

використовуються при символному топологічному аналізі лінійних кіл з постійними параметрами.

4. Використання запропонованої частотної моделі елемента не приводить до появи у колі додаткових джерел сигналів.

5. Використання запропонованої частотної моделі дозволяє проводити частотний символний аналіз, статистичні дослідження та оптимізацію параметричних кіл виключно у частотній області, не звертаючись до часової.

1. Шаповалов Ю.І., Шувар Б.А. Підвищення ефективності частотних методів аналізу параметричних кіл // Вісн. ДУ «Львівська політехніка», №302, 1996, с.71
2. Ю. Шаповалов. Моделювання лінійних параметричних кіл частотним символним методом // Вісн. ДУ „Львівська політехніка”. - №343. - 1998. - С126-132.
3. Шаповалов Ю., Мандзій Б. Символьний аналіз лінійних параметричних кіл: стан питань, зміст і напрямки застосування // Теоретична електротехніка. 2007. Вип. 59, с.3-9.
4. Солодов А.В., Петров Ф.С. Линейные автоматические системы с переменными параметрами. – М.: Наука. Гл. ред. Физ.-мат.лит., 1971. – 620 с.
5. Zadeh L. A. Frequency Analysis of Variable Networks // Proc. of the IRE, vol.39, 1950.
6. Гоноровский И.С. Радиотехнические цепи и сигналы // М.: Сов. радио. 1977.
7. Тафт В.А. Электрические цепи с переменными параметрами. - М.: Энергия, 1968. – 328 с.
8. Ю. Шаповалов. Формування рівнянь лінійних параметричних кіл топологічним методом // Вісн. «Радіоелектроніка та телекомунікації» НУ Львівська політехніка. -№ 595. - 2007. с. 3-6.
9. Сигорский В.П., Петренко А.И. Основы анализа электронных схем. - К.: Вища школа, 1971. - 568 с.

Поступила 19.01.2009р.

УДК 004.942

В.М.Теслюк, к.т.н., доцент кафедри САП, НУ «Львівська політехніка»,
А.Я.Зелінський, аспірант НУ «Львівська політехніка»,
Хамза Алі Юсеф Аль Шавабкех, аспірант НУ «Львівська політехніка».

ЗАСТОСУВАННЯ МЕТОДУ ГІЛОК ТА ГРАНИЦЬ ДЛЯ ЗМЕНШЕННЯ ПОТУЖНОСТІ МНОЖИНИ АЛЬТЕРНАТИВНИХ РІШЕНЬ ПРИ РОЗВ'ЯЗАННІ ЗАДАЧ СИНТЕЗУ МЕМС

В роботі застосовано метод гілок та границь для зменшення потужності множини альтернативних структур МЕМС. На основі цього алгоритму розроблено підсистему, яка дає змогу автоматизувати процес зменшення потужності множини альтернативних рішень та наведено результати чисельних експериментів.

In this article usage of branch and bounds method for decreasing multitude power of alternative MEMS structures is described. On the basis of this method a subsystem has been developed, which allows to automatize the process of decreasing alternative solutions multitude power.

Вступ

На сьогодні надзвичайно активно розвиваються міждисциплінарні наукові області. Однією з таких галузей є мікроелектромеханічні системи [1-3]. Особливістю пристроїв цього типу є їх мікронні розміри, що значно ускладнює процеси операцій тестування, постановки експериментів, виготовлення тощо. Тому особливого значення набувають програмні системи, технології, методи та моделі, які дають змогу отримати результати моделювання та проектування з високою точністю.

Як правило, в процесі розроблення МЕМС використовується багаторівневий підхід [4], який включає системний, схемотехнічний та компонентний рівні. Множина альтернативних рішень (МАР), яку генерують в процесі розв'язання задач структурного синтезу потребує зменшення її потужності шляхом застосування адаптивних методів та засобів. Для розв'язання цієї задачі застосовано метод гілок та границь, що визначає актуальність роботи.

1. Розроблення алгоритму застосування методу гілок та границь

Метод гілок і границь (МГГ) є одним із найбільш популярних і ефективних методів неперервної та дискретної глобальної оптимізації. В основу МГГ покладено той факт, що для отримання розв'язання задачі оптимізації на деякій допустимій множині рішень можна розбити цю множину на підмножини, знайти розв'язок задачі на кожній з підмножин і у якості відповіді взяти найкращий з отриманих розв'язків, у відповідності з критерієм якості задачі (найбільшим чи найменшим значенням цільової функції).

Розвиваючи описану ідею можна прийти до висновку, що замість знаходження точного розв'язання задачі на підмножині (що може бути надзвичайно складно), можна знаходити лише його оцінку, а потім уточнювати її шляхом подрібнення складових підмножин. Іншою особливістю МГГ є можливість відкидання так званих «безперспективних» підмножин, тобто підмножин, на яких гарантовано не може бути розв'язання задачі. Алгоритм гілок та границь працює за таким алгоритмом:

Крок 0. Початкове розбиття допустимої множини на декілька підмножин і визначення для кожної з них оцінок зверху і знизу (мінімального значення цільової функції).

Крок 1. Визначення поточної підмножини шляхом відшукування найменшої серед оцінок знизу на підмножинах.

Крок 2. Перевірка умови зупинки алгоритму шляхом порівняння різниці між оцінками зверху і знизу на черговій поточній підмножині з заданою

точністю. Якщо точність досягнута, то перехід до кроку 6.

Крок 3. Розбиття поточної підмножини на нові підмножини і обчислення оцінок мінімального значення цільової функції на кожній з отриманих підмножин.

Крок 4. Визначення найменшої оцінки зверху на підмножинах і відкидання тих підмножин, у яких оцінки знизу будуть більшими за отриману найменшу оцінку зверху.

Крок 5. Перехід до кроку 1.

Крок 6. В якості відповіді можна взяти центр ваги поточної рекордної підмножини.

На сучасному рівні розвитку обчислювальної техніки, зокрема персональних комп'ютерів, основним шляхом збільшення її продуктивності є перехід до багатоядерної архітектури мікропроцесорів. Вже зараз будь-який пересічний користувач персонального комп'ютера може дозволити собі працювати з багатоядерною обчислювальною системою. Проте необхідною умовою для зростання продуктивності, в цьому випадку, є здатність програмного забезпечення працювати у багатопроцесорному (ядерному) режимі, тобто бути реалізованим за допомогою технологій і засобів паралельного програмування.

Ще однією особливістю методу гілок та границь є його природний паралелізм, який витикає з описаної вище можливості розв'язання незалежних задач оптимізації на кожній з підмножин розбиття. Саме притаманність методу природного паралелізму, тобто можливості розпаралелювання без додаткових обмежень та модифікацій і стала причиною обрання його для реалізації різними засобами створення паралельних програм. До того ж паралельна реалізація методу досить пристойно зарекомендувала себе у роботі на кластерних системах.

Алгоритм паралельного варіанту методу гілок і границь має такий вигляд:

1. Введення даних.
2. Початкове розбиття допустимої множини на підмножини.
3. Розподілення отриманих підмножин між процесорами (ядрами).
4. Обчислення центра ваги, верхньої і нижньої оцінок кожним процесором для своїх підмножин.
5. Визначення підмножини з мінімальною оцінкою знизу.
6. Якщо не виконуються умови виходу (різниця між оцінкою зверху і оцінкою знизу менше заданої точності) або умова синхронізації процесів (виконана певна кількість ітерацій), то відбувається розбиття знайденої підмножини на дві нові підмножини і обчислення для них центрів ваги, верхньої і нижньої оцінок, інакше переходимо до пункту 8.
7. Видалення безперспективних підмножин і перехід до пункту 5.
8. Якщо на одному з процесів виконалася умова виходу з ітераційного

процесу, а на всіх інших – умова синхронізації, то відбувається порівняння нижніх оцінок перспективних підмножин. Якщо оцінка знизу перспективної підмножини є найменшою, то формується відповідь — повідомлення про завершення ітераційного процесу та перехід до пункту 11.

9. Якщо нема повідомлення про завершення ітераційного процесу, то відбувається видалення безперспективних підмножин за найменшою серед усіх оцінкою зверху.

10. Перерозподіл множин між ядрами та перехід до пункту 5.

2. Вибір і обґрунтування критеріїв розвитку

Застосування МГГ на етапі синтезу структур МЕМС потребує розв’язання двох підзадач, а саме: визначення алгоритму чи підходу до поділу множини альтернативних рішень на підмножини та визначення критерію згідно з яким необхідно проводити відкидання підмножин.

Метод поділу на підмножини на системному рівні передбачає використання єдиної технології виготовлення МЕМС (Рис.1), а критерій вибору підмножини – вартість виготовлення (хоча можливі інші критерії чи їх комбінації при потребі).



Рис. 1. Ієрархія технологій виготовлення МЕМС

3. Розроблення структурної схеми та програмного забезпечення підсистеми

Структура підсистеми для мінімізації альтернативних рішень містить декілька незалежних один від одного модулів. Такий підхід забезпечує можливість кардинальної зміни роботи її окремих частин без можливих втрат у решті функціоналу. Завдяки методам об'єктно-орієнтованого програмування вдалось перенести значну частину архітектури програми на максимально абстрактний рівень, що, в свою чергу, забезпечує гнучкість у зміні її окремих частин.

Програма складається з таких модулів:

Перелік робочих модулів програми

Таблиця 1.

Назва модулю	Функціонал	Особливості
Парсер вхідного файлу	Читання з файлу, верифікація даних, конвертація даних у внутрішню структуру	Вхідний файл може бути як переданий програмі, так вказаний перед тим. Алгоритм валідації даних що зчитуються легко замінити
SQL адаптер	З'єднання за БД, співставлення альтернативного розв'язку з його властивостями, збереження у БД, асинхронний зв'язок з БД	Властивості альтернативного розв'язку можуть бути зчитані з будь-якого іншого місця при коректній реалізації алгоритму запису значень
Синхронізуючий модуль	Співставлення отриманих альтернативних розв'язків з властивостями елементів, що містяться у БД	Синхронізація відбувається в окремому потоці, що дає змогу забезпечити максимальну інтерактивність користувацького інтерфейсу
Віртуалізуюча UI панель	Візуальне відображення множини альтернативних / мінімізованих розв'язків	Специфічний алгоритм віртуалізації UI елементів дає змогу відображати множини розв'язків досить великих порядків
Модуль побудови критерію відбору	Побудова критеріальної оцінки елементу альтернативного розв'язку	Алгоритм побудови критерію може бути доповнений/ змінений/ модифікований в залежності від бажань користувача
Конструктор альтернативних рішень	Зменшення потужності множини альтернативних рішень	Алгоритм (в даному випадку гілок та границь) може бути легко змінено чи доповнено

Гнучкість забезпечується існуванням базових класів/інтерфейсів, породження від яких надає можливість зміни функціоналу. Ядро програми ґрунтується на використанні цих абстрактних базових класів.

Робота підсистеми ґрунтується на основі взаємодії окремих модулів, кожен з яких можна дописати (змінити), без втручання в роботу підсистеми в цілому.

Вся робота програми керується ядром. Зчитавши вхідний файл, після команди користувача відбувається синхронізація зчитаних даних з записами в базі даних. Підчас синхронізації користувач має змогу переглянути отримані з БД альтернативні рішення. У випадку коли рішення містить відповідний AutoCAD файл, програма надає користувачеві можливість переглянути і його. У випадку наявності зображення елементу в базі даних воно буде відображене. Більшість операцій користувача з програмою може бути здійснено за допомогою «гарячих клавіш».

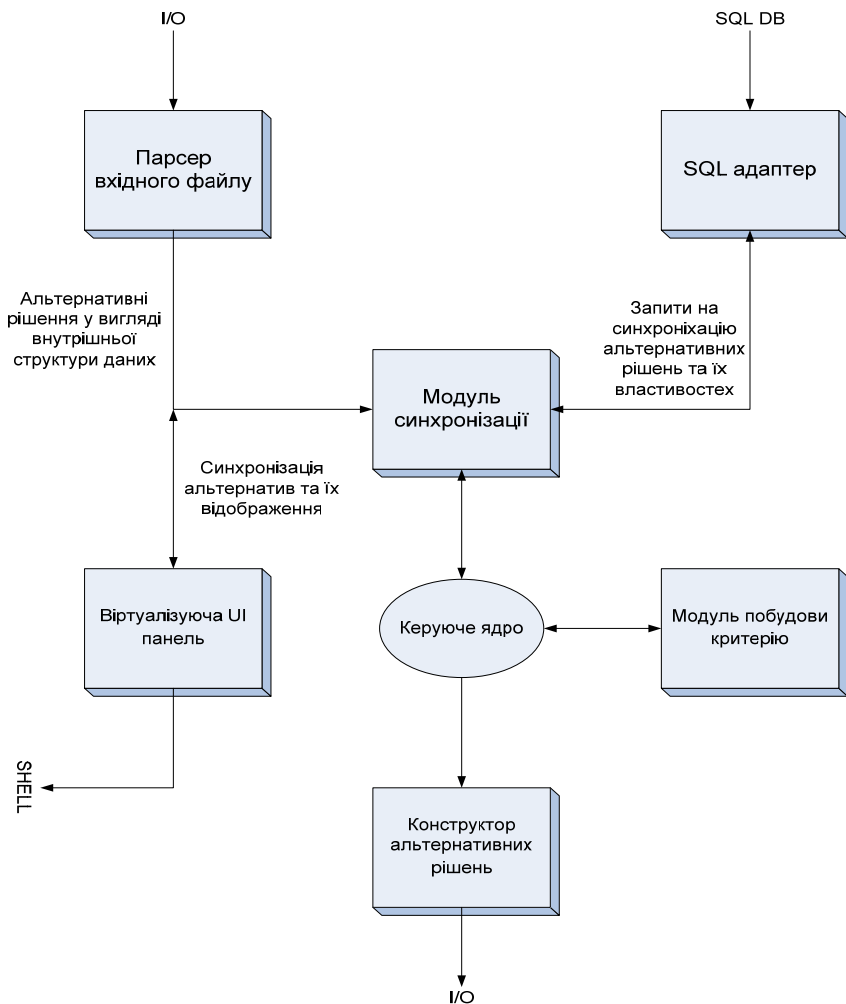


Рис. 2. Структурна схема підсистеми

4. Результати досліджень роботи підсистеми

Основним з параметрів роботи будь-якої системи є її швидкодія. Було прийнято рішення провести ряд тестувань підсистеми, зокрема: визначити час завантаження, опрацювання та обрахунку даних в залежності від їх кількісних характеристик.

Тестування часу завантаження даних (читання вхідного файлу, валідація, розпізнавання МЕМС елементів, тощо).

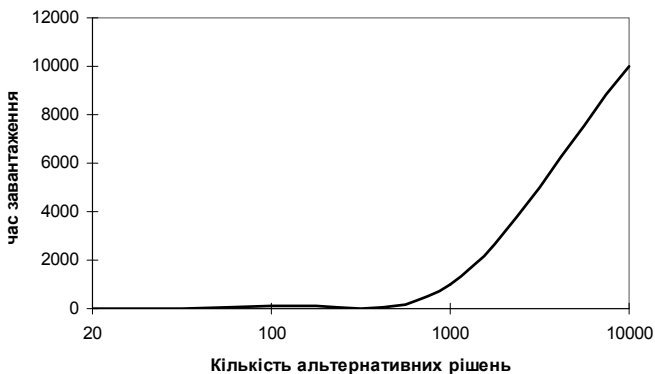


Рис. 3. Залежність часу завантаження даних від кількості альтернативних рішень.

Як видно з рис. 3, час завантаження значно зростає при використанні великих об'ємів вхідних даних. Це може бути обумовлене особливістю побудови зв'язаної спискової структури, що використовується для збереження даних.

Тестування часу синхронізації параметрів елементів в залежності від їх кількості наведено на рис.4. Критерій доступу до бази даних може змінюватись в залежності від каналу, що з'єднує підсистему і БД. Тестування проводилось за найбільш сприятливих умов: локально.

Тестування показує критичну кількість в чотири параметри, після якої час доступу зростає дещо повільніше. Граничним значенням критерію зчитування будемо вважати 35 мс.

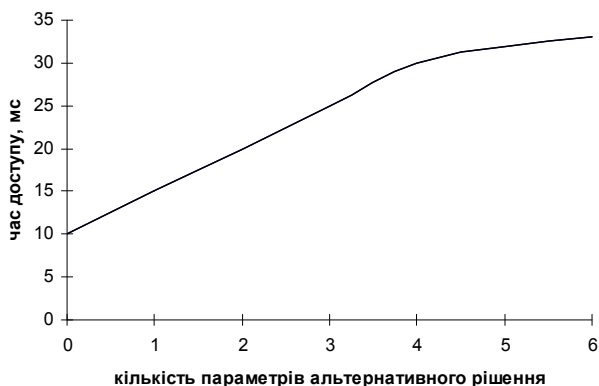


Рис.4. Залежність часу доступу до параметрів елементів пристрою від їх кількості.

Затрати часу на оптимізацію задачі в залежності від кількості альтернативних рішень наведено на рис.5. Відповідно до розробленого алгоритму, час мінімізації виступає в ролі одного з основних критеріїв.

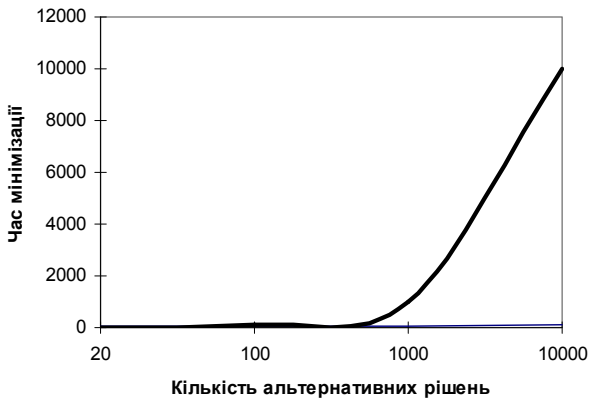


Рис. 5. Залежність часу мінімізації в залежності від кількості вхідних рішень.

Висновки

1. В роботі застосовано метод гілок та границь для зменшення потужності множини альтернативних рішень при вирішенні задачі синтезу структури MEMC.

2. Обґрунтовано вибір критеріїв для розбиття початкової множини альтернативних рішень на підмножини та відсіювання незадовільних варіантів. У випадку, коли множину альтернативних рішень складають варіанти структури MEMC даними критеріями є технологія виготовлення MEMC та їх вартість.

3. Розроблено програмне забезпечення, яке дає змогу автоматизувати процес зменшення потужності множини альтернативних рішень. Для збільшення швидкодії був використаний паралельний варіант методу гілок та границь, що дає змогу розподілити обчислювальні операції між ядрами процесора.

4. Проведено тестування підсистеми, які дали змогу отримати результати пов'язані з можливими шляхами підвищення ефективності розроблення MEMC на системному рівні і встановити, що залежність між затратами ресурсу персонального комп'ютера та розмірністю задачі носить поліноміальний характер.

1. *Теслюк В.М.* Моделі та інформаційні технології синтезу мікроелектромеханічних систем: Монографія. – Львів: Видавництво ПП "Вежа і Ко", 2008 – 192 с.

2. *Луцинин В.В.* Микросистемная техника. Направления и тенденции развития // Научное приборостроение. 1999. Т. 9. № 1. С. 3-18.

3. *Лысенко И.Е.* Проектирование сенсорных и актюаторных элементов микросистемной техники. – Таганрог: Изд-во ТРТУ. 2005. – 103 с.
4. Норенков И.П. Основы теории и проектирования САПР. – М.: Высш. Шк., 1990. – 334 с.
5. Системы автоматизированного проектирования: Учеб. пособие для вузов: В 9 кн. / *И.П.Норенков*. Кн.1. принципы построения и структура. – М.: Высш.шк., 1986. – 127 с.
6. Microsoft Corporation. Проектирование и реализация баз данных Microsoft SQL Server 2000. / Пер. с англ – 3-е изд. – Издательско-торговый дом “Русская Редакция”; СПб.: Питер, 2006.- 512стр.: ил.
7. *Троелсен, Эндрю.* Язык программирования С# и платформа .NET 2.0, 3-е издание. : Пер.с англ. – М.: ООО “И.Д.Вильямс”, 2007 – 1168с.: ил.

Поступила 30.01.2009р.

УДК 004.652.4+004.827

Н.Б.Шаховська, Д.І.Угрин

ІНТЕГРАЦІЯ РОЗРІЗНЕНИХ ДАНИХ ПРО КЛІЄНТА З ВИКОРИСТАННЯМ ІНТЕЛЕКТУАЛЬНОГО АГЕНТА ВИЗНАЧЕННЯ СТРУКТУРИ ДЖЕРЕЛА

An order, methods and facilities of getting, concordance, integration of client information, creation of operative depositories of information and load of information, is in-process worked out in a central depository.

ВСТУП

Туристичний бізнес – це сучасна галузь, що динамічно розвивається. Впровадження спеціалізованих програмних продуктів, розроблених винятково для потреб туристичної галузі, компанії-розроблювачі почали вже понад п'ятнадцять років тому. Проте наявність великої кількості програмних продуктів не дає змоги здійснювати обмін даними між туроператорами, що б дозволило значно підвищити їхні прибутки, оскільки дані зберігаються у різних моделях, керуються різними операційними системами тощо. Технологія Customer Data Integration (CDI) дозволяє збільшити швидкість і точність ідентифікації клієнтів і не потребує при цьому створення єдиного сховища даних про клієнтів. Ця технологія інтеграції відрізняється від попередніх тим, що в ній використовується так званий "заснований на знаннях" підхід. В основі CDI є каталог даних про клієнтів та опис методів доступу до інформації про них.