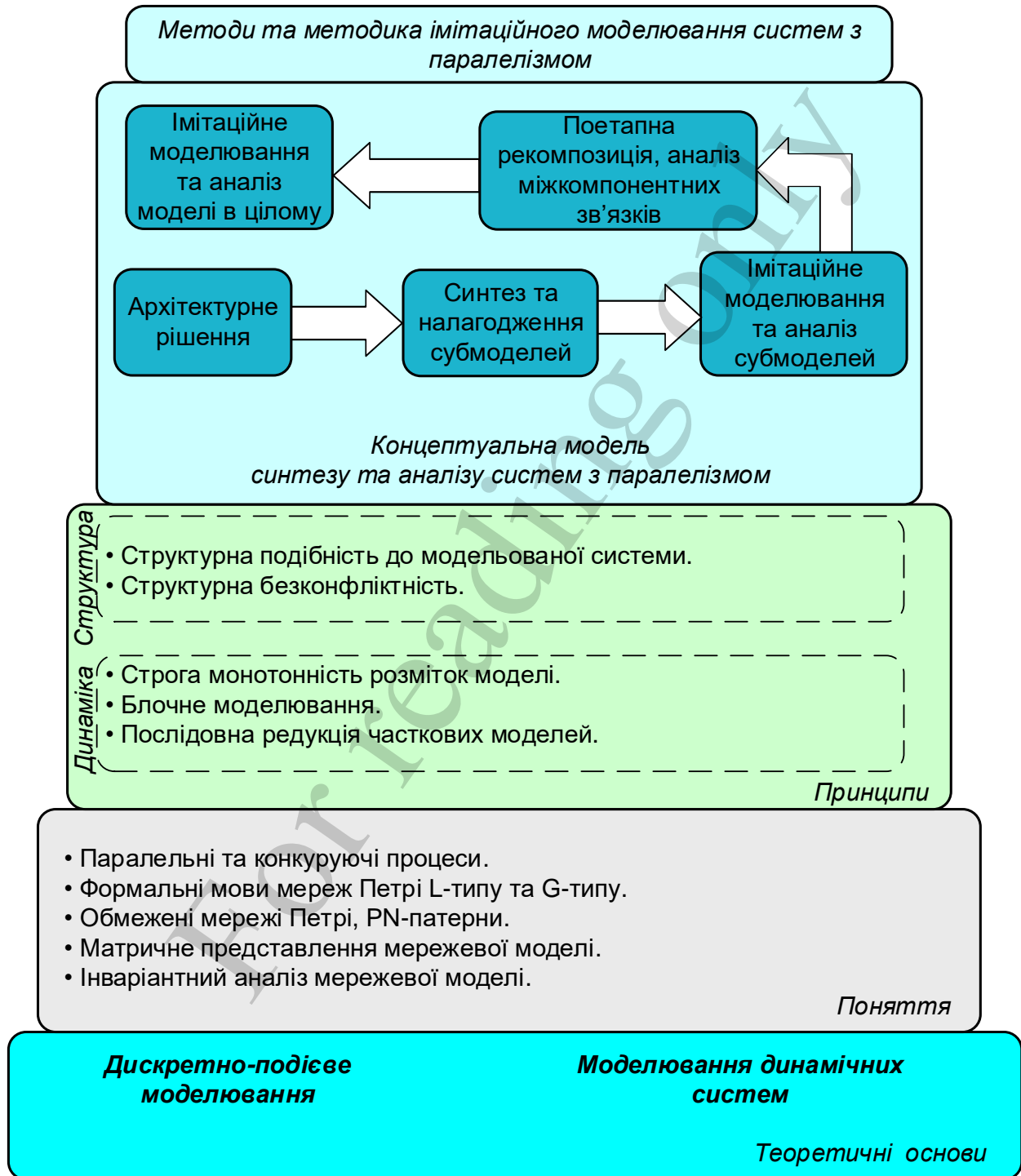


Рис. 4. Архітектура комбінованого підходу до імітаційного моделювання систем з паралелізмом

Для аналізу побудованих компонентів моделі (субмоделей) та моделі в цілому використовується імітаційне моделювання та матричне представлення їх структури [13], метод інваріантів [22]. Для визначення динамічних властивостей досліджуваних субмоделей та часткових моделей системи з паралелізмом, що формуються на кожному етапі збирання моделі, проводиться підрахунок та аналіз інваріантів і рангу матриці інцидентності.

Елементами комбінованого підходу до моделювання систем з паралелізмом виступають дві групи принципів. До першої групи належать принципи, які стосуються структурної будови моделі:

- принципи структурної подібності;
- принцип структурної безконфліктності [23].

До другої групи належать принципи, що визначають особливості моделювання та аналізу динамічних властивостей моделі:

- принцип строгої монотонності розміток моделі;
- принцип блочного моделювання;
- принцип послідовної редукції часткових моделей.

Так, принцип структурної подібності значно спрощує перевірку моделі у змішаному режимі (аналітично та візуально), принцип структурної безконфліктності дозволяє виправити помилки при побудові моделі на основі переліку правил [23]. Принцип строгої монотонності розміток дозволяє однозначно описати функціонування моделі та перевірити важливі властивості системи, такі як керованість та обмеженість. Принцип блочного моделювання, що застосовується в алгебрі процесів [24], ієрархічних мережах [13] та компонентно-орієнтованому підході до проектування програмних систем, дозволяє проводити процес побудови і дослідження імітаційної моделі системи з паралелізмом поетапно. Принцип редукції часткових моделей дозволяє зменшити розмірність досліджуваної моделі для її аналізу методом інваріантів.

У концептуальній моделі синтезу та аналізу систем з паралелізмом (рис. 4) відображено поетапний процес побудови, перевірки та відлагодження моделі з паралелізмом. На початку процесу синтезу та аналізу моделі системи з паралелізмом будується архітектурне рішення, яке відображає загальне уявлення про систему на момент постановки задачі і буде уточнюватись. Наступним етапом є побудова та налагодження субмоделей, що дозволяє отримати компоненти моделі, які складаються з відносно невеликої кількості елементів і дозволяють підготувати субмоделі до аналізу динамічних властивостей на мережевому та матричному представленні. На основі налагоджених субмоделей проводиться їх імітаційне моделювання, яке дозволяє перевірити логіку функціонування моделі, а також наявність потенційно небезпечних конструкцій, наприклад, тупиків та пасток.

Стан тупика у моделі виникає, коли не досягнута підсумкова розмітка i , разом з тим, немає передумов для спрацьовування жодного переходу. Він може виникнути, наприклад коли один процес потребує ресурсу, що в цей час використовується іншим процесом, або при недостатності ресурсів для ініціювання певного процесу. Пастка не приведе до явного припинення функціонування моделі або до стану тупика, але кінцевого результату моделювання не буде досягнуто.

Після імітаційного моделювання проводиться аналіз субмоделей на матричному представленні з використанням методу інваріантів, який дозволяє перевірити основні динамічні властивості субмоделей. Також при інваріантному аналізі можна виявити ознаки потенційно небезпечних сценаріїв функціонування моделі, наприклад, таких як прихований тупик, аналіз якого продемонстровано на моделі, що представлена на рис. 2.

На четвертому етапі для отримання моделі системи з паралелізмом проводиться поетапна рекомпозиція субмоделей у часткову модель. При цьому проводиться аналіз міжкомпонентних зв'язків, який здійснюється на матричному представленні часткової моделі з використанням методу інваріантів та рангу матриці інцидентності. Відмінністю цього етапу від другого етапу є не тільки об'єкт – часткова модель, на відміну від субмоделі, але і необхідність проводити згортку елементів часткової моделі за загортковим методом з метою зменшення розмірності моделі для інваріантного аналізу.

Останній п'ятий складовий концептуальної моделі передбачає імітаційне моделювання моделі системи з паралелізмом в цілому, при цьому теж застосовується згортка підпорядкованих моделей, що забезпечує можливість аналізу мережевої моделі та її матричного представлення за методом інваріантів. Якщо потрібно перевірити визначені елементи на тлі проведення імітаційного моделювання цілої системи, вони можуть деталізуватися.

6. Обговорення результатів дослідження засобів моделювання та аналізу систем з паралелізмом

Проаналізовані парадигми імітаційного моделювання. Запропоновано створити підхід до моделювання програмних систем на основі сполучення двох парадигм імітаційного моделювання: дискретно-подієвого моделювання та моделювання динамічних систем. Засоби дискретно-подієвого моделювання представлені обмеженими мережами Петрі, на основі яких будуються локальні моделі проектних рішень, що аналізуються при імітаційному експерименті. Моделювання динамічних систем дозволяє отримати однозначний математичний опис мережевої моделі у матричному вигляді та розрахувати значення інваріантів, аналіз яких надає можливість визначити динамічні характеристики моделі.

Описані особливості та обмеження формальних мов мереж Петрі дозволили оцінити обмеження характеристик формальних мов L-типу для опису моделей програмних систем. Це дозволило визначити необхідні для доповнення мов L-типу властивості мов G-типу для представлення проектних рішень. Зокрема, розширена мова L_G -типу (1) не має обмежень на початкові та проміжні розмітки, які належать до кінцевої множини підсумкових розміток. Таким чином, обмеження WF-мереж [11], що описуються мовами L-типу, вдалося розширити. Обрані інтерпретації обмежених мереж Петрі для побудови комбінованих інструментальних засобів моделювання систем з паралелізмом, які мають більшу потужність моделювання ніж WF-мережі. Це безпечні, оціночні та управляючі мережі Петрі, які дозволяють подолати недоліки, зазначені у роботі [19]. Також запропоновані мережі Петрі можуть мати скінченну множину контролюючих ве-

ршин, що дозволяє описувати паралельні та конкуруючі процеси, які утворюються при функціонуванні програмних додатків.

Проаналізовані властивості обмежених мереж Петрі, що описуються класом формальних мов L_G -типу, такі як живість, досяжність, обмеженість та збережуваність. Зокрема, властивість досяжності може застосовуватись для визначення тупикових розміток (рис. 2). Про повну керованість моделі свідчить покриття ненульовими значеннями всіх елементів T - та R -інваріантами разом з рівністю рангу матриці інцидентності мінімальній розмірності потужності множин вершин переходів чи вершин місць. Для забезпечення передбачуваності функціонування моделі застосовується властивість нестрогої збережуваності (3). Замкнутість обмежених мереж Петрі по відношенню до кінцевої підстановки дозволяє використовувати їх для формування локальних моделей, які у подальшому збирають (рис. 4) у цілісну модель програмної системи.

Сформовані принципи для комбінованого підходу до імітаційного моделювання систем з паралелізмом. Вони дозволяють забезпечити аналіз моделі на всіх етапах роботи з нею, а також закладають основу для концептуальної моделі синтезу та аналізу систем з паралелізмом. На відміну від частково узгоджених елементів аналізу [6, 7] та модифікацій мереж Петрі з недоведеними властивостями чи прихованими елементами [15–18], пропонується концептуальна модель дозволяє об'єднати необхідні засоби синтезу та аналізу моделі з паралелізмом. Це засоби явного графічного опису моделі, його аналітичного представлення, яке дозволяє виявити непередбачувані ситуації та визначити структури моделі, що їх викликають. Дана концептуальна модель дозволяє пропонувати та перевірити проектні рішення, які забезпечать працездатність та передбачуваність функціонування досліджуваної системи.

На основі вищезазначених складових сформована архітектура комбінованого підходу для покрокової побудови складових моделі системи з паралелізмом, та на їх основі представлене формування цілісної моделі програмної системи. У даному підході враховані обмеження розмірності локальних моделей до 40–45 елементів найбільшої множини з множин T та R , що визначають максимальну розмірність матриці інцидентності. Для формування цілісної моделі системи та її аналізу передбачена рекомпозиція часткових моделей.

До недоліків комбінованого підходу належить віднести досить праце містку побудову субмоделей та моделі програмної системи. На зменшення даного недоліку можливо сформувати шаблони конструкцій, які часто зустрічаються у моделях програмних систем.

Перспективами даного дослідження є доповнення інструментальних засобів елементами часових мереж Петрі, що теж належать до обмежених мереж Петрі, а також автоматизоване формування шаблонів для виправлення ділянок моделі з непередбачуваною поведінкою.

7. Висновки

1. Запропоновано на основі сполучення двох парадигм імітаційного моделювання: дискретно-продієвого моделювання та моделювання динамічних систем побудувати комбінований підхід та інструментальні засоби моделювання

систем з паралелізмом. Проведений аналіз формальних мов мереж Петрі дозволив визначити особливості та обмеження мови L-типу для побудови інструментальних засобів моделювання систем з паралелізмом. Обмежені мережі Петрі, які описуються формальною мовою L_G -типу, дозволяють розширити потужність моделювання у порівнянні з засобами на основі мов L-типу, що важливо при моделюванні програмних систем. Зокрема, запропоновані інструментальні засоби дозволяють починати імітаційне моделювання з будь-якої розмітки, що належить до кінцевої множини підсумкових маркувань. Також можливо утворювати скінченну кількість контролюючих вершин, розмітка яких має відновлюватись по закінченні сеансу імітації моделі.

2. Представлена методика виявлення конфліктної ситуації на прикладі прихованого тупика з використанням інваріантного методу та аналізу матриці інцидентності. Ця методика дозволяє проаналізувати причини виникнення тупика, запропонувати корекцію моделі та перевірити її динамічні властивості. Так, якщо у модель може виникнути критична ситуація, покриття складових T- та P-інваріантів ненульовими елементами не достатньо, потрібно щоб ранг матриці інцидентності дорівнював найменшій розмірності з потужності множин вершин переходів $|T|$ чи вершин місць $|P|$.

3. Сформовані принципи та концептуальна модель для розробки архітектури комбінованого підходу до імітаційного моделювання систем з паралелізмом. Концептуальна модель дозволяє проаналізувати модель програмної системи з використанням засобів імітаційного моделювання. Також вона передбачає використання аналітичного представлення, що створює підґрунтя для автоматизованої перевірки моделі при побудові й моделюванні функціонування програмних засобів в умовах, наближених до реальних. Принципи, що стосуються структури моделі, дозволяють забезпечити самоперевірку при побудові компонентів моделі та візуальний аналіз ділянок моделі при виявленні конфліктних ситуацій. Принципи, що стосуються динаміки моделі, дозволяють забезпечити аналіз динамічних властивостей як компонентів моделі, так і часткових моделей та моделі в цілому. Це важливо для забезпечення цілісного аналізу моделі програмної системи, особливо при наявності численних паралельних та конкуруючих процесів. Можливість застосування у комбінованому підході принципів побудови ієрархічних мереж Петрі, які описуються формальними мовами G-типу, дозволяє дотримуватись вимог стійкості до змін та масштабованості для програмних систем. Ці властивості є важливими при розробці сучасних програмних систем.

Література

1. Стоян, В. А. (2008). Моделювання та ідентифікація динаміки систем із розподіленими параметрами. Київ: Київський університет, 201.
2. Строгалева, В. П., Толкачева, И. О. (2008). Имитационное моделирование. Москва: Изд-во МГТУ им. Н.Э. Баумана, 280.
3. Самарский, А. А., Михайлов, А. П. (2001). Математическое моделирование. Идеи. Методы. Примеры. Москва: Физматлит, 320.
4. Супруненко, О. О. (2013). Парадигми імітаційного моделювання при дослідженні складних систем з паралелізмом. Восточно-Европейский жур-

нал передовых технологий, 5 (4 (65)), 63–67. URL: <http://journals.uran.ua/eejet/article/view/18353/16394>

5. Сергієнко, І. В. (2018). Математичне та програмне моделювання складних систем з використанням суперкомп'ютерних технологій. Вісник Національної академії наук України, 3, 39–48. doi: <https://doi.org/10.15407/visn2018.03.039>

6. Карпов, Ю. Г. (2005) Имитационное моделирование систем. Введение в моделирование с AnyLogic 5. Санкт-Петербург: БХВ-Петербург, 400.

7. Брауде, Э. (2004). Технология разработки программного обеспечения. СПб.: Питер, 655. URL: http://www.immsp.kiev.ua/postgraduate/Biblioteka_trudy/TekhnologiyaRazrabProgrBraude2004.pdf

8. Van Hee, K. (2002). Workflow management: models, methods, and systems. The MIT Press. doi: <https://doi.org/10.7551/mitpress/7301.001.0001>

9. Карпов, Ю. Г. (2010). Model Checking. Верификация параллельных и распределённых программных систем. Санкт-Петербург: БХВ-Петербург, 560.

10. Кузьмук, В. В., Супруненко, О. А. (2014). Средства описания информационных потоков в динамических моделях медицинских программно-аппаратных систем. Theoretical & Applied Science, 7 (15), 11–18. doi: <https://doi.org/10.15863/tas.2014.07.15.2>

11. Van der Aalst, W. M. P. (2013). Business Process Management: A Comprehensive Survey. ISRN Software Engineering, 2013, 1–37. doi: <https://doi.org/10.1155/2013/507984>

12. Jensen, K., Rozenberg, G. (Eds.) (1991). High-level Petri Nets: Theory and Application. Springer, 724. doi: <https://doi.org/10.1007/978-3-642-84524-6>

13. Кузьмук, В. В., Супруненко, О. О. (2010). Модифицированные сети Петри и устройства моделирования параллельных процессов. Киев: Маклаут, 252.

14. Петерсон, Дж. (1984). Теория сетей Петри и моделирование систем. Москва: Мир, 264.

15. Ломазова, И. А. (2004). Вложенные сети Петри: моделирование и анализ распределённых систем с объектной структурой. Москва: Научный мир, 208.

16. Ломазова, И. А. (2009). Адаптивное и динамическое моделирование потоков работ на основе взаимодействующих сетей Петри. Методы и средства обработки информации. Труды III Всероссийской научной конференции. Москва: Издательский отдел факультета вычислительной математики и кибернетики МГУ, 32–37.

17. Bashkin, V. A. (2012). Approximating bisimulation in one-counter nets. Automatic Control and Computer Sciences, 46, 317–323. doi: <https://doi.org/10.3103/s014641161207005x>

18. Bashkin, V. A. (2017). On the Resource Equivalences in Petri nets with Invisible Transitions. Petri Nets and Software Engineering (PNSE'17). Zaragoza, 51–68.

19. Belusso, C. L. M., Sawicki, S., Roos-Frantz, F., Frantz, R. Z. (2016). A Study of Petri Nets, Markov Chains and Queueing Theory as Mathematical Modeling Languages Aiming at the Simulation of Enterprise Application Integration Solutions: A First Step. Procedia Computer Science, 100, 229–236. doi: <https://doi.org/10.1016/j.procs.2016.09.147>

20. Кузьмин, Е. В., Соколов, В. А. (2005). Вполне структурированные системы помеченных переходов. Москва: Физматлит, 176.

21. Hack, M. (1975). Decision Problems for Petri Nets and Vector Addition Systems, Computation Structures Group Memo 95, Project MAC. Massachusetts Institute of Technology, Cambridge, Massachusetts, March 1974, pp. 79. revised as Memo 95-1, August 1974; Technical Memo 59, Project MAC, Massachusetts Institute of Technology, Cambridge, Massachusetts, March 1975, pp. 7.

22. Murata, T. (1989). Petri nets: Properties, analysis and applications. Proceedings of the IEEE, 77 (4), 541–580. doi: <https://doi.org/10.1109/5.24143>

23. Супруненко, О. О. (2019). Комбінований підхід до імітаційного моделювання динаміки програмних систем на основі інтерпретацій мереж Петрі. KPI Science News, 5-6, 43–53. doi: <https://doi.org/10.20535/kpi-sn.2019.5-6.174596>

24. Нестеренко, Б. Б., Новотарский, М. А. (2007). Алгебра процессов для моделирования сложных систем с реальной рабочей нагрузкой. Реестрация, зберігання та обробка даних, 9 (4), 49–59.

For reading Only