УДК 519.852: 004.424.27.021

Б.И. Юхименко, канд. экон. наук, доц.,
Ю.Ю. Козина, магистр,
Одес. нац. политехн. ун-т

СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА АЛГОРИТМОВ МЕТОДА ВЕТВЕЙ И ГРАНИЦ ДЛЯ РЕШЕНИЯ ЗАДАЧ ЦЕЛОЧИСЛЕННОГО ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

Б.І. Юхименко, Ю.Ю. Козіна. **Порівняльна характеристика алгоритмів методу гілок та меж для розв'язання задач цілочисельного лінійного програмування.** Наведено результати дослідження швидкості збіжності деяких алгоритмів методу гілок та меж. Алгоритми однакові за способами оцінювання безлічі варіантів розв'язання задач та відрізняються способами галуження. Оцінювання здійснено шляхом розв'язання декількох одновимірних задач про ранець, проте, змінним присвоєно: значення "нуль" або значення умовної одиниці, що обчислюється на основі системи обмежень задачі. Розглянуто два способи галуження: з послідовним вибором змінної, що конкретизується, та комбінований, при якому номер змінної, що конкретизується, обчислюється спеціальним способом.

B.I. Yukhimenko. Yu.Yu. Kozina. **Comparative characteristic of branch and bound method algorithms for solving the problems of integer linear programming.** The results of studying some branch and bound method algorithms for convergence speed are presented. The algorithms are alike by the methods of estimating the variants set of problems solving and are different by the ways of branching. Estimation is performed by solving several one-dimensional knapsack problems, assuming, however, that the variable accepts the value 0 or a so-called standard unit, that is calculated on the basis of the task restrictions system. There are two ways of branching: by sequential variable selection and a combined one, when the number of a concretized variable is calculated in a special way.

Для комбинаторных методов решения задач целочисленного линейного программирования (ЦЛП) характерно использование конечности множества вариантов решения и замена полного перебора всех вариантов решения частичным. Уменьшение перебора осуществляется отсеиванием неперспективных решений, заведомо несодержащих оптимального.

Одно из центральных мест среди методов этой группы занимает метод ветвей и границ [1]. Метод применяется для определения максимального значения функции $f(x)$ на некотором конечном множестве G вариантов решения. Он опирается на следующие построения:

— вычисление верхней границы значения целевой функции $f(x)$ либо на множестве G, либо на некотором его подмножестве (оценивание);

— постепенное разбиение множества G на подмножества (ветвление);

— пересчет оценок при постепенном разбиении множеств;

— нахождение конкретных вариантов решения;

— проверка вариантов на оптимальность.

Алгоритмы, реализующие метод ветвей и границ, в основном отличаются способами ветвления и вычислением оценок [2]. Близким к ним является аддитивный алгоритм Балаша для решения задач ЦЛП с булевыми переменными [3].

Известны различные подходы к вычислению оценок и проведению процедуры ветвления [4, 5]. Сравнительная характеристика этих подходов не сделана, и дать оценку скорости сходимости невозможно.

Сделана попытка обобщить различные способы получения оценок. Дана экспериментальная оценка скорости сходимости при последовательном и комбинированном способах ветвления.

Пусть рассматривается оптимизационная задача в формальной постановке

$$z = \max_{x \in G} f(x),$$

где G — конечное множество.

Для получения оценок обычно создается такое множество R (расширение множества вариантов решения), что $G \subseteq R$. На R определяется функция $g(x)$ такая, что $g(x) \geq f(x)$ для всех $x \in G$.

Простейшим способом расширения множества G до множества R в задачах ЦЛП является отбрасывание требования целочисленности искомых величин и, тем самым, получение задачи линейного программирования (ЛП). Задача ЦЛП формулируется как

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

при ограничениях

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i , \; i = \overline{1,m}, \tag{2}$$

$$x_j \geq 0, \; j = \overline{1,n}, \tag{3}$$

$$x_j \text{ — целое, } j = \overline{1,n}, \tag{4}$$

где c_j, a_{ij}, b_i, m, n — константы, предопределяющие аналитический вид функций и размерность задачи.

Совокупность ограничений (2)...(4) составляет множество G, а совокупность (2) и (3) — множество R. Очевидно, что $G \subseteq R$. Тогда функция $g(x)$ предопределяет решение задачи ЛП, как задачи в постановке

$$z = \max \sum_{j=1}^{n} c_j x_j ,$$

$$x \in R .$$

Расширение множества вариантов решений при решении задач ЦЛП с булевыми переменными, когда ограничения (3), (4) представляются в виде

$$x_j = 0 \vee 1, \; j = \overline{1,n}, \tag{5}$$

осуществляется путем решения m ($i = \overline{1,m}$) нецелочисленных задач о ранце [6] вида

$$z_i = \max \sum_{j=1}^{n} c_j x_j ,$$

при ограничениях

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i , \tag{6}$$

$$x_j \in [0,1], \; j = \overline{1,n}. \tag{7}$$

Естественно, что множество вариантов решений, определяемых ограничениями (2) и (5), включаются во множество, определяемое ограничениями (6) и (7). Оценкой множества вариантов считается величина ξ, для которой справедливо соотношение

$$\xi = \min_i z_i \,.$$

Изложенный подход расширения множества вариантов решений был использован для решения задач ЦЛП в постановке (1)…(4) [7]. Предполагается, что целочисленная искомая величина x_j принимает значение либо “нуль”, либо “условная единица”, т.е. $x_j = 0 \vee d_j$, где $d_j = \min_i \left[\frac{b_i}{a_{ij}} \right]$.

Для получения оценки решается m нецелочисленных задач о ранце, но ограничение типа (7) имеет вид

$$x_j \in [0, d_j], \; j = \overline{1,n}.$$

Способ постепенного разбиения множества вариантов G на подмножества (процедура ветвления) также значительно влияет на скорость сходимости алгоритмов метода ветвей и границ [5]. Возможно исключить из рассмотрения нецелочисленные значения искомой величины в окрестности ее оптимального значения путем введения соответствующих дополнительных ограничений [1]. В алгоритме Балаша для решения задач с булевыми переменными используется процедура одностороннего ветвления — выбор компонент, которым “выгодно” присвоить значение “1” [3]. В данном алгоритме сформирован несколько специфический признак оптимальности, который является довольно слабым, и при решении задачи этим методом уходит много времени на доказательство оптимальности давно полученного оптимального решения.

Используя идею последовательного построения решений [8], процедуру ветвления можно представить в виде последовательной компоновки решения, на каждом шаге которой конкретизируется значение конкретной переменной из интервала возможных ее значений. Вершинам дерева подмножеств будет соответствовать подмножество вариантов, содержащее частичное решение — некоторое σ^k, в котором конкретизировано k компонент варианта решения ($k < n$). Переход от одного уровня дерева к другому — это переход от частичного решения σ^k к σ^{k+1}. Ветви уровня σ^k сопоставляются с процедурой конкретизации некоторой, определенным образом выбранной компоненты. Вершины уровня σ^{k+1}— это подмножества вариантов, содержащие частичные решения, где конкретизируемые переменные принимают некоторые возможные значения.

Формально сказанное представляется так. Если σ^k — имеющееся частичное решение ($k < n$), то $\sigma^{k+1}=\{\sigma^k \bigcup x_l\}$, где x_l — выбранная компонента. Частичных решений σ^{k+1} будет столько, сколько возможных целочисленных значений может принять переменная x_l.

Пусть $J\sigma^k$ — множество индексов конкретизированных переменных в частичном решении σ^k

$$J\sigma^k = \{ j / x_j \in \sigma^k \}.$$

Тогда наибольшее значение, которое может иметь конкретизируемая переменная на $k+1$ уровне, определяется как

$$d_{x_l} = \min_i \left[\frac{b_i - \sum\limits_{j \in J_{\sigma^k}} a_{ij} x_j}{a_{i_l}} \right].$$

Переменная x_l принимает все целочисленные значения из интервала $[0, d_{x_l}]$, т.е. $x_l \in [0, d_{x_l}]$.

Скорость сходимости алгоритмов метода ветвей и границ также значительно зависит от способа ветвления. Получены результаты экспериментального исследования на скорость сходимости алгоритмов при различных процедурах выбора конкретизируемой переменной, и способах разбиения подмножеств вариантов на более мелкие подмножества. Пусть имееются две функции $\Psi_1(\sigma^k)$ и $\Psi_2(\sigma^k)$ на множестве вариантов G, при помощи которых выбирается переменная x_l, подлежащая конкретизации при переходе от частичного решения σ^k к σ^{k+1}.

Функция $\Psi_1(\sigma^k)$ — последовательный выбор — предопределяет переменную x_l, номер которой соответствует уровню дерева подмножеств, т.е. если

$$\sigma^k=\{x_1,x_2,\ldots,x_k\}\text{, то } \sigma^{k+1}=\{x_1,x_2,\ldots,x_k,x_{k+1}\}\ .$$

Функция $\Psi_2(\sigma^k)$ — комбинированный выбор — предопределяет переменную x_l, учитывая величину неувязки в системе ограничений (2), относительный разброс коэффициентов целевой функции (1), а также величину условной единицы переменных, которые являются претендентами на очередную конкретизацию.

Коэффициенты правых частей (неувязки) системы ограничений определяются по формуле

$$b_i^s=b_i-\sum_{j\in J_{\sigma^k}} a_{ij}x_j\ ,\qquad i=\overline{1,m}\ ,$$

где σ^k — имеющееся частичное решение,

b_i^s — множество всех ограничений задачи.

Множество индексов переменных — претендентов на конкретизацию — определяется как множество V_j^{k+1}, которым можно присвоить значение, отличное от нуля, т.е.

$$V_j^{k+1}=\{j/b_i^s-a_{ij})>0 \text{ для всех } i=\overline{1,m} \text{ и } j\notin J\,\sigma^k.$$

Каждый из претендентов оценивается величиной

$$P_j=\sum_{i=1}^{m}(b_i^s-a_{ij})\sqrt{c_j/\min c_j}\ d_j\ ,\quad j\in V_j^{k+1}\ ,$$

где d_j — величина условной единицы.

Конкретизации подлежит компонента x_l с наибольшей оценкой

$$P_l=\max_{j\in V_j^{k+1}} P_j\ .$$

Эффективность функций $\Psi_1(\sigma^k)$ и $\Psi_2(\sigma^k)$ и их влияние на скорость сходимости алгоритмов метода ветвей и границ исследовались экспериментальным путем. Разработан программный продукт, реализованный на персональном компьютере. Результаты проведенного эксперимента приведены в таблице.

При проведении эксперимента для каждой размерности задачи было сгенерировано 25 задач. Приведенные время сходимости и количество итераций являются средними. Все составляющие задач являются равномерно распределенными величинами из интервала 2…150.

Экспериментальные данные о скорости сходимости алгоритмов метода ветвей и границ при различных процедурах выбора конкретизируемой переменной

Размерность задачи, $m\times n$	Тип выбора конкретизируемых переменных			
	$\Psi_1(\sigma^k)$		$\Psi_2(\sigma^k)$	
	Время сходимости, с	Число итераций	Время сходимости, с	Число итераций
3×5	0,10	4	0,0	3
7×15	0,20	11	0,10	9
11×25	0,25	21	0,12	19
15×35	0,30	34	0,20	30
19×45	0,35	41	0,24	38

23×55	0,40	50	0,30	42
27×65	0,45	64	0,32	61
31×75	0,50	71	0,20	63
35×85	0,80	82	0,60	70
39×95	2,00	90	1,20	82
43×105	2,50	102	1,80	84
47×168	4,50	162	3,20	148
51×231	8,90	229	6,30	168
55×294	11,00	293	9,40	198
59×357	12,50	354	10,00	268
63×420	13,70	418	11,20	390
67×483	16,40	480	14,60	440
71×546	18,90	542	16,60	530
75×609	19,00	603	17,00	590
79×672	19,60	671	17,80	654
83×735	21,30	731	18,00	690
87×798	22,00	797	18,20	740
91×861	22,40	860	18,60	820
100×1000	24,80	1001	22,00	974

Исходя из полученных результатов, можно сделать вывод о том, что скорость сходимости алгоритма с использованием функции $\Psi_2(\sigma^k)$ выше, особенно для задач меньшей размерности. С увеличением размерности задачи разница между количествами итераций с различными способами выбора конкретизируемой переменной уменьшается. Это говорит от том, что скорость сходимости алгоритмов метода ветвей и границ все равно главным образом зависит от количества искомых величин и меньше зависит как от способов ветвления, так и оценивания, т.е. алгоритм остается P-полным алгоритмом.

Приведены деревья решений с различными способами ветвления задачи (см. рисунок).

$$Z = \max(4x_1 + 3x_2 + 2x_3 + 10x_4),$$

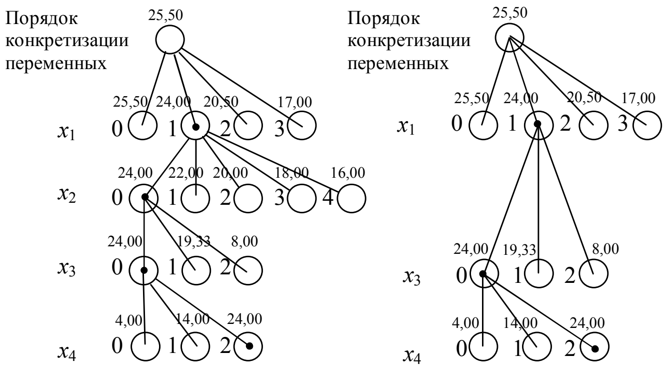
при ограничениях

$$\begin{cases} 2x_1 + 1x_2 + 4x_3 + 3x_4 \le 10, \\ 3x_1 + 2x_2 + 1x_3 + 4x_4 \le 11, \end{cases}$$

$$x_j \ge 0, \qquad j = \overline{1,4},$$

$$x_j \text{ — целое, } j = \overline{1,4}.$$

Очевидно, что некоторые компоненты варианта решений при комбинированном выборе вообще не рассматриваются, что уменьшает количество пересматриваемых вершин и ускоряет сходимость алгоритма метода ветвей и границ.



а *б*

Деревья решений с последовательным (а) и комбинированным (б) выбором конкретизируемых переменных, выделенные линии указывают процесс получения оптимального варианта решения задачи

Литература

1. Land A.H. An Automatic method of solving discrete programming problems / Land A.H., Doig A.G. // Economet-

rica. — 1960. — Vol. 28, № 3. — P. 497 — 520.
2. Корбут А.А. Метод ветвей и границ (обзор теории, алгоритмов, программ и приложений) / Корбут А.А. Сигал И.Х., Финкельштейн Ю.Ю. // Math. Operationsforsch. Statist. Ser.Optimization. — 1997. —Vol. 20, № 2. — P. 253 — 280.
3. Balas E. An additive algorithm for solving linear programs with zero – one variables // Oper. Res. — 1965. — Vol. 13, № 4. — P.517 — 546.
4. Корбут А.А. Дискретное программирование / Корбут А.А., Финкельштейн Ю.Ю. — М.: Наука, 1969. — 320 с.
5. Юхименко Б.І. Вибір ефективного алгоритму розв’язання задач цілочисельного лінійного програмування // Тр. Одес. политехн. ун-та. — 2003. — Вып. 2. — С. 172 — 176.
6. Юхименко Б.И. Разработка и обоснование некоторых схем и методов дискретной оптимизации в автоматизации функций распределения – размещения в АСУ: Автореф дис… канд. экон. наук. — Одесса, 1982. — 20 с.
7. Юхименко Б.И. Ускоренный алгоритм метода ветвей и границ для решения задачи целочисленного линейного программирования // Тр. Одес. политехн. ун-та. — 2004. — Вып. 2. — С. 223 — 226.
8. Михалевич В.С. Последовательные алгоритмы оптимизации и их применение // Кибернетика. — 1965, № 1. — С. 45 — 55.

Поступила в редакцию 20 апреля 2005 г.