

ПОРІВНЯЛЬНИЙ АНАЛІЗ МЕТОДІВ ОПТИМІЗАЦІЇ ДЛЯ ЗАДАЧІ ПЛАНУВАННЯ ПОСТАВОК

Вінницький національний технічний університет

Анотація

Здійснюється аналіз методів оптимізації з точки зору ефективності їх застосування для оптимізації управління складськими запасами. Побудовано класифікацію методів, розглянуті точні та евристичні методи. Обґрунтовано вибір генетичного алгоритму.

Ключові слова: оптимізація, алгоритми пошуку, евристики, метаевристики, генетичний алгоритм

Abstract

The optimization methods analysis from the point of view of their application efficiency for the warehouse stocks management optimization is carried out. The classification of methods is constructed, exact and heuristic methods are considered. The choice of genetic algorithm is substantiated.

Keywords: optimization, search algorithms, heuristics, metaheuristics, genetic algorithm.

Вступ

Запаси посідають особливе місце у складі майна та домінуючі позиції у структурі витрат підприємств різних сфер при визначенні результатів господарської діяльності підприємства та при висвітленні інформації про його фінансовий стан. В цій ситуації удосконалення побудови своєчасного та достовірного контролю, та планування рівня запасів відіграє надзвичайно важливу роль [1]. На ринку є не мало програмних продуктів, що сприяють підвищенню ефективності планування та контролю рівня запасів. Але постійний розвиток методів оптимізації та бажання розширення функціоналу оптимізаційних програм залишає задачу планування поставок актуальною.

Загальна характеристика задачі оптимізації

Задача оптимізації є однією з центральних проблем науки, техніки та повсякденного життя. Прийняття рішення в більшості випадків полягає в генерації всіх можливих альтернатив рішень, їх оцінці та вибору кращої з них. Прийняти правильне рішення – значить обрати такий варіант з числа можливих, в якому при врахуванні усіх різноманітних факторів і суперечливих вимог буде оптимізована деяка загальна цінність, тобто рішення буде в максимальному ступені сприяти досягненню поставленої мети [2,3].

Формалізація тієї чи іншої задачі оптимізації, у загальному випадку, припускає опис усіх важливих факторів, які впливають на досягнення мети, їх взаємодії, обмежувальних умов та критерію оцінки якості рішення, що приймається, на базі якого можна здійснювати вибір між альтернативами. У якості критерію оцінки виступає цільова функція, аргументами якої є кількісні характеристики, які описують становище факторів, що впливають на досягнення мети в розв'язуваній задачі. При цьому, рішення, яке приводить до найкращого результату, як правило, відповідає екстремальним значенням цільової функції, тобто точка її максимуму або мінімуму [4].

Отже, процес генерації варіантів рішення та вибору найкращої з отриманих альтернатив зводиться до створення всіх можливих комбінацій значень характеристик, які впливають на цільову функцію, і знайдення такої комбінації, яка приводить до її екстремального значення. Всі можливі комбінації аргументів при цьому утворюють простір пошуку задачі, розмірність якого визначається кількістю аргументів цільової функції. А кожна з зазначених комбінацій утворює точку в даному просторі [5].

Слід відзначити, що цільова концепція при постановці задачі вибору може знаходити відображення як при завданні множини допустимих альтернатив, так і при формулюванні вимог до отримання ефективних, зокрема строго оптимальних рішень.

Таким чином, метою задачі оптимізації як практичного, так і теоретичного характеру є вибір "найкращої" (припустимої або оптимальної) конфігурації з множини альтернатив для досягнення деякої поставленої мети. При цьому "найкращою" – в смислі забезпечення оптимуму (максимального або мінімального значення залежно від конкретної постановки задачі) заданої цільової функції при задоволенні певних обмежень.

Класифікація методів та алгоритмів оптимізації

Наявність значних труднощів та специфічних особливостей у вирішенні задач оптимізації породило велику кількість методів і алгоритмів. Як показав аналіз, при їх розробці залучався дуже різноманітний математичний апарат від математичної логіки до теорії динамічних систем, що ускладнює проведення відповідної класифікації. Однак, усі розроблені на сьогоднішній день методи, залежно від необхідної якості отримання результату, можна віднести до одного з двох класів:

- методи, які завжди приводять до знаходження оптимального рішення, але потребують для цього в найгіршому випадку неприпустимо великої кількості операцій;
- методи, які не завжди приводять до знаходження оптимального рішення, але потребують для отримання цього рішення припустимої кількості операцій.

Докладну класифікацію наведено в таблиці 1.

Точні методи

Точні методи вирішення в теорії дискретного програмування є найбільш загальними, що призводить до появи все нових різновидів методів і алгоритмів. До даної групи відносяться алгоритми, в яких робиться спроба як повного перебору при невеликій розмірності задачі, так і максимального скорочення обсягу перебору в протилежному випадку. При цьому має місце неминучість експоненціального часу роботи алгоритмів [6].

Найбільш поширеними прийомами скорочення перебору є прийоми ціле-спрямованого перебору, які засновані на методі "гілок та границь" або методи "неявного перебору". Ці прийоми складаються з побудови "часткових рішень", які представлені у вигляді дерева пошуку, та з використання потужних методів побудови оцінок. Ці оцінки використовуються для впізнання безперспективних часткових рішень, в результаті чого від дерева пошуку відразу відсікаються цілі гілки.

Проаналізовано інші підходи, коли процес пошуку організований інакше (вони інколи використовуються спільно з методом гілок та границь). До них відноситься метод динамічного програмування, метод відсічень (площин що відтинають) та метод Лагранжа.

Використання даних алгоритмів дозволяє в тій чи іншій мірі прискорити пошук точного рішення при збереженні в загальному випадку експоненціального закону залежності часу виконання алгоритму (трудомісткості) від вихідної кількості.

Алгоритми даного підходу при вирішенні практичних задач оптимізації мають лише теоретичний інтерес і, як показав аналіз, ефективність своєї реалізації для невеликої кількості вихідних даних [7].

Евристичні методи

До евристики відносяться алгоритми послідовного покращання рішень, які є дуже розвиненими в теорії математичного програмування. Вони використовують прийом, який можна назвати "зниженням вимог". Він полягає у відмовленні від оптимального рішення, але в знаходженні "гарного" рішення за припустимий час.

Виявлено, що основою даного напрямку є така організація пошуку на множині альтернатив, при якій поступово виділялися б все більш кращі припустимі рішення. Алгоритми, засновані на цьому прийомі, звичайно називають "евристичними", оскільки вони використовують розумні міркування без строгих обґрунтувань. При їх розробці використовуються інтуїтивні міркування, які не підкріплюються відповідним математичним обґрунтуванням.

Визначено, що при використанні подібних алгоритмів природно виникає питання про можливість досягнення при їх використанні глобального екстремуму або наближення до нього із заданою точністю. Виявляється, що в більшості випадків на практиці може бути гарантоване лише досягнення локального екстремуму. Дані алгоритми є приблизними, але вони принципово відрізняються від

приблизних алгоритмів оптимізації, які будуються на базі точних методів послідовного звуження множини рішень та гарантують наближення до екстремуму з вказаною точністю [8-10].

Таблиця 1 - Методи і алгоритми оптимізації

Найменування алгоритму (методу)		Переваги		Недоліки		
Точні методи	Методи і алгоритми послідовного звуження множини рішень (повний та ціле-спрямований перебір)	Повний перебір варіантів		Оптимальне рішення	NP-повнота. Неможливість реалізації при великій кількості вихідних даних	
		Метод гілок і границь				
		Метод динамічного програмування				
		Метод множників Лагранжа				
		Алгоритми попереднього розширення і послідовного звуження	Алгоритми відсічень Алгоритми кінцевого розширення і послідовного звуження			
Композитні алгоритми						
Приблизні методи	Методи і алгоритми послідовного покращання рішень (евристичні алгоритми)	Ітеративні (локальні) алгоритми	Алгоритм найближчого сусіда	Висока швидкість отримання результату	Зріст погрішності зі збільшенням вихідних даних	
			Алгоритм середньої величини			
		Стохастичні алгоритми	Випадковий пошук	Перегляд області рішень із заданою імовірністю		Імовірнісний пошук
			Спрямований випадковий пошук			
		Імітація відпалу	Рішення, близьке до оптимального	Низька швидкість отримання результату		
		Методи паралельної обробки даних	Нейронні мережі	Висока швидкість і рішення, близьке до оптимального	Складність аналізу, програмної реалізації	
	Локально-стохастичні методи (методи, які використовують елемент випадковості - еволюційні методи)	Еволюційні алгоритми	Генетичні алгоритми	Широкий простір рішень, паралельність	Проблеми збіжності, обмеження вибору параметрів	
Ройові алгоритми		Алгоритм колонії мурах	Паралельність, надійність, легкість комбінування з іншими методами	Низька збіжність, залежність від експериментального налагодження		

Евристичні алгоритми, побудовані вказаними вище способами, на практиці виявляються дуже задовільними. Однак, для отримання задовільних характеристик за часом і точністю отримання результату звичайно потрібна велика робота по його доробці та синтезі нових підходів знаходження глобального екстремуму цільової функції. В результаті вдається тільки в дуже рідких випадках передбачати й оцінювати поведінку таких алгоритмів. Замість цього такі алгоритми оцінюються та

порівнюються на базі сполучення емпіричних даних і аргументів, що спираються на здоровий глузд. Однак, евристичні алгоритми не завжди є настільки складними для формального аналізу. В деяких випадках вдається довести, що рішення, отримані евристичними алгоритмами, завжди будуть відрізнятися в процентному відношенні від оптимального не більш ніж на певну величину [11-14].

Генетичний алгоритм

Іншим способом вирішення оптимізаційних задач є генетичний алгоритм (ГА), що використовується для вирішення задачі шляхом послідовного підбору, комбінування і варіації шуканих параметрів з використанням механізмів, що нагадують біологічну еволюцію.

Особливістю генетичного алгоритму є акцент на використання оператора "схрещування", який виконує рекомбінацію рішень-кандидатів, аналогічну ролі схрещування в живій природі [15].

ГА відрізняються від інших оптимізаційних і пошукових процедур тим, що: працюють переважно не з параметрами задачі, а із закодованою множиною параметрів; здійснюють пошук покращенням не одного рішення, а одразу декількох альтернатив на заданій множині рішень; для генерації нових рішень фактично не потребують знань залежності результату від ознак властивостей рішення.

ГА, являючись однією з парадигм еволюційних досліджень, є алгоритмами пошуку, побудованими на принципах природного добору і генетики. Вони об'єднують у собі принцип виживання найбільш перспективних особин-рішень і структурований випадково-детермінований обмін інформацією, в якому присутній елемент випадковості, що моделює природні процеси спадкування і мутації.

Популярність генетичних алгоритмів обумовлена тим, що вони дозволяють знайти більш гарні або "раціональні" рішення NP-повних практичних задач оптимізації за менший час, ніж інші методи. Звичайно ж, термін "гарні" або "раціональні" не є строгим в математичному значенні.

Під "раціональними" маються на увазі рішення, які задовольняють дослідника. Адже в більшості реальних задач немає необхідності знаходити саме глобальний оптимум. Найчастіше метою пошуку є рішення, які задовольняють певним обмеженням. Наприклад, час іспитів обладнання не повинен перевищувати певної заданої величини. В такому випадку достатньо знайти саме "раціональне", тобто розумне рішення. Друга немаловажна причина популярності ГА забезпечується стрімким зростанні продуктивності сучасних комп'ютерів [16,17].

Переваги генетичних алгоритмів стають більш очевидними, якщо розглянути п'ять їх основних відмінностей від традиційних методів:

- робота з кодами, в яких представлений набір параметрів, безпосередньо залежних від аргументів цільової функції;
- одночасне використання для пошуку декількох точок пошукового простору (розпаралелювання), а не перехід від точки до точки. Тобто ГА оперує одночасно всією сукупністю припустимих рішень;
- відсутність потреби в процесі роботи будь-якої додаткової інформації, що підвищує швидкість їх роботи;
- використання як імовірнісних правил для породження нових точок пошуку, так і детермінованих правил для переходу від одних точок до інших;
- здійснення пошуку оптимального рішення за однією й тією ж стратегією, як для унімодальних, так і для багатоекстремальних функцій.

Висновок

Аналіз переваг і недоліків методів вирішення оптимізаційних задач було визначено, що генетичний алгоритм має суттєві переваги для оптимізації планування поставок, які полягають в простоті використання цільової функції (зменшення розміру втрат), паралельному пошуку рішення у багатьох напрямках, легкому масштабуванні алгоритму відносно кількості вхідних параметрів, низькій імовірності застрягання в локальному рішенні, отриманні рішень близьких до оптимального.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Іванова О. А. Економіка сучасного підприємства : Навч. посіб. / О. А. Іванова, Є. Ф. Пеліхов, О.М. Сумець ; Нар. укр. акад.; [каф. економіки підприємства. – 2-е вид., випр.]. – Харків: Вид-во НУА, 2017. – 312

2. Тасьмук Д.І., Месюра В.І. Оптимізація міського трафіку за допомогою генетичного алгоритму// «Інтернет-Освіта-Наука-2018», Одинадцята міжнародна науково-практична конференція ІОН-2018, 22-25 травня, 2018: Збірник праць. –Вінниця: ВНТУ, 2018 –ст. 24-25 с. –ISBN978-966-641-728-5
3. Тасьмук Д., Месюра В. Визначення параметрів алгоритму оптимізації керування рухом на перехресті / Контроль і управління в складних системах (КУСС-2018). XIV Міжнародна конференція. Тези доповідей. Вінниця, 15-17 жовтня 2018 року. – Вінниця: ВНТУ. – 2018. – с.138.
4. Бродецкий Г. Л. Управление запасами : Многофакторная оптимизация процесса поставок : Учебник для академического бакалаврата / Г. Л. Бродецкий, В. Д. Герами, А. В. Колик, И. Г. Шидловский. – М. : Издательство Юрайт, 2019. – 322 с.
5. Люгер Д. Ф. Искусственный интеллект. Стратегии и методы решения сложных проблем/ Д. Ф. Люгер - Вильямс, 2005. - 864 с.
6. Рафгарден Тим. Совершенный алгоритм. Алгоритмы для NP-трудных задач. СПб.: Питер, 2021. – 304 с.
7. Кормен Т. Г. Вступ до алгоритмів : Переклад з англійської третього видання : [укр.] = Introduction to Algorithms : Third Edition : [пер. з англ.] / Томас Г. Кормен, Чарлз Е. Лейзерсон, Роналд Л. Рівест, Кліфорд Стайн. К.: К. І. С., 2019. – 1288 с.
8. Бузовський П. О., В. І. Месюра. Аналіз метаевристичних підходів до розв'язання задачі складання навчальних розкладів// XLIX Науково-технічна конференція підрозділів Вінницького національного технічного університету (2021) / Електронне наукове видання матеріалів конференції. – Вінниця, 2020. – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2020/paper/view/9816/8804>
9. Ліщинський В. О., Месюра В. І. Обґрунтування вибору метаевристики для визначення оптимального маршруту // XLIX Науково-технічна конференція підрозділів Вінницького національного технічного університету (2021) / Електронне наукове видання матеріалів конференції. – Вінниця, 2020. – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2020/paper/view/9892/8241>
10. Гранік М.О. Використання методу імітації відпау для розв'язання задачі про розфарбування графу / М.О.Гранік, В.І.Месюра // Інформаційні процеси і технології «Інформатика - 2013»: матеріали VI Міжнар. наук.-практ. конф. молодих учених і студентів, Севастополь, 22 – 26 квіт. 2013 р. / М-во освіти і науки України, Севастоп, нац. техн. ун-т; наук. ред. С.В.Доценко – Севастополь: СевНТУ, 2013. – С. 77-78. – ISBN 978-966-335-393-7.
11. Бендерук Ю. А., Динамічна зміна коефіцієнтів соціалізації та персоналізації методу рою часток під час розв'язання задачі про розподіл економічного навантаження / Ю.А. Бендерук, М. О. Гранік, В. І. Месюра. - Вісник Вінницького політехнічного інституту -. – 2013. – №3. – С. 96-98.
12. Бендерук Ю.А. Підбір константних параметрів методу рою часток за методом імітації відпау під час розв'язання задачі розподілу виробничого навантаження / Ю. А. Бендерук В. І. Месюра //Наукові праці Вінницького національного технічного університету. – 2013. – №3. – [Електронне наукове фахове видання] – Режим доступу до журн.:http://irbis-nbuv.gov.ua/cgi-bin/irbis_nbuv/cgiirbis_64.exe?C21COM=2&I21DBN=UJRN&P21DBN=UJRN&Z21ID=&IMAGE_FILE_DOWNLOAD=1&Image_file_name=PDF/VNTUV_2013_2_8.pdf
13. Сімоненко, Д. В., Месюра, В. І., «Мультиагентна система маршрутизації на основі мурашкового алгоритму» в Матеріали конференції «XLVI Науково-технічна конференція підрозділів Вінницького національного технічного університету (2017)», Вінниця, 2017. [Електронний ресурс]. Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/index/pages/view/zbirn2017> Дата звернення: Черв. 2017
14. Корчиста О.В., Месюра В.І. Мурашковий алгоритм на базі нечіткої логіки // Сучасні інформаційні технології, засоби автоматизації та електропривод : матеріали II Всеукраїнської науково-технічної конференції, 19–21 квітня 2018 р. / За заг. ред. О. Ф. Тарасова. – Краматорськ : ДДМА, 2018. – С. 104-105.
15. Сидоренко С. О., Месюра, В. І. Інтелектуальний модуль для налаштування параметрів генетичного алгоритму // Матеріали конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2019)». – Вінниця, 2019. [Електронний ресурс]. Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-mn/index/pages/view/zbirn2019> , Дата звернення: Берез. 2020.

16. Тасьмук Д.І., Месюра В.І. Генетичний алгоритм для керування рухом на перехресті // Сучасні інформаційні технології, засоби автоматизації та електропривод : матеріали II Всеукраїнської науково-технічної конференції, 19–21 квітня 2018 р. / За заг. ред. О. Ф. Тарасова. – Краматорськ : ДДМА, 2018. – С. 121-122. ISBN 978-966-379-869-1.

17. Донець В.В., Месюра В.І. Аналіз задачі маршрутизації транспортних засобів для перевезення товарів // I Науково-технічна конференція підрозділів Вінницького національного технічного університету (2021) / Електронне наукове видання матеріалів конференції. – Вінниця, 2021. – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2021/paper/view/12847/10772>

Мялковський Дмитро Васильович — студент групи 2КН-186, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, email: mylkovsky@gmail.com

Месюра Володимир Іванович — канд. техн. наук, професор кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця

Арсенюк Ігор Ростиславович – канд. техн. наук., доцент, доцент кафедри комп'ютерних наук, Вінницький національний технічний університет, м. Вінниця.

Dmytro V. Mialkovskiy – student of the Intelligent Information Technologies and Automation Department, Vinnytsia National Technical University, Vinnytsia, e-mail: mylkovsky@gmail.com

Volodymyr I. Mesyura– Cand. Sc. (Eng.), Professor of the Computer Science Chair, Vinnytsia National Technical University, Vinnytsia.

Igor R. Arsenyuk – Cand. Sc. (Eng.), Assistant Professor of the Chair of Computer Science, Vinnytsia National Technical University, Vinnytsia.