Реализация метода ветвей и границ в задаче о бинарном рюкзаке

Кононов Сергей 411 группа

April 2019

1 Постановка задачи

Задачей о рюкзаке или ранце называется задача целочисленного линейного программирования с одним ограничением:

$$\sum_{j=1}^{n} c_j x_j \to \max$$
$$\sum_{j=1}^{n} c_j x_j \leq b,$$
$$x_j \geq 0 \text{ и целые}, \ j = 1, ..., n$$

Числа c_j, a_j $(j = 1, ..., n)$, b можно считать положительными и целыми, $a_j \leq b$.

Своё название задача получила благодаря следующей её интерпретации. Имеется n видов неделимых предметов со стоимостями $c_1, ..., c_n$ и весами $a1, ..., a_n$. Требуется так упаковать ими рюкзак, чтобы его вес не превышал b, а суммарная стоимость упакованных предметов была максимальной. Вводя переменные x_j, $j = 1, ..., n$, для количества упакованных предметов каждого типа, получаем задачу (1.2) – (1.4).
Иногда рассматривается задача о рюкзаке с ограничением типа равенства

$$\sum_{j=1}^{n} a_j x_j = b,$$

что соответствует требованию полной загрузки рюкзака грузоподъёмностью b. Задача о рюкзаке (1.2) – (1.4) может быть сведена к задаче с ограничением типа равенства введением дополнительной неотрицательной целочисленной переменной

$$x_{n+1} = b - \sum_{j=1}^{n} a_j x_j.$$

Если считать, что имеется ровно по одному предмету каждого типа, то ограничение следует заменить условием

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

то есть $x_j = 1$, если j-й предмет кладется в рюкзак и $x_j = 0$, если нет. Полученная задача называется задачей о бинарном или 0-1 рюкзаке. При этом предполагается дополнительно, что $\sum_{j=1}^{n} a_j > b$, то есть все предметы в рюкзак упаковать нельзя. Множество D в этой задаче – множество n-мерных булевых векторов с компонентами 0, 1, удовлетворяющих условию. Очевидно, что $D = 2^n$.

2 Общая схема ветвей и границ

Рассматривается задача частично программирования в стандартной форм

Пусть стоит задача, где D – конечное множество. На каждой итерации алгоритма происходит работа с некоторым подмножеством концевых вершин дерева поиска. Это множество называется списком задач-кандидатов или задач для ветвления. Задача решена, если список задач-кандидатов пуст, т.е. если все они отсеяны по правилам отсева.
На начальном шаге процесса список состоит из множества D. Некоторым способом вычисляется значение нижней оценки $\xi(D)$ для целевой функции.

Пусть можно указать план $x' \in D$. Если $f(x) = \xi(D)$, то x' – оптимальное решение задачи. В противном случае полагаем $x_0 = x'$ – рекордное решение, рекорд равен $f_0 = f(x_0)$. Если допустимых планов не найдено, $f_0 = +\infty$. Стандартная (k-я) итерация алгоритма состоит из следующих этапов:

1. Если список кандидатов пуст, прекратить работу, при этом, если рекорд конечен, рекордное решение является оптимальным, в противном случае задача не имеет допустимых решений.

2. Выбрать для ветвления одно из множеств списка.

3. Осуществить ветвление. Модифицировать список.

4. Для каждого подмножества, получившегося в результате ветвления, найти нижние границы $\xi(D_i^k), i = 1, 2, ..., r_k$

5. Если становятся известны допустимые решения (например, если, $D_i^k = \{x\}$ откорректировать сведения о рекорде. Пусть X_k – множество допустимых решений, полученных на k-й итерации. Тогда рекорд равен

$$f_k = min\{f_{k-1}, \min_{x \in X_k} f(x)\} \tag{1}$$

   и рекордное решение x^k есть допустимое решение, на котором достигается минимум в

6. Проверить выполнение правила отсева. Если для некоторого множества $D_i^k = \{x\}$, то исключить его из списка.

Алгоритм ветвей и границ можно остановить до его завершения, используя рекордное решение x_k в качестве приближенного решения задачи. Согласно, относительная погрешность такого решения не превосходит

$\frac{f(x^k)-\xi_0}{\xi_0}$

3 Метод Ленд и Дойг для задачи частично целочисленного линейного программирования

Рассматривается задача частично программирования в стандартной форме:

$$\begin{gathered} f(x) = \sum_{j=1}^{n} c_j x_j \to \min \\ \sum_{j=1}^{n} a_i j x_j \le b_i,\ i = 1, ..., m \\ l_j \le x_j \le d_j,\ j = 1, ..., n \\ x_j \text{ и целые},\ j = 1, ..., n_1, n_1 \le n \end{gathered}$$

Предполагается, $l_j \ge 0,\ j = 1, ..., n$. Если ограничения не заданы изначально, их можно получить, решив 2n задач линейного программирования

$$x_j \to min \text{ и } x_j \to max$$

при условиях и $x_j \ge 0, j = 1, ..., n$ и положив $d_j = \max\lfloor x_j \rfloor$ и $l_j = \min x_j$, если это число целое и $l_j = min\ \lfloor x_j \rfloor\ +1$ в противном случае.
Очевидно, что многогранное множество X, описываемое условиями, ограниченное.

Алгоритм осуществляется по схеме, описанной в предыдущем параграфе. Вершины дерева поиска – это задачи частично целочисленного линейного программирования

$$\begin{gathered} f(x) \to \min, x \in D', \\ D' = \{x = (x_1, ..., x_n) \in X' \subseteq X : z_j \in Z,\ j = 1, ..., n_1\}, \end{gathered}$$

а оценочные задачи для них – соответствующие задачи линейного программирования, получаемые отбрасыванием условий целочисленности.

Если оценочная задача неразрешима, не имеет решений и задача и, следовательно, соответствующая вершина дерева поиска удаляется из списка. Если решение оценочной задачи удовлетворяет условиям целочисленности, то оно является допустимым решением исходной задачи и используется для корректировки сведений о рекорде.

Предположим, на k-й итерации алгоритма ($k = 0, 1, ...$) выбрано для ветвления множество D k (на начальном этапе множество D 0 определяется условиями), $x = (x_1, ..., x_n)$ – решение соответствующей оценочной задачи, и его компонента x_{j_0} , где $1 \le j_0 \le n_1$, не является целой. Тогда множество D^k разбивается на множества D_1^k и D_2^k , первое из них образовано добавлением к ограничениям, задающим D_k , условия $x_{j_0} = \lfloor x_{j_0} \rfloor$, второе – добавлением условия $x_{j_0} = \lfloor x_{j_0} \rfloor + 1$. Если $\lfloor x_{j_0} \rfloor < l_{j_0}$ или $\lfloor x_{j_0} \rfloor \ge d_{j_0}$ то множество D_1^k или D_2^k соответственно пусто.

Если x_{j_0} — булева переменная, то ветвление из вершины D_k соответствует заданию условия $x_{j_0} = 0$ для задачи D_1^k и условия $x_{j_0} = 1$ для задачи D 2 k , то есть, по сути, осуществляется покомпонентное ветвление.

Если нецелочисленных компонент вектора x несколько, ветвление осуществляется, например, по дробной компоненте, имеющей наименьший номер.

4 Метод Ленд и Дойг для задачи о рюкзаке

Ограничения в задачах о рюкзаке имеют вид, где $n_1 = n$ и $l_j = 0, d_j = 1$ для $j = 1, ..., n$, если рассматривается бинарный рюкзак. Поэтому может применяться рассмотренный в предыдущем параграфе алгоритм, соответствующим образом модифицированный для решения задачи на максимум.

Конкретизируем элементы метода Ленд и Дойг для задачи о бинарном ранце

$$f(x) = \sum_{j=1}^{n} c_j x_j \to \max$$
$$\sum_{j=1}^{n} a_j x_j \leq b$$
$$x_j \in \{0, 1\}; j = 1, ..., n$$
$$c_j > 0,\ 0 \leq a_j \leq b,\ j = 1, ...n$$

При ветвлении значение одной из переменных полагается равным 0 или 1, поэтому подзадачи также имеют вид.

Оценочной для задачи является задача линейного программирования

$$f(x) = \sum_{j=1}^{n} c_j x_j \to \max$$
$$\sum_{j=1}^{n} a_j x_j \leq b$$
$$0 \leq x_j \leq 1,\ j = 1, ...n$$

Алгоритм поиска оптимального решения этой задачи сформулирован в следующей теореме.

Теорема (правило Данцига). Пусть переменные $x_j, 1 \leq j \leq n$ перенумерованы так, что $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$. где $\lambda_n = \frac{c_j}{a_j}$ Тогда оптимальное решение $\overline{x} = (\overline{x_1}, ..., \overline{x_n})$ задачи имеет вид

$$\overline{x_1} = \overline{x_2} = ... = \overline{x_{s-1}} = 1$$
$$\overline{x_s} = \frac{b - \sum_{j=1}^{s-1} a_j}{a_s}$$
$$\overline{x_{s+1}} = \overline{x_{s+2}} = ... = \overline{x_n} = 0$$

где s определяется из условия

$$\sum_{j=1}^{s-1} a_j \leq b \leq \sum_{j=1}^{s} a_j$$

Если вектор, полученный по правилу Данцига, целочисленный, то он является оптимальным решением соответствующей задачи. В противном случае, при $0 < x_s < 1$, можно получить допустимое решение x' задачи, обнулив s-ю компоненту вектора x , тогда

$$f(\overline{x}) - f(x') = c_s \overline{x_s} = \lambda_s (b - \sum_{j=1}^{s-1} a_j)$$

В соответствии с замечанием, если в задаче о рюкзаке все с j целые, множество D^k может быть отсеяно, если

$$0 \leq \xi(D^k) - f(x') < 1$$

где x' — рекордное решение. Задача решена, если $= max\ \xi(D^k) - f(x') < 1$, при этом x' – оптимальное решение.