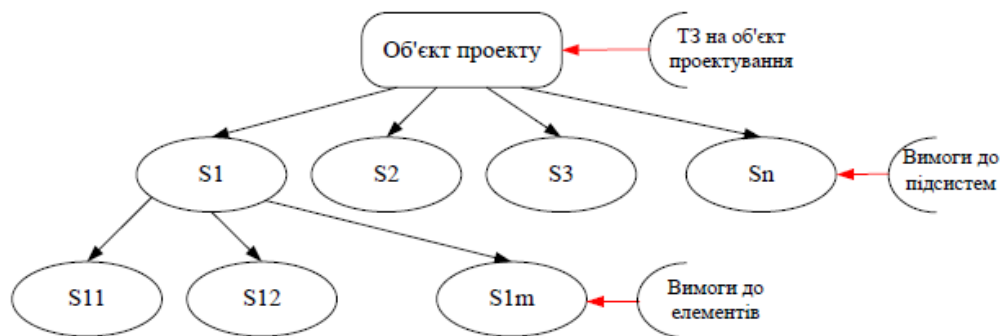


РОЗШИРЕНЕ ПРОЕКТУВАННЯ МІКРОСИСТЕМНИХ ПРИСТРОЇВ

Навчальний посібник



№530785-TEMPUS-1-2012-1-PL-TEMPUS-JPCR



Tempus

Навчальний посібник **"Розширене проектування мікросистемних пристроїв"** створено для допомоги вищим навчальним закладам України впровадити нову магістерську навчальну програму "Проектування мікросистем".

Посібник **"Розширене проектування мікросистемних пристроїв"** створено при підтримці Європейського Союзу за Спільним Європейським Проектом "Curricula Development for New Specialization: Master of Engineering in Microsystems Design" (MastMST), ідентифікаційний номер 530785-TEMPUS-1-2012-1-PL-TEMPUS-JPCR.

Координатор проекту проф. Збігнєв Лісік, Технічний університет м.Лодзь, Польща.

Учасники проекту:

- Національний університет "Львівська політехніка", м. Львів, Україна, координатор проф. Михайло Лобур.
- Київський Національний університет ім. Тараса Шевченка, м. Київ, Україна, координатор проф. Валерій Скришевський.
- Харківський національний університет радіоелектроніки, м. Харків, Україна, координатор проф. Володимир Хаханов.
- Донецький національний технічний університет, м. Красноармійськ, Україна, координатор проф. Володимир Святний.
- Технічний університет м.Ільменау, Німеччина, координатор проф. Іво Рангелов.
- Ліонський Національний інститут прикладних наук, Франція, координатор проф. Александра Апостолук
- Університет Павії, м.Павії, Італія, координатор проф. Паоло Ді Барба

Посібник схвалено редакційним комітетом (проф. Паоло Ді Барба (Університет Павії) - співголова, проф. Александра Апостолук(Ліонський Національний інститут прикладних наук) – співголова, члени: проф. Збігнєв Лісік(Технічний університет м.Лодзь), д-р Яцек Подгурські (Технічний університет м.Лодзь), Д-р Януш Возний (Технічний університет м.Лодзь), Д-р Валентин Іщук (Технічний університет м.Ільменау), Д-р Марія-Евеліна Могначі (Університет Павії), Д-р Роберто Галді (Університет Павії)) 6 травня 2016, м. Павія, Італія

Автори висловлюють глибоку вдячність керівництву вищеназваних університетів за всебічну підтримку Проекту.

Textbook "**Advanced Design for Microsystem Devices**" developed to help higher education institutions in Ukraine to introduce new master's educational program "Designing microsystems".

Textbook "**Advanced Design for Microsystem Devices**" was created with the support of the European Union within the Joint European Project "Curricula Development for New Specialization: Master of Engineering in Microsystems Design" (MastMST), identification number 530785-TEMPUS-1-2012-1-PL-TEMPUS-JPCR.

Project Coordinator prof. Zbigniew Lisik, Lodz University of Technology, Lodz, Poland.

Учасники проекту:

- Lviv Politechnical National University, Lviv, Ukraine ,
Coordinator prof. Mykhailo Lobur.
- Taras Shevchenko National University of Kyiv, Ukraine,
Coordinator prof. Valeriy Skryshevsky.
- Kharkiv National University of Radioelectronics, Ukraine ,
Coordinator prof. Vladimir Hahanov.
- Donetsk National Technical University, Krasnoarmiysk,
Coordinator prof. Volodymyr Sviatny.
- Ilmenau University of Technology, Germany,
Coordinator prof. Ivo Rangelow.
- Lyon Institute of Applied Sciences, France,
Coordinator prof. Alexandra Apostoluk.
- University of Pavia, Italy,
Coordinator prof. Paolo Di Barba.

The Handbook was approved by Editorial Committee (prof. Paolo Di Barba (University of Pavia) - Co-Chair, prof. Alexandra Apostoluk (Lyon Institute of Applied Sciences) – Co-Chair, members: prof. Zbigniew Lisik (Lodz University of Technology), Dr Jacek Podgorski (Lodz University of Technology), Dr Janusz Wozny (Lodz University of Technology), Dr Valentyn Ishchuk (Ilmenau University of Technology), Dr Maria Evelina Mognaschi (University of Pavia), Dr Roberto Galdi (University of Pavia) May 6, 2016, Pavia, Italy.

The authors express their deep gratitude to the aforementioned universities for full support of the project.

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

Національний університет "Львівська політехніка"

*Василь Теслюк, Андрій Зелінський,
Володимир Каркульовський, Ярослав Василюк*

**ADVANCED DESIGN FOR
MICROSYSTEMS DEVICES**

Навчальний посібник

**Підготовлено в рамках проекту
№530785-TEMPUS-1-2012-1-PL-TEMPUS-JPCR**

ЛЬВІВ 2016

Теслюк В.М., Зелінський А. Я., Каркульовський В.І., Василюк Я.Р.

Advanced design for microsystems devices: Навчальний посібник. – Львів: Видавництво Національного університету "Львівська політехніка", 2016. – 200с.

Робота виконана в рамках проекту **Curricula Development for New Specialization: Master of Engineering in Microsystems Design / MastMST**, Ідентифікаційний номер **517238-TEMPUS-1-2011-1-PL-TEMPUS-JPCR**. Програма фінансування ЄК: Міжрегіональна програма Європейського сусідства і партнерства.

Цільова група: студенти, випускники, викладачі та адміністрація університетів, керівники промислових підприємств, міністерство освіти і науки України.

В посібнику розглянуто особливості розв’язання задач аналізу та синтезу з використанням сучасних методів та моделей в процесі автоматизованого проектування МЕМС на системному рівні проектування. Для розв’язання сформульованих задач використовується методи багатокритеріальної оптимізації, морфологічний метод, методи синтезу з використанням онтологій і теорії чутливості та методів діакоптичного підходу.

Запропонований посібник буде корисним студентам, аспірантам та науковцям, які спеціалізуються в області автоматизованого проектування та математичного моделювання мікроелектромеханічних систем.

ВСТУП

Складність сучасних технічних систем зростає з кожним роком, нарощується їх функціональна та інтелектуальна складові. В таких умовах все частіше використовуються технології мікроелектромеханічних систем (МЕМС) для покращення технічних параметрів проєктованих виробів. Відповідно, автоматизація проєктування МЕМС є гострою проблемою сьогодення.

В запропонованому посібнику, в першому розділі, розглянуто особливості МЕМС, специфіка їх проєктування на ієрархічних рівнях в процесі використання блочно-ієрархічного підходу. Особлива увага приділена аспектам проєктування МЕМС, які дають змогу врахувати мультидисциплінарність мікросистем.

В другому розділі посібника описано особливості задач синтезу та аналізу МЕМС, наведено їх класифікацію. Розглянуто підходи та методи розв'язування типових задач синтезу та аналізу.

У третьому розділі описано особливості проєктування МЕМС з використанням конструкторської бази даних (КБД). Наведено основні функції КБД, її архітектура, програмна та інформаційна моделі.

Четвертий розділ посібника включає інформацію про особливості наповнення КБД МЕМС. Зокрема, класифікацію базових елементів МЕМС та базових структур мікросистем. Наведено приклад використання КБД МЕМС.

В п'ятому, шостому та сьомому розділах описано основні методи розв'язання задач багатокритеріальної оптимізації, які використовують в процесі структурного та параметричного синтезу МЕМС.

У восьмому розділі посібника описано особливості застосування морфологічного методу при розв'язанні задачі структурного синтезу.

Дев'ятий розділ присвячено розв'язанню задач структурного та параметричного синтезу з використанням онтологій, методів діакоптичного підходу та теорії чутливості в процесі проєктування МЕМС.

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1. ОСОБЛИВОСТІ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ MEMC	8
1.1. Специфіка мікроелектромеханічних систем	8
1.2. Застосування блочно-ієрархічного підходу до автоматизованого проектуювання MEMC	10
1.3. Особливості автоматизованого проектування MEMC на ієрархічних рівнях	12
1.4. Аспекти автоматизованого проектування MEMC	23
1.5. Контрольні запитання до розділу	24
1.6. Список використаної літератури до першого розділу	25
РОЗДІЛ 2. ОСНОВИ ЗАДАЧ СТРУКТУРНОГО ТА ПАРАМЕТРИЧНОГО СИНТЕЗУ MEMC	27
2.1. Поняття задач структурного та параметричного синтезу	28
2.2. Класифікація задач синтезу	32
2.2.1. Класифікація задач параметричного синтезу	32
2.2.2. Класифікація задач структурного синтезу	38
2.3. Методи розв'язання задач параметричного синтезу	40
2.4. Підходи до розв'язання задач структурного синтезу	43
2.5. Контрольні запитання до розділу	47
2.6. Список використаної літератури до другого розділу	49
РОЗДІЛ 3. АВТОМАТИЗАЦІЯ ПРОЕКТУВАННЯ MEMC ВИКОРИСТАННЯМ КОНСТРУКТОРСЬКОЇ БАЗИ ДАНИХ	50
3.1. Архітектура підсистеми роботи з КБД MEMC	50
3.2. Конструкторсько-технологічна база даних елементів MEMC	53
3.3. Побудова інформаційної моделі КБД	58
3.4. Розроблення програмної моделі представлення MEMC в КБД	65
3.5. Розроблення лінгвістичного забезпечення системи управління КБД	71
3.6. Контрольні запитання до розділу	75
3.7. Список використаної літератури до третього розділу	75

РОЗДІЛ 4. ОСОБЛИВОСТІ НАПОВНЕННЯ КОНСТРУКТОРСЬКОЇ БД СКЛАДОВИХ МЕМС	77
4.1. Класифікація складових МЕМС	77
4.2. Особливості розроблення базових складових МЕМС	86
4.3. Особливості розроблення базових структур МЕМС	107
4.4. Особливості реалізації підсистеми керування КБД	111
4.5. Контрольні запитання до розділу	118
4.6. Список використаної літератури до четвертого розділу	118
РОЗДІЛ 5. ОСНОВИ ЗАДАЧ БАГАТОКРИТЕРІАЛЬНОЇ ОПТИМІЗАЦІЇ	123
5.1. Основні елементи теорії оптимізації	123
5.2. Особливості об'єкта оптимізації	127
5.3. Постановка оптимізаційної задачі	129
5.4. Причини багатокритеріальності в задачах оптимального проектування МЕМС	132
5.5. Постановка задачі багатокритеріальної оптимізації	134
5.6. Контрольні запитання до розділу	135
5.7. Список використаної літератури до п'ятого розділу	135
РОЗДІЛ 6. МЕТОДИ РОЗВ'ЯЗАННЯ БАГАТОКРИТЕРІАЛЬНИХ ЗАДАЧ НА ОСНОВІ ОДНОКРИТЕРІАЛЬНОЇ ОПТИМІЗАЦІЇ	137
6.1. Метод головної компоненти	138
6.2. Лексикографічний метод	143
6.3. Метод поступок	148
6.4. Методи цільового програмування	152
6.4.1. Метод умовного центра мас	153
6.4.2. Метод ідеальної точки	157
6.5. Контрольні запитання до розділу	158
6.6. Список використаної літератури до шостого розділу	160
РОЗДІЛ 7. МЕТОДИ РОЗВ'ЯЗАННЯ ЗАДАЧ БАГАТОКРИТЕРІАЛЬНОЇ ОПТИМІЗАЦІЇ З ВИКОРИСТАННЯМ УЗАГАЛЬНЕНОГО (ІНТЕГРАЛЬНОГО) КРИТЕРІЮ ОПТИМАЛЬНОСТІ	161

7.1. Нормалізація узагальненого критерію	162
7.2. Адитивний критерій	162
7.3. Мультиплікативний критерій	166
7.4. Максимальний (мінімальний) критерій	171
7.5. Методи визначення значень вагових коефіцієнтів	176
7.5.1. Метод ранжування	176
7.5.2. Метод приписування балів	178
7.6. Контрольні запитання до розділу	179
7.7. Список використаної літератури до сьомого розділу	180
РОЗДІЛ 8. ЗАСТОСУВАННЯ МОРФОЛОГІЧНОГО МЕТОДУ ДО АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ MEMC	181
8.1. Розроблення моделі для проектування MEMC на системному рівні	182
8.2. Побудова І-АБО дерева для інформаційної технології синтезу конструкції MEMC на системному рівні	184
8.3. Особливості розроблення продукційних правил	187
8.4. Розроблення інформаційної моделі альтернативного рішення	188
8.5. Розроблення методу зменшення потужності множини альтернативних рішень	191
8.6. Розроблення методу зменшення потужності множини альтернативних рішень на методу гілок та границь	193
8.7. Застосування методу гілок і границь для синтезу структурних схем MEMC	198
8.7.1. Побудова критерію відбору груп альтернативних рішень структур MEMC	204
8.7.2. Побудова критерію оцінки груп альтернативних розв'язків структур MEMC	207
8.8. Розроблення основних алгоритмів для синтезу структурних схем MEMC	210
8.9. Побудова інформаційних моделей для реалізації методу гілок і границь	214

	7
8.10. Контрольні запитання до розділу	216
8.11. Список використаної літератури до восьмого розділу	217
РОЗДІЛ 9. СУЧАСНІ МЕТОДИ СТРУКТУРНОГО ТА ПАРАМЕТРИЧНОГО СИНТЕЗУ MEMC	220
9.1. Застосування теорії чутливості в процесі автоматизованого проектування MEMC	220
9.2. Онтологічний підхід до автоматизації структурного синтезу MEMC	223
9.2.1. Особливості застосування онтологічного підходу для проектування MEMC	223
9.2.2. Особливості побудови онтологічної моделі	225
9.2.3. Аналіз моделей та основних методів представлення онтологій	226
9.2.4. Розроблення моделі онтології для елемента, підсистеми та системи	231
9.2.5. Приклади застосування онтологій для проектування MEMC	236
9.2.6. Особливості реалізації програмного забезпечення модуля роботи з онтологією	240
9.2.7. Результати роботи підсистеми взаємодії з онтологією	245
9.3. Методи діакоптичного підходу в процесі проектування MEMC	248
9.4. Контрольні запитання до розділу	248
9.5. Список використаної літератури до дев'ятого розділу	250

РОЗДІЛ 1.

ОСОБЛИВОСТІ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ MEMS

1.1. Специфіка мікроелектромеханічних систем

Мікроелектромеханічні системи з'явилися в процесі розвитку мікротехнологій, на яких базувалася мікроелектроніка [1, 14]. К. Петерсон (фірма IBM) в 1982 р опублікував статтю [15], у якій він вперше звернув увагу на появу нового наукового та прикладного напрямку [16], який ґрунтувався на використанні методів та підходів напівпровідникової мікротехнології та використанні кремнію як матеріалу, не в цілях мікроелектроніки.

Важливо відзначити, що поштовхом для становлення обох напрямків є технічні досягнення, а не загальні теоретичні положення, як було в історії робототехніки [17].

З вище наведеного можна зробити висновок, що MEMS – природне продовження мікроелектроніки, технології виробництва якої використані для створення мікромеханічних і мікроелектромеханічних систем та інших мікросистем, пов'язаних функціональними зв'язками, на основі об'єму твердого тіла. Таким чином, MEMS відрізняється від мікроелектроніки розширенням предметної області та використанням у зв'язку з цим нових мікротехнологій.

З початку 90-х років досить багато комерційних компаній та університетів по всьому світу інтенсивно ведуть розробки в області розвитку технологій MEMS. У зв'язку з цим з'явилась велика кількість нових технічних рішень і до 2015 року різко зросла загальна кількість зареєстрованих патентів в області технологій MEMS. На даний час, в цій галузі щорічно реєструється більше 300 патентів [15, 18 – 20].

Історично склалося так, що перші пристрої MEMS були розроблені та виготовлені набагато раніше від появи окремої науково-технічної області MEMS. Так в 1961 році був продемонстрований перший кремнієвий давач тиску, а в 1967 р. – розроблена технологія поверхневої мікромеханічної

опрацювання з використанням “жертвовного” шару для відокремлення мікрOMEханічного пристрою від його основи. Перший кремнієвий акселерометр був продемонстрований в 1970 році, а в 1980 році – виготовлений перший одноразовий мікродавач кров’яного тиску. В 1982 році був розроблений LIGA процес (LIGA – скорочення німецьких слів, що описують стадії процесу: Lithographie – літографія, Galvanoformung – гальваніка, Abformung – формування) для виготовлення MEMC, а перша конференція, присвячена MEMC відбулася в 1988 р. (перший комерційний акселерометр – 1993 р., BioMEMS -1995 р., OptoMEMS - 2000 р.). Починаючи з 90-х років минулого сторіччя і до нашого часу, практично кожний рік характеризувався появою нових пристроїв MEMC, з використанням нових фізичних, хімічних і ефектів іншої природи та впровадженням їх в нові галузі людського життя [21].

Науково-прикладна область мікроелектромеханічних систем [1 – 4] акумулює в собі найновіші досягнення більшості наукових та інженерних галузей (рис.1.1) і належить до так званих “критичних технологій”, які визначають рівень інтелектуального потенціалу нації та здатність держави забезпечити її розвиток. Країни, які в ранг державних пріоритетів відносять науково-технічний прогрес, домінують на світовому ринку нових інформаційних технологій, а в наступні роки розвитку людства ця тенденція ще більше буде посилюватися.

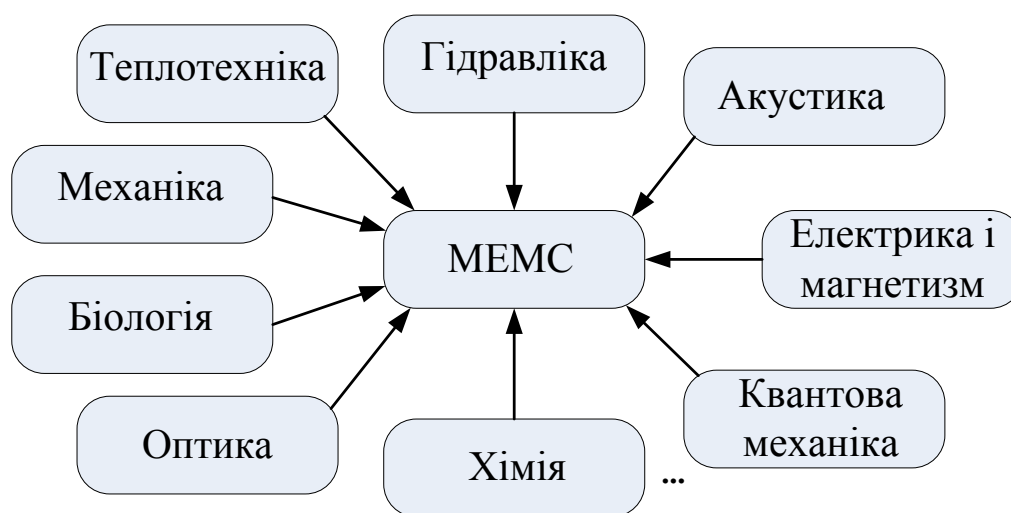


Рис.1.1. Основні складові MEMC

1.2. Застосування блочно-ієрархічного підходу до автоматизованого проектування МЕМС

Найбільш перспективними на сьогоднішній день є наступні технології проектування: наскрізного проектування, паралельного проектування, низхідного та висхідного проектування, CALLS технології [1, 2], та інші.

Наскрізне проектування. Зміст наскрізної технології проектування, полягає в ефективній передачі даних і результатів конкретного поточного етапу розроблення відразу на всі наступні етапи. Це потрібно для того, щоб на необхідному рівні виконати який-небудь з етапів проектування. Розробнику часто не вистачає регламентованої інформації від попереднього етапу і необхідна більш повна та різноманітна інформація, що могла бути сформульована на одному з ранніх етапів проектування (не обов'язково на сусідньому). У розробників, які виконують різні етапи проектування, може бути одночасно з першим етапом проектування отримане технічне завдання і таким чином, усі розробники можуть одночасно почати продумувати як більш успішно реалізувати свій етап. Дана технологія ґрунтується на модульній побудові САПР, на використанні загальних баз даних і баз знань, і характеризується широкими можливостями моделювання та контролю на всіх етапах проектування. Наскрізні САПР, як правило, є інтегрованими, тобто мають альтернативні алгоритми реалізації окремих проектних процедур.

Паралельне проектування. При паралельному проектуванні, інформація щодо яких-небудь проміжних чи остаточних характеристик розроблювального виробу, формується і надається всім учасникам робіт, починаючи із самих ранніх етапів проектування. В цьому випадку інформація носить прогностичний характер. Її одержання базується на математичних моделях і методах прогностичної оцінки критеріїв якості проектного рішення. Оцінка може вироблятися на основі аналітичної моделі, статистичних методах і методах експертних систем. Технологія паралельного проектування реалізується на основі інтегрованих інструментальних засобів прогностичної оцінки й аналізу альтернативних проектних рішень з наступним вибором

базового проектного рішення. Очікується, що інженер починає працювати над проектом на високому рівні абстракції з наступною деталізацією проекту.

Принциповою відмінністю паралельного проектування від наскрізного проектування (хоча паралельне проектування одержало розвиток на основі наскрізного) у тому, що інформація не просто надходить на всі наступні етапи проектування, але і власне кажучи ці етапи починають виконуватися одночасно.

Фірма MENTOR GRAPHICS вперше створила середовище паралельного проектування на основі принципу об'єднання всіх інструментальних засобів проектування і даних в одному безупинному і гнучкому процесі створення виробу.

До складу цієї інфраструктури входять: середовище керування проектуванням, система керування даними проекту, система підтримки прийняття рішень (СППР).

Низхідне проектування. Передбачається, що інженер починає працювати над проектом на високому рівні абстракції з наступною деталізацією проекту. Основною задачею керівника чи інженера є визначення оптимального концептуального рішення, вибір функціональних алгоритмів проектування, а також вибір найбільш ефективних інструментальних засобів проектування.

Іншими словами визначення правильної стратегії проектування на основі досить загальної і найчастіше невизначеної інформації. Дана програма вирішується на основі застосування предиктивних інструментальних засобів, тобто програм, які забезпечують зв'язок етапів функціонально-логічного, конструкторського проектування й етапу технологічної підготовки виробництва. При цьому предиктивний інструментарій використовується як на рівні окремих проектних процедур, так і на рівні проекту в цілому. Низхідне проектування дає можливість одержати виріб з іншими характеристиками, створити надійний пристрій. Як правило, усі сучасні виробники працюють на рівні низхідного проектування. Хоча, по мірі накопичення інформації та результатів проектування певної групи об'єктів, все частіше з часом та накопиченим досвідом починає використовуватися висхідне проектування. Для

прикладу, це проектування електричних цифрових схем на основі базових комірок, проектування автомобілів зі стандартних деталей та вузлів тощо.

CALLS технологія. Основна ідея полягає в створенні електронного опису і супроводу виробу на всіх етапах його життєвого циклу. Електронний опис має відповідати прийнятим вітчизняним і міжнародним стандартам у даній предметній області. Це технологія інформаційного супроводу створення виробу.

1.3. Особливості автоматизованого проектування МЕМС на ієрархічних рівнях

На рис. 1.2 зображено типову послідовність проектних процедур ієрархічного рівня на одному з етапів розробки „зверху-вниз” [1].

На попередньому етапі розв’язують задачі k -го ієрархічного рівня, одним з результатів розв’язання цих задач, при проектуванні „зверху-вниз”, є формулювання технічного завдання (ТЗ) на розробку системи на $(k+1)$ -му ієрархічному рівні.

Розроблення системи починається із синтезу вихідного варіанта її структури. Для оцінки цього варіанта створюється модель: математична – при автоматизованому проектуванні, експериментальна чи стендова модель при неавтоматизованому проектуванні. Після вибору вихідних значень параметрів елементів проводиться аналіз отриманого варіанта рішення, за результатами якого стає можлива його оцінка. Звичайно оцінка полягає в перевірці виконання умов працездатності об’єкта, сформульованих у ТЗ. Якщо умови працездатності виконуються належною мірою, то отримане проектне рішення приймається, система $(k+1)$ -го рівня описується в прийнятій формі та формулюються ТЗ на розроблення елементів даного рівня (тобто систем наступного ієрархічного рівня). Якщо ж отримане проектне рішення незадовільне, то вибирається один з можливих шляхів покращення проекту.

Звичайно, найпростіше провести зміни числових значень параметрів елементів, що складають вектор вхідних даних X .

Сукупність процедур модифікації вектора X , аналізу та оцінки результатів аналізу являє собою процедуру параметричного синтезу. Якщо модифікації вектора X ціленаправлені та підлеглі стратегії пошуку направлені на покращення значення деякого показника якості, то процедура параметричного синтезу є процедурою оптимізації. Можливо, що шляхом параметричного синтезу не вдасться домогтися прийняттого ступеня виконання умов працездатності пристрою. Тоді використовують інший шлях, пов'язаний з модифікаціями структури пристрою.

Новий варіант структури синтезується, і для нього повторюються процедури формулювання моделі і параметричного синтезу. Якщо не вдасться одержати прийнятне проектне рішення і на цьому шляху, то ставиться питання про коректування ТЗ, сформульованого на попередньому етапі розробки. Таке коректування може вимагати повторного виконання ряду процедур k -го ієрархічного рівня, що й обумовлює ітераційний характер проектування.

Інформація, зображена на рис. 1.2 дає можливість встановити характерну рису взаємозв'язку проектних процедур аналізу та синтезу [1]. Цей взаємозв'язок має характер вкладеності процедур аналізу в процедуру оптимізації (параметричного синтезу) і процедури оптимізації в процедуру синтезу, що поєднує синтез структурний і параметричний. Вкладеність процедур показана на рис. 1.3.

Вкладеність означає, по-перше, що аналіз входить як складова частина в оптимізацію, а оптимізація в синтез і по-друге, однократне виконання оптимізації потребує багаторазового виконання аналізу, а однократне виконання синтезу потребує багатократного виконання процедури оптимізації. Очевидно, що такий же характер взаємодії мають процедури аналізу – однократний багатоваріантний аналіз заснований на багаторазовому одноваріантному аналізі. Неважко підрахувати, що синтез проектного рішення на черговому етапі розробки може вимагати виконання надмірно великої кількості варіантів аналізу і відповідно надзвичайно великих затрат машинного часу та людських ресурсів.

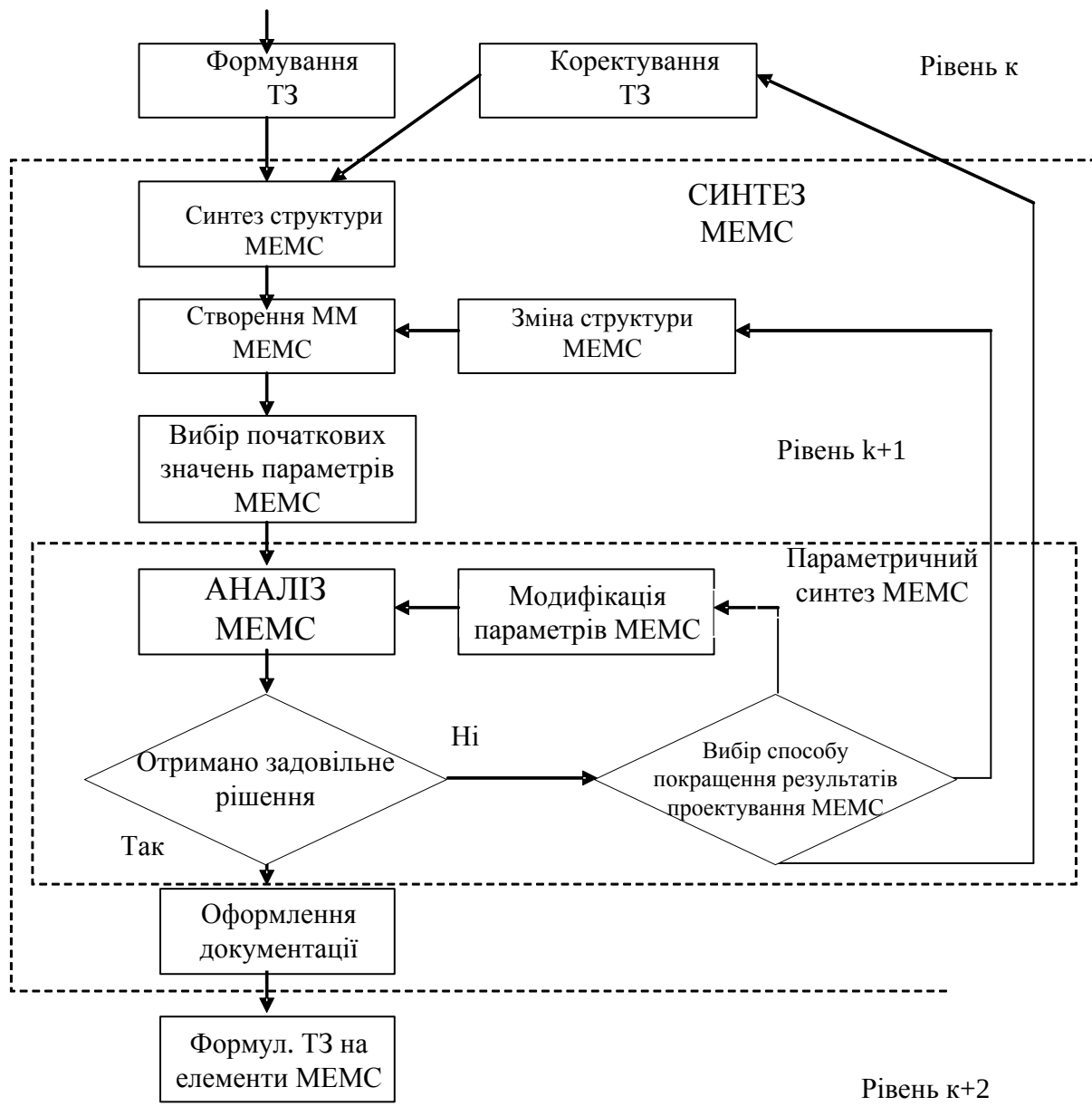


Рис. 1.2. Схема процесу проектування на ієрархічному $(k+1)$ -му рівні розроблення системи

Один зі шляхів рішення цієї проблеми – застосування достатньо точних і складних математичних моделей і алгоритмів аналізу тільки на завершальних ітераціях синтезу. Для більшості розглядуваних варіантів структури, при цьому, виконується лише попередня оцінка на основі непрямих критеріїв, спрощених моделей та алгоритмів. Така оцінка дасть змогу, без суттєвих витрат обчислювальних ресурсів, відсіяти більшість безперспективних варіантів і залишити для ретельного аналізу мале число варіантів.



Рис. 1.3. Вкладеність проектних процедур

Розроблення MEMС, на відміну від розробки макропристроїв, має свою специфіку та особливості. Структурна схема проектування мікроелектромеханічної системи для одного з ієрархічних рівнів наведена на рис.1.4. Проектування будь-якого об'єкта передбачає на першому етапі науково-дослідні роботи, які дають змогу сформулювати технічне завдання (ТЗ) на мікроелектромеханічний пристрій (МЕМП). Згідно сформульованого ТЗ виконується операція синтезу структури MEMС. В ряді робіт запропоновано використовувати бібліотеку елементів, компонент та пристроїв MEMС (БЕКП MEMС) і модуль технології виготовлення інтегральних пристроїв БЕКП MEMС, що дає можливість на базі стандартних конструкцій елементів, компонент та пристроїв синтезувати конструкцію MEMС, а, при розробленні нових конструкцій пристроїв MEMС скористатись вже набутим досвідом в області проектування пристроїв даного класу [2].

Модуль технології призначений для дослідження можливості виготовлення даного інтегрального пристрою за допомогою відомих стандартних та нестандартних інтегральних технологій.

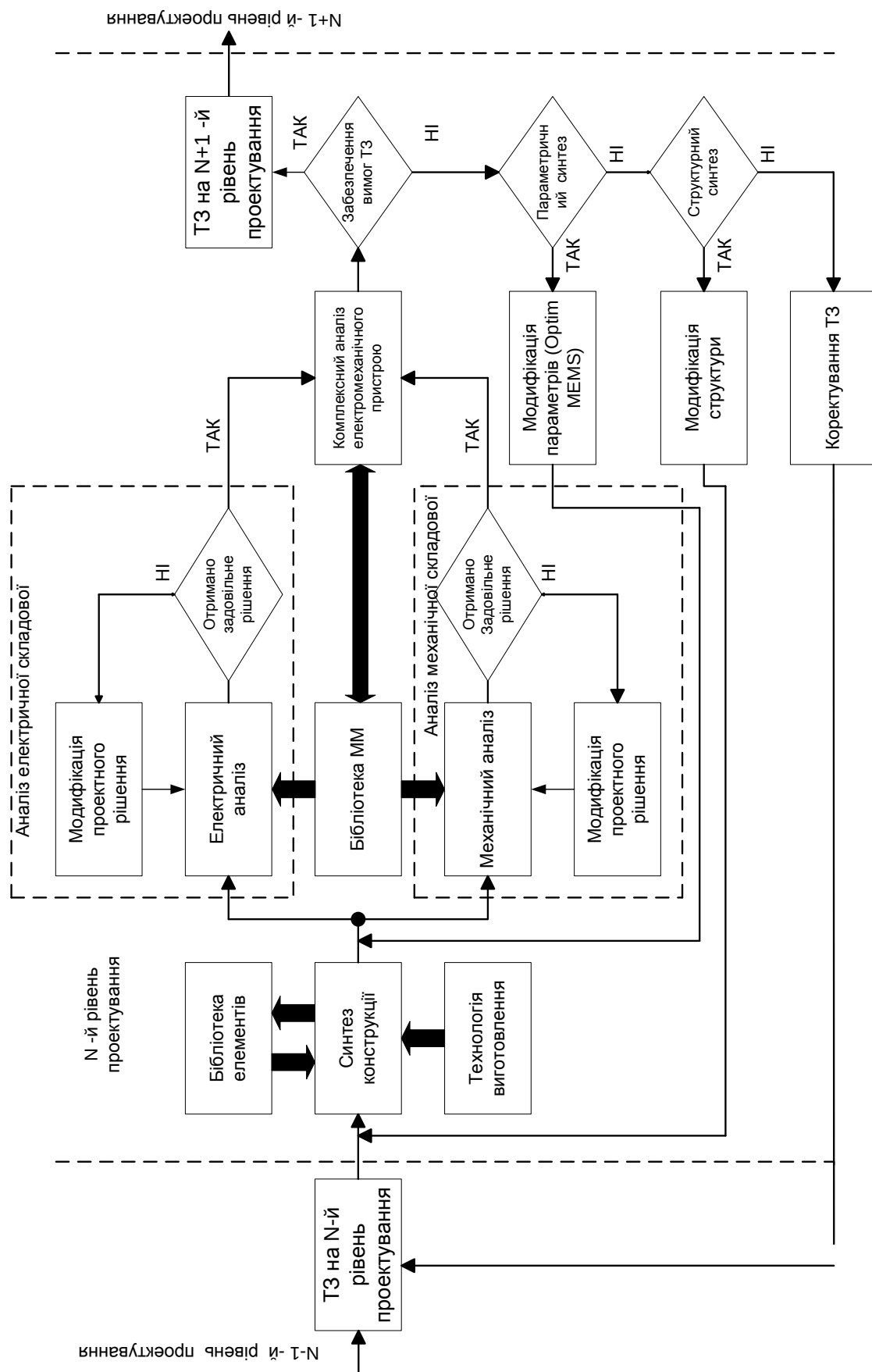


Рис. 1.4. Структурна схема проектування MEMS на одному з ієрархічних рівнів

При проектуванні електромеханічного інтегрального пристрою, його вхідними та вихідними даними є як електричні параметри (ємність, напруга та ін.), так і механічні та конструктивні параметри (товщина балки, жорсткість пружини, тощо). Для аналізу синтезованої конструкції, аналіз механічної та електричної складових, проводиться окремо з використанням бібліотеки математичних моделей. Відповідні операції є циклічними, а вихід з циклу виконується тоді, коли вихідні механічні або електричні параметри спроектованого інтегрального пристрою задовольняють ТЗ.

Наступний крок передбачає комплексний аналіз роботи електромеханічного інтегрального пристрою. У випадку забезпечення параметрів ТЗ – процес проектування електромеханічного елемента МЕМС завершується. В разі невиконання параметрів ТЗ виконується операція параметричного чи структурного синтезу з використанням підсистеми оптимізації, а в окремих випадках може виникнути потреба в операції коректування технічного завдання.

Як було зазначено вище, МЕМС можуть бути побудовані на основі дії різних фізичних принципів. Так, для прикладу, якщо необхідно спроектувати електротермічний мікропристрій, то в структурній схемі на рис.1.4 необхідно блок аналізу механічної складової замінити на блок термічного аналізу, який зображено на рис.1.5.

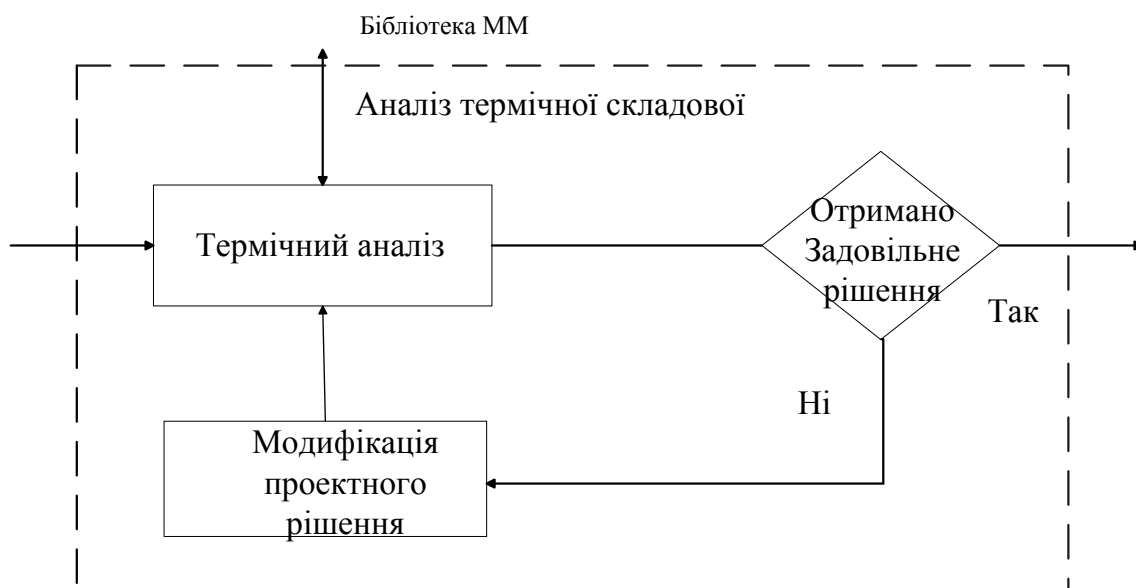


Рис.1.5. Блок для аналізу термічної складової

Можлива й інша ситуація, коли до стандартної схеми на рис.1.4 до блоку аналізу механічної складової необхідно додати паралельно ще один блок. Така ситуація може виникнути при проектуванні гідроелектромеханічних пристроїв. Приклад блоку для гідравлічного аналізу наведений на рис.1.6.

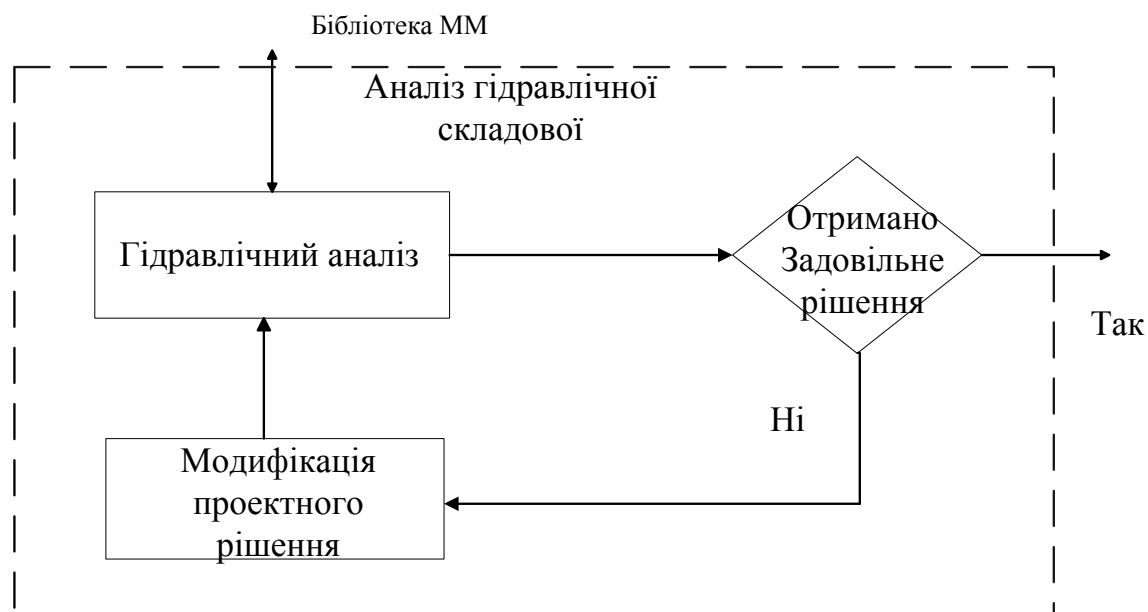


Рис. 1.6. Блок для аналізу гідравлічної складової інтегрального пристрою

Аналогічні блоки використовуються при аналізі оптичної, магнітної, акустичної, механічної обертової складових та інших.

Разом з тим, використовувати паралельний аналіз складових інтегрального пристрою, не завжди доцільно, хоча такий підхід дає можливість значно прискорити процес проектування інтегрального виробу. Це пояснюється тим, що отримані рішення досить важко поєднати в одному пристрої, так як, вони можуть бути взаємозаперечувальними на етапі комплексного аналізу роботи виробу. В таких випадках використовують послідовний аналіз складових інтегрального пристрою. Приклад відповідної схеми для інтегральної електромеханічної системи наведено на рис.1.7.

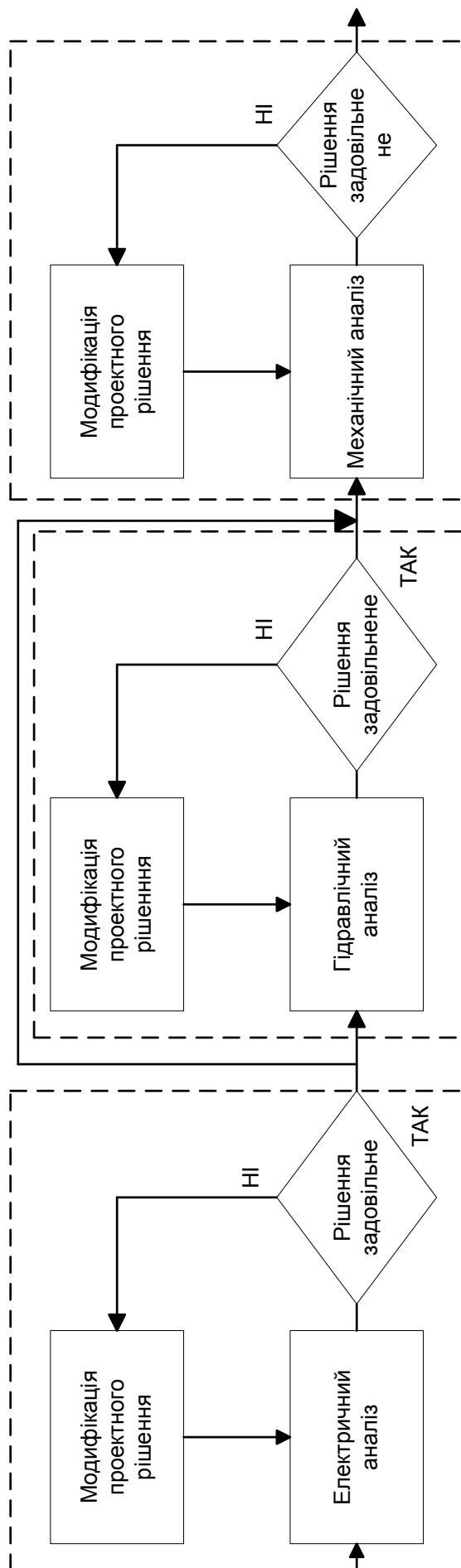


Рис.1.7. Приклад модуля трикомпонентного наскрізного аналізу

Основна ідея цього підходу полягає в тому, щоб результати електричного аналізу були враховані на етапі аналізу механічної складової при проходженні першої ітерації. У випадку, коли необхідно провести аналіз по трьох складових, використовується наскрізне проектування на ієрархічному рівні (рис.1.8), основна ідея якого полягає в тому, що результати після аналізу першої складової передаються паралельно на вхід другої та третьої складових. Особливістю проектування складових MEMC на функціональному рівні є те, що на цьому рівні розв'язують задачі кінематики, які направлені на визначення максимальних зміщень, траєкторій руху складових конструкції тощо. Специфіку проектування MEMC на схемотехнічному та компонентному рівнях розглянуто в роботах [3 - 7].

Підвищити якість та швидкість проектування пристроїв MEMC можна за допомогою використання методів, пов'язаних з визначенням чутливості та побудови таблиці кореляційних залежностей між вхідними та вихідними параметрами [8]. На рис.1.9 наведена схема проектування MEMC з використанням модуля чутливості та модуля визначення кореляційних залежностей між параметрами різної фізичної природи.

Приклад організації оптимізації параметрів пристрою MEMC, який передбачає визначення матриці чутливості, на одному з ієрархічних рівнів наведено на рис.1.9.

Визначення матриці чутливостей між вхідними та вихідними параметрами MEMC (електричними, механічними, магнітними, термічними та іншими), дає можливість ціленаправлено змінювати вхідні параметри для забезпечення вимог ТЗ на даному ієрархічному рівні автоматизованого проектування.

Математичний апарат теорії чутливості та кореляції [9 - 12] досить широко використовується в різних областях науки та техніки. Зокрема, при розв'язанні задач параметричної оптимізації, призначення допусків тощо.

Разом з тим, при розв'язанні задач аналізу та синтезу в процесі проектування пристроїв MEMC, виникає ряд проблем, а саме: при розв'язанні задач параметричної оптимізації.

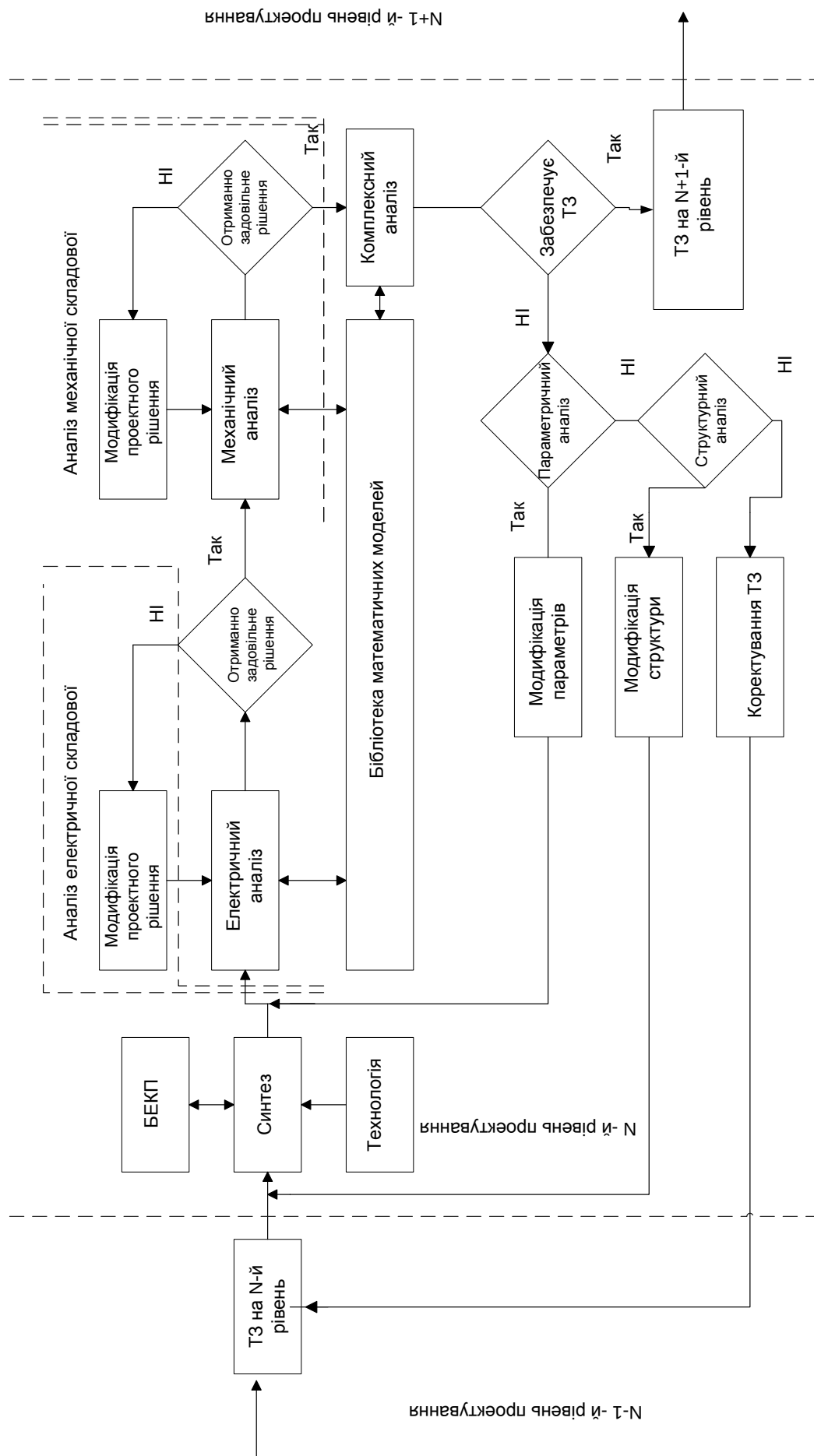


Рис. 1.8. Структурна схема проектування МЕМС з послідовним аналізом складових інтегрального пристрою

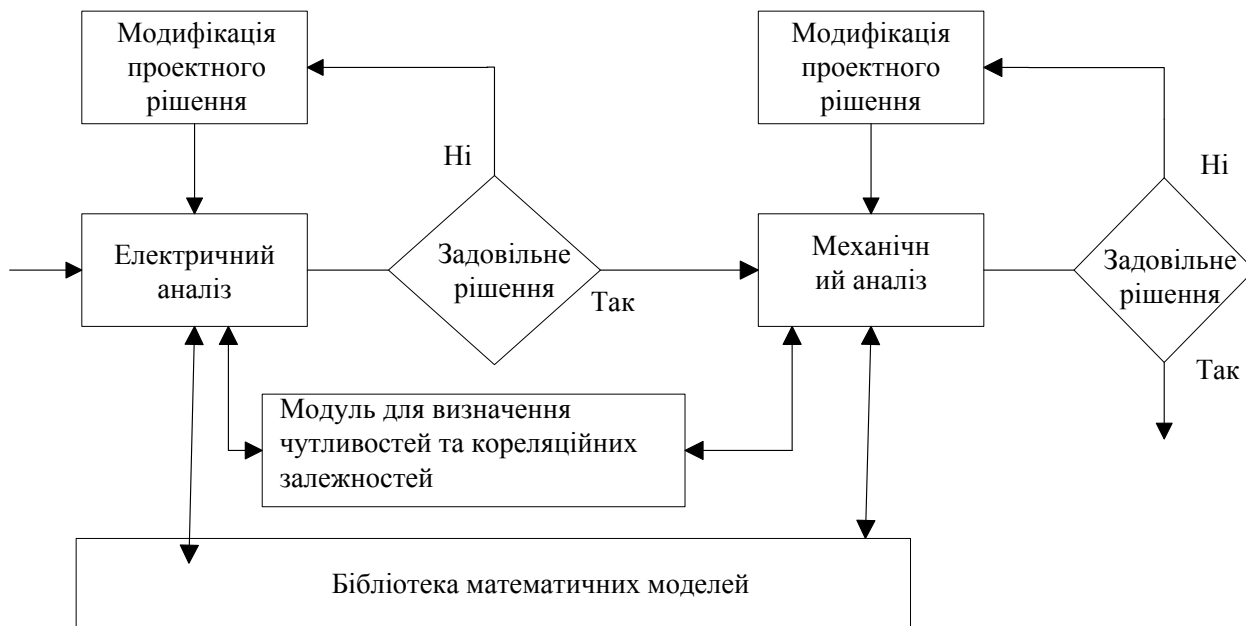


Рис.1.9. Розв'язання задач аналізу та синтезу MEMC на ієрархічних рівнях з використанням теорії чутливості

Складність проектування MEMC закладена в самій суті цих пристроїв і пов'язана з тим, що їх вхідні та вихідні параметри відносяться до різних наукових областей. Для прикладу, електростатичний давач тиску має реєструвати параметри, які відносяться до електрики і механіки. При проектуванні інтегральних пристроїв цього типу, за допомогою математичних моделей, необхідно встановити зв'язок між такими групами параметрів, що є досить складною проблемою.

Процес проектування інтегральних пристроїв MEMC носить ітераційний характер і поява помилок на етапі розробки виробу призводить до виконання операції перепроєктування, що значно збільшує вартість проектних робіт. Оціночну вартість внесення змін в проектне рішення на різних стадіях виробництва мікроелектронних виробів за повідомленням всесвітньої організації IEEE, наведено в табл. 1.1. Звести до мінімуму та по-можливості усунути операцій перепроєктування, а також знизити вартість та зменшити час проектування можна шляхом зменшення кількості ітераційних процедур, які використовуються при параметричній оптимізації. Провести цілеспрямовану

оптимізацію можна використавши результати дослідження кореляційних залежностей та чутливості між вихідними і вхідними параметрами різних груп.

Вартість внесення змін в проектне рішення на різних стадіях виробництва мікроелектронних виробів (IEEE) Таблиця 1.1

Стадія	Вартість (в амер. доларах)
Виробництво	10.000.000
Випробування	1.000.000
Планування	100.000
Тестування	10.000
Проектування	1.000

1.4. Аспекти автоматизованого проектування МЕМС

Особливістю проектування МЕМС, зокрема автономних, є те, що вони функціонують за наперед розробленими алгоритмами і об'єм проектних робіт пов'язаних з розробкою алгоритмічного та програмного забезпечень, зростає з кожним роком відповідно до підвищення складності цих пристроїв, нарощення їх функціональної складової та нарощенням потужності виробництва.

Оскільки розроблення алгоритмів займає значну частку проектувальних робіт, при розробці МЕМС, на відміну від проектування ІС, доцільно ввести четвертий аспект, назовемо його – алгоритмічний, який має бути врахований на кожному з етапів багаторівневого блочно-ієрархічного проектування. Введення даного аспекту дасть можливість підвищити якість та ефективність проектування МЕМС. Основні задачі, що виникають при ієрархічній розробці з врахуванням алгоритмічного аспекту, наведені в табл. 1.2. В ряді випадків, для підвищення ефективності проектувальних робіт, є зміст вводити додаткові підаспекти в функціональний аспект, а саме: механічний, електричний, хімічний, рідинний, оптичний тощо. Це обумовлено гетерогенністю елементів МЕМС та складністю фізичних процесів, які відбуваються в цих пристроях.

Рівні	Задачі
Системний	Розробка структурної схеми програмного забезпечення MEMC. Визначення способу реалізації функції: програмно чи апаратно.
Функціональний	Розробка та уточнення модулів для реалізації функцій програмним чи апаратним шляхом.
Схемотехнічний	Розробка, уточнення та програмна реалізація алгоритмів для функцій MEMC.
Компонентний	Програмування та тестування мікропрограм.

1.5. Контрольні запитання до розділу

1. Що Ви розумієте під терміном MEMC?
2. Які складові включає MEMC?
3. В чому полягає ідея застосування блочно-ієрархічного підходу до проектування MEMC?
4. Які особливості проектування MEMC “зверху вниз”?
5. Які особливості проектування MEMC “знизу вверх”?
6. Які особливості паралельного проектування MEMC?
7. Які задачі необхідно розв’язати в процесі автоматизованого проектування MEMC?
8. Які рівні проектування MEMC Ви знаєте?
9. Які особливості проектування MEMC на ієрархічних рівнях?
10. Яка різниця між синтезом та аналізом в процесі автоматизованого проектування?
11. Які аспекти проектування MEMC Ви знаєте?
12. Які задачі необхідно розв’язати для алгоритмічного аспекту проектування MEMC?
13. Які аспекти проектування MEMC окрім механічного та електричного Ви знаєте?

14. Які технології використовують в процесі виготовлення MEMC?

15. Яке місце технологій інтегральних схем в процесі виготовлення MEMC?

16. Які основні елементи включає схема проектування MEMC на ієрархічному рівні?

1.6. Список використаної літератури до першого розділу

- [1] 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.
- [2] Теслюк В.М. Методи проектування мікроелектромеханічних систем // Всеукраїнський міжведомственный научно-технический сборник “Автоматизированные системы управления и приборы автоматики”, Выпуск 134. - Харків, 2006. - С. 82 – 89.
- [3] Ладугубец В.В. Состояние и перспективы развития автоматизированного схемотехнического проектирования // Научно-технический сборник „Электроника и связь”, Тематичний випуск „Проблеми електроніки”, Ч.1. - Київ, НТУУ “КПИ”. - 2005. – С.121 – 127.
- [4] Петренко А.І. Мережний пакет для комп’ютерного проектування мікроелектромеханічних систем (MEMC) // Труды международной научной конференции “Развитие информационно-коммуникационных технологий и построение информационного общества в Украине” (CeBIT - 2007). – Київ, 2007. – С.143 – 156.
- [5] Теслюк В.М. Моделі та алгоритми для автоматизованого проектування MEMC на функціональному рівні // Комп’ютерні технології друкарства. - № 16. - Львів, 2006. - С. 222 - 230.
- [6] Теслюк В.М. Розробка мови та структури даних для збереження інформації про множину альтернативних рішень при автоматизованому проектуванні MEMC на системному рівні // Збірник наукових праць Інституту проблем моделювання в енергетиці ім.Г.Є.Пухова НАН України. - Вип. 40. - Київ, 2007. – С.37 - 43.
- [7] Теслюк В.М. Застосування мереж Петрі при проектуванні MEMC на системному рівні // Вісник Національного університету «Львівська політехніка»: Комп’ютерні системи проектування. Теорія і практика. - 2006. - № 564. - С. 45 - 53.
- [8] Свірідова Т.В. Математичні моделі мікрорезонаторів електростатичного типу для вбудованих систем. Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 05.13.02 – Математичне моделювання та обчислювальні методи. – Національний університет “Львівська політехніка”, Львів, 2006. – 20 с.
- [9] Лобур М. В., Свірідова Т.В., Теслюк В.М. Аналіз кореляційних залежностей між механічними і електричними параметрами мікробудованих систем // Вісник Національного університету «Львівська політехніка»: Радіотехніка та телекомунікації. - Львів, 2005. - № 534. – С. 147 - 152.
- [10] Lobur M., Sviridova T., Teslyuk V. Desing of micromechanical filter using correlation dependencies between mechanical and electrical parameters // Proc. of The 12-th Intern. Conf. on Mixed Design Of Integrated Circuits And Systems (MIXDES 2005). - Krakov, Poland, 2005. - P. 325 – 327.
- [11] Teslyuk V., Lobur M., Zaharyuk R., Al Omari Tarik Sensitivity theory application in MEMS design // Proc of the IX-th Intern. Conf. on The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM’2007). – Lviv – Polyana, Ukraine, 2007. – P. 536 – 537.
- [12] Mykhaylo Melnyk, Vasyl Teslyuk, Ihor Farmaha, Volodymyr Karkulyovskyy, Svitlana Moroz. The Development of Model for Parameters for MEMS’ Operating Reliability Determination. IIId International Conference of Young Scientists MEMSTECH’2007, May, pp.23-26, Lviv, Polyana, pp.131-132.

- [13] Максвелл Дж. К. Тракта́т об электричестве и магнетизме. В двух томах. Т.1. - М.: Наука., 1989.
- [14] Pereyma M., Denysyuk P., Teslyuk V. Feature Of Optical Embedded System Design // Proc. of the 1st Inter. Conf. of Young Scientists "Perspective Technologies and Methods in MEMS Design" (MEMSTECH 2005).- Lviv-Polyana, Ukraine, 2005. – P. 31-33.
- [15] Antonyuk M., Lobur V., Teslyuk V. Bio-Robot For Research And Analysis Of Human Organism // Proc. of the 1st Inter. Conf. of Young Scientists "Perspective Technologies and Methods in MEMS Design" (MEMSTECH 2005).- Lviv-Polyana, Ukraine, 2005. – P. 18-19.
- [16] Денисюк П., Теслюк В.М., Перейма М. Є. Математична модель базової конструкції гідравлічної мікропомпи з п'єзоелектричним приводом для компонентного рівня проектування // Вісник Національного університету „Львівська політехніка”: Комп'ютерні системи проектування. Теорія і практика. - Львів, 2005. – № 548. – С. 106 - 111.
- [17] Теслюк В.М., Денисюк П.Ю. Математична модель термоактюатора рідинного типу // Вісник Національного університету «Львівська політехніка»: Комп'ютерні системи проектування. Теорія і практика. - Львів, 2004. – № 522. – С. 181 - 185.
- [18] Физическая энциклопедия. / В пяти томах. –М.:, Советская энциклопедия, 1998.
- [19] Теслюк В.М. Побудова множини можливих рішень з допомогою І - АБО – дерева при вирішенні задач структурного синтезу на системному рівні проектування МЕМС // Науково-технічний журнал "Інформаційні технології і системи". – Львів. – 2006. - Том 9, № 1. – С. 101- 108.
- [20] Lab-on-a-Chip and Microarrays for Post-Genome Applications // Cambridge Helthtech Institute's Fourth Annual. – Zurich. – 2002. – 608 p.
- [21] Корляков А.В., Крапивина Е.В., Лучинин В.В. Физико-топологические основы микрофлюидных чипов // Петербургский журнал электроники. – 2001. - № 4. – С.43 – 53.

РОЗДІЛ 2.

ОСНОВИ ЗАДАЧ СТРУКТУРНОГО ТА ПАРАМЕТРИЧНОГО СИНТЕЗУ МЕМС

2.1. Поняття задач структурного та параметричного синтезу

Процес проектування будь-якого об'єкта передбачає, як правило, використання типових задач [1 - 6]:

- задача аналізу;
- задача синтезу.

Проектна задача називається типовою, якщо вона призначена для багаторазового використання в процесі проектування багатьох типів об'єктів. Класичну класифікацію типових проектних задач [1] наведена на рис. 2.1

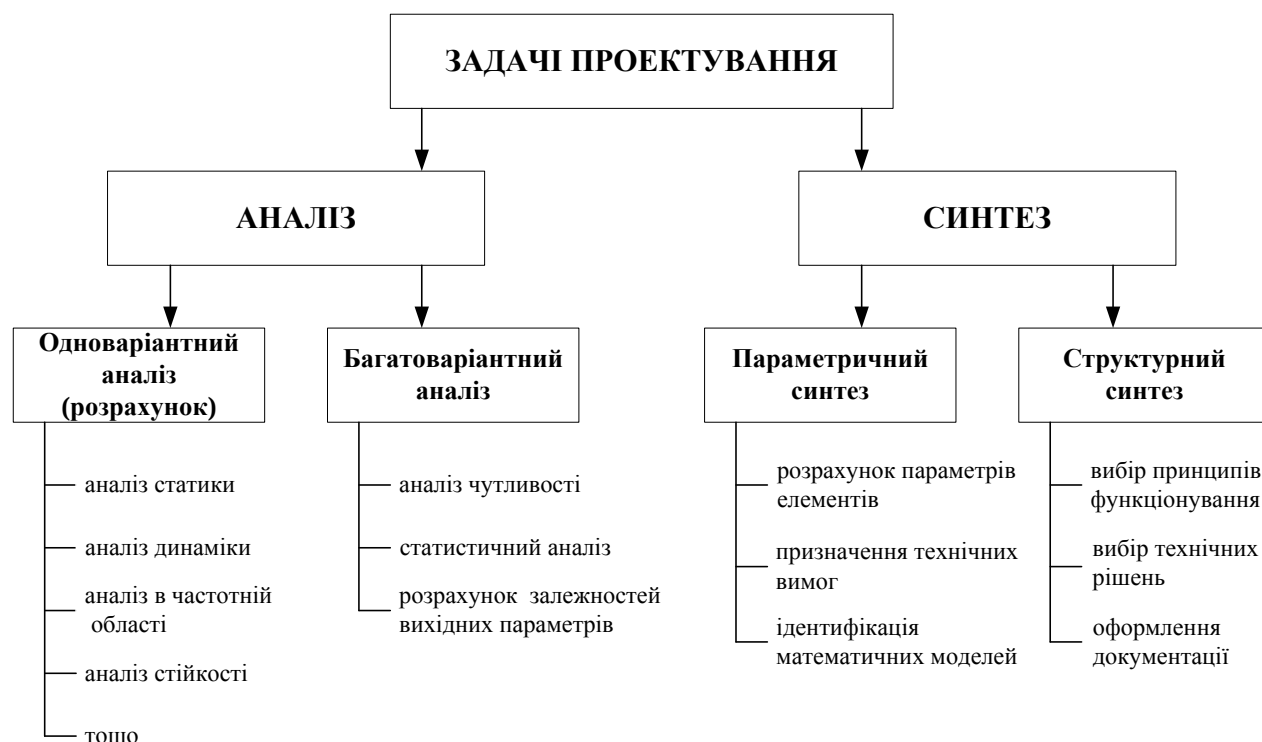


Рис.2.1. Класифікація типових проектних задач [1]

Отже, в загальному випадку, розрізняють проектні задачі аналізу та синтезу. *Синтез* полягає в створенні опису об'єкта, а *аналіз* — в визначенні

властивостей та дослідженні працездатності об'єкта за його описом, тобто при синтезі створюються, а при аналізі оцінюються проекти об'єктів.

Схематично різницю між задачами аналізу та синтезу, можна зобразити таким чином, як показано на рис.2.2.

Отже, при синтезі відомі вихідні параметри об'єкта проектування і необхідно визначити вхідні, а при аналізі – обернена ситуація: відомо вхідні параметри і необхідно визначити вихідні.

В свою чергу, задачі аналізу поділяються на задачі одно- чи багатоваріантного аналізу.

При *одноваріантному аналізі (задача розрахунку)* відомі значення внутрішніх та зовнішніх параметрів, необхідно визначити значення вихідних параметрів об'єкта. Корисно використовувати геометричну інтерпретацію цієї задачі, пов'язану з поняттям області визначення внутрішніх параметрів. Цей n -вимірний простір, в якому для кожного з n внутрішніх параметрів виділена координатна вісь X_n . При одноваріантному аналізі задається також деяка точка в просторі внутрішніх параметрів і необхідно в цій точці визначити значення вихідних параметрів. Така задача, як правило, зводиться до одноразового розв'язку рівнянь, які складають математичну модель, що й обумовлює назву цього виду аналізу.

Для прикладу, це може бути одноразове визначення значення сили між пластинами електростатичного актюатора (рис.2.3.) чи давача за формулою

$$F_{el} = C \frac{U^2}{2(d-x)}, \quad C = \varepsilon_0 \varepsilon \frac{S}{d-x},$$

де C – електрична ємність; S – площа пластин; U – прикладена напруга; d – початкова відстань між пластинами конденсатора; ε , ε_0 – діелектрична проникність середовища між пластинами конденсатора та діелектрична проникність у вакуумі.

Багатоваріантний аналіз полягає в дослідженні властивостей об'єкта в деякому просторі області внутрішніх параметрів. Такий аналіз потребує багаторазового розв'язання системи рівнянь (багаторазового виконання одноваріантного аналізу).



а)



б)

Рис. 2.2. Схематичне представлення задач синтезу а) та аналізу б)

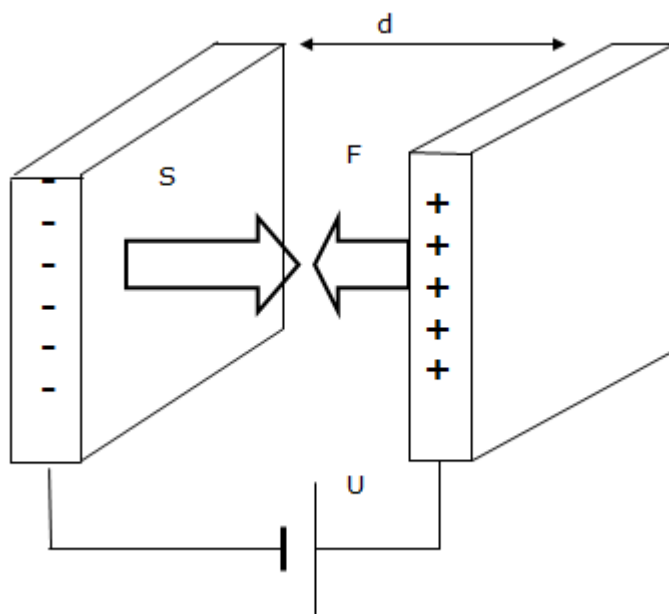


Рис. 2.3. Спрощена конструкція електростатичного актюатора

Прикладом задач багатоваріантного аналізу може бути задача одноваріантного аналізу, яка виконується багато разів. В цьому випадку ми

отримаємо залежність вихідного параметра (сили) електростатичного актюатора від параметрів конструкції. У кінцевому випадку ми отримаємо графік даної залежності, який побудований шляхом багаторазового розв'язання задач одноваріантного аналізу.

Задачі синтезу поділяються на задачі структурного та параметричного синтезу. Ціллю задачі *структурного синтезу* є визначення *структури об'єкта* — списку типів елементів, з яких складається об'єкт, і зв'язків між елементами в складі об'єкта.

Параметричний синтез полягає в визначенні числових значень параметрів елементів при заданій структурі та умовах працездатності на вихідні параметри об'єкта, тобто при параметричному синтезі необхідно знайти точку чи область в просторі внутрішніх параметрів, де виконуються ті чи інші умови (як правило, умови працездатності).

Для прикладу, в процесі проектування інтелектуального МЕМС давача тиску ємнісного типу, розроблення його структури є задачею структурного синтезу. Побудована структура буде включати два основні структурні елементи (рис.2.4): безпосередньо давач та елемент (схема), який буде реалізовувати інтелектуальні функції:

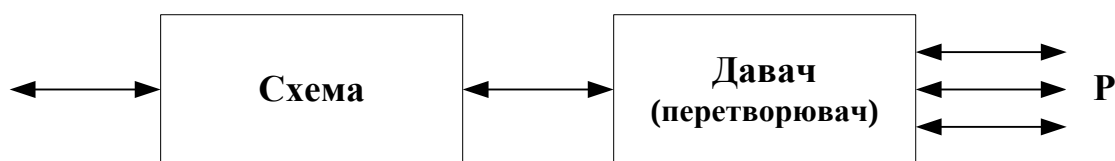


Рис. 2.4. Структура інтелектуального давача тиску

Після синтезу структури, необхідно визначити параметри цих двох елементів. В цьому випадку необхідно розв'язати вже задачу параметричного синтезу. Тобто, мається на увазі, визначення конструктивних параметрів давача (розміри, матеріали, тощо), параметри схеми (реалізація на основі штучних нейронних мереж, швидкодію, енергію споживання, функції, які мають реалізувати елемент “схема” та інші).

Загалом, в процесі автоматизованого проектування складних об'єктів та систем, використовують дві типові задачі (аналіз та синтез). Разом з тим, ведеться мова про задачі оптимізації, які включає як задача структурного та параметричного синтезу і задачу верифікації. Схема поєднання задач аналізу, оптимізації та синтезу, зображена на рис.1.2. та рис.2.5.

Задача верифікації проектного рішення полягає у перевірці відповідності отриманого проектного рішення вимогам технічного завдання.

Неважко підрахувати, що синтез проектного рішення на черговому етапі проектування може вимагати виконання надмірно великої кількості варіантів аналізу і відповідно надзвичайно великих затрат машинного часу та людських ресурсів. Зокрема на рис.2.5 наведено схему взаємодії задач синтезу, оптимізації та аналізу.

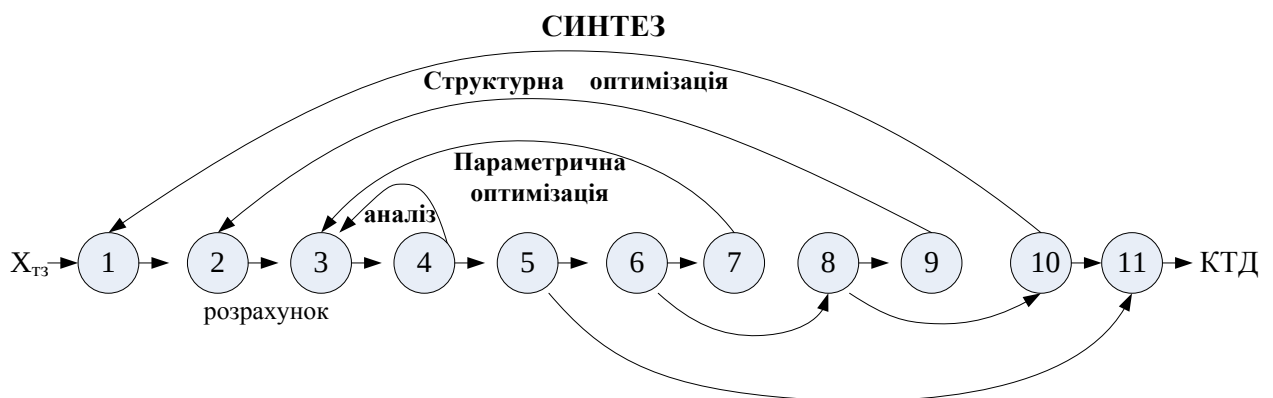


Рис.2.5. Схема взаємодії задач синтезу, оптимізації та аналізу

Для наведеної схеми визначимо кількість разів розв'язання задачі аналізу.

Отже, загальна кількість ітерацій буде рівною:

$$K_{\text{ітер.}} = K_{\text{аналізу}} * K_{\text{парам.оптм.}} * K_{\text{структ.оптм.}} * K_{\text{синтезу}}$$

Приймемо, що:

$$K_{\text{аналізу}} = 40, \quad K_{\text{парам.оптм.}} = 500,$$

$$K_{\text{структ.оптм.}} = 20, \quad K_{\text{синтезу}} = 1.$$

Тоді, загальна кількість ітерацій (кількість розв'язання задачі аналізу) буде рівною:

$$K_{\text{итер.}} = 40 * 500 * 20 * 1 = 400000 \text{ ітерацій.}$$

Відповідно, отримали величезну кількість ітерацій, що, без сумніву, призведе до громіздких обчислень і величезних затрат ресурсу комп'ютера.

Один зі шляхів рішення цієї проблеми - застосування достатньо точних і складних математичних моделей і алгоритмів аналізу тільки на завершаючих ітераціях синтезу.

Для більшості розглядуваних варіантів структури, при цьому, виконується лише попередня оцінка на основі непрямих критеріїв, **спрощених моделей та алгоритмів**. Така оцінка дасть змогу без суттєвих витрат обчислювальних ресурсів відсіяти більшість безперспективних варіантів і залишити для ретельного аналізу мале число варіантів.

2.2. Класифікація задач синтезу

Для кращого розуміння особливостей задач синтезу, які властиві процесу автоматизованого проектування, наведемо приклади їх класифікації згідно з різними критеріями.

2.2.1. Класифікація задач параметричного синтезу

Отже, усі задачі параметричного синтезу можна розділити на три основні групи (рис.2.6.) [1 - 6]:

- задачі призначення технічних умов;
- задачі обчислення параметрів елементів;
- задачі ідентифікації математичних моделей.

В свою чергу, до задач параметричного синтезу призначення технічних умов, належать підзадачі: задачі призначення технічних умов при зовнішньому проектуванні та задачі призначення технічних умов при проектуванні “зверху-вниз”.

Група 1 задач параметричного синтезу пов'язана з призначенням технічних вимог до вихідних параметрів об'єкта проектування.

На верхньому рівні проектування чи на кожному ієрархічному рівні задача призначення технічних вимог не може бути повністю формалізована. Як правило, початкове ТЗ відображає лише загальні вимоги до об'єкта, що потребує подальшої конкретизації: та узгодження, тобто визначення вектора технічних вимог T_j , а саме: забезпечення умов працездатності $y_i \leq T_j$.

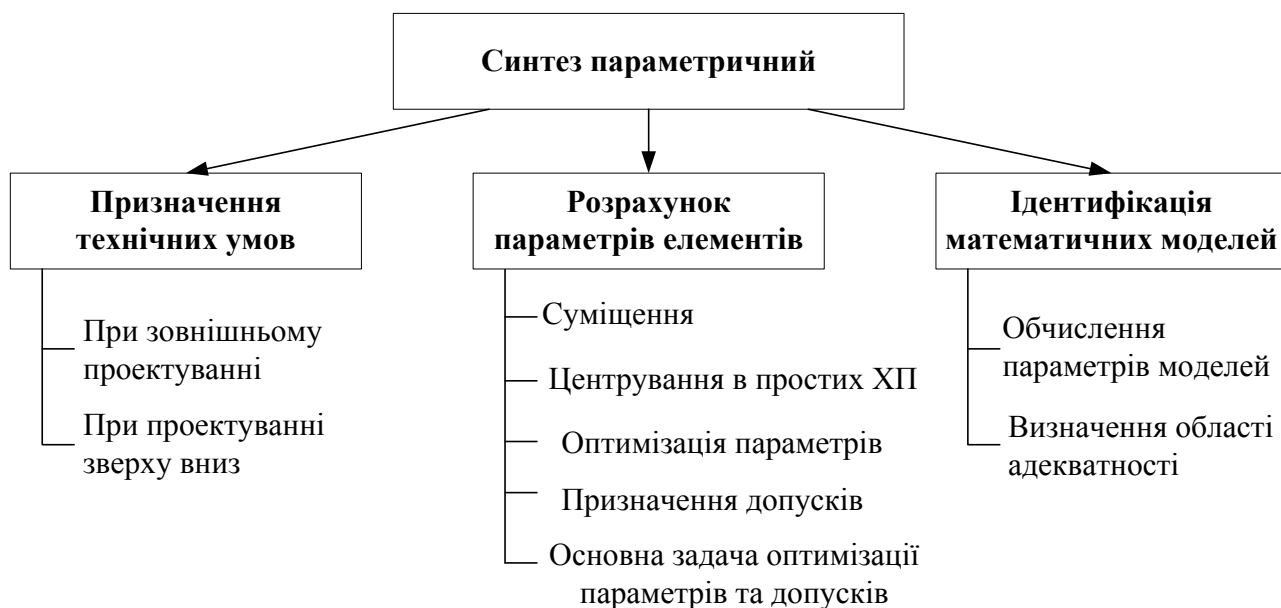


Рис. 2.6. Класифікація задач параметричного синтезу

Задача визначення вектора технічних вимог T_j — основна задача параметричного синтезу, яку розв'язують при зовнішньому проектуванні.

На всіх ієрархічних рівнях проектування зверху –вниз, окрім найвищого, задача призначення T може бути формалізована та представлена як задача оптимального перетворення вектора T до вихідних параметрів об'єкта на k -му рівні в T до вихідних параметрів складових об'єкта на $(k+1)$ -му рівні (рис.2.7).

Група 2 задач параметричного синтезу пов'язана з визначенням параметрів елементів об'єкта при заданій структурі об'єкта проектування.

Параметри проєктованих об'єктів, як правило, являються випадковими величинами, які не піддаються строгому врахуванню виробничих похибок виготовлення та випадкового характеру параметрів вхідних матеріалів. Тому, в

найбільш загальній постановці визначення параметрів розуміється обчислення як вектора номінальних значень параметрів $X_{ном}$, так і вектора їх допусків G . Як правило відомості про характер розподілу вектора X при проектуванні досить приблизні. При цьому, під номінальним значенням $X_{номі}$ параметра X_i найчастіше всього є його математичне очікування, а під допуском g_i – половина інтервалу з значенням $X_{номі}$ в центрі та ймовірністю попадання в нього значень X_i , рівною заданій величині P (рис.2.8).

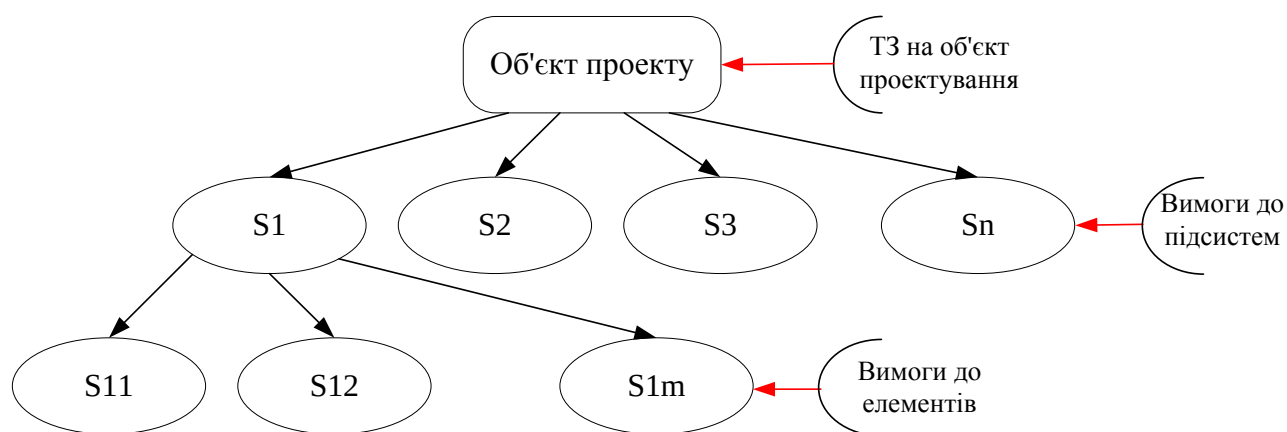


Рис. 2.7. Схема формування вимог до підсистем та елементів підсистем

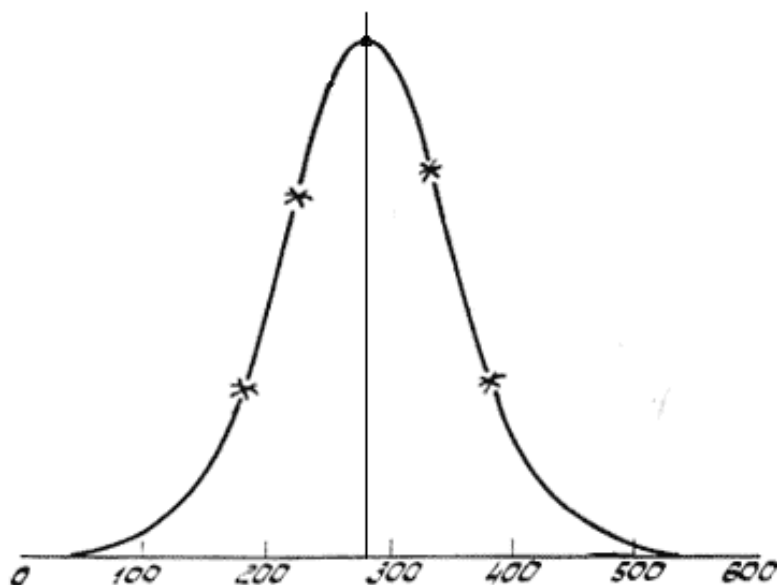


Рис. 2.8. Приклад розподілу вхідного чи вихідного параметра

В САПР зустрічаються наступні різновиди задач визначення параметрів елементів []:

Задача суміщення, яка розв'язується при відомій формі (часто і при відомих розмірах) області **XG** і зводяться до такого суміщення областей **XP** і **XG**, при якому ймовірність виконання заданих умов роботоспроможності максимальна.

Задача центрування, яка найчастіше зустрічається і є частковим випадком задачі суміщення, коли відомості про вірогідну кореляцію і асиметрію розподілу параметрів відсутні. Задача зводиться до знаходження центра **X*** області **XP** в нормованому просторі параметрів, цей центр і приймається в якості шуканої точки **Xном**.

Задача оптимізації параметрів без врахування відомостей про їх розподіл, яка зводиться до задачі математичного програмування. Для нормування вихідних параметрів корисно мати відомості про допускову область, але не в просторі **XП** керованих параметрів, а в просторі **УП** вихідних параметрів.

Задача призначення допусків, яка розв'язується при заданому векторі **Xном** і припущення стосовно економічно оправданих співвідношень між допусками гі окремих параметрів. Ця задача зводиться до вибору прийнятного відсотка виходу готових виробів при їх виробництві і вписування гіперпаралелепіпеда допусків у **XP**.

Основна задача оптимізації параметрів і допусків, полягає в визначенні векторів **Xном** і **G** при заданих співвідношеннях, між допусками гі окремих параметрів

Група задач 3 параметричного синтезу пов'язана з визначенням параметрів, які використовуються в САПР математичних моделей та визначенням областей їх адекватності.

Необхідно зауважити, що точність моделі залежить від умов функціонування об'єкту-оригіналу в просторі зміни вхідних параметрів. В цьому випадку, зручніше оперувати поняттям області адекватності моделі [6], що визначає область в просторі зміни вихідних параметрів, де виконується умова $\varepsilon_{\text{model}} < \varepsilon_{\text{гранич}}$ ($\varepsilon_{\text{гранич}}$ - наперед задане граничне значення похибки ММ)

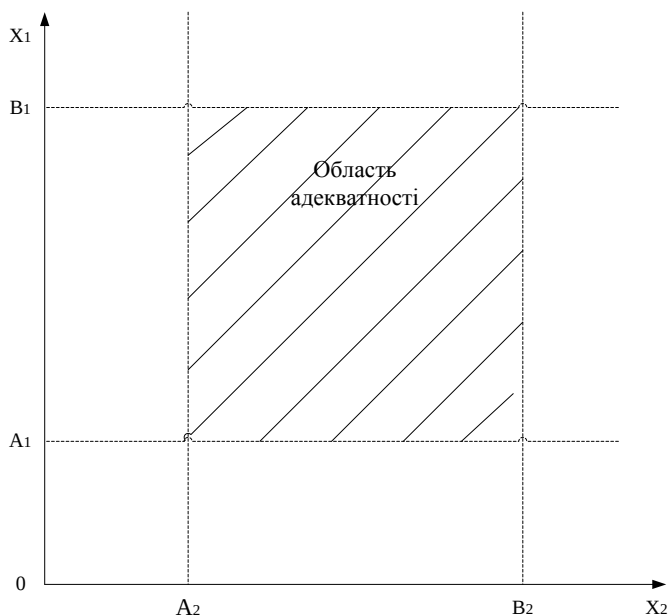


Рис. 2.9. Приклад області адекватності моделі

Отже, найважливішою вимогою до математичної моделі є вимога її **адекватності** (відповідності) об'єкту-оригіналу відносно вибраної системи його характеристик. Під цим, як правило розуміють: 1) правильний якісний опис об'єкта за вибраними характеристиками; 2) правильний кількісний опис об'єкта за вибраними характеристиками з необхідною точністю в розумних межах. Отже, під **адекватністю ММ** будемо розуміти можливість відображення заданих властивостей моделі з заданою точністю. **Точність** визначається як ступінь співпадіння вихідних параметрів моделі та об'єкту-оригіналу.

Оберненою величиною до точності моделі є її похибка. Для прикладу, відносна похибка моделі i -го вихідного вихідного параметра визначається з виразу

$$\varepsilon_i = \left| \frac{(y_{\text{model},i} - y_{\text{original},i})}{y_{\text{original},i}} \right| * 100\%,$$

де ε_i – похибка моделі i -го вихідного вихідного параметра, $y_{\text{model},i}$ – значення i -го вихідного вихідного параметра порахованого з використанням побудованої моделі, а $y_{\text{original},i}$ – значення i -го вихідного вихідного параметра порахованого з використанням об'єкту-оригіналу.

При визначенні похибки математичної моделі ε_{model} використовується

одна з норм вектора $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)$, а саме: $\varepsilon_{model} = \max |\varepsilon_i|$, де $i = \overline{1, n}$, або

$$\varepsilon_{model} = \sqrt{\varepsilon_1^2 + \varepsilon_2^2 + \dots + \varepsilon_n^2}.$$

Розглянемо, для прикладу, побудовану математичну модель, яка дає змогу визначити чотири вихідних параметрів деякого об'єкта проектування. Значення порохованих вихідних параметрів з використанням розробленої ММ наступні (в безрозмірному вигляді): 10.5, 23.4, 100.1 та 56.7, а значення отримані з допомогою експерименту: 9.3, 25, 112.4 і 49.2.

Визначимо величини відносних похибок для кожного вихідного параметра і загальну похибку побудованої ММ:

$$\varepsilon_1 = \left| \frac{(10.5 - 9.3)}{9.3} \right| * 100\% = 12.9\%, \quad \varepsilon_2 = \left| \frac{(23.4 - 25)}{25} \right| * 100\% = 6.4\%,$$

$$\varepsilon_3 = \left| \frac{(100.1 - 112.4)}{112.4} \right| * 100\% = 10.9\%, \quad \varepsilon_4 = \left| \frac{(56.7 - 49.2)}{49.2} \right| * 100\% = 15.2\%,$$

$$\varepsilon_{model} = \sqrt{12.9^2 + 6.4^2 + 10.9^2 + 15.2^2} = \sqrt{166.41 + 40.96 + 118.81 + 231.04} = \sqrt{557.22} = 23.61\%.$$

Наведений приклад показує, що похибка моделі не може бути меншою за найбільше значення похибки одного з вихідних параметрів математичної моделі.

Слід підкреслити відносний характер наведеного поняття адекватності, тобто прив'язку адекватності моделі до характеристик об'єкту дослідження, які прийнято за основні в рамках цього дослідження. Так, наприклад, якщо вивчається реакція об'єкта на зовнішні збурення того чи іншого класу, то модель, адекватна відносно одного класу збурень, може виявитися неадекватною відносно іншого класу збурень. Іншим характерним прикладом може бути класичне рівняння теплопровідності, отримане на основі закону Фур'є. Це рівняння добре описує еволюцію температури, тобто є адекватним у кількісному відношенні. Крім того, з нього випливає ряд наслідків якісного характеру, які також правильно описують реальний процес поширення тепла: збереження кількості тепла, неможливість концентрації та осциляції температури і т.д. Але з іншої сторони, з цього рівняння випливає також фізично абсурдний висновок про нескінченність швидкості поширення тепла.

Таким чином, якщо суттєвою характеристикою процесу вважати швидкість поширення тепла (хоча потреба в цьому виникає досить рідко), то класичне рівняння теплопровідності, як модель реального процесу поширення тепла, стає неадекватним не тільки у кількісному, а й у якісному відношеннях. Для того, щоб отримати модель адекватну і за швидкістю поширення тепла, необхідно уточнити закон Фур'є шляхом врахування інерційності молекул.

Упущення того факту, що адекватність математичної моделі завжди відносна і має свої рамки застосування може привести (і, на жаль, не раз приводило) до спроб нав'язати реальному об'єкту властивості його моделі, наприклад, серйозному сприйнятті твердження, що швидкість поширення тепла є дійсно нескінченною. На жаль, у більш складних випадках, неадекватність моделі виявити набагато складніше, тому застосування неадекватної моделі може привести до того, що можна або не виявити, або занадто спотворити те що насправді є, натомість вивчати те, що нам не потрібно і навіть те, чого взагалі не існує. Як правило у таких випадках перевірка адекватності здійснюється на раціональному рівні, іншими словами “на хлопський розум”, що, безумовно, вступає у конфлікт з чисто дедуктивним (логічним) рівнем і є предметом гострої критики так званих “чистих” математиків. Перевірка адекватності моделі може слугувати також раціональним обґрунтуванням законності застосування прийнятих гіпотез та припущень.

2.2.2. Класифікація задач структурного синтезу

Наведемо риклади класифікації задач структурного синтезу в залежності від різних критеріїв класифікації [1 - 6].

В залежності від стадії проектування розрізняють наступні задачі, а саме:

1. Вибір основних принципів функціонування об'єкта проектування (*тут йде мова про принципи інформаційних, організаційних, фізичних і т.д.*) найчастіше в цій групі процедур виокремлюють групу процедур вибору фізичних принципів дії.

2. Вибір технічного рішення в рамках заданих принципів функціонування.

3. Оформлення технічної документації.

*В залежності від можливостей формалізації задачі синтезу поділяються на **п'ять** рівнів складності.*

До **1-го рівня складності** належать задачі, в яких потрібно виконати тільки параметричний аналіз, а структура об'єкта визначена або специфікою ТЗ, або результатами процедур, виконаних на попередніх етапах проектування.

2-й рівень складності. Сюди належать задачі, **в яких можливий повний перебір відомих розв'язків**. Отже, це комбінаторні задачі, тобто задачі вибору елементів в скінчених множинах, причому в множинах малої потужності. В цих задачах або елементи множини структур представляють собою наперед складені і включені в базу даних описи структур, або є алгоритм, який дає можливість по чергово отримувати і аналізувати всі елементи множини за допустимий час.

До **3-го рівня складності** відносять комбінаторні задачі, які при існуючих технічних і програмних засобів не можуть бути розв'язані шляхом повного перебору за прийнятний час. Існує велика кількість практично важливих задач синтезу, які належать до 3-го рівня. Приклади таких задач – задачі компоновання і розташування заданого обладнання в обмеженому просторі, проведення трас, більшість процедур оформлення технічної документації. *До 3-го рівня зводяться багато задач синтезу більш високих рівнів при прийнятті відповідних обмежень і допусків.*

4-й рівень складності. Сюди відносять задачі пошуку **варіантів структур в рахункових множинах невідомої чи необмеженої потужності**. Формалізація саме таких задач має найбільшу складність.

До **5-го рівня складності** належать задачі синтезу, розв'язок яких є проблематичним. Якщо в задачах 4-рівня можливості створення структур беззаперечно і головна проблема полягає в знаходженні серед багатьох згенерованих структур деякої, яка б задовольняла певним кількісним вимогам,

то в задачах 5-го рівня – одержання рішення еквівалентне пропонуванню принципово нових основ побудови цілого класу технічних об'єктів.

В залежності від типу синтезованих структур.

Одновимірний синтез. Розв'язуються задачі впорядкування елементів структури в одновимірних просторах (наприклад, задачі складання розкладів, синтезу процесів, які представляються у вигляді впорядкованої послідовності елементів.

При схемному синтезі визначається структура об'єкта без конкретизації його геометричних форм. Характерні приклади – синтез кінематичних, електричних, функціональних схем і т.п.

Геометричний синтез. Полягає в конкретизації геометричних властивостей проєктованих об'єктів і включає в себе охарактеризовані вище задачі оформлення конструкторської документації, а також задачі позиціонування і синтезу поверхонь і траєкторій.

До **задач позиціонування** належать задачі взаємного розміщення в просторі деталей заданої геометричної форми, наприклад задачі вибору баз для механічної опрацювання деталей складної форми, синтез композицій із заданих деталей.

До **синтезу поверхонь та траєкторій** відносять задачі проєктування поверхонь, які обтікають потоки газу чи рідини чи направляючих такий потік (крило літака, корпус автомобіля, лопатка турбіни), синтеза траєкторій рухомих робочих органів технологічних автоматів, синтез профілів несучих конструкцій та ін.

2.3. Методи розв'язання задач параметричного синтезу

В загальному випадку задача параметричного синтезу формулюється як задача оптимізації, де необхідно визначити оптимальні параметри розроблювального виробу при заданій структурі.

Наведемо приклад формулювання задачі параметричного синтезу шляхом побудови оптимізаційної задачі.

Одним з основних параметрів будь-якого мікродавача є його чутливість

[7 - 9], яка визначається як відношення приросту вихідного параметра до приросту вхідного, тобто, у випадку ємнісного мікродавача тиску отримаємо:

$$S_P^C = \frac{\Delta C}{\Delta P}, \Delta C = C_1 - C_0, \quad (2.1)$$

де ΔC – зміна ємності; ΔP – зміна прикладеного тиску.

Переважно пластина кругла за формою, отже, можна скористатися відомими виразами для оцінки зміни ємності мікродавача. Так, для визначення ємності мікродавача без прикладеного навантаження скористаємося формулою

$$C_0 = \varepsilon_0 \varepsilon \frac{A}{d}, \quad (2.2)$$

де ε – діелектрична проникність речовини між пластинами; ε_0 – діелектрична проникність вакууму; A – площа мембрани, R – радіус пластини); d – відстань між пластинами.

У випадку прикладання тиску до мембрани – вона зміститься на відстань Δd , тоді формулу для визначення ємності (2.2) необхідно переписати у вигляді:

$$C_1 = \varepsilon_0 \varepsilon \frac{A}{d - \Delta d}.$$

Відповідні формули справедливі у тому випадку, якщо товщина пластини круглої форми набагато менша за її діаметр (довжину чи ширину – для пластин іншої форми).

Для визначення зміщення x скористаємося теорією тонких пластин і вважатимемо, що максимальні зміщення пластини, яка жорстко защемлена по її краях, вдвічі менші від її товщини t . Ввівши такі обмеження, можемо скористатися таким аналітичним виразом для визначення зміщення:

$$x(r) = \frac{3}{16} p \frac{1 - \nu^2}{E} \frac{(R - r)^2 (R + r)^2}{t^3} \text{ при } x(r) \leq \frac{t}{2}, \quad (2.3)$$

де ν – коефіцієнт Пуассона; E – модуль Юнга; r – поточне значення радіуса; R – радіус тонкої пластини.

Пам'ятаючи про припущення, що діаметр круглої пластини набагато більший за її товщину, можна вважати, що x дорівнює максимальному зміщенню, визначеному за виразом (2.3). Тоді

$$x = \frac{3}{16} p \frac{(1 - \nu^2) R^4}{E t^3}. \quad (2.4)$$

У випадку квадратної тонкої пластини зміщення визначаємо за виразом

$$x(z, y) = p \frac{12a^2}{Et^2} \left(1 - \nu^2\right) \left(1 - \frac{z^2}{a^2}\right)^2 \left(1 - \frac{y^2}{a^2}\right) * \\ * \left[0,0202 + 0,0054 \left(\frac{z^2 + y^2}{a^2}\right) + 0,0063 \left(\frac{z^2 y^2}{a^4}\right)\right], \quad (2.5)$$

де a – сторона квадратної пластини.

Разом з тим, існують обмеження конструктивного та технологічного характеру. Зокрема, будь-який пристрій займає певну площу (в інтегральних технологіях завжди важливим було питання мінімізації площі), і в процесі оптимізації необхідно враховувати цю особливість. Тобто площа пристрою, який розробляють, не має перевищувати $A \leq A_{\max}$.

З виразів (2.3 – 2.5) та (2.1) випливає, що для підвищення чутливості мікродавача необхідно зменшувати товщину пластини. З одного боку, існують технологічні обмеження, тобто товщина пластини не може бути меншою за певне значення $t \geq t_{\min}$ та необхідно, щоб виконувалась умова, що товщина має бути набагато меншою за найменший лінійний розмір пластини (діаметр, ширина, довжина) $t \ll L$ (для справедливості вищенаведених виразів щодо визначення зміщень і напружень). З іншого боку, зміщення призводять до генерації напружень, які не мають перевищувати максимально допустимі для матеріалу пластини напруження, з відповідним коефіцієнтом запасу $\sigma_x \leq G_{\max}$ чи $\sigma_y \leq G_{\max}$.

Можливість появи напруги пробою накладає обмеження на відстань між пластинами ємнісного елемента $0 \leq U \leq 3 \cdot 10^6 (d - \Delta d)$.

Отже, необхідно знайти максимум виразу (2.1) (чутливості інтегрального мікродавача):

$$\max S(t, L, \Delta d, U, \sigma_x, \sigma_y), \quad (2.6)$$

із задоволенням таких умов:

$$A \leq A_{\max}, t \geq t_{\min}, t \ll L, \sigma_x \leq G_{\max},$$

$$\sigma_y \leq G_{\max}, 0 \leq U \leq 3 \cdot 10^6 (d - \Delta d).$$

В процесі розв'язання цієї задачі використовується один з методів багатокритеріальної оптимізації.

2.4. Підходи до розв'язання задач структурного синтезу

Існує три основні підходи до розв'язання задач структурного синтезу [1 - 6], а саме:

1. Перебір.
2. Послідовний синтез.
3. Трансформація описів різних аспектів.

Класифікація даних підходів зображена на рис. 2.10.



Рис. 2.10. Класифікація підходів до розв'язання задач структурного синтезу

Отже, маємо три групи підходів до розв'язання задач структурного синтезу. Опишемо коротко кожен з вищеперечислених.

Алгоритми перебору характеризуються можливостями оцінки тільки варіантів готових кінцевих структур.

Такі структури або створюються заздалегідь, або зберігаються в базі даних, або генеруються за тими чи іншими правилами із заданого набору елементів. Повний перебір можливий тільки лише в найпростіших випадках.

Як правило, перебір має включати в себе такі складові:

- 1) вибір чи генерація чергового варіанту;
- 2) оцінка варіанту;
- 3) прийняття рішення про відповідність чи невідповідність технічному завданню.

Алгоритми трансформації описів різних аспектів широко використовуються в діючих САПР.

Так на основі відповідності між функціональними і структурними описами ґрунтуються програми так званих кремнієвих компіляторів в САПР великих інтегральних схем. В кремнієвому компіляторі кожній функціональній комірці ставиться у відповідність певна конструктивна комірка. Трансформація описів лежить також в основі формального синтезу функціональних схем ЕОМ за заданим алгоритмом функціонування. Іншим прикладом є функціонування систем виготовлення конструкторської документації, в яких формалізовано перетворення результатів конструкторського проектування в графічній формі, виконаного за правилами проекційного креслення.

Однією з таких систем, де реалізовано зазначений підхід – є система Cadence (одне з основних меню якого зображено на рис.2.11).

Послідовні алгоритми характеризуються поетапним розв'язанням задачі синтезу з можливостями оцінки одержаних проміжних структур. Розрізняють два способи одержання закінченої структури:

- 1) нарощування;
- 2) виокремлення.

При нарощуванні відбувається почергове додавання елементів до деякої початкової структури (в частковому випадку за початкову структуру може бути прийнятий деякий елемент). Прикладами алгоритмів нарощування можуть бути послідовні алгоритми компонування і розміщення обладнання. Наприклад, при

розподілі обладнання по конструктивних блоках основною оцінкою варіантів може служити кількість міжблочних зв'язків (рис.2.12 та рис.2.13). Тоді перевагу слід віддавати тим проміжним варіантам, при яких число зв'язків виявляється сконцентрованими в межах одного блоку.

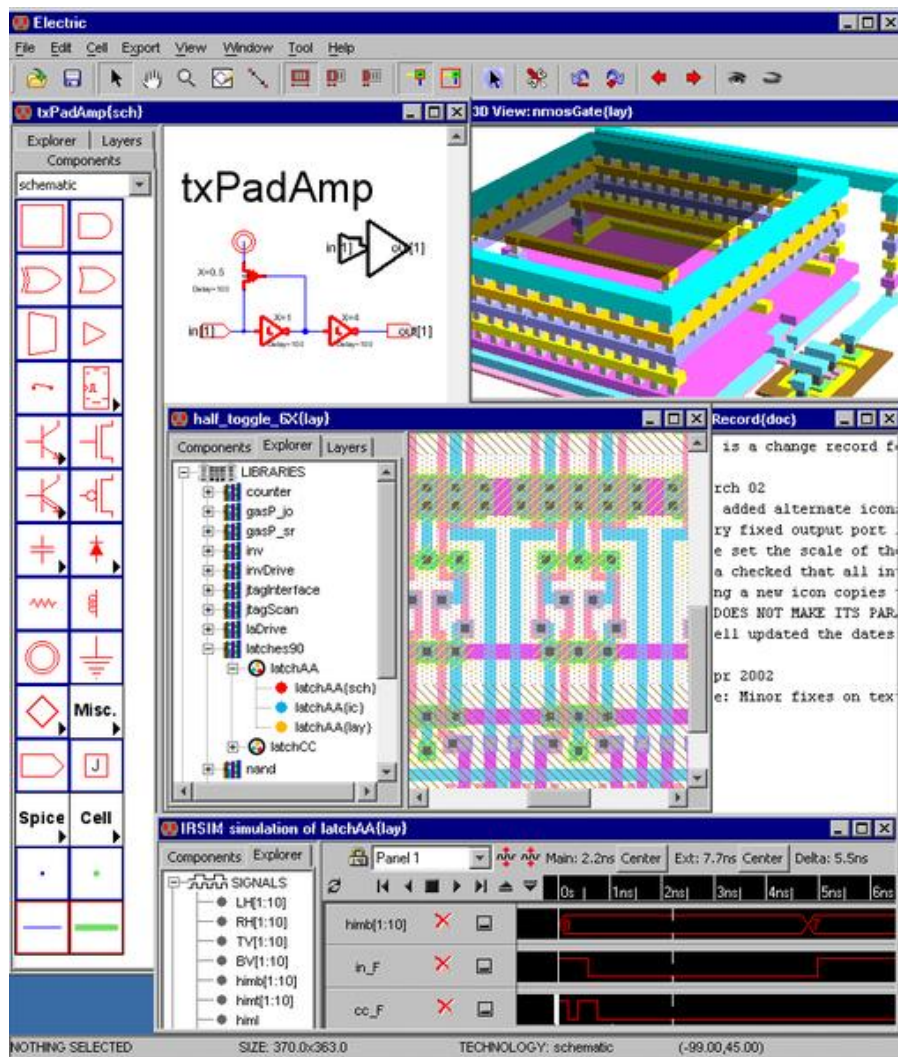


Рис. 2.11. Приклад одного з основних меню системи

При виокремленні із деякої надлишкової структури поступово відкидаються зайві елементи (рис. 2.14 та рис.2.15). Алгоритми виокремлення можуть використовуватися, якщо попередньо складена узагальнена структура для класу об'єктів, які розглядаються. Прикладами таких узагальнених структур можуть бути узагальнені технологічні маршрути опрацювання деталей деяких класів на металообробних станках. В узагальнений маршрут включають

операції, які можуть зустрічатися при різних поєднаннях конструктивних особливостей в деталях даного класу. Подальше зіставлення креслення конкретної деталі і узагальненого маршруту дає можливість відкинути зайві операції і сформувати конкретний технологічний маршрут.

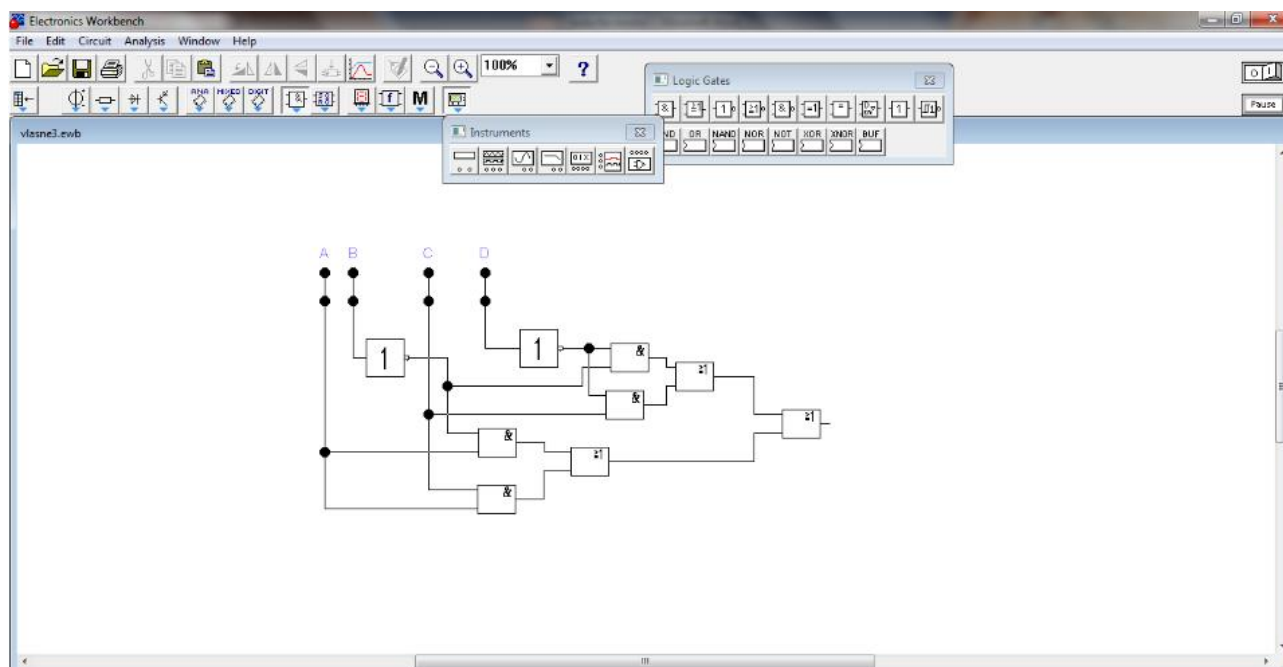


Рис. 2.12. Приклад меню системи, яка реалізує операцію “Нарощування”

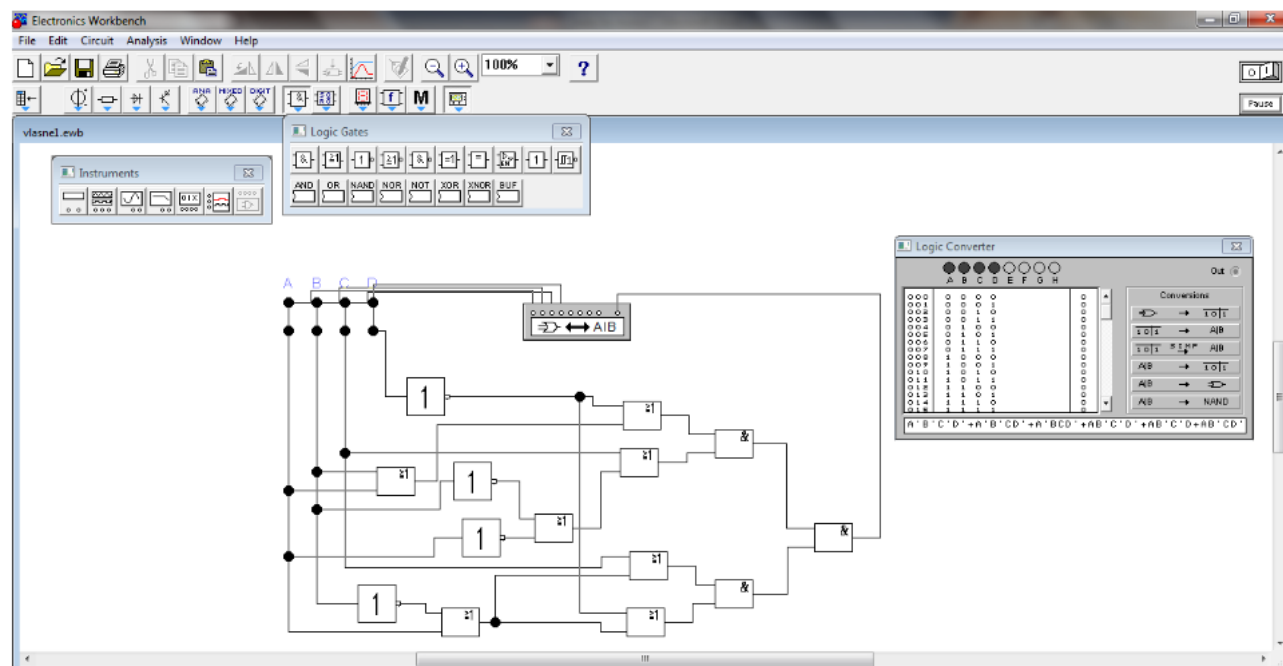


Рис. 2.13. Приклад меню реалізації операції “Нарощування”

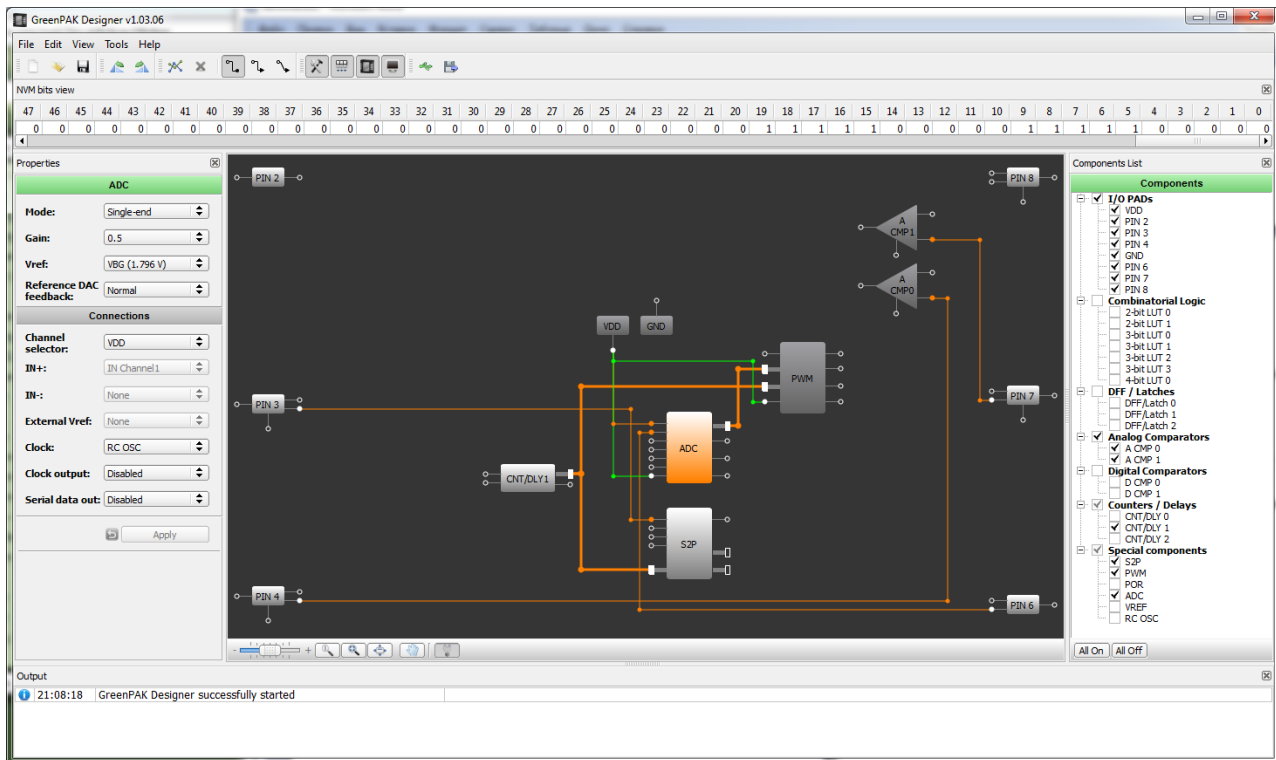


Рис. 2.14. Приклад меню системи, яка реалізує операцію виокремлення
GreenPAK Designer

2.5. Контрольні запитання до розділу

1. Які типові задачі проектування Ви знаєте?
2. Яка задача називається задачею синтезу?
3. Яка різниця між задачами параметричного і структурного синтезу?
4. Яка задача називається задачею аналізу?
5. Яка різниця між задачами одноваріантного і багатоваріантного аналізу?
6. Яка проектна задача називається типовою?
7. Наведіть приклади задач аналізу та синтезу, які використовують в процесі проектування MEMС.
8. Наведіть приклади типових задач аналізу та синтезу, які використовують в процесі проектування.
9. Які алгоритми структурного синтезу Ви знаєте?
10. Яка ідея алгоритмів перебору в процесі структурного синтезу?
11. Яка ідея алгоритмів трансформації описів в процесі розв'язання задач структурного синтезу?

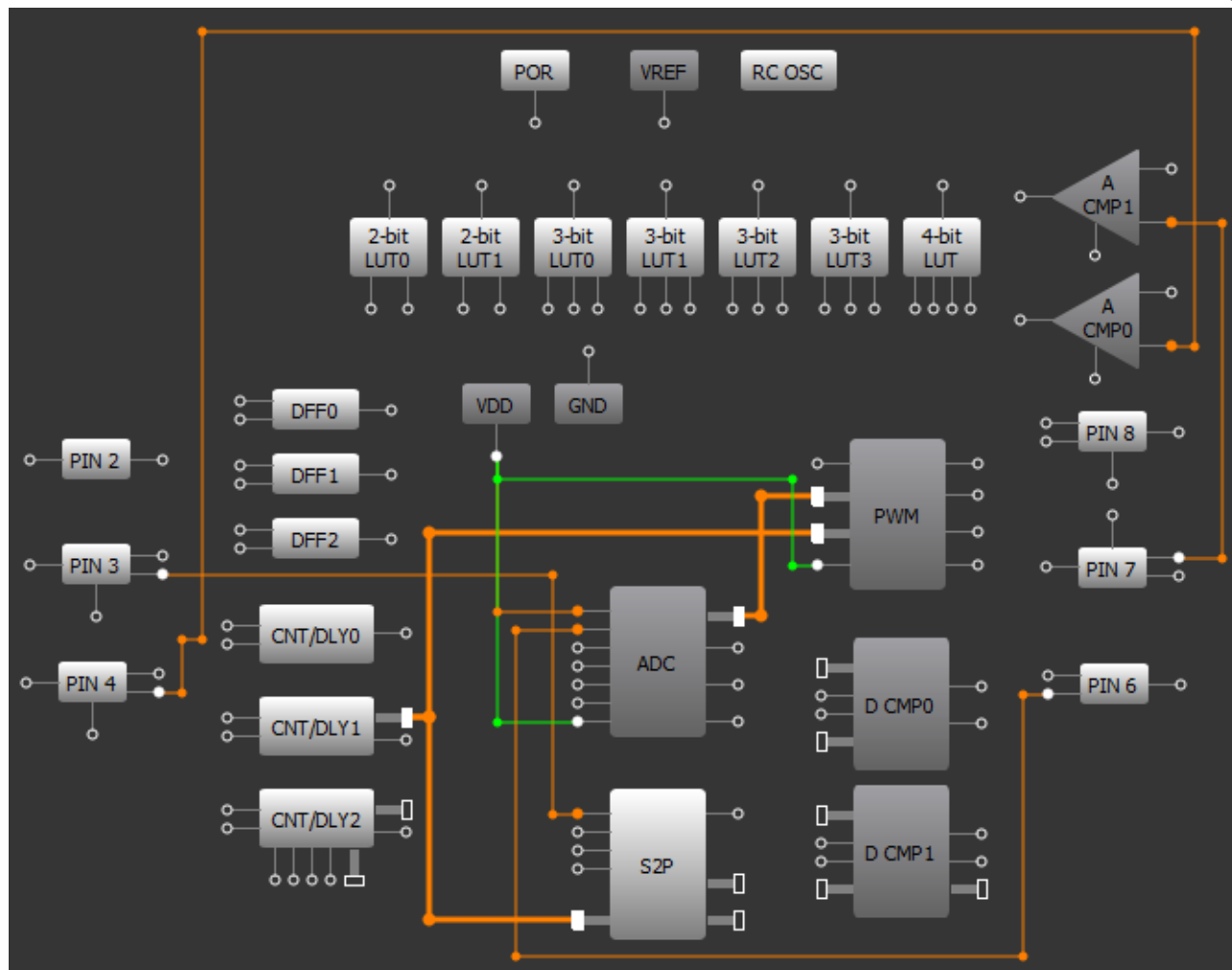


Рис. 2.15. Меню для реалізації виокремлення

12. Яка ідея основних алгоритмів структурного синтезу?
13. Назвіть методи дискретного математичного програмування.
14. Яка різниця між “Виокремленням” та “Нарощуванням”?
15. Як формулюється задача структурного синтезу?
16. Як формулюється задача параметричного синтезу?
17. Яке місце займає задача оптимізації в процесі структурного та параметричного синтезу?
18. Наведіть класифікацію задач структурного синтезу за критерієм складності задач.
19. Наведіть приклади задач параметричного синтезу.
20. Наведіть приклади задач структурного синтезу.
21. Наведіть приклади задач одноваріантного та багатоваріантного аналізу.
22. Наведіть класифікацію задач параметричного синтезу.

2.6. Список використаної літератури до другого розділу

- [1] Системы автоматизированного проектирования: / И.П. Норенков. Учеб. пособие для вузов: В 9-ти кн. Кн.1. Принципы построения и структура. – М.: Высшая школа, 1986. – 127 с.
- [2] Коваль В. О., Лобур М, В. Автоматизация технологического моделирования полупроводниковых ИС: Учебное пособие. – Львов: ЛПИ, 1987. – 84 с.
- [3] Петренко А. И., Семенов А. И. Основы построения систем автоматизированного проектирования. – К.: Вища школа., 1984. – 296 с.
- [4] Гранат П.П., Теслюк В.М. Система наскрізного проектування ІС – “ПроМІС” // Вісник Державного університету “Львівська політехніка”: Радіoeлектроніка та телекомунікації. – Львів, 2000. – № 399. – С. 164 – 169.
- [5] Норенков И. П. Основы теории и проектирования САПР / И. П. Норенков. – М. : Высш. Шк., 1990. – 334 с.
- [6] Корячко В. П., Курейчик В.М., Норенков И. П. Теоретические основы САПР : учеб. для вузов. / В. П. Корячко, В. М. Курейчик, И. П. Норенков. – М. : Энергоатомиздат, 1987. – 400 с.
- [7] Лобур М. В., Свірідова Т.В., Теслюк В.М. Аналіз кореляційних залежностей між механічними і електричними параметрами мікровбудованих систем // Вісник Національного університету “Львівська політехніка”: Радіотехніка та телекомунікації. – Львів, 2005. – № 534. – С. 147 – 152.
- [8] Lobur M., Sviridova T., Teslyuk V. Desing of micromechanical filter using correlation dependencies between mechanical and electrical parameters // Proc. of The 12-th Intern. Conf. on Mixed Design Of Integrated Circuits And Systems (MIXDES 2005). – Krakov, Poland, 2005. – P. 325 – 327.
- [9] Teslyuk V., Lobur M., Zaharyuk R., Al Omari Tarik Sensitivity theory application in MEMS design // Proc of the IX-th Intern. Conf. on The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM'2007). – Lviv – Polyana, Ukraine, 2007. – P. 536 – 537.

РОЗДІЛ 3.

АВТОМАТИЗАЦІЯ ПРОЕКТУВАННЯ МЕМС З ВИКОРИСТАННЯМ КОНСТРУКТОРСЬКОЇ БАЗИ ДАНИХ

3.1. Архітектура підсистеми роботи з КБД МЕМС

Провівши огляд існуючих систем автоматизованого проектування МЕМС [1], проаналізувавши їх сильні та слабкі сторони, стало очевидно, що існуючі системи спрощують процес проектування МЕМС, забезпечують користувача потужними інструментами дизайну та візуалізації, проте не існує потужних систем, які б давали змогу автоматизувати процес синтезу нових проектних рішень. Особливе місце в процесі автоматизованого синтезу проектних рішень займає конструкторська база даних. Переваги наявності такої бази даних очевидні, а саме:

- існує можливість отримання і внесення інформації про структурні елементи МЕМС необмеженою кількістю користувачів, що в свою чергу дасть можливість швидко наповнити інформацією таку базу даних;
- структурні елементи МЕМС з такої бази даних можуть бути використані в системі автоматизованого синтезу нових структур МЕМС;
- інформацію про існуючі елементи можна використати при розробленні бази знань та правил поєднання структурних елементів МЕМС;
- при розробленні власної КБД елементів МЕМС можна застосувати новітні технології проектування та програмування, що дасть змогу підвищити рівень надійності та швидкодії роботи такої БД.

Водночас, перед розробником постає низка вимог [2 - 5], яких необхідно дотриматись для забезпечення стабільного функціонування, наповнення та неодноразового використання інформації, що зберігається в базі даних. До таких вимог слід віднести:

- структура бази даних має забезпечити швидкий доступ до структурних елементів МЕМС;

- база даних має забезпечити можливість швидкого, багатокритеріального пошуку структурних елементів;
- система захисту бази даних має гарантувати безпеку даних;
- валідація вхідних даних має гарантувати ввід змістовних даних про структурні елементи МЕМС;
- база даних має забезпечити можливість інтеграції даних, що зберігаються в САПР МЕМС;
- має існувати можливість представити інформацію про структурні елементи МЕМС в форматі зрозумілому для проектанта (інженера).

Основним призначенням конструкторської бази даних структурних елементів МЕМС є використання інформації, що в ній зберігається, в процесі синтезу структурних схем МЕМС. Очевидно, що структура бази даних має бути адаптованою під систему, яка буде її використовувати. Іншими словами, від логіки роботи математично-логічного блоку системи, від методу розв'язання задачі синтезу структурних схем МЕМС, буде залежати структура конструкторської бази даних МЕМС. Адаптація структури бази даних - необхідний етап в процесі побудови системи синтезу, оскільки це дасть можливість збільшити продуктивність такої КБД, а саме:

- зберігати інформацію про МЕМС у структурах, оптимальних для використання експертною системою;
- прискорити процеси вибірки та запису;
- організувати внутрішні механізми роботи зі структурами даних, що пришвидшить роботу експертної системи.

Отже, першочерговою задачею є вибір методів та моделей, за допомогою яких буде здійснюватись розв'язання проектної задачі синтезу МЕМС.

Для розв'язання задачі синтезу нових структурних схем МЕМС, використовуються алгоритми оптимізації і/або алгоритми штучного інтелекту [1 - 5]. Існує розроблена КБД на основі ситуативно-базованій аргументації [6] (СБА) для розв'язання задач синтезу та оптимізації структурних рішень МЕМС. СБА – це перспективний метод, в напрямку якого зараз ведуться розробки провідними фахівцями в сфері САПР і МЕМС.

Конструкторська база даних використовується у системі і ґрунтується на методах СБА, має задовольняти наступні вимоги:

- структура КБД адаптована для використання переваг методів СБА;
- КБД зберігає інформацію не тільки про структурні елементи МЕМС, а й про їх властивості і навіть готові проектні рішення, які вже були використані для розв'язання попередніх задач структурного синтезу;
- реалізовує швидкі алгоритми пошуку та роботи з даними;
- підтримує можливість збереження інформації в форматі сприйнятному для користувача;
- забезпечити можливість інтеграції з САПР МЕМС.

Враховуючи ці аспекти, розроблено архітектуру підсистеми роботи з КБД [7, 8], яка виконує функції наповнення, редагування та валідації інформації, що зберігається в КБД МЕМС, а також полегшує роботу з конструкторською базою даних. Користувачу не потрібно бути спеціалістом у галузі баз даних для того, щоб додавати або редагувати інформацію про структурні елементи МЕМС в КБД, отже підсистема роботи з КБД має інтуїтивний інтерфейс користувача, який дає змогу без особливих труднощів працювати з таким програмним продуктом.

Оскільки в системі структурного синтезу МЕМС будуть використовуватись вже існуючі структурні елементи МЕМС [7], потрібно забезпечити можливість зберігання конструкторсько-технологічної документації елемента та забезпечити можливість зручного її перегляду. Для цього потрібно розробити функціонал, який надав би можливість підсистемі працювати/інтегруватися з такими САПР як Autocad, КОМПАС, тощо.

Розроблена архітектура підсистеми роботи з КБД МЕМС [8] представлена на рис. 3.1. Розроблена архітектура складається з чотирьох основних модулів:

1. Модуль роботи з базою даних;
2. Модуль інтеграції з САПР;
3. Модуль автоматизованого синтезу представлення МЕМС у XML форматі;
4. Інтерфейс користувача.

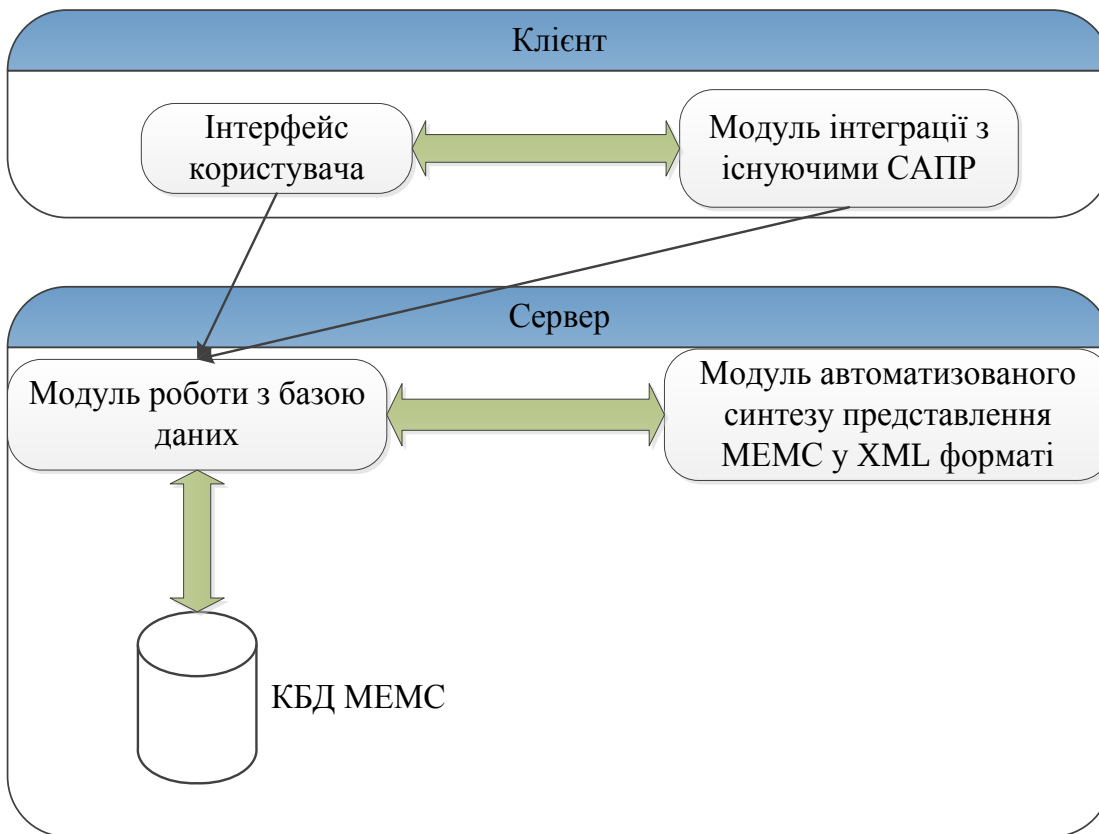


Рис. 3.1. Архітектура підсистеми роботи з КБД

Всі модулі є окремими програмними одиницями, що спроектовані як ієрархія класів, які реалізують певний функціонал. Кожен з модулів підтримує визначений інтерфейс, через який і здійснюється взаємодія між окремими модулями та забезпечення повної функціональності підсистеми роботи з КБД. Слід зазначити, що для кожного модуля розроблений клас виключень, які повністю характеризують і дають змогу ефективно опрацювати і повідомити користувача про системні помилки, що можуть виникнути при роботі з даним модулем.

3.2. Конструкторсько-технологічна база даних елементів MEMC

Складність розроблення КБД для зберігання інформації про MEMC полягає в ієрархічній структурі самих MEMC, в якій на кожному рівні ієрархії потрібно зберігати величезний обсяг інформації про об'єкти цього рівня

ієрархії, їх властивості, правила поєднання та конструкторсько-технологічну документацію. На відміну від існуючих баз даних елементів МЕМС, ключовою особливістю розробленої КБД є можливість її інтеграції в систему синтезу структурних схем МЕМС та обсяг інформації, який в ній зберігається. Спроектowana КБД [7, 8] містить не тільки інформацію про компоненти, елементи чи самі МЕМС, вона також охоплює всю складну ієрархічну структуру МЕМС, завдяки використанню чотирьох ієрархічних рівнів структурних складових МЕМС, та забезпечує можливість інтеграції з іншими підсистемами і використання даних, що в ній зберігаються для синтезу нових структурних схем МЕМС.

Класичним представленням МЕМС можна вважати трирівневу (системи, підсистеми, елементи) ієрархію, що включає МЕМС і її підсистеми, а також елементи, з яких складаються ці підсистеми. Однак, як доведено в роботах [9, 10, 11], часто бажані характеристики роботи елементів можна отримати оптимізувавши їх структурні елементи чи спосіб їх поєднання, тобто компоненти, з яких вони складаються. До таких компонентів можна віднести: балки, пружини, опори і нерухомі маси, тощо. Існує низка методів оптимізації та синтезу нових структурних рішень елементів МЕМС [12 - 16], що ґрунтуються на зміні чи модифікації їх складових компонентів. Взявши це до уваги, була розроблена чотирирівнева ієрархія МЕМС, що розширює і вдосконалює класичну трирівневу ієрархію завдяки впровадженню додаткового (найнижчого) рівня ієрархії – компонентного рівня, який дає можливість враховувати особливості елементів цього рівня та розв'язувати поставлені задачі за допомогою структурних змін на рівні компонентів. На рисунку 3.2 зображена вдосконалена чотирирівнева структура КБД МЕМС. Отже, КБД складається з таких чотирьох ієрархічних рівнів:

1. Рівень представлення МЕМС. Зберігає інформацію про найвищий рівень ієрархії МЕМС – саму систему. На цьому рівні зберігається інформація про базові властивості МЕМС, середовище та призначення системи, методи та області застосування, а також конструкторсько-технологічна документація системи.



Рис. 3.2. Структурне представлення рівнів КБД

2. Рівень представлення підсистем. Даний рівень відповідає за збереження інформації про підсистеми МЕМС (підсистема давачів, підсистема живлення, підсистема актюаторів тощо). Окрім загальних характеристик та властивостей підсистем, цей рівень містить структурні схеми підсистем, їх основні фізичні характеристики та вимоги до них.

3. Рівень представлення елементів. На цьому рівні зберігається інформація про множину елементів, які входять до складу підсистеми (актюатори, гіроскопи, дзеркала, резонатори, тощо). Окрім інформації про тип, спосіб застосування та ключові характеристики елементів МЕМС, даний рівень зберігає інформацію про математичні моделі елементів та способи обчислення ключових фізичних властивостей.

4. Рівень представлення компонентів. Цей рівень є доповненням до класичної тривірневої структури МЕМС. Оскільки метою його введення є потреба в оптимізації структури МЕМС на рівні елементів, то очевидно, що цей рівень повинен містити інформацію про фізичні властивості елементарних частинок – компонентів, з яких складаються елементи МЕМС. Логічним, також, є збереження інформації про властивості, які можуть бути зміненими в процесі оптимізації та синтезу нових структурних схем.

Окрім чотирьох основних рівнів ієрархії, КБД включає також три проміжні рівні, іншими словами три рівні зв'язків. Призначення цих рівнів полягає у зберіганні інформації про зв'язки між сусідніми ієрархічними рівнями МЕМС. Саме об'єкти цих рівнів зберігають інформацію про ієрархічну структуру конкретних МЕМС об'єктів, іншими словами вони визначають з яких підсистем складається та чи інша МЕМС, які елементи входять до складу підсистеми, чи які базові компоненти використані для побудови елементу МЕМС. Окрім цього, проміжні КБД містять інформацію про правила поєднання між рівнями ієрархії. Проміжні рівні забезпечують використання знань про поєднання структурних елементів різних ієрархічних рівнів МЕМС в процесі синтезу нових структурних схем МЕМС чи в процесі генерації множини альтернативних рішень. Спрощена структурна схема КБД представлена на рис. 3.3.

Під терміном “архітектура” розуміють структурну схему в поєднанні з топологічним розміщенням та функціональним призначенням елементів структури. Оскільки складність розробки, реалізації і підтримки КБД полягає також в обсязі інформації, яку вона має зберігати (якщо на рівні МЕМС систему можна охарактеризувати десятками властивостей, то з кожним ієрархічним рівнем їх кількість буде зростати в геометричній прогресії), то постає питання надійності роботи КБД, насамперед питання збереження інформації. Здійснивши детальний аналіз існуючих методів захисту інформації [17], та з метою підвищення надійності роботи КБД, прийнято рішення забезпечити можливість кластеризації КБД з використанням кількох серверів, на яких буде розміщуватись КБД.

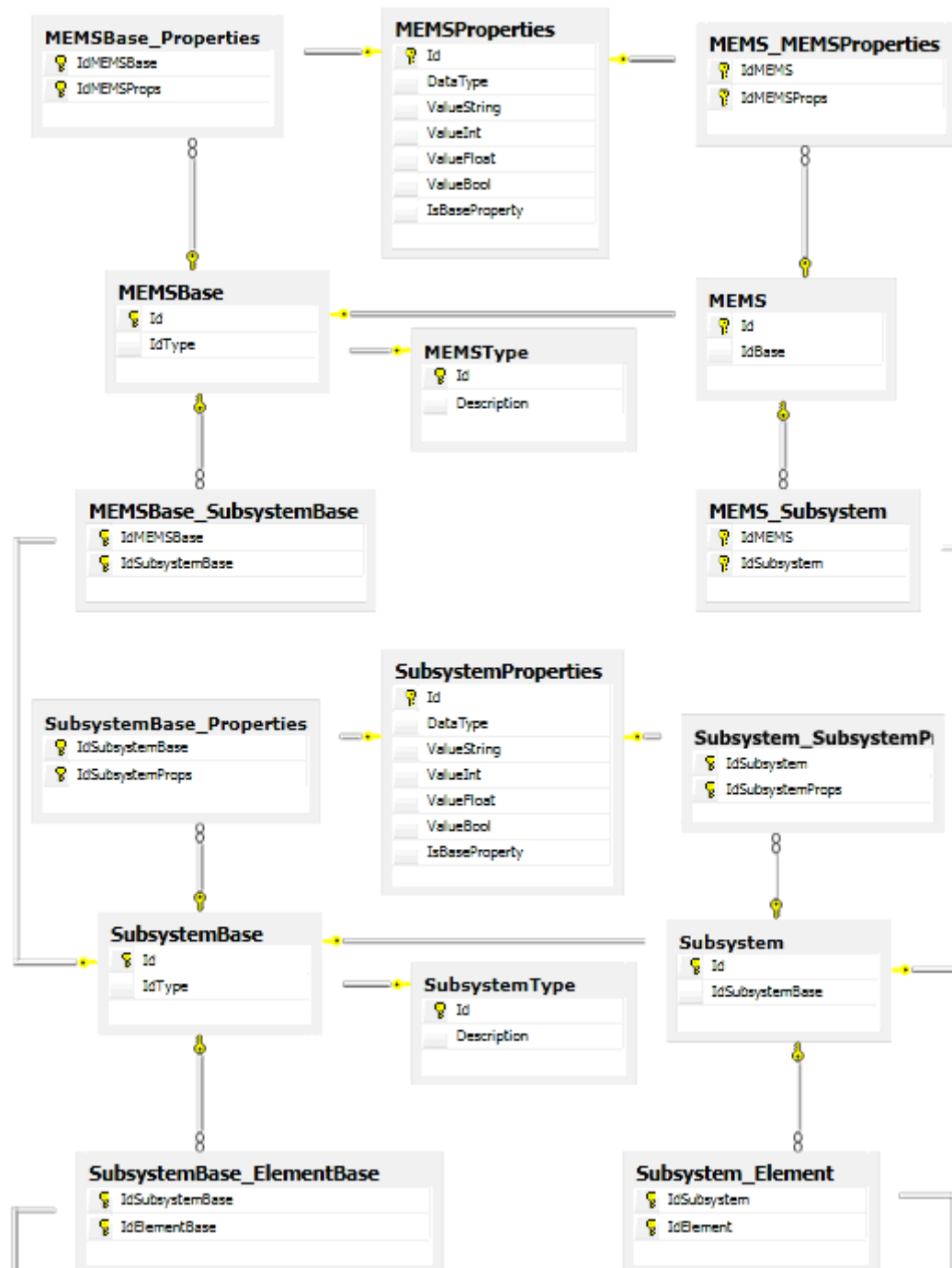


Рис. 3.3. Структурна схема КБД

Для кластеризації обраний тип кластеру «Активний/Пасивний» [18], що полягає в спільній роботі двох серверів, один з яких перебуває в активному стані, а інший в стані сну. У випадку збоїв в роботі активного серверу стабільність КБД буде забезпечено пасивним сервером. Такий тип кластеризації вимагає більших фінансових затрат, але забезпечить зручність в налаштуванні та адмініструванні, а також збільшить швидкодію роботи сервера КБД. Схема

кластеру «Активний/Пасивний» зображена на рис. 3.4. З метою зменшення витрат можна використовувати 2 серверних кластери.

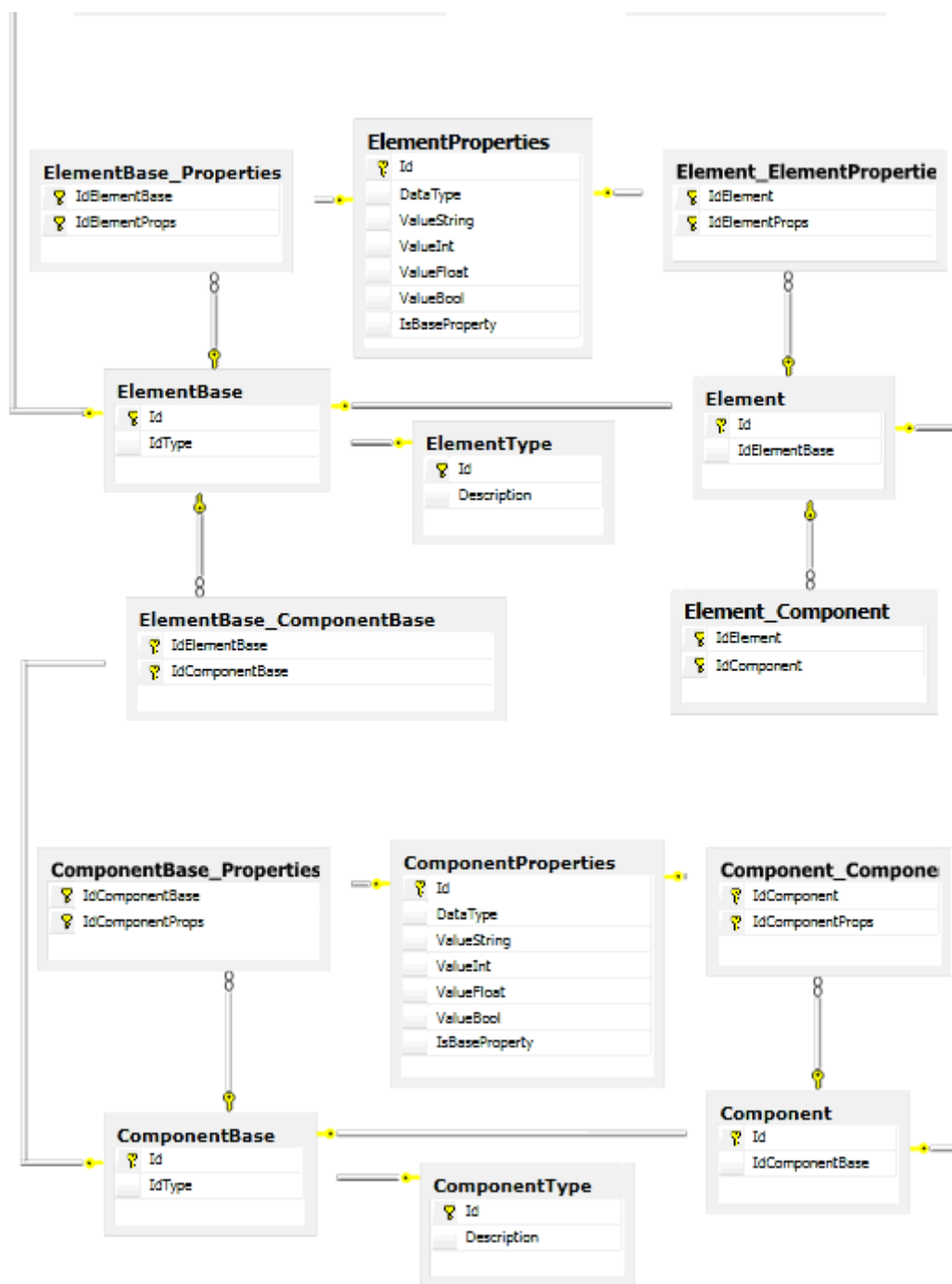


Рис.3.3. Структурна схема КБД (продовження)

3.3. Побудова інформаційної моделі КБД

КБД МЕМС ґрунтується на блочно-ієрархічному підході і складається з кількох ієрархічно пов'язаних рівнів, кожен з яких зберігає інформацію про

один з основних структурних рівнів МЕМС. Однак, всі структурні рівні МЕМС містять певну схожу інформацію, таку як:



Рис. 3.4. Кластеризація типу Активний/Пасивний

Дані про об'єкт, що описується, а саме: тип об'єкту, його призначення, конструкторсько-технологічну документацію, параметричні моделі об'єкту, тощо.

Властивості об'єкту характеризують структурну одиницю МЕМС як фізичний об'єкт. Властивості, що зберігаються в КБД, можуть бути як фізичними, наприклад розмір чи площа поверхні, так і властивостями МЕМС, такими як чутливість чи точність.

Взявши до уваги наявність структур даних спільних для усіх ієрархічних рівнів МЕМС, розроблено інформаційну модель узагальненого представлення кожного зі структурних рівнів МЕМС. Спільні структури даних для кожного з рівнів поєднано таким чином, щоб згодом спростити роботу підсистем, які будуть оперувати інформацією з КБД МЕМС. Іншими словами, розроблена інформаційна модель ієрархічного рівня МЕМС була оптимізована для використання методами та засобами об'єктно-орієнтованого програмування та забезпечує швидкий доступ і вибірку інформації з різних рівнів ієрархії МЕМС за допомогою шаблонних структур, що в свою чергу призводить до підвищення

рівня автоматизації процесу проектування. Всі переваги розробленої інформаційної моделі можна розглянути на прикладі структури одного із ієрархічних рівнів MEMC. На рис. 3.5 зображена структура ієрархічного рівня підсистем MEMC.

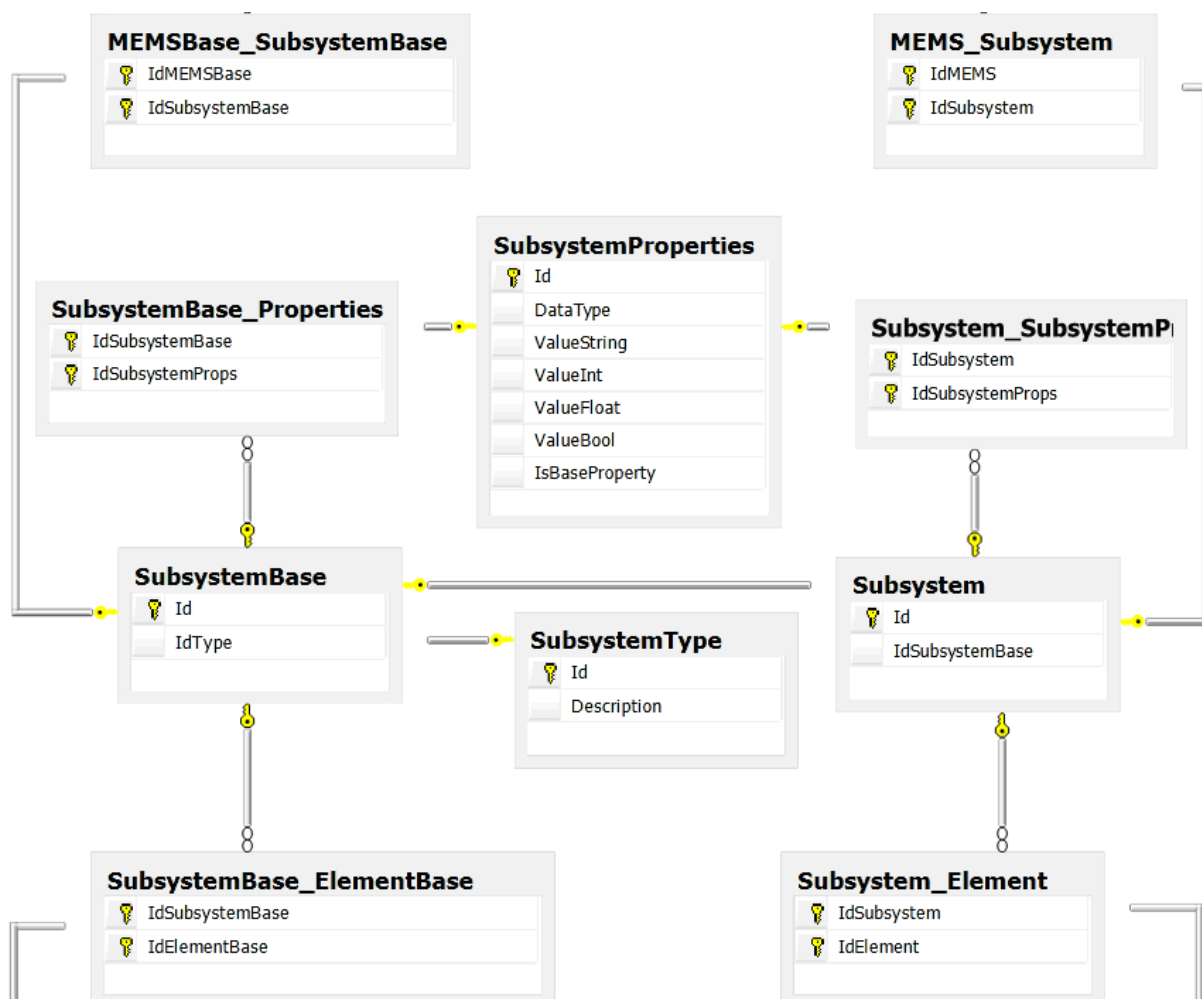


Рис. 3.5. Структура ієрархічного рівня підсистем MEMC

Принцип побудови кожного з рівнів спрощено можна зобразити за допомогою схеми (рис. 3.6). Як видно зі схеми на рисунку 3.6, кожен ієрархічний рівень містить інформацію про базовий компонент та дані про поєднання цього компонента з вищим та нижчим рівнями ієрархії. Виключення становлять тільки рівні представлення MEMC і компонентів, бо ці рівні є найвищим і найнижчим рівнями представлення структурних об'єктів MEMC, і

вони не містять інформації про поєднання компонентів свого рівня з вищими та нижчими рівнями ієрархії, відповідно.

Слід відзначити, що з метою представлення узагальненої інформації про різноманітні шаблонні типи мікроелектромеханічних систем до розробленої інформаційної моделі було додано структури даних для представлення так званих шаблонів об'єктів MEMC. Під шаблоном слід розуміти узагальнений об'єкт певного типу, який має тільки ті властивості, що характерні для об'єктів цього типу.

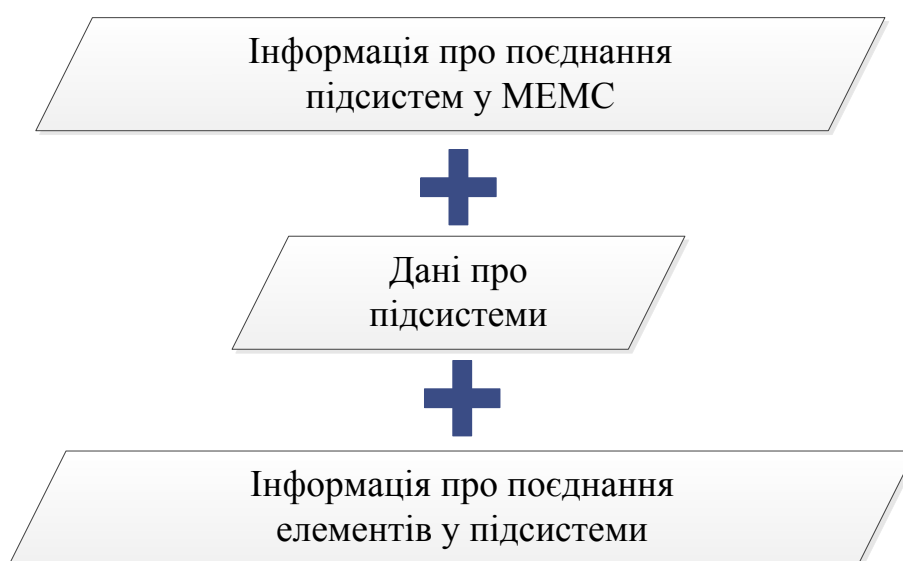


Рис. 3.6. Схема принципів побудови кожного з ієрархічних рівнів представлення MEMC

Шаблон об'єкту також може містити інформацію про притаманні для всіх об'єктів даного типу способи поєднання з іншими структурними елементами MEMC, тощо. Керуючись принципами об'єктно-орієнтованого програмування, шаблони стали “батьками” для всіх структурних елементів даного типу, що зберігаються в базі даних. В КБД MEMC поряд з блочно-ієрархічною структурою елементів MEMC існує структура шаблонів MEMC, яка поєднується за тими ж правилами, що і елементи. Об'єкти, що представляють інформацію про конкретну реалізацію мікросистеми, поєднуються за правилами поєднання шаблонів, оскільки кожен елемент походить від шаблону.

Схему ієрархічного рівня підсистем, яка відображає зв'язок об'єктів і шаблонів, зображено на рис. 3.7.

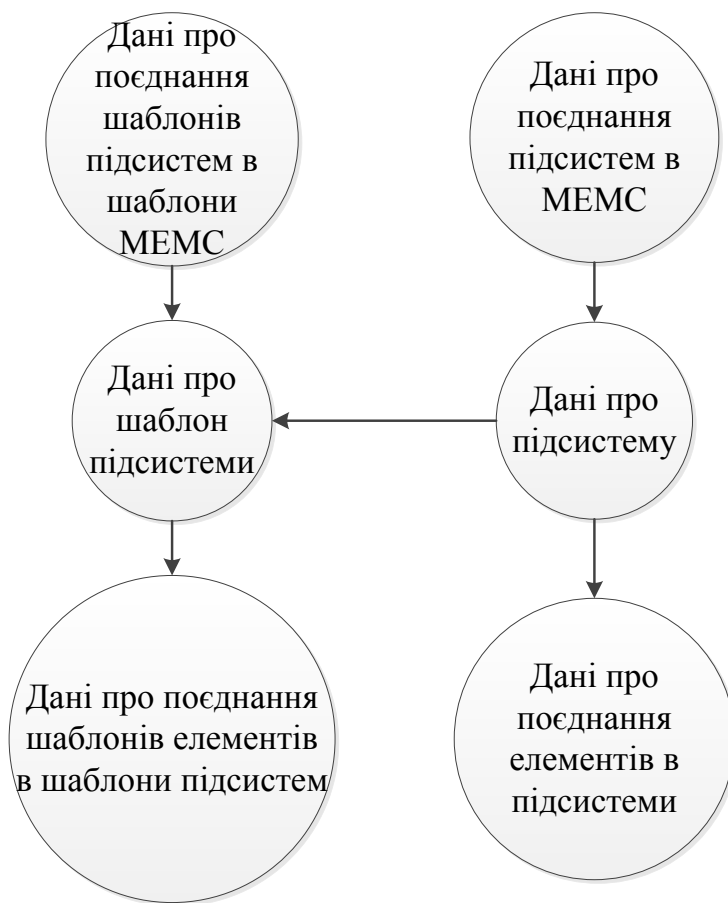


Рис. 3.7. Схема зв'язків між ієрархічними рівнями шаблонів та об'єктів

Детальніше інформаційну модель представлення кожного зі структурних рівнів MEMC краще розглядати на прикладі таблиць рівнів підсистем. Отже, для представлення шаблону MEMC використовуються 4 таблиці даних, що зображені на рисунку 3.8:

1. Тип підсистеми (SubsystemType). Містить дані про тип підсистеми. Оскільки об'єкти підсистем базуються на шаблонах, тип шаблону поширюється на об'єкт.
2. Шаблон підсистеми (SubsystemBase). Містить інформацію про шаблон підсистеми.

3. Властивості підсистеми (SubsystemProperties). Зберігає набір властивостей характерних для всіх об'єктів даного типу. Слід відзначити, що конкретні об'єкти можуть мати свої власні відмінні від шаблонних властивості.

4. Проміжна таблиця зв'язків між шаблоном підсистеми та властивостями (SubsystemBase_Properties). Забезпечує зв'язок багато до багатьох, адже певні властивості можуть бути характерними кільком шаблонам, наприклад розмір чи технологія виробництва.

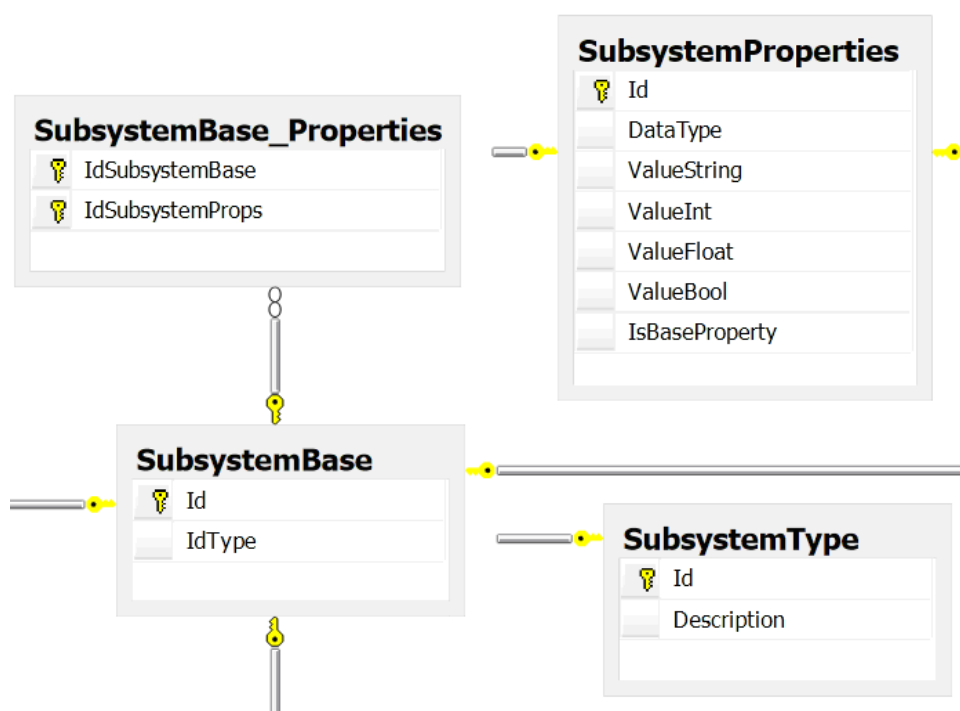


Рис. 3.8. Таблиці представлення інформації про шаблон підсистеми

Для опису підсистеми використовуються такі таблиці (рисунок 3.9):

Підсистеми (Subsystem). Зберігає вказівник на шаблон підсистеми та інформацію про підсистему.

Проміжна таблиця зв'язків між підсистемою та властивостями (Subsystem_SubsystemProperties). Забезпечує зв'язок «багато до багатьох». По суті, взаємозв'язок між властивостями та об'єктами підсистем можна було б описати зв'язком «один до багатьох», адже конкретне значення властивості притаманне лише одному конкретному об'єкту. Зв'язок «багато до багатьох» використовується у зв'язку з тим, що таблиця властивостей спільна для

шаблонів підсистем і об'єктів підсистем. Це дає змогу збільшити швидкодію і ефективність пошуку за властивостями, але унеможлиблює зв'язок «один до багатьох», оскільки наявність зовнішнього ключа ідентифікатора об'єкта підсистеми в таблиці властивостей робить неможливим додавання в неї записів про властивості шаблонів без наявності об'єкта, що є абсурдним, оскільки шаблон підсистеми має бути створеним раніше, ніж об'єкт.

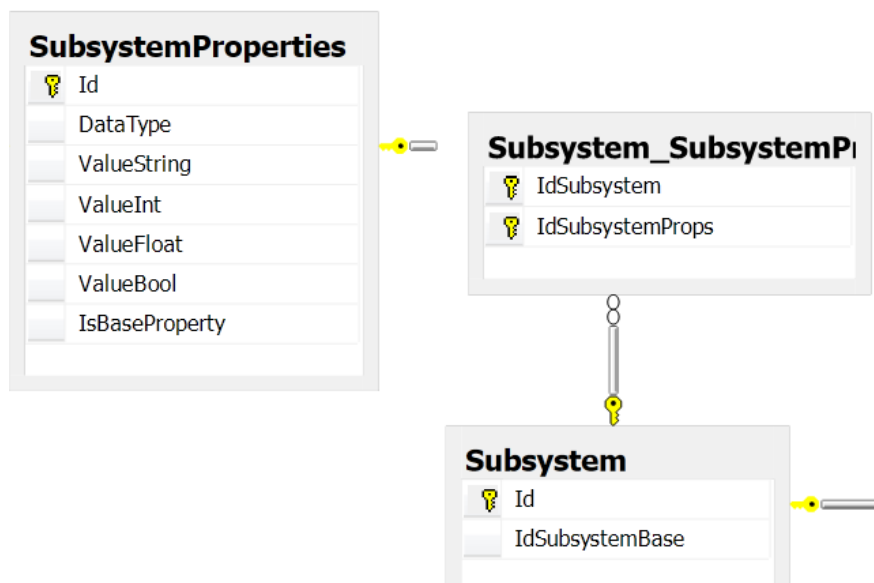


Рис. 3.9. Таблиці представлення інформації про підсистеми

Таблиця властивостей (**SubsystemProperties**). Оскільки таблиця властивостей є спільною для шаблонів і об'єктів підсистем, її детальну структуру представлено на рис. 3.10.

SubsystemProperties			
	Column Name	Data Type	Allow Nulls
▶	Id	int	☐
	DataType	int	☐
	ValueString	nvarchar(50)	☒
	ValueInt	int	☒
	ValueFloat	float	☒
	ValueBool	bit	☒
	IsBaseProperty	bit	☐
			☐

Рис. 3.10. Структура таблиці властивостей підсистеми

На перший погляд, таблиця для зберігання властивостей підсистем не відповідає вимогам нормалізації бази даних, оскільки містить надлишкову інформацію (поля для збереження даних різних типів). Але переваги запропонованої структури значно переважають втрати, пов'язані з денормалізацією цієї таблиці. Насамперед, в даній таблиці зберігаються дані як про властивості шаблонів, так і про властивості об'єктів. Про приналежність властивості до шаблону свідчить прапорець `IsBaseProperty`. Нестандартно є також реалізована модель зберігання властивостей, оскільки замість звичної для нормалізованої бази даних наявності кількох таблиць з даними певного типу, таблиця властивостей містить чотири типи значень (`ValueInt`, `ValueBool`, `ValueFloat` і `ValueString`) і поле, в якому зберігається інформація про тип даних. Переваги такої моделі перш за все в швидкості доступу до даних, пошуку і універсальності властивостей, які можуть зберігатись, адже така модель може зберігати властивості не тільки вказаних чотирьох типів, але, наприклад, конструкторсько-технологічну документацію, файл якої знаходиться в файловій структурі підсистеми керування КБД, а шлях до цього файлу зберігається в полі `ValueString`. Тип даних при цьому буде відповідати значенню `BinaryData`.

Зв'язок між шаблонами підсистем і об'єктами підсистем з вищим і нижчим рівнями ієрархії реалізується за допомогою таблиць зв'язку шаблонів МЕМС та шаблонів підсистем, шаблонів підсистем та шаблонів елементів і дзеркальної комбінації для об'єктів підсистем. На рис. 3.11 зображений зв'язок між вищеописаними рівнями.

3.4. Розроблення програмної моделі представлення МЕМС в КБД

Структури даних, що зберігаються в КБД і представляють інформацію про МЕМС, мають ієрархічну структуру, оскільки описують елементи чотирьох рівнів, з яких складаються мікроелектромеханічні системи. Можна провести

паралель між структурою представлення МЕМС і структурою представлення знань – фреймом [19 - 23]. Типова структура фрейму зображена на рис. 3.12:

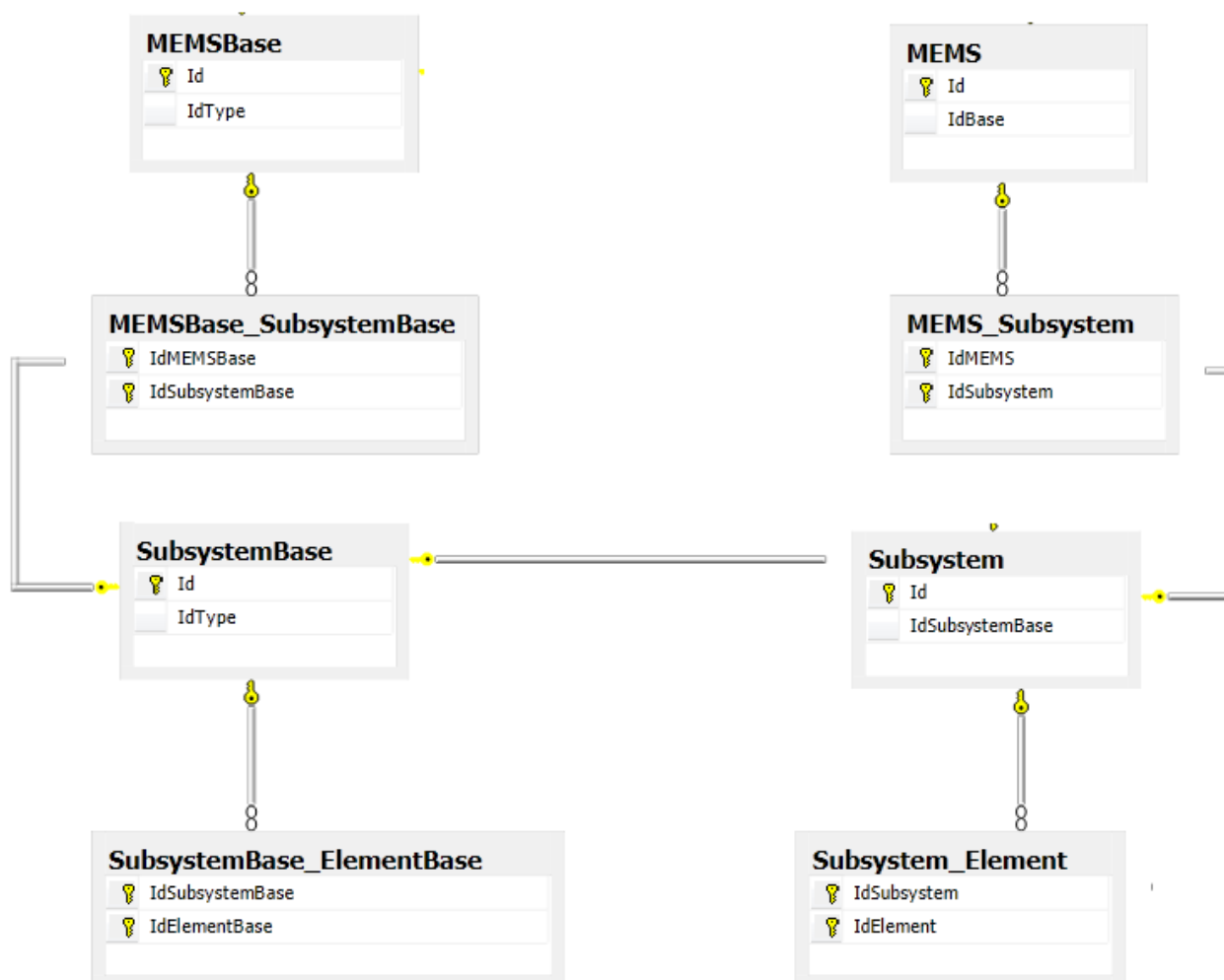


Рис. 3.11. Зв'язок підсистем з верхнім та нижнім рівнями ієрархії

На рис. 3.12 зображено типову структуру фрейму, яка є однорідною і складається з назви фрейму (тобто сутності) і слотів, які характеризують даний фрейм. Ключовою особливістю фреймів є те, що роль слоту можуть виконувати інші фрейми. Такі поєднання утворюють фреймову ієрархію. Структура даних МЕМС дуже схожа, вона включає інформацію, що характеризує об'єкти МЕМС, та інформацію про об'єкти, що входять до їх складу. Для представлення інформації про ієрархічні структури ідеально підходить формат

XML [24, 25], який за своїм визначенням є “стандартом побудови мов розмітки ієрархічно структурованих даних для обміну між різними застосуваннями”.

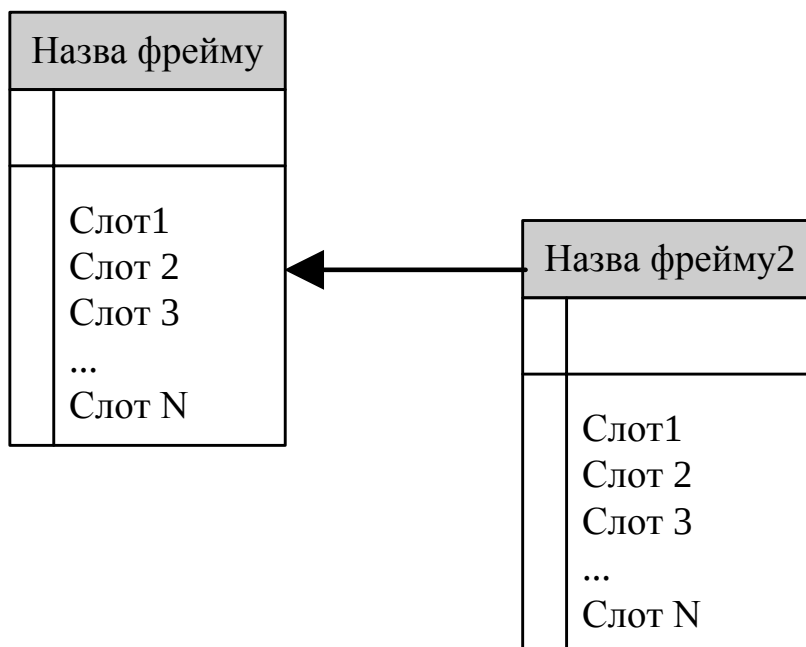


Рис. 3.12. Типова структура фрейму

Використання формату XML є доцільним також тому, що в процесі структурного синтезу MEMC в проектанта може виникнути потреба в перегляді інформації про MEMC на поточному етапі проектування. Саме так система може надати користувачу інформацію про поточний стан процесу синтезу у зрозумілому форматі. Типова структура рівнів програмного забезпечення представлена на рис. 3.13.

Оскільки процес синтезу структурних схем MEMC [26 - 28] є надзвичайно складним, то система синтезу буде складатися з кількох підсистем, кожна з яких буде виконувати певні вузькоспеціалізовані задачі. Кожна з таких підсистем має отримувати вхідну інформацію у зрозумілому для неї форматі, а також після завершення роботи має сформулювати розв’язок таким чином, щоб він був зрозумілим не тільки для інших підсистем, але і для користувача. Для реалізації функції обміну інформацією між підсистемами і формування проміжної інформації про стан проектування на різних етапах синтезу

розроблено XML представлення інформації про MEMC, яке може бути сформоване системою на будь-якому з етапів процесу синтезу нових проектних рішень.

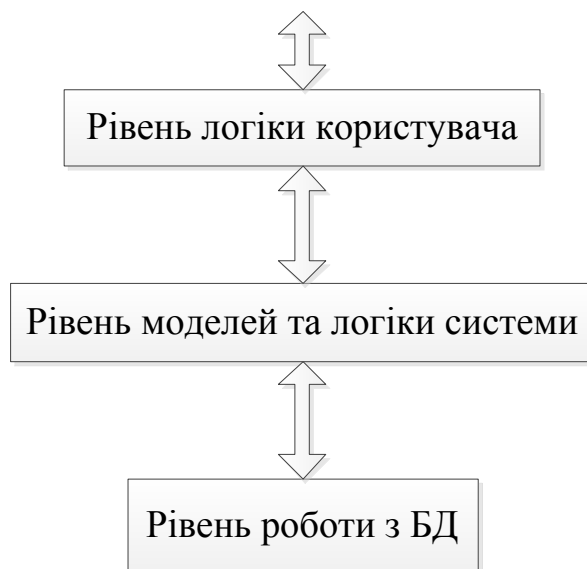


Рис. 3.13. Типова структура рівнів ПЗ

Оскільки інформацію про стан проектування, про згенеровані рішення чи про існуючі MEMC моделі потрібно генерувати в процесі прийняття рішення, актуальною є задача автоматизації процесу створення XML представлень об'єктів КБД.

Для швидкого та надійного перетворення інформації про MEMC та процес проектування в формат XML чи з нього (у випадку опрацювання підсистемою вхідних даних) використано механізм серіалізації/дсеріалізації. Суть даних перетворень полягає в процесі приведення об'єкта даних в форму XML структури, а при десеріалізації відбувається зворотній процес. Серіалізація/десеріалізація об'єктів - це автоматизований налаштований процес, що конфігурується за допомогою атрибутів і дає можливість отримати з програмної моделі XML представлення необхідної структури. Отже, для автоматизації процесу перетворення інформації про MEMC у XML структуру необхідним є розроблення об'єктної моделі представлення MEMC.

Побудована програмна модель представлення MEMC ґрунтується на принципах об'єктно-орієнтованого програмування, вона адаптована (базується

на схожих принципах та дає змогу працювати зі структурами даних КБД без додаткових перетворень) до структури КБД і забезпечує не тільки можливість представлення інформації про МЕМС в XML форматі, але й ефективну роботу з КБД, підвищення швидкодії і економію фізичної пам'яті в процесі синтезу структурних схем.

Ядром програмної моделі є дві сутності – базовий об'єкт і властивість. Як описано вище, структура КБД реалізована таким чином, що властивості об'єктів і шаблонів зберігаються разом. Також у шаблонів кожного ієрархічного рівня МЕМС та його об'єктів є спільні сутності (це зображено на рис. 3.14).

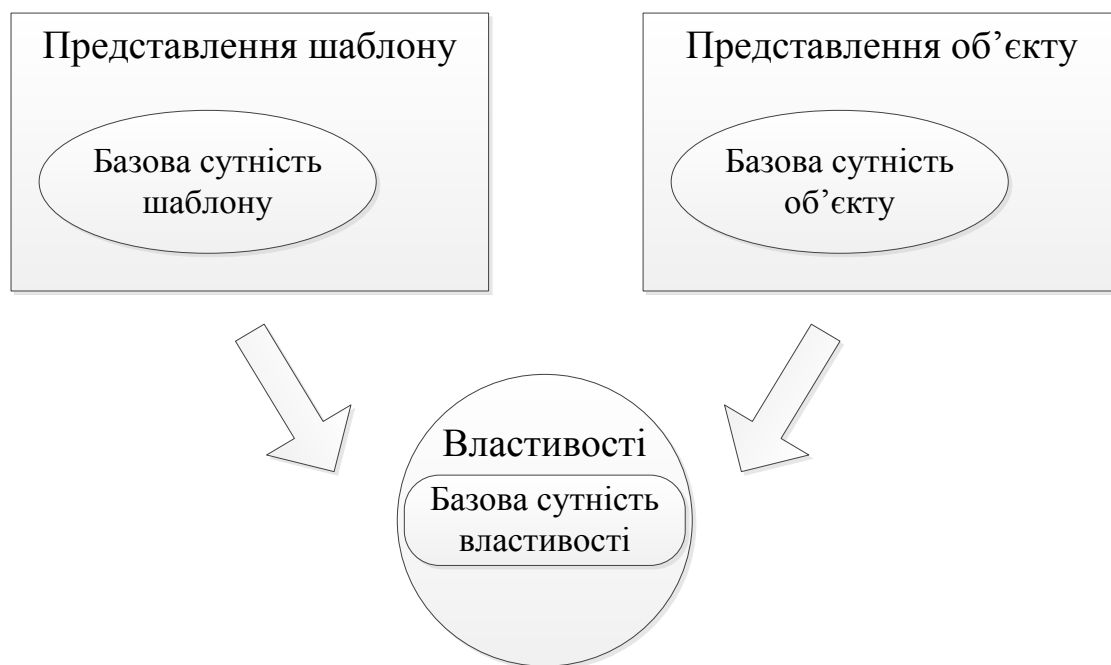


Рис. 3.14. Базові сутності і зв'язки між ними

Фактично на всіх рівнях ієрархії МЕМС можна виокремити дві базові сутності – базовий об'єкт і базову властивість. Це дає суттєву перевагу, оскільки величезний обсяг інформації на всіх рівнях ієрархії МЕМС можна представити використавши лише одну програмну модель. Ієрархічну схему представлення МЕМС з допомогою базових сутностей можна звести до вигляду зображеного на рисунку 3.15. На практиці, забезпечивши в системі механізм

серіалізації та десеріалізації об'єктів цих сутностей фактично буде реалізовано механізм XML перетворень основної частини даних про MEMC в КБД.

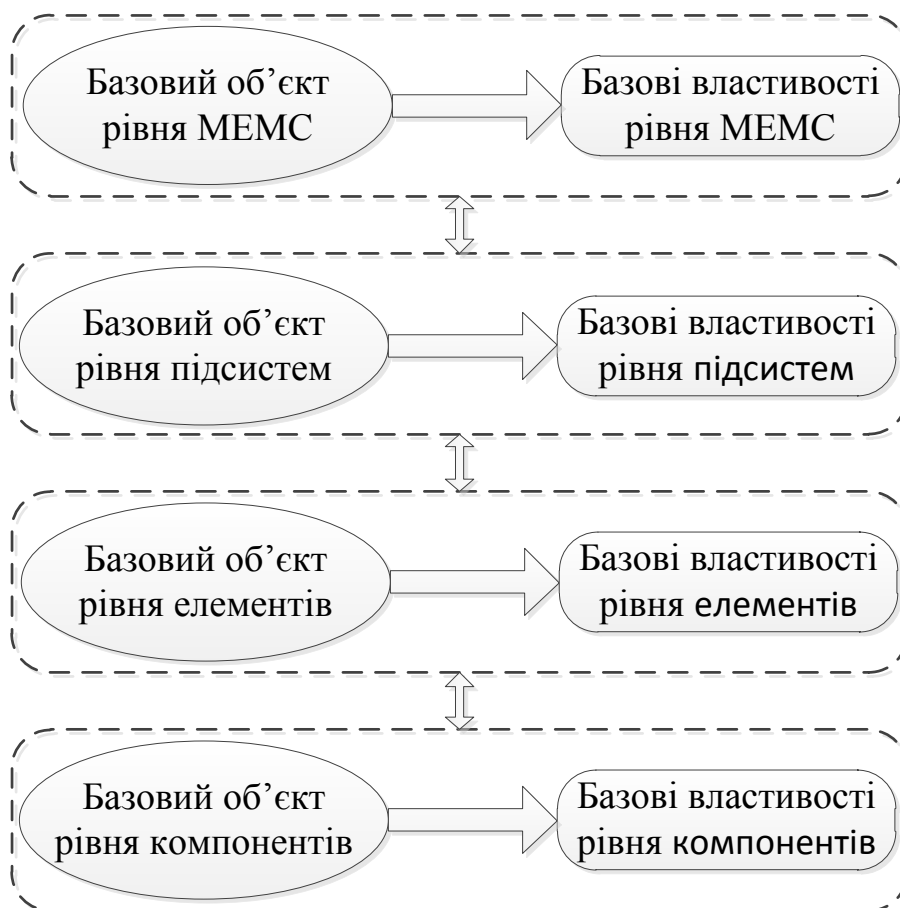


Рис. 3.15. Представлення ієрархічної схеми за допомогою базових сутностей

Для того, щоб повністю описати програмну модель представлення ієрархічної структури MEMC об'єктів, базові сутності об'єктів було розширено сутностями, що характерні для окремих рівнів ієрархії. Наприклад MEMC, окрім власних параметрів, конструкторсько-технологічної документації та інших властивостей, складається з набору підсистем, які в свою чергу є поєднанням елементів, що сформовані поєднанням компонентів. Отже, базовою структурою для всіх об'єктів різних рівнів ієрархії є базова сутність представлення елемента ієрархічного рівня MEMC. Структурна схема

наслідування розширень ієрархічних рівнів від базового об'єкта зображена на рис. 3.16.

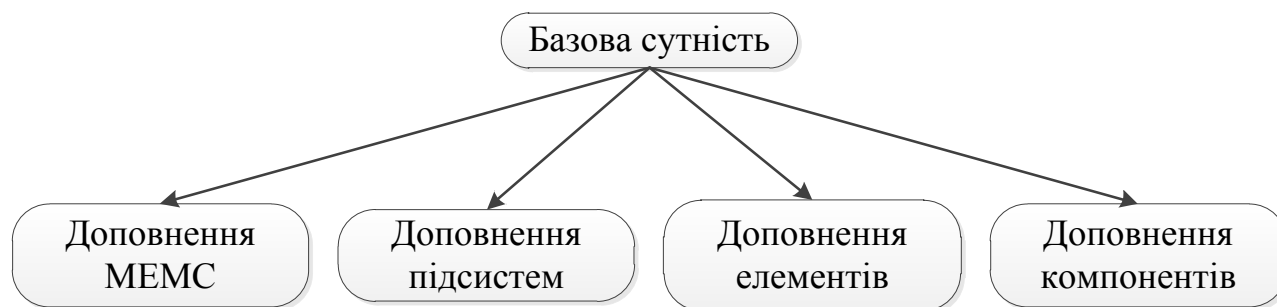


Рис. 3.16. Ієрархія наслідування доповнень рівнів від базового об'єкта

Отже, описавши доповнення для кожного з ієрархічних рівнів MEMC, отримано програмну модель представлення MEMC, що за допомогою алгоритмів серіалізації автоматично приводиться до XML представлення. Структура XML представлення визначається за допомогою атрибутів серіалізації, що застосовуються до властивостей програмної моделі.

Повна ієрархічна структура програмної моделі представлення MEMC в КБД елементів MEMC зображено на рис. 3.17. Всі ієрархічні зв'язки зображені на рис. 3.17 зберігаються у XML представленні MEMC.

3.5. Розроблення лінгвістичного забезпечення системи управління КБД

Оскільки елементи MEMC, які будуть зберігатися в конструкторській базі даних мають містити не тільки інформацію про базові характеристики, але й конструкторсько-технологічну документацію, було прийнято рішення долучити до інформації про елемент файли з відповідними даними.

Проаналізувавши ряд форматів представлення конструкторських даних, для збереження конструкторсько-технологічної документації елемента MEMC було вибрано формат dwg [29], який є основним форматом для деяких САПР, зокрема, AutoCad [30], nanoCAD, IntelliCAD. До переваг даного формату слід віднести те, що формат dwg підтримується багатьма САПР програмами

побічно: тобто дані з одного формату даних переміщуються в іншій через функції імпорт-експорт. Це і стало основним чинником, який обумовив вибір даного формату. Так, формат dwg підтримують усі відомі системи проектування та створення конструкторсько-технологічної документації, такі як SolidWorks, DataCad тощо.

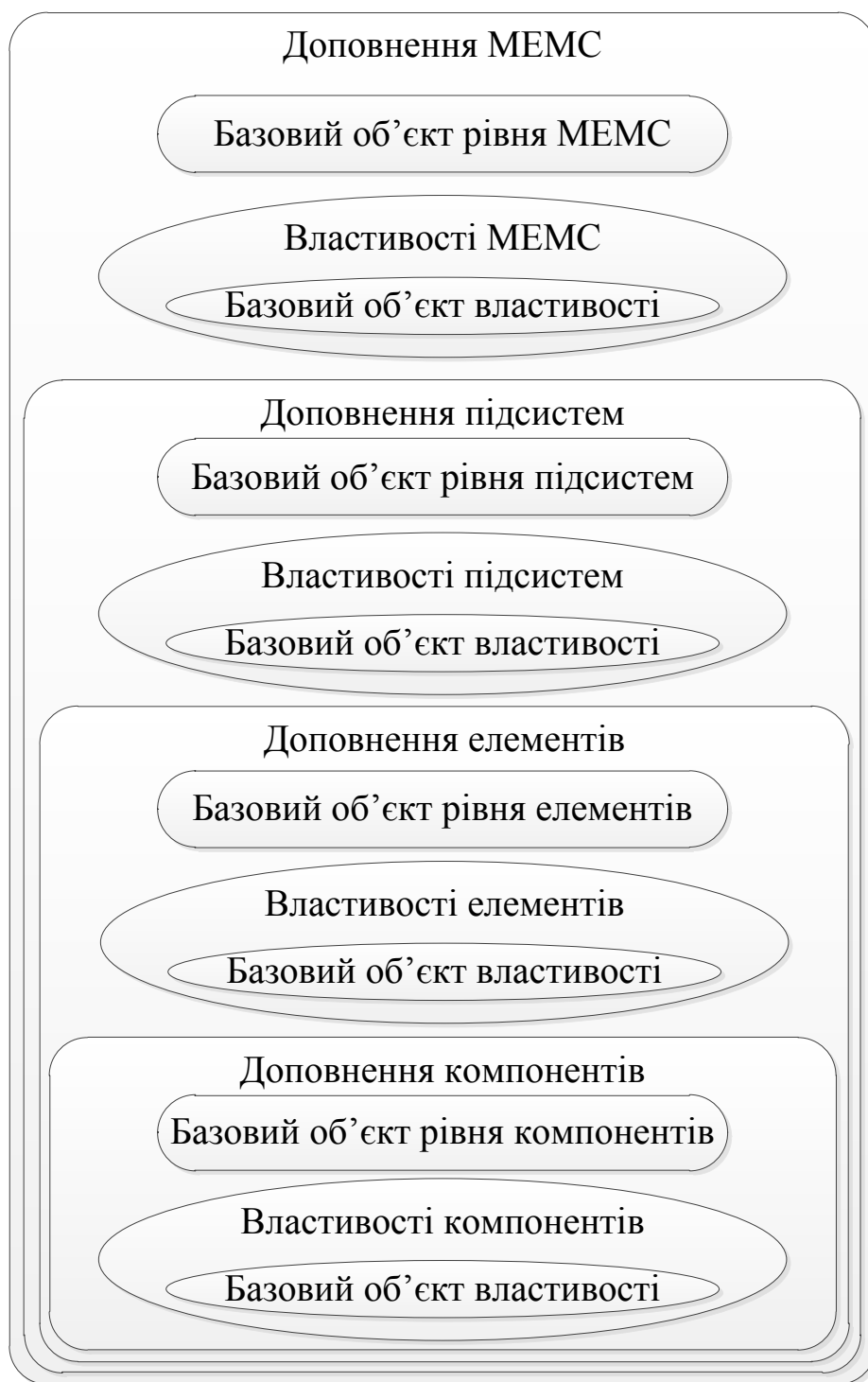


Рис. 3.17. Повна ієрархічна структура програмної моделі представлення MEMС в КБД

На рис. 3.18 показана взаємодія підсистеми роботи з КБД та різноманітних САПР через файли формату dwg.

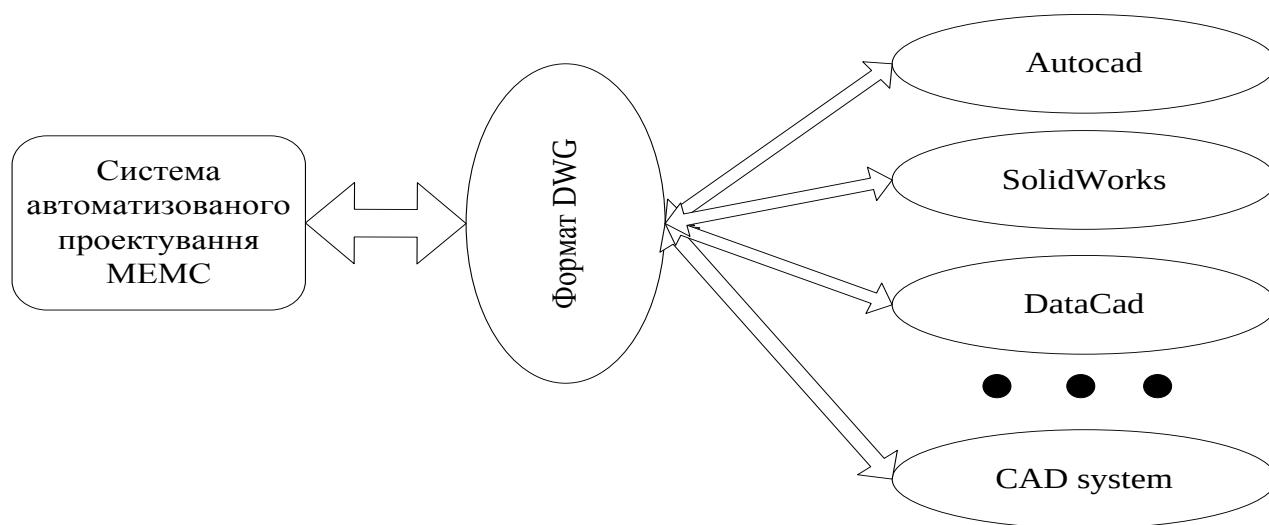


Рис. 3.18. Схема взаємодії підсистеми роботи з КБД та різноманітних САПР за допомогою файлів в форматі dwg

Формат dwg є універсальним форматом для збереження конструкторсько-технологічної інформації про елемент MEMС. Для представлення інформації про MEMС та структурні елементи MEMС у форматі зрозумілому для користувача в програмі використовується формат XML.

Завдяки розробленій моделі представлення властивостей MEMС, підсистема роботи з КБД MEMС забезпечує зручний спосіб інтеграції з іншими системами. Оскільки структура представлення властивостей може зберігати значення всіх основних типів даних, то файли конструкторсько-технологічної документації чи результати моделювання MEMС можна зберегти наступним чином: створити нову властивість з відповідною назвою, наприклад “Результати моделювання”, і в поле значення записати шлях до файлу, що зберігає інформацію про результати моделювання. В даному випадку можна перемістити файл до структурованого сховища даних системи, або залишити адресу його розміщення без змін. Підсистема роботи з КБД забезпечує можливість налаштування і співставлення форматів даних та систем, що з ними працюють. Отже, можна налаштувати автоматичне відкривання результатів моделювання відповідною системою, наприклад використати ANSYS.

Для обміну даними між підсистемами і представлення інформації про структурні елементи МЕМС розроблено XML модель для опису компонентів МЕМС. На рисунку 3.19 зображено XML представлення структурного компонента МЕМС елементів.

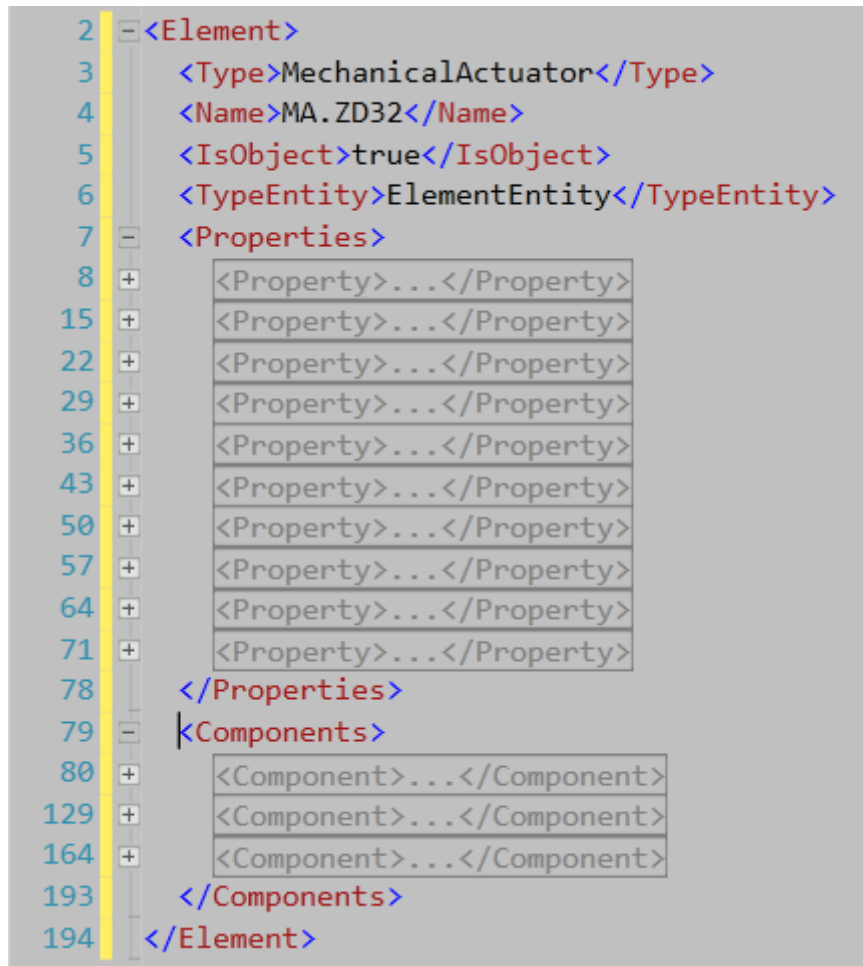


Рис. 3.19. XML представлення елементу МЕМС

Як видно з рис. 3.19, для обміну інформацією про МЕМС необхідно представити у вигляді XML наступну інформацію:

- дані про елемент, який описується;
- інформацію про властивості цього елемента;
- інформацію про компоненти, з яких складається даний елемент.

Можна обрати два методи опису компонентів: загальний і детальний. При детальному описі компонентів інформація про них буде деталізована так само,

як інформація про елемент МЕМС. При загальній деталізації XML модель буде містити тільки загальну інформацію про компонент.

3.6. Контрольні запитання до розділу

1. Що ви розумієте під конструкторською базою даних?
2. Яке призначення КБД?
3. Які переваги використання КБД в процесі синтезу структур МЕМС?
4. Які основні складові КБД МЕМС?
5. Що Ви розумієте під інформаційною моделлю?
6. Які основні складові включає інформаційна модель КБД МЕМС?
7. Які функції виконує КБД МЕМС?
8. Як враховано ієрархічну структуру МЕМС в процесі розроблення КБД?
9. Яка особливість програмної моделі КБД МЕМС?
10. Для чого використано мову XML в процесі побудови КБД МЕМС?

3.7. Список використаної літератури до третього розділу

- [1] Программное обеспечение для проектирования MEMS устройств компании Coventor [Електронний ресурс].— Режим доступу: <http://www.eurointech.ru/coventor>.
- [2] Василюк Я. Р. Аналіз принципів, моделей, методів та засобів для побудови онтологій області мікроелектромеханічних систем / Я. Р.Василюк, В. М. Теслюк, А. Я.Зелінський // Зб. наук.—техн. пр.: Науковий Вісник НЛТУ України.— 2011.— Вип. 21.12.— С. 322—330.
- [3] Згуровський М. З. Основи системного аналізу / М. З. Згуровський, Н. Д. Панкратова.— К., 2007.— 544 с.
- [4] Норенков И. П. Основы автоматизированного проектирования: Учеб. для вузов / И. П. Норенков.— 2-е изд., перераб. и доп.— М. : Изд—во МГТУ им. Н.Э. Баумана, 2002.— 336 с.
- [5] Теслюк В. М. Моделі та інформаційні технології синтезу мікроелектромеханічних систем / В. М. Теслюк // Монографія.— Львів: Вежа і Ко, 2008.— 192 с.
- [6] Leake D. B. Case—based reasoning / D. B. Leake // The Knowledge Engineering Review.— Cambridge University Press.— 1994.— P. 61—64
- [7] Розроблення системи управління та бази даних елементів МЕМС / В. М. Теслюк, А. Я. Зелінський, Х. Аль Шавабкех, С. І. Бобало // Збірник наукових праць інституту проблем моделювання в енергетиці ім. Г. Є. Пухова НАН України.— Київ, 2008.— Вип. 48.— С.192—199.
- [8] Zelinsky A. Database design for system that solves MEMS synthesis tasks / A. Zelinsky, V. Teslyuk // Proc. of the XI Intern. Conf. on The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM'2011).— Lviv – Polyana, Ukraine.— 2011.— P. 420.
- [9] Zhou N. Automated Design Synthesis for Micro-Electro-Mechanical Systems (MEMS) / N. Zhou, A. M. Agogino, K. S. Pister // In Proc. of the ASME Design Automation Conference.— 2002.

- [10] Evolutionary Synthesis of Microelectromechanical Systems (MEMS) Design / N. Zhou, B. Zhu, A. M. Agogino, K. S. Pister // In Proc. of ANNIE 2001: Intelligent Engineering System through Artificial Networks.— ASME Press, 2001.— P. 197—202.
- [11] Goel A. K. Kritik: An Early Case-Based Design System / A. K. Goel, S. Bhatta, E. Stroulia // Issues and Applications of Case-Based Reasoning in Design / ed. M. Maher, P. Pu.— Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.— 1997 —P. 87—132.
- [12] Особливості розроблення веб-систем для еволюційних обчислень / [Ткаченко С., Кривий Р., Теслюк В., Зелінський А.] // Вісник Національного університету «Львівська політехніка»: Комп'ютерні системи проектування. Теорія і практика.— Львів, 2011.— №711. — С. 22—26.
- [13] Теслюк В. М. Методологія автоматизованого проектування МЕМС на системному рівні / В. М. Теслюк, А. Я. Зелінський, А. І. Пукач // Зб. наук. пр. ІППМЕ ім. Г. Є. Пухова НАН України.— Київ, 2011.— Вип. 61.— С. 149—157.
- [14] Teslyuk V. Information support system analysis and synthesis of microelectromechanical systems / Vasyl Teslyuk, Andrii Pukach, Andriy Zelynskiy // Proc. of the III-d Intern. Conf. of Computer Science & Engineering, (CSE'2011).— Lviv.— P. 72 – 75.
- [15] Теслюк В.М. Застосування методу гілок та границь для зменшення потужності множини альтернативних рішень при розв'язанні задач синтезу МЕМС / В. М. Теслюк, А. Я. Зелінський, Х. Аль Шавабкех // Моделювання та інформаційні технології. Зб. наук. пр. ІППМЕ ім. Г. Є. Пухова НАН України.— Київ, 2009.— Вип. 50.— С.157—165.
- [16] Методологія автоматизованого проектування МЕМС на системному рівні / [Теслюк В. М., Денисюк П. Ю., Зелінський А. Я., Василюк Я. Р.] // Матеріали науково—методичної конференції „Сучасні проблеми телекомунікацій і підготовка фахівців в галузі телекомунікацій — 2011”. — 27 - 30 жовтня, Львів, 2011 р.— С. 86—89.
- [17] Барсуков В. С. Современные технологии безопасности / В. С. Барсуков, В. В. Водолазкий.— М. : “Нолидж”, 2000.— 496 с.
- [18] Терминология и варианты конфигурации кластера [Електронний ресурс].— Режим доступу: http://www.askit.ru/custom/sql2005_admin/m7/07_01_02_cluster_types.htm.
- [19] Гаврилова Т. А. Базы знаний интеллектуальных систем. Учебник / Т. А. Гаврилова, В. Ф. Хорошевский.— СПб.: Питер, 2000.— 384 с.
- [20] Искусственный интеллект. Кн. 2: Модели и методы: Справочник / Под ред. Э.В. Попова.— М.: Радио и связь, 1990.— 303 с.
- [21] Левин Р. Практическое введение в технологию искусственного интеллекта и экспертных систем с иллюстрациями на Бейсике / Р. Левин, Д. Дранг, Б. Эделсон.— М.: Финансы и статистика, 1990.— 239 с.
- [22] Нильсон Н. Принципы искусственного интеллекта / Н. Нильсон.— М.: Радио и связь. 1990.— 376 с.
- [23] Минский М. Фреймы для представления знаний / М. Минский.— М.: Мир, 1979.— 152 с.
- [24] XML [Електронний ресурс].— Режим доступу: <http://uk.wikipedia.org/wiki/XML>.
- [25] Денисюк П. Ю. Застосування XML-формату для опису конструкцій гідравлічних МЕМС / П. Ю. Денисюк // Зб. наук. пр. УАД Комп'ютерні технології друкарства.— Львів, 2007.— №17.— С. 93—99.
- [26] Yong Z. Layout Synthesis of Accelerometers / Z. Yong // Department of Electrical and Computer Engineering.— Carnegie Mellon University.— August, 1998.
- [27] Gupta V. Layout Synthesis of CMOS MEMS Accelerometers / V. Gupta, T. Mukherjee // Department of ECE.— Carnegie Mellon University.— Pittsburgh, PA-15213.— 2000.
- [28] Gupta V. Approaches to Synthesis of a CMOS Accelerometer / V. Gupta // Department of Electrical and Computer Engineering, Carnegie Mellon University.— Pittsburgh, Pennsylvania.— August, 2000.
- [29] AutoCAD [Електронний ресурс].— Режим доступу: <http://usa.autodesk.com/autocad/>.
- [30] DWG [Електронний ресурс].— Режим доступу: <http://en.wikipedia.org/wiki/dwg>.

РОЗДІЛ 4.

ОСОБЛИВОСТІ НАПОВНЕННЯ КОНСТРУКТОРСЬКОЇ БД СКЛАДОВИХ МЕМС

4.1. Класифікація складових МЕМС

Процес дослідження, розробки та моделювання МЕМС невід’ємно пов’язаний з процесами класифікації об’єктів проектування, які дають можливість глибше зрозуміти принципи їх роботи і особливості для значно ефективнішої організації процесу аналізу та синтезу МЕМС.

Будь-яка класифікація визначається критерієм класифікації. При класифікації МЕМС, таких критеріїв може бути декілька. Якщо в якості критерію класифікації вибрати тип елемента, що виконує перетворення енергії одного виду в інший, то всі складові МЕМС можна розділити на основні групи, які наведені на рис.4.1. Отже, до першої групи відносять мікродавачі, до другої групи відносять інтегральні пристрої, які виконують роботу, і такі пристрої називають мікроактюаторами [1]. До третьої групи відносять мікропристрої відображення, які призначені для відображення будь-яких змін в оточуючому середовищі, зрозуміло, що ці зміни попередньо мають бути зафіксовані мікродавачем.

Існує і окрема група мікропристроїв, яка не виконує перетворень енергії та переміщень, до них відносять зокрема мікролінзи, мікроканали, мікросопла, але є невід’ємною частиною МЕМС, зокрема: оптичних [2], біопристроїв [3], рідинних [4, 5], тощо.

Якість виконання функцій МЕМС, які не пов’язані енергетичними зв’язками з основною системою залежить від її енергетичного забезпечення. Тому, на сьогодні, надзвичайно активно розвивається область мікропристроїв генерації або накопичення енергії (мікроаккумулятори, мікрореактори, мікроджерела живлення на основі сонячної енергії, енергії вітру, енергії різних видів коливань тощо).

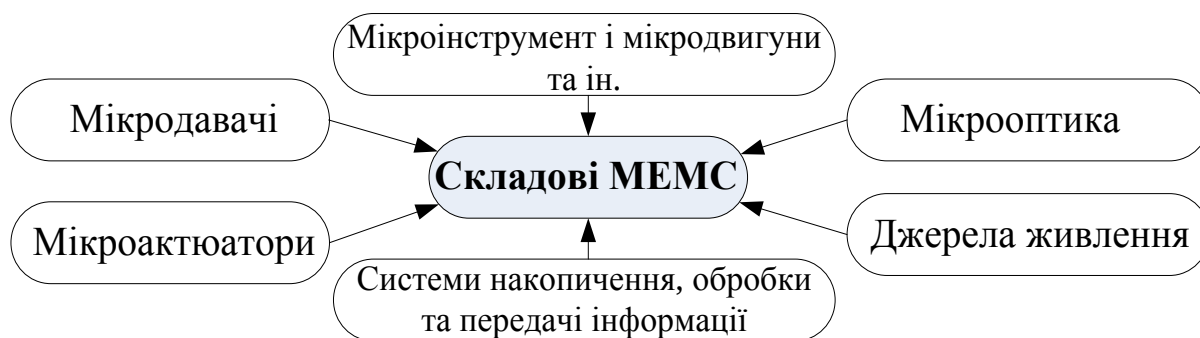


Рис.4.1. Класифікація складових MEMC за типом (функціональним призначенням) елементів

Відповідно до наведених типів елементів MEMC, усі вони можуть бути розміщені на одному кристалі або на декількох. В першому випадку маємо однокристальну MEMC, а в другому – багатокристальну (мультичіпову) (рис.4.2). Підхід до мультичіпового з’єднання елементів MEMC обумовлений тим, що дуже важко з допомогою однієї мікротехнології виготовити усі складові єдиної інтегральної мікросистеми. Тому, досить часто, окремі складові MEMC виготовляють за різними мікротехнологіями з їх подальшим розміщенням на мультичіповій платі.

За іншою класифікацією, усі MEMC можна розділити на такі дві великі групи: автономні та вбудовані (рис.4.3). До автономних можна віднести, для прикладу, мікророботи, мікролабораторії на “кристалі”, пристрої типу “розумний пил”, а до вбудованих – системи аварійного викидання подушок в автомобілі, системи випорскування чорнила в принтерах та інші.

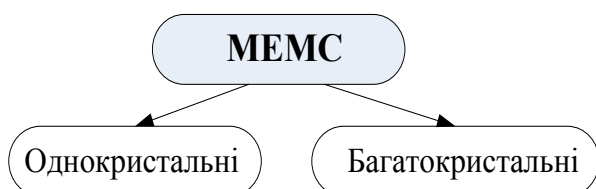


Рис.4.2. Поділ MEMC на однокристальні та багатокристальні

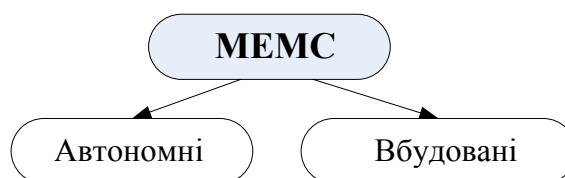


Рис.4.3. Поділ MEMC на автономні та вбудовані

Якщо в якості критерію класифікації вибрати технологію виготовлення MEMC, що є надзвичайно актуальним питання для області даних інтегральних пристроїв, то розрізняють наступні (рис.4.4): технологія об'ємної мікроопрацювання (ТОмО); технологія поверхневої мікроопрацювання (ТПмО); LIGA технологія; SIGA технологія; MUMPs технологія; технологія виготовлення полімерних елементів MEMC (ТВПЕ MEMC); тривимірна мікротехнологія та ін. На даний момент, у світі, нараховують близько двадцяти технологій, які використовують при виготовленні MEMC, і з кожним роком їх кількість лише зростає.

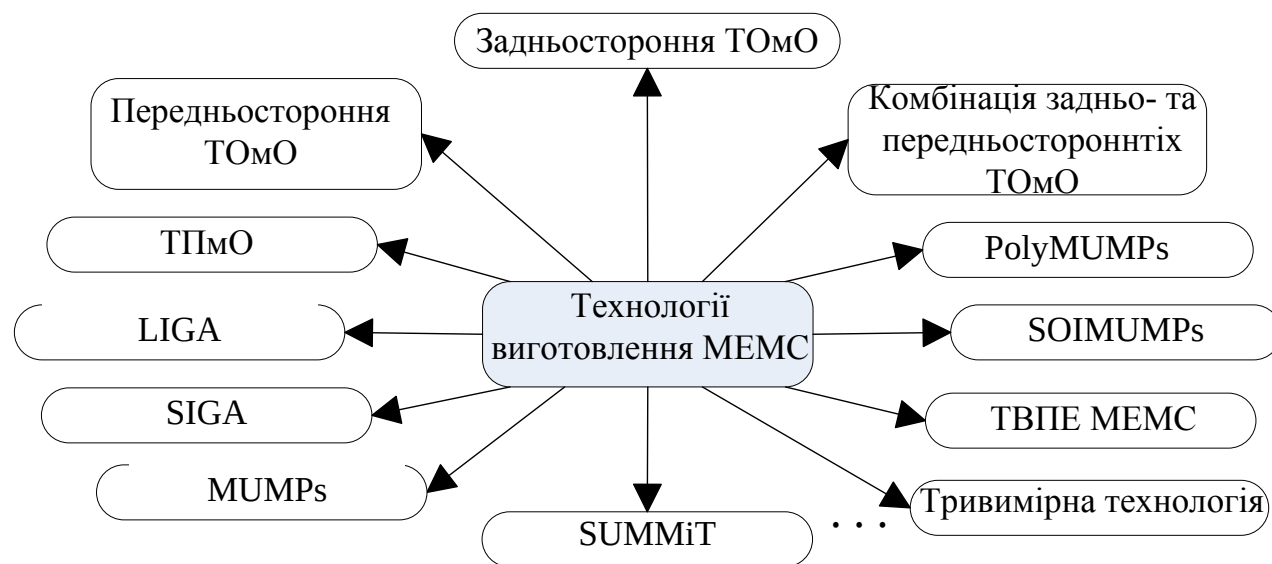


Рис.4.4. Класифікація MEMC за технологією виготовлення

Класифікація мікродавачів. Усі мікродавачі можна поділити на активні та пасивні; контактні та неконтактні та інші.

Активні мікродавачі в процесі роботи (вимірювання) потребують впливу на чутливий елемент енергії зовнішнього середовища, а пасивні мікродавачі генерують вихідний електричний сигнал без впливу енергії зовнішнього середовища.

Контактні мікродавачі для роботи (вимірювання) вимагають безпосереднього контакту з зовнішнім середовищем, а неконтактні – не потребують контакту з зовнішнім середовищем.

Підсумовуючи вищесказане можна навести класифікацію мікродавачів згідно наступних критеріїв: за призначенням використання мікродавачів (мікродавач тиску, вологості, сили і т.д.) (рис. 4.5); за вихідним контрольованим параметром (ємнісний мікродавач, п'єзореzystивний тощо) (рис.4.6); згідно фізичних ефектів, що використовуються в процесі роботи мікродавачів (мікродавач на основі п'єзоефекту, мікродавач на основі магнітострикційного ефекту тощо) (рис.4.7); згідно параметрів, для вимірювання яких вони призначені (рис.4.8); згідно технологій їх виготовлення (гібридно-плівкові, твердотільні біполярні, МОН – структури та технології виготовлення МЕМС (див.рис.4.4)).

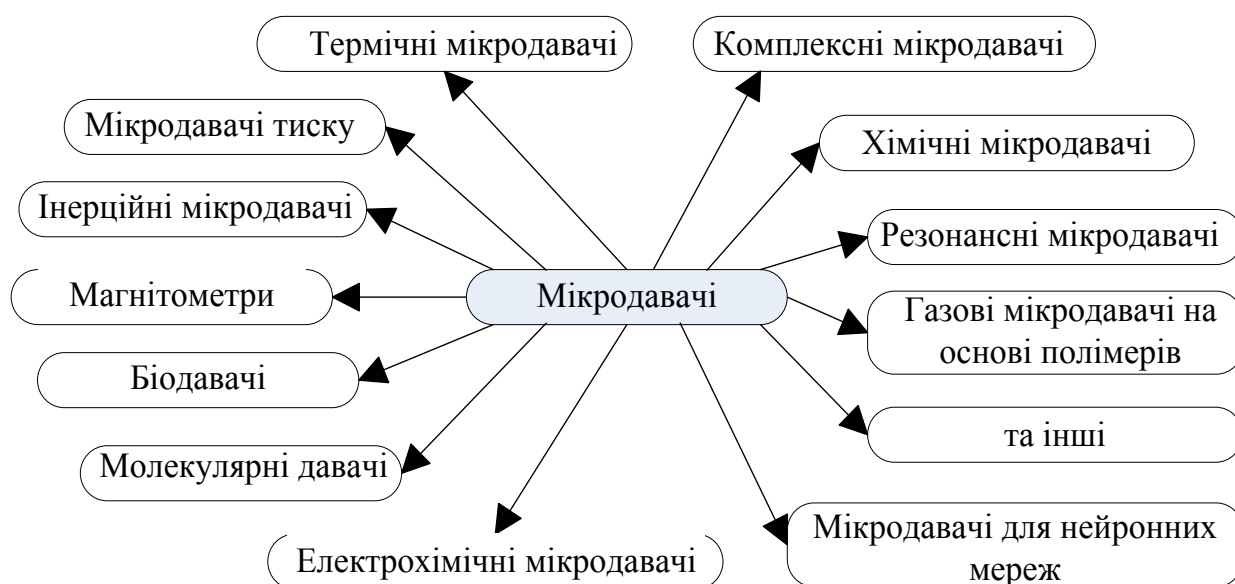


Рис.4.5. Класифікація мікродавачів за призначенням використання

Слід зауважити, що можна додати ще багато різних критеріїв класифікації, зокрема: за швидкістю реакції, частотним діапазоном, лінійністю, вартістю, розмірами та масою, за точністю вимірювання давачами контрольованої величини (для прикладу, високоточні (до 0,1 %), точні (0,1 – 0,3 %), середньої (0,3 – 1,5 %) та низької (більше 1,5 %) точності); за чутливістю (малої та високої чутливості); за діапазоном виміру вхідної величини (малих, середніх та великих значень). Такий поділ умовно вже існує в цілому ряді випадків. Так, низькими вважаються температури нижче 273 °K, середніми 273

– 1300 °K , високими – вище 1300 °K . Піддіпазони зміни об’ємних витрат рідин 0 – $15 \cdot 10^{-4}$ м³/с (малі), $15 \cdot 10^{-4}$ – 0,5 м³/с (середні) і від 0,5 м³/с та вище (великі) і інші. Ряд вищенаведених критеріїв класифікації мікродавачів можна віднести і до класифікації мікроактюаторів.

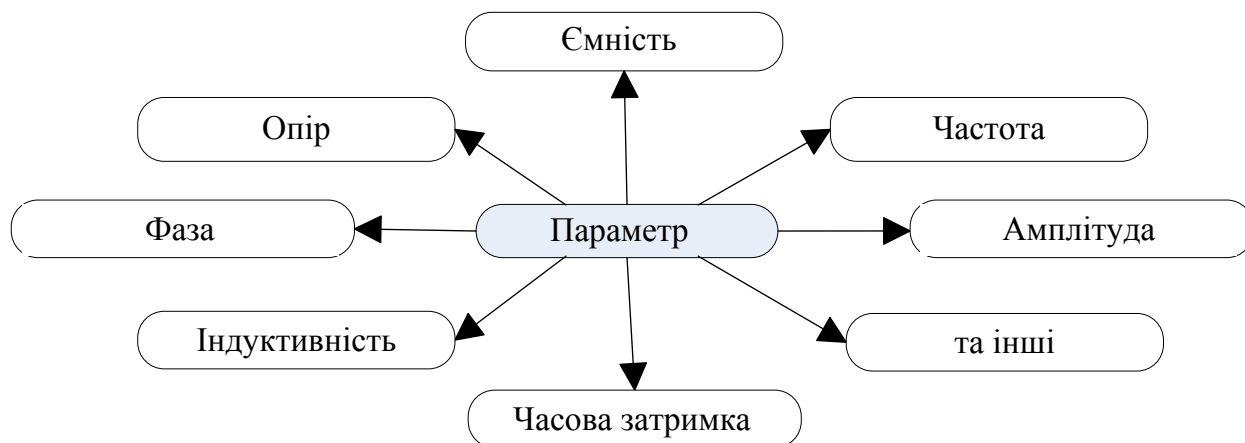


Рис.4.6. Класифікація мікродавачів за вихідним контрольованим параметром

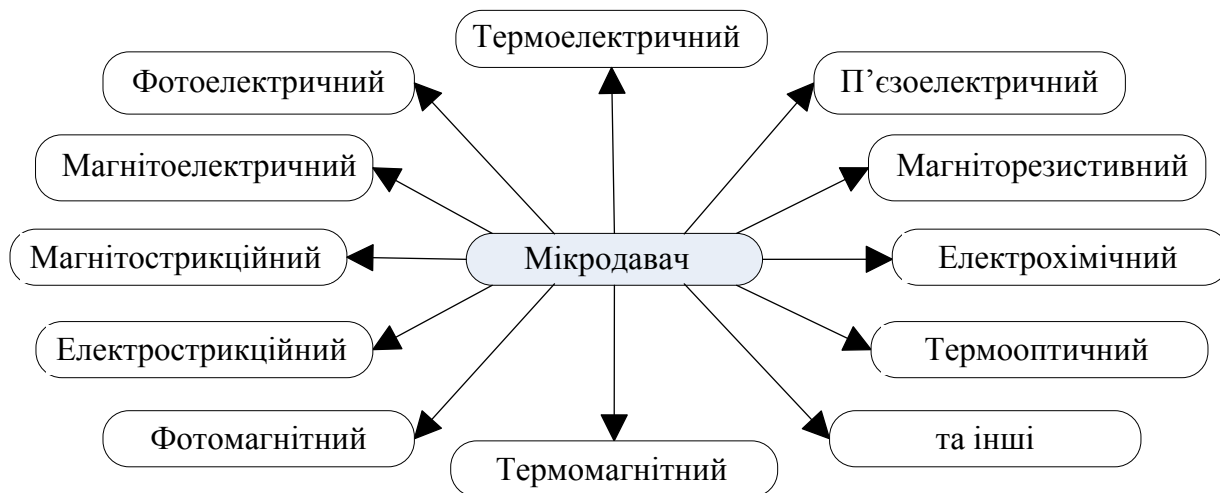


Рис.4.7. Класифікація мікродавачів за ефектами

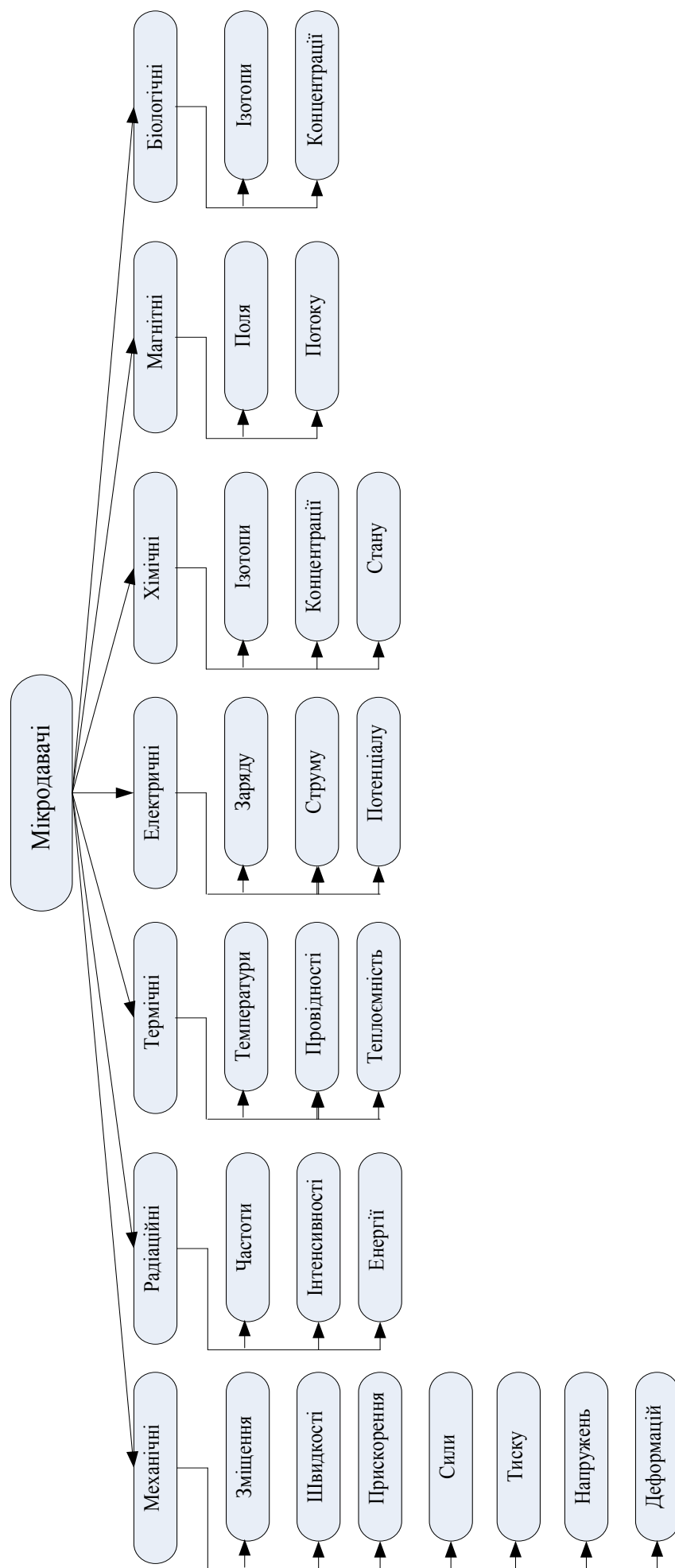


Рис.4.8. Класифікація мікродавачів по параметрах, для вимірювання яких вони призначені

Класифікація мікроактюаторів. Як було зазначено вище, до кожного актюатора необхідно підвести енергію (вхідним параметром є електричний сигнал, а саме: напруга, струм, частота, фаза та ін.) щоб він виконав переміщення чи іншу функцію (енергія світла, нагрівання, відображення та ін.). Тому, згідно вхідної енергії, можна навести наступну класифікацію мікроактюаторів за використанням різних видів енергії (рис.4.9) [6]: енергії тиску; електричної; термічної; оптичної; акустичної та інших.

Кожна енергія може згенерувати різні сили, які виникають в процесі роботи актюатора при виконанні переміщення. Так, для прикладу, електрична енергія, дає змогу генерувати два типи сил: молекулярні сили і сили електричного та магнітного полів. В залежності від сили, розрізняють різні види мікроактюаторів. У випадку молекулярних сил ми отримаємо три види мікроактюаторів: магнітострикційний, п'єзоелектричний та електрореологічний, а для випадку сил електричного та магнітного полів – електростатичний та електромагнітний мікроактюатори.

Термічна енергія дає змогу згенерувати також два види сил, а саме: сили, які виникають при термічному розширенні речовини (біметалічний мікроактюатор, термічний мікроактюатор) та сили на основі ефекту форми (мікроактюатор на основі ефекту форми).

Аналогічним чином відбувається поділ на згенеровані сили з допомогою рідинного і хімічного видів енергій, а сили, в свою чергу, обумовлюють появу відповідних видів мікроактюаторів. Так хімічна енергія може генерувати електролітичні сили (електрохімічний пристрій) і сили вибуху (піротехнічний пристрій). Енергія рідини генерує гідравлічні та пневматичні сили. Відповідно є мікроактюатори гідравлічні, пристрої високого і низького тиску.

Вищенаведену класифікацію можна розширити для акустичної, оптичної та інших видів енергій. Можна класифікувати мікроактюатори за видом переміщення, яке вони виконують (лінійне, обертове, переміщення в одновимірному напрямку, двовимірному та тривимірному), згідно кількості енергії, яку підводять до мікроактюаторів та інші.

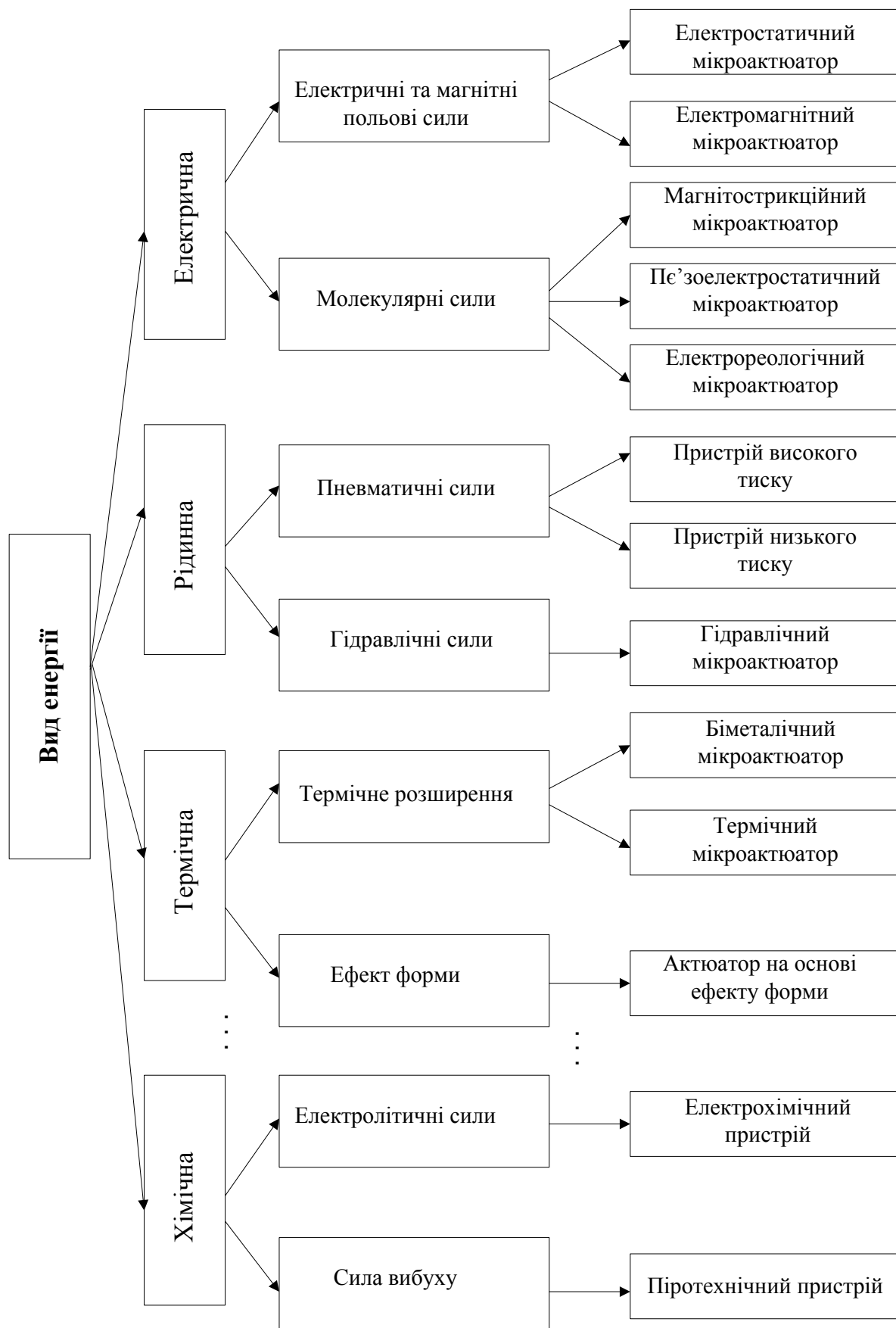


Рис.4.9. Класифікація мікроактюаторів згідно вхідної енергії

Разом з тим, при порівнянні необхідно розглянути й інші характеристики та параметри мікроактюаторів.

В загальному випадку теплові мікроактюатори мають малий час реакції (повільніші від інших, в межах кількох мілісекунд) і високе енергоспоживання (в межах десятків мВт). Для порівняння, електромагнітний мікроактюатор може бути набагато швидшим (мікросекунди) і споживати менше енергії, особливо електростатичний мікроактюатор.

Розробка теплового мікроактюатора часто вимагає використовувати багат шарову конструкцію з матеріалів, які мають різні фізичні властивості. Це часто ускладнює розробку та технологічне виготовлення мікропристроїв даного типу, зокрема завдяки залишковим напруженням в тонких плівках.

П'єзоелектричні мікроактюатори та мікродавачі вимагають використання додаткових матеріалів (плівок), що ускладнює технологію їх виготовлення. На відміну, електростатичний мікроактюатор використовує стандартні матеріали та технологічні операції для виготовлення інтегральних схем, що є безумовною перевагою мікропристроїв електростатичного типу. Перевагами п'єзоелектричних і термічних мікроактюаторів є те, що вони дають можливість згенерувати досить велику силу в мікросвіті.

Генерація сили в електростатичному мікроактюаторі вимагає подачі напруги до пластин конденсатора через ізолюючий шар діелектричної плівки, що обумовлює майже нульові втрати електричної енергії. Магнітні мікроактюатори вимагають при виготовленні магнітних матеріалів, які не є стандартними в технології виробництві інтегральних схем і досить часто вимагають деяких видів ручних робіт при їх складанні.

Класифікація систем опрацювання, збереження та передачі даних. Системи опрацювання, збереження та передачі даних (СОЗПД) можна розділити на три основні групи, зокрема (рис.4.10):

- системи програмної опрацювання інформації (СПОІ);
- системи апаратної опрацювання інформації (САОІ);
- гібридні системи опрацювання інформації (ГСОІ).

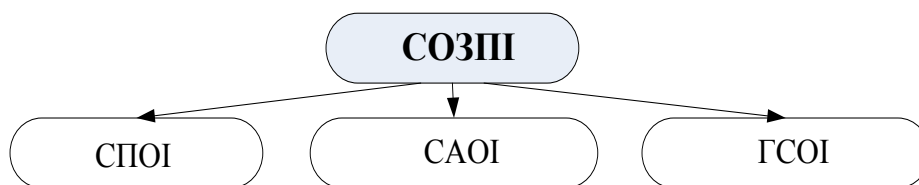


Рис.4.10. Класифікація систем опрацювання інформації

4.2. Особливості розроблення базових складових МЕМС

Особливе, або з впевненістю можна стверджувати, що лідируюче становище займають МЕМС з електростатичним принципом дії [7, 8]. Оскільки вони мають ряд переваг над інтегральними МЕМС з іншими принципами роботи, а саме: простота виготовлення мікропристроїв даного типу, як правило, використовуються технології виготовлення, які є сумісні з технологіями інтегральних схем, або технології, що базуються на них; використовують матеріали, які добре відомі для технології виготовлення ІС; величина сили, яка виникає між обкладками мікропристрою, має досить велике значення; відомий математичний апарат для визначення величини сили на етапах аналізу та синтезу таких інтегральних пристроїв та ін.

В даний час розроблено гіроскопи [9], давачі [10 – 12] акселерометри [13 – 15], актюатори [16], мікропомпи [17 – 20], резонатори [21, 22] та інші інтегральні пристрої МЕМС [23 – 24], які використовують електростатичну силу, зміну ємності тощо. Саме перший в світі кремнієвий давач, який був продемонстрований в 1961 році, та кремнієвий акселерометр (1970) були електростатичного типу. На даний час, багато робіт присвячено розробці актюаторів електростатичного типу, але досить мало приділяється уваги мікродавачам.

Особливе значення серед інтегральних пристроїв займають гребінчасті конструкції або конструкції, подібні до них. На базі цієї структури розроблено давачі [25], акселерометри та гіроскопи [26, 27]. Наявні математичні моделі володіють низькою точністю та не враховують специфіку інформаційних

технологій автоматизованого багаторівневого ієрархічного розроблення МЕМС.

Приклад конструкції інтегрального давача ємнісного типу наведено на рис.4.11 та рис. 4.13. Виготовлення відповідної конструкції давача з допомогою мікроелектронних технологій не представляє складності і детально описано в літературі [28 – 30]. Виготовляють даний пристрій на пластині з діелектрика чи напівпровідника, а в якості провідника використовують метал або сильно леговану область напівпровідника. В загальному випадку даний давач представляє конденсатор, ємність якого залежить не лише від конструкції, але і від діелектричної проникності речовини, яка розміщена між та над його обкладками [31]. Тому контроль фізичних властивостей речовини проводиться саме на основі цього параметра.

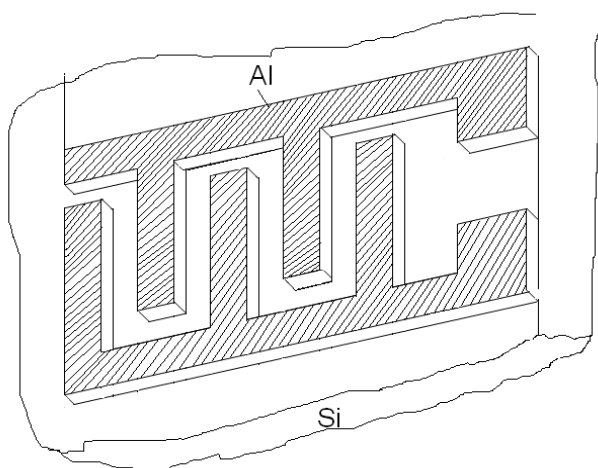


Рис.4.11. Структура інтегрального ємнісного давача зустрічно-стрижневої конструкції

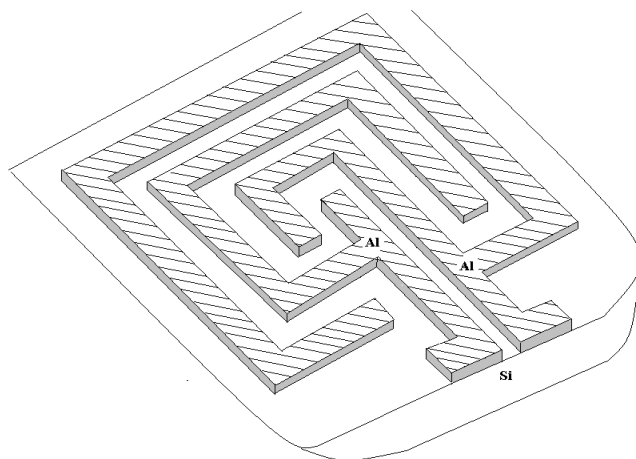


Рис. 4.12. Конструкція інтегрального ємнісного давача типу еквідистанційної спіралі Архімеда

Побудуємо вигляд зверху для конструкції давача, яка наведена на рис.4.11 і зробимо переріз А-А, що зображено на рис.4.14. В результаті отримаємо область моделювання, яка зображена на рис.4.15.

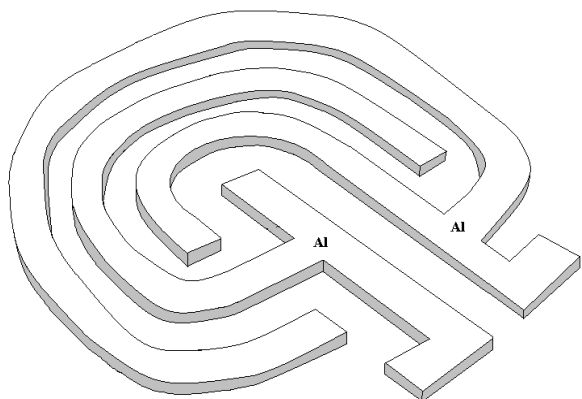


Рис. 4.13. Конструкція інтегрального ємнісного датчика типу еквідистанційної спіралі Архімеда

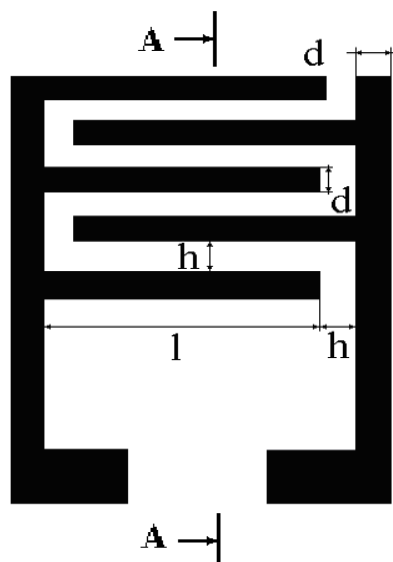


Рис.4.14. Конструкція інтегрального електростатичного ємнісного датчика (зустрічно-стрижнева конструкція, вид зверху)

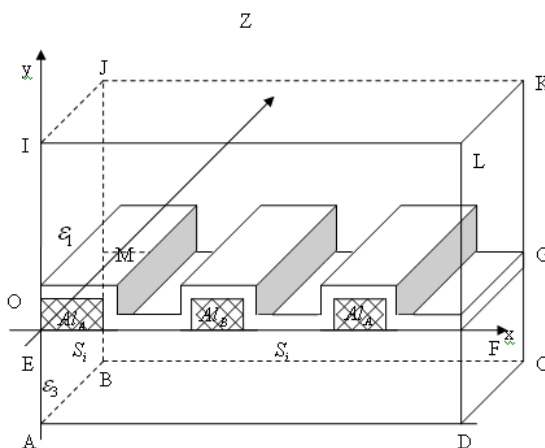


Рис.4.15. Тривимірний простір моделювання

Математична модель, яка передбачає визначення розподілу потенціалу в тривимірному випадку для однорідного середовища, вимагає розв'язку системи диференціальних рівнянь в частинних похідних [31, 32]:

$$\frac{\partial^2 \varphi_1}{\partial x^2} + \frac{\partial^2 \varphi_1}{\partial y^2} + \frac{\partial^2 \varphi_1}{\partial z^2} = 0, \text{ при визначенні потенціалу } \varphi_1 \text{ в середовищі з}$$

$$\text{діелектричною проникністю } \varepsilon_1, \quad \frac{\partial^2 \varphi_2}{\partial x^2} + \frac{\partial^2 \varphi_2}{\partial y^2} + \frac{\partial^2 \varphi_2}{\partial z^2} = 0, \text{ при визначенні}$$

потенціалу φ_2 в середовищі з діелектричною проникністю ε_2 ,

$$\frac{\partial^2 \varphi_3}{\partial x^2} + \frac{\partial^2 \varphi_3}{\partial y^2} + \frac{\partial^2 \varphi_3}{\partial z^2} = 0, \text{ при визначенні потенціалу } \varphi_3 \text{ в середовищі з}$$

діелектричною проникністю ε_3 , де $\varphi_1, \varphi_2, \varphi_3$ - величина електростатичного потенціалу в середовищах 1, 2 та 3; x, y, z – просторові координати.

До даної системи диференціальних рівнянь необхідно додати крайові

$$\text{умови: } \frac{\partial \varphi_1}{\partial n} = \frac{\partial \varphi_2}{\partial n} = \frac{\partial \varphi_3}{\partial n} = 0, \quad \text{при } n \in S, \varphi_2 = \varphi_3 = \varphi_0, \quad \text{при}$$

$$\varphi_{2,3} \in Al_A, \varphi_2 = \varphi_3 = -\varphi_0, \quad \text{при } \varphi_{2,3} \in Al_B,$$

$$\varepsilon_1 \left(\frac{\partial \varphi_1}{\partial m} \right) \varepsilon_1 / \varepsilon_2 = \varepsilon_2 \left(\frac{\partial \varphi_2}{\partial m} \right) \varepsilon_1 / \varepsilon_2, \varepsilon_2 \left(\frac{\partial \varphi_2}{\partial m} \right) = \varepsilon_3 \left(\frac{\partial \varphi_3}{\partial m} \right), \quad \varphi_1 \varepsilon_1 / \varepsilon_2 = \varphi_2 \varepsilon_1 / \varepsilon_2,$$

$$\varphi_2 \varepsilon_2 / \varepsilon_3 = \varphi_3 \varepsilon_2 / \varepsilon_3, \text{ де } n - \text{ нормаль до бічної поверхні області моделювання}$$

АВСВІІLK; $\varepsilon_1, \varepsilon_2, \varepsilon_3$ - діелектрична проникність середовища 1, 2 та 3, відповідно; m – нормаль до границі розділу двох середовищ.

Для визначення розподілу напруженості електричного поля в тривимірному випадку, необхідно розв'язати наступні рівняння в області моделювання [31]:

$$\vec{i} \frac{\partial \varphi_1}{\partial x} + \vec{j} \frac{\partial \varphi_1}{\partial y} + \vec{k} \frac{\partial \varphi_1}{\partial z} = -\vec{E}, \quad \vec{i} \frac{\partial \varphi_2}{\partial x} + \vec{j} \frac{\partial \varphi_2}{\partial y} + \vec{k} \frac{\partial \varphi_2}{\partial z} = -\vec{E}, \quad \vec{i} \frac{\partial \varphi_3}{\partial x} + \vec{j} \frac{\partial \varphi_3}{\partial y} + \vec{k} \frac{\partial \varphi_3}{\partial z} = -\vec{E}.$$

Розподіл вектора електричної індукції визначається з виразу [33]: $D = \varepsilon_1 E$, в області з діелектричною провідністю ε_1 , $D = \varepsilon_2 E$, в області з діелектричною провідністю ε_2 , $D = \varepsilon_3 E$, в області з діелектричною провідністю ε_3 , а на границі розділу двох середовищ необхідно врахувати

наступні краєві умови: $D_{\varepsilon_1/\varepsilon_2} = \left(\frac{\varepsilon_1 + \varepsilon_2}{2}\right) E_{\varepsilon_1/\varepsilon_2}$, $D_{\varepsilon_2/\varepsilon_3} = \left(\frac{\varepsilon_2 + \varepsilon_3}{2}\right) E_{\varepsilon_2/\varepsilon_3}$, де $E_{\varepsilon_1/\varepsilon_2}$

– значення напруженості електричного поля на границі розділу середовищ 2 та 3; $E_{\varepsilon_1/\varepsilon_2}$ – значення напруженості електричного поля на границі розділу

середовищ 1 та 2; $D_{\varepsilon_1/\varepsilon_2}$ – значення вектора електричної індукції на границі

розділу середовищ 1 та 2; $D_{\varepsilon_2/\varepsilon_3}$ – значення вектора електричної індукції на

границі розділу середовищ 2 та 3.

Згідно класичного визначення ємності [34] вона визначається як відношення заряду q до різниці потенціалів U_0 , тобто: $C = q/U_0$, або відношенню енергії до квадрату різниці потенціалів:

$$C = 2W/U_0^2, \quad (4.1)$$

де W - енергія.

В нашому випадку, якщо ми досліджуємо зустрічно-стрижневу конструкцію ємнісного давача, то енергія на одиницю довжини визначається з виразу [34]

$$W = \frac{1}{2} \int_S \bar{D}(x,y) \bar{E}(x,y) dS, \quad (4.2)$$

де S – площа, яка обмежена контуром l ; $D(x,y)$ - розподіл вектора електричної індукції в області моделювання; $E(x,y)$ - розподіл напруженості електричного поля в області моделювання.

Застосовуючи вище наведені формули до задачі визначення ємності давача зустрічно-стрижневої конструкції, необхідно визначити інтеграл (4.2). Для цього застосуємо стандартні формули квадратур [35], а попередньо необхідно модифікувати інтеграл, щоб звести його до нормованої форми. В кінцевому випадку отримаємо

$$W = \int_{-1}^1 \int_{-1}^1 D(x,y)E(x,y)dx dy \approx \sum_{j=1}^n \sum_{i=1}^n D(x_i,y_j)E(x_i,y_j)\omega_i\omega_j, \quad \text{де}$$

$D(x_i,y_j)$ та $E(x_i,y_j)$ - значення вектора електричної індукції і напруженості електричного поля в точці з координатами x_i,y_j ; ω_i,ω_j - вагові коефіцієнти.

Маючи значення енергії одиниці довжини та різниці потенціалів, порахуємо ємність одиниці довжини за формулою (4.2). Отримане значення ємності слід перемножити на довжину стрижня. Кінцева формула для визначення ємності між двома стрижнями буде наступною:

$$C = \frac{2L}{U_0^2} \sum_{j=1}^n \sum_{i=1}^n D(x_i,y_j)E(x_i,y_j)\omega_i\omega_j, \quad \text{де } L - \text{довжина стрижня.}$$

У випадку не зустрічно-стрижневої конструкції мікродавача слід використовувати формулу для визначення ємності (4.1), а енергія визначається з виразу:

$$W = \int_{-1}^1 \int_{-1}^1 \int_{-1}^1 D(x,y,z)E(x,y,z)dx dy dz \approx \sum_{j=1}^n \sum_{i=1}^n \sum_{k=1}^n D(x_i,y_j,z_k)E(x_i,y_j,z_k)\omega_i\omega_j\omega_k,$$

де $D(x_i,y_j,z_k), E(x_i,y_j,z_k)$ - значення вектора електричної індукції і напруженості електричного поля в точці з координатами x_i,y_j,z_k ; $\omega_i,\omega_j,\omega_k$ - вагові коефіцієнти.

Приклад базової конструкції актюатора електростатичного типу наведено на рис. 4.16. Вона включає дві паралельні пластини, до яких прикладено напругу. Між пластинами виникає сила, яка прагне наблизити ці дві пластини одна до одної. У випадку жорсткого защемлення першої пластини та реалізації іншої в формі тонкої пластини, жорстко защемленої по її контуру чи защемленої з однієї сторони для прямокутної форми пластини, то друга пластина буде деформуватися під дією електростатичної сили. На основі цієї конструкції розроблено багато різних модифікацій, але принцип роботи та основні елементи конструкції залишаються незмінними.

Визначення вихідних параметрів актюатора електростатичного типу передбачає послідовне розв'язання двох задач. Перша задача призначена для

визначення сили, яка виникає між пластинами, а друга задача – зміщення елементів вільної, в своєму русі, пластини (тобто тої пластини, яка має змогу деформуватися повністю або хоча б її частина).

Особливістю більшості математичних моделей на компонентному рівні проектування для електростатичних актюаторів є те, що на першому етапі проводять обчислення сили за спрощеними виразами з наступним визначенням деформацій в рухомому або пружному елементі електростатичного інтегрального пристрою. Другий етап передбачає точніше визначення ємності системи та сили з врахуванням деформацій конструкції і наступним уточненням вихідних параметрів про деформації та зміщення елементів мікроактюатора на базі уточнених значень діючої сили. Часом виникає потреба в повторенні декількох разів даної операції для досягнення необхідної точності вихідних результатів.

Електростатичні мікроактюатори використовуються для реалізації мікроперемикачів, приводів для повороту мікродзеркала, мікродвигунів тощо. Приклад мікродвигуна електростатичного типу наведено на рис.4.17. Математична модель його аналогічна до вищеописаної.

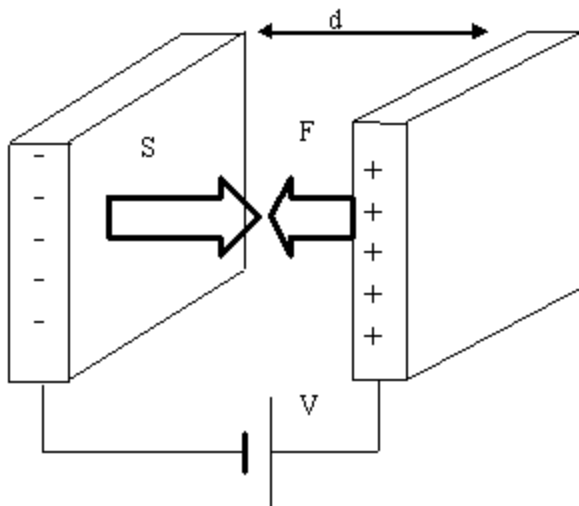


Рис.4.16. Конструкція ємнісного актюатора

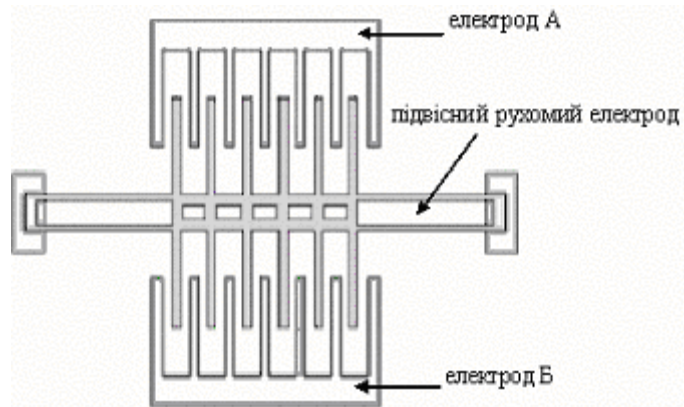


Рис.4.17. Конструкція гребінчастого електростатичного мікродвигуна

Типова конструкція акселерометра ємнісного типу складається із рухомої сейсмічної маси, яка прикріплена на балках до підкладки та двох обкладок. Між

обкладками та сейсмічною масою розміщуються електроди, які фіксують зміну ємності при зміщенні сейсмічної маси. Варто зауважити, що при такій побудові конструкції зміщення будуть відбуватись лише вздовж однієї осі.

В ємнісних давачах прискорення тобто в акселерометрах, основа приладу та маса відділяються одна від одної і вимірюється ємність або заряд між ними. При зменшенні відстані між обкладками ємність зростає і електричний струм тече в напрямку до давача, і навпаки – при зростанні відстані – струм тече у зворотньому напрямку. Давач перетворює прискорення сейсмічної маси в електричний струм, заряд або напругу. Чутливий елемент давача для вимірювання прискорення виготовлений, як правило, з кремнію, полікремнію або скла. Ємнісний чутливий елемент вимірює прискорення в додатному і від’ємному напрямках і є чутливим до статичного прискорення (вектор земного тяжіння) та вібрацій [36].

Ємнісні давачі прискорення це пристрої, які мають декілька переваг над давачами прискорення інших типів, а саме: високу чутливість; низький рівень шумів та дрейфу струму; низька чутливість до температури; мале значення розсіювання потужності.

Більше того, додаткові конденсатори можуть бути вбудовані і використані при доробці електричної схеми компенсації кільцевого зворотнього зв’язку чи для тестових цілей.

Недоліком ємнісного давача прискорення є значне ускладнення електронної схеми. Електронна схема спряження має бути розташована якомога ближче до давача, щоб зменшити паразитні ємності. Для того, щоб запобігти електромагнітним завадам, на пристрій необхідно встановити відповідне екранування [37].

Приклади інших конструкцій давачів прискорення ємнісного типу наведено на рис.4.18 та рис. 4.19 [38].

Математична модель для базових конструкцій акселерометрів на елементному рівні розроблення, в деякій мірі, є зворотною до моделі електростатичного актюатора. Вона також передбачає послідовне вирішення двох задач. Перша задача призначена для визначення деформацій в конструкції

акселерометра, а друга – ємності чи її зміни.

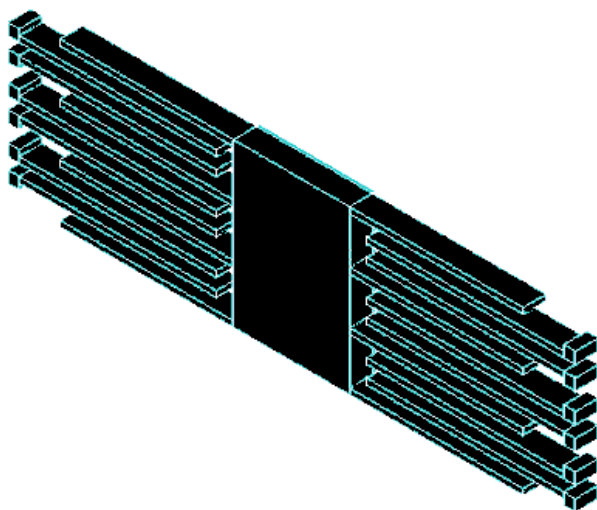


Рис. 4.18. Конструкція гребінчастого давача прискорення

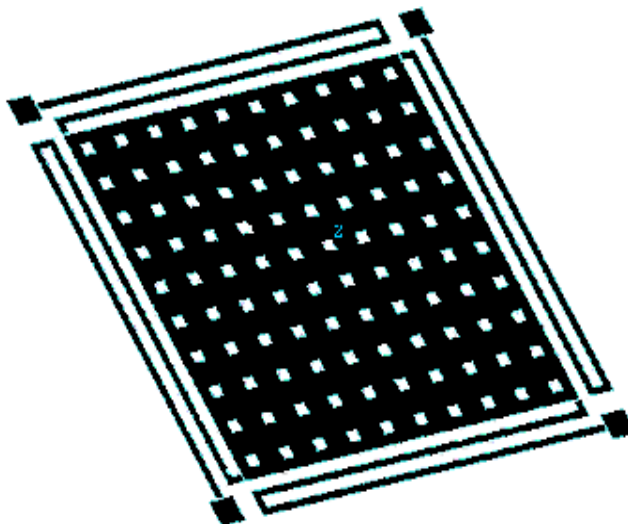


Рис. 4.19. Конструкція ємнісного акселерометра

Вирішення першої задачі по визначенню деформацій та напружень [39] в конструкції мікродавача прискорення проводиться з допомогою методу скінченних елементів. Основні математичні співвідношення наведені в роботі [40].

В процесі виготовлення електростатичних елементів MEMC використовують технології поверхневої та об'ємної мікрообробки і MUMPs – технологія.

Особливе значення серед MEMC посідають інтегральні пристрої на базі магнітних фізичних процесів та законів [41, 42], які володіють рядом переваг у порівнянні з електростатичними та іншими принципами роботи елементів мікросистем, зокрема: в мікромашинах, фрикційна сила [7] – недосяжна для інших видів енергії; магнітне поле надає можливість віддаленого керованого руху через магнітний ефект поля без будь-якого фізичного контакту; мікродвигуни, які широко використовуються, в основному, приводяться в дію з допомогою магнітного поля та обертового моменту.

Наведені переваги, в даний час, обумовлюють широке використання магнітних принципів дії при розробці нових мікродавачів, мікроактюаторів та

мікросистем [43], які потім використовуються в оптиці [44], мікрогідролічних пристроях [45], біологічних та медичних науках [46, 47], області передачі даних [48] та ін [49]. Для виготовлення електромагнітних елементів MEMS найчастіше використовують LIGA –технології, MUMPs – технологію та мікротехнології з використанням жертвовного шару.

Електромагнітний давач призначений для вимірювання величини магнітного поля. Разом з тим, цей інтегральний пристрій може використовуватися в оптичних мікропристроях, в системах для вимірювання електричних коливальних процесів тощо.

Конструкція актюатора включає тонку діелектричну пластину, як правило з окислу кремнію чи нітриду кремнію, на якій розміщено контур зі струмом та п'єзореzystори (рис.4.20).

Принцип роботи мікродавача полягає в наступному. На рамку зі струмом в магнітному полі діють сили, які прикладені до протилежних частин пластини і направлені в протилежних напрямках (рис.4.21). Це призводить до деформації тонкої пластини. В місцях виникнення найбільших напружень розміщені п'єзореzystори, які змінюють власний опір в залежності від змін напружень в конструкції п'єзореzystорів. П'єзореzystори з'єднані по схемі містка (рис.4.22). Вихідним параметром мікродавача є напруга, по змінах якої в залежності від змін п'єзоопорів і контролюють величину магнітного поля, в яке поміщений інтегральний пристрій.

Для ефективного аналізу та синтезу базових електромагнітних пристроїв необхідно розробити їх математичні моделі. З принципу роботи електромагнітного давача слідує, що необхідно послідовно розв'язати наступні задачі [50]: визначити сили, які діють на інтегральну рамку з контуром у формі тонкої пластини; визначити розподіл переміщень та напружень в тонкій пластині під дією прикладених сил; визначити зміни п'єзореzystивних опорів; визначити зміну вихідної напруги.

Розробка LIGA технології дала змогу проектувальникам інтегральних мікросистем використовувати феромагнітні матеріали (нікель) в процесі

виготовлення МЕМС, що послужило поштовхом для розвитку та використання в мікротехнологіях магнітних актюаторів [51, 52].

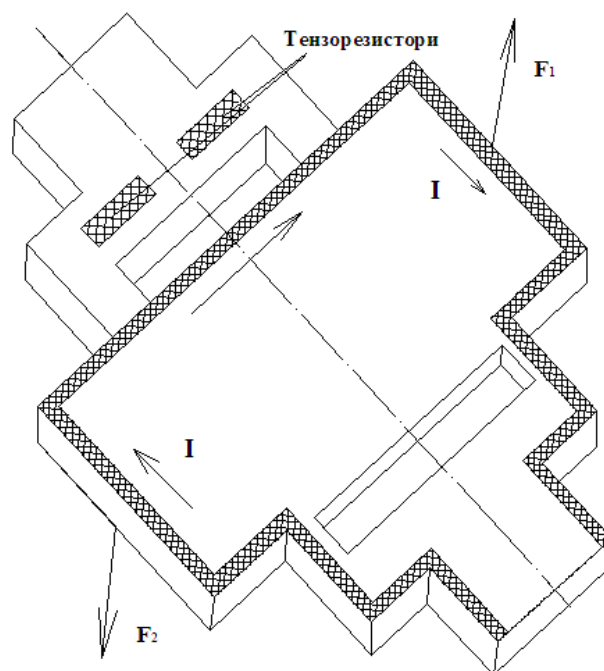


Рис.4.20. Конструкція електромагнітного давача

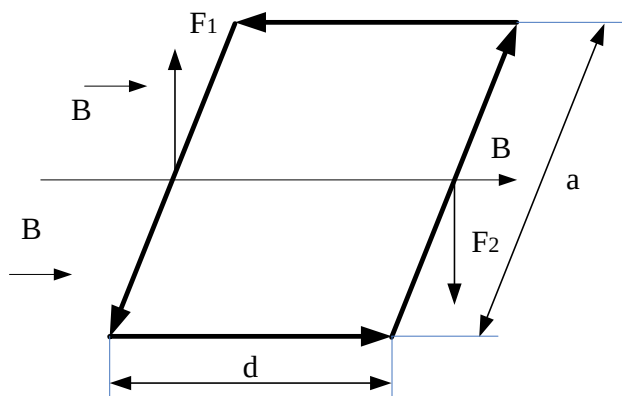


Рис.4.21. Рамка зі струмом в однорідному магнітному полі

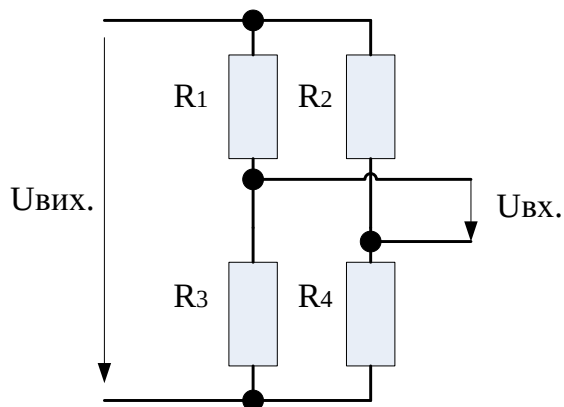


Рис.4.22. Місткова схема підключення п'єзорезисторів мікродавача

Приклад базової конструкції електромагнітного актюатора зображено на рис. 4.23. Вона включає балку на якій розміщено пермалоеву пластину. У випадку відсутності зовнішнього магнітного поля пермалоева пластина з балкою знаходяться в стані спокою (рис. 4.23а), а при прикладанні зовнішнього

магнітного поля по нормалі до площини структурної пластини, всередині пермалоєвої ділянки пластини виникає вектор намагніченості M , який взаємодіє з зовнішнім магнітним полем [53] при цьому генерує обертовий момент (M_{mag}) і силу, яка примушує згинатись вільний кінець балки (рис. 4.23б).

В процесі побудови математичної моделі пермалоєвий матеріал розглядається як матеріал, який володіє постійним плоскопаралельним вектором намагніченості $M_{нас}$. При впливі на пермалоєву пластину зовнішнього магнітного поля, відбувається генерація двох складових сили. Перша F_1 , яка діє на верхню грань пермалоєвої пластини та F_2 , яка діє на нижню грань. Для визначення величин сил використовують наступні вирази [31]: $F_1 = M_s z K H_1$, $F_2 = M_s z K H_2$, де H_1 і H_2 - напруженість магнітного поля на верхній та нижній гранях пластини (в наведеній конфігурації $H_1 < H_2$); z , K - ширина та товщина балки; M_s - намагніченість насичення.

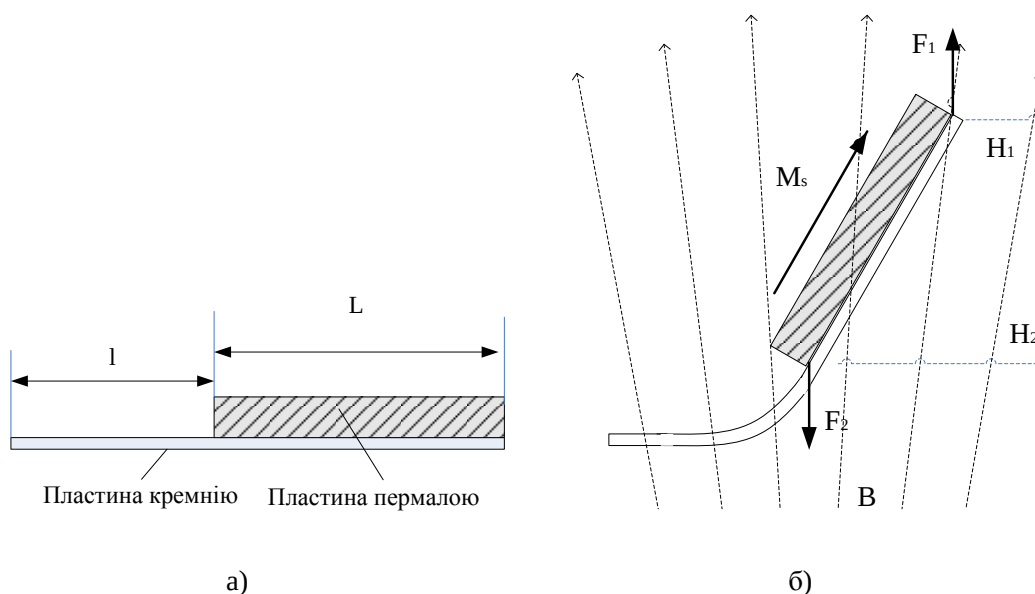


Рис.4.23. Конструкція електромагнітного актюатора

При зменшенні розмірів інтегральних пристроїв одним з напрямів, який на сьогодні надзвичайно активно розвивається, є розроблення та виготовлення теплових актюаторів та давачів. Аналіз таких пристроїв дає змогу стверджувати, що теплові актюатори можуть створювати порівняно великі сили, але немає конструкції, яка б дозволяла це зробити з позиції ефективного

використання енергії. Результат стає кращим при збільшенні різниці між коефіцієнтами теплового розширення матеріалу робочого елемента α і при більшому перепаді температури ΔT , однак коефіцієнт корисної дії (ККД), що досягається при цьому, все одно залишається малим. Гази і рідини мають набагато більший коефіцієнт теплового розширення, ніж тверді тіла, і це можна використати в термопневматичних та терморідинних мікроактюаторах.

Наведемо основні переваги теплових мікроактюаторів: простота конструкції, робочими елементами є резистори, які досить просто виготовити за допомогою групових мікротехнологій; розміри інтегральних пристроїв розміщені в мікродіапазоні, де виконується співвідношення (закон) Фур'є (швидкодія при зменшенні розмірів зростає квадратично); як активні елементи використовують практично будь-які матеріали, які окрім різних коефіцієнтів розширення мають достатню міцність.

Як правило, в термоактюаторах в якості нагрівального елемента використовують резистори хвилеподібної форми, які можна досить легко виготовити з використанням тонко – або товстоплівкової технологій [54].

До недоліків теплових актюаторів можна віднести такі: нагрівальний елемент потребує дуже багато енергії, тобто невисокий ККД пристроїв даного типу; нагрівальний елемент необхідно охолоджувати, щоб повернути актюатор в початкове положення.

Приклад базової конструкції мікротермодавача наведено на рис.4.24. Вона включає двохшарову пластину, на якій розміщено п'єзореzystор. Особливістю виготовлення даного інтегрального пристрою є те, що кожний з шарів виготовляють з матеріалів з різними коефіцієнтами теплового розширення, для прикладу, алюміній та кремній, які є стандартними матеріалами для технології виготовлення ІС.

Такий термодавач призначений для вимірювання температури. Принцип його роботи базується на тому, що при збільшенні температури від значення T_1 до T_2 матеріали 1 та 2 будуть збільшувати власний об'єм. Оскільки коефіцієнти теплового розширення для кожного з матеріалів різні, а границя між двома матеріалами жорстко защемлена, то дане збільшення температури призведе до

деформації даного інтегрального пристрою. Наявність термодеоформацій призведе до появи термонапружень.

Математична модель термодавача включає три основні підзадачі: в залежності від прикладеної температури необхідно визначити поле деформацій [55]; по відомому полю деформацій визначити розподіл термонапружень в конструкції термодавача; визначення зміни опору та вихідної напруги по відомих параметрах поля напружень.

Для розв'язку першої та другої підзадач використовується метод скінченних елементів [55].

Приклад базової конструкції термоактюатора наведено в літературі [56]. Математична модель для компонентного рівня розроблення передбачає визначення зміщення вільного кінця актюатора, а саме: в залежності від прикладеної температури визначення поля деформацій. При необхідності визначення критичних параметрів мікротермоактюатора необхідно провести розрахунок розподілів термодеоформацій та термонапружень в конструкції елемента МЕМС. Особливості розв'язання цих задач детально описано в роботі [57, 58]. При виготовленні термоактюаторів найчастіше використовують MUMPs – технології і технології поверхневої мікрообробки.

Основним елементом будь-якої гідравлічної мікросистеми є мікропомпи. На рис. 4.25 зображена мікропомпа з електростатичним приводом (актюатором). Наведена конструкція мікропомпи включає дві основні частини: привідна частина конструкції під дією прикладеної напруги генерує циклічні переміщення мембрани та клапанна частина призначена для управління напрямком течії робочої рідини.

Підключення електричної напруги, у формі прямокутних імпульсів, до верхнього та нижнього електродів мікропомпи призводить до виникнення електростатичної сили, яка є рівномірно розподілена по мембрані. В результаті дії прикладеної сили – мембрана вигинається (рис. 4.25а), виникає різниця тисків, за рахунок цього всмоктувальний клапан відкривається, а рідина втягується до робочої камери мікропомпи. Після зняття напруги з електродів (за рахунок сил пружності пластина повертається в початкове положення) чи

прикладання напруги іншої полярності призводить до руху мембрани в протилежному напрямку і, відповідно, виштовхування робочої рідини з камери мікропомпи (рис.4.25б). При цьому виникаючий додатковий тиск відкриває вихідний клапан і робоча рідина виштовхується в вихідний канал.

Математична модель гідравлічної мікропомпи з електростатичним приводом передбачає розв'язання послідовності з двох задач. Перша задача пов'язана з визначенням зміщень тонкої пластини по відношенню до початкового положення, а друга – визначення напружень в мембрані.

З курсу теорії пружності відомо, що якщо товщина пластини є набагато менша за її лінійні розміри, то для опису поведінки, в даному випадку переміщень пружного елемента мікропомпи, можна використати теорію тонких пластин [59]. Для опису переміщень тонкої пластини використаємо бігармонійне рівняння. Будемо вважати, що розподіл сили по пластині є рівномірний, а її краї - жорстко защемлені. Відповідно, для визначення деформацій в пластині використаємо наступне диференціальне рівняння:

$$D \frac{\partial^4 w(x,y,t)}{\partial x^4} + 2D \frac{\partial^4 w(x,y,t)}{\partial x^2 \partial y^2} + D \frac{\partial^4 w(x,y,t)}{\partial y^4} - \rho h \frac{\partial^2 w^2(x,y,t)}{\partial t^2} = P(t),$$

$$D = \frac{Eh^3}{12(1-\nu^2)}, \quad (4.3)$$

де $w(x,y,t)$ – прогини пластини (вертикальні переміщення пластини); x, y – напрямки осей, які формують координатну площину xu , що рівновіддалена від основ пластинки; h – товщина пластини; P – інтенсивність розподіленого на поверхні пластини навантаження.

Для визначення напружень в тонкій пластині скористаємося наступними рівняннями:

$$\sigma_x(t) = \frac{Eh}{2(1-\nu^2)} \left(\frac{\partial^2 w(x,y,t)}{\partial x^2} + \nu \frac{\partial^2 w(x,y,t)}{\partial y^2} \right), \quad (4.4)$$

$$\sigma_y(t) = -\frac{Eh}{2(1-\nu^2)} \left(\frac{\partial^2 w(x,y,t)}{\partial y^2} + \nu \frac{\partial^2 w(x,y,t)}{\partial x^2} \right), \tau_{xy}(t) = -\frac{Eh}{2(1-\nu^2)} \left(\frac{\partial^2 w(x,y,t)}{\partial x \partial y} \right), \quad (4.5)$$

де σ_x - напруження в напрямку осі x , σ_y - напруження в напрямку осі y , τ_{xy} - дотичне напруження.

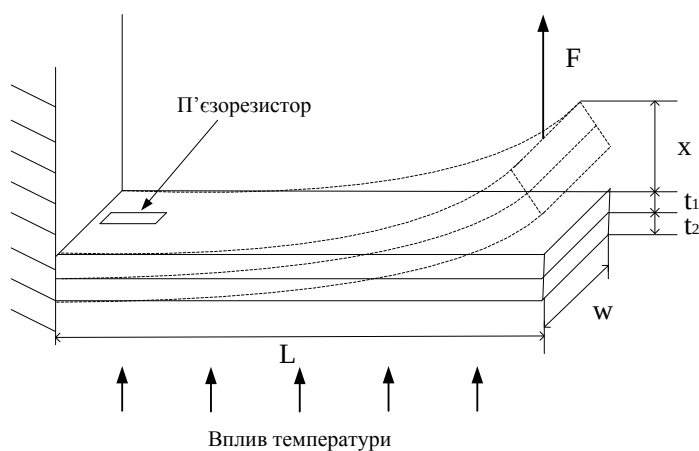


Рис.4.24. Базова конструкція мікротермодавача

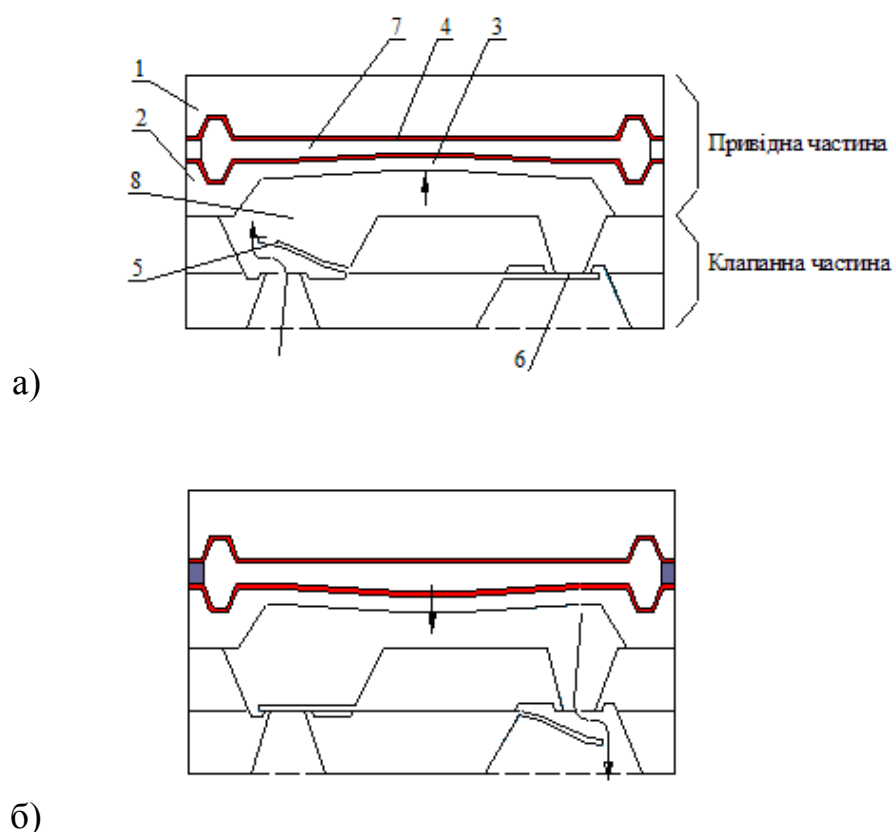


Рис. 4.25. Мікропомпа з електростатичним приводом: а) в фазі всмоктування, б) в фазі виштовхування рідини. (1 - верхній електрод; 2 - нижній електрод; 3 - мембрана помпи; 4 - ізоляційний шар; 5 – всмоктувальний клапан; 6 – вихідний клапан; 7 - камера керування; 8 – робоча камера).

Краєві та початкову умови запишемо в наступній формі:

$$w|_{\partial p} = w_0 = 0, \quad \left. \frac{\partial w}{\partial N} \right|_{\partial p} = 0, \quad \frac{\partial w(x,y,t=0)}{\partial t} = 0, \quad w(x,y,t=0) = 0, \quad (4.6)$$

де n – нормаль до границі області моделювання.

Електростатична сила, яка виникає між пластинами (електродами) мікропомпи визначається з допомогою наступного виразу [31]:

$$F_{el} = \frac{1}{2} \frac{\varepsilon_0 \varepsilon_c S U^2}{d^2} = \frac{1}{2} \frac{C U^2}{d}.$$

Для розв'язання задачі (4.3 – 4.6) визначення деформацій і напружень в тонкій пластині використано метод скінченних різниць.

Приклад базової конструкції мікропомпи з п'єзоелектричним приводом зображено на рис. 4.26. Зміщення тонкої пластини мікропомпи виникають завдяки використанню п'єзоелектричного елемента, який є жорстко з'єднаний з мембраною мікропомпи. Особливості роботи мікропомпи з п'єзоелектричним елементом до якого прикладено електричну напругу зображено на рис. 4.26а – в фазі всмоктування робочої рідини. Прикладання нульового значення напруги, або її зняття – призводить до повернення пластини в початкове положення і, відповідно, виштовхування рідини з робочої камери мікропомпи (рис. 4.26б). Значення робочої напруги у формі імпульсів складає сотні вольт.

Приклад спрощеної конструкції привідної частини п'єзоелектричної мікропомпи наведено на рис. 4.27. Математична модель для визначення зміщень тонкої пластини з п'єзоелектриком включає стаціонарне бігармонійне

диференціальне рівнянням $\frac{\partial^4 w(x,y)}{\partial x^4} + 2 \frac{\partial^4 w(x,y)}{\partial x^2 \partial y^2} + \frac{\partial^4 w(x,y)}{\partial y^4} = \frac{F_z}{D}$, рівняння для

визначення напружень в тонкій пластині $\sigma_x = \frac{Eh}{2(1-\nu^2)} \left(\frac{\partial^2 w(x,y)}{\partial x^2} + \nu \frac{\partial^2 w(x,y)}{\partial y^2} \right)$,

$$\sigma_y = -\frac{Eh}{2(1-\nu^2)} \left(\frac{\partial^2 w(x,y)}{\partial y^2} + \nu \frac{\partial^2 w(x,y)}{\partial x^2} \right), \quad \tau_{xy} = -\frac{Eh}{2(1-\nu^2)} \left(\frac{\partial^2 w(x,y)}{\partial x \partial y} \right) \text{ та краєві умови}$$

$$w|_{\partial p} = w_0 = 0, \quad \left. \frac{\partial w}{\partial N} \right|_{\partial p} = 0.$$

Для визначення величини п'єзоелектричної сили в залежності від геометричних параметрів прямокутної п'єзопластини та прикладеної до неї напруги можна використати наступні вирази [60]: $F_z = \varepsilon_o \varepsilon \frac{L U}{dxz}$, $F_y = \varepsilon_o \varepsilon \frac{W U}{dxz}$, $F_x = \varepsilon_o \varepsilon \frac{U L W}{dxzh}$, де U - прикладена напруга; h - висота п'єзопластини; W - ширина п'єзопластини; L - довжина п'єзопластини.

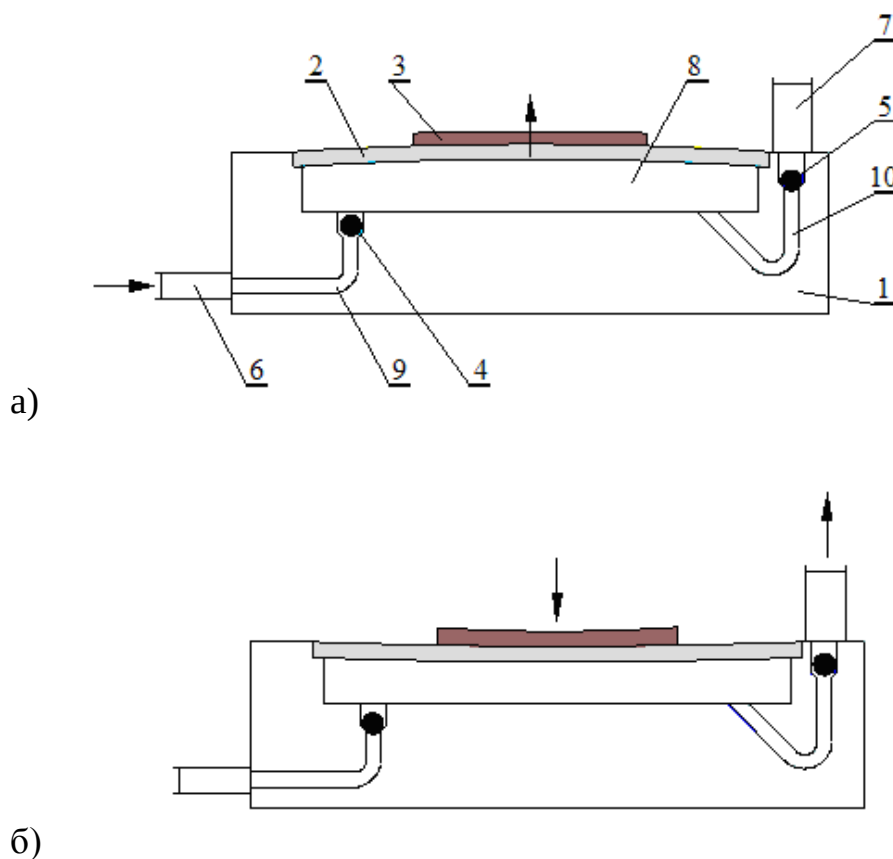


Рис. 4.26. Мікропомпа з п'єзоелектричним приводом:

а) в фазі всмоктування, б) в фазі виштовхування рідини, 1 – корпус, 2 – мембрана, 3 – п'єзоелектричний елемент, 4 – вхідний клапан, 5 – вихідний клапан, 6 – вхідний канал мікропомпи, 7 – вихідний канал мікропомпи, 8 – робоча камера помпи, 9 – вхідний канал клапану, 10 – вихідний канал клапану

У випадку круглої пластини п'єзоелектрика з товщиною h та діаметром $Diam$ (рис. 5.26), зміна його лінійних розмірів під дією прикладеної напруги U можна визначити з допомогою використання наступних виразів [61, 62] (рис.5.27): $\Delta h = d_{33}U = d_{33}Eh$, $\Delta Diam = d_{33}U = d_{33}EDiam$,

$F_z = \frac{U E d_{33} \pi \left(\frac{Diam}{2} \right)^2}{h}$, де Δh - прирість товщини п'єзоелектрика; $\Delta Diam$ - прирість діаметра п'єзоелектрика; E – напруженість електричного поля.

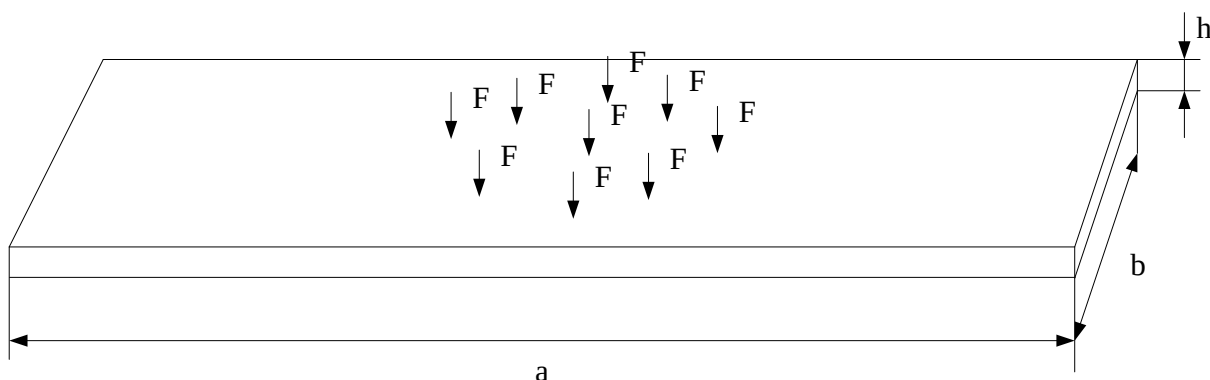


Рис.4.27. Спрощена конструкція привідної частини п'єзоелектричної мікропомпи

Для розв'язання сформульованої задачі використано метод скінченних різниць.

Конструкція п'єзорезистивного мікродавача наведена на рис. 4.27. Ця конструкція включає основу, пружний елемент та п'єзорезистори. Вигляд зверху цієї конструкції наведено на рис. 4.28.

Принцип дії цього мікродавача полягає в наступному. На пружний елемент інтегрального пристрою діє тиск, який рівномірно розподілений по поверхні та призводить до деформації пружного елемента (тонкої кремнієвої пластини). З теорії визначення деформацій та напружень [63], для даного випадку, відомо, що максимальні напруження будуть зосереджені біля границь пружного елемента, тому для досягнення максимальної чутливості

мікродавача п'єзореzystори розміщують на цих границях або в безпосередній близькості від них [64].

В мікродавачі використано чотири п'єзореzystори, які з'єднані в ланку в формі містка. В кінцевому випадку зміна опорів резисторів призводить до зміни вихідної контрольованої напруги, по змінах якої визначають величину прикладеного тиску до поверхні пружного елемента мікродавача.

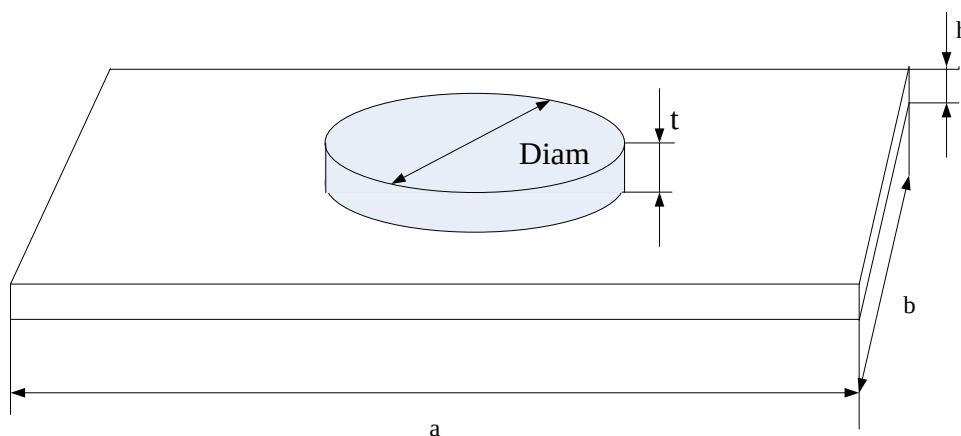


Рис.5.26. Конструкція тонкої пластини мікропомпи з п'єзоелектричним диском

Математична модель по визначенню змін вихідної напруги містка в залежності від прикладеного тиску до пружного елемента інтегрального мікродавача передбачає послідовне вирішення наступних підзадач: визначення розподілу переміщень, які виникають в конструкції пружного елемента та п'єзореzystорів зокрема при прикладанні тиску до мікродавача; визначення розподілу напружень в тонкій кремнієвій пластині та конструкції п'єзореzystорів по відомому розподілі переміщень; розрахунок змін опорів п'єзореzystорів; визначення зміни вихідної напруги містка.

Пружний елемент мікродавача, з точки зору механіки, є тонкою пластиною. Тому при визначенні розподілів переміщень та напружень використовується теорія тонких пластин. Для визначення розподілу переміщень необхідно розв'язати бігармонійне рівняння з краєвими умовами, які

враховують жорстке закріплення пластини по її краях. Одержаний розв'язок дає змогу визначити вертикальний прогин середньої площини пластини.

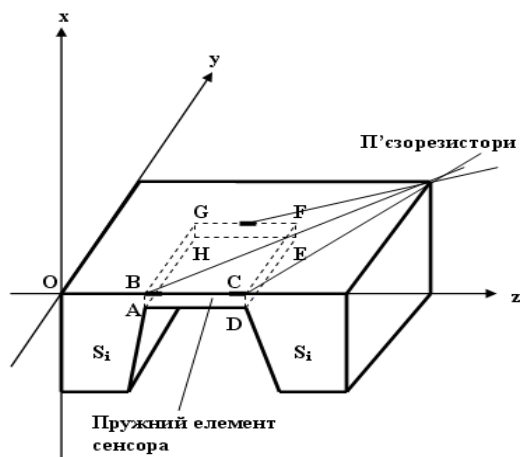


Рис.4.27. Конструкція п'єзореzystивного мікродавача

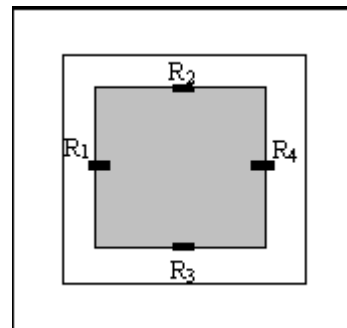


Рис.4.28. Вид зверху та збоку інтегрального мікродавача

Приклад базових конструкцій п'єзореzystивних акселерометрів наведено на рис. 4.29 та рис. 4.30. Виготовляють такі конструкції п'єзореzystивних акселерометрів з допомогою об'ємної мікрообробки. Основним матеріалом може бути пластина або балка з кремнію чи полікремнію. Даний акселерометр балочного типу включає інерційну масу, концентратор напружень, п'єзореzystор та основу, до якої прикріплено інтегральний пристрій.

Принцип роботи акселерометрів передбачає, що під дією прикладеної сили до інерційної маси – конструкція вигинається і найбільші напруження будуть сконцентровані в найтоншій області акселерометра, де розміщені п'єзореzystори.

П'єзореzystори в залежності від змін напружень змінюють опір, який реєструється відповідною схемою. Математична модель п'єзореzystивного акселерометра передбачала розв'язання послідовності таких задач: визначення напружень в “концентраторі напружень” під дією прикладеної сили; визначення зміни опору по відомих значеннях напружень.

Для досягнення вищої точності при визначенні напружень необхідно використати метод скінченних елементів.

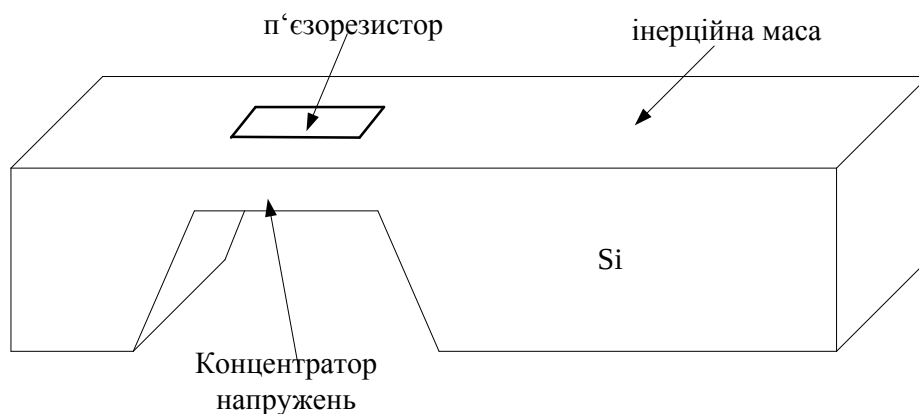


Рис.4.29. Конструкція п'єзореzystивного акселерометра балочного типу

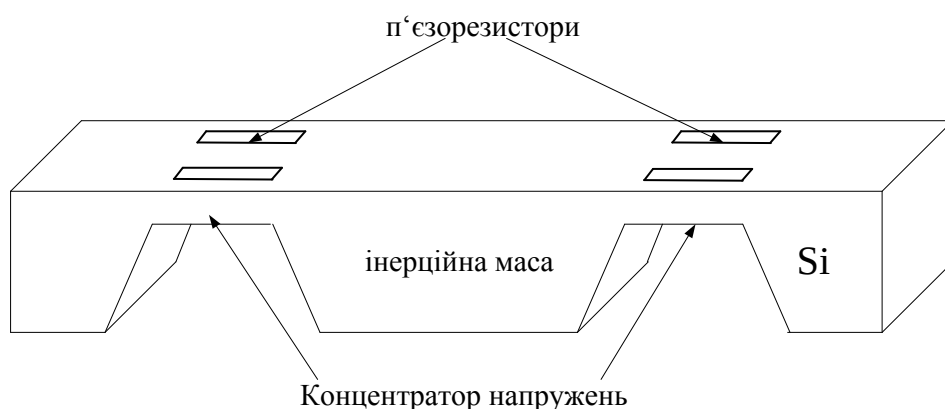


Рис.4.30. Конструкція п'єзореzystивного акселерометра балочного типу з двома концентраторами напружень

4.3. Особливості розроблення базових структур МЕМС

Загальна структура МЕМС наведена на рис. 4.31. Вона включає вхідний перетворювач, мікропроцесор (пристрій для опрацювання, збереження, передачі інформації) та вихідний перетворювач.

Отже, вхідний перетворювач (надалі мікродавач) призначений для визначення змін чи впливу оточуючого середовища на інтегральний пристрій. В багатьох мікропроцесорах, в якості вхідного електричного параметра, можуть виступати зміна опору, ємності, частоти, напруги, струму тощо.

Оскільки, безпосередньо, аналогову величину напруги чи струму мікропроцесор обробляти не може, то після мікродавача використано аналого-

цифровий перетворювач (АЦП), з якого вже цифровий сигнал поступає на мікропроцесор. Мікропроцесор обробляє отримані дані згідно попередньо визначеного алгоритму, а результат опрацювання, у формі цифрового сигналу, видає на цифро-аналоговий перетворювач (ЦАП). ЦАП перетворює код в аналоговий сигнал, який безпосередньо подається на вихідний перетворювач. В якості вихідного перетворювача виступають актюатори.

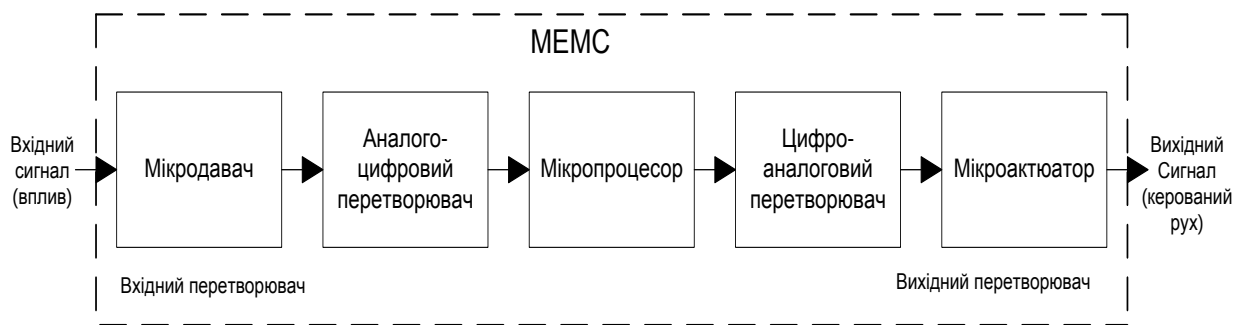


Рис.4.31. Загальна структура MEMC

Разом з тим, можлива й інша структура, яка наведена на рис.4.32. Її особливістю є те, що вона обробляє лише аналоговий сигнал і включає, відповідно, мікродавач, схему керування та опрацювання аналогового сигналу і мікроактюатор. Принцип дії такої MEMC аналогічний до попередньої.

Наведені структури MEMC на рис.4.31 та рис.4.32 відносяться до найпростіших. Особливістю їх є те, що реалізувати таку MEMC можна за допомогою єдиної технології виготовлення, хоча можуть виникнути проблеми виготовлення мікродавача та мікроактюатора за єдиною технологією та розміщення їх на одному кристалі. Таким структурам притаманна найвища швидкодія (з двох вищенаведених структур кращі параметри швидкодії має структура з аналоговою схемою керування та опрацювання сигналу). Живлення в таких пристроях подається від макросистеми. В більшості випадків для реалізації мікродавача та мікроактюатора використовують технології поверхневої чи об'ємної обробки, або їх похідні та КМОН технологія для виготовлення мікропроцесора, яка є сумісною з двома вище перерахованими.

Прикладом MEMC, які використовують такі структури, є підсистема викидання подушок безпеки в автомобілі, системи контролю цукру в крові людини тощо.

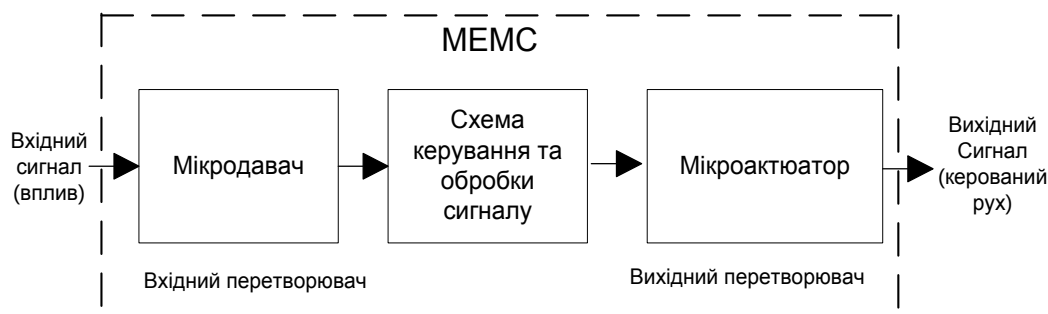


Рис.4.32. Структура MEMC для опрацювання аналогового сигналу

Складні структури MEMC можуть включати десятки мікродавачів, кілька каналів опрацювання інформації та десятки виконуючих пристроїв. Прикладом MEMC з такою структурою може бути “мікролабораторія-на-кристалі” (рис.4.33 та рис. 4.34) [65, 66, 67, 68, 69, 70, 71]. Різниця між структурами на рис.4.33 та рис.4.34 полягає в тому, що на рис.4.33 структура включає один канал опрацювання даних, а на рис.4.34 – декілька, що дає змогу підвищити швидкодію системи.

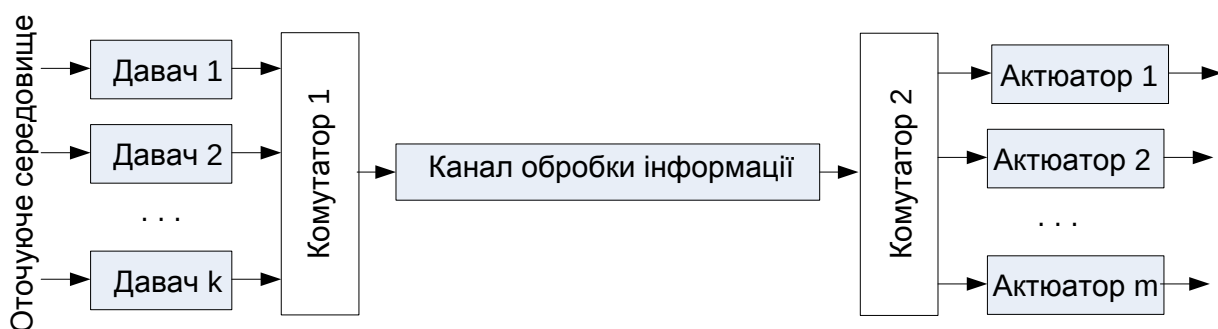


Рис.4.33. Приклад складної структури MEMC з одним каналом опрацювання інформації

За кількістю елементів структура на рис.4.34 - складніша, але структура на рис.4.33 має кращі параметри швидкодії та надійності.

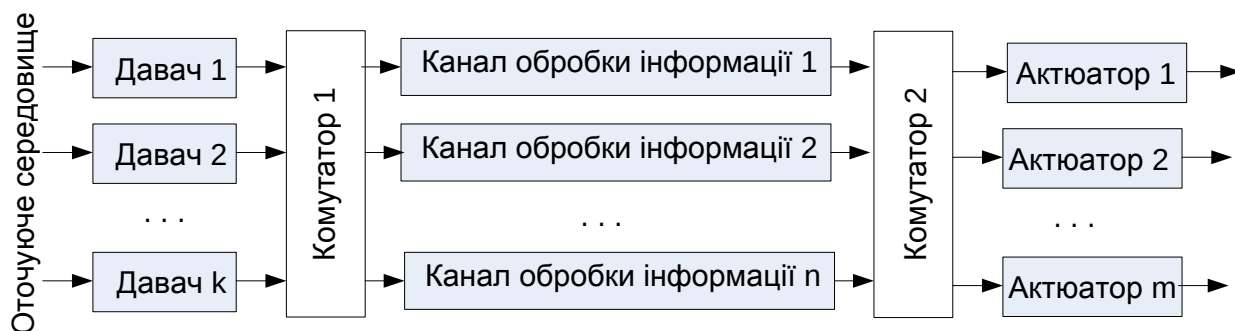


Рис. 4.34. Приклад складної структури МЕМС з кількома каналами
опрацювання інформації

Усі канали опрацювання інформації на рис.4.34 можна розділити на дві групи. Перша група характеризується тим, що опрацювання інформації відбувається в програмному режимі і, відповідно, для неї притаманні певні особливості. Зокрема, до переваг програмних каналів опрацювання інформації можна віднести можливість зміни програм опрацювання не змінюючи структури системи, що дає змогу змінювати функції, які вона виконує. Друга група каналів обробляє інформацію від давачів за допомогою апаратних засобів і має кращі параметри швидкодії ніж перша група. Разом з тим, для каналів кожної групи притаманні і недоліки, зокрема: канали з програмною обробкою інформації мають гірше значення швидкодії ніж канали з апаратними засобами опрацювання, які не надають можливості змінювати алгоритми опрацювання інформації від мікродавачів. Тому доцільно в одній МЕМС передбачити як канали для програмної опрацювання інформації, так і канали для апаратної опрацювання. Приклад відповідної структури наведено на рис.4.35. Побудова МЕМС з такою структурою вимагає вирішення на верхніх рівнях задач пов'язаних з вибором методу реалізації функцій системи (програмної чи апаратної опрацювання), розробкою структури програмного і математичного забезпечень для програмної опрацювання інформації від мікродавачів.

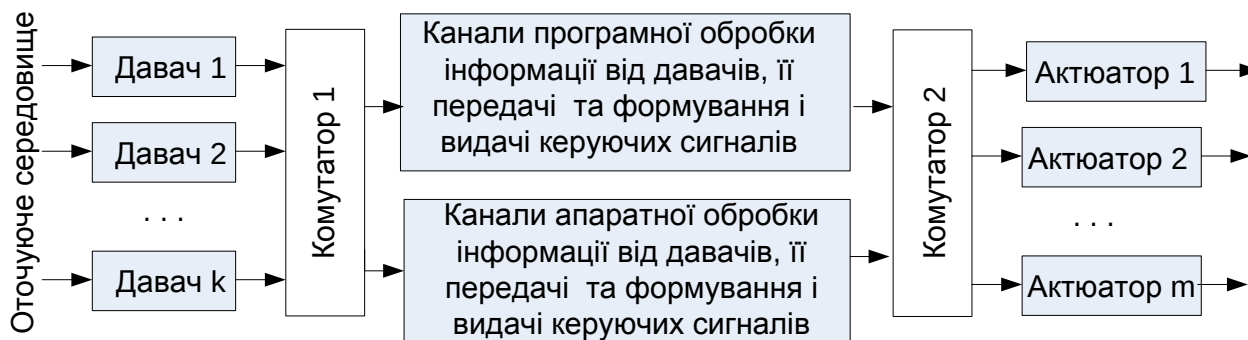


Рис. 4.35. Приклад складної структури MEMC з каналами апаратної та програмної опрацювання інформації від давачів, її передачі та формування і видачі керуючих сигналів

В більшості випадків MEMC є системами реального часу, тому особлива увага приділяється такому параметру як швидкодія, та методам її підвищення. Одним з архітектурних методів підвищення швидкодії MEMC є реалізація мікросистеми на основі розподіленої структури, приклад якої наведено на рис.4.36. Вона може використовуватися при реалізації спеціального виду MEMC так званих “розумних структур”, “розумних поверхонь” тощо.

Приклад структури MEMC, яка дає змогу встановити зв'язок з іншими зовнішніми системами та їй подібними зображено на рис.4.37. Основне призначення мікросистем даного типу полягає в зборі інформації про оточуюче середовище, її часткову чи повну обробку, передача, обмін інформацією між подібними їй пристроями тощо. Запропонована структура може використовуватися при реалізації пристроїв типу “розумний пил” та інші.

4.4. Особливості реалізації підсистеми керування КБД

Оскільки програма працює з елементами структур MEMC та їх властивостями, в систему введено таке поняття, як шаблон. У роботі з базою даних може виникнути необхідність створення декількох елементів одного типу і було б недоцільним кожного разу створювати спільні властивості для всіх

однотипних елементів. Шаблони елементів включають в себе не тільки загальну інформацію, характерну всім елементам цього типу, а й набір шаблонних властивостей елемента (передбачено, що ці властивості будуть спільними для усіх елементів одного типу). Наприклад шаблон актюатора містить наступні властивості: матеріал виготовлення; тип; точність; чутливість; форма; вага.

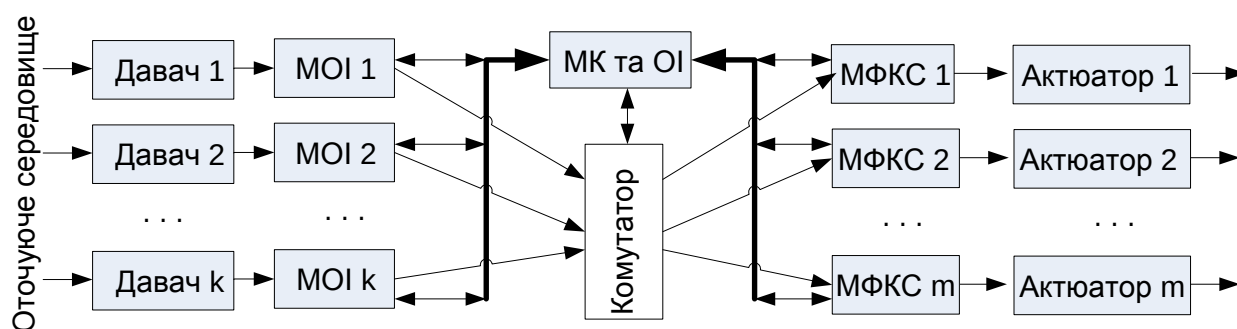


Рис. 4.36. Приклад складної розподіленої структури МЕМС (МОІ – модуль опрацювання інформації, МКтаОІ – модуль керування та опрацювання інформації, МФКС – модуль формування керуючого сигналу)

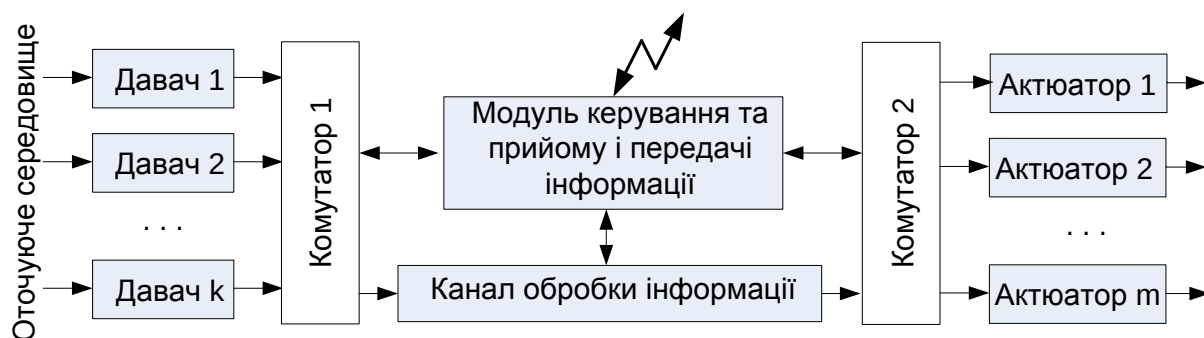


Рис. 4.37. Приклад структури МЕМС з зовнішнім керуванням

Властивості шаблону не будуть містити конкретних значень, лише узагальнені дані про властивості такі, як назва, одиниці виміру та тип. При створенні елемента користувач може обрати шаблон для цього елемента і все,

що залишиться зробити, - це визначити значення тих чи інших властивостей і додати унікальні властивості. Крім особливих шаблонів для типових елементів в системі передбачений шаблон General, в якому буде міститися перелік усіх можливих властивостей. Якщо користувач захоче створити елемент без використання типового шаблону, він зможе завантажити шаблон General і вибрати зі списку його властивостей ті, які необхідні для визначення нового елемента. Окрім того, коли до певного елемента додається властивість якої ще не містить шаблон General, то вона буде автоматично додана в цей шаблон і надалі користувач зможе вибирати її з вже існуючого шаблону.

Інтерфейс користувача представлений у формі прикладної програми Windows. Враховуючи те, що КБД повинна бути відкритою і для її наповнення необхідно залучити максимальну кількість користувачів, є сенс розробки веб-інтерфейсу для реалізації надання одночасного доступу до конструкторської бази даних багатьом користувачам. Веб-інтерфейс повинен передбачати можливість експорту та імпорту інформації про структурні елементи MEMC у форматі представлення даних XML.

На рис. 4.38. відображено дані про елементи, що містяться в базі даних.

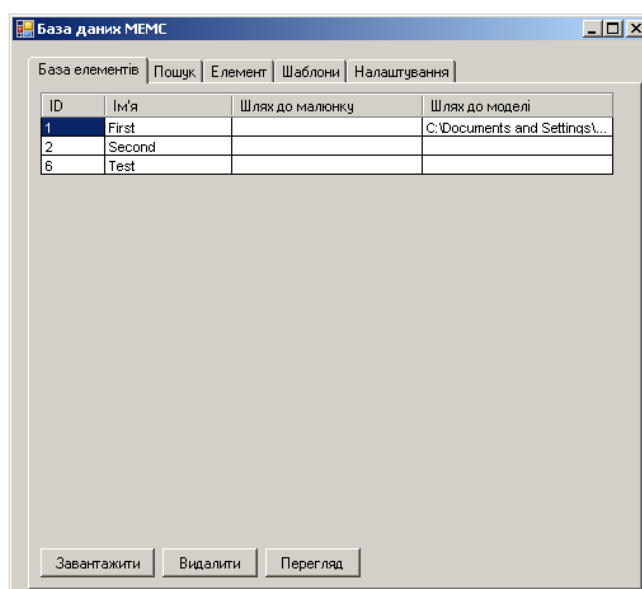


Рис. 4.38. Закладка “База елементів”. Загальний вигляд

Як бачимо на рис. 4.39, для видалення елемента достатньо виділити будь-яку комірку з таблиці елементів та натиснути кнопку “Видалити”. З бази даних буде видалено той елемент, який було обрано в таблиці.

При натисненні на кнопку “Перегляд” відкривається вікно перегляду властивостей елемента (Рис. 4.40). У цьому вікні відображається вигляд елемента (якщо існує відповідний файл). При бажанні, користувач може переглянути конструкторсько-технологічну документацію натиснувши на кнопку “Перегляд”.

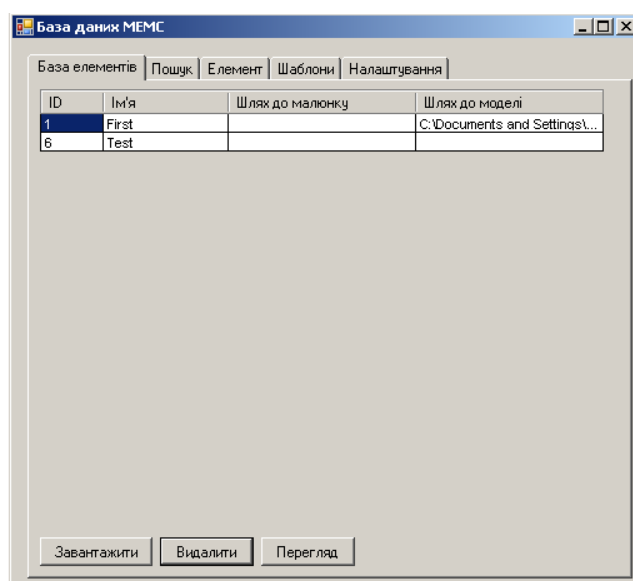


Рис. 4.39. Видалення елемента з бази за допомогою кнопки “Видалити”

На закладці “Елемент” містяться всі функції для створення та редагування елементів. На рис. 4.41 зображено закладку “Елемент”, в переліку властивостей завантажений шаблон General. Користувач може відзначити необхідні властивості і на основі їх створити новий елемент.

Для створення елемента користувач відзначає необхідні властивості і натискає на клавішу “Створити” (рис. 4.42). Після натиснення на дану клавішу користувачу пропонується заповнити наступні поля:

- ім'я елемента;
- шлях до файлу з зображенням елемента;
- шлях до файлу з конструкторсько-технологічною документацією.

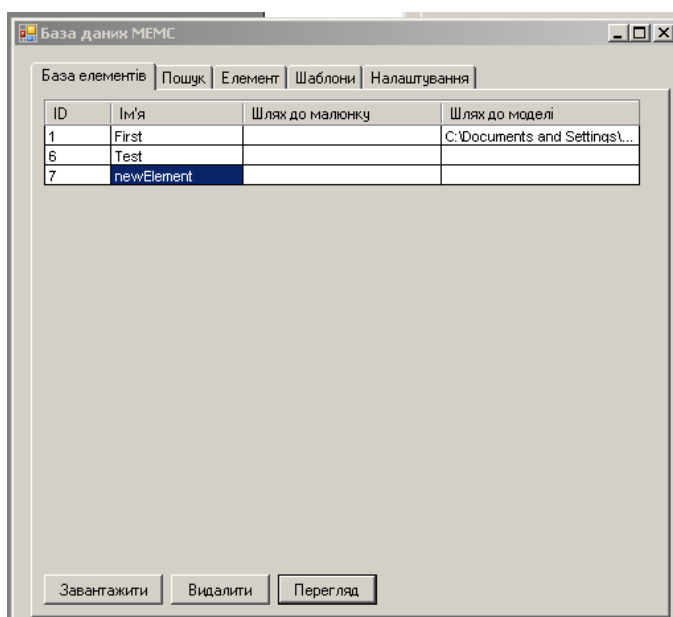


Рис. 4.40. Перегляд елементу

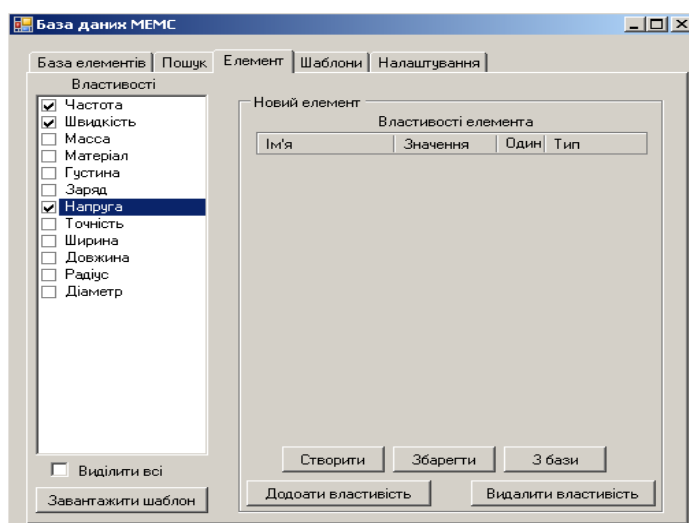


Рис. 4.41. Закладка “Елемент”

Після того, як користувач ввів загальні дані про елемент, програма пропонує йому визначити властивості даного елементу (рис. 4.43). Користувачу пропонується ввести значення всіх обраних ним властивостей. Вікна вводу властивостей з’являються по чергову.

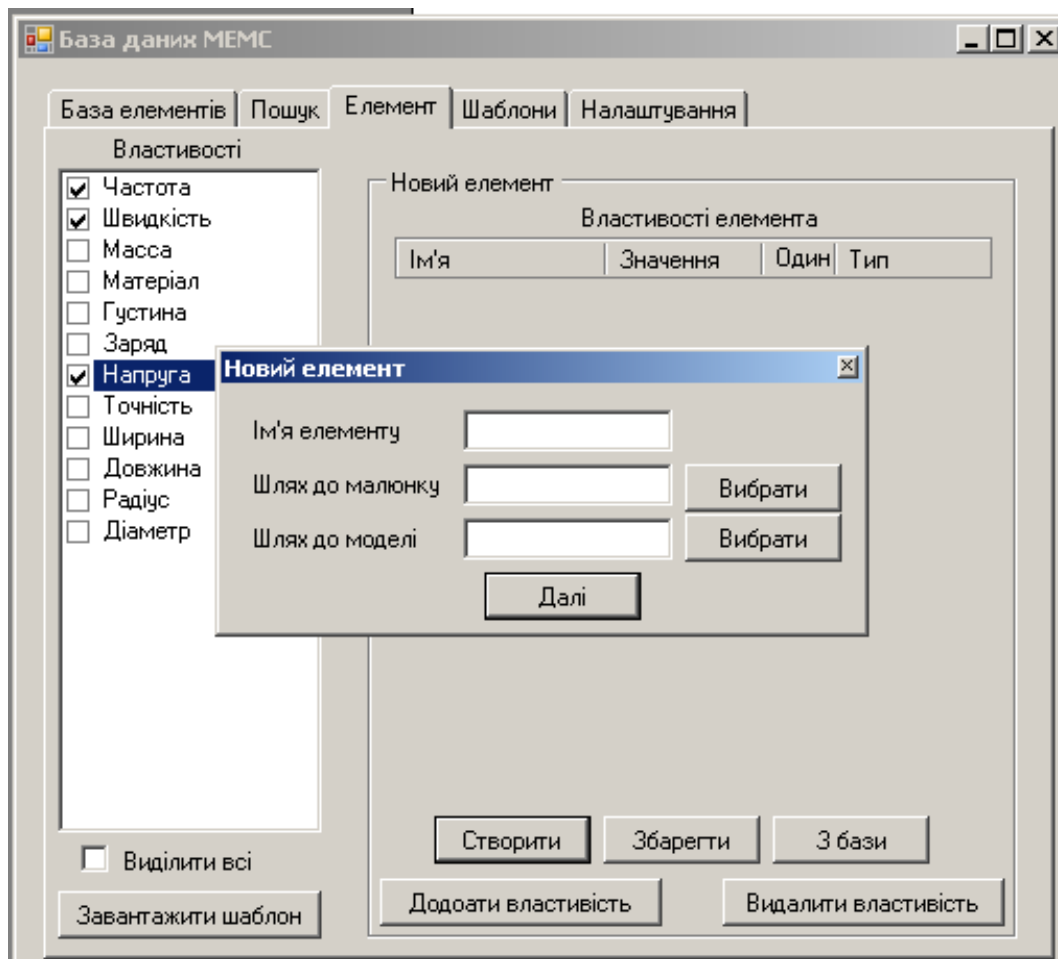


Рис. 4.42. Створення нового елементу

Після збереження елемент одразу ж відображається на закладці “Елемент”. На рис. 4.44. показано як виглядає новостворений елемент.

Закладка “Елемент” дає можливість редагувати властивості елемента. За допомогою кнопок “Додати властивість” (рис. 4.45) та “Видалити властивість” користувач може додавати або видаляти властивості елемента.

Кнопка “Видалити властивість” видаляє ту властивість, на яку є фокус у таблиці. При допомозі кнопки “Зберегти” новостворений елемент записується в базу.

Програма також пропонує користувачу зручний спосіб пошуку і редагування елементів у базі. Після натиснення на кнопку “Пошук” з’являється діалогове вікно, в якому користувачу дається змога вибрати тип пошуку: пошук за ID, пошук за ім’ям (рис. 4.46). Після того, як елемент був знайдений, його властивості відображаються в закладці “Елемент” і далі користувач може працювати з елементом та його властивостями.

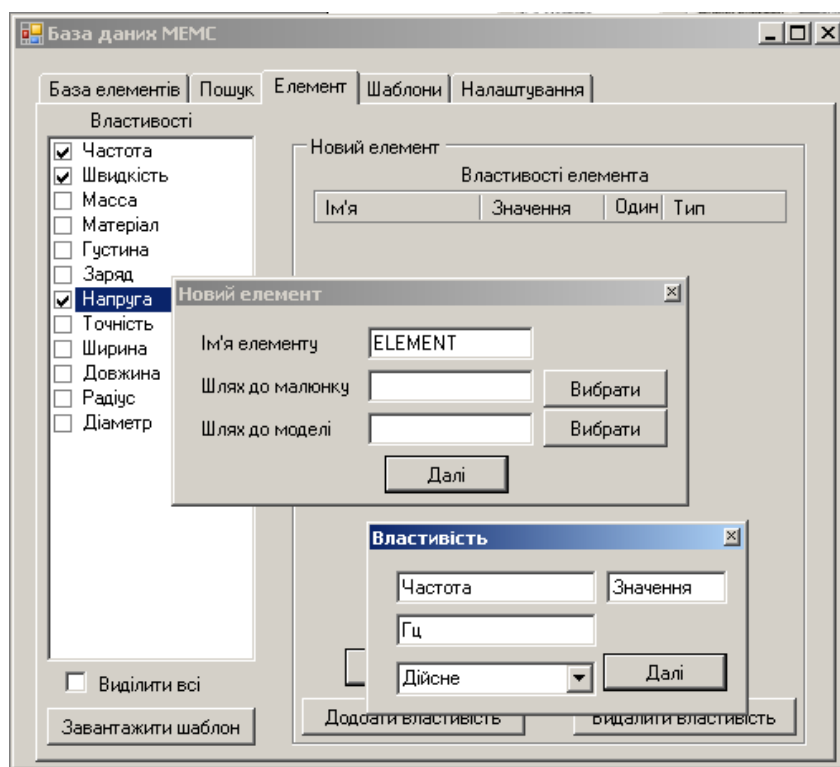


Рис. 4.43. Визначення властивостей елемента.

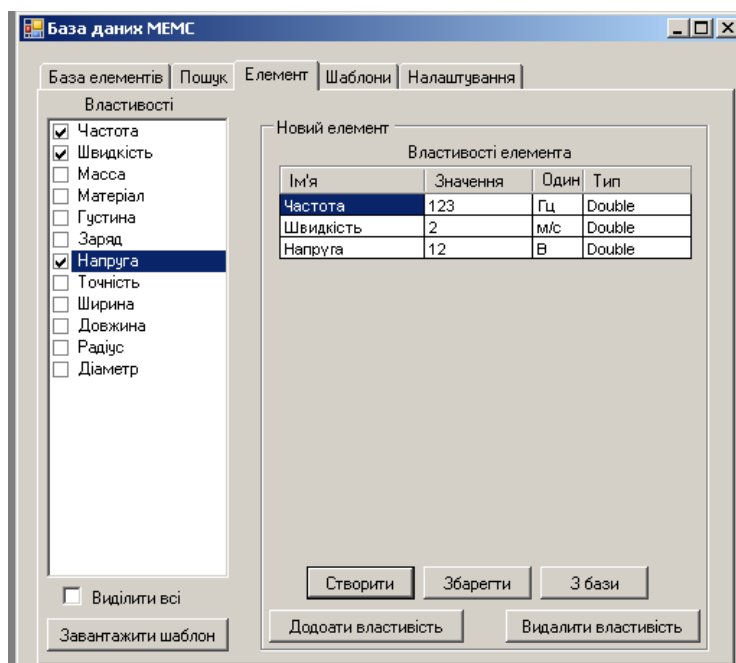


Рис. 4.44. Новостворений елемент

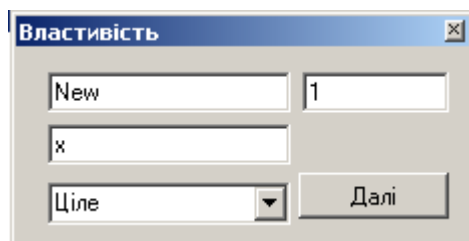


Рис. 4.45. Додавання властивості за допомогою кнопки “Додати властивість”

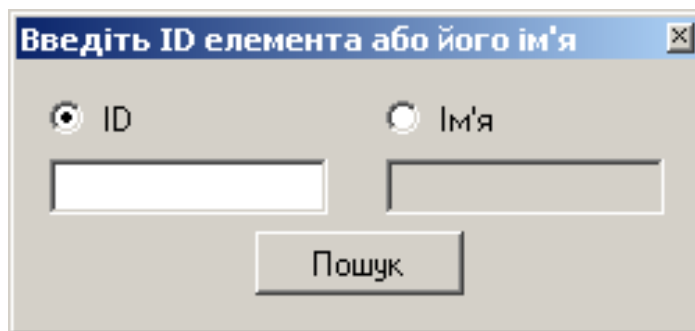


Рис. 4.46. Вікно пошуку елемента

4.5. Контрольні запитання до розділу

1. Які елементи системи називають базовими?
2. Наведіть приклади базових елементів MEMC?
3. Яка особливість базових структур MEMC?
4. Наведіть приклад базових структур MEMC?
5. Які критерії використовують для класифікації MEMC?
6. Наведіть приклади класифікацій давачів?
7. Наведіть приклади класифікацій актюаторів?
8. Наведіть приклади класифікацій систем опрацювання даних від підсистеми давачів?
9. Наведіть приклади технологій виготовлення MEMC?
10. Які переваги та недоліки типових структур MEMC?

4.6. Список використаної літератури до четвертого розділу

- [1] Максвелл Дж. К. Трактат об електричестві и магнетизме. В двух томах. Т.1. - М.: Наука., 1989.
- [2] Pereyma M., Denysyuk P., Teslyuk V. Feature Of Optical Embedded System Design // Proc. of the 1st Inter. Conf. of Young Scientists “Perspective Technologies and Methods in MEMS Design” (MEMSTECH 2005).- Lviv–Polyana, Ukraine, 2005. – P. 31-33.

- [3] Antonyuk M., Lobur V., Teslyuk V. Bio-Robot For Research And Analysis Of Human Organism // Proc. of the 1st Inter. Conf. of Young Scientists "Perspective Technologies and Methods in MEMS Design" (MEMSTECH 2005).- Lviv-Polyana, Ukraine, 2005. – P. 18-19.
- [4] Денисюк П., Теслюк В.М., Перейма М. Є. Математична модель базової конструкції гідравлічної мікропомпи з п'єзоелектричним приводом для компонентного рівня проектування // Вісник Національного університету „Львівська політехніка”: Комп'ютерні системи проектування. Теорія і практика. - Львів, 2005. – № 548. – С. 106 - 111.
- [5] Теслюк В.М., Денисюк П.Ю. Математична модель термоактюатора рідинного типу // Вісник Національного університету «Львівська політехніка»: Комп'ютерні системи проектування. Теорія і практика. - Львів, 2004. – № 522. – С. 181 - 185.
- [6] Физическая энциклопедия. / В пяти томах. –М.: Советская энциклопедия, 1998.
- [7] Савельев И. В. Курс общей физики: Учеб. Пособие: для вузов. В 5 кн. Кн.2. Электричество и магнетизм / И. В. Савельев : 4-е изд., перераб. – М. : Наука, Физматлит, 1998. – 336 с.
- [8] Mitcheson P. D. Mems electrostatic micro - power generator for low frequency operation / P. D. Mitcheson, P. Miao, B. H. Stark [et al.] // Solid-State Transducers (Euroensors XVII) : 17-th European Intrn. Conf. – 2004. – P. 523 – 529.
- [9] Huikai Xie Fabrication, Characterization and analysis of a DRIE CMOS – MEMS Gyroscope / Xie Huikai, K. Fedder. Gary // IEEE Sensors Journal. – 2003. – October, Vol. 3, № 5. – P. 622 – 631.
- [10] Xie H. A CMOS – MEMS lateral - axis gyroscope / H. Xie, G. K. Fedder // Microelectromechanical Systems : Proc. 14-th IEEE Int. Conf. – Interlaken, Switzerland. – 2001. – P. 162–165.
- [11] Luo H. A copper CMOS - MEMS z - axis gyroscope / H. Luo, X. Zhu, H. Lakdawala [et al.] // Microelectromechanical Systems : Proc. 15-th IEEE Int. Conf. – Las Vegas, NV, 2001. – P.631 – 634.
- [12] Gupta R. K. Zorian Y. Introducing in core - based system design / R.K. Gupta, Y. Zorian // Design & Test of Computer. – 1997, Oct/Dec. – Vol. 14, №. 4. – P. 15 – 25.
- [13] Xie H. Post – CMOS processing for high - aspect - ratio integrated silicon microstructures / H. Xie, L. Erdmann, X. Zhu [et al.] // IEEE/ASME J. Microelectromech. Syst. – 2002. – Vol. 11, Feb. – P. 93 – 101.
- [14] Yeh C. CMOS interface circuitry for a lowvoltage micromachined tunneling accelerometer / C. Yeh, K. Najafi // Journal of Microelectromechanical Systems. – 1998. – Vol. 7, №. 1. – P. 6 – 15.
- [15] Liu C. H. Characterization of a highsensitivity micromachined tunnelingaccelerometer with micro - resolution / C. H. Liu, T. W. Kenny [et al.] // Journal of Microelectromechanical Systems. – 1998. – Vol. 7, № 2. – P. 235 – 244.
- [16] Федасюк Д. В. Синтез конструктивних параметрів резистивного обмежувача струму із застосуванням математичних моделей теплоелектричних процесів у напівпровідниках / Д. В. Федасюк, П. Сердюк // Комп'ютерні науки та інженерія : матер. 1-ї міжнар. наук. конф. молодих науковців. – Львів, 2006. – С. 113 – 115.
- [17] Zengerle R. A bidirectional silicon icropump / R. Zengerle, S. Kluge, M. Richter, A. Richter // Proc. IEEE MEMS'95. – Amsterdam, Netherlands. – 1995. – P. 19 – 24.
- [18] Judy J. W. Surface – machined microchannel membrane pump / J. W. Judy, T. Tamagawa, D. L. Polla // Proc. IEEE MEMS'91. – Nara, Japan. – 1991. – P. 182 – 186.
- [19] Van Kessel P. F. A MEMS – based projection display / P. F. Van Kessel, L. J. Hornbeck, R. E. Meier, M. R. Douglass // Integrated Sensors, Microactuators and Microsystems : IEEE Proceedings, Special Issue. – 1998, Aug. – P. 1687 – 1704.
- [20] Goldsmith C. Characteristics of Micromachined Switches at Microwave Frequencies / C. Goldsmith, J. Randall, S. Eshelman [et al.] // IEEE MTT - S International Microwave Symposium Digest 2. – 1996. – P. 1141 – 1144.
- [21] Wang K. High – Order Micromechanical Electronic Filters / K. Wang, C. T.-C. Nguyen // IEEE MEMS Workshop. – Japan. – 1997, January. – P. 25 – 30.

- [22] Лобур М. В. Математическое моделирование исходных электрических и конструктивных параметров резонатора емкостного типа / М. В. Лобур, В. Н. Теслюк, Т. В. Свиридова, П. Ю. Денисюк, П. Ю. Раевський П.Ю., С. Г. Мосин // Электронная техника : Межвуз. сб. науч. трудов. – Ульяновск : УлГТУ. – 2004. – С. 89 – 96.
- [23] Матвійків М. Д. Аналіз доцільності побудови давачів мікроелектромеханічних систем на активних напівпровідникових елементах / М. Д. Матвійків, М. В. Лобур, В. М. Теслюк, Т. М. Матвійків // Вісник Національного університету “Львівська політехніка” : Радіотехніка та телекомунікації. – 2006. – № 557. – С. 20 – 24.
- [24] Pereyma M. Approaches for power output increasing of the vibration – based energy harvesting device / M. Pereyma, V. Teslyuk, A. Holovaty // The experience of designing and application of CAD systems in microelectronics (CADSM 2007) : Proc. of the IXth International Conference. – Lviv - Polyana, 2007. – P. 551 – 552.
- [25] Tang W. C. Electrostatic - comb drive of lateral polysilicon resonators / W. C. Tang, T.-C. H. Nguyen, M.W. Judy, R. T. Howe // Solid – State Sensors and Actuators : Proc. 5-th Int. Conf. – Montreux, 1989. – Vol. 2. – P. 328 – 331.
- [26] Lemkin M. A Micromachined fully differential lateral accelerometer / M. Lemkin, B. E. Boser // Custom Integrated Circuits (CICC’96) : Proc. Int. Conf. – San Diego. – 1996, May. – P. 315 – 318.
- [27] Luo H. An elastically gimbaled Z - Axis CMOS - MEMS gyroscope / H. Luo, G. K. Fedder, L. R. Carley // Smart Structures & Microsystems : Proc. Int. Sym. – Hong Kong, 2000, Oct. – P. 19 – 21.
- [28] Тилл У. Интегральные схемы / У. Тилл, Дж. Лаксон. – М. : – 1985. – 504 с.
- [29] Таруи Я. Основы технологии СБИС / Я. Таруи. – М. : Радио и связь. – 1985. – 480 с.
- [30] Pister K. S. J. Microfabricated hinges / K. S. J. Pister, M.W. Judy, S.R. Burgett, R.S. Fearing // Sensors and Actuators. – 1992. – Vol. A33, № 3. – P. 249 – 246.
- [31] Тамм И. Е. Основы теории электричества: учеб. пособие для вузов. / И. Е. Тамм. – М. : Наука. Гл. ред. физ. - мат. лит., 1989. – 504 с.
- [32] Teslyuk V. Design of output parameters of integral capacity sensor / V. Teslyuk, M. Lobur, P. Rayevskyy // The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM’2005) : Proc of the VIII - th Intern. Conf. – Lviv - Polyana, Ukraine, 2005. – P. 528 – 530.
- [33] Марков Г. Т. Электродинамика и распространение радиоволн: учеб. пособие для вузов. / Г. Т. Марков, Б. М. Петров, Г. П. Грудинский. – М. : Сов. Радио, 1979. – 376 с.
- [34] Миролюбов Н. Н. Методы расчета электростатических полей. / Н. Н. Миролюбов, М. В. Костенко, М. Л. Левинштейн, Н. Н. Тиходеев. – М. : Наука, 1963. – 416 с.
- [35] Бреббия К. Методы граничных элементов. / К. Бреббия, Ж. Теллес, Л. Вроубел: пер. с англ. – М. : Мир, 1987. – 524 с.
- [36] VTI Technologies, Accelerometers and inclinometers [Електронний ресурс]. – Режим доступу: <http://www.vti.fi/technology/accelometers.html>.
- [37] Elwenspoek M. Mechanical microsensors / M. Elwenspoek, R. Wiegerink // Sprinkler - Verlag. – Berlin, Heidelberg, New York. – P. 230 – 236.
- [38] Yu H. G. Lead zirconate titanate MEMS accelerometer using interdigitated electrodes / H. G. Yu [et al.] // Sensors and Actuators A: Physical. – 2003. – Vol. 107, Issue 1. – P. 26 – 35.
- [39] Teslyuk V. Research of output mechanical parameters of the capacitive microaccelerometer construction / V. Teslyuk, R. Zaharyuk, A. Holovaty, A. Romanyuk // CAD in Machinery Design. Implementation and Educational Problems : Proc. of the XIV Ukrainian – Polish Conf. – Polyana, 2006. – P. 113.
- [40] Самарский А. А. Теория разностных схем. / А. А. Самарский. – М. : Наука, 1983. – 432 с.
- [41] Парсел Э. Электричество и магнетизм. Том II. / Э. Парсел : пер. с англ. – М. Наука, Главная редакция физ-мат литературы. – 1975. – 439 с.
- [42] Holmes A. S. Axial - flux permanent magnet machines for micropower generation / A. S. Holmes, H. Guodong, K. R. Pullen // Journal of Microelectromechanical Systems. – 2005. – Vol. 14. – P. 54 – 62.

- [43] Jaephil D. A Low cost magnetic inter - digitated array on a plastic wafer / D. Jaephil, C. Jin-Woo, H. A. Chong // IEEE Transactions on Magnetics. – 2004. – Vol. 40, № 4. – P. 3009 – 3011.
- [44] Lee S. S. Surface micromachined freespace fiber optics switches with integrated microactuators for optical fiber communication systems / S. S. Lee, E. Motamedi, M. C. Wu // Solid - State Sens. Actuators : Tech. Dig. '97 Int. Conf. – Chicago, IL. – 1997. – P. 85.
- [45] Ho C. M. Microelectromechanical systems and fluid flows / C. M. Ho, Y. C. Tai // Ann. Rev. Fluid Mech. – 1998. – Vol. 30. – P. 579 – 612.
- [46] Manz A. Miniaturized chemical analysis systems based on electroosmotic flow / A. Manz // Micro Electromech. Syst.: 10 - th IEEE Workshop. – Nagoya, Japan. – 1997. – P. 14 – 18.
- [47] Абакумов А. А. Полупроводниковые сканеры распределенного магнитного поля / А. А. Абакумов, А. А. Абакумов (мл.), А. И. Галушков, Ю. А. Чаплыгин // Известия вузов. Электроника. – 1997, № 3 – 4. – С. 117 – 125.
- [48] Hirano T. Invar MEMS milliactuator for hard disk drive applications / T. Hirano, L. S. Fan, J. Q. Gao // Micro Electromech. Syst. : 10-th IEEE Workshop – Nagoya, Japan. – 1997. – P. 378–382.
- [49] Бараночников М. Л. Микромагнитоэлектроника: в 2-х томах / М. Л. Бараночников. – М. : ДМК Пресс, 2001. – 544 с.
- [50] Теслюк В. М. Математична модель магнітного інтегрального мікродавача / В. М. Теслюк, К. К. Колесник, П. Ю. Раєвський, Р. Т. Панчак // Вісник Національного університету “Львівська політехніка” : Комп’ютерні системи проектування. Теорія і практика. – 2005. – № 548. – С. 125 – 129.
- [51] Cho H. J. A Bi – directional magnetic microactuator using electroplated permanent magnet arrays / H. J. Cho, C. H. Ahn // IEEE Journal of Microelectromechanical Systems. – 2002. – Vol. 11, № 1. – P. 78 – 84.
- [52] Tilmans H. A Fully – packaged electromagnetic microrelay / H. Tilmans, E. Fullin, H. Ziad [et al.] // IEEE MEMS. – Orlando : USA, 1999. – P. 25 – 30.
- [53] Wagner B, Kreutzer M, Benecke W. Linear and rotational magnetic micromotors fabricated using silicon technology / B. Wagner, M. Kreutzer, W. Benecke // Micro Electro Mechanical Systems : Proc. IEEE. – Travemünde, Germany, 1992. – P. 183 – 189.
- [54] Черняев В. Н. Технология производства интегральных микросхем и микропроцессоров / В. Н. Черняев: учебник для вузов. – М. : Радио и связь, 1987. – 464 с.
- [55] Зенкевич О. Метод конечных элементов в технике. / О. Зенкевич. – М. : Мир, 1975. – 541 с.
- [56] Waanders J. W. Piezoelectric Ceramics - Properties and Applications / J. W. Waanders – 1991. – 38 p.
- [57] Теслюк В. М. Математична модель для обчислення термонапружень та переміщень в актуаторі на базі двохшарової пластини / В. М. Теслюк, М. В. Лобур, К. К. Колісник, П. Ю. Денисюк // Вісник Національного університету “Львівська політехніка” : Комп’ютерна інженерія та інформаційні технології. – 2003. – № 496. – С. 94 – 99.
- [58] Lobur M. Thermomechanical characteristics research of actuators based on the bimetalic plates / M. Lobur, V. Teslyuk, K. Kolisnyk, M. Pereyma // Advanced Computer Systems And Networks: Design And Application : Proc. of the 1 – st Intern. Conf. – Lviv, 2003. – P. 39 – 42.
- [59] Писаренко Г. С. Сопротивление материалов. / Г. С. Писаренко – К. : Техника, 1967. – 791 с.
- [60] Богуш М. В. Пьезоэлектрическое приборостроение: сборник в 3 томах. Т. 3. Пьезоэлектрические датчики для экстремальных условий эксплуатации. / М.В. – Богуш Ростов-на-Дону, Издательство СКНЦ ВШ, 2006. – 346 с.
- [61] Богуш М. В. Проектирование пьезоэлектрических датчиков с использованием конечно-элементных математических моделей / М. В. Богуш // Приборы – 2007. – № 12. – С. 30 – 38.

- [62] Robertson R. G. A Survey of micro-actuator technologies for future spacecraft missions / R. G. Robertson, J. D. Busch // The Journal of The British Interplanetary Society. – 1996. – Vol. 49. – P. 129 – 138.
- [63] David L. Churchill Strain energy harvesting for wireless sensor networks. In Smart Structures and Materials 2003: Smart Electronics, MEMS, BioMEMS, and Nanotechnology / David L. Churchill, Christopher P. Townsend, Steven W. Arms // Proceedings of SPIE. – 2003. – Vol. 5055. – P. 319 – 327.
- [64] Lesieutre G. K. Damping as a result of piezoelectric energy harvesting / G. K. Lesieutre, G. A. Ottman, H. F. Hofmann // Journal of Sound and Vibration. – 2004., January. – Vol. 269. – P. 991 – 1001.
- [65] Теслюк В.М. Побудова множини можливих рішень з допомогою І - АБО – дерева при вирішенні задач структурного синтезу на системному рівні проектування МЕМС // Науково-технічний журнал "Інформаційні технології і системи". – Львів. – 2006. - Том 9, № 1. – С. 101- 108.
- [66] Lab-on-a-Chip and Microarrays for Post-Genome Applications // Cambridge Helthtech Institute's Fourth Annual. – Zurich. – 2002. – 608 p.
- [67] Корляков А.В., Крапивина Е.В., Лучинин В.В. Физико-топологические основы микрофлюидных чипов // Петербургский журнал электроники. – 2001. - № 4. – С.43 – 53.
- [68] Зимина Т.М., Лучинин В.В., Крапивина Е.В., Ресин А.С. Микросистемная техника и проблемы биомедицинского анализа // Микросистемная техника. – 2000.- № 2. – С.37 - 42.
- [69] Kim D. S., Lee S. H., Ahn C. H., Lee J. Y. and Kwon T. H.. Disposable Integrated Microfluidic Biochip for Blood Typing by Plastic Microinjection Molding // Lab Chip. - Vol. 6. – 2006. - P. 794-802.
- [70] Ahn C. H., Choi J. -W., Beaucage G., Nevin J., Lee J. -B., Puntambekar A., and Lee J. Y. Disposable Smart Lab on a Chip for Point-of-Care Clinical Diagnostics // Proc. of the IEEE, Special Issue on Biomedical Applications for MEMS and Microfluidics. - 2004. - Vol. 92. - P. 154 – 173.
- [71] Kwon J.W., Kamal-Bahl S. and Kim E.S. Film Transfer and Bonding Technique to Cover Lab on a Chip // IEEE Intern. Conf. on Solid-State Sensors and Actuators(Transducers '05). - Seoul, Korea. - June 5-9, 2005. - P. 940-943.

РОЗДІЛ 5.

ОСНОВИ ЗАДАЧ БАГАТОКРИТЕРІАЛЬНОЇ ОПТИМІЗАЦІЇ

5.1. Основні елементи теорії оптимізації

Під терміном “оптимізація”, як правило, в техніці розуміють процес, який дає можливість отримати найкраще рішення в заданих умовах. Слід зазначити що “найкраще рішення” деякої задачі з точки зору певної особи, яка визначає, що це є саме “найкраще рішення”, чи по іншому його називають оптимальне. З точки зору іншої особи, це рішення може бути вже не “найкраще”, а близьке до оптимального. Тому у визначенні “найкращого рішення” присутній певний суб’єктивізм, який, зрозуміло, що по можливості в процесі розв’язання оптимізаційних задач бажано усунути.

Відповідно, в процесі розв’язання складних технічних оптимізаційних задач, оцінку “найкращого рішення” має приймати проектувальник (інженер), яка має значний досвід в цій області чи колектив досвідчених науковців.

Наступним моментом вище наведеного визначення є те, що це отримане оптимальне рішення “в заданих умовах”. В інших умовах отриманий результат вже може і не бути найкращим, оптимальним.

Загалом існує багато визначень терміну “оптимізації”, але усі вони містять вищезазначені елементи, і можуть відрізнятися врахуванням лише специфіки та особливості області застосування.

Зокрема, деякі з них:

Оптиміза́ція — 1. процес вибору найкращого варіанта з можливих. 2. Процес приведення системи до найкращого (оптимального) стану [1].

Оптиміза́цією (від латів. *optimus* - найкраще) називається процес знаходження екстремуму (максимуму або мінімуму) певної функції або вибору найкращого (оптимального) варіанту з множини можливих.

Оптиміза́ція (від латів. *optimus* - найкращий) — процес надання будь-чому найвигідніших характеристик, співвідношень (наприклад, оптимізація виробничих процесів і виробництва)[2].

З досвіду відомо що, в більшості випадків в процесі розв'язання практичних оптимізаційних задач вдається лише покращити вже існуюче рішення, а не знайти “найкраще”. Відповідно, необхідно ставитися до цього з розумінням, і це пов'язано з тим, що процес оптимізації є надзвичайно складним на який впливають різні за природою параметри і кількість яких є надзвичайно велика. Більше того, досить часто, ми не знаємо чи існує те “найкраще рішення” взагалі в даній області зміни проектних параметрів (задані умови) і де його шукати.

Перш ніж приступити до обговорення питань оптимізації, введемо ряд визначень.

Проектні параметри (шукані змінні). Цим терміном позначають незалежні змінні параметри, які повністю і однозначно визначають вирішувану задачу проектування.

Проектні параметри – невідомі величини, значення яких обчислюються в процесі оптимізації. Як проектні параметри можуть бути будь-які основні або похідні величини, які призначені для кількісного опису об'єкта проектування. Так, це можуть бути невідомі значення об'єму оперативної пам'яті, тактова частота мікропроцесора, довжини мікродавача, маса чутливого елемента мікродавача, температура, тощо. Число проектних параметрів характеризує ступінь складності даного завдання проектування деякого технічного процесу чи об'єкту. Звичайне число проектних параметрів позначають через n , а самі проектні параметри через x з відповідними індексами. Таким чином n проектних параметрів певного завдання позначатимемо через

$$x_1, x_2, x_3, \dots, x_n.$$

Додаткову інформацію про класифікація та особливості проектних параметрів можна знайти в роботах [3, 4].

В процесі вивчення та формалізації оптимізаційних задач оперують такими складовими, як критерій оптимізації (КО) та цільова функція (ЦФ).

Критерій оптимізації [4] – це є певна властивість об'єкта проектування. (**Критерій оптимальності** – показник чи система показників якості роботи

деякої системи, значення якої має бути мінімізовано (максимізовано) [5])

Для прикладу: вартість виготовлення мікродавача, максимальна швидкість автомобіля, максимальна потужність підсилювача низької частоти, швидкодія мікроконтролера та ін. Визначає критерій оптимізації, як правило, інженер чи науковець, який володіє знаннями в певній галузі та має досвід. В кожній оптимізаційній задачі необхідно знайти найбільше чи найменше значення КО.

Для того, щоб оперувати критерієм оптимізації в процесі розв'язання оптимізаційної задачі, необхідно мати справу з його кількісною оцінкою, а не з якісною. Тому можна сказати так, що формалізують КО, тобто записують в математичній формі залежність КО від проектних параметрів. Відповідно, даний запис називається **цільовою функцією**.

(Цільова функція, функція цілі – функція, найбільше чи найменше значення якої шукається в задачах математичного програмування з врахуванням наявних обмежень [5]).

В процесі розв'язання оптимізаційної задачі цільова функція дає змогу кількісно порівняти два чи більше альтернативних рішення. З математичної точки зору цільова функція описує деяку $(n+1)$ - вимірну поверхню. Її значення визначається проектними параметрами

$$f = f(x_1, x_2, \dots, x_n).$$

Більше інформації про критерії оптимальності та цільові функції можна почерпнути з робіт [6 - 9].

Прикладами цільових функцій, які часто зустрічаються в інженерній практиці, можуть бути математичний вираз для визначення вартості виробу, його ваги, міцності, ККД та інші. У випадку, коли цільова функція містить тільки один проектний параметр, то цільову функцію можна представити з допомогою кривої на площині. Якщо проектних параметрів два, то цільова функція зображуватиметься поверхнею в просторі трьох змінних. При трьох і більше проектних параметрах поверхні, що задаються цільовою функцією, називаються *гіперповерхнями* і не піддаються зображенню звичайними засобами. Топологічні властивості поверхні цільової функції відіграють велику роль в процесі оптимізації, оскільки від них залежить вибір найбільш

ефективного методу пошуку оптимальних рішень.

Цільова функція в ряді випадків може приймати найнесподіваніші форми. Наприклад, її не завжди вдається виразити в математичній формі, в інших випадках вона може бути кусково-гладкою функцією. Для задання цільової функції іноді може знадобитися таблиця технічних даних (наприклад таблиця стану водяної пари) або може знадобитися провести експеримент. В ряді випадків, проектні параметри приймають тільки цілі значення. Прикладом може бути кількість комірок пам'яті персонального комп'ютера чи кількість мікропроцесорів. Іноді проектні параметри мають тільки два значення - так чи ні. Якісні параметри, такі як задоволення, яке отримує покупець, що придбав виріб, надійність, естетичність, теж можливо враховувати в процесі оптимізації, хоча їх складно охарактеризувати кількісно. Проте, в якому б вигляді не була представлена цільова функція, вона має бути однозначною функцією проектних параметрів.

В ряді оптимізаційних задач потрібне введення більше однієї цільової функції. Іноді одна з них може виявитися несумісною з іншою. Прикладом служить проектування ноутбуків, коли одночасно потрібно забезпечити максимальну потужність, мінімальну вагу та мінімальну вартість. В таких випадках розробник має ввести систему пріоритетів і поставити у відповідність кожній цільовій функції деякий безрозмірний множник. В результаті з'являється так звана "функція компромісу", що дає змогу в процесі оптимізації користуватися однією складною цільовою функцією (в даному випадку розробник має справу з задачею багатокритеріальної оптимізації оскільки присутні декілька критеріїв оптимізації).

Якщо порівняти між собою КО і ЦФ, то можна стверджувати, що критерій оптимальності визначає якісну оцінку якоїсь властивості об'єкта проектування, або його властивість чи властивості, а цільова функція – кількісну оцінку критерію чи групи критеріїв оптимізації.

Пошук мінімуму і максимуму. Одні алгоритми оптимізації пристосовані для пошуку максимуму, інші – для пошуку мінімуму. Проте, незалежно від типу розв'язуваної задачі на екстремум, можна користуватися одним і тим же

алгоритмом, оскільки задачу мінімізації можна легко перетворити на задачу пошуку максимуму, помінявши знак цільової функції на зворотний. Чим, на практиці, досить часто користуються проєктувальники в процесі проєктування технічних пристроїв.

Множина допустимих рішень. Так називається область, визначена всіма n проектними параметрами. Простір рішення не такий великий, як може здаватися на перший погляд, оскільки він, зазвичай, обмежений рядом умов, пов'язаних з фізичною суттю завдання. Обмеження можуть бути такими суттєвими, що задача не матиме жодного задовільного розв'язання. Слід зазначити, що дуже часто у зв'язку з обмеженнями оптимальне значення цільової функції досягається на одній з меж області множини допустимих розв'язань задачі.

Локальний оптимум. Так називається точка простору рішень, в якій цільова функція має найбільше значення в порівнянні з її значеннями в усіх інших точках її околу.

Досить часто простір проєктування містить багато локальних оптимумів і слід дотримуватися обережності, щоб не прийняти перший з них за оптимальне розв'язання задачі.

Глобальний оптимум. Глобальний оптимум – це оптимальне рішення для всієї множини допустимих рішень. Воно краще за всі інші рішення, відповідні локальному оптимуму, і саме його шукає розробник. Можливий випадок декількох рівних глобальних оптимумів, розташованих в різних частинах простору проєктування.

5.2. Особливості об'єкта оптимізації

Для більш повного уявлення про оптимізаційні задачі, зупинимося докладніше на характеристиках об'єкта оптимізації і сукупності даних, необхідних для оптимізації об'єкта.

Об'єкти оптимізації можна класифікувати по ряду ознак. До таких ознак відносяться:

- число оптимізованих параметрів об'єкта;
- число екстремумів характеристики об'єкта, використаної як показник якості;
- обсяг апіорної інформації про об'єкт;
- спосіб математичного опису об'єкта.

По числу змінних параметрів розрізняють одно- і багатопараметричні об'єкти. В залежності від кількості екстремумів, об'єкти поділяються на одноекстремальні і багатоекстремальні, причому в останньому випадку, оптимізаційна задача зводиться до пошуку глобального екстремуму, тобто мінімального мінімуму і максимального максимуму.

В залежності від обсягу апіорної інформації, можуть бути екстремальні об'єкти, для яких існує математичний опис, і залежність показника якості Q від параметрів оптимізації X відома. Для таких об'єктів є достатній обсяг апіорної інформації. Існує також великий клас об'єктів, для яких немає ніякого математичного опису. Малий обсяг апіорної інформації про подібні об'єкти послужив приводом називати їх об'єктами типу "чорний ящик".

Сукупність параметрів, які оптимізуються, утворює вектор параметрів об'єкта оптимізації і характеризує вид оптимізаційної задачі. Якщо число параметрів, які оптимізуються, більше одиниці ($n > 1$), то задача відноситься до багатопараметричних, а при $n = 1$ вона переходить в одно параметричну оптимізаційну задачу.

Сукупність показників якості утворить вектор показників якості об'єкта $F = \{f_1, \dots, f_m\}$.

При необхідності характеризувати об'єкт групою показників якості, задача класифікується як багатокритеріальна чи векторна, якщо ж для оптимізації обраний лише один показник якості, то задача переходить в однокритеріальну чи скалярну.

5.3. Постановка оптимізаційної задачі

В процесі побудови оптимізаційної задачі (ОЗ) необхідно інженерові чи науковцеві вирішити такі завдання:

- 1) Вибір та визначення критерію оптимізації.
- 2) Визначити проектні параметри.
- 3) Побудувати цільову функцію (модель залежності між КО та проектними параметрами).
- 4) Визначити обмеження.
- 5) Вибір методу розв'язання оптимізаційної задачі.

Надалі будемо вважати, що ОЗ сформульована, якщо вона містить усі вище перераховані елементи. В ряді випадків, у процесі автоматизованого проектування деякого технічного об'єкта, КО можна отримати з технічного завдання, де записані вимоги до приладу чи технологічного процесу його виготовлення. Проблеми виникають в тих випадках, коли КО є багато і необхідно вибрати з них один, або декілька.

Не менш важливим питанням є визначення переліку множини проектних параметрів. Кількість проектних параметрів є необмеженою. В даному випадку, йде мова про параметри, які мають суттєвий вплив на критерій оптимальності (тобто якусь властивість об'єкта проектування). Тобто, на практиці необхідно відкинути ті проектні параметри, вплив яких виражається значеннями на рівні похибки вихідного параметра.

Дане питання детально розглядається в теорії побудови математичних моделей об'єкта проектування, лишень з тою відмінністю, що замість вихідних параметрів ми оперуємо критерієм оптимальності [4].

Дещо інша ситуація з етапом побудови цільової функції. Все дуже красиво виглядає на лекціях, коли в якості прикладу береться стандартна задача і без проблем будується ЦФ. На практиці, під час проектування технічного пристрою, ще невідомо остаточно його структуру, параметри елементів структури, тощо. Тому побудувати ЦФ надзвичайно складно. Тим більше побудувати адекватну модель залежності вихідного параметра від вхідних, а на

основі моделі – ЦФ. Інколи взагалі така залежність невідома, або побудувати математичну залежність КО від проектних параметрів неможливо. Тому, необхідно володіти певним досвідом у цій галузі і ставитись до цього етапу надзвичайно серйозно. Оскільки, від адекватності розробленої математичної моделі, залежності між критерієм оптимальності та проектними параметрами – залежить точність отриманих результатів оптимізації.

Сукупність обмежень відіграє дуже важливу роль при постановці і розв'язання оптимізаційної задачі. Найбільш часто зустрічаються обмеження виду рівності ($x_i = x_{i0}$) чи нерівності ($x_{imin} \leq x_i \leq x_{imax}$). Обмеження накладаються на змінні проектні параметри, а також на показники якості. Зміст цього етапу полягає в тому, що часто якість системи характеризується не одним, а групою показників якості, тому якщо система оптимізується по одному показнику якості, то інші можуть досягти такої величини. Отже, якщо обраний якийсь параметр системи як критерій якості, то на інші показники якості і змінні параметри накладаються обмеження. Якщо в задачі багатокритеріальної (векторної) оптимізації перевести частину показників якості в розряд обмежень, то можна її звести до однокритеріальної (скалярної) задачі.

Четвертий етап, в більшості випадків не представляє складності, оскільки обмеження визначаються вимогами технічного завдання на проектування.

Завершальними етапами розв'язання оптимізаційної задачі є вибір методу оптимізації і сама оптимізація, тобто знаходження оптимальних значень проектних параметрів (вхідних змінних), при яких цільова функція (якості) досягає мінімуму чи максимуму.

Вибір методу оптимізації залежить, в основному, від виду цільової функції, що у свою чергу визначається особливостями об'єкта оптимізації: його складністю, структурою, математичним описом об'єкта, наявністю апріорних даних про об'єкт проектування.

Найбільш повно розроблені методи оптимізації, які отримали назву "методи математичного програмування". До них відносяться методи лінійного, геометричного, випуклого, нелінійного, стохастичного програмування.

Для застосування того чи іншого методу необхідно, щоб цільова функція відповідала певним вимогам, а саме: її аналітичний вираз повинен бути певного виду.

Варто мати на увазі, що структура вираження цільової функції цілком залежить від особливостей математичного опису об'єкта оптимізації, тому що функція якості знаходиться із математичного опису об'єкта (його математичної моделі).

Як раніше вказувалося, опис об'єкта може бути представлено або у вигляді аналітичних співвідношень, або у вигляді алгоритмів. Остання форма опису практикується при описі складних об'єктів, коли доцільне застосування методів машинного моделювання. Тому не завжди вдається отримати цільову функцію у формі явної аналітичної залежності.

Усе це ускладнює в багатьох випадках застосування добре розроблених методів математичного програмування.

Для розв'язання багатопараметричних оптимізаційних задач при алгоритмічному методі опису об'єктів проектування можуть бути використані пошукові методи оптимізації.

Отже, в якості прикладу наведемо формулювання однокритеріальної оптимізаційної задачі.

Необхідно знайти такі значення проектних параметрів x_1, x_2, x_3, x_4 та x_5 об'єкта дослідження, для яких цільова функція досягає максимального значення

$$\max f(x_1, x_2, x_3, x_4, x_5),$$

при цьому виконуються такі обмеження нерівності та рівності:

$$g_1(x_1, x_2, x_3, x_4, x_5) \geq 10,$$

$$g_2(x_1, x_2, x_3, x_4, x_5) \leq 8,$$

$$h_1(x_1, x_2, x_3, x_4, x_5) = 12,$$

$$h_2(x_1, x_2, x_3, x_4, x_5) = 3,$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0.$$

В наведеному прикладі кількість проектних параметрів рівна п'яти. При чому їхні значення можуть бути більшими, або рівними нулю. Два обмеження у формі рівностей і два обмеження у формі нерівностей.

5.4. Причини багатокритеріальності в задачах оптимального проектування МЕМС

Майже будь-яке складне технічне завдання прийняття рішення є багатокритерійним, оскільки при виборі найкращого варіанту доводиться враховувати багато різних вимог, і серед цих критеріїв зустрічаються такі, що суперечать один одному. Проте, майже всі математичні методи оптимізації призначені для знаходження екстремуму однієї функції – тобто для знаходження оптимального розв'язання за одним критерієм. Тому найчастіше намагаються звести багатоцільове завдання до одноцільового. Ця процедура в більшості випадків призводить до серйозного спотворення сутності проблеми і, отже, до невиправданої заміни одного завдання іншим.

Якщо при розв'язанні одноцільових задач методологічних проблем не виникає, а можливі тільки обчислювальні труднощі, то інакше йде справа з багатокритерійними задачами. Тут основні нюанси пов'язані з наступною проблемою: що слід вважати якнайкращою альтернативою в завданні з декількома цільовими функціями, які суперечливі між собою, і досягають максимуму в різних точках множини альтернатив?

Цілі, багатокритеріальної задачі, можуть знаходитися одна з одною в наступних відносинах:

1. Цілі взаємно нейтральні. В такій ситуації, система може стосовно окремих цілей характеризуватися і розглядатися незалежно.

2. Цілі кооперуються. Тут, як правило, систему вдається розглянути стосовно однієї мети, а інші досягаються одночасно.

3. Цілі конкурують. В цьому випадку одну з цілей можна досягти лише за рахунок іншої.

Якщо цілі частково нейтральні, частково кооперовані і частково конкурують між собою, то завдання формулюється таким чином, що потрібно брати до уваги тільки конкуруючі цілі. Розгляд нейтральних або кооперативних цілей не представляє особливих труднощів, так що проблеми, орієнтовані на множину цілей, перш за все повинні бути розглянуті в частині конкуруючих цілей, якщо всі вони разом не можуть бути виражені одновимірним параметром.

Найбільш загальною математичною моделлю прийняття оптимального рішення є завдання багатопараметричної оптимізації. Наведемо декілька причин, які приводять до багатокритеріальних завдань.

1. Однією з причин, що приводить до багатокритеріальності, є *множина технічних вимог*, які пред'являються до характеристик проектного пристрою.

2. Наступною причиною багатокритеріальності є *необхідність забезпечення оптимальності проектного пристрою за різних умов його функціонування*, тобто забезпечення екстремальних значень критерію оптимальності при невизначеності умов, в яких доводиться працювати пристрою.

3. В тих випадках, коли *проектований пристрій складається з декількох взаємозв'язаних вузлів і блоків*, оптимальність всього пристрою визначається ефективністю і якістю його окремих частин, кожна з яких може бути охарактеризована, принаймні, хоч би одним власним критерієм оптимальності $f_i(\vec{x})$.

В цьому випадку, функціонування всього пристрою можна вважати якнайкращим, якщо за рахунок вибору керованих параметрів $\vec{x}$ забезпечуються екстремальні значення всіх приватних критеріїв оптимальності як основних підцілей однієї загальної мети проектування.

5.5. Постановка задачі багатокритеріальної оптимізації

В загальному випадку задачами багатокритеріальної оптимізації прийнято вважати такі задачі, де існує декілька критеріїв оптимізації.

Одне з можливих визначень наведено наступне [6, 7]:

Багатокритеріальна оптимізація або програмування (англ. Multi-objective optimization) — це процес одночасної оптимізації двох або більше конфліктуючих цільових функцій в заданій області визначення.

Багатокритеріальності проблема — задача вибору рішення при наявності декількох функцій цілі [5].

Задача багатокритеріальної оптимізації зустрічаються в багатьох галузях науки та техніки.

Постановка багатокритеріальної оптимізації вже оперує з такими елементами, як цільові функції, а не критерії.

В загальному випадку задачу багатокритеріальної оптимізації можна сформулювати наступним чином [8, 9].

Знайти такі значення проектних параметрів, для яких забезпечуються екстремальні значення цільових функцій:

$$\min(\max) f_i(\vec{x}), (i = 1, n), j = 1, m, \quad (5.1)$$

при виконанні наступної системи обмежень

$$g_k(\vec{X}) \leq \alpha_k, (k = 1, l), \quad (5.2)$$

$$h_m(\vec{X}) = \beta_m, (m = 1, j),$$

$$b_1 \leq x_j \leq b_m.$$

де вектор розв'язків $\vec{x} = (x_1, x_2, \dots, x_n)^T$ належать до не порожньої області визначення D .

Отже, задача багатокритеріальної оптимізації полягає у пошуку вектора цільових змінних, який задовольняє накладеним обмеженням та оптимізує векторну функцію, елементи якої відповідають цільовим функціям. Ці функції утворюють математичний опис критерію задовільності та, зазвичай, взаємно

конфліктують. Звідси, «оптимізувати» означає знайти такий розв'язок, за якого значення цільових функцій були б прийнятними для особи, що поставила задачу [9].

5.6. Контрольні запитання до розділу

1. Що таке критерій оптимізації?
2. Що Ви розумієте під цільовою функцією?
3. Яка задача називається задачею багатокритеріальної оптимізації?
4. Які види критеріїв оптимізації Ви знаєте?
5. Який критерій оптимізації називається векторним?
6. Який критерій оптимізації називається скалярним?
7. Що таке проектні параметри?
8. Що таке локальний оптимум?
9. Що таке глобальний оптимум?
10. Які причини багатокритеріальності Ви знаєте?
11. Наведіть приклад багатокритеріальної оптимізаційної задачі?
12. Які методи розв'язання задач багатокритеріальної оптимізації Ви знаєте?
13. На основі якої інформації визначаються проектні параметри?
14. На основі якої інформації визначаються критерії оптимізації?
15. Яка різниця між критерієм оптимізації та цільовою функцією?
16. Які основні кроки включає процес розв'язання оптимізаційних задач?
17. Які види обмежень Ви знаєте?
18. Наведіть приклад багатокритеріальної оптимізаційної задачі?
19. Наведіть приклад взаємно нейтральних критеріїв оптимальності?
20. Наведіть приклад критеріїв оптимальності, які кооперуються в процесі рішення деякої задачі?
21. Наведіть приклад критеріїв оптимальності, які взаємнонейтральні в процесі рішення задачі проектування?

5.7. Список використаної літератури до п'ятого розділу

[1] Сучасний тлумачний словник української мови: 60 000 слів / За заг. Ред. д-ра філол. наук,

- проф. В.В. Дубічинського. – Х. : ВД «ШКОЛА», 2007. – 832 с.
- [2] Словник іншомовних слів. За редакцією член-кореспондента АН УРСР О.С. Мельничука. – Київ, Головна редакція Української радянської енциклопедії, 1977. – 776 с.
- [3] <http://wiki.tntu.edu.ua/параметри> оптимізації
- [4] Теслюк В.М. Математичне моделювання в САПР: Ч.1. Конспект лекцій з курсу “Математичне моделювання в САПР” для студентів базового напрямку “Комп’ютерні науки”. – Львів: Видавництво Національного університету “Львівська політехніка”, 2009. – 64 с.
- [5] Словарь по кибернетике: Св. 2000 ст./Под ред. В.С. Михалевича, 2-е изд.-К.: Гл.ред.УСЭ им.М.П.Бажана, 1989. – 751с.
- [6] Steuer, R.E. Multiple Criteria Optimization: Theory, Computations, and Application. — New York : John Wiley & Sons, Inc , 1986. ISBN 047188846X.
- [7] Sawaragi, Y. Theory of Multiobjective Optimization (vol. 176 of Mathematics in Science and Engineering). — Orlando, FL : Academic Press Inc , 1985. ISBN 0126203709.
- [8] Jürgen Branke, Kalyanmoy Deb, Kaisa Miettinen та Roman Slowinski Multiobjective Optimization: Interactive and Evolutionary Approaches (Lecture Notes in Computer Science). — Springer, 2008. ISBN 3-540-88907-8.
- [9] Лотов А.В., Поспелова И.И. Многокритериальные задачи принятия решений: Уч. пособие. М.: Издат. отдел ф-та ВМиК МГУ, МАКС Пресс, 2008. – 197 с.
- [10] Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач.- М.: Наука, 1982.- 256 с.

РОЗДІЛ 6.

МЕТОДИ РОЗВ'ЯЗАННЯ БАГАТОКРИТЕРІАЛЬНИХ ЗАДАЧ НА ОСНОВІ ОДНОКРИТЕРІАЛЬНОЇ ОПТИМІЗАЦІЇ

В п'ятому розділі посібника було зазначено, що задачі багатокритеріальної оптимізації (ЗБО) – це такі задачі оптимізації, в яких використовується не один, а кілька критеріїв оптимальності. На практиці такі задачі виникають, коли проєктований об'єкт (чи вихідний параметр об'єкта розроблення) не може бути описаний однокритеріальною залежністю, або в тій ситуації, коли об'єднати окремі критерії в єдиний узагальнений критерій не представляється можливим. В ряді випадків таке об'єднання критеріїв у єдиний узагальнений критерій застосовується і воно буде розглянуто в наступному розділі даної роботи. Але це об'єднання, як правило, буває формальним, чи як по іншому його називають – штучним, яке не несе ніякого фізичного навантаження.

З математичної точки зору не існує ідеального методу або способу розв'язання таких задач. Кожний з них має свої певні переваги та недоліки і область застосування. В цьому розділі розглянемо методи, які можна звести до одного критерію та методи, які дають змогу попередньо розв'язати задачі оптимізації як однокритеріальні з наступними перетвореннями, що дають змогу вважати отримані результати, як результати розв'язання ЗБО. Отже розглянемо більш детально такі методи розв'язання багатокритеріальних задач оптимізації, а саме [1 - 12]:

- метод головної компоненти (МГК);
- метод комплексного критерію;
- лексикографічний метод (ЛМ);
- метод поступок (МП);
- метод ідеальної точки (МІТ);
- метод умовного центра мас (МУЦМ).

6.1. Метод головної компоненти

Ідея методу головної компоненти полягає в тому, що узагальнений критерій (часом його називають критерій якості) зв'язується з одним із критеріїв, що вибраний в ролі основного (головного), а на основі інших критеріїв будують обмеження. В такому випадку отримують задачу однокритеріальної оптимізації для розв'язання якої використовують відповідні методи на основі виду та особливостей цільової функції обмеження. Блок-схема алгоритму роботи цього методу наведена на рис.6.1. Наведена блок схема включає такі основні кроки, як вибір основного критерію оптимальності, побудова обмежень на основі інших критеріїв та, відповідно, розв'язання отриманої однокритеріальної задачі.

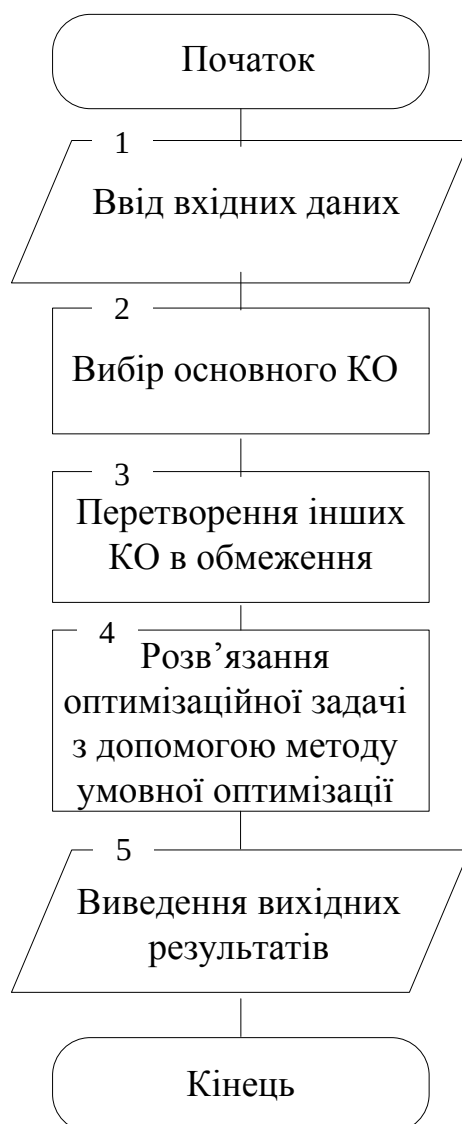


Рис.6.1. Блок-схема алгоритму застосування методу головної компоненти

В загальному випадку припустимо, що нам необхідно в процесі автоматизованого проектування деякого складного об'єкта розв'язати ЗБО, яка має n критеріїв оптимальності:

$$F = \{\max f_1(\bar{X}), \max f_2(\bar{X}), \dots, \min f_{n-1}(\bar{X}), \min f_n(\bar{X})\}, \quad (6.1)$$

де $f_1(\bar{X})$ – перша цільова функція, яка описує, для прикладу, продуктивність деякого об'єкта, $f_2(\bar{X})$ – друга цільова функція, яка описує надійність цього технічного пристрою, ..., $f_n(\bar{X})$ – n -а цільова функція, яка описує його собівартість.

З обмеженнями наступного виду

$$g_k(\bar{X}) \leq \alpha_k, \quad (k=1, l), \quad (6.2)$$

$$h_m(\bar{X}) = \beta_m, \quad (m=1, j),$$

де $g_k(\bar{X}) \leq \alpha_k$ – k -те обмеження нерівності, $h_m(\bar{X}) = \beta_m$ – m -те обмеження рівності, α_k та β_m – деякі константи, l – кількість обмежень нерівностей, j – кількість обмежень рівностей.

Припустимо, що на основі деякого методу розробник визначив, що головним критерієм оптимальності є собівартість його виготовлення ($f_n(\bar{X})$). Відповідно, задача багатокритеріальної оптимізації, яку необхідно розв'язати з допомогою методу головної компоненти, буде сформульована наступним чином:

знайти

$$\min f_n(\bar{X}), \quad (6.3)$$

при виконанні таких обмежень:

$$f_1(\bar{X}) \geq A_1, \quad f_2(\bar{X}) \geq A_2, \quad \dots, \quad f_{n-1}(\bar{X}) \leq A_{n-1}, \quad (6.4)$$

$$g_k(\bar{X}) \leq \alpha_k, \quad (k=1, l),$$

$$h_m(\bar{X}) = \beta_m, \quad (m=1, j),$$

$$A_{<i>} = \langle A_1, A_2, \dots, A_n \rangle,$$

де $A_i (i=1, 2, \dots, n)$ – граничні значення критеріїв оптимальності (константи).

Отже, метод головної компоненти полягає в довільному виборі одного із критеріїв в якості головного, по якому проводиться оптимізація і вибирається рішення. При цьому інші компоненти переводяться в розряд обмежень.

Цей метод простий, наочний і часто застосовується при проектуванні, в машинобудівній практиці, однак принциповим його недоліком є довільність у виборі головного критерію. Можна навести багато прикладів із історії науки і техніки, коли довільний і невірний вибір цього критерію призводив до трагічних наслідків чи, щонайменше, до малоефективних результатів.

В літературі, присвяченій питанню оптимізації програм, в якості головного показника вибирають, наприклад, їх собівартість чи термін експлуатації. При оптимізації металоконструкцій – металоємність. Коли вирішуються технологічні задачі в якості головного критерію. прийнято використовувати, продуктивність праці, продуктивність обладнання.

Для кращого розуміння особливостей застосування цього методу, наведемо приклад ЗБО, яка включає три цільові функції та три обмеження. Отже необхідно знайти такі значення проектних параметрів x_1 , x_2 , x_3 , x_4 та x_5 , для яких цільові функції приймають максимально можливі значення:

$$\begin{aligned}\max f_1(\bar{x}) &= 2x_1 + 3x_2 + x_3 + 2x_4, \\ \max f_2(\bar{x}) &= 3x_1 - x_2, \\ \max f_3(\bar{x}) &= x_1 - 4x_2,\end{aligned}\tag{6.5}$$

при виконанні таких умов:

$$\begin{aligned}x_1 + 2x_2 + x_3 &= 6, \\ 2x_1 + x_2 + x_4 &= 8, \\ -x_1 + x_2 + x_5 &= 4, \\ x_1, x_2, x_3, x_4, x_5 &\geq 0.\end{aligned}\tag{6.6}$$

Припустимо, що головним критерієм є перший, а другий та третій, відповідно, необхідно ввести в список обмежень. Оскільки, в другому та третьому КО шукаємо максимум, то цільові функції мають бути не менше деякої уявної величини (що в процесі проектування може визначатися технічним завданням на виріб).

Для прикладу, в процесі проектування автомобілів, це може бути його максимальна швидкість (тобто більшою може бути, але не меншою деякого значення) у випадку проектування підсилювача звукової частоти – вихідна потужність у випадку розроблення комп'ютера – об'єм пам'яті, та ін.

Повертаючись до нашого прикладу прийнемо, що значення другого КО, має бути більше-рівне чотирьох, а третього – шести. Тоді два додаткові обмеження можна записати у наступній формі:

$$3x_1 - x_2 \geq 4,$$

$$x_1 + 4x_2 \geq 6.$$

Задача, яку в кінцевому випадку необхідно розв'язати, буде включати цільову функцію $\max f_1(\bar{x}) = 2x_1 + 3x_2 + x_3 + 2x_4$ та обмеження з задачі (6.5) плюс обмеження (6.6).

Неозброєним оком видно, що ця задача є задачею лінійного програмування і для розв'язання якої використаємо метод великих чисел.

В процесі застосування М-методу, модифікуємо обмеження, тобто перетворимо 4-е та 5-е обмеження нерівності в рівності, при цьому віднімаючи змінні x_6 та x_7 :

$$3x_1 - x_2 - x_6 = 4, \tag{6.7}$$

$$x_1 + 4x_2 - x_7 = 6.$$

До цих обмежень додамо штрафи R_1 та R_2 .

$$3x_1 - x_2 - x_6 + R_1 = 4, \tag{6.8}$$

$$x_1 + 4x_2 - x_7 + R_2 = 6.$$

Після врахування штрафів у цільовій функції, отримаємо таку оптимізаційну задачу:

$$\begin{aligned} \max f_1(\bar{x}) &= 2x_1 + 3x_2 + x_3 + 2x_4 - MR_1 - MR_2, \\ x_1 + 2x_2 + x_3 &= 6, \end{aligned} \tag{6.9}$$

$$2x_1 + x_2 + x_4 = 8,$$

$$-x_1 + x_2 + x_5 = 4,$$

і додаткові обмеження з введеними штрафами

$$3x_1 - x_2 - x_6 + R_1 = 4,$$

$$x_1 + 4x_2 - x_7 + R_2 = 6,$$

$$x_1, x_2, x_3, x_4, x_5, x_6, x_7, R_1, R_2 \geq 0.$$

Слід зауважити, що штрафи перемножені на велике число і введені у ЦФ зі знаком мінус, бо маємо задачу максимізації.

Надалі модифікуємо ЦФ з (6.9) шляхом введення замість штрафів R_1 та R_2 , вирази, які визначимо з виразів (6.8). Тоді отримаємо:

$$R_1 = 4 - 3x_1 + x_2 + x_6, \quad (6.10)$$

$$R_2 = 6 - x_1 - 4x_2 + x_7.$$

Модифікуємо ЦФ з (6.9) і будемо мати такий вираз:

$$\begin{aligned} \max f_1(\bar{x}) &= 2x_1 + 3x_2 + x_3 + 2x_4 - M(4 - 3x_1 + x_2 + x_6) - M(6 - x_1 - 4x_2 + x_7) = \\ &= 2x_1 + 3Mx_1 + Mx_1 + 3x_2 - Mx_2 + 4Mx_2 + x_3 + 2x_4 - Mx_6 - Mx_7 - 4M - 6M = \quad (6.11) \\ &= (2 + 4M)x_1 + (3 + 3M)x_2 + x_3 + 2x_4 - Mx_6 - Mx_7 - 10M \end{aligned}$$

Приймемо, що $M = 1000$, тоді ЦФ прийме такий вид:

$$\begin{aligned} \max f_1(\bar{x}) &= (2 + 4M)x_1 + (3 + 3M)x_2 + x_3 + 2x_4 - Mx_6 - Mx_7 - 10M = \\ &= 4002x_1 + 3003x_2 + x_3 + 2x_4 - 1000x_6 - 1000x_7 - 10000. \end{aligned}$$

Надалі необхідно розв'язати симплекс-методом задачу ЛП (6.9), де необхідно замінити ЦФ на модифіковану (6.11).

В кінцевому випадку отримуємо такі значення змінних та ЦФ:

$$x_1 = 1,69, \quad x_2 = 1,08, \quad x_3 = 2,15, \quad x_4 = 3,54, \quad x_5 = 4,6,$$

$$f_1(\bar{x}) = -6,15,$$

$$f_1(\bar{x}) = 1,69 * 2 + 1,08 * 3 + 2,15 + 2 * 3,54 = 3,38 + 3,24 + 2,15 + 7,08 = 15,85.$$

Отже, метод головного критерію застосовується в таких задачах, як мінімізація затрат при умові виконання плану по виробництву різних видів продукції, максимізація випуску комплектних наборів при обмеженні на використовувані ресурси та ряд інших.

До переваг МГК можна віднести такі характеристики як простоту та наочність, а його недоліком є певна присутність суб'єктивності в процесі вибору головного критерію (свавілля).

6.2. Лексикографічний метод

Ідея методу ґрунтується на тому, що спершу часткові критерії оптимізації ранжують по їх відносній важливості і поступово розв'язують задачі однокритеріальної оптимізації, починаючи з найважливішого критерію оптимізації.

В загальному випадку алгоритм (рис.6.2) розв'язання ЗБО з використанням лексикографічного методу включає такі кроки:

Крок 0. На попередньому кроці розв'язання ЗБО ранжуємо часткові критерії в порядку спадання їхньої важливості. Пронумерувавши після цього критерії, можна вважати, що перший критерій – найважливіший.

Крок 1. Розв'язати задачу багатокритеріальної оптимізації (знайти такі значення проектних параметрів $\bar{x}$, для яких):

$$\max f_1(\bar{x}),$$

$$g_i(\bar{x}) \geq B_i,$$

$$h_j(\bar{x}) = A_j.$$

і знаходиться максимальне значення критерію $f_1^*(\bar{x}_1^*)$.

Крок 2. Розв'язати задачу багатокритеріальної оптимізації

$$\max f_2(\bar{x})$$

$$g_i(\bar{x}) \geq B_i,$$

$$h_j(\bar{x}) = A_j.$$

$$f_1(\bar{x}) = f_1^*(\bar{x}_1^*)$$

Знайдене розв'язання $\bar{x}_2^*$ максимізує другий критерій, який задовольняє при цьому додатковому обмеженню, при виконанні якого досягається максимум по першому критерію.

Крок k. Розв'язати задачу багатокритеріальної умовної оптимізації

$$\max f_k(\bar{x})$$

$$g_i(\bar{x}) \geq B_i,$$

$$h_j(\bar{x}) = A_j,$$

$$f_1(\bar{x}) = f_1^*(\bar{x}_1^*), f_2(\bar{x}) = f_2^*(\bar{x}_2^*), \dots, f_{k-1}(\bar{x}) = f_{k-1}^*(\bar{x}_{k-1}^*).$$

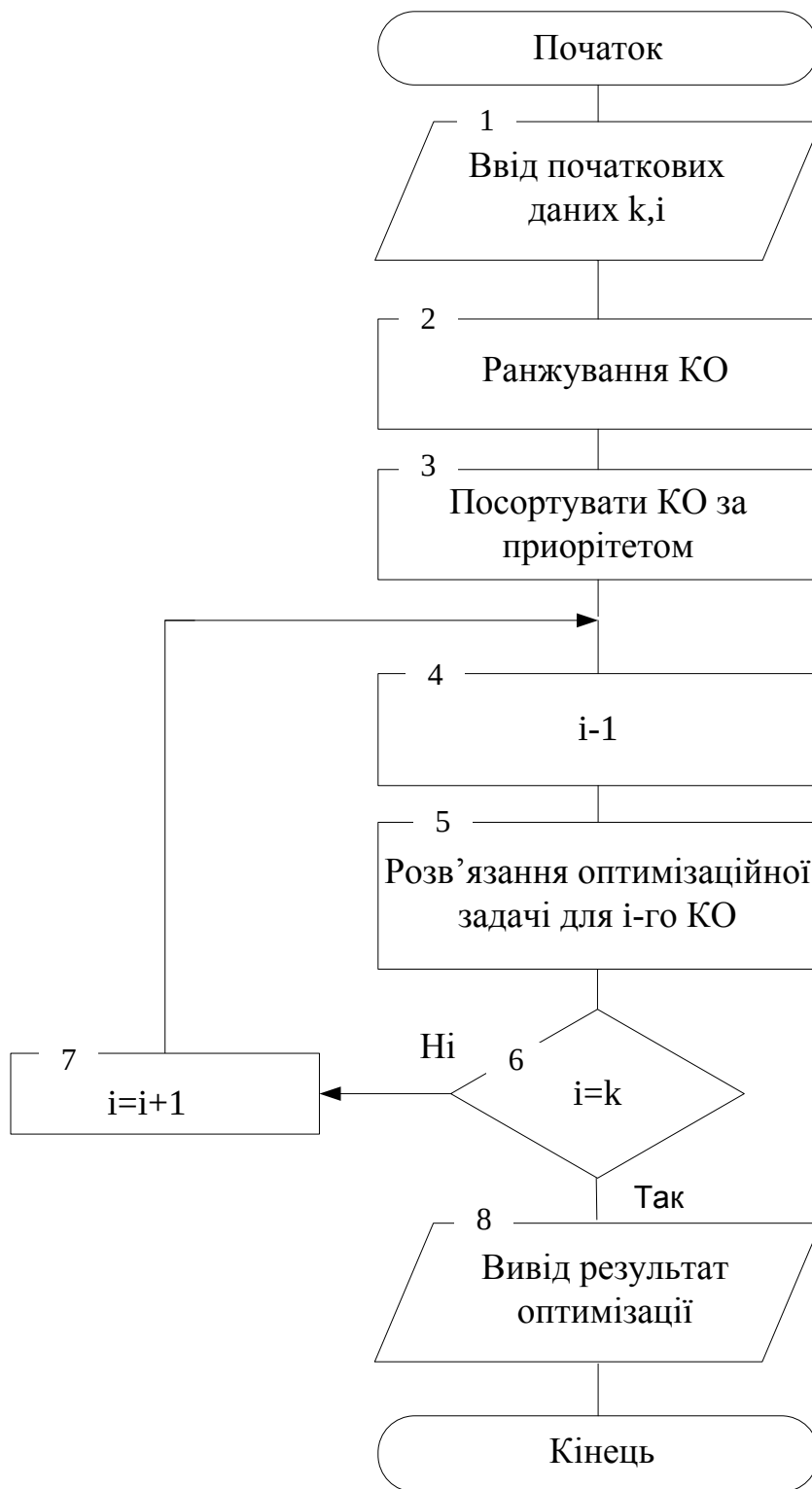


Рис.6.2 Блок-схема алгоритму розв'язання ЗБО з використанням лексикографічного методу

Розв'язання $\bar{x}_k^*$ максимізує k – й критерій оптимальності, одночасно задовольняючи обмеження на значеннях попередніх критеріїв оптимальності.

Ця процедура продовжується до тих пір, поки не буде максимізоване значення останнього із часткових критеріїв, після чого процедура завершується.

В процесі застосування вищенаведеного алгоритму після 1-го, 2-го, ..., $k-1$ -го кроків необхідно цільову функцію перевести в список обмежень, що наведено нижче:

$$f_1(\bar{x}) \geq f_1^* - \text{після 1-го кроку}, \quad (6.12)$$

$$f_2(\bar{x}) \geq f_2^* - \text{після 2-го кроку},$$

...

$$f_{k-1}(\bar{x}) \geq f_{k-1}^* - \text{після } k-1\text{-го кроку},$$

де $f_1^*, f_2^*, \dots, f_{k-1}^*$ - отримане значення цільової функції після розв'язання задачі на першому, другому, ..., $k-1$ -му кроці.

Звернемо увагу на те, що лексикографічний метод не призводить до нескінченної ітераційної процедури, яка зупиняється, коли одержаний результат задовольняє розробника. Навпаки, описана процедура має наперед відоме обмежене число кроків, яке не більше числа кількості часткових критеріїв.

Слід зауважити, що в лексикографічному методі часто можливості вибору не залишається вже після оптимізації по першому критерію, так що процес зразу ж зупиняється. В цьому випадку, задача багатокритеріальної оптимізації виявляється зведеною до однокритеріальної задачі з найбільш важливим критерієм, причому, значеннями інших критеріїв практично нехтують. Якщо ж після оптимізації першого критерію і залишається якась свобода дій, то її може виявитися недостатньо для одержання задовільних значень інших критеріїв оптимальності.

Необхідно додати, що задача ранжування критеріїв оптимальності, за важливістю зовсім не проста. Окрім того, лексикографічна процедура, загалом, некоректна з обчислювальної точки зору, так як малі збурення параметрів початкової задачі можуть призвести до серйозної помилки в результатах. Хоча і

були розроблені стійкі обчислювальні методи розв'язку цієї задачі, використання лексографічної процедури у випадку нестійкості до початкових даних вимагає особливої уваги інженера чи розробника. Теоретичною основою лексографічного методу є лексографічний порядок, який задається на множині векторів. Будучи повним, він тим не менше, не володіє властивістю неперервності, що може привести до несподіваних наслідків.

Для кращого розуміння особливостей та ідеї методу наведемо приклад, а саме: необхідно розв'язати таку задачу багатокритеріальної оптимізації (задача лінійного програмування) з трьома цільовими функціями, трьома обмеженнями у вигляді рівностей і п'ятьма змінними (Див.формули 6.5):

Після проведення операції ранжування, для прикладу, маємо таку ситуацію:

1. $\max f_1(\bar{x}) = 2x_1 + 3x_2 + x_3 + 2x_4$ – найважливіший КО;
2. $\max f_2(\bar{x}) = 3x_1 - x_2$ – менш важливий КО;
3. $\max f_3(\bar{x}) = x_1 - 4x_2$ – найменш важливий КО.

Згідно з вищенаведеним алгоритмом, на першому кроці необхідно розв'язати таку задачу однокритеріальної оптимізації:

$$\begin{aligned} \max f_1(\bar{x}) &= 2x_1 + 3x_2 + x_3 + 2x_4, & (6.13) \\ x_1 + 2x_2 + x_3 &= 6, \\ 2x_1 + x_2 + x_4 &= 8, \\ -x_1 + x_2 + x_5 &= 4, \\ x_1, x_2, x_3, x_4, x_5 &\geq 0. \end{aligned}$$

Застосувавши симплекс – метод до розв'язання задачі (2.13) отримаємо такі результати:

$$\begin{aligned} f_1^{\max} &= f_1^* = 22, & (6.14) \\ x_1 &= 0, \quad x_2 = 0, \quad x_3 = 6, \quad x_4 = 8, \quad x_5 = 4. \end{aligned}$$

Наступний крок алгоритму передбачає введення першого критерію оптимізації в список обмежень (справу ми маємо з цільовою функцією, яка описує цей критерій і дає змогу отримати числове значення оцінки цього

критерію) і розв'язання однокритеріальної оптимізаційної задачі з другим критерієм. Тобто, оптимізаційну задачу наступного виду:

$$\max f_2(\bar{x}) = 3x_1 - x_2, \quad (6.15)$$

$$x_1 + 2x_2 + x_3 = 6,$$

$$2x_1 + x_2 + x_4 = 8,$$

$$-x_1 + x_2 + x_5 = 4,$$

$$2x_1 + 3x_2 + x_3 + 2x_4 \geq 22 \quad - \text{додаткове обмеження на основі } f_1(\bar{x}),$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0.$$

В результаті розв'язання задачі (2.12) отримаємо такі значення:

$$f_2^* = 0, f_1^* = 22,$$

$$x_1 = 0, x_2 = 0, x_3 = 6, x_4 = 8, x_5 = 4.$$

Останній етап розв'язання задачі (6.5) з використанням лексикографічного методу передбачає введення другої цільової функції в список обмежень і розв'язання нижченаведеної оптимізаційної однокритеріальної задачі лінійного програмування:

$$\max f_3(\bar{x}) = x_1 - 4x_2, \quad (6.16)$$

$$x_1 + 2x_2 + x_3 = 6, 2x_1 + x_2 + x_4 = 8, -x_1 + x_2 + x_5 = 4,$$

$$2x_1 + 3x_2 + x_3 + 2x_4 \geq 22 \quad - 1\text{-ше додаткове обмеження на основі } f_1(\bar{x}),$$

$$3x_1 - x_2 \geq 0 \quad - 2\text{-ге додаткове обмеження на основі } f_2(\bar{x}),$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0.$$

Отримані кінцеві результати розв'язання задачі (6.5) наступні:

$$f_3^* = 0, f_2^* = 0, f_1^* = 22, \quad (2.17)$$

$$x_1 = 0, x_2 = 0, x_3 = 6, x_4 = 8, x_5 = 4.$$

Необхідно зауважити, що (підтверджено наведеним прикладом) в лексикографічному методі часто можливості вибору не залишається вже після оптимізації по першому критерію, так що процес зразу же зупиняється, як наслідок використовується цей метод, на практиці, дуже рідко.

Розвитком лексографічного методу є метод поступок.

6.3. Метод поступок

Для задач, у яких критерії не рівнозначні, застосовують інший метод розв'язання, а саме: метод *поступок*. Перш ніж розв'язувати поставлену задачу за методом поступок, необхідно:

Крок 1. Розмістити критерії оптимальності за їхньою значимістю (найважливіший необхідно розмістити першим, потім менш важливий і т.д.).

Крок 2. Відшукати оптимальне значення f_1^* для цільової функції f_1 .

Крок 3. Зробити поступку по першому показнику ефективності, тобто погіршити величину f_1^* до значення $f_1^{**} = k_1 f_1^*$.

Крок 4. Ввести в задачу додаткове обмеження $f_{1i} f_1^{**}$.

Крок 5. Відшукати оптимальне значення f_2^* цільової функції f_2 .

Крок 6. Зробити поступку по другому показнику ефективності, тобто погіршити величину f_2^* до значення $f_2^{**} = k_2 f_2^*$.

Крок 7. Ввести в оптимізаційну задачу додаткове обмеження $f_{2i} = f_2^{**}$.

Крок 8. Нову задачу з двома додатковими обмеженнями розв'язати по третьому показнику ефективності і т.д.

Крок 9. Процес розв'язання задачі закінчується, коли рішення буде отримано по всім показникам. Кінцевий результат і буде найбільш раціональним – отримано оптимальне значення найменш важливого критерію при умові гарантованих значень попередніх показників ефективності.

Приклад блок-схеми алгоритму застосування методу поступок до розв'язання ЗБО наведено на рис.6.3. Особливістю якого є те, що важливість критеріїв оптимальності також визначається на основі знань та досвіду інженера, що також є суб'єктивним фактором.

Необхідно додати основні математичні співвідношення для процесу введення додаткових обмежень після виконання поступки.

$$f_1(\bar{x}) \geq f_1^* - \Delta_1 = f_1^{**} \quad \text{– додаткове обмеження після 1-ї поступки;}$$

$$f_2(\bar{x}) \geq f_2^* - \Delta_2 = f_2^{**} \quad \text{– додаткове обмеження після 2-ї поступки;}$$

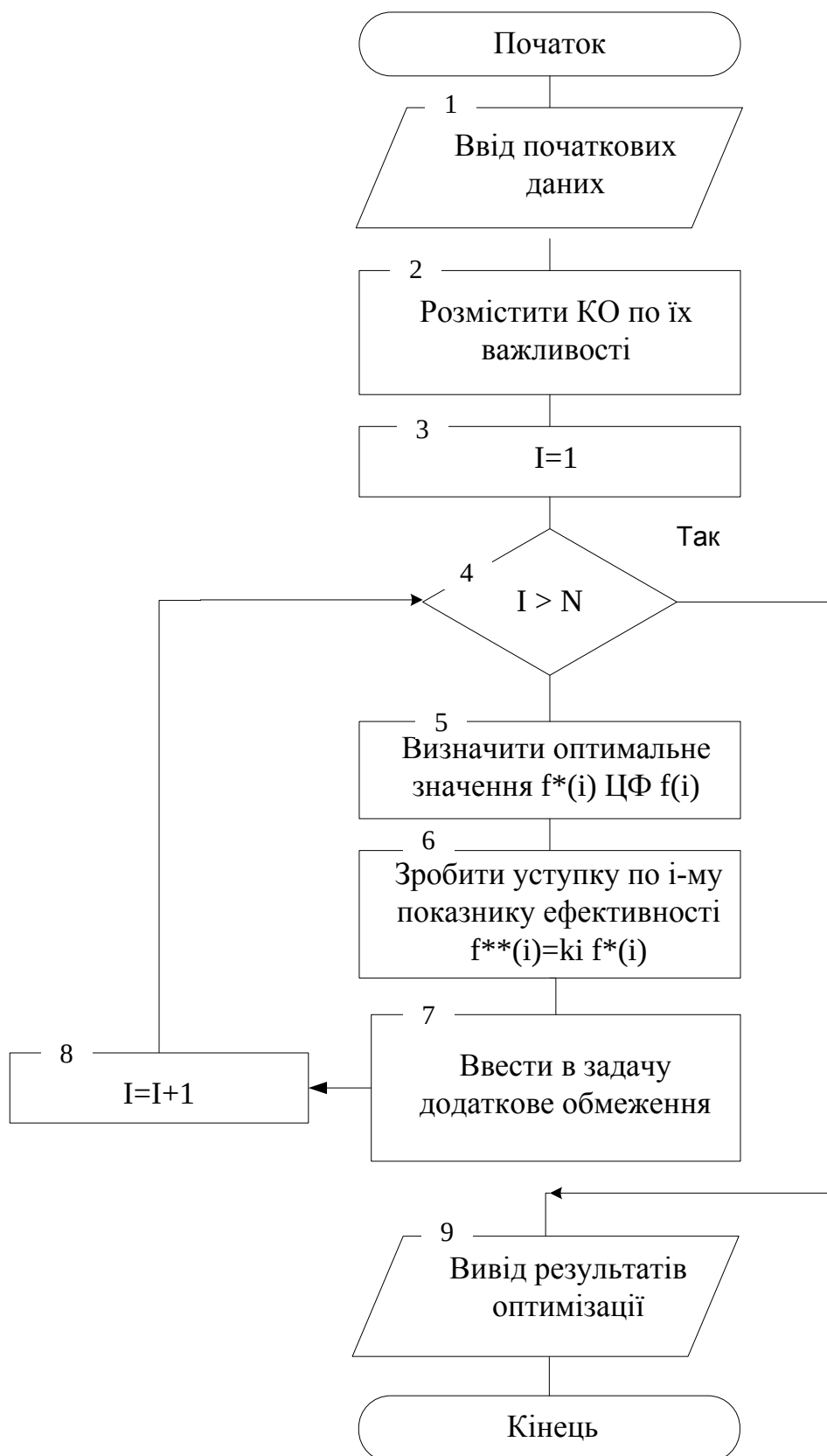


Рис.6.3 Блок-схема алгоритму застосування методу поступок

...

$f_{n-1}(\bar{x}) \geq f_{n-1}^* - \Delta_{n-1} = f_{n-1}^{**}$ – додаткове обмеження після $n-1$ -ї поступки.

Застосуємо метод поступок до розв'язання задачі багатокритеріальної оптимізації (6.5). Будемо вважати, що критерії оптимізації уже розміщені по їх важливості. Найбільш важливий розміщений першим, тобто перший крок алгоритму уже виконали. Розв'язання однокритеріальної оптимізаційної задачі (6.13) ми також вже виконали з першим КО (Крок 2 алгоритму) і розв'язання наведено вище (Див. вирази 6.14). Ці два кроки ми вже виконали в процесі застосування лексикографічного методу до розв'язання ЗБО (6.5). Третій крок передбачає операцію виконання поступки. Для розглядуваної задачі зробимо поступку в розмірі 33%, а саме ($k_1 = 0,33$):

$$f_1^{**} = f_1^* - \Delta_1 = f_1^* - 0,33 * f_1^* = 22 - 0,33 * 22 = 22 - 7,27 = 14,7. \quad (6.18)$$

Введемо в задачу відповідно наступне додаткове обмеження (Крок 4), а саме будемо мати таку форму:

$$2x_1 + 3x_2 + x_3 + 2x_4 \geq 14,7 \text{ – 1-ше додаткове обмеження на основі } f_1(\bar{x}).$$

В даному випадку ми погіршили значення першого критерію на 33%, зменшили значення цільової функції тобто з 22 до 14,7.

П'ятий крок алгоритму передбачає розв'язання однокритеріальної оптимізаційної задачі з другим КО та модифікованою системою обмежень, що наведено нижче:

$$\max f_2(\bar{x}) = 3x_1 - x_2, \quad (6.19)$$

$$x_1 + 2x_2 + x_3 = 6,$$

$$2x_1 + x_2 + x_4 = 8,$$

$$-x_1 + x_2 + x_5 = 4,$$

$$2x_1 + 3x_2 + x_3 + 2x_4 \geq 14,7 \text{ – додаткове обмеження на основі } f_1(\bar{x}),$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0.$$

Слід зауважити, що задача (6.19) аналогічна до (6.15) лишень з відмінністю в додатковому обмеженні, де змінено максимальне значення цільової функції 22 на 14,7.

Результати розв'язання задачі (6.19) наведено нижче:

$$f_3^* = 2,42, f_2^* = 7,26, f_1^* = 14,7, \quad (6.20)$$

$$x_1 = 2,42, x_2 = 0, x_3 = 3,58, x_4 = 3,16, x_5 = 6,42.$$

Надалі, знову необхідно зробити поступку, але вже для другого критерію оптимальності. Для прикладу, виберемо величину поступки для другого критерію оптимальності в 40 %:

$$f_2^{**} = f_2^* - \Delta_2 = f_2^* - 0,4 * f_2^* = 7,26 - 0,4 * 7,26 = 7,26 - 2,9 = 4,36. \quad (6.21)$$

Тоді додаткове обмеження набуде наступного вигляду:

$$3x_1 - x_2 \geq 4,36 - 2\text{-ше додаткове обмеження на основі } f_2(\bar{x}).$$

Наступний крок алгоритму передбачає розв'язання вже такої задачі (з третім КО і двома додатковими обмеженнями):

$$\max f_3(\bar{x}) = x_1 - 4x_2, \quad (6.22)$$

$$x_1 + 2x_2 + x_3 = 6,$$

$$2x_1 + x_2 + x_4 = 8,$$

$$-x_1 + x_2 + x_5 = 4,$$

$$2x_1 + 3x_2 + x_3 + 2x_4 \geq 14,7 - 1\text{-ше додаткове обмеження на основі } f_1(\bar{x}),$$

$$3x_1 - x_2 \geq 4,36 \quad - 2\text{-ге додаткове обмеження на основі } f_2(\bar{x}),$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0.$$

Отримані кінцеві результати розв'язання задачі (6.22) подані нижче :

$$f_1^* = 14,7, f_2^* = 4,36, f_3^* = 7,74, \quad (6.23)$$

$$x_1 = 1,94, x_2 = 1,45, x_3 = 1,16, x_4 = 2,68, x_5 = 4,48.$$

Отже, метод послідовних поступок доцільно застосовувати для розв'язання тих багатокритеріальних задач, в яких усі часткові критерії природнім чином впорядковані за ступенем важливості, причому кожний критерій настільки істотно більш важливіший за наступний, що можна

обмежитися врахуванням тільки попарного зв'язку критеріїв і вибрати допустиме зниження чергового критерію з врахуванням поведінки тільки одного наступного критерію.

Особливо зручним є випадок, коли вже в результаті попереднього аналізу багатокритеріальної задачі виявляється, що можна допустити поступки тільки в межах “інженерної” точності (5-10% від найбільшої величини критерію).

6.4. Методи цільового програмування

Основна ідея методів цієї групи полягає в припущенні, що існує певна точка, де значення усіх критеріїв $f_i(\bar{x})$, $i=1, n$ досягають найоптимальнішого (цільового, найбільшого $f_i^{\max}(\bar{x})$) значення (тобто досягається певна ціль). Відповідно, в самому загальному випадку задача цільового програмування (ЦП) може бути сформульована як мінімум сум відхилень цільових функцій (критеріїв) від цільових значень (оптимальних) з нормованими коефіцієнтами $(\alpha_1, \alpha_2, \dots, \alpha_n)$:

$$\Delta(F(\bar{x}), F^{\max}) = \left[\sum_{i=1}^n \alpha_i |f_i(\bar{x}) - f_i^{\max}|^p \right]^{\frac{1}{p}} \rightarrow \min, \quad (6.24)$$

де $F^{\max} = (f_1^{\max}, f_2^{\max}, \dots, f_n^{\max})$ - вектор цільових значень КО; $\Delta(F(\bar{x}), F^{\max})$ - відхилення між $f(\bar{x})$ та $f^{\max}$, $1 \leq p < \infty$.

Досить часто в літературі точку $F^{\max}$ називають ідеальною чи утопічною.

Наведемо приклад розв'язання ЗБО з використанням виразу (6.24).

$$f_1 = (x_1 + 2x_2) \exp(-x_2) \rightarrow \max, \quad (6.25)$$

$$f_2 = (3x_1 + 2x_2) \exp(-(3x_1 + x_2)) \rightarrow \max,$$

$$f_3 = x_1 + x_2,$$

$$2x_1 + x_2 \leq 2,$$

$$x_2 - x_1 \leq 3,$$

$$x_1, x_2 \geq 0.$$

Прийmemo, що $\alpha_1 = 0,5$, $\alpha_2 = 0,3$ та $\alpha_3 = 0,2$.

Визначити

$$F(\bar{x}) = \{f_1, f_2, f_3\} \rightarrow \max.$$

В процесі розв'язання задачі отримали наступні значення ЦФ

$$f_1^{\max} = 1,0748,$$

$$f_2^{\max} = 0,7357,$$

$$f_3^{\max} = 2.$$

При визначенні $\Delta(F(\bar{x}), F^{\max})$ отримали наступне значення:

$$\Delta(F(\bar{x}), F^{\max}) = 0,32534.$$

Значення компонент вектора, відповідно:

$f_1^{\max}$	$f_1(\bar{x})$	$f_2^{\max}$	$f_2(\bar{x})$	$f_3^{\max}$	$f_3(\bar{x})$
1,0748	0,7815	0,7358	0,3609	2	1,16784

6.4.1. Метод умовного центра мас

Метод умовного центра мас знайшов досить широке розповсюдження в процесі автоматизованого проектування. На основі цього методу розв'язані задачі забезпечення ефективності лісогосподарських машин для рубки, догляду, лісомеліоративних агрегатів, гідроманіпуляторів трельовочних машин.

Нехай послідовно знайдені значення екстремумів для кожної цільової функції $f_j(\bar{x})$, що відповідає точкам у просторі проектних параметрів з координатами $\{x_1^{i*}, x_2^{i*}, \dots, x_n^{i*}\}$.

Введемо поняття “умовної маси” точки [3]:

$$m_i = \frac{\sum f_i(x_1^{i*}, x_2^{i*}, \dots, x_n^{i*})}{f_i(x_1^{i*}, x_2^{i*}, \dots, x_n^{i*})},$$

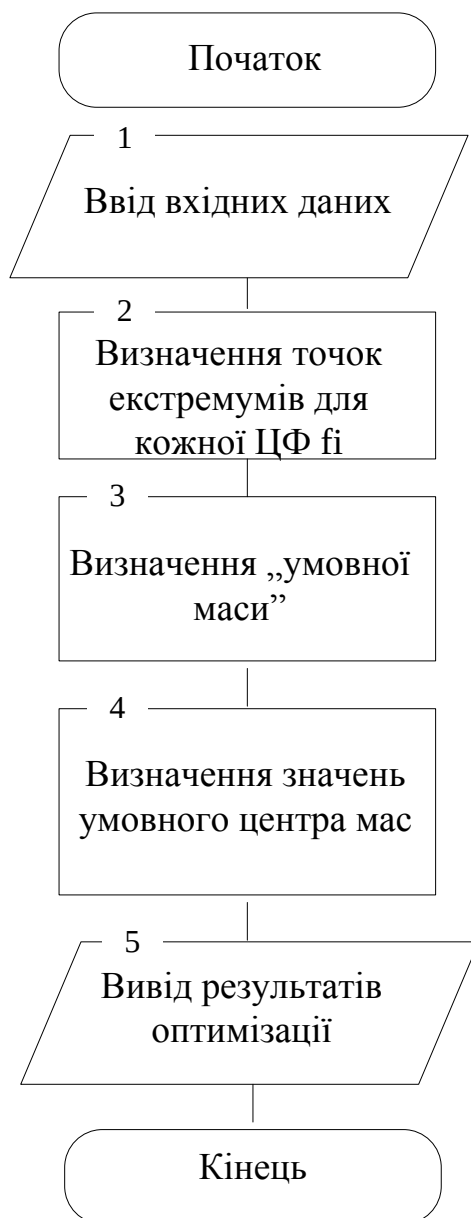


Рис.6.4 Блок-схема алгоритму застосування методу умовного центра мас

де $f_i(x_1^{i*}, x_2^{i*}, \dots, x_n^{i*})$ - значення i -ї цільової функції при сукупності керованих параметрів, що забезпечують екстремальне його значення. Будемо вважати, що компромісному розв'язанню буде задовольняти набір параметрів, які відповідають точці з координатами “умовного центра мас”:

$$x_j^{**} = \frac{\sum m_i x_j^{i*}}{\sum m_i}.$$

Знайдені з використанням цього методу середньозважені значення параметрів x_j^{**} враховують не тільки інтереси всіх показників якості, але і чутливість кожного по відношенню до даного проектного параметра.

Наведемо приклад розв'язання ЗБО з використанням методу умовного центра мас. Отже, в якості прикладу, візьмемо задачу, де є три цільові функції, три обмеження та п'ять змінних і якщо формулюється таким чином, що необхідно знайти такі значення проектних параметрів, щоб забезпечити максимум ЦФ- й.

$$\max f_1(\bar{x}) = 2x_1 + x_2 + 3x_3 + x_4 + 4x_5, \quad (6.26a)$$

$$\max f_2(\bar{x}) = x_1 + 3x_2, \quad (6.26б)$$

$$\max f_3(\bar{x}) = 2x_1 + 2x_2, \quad (6.26в)$$

і виконувалися такі обмеження:

$$2x_1 + x_2 + x_3 = 8, \quad (6.27)$$

$$x_1 + 3x_2 + x_4 = 3,$$

$$x_1 + x_2 + x_5 = 5,$$

$$x_1, x_2, x_3, x_4, x_5 \geq 0.$$

Розв'яжемо послідовно три оптимізаційні задачі з одним критерієм оптимальності. Перша задача включає ЦФ (6.26a) та обмеження (6.27), друга – (6.26б) і (6.27), а третя (6.26в) і (6.27).

В результаті застосування симплекс-методу отримали наступні розв'язання задач багатокритеріальної оптимізації:

для першої:

$$f_1^* = 47, \quad (6.28)$$

$$x_1 = 0, x_2 = 0, x_3 = 8, x_4 = 3, x_5 = 5.$$

для другої:

$$f_2^* = 3, \quad (6.29)$$

$$x_1 = 0, x_2 = 1, x_3 = 7, x_4 = 0, x_5 = 3.$$

для третьої:

$$f_3^* = 6, \quad (6.30)$$

$$x_1 = 3, x_2 = 0, x_3 = 2, x_4 = 0, x_5 = 2.$$

Визначимо параметри “умовного центра мас”, зокрема для нашого випадку:

$$m_i = \frac{\sum f_i(x_1^{i*}, x_2^{i*}, x_3^{i*}, x_4^{i*}, x_5^{i*})}{f_i(x_1^{i*}, x_2^{i*}, x_3^{i*}, x_4^{i*}, x_5^{i*})},$$

$$m_1 = \frac{56}{47} \approx 1.19, m_2 = \frac{56}{3} \approx 18.67, m_3 = \frac{56}{6} \approx 9.33. \quad (6.31)$$

Значення проектних параметрів “умовного центра мас”:

$$x_1^{**} = \frac{0 * 1.19 + 0 * 18.67 + 9.33 * 3}{1.19 + 18.67 + 9.33} = \frac{27.99}{29.19} \approx 0.96, \quad (6.32)$$

$$x_2^{**} = \frac{0 * 1.19 + 1 * 18.67 + 0 * 9.33}{1.19 + 18.67 + 9.33} = \frac{18.67}{29.19} \approx 0.64,$$

$$x_3^{**} = \frac{8 * 1.19 + 7 * 18.67 + 2 * 9.33}{1.19 + 18.67 + 9.33} = \frac{158.87}{29.19} \approx 5.44,$$

$$x_4^{**} = \frac{3 * 1.19 + 0 * 18.67 + 0 * 9.33}{1.19 + 18.67 + 9.33} = \frac{3.57}{29.19} \approx 0.12,$$

$$x_5^{**} = \frac{5 * 1.19 + 4 * 18.67 + 2 * 9.33}{1.19 + 18.67 + 9.33} = \frac{99.29}{29.19} \approx 3.4.$$

Відповідно значення цільових функцій:

$$f_1^* = 2 * 0.96 + 1 * 0.64 + 3 * 5.44 + 1 * 0.12 + 4 * 3.4 = \quad (6.33)$$

$$= 1.92 + 1.28 + 16.32 + 0.12 + 13.6 = 33.24,$$

$$f_2^* = 1 * 0.96 + 3 * 0.64 = 0.96 + 1.92 = 2.88,$$

$$f_3^* = 2 * 0.96 + 2 * 0.64 = 1.92 + 1.28 = 3.2.$$

Метод умовного центра мас широко використовується в процесі розв’язання задач багатокритеріальної оптимізації, в процесі автоматизованого проектування складних об’єктів і систем.

6.4.2. Метод ідеальної точки

Ідея методу “ідеальної точки” ґрунтується на тому, що постулюється існування “ідеальної точки” для розв’язку задачі, у якій досягається екстремум всіх критеріїв (принцип Джофріона). Оскільки ідеальна точка, в абсолютній більшості випадків не знаходиться серед припустимих, виникає проблема знаходження точки, що „найближча” до ідеальної, і належить до множини припустимих рішень. Все було б добре, якщо б існувало єдине об’єктивне поняття “віддалі”, однак це не так – якщо на площині ми можемо з тим чи іншим обґрунтуванням застосовувати Евклідову метрику, то, наприклад, на поверхні кулі найкоротшою віддаллю буде дуга, а не пряма.

Таким чином, для розв’язання задачі багатокритеріальної оптимізації за допомогою методу “ідеальної точки” необхідно насамперед визначити її координати, і надалі визначити метрику, за допомогою якої можна було б виміряти віддаль до оптимальної точки. Для визначення координат “ідеальної точки” розв’язуємо n однокритеріальних задач за кожним з критеріїв оптимізації $\max f_i(a, x) \quad x \in X, \quad i = \overline{1, n}$. Сукупність оптимальних значень критеріїв кожної з однокритеріальних задач $f_i = \max f_i(a, x), \quad x \in X$ і визначаємо координати ідеальної точки $f = (f_1, \dots, f_n)$ в просторі критеріїв. Якщо “ідеальна точка” належить до множини припустимих (що зустрічається вкрай рідко), то розв’язок отриманий.

В іншому випадку визначаємо “віддаль” до ідеальної точки, вводячи метрику, і розв’язуємо однокритеріальну задачу знаходження точки з числа припустимих, яка найменш віддалена від ідеальної. Таким чином задача матиме вигляд $f(x) = \rho(f(a, x) - f^*) \Rightarrow \min, \quad x \in X$. Якщо обрана Евклідова метрика,

то критерій буде мати вигляд $f(x) = \sqrt{\sum_{i=1}^n (f_i(x) - f_i^*)^2} \Rightarrow \min$.

Застосуємо метод ідеальної точки до розв'язання ЗБО (6.22), (6.23).

Результати розв'язання однокритеріальних задач наведено на рис.6.5 та на рис.6.7.

В завершення можна додати ти, що метод ідеальної точки теж широко використовується в процесі розв'язання ЗБО.

Початок роботи програми

$f_1=47$	$f_2=3$	$f_3=6$
$x_1=0$	$x_1=0$	$x_1=3$
$x_2=0$	$x_2=1$	$x_2=0$
$x_3=8$	$x_3=7$	$x_3=2$
$x_4=3$	$x_4=0$	$x_4=0$
$x_5=5$	$x_5=4$	$x_5=2$

Рис.6.5. Результати розв'язання однокритеріальних задач

Розрахунок проведено

$x_1=1$
 $x_2=0,3333333333333333$
 $x_3=5,666666666666667$
 $x_4=1$
 $x_5=3,666666666666667$
 $f_1=35$
 $f_2=2$
 $f_3=2,666666666666667$

Рис. 6.7. Результати розв'язання ЗБО з використанням методу ідеальної точки.

6.5. Контрольні запитання до розділу

1. Яка ідея методу головної компоненти?
2. Яка ідея лексикографічного методу?
3. Яка ідея методу поступок?
4. Яка ідея методу ідеальної точки?

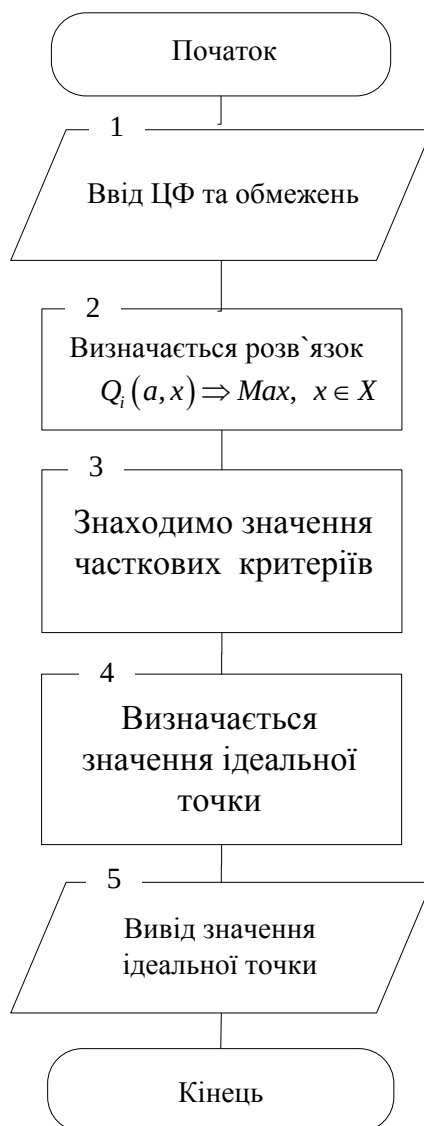


Рис. 6.6. Алгоритм розв'язку задачі багатокритеріальної оптимізації за допомогою методу ідеальної точки

5. Яка ідея методу умовного центра мас?
6. Які переваги та недоліки МГК?
7. Які переваги та недоліки лексикографічного методу?
8. Які переваги та недоліки методу поступок?
9. Які переваги та недоліки методу ідеальної точки?
10. Які переваги та недоліки методу умовного центра мас?
11. Які основні кроки включає алгоритм застосування МГК при розв'язанні ЗБО?

12. Які основні кроки включає алгоритм застосування лексикографічного методу при розв'язанні ЗБО?
13. Які основні кроки включає алгоритм застосування методу поступок при розв'язанні ЗБО?
14. Які основні кроки включає алгоритм застосування методу поступок при розв'язанні ЗБО?
15. Які основні кроки включає алгоритм застосування методу умовного центра мас при розв'язанні ЗБО?
16. Яка ідея методу комплексного критерію?
17. Які переваги та недоліки методу комплексного критерію?

6.6. Список використаної літератури до п'ятого розділу

- [1] Штойер Р. Многокритериальная оптимизация. Теория, вычисления, и приложения.- М.: Радио и связь, 1992.- 504 с.
- [2] Р. Л. Кини, Х. Райфа. Принятие решений при многих критериях: предпочтения и замещения. - М.: Радио и связь, 1981.
- [3] О. И. Ларичев. Теория и методы принятия решений. - М.: Логос, 2000.
- [4] В. В. Подиновский, В. Д. Ногин. Парето-оптимальные решения многокритериальных задач. - М.: Наука, 1982.
- [5] Соболев И.М. Выбор оптимальных параметров в задачах со многими критериями. М.: Наука, 1981.
- [6] Подиновский В.В., Гаврилов В.М. Оптимизация по последовательно применяемым критериям. М.: Сов. радио, 1975.
- [7] Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач. М.: Наука, 1982.
- [8] Лотов А. В., Бушенков В. А., Каменев Г. К., Черных О. Л.. Компьютер и поиск компромисса. Метод достижимых целей. - М.: Наука, 1997.
- [9] Таха Х. Введение в исследование операций: В 2-х кн. Кн. 1. Пер. с англ. - М.: Мир, 1985. - 479 с.
- [10] Таха Х. Введение в исследование операций: В 2-х кн. Кн. 2. Пер. с англ. - М.: Мир, 1985. - 496 с.
- [11] Теслюк В.М., Андрійчук М.І. Конспект лекцій з курсу «Методи синтезу та оптимізації» для студентів базового напрямку «Комп'ютерні науки», Ч.1. - Львів, 2005 – 64 с.
- [12] Теслюк В.М. Моделі та інформаційні технології синтезу мікроелектромеханічних систем: Монографія. – Львів: Видавництво ПП "Вежа і Ко", 2008 – 192 с.

РОЗДІЛ 7.

МЕТОДИ РОЗВ'ЯЗАННЯ ЗАДАЧ БАГАТОКРИТЕРІАЛЬНОЇ ОПТИМІЗАЦІЇ З ВИКОРИСТАННЯМ УЗАГАЛЬНЕНОГО (ІНТЕГРАЛЬНОГО) КРИТЕРІЮ ОПТИМАЛЬНОСТІ

Суть цього методу полягає в тому, що власні критерії $f_i(\vec{x}), i = \overline{1, n}$ будь-яким чином поєднуються в один інтегральний критерій $F(x) = \Phi(f_1(x), f_2(x), \dots, f_n(x))$, а потім знаходиться максимум чи мінімум цієї цільової функції [1 – 9].

Якщо об'єднання власних критеріїв будується, виходячи з об'єктного взаємозв'язку власних критеріїв і критерію узагальненого, то тоді оптимальне рішення оптимізаційної задачі буде коректне. Але таке об'єднання здійснити вкрай складно чи неможливо, тому, як правило, узагальнений критерій є результатом чисто формального об'єднання власних (часткових) критеріїв оптимальності.

В залежності від того, яким чином власні критерії поєднуються в узагальнений критерій оптимальності розрізняють наступні види узагальнених критеріїв:

1. Адитивний критерій;
2. Мультиплікативний критерій;
3. Максимінний (мінімаксний) критерій.

Маючи декілька рішень, кожне з яких отримано на основі врахування багатьох критеріїв оптимізації, необхідно їх порівняти між собою, або надати перевагу одному з них. Для розв'язання цієї задачі будується функція корисності, яка дає змогу визначити показник ефективності рішення і процес надання переваги зводиться до порівняння чисел-значень.

При цьому особа, що приймає рішення (ОПР), враховує, що один набір значень локальних критеріїв володіє перевагою над іншими, якщо йому відповідає більше значення функції переваги.

7.1. Нормалізація узагальненого критерію

В процесі порівняння значень критеріїв оптимальності їх необхідно звести до однієї розмірності, або до безрозмірного виду.

В якості стандарту вибрано перетворення в шкалу зі значеннями в діапазоні від 0 до 1.

1. Якщо відомо деяке еталонне значення (f^{etalon}), то

$$f_{H,i} = \frac{f_i^0}{f_i^{etalon}}, \quad \text{де } i = 1, 2, \dots, m. \quad (7.1)$$

2. Якщо відомо максимально можливе значення f^{max} , то

$$f_{H,i} = \frac{f_i^0}{f_i^{max}}, \quad \text{де } i = 1, 2, \dots, m. \quad (7.2)$$

3. Якщо відомо діапазон зміни від f^{min} до f^{max} , то

$$f_{H,i} = \frac{f_i^0}{f_i^{max} - f_i^{min}}, \quad \text{де } i = 1, 2, \dots, m, \quad (7.3)$$

$$\text{чи } f_{H,i} = \frac{f_i^0 - f_i^{min}}{f_i^{max} - f_i^{min}} \quad \text{де } i = 1, 2, \dots, m. \quad (7.4)$$

Вирази для зведення безрозмірного виду наведені на (7.1 - 7.4) найчастіше зустрічаються в задачах проектування, а більшість еталонних значень, максимально можливих та діапазон змін беруться з технічного завдання на виріб.

7.2. Адитивний критерій

Цільову функцію будують шляхом додавання нормованих значень власних критеріїв. В загальному випадку узагальнена цільова функція має наступний вид:

$$F(\vec{x}) = \sum_{i=1}^n \alpha_i \frac{f_i(\vec{x})}{f_i^0(\vec{x})} = \sum_{i=1}^n \alpha_i f_{H,i}(\vec{x}) \rightarrow \max(\min), \quad (7.5)$$

де n – кількість об'єднаних часткових критеріїв; C_i – ваговий коефіцієнт i – го часткового критерію; $f_i(\vec{x})$ – числове значення i – го часткового критерію;

$f^{(0)}_i(\bar{x})$ – i – й нормований дільник; $f_{H,i}(\bar{x})$ – нормоване значення i – го часткового критерію.

Власні критерії мають різну фізичну природу і тому різну розмірність. А значить просто підсумовувати їх некоректно. У зв'язку з цим, у попередній формулі числові значення власних критеріїв поділяються на деякі дільники, що нормують і призначається в такий спосіб:

1. Як нормуючі дільники приймаються директивні значення чи параметри критеріїв, які задаються замовником (що можна отримати з технічного завдання). Вважається, що значення проектних параметрів, закладені в технічному завданні, є оптимальними чи найкращими.

2. Як нормуючі дільники приймаються максимальні (мінімальні) значення критеріїв, що досягаються в області допустимих рішень.

Розмірності власних критеріїв оптимальності та відповідних дільників, що нормують, однакові, тому в підсумку узагальнений адитивний критерій виходить безрозмірною величиною. Алгоритм розв'язання задачі багатокритеріальної оптимізації зображено на рис.7.1.

Наведемо приклад, який призначений для визначення оптимального варіанта машини з використанням узагальненого (інтегрального) адитивного критерію. Частковими критеріями, з допомогою яких оцінені варіанти машини, являються її продуктивність та надійність. Два критерії “працюють” на максимум, тобто найкращими варіантами машини являються ті з них, які забезпечують найбільшу її продуктивність та надійність. Початкові дані для розв'язання задачі наведені в таблиці 7.1.

Цільова функція на основі узагальненого адитивного критерію запишеться в такий спосіб:

$$F(x) = \alpha_1 \frac{f_1(\bar{x})}{f_1^{(0)}(\bar{x})} + \alpha_2 \frac{f_2(\bar{x})}{f_2^{(0)}(\bar{x})} \rightarrow \max. \quad (7.6)$$

Як нормуючі дільники в цій задачі приймемо найкращі (максимальні) значення власних критеріїв:

$$f_1^{(0)}(\bar{x}) = 4000 \text{ шт/год}, \quad f_2^{(0)}(\bar{x}) = 1500 \text{ год}.$$

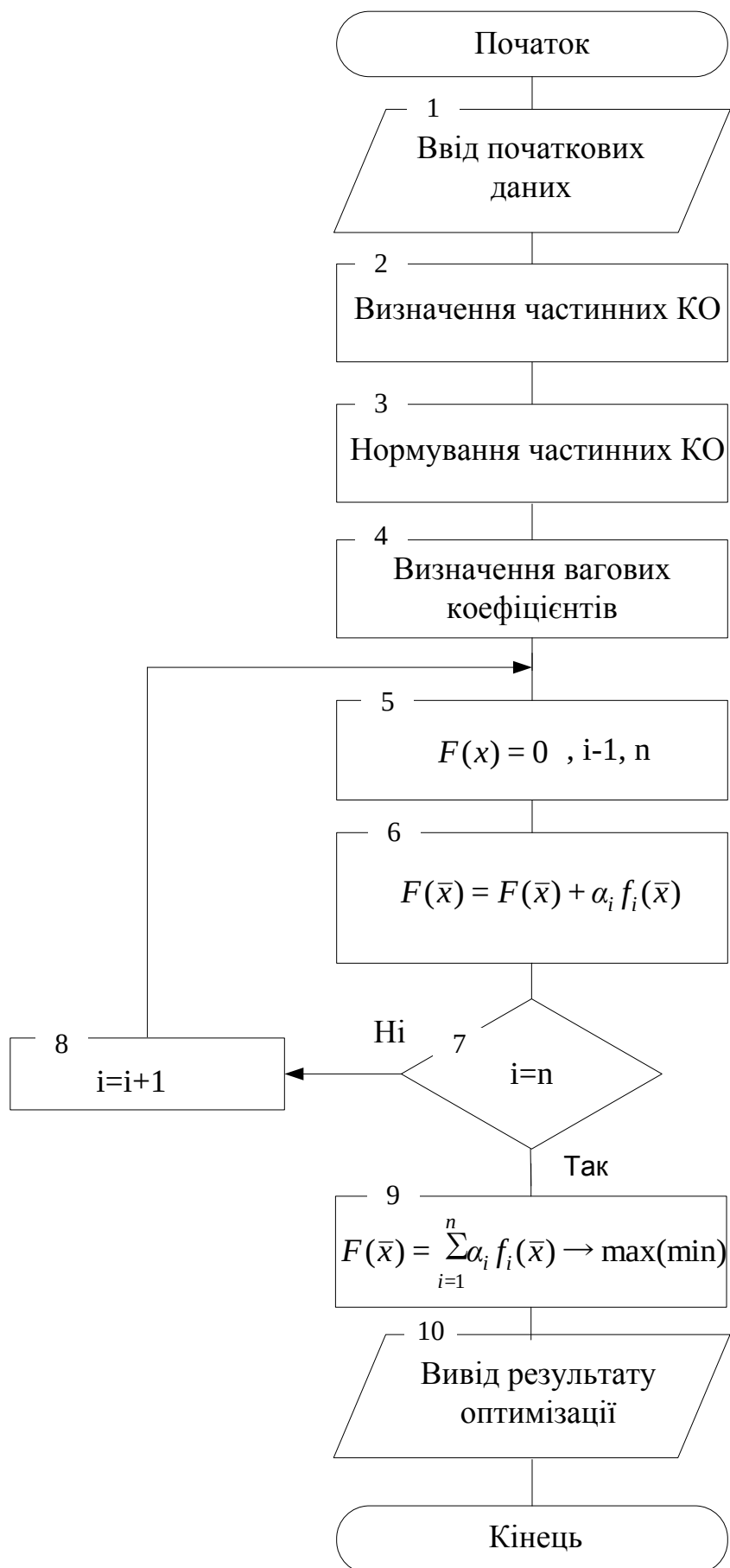


Рис.7.1. Блок-схема алгоритму розв'язання ЗБО з використанням узагальненого адитивного критерію

Початкові дані для визначення оптимального варіанта

Таблиця 7.1.

виконання машини

Критерій МЕМС	Ваговий коефіцієнт	Значення критеріїв для варіантів виконання машини		
		Варіант 1	Варіант 2	Варіант 3
Продуктивність, шт.год	0,6	1000	2000	4000
Надійність, год	0,4	1500	1000	500

Значення узагальненого адитивного критерію розраховуються для кожного варіанта машини:

$$\text{Варіант 1. } F(\bar{x}) = 0,6(1000/4000) + 0,4(1500/1500) = 0,55, \quad (7.7)$$

$$\text{Варіант 2. } F(\bar{x}) = 0,6(2000/4000) + 0,4(1000/1500) = 0,558,$$

$$\text{Варіант 3. } F(\bar{x}) = 0,6(4000/4000) + 0,4(500/1500) = 0,732.$$

Оптимальним є 3-й варіант машини, тому що йому відповідає максимальне значення узагальненого адитивного критерію.

Один з недоліків цього методу полягає в тому, що вагові коефіцієнти призначає, в більшості випадків, сам проектувальник. Різні проектувальники можуть призначати різні вагові коефіцієнти. Розглянемо дещо іншу ситуацію, для прикладу, $\alpha_1 = 0,4$; $\alpha_2 = 0,6$. Визначимо тепер значення адитивних критеріїв для варіантів машини:

$$\text{Варіант 1. } F(\bar{x}) = 0,4 * 0,25 + 0,6 * 1 = 0,7, \quad (7.8)$$

$$\text{Варіант 2. } F(\bar{x}) = 0,4 * 0,5 + 0,6 * 0,67 = 0,602,$$

$$\text{Варіант 3. } F(\bar{x}) = 0,4 * 1 + 0,6 * 0,33 = 0,598.$$

Тобто, при такій зміні значень вагових коефіцієнтів оптимальним уже буде 1 варіант машини.

Перевага даного методу полягає в тому, що, як правило, завжди вдається визначити єдиний оптимальний варіант розв'язання оптимізаційної задачі.

До недоліків цього методу можна віднести таке:

1. Труднощі (суб'єктивізм) у визначенні вагових коефіцієнтів.

2. Адитивний критерій не впливає з об'єктної ролі власних критеріїв і тому виступає як формальний математичний прийом.

3. В адитивному критерії відбувається взаємна компенсація власних критеріїв, тобто зменшення одного з них може бути компенсовано збільшенням іншого критерію.

7.3. Мультиплікативний критерій

Цільова функція з використанням мультиплікативного критерію записується наступним чином:

$$F(\bar{x}) = \prod_{i=1}^n \alpha_i f_i(\bar{x}) \rightarrow \max(\min), \quad (7.9)$$

де $\prod$ – знак добутку; α_i - ваговий коефіцієнт i -го часткового критерію; $f_i(\bar{x})$ - числове значення i -го часткового критерію.

На рис.7.2 зображено блок-схема алгоритму розв'язання ЗБО з використанням узагальненого мультиплікативного критерію.

Переваги мультиплікативного критерію є наступні:

1. Не потрібно нормування власних критеріїв.
2. Практично завжди визначається одне оптимальне рішення.

До недоліків можна віднести наступне:

1. Труднощі (суб'єктивізм) у визначенні вагових коефіцієнтів.
2. Перемножування різних розмірностей.
3. Взаємна компенсація значень власних критеріїв.

Застосовуємо цей метод до розв'язання задачі (Див. табл.3.1)

Цільова функція на основі мультиплікативного критерію запишеться в такий спосіб:

$$F(\bar{x}) = \alpha_1 \frac{f_1(\bar{x})}{f_1^{(0)}(\bar{x})} * \alpha_2 \frac{f_2(\bar{x})}{f_2^{(0)}(\bar{x})} \rightarrow \max. \quad (7.10)$$

Як нормуючі дільники в цій задачі приймемо найкращі (максимальні) значення власних критеріїв:

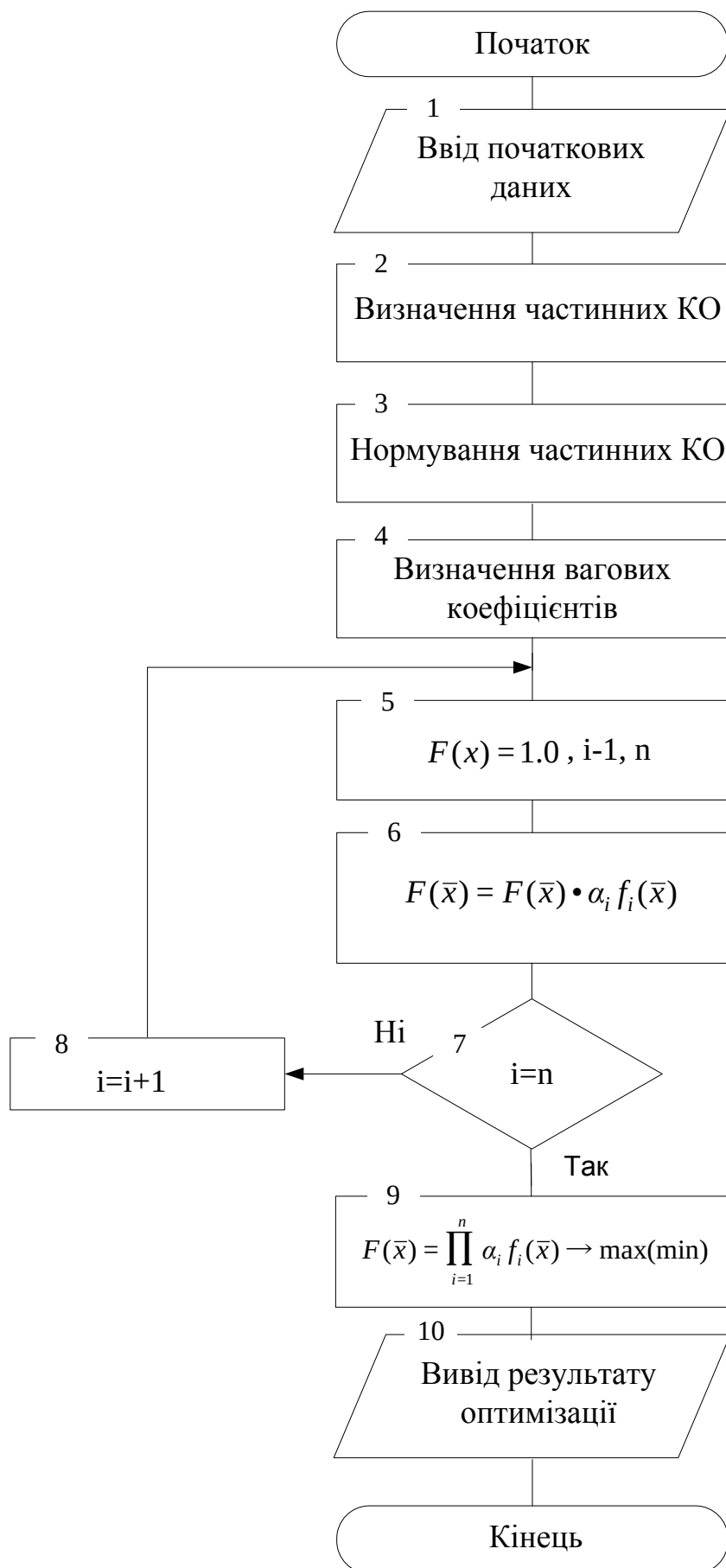


Рис.7.2. Блок-схема алгоритму розв'язання ЗБО з використанням узагальненого мультиплікативного критерію

$$f_1^{(0)}(\bar{x}) = 4000 \text{шт/год}, f_2^{(0)}(\bar{x}) = 1500 \text{год}.$$

Значення узагальненого мультиплікативного критерію розраховуються для кожного варіанта машини і отримаємо:

$$\text{Варіант 1. } F(\bar{x}) = \{0,6(1000/4000)\} * \{0,4(1500/1500)\} = 0,15 * 0,4 \approx 0,06.$$

$$\text{Варіант 2. } F(\bar{x}) = \{0,6(2000/4000)\} * \{0,4(1000/1500)\} = 0,3 * 0,267 \approx 0,08.$$

$$\text{Варіант 3. } F(\bar{x}) = \{0,6(4000/4000)\} * \{0,4(500/1500)\} = 0,6 * 0,13 \approx 0,08.$$

Оптимальними є 2-й варіант і 3-й варіанти машини.

Один з недоліків цього методу полягає також в тому, що вагові коефіцієнти призначає проектувальник. Різні проектувальники можуть призначати різні вагові коефіцієнти. Нехай, для прикладу, $\alpha_1 = 0,4$; $\alpha_2 = 0,6$. Визначимо тепер значення мультиплікативних критеріїв для варіантів машини:

$$\text{Варіант 1. } F(\bar{x}) = (0,4 * 0,25) * (0,6 * 1) \approx 0,06.$$

$$\text{Варіант 2. } F(\bar{x}) = (0,4 * 0,5) * (0,6 * 0,67) \approx 0,08.$$

$$\text{Варіант 3. } F(\bar{x}) = (0,4 * 1) * (0,6 * 0,33) \approx 0,08.$$

Тобто, при такій зміні значень вагових коефіцієнтів не змінилося.

Розглянемо іншу ситуацію, для прикладу $\alpha_1 = 0,1$; $\alpha_2 = 0,9$. Визначимо тепер значення адитивних та мультиплікативних критеріїв для варіантів машини:

$$\text{Варіант 1. } F(\bar{x}) = 0,1 * 0,25 + 0,9 * 1 = 0,925.$$

$$\text{Варіант 2. } F(\bar{x}) = 0,1 * 0,5 + 0,9 * 0,67 = 0,653.$$

$$\text{Варіант 3. } F(\bar{x}) = 0,1 * 1 + 0,9 * 0,33 = 0,397.$$

Тобто, при такій зміні значень вагових коефіцієнтів оптимальним буде 1 варіант машини.

$$\text{Варіант 1. } F(\bar{x}) = (0,1 * 0,25) * (0,9 * 1) \approx 0,0225.$$

$$\text{Варіант 2. } F(\bar{x}) = (0,1 * 0,5) * (0,9 * 0,67) \approx 0,03.$$

$$\text{Варіант 3. } F(\bar{x}) = (0,1 * 1) * (0,9 * 0,33) \approx 0,03.$$

Оптимальними є 2-й варіант і 3-й варіанти машин.

Отже, з даного прикладу можна зробити висновок, що мультиплікативний критерій менш чутливий до неточності визначення вагових коефіцієнтів.

Розглянемо іншу ситуацію, для прикладу $\alpha_1 = 0,0$; $\alpha_2 = 1,0$. Визначимо тепер значення адитивних та мультиплікативних критеріїв для варіантів машини (адитивний):

$$\text{Варіант 1. } F(\bar{x}) = 0,0 * 0,25 + 1,0 * 1 = 1,0.$$

$$\text{Варіант 2. } F(\bar{x}) = 0,0 * 0,5 + 1,0 * 0,67 = 0,67.$$

$$\text{Варіант 3. } F(\bar{x}) = 0,0 * 1 + 1,0 * 0,33 = 0,33.$$

Тобто, при такій зміні значень вагових коефіцієнтів оптимальним буде 1 варіант машини (мультиплікативний).

$$\text{Варіант 1. } F(\bar{x}) = (0,0 * 0,25) * (1,0 * 1) = 0,0.$$

$$\text{Варіант 2. } F(\bar{x}) = (0,0 * 0,5) * (1,0 * 0,67) = 0,0.$$

$$\text{Варіант 3. } F(\bar{x}) = (0,0 * 1) * (1,0 * 0,33) = 0,0.$$

Оптимальні альтернативи відсутні.

Отже, з даного прикладу можна зробити висновок, що якщо в мультиплікативному критерії рівний нулю (або близький до нуля) хоча б один частковий критерій, то результуючий критерій рівний нулю також.

Наведемо інший приклад застосування адитивної та мультиплікативної згорток в процесі розв'язання задачі багатокритеріальної оптимізації.

Задача векторної оптимізації (знайти максимум)

$$f_1(x_1, x_2) = 2x_1 + 5x_2 + 4x_3,$$

$$f_2(x_1, x_2) = 3x_1 + 4x_2 + 4x_3,$$

$$x_1 + x_2 + x_3 \leq 45,$$

$$x_1 + 2x_2 + x_3 \leq 40,$$

$$2x_1 + 3x_3 \leq 75,$$

$$2x_1 + 4x_3 \leq 90,$$

$$3x_1 + 4x_2 \leq 50.$$

$$3x_1 + 2x_2 \leq 25$$

$$x_1, x_2 \geq 0.$$

Відомо, що $\alpha_1 = 0,2$, а $\alpha_2 = 0,8$.

(Розв'язання: при $\alpha_1 = 0,2$, а $\alpha_2 = 0,8$, $F_{\max}^{(1)} = 126,75$.

$\alpha_1 = 0,7$, а $\alpha_2 = 0,3$, $F_{\max}^{(1)} = 131,125$)

Використаємо адитивну згортку

$$\max F(x_1, x_2) = \alpha_1 f_1(x_1, x_2) + \alpha_2 f_2(x_1, x_2),$$

$$x_1 + x_2 + x_3 \leq 45,$$

$$2x_1 + 3x_3 \leq 75,$$

$$3x_1 + 4x_2 \leq 50$$

$$x_1 + 2x_2 + x_3 \leq 40,$$

$$2x_1 + 4x_3 \leq 90,$$

$$3x_1 + 2x_2 \leq 25,$$

$$x_1, x_2 \geq 0.$$

Можемо використати графічний метод розв'язання задачі лінійного програмування чи симплекс метод.

$$\begin{aligned} \max F(x_1, x_2) &= 0,2(2x_1 + 5x_2 + 4x_3) + 0,8(3x_1 + 4x_2 + 4x_3) = \\ &= 0,4x_1 + x_2 + 0,8x_3 + 2,4x_1 + 3,2x_2 + 3,2x_3 = 2,8x_1 + 4,2x_2 + 4x_3 \end{aligned}$$

$$x_1 + x_2 + x_3 + x_4 = 45,$$

$$2x_1 + 3x_3 + x_5 = 75,$$

$$3x_1 + 4x_2 + x_6 = 50$$

$$x_1 + 2x_2 + x_3 + x_7 = 40,$$

$$2x_1 + 4x_3 + x_8 = 90,$$

$$3x_1 + 2x_2 + x_9 = 25$$

$$\alpha_1 = 0,4, \quad \alpha_2 = 0,6,$$

$$\begin{aligned} \max F(x_1, x_2) &= 0,4(2x_1 + 5x_2 + 4x_3) + 0,6(3x_1 + 4x_2 + 4x_3) = \\ &= 0,8x_1 + 2x_2 + 1,6x_3 + 1,8x_1 + 2,4x_2 + 2,4x_3 = 2,6x_1 + 4,4x_2 + 4x_3, \end{aligned}$$

$$\alpha_1 = 0,6, \quad \alpha_2 = 0,4,$$

$$\max F(x_1, x_2) = 0,6(2x_1 + 5x_2 + 4x_3) + 0,4(3x_1 + 4x_2 + 4x_3) =$$

$$= 1,2x_1 + 3x_2 + 2,4x_3 + 1,2x_1 + 1,6x_2 + 1,6x_3 = 2,4x_1 + 4,6x_2 + 4x_3,$$

$$\alpha_1 = 0,7, \quad \alpha_2 = 0,3,$$

$$\begin{aligned} \max F(x_1, x_2) &= 0,7(2x_1 + 5x_2 + 4x_3) + 0,3(3x_1 + 4x_2 + 4x_3) = \\ &= 1,4x_1 + 3,5x_2 + 2,8x_3 + 0,9x_1 + 1,2x_2 + 1,2x_3 = 2,3x_1 + 4,7x_2 + 4x_3, \end{aligned}$$

Згрупуємо отримані результати оптимізації:

$$\alpha_1 = 0,2, \text{ а } \alpha_2 = 0,8, \quad \text{тоді} \quad F_{\max}^{(1)} = 126,75,$$

$$\alpha_1 = 0,4, \text{ а } \alpha_2 = 0,6, \quad \text{тоді} \quad F_{\max}^{(1)} = 128,5,$$

$$\alpha_1 = 0,6, \text{ а } \alpha_2 = 0,4, \quad \text{тоді} \quad F_{\max}^{(1)} = 130,25,$$

$$\alpha_1 = 0,7, \text{ а } \alpha_2 = 0,3, \quad \text{тоді} \quad F_{\max}^{(1)} = 131,125,$$

$$\alpha_1 = 1,0, \text{ а } \alpha_2 = 0,0, \quad \text{тоді} \quad F_{\max}^{(1)} = 133,75.$$

До недоліків цих методів можна віднести те, що рішення, які оптимізують узагальнену функцію, може виявитися незадовільним по одному чи декількох часткових критеріїв.

Це можна пояснити тим, що при досягненні максимуму узагальненої функції надзвичайно малі значення деяких показників $f_i(\bar{x})$ компенсуються великими значеннями інших.

7.4. Максимінний (мінімаксний) критерій

Максимінний (мінімаксний) критерії працюють за принципом компромісу, який ґрунтується на ідеї рівномірності. Сутність принципу максиміна полягає в наступному. При проектуванні складних технічних систем і наявності великого числа власних критеріїв встановити між ними аналітичний взаємозв'язок дуже складно, а в більшості випадків і неможливо. Тому намагаються знайти такі значення проектних (параметрів) $\bar{x} = \{x_1, x_2, \dots, x_m\}$, при яких нормовані значення всіх власних критеріїв рівні між собою:

$$\alpha_i f_i(X) = K, \quad (7.11)$$

де $f_i(X)$ – нормоване значення i -го часткового критерію; K – константа.

При великій кількості власних критеріїв через складні взаємозв'язки домогтися виконання зазначеного вище співвідношення дуже складно. Тому, на практиці, так варіюють значеннями проектних змінних проектування $x_1, x_2, \dots, x_m$, при яких послідовно "підтягуються" ті нормовані критерії, чисельні значення яких у вихідному рішенні виявилися найменшими. Оскільки ця операція виробляється в області компромісу, підтягування "відстаючого" критерію неминуче призводить до зниження значень частини інших критеріїв. Але при проведенні ряду кроків можна домогтися визначеного ступеня зрівноважування суперечливих власних критеріїв, що і є метою принципу максиміна.

Формально принцип максиміна формулюється в такий спосіб: вибрати такий набір змінних $X^{(0)} \in X$, при якому реалізується максимум з мінімальних нормованих значень власних критеріїв, тобто $F(X^{(0)}) = \max \min f_i(X)$.

Такий принцип вибору $X^{(0)}$ іноді називають гарантований результат. Він запозичений з теорії ігор, де є основним принципом. Блок-схема алгоритму розв'язання ЗБО з використанням максимінного підходу (рис.7.3).

Якщо часткові критерії необхідно мінімізувати, то найбільш відстаючим критерієм є той, який приймає максимальне значення. В цьому випадку застосовують принцип мінімакса: $F(X^{(0)}) = \max \min f_i(X)$.

Наведемо приклад застосування максимінного підходу. Отже в Таблиці 7.2 наведено результати вхідних даних.

Таблиця 7.2

$$K_1^* = 8, K_2^* = 9, K_3^* = 6, K_4^* = 10.$$

	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇	S ₈	S ₉
K ₁	3	6	7	4	8	5	9	3	2
K ₂	3	4	4	9	4	8	7	7	5
K ₃	3	7	9	2	4	7	6	6	9
K ₄	9	5	3	7	9	3	7	5	3

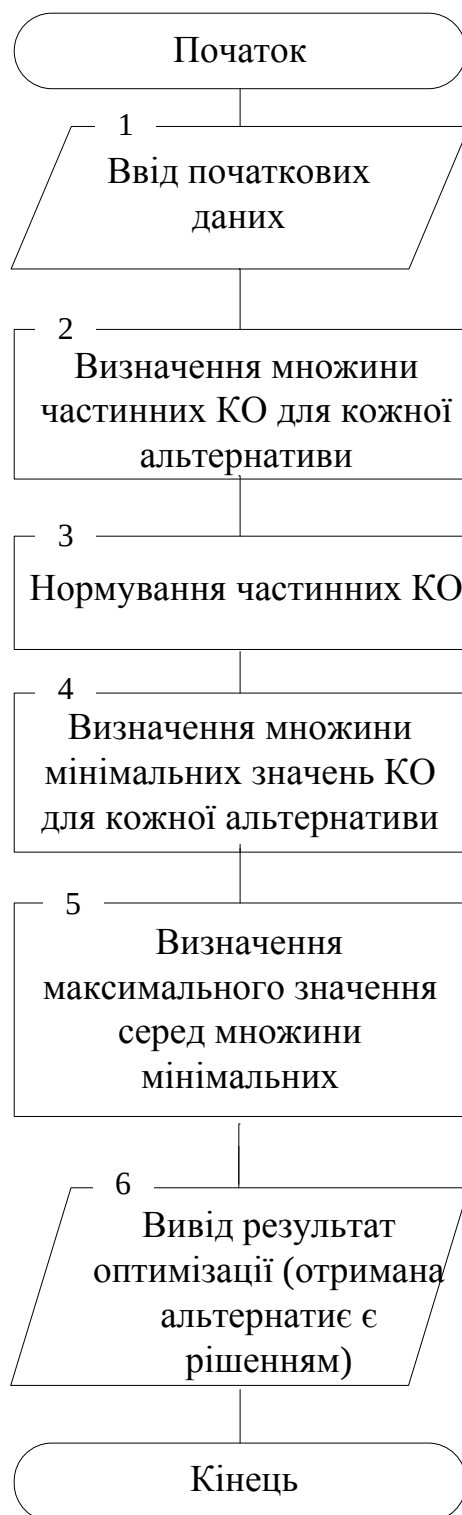


Рис.7.3 Блок-схема алгоритму розв’язання ЗБО з використанням максимінного підходу

В Таблиці 7.2 S_i - i -й варіант рішення задачі багатокритеріальної оптимізації, K_i - i -й критерій оптимізації, K_i^* - номінальне значення i -го КО. В кожній комірці таблиці розміщено значення частинного критерію

оптимальності для i -го варіант рішення частинного критерію. На наступному кроці необхідно звести усі значення критеріїв до безрозмірного виду. Для цього розділимо кожне значення частинного критерію оптимальності на номінальне значення даного КО. Отримані результати зображено в Таблиці 7.3 (від другої до п'ятої стрічки таблиці). Далі для кожної альтернативи вибирається мінімальне значення безрозмірного значення критерію (шоста стрічка Таблиці 7.3). Наступний крок алгоритму передбачає вибір максимального значення з шостої стрічки таблиці. В даному випадку вибраним варіантом є друга альтернатива.

Результати розв'язання максимінної задачі

Таблиця 7.3

	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇	S ₈	S ₉
K₁	3/8	6/8	7/8	4/8	8/8	5/8	9/8	2/8	2/8
K₂	3/9	4/9	4/9	9/9	3/9	8/9	7/9	7/9	5/9
K₃	3/6	7/6	9/6	2/6	4/6	7/6	6/6	6/6	9/6
K₄	9/10	5/10	3/10	7/10	9/10	3/10	4/10	5/10	3/10
Min	3/9	4/9	3/10	2/6	3/9	3/10	4/10	2/8	2/8
	0,33	0,44	0,3	0,33	0,33	0,3	0,4	0,25	0,25
		Max							

Інший приклад, де наведемо особливості застосування мінімаксного підходу. Вхідні дані розміщені в тій же Таблиці 7.2, а результати в Таблиці 7.4.

Отже, перші кроки застосування мінімаксного підходу подібні до застосування максимінного підходу. Тобто, необхідно отримати значення часткових критеріїв оптимізації для кожної альтернативи і звести до безрозмірного виду шляхом ділення на номінальне значення кожного критерію оптимальності (див. від другої до п'ятої стрічки Таблиці 7.4). Далі для кожної альтернативи вибирається максимальне значення безрозмірного значення критерію (шоста стрічка Таблиці 7.4). Наступний крок алгоритму передбачає вибір мінімальне значення з шостої стрічки таблиці. В даному випадку вибраним варіантом є перша альтернатива (див. рис.7.4).

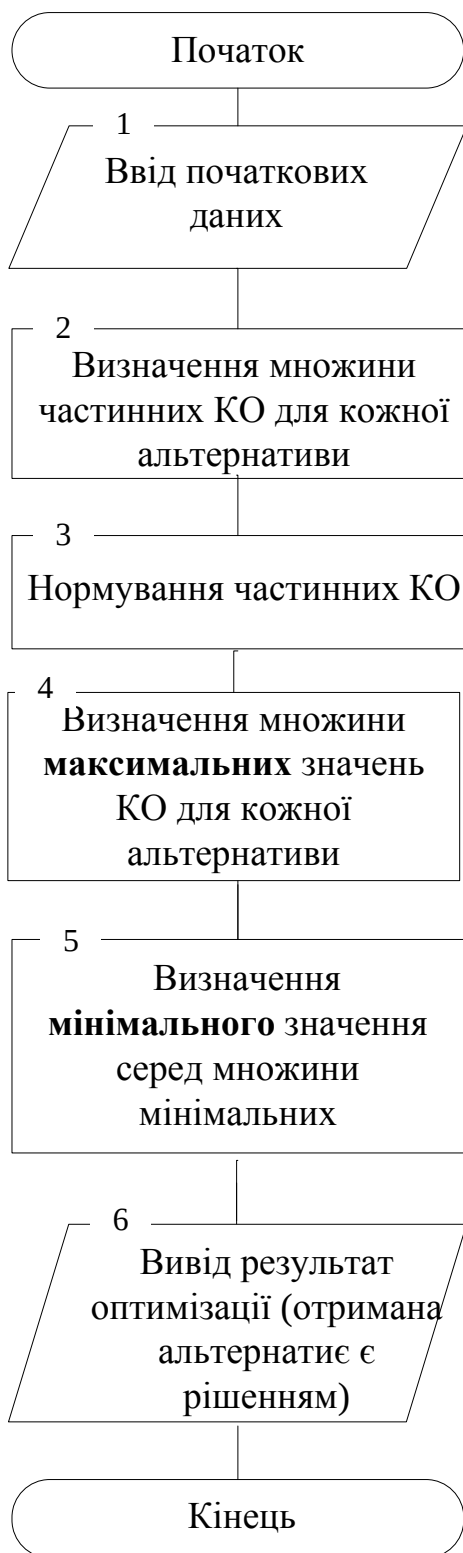


Рис.7.4 Блок-схема алгоритму розв'язання ЗБО з використанням мінімаксного підходу

	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇	S ₈	S ₉
K ₁	3/8	6/8	7/8	4/8	8/8	5/8	9/8	2/8	2/8
K ₂	3/9	4/9	4/9	9/9	3/9	8/9	7/9	7/9	5/9
K ₃	3/6	7/6	9/6	2/6	4/6	7/6	6/6	6/6	9/6
K ₄	9/10	5/10	3/10	7/10	9/10	3/10	4/10	5/10	3/10
Max	9/10	7/6	9/6	9/9	8/8	7/6	9/8	6/6	9/6
	0,9	1,17	1,5	1,0	1,0	1,17	1,125	1,0	1,5
Min									

7.5. Методи визначення значень вагових коефіцієнтів

У випадку рівноцінних критеріїв оптимальності w_i вибираються однаковими [4]

$$w_i = \frac{1}{n}, i = 1, 2, \dots, n, \quad (7.13)$$

де n - кількість критеріїв оптимальності.

Для нерівноцінних критеріїв, тобто критеріїв, для яких особа, що приймає рішення може встановити пріоритет по важливості. В цьому випадку значення вагових коефіцієнтів вибираються відповідно до важливості критерію.

$$w_i > 0, \sum_{i=1}^n w_i = 1, i = 1, 2, \dots, n. \quad (7.14)$$

Найбільш відомими, які широко використовуються проектуванні є метод ранжування та метод приписування балів.

7.5.1. Метод ранжування

Маємо: l - експертів та n - часткових критеріїв $f_i(\bar{x})$, де $i = \overline{1, n}$.

Цифрою 1 – позначаємо найбільш важливий КО, 2 – наступний за важливістю і цифрою n - критерій з найменшим значенням важливості [4].

Потім присвоюємо ранг кожному КО. Ранг 1 має оцінку n і є найбільш важливим з точки зору експерта, а ранг 2 – оцінка $(n-1)$ і так далі до рангу n , якому присвоюємо оцінку 1.

Позначимо ранг i -го критерію (всіх $\in n$) k -им експертом (який $\in l$) $r_i^{(k)}$ вагові коефіцієнти w_i , $i = \overline{1, n}$ визначається з допомогою наступної формули:

$$C_i = \frac{\sum_{k=1}^l r_i^{(k)}}{\sum_{i=1}^n \sum_{k=1}^l r_i^{(k)}}, \quad i = \overline{1, n}. \quad (7.15)$$

Наведемо приклад застосування методу ранжування. Візьмемо п'ять критеріїв $F_i(\bar{x})$ і трьох експертів. Припустимо, що поставлені оцінки експертами наступні:

Таблиця 7.5.

експерт критерій	експерт 1	експерт 2	експерт 3
f_1	1 (5)	3 (3)	2 (4)
f_2	4 (2)	2 (4)	3 (3)
f_3	3 (3)	1 (5)	1 (5)
f_4	5 (1)	4 (2)	4 (2)
f_5	2 (4)	5 (1)	5 (1)

Результати розв'язання задачі:

$$\sum_{i=1}^n \sum_{k=1}^l r_i^{(k)}, \quad i = \overline{1, n} = 3 * 15 = 45,$$

$$i = 1, C_1 = \frac{\sum_{k=1}^l r_1^{(k)}}{45} = \frac{(5+3+4)}{45} \approx 0,27, \quad (7.16)$$

$$i = 2, C_2 = \frac{\sum_{k=1}^l r_2^{(k)}}{45} = \frac{(2+4+3)}{45} \approx 0,2,$$

$$i = 3, C_3 = \frac{\sum_{k=1}^l r_3^{(k)}}{45} = \frac{(3+5+5)}{45} \approx 0,29,$$

$$i = 4, C_4 = \frac{\sum_{k=1}^l r_4^{(k)}}{45} = \frac{(1+2+2)}{45} \approx 0,11,$$

$$i = 5, C_5 = \frac{\sum_{k=1}^l r_5^{(k)}}{45} = \frac{(4+1+1)}{45} \approx 0,13.$$

7.5.2. Метод приписування балів

Використаємо в якості вхідних даних з попереднього прикладу. Визначимо параметри ваги i – го критерію на основі k – го експерта

Таблиця 7.6

експерт \ критерій	експерт 1	експерт 2	експерт 3
f_1	5/15=0,33	3/15=0,20	4/15=0,27
f_2	2/15=0,13	4/15=0,27	3/15=0,20
f_3	3/15=0,20	5/15=0,33	5/15=0,33
f_4	1/15=0,07	2/15=0,13	2/15=0,13
f_5	4/15=0,27	1/15=0,07	1/15=0,07

Експерти оцінюють важливість КО по шкалі від 0 до 10 балів. Причому, можуть бути оцінки з дробовим значеннями та оцінки з однаковими значеннями для різних КО.

Знаючи бали $h_i^{(k)}$ i – го критерію на основі k – го експерта

$$H_i^{(k)} = \frac{h_i^{(k)}}{\sum_{i=0}^n h_i^k}. \quad (7.17)$$

Вага визначення для i – го критерію $F_i(\bar{x})$ на основі оцінок k – го експерта. Потім використовують формулу

$$C_i = \frac{\sum_{k=1}^l H_i^{(k)}}{\sum_{i=1}^n \sum_{k=1}^l H_i^{(k)}}, i = \overline{1, n},$$

$$i=1, C_1 = \frac{\sum_{k=1}^l H_1^{(k)}}{3} = (0,33 + 0,2 + 0,27)/3 \approx 0,27, \quad (7.18)$$

$$i=2, C_2 = \frac{\sum_{k=1}^l H_2^{(k)}}{3} = (0,13 + 0,27 + 0,2)/3 \approx 0,2,$$

$$i=3, C_3 = \frac{\sum_{k=1}^l H_3^{(k)}}{3} = (0,2 + 0,33 + 0,33)/3 \approx 0,29,$$

$$i=4, C_4 = \frac{\sum_{k=1}^l H_4^{(k)}}{3} = (0,07 + 0,13 + 0,13)/3 \approx 0,11,$$

$$i=5, C_5 = \frac{\sum_{k=1}^l H_5^{(k)}}{3} = (0,27 + 0,07 + 0,07)/3 \approx 0,13.$$

Отже, наведемо переваги та недоліки використання методів на основі згорток для розв'язання задач багатокритеріальної оптимізації.

Переваги:

1. Широко використовується в процесі проектування.
2. Дає змогу максимізувати запас по працездатності технічного вибору.
3. Шляхом зміни значень вагових коефіцієнтів є можливість досліджувати область слабо ефективних рішень (оптимальних за Слейтером).

Недоліки:

В процесі розв'язання задачі приходить вирішувати багато однокритеріальних задач, які є складними та нелінійними і інколи зробити це неможливо.

7.6. Контрольні запитання до розділу

1. Що Ви розумієте під узагальненим критерієм?

2. Як будується адитивний узагальнений критерій?
3. Які переваги та недоліки адитивного критерію?
4. Як будується мультиплікативний узагальнений критерій?
5. Які переваги та недоліки мультиплікативного узагальненого критерію?
6. Який алгоритм розв'язання ЗБО з використанням адитивної згортки?
7. Який алгоритм розв'язання ЗБО з використанням мультиплікативної згортки?
8. З якою метою виконують нормування часткових КО?
9. Який алгоритм застосування максимінного(мінімаксного) узагальненого критерію?
10. Які переваги та недоліки максимінного(мінімаксного) узагальненого критерію?
11. Які методи визначення вагових коефіцієнтів Ви знаєте?
12. Наведіть приклади зведення критеріїв оптимізації до безрозмірного виду.
13. Який алгоритм розв'язання ЗБО з використанням максимінної згортки?
14. Який алгоритм розв'язання ЗБО з використанням мінімаксної згортки?

7.7. Список використаної літератури до п'ятого розділу

- [1] Лотов А. В., Бушенков В. А., Каменев Г. К., Черных О. Л. Компьютер и поиск компромисса. Метод достижимых целей. - М.: Наука, 1997.
- [2] Таха Х. Введение в исследование операций: В 2-х кн. Кн. 1. Пер. с англ. - М.: Мир, 1985. - 479 с.
- [3] Таха Х. Введение в исследование операций: В 2-х кн. Кн. 2. Пер. с англ. - М.: Мир, 1985. - 496 с.
- [4] Корячко В. П., Курейчик В.М., Норенков И. П. Теоретические основы САПР : учеб. для вузов. / В. П. Корячко, В. М. Курейчик, И. П. Норенков. – М. : Энергоатомиздат, 1987. – 400 с.
- [5] Грицюк Ю.І., Грицюк М.Ю. Особливості мультиплікативного згортання часткових критеріїв в узагальнений показник // Зб. наук.-техн. пр. : Науковий Вісник НЛТУ України. – 2014. – Вип. 24.11. – С. 341 – 352.
- [6] Кини Р. Л., Райфа Х. Принятие решений при многих критериях: предпочтения и замещения. - М.: Радио и связь, 1981.
- [7] Steuer R.E. Multiple Criteria Optimization: Theory, Computations, and Application. — New York : John Wiley & Sons, Inc , 1986. ISBN 047188846X.
- [8] Sawaragi Y. Theory of Multiobjective Optimization (vol. 176 of Mathematics in Science and Engineering). — Orlando, FL : Academic Press Inc , 1985. ISBN 0126203709.
- [9] Jürgen Branke, Kalyanmoy Deb, Kaisa Miettinen ma Roman Slowinski Multiobjective Optimization: Interactive and Evolutionary Approaches (Lecture Notes in Computer Science). — Springer, 2008. ISBN 3-540-88907-8.

РОЗДІЛ 8.

ЗАСТОСУВАННЯ МОРФОЛОГІЧНОГО МЕТОДУ ДО АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ МЕМС

Процес проектування МЕМС невід’ємно пов’язаний з процесами класифікації об’єктів розроблення, що дає змогу глибше зрозуміти принципи їх роботи і особливості, які дають змогу значно ефективніше організувати процес їх розроблення.

Будь-яка класифікація тісно пов’язана з критерієм класифікації. При класифікації МЕМС їх може бути декілька. Отже, в роботах [1, 2, 3, 4] обґрунтовано та проаналізовано класифікації мікродавачів згідно наступних критеріїв: за призначенням мікродавачів (мікродавач тиску, вологості, сили тощо); за вихідним контрольованим параметром (ємнісний мікродавач, п’єзорезистивний тощо); згідно фізичних ефектів, які використовуються в процесі роботи мікродавачів (мікродавач на основі п’єзоефекту, мікродавач на основі магнітострикційного ефекту тощо); згідно параметрів, для вимірювання яких вони призначені (мікродавачі тиску, інерційні мікродавачі, магнітометри, біодавачі, електрохімічні мікродавачі, молекулярні давачі, газові мікродавачі на основі полімерів, резонансні мікродавачі, хімічні мікродавачі, комплексні мікродавачі, термічні мікродавачі).

Проведена класифікація актюаторів згідно вхідної енергії, а саме: електричної, енергії тиску; термічної енергії; хімічної енергії; акустичної енергії тощо. Наступний критерій відображає, які сили використані при роботі мікроактюатора. З допомогою кожного виду енергії можна згенерувати певні сили. Так, для прикладу, електрична енергія дає змогу згенерувати два типи сил: молекулярні сили та сили електричного і магнітного полів. Відповідні сили дають змогу побудувати різні види актюаторів.

Отримані результати аналізу класифікацій дали змогу визначити базові елементи МЕМС (мікродавачі, мікроактюатори та інші), які характеризують основні параметри цілої групи пристроїв з наступним їх використанням в

процесі розв'язання проектних процедур на системному та інших рівнях багаторівневого автоматизованого проектування MEMC.

8.1. Розроблення моделі для проектування MEMC на системному рівні

Задачу прийняття рішення в процесі проектування MEMC на системному рівні [5, 6] можна сформулювати з допомогою наступного кортежа:

$$Z_{pr} = \langle A, K_p, Model, Main \rangle, \quad (3.1)$$

де $A = \{A_1, A_2, \dots, A_n\}$ - множина альтернатив проектного рішення; $K_p = \{K_1, K_2, \dots, K_m\}$ - множина критеріїв, які визначають можливі зміни вихідних параметрів; $Model = \{Model_1, Model_2, \dots, Model_i\}$ - множина моделей, які дозволяють визначити на основі альтернативи, множину критеріїв; $Main$ - основний критерій, який дає змогу, в ситуації багатокритеріальної задачі, визначити найкращий результат структури системи згідно сформульованого критерію оптимальності.

Слід зазначити, що кожний i -й елемент множини критеріїв є підмножиною і включає мінімальне та максимальне значення критерію, тобто

$$K_i = \{k_{\min}^i, k_{\max}^i\}.$$

Алгоритм розв'язання задачі на системному рівні проектування передбачає послідовне розв'язання наступної сукупності задач: постановка задачі, формування множини критеріїв, генерація множини альтернативних рішень (АР), побудова множини математичних моделей, побудова основного критерію оптимальності, вибір оптимального рішення, аналіз отриманих результатів.

На етапі системного проектування технічне завдання та множина критеріїв є вже сформовані. Процес генерації множини альтернатив базується на побудові множини альтернативних рішень з допомогою І-АБО дерева та правил, які враховують специфіку мікроелектромеханічних систем і вимоги ТЗ. Особливості збереження інформації про альтернативне рішення та мову опису АР детально розглянуто в роботі [7].

На системному рівні проектування МЕМС [8, 9], в основному, використовується математичний апарат теорії систем масового обслуговування і мережі Петрі. Приклад застосування мереж Петрі для побудови математичних моделей, детально описано в роботах [7 – 9].

В якості основного критерію вибору альтернативного рішення використовуємо адитивну його форму, зокрема [10]:

$$Main = \sum_{e=1}^m K_e \omega_e, \quad \omega_e > 0, \quad \sum_{e=1}^m \omega_e = 1,$$

де ω_e - коефіцієнт відносної важливості K_e -го критерію.

Набагато зручніше працювати з нормованим значенням K_e , тому використовуємо наступний вираз при мінімізації цільової функції:

$$Main = \sum_{e=1}^m \left(\frac{k_{\max}^e - k^e}{k_{\max}^e - k_{\min}^e} \right) \omega_e.$$

При необхідності усунути можливість компенсації якості одного часткового критерію іншим, використовують наступний вид критерію оптимальності. У випадку пошуку мінімуму цільової функції

$$Main = \sqrt{\sum_{e=1}^m \omega_e \left(\frac{k^e}{k_{\min}^e} - 1 \right)^2}.$$

Вибір оптимального рішення, в кінцевому випадку, проводиться шляхом розв'язання наступної оптимізаційної задачі (визначити мінімум цільової функції):

$$\min Main_d(K_p) \\ K_p \subset kD,$$

при виконанні системи обмежень $k_{\max}^i \geq k^i \geq k_{\min}^i$, де kD - допустима область можливих рішень.

Необхідно зауважити, що величина кожного критерію визначається на основі математичної моделі: $k^i = Model_i(X)$.

8.2. Побудова І-АБО дерева для інформаційної технології синтезу конструкції МЕМС на системному рівні

Згідно вищепроведеного аналізу наведених класифікацій побудуємо І-АБО дерево для генерації альтернатив структури МЕМС. Отже, на самому верхньому рівні маємо вершину АБО, яка визначає два можливі види МЕМС, а саме: вбудована МЕМС чи автономна. Кожний вид МЕМС, в свою чергу, включає певні підсистеми (підсистема давачів, підсистема актюаторів, підсистема опрацювання передачі та збереження даних, підсистема живлення). По мірі розвитку даної області множина можливих підсистем буде лише зростати. Даний рівень є вершиною типу І. Для прикладу, згідно своєї специфіки, вбудована МЕМС включає три підсистеми: підсистему давачів, підсистему актюаторів і підсистему опрацювання, збереження та передачі даних.

Наступний рівень включає вершини типу АБО. Згідно рис. 8.1 підсистема актюаторів може включати один, або кілька цих пристроїв в залежності від задач, які вирішує МЕМС. Оскільки актюаторів є надзвичайно багато, то ми наведемо кілька базових пристроїв, що характеризують цілий певний клас згідно класифікації, яка наведена в роботі [3, 11] (електростатичний, електромагнітний, магнітострикційний та інші актюатори). Підсистема давачів може включати, як і у випадку підсистеми актюаторів, один, або декілька інтегральних давачів. Підсистема живлення може включати статичний (типу акумулятор) та динамічні пристрої (сонячні батареї, підсистема живлення (ПЖ) на основі п'єзоефекту та ін.). Підсистема опрацювання даних може включати цифрову або аналогову схеми чи схему на базі штучних нейронних мереж. Кожна з цих схем має свої переваги та недоліки.

Четвертий рівень включає вершини типу І. Зрозуміло, що цифрова схема опрацювання обов'язково має включати такі елементи як АЦП, мультиплексор, МП, ЦАП, оперативний запам'ятовуючий пристрій (ОЗП) чи постійний запам'ятовуючий пристрій (ПЗП) або те і інше тощо. Відповідно аналогова

Відповідно вбудована МЕМС, згідно наведеного вище дерева, включає множини давачів, актюаторів та схем опрацювання даних, а у випадку автономної МЕМС – необхідно додати множину елементів живлення, а саме:

Вбудовані(Підсистеми) = ПідсисДавачів(БазДавачі) AND ПідсисАктюаторів(БазАктюатори) AND ПідсисОпрацюванняДаних(ТипиСхем).

Автономна(Підсистеми) = ПідсисДавачів(БазДавачі) AND ПідсисАктюаторів(БазАктюатори) AND ПідсисОпрацюванняДаних(Типи схем) AND ПідсисЖивлення(ТипиБЖ).

Кожна множина включає один, або більше базових елементів, тоді:

ПідсисДавачів(БазДавачі) = ЕлектростатичнийДавач OR П'єзоелектричнийДавач OR ... OR ЕлектромагнітнийДавач;

ПідсисАктюаторів(БазАктюатори)= ЕлектростатичнийАктюатор OR П'єзоелектричний Актюатор OR ... OR Електромагнітний Актюатор;

ПідсисОпрацюванняДаних(ТипиСхем) = ЦифроваСх(Пристрої) OR АналоговаСх(ПідСх) OR СхемаНаОсновіНейроннихМереж;

ПідсисЖивлення(ТипиБЖ) = Статичний OR СонячніБатареї OR ... OR НаБазіП'єзоефекту.

Кожна з підмножин *ЦифроваСх(Пристрої)* та *АналоговаСх(Пристр)* включає підсхеми та пристрої:

ЦифроваСх(Пристрої)=АЦП AND ЦАП AND ... AND Мультиплексом;

АналоговаСх(ПідСх) = Вхідний каскад AND ДифКаскад AND ... AND Мультиплексор.

При програмній реалізації для збереження інформації про дерево альтернативних рішень використана структура даних на основі списків.

Разом з тим, вищенаведене дерево можна заразити у формі безконтурного І-АБО орграфа, який зображений на рис. 8.2 і який зручно використовувати в процедурі формування множини альтернативних рішень.

8.3. Особливості розроблення продукційних правил

При генерації множини альтернативних рішень структури МЕМС використовують продукційні правила у формі:

Pr1: Якщо $f_1 = A_1$, тоді $Elem_1 = B_1$;

Pr2: Якщо $f_2 = A_2$, тоді $Elem_2 = B_2$;

...

Prm: Якщо $f_n = A_m$, тоді $Elem_{l+r+t} = B_m$,

де A_m - лінгвістичні терми; B_m - коди пристроїв МЕМС.

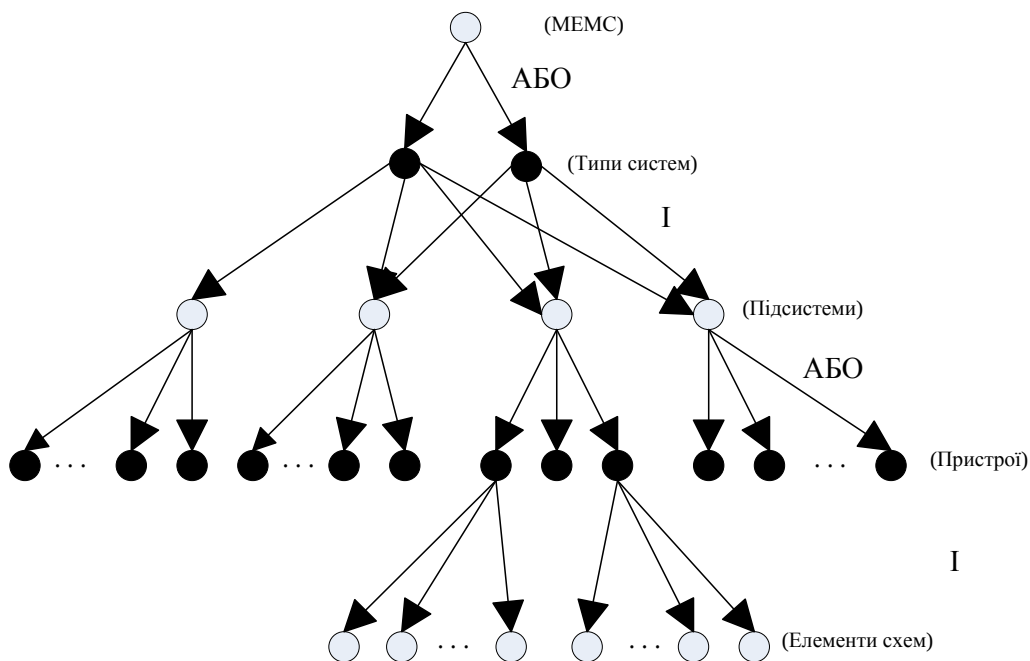


Рис. 8.2. Безконтурний І-АБО оргграф для формування множини альтернативних рішень

Разом з тим, в використовуються складніші конструкції продукційних правил, які потребують для виконання певної функції використання декількох базових пристроїв:

Prz: Якщо $f_z = A_z$, тоді $Elem_z = B_z$, $Elem_{z+1} = B_{z+1}$, ..., $Elem_{z+k} = B_{z+k}$.

8.4. Розроблення інформаційної моделі альтернативного рішення

Для збереження даних про альтернативне рішення розроблена спеціальна інформаційна модель [12, 13]. Позначимо множину можливих рішень синтезу структури МЕМС через M . Кожне альтернативне рішення позначимо E_i , де $i = \overline{1, n}$. Тоді множина можливих рішень буде мати наступний вигляд, а саме: $M(E_1, E_2, \dots, E_n)$, де потужність цієї множини рівна n . Формування кожного можливого рішення проводиться згідно правил зазначених вище та безконтурного І-АБО оргграфа можливих рішень (рис.8.3). Кожне альтернативне рішення виражається числом, де закодовано альтернативу структури МЕМС.

Кожна альтернатива включає таку інформацію: код типу МЕМС, кількість підсистем та їх коди. Потім згідно зазначених кодів підсистем записується кількість пристроїв в кожній підсистемі та їх коди. У випадку наявності в альтернативі й структурі елементів четверного рівня, то після коду пристрою записується кількість елементів, а потім йдуть коди використаних елементів. Тобто маємо структуру даних, яка наведена нижче (рис.8.3). Для опису структури можна використати нормальну форму Бекуса (НФБ).

Кожне альтернативне рішення включає інформацію про:

- тип МЕМС;
- кількість підсистем, які включає МЕМС;
- опис кожної підсистеми.

Запишемо сказане вище, використовуючи НФБ, зокрема:

```
<опис структури МЕМС> ::= { <опис АР> };
<опис АР> ::=
    <тип МЕМС>;
    <кількість підсистем>;
    <опис підсистем МЕМС>.
```

В загальному випадку, розрізняють два види МЕМС:

```
<тип МЕМС> ::= <вбудована> | <автономна>.
```

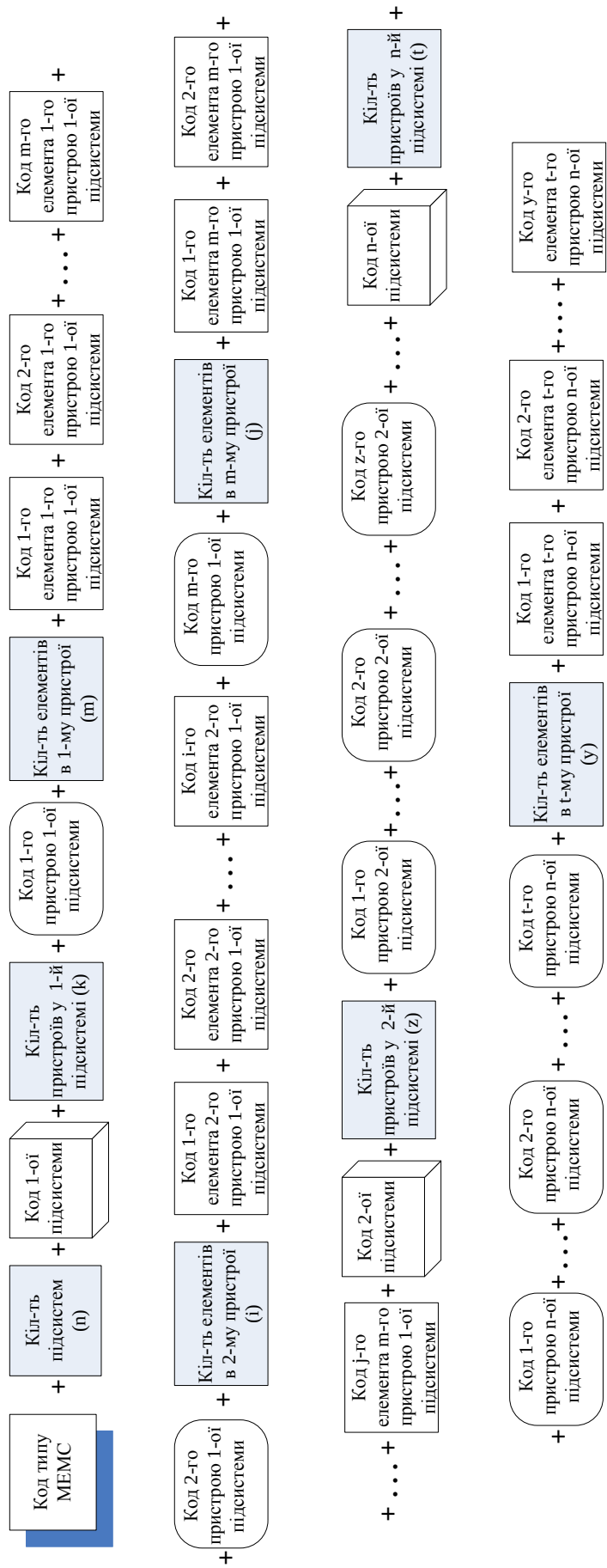


Рис. 8.3. Структура запису альтернативного рішення

Оскільки дана область динамічно розвивається, то закладено в код типу МЕМС можливість розширення, тому код типу МЕМС позначено двоцифровим числом, а саме:

<вбудована>::=00;

<автономна>::=01.

Кількість підсистем МЕМС може приймати наступні значення:

<кількість підсистем>::=<число>;

<цифра>::=<число><цифра>|<число>;

<цифра>::=0|1|2|3|4|5|6|7|8|9.

Для опису кожної підсистеми МЕМС використовують наступні записи:

<опис підсистем МЕМС>::= <код підсистеми>;

<кількість пристроїв у підсистемі>;

<опис пристроїв підсистеми>.

В МЕМС можливі наступні підсистеми:

<код підсистеми>::=<підсистема давачів> | <підсистема актюаторів> |
<підсистема опрацювання, збереження та передачі даних> | <підсистема енергозабезпечення>.

По мірі розвитку даної області кількість підсистем буде зростати. Кожна з підсистем має відповідний код, а саме:

<підсистема давачів>::=S1;

<підсистема актюаторів>::=S2;

<підсистема опрацювання, збереження та передачі даних>::=S3;

<підсистема енергозабезпечення>::=S4.

Кількість пристроїв у підсистемі:

<кількість пристроїв у підсистемі>::=<число>.

Кожний пристрій може включати кілька елементів, а кожний елемент, в свою чергу, визначається кодом:

<опис пристрою підсистеми>::= <код пристрою>; <кількість елементів>;

<коди елементів МЕМС>.

В альтернативному рішенні, в залежності від підсистеми, елементи можуть бути відсутні.

Наведемо коди можливих пристроїв, де відображено їх принцип дії.

<код пристрою>::=<електромеханічний> | <електростатичний> |
 <електромагнітний> | <термічний> | <гідравлічний> | <оптичний> |
 <акустичний> | <п'єзоелектричний>.

Слід зауважити, що дана група може з часом збільшуватися за рахунок інших фізичних принципів роботи пристроїв MEMC.

За визначеним принципом дії може функціонувати різна кількість пристроїв, які один від одного відрізняються за допомогою коду:

<електромеханічний>::=EM1|EM2|EM3|EM4|,...|EMK;
 <електростатичний>::=EC1|EC2|EC3|EC4|,...|ECL;
 <електромагнітний>::=EMag1|EMag2|EMag3|EMag4|,...|EMag;
 <термічний>::=T1|T2|T3|T4|,...|TN; <гідравлічний>::=Г1|Г2|Г3|Г4|,...|ГІ;
 <оптичний>::=O1|O2|O3|O4|,...|OY; <акустичний>::=A1|A2|A3|A4|,...|АП;
 <п'єзоелектричний>::=П1|П2|П3|П4|,...|ПYY;

В разі присутності елементів у пристрої, кількість елементів:

<кількість елементів>::=<число>; <коди елементів>::=E1|E2|E3|E4|,...|EZ.

В даному випадку Z – загальна кількість елементів, яка присутня в БД елементів MEMC.

8.5. Розроблення методу зменшення потужності множини альтернативних рішень

Знаходження оптимального розв'язання на системному рівні проектування надзвичайно складна задача. В ряді робіт запропоновано використати евристичний алгоритм, який ґрунтується на досвіді проектування MEMC та особливостях даної області. Отже, даний евристичний алгоритм зменшення потужності множини альтернативних рішень передбачає виконання наступної послідовності кроків.

1. Визначення та використання базових елементів та структур MEMC;

2. Формування множини альтернативних рішень на основі розробленого І-АБО дерева альтернатив та продукційних правил, які дають змогу відсіяти не перспективні рішення на даному кроці.

3. Відкидання альтернативних рішень, які обумовлені специфікою технології виготовлення.

4. Визначення оптимальної конструкції на основі адитивного критерію оптимальності.

Перший пункт даного алгоритму ґрунтується на тій особливості MEMC, що її структура включає набір складових, призначення яких наперед відоме.

Тобто, для прикладу, стандартна структура, у випадку вбудованої MEMC, включає мікродавач, схему опрацювання, збереження та передачі даних і мікроактюатор. В якості мікродавача та мікроактюатора можуть бути пристрої з різним принципом дії, різноманітних конструкцій, технологій виготовлення тощо.

Як правило, інтегральні пристрої одного принципу дії мають багато спільних або близьких за величиною параметрів. Тому, запропоновано синтезувати структуру MEMC на основі базових пристроїв з можливістю, в кінцевому випадку, певної зміни конструктивних та інших параметрів даного пристрою. Відповідний підхід дає змогу відкинути з розгляду подібні альтернативні рішення, або назвемо їх рішеннями, що впливають одне з іншого.

Другий етап відкидання неперспективних рішень спирається на ряд правил, які дають змогу сформувати множину AP і відкинути ті, які не задовольняють ТЗ на MEMC.

Третій етап зменшення потужності множини AP базується на тій особливості, що існує кілька технологій виготовлення пристроїв MEMC і лише певна група пристроїв може бути виготовлена за однією з технологій. Вартість технологічного обладнання надзвичайно висока, тому виробники та розробники, як правило, притримуються однієї чи двох технологій. Відповідно, запропоновано відкинути ті альтернативні рішення, які використовують технології, що не планується використовувати в процесі виготовлення MEMC.

На четвертому етапі алгоритму формується адитивний критерій оптимальності, який враховує параметри об'єкта проектування і визначається оптимальне рішення або група оптимальних рішень для синтезу структури МЕМС.

8.6. Розроблення методу зменшення потужності множини альтернативних рішень на методу гілок та границь

Для розроблення МЕМС, переважно, використовують багаторівневий підхід [14, 15], який складається кількох рівнів: системного, схемотехнічного [16] і компонентного [17]. Найскладнішими задачами, які потрібно розв'язати на системному рівні є задачі структурного і параметричного синтезу та задачі багатоваріантного аналізу. Задача вибору принципів функціонування мікроелектромеханічної системи і задачі синтезу структур мікросистеми є також типовими задачами структурного синтезу. Спочатку необхідно сформулювати множину альтернативних рішень (МАР) структур МЕМС, які генерують в процесі розв'язання задач структурного синтезу. Наступний крок - необхідно зменшити її потужність застосувавши відповідні адаптивні методи і програмні засоби. Для розв'язку цієї підзадачі, використовують метод гілок і границь (Див.рис.8.4).

Метод гілок і меж (МГГ) - один з комбінаторних методів [18, 19, 20, 21]. Його суть полягає в упорядкованому переборі варіантів і розгляді лише тих з них, які є, за певними ознаками, перспективними, і відкиданні безперспективних варіантів рішень (альтернатив) завдання. Ідея методу полягає в наступному: множина допустимих рішень (множина альтернатив) певним способом розбивається на підмножини, кожна з яких знову розбивається на підмножини. Процес триває доти, поки не буде отримано оптимальне розв'язання початкової задачі у вигляді цілого числа.

Отже, процес автоматизованого проектування структур МЕМС на системному рівні передбачає побудову множини альтернативних рішень [22,

23], що зображено на рис.8.4. MAP генерується з використанням методів і моделей, розроблених в роботах [24].

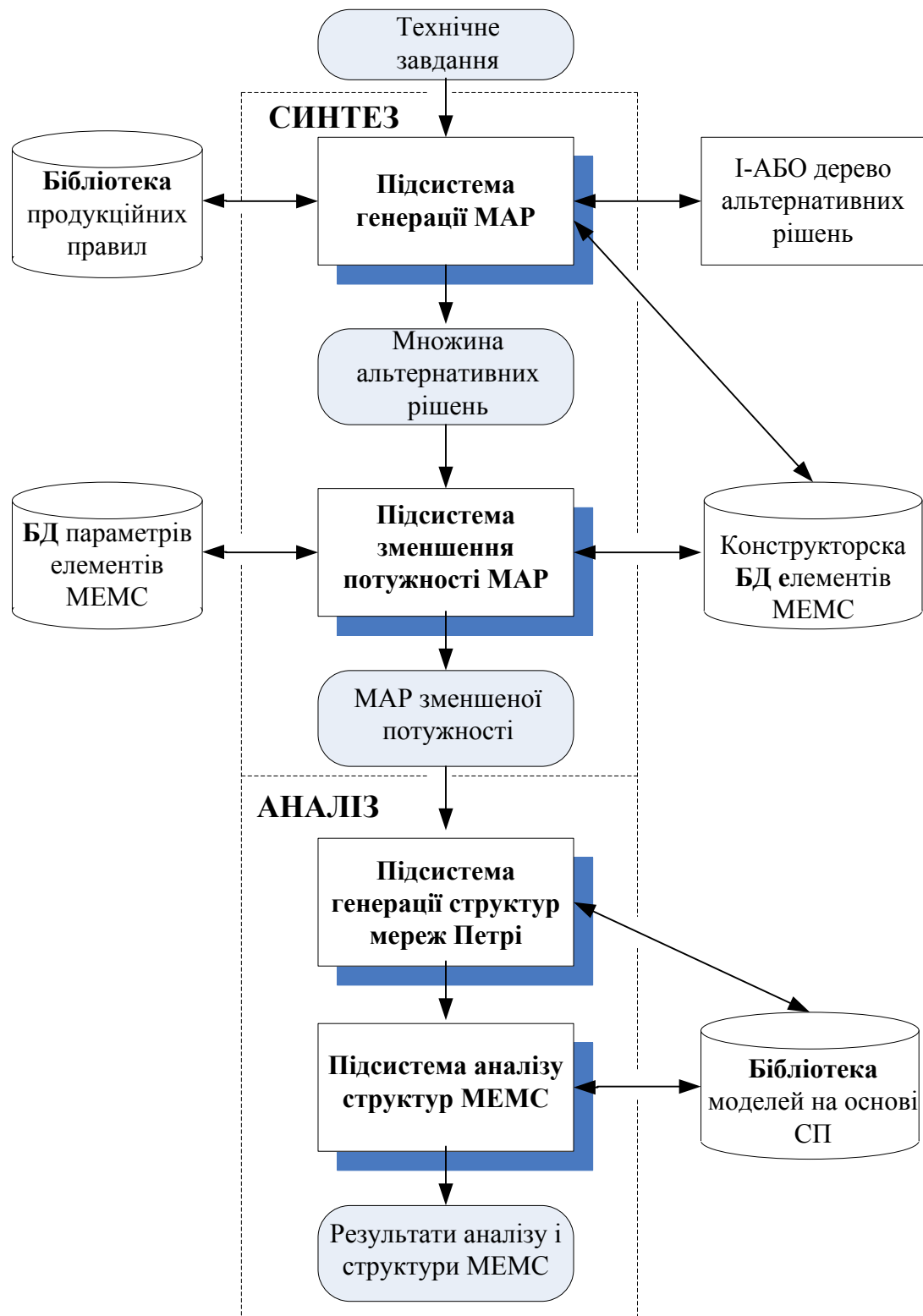


Рис.8.4. Структурна схема розв'язку задач аналізу і синтезу на системному рівні розроблення мікроелектромеханічних систем

Дана операція виконується з використанням програмних засобів в автоматизованому режимі. Генерування множини альтернативних рішень і зміст кожної альтернативи визначається технічним завданням на інтегральний пристрій і розробленими в роботах [24] І-АБО деревом, попередньо побудованої множини продукційних правил на базі знань і досвіду розробників мікросистем.

Результат синтезу множини альтернативних рішень передбачає генерування файлу з набором альтернатив на основі І-АБО дерева альтернативних рішень і попередньо побудованих альтернативних правил. Кількість альтернативних структур мікроелектромеханічних систем може досягати тисяч і десятків тисяч елементів. Інша частина рішення задачі синтезу передбачає зменшення потужності МАР. Для розв'язання цієї підзадачі використовують метод гілок і границь. В результаті застосування МВГ отримують зменшену множину альтернативних рішень, яка може бути порожньою множиною, або включати від одиниць до десятків альтернатив. Кожна з цих альтернатив відповідає вимогам технічного завдання (ТЗ).

Наступне завдання на системному рівні передбачає визначення вихідних параметрів альтернатив. Відповідний аналіз проводиться з використанням розроблених моделей на підставі теорії розширених мереж Петрі, а саме: стохастичних, інгібіторних, часових, мереж з пріоритетами, ієрархічних, функціональних, кольорових і ін.

При цьому, попередньо використовується підсистема, яка на підставі побудованої інформаційної моделі структури альтернативи, в автоматизованому режимі, генерує структуру мережі Петрі.

Однією з основних і перших завдань, які необхідно визначити проектувальнику МЕМС є синтез її структури. Це є типовим завданням синтезу проектних рішень і, разом з тим, найскладнішим для формалізації та реалізації в порівнянні з завданнями на інших рівнях аналізу та синтезу мікросистем.

Зрозуміло, що досягти повної автоматизації при вирішенні цього завдання неможливо, тому ряд проектних операцій виконує проектувальник, а

іншу частину електронно-обчислювальні машини (ЕОМ). Проектні операції, які виконує ЕОМ є добре формалізовані, типові і т.п.

Разом з тим, розв'язання цієї задачі зводиться до вирішення задачі прийняття рішень, яку можна записати у вигляді наступного алгоритму (Рис.8.5) і виразу:

$$Z_{np.} = (Alt, K, MM, O_k),$$

де Alt – множина альтернативних рішень; K – множина критеріїв, за якими проводиться оцінка можливого рішення; MM – модель, яка дає можливість визначити множину критеріїв K для можливого рішення; O_k – основне правило для вибору найважливішого можливого рішення при багатокритеріальних ситуаціях.

На цьому етапі синтезу структури МЕМС згідно із завданням ТЗ може використовуватися явне задання використаних підсистем в множині можливих варіантів Alt . При цьому використовуються наступні правила для зменшення потужності множини Alt під час проектування МЕМС і, які сформульовані в роботах [25]:

- проектувальник може задати параметр заборони використання певних підсистем і протилежна ситуація, коли проектувальник задає пріоритет використання того чи іншого елемента в структурі альтернатив МЕМС (де можна надавати перевагу елементам виготовленими за певної технологією, елементам з відповідним фізичним принципом роботи, швидкодією, енергією живлення і т .п.);
- формування множини можливих підсистем в системі слід починати з вибору актюатора, оскільки цей елемент є визначальним при оцінці ряду параметрів системи;
- кожне можливе проектне рішення формується на основі базових підсистем МЕМС, які враховують специфіку та особливості інтегральних пристроїв цього типу.

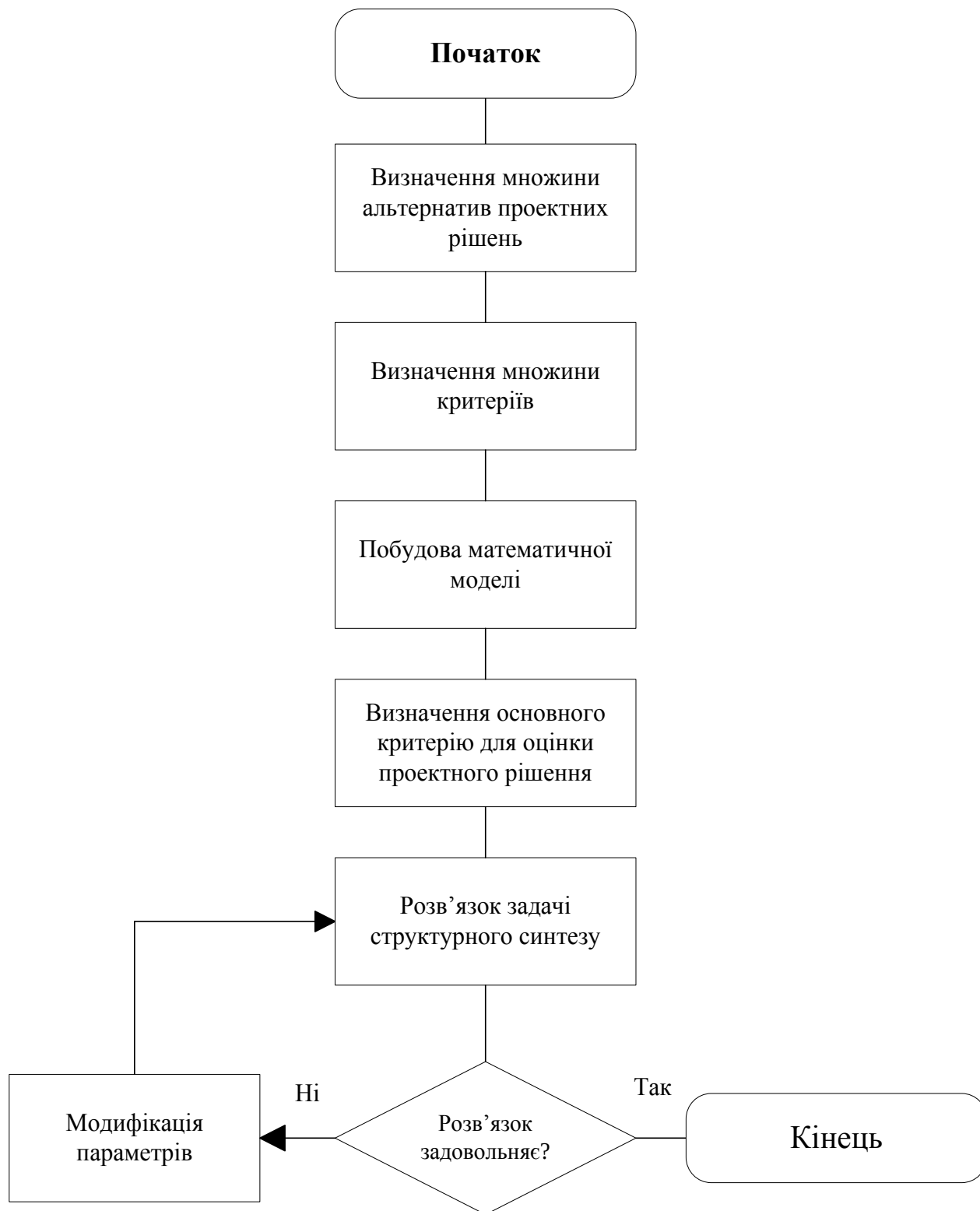


Рис. 8.5. Блок-схема алгоритму розв'язання задачі прийняття рішення на системному рівні аналізу і синтезу МЕМС

У загальному випадку базові структури МЕМС є відомими. При вирішенні завдання на системному рівні необхідно визначити параметри

складових систем МЕМС і вихідних параметрів системи, в цілому.

Більшість моделей базових складових МЕМС розроблені в ряді робіт [26, 27, 28, 29], згруповані в бібліотеки або наведені методи їх побудови. При їх відсутності можна скористатися літературними джерелами, де наведені експериментальні дані, використовуючи які, за допомогою методів апроксимації та планування експерименту, можна побудувати деякі макромоделі.

Разом з тим, для оцінки такого параметра як швидкодія систем або надійність, які надзвичайно актуальні для систем реального часу, можна використовувати теорію розширених мереж Петрі, для задач синтезу структури - теорію систем масового обслуговування та ін.

8.7. Застосування методу гілок і границь для синтезу структурних схем МЕМС

Отже, як було показано вище, метод гілок і границь належить до групи комбінаторних методів дискретного програмування і є одним з найбільш поширених методів цієї групи. Вперше він був запропонований Лендом і Дойгом в 1960 році для вирішення завдань лінійного цілочисельного програмування. Однак ця робота не зробила помітного внеску в розвиток ідей дискретного програмування. Друге народження методу - пов'язане з роботою Літла, Мурті, Суїні і Керель, 1963 р присвяченій задачі комівояжера.

У загальному випадку, застосування методу гілок і меж передбачає розв'язання задачі дискретного програмування в такій формі:

Мінімізувати

$$\min Alt_end = f(x),$$

при виконанні таких умов:

$$x \in A,$$

де A – скінчена множина; $f(x)$ – цільова функція.

На основі методу гілок і меж лежать операції, які дають можливість у певних випадках істотно скоротити процес перебору варіантів розв'язання задачі.

Застосування МВГ передбачає розв'язання двох таких підзадач:

- визначення правила розподілу множини альтернативних рішень на підмножини;
- розробка критерію для визначення оцінки нижньої (верхньої) границі підмножин.

Під час обчислення нижньої межі (оцінки) $\min Alt_end$ нерідко вдається, не розв'язавши задачу, знайти таку нижню границю цільової функції на множині рішень A (чи на її визначеній підмножині) $\xi(A)$, що для всіх $x \in X$:

$$f(x) \geq \xi(A),$$

де $\xi(A)$ – оцінка значень нижньої (верхньої) границі множини можливих рішень Alt .

Особливості розбиття множини альтернативних рішень на підмножини. Реалізація методу гілок і границь пов'язана з постійним розбиттям множини альтернативних рішень на дерево підмножин за наступною схемою.

На першому кроці відома початкова множина $A = A^{(0)}$. Певним способом його розбивають на скінчене число підмножин

$$A_1^{(1)}, A_2^{(1)}, \dots, A_r^{(1)}, \dots, A_n^{(1)},$$

або

$$A^{(0)} = \sum_{i=1}^n A_i^{(1)},$$

де 1 – номер ітерації (кроку), що позначається індексом вверху (в даному випадку номер ітерації одиниця); n – кількість підмножин на нульовому кроці (індекс внизу).

Нехай множини $A_1^{(k)}, A_2^{(k)}, \dots, A_r^{(k)}, \dots, A_n^{(k)}$ ще не підлягали розбиттю.

Згідно правилу, описаному нижче, серед них необхідно вибрати таку множину $A_V^{(k)}$ і розбити її на скінчене число підмножин $A_1^{(k+1)}, A_2^{(k+1)}, \dots, A_r^{(k+1)}, \dots$.

В процесі розв'язку задачі, при розбитті множини на підмножини $A_1, A_2, \dots, A_5 (A_0 = \bigcup_{i=1}^5 A_i)$ завжди враховують, що оцінка для кожного з них є не менше (більше) оцінки для початкової множини A_0 , тобто для всіх підмножин A_i , а саме:

$$\xi(A_i) \geq \xi(A_0).$$

Під час пошуку розв'язків задачі для певних конкретних задач можуть бути вказані різні способи визначення рішень в послідовно розгалужених підмножинах. Будь-який такий спосіб ґрунтується на специфіці та особливостях завдання, які вирішує інженер.

Розглянемо ситуацію, коли отримано оптимальний розв'язок задачі. Для прикладу, нехай попередня множина A розбита за певним правилом на п'ять підмножин, тобто $A = \bigcup_{i=1}^5 A_i$ і отримано розв'язок задачі, такий що $x_0 \in A_V$.

Якщо, при цьому, $f(x_0) = \xi(A_V) \leq \xi(A_i)$ для всіх i , тоді x_0 - оптимальне розв'язання задачі.

Для оцінки точності наближеного розв'язання задачі синтезу, розглянемо таку ситуацію. Нехай, як і в попередньому випадку $A = \bigcup_{i=1}^5 A_i$, і відомо, що

$$\xi = \min_i \xi(A_i). \text{ Якщо } x \text{ - певне розв'язання задачі, тоді } \xi \leq \min_x f(x) \leq f(x).$$

Якщо різниця $\Delta = f(x) - \xi$ не є великою, то x можна прийняти за наближений розв'язок задачі синтезу, а Δ буде оцінкою його точності одержаного розв'язку задачі.

У загальному випадку, алгоритм методу гілок і границь має обов'язково включати такі основні кроки (повний алгоритм включає ряд додаткових кроків):

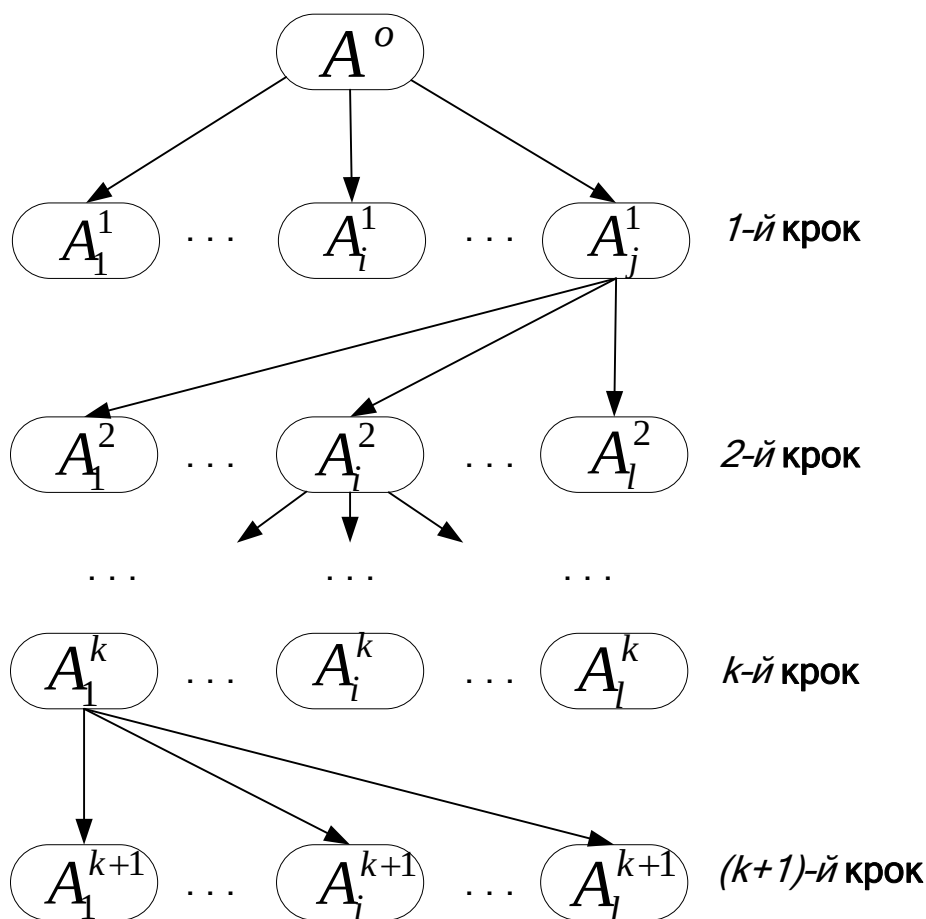


Рис. 8.6. Схема ділення на підмножини

Нульовий крок. Обчислити початкову оцінку множини альтернативних розв'язків задачі синтезу $\xi(A) = \xi(A^{(0)})$. Якщо, при цьому, вдається знайти такий розв'язок задачі x_0 , що $f(x_0) = \xi(A^{(0)})$, то x_0 - оптимальний розв'язок задачі і, в цьому випадку, необхідно завершити обчислення.

Якщо оптимальний розв'язок x_0 не знайдено, тоді слід розбити початкову множину можливих розв'язків $A^{(0)}$ визначеним способом на скінчене число підмножин $A_1^{(1)}, A_2^{(1)}, \dots, A_{\eta}^{(1)}, \dots$, таких, що їх об'єднання має відповідати такій умові $A^{(0)} = A_1^{(1)} \cup A_2^{(1)} \cup \dots \cup A_{\eta}^{(1)}$.

Далі, для продовження роботи, необхідно перейти до першого кроку алгоритму.

Перший крок. Обчислити оцінки верхньої (нижньої) межі для кожної з

підмножин $\xi(A_i^{(1)})$ при $i = 1, 2, \dots, r_1 \dots$. Якщо вдається знайти такий розв'язок проектної задачі синтезу x_0 , що $x_0 \in A_p^{(1)}$ при $(1 \leq p \leq r_1)$ і $f(x_0) = \xi(A_p^{(1)}) \leq \xi(A_i^{(1)})$, при $i = 1, 2, \dots, r_1$, тоді x_0 - оптимальний розв'язок задачі синтезу. В іншому випадку, для подальшого розбиття вибирають найперспективнішу множину $A_v^{(1)}$ за наступним правилом:

$$\xi(A_v^{(1)}) = \min_i \xi(A_i^{(1)}).$$

Далі необхідно розбити множину $A_v^{(1)}$ на кілька підмножин $A_{v,i}^{(1)} = \bigcup_{i=1}^{\eta} A_{v,i}^{(1)}$.

Тим підмножинам, що не підлягають розбиттю, змінюють позначення $A_1^{(2)}, A_2^{(2)}, \dots, A_{r_2}^{(2)}$ і переходять до другого кроку.

k - ий крок. Обчислити оцінки для підмножин k -ої ітерації $\xi(A_i^{(k)})$, при $i = 1, 2, \dots, r_k \dots$. Якщо вдається знайти такий розв'язок задачі x_0 , для якого $f(x_0) = \xi(A_p^{(k)}) \leq \xi(A_i^{(k)})$ для всіх $i = 1, 2, \dots, r_k \dots$, тоді одержаний розв'язок x_0 - оптимальний розв'язок задачі. Якщо ж оптимальний розв'язок задачі не знайдено, тоді знов необхідно вибрати найперспективнішу підмножину $A_v^{(k)}$ таку, яка задовольняє наступній умові $\xi(A_v^{(k)}) = \min_i \xi(A_i^{(k)})$.

Далі розбити множину $A_v^{(k)}$ на кілька підмножин $A_{v,i}^{(k)} = \bigcup_{i=1}^n A_{v,i}^{(k)}$ і перейти

до (k+1) – го кроку.

Під час реалізації вищеописаної загальної схеми методу гілок і границь для окремих задач дискретного програмування, необхідно розробляти конкретні правила розбиття початкової множини на підмножини, способи обчислень оцінок верхньої (нижньої) границь і пошуку рішень завдань, виходячи з специфіки та особливостей конкретних практичних завдань аналізу та синтезу.

Наведемо опис ряду основних кроків алгоритму застосування методу гілок і границь для розв'язку задач цілочисельного програмування.

Нульовий крок.

1. Визначити множину A_0 .
2. Знайти оптимальний розв'язок задачі x_0 .
3. Обчислити оцінку $\xi(A_0) = f(x_0)$.

Якщо розв'язок задачі x_0 задовольняє умову цілочисельності, тоді він є шуканим. В іншому випадку переходимо до першого кроку.

Перший крок.

1. Вибрати певну нецілочислену компоненту $x_i = x_{i_0}, 1 \leq i \leq n_1$.

Множину A_0 розбити на дві підмножини $A_1^{(1)}, A_2^{(1)}$, таким чином, щоб:

$$A_1^{(1)} = \left\{ x : x \in A_0 \text{ та } x_i \leq \lfloor x_{i_0} \rfloor \right\},$$

$$A_2^{(1)} = \left\{ x : x \in A_0 \text{ та } x_i \geq \lfloor x_{i_0} \rfloor + 1 \right\}.$$

2. Розв'язати задачу на множині $A_1^{(1)}$ і знайти оптимальний розв'язок задачі $x_1^{(1)}$. Відповідно розв'язати задачу на множині $A_2^{(1)}$ і знайти розв'язок задачі $x_2^{(1)}$.

3. Обчислити оцінки $\xi(A_1^{(1)}) = f(x_1^{(1)})$ і $\xi(A_2^{(1)}) = f(x_2^{(1)})$.

Перевірити ознаки оптимальності.

Якщо $x_i^{(1)}$ – цілочислене і $\xi(A_1^{(1)}) = \min \{ \xi(A_1^{(1)}), \xi(A_2^{(1)}) \}$, тоді $x_i^{(1)}$ – шуканий розв'язок задачі.

В іншому випадку, перейти до наступної ітерації і виконати процедуру розгалуження.

(k+1)-ий крок.

Нехай виконано k кроків, в результаті яких були побудовані підмножини $A_i^{(k)}, i = \overline{1, r_k}$, визначені оцінки $\xi(A_i^{(k)})$, але ще не знайдено оптимальний

розв'язок задачі. Тоді вибираємо найперспективнішу підмножину $A_v^{(k)}$,

таку, щоб $\xi(A_v^{(k)}) = \min_i \xi(A_i^{(k)})$.

1. Розбиваємо множину $A_v^{(k)}$ на підмножини $A_{v1}^{(k)}$, $A_{v2}^{(k)}$ таким чином, щоб:

$$A_v^{(k)} = A_{v1}^{(k)} \cup A_{v2}^{(k)} \quad \text{і} \quad A_{v1}^{(k)} \cap A_{v2}^{(k)} = \emptyset.$$

З цією метою вибрати визначену нецілочислену компоненту розв'язку задачі $x_v^{(k)}$, наприклад $x_{s,v}^{(k)} = x_{s0}$, $1 \leq s \leq n_1$.

2. Тоді підмножини $A_{v1}^{(k)}$ і $A_{v2}^{(k)}$ визначаємо з таких умов:

$$A_{v1}^{(k)} = \left\{ x : x \in A_v^{(k)} \text{ та } x_s \leq [x_{s0}] \right\}; \quad A_{v2}^{(k)} = \left\{ x : x \in A_v^{(k)} \text{ та } x_s \geq [x_{s0}] + 1 \right\}.$$

3. Розв'язати задачу на підмножинах $A_{v1}^{(k)}$, $A_{v2}^{(k)}$ і знайти оптимальні розв'язки задачі $x_{v1}^{(k)}$, $x_{v2}^{(k)}$ і відповідні їм оцінки:

$$\xi(G_{v1}^{(k)}) = f(x_{v1}^{(k)}); \quad \xi(A_{v2}^{(k)}) = f(x_{v2}^{(k)}).$$

4. Якщо $x_{v1}^{(k)}$ задовольняє умові цілочисленості, тоді вершина $A_{v1}^{(k)}$ є кінцевою і не підлягає подальшому розбиттю. Також, якщо, при цьому,

$$\xi(A_{v1}^{(k)}) = \min_i \xi(A_i^{(k)}),$$

де $\{A_i^{(k)}\}$ – множина всі висячі вершини; $x_{v1}^{(k)}$ – шукане цілочислений розв'язок задачі (це ознака оптимальності).

В іншому випадку, необхідно продовжити процес розгалуження.

8.7.1. Побудова критерію відбору груп альтернативних рішень структур МЕМС

Застосування МВГ на етапі синтезу структур МЕМС, як було зазначено вище, вимагає розв'язання двох підзадач, а саме [30, 31]: визначення алгоритму

або підходу до поділу множини альтернативних рішень на підмножини і визначення критерію згідно з яким необхідно проводити відкидання підмножин.

Для методу поділу на підмножини на системному рівні синтезу MEMC запропоновано використовувати, як один з варіантів, єдину технологію виготовлення мікроелектромеханічних (див. рис.8.7 і рис.8.8), а критерій вибору підмножини - вартість виготовлення (хоча можливі інші критерії або їх комбінації для оцінки нижньої (верхньої) границі при потребі).

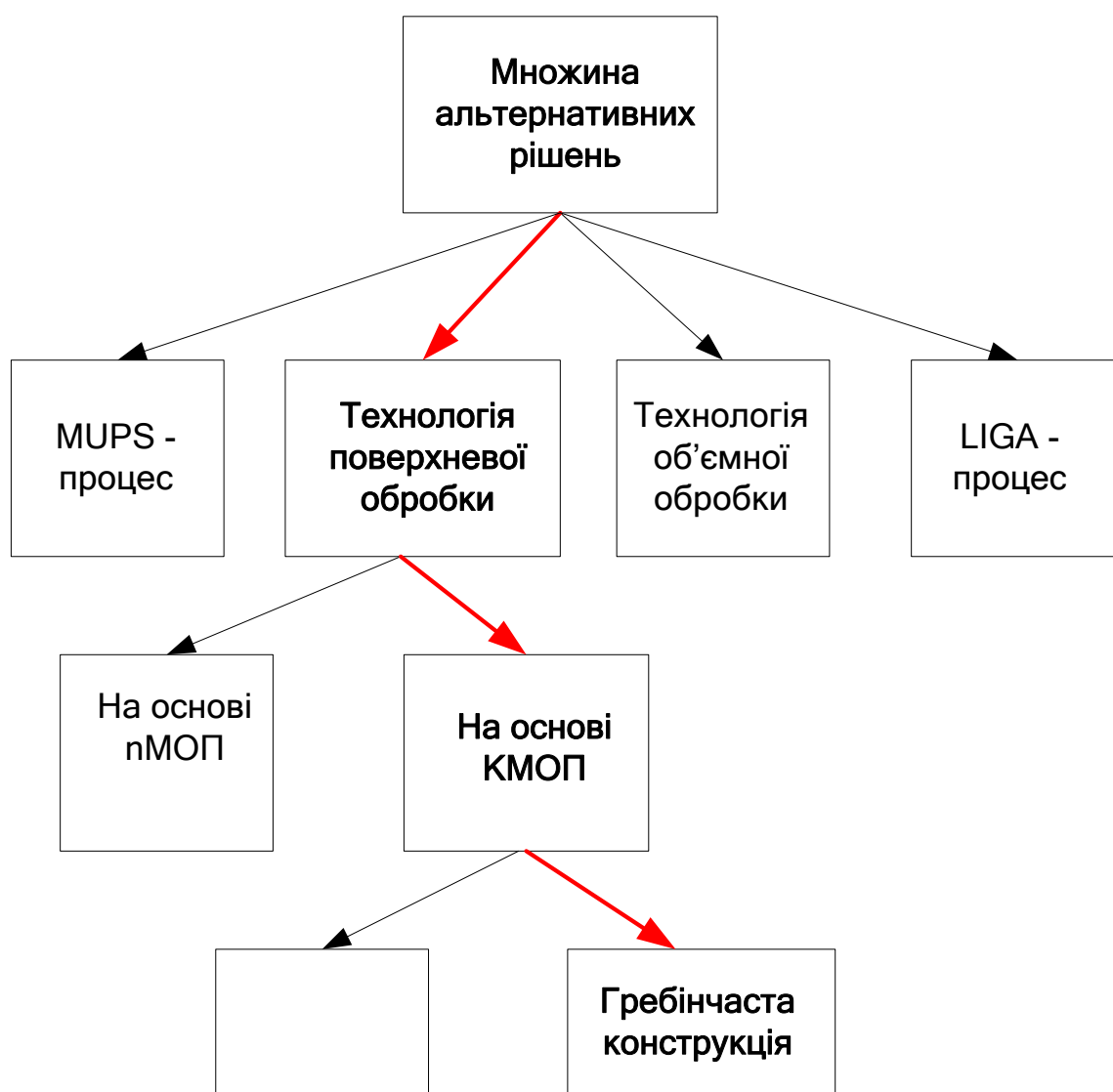


Рис.8.7. Схема застосування методу гілок і границь (критерій єдина технологія виготовлення MEMC)

Отже, більшість MEMC виготовляють з використанням технологій та операцій мікроелектроніки (дифузії, окислення, іонного легування, епітаксії, травлення, фотолітографії та ін.), а також спеціально розроблених мікротехнологій, що використовують операції анізотропного і селективного травлення, метод анодного зварювання або плавлення для з'єднання пластин, а саме: кремнієвого об'ємного мікровиготовлення (ТОМІ), технології поверхневого мікровиготовлення (ТПМІ), процес з використанням локального окислення кремнію (LOCOS процес), технологія виготовлення MEMC з використанням імплантації іонів водню і формування пористого кремнію (HIPS технологія), комерційний варіант тришарової полікремневої ТПМІ (MUMPs технологія), комерційний варіант чотиришаровій полікремневої ТПМІ (SUMMiT), LIGA процесів [[32, 33, 34] та інші.

На сьогоднішній день, активно розвиваються технології і процеси трьохмірного мікровиготовлення [35, 36, 37], які появились недавно і які використовують для виготовлення біомедичних пристроїв і мікроактюаторів з підвищеними вимогами до точності елементів MEMC.

Використання єдиної технології виготовлення мікроелектромеханічних системи дає можливість істотно зменшити вартість інтегрального пристрою і підвищити його надійність і швидкодію.

Більше того, можливі варіанти розбиття на підмножини структур альтернатив MEMC, де використовується об'єднання двох і більше технологій виготовлення. У цьому випадку мається на увазі, наприклад, виготовлення мікросенсора і підсистеми опрацювання даних - за однією технологією, а - актюатора, або групи актюаторів - за іншою.

Приклад використання розбиття на підмножини за критерієм єдиної технології виготовлення зображений на рис. 8.8. У даному випадку було вибрано як базову технологію об'ємного мікровиготовлення.

8.7.2. Побудова критерію оцінки груп альтернативних розв'язків структур MEMC

В якості критерію оцінки нижньої (верхньої) границі підмножин альтернативних рішень можна використовувати різні критерії в процесі синтезу структур MEMC, а саме:

- вартість мікротехнологій виготовлення MEMC;
- єдина мікротехнологія виготовлення MEMC;
- швидкодія MEMC;
- надійність мікроелектромеханічної системи;
- енергетичні витрати;
- комплексний критерій;
- та інші.

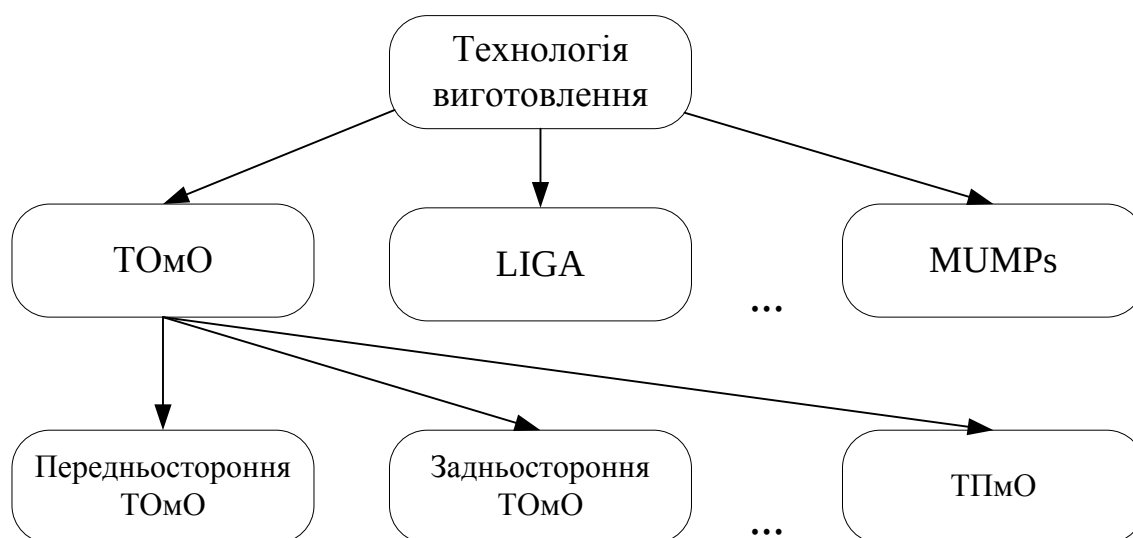


Рис. 8.8. Ієрархія технологій виготовлення MEMC

На рис.8.9 зображено вартість витрат на виготовлення ємнісних елементів мікроелектромеханічних систем (використано різні технології виготовлення) залежно від технологічної точності виготовлення, яка впливає на функціональні параметри розроблюваного інтегрального приладу.

Разом з тим, для забезпечення вимог технічного завдання, необхідно враховувати значення групи вихідних параметрів МЕМС. Розв'язання такої задачі структурного синтезу вимагає побудови множини критеріїв. При синтезі структури МЕМС можуть використовуватися такі критерії як:

- швидкодія системи (K_1);
- енергія живлення (K_2);
- вартість системи (K_3);
- надійність системи (K_4);
- інше.

Отже, в загальному випадку, множина критеріїв, за якими оцінюється проектне рішення, можна записати в наступному вигляді (для випадку n -критеріїв):

$$K = \{K_1, K_2, \dots, K_n\}.$$

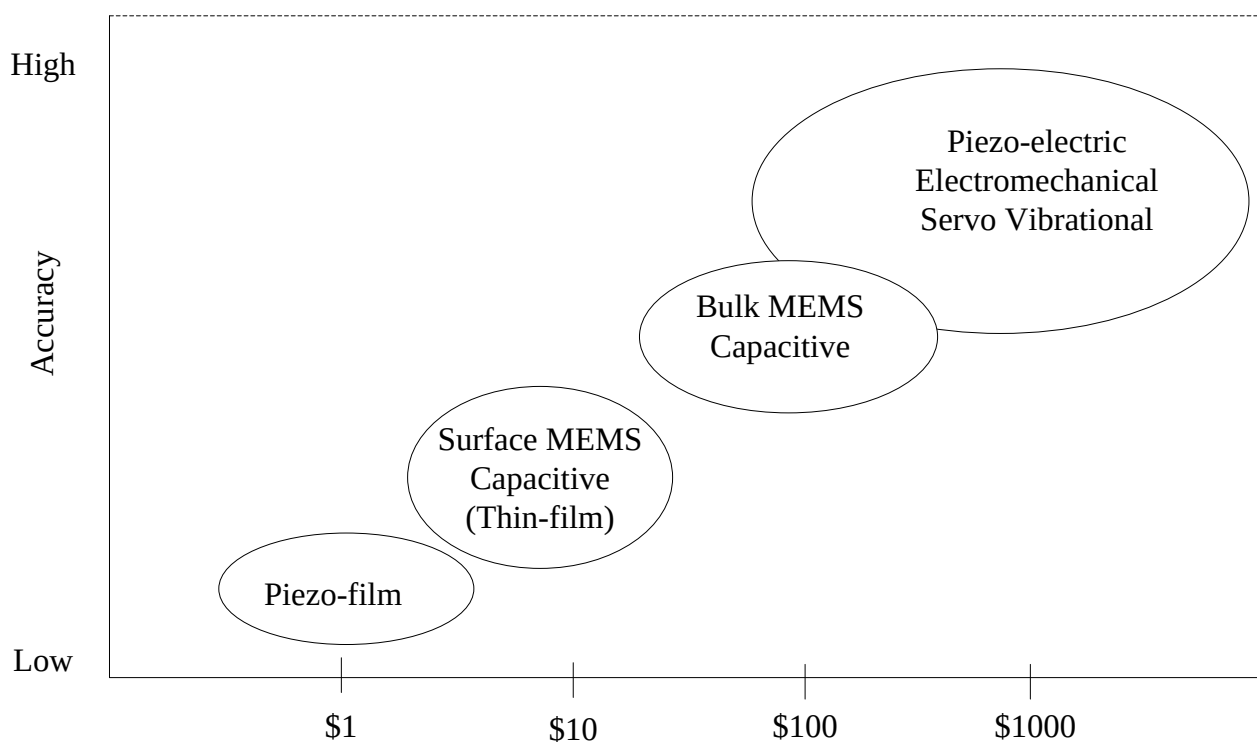


Рис.8.9. Особливості побудови інтегрального критерію

Для прикладу, швидкодія системи у випадку стандартної структури МЕМС залежить від наступних параметрів (у випадку простої структури

MEMC, де присутній один актюатор, канал опрацювання даних і мікросенсор), а саме:

- час, який витрачається на сприйняття сенсором змін оточуючого середовища $t_{\text{сенсора}}$;
- час, який витрачається на опрацювання і видачу керуючого сигналу на актюатор ($t_{\text{сист.обробл.}}$);
- затрати часу на спрацювання актюатора ($t_{\text{актюатора}}$).

Тоді час реакції MEMC буде рівний (швидкодія):

$$t_{\text{системи}} = t_{\text{сенсора}} + t_{\text{сист.обробл.}} + t_{\text{актюатора}}.$$

Для випадку коли, існують кілька сенсорів, актюаторів і складна конструкція підсистеми опрацювання сигналів можна використовувати теорію часових мереж Петрі.

Досить часто ряд критеріїв необхідно порівняти між собою або сформулювати узагальнюючий критерій, для цього можна кожен критерій звести до безрозмірного виду і ввести вагові коефіцієнти. Так, для прикладу, використовуючи адитивний критерій оцінки верхньої (нижньої) границі буде мати таку форму:

$$f(K) = \sum_{i=1}^m w_i \frac{K_i}{K_{\text{ном},i}},$$

де m – кількість критеріїв; w_i – ваговий коефіцієнт; $K_{\text{ном},i}$ – номінальне значення i - го критерію.

Вище наведений критерій можна використовувати в якості основного правила для вибору оптимального розв'язку задачі. Разом з тим, за допомогою вагових коефіцієнтів можна визначити відносну важливість того чи іншого критерію на підставі методу ранжування або інших методів [38].

Більш того, для побудови комплексних критеріїв можна використати мультиплікатний [38]:

$$f(\bar{K}) = \prod_{i=1}^n w_i K_i(\bar{X}),$$

мінімакний [38]:

$$f(\overline{K}) = \max_X \min_i \{K_i(\overline{X})\}, \text{ где } i = \overline{1, n}, X = (x_1, \dots, x_m),$$

чи максимінний критерії [38]:

$$f(\overline{K}) = \min_i \max_X \{K_i(\overline{X})\},$$

де $i = \overline{1, n}$, $X = (x_1, \dots, x_m)$.

8.8. Розроблення основних алгоритмів для синтезу структурних схем МЕМС

Метод гілок і границь є одним з найбільш популярних і ефективних методів безперервної і дискретної глобальної оптимізації. В основу МГГ покладений той факт, що для отримання рішення задачі оптимізації на деякій допустимій множині рішень, можна розбити цю множину на підмножини, знайти розв'язок задачі на кожній з підмножин і як відповідь взяти найкраще з отриманих рішень, відповідно до критерію якості задачі (найбільшим або найменшим значенням цільової функції).

Розвиваючи описану ідею можна прийти до висновку, що замість знаходження точного розв'язку задачі на підмножину (що може бути надзвичайно складно), можна знайти тільки його оцінку, а потім уточнити її шляхом дроблення 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 і границь включає наступні кроки відповідно до роботам [39, 40] (Рис.8.10):

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

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

Крок 2. Перевірка умови зупинки алгоритму шляхом порівняння різниці між оцінками зверху і знизу на черговій поточній підмножині із заданою точністю. Якщо точність досягнута, то перехід до кроку 6.

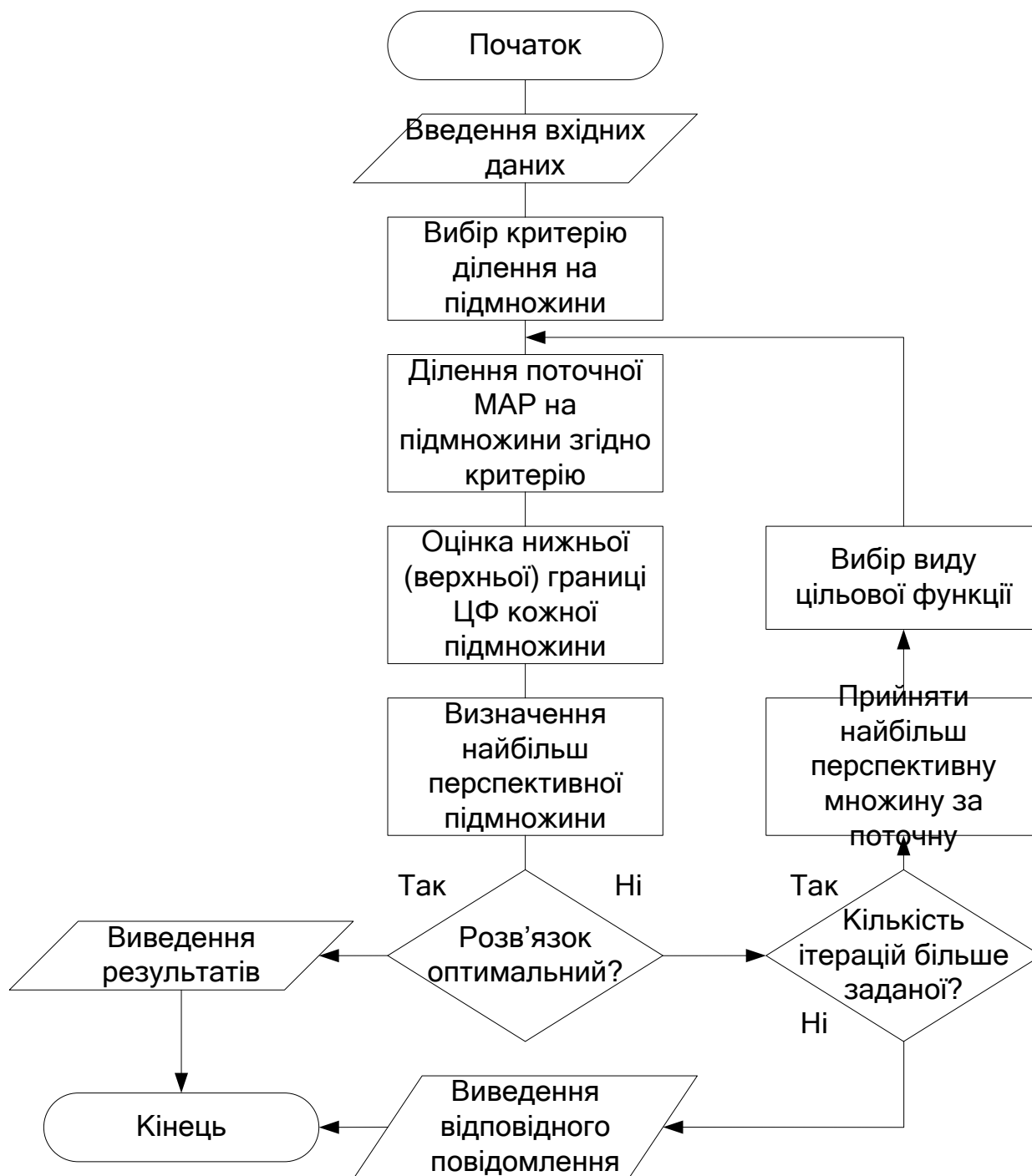


Рис.8.10. Блок-схема алгоритму застосування методу гілок і границь для розв'язку задач синтезу структурних схем MEMC

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, в яких оцінки знизу будуть більше за отриману найменшу оцінку зверху.

Крок 5. Перехід до кроку 1.

Крок 6. Як відповідь, можна взяти центр ваги поточної рекордної підмножини.

На сучасному рівні розвитку обчислювальної техніки, зокрема персональних комп'ютерів, основним шляхом підвищення 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. Однак необхідною умовою для підвищення продуктивності, в цьому випадку, є здатність програмного забезпечення працювати в багатопроцесорному (ядерному) режимі, тобто бути реалізованим за допомогою технологій і засобів паралельного програмування.

Ще однією особливістю методу гілок і границь є його природний паралелізм, який слідує з описаної вище можливості розв'язку незалежних задач оптимізації на кожній з підмножин розбиття. Власне притаманність методу природного паралелізму, тобто можливості розпаралелювання без додаткових обмежень і модифікації і стала причиною обрання його для реалізації різними засобами на створення паралельних програм. Крім того, паралельна реалізація методу досить добре зарекомендувала себе в роботі на кластерних системах.

Алгоритм паралельного варіанта методу гілок і границь має такий вид [39, 40] (Рис.8.11):

Крок 1. Введення даних.

Крок 2. Початкове розбиття допустимої множини на підмножини.

Крок 3. Розподіл одержаних підмножин між процесорами (ядрами).

Крок 4. Обчислення центру ваги, верхньої і нижньої оцінок кожним процесором для своїх підмножин.

Крок 5. Визначення підмножин з мінімальною оцінкою знизу.



Рис.8.11. Блок-схема алгоритму застосування методу гілок і границь для

Крок 6. Якщо не виконується умова виходу (різниця між оцінкою зверху і оцінкою знизу менше заданої точності) або умова синхронізації процесів (виконано певну кількість ітерацій), то відбувається розбиття знайденої підмножини на два нові підмножини і обчислення для них центрів ваги, верхньої та нижньої оцінок, інакше переходимо до пункту 8.

Крок 7. Вилучення безперспективних підмножин і перехід до пункту 5.

Крок 8. Якщо на одному з процесів виконується 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 і перехід до пункту 11.

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

Крок 10. Перерозподіл множини між ядрами і перехід до пункту 5.

Крок 11. Завершення обчислень.

8.9. Побудова інформаційних моделей для реалізації методу гілок і границь

Базуючись на тому, що програмні засоби інформаційної технології синтезу МЕМС орієнтовані на роботу з великою кількістю інформації у поєднанні з високими вимогами до швидкодії програмного продукту, тому доцільно приділити окрему увагу проектуванню методу та інформаційним моделям збереження даних.

Після зчитування вхідного потоку підсистемою, дані необхідно зберегти в структурі з максимально малим часом доступу (читання, запису, переміщення) до них. Проаналізувавши існуючі методи збереження і шаблони організації даних, запропоновано використовувати спискові структури [41]. Тобто, тип даних, що формалізує концепцію впорядкованої множини елементів.

Такий абстрактний тип даних для безтипних, змінних списків можна визначити за допомогою конструктора і чотирьох основних операцій [41, 42]:

- конструктор для створення пустого списку;
- операція визначення пустоти списку;
- операція для додавання елемента в початок списку;
- операція одержання першого елемента списку (чи «голови») списку;
- операція для визначення списку, що складається з усіх елементів списку крім першого (або його «хвоста»).

З такими структурами даних виконуються операції, що задовольняють вимоги до програмних засобів інформаційної технології синтезу структур мікроелектромеханічних систем, а саме: отриманні k -го елемента списку для читання або запису в нього нового значення; додавання нового елемента в будь-яку позицію в списку; видалення елемента списку; об'єднання в одному списку двох або більше лінійних списків; розбиття списку на два або більше фрагментів; створення копії списку; визначення кількості елементів у списку; сортування елементів списку; пошук елемента, що задовольняє певні критерії.

У порівнянні з масивами списки мають ряд переваг. Зокрема, вони досить ефективні щодо операцій додавання або видалення елемента в довільному місці списку, виконуючи їх за постійний час, тоді як масиви для цього вимагають часу $O(n)$, тобто час зростає з ростом кількості елементів масиву.

На підставі списочних структур даних досить зручно будувати складні структури даних, такі як дерева. У даному випадку мається на увазі збереження інформації про І-АБО дереві альтернативних рішень синтезу структур МЕМС.

У списках також не існує проблеми "розширення", яка рано чи пізно постає в масивах фіксованого розміру, коли виникає необхідність включити в нього додаткові елементи. Точно так, фіксований масив, з якого було видалено багато елементів (або вони просто не виконуються) є дуже неефективним з точки зору використання пам'яті. Функціонування списків можливо в ситуації, коли пам'ять комп'ютера фрагментована, тоді як масиви переважно вимагають неперервної області для збереження.

З іншого боку, масиви дають можливість безпосереднього доступу до будь-якого елемента. Односторонньо зв'язані списки, навпаки, вимагають проходження всіх попередніх елементів. Це призводить до складності застосування списків в задачах, де необхідно швидко знаходити елемент за його індексом, наприклад в алгоритмах сортування. Кеширование списків в таких випадках майже не дає ефекту.

Іншим очевидним недоліком списків є необхідність разом з корисною інформацією, додатково зберігати інформацію про покажчики, що відображається на ефективності використання пам'яті цими структурами.

Двостороннє зв'язування вимагає більше пам'яті на елемент і великих обчислювальних витрат на виконання елементарних операцій. Але такими списками легко маніпулювати, тому що вони дозволяють проходження за списком в обох напрямках, що є необхідною умовою синтезу структури МЕМС і функціонування деяких алгоритмів.

Проаналізувавши вищезгадані спискові структури даних, вважаємо доцільним використання двонаправлених списочних структур даних.

8.10. Контрольні запитання до розділу

1. Яка ідея морфологічного методу?
2. Які методи використовують для зменшення потужності множини альтернативних рішень?
3. Сформулюйте задачу прийняття рішень в процесі розв'язання задачі синтезу МЕМС на системному рівні.
4. Що ви розумієте під множиною альтернативних рішень?
5. Що ви розумієте під І-АБО деревом?
6. Який математичний апарат використовують на системному рівні проектування?
7. Яка ідея методу гілок та границь?
8. Які основні кроки передбачає застосування МГГ в процесі синтезу МЕМС?
9. Що таке альтернативна структура МЕМС?

10. Як відбувається генерування альтернатив в процесі автоматизованого проектування МЕМС на системному рівні проектування?
11. Для чого використовують продукційні правила?
12. Які види продукційних правил ви знаєте?

8.11. Список використаної літератури до восьмого розділу

- [1] Теслюк В. М. Моделі та інформаційні технології синтезу мікроелектромеханічних систем / В. М. Теслюк // Монографія. – Львів : Вежа і Ко, 2008. – 192 с.
- [2] Лобур М. В. Методи та моделі для наскрізного проектування вбудованих систем : автореф. дис. на здобуття наук. ступеня докт. техн. наук : 05.13.12 “Системи автоматизації проектувальних робіт” / М. В. Лобур – Київ, НТУУ “КПІ”, 2004. – 32 с.
- [3] Лысенко И. Е. Проектирование сенсорных и актюаторных элементов микросистемной техники / И. Е. Лысенко. – Таганрог : ТРТУ, 2005. – 103 с.
- [4] Колпаков Ф. Ф. Микроэлектромеханические устройства в радиотехнике и системах телекоммуникаций : [учеб. пособие] / Ф. Ф. Колпаков, Н. Г. Борзяк, В. И. Картунов. – Харьков : НАУ ХАИ, 2006. – 82 с.
- [5] Гранат П. П. Алгоритмізація задачі синтезу технологічних маршрутів формування структури кристалів ІС / П. П. Гранат, А. В. Назар, В. М. Теслюк // Вісник Державного університету “Львівська політехніка” : Комп’ютерна інженерія та інформаційні технології. – 2000. – № 413. – С. 51 – 56.
- [6] Назар А. В. Структуризація процесу прийняття рішень в задачах технологічного проектування ІС / А. В. Назар, В. М. Теслюк // Вісник Державного університету “Львівська політехніка” : Комп’ютерна інженерія та інформаційні технології. – 1999. – № 370. – С. 34 – 40.
- [7] Теслюк В.М. Побудова множини можливих рішень з допомогою І - АБО – дерева при вирішенні задач структурного синтезу на системному рівні проектування МЕМС // Науково-технічний журнал “Інформаційні технології і системи”. - Львів, Том 9, № 1, 2006. – С. 101- 108.
- [8] Теслюк В.М. Методологія проектування МЕМС на системному рівні проектування // Збірник наукових праць Української академії друкарства “Комп’ютерні технології друкарства”, № 18. - Львів, 2007. - С. 56 – 63.
- [9] Теслюк В. М. Методологія автоматизованого проектування МЕМС на системному рівні проектування / В. М. Теслюк, М. В. Лобур, П. Ю. Денисюк // Інтелектуальні системи прийняття рішень та прикладні аспекти інформаційних технологій: матер. наук - техн. конф. – 2007. – Херсон: Вишемирський В.С. – Том. 2. – С. 226 – 229.
- [10] Петренко А. И. Основы построения систем автоматизированного проектирования / А. И. Петренко, А. И. Семенов. – К. : Вища школа, 1984. – 296 с.
- [11] Лучинин В. В. Микросистемная техника. Направления и тенденции развития / В. В. Лучинин // Научное приборостроение. – 1999. – Т. 9, № 1. – С. 3 – 18.
- [12] Теслюк В. М. Моделі та алгоритми для автоматизованого проектування МЕМС на функціональному рівні / В. М. Теслюк // Комп’ютерні технології друкарства. Зб. наук. пр. – Львів : УАД, 2006. – Вип. 16. – С. 222 – 230.
- [13] Denysyuk P. XML application for microfluidic devices description / P. Denysyuk, V. Teslyuk, I. Khimich, I. Farmaga // The experience of designing and application of CAD systems in microelectronics (CADSM 2007) : Proc. of the IXth International Conference. – Lviv - Polyana, 2007. – P. 567 – 569.
- [14] Норенков И.П. Основы теории и проектирования САПР. – М.: Высш. Шк., 1990. – 334 с.
- [15] 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.

- [16] Zaharyuk R. A VHDL-AMS Model of a Capacitive Accelerometer Based on the Spring Elasticity Constant / R. Zaharyuk, V. Teslyuk, I. R. Kozak, Hamza Ali Yousef AlShawabkeh // Proc. of the V-th International Conference of Young Scientists (MEMSTECH'2009) – Lviv - Polyana, 2009. – P.49.
- [17] Teslyuk V. Method for automated synthesis of micromodels of acoustic elements of MEMS / V. Teslyuk, P. Denysyuk, Hamza Ali Yousef Alshawabkeh, M. Melnyk // Book of abstracts XVII Polish-Ukrainian Conf. CAD in Machinery Design (CADMD 2009). – 2009, Poland, Krasiczyn. – P. 53 – 54.
- [18] Maluf N. An introduction to Microelectromechanical Systems Engineering / N. Maluf, K. Williams. – Boston, Artech House, 2000. – 283 p.
- [19] Land A.H., and Doig A.G. An automatic method of solving discrete programming problems. *Econometrica*. v28 (1960), pp. 497-520.
- [20] Таха Х. Введение в исследование операций: В 2-х кн. Кн. 1. Пер. с англ. - М.: Мир, 1985. - 479 с.
- [21] Ковалев М. М. Дискретная оптимизация (целочисленное программирование). Изд.2. - Минск, 2003. – 192 с.
- [22] Teslyuk V. Computer-aided design of MEMS at system level / V. Teslyuk, Tarik Al Omari, Hamza Alshavabkeh, P. Denysyuk, M. Melnyk // Journal Machine Dynamics Problems. – Warsaw University of Technology, Poland. – 2007. – Vol. 31, № 3. – P. 92 – 104.
- [23] Teslyuk V. Computer – aided design of MEMS at system level / V. Teslyuk, Tarik Al Omari, Hamza Alshavabkeh, P. Denysyuk, M. Melnyk // CAD in Machinery Design – Implementation and Educational Problems : Proc. of the XV Polish – Ukrainian Conf. – Naleczow, Poland, 2007. – P. 35 – 37.
- [24] Теслюк В.М. Побудова множини можливих рішень з допомогою І - АБО – дерева при вирішенні задач структурного синтезу на системному рівні проектування МЕМС // Інформаційні технології і системи. - Том 9, № 1. - Львів, 2006. - С. 101- 108.
- [25] Теслюк В. М. Інформаційні технології аналізу та синтезу мікроелектромеханічних систем : автореф. дис. на здобуття наук. ступеня докт. техн. наук : 05.13.06 “інформаційні технології” / В. М. Теслюк. – Львів, Видавництво НУ “Львівська політехніка”, 2009. – 36 с.
- [26] Teslyuk V., Lobur M. Model of integral electromagnetic actuator // Прикладна радіоелектроніка. - Т. 5, № 2. - Харків, 2006. - С. 284 - 288.
- [27] Лобур М.В., Теслюк В.М., Колесник К.К. Математична модель для аналізу вихідних характеристик інтегрального давача тиску на компонентному рівні проектування // Електроніка и связь: Тематичний випуск “Проблеми електроніки”. - Ч.2. - Київ: НТУУ “КПІ”, 2005. - С. 109-111.
- [28] В.Варадан, К. Виной, Л. Джозе ВЧ МЭМС и их применения. – Издательство: Техносфера. – 2004. – 528 с.
- [29] Beeby S.P., Ensel G., Kraft M. — MEMS Mechanical Sensors Artech House, Inc. – 2004. – 269 с.
- [30] Теслюк В. М. Застосування методу гілок та границь для зменшення потужності множини альтернативних рішень при розв’язанні задач синтезу МЕМС / В. М. Теслюк, А. Я. Зелінський, Хамза Алі Юсеф Аль Шавабкех // Моделювання та інформаційні технології. Зб. наук. пр. ІППМЕ ім.Г.Є.Пухова НАН України. – Київ, 2009, Вип. 50. – С.157 – 165.
- [31] Hamza Ali Yousef AlShawabkeh Desreasing Alternative Solutions Multitude Power Using Branch and Bound Algorithm During Solving MEMS Synthesis Task / Hamza Ali Yousef AlShawabkeh, V. Teslyuk, A. Zelinsky, P. Denysyuk, K. Bobak // Proc. of the X Intern. Conf. on The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM'2009). – Lviv – Polyana, Ukraine, 2009. – P. 499.
- [32] Guckel H, High-aspect-ratio micromachining via deep X-ray lithography // Proc. IEEE. –1998. – Vol.86 (8). – P. 1586–1593.
- [33] Bustillo J M, Howe R T and Muller R S. Surfacemicromachining for microelectromechanical systems // Proc. IEEE. –1998. – Vol.86 (8). – P. 1552–1574.

- [34] Kovacs G T A, Maluf N I, Petersen K E. Bulk micromachining of silicon // Proc. IEEE. – 1998. – Vol.86 (8). – P. 1536 – 1551.
- [35] Beluse L. et al. Microstereolithography: a new process to build complex 3D objects // Proc. SPIE. – 1999. – Vol. 3680. – P. 808 – 817.
- [36] Maruo S., Ikuta K. Two-photon micro stereolithography with submicron resolution // Proc. Eighth. Intern. Conf. of Rapid Prototyping. – Japan. – 2000. – P. 201 – 206.
- [37] Bertsch A. et al. 3D microfabrication by combining microstereolithography and thick resist UV lithography // Sensors and Actuators. – 1999. – Vol. A73. – P. 14 – 23.
- [38] Корячко В.П. и др. Теоретические основы САПР : Учебник для вузов / В. П. Корячко, В. М. Курейчик, И. П. Норенков. – М. : Энергоатомиздат, 1987. – 400 с.
- [39] Нестеренко М. І. Опис технологій і средств розробки програм на ПК з багатоядерними процесорами на базі методу ветвей і границ [Електронний ресурс]. / М. І. Нестеренко. – Режим доступа: <http://s-journal.cdu.edu.ua/base/2008/v2/v2pp256-257.pdf>. – Дата доступу 21.05.2010.
- [40] Онищенко Б.О. Решение одной задачи стохастической глобальной оптимизации параллельным методом ветвей и границ на кластере // Компьютерная математика. – Вып. 3. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2005. – С. 153-160.
- [41] Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ, 2-е издание. : Пер. с англ. — М. : Издательский дом "Вильямс", 2005. — 1296 с.
- [42] Вирт Н. Алгоритмы + структуры данных = программы. Пер. с англ. – М. : Мир, 1985. – 406 с.

РОЗДІЛ 9.

СУЧАСНІ МЕТОДИ СТРУКТУРНОГО ТА ПАРАМЕТРИЧНОГО СИНТЕЗУ
МЕМС**9.1. Застосування теорії чутливості в процесі автоматизованого
проектування МЕМС**

При розробці МЕМС на кожному ієрархічному рівні виникають задачі, пов'язані з необхідністю покращення проектного рішення, і не завжди при цьому можна використати методи оптимізації. Так, як складові МЕМС функціонують за різними фізичними принципами, то особливого значення набуває теорія чутливості [1, 2], яка дає можливість цілеспрямовано впливати на вхідні параметри пристрою проектування для досягнення заданих, згідно технічного завдання, вихідних параметрів.

Отже, в загальному випадку, під чутливістю розуміють деяку числову величину, яка дає можливість отримати додаткову інформацію про поведінку фізичної системи та визначити як впливають зміни одного параметра на інший [1, 2].

Для визначення коефіцієнта чутливості між j -м вихідним та i -м вхідним параметрами деякого пристрою МЕМС [3, 4], який описується множинами вхідних $x_i (i = 1, 2, \dots, n)$ та вихідних $y_j (j = 1, 2, \dots, m)$ параметрів, використовується похідна від диференційованої функції y по параметру x , а саме:

$$S_{i,j} = \frac{dy_j(x_1, x_2, \dots, x_n)}{dx_i}. \quad (9.1)$$

Значення коефіцієнта чутливості (9.1) не є безрозмірною величиною, що викликає певні труднощі при виконанні різного роду операцій з ним, тому використовують нормоване значення чутливості:

$$S_{i,j}^n = \frac{x_i}{y_j} \frac{dy_j}{dx_i}. \quad (9.2)$$

Слід зауважити, що у випадку рівності нулю одного з параметрів необхідного використовувати напівнормовані значення чутливості:

$$S_{i,j}^n = x_i \frac{dy_j}{dx_i}, \quad (9.3)$$

$$S_{i,j}^n = \frac{1}{y_j} \frac{dy_j}{dx_i}. \quad (9.4)$$

Формула (9.3) використовується при рівності нулю вихідного параметра, а формула (9.4) – вхідного параметра.

В загальному випадку можемо побудувати на основі виразів (9.1 – 9.4) матрицю коефіцієнтів чутливості між вхідними та вихідними параметрами пристроїв МЕМС. Попередньо опустивши знак нормування та приймаючи, що кількість стовпчиків рівна кількості вихідних параметрів, а кількість стрічок – кількості вхідних параметрів об'єкта розробки.

$$S = \begin{bmatrix} S_{1,1} & S_{1,2} & \dots & S_{1,m} \\ S_{2,1} & S_{2,2} & \dots & S_{2,m} \\ \dots & \dots & \dots & \dots \\ S_{n,1} & S_{n,2} & \dots & S_{n,m} \end{bmatrix}. \quad (9.5)$$

Для визначення зміни вихідного параметра при заданій зміні вхідного використовується наступний вираз:

$$\delta y_{j,i} = \left(\frac{\Delta y_j}{y_j} \right)_i = S_{i,j} \frac{\Delta x_i}{x_i}, \quad (9.6)$$

де $\delta y_{j,i}$ - відносна зміна j -го вихідного параметра при зміні i -го вхідного параметра.

При необхідності визначення одночасного впливу всіх вхідних параметрів на j -й вихідний параметр використовують наступний вираз

$$\delta y_j = \frac{\Delta y_j}{y_j} = \sum_{i=1}^n \frac{\partial y_j}{\partial x_i} \frac{\Delta x_i}{x_i}, \quad (9.7)$$

де $\delta x_i = \frac{\Delta x_i}{x_i}$.

Для визначення коефіцієнтів матриці чутливості можна використовувати метод приростів та інші.

Особливістю МЕМС є те, що електромеханічний пристрій володіє як механічною чутливістю, так і електричною (власні чутливості). У випадку давача тиску на основі пластинчастої конструкції будемо мати наступні вирази:

$$S_{mechanical} = \frac{\Delta h}{\Delta P},$$

$$S_{electrical} = \frac{\Delta C}{\Delta h},$$

де ΔP - приріст прикладеного тиску, Δh - приріст зміщення тонкої пластини, ΔC - зміна ємності.

Чутливість давача тиску пластинчастої конструкції рівна

$$S = S_{mechanical} * S_{electrical} = \frac{\Delta h}{\Delta P} * \frac{\Delta C}{\Delta h} = \frac{\Delta C}{\Delta P}.$$

В загальному випадку, чутливість складного пристрою визначається з допомогою наступної формули при відомих власних чутливостях:

$$S_{system} = S_1 S_2, \dots, S_n = \frac{dy_1}{dx_1} \frac{dy_2}{dx_2}, \dots, \frac{dy_n}{dx_n}.$$

Специфіка МЕМС, як правило, полягає в тому, що $x_2 = y_1$, $x_3 = y_2$, ..., $x_n = y_{n-1}$, тобто: вихід перетворювача першого є входом другого, вихід перетворювача другого є входом третього і так далі. Тому, загальний вираз для визначення чутливості, з врахуванням особливості МЕМС, можна переписати в наступній формі:

$$S_{system} = S_1 S_2, \dots, S_n = \frac{dy_n}{dx_1}.$$

As an example, a sensitivity analysis was carried out. The sensitivity to measure was the one between an input and an output parameter of a capacitive microsensor. The structure of the device is given in Fig. 4.11 [5]. The results obtained are shown below [6]: $S_C^L = 0.31$; $S_{C_1}^L = 0.24$; $S_C^h = -0.32$; $S_C^d = 0.0008$.

The results allow to claim that capacitance increase is reached through finger lengthening and widening, and capacitance decrease occurs while moving fingers away from each other.

9.2. Онтологічний підхід до автоматизації структурного синтезу МЕМС

9.2.1. Особливості застосування онтологічного підходу для проектування МЕМС

Інформатизація суспільства, накопичення великих обсягів інформації, впровадження обчислювальної техніки у всі сфери життя та виробництва, потреба у нових промислових виробках та системах зумовлюють необхідність опрацювання великого обсягу даних та виконання низки проектувальних робіт. При створенні (електронних) бібліотек та бібліотек з даними для проектування складних об'єктів та систем, виникає серйозна проблема щодо запису та опрацювання великого обсягу інформації та доступу до неї. Для вирішення цієї проблеми розробляють складні схеми класифікації та правила категоризації при утворенні та зберіганні метаданих. Такий підхід спростив процедуру пошуку потрібної інформації користувачем та підвищив ефективність його роботи загалом. Проте, залишається актуальною проблема представлення метаданих у адаптованому вигляді (зрозумілому) для машинного опрацювання в процесі автоматизованого проектування. Для досягнення зазначеної мети запропоновано використовувати онтологічний підхід із застосуванням спеціальних моделей онтологій.

Для розуміння значення та можливостей прикладного застосування онтологій потрібно розуміти їхні основні визначення, принципи. Проте, до тепер не існує єдиного визначення для поняття «онтології». В перекладі з грец. «онто» означає суще, а «логос» – вчення, поняття, ідея. Давні філософи розуміли онтологію, як науку, що вивчає буття. Переважно значення поняття «онтологія» формується в залежності від області, в якій вона функціонує. Поняттю «онтологія» присвоюється багато значень [7], а саме:

- певна логічна теорія, що складається зі словника та певних тверджень, які відображаються певною мовою логіки;
- основа для комунікації між людиною та комп'ютерними агентами;

- це ієрархічно структурований набір термінів для опису області, який можна використати як базис для формування бази знань та ін.

Також відомо низку інших значень, запропонованих Губертом. Серед них можна виділити такі [7]:

- онтологія – це експліцитна специфікація концептуалізації, де описуються множини об'єктів предметної області та зв'язки між ними;
- онтологія – це база знань, яка описує факти, яким завжди присвоюються значення «істинні» в межах певного соціуму на основі загальноприйнятого змісту прикладного словника;
- онтологія – це знання, які формально представлені на базі концептуалізації. З формальної точки зору, онтології складаються із термінів, що організовані в таксономію, їх визначень та атрибутів, аксіом та правил виведення;
- онтологія – це формальна специфікація сепаративної специфікації, яка характеризує певну предметну область.

Однак, в найбільшій мірі дослідники штучного інтелекту користуються класичним визначенням онтології, запропонованим Губертом: «онтологія – це специфікація концептуалізації».

Інколи спеціалісти комп'ютерної лінгвістики присвоюють онтології такі значення, як штучний словник, таксономія чи база знань. Проаналізувавши вище сказане, можна стверджувати, що синонімія даних понять спричинена деякими спільними властивостями.

Штучний словник використовується переважно в лінгвістиці. Це набір спеціалізованих термінів, які семантично належать до певної області та якими оперує користувач в даній області. Прикладом штучного словника є тезаурус. Поняття «таксономія» є певною класифікацією реальності, реального світу тощо, яка складається з набору певних об'єктів чи термінів, які мають ієрархічну структуру. В штучному інтелекті база знань складається з двох частин: термінологічної частини (Т-Box), частини суджень (А-Box). Термінологічна частина формується набором концептів і їхніх значень. Частина суджень утворюється набором прикладів даних концептів та суджень між ними.

Спільною властивістю онтологій та баз знань є можливість представлення знань.

Онтології дають змогу подати нові поняття зрозуміло для системи і поєднати їх з уже існуючими (записаними) описами вже відомих класів, об'єктів, відношень та властивостей реального світу. Так як будь-яка онтологія складається з певних компонентів, які залежать від парадигми представлення знань. Переважно всі моделі онтологій включають у себе концепти (поняття, класи, сутності, категорії), властивості концептів (слоти, атрибути, ролі), відношення між концептами (зв'язки, залежності, функції), обмеження (визначальні аксіоми, фасети), що уможливорює повноцінний опис будь-якої предметної області.

При класифікації моделі онтологій можна розподілити таким чином:

- прості онтології (складаються лише з концептів);
- онтології, побудовані на основі фреймів (складаються лише з концептів та їх властивостях);
- онтології, побудовані на основі логік (наприклад Ontolingua, DAML-OIL).

Щодо відношень онтологій, то вони відображають типи взаємодій між концептами певної предметної області. Прикладом бінарного зв'язку служить відношення «є частиною» (is_part). Аксіоми використовуються для моделювання тверджень, які за сутністю є істинними.

9.2.2. Особливості побудови онтологічної моделі

При побудові будь-якої моделі онтологій слід розуміти та базуватись на таких основних принципах [8]:

1. Зрозумілість (Clarity). Терміни (і поняття) онтологій повинні відображати реалії світу. Їхнє символічне представлення повинне створюватись на основі загальноприйнятих правил семіотики, і вони повинні виражати встановленні значення реальних об'єктів. Ці значення отримують із **словників**

ПрО. Судження формалізуються на основі формального апарата у вигляді істинних за значенням аксіом.

2. Розширюваність (Extensibility). На початковій фазі ядро будь-якої онтології формується введеними поняттями та аксіомами, які описують ці поняття. В онтології повинен бути передбачений механізм розширення (обмеження) залучених словників понять без порушення цілісності системи загалом.

3. Обґрунтованість, зв'язність (Coherency). Формулювання початкового набору понять онтології і їхнє подальше додавання повинне бути обґрунтоване і визначене переліком завдань, які мають бути вирішені. Логічні аксіоми початкового набору понять не повинні суперечити між собою щодо встановлених ними обмежень.

4. Мінімальний вплив кодування (minimal encoding bias). В онтології має бути реалізований принцип сумісного використання декількох онтологій, який передбачає специфікацію онтологій на рівні представлення, а не на рівні символічного кодування. Таку специфікацію, здійснену на загальноприйнятому та платформо незалежній мові при описі онтології, може застосовувати будь-який програмний агент чи користувач.

5. Мінімальні онтологічні зобов'язання (Minimal ontological commitment). За змістом цей принцип схожий на принципи обґрунтованості та розширюваності/обмеження. Важливим фактором для будь-якої онтології є представлення структури предметної області з допомогою множини понять онтології. Процес відокремлення концептуальних знань від фактичних формує стратегію побудови онтологічних систем, онтологічних баз даних.

9.2.3 Аналіз моделей та основних методів представлення онтологій

Традиційно модель онтології можна відобразити таким чином: $O = \langle T, R, F \rangle$ – це три впорядковані множини, де T – терміни предметної області (ПрО), яку описує онтологія (O); R – відношення між термінами заданої предметної

області (ПрО); F – функції інтерпретації, які задані на термінах і/чи відношеннях онтології.

Проте, Аманн та Фундулак [9] запропонували таку математичну модель представлення, яка складається з трьох членів: $O = (C, S, isa)$; де 1) $C = \{c_1, c_2, \dots, c_m\}$ – це набір класів, в якому кожен клас c_i відповідає певним об'єктам реального світу (екземплярам класу); 2) $S = \{s_1, s_2, \dots, s_n\}$ – це набір слотів, де кожен слот s_i може відноситись до: а) властивість класу за значенням простого типу; б) подвійний зв'язок (представлення відношення між класами); 3) $isa = \{isa_1, isa_2, \dots, isa_p\}$ – це набір успадкованих відношення між класами. Успадковані відношення відображають семантику підмножин (класів) та визначають частковий порядок класів, що формує одну або декілька дерево подібних структур.

Для кращого розуміння та представлення окремих екземплярів (класів) пропонуємо ввести ще один член-елемент $I = \{i_1, i_2, \dots, i_q\}$ до вищенаведеної моделі, де i_w є екземпляром певного класу $c_x \in C$. Такий екземпляр включає конкретне значення для кожного слоту s_y , що асоціюється з класом c_x чи попереднім класом.

Загальновідомі методи візуалізації, в основному, представляли онтології у дерево подібному чи графовому вигляді. Вони, наприклад, використовувались для візуалізації системи організації файлів чи категоризації документів тощо.

Потрібно розуміти, що методи представлення онтологій потребують при їх практичному застосуванні певних модифікацій відповідно до специфіки предметної області. Тому припустимо, що такі методи мають представляти складові онтологій, а саме класи, зв'язки (відношення), екземпляри, властивості (чи слоти). Наприклад, пряму рівнозначність між об'єктами файлової системи, категоризованими документами. Відповідно до зазначених вимог та на основі різних характеристик відображення онтологій, а саме технік взаємодії, функціональності чи розмірної візуалізації запропоновано категоризувати методи представлення онтологій таким чином:

1. Дворівневий список.

2. Дерево подібне та вузлове представлення.
3. Масштабоване представлення.
4. Просторове представлення.
5. Фокусне представлення або деформація.
6. 3D інформаційна архітектура.

Метод дворівневого списку. Більшість систем представлення онтологій (Protege, OntoEdit, Kaon, OntoRama) при описі онтологій формують дерево подібну структуру. Такі способи репрезентації онтології відображають їх, як набір вузлів, що утворюють вертикальну чи горизонтальну структури. Таке представлення дає змогу користувачу при потребі деталізації інформації розширювати чи видаляти певні вузли і сформовані ними дерева.

Методики методу дерево подібного та вузлового представлення полягає в представленні онтологій як набору взаємопов'язаних вузлів, що утворюють вертикальну чи горизонтальну структури. Таке представлення дає змогу користувачу при потребі деталізації інформації розширювати чи видаляти певні вузли і сформовані ними дерева. Методики даної категорії поділяються на дво- та трьохрозмірні. До перших належать:

1) OntoViz є модулем візуалізації редактора Protégé, який використовує бібліотеку GraphViz для побудови звичайного 2D графа [10].

2) IsaViz є середовищем представлення RDF онтологій з допомогою орієнтованих графів [10].

До других належать:

1) Cone Tree [11] – це спосіб представлення онтологій, який базується на 3D графові, що утворює за принципом «знизу-вверх» 3D дерево.

2) Reconfigurable Disk Tree [12] є розширеною варіацією Cone Tree, в якій користувач має змогу змінювати висоту конуса кожного піддерева з метою покращення зрозумілості та читабельності всіх вузлів.

3) Tree Viewer [13] утворює дерево подібну структуру у формі реального дерева.

4) OntoSphere [14] пропонує використовувати вузловий зв'язок між трьома різними онтологіями для їх опису та деталізації відповідно до потреб користувача.

Категорія методу масштабованого представлення складається з методик, які дають змогу відобразити вузли початкових рівнів ієрархії, що входять до складу «батьківських» вузлів. До таких методик належать дво- та трьохрозмірні способи представлення. Розглянемо їх більш детально. До дворовмірного відносяться:

1) Grokker [15], [<http://www.groxis.com>] – це репрезентаційна система карт знань. Вона графічно відображає інформацію, наприклад, як результати роботи пошукової системи чи системи пошуку файлів загалом.

2) Jambalaya [16] – візуалізаційний модуль Protégé, що використовує техніку представлення 2D SHriMP.

3) CropCircles [17] є методом представлення онтологій, що відображає ієрархію класів, як набір концентричних кіл.

До трьохрозмірного відносять:

1) Information Cube [18] використовує вкладені куби для відображення категорій об'єктів, що формують ієрархію.

2) Information Pyramids [19] відображають онтологію та її складові у вигляді пірамід, які мають плоску верхівку та розміщуються одна на одній.

Суть методу просторового представлення полягає в розподіленні концептів, а саме вузлів та їх похідних вузлів в допустимих межах екрану. Розмір кожного розподілу залежить від властивості, яку відображає вузол, його розміру, числа вміщених похідних вузлів тощо. Цей метод поділяють на дво- та трьохрозмірні. До першої категорії відносяться:

1) Метод візуалізації TreeMaps [20] ґрунтується на способі 2D з використанням прямокутної області, що складається з прямокутних секцій, для представлення ієрархій класів-об'єктів.

2) SequoiaView [21] відображає ієрархію об'єктів онтологій, як дерево подібні структури TreeMap.

3) Техніка візуалізації Information Slices [22] використовує напівкруглі диски для більш компактного відображення великих ієрархій класів в 2D просторі.

До трьохрозмірних відносять:

1) BeamTrees [23] об'єднує метод візуалізації Treemaplike та метод 3D візуалізації вузлового зв'язку.

Методики візуалізації методу фокусного представлення або деформації базується на понятті деформації графу для об'єднання контексту та фокусу. Вузол фокусу є основним (батьківським) вузлом, від якого відходять вузли в послідовності зменшення їх розміру. Традиційно такий спосіб представлення описується гіперболічним рівнянням. Цей метод ґрунтується на дво- та трьохрозмірного представленні онтологій. До дворозмірних відносяться:

1) 2D hyperbolic tree використовується для відображення онтології бразильського товариства з питань сільськогосподарських досліджень. Гіперболічне дерево базується на гіперболічній трансформації. Основа дерева розміщена в середині області круга, від якої відходять похідні вузли.

2) OntoRama [24] є Java аплікацією, яка гіперболічно відображає структуру онтології.

До трьохрозмірних відносять:

1) Метод візуалізації 3D Hyperbolic Tree був створений для відображення веб-сайтів та застосовувався як браузер файлів також.

В середовищах віртуальної реальності досить поширеним для управління документами є поняття «інформаційної метафори», де в переліку документів документи розміщуються відповідно до (зазначеного) кольору та розміру закодованих 3D об'єктів. До категорії методу інформаційної архітектури відносять лише дві системи, які дещо відрізняються одна від одної:

- File System Navigator (FSN) [25], в якій висота вузла відображає кількість вміщених файлів, щодо онтологій – екземпляри;
- Harmony Information Landscape [26], створений для опрацювання гіпертекстів.

9.2.4. Розроблення моделі онтології для елемента, підсистеми та системи

Метод автоматизованого проектування потребує моделі онтології, на основі якої повинні формуватись знання про складові мікросистем, їх властивості, зв'язки між рівнями MEMC, відношення між елементами тощо. Загальновідомо, що для проектування MEMC використовується блочно-ієрархічний підхід («згори-донизу», «знизу-догори», «паралельне розроблення») [27], який передбачає використання принципу ієрархічності для структурування представлень про об'єкти за ступенем деталізації описів та принцип декомпозиції для розбиття представлень кожного рівня на завершені блоки з можливістю їх по блокового аналізу та синтезу [27]. Поділ на блоки виконується, як правило, за функціональною ознакою. Інформаційна технологія аналізу та синтезу MEMC потребує чотири рівні ієрархії (див. рис.9.1) – системний, схемотехнічний, компонентний та елементний [28].

Це дає змогу нам описати кожен рівень як окрему модель, що складається з відповідних блоків певного рівня розбиття. Загалом чотирирівневу модель онтології гетерогенних САПР, задіяних у розроблення MEMC можна описати з допомогою наступної моделі []:

$$S_{CAS}^1 = \bigcup_{j=1}^n S_{CAS}^{2,j} \bigcup_{l=1}^{K_j} S_{CAS}^{3,l} \bigcup_{z=1}^{Z_l} S_{CAS}^{4,z},$$

де S_{CAS}^1 – верхній рівень CAS, $S_{CAS}^{2,j}$ – третій рівень CAS, $S_{CAS}^{3,l}$ – другий рівень CAS, $S_{CAS}^{4,z}$ – перший рівень CAS, j – номер блоку 3-го рівня розбиття ($j=1,2,\dots,n$), K_j – кількість блоків другого рівня, l – номер блока другого рівня, Z_l – кількість блоків першого рівня.

Для повторного використання онтологій MEMC запропоновано розділити таксономічно організовану термінологію MEMC, виходячи з принципів диз'юнктивності, на основі загальноприйнятих формулювань, які описують оточуючий світ, та конкретних вимог щодо моделі даних, що застосовується для проектування, аналізу, технічних даних обміну та валідації документації.

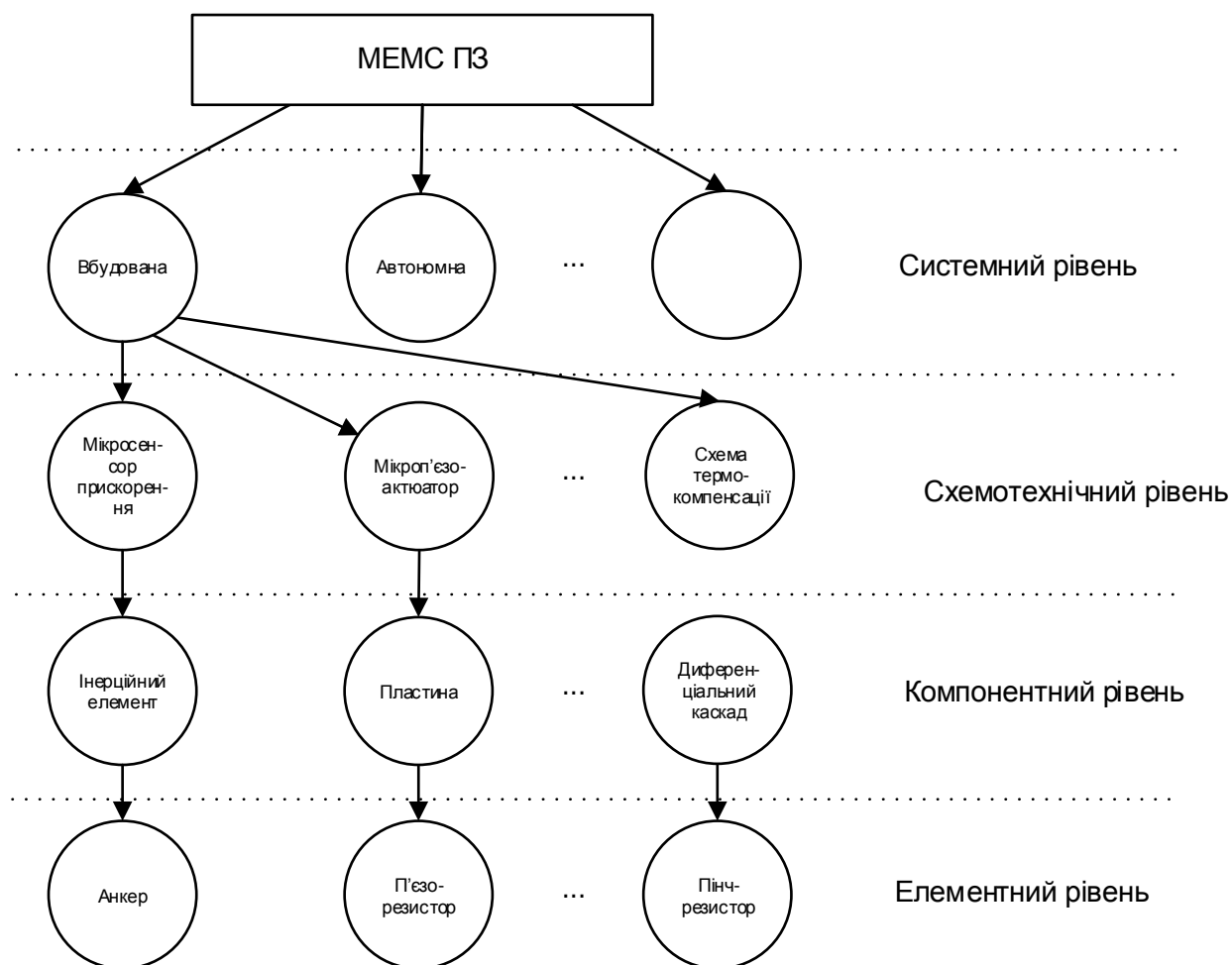


Рис. 9.1. Паралельне проектування елементів MEMC

Базуючись на вищенаведених принципах в роботі запропоновано застосовувати чотирьохрівневу модель онтології класів та екземплярів (ланок) (рис. 9.2).

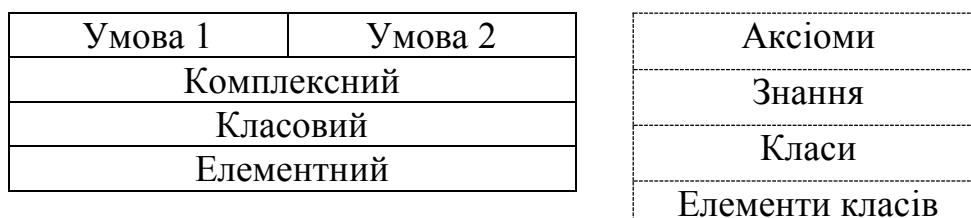


Рис. 9.2. Чотирирівнева модель онтології для класів та екземплярів

Перший рівень – елементний. Він складається з елементів, атрибутів та властивостей класів. На даному рівні наведено класифікацію та опис характеристик певного елемента MEMC, наприклад анкера чи пінчрезистора (рис. 9.2).

Другий рівень – це рівень моделей онтологій, який складається з класів, що утворюють таксономічне дерево – термінологію області та відношення у вигляді OWL сутностей, а саме в даному випадку класу, властивостей об'єкту, типу даних властивості, що асоціюються з точним визначенням. Відповідно до ідеї нормалізації Алана Ректора [29] кожна сутність має одну вищу сутність типу «батько», формуючи таким чином дерево примітивного характеру. Так як побудова такого дерева з автономних концептів є процесом нечітким, невизначеним та складним, в даній роботі запропоновано зменшити важливість критерію диз'юнктивності у такому процесі, зважаючи на можливість потенційної несумісності. Таке упущення пов'язане з можливим розширенням класового рівня, на якому відбувається реалізація екземплярів класу у складні класи. Прості рівні дають змогу користувачу отримувати користь від таксономічно організованих термінологій області та змінювати семантику рівня, що може призвести до несумісності між додатковими знаннями. Більш того, користувач може обмінюватись даними визначених класів та відношень.

Третій рівень – комплексний. Він містить необхідні чи відповідні до умов знання певної предметної області. Комплексний рівень доповнює класовий рівень введенням певних обмежень-умов щодо категоризації. Це такі аспекти OWL-DL мови [30, 31]: диз'юнкція, екзистинційні та універсальні обмеження, об'єднання, перетин, доповнення, обмеження потужності, транзитивність та гнучкість властивостей тощо. Комплексний рівень посилює зв'язок онтології з знаннями області, що спричиняє появу більш реалістичних припущень щодо рішень поставлених проблем та досягнення цілей проекту, та включає співставлення з вищим рівнем онтологій.

Четвертий рівень утворюється за допомогою встановлення додаткових аксіом (обмежень/умов), які потрібні для достатнього опису певної моделі даних. На даному рівні наводяться правила можливого об'єднання об'єктів системи при проектуванні, правилами відношень між об'єктами та їх даними. Загалом, даний рівень визначає структуру знань, які отримуються при задіянні декількох об'єктів та дають змогу модифікувати рішення реалізації.

Таким чином, при застосуванні такої моделі OWL онтології розробник може отримати додаткові знання при взаємодії різних елементів системи, наприклад анкера та п'єзоризистора, які дадуть змогу повторно використовувати їх в майбутніх дослідженнях.

Модель знань 4-х рівневої онтології складається з концептуальних знань та фізичних знань про продукт. Перший тип знань стосується основних концептуальних знань, що стосуються процесу проектування продукту (формує знання про перебіг проектування, знання про планування процесу вироблення продукту та спільні знання). Фізичні знання описують фізичні знання суті перебігу процесу проектування та поділяються на такі типи [32]: знання «що», знання «чому», знання «як», знання «коли», знання «хто», знання «де», знання «з допомогою чого» та знання комунікації. Кожен тип знань представляють в форматах XML/HTML [32], що дає змогу зберігати та використовувати в подальшому такі знання інженерами МЕМС області.

Для прикладу концептуальні та фізичні знання такого пристрою як «давач тиску» можна описати за схемою (рис.9.3), яка складається [32]: знання про концепт «давач тиску», основне значення концепту, його визначення, синонімічний ряд концепту, фізичні знання концепту, відношення концепту з іншими концептами тощо.

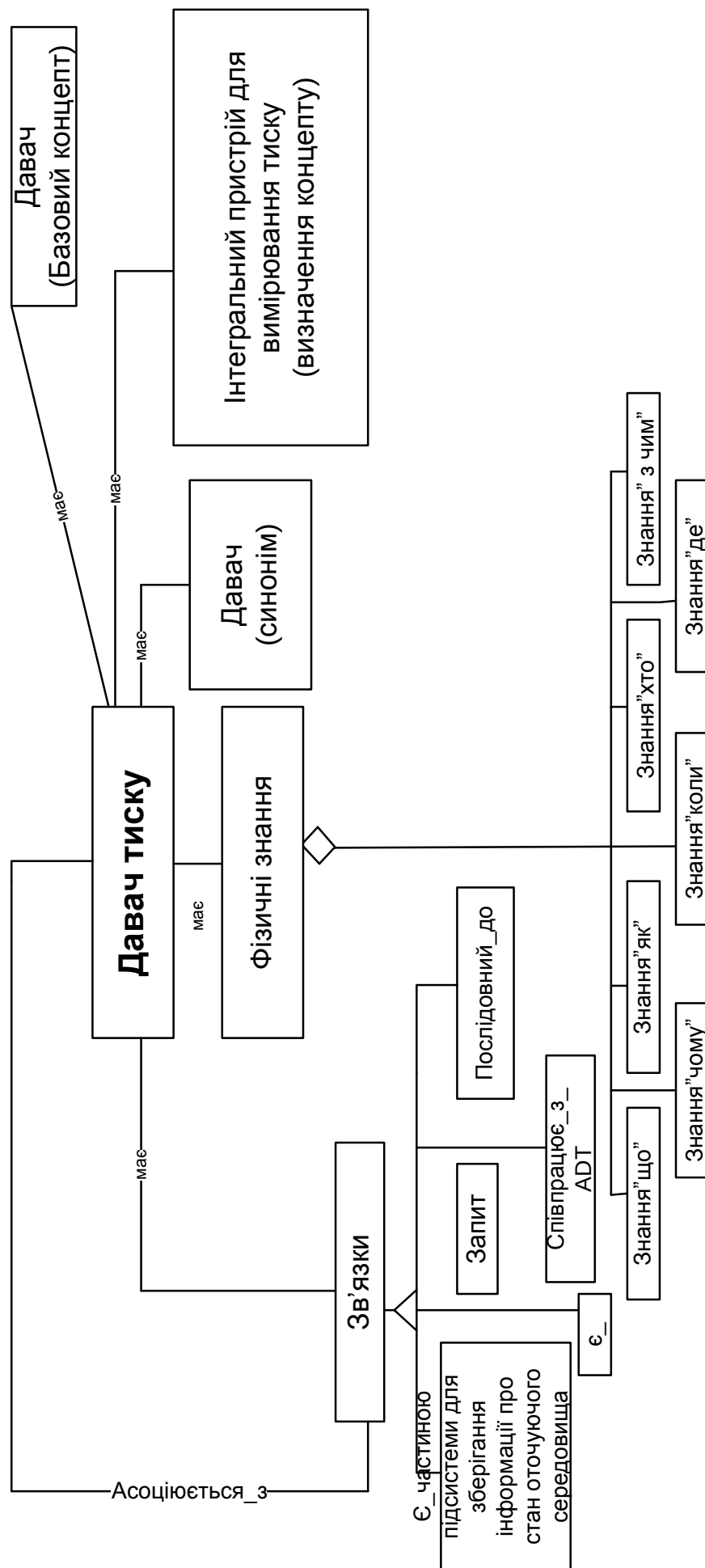


Рис. 9.3. Опис знань концепту «давач тиску» (онтологія концепту)

Таким чином, розроблена чотирьохрівневу модель онтології дає змогу детальніше збирати та описувати інформацію кожного рівня моделювання, аналізу та синтезу, що прискорить рівень автоматизації проектних процедур, збільшить кількість повторного використання знань та даних кожного рівня, уможливить розвиток, розширення онтологій та сформує більш сумісну структуру та зв'язки між поняттями різних Computer-aided systems.

9.2.5. Приклади застосування онтологій для проектування MEMC

На початковому етапі розроблення програмної реалізації підсистеми роботи з моделями онтологій доцільно сформулювати перелік вимог, які повинні вирішувати поставленні онтологічні задачі системи проектування мікроелектромеханічних систем загалом [33]. На основі таких вимог розроблено архітектуру програмного модуля взаємодії з онтологією. Проаналізувавши їх було виокремлено такі задачі:

- автоматизоване створення концептів моделей онтологій через різні джерела постачання даних та знань (мануальний, документи різних форматів, конструкторські бази даних) ;
- формування структурованої ієрархії онтології;
- інтерактивна взаємодія з модулем синтезу MEMC;
- реверсне перетворення концептів онтології в об'єкти об'єктно-орієнтовано мови;
- трансформація даних з формату *.xml, *.owl, *.json;
- зберігання, модифікація даних в онтології;
- контролювання цілісності знань онтології.

Відповідно, базуючись на таких вимогах було визначено перелік сервісів, які потрібно реалізувати в програмному модулі роботи з онтологією. Кожен сервіс розв'язує одну задачу. Такий підхід дає змогу в подальшому розширювати функціональність системи загалом та робити її більш гнучкою до інтеграції з іншими системами автоматизованого проектування.

Традиційний підхід до архітектури будь-якої програмної системи ґрунтується на трьох рівнях:

- рівень презентації (користувачський інтерфейс, де презентується ієрархія класів онтології MEMC (рис. 9.4));
- рівень опрацювання логіки (здійснюється трансформація даних в залежності від вхідного типу, підготовка даних до збереження в онтології MEMC, створення та модифікація концептів);
- рівень роботи з базою даних (реалізація CRUD операцій, отримання концептів онтології у відповідності до пошукового запиту).

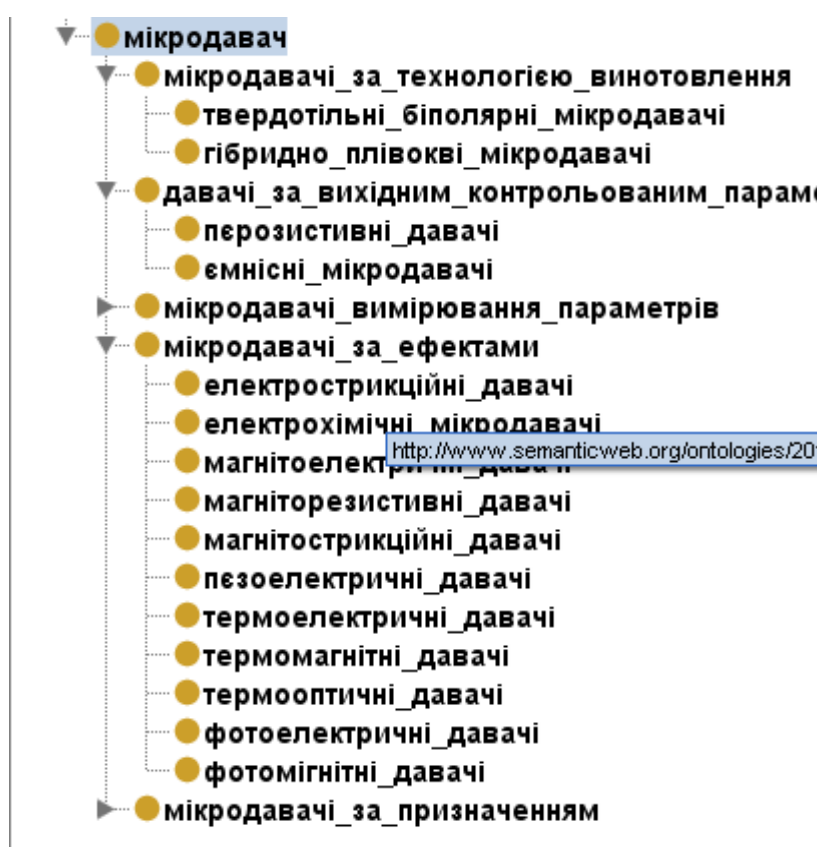


Рис. 9.4. Ієрархія класів онтології MEMC

Проте, в програмному модулі взаємодії з онтологіями додатково реалізовано ще один модуль – модельний рівень, який призначений створювати дерево концептів в онтології та відповідно ініціалізувати об'єкти об'єктно-орієнтовної мови. На рис. 9.5. представлено архітектуру програмного модуля взаємодії з онтологією MEMC:

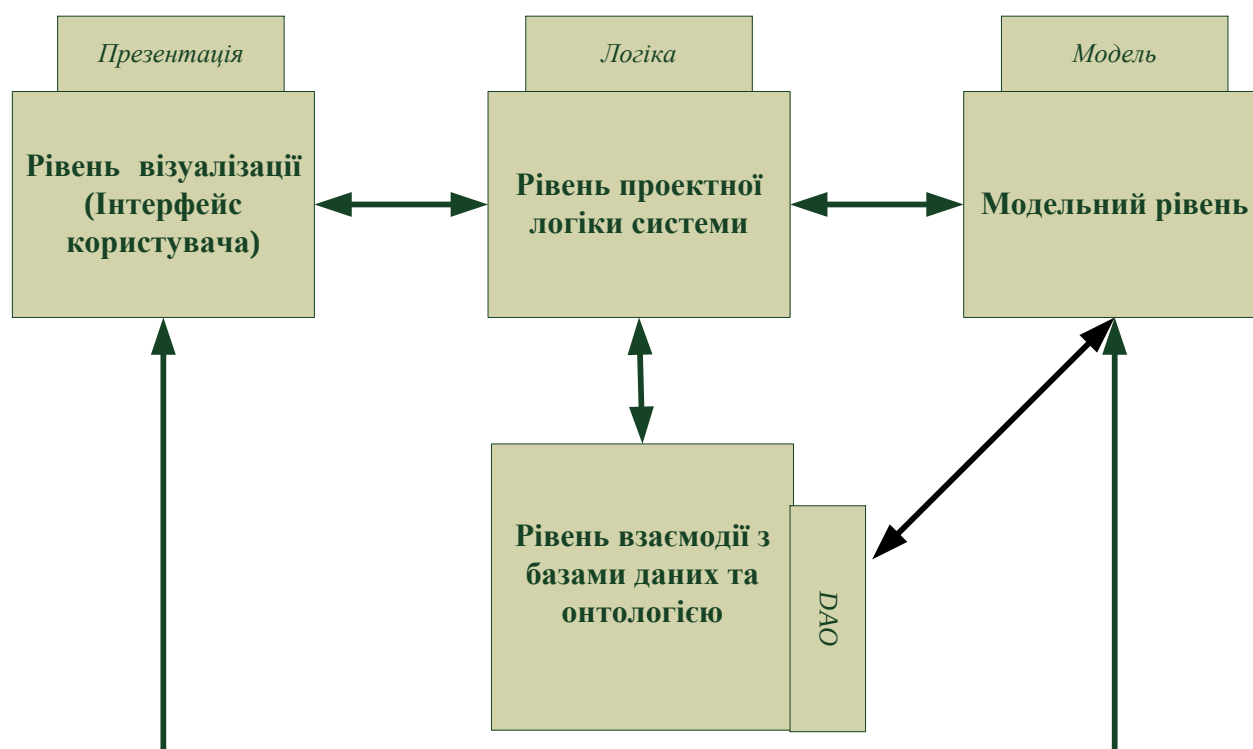


Рис. 9.5. Чотирирівнева архітектура модуля роботи з онтологією

Таким чином, такий підхід до архітектури системи спростив процедуру внесення будь-якої програмної зміни в модулі взаємодії з онтологією MEMC, зменшену зв'язність між рівнями системи, можливість використання однієї презентації з різними моделями-даними, розпаралелення транзакційних процесів до онтології MEMC, одночасне опрацювання запитів від декількох користувачів.

Описавши загальні переваги архітектури програмного модуля роботи з онтологією мікросистем, потрібно більш детально описано сервіси, які реалізовані програмно в нашій системі та які забезпечують повну функціональність відповідно до сформованих вимог до автоматизованої підсистеми роботи з онтологією мікросистем.

На рис. 9.6. представлена діаграма послідовності виклику програмних операцій в залежності від дій користувача, які здійснюються через зовнішній інтерфейс.

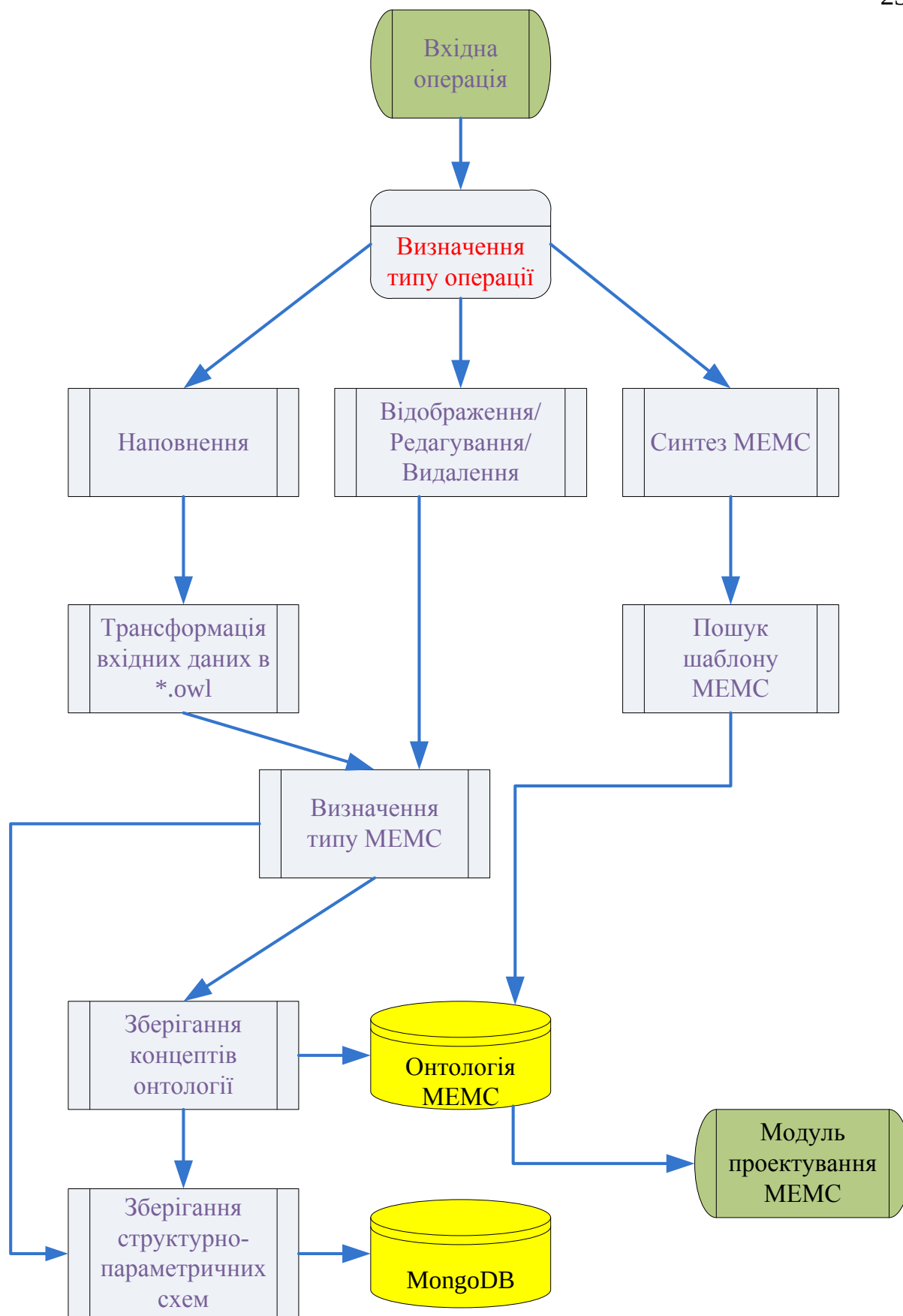


Рис. 9.6. Діаграма операцій програмного модуля роботи з онтологією MEMC

Існує три сценарії, які опрацьовуються в підсистемі:

- наповнення даними та знаннями онтології МЕМС;
- відображення, редагування концептів онтологічних моделей;
- видалення онтологічних концептів (операція здійснюється тільки адміністратором системи);
- синтез нового проектного рішення структури мікросистеми.

Завдяки 4-ри рівневій архітектурі на логічному рівні модуль автоматично розрізняє який сценарій потрібно почати функціонально виконувати, ґрунтуючись на даних зовнішнього запиту (рівня візуалізації).

9.2.6. Особливості реалізації програмного забезпечення модуля роботи з онтологією

Виходячи з чотирьохрівневої архітектури підсистеми роботи з онтологією мікросистем, кожен рівень передбачає застосування специфічних бібліотек та додатків в процесі реалізації системи. Для рівня представлення було застосовано такі технології:

- JOWL (додаток, призначений для навігації та візуалізації OWL-RDFS документів, які формуються з ієрархії класів, екземплярів, їхніх властивостей та зв'язків між ними) [34];
- JQuery (javascript додаток, який дає змогу здійснювати маніпуляції з HTML документами, опрацьовувати веб сторінки відображення онтології МЕМС, зв'язувати серверну логіку з клієнтською стороною, посилати клієнтські AJAX запити на серверну частину) [35];
- Backbone.js [36] (дає можливість структурувати розроблену веб-аплікацію за принципом кожен об'єкт є моделлю типу «ключ:значення», реалізовувати користувацькі події (custom web events) та логіку їх опрацювання, зв'язувати і передавати дані за допомогою REST вебсервісів у форматі JSON);
- Underscore.js (бібліотека, яка забезпечує набором функцій для спрощення роботи з колекціями веб об'єктів, об'єднання логіки декількох

функцій в одну, javascript шаблонізацію сторінок, використання на стороні клієнта серверних функцій напрямую);

Судячи зі низки сервісів, які реалізовані в системі автоматизованого проектування МЕМС загалом та модуля взаємодії з онтологією, для імплементації логічної частини системи, обрано об'єктно-орієнтовну мову програмування JAVA [37]. Завдяки нижченаведеним перевагам JAVA уможливорює реалізацію всіх поставлених задач, від додавання даних в онтологію до синтезу нових структурно-параметричних схем мікросистем:

- платформи незалежна;
- наявність низки бібліотек для роботи з семантичними фреймворками;
- інтеграції з онтологічними середовищами;
- реверсне перетворення концептів онтологічних моделей в JAVA об'єкти.

Відомо декілька платформ JAVA: JAVA EE (від 1.5 до 4.3) – для створення багаторівневих систем, які покликані вирішувати складні завдання, що передбачають інтеграцію з різними базами даними, іншими аплікаціями тощо; JAVA SE (від 1.5 до 4.3) – призначена для розроблення програмного забезпечення для настільних програмних систем, веб систем; JAVA ME (від 1.5 до 3.4) – призначені для реалізації програмного забезпечення для мобільних телефонів переважно. Ядро системи автоматизованого проектування МЕМС та рівень проектної логіки написані на основі платформи JAVA SE 1.7. Додатково, для спрощення імплементації логічної частини підсистеми роботи з онтологією було використано такі технології:

- Spring APIs (відкритий програмний фрейворк, який дає змогу проводити аутентифікацію та авторизацію користувачів САПР МЕМС, конфігурувати компоненти аплікації та її об'єктами, забезпечувати взаємодію з базами даними, розпаралелювати транзакції, здійснювати оповіщення користувачів через різнотипні канали зв'язку, налаштовувати характер роботи сторонніх («Third-parties») бібліотек);

- Hibernate [38] (об'єктно-реляційний фреймворк (ORM-framework), який дає змогу автоматизувати процеси звертання до баз даних та створення JAVA об'єктів в базах даних);
- Guava APIs (спрощує роботу з колекціями JAVA об'єктів, дає можливість написання універсальних функцій, розв'язання багато потокових задач, кешування, роботу з вхідними та вихідними даними) [39];
- Maven (програмне забезпечення, що автоматизує процес управління збирання розробленого програмного продукту для проектування мікросистем, додавання нових бібліотек, компіляцію, створення jar, генерації документації проекту) [40];
- BoneCP [41] (забезпечує управління з'єднаннями з базами даних MongoDB, конфігураційною базою даних CAIP MEMC);
- Rest (Representational State transfer) API [42] (забезпечує отримання, модифікацію та видалення даних та знань з онтологію через URI (Unified resource identifier), який присвоюється кожному елементу онтології MEMC);
- Liquibase API [43] (проводить автоматично керувати версіями баз даних та змінювати структурну схему даних);

Для створення онтологій та проведення маніпуляцій з її даними та знаннями про мікроелектромеханічні системи було використано середовище розробки Protégé (версія 4.2) [44]. Protégé – це графічний редактор онтологій, який характеризується гнучкістю та модульністю, що забезпечується наявністю і вільному доступу до низки додатків-програмних плагінів та бібліотек, та відкритістю архітектури. Він дає можливість оперувати великим набором операторів для роботи з класами онтологій, декларативно представляти дані та знання в форматі, як вони описані і тлумачаться в реальному світі в предметній області, додавати знання з інших онтологій, формувати цілісну ієрархію класів, зв'язки між екземплярами класів, встановлювати умови для елементів одного рівня та їхніх властивостей, визначати типи властивостей класів та екземплярів тощо. На рис. 9.7. представлено інтерфейс Protégé.

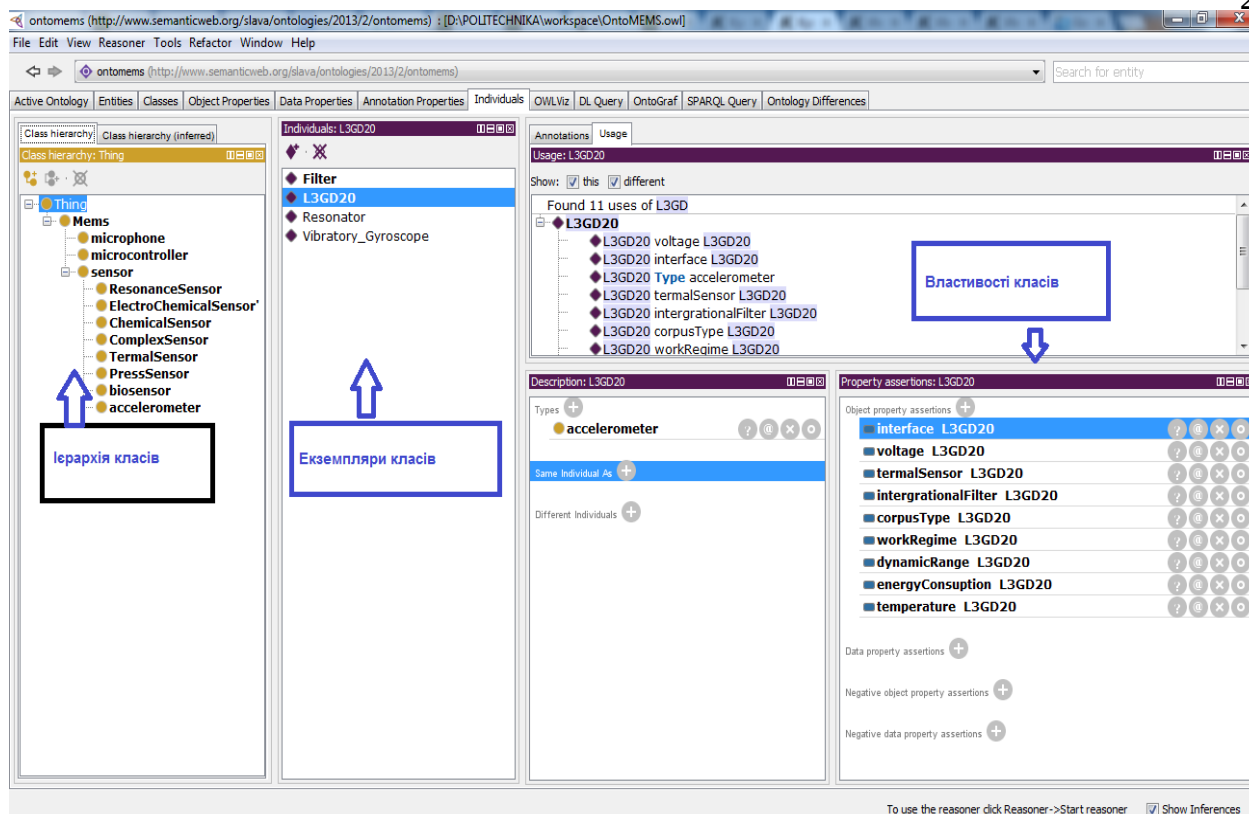


Рис. 9.7. Інтерфейс редактора онтологій Protege

Дані та знання в онтології представляються за допомогою мови OWL, розробленою Semantic Web, що базується на форматі RDF [45], який є розширенням до традиційного формату XML. Формат RDF дає змогу створити просту модель даних про MEMC, доступатись до кожного класу онтології через унікальний шлях – URI (Unified Resource Identifier) [46], застосовувати при описі онтології мікросистем XML формат. Ядром структури RDF вважається триплет (рис.9.8).

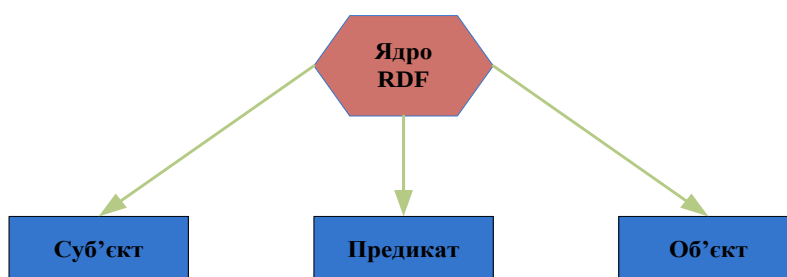


Рис. 9.8. Структура формату RDF

В наведеній схемі Суб'єкт – це опис реальний об'єкт предметної області, який описується; Предикат – це властивість об'єкта предметно області; Об'єкт – це значення властивості об'єкта предметно області.

Таким чином, акселерометра можна описати як:

- Акселерометр – це клас онтології MEMC;
- L3GD20 – це екземпляр, який відноситься до класу акселерометр;
- L3GD20 – володіє властивістю «давач температури»;
- Давач температури – 8 біт, літерал.

Предметна область мікроелектромеханічних систем є складною і охоплює низку термінів. Тому для опису MEMC було використано OWL, яка дає можливість представляти комплексну інформацію про різні класи та індивіди мікросистем. Всі компоненти MEMC, відношеннями між ними трактувались в перерізі OWL як окремі сутності. За допомогою OWL було згруповано дані та знання в онтологію у вигляді документів з розширенням *.owl, та створено таксономію області мікросистем. Таке представлення дає змогу повторно використовувати накопичені знання в іншому програмному забезпеченні. Оскільки формат представлення owl схожий на xml.

Наступним ключовою задачею в реалізації взаємодії між онтологією MEMC та програмною підсистемою є реверсна трансформація онтологічних об'єктів в об'єктно-орієнтовні об'єкти. Цей етап є складним та важливим, оскільки від нього залежить цілісність роботи системи загалом, правильність функціонування всіх інших модулів системи автоматизованого проектування MEMC, представлення даних як в онтології, так і в базі збережених проектних рішень мікросистем.

Тому реалізовано програмну процедуру автоматичного конвертування онтологічних даних в класи JAVA. На рис. 9.9 наводить схема відповідності елементів онтології та об'єктних моделей.

Таким чином, розроблений та програмно реалізований модуль роботи з онтологією забезпечує виконання всіх поставлених вимог, які передбачають нормалізоване опрацювання вхідних даних та знань про MEMC, додавання

нової інформації про мікросистеми, автоматичне проведення реверсних конвертацій даних з формату owl, xml для можливості подальшого використання.

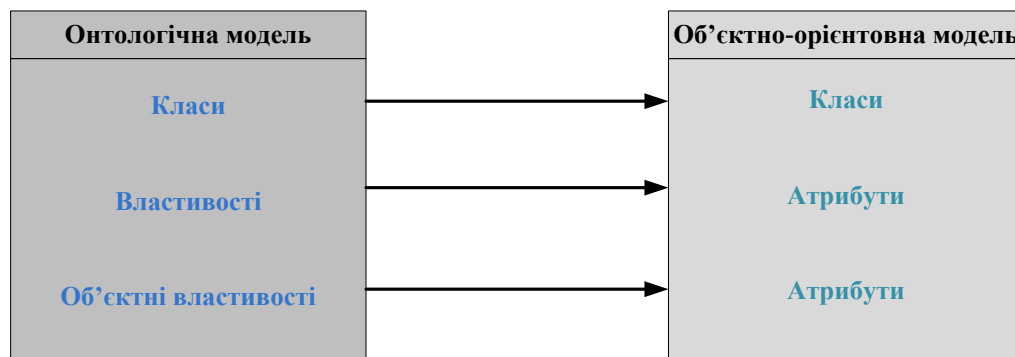


Рис. 9.9. Відповідність онтологічної моделі до об'єктно-орієнтовної моделі

9.2.7. Результати роботи підсистеми взаємодії з онтологією

Суть процедури трансформації полягає в точному мапінгу об'єктів онтологічної парадигми в об'єктно-орієнтовну парадигму. У випадку, наприклад, при додаванні нових знань дані зберігаються не тільки в базі знань онтології, а й в базі проектних рішень, тому потрібно правильно сформулювати JAVA об'єкт типу *Document*, який описуватиме структуру елементів мікросистеми, відповідні значення елементів, значеннєві обмеження на властивості та правила поєднання елементів MEMC. Для прикладу розглянемо трансформацію даних про екземпляр L3GD20 онтологічного класу "accelerometer". Представлення даного індивіду онтології на мові OWL відображено на рис. 9.10.

Наступним етапом роботи підсистеми є автоматична конвертація вхідного документу owl в JAVA об'єкт типу *DefaultAccelerometer*. Характерною рисою процедури конвертації онтологічних даних в JAVA об'єкти є створення об'єктів зі стандартними методами отримання атрибуту ("*setAttribute()*") та встановлення ("*getAttribute()*") значення атрибуту. Атрибути утворюють клас типу *ENUM Vocabulary*, де і оголошується їхній перелік відповідно до

онтологічного екземпляру, та в конструкторі кожного передається унікальний ідентифікатор (URI) доступу до елемента в графі елементів онтології, в даному випадку L3GD20. JAVA об'єкт типу *Vocabulary* зберігається відповідність між ієрархічними рівнями моделі онтології. Базуючись на такому результаті, можна ствердно сказати, що модуль реверсної взаємодії з онтологією функціонує правильно.

```

225     <rdf:type rdf:resource="&ontomems;Mems"/>
226   </owl:NamedIndividual>
227
228
229
230   <!-- http://www.semanticweb.org/slava/ontologies/2013/2/ontomems#L3GD20 -->
231
232   <owl:NamedIndividual rdf:about="&ontomems;L3GD20">
233     <rdf:type rdf:resource="&ontomems;accelerometer"/>
234     <voltage rdf:datatype="&xsd;long"/></voltage>
235     <interface rdf:datatype="&xsd:string"> SPI/I2C</interface>
236     <dynamicRange rdf:datatype="&xsd;long">-20006 2000</dynamicRange>
237     <corpusType rdf:datatype="&xsd:string">LGA-16</corpusType>
238     <integrationFilter rdf:datatype="&xsd:string">VCH,NCH</integrationFilter>
239     <temperatureSensor rdf:datatype="&xsd:string">eight</temperatureSensor>
240     <port rdf:datatype="&xsd:string">input, output</port>
241     <used rdf:datatype="&xsd:string">mobile, gadget, game controllers, navigaion systems</used>
242     <energyConsumption rdf:datatype="&xsd:string">power-down, power-sleep
243   </energyConsumption>
244     <rdfs:comment rdf:datatype="&xsd:string">used in spheres of mobile,tablets, game controllers
245   </owl:NamedIndividual>
246
247
248
249   <!-- http://www.semanticweb.org/slava/ontologies/2013/2/ontomems#Resonator -->
250
251   <owl:NamedIndividual rdf:about="&ontomems;Resonator"/>

```

Рис. 9.10. OWL представлення екземпляру L3GD20

Звичайно, при великій кількості знань потрібно буде вдосконалити механізм конвертації, наприклад, створювати тип *Vocabulary* як батьківський, який буде містити тільки загальні властивості, а всі інші будуть нащадками, які описуватимуть властивості визначеного класу, екземпляру класу тощо.

Оскільки підсистема взаємодії з онтологією мікросистем повинна не тільки забезпечувати перетворення онтологічних знань в ОО об'єкти, потрібно реалізувати функціональність по автоматичній конвертації JAVA об'єктів в онтологічні індивіди. Для розв'язання даної задачі було використано бібліотеку Jena [47] та Protégé APi [48].

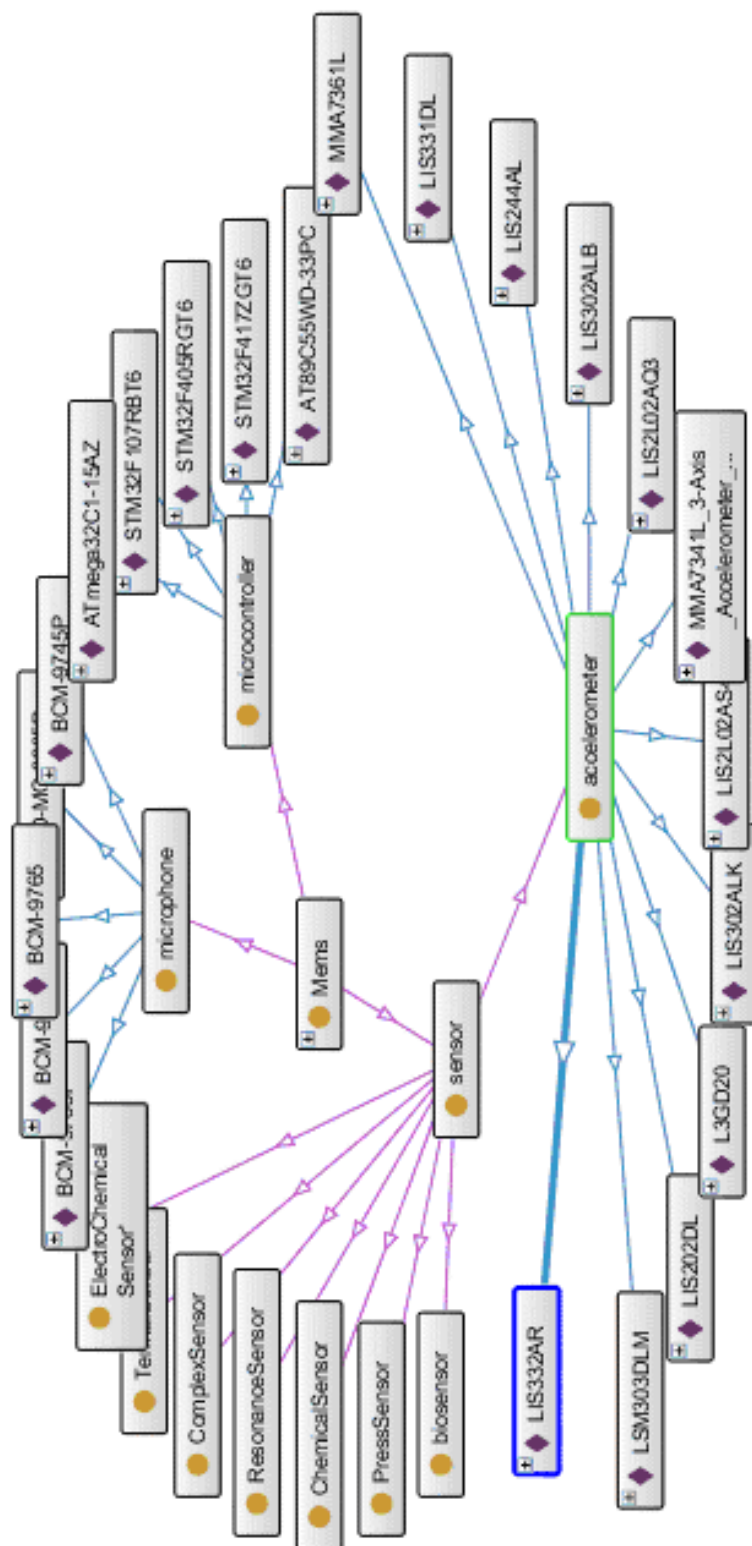


Рис. 9.11. Граф ієрархії класів, екземплярів та зв'язків онтології МЕМС

Як результат, на рис. 9.11 представлено звичайний граф доданих до існуючої онтології нових індивідів, а саме до класу *Microcontrolle*, *Microphone*, а на рис. 9.12 – детальний граф онтологічних елементів мікросистеми з посиланнями на них через URI (Unified resource identifier).

Таким чином, розроблений програмний модуль роботи з онтологією МЕМС виконує всі поставлені задачі:

- опрацювання вхідної інформації незалежно від типу її представлення;
- трансформацію об'єктів онтологічної парадигми в об'єктно-орієнтовну, і навпаки;
- конвертування даних у формат сумісний з форматом даних для збереження в базі готових проектних рішень мікросистем, MongoDB.

9.5. Контрольні запитання до розділу

1. Що ви розумієте під онтологією?
2. Які складові включає онтологія?
3. Які переваги в процесі проектування МЕМС надає онтологія?
4. Які елементи включає архітектура системи з використанням онтології для автоматизації проектування МЕМС?
5. Які моделі побудови онтології предметної області ви знаєте?
6. Які складові включає модель онтології предметної області?
7. Що ви розумієте під терміном чутливість?
8. Які переваги надають моделі чутливості в процесі проектування МЕМС?

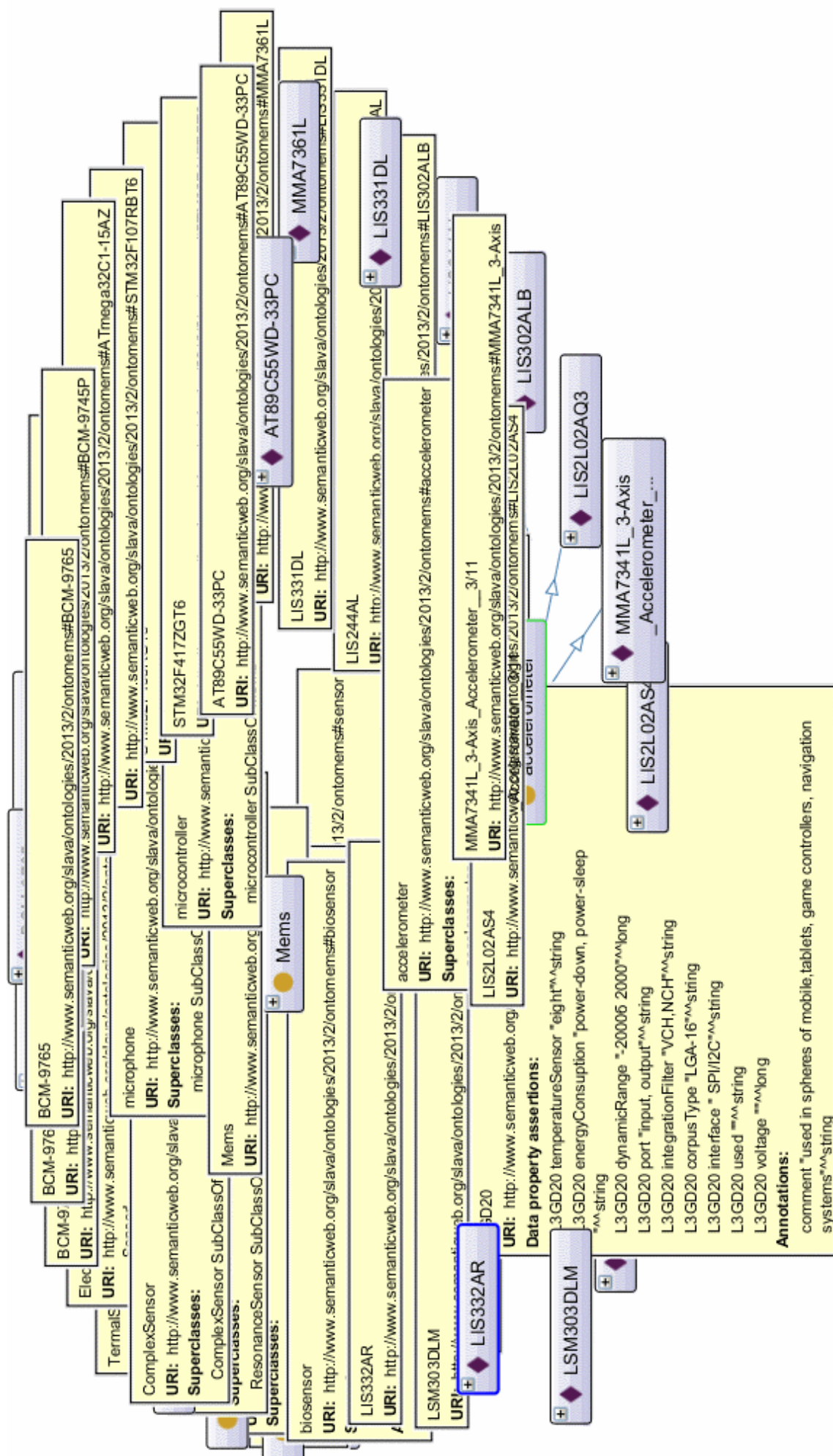


Рис. 9.12. Граф деталізованої організації онтології MEMC

9.6. Список використаної літератури до восьмого розділу

- [1] Влах И., Сингхал К. Машинные методы анализа и проектирования электронных схем. Пер. с англ. - М.: Радио и связь, 1988.-560с.
- [2] Хомович Р., Вакубратович М. Общая теория чувствительности. Пер. с сербс. и с англ., под. ред. Цыпкина Я.З. – М.: Советское радио, 1972. – 240 с.
- [3] Лучинин В.В. Микросистемная техника. Направления и тенденции развития // Научное приборостроение. 1999. Т. 9. № 1. С. 3-18.
- [4] Бочаров Л.Ю., Мальцев П.П. Состояние и перспективы развития микроэлектромеханических систем за рубежом. Микросистемная техника. 1999. № 1. С. 41-46.
- [5] Teslyuk V., Lobur M., Kernyskyi A., Denysyuk P. Mathematical Model for Optimization of Input Design Parameters Of Integrated Electrostatic Sensors // Proc. of XIII Polish-Ukrainian Conf. on CAD in Machinery Design – Implementation and Educational Problems. – Jurata, Poland, 2005, - P. 71-78.
- [6] Teslyuk V., Lobur M., Zaharyuk R., Al Omari Tarik Sensitivity theory application in MEMS design // Proc of the IX-th Intern. Conf. on The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM'2007). – Lviv – Polyana, Ukraine, 2007. – P. 536 – 537.
- [7] Andrews K. Information PyramidsTM: A new approach to visualizing large hierarchies. / Wolte J., Pichler, M. // In Proceedings of the IEEE Visualization '97, Phoenix, Arizona. –1997. – P.52-59.
- [8] Andrews K. Visual exploration of large hierarchies with information pyramids // In Proceedings of the Sixth International Conference on Information Visualization (IV'02). IEEE Computer Society Press, London, England. – 2002. – P.798-793.
- [9] Amann B.. Integrating ontologies and thesauri to build RDF schemas / Fundulaki I. // In Proceedings of the Third European Conference on Research and Advanced Technology for Digital Libraries. – 1999 – P.234–253.
- [10] Arpirez J.C. WebODE: a scalable workbench for ontological engineering / Corcho O., Fernandez-Lopez M., Gomez-Perez A.// First Intern. Conf. Knowledge Capture (KCAP2001). Victoria. Canada, Oct. – 2001.
- [11] Bechhofer S. OilEd: A Reason-able Ontology Editor for the Semantic Web / Horrocks I., Goble C., Stevens R. // Joint German/Austrian conf. on Artificial Intelligence (KI'01). Lecture Notes in Artificial Intelligence LNAI 2174, Springer-Verlag, Berlin. – 2001. –P. 396-408.
- [12] Bosca A. OntoSphere: more than a 3D ontology visualization tool / Bomino D., and Pellegrino P. // In Proceedings of SWAP, the 2nd Italian Semantic Web Workshop, Trento, Italy, December 14–16, CEUR. Workshop Proceedings, ISSN 1613-0073, – 2005. – <http://ceur-ws.org/Vol-166/70.pdf>.
- [13] Ceccaroni L. Ontowedss – An Ontology-based Environment Decision Support System for Management of Wastewater Treatment Plants // Ph.D. Thesis. Politechnic University of Catalonia. – 2001. – P. 9-45.
- [14] Ceravolo P. Introduction to Ontology Engineering / Damiani E. // Semantic Knowledge Management: An Ontology-Based Framework, 2009. – P. 47-73.
- [15] Domingue J. Tadzebao and WebOnto: Discussing, Browsing, and Editing Ontologies on the Web // Proc. of the Eleventh Workshop on Knowledge Acquisition, Modeling and Management, KAW'98, Banff, Canada. – 1998.
- [16] Chaudhri V. OKBC: A Programmatic Foundation for Knowledge Base Interoperability / A. Farquhar, R. Fikes P. Karp J. Rice // Fifteenth National Conf. on Artificial Intelligence. AAAIPres/The MIT Press, Madison. – 1998. – P. 600-607.
- [17] Eklund P.W. OntoRama: Browsing an RDF ontology using a hyperbolic like browser / Roberts N., Andgreen S.P. // In Proceedings of the First International Symposium on CyberWorlds (CW2002). Theory and Practices, IEEE press. – 2002. – P.405-411.

- [18] Eyl M. The Harmony Information Landscape: Interactive, Three Dimensional Navigation Through an Information Space//Master's thesis, Graz University of Technology, Austria. – 1995.
- [19] Farquhar A. The Ontolingua server: A tool for collaborative ontology construction / Fikes R., Rice J.// International Journal of Human-Computer Studies, 46 (6). – 1997. – P. 707-728.
- [20] Fernandez M. A Building a Chemical Ontology Using Methontology and the Ontology Design Environment / Gomez-Perez A., Pazos J.// IEEE Intelligent Systems, Jan./Feb. – 1999. – P. 37-46.
- [21] Fernandez M. METHONTOLOGY: From Ontological Art Towards Ontological Engineering / Gomez-Perez A., Juristo N.// AAAI-97 Spring Symposium on Ontological Engineering, Stanford University. –1997.
- [22] Gruber T.R. A translation approach to portable ontology specifications // Knowledge Acquisition. – 1993. – 5 (2). – P. 200-199.
- [23] Jeong C. Reconfigurable disc trees for visualizing large hierarchical information space / Pang A. // In Proceedings of Information Visualization. –1998. – P. 19-25.
- [24] Kifer M. Logical Foundations of Object-Oriented and Frame-Based Languages / Lausen G., Wu J.// Journal of the ACM. – 1995.
- [25] Kleiberg E. Botanical visualization of huge hierarchies / Van de Wetering H., Van Wijk J. // In Proceedings of the IEEE Symposium on Information Visualization (InfoVis'2001). IEEE Computer Society Press. – 2001.
- [26] Maedche A. Tutorial on Ontologies: Representation, Engineering, Learning and Application / Staab S. // ISWC'2002.
- [27] Теслюк В.М. Моделі інформаційних технологій синтезу мікроелектромеханічних систем: Монографія. – Львів : Видавництво ПП "Вежа і Ко", 2008. – 192 с.
- [28] Rector, A.L.: Modularisation of domain ontologies implemented in description logics and related formalisms including OWL. KC2003. ACM Press, Sanibel Island, FL, USA (2003).
- [29] Yuh-Jen Chen Knowledge integration and sharing for collaborative molding product design and process development, Computers in Industry, - 2010, - p.659-676.
- [30] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Tractable reasoning and efficient query answering in description logics: The DL-Lite family. J. of Automated Reasoning, 39(3), 2007. – pp.385–429.
- [31] B. Motik, I. Horrocks, and U. Sattler. Adding integrity constraints to OWL. In Proceedings of the Workshop OWLED, 2007.
- [32] Milko Marinov Using XML to Represent Knowledge by Frames// Proc. International Conference on Computer Systems and Technologies, 2004.
- [33] Biao Li , Xiaosong Tang, Huimin Xie, Xin Zhanga Strain analysis in MEMS/NEMS structures and devices by using focused ion beam system // Sensors and Actuators (A 111), 2004. – pp. 57–62.
- [34] [Електронний ресурс] : <http://jowl.ontologyonline.org/>
- [35] [Електронний ресурс] : <http://www.w3schools.com/jquery/>
- [36] [Електронний ресурс] : <http://backbonejs.org/>
- [37] [Електронний ресурс] : <http://www.oracle.com/us/technologies/java/standard-edition/advanced-suite/overview/index.html>
- [38] [Електронний ресурс] : <http://www.hibernate.org/>
- [39] [Електронний ресурс] : <https://code.google.com/p/guava-libraries/>
- [40] [Електронний ресурс] : <http://maven.apache.org/>
- [41] [Електронний ресурс] : <http://jolbox.com/>
- [42] [Електронний ресурс] : <http://docs.oracle.com/javaee/6/tutorial/doc/giepu.html>
- [43] [Електронний ресурс] : <http://www.liquibase.org/>
- [44] [Електронний ресурс] : <http://protege.stanford.edu/>
- [45] [Електронний ресурс] : <http://www.w3.org/RDF/>
- [46] [Електронний ресурс] : http://en.wikipedia.org/wiki/Uniform_resource_identifier
- [47] [Електронний ресурс] : <http://jena.apache.org/>
- [48] [Електронний ресурс] : <http://protege.stanford.edu/plugins/owl/api/>

НАВЧАЛЬНЕ ВИДАННЯ

*Теслюк Василь Миколайович,
Зелінський Андрій Ярославович,
Каркульовський Володимир Іванович,
Василюк Ярослав*

ADVANCED DESIGN FOR MICROSYSTEMS DEVICES

НАВЧАЛЬНИЙ ПОСІБНИК

Редактор
Комп'ютерне верстання