

## **Адаптивний метод проектування програмного забезпечення**

Кушнірук С.Л.

Науковий керівник: к.т.н. доц. Муляр І.В.

Хмельницький національний університет

В теперішніх умовах багато уваги приділяється проблемам, що пов'язані з проектуванням складних програмних систем. При цьому значний комплекс проектних завдань пов'язаний не лише з створенням нових програмних комплексів, але і з необхідністю структурної та параметричної перебудови діючих систем з метою їх реформування та модернізації. Динамічні зміни зовнішніх умов, потребують постійного технічного та алгоритмічного удосконалення [2]. Це призводить до необхідності розробки математичного та програмного забезпечення для комплексного рішення задач аналізу та синтезу складних програмних систем з урахуванням ієрархічності та можливості перебудови структури програмного забезпечення.

Процес структурного та параметричного синтезу програмних систем ускладнюється високою динамікою, нестабільністю та стохастичністю вимог до програмного забезпечення, що ускладнює використання аналітичних моделей для прийняття проектних рішень. Це призводить до необхідності застосування алгоритмічних оптимізаційних моделей, в яких відсутні явні аналітичні формулювання критеріїв оптимальності та обмежень, а наявна лише можливість визначення їх значень для кожного із варіантів з застосуванням різних моделюючих процедур. Складність оцінки властивостей таких моделей обмежує можливість використання стандартних оптимізаційних процедур, що в кінцевому результаті знижує ефективність процесу оптимального проектування [2].

Рішенням вказаної проблеми може бути досягнуто при використанні адаптивного підходу до проектування програмних систем. Даний підхід передбачає побудову комплексу алгоритмів оптимізації які забезпечують поєднання процесу більш повної формалізації задачі з її рішенням, та їх поєднання з процедурами багатоваріантного моделювання в інтелектуальну адаптивне середовище з можливістю його динамічного налаштування на різні класи вирішуваних задач оптимального проектування. При цьому, важливою вимогою до розроблюваного математичного забезпечення являється можливість ефективного рішення задач що описані складними алгоритмічними моделями.

Таким чином, актуальність магістерської роботи визначається необхідністю розробки теоретичних основ, математичного та програмного забезпечення для рішення задач пошуку оптимальних варіантів складних програмних систем при їх реформуванні та модернізації з можливістю врахування динамічних та стохастичних аспектів роботи.

На сучасному етапі розвитку ринкової економіки актуальними

проблемами залишаються проблеми створення нового програмного забезпечення та комп'ютерних систем а також, не менш важливими проблемами, залишається модернізація та реструктуризація вже створених та працюючих програм та систем .

Першим етапом розробки нової програмної системи або удосконалення вже існуючої, є комплексний всебічний аналіз всіх характеристик яким система повинна відповідати. Для того щоб сформувати загальне уявлення в цілому про програмну систему як про систему яка проектується, необхідно проаналізувати наступні характеристики [3]:

Група 1:

- 1) Алгоритмічна складність (логіка алгоритмів обробки інформації).
- 2) Склад та глибина роботи функцій обробки які будуть реалізовані.
- 3) Повнота та системність функцій обробки.
- 4) Об'єм файлів програми.
- 5) Вимоги до операційної системи та до технічних засобів обробки

програмними засобами.

- об'єм дискового простору.
- необхідна кількість оперативної пам'яті для запуску.
- тип процесора.
- версія операційної системи.

Наявність обчислювальної мережі.

Наступна ступінь диференціації характеристик виникає при переході до аналізу окремих компонентів програми. При цьому відбувається деталізація характеристик першої групи. Так, структура програми деталізується у вигляді схем основних технологічних взаємозв'язків в середині кожного з модулів [4].

До другої групи характеристик відносяться характеристики форм організації основних функціональних процесів програми:

Група 2:

- 1) Технології, мови програмування за допомогою яких буде проводитись розробка.
- 2) Організація зв'язків між компонентами програми.
- 3) Створення інтерфейсів для комунікації між компонентами створених з використанням різних технологій або мов програмування.
- 4) Специфікація даних з якими буде працювати той чи інший компонент системи.
- 5) Рівень завантаженості каналів зв'язку.
- 6) Характер передачі даних між компонентами програми.
- 7) Спосіб зберігання даних.
- 8) Швидкість обробки даних.

На основі аналізу цих характеристик визначаються напрямки створення програмної системи та визначаються можливі напрямки розвитку

при її подальшій модернізації.

На рисунку 1 показана схема процесу проектування програмної системи яка відповідає трьом рівням оптимізації.

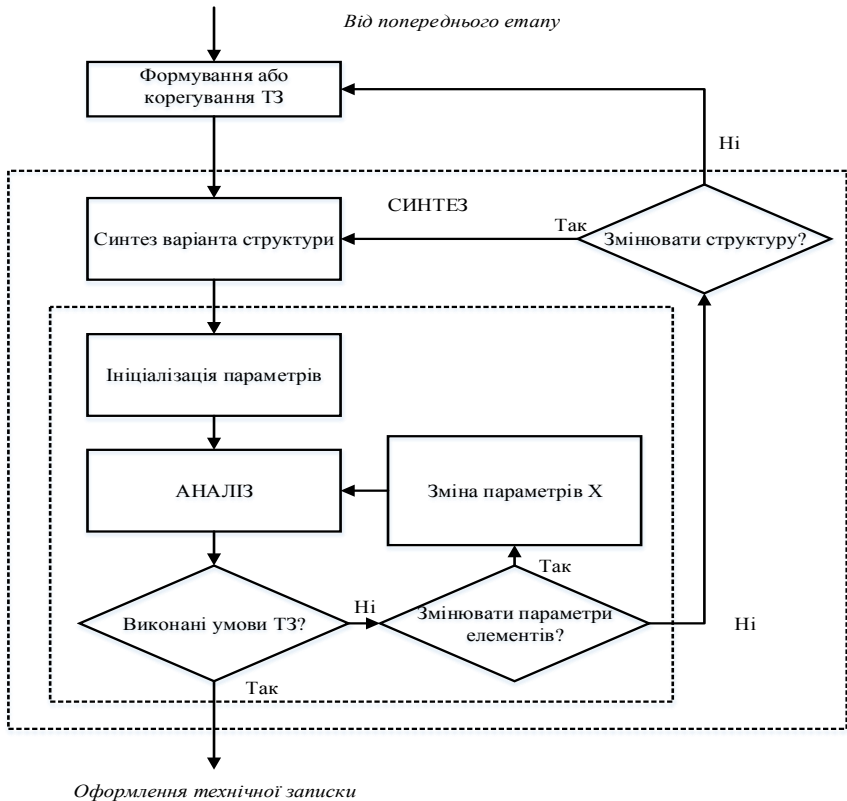


Рисунок 1 - Схема типового маршруту проектування

Перший рівень полягає у виборі найкращої ідеї або принципу дії об'єкта проектування; другий рівень – у пошуці найкращого структурного рішення або принципу дії; третій рівень – у визначенні найкращих параметрів для вибраної структури. Задачі першого рівня характерні для етапу зовнішнього проектування і вирішуються з використання експертних підходів і методів. Математичне забезпечення сучасних САПР орієнтовано на рішення задач другого та третього рівня.

Отримання оптимального рішення пов'язано з вибором найкращого варіанта з деякої допустимої множини рішень. Склад оптимізаційного процесу являє собою набір варіантів, які відповідають вимогам до

функціонування програмної системи, а також набору експлуатаційних і технічних обмежень.

#### Література

1. Информационные системы и технологии: Научное издание. / Под ред. Ю.Ф. Тельнова. - М.: ЮНИТИ, 2016. - 303 с.
2. Черников Б.В., Поклонов Б.Е. Оценка качества программного обеспечения практикум. - М.: ИД. "Форум" : ИНФРА - М. 2012-400с.
3. Хеффельфингер, Д. Разработка приложений Java EE 6 в NetBeans 7 / Д. Хеффельфингер. - М.: ДМК Пресс, 2013. - 330 с.
4. Максимов, Н.В. Современные информационные технологии / Н.В. Максимов, Т.Л. Партыка, И.И. Попов. - М.: Форум, 2013. - 512 с.