

Н. К. Тимофієва (Київ)

ПРО СПОСОБИ ОЦІНКИ РЕЗУЛЬТАТІВ РОЗВ'ЯЗКУ ЗАДАЧ КОМБІНАТОРНОЇ ОПТИМІЗАЦІЇ

При розв'язанні задач комбінаторної оптимізації оцінку результату проводять з використанням кореляційних методів або аналізують структуру вхідної інформації. Різні підходи впливають на точність результату і на швидкодню алгоритму. Методами, які ґрунтуються на розпізнаванні структури вхідної інформації в порівнянні з кореляційними, глобальний розв'язок для великих розмірностей задачі досить часто знаходиться поліноміально. Це пов'язано з тим, що комбінаторна конфігурація (аргумент цільової функції) будується в процесі розпізнавання елементів множини вхідних даних, завдяки чому між ними встановлюється функціональна зв'язаність. До того ж для деяких задач із штучного інтелекту, в яких виникає ситуація невизначеності, такі підходи можуть бути єдино можливими для їхнього розв'язання.

Постановка задачі. Для оцінки результату розв'язку задач комбінаторної оптимізації на основі вхідної інформації моделюється цільова функція. Вхідні дані в задачах цього класу – випадкові величини, які мають безладну структуру. Тому для отримання адекватної оцінки важливо вибрати такі методи аналізу даних, за допомогою яких можна отримати коректний результат в реальному часі.

Основна частина. Як правило, для різних підходів, що використовуються для розв'язання задач комбінаторної оптимізації, цільова функція моделюється на формальному рівні, а вхідні дані вважають її аргументом. Але в задачах цього класу аргументом цільової функції є комбінаторні конфігурації різних типів. Вхідні дані виступають як критерії, за якими оцінюється результат.

Для розв'язання задач із класів задач комбінаторної оптимізації виділимо такі основні підходи [1]: а) ітераційні методи та алгоритми, що ґрунтуються на частковому переборі варіантів; б) методи та алгоритми, що ґрунтуються на розпізнаванні структури вхідної інформації. Їх ще називають евристичними, такими, в яких моделюються правила вибору оптимального рішення людини в ручному режимі. В ітераційних методах спочатку визначається аргумент цільової функції (випадково або за заданими правилами), а потім для нього за змодельованим виразом обчислюється значення цільової функції. В методах, які ґрунтуються на розпізнаванні структури вхідних даних, шляхом визначення зв'язків між елементами базової множини, якою задається певна задача, послідовно будується комбінаторна конфігурація (аргумент), для якої цільова функція набуває оптимального значення.

В ітераційних методах часто використовують кореляційний аналіз. Але цей аналіз даних відображає лише лінійну залежність між випадковими величинами та не відображає їхньої функціональної зв'язаності. В перебірних методах комбінаторна конфігурація (аргумент цільової функції) знаходиться не з урахуванням структури вхідної інформації, а визначається на певній ітерації випадково або за певними правилами. Розроблені на основі цих методів алгоритми поліноміально визначають лише локальний розв'язок. Глобальний оптимум цими підходами знаходиться для невеликої розмірності задачі, а для великої знаходиться експоненціально. Але його інколи знайти неможливо навіть повним перебором із-за ситуації невизначеності.

При використанні евристичних методів для визначення функціональної зв'язаності між елементами множин вхідних даних використовують підходи, що ґрунтуються на розпізнаванні їхньої структури (кластерний та факторний аналізи, метод опорних векторів тощо). В процесі розв'язання цих задач послідовно будується комбінаторна конфігурація (аргумент цільової функції), який може бути і глобальним розв'язком. Такий підхід реалізовано і в методі структурно-алфавітного пошуку.

Висновок. Отже, використання різних підходів до аналізу вхідних даних для оцінки результату впливає на швидкодню та точність алгоритмів, що розробляються для розв'язання задач комбінаторної оптимізації.

Список літературних джерел

1. Тимофієва Н.К. Теоретико-числові методи розв'язання задач комбінаторної оптимізації. Автореф. дис... докт. техн. наук / – Ін-т кібернетики ім. В.М. Глушкова НАН України, Київ. – 2007. – 32 с.