УДК 658.52.011.56

РІШЕННЯ ЗАДАЧІ ПРО РЮКЗАК ЕВРИСТИЧНИМ МЕТОДОМ

М. В. Костікова
Харківський національний автомобільно-дорожній університет

Задача про рюкзак – класичне задача дискретної оптимізації [1], [2]. Дана задача і її варіанти широко використовуються для моделювання великої кількості практичних задач. У загальному виді задачу можна сформулювати так: із заданої множини предметів із властивостями «вартість» і «вага», потрібно відібрати якесь число предметів таким чином, щоб одержати максимальну сумарну вартість при одночасному дотриманні обмеження на сумарну вагу.

Свою назву задача одержала від максимізаційної задачі укладання як можна більшого числа коштовних речей у рюкзак за умови, що загальний обсяг (або вага) усіх предметів, здатних поміститися в рюкзак, обмежений. Безпосередньо термін «рюкзак» може бути інтерпретований досить широко. Задача про рюкзак і її модифікації часто виникають в економіці, прикладній математиці, криптографії, генетиці й логістиці для знаходження оптимального завантаження транспорту (автомобіля, контейнера, літака, поїзда, трюму корабля) або складу.

До класичного формулювання задачі можуть зводитися багато інших задач різних розмірностей. У якості вартості й ваги можуть використовуватися зовсім різні характеристики й навіть їх комбінації. У цім питанні необхідна лише наявність деякого перетворення (функції), яка дозволить звести необхідну задачу до класичної.

При виборі алгоритму рішення доводиться вибирати між точними алгоритмами, які не застосовні для рюкзаків великої розмірності, і наближеними, які працюють швидко, але не забезпечують оптимального рішення задачі. Вибір використання того або іншого методу є спірним питанням. Усе залежить від постановки задачі, а також від того, які цілі поставлені.

Порівняння різних методів рішення задачі про рюкзак широко представлено в літературі й Інтернеті. Розробці методів рішення задачі, і в першу чергу ефективних, приділене досить багато уваги [1 – 10].

Ціль роботи полягає в розробці евристичного методу призначеного для рішення задачі про рюкзак, який дає наближені рішення задачі.

Розглянута нами задача є NP-повною, і проблема в тому, що не існує поліноміального алгоритму, що вирішує її за розумний час.

У загальному виді задача про рюкзак формулюється в такий спосіб. Перед походом у рюкзак, місткістю не більш A одиниць ваги, з набору $I = \{1, 2, ..., n\}$ предметів, кожний вагою a_i і «цінністю» c_i, необхідно покласти ті предмети x_i, $i \in I$, які максимізують сумарну «цінність» вантажу й містяться по вазі в рюкзак.

У математичній формі задача представляється в такий спосіб: потрібно знайти x_i, $i \in I$, що доставляє

$$\max \sum_{i \in I} c_i x_i = W \tag{1}$$

при умовах

$$\sum_{i \in I} a_i x_i \le V , \tag{2}$$

$$x_i = \begin{cases} 1, & \text{якщо предмет поміщений у рюкзак,} \\ 0, & \text{якщо предмет не поміщений у рюкзак.} \end{cases} \tag{3}$$

Таким чином, задача про рюкзак – це задача цілочисельного лінійного програмування з булевими змінними. Її рішення досягається на деякій підмножині 2^n комбінацій, формованій різними наборами змінних x_i, що задовольняють обмеженню (2).

Задача є NP-повною, тобто для неї теоретично не існує поліноміального за часом рішення алгоритму [1]. Серед експоненціальних алгоритмів найбільш підходящим є ті, які реалізують схему гілок і границь, метод динамічного програмування, адитивний алгоритм Балаша [2] і інші розроблені алгоритми. Хоча зазначені алгоритми й оцінені по обчислювальній складності, вони разом з тим не мають практичних оцінок рішення задач, оцінок роботи алгоритмів у середньому, що дуже важливо для застосування. Тому іноді необхідно вибирати або швидкий алгоритм, але він не завжди вирішує задачу найкращим чином, або вибирати точний, який не є працездатним для великих значень.

У методиці гілок і границь є два моменти алгоритмізації, визначених специфікою задачі: розбивка вихідної множини комбінацій на підмножини з подальшим вибором підмножини для чергової розбивки й обчислення нижніх (верхніх) границь (оцінок) значень функції, що оптимізується, на підмножинах. Розбивка множини на підмножини називається розгалуженням, а вибір підмножини для розбивки – її стратегією. Обчислення помилок тлумачать як рішення оцінних задач.

Наочним результатом розгалуження й рішення оцінних завдань є n-ярусне дерево пошуку рішень з оцінками вершин підмножин кожного ярусу. Оцінка вершини останнього ярусу – рекорд – являє собою поточне значення функції, що оптимізується, яке далі порівняється з оцінками вершин попередніх ярусів, у результаті чого безперспективні для розгалуження підмножини відсіваються, а перспективні розбиваються, доповнюючи дерево рішень. Алгоритм закінчує роботу тоді, коли будуть порівняні з рекордом усі поточні й знову породжувані оцінки вершин.

Практика показує, що більш ефективним є ті алгоритми, які будують бінарні дерева рішень, тобто реалізують розбивку кожної чергової множини на дві підмножини, вибір підмножини для розгалуження здійснюють по максимальній (мінімальної) оцінці функції, що оптимізується, оцінні задачі формуються так, що більш точно обчислюються оцінки.

Тому при розробці алгоритму й наявності вибору слід дотримуватися зазначених правил.

Що стосується задачі, що розглядається, то процес розбивки чергової множини здійснюється на дві підмножини, перша з яких містить комбінації компонентів вектора з $x_i = 0$, друга – з $x_i = 1$ Стратегія розгалуження полягає в тому, щоб вибирати чергову підмножину для розбивки по максимальній оцінці верхньої границі функції, що оптимізується.

Для обчислення оцінок верхніх границь функції, що оптимізується, на підмножинах формується наступна оцінна задача:

$$Q = \max \sum_{i \in I} c_i x_i \tag{4}$$

при умовах

$$\sum_{i \in I} a_i x_i \le V , \tag{5}$$

$$0 \le x_i \le 1, \; i \in I . \tag{6}$$

Іншими словами, для якогось x_i допускається, що воно не булево, а лежить в інтервалі $[0,1]$. Така релаксація строгості умов задачі приводить до того, що оцінка Q для кожної вершини дерева пошуку виявляється більше W, тобто вона дійсно є верхньою границею функції, що оптимізується, на підмножинах комбінацій.

У теж час оцінна задача легко розв’язувана. Згідно [3] слід знайти «цінності» одиниць ваг вантажів c_i / a_i, $i = 1, 2, ..., n$ і впорядкувати їх по убуванню $c_1/a_1 \ge c_2/a_2 \ge ... \ge c_n/a_n$. Тоді всі $x_i \in I$, обирані в порядку послідовності «цінностей» і такі, що $\sum_{i=1}^{k} a_i < V$, покладаються рівними одиниці. Чергове значення x_{k+1} таке, що $\sum_{i=1}^{k+1} a_i \cdot x_i > V$ обчислюється по виразу $x_{k+1} = (V - \sum_{i=1}^{k} a_i) / a_{k+1}$, тобто шукається значення тієї змінної x_i, $i \in I$, яка покладається не булевой. Інші x_i, $i = k+2, ..., n$ приймаються рівними нулю. У результаті одержуємо оцінку $Q = \sum_{i=1}^{k+1} c_i x_i$ й обмеження $\sum_{i=1}^{k+1} a_i < V$.

Розглянемо евристичний алгоритм, що приблизно вирішує оцінну задачу (4).

Він включає наступні дії.

Крок 1. Установити n. Сформувати вектор «цінностей» одиниць вантажів $Y = (c_1/a_1, \mathrm{c}_2/\mathrm{a}_2, .., \mathrm{c_n}/a_n)$.

Крок 2. Упорядкувати компоненти вектора Y по убуванню й одержати відповідний вектор індексів предметів X.

Крок 3. Покласти початкову вагу рюкзака $L = 0$, початкову «цінність» вантажу $z = 0$.

Крок 4. Покласти $i = 1$.

Крок 5. Обчислити $L = L + A(x_i)$.

Крок 6. Якщо $L \le V$, обчислити $z = z + c(x_i)$, інакше обчислити $L = L - c(x_i)$ й перейти до кроку 8.

Крок 7. Покласти $i = i + 1$; якщо $i \le n$, повернутися до кроку 5.

Крок 8. Обчислити недовантаження рюкзака $\Delta V = V - L$ й зупинитися.

На практиці, при оперативному плануванні, якщо потрібне швидке наближене рішення задачі, те можна успішно застосувати евристичний метод рішення задачі.

Дана задача дуже важлива з погляду її застосування в реальному житті. Існує багато алгоритмів рішення задачі про рюкзак. Однак, кожний із цих алгоритмів успішно вирішує певне коло задач. Інтерес до рішення цієї задачі зростає. Усі програми оптимізації завантаження транспорту вирішують задачу відому в науці як задача про ранець. Так, наприклад, оптимальне завантаження транспорту з обліком самих різних обмежень, допомагає скорочувати витрати, одержувати більший прибуток від перевезень вантажів.

Література:

1. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. М.: Мир, 1985. 510 с.

2. Корбут А. А., Финкельштейн Ю. Ю. Дискретное программирование. М.: Наука, 1969. 368 с.

3. Шкурба В. В. Задачи 3-х станков. М.: Наука, 1976. 96 с.

4. Саати Т. Целочисленные методы оптимизации и связанные с ними экстремальные проблемы. М.: Мир, 1973. 304 с.

5. Сигал И. Х., Иванова А. П. Введение в прикладное дискретное программирование: модели и вычислительные алгоритмы: Учебное пособие. М.: Физматлит, 2002. 240 с.

6. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ. М.: Вильямс, 2013. 1328 с.

7. Буркова И. В., Пужанова Е. О., Марин О. Л. Задача о ранце и её модификации. *Научный вестник Воронежского государственного архитектурно-строительного университета. Серия: Управление строительством.* Воронеж: ВГАСУ, 2014. Вып. № 1 (6). С. 103 – 111.

8. Федорин А. Н. Эволюционно-генетические алгоритмы решения задач о ранце. // *Информационные технологии моделирования и управления*. 2007. № 9 (43). С. 1054 – 1062.

9. Васильчиков В. В. О рекурсивно-параллельном алгоритме решения задачи о рюкзаке. *Моделирование и анализ информационных систем*. 2018. Т. 25, № 2. С. 155 – 164. DOI: https://doi.org/10.18255/1818-1015-2018-2-155-164.

10. Чебаков С. В., Серебряная Л. В. Определение структуры оптимального подмножества в задаче о ранце. *Доклады БГУИР*. 2019. № 6 (124). С. 72 – 79. – DOI: https://doi.org/10.35596/1729-7648-2019-124-6-72-79.