КОМПЬЮТЕРНАЯ МАТЕМАТИКА

УДК 519.854.33

В.П. ШИЛО, В.О. РОЩИН, І.П. ГРАДИНАР

НАБЛИЖЕНИЙ АЛГОРИТМ РОЗВ'ЯЗАННЯ ЗАДАЧІ УПАКОВКИ*

Встановлено еквівалентність задачі упаковки та задачі знаходження незалежної множини вершин графу максимальної ваги. Для їх розв'язання розроблено наближений алгоритм. Ефективність його підтверджена експериментально при розв'язанні задач упаковки великої розмірності та порівнянні отриманих результатів з відомими.

Вступ. Однією з широко поширених екстремальних задач на множинах є задача про упаковку множини максимальної ваги, іменована в подальшому задачею упаковки. Вона виникає при складанні графіків руху поїздів, суден і літаків, розподілі роботи, у брокерській діяльності тощо [1 – 3]. Ця задача описується моделлю булевого лінійного програмування з яскраво виявленими структурними особливостями, які дозволяють розробляти спеціальні алгоритми її розв'язання. Такі алгоритми, як правило, є більш ефективними, ніж загальні методи цілочислового програмування. Побудові та дослідженню одного з них і присвячена дана робота.

Постановка задачі. Задача упаковки полягає у пошуку

$$\max\left\{ f(x) = \sum_{j=1}^{n} c_j x_j \right\} \tag{1}$$

за умов

$$\sum_{j=1}^{n} a_{ij} x_j \le 1, \ i = 1,...,m, \tag{2}$$

$$x_j \in \{0,1\}, \ j = 1,...,n, \tag{3}$$

де $c_j \ge 0$, $a_{ij} \in \{0,1\}$, $j = 1,...,n$, $i = 1,...,m$.

Еквівалентність задач. Розглянемо нерівність

$$x_1 + x_2 + ... + x_n \le 1 \tag{4}$$

та систему нерівностей

* Робота виконана за часткової фінансової підтримки Українського науково-технологічного центру (грант № 5710)

$$\left.\begin{aligned} x_1 + x_2 &\le 1, \\ &\dots \\ x_1 + x_n &\le 1, \\ x_2 + x_3 &\le 1, \\ &\dots \\ x_{n-1} + x_n &\le 1. \end{aligned}\right\} \quad (5)$$

Система (5) містить нерівності з усіма парами змінних $x_1, x_2, ..., x_n$.

Твердження 1. Співвідношення (4) і (5) при виконанні умови (3) еквівалентні.

Доведення. Очевидно, що нерівність (4) задовольняє нульовий вектор та всі булеві вектори, в яких тільки одна компонента дорівнює одиниці, а решта – нулі.

Нульовий вектор є також допустимим розв'язком системи нерівностей (5). Припустимо, що деяка компонента x_j, $j = 1,...,n$, вектора x, наприклад, x_1, приймає значення 1. Просумувавши всі нерівності з (5), що містять x_1, отримаємо $(n-1)x_1 + x_2 + ... + x_n \le (n-1)$. Оскільки $x_1 = 1$, то $x_2 + x_3 + ... + x_n \le 0$. З умови (3) випливає, що $x_i = 0$, $i = 2,...,n$. Якщо інша компонента вектора x дорівнює 1, доведення проводиться аналогічно.

Отже, як і нерівність (4), систему нерівностей (5) задовольняють тільки нульовий вектор або вектори, в яких тільки одна компонента приймає значення 1, а інші дорівнюють 0. Таким чином, при виконанні умови (3) співвідношення (4) та (5) еквівалентні. Твердження доведено.

Розглянемо задачу знаходження незалежної множини вершин графу максимальної ваги та доведемо еквівалентність її задачі упаковки. Нехай задано неорієнтований граф $G = G(V, E)$ з множиною вершин $V = \{v_1,...,v_n\}$ та множиною ребер E. Підмножина I множини V вершин графу G називається незалежною, якщо ніякі дві її вершини не зв'язані ребром. Нехай кожній вершині графу G відповідає її вага – довільне число $w_j \ge 0$, $j = 1, …, |V|$, $|V|$ – потужність множини V. Задача полягає у знаходженні незалежної множини вершин графу максимальної сумарної ваги. Поставимо у відповідність кожній вершині $v_j \in V$ графу G змінну $x_j \in \{0,1\}$, $j = 1,...,n$; $x_j = 1$, якщо $v_j \in I$, у протилежному випадку – $x_j = 0$. Тоді математичну модель цієї задачі можна подати у вигляді: знайти

$$\max\left\{ f(x) = \sum_{j=1}^{n} w_j x_j \right\} \quad (6)$$

за умов

$$x_i + x_j \le 1,\ (v_i, v_j) \in E,\ i, j = 1,...,n, \quad (7)$$

$$x_j \in \{0,1\},\quad j = 1,...,n, \quad (8)$$

де $w_j \ge 0$, $j = 1,...,n$.

Твердження 2. Математичні моделі (1)–(3) та (6)–(8) еквівалентні.

Доведення. Оскільки у задачах (1)–(3) і (6)–(8) цільові функції (1) та (6) однакові, на основі твердження 1 робимо висновок про справедливість даного твердження.

Фактично (1)–(3) – математична модель задачі знаходження незалежної множини вершин графу максимальної ваги, коли у кожному рядку матриці $[a_{ij}]_{mxn}$ не більше двох елементів дорівнює 1. У протилежному випадку умови (2) відображають кліки в графі, і може бути вибрана тільки одна вершина з такої кліки. Якщо одну нерівність (2) замінити системою нерівностей вигляду (5), це означає, що з усіх розглянутих вершин, знову ж таки, може бути вибрана тільки одна. Тобто, навіть коли кількість ненульових елементів принаймні в одному рядку матриці $[a_{ij}]_{mxn}$ більше двох, задачу упаковки теж можна розглядати як задачу знаходження незалежної множини вершин графу максимальної ваги. Про це свідчить попереднє твердження.

Алгоритм. Оскільки згідно твердження 2 моделі (1)–(3) та (6)–(8) еквівалентні, до задачі упаковки можна застосовувати алгоритми знаходження незалежної множини вершин графу максимальної ваги. З цією метою нами використано підхід до наближеного розв'язання деяких задач дискретної оптимізації на графах, заснований на ідеях методу глобального рівноважного пошуку [4, 5]. Цей підхід був успішно застосований, зокрема, до задач знаходження максимальної ρ-щільної множини вершин графу [6] та максимального k-plex (co-k-plex) графу [7].

Найкращий із усіх допустимих розв'язків задачі, розглянутих до даного кроку обчислювального процесу, будемо називати рекордним, а відповідне значення цільової функції – рекордом.

Наближений алгоритм розв'язання задачі упаковки вигляду (1)–(3) складається з двох етапів. Спочатку вибираємо початкове наближення $x^0 = (0,...,0)$ – допустимий розв'язок даної задачі. Далі випадковим чином вибираємо компоненту $x_j^k = 0,\ \ j = 1,...,n,\ \ k = 0,1,...,$ вектора $x^k = (x_1^k,...,x_n^k)$ та змінюємо її на $x_j^k = 1$, а k – на k+1. Якщо вектор x^{k+1} є допустимим розв'язком задачі (1)–(3), то повторюємо цей крок. У протилежному випадку компоненту $x_j^k = 0$ залишаємо без змін. На кожному кроці запам'ятовуємо лише рекордний розв'язок. Повторення описаних кроків здійснюємо доти, поки не знайдеться компонента $x_j^k = 0$, яку можна замінити на $x_j^k = 1$ при виконанні умови (2) для відповідного вектора.

На другому етапі для отримання кращого розв'язку використовуємо ідею методу глобального рівноважного пошуку [4, 5]. Вона полягає в адаптивній імовірнісній зміні частини одиничних елементів знайденого рекордного розв'язку на нульові. Далі у відповідності з вищеописаною процедурою поповнюємо отриманий вектор одиничними елементами замість нульових. Цей процес триває до виконання деякого критерію (наприклад, часу, виділеного на розв'язання задачі, отримання заданого рекорду та ін.).

Оскільки запропонований алгоритм має РЕСТАРТ-розподіл часу розв'язання задач вигляду (1)–(3), до нього була застосована РЕСТАРТ-технологія [8]. Це дозволило прискорити час розв'язання таких задач.

Схема запропонованого алгоритму має такий вигляд:

while виділено_час_або_кількість_ітерацій ( );
  знаходження_розв'язку_на_першому_етапі ( );
  while перевірка_умови_рестарту ( );
    заміна_частини_одиничних_елементів_розв'язку_на_нульові ( );
    пошук_кращого_розв'язку ( );
    запам'ятовування_рекорду ( );
  end while;
end while.

Результати експериментальних розрахунків. Для дослідження ефективності запропонованого алгоритму проведено обчислювальні експерименти з використанням PC з Intel® Core^{TM}2 QUAD CPU, Q9550 2.83GHz, 8.0GB оперативної пам'яті при 25 % завантаженні. Розв'язано 64 тестові задачі упаковки великої розмірності, взяті з сайту http://www.emse.fr/~delorme/SetPacking.html [3]. Кожна задача розв'язувалась 10 разів при різних початкових наближеннях. У табл. 1, 2 містяться результати цих досліджень. Для 56 задач проведено їх порівняння з результатами розв'язання таких задач алгоритмами GRASP та мурашиних колоній (ACO) [1] на PC Pentium III 800 MHz. У табл. 1. використано наступні позначення: n і m – відповідно кількість змінних та обмежень задачі; ρ – щільність матриці $[a_{ij}]_{mxn}$ (під щільністю розуміємо відношення числа одиничних елементів до загальної кількості елементів матриці); * біля відомого рекорду означає оптимальність знайденого значення цільової функції; $f_{сер}$ і $t_{сер}$ – відповідно середнє значення цільової функції та часу (в сек.) розв'язання задачі розробленим алгоритмом; $t_{сер}(GRASP)$ і $t_{сер}(ACO)$ – середній час (в сек.) розв'язання задачі відповідно алгоритмами GRASP і мурашиних колоній ACO.

ТАБЛИЦЯ 1. Результати розв'язання задач упаковки

№	Задача	n	m	ρ (%)	Відомий рекорд	Отриманий рекорд	$f_{сер}$	$t_{сер}$	$\frac{t_{сер}(GRASP)}{t_{сер}}$	$\frac{t_{сер}(ACO)}{t_{сер}}$
1	2	3	4	5	6	7	8	9	10	11
1.	pb_100rnd0100	100	500	2.00	**372***	372.000	372.00	<0.001	1970,0	3330,0
2.	pb_100rnd0200	100	500	2.00	**34***	34.000	34.00	<0.001	1310,0	2000,0
3.	pb_100rnd0300	100	500	2.96	**203***	203.000	203.00	<0.001	1140,0	2000,0
4.	pb_100rnd0400	100	500	3.03	**16***	16.000	16.00	<0.001	1290,0	670,0
5.	pb_100rnd0500	100	100	2.00	**639***	639.000	639.00	<0.001	800,0	1670,0
6.	pb_100rnd0600	100	100	2.00	**64***	64.000	64.00	<0.001	690,0	1000,0
7.	pb_100rnd0700	100	100	2.93	**503***	503.000	503.00	0.008	125,0	125,0
8.	pb_100rnd0800	100	100	3.07	**39***	39.000	39.00	<0.001	570,0	670,0
9.	pb_100rnd0900	100	300	2.00	**463***	463.000	463.00	<0.001	1260,0	1670,0
10.	pb_100rnd1000	100	300	2.00	**40***	40.000	40.00	<0.001	1280,0	1000,0

Продовження табл. 1

1	2	3	4	5	**6**	7	8	9	10	11
11.	pb_100rnd1100	100	300	3.08	**306***	306.000	306.00	<0.001	680,0	1670,0
12.	pb_100rnd1200	100	300	2.97	**23***	23.000	23.00	<0.001	1130,0	330,0
13.	pb_200rnd0100	200	1000	1.49	**416***	416.000	416.00	0.019	385,3	1438,4
14.	pb_200rnd0200	200	1000	1.49	**32***	32.000	32.00	<0.001	7350,0	14670,0
15.	pb_200rnd0300	200	1000	1.00	**731***	731.000	731.00	<0.001	10810,0	44330,0
16.	pb_200rnd0400	200	1000	1.00	**64***	64.000	64.00	0.006	1520,0	4055,0
17.	pb_200rnd0500	200	1000	2.48	**184***	184.000	184.00	<0.001	4620,0	16000,0
18.	pb_200rnd0600	200	1000	2.49	**14***	14.000	14.00	<0.001	3480,0	4000,0
19.	pb_200rnd0700	200	200	1.53	**1004***	1004.000	1004.00	0.128	32,8	49,5
20.	pb_200rnd0800	200	200	1.50	**83***	83.000	83.00	<0.001	2710,0	2670,0
21.	pb_200rnd0900	200	200	1.00	**1324***	1324.000	1324.00	2.950	1,3	2,5
22.	pb_200rnd1000	200	200	1.00	**118***	118.000	118.00	<0.001	3640,0	4000,0
23.	pb_200rnd1100	200	200	2.48	**545***	545.000	545.00	0.016	147,5	270,6
24.	pb_200rnd1200	200	200	2.57	**43***	43.000	43.00	<0.001	1010,0	1330,0
25.	pb_200rnd1300	200	600	1.50	**571***	571.000	571.00	0.011	546,4	1848,2
26.	pb_200rnd1400	200	600	1.49	**45***	45.000	45.00	<0.001	3920,0	8670,0
27.	pb_200rnd1500	200	600	1.00	**926***	926.000	926.00	0.008	527,5	3375,0
28.	pb_200rnd1600	200	600	1.00	**79***	79.000	79.00	<0.001	6800,0	15330,0
29.	pb_200rnd1700	200	600	2.48	**255***	255.000	255.00	<0.001	3610,0	11000,0
30.	pb_200rnd1800	200	600	2.56	**19***	19.000	19.00	0.042	56,0	71,4
31.	pb_500rnd0100	500	2500	1.23	**323**	323.000	323.00	0.067	478,8	2308,5
32.	pb_500rnd0200	500	2500	1.20	**25**	25.000	25.00	1.294	19,8	20,6
33.	pb_500rnd0300	500	2500	0.70	**776**	776.000	776.00	0.211	333,3	1156,4
34.	pb_500rnd0400	500	2500	0.70	**62**	62.000	62.00	0.089	643,8	775,3
35.	pb_500rnd0500	500	2500	2.22	**122***	122.000	122.00	0.258	60,0	275,2
36.	pb_500rnd0600	500	2500	2.19	**8**	8.000	8.00	0.025	483,2	386,8
37.	pb_500rnd0700	500	500	1.20	**1141***	1141.000	1141.00	0.030	447,7	2000,0
38.	pb_500rnd0800	500	500	1.19	**89***	89.000	89.00	<0.001	15800,0	21670,0
39.	pb_500rnd0900	500	500	0.70	**2236***	2236.000	2236.00	0.245	95,7	344,2
40.	pb_500rnd1000	500	500	0.70	**179***	179.000	179.00	<0.001	18200,0	44670,0
41.	pb_500rnd1100	500	500	2.26	**424***	424.000	424.00	0.511	37,7	73,1
42.	pb_500rnd1200	500	500	2.18	**33**	33.000	33.00	<0.001	11910,0	8000,0
43.	pb_500rnd1300	500	1500	1.21	**474**	474.000	474.00	0.017	1934,1	6176,5
44.	pb_500rnd1400	500	1500	1.21	**38**	38.000	38.00	0.038	546,6	675,3
45.	pb_500rnd1500	500	1500	0.69	**1196**	1196.000	1196.00	0.064	927,5	2526,1
46.	pb_500rnd1600	500	1500	0.70	**88**	88.000	88.00	0.005	7262,0	12134,0
47.	pb_500rnd1700	500	1500	2.17	**192**	192.000	192.00	0.175	105,0	325,7
48.	pb_500rnd1800	500	1500	2.20	**13**	13.000	13.00	0.006	2005,0	1500,0
49.	pb_1000rnd0100	1000	5000	2.60	**67***	67.000	67.00	610.656	0,1	0,2
50.	pb_1000rnd0200	1000	5000	2.59	**4***	4.000	4.00	551.427	0,1	<0,1
51.	pb_1000rnd0300	1000	5000	0.60	**661**	661.000	661.00	0.644	343,5	1088,0
52.	pb_1000rnd0400	1000	5000	0.60	**48**	48.000	48.00	0.341	439,0	317,7
53.	pb_1000rnd0500	1000	1000	2.60	**222**	222.000	222.00	0.305	212,5	280,9
54.	pb_1000rnd0600	1000	1000	2.65	**15**	15.000	15.00	0.295	140,3	27,1
55.	pb_1000rnd0700	1000	1000	0.58	**2260**	2260.000	2260.00	26.416	4,5	11,2
56.	pb_1000rnd0800	1000	1000	0.60	**175**	175.000	175.00	1.464	56,4	64,2
57.	pb_2000rnd0100	2000	10000	2.54	**40***	40.000	33.70	1104.223		

Закінчення табл. 1

1	2	3	4	5	6	7	8	9	10	11
58.	pb_2000rnd0200	2000	10000	2.55	**2***	2.000	2.00	0.541		
59.	pb_2000rnd0300	2000	10000	0.55	**478**	478.000	478.00	38.169		
60.	pb_2000rnd0400	2000	10000	0.55	32	**33.000**	**32.80**	1263.610		
61.	pb_2000rnd0500	2000	2000	2.55	**140**	140.000	140.00	205.891		
62.	pb_2000rnd0600	2000	2000	2.56	**9**	9.000	9.00	0.200		
63.	pb_2000rnd0700	2000	2000	0.56	**1811**	1811.000	1811.00	13.814		
64.	pb_2000rnd0800	2000	2000	0.56	**135**	135.000	135.00	33.595		

Аналіз наведених у табл. 1, 2 даних показує, що за допомогою розробленого алгоритму отримано відомі рекорди для всіх 64-х тестових задач упаковки великої розмірності, причому 40 розв'язків є точними [3]. Для задачі pb_2000rnd0400 знайдено новий рекорд. Крім того, розв'язання цих задач запропонованим алгоритмом потребує значно менших часових затрат у порівнянні з алгоритмами GRASP і мурашиних колоній ACO [1]. (Виключенням є лише 3 задачі).

ТАБЛИЦЯ 2. Порівняння результатів експериментальних розрахунків

Алгоритми	Розроблений	GRAS P	ACO
Кількість спроб розв'язання кожної задачі	10	16	16
Кількість розв'язаних задач	64	56	56
Кількість задач, розв'язаних трьома алгоритмами (по них проводилось порівняння)	**56**		
Кількість задач, для яких досягнуто відомий рекорд хоча б при одній спробі розв'язання	56	46	52
Кількість задач, для яких досягнуто відомий рекорд при всіх спробах розв'язання	55	22	14
Кількість задач, середній час розв'язання яких < 1 с	50	4	3
Кількість задач, середній час розв'язання яких < 0,01с	29	0	0

Висновок. У роботі доведено еквівалентність задачі упаковки та задачі знаходження незалежної множини вершин графу максимальної ваги. Розроблено наближений алгоритм розв'язання цих задач, що базується на використанні ідей методу глобального рівноважного пошуку. В результаті проведених експериментальних розрахунків отримано відомі рекорди для всіх 64 тестових задач упаковки великої розмірності, причому 40 розв'язків є точними [3]. Для однієї задачі знайдено новий рекорд. Крім того, порівняння отриманих результатів з результатами роботи алгоритмів GRASP і мурашиних колоній ACO [1] показало, що розроблений алгоритм майже у всіх випадках значно кращий за швидкодією.

В.П. Шило, В.А. Рощин, И.П. Градинар

ПРИБЛИЖЕННЫЙ АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ УПАКОВКИ

Установлена эквивалентность задачи упаковки и задачи нахождения независимого множества вершин графа максимального веса. Для их решения разработан приближенный алгоритм. Эффективность его подтверждена экспериментально при решении задач упаковки большой размерности и сравнении полученных результатов с известными.

V.P. Shylo, V.O. Roshchyn, I.P. Gradinar

AN APPROXIMATE ALGORITHM TO SOLVE PACKING PROBLEM

In the paper, the equivalence of packing problem and the problem of finding an independent set of graph nodes of maximum weight is established. The approximation algorithm is proposed for this problem. Its efficiency is confirmed experimentally by solving packing problems of high dimensionality and comparing the results obtained with known ones.

1. *Gandibleux X., Delorme X., T'Kindt V.* An ant colony algorithm for the set packing problem // 4th International Workshop «Ant Colony Optimization and Swarm Intelligence» (ANTS 2004) / Lecture Notes in Computer Science / Edited by M. Dorigo, M. Birattari, C. Blum, L. M. Gambardella, F. Mondada and F. Stutzle. – Berlin: Springer, 2004. – **3172.** – P. 49–60. – Mode of access: http://www.univ-valenciennes.fr/ROI/WP/download/2004/13-2004.pdf (October 30, 2011).
2. *Guo Y.*, *Lim A.*, *Rodrigues B.*, *Zhu Y.* Heuristics for a brokering set packing problem // The 8th International Symposium on Artificial Intelligence and Mathematics (AMAI 2004). – Mode of access: http://www.cs.cornell.edu/~guoys/publications/AIMA2004_Guo.pdf (October 30, 2011).
3. *Benchmarks* for the Set Packing Problem: (X. Delorme. Personal Homepage). – Mode of access: http://www.emse.fr/~delorme/SetPacking.html (October 30, 2011).
4. *Шило В.П.* Метод глобального равновесного поиска // Кибернетика и системный анализ. – 1999. – № 1. – С. 74 – 81.
5. *Сергиенко И.В.*, *Шило В.П.* Задачи дискретной оптимизации. Проблемы, методы решения, исследования. – Киев: Наукова думка, 2003. – 264 с.
6. *Шило В.П.*, *Рощин В.А.*, *Градинар И.П.* Приближенное решение задачи нахождения максимального ρ-плотного множества вершин графа // Компьютерная математика. – 2011. – № 1. – С. 157 – 164.
7. *Шило В.П.*, *Градинар І.П.*, *Ляшко В.І.* Наближений алгоритм знаходження максимального k -plex (co-*k*-plex) графу // Наукові записки НаУКМА: Комп'ютерні науки. – 2011. – **125.** – С. 17 – 22.
8. *Сергиенко И.В*., *Шило В.П.*, *Рощин В.А*. РЕСТАРТ-технология решения задач дискретной оптимизации // Кибернетика и системный анализ. – 2000. – № 5. – С. 32 – 40.

Одержано 29.01.2013

Про авторів:

Шило Володимир Петрович,
доктор фізико-математичних наук, провідний науковий співробітник
Інституту кібернетики імені В.М. Глушкова НАН України,
E-mail: v.shylo@gmail.com

Рощин Валентина Олексіївна,
кандидат фізико-математичних наук, старший науковий співробітник
Інституту кібернетики імені В.М. Глушкова НАН України,
E-mail: dopt135@gmail.com

Градинар Іван Петрович,
асистент Ужгородського національного університету.
E-mail: vgradinar@gmail.com