

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 неприсвоєним 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 неприсвоєним 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 чергову 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... };
```

Як бачимо з сигнатури властивості, користувач не може змінити її значення, його можливо отримати, тільки проаналізувавши поточний стан усіх змінних та їх значень. Отже, функція визначення приймача матиме такий вигляд:

```
public Variable GetNextVariable ()
{return new Variable (varList.AsQueryable ().Max (w=>w. ConflictsPower));}
```

Функція присвоєння значення приймачу. Дії цієї функції мають бути частково протилежними до дій, які виконувала властивість ConflictsPower, тобто вона повинна з множини значень, які може набувати змінна-приймач, визначити значення, яке буде мати найменший ступінь конфліктності з обмеженнями. Сигнатура цього методу має такий вигляд:

```
public bool SetValue (Variable v, ArrayList<Values> avVals)
{у випадку, якщо вдалось знайти значення, яке має менший ступінь
конфліктності, функція повертає значення true, в іншому разі – false.}
```

Після кожного присвоєння потрібно виконати функцію перевірки цілі для того, щоб визначити, отримали ми розв'язок задачі чи ні. Але ця функція потребує певних затрат часу, бо така перевірка тягне за собою перевірку ступеня конфліктності кожної змінної зокрема, і у разі негативного результату, у найгіршому випадку, ми знайдемо несумісність під час перевірки останньої змінної. Тому потрібно максимально мінімізувати використання цієї функції,



Рис. 1. Програмна логіка роботи модуля розв'язання 330.

насамперед потрібно визначити чіткі випадки, коли її використання є доцільним. Наприклад, якщо функція призначення нового значення приймачу не змогла знайти таке значення, за якого кількість конфліктів з обмеженнями

дорівнювала б нулю, то запуск функції перевірки рішення є однозначно недоцільним. Отже, виклик функції перевірки результату потрібно здійснювати тільки у випадках, коли поточному приймачу було присвоєне значення зі ступенем конфліктності 0. Програмну логіку роботи модуля розв'язання 330 зображено на рис. 1.

Оскільки описана схема розв'язання 330 припиняє спроби розв'язання поставленої задачі у разі знайдення першого ж рішення, вона є недосконалою і в такому вигляді її не може бути використано в процесі структурного синтезу МЕМС. За такого підходу існує ризик, що перший знайдений розв'язок буде не оптимальним, тому потрібно вдосконалити існуючу структуру і забезпечити можливість генерації множини альтернативних рішень, які згодом буде використано в подальших етапах процесу синтезу структурної схеми МЕМС. Для забезпечення генерації множини альтернативних рішень, ми розробили програмну модель, яка рекурсивно звертається до алгоритму розв'язання 330 та зберігає результати його роботи у двоспрямованому списку.

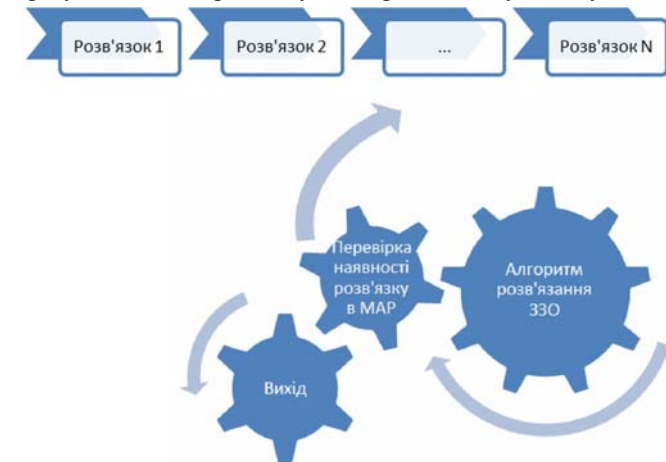


Рис. 2. Програмна модель роботи підсистеми розв'язання 330

Для забезпечення унікальності рішень, згенерованих підсистемою, алгоритм розв'язання 330 був модифікований: під час перевірки досягнення результату модифікований алгоритм перевіряє наявність такого результату у двоспрямованому списку розв'язків і якщо такий результат вже є – продовжує пошук. Задачу вважають розв'язаною у випадку, коли алгоритм вирішення 330 не спроможний знайти нових рішень, окрім тих, які вже зберігаються у програмних структурах, що представляють множини альтернативних рішень. Програмну модель роботи вдосконаленого алгоритму зображено на рис. 2.

Висновки:

1. Провівши аналіз процесу структурного синтезу МЕМС, доведено, що частину цього процесу, а саме, підбір структурних елементів можна звести до вигляду 330 та автоматизувати її розв'язання.

2. Розроблено модель збереження даних про ЗЗО, властивості та обмеження, що накладаються на MEMC, яка має вирішувати цю задачу.
3. Розроблена об'єктно-орієнтована модель, що дає змогу працювати зі сутностями, як з програмними об'єктами.
4. Вдосконалено існуючі алгоритми розв'язання ЗЗО, забезпечивши можливість генерації множини альтернативних рішень ЗЗО, яку згодом буде використано для вибору оптимального розв'язку поставленої задачі.
5. Розроблено програмні структури для зберігання інформації про рішення ЗЗО та множини альтернативних рішень.

Література

1. Теслюк В.М. Моделі та інформаційні технології синтезу мікроелектромеханічних систем : монографія / В.М. Теслюк. – Львів : Вид-во ПП "Вежа і Ко", 2008. – 262 с.
2. Лысенко И.Е. Проектирование сенсорных и актюаторных элементов микросистемной техники / И.Е. Лысенко. – Таганрог : Изд-во ТРТУ, 2005. – 235 с.
3. Лучинин В.В. Микросистемная техника. Направления и тенденции развития / И.Е. Лысенко // Научное приборостроение, 1999. – 432 с.
4. Russel S.J. "Artificial Intelligence A Modern Approach" Second Edition / S.J. Russel, P. Norvig // Prentice Hall, 2010. – 242 p.
5. Джордж Ф. Люгер Искусственный интеллект: стратегии и методы решения сложных проблем / Ф. Джордж. – Изд. 4-ое, [перераб. и доп.], 2008. – 232 с.
6. Троелсен, Эндрю. Язык программирования C# и платформа .NET 2.0 : пер. с англ. / Эндрю Троелсен. – Изд. 3-е, [перераб. и доп.]. – М. : ООО "И.Д. Вильямс", 2007-1168 с.

Зелинский А.Я., Теслюк В.М. Программная модель решения задачи удовлетворения ограничений в процессе структурного синтеза микроэлектромеханических систем

Разработана программная модель решения задачи удовлетворения ограничений (ЗЗО) в процессе структурного синтеза микроэлектромеханических систем (MEMC).

Ключевые слова: MEMC, модели сохранения данных, ЗЗО, синтез, элементы MEMC, ограничения задачи синтеза MEMC.

Zelinsky A.Ya., Teslyuk V.M. A programmatic model of decision of task of zadovolnya limitations is in the process of structural synthesis of the microelectromechanic systems

This article is devoted to program model development for solving constraints satisfaction problems (CSP) that are part of MEMS structural design process.

Keywords: MEMS, data storing models, CSP, synthesis, MEMS elements, MEMS synthesis constraints.

УДК 336.14 **Здобувач В.В. Кулешир¹ – Львівський НУ ім. Івана Франка**

АНАЛІЗ ЕФЕКТИВНОСТІ ВИКОРИСТАННЯ КОШТІВ БЮДЖЕТІВ РОЗВИТКУ В УКРАЇНІ

Проаналізовано стан і сучасний розвиток інвестиційної активності регіонів України, досліджено динаміку витраток місцевих бюджетів розвитку та їх структуру. Наведено основні причини необхідності вдосконалення механізму управління витратками місцевих бюджетів розвитку. Розглянуто пріоритетні напрями використання коштів місцевих бюджетів розвитку та розроблено методичні прийоми ефективного управління ними.

¹ Наук. керівник: проф. О.М. Ковалюк, д-р екон. наук

Ключові слова: бюджет розвитку, інвестиційна програма, капітальні видатки, метод проектів, методи управління, організаційне забезпечення, регіональні інвестиції, розширене відтворення, спеціальний фонд, структура витратків.

Постановка проблеми та її зв'язки з науковими чи практичними завданнями. В Україні існує проблема інвестиційної активності, що зумовлено несприятливими умовами для інвестування, зокрема: недосконалим законодавством з питань захисту права власності, адміністрування податків, отримання у власність земельних ділянок, недостатнім рівнем державної підтримки інвестиційної діяльності, неналежним чином розвиненою системою підготовки інвестиційних програм і відсутністю відповідних проектів для покращення використання інвестування як важливого фінансового методу.

Активізація інвестиційної діяльності неможлива без реформ у цій галузі на державному рівні. На низьку інвестиційну активність у 2008-2010 рр. вплинула світова фінансово-економічна криза, високий інфляційний тиск та низька конкурентоспроможність економіки, які призвели до зниження інвестиційної привабливості країни. Але поряд із проблемою недостатності коштів для фінансування державних інвестиційних проблем постає проблема ефективності використання цих коштів. Саме від ефективного інвестиційного менеджменту залежить розвиток регіональних інноваційно-інвестиційних програм.

Аналіз останніх досліджень. Проблеми інвестиційного розвитку регіонів загалом з методами ефективного управління використанням коштів місцевих бюджетів приділяли увагу такі дослідники, як: Н. Винниченко [4], Н. Гринчук [5], І. Запатріна [6], В. Зубенко [2; 7], В. Кудряшов [8], О. Панченко [7], І. Щербина [2, 7] та ін.

Так, Н. Винниченко [4, с. 7] вважає, що причиною низького рівня інвестиційного забезпечення соціальної сфери є неефективна побудова організаційного механізму управління витратками місцевих бюджетів розвитку. На думку А. Вінграновського, ефективність впливу бюджету розвитку місцевого бюджету на соціально-економічний розвиток регіонів залежить не тільки від обсягу надходжень, але й від напрямів використання коштів бюджету розвитку [3, с. 72]. Як один з основних напрямів зростання ефективності використання витраток місцевих бюджетів учений пропонує запровадити їх програмну класифікацію. О. Панченко та І. Щербина аналізували питання бюджетного моніторингу виконання бюджетів розвитку [2, 7].

Виділення невирішених частин загальної проблеми. Дослідження науковців, які присвячені проблемі інвестиційного розвитку регіонів, зосереджені переважно на пошуку шляхів наповнення місцевих бюджетів розвитку, тоді питанням удосконалення механізмів управління витратками даних бюджетів у науковій літературі приділено недостатню увагу.

Виклад основного матеріалу. Досвід європейських країн засвідчив, що основним трендом світової економіки є децентралізація державних фінансових потоків, децентралізація бюджетів. Практика надання більшої автономії місцевим бюджетам довела свою ефективність у багатьох країнах світу.