

10. Ecosystems and Human Wellbeing: Synthesis // Millennium Ecosystem Assessment. – Washington : Island Press, 2005. [Electronic resource]. – Mode of access <http://www.millenniumassessment.org/ru/Index.aspx>

11. Kant S. A social choice approach to sustainable forest management: an analysis of multiple forest values in Northwestern Ontario / S. Kant, S. Lee // Forest Policy and Economics. – 2004. – № 6 (3-4). – Pp. 215-227. [Electronic resource]. – Mode of access http://www.forestry.toronto.edu/people/shashi_site/Ref_05_01_files/Kant_2004_a.pdf

12. Kearney A. Stakeholder perspectives on appropriate forest management in the Pacific Northwest / A. Kearney, G. Bradley, R. Kaplan and S. Kaplan // Forest Science. – 1999. – № 45(1). – Pp. 62-73.

13. Layard R. Cost-Benefit Analysis / R. Layard, S. Glaister. – Cambridge : Cambridge University Press, 1996.

14. Lee S. Forest Values, Perceptions and Co-management in Northwestern Ontario / S. Lee, S. Kant. – Toronto : Toronto University, Sustainable Forest Management Network Research Communication, April, 2003. – 49 p.

15. Newbold P. Statistics for Business and Economics / P. Newbold // Fourth edition. New Jersey: Prentice Hall, 1995. – 876 p.

16. Rozen S. Hedonic prices and implicit markets: product differentiation in pure competition / S. Rozen // Journal of Political Economy. – 1974. – № 82. – Pp. 34-55.

Бас Т.Б. Анализ предпочтений стейкхолдеров относительно услуг лесных экосистем с использованием непараметрического статистического аппарата

Исследованы предпочтения стейкхолдеров относительно услуг лесных экосистем (УЛЭ), обработана полученная информация с помощью непараметрического статистического аппарата. Доказана статистическая значимость различий в предпочтениях разных социальных групп (стейкхолдеров) относительно УЛЭ с использованием тестовой статистики Фридмана.

Ключевые слова: услуги лесных экосистем, непараметрическая статистика, тест Фридмана.

Bas T.B. Analysis of stakeholder's preferences regarding forest ecosystem services using non-parametric statistical methods.

Stakeholder's preferences regarding forest ecosystems services (FES) have been investigated. The information received by non-parametric statistical methods was processed. Statistical significance of differences in preferences of different social groups (stakeholders) regarding FES was proved using the Friedman test.

Keywords: forest ecosystem services, non-parametric statistics, Friedman test.

УДК 004.942

*Доц. П.Ю. Денисюк, канд. техн. наук;
проф. В.М. Теслюк, д-р техн. наук; здобувач М.Є. Перейма;
студ. Т.В. Теслюк – НУ "Львівська політехніка"*

ВИКОРИСТАННЯ ІЄРАРХІЧНИХ МЕРЕЖ ПЕТРІ ДЛЯ МОДЕЛЮВАННЯ МЕМС¹

Розроблено модель для аналізу МЕМС з використанням ієрархічних мереж Петрі та реалізовано програмні засоби, як дають змогу автоматизувати побудову моделі на основі МП і аналіз динаміки.

Ключові слова: мережа Петрі, мікроелектромеханічні системи (МЕМС).

¹ Матеріали підготовлені за підтримки гранту Президента України докторам наук для здійснення наукових досліджень на 2011 рік. Тема: "Розроблення інформаційних технологій автоматизації структурного синтезу та аналізу мікроелектромеханічних систем" (Ф35/541-2011, №0111U009116)

Вступ. Мережі Петрі є популярним сучасним формалізмом для моделювання й аналізу розподілених, паралельних та складних об'єктів і систем [1-3]. Компоненти довільної системи та їхні дії зручно представляти абстрактними подіями. Подія може відбутися один раз, повторитися багаторазово, породжуючи конкретні інші дії, чи не відбутися жодного разу. Сукупність дій, що виникають як реалізації подій, утворює процес. У загальному випадку, одна і та сама система може функціонувати в одних і тих самих умовах по-різному, породжуючи деяку множину процесів. Складні системи, в яких однаковий набір вхідних параметрів може викликати деяку множину вихідних параметрів, зручно описувати термінами мереж Петрі.

Ієрархічні мережі Петрі (МП) – це розширення стандартного формалізму мереж Петрі [1], за якого маркери, що визначають розмітку мережі, самі є маркованими мережами. Назва "ієрархічні" тут вказує на те, що елементи мереж самі є мережами, подібно до того, як у системі ієрархічних множин елементами деякої множини можуть бути множини.

Під час моделювання систем оброблення та передачі інформації на системному рівні проектування мікроелектромеханічних систем (МЕМС) [1] використання ієрархічного підходу дає змогу значно спростити структурну схему системи. Крім того, це дасть змогу розробнику складних мікроелектромеханічних систем використовувати бібліотеку вже розроблених проектних рішень, що зменшить загальний час та ефективність автоматизованого проектування на системному рівні.

1. Побудова моделі на основі ієрархічної мережі Петрі. На системному рівні проектування МЕМС для аналізу структурних рішень використовуються моделі на основі мереж Петрі [3-8]. Розширити можливості аналізу структур МЕМС можна за допомогою ієрархічних мереж Петрі, які дають змогу використовувати як елементи – мережі Петрі. Математична модель МЕМС на основі ієрархічних мереж Петрі має такий вигляд:

$$N_h = \{S, T, F, M_0, A, A_{in}, A_{out}\},$$

де: $S = \{S_1, S_2, \dots, S_g\}$ – множина позицій (стани); $T = \{t_1, t_2, \dots, t_v\}$ – множина переходів; F – множина дуг, яка включає дві підмножини вхідних та вихідних дуг по відношенню до переходу; M_0 – множина, яка задає початкове маркування мережі Петрі; g, v – кількість позицій та переходів, відповідно; A – множина вкладених мереж Петрі в S_i позицій; A_{in} – множина, яка відображає доступну множину вхідних позицій вкладених мереж Петрі; A_{out} – множина маркерів вихідних позицій вкладених мереж Петрі.

Розроблена модель системного рівня проектування дає змогу провести аналіз динаміки МЕМС.

2. Програмне забезпечення системи для моделювання МЕМС. Мова програмування Java обрана як програмний засіб реалізації підсистеми. Це дало можливість створити незалежну від архітектурної платформи комп'ютера підсистему аналізу моделей на основі ієрархічних мереж Петрі. Окрім того, використання об'єктно-орієнтованого підходу під час програмної реалізації, дало змогу користувачу в подальшій роботі підключати зовнішні програмні модулі та системи. Головною перевагою підсистеми аналізу ієрархіч-

них мереж Петрі є можливість отримати протокол розв'язання задачі, та відслідковувати покрокове виконання вкладених ієрархічних мереж Петрі, що спрощує розуміння та аналіз отриманих даних.

Розроблене програмне забезпечення підсистеми можна поділити на основні чотири групи класів: класи, які дають змогу проводити аналіз мереж Петрі (controller); моделі ієрархічних мереж Петрі (model); класи роботи з файлами (util); класи, призначені для організації візуалізації роботи підсистеми (ui).

Основною перевагою цього програмного продукту є можливість використовувати вже розроблені структурні схеми мереж Петрі для моделювання складних мікросистем. За допомогою розробленого інформаційного забезпечення користувач має змогу зберігати розроблені моделі MEMC у бібліотеку моделей, або редагувати існуючі. Крім того, система веде протокол роботи процесу моделювання, що дає змогу краще проаналізувати та зрозуміти отримані результати.

Результати. Отже, MEMC є складною системою, яка містить підсистеми, які, своєю чергою, можна розглядати як окремі зв'язані між собою системи. У такому випадку зручно для системного рівня проектування MEMC використовувати ієрархічні МП. Як приклад, розглянемо просту модель структури MEMC (рис. 1) з використанням ієрархічних мереж Петрі, що зображено на рис. 2.

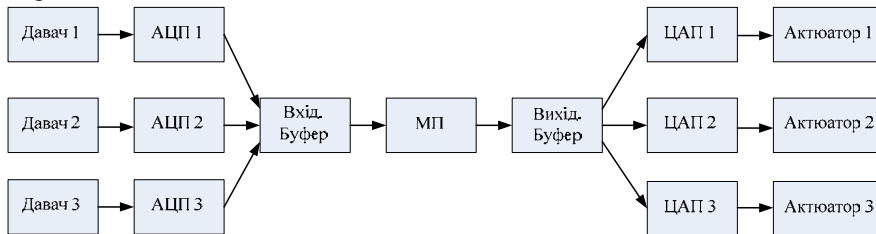


Рис. 1. Структура мікроелектромеханічної системи

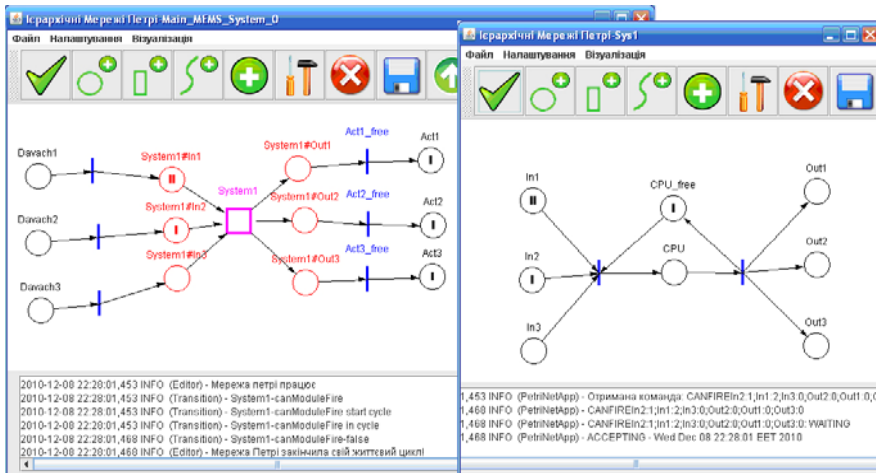


Рис. 2. Результати моделювання MEMC

У разі необхідності врахування пріоритетів до мережі на рис. 2, потрібно додати відповідну інформацію, а за потреби врахування часових параметрів, необхідно в кольоровій мережі врахувати часовий механізм.

Висновок. Розроблено та формалізовано модель для аналізу структур MEMC за допомогою ієрархічних мереж Петрі. Отримані результати аналізу схемних моделей дають змогу стверджувати про зменшення та спрощення процесу моделювання складних мікроелектромеханічних пристроїв.

Література

1. Теслюк В.М. Моделі та інформаційні технології синтезу мікроелектромеханічних систем : монографія / В.М. Теслюк. – Львів : Вид-во ПП "Вежа і Ко", 2008-192 с.
2. Ломазова И.А. Моделирование мультиагентных динамических систем вложенными сетями Петри / И.А. Ломазова // Программные системы: Теоретические основы и приложения : Изд-во "Наука". Физ-матлит, 1999. – С. 143-156.
3. Теслюк В.М. Застосування мереж Петрі при проектуванні MEMC на системному рівні / В.М. Теслюк // Вісник Національного університету "Львівська політехніка". – Сер.: Комп'ютерні системи проектування. Теорія і практика. – Львів : Вид-во НУ "Львівська політехніка", 2006. – № 564. – С. 45-53.
4. Теслюк В.М. Методологія проектування MEMC на системному рівні проектування // Збірник наукових праць Української академії друкарства. – Сер.: Комп'ютерні технології друкарства. – Львів : Вид-во УАД, 2007. – № 18. – С. 56-63.
5. Teslyuk Vasyi. Computer-Aided Design of MEMS at System Level / Teslyuk Vasyi, Tarik Al Omari, Hamza Alshavabkeh, Pavlo Denysyuk, Mykhaylo Melnyk // Journal Machine Dynamics Problems. – Poland, Warsaw University of Technology. – 2007. – Vol. 31, No. 3. – Pp. 92-104.
6. Теслюк В.М. Розроблення математичних моделей MEMC на основі мереж Петрі для системного рівня автоматизованого проектування / В.М. Теслюк, П.Ю. Денисюк, Хамза Алі Юсеф Аль Шавабкех, Тарік (Мох'д Тайсір) Алі Аль Омари // Моделювання та інформаційні технології : зб. наук. праць ін-ту проблем моделювання в енергетиці ім. Г.С. Пухова НАН України. – К., 2008. – Вип. 46. – С. 120-126.
7. Hamza Ali Yousef Alshawabkeh. Information models development for mems analysis subsystem based on stochastic Petri nets. Proc. of the Xth Inter. Conf. on Modern Problems of Radio Engineering / Hamza Ali Yousef Alshawabkeh, Vasyi Teslyuk, Andriy Kernysky, Pavlo Denysyuk // Telecommunications and Computer Science, (TCSET'2010). – Lviv, 2010. – Pp. 46.
8. Vasyi Teslyuk. Models and means of mems analysis based on stochastic Petri nets, Proc. of the Xth Inter. Conf. on Modern Problems of Radio Engineering / Vasyi Teslyuk, Andriy Kernysky, Pavlo Denysyuk, Hamza Ali Yousef Alshawabkeh // Telecommunications and Computer Science, (TCSET'2010). – Lviv-Slavsko, Ukraine, 2010. – Pp. 45-48.

Денисюк П.Ю., Теслюк В.М., Перейма М.Е., Теслюк Т.В. Использование иерархических сетей Петри для моделирования МЭМС

Разработана модель для анализа МЭМС с использованием иерархических сетей Петри и реализованы программные средства, которые дают возможность автоматизировать построение модели на основе МП и анализ динамики.

Ключевые слова: сеть Петри, микроэлектромеханические системы (МЭМС).

Denysyuk P. Yu., Teslyuk V.M., Pereyma M. Ye., Teslyuk T.V. Use hierarchical Petri nets for modeling MEMS

In the paper a model of MEMS analysis by using the hierarchical Petri nets is developed and software tools are realized that enable to automate the model design on the basis of Petri nets and dynamics analysis.

Keywords: Petri nets, microelectromechanical systems (MEMS).

ПРОГРАМНА МОДЕЛЬ РОЗВ'ЯЗАННЯ ЗАДАЧІ ЗАДОВОЛЕННЯ ОБМЕЖЕНЬ У ПРОЦЕСІ СТРУКТУРНОГО СИНТЕЗУ МІКРОЕЛЕКТРОМЕХАНІЧНИХ СИСТЕМ¹

Розроблено програмну модель розв'язання задачі задоволення обмежень (ЗЗО) у процесі структурного синтезу мікроелектромеханічних систем (МЕМС).

Ключові слова: МЕМС, моделі збереження даних, ЗЗО, синтез, елементи МЕМС, обмеження задачі синтезу МЕМС.

Вступ. У процесі проектування сучасних мікроелектромеханічних систем [1] використовують дані, які стосуються середовища та характеру їх функціонування [2]. Ця інформація є визначальною при виборі технології та матеріалів виготовлення МЕМС, структури підсистем та компонентів окремих елементів МЕМС тощо. Типовою задачею, яку необхідно розв'язати в процесі структурного синтезу МЕМС, є задача підбору елементів МЕМС, які б задовольняли граничні умови, що накладаються характером та середовищем використання МЕМС та його призначенням [3]. Іншими словами, в процесі синтезу структурної схеми МЕМС проектувальнику необхідно розв'язати задачу задоволення обмежень [4], і кількість таких обмежень буде збільшуватись зі зростанням складності МЕМС, а отже буде зростати складність ЗЗО. Здійснивши літературний огляд існуючих систем штучного інтелекту та розглянувши алгоритми [5], на основі яких такі системи функціонують, ми прийшли до висновку, що певні алгоритми штучного інтелекту (ШІ) можуть бути використані в процесі автоматизованого проектування МЕМС для розв'язання задач задоволення обмежень.

Будь-яка задача задоволення обмежень визначається множиною змінних, $x_1, x_2, \dots, x_N$, і множиною обмежень $C_1, C_2, \dots, C_M$. Кожна змінна x_i має непусту область визначення D_i можливих значень. Кожне обмеження C_i містить деяку підмножину змінних і задає допустимі комбінації значень для цієї підмножини. Стан задачі визначається шляхом присвоєння значень деяким або всім змінним, $\{x_i = v_i, x_j = v_j, \dots\}$. Присвоєння, яке задовольняє усі обмеження, називають *сумісним або допустимим присвоєнням*. Повним *присвоєнням* називають таке присвоєння, в якому бере участь кожна змінна, а *розв'язанням задачі* називають повне присвоєння, яке задовольняє всі умови. Будь-якій задачі задоволення обмежень (ЗЗО) можна дати інкрементне формулювання:

- *початковий стан*, в якому жодній змінній не присвоєно значення.
- *функція визначення приймача*. Значення може бути присвоєно будь-якій змінній з неprisвоєним значенням, при умові, що змінна не буде конфліктувати з іншими змінними, значення яких були присвоєні раніше.
- *Перевірка мети*. Поточне присвоєння є повним?
- *Вартість шляху*.

¹ Матеріали статті підготовлені за підтримки Гранту Президента України для докторів наук до 45 років. Тема роботи "Розроблення інформаційних технологій автоматизації структурного синтезу та аналізу мікроелектромеханічних систем", 2011 р. (Ф35/541-2011, №0111U009116).

Під час вибору і присвоєння значення змінній в наступному вузлі пошуку доцільно вибирати змінну з найменшою кількістю допустимих значень, цей процес називають *евристикою з мінімальною кількістю допустимих значень*, її також називають евристикою зі "змінною, на яку поширюється найбільша кількість обмежень", або евристикою "до першого невдалого завершення". Продуктивність пошуку таким методом підвищується від 3 до 3000 разів порівняно зі звичайним пошуком. Перегляд обмежень на різних етапах пошуку або навіть перед початком пошуку може різко зменшити простір пошуку.

Попередня перевірка. Під час кожного присвоєння значення змінній X в процесі попередньої перевірки переглядають кожну змінну Y з неprisвоєним значенням, яка з'єднана зі змінною X певним обмеженням; з області визначення змінної Y видаляють будь-яке значення, несумісне зі значенням, вибраним для X .

Розроблення програмної моделі. Розглянувши кілька методів і алгоритмів розв'язання ЗЗО [5], можна виокремити три загальні функції, які є у кожному з них, а саме:

- функція вибору приймача;
- функція перевірки цілі (функція досягнення мети);
- функція задання початкового стану;
- функція присвоєння значення приймачу;
- функція повернення до попереднього вузла.

Отже, універсальний інтерфейс, що зможе об'єднати декілька методів розв'язку ЗЗО, повинен однозначно містити сигнатури цих функцій. Спробуємо розробити клас, сумісний з таким інтерфейсом, який буде розв'язувати ЗЗО методом локального пошуку.

Функція задання початкового стану. Існує два методи визначення початкового стану під час розв'язання ЗЗО:

- не використовуючи початкове присвоєння, по чергово виконати функцію вибору приймача і функцію вибору значення для змінної. Під час ініціалізації кожної наступної змінної необхідно враховувати результати попередніх присвоєнень;
- випадковим чином вибрати значення і визначити всі змінні.

Хоча 1-й метод і потребує більших затрат часу, його використання надалі може скоротити час пошуку розв'язку, оскільки початкову підбірку вже буде здійснено. Сигнатура методу:

```
public void InitVariables (ArrayList<Variable> varList) { }
```

Функція вибору приймача. Метод локального пошуку передбачає зміну значення змінної, яка на цей момент має найбільше конфліктів з існуючими обмеженнями, отже функція вибору приймача, себто функція, що визначає змінну, якій треба присвоїти нове значення, повинна реалізувати механізм визначення змінної, яка має на цей момент найвищий ступінь конфліктності. Для цього у класі Variable розроблено властивість [6]:

```
public int ConflictsPower  
{get {...визначає поточну кількість конфліктів... };
```