Лекция 10

ОСНОВЫ ЦЕЛОЧИСЛЕННОГО ПРОГРАММИРОВАНИЯ

Под задачей целочисленного программирования (ЦП) понимается задача, в которой все или некоторые переменные должны принимать целые значения. В том случае, когда ограничения и целевая функция задачи представляют собой линейные зависимости, задачу называют *целочисленной задачей линейного программирования.* В противном случае, когда хотя бы одна зависимость будет нелинейной, это будет *целочисленной задачей нелинейного программирования.*

Особый интерес к задачам ЦП вызван тем, что во многих практических задачах необходимо находить целочисленное решение ввиду дискретности ряда значений искомых переменных. К их числу относятся, например, следующие задачи:

- задачи оптимизации раскроя;
- оптимальное проектирование машин и оборудования;
- оптимизации системы сервиса и технического обслуживания машинно-тракторного парка;
- и т.д.

Как уже отмечалось, часто задачу ЦП решают без учета условий целочисленности переменных, а затем округляют полученное решение с избытком или недостатком. Это не гарантирует получение оптимального целочисленного решения задачи. Поэтому для нахождения оптимального решения целочисленных задач применяют специальные методы, в которых учитывается, что число возможных решений любой целочисленной задачи является конечным. Следовательно, можно рассмотреть все возможные сочетания целочисленных переменных и проверить, удовлетворяют ли они ограничениям, и из числа удовлетворяющих ограничениям, выбрать наилучшее с точки зрения целевой функции. Такой метод называют *методом полного перебора*. Его трудоемкость с ростом числа переменных и расширением области граничных условий значительно возрастает. Поэтому для реальных задач он неприменим.

На практике для решения реальных задач следует использовать методы, в котором все возможные альтернативы не рассматриваются. Наиболее распространенным является *метод ветвей и границ.*

Метод ветвей и границ

Алгоритм метода ветвей и границ представляет собой эффективную процедуру перебора всех целочисленных допустимых решений.

1. Решаем сформулированную задачу как задачу ЛП (обозначим ЛП-1), рассматривая все ее переменные как непрерывные. Пусть W_1 - оптимальное значение ее целевой функции. Допустим в оптимальном решении ЛП-1 некоторые целочисленные переменные принимают дробные значения. Тогда оптимальное решение исходной задачи не совпадает с оптимальным решением ЛП-1. В этом случае W_1 - *верхняя граница* оптимального значения W исходной задачи.

Пример 1. Линейное программирование.

Лесничество имеет 24 га свободной земли под паром и заинтересовано извлечь из нее доход. Оно может выращивать саженцы быстрорастущего гибрида новогодней ели, которые достигают спелости за один год, или бычков, отведя часть земли под пастбище. Деревья выращиваются и продаются в партиях по 1000 штук. Требуется 1.5 га для выращивания одной партии деревьев и 4 га для вскармливания одного бычка. Лесничество может потратить только 200 ч. в год на свое побочное производство. Практика показывает, что требуется 20 ч. для культивации, подрезания, вырубки и пакетирования одной партии деревьев. Для ухода за одним бычком также требуется 20 ч. Лесничество имеет возможность израсходовать на эти цели 6 тыс. руб. Годовые издержки на одну партию деревьев выливаются в 150 руб. и 1,2 тыс. руб. на одного бычка. Уже заключен контракт на поставку 2 бычков. По сложившимся ценам, одна новогодняя ель принесет чистый доход в 2,5 руб., один бычок - 5 тыс. руб.

Постановка задачи.

1. В качестве *показателя эффективности* целесообразно взять доход за операцию (годовой чистый доход с земли в рублях).
2. В качестве управляемых *переменных* задачи следует взять:

   x_1 - количество откармливаемых бычков в год;

   x_2 - количество выращиваемых партий быстрорастущих новогодних елей по 1000 шт. каждая в год.
3. *Целевая функция*:

$$5000\, x_1 + 2500\, x_2\ (\rho)\ \max,$$

где

   5000- чистый доход от одного бычка, руб.;

   2500 - чистый доход от одной партии деревьев (1000 шт. по 2,5 руб.).

4. *Ограничения*:

4.1. По использованию земли, га:

$$4\, x_1 + 1{,}5\, x_2 \le 24.$$

4.2. По бюджету, руб.:

$$1200\, x_1 + 150\, x_2 \le 6000.$$

4.3. По трудовым ресурсам, ч:

$$20\, x_1 + 20\, x_2 \le 200.$$

4.4. Обязательства по контракту, шт.:

$$x_1 \ge 2.$$

4.5. Областные ограничения:

$$x_1 \ge 0,\ x_2 \ge 0.$$

Пример 2.Целочисленное программирование.

В решенном нами Примере 1 верхняя граница соответствует 34 млн. руб. при x1=3.6 и x2=6.4. Очевидно, что переменные x1 (число откармливаемых бычков) и x2 (количество выращиваемых партий быстрорастущих новогодних елей) должны быть целочислены.

2. Производим ветвление по одной из целочисленных переменных, имеющих дробное значение в оптимальном решении ЛП-1. Выбор переменной, по котором производим ветвление, осуществляется по ряду правил:

- выбор целочисленной переменной, значение которой в оптимальном решении ЛП-1 имеет наибольшее дробное значение;
- расстановка приоритетов и ветвление по переменным с наибольшим приоритетом (данная переменная представляет важное решение, ее коэффициент стоимости или прибыли в целевой функции существенно превосходит остальные и т.д.);
- Произвольные правила выбора (например, переменную с наименьшим порядковым номером).

В рассматриваемой целочисленной версии примера 1 в качестве переменной, по которой будет вестись ветвление, следует рассматривать x_1, т.к. коэффициент стоимости при этой переменной максимален.

3. Пусть ветвление происходит по x_i, дробное значение которой в ЛП-1 равно β_i. Далее рассматриваются уже две новые задачи ЛП-2 и ЛП-3, которые образуются путем введения двух дополнительных ограничений $x_i \le \alpha_i$, $x_i \ge \gamma_i$, где α_i - наибольшее целое, не превосходящее β_i, γ_i - наименьшее целое, большее β_i. Допустим, оптимальные значения ЛП-2 и ЛП-3 содержат дробные значения целочисленных переменных и поэтому не являются допустимыми.

Для рассматриваемого примера β_1=3,6, α_1=3, γ_1=4.

4. Производим ветвление в вершине 2 или 3, вводя новое ограничение. Выбор вершины (задачи ЛП) для дальнейшего ветвления осуществляется с помощью специальных правил:

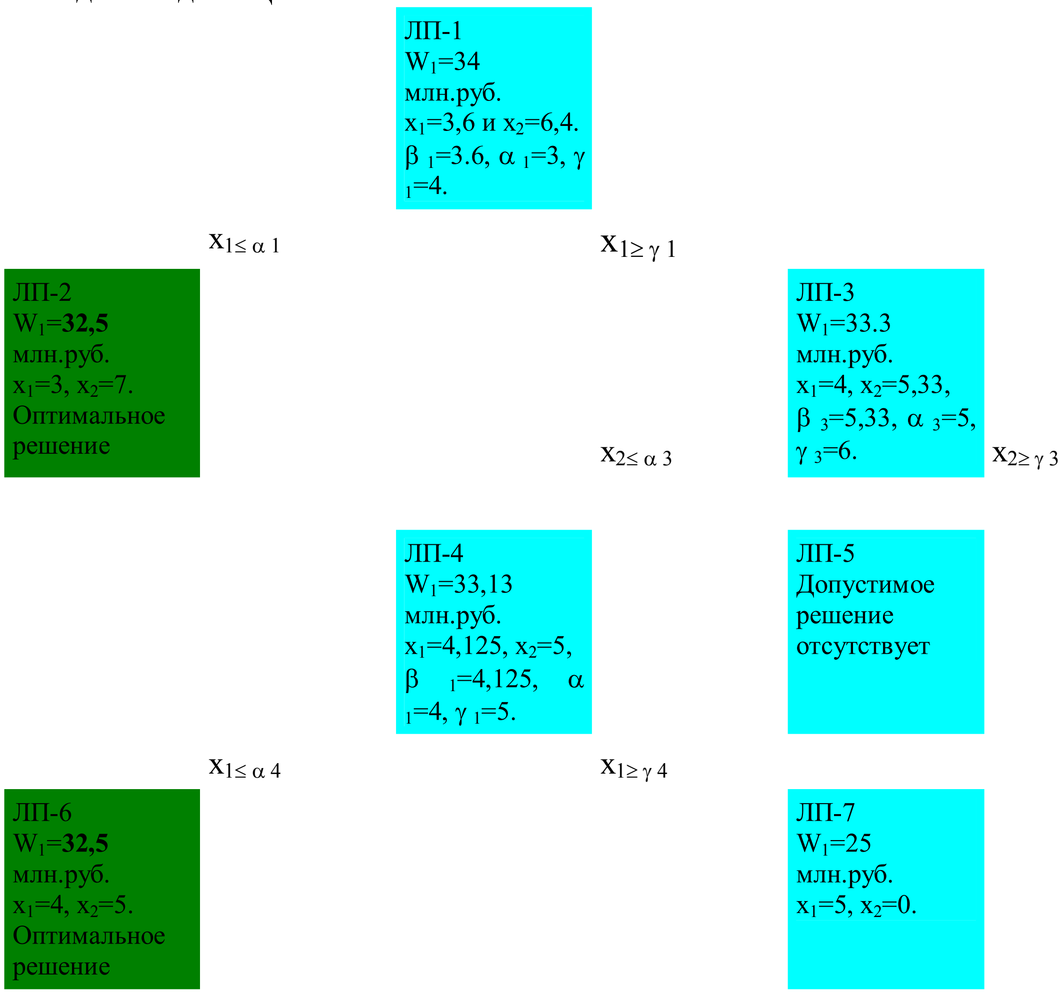
- использование наибольшего оптимального значения целевой функции;
- "последним пришел - первым вышел".

5. Процесс ветвления и решения задач ЛП продолжаем до получения целочисленного решения одной из задач ЛП. Значение W в полученной точке

представляет собой *нижнюю границу* оптимального значения ЦФ исходной задачи ЦП. На этом этапе все вершины (задачи ЛП), для которых оптимальное значение W не превосходит полученной нижней границы ("прозондированные" вершины). Вершина является прозондированной в следующих случаях:

- оптимальное решение, соответствующее данной вершине, целочислено;
- задача ЛП не имеет допускаемого решения;
- оптимальное значение W не превосходит текущей нижней границе.

Ветвление происходит до тех пор, пока остается хотя бы одна непрозондированная вершина. Прозондированная вершина с наилучшим W дает оптимальное решение исходной задачи ЦП.



Процесс ветвления и решения рассматриваемой задачи представлен на рис.4.1. Оптимальное решение - W=32,5 млн. руб.; x_1=4; x_2=5.

Рекомендации по формулировке и решению ЦП

1. Количество целочисленных переменных уменьшать насколько возможно. Например, целочисленные переменные, значения которых должно быть не менее 20, можно рассматривать как непрерывные.
2. В отличие от общих задач ЛП, добавление новых ограничений особенно включающих целочисленные переменные, обычно уменьшают время решения задач ЦП.
3. Если нет острой необходимости в нахождении точного оптимального целочисленного решения, отличающегося от непрерывного решения, например, 3%. Тогда реализацию метода ветвей и границ для задачи максимизации можно заканчивать, если отношение разницы между верхней и нижней границ к верхней границы меньше 0,03.

Метод ветвей и границ можно применять для решения задач нелинейного программирования.

Задачи оптимизации раскроя

В качестве примера рассмотрим задачу раскроя сырья. Поэтому экономное расходование сырья является одной из важнейших задач, подлежащих решению при планировании, организации и управлении производственными процессами. Эта задача приобретает еще большую актуальность в связи с экологическими проблемами, вопросами охраны окружающей среды.

При построении модели оптимизации раскроя обычно исходят из следующей ситуации. Имеется предприятие, на которое поступает определенное количество сырья (листовые, брусковые и т.д.). Предприятие выпускает продукцию, для изготовления которой исходное сырье подлежит раскрою на заготовки данной конфигурации и размеров.

Знакомство с задачами раскроя начнем с линейного раскроя.

Пример 3. Из 50 шт. бревен длиной 6,5 м необходимо изготовить наибольшее число комплектов, в состав каждого из которых входит 2 шт. детали длиной 2 м и 3 шт. детали длиной 1,25 м.

Есть два подхода к решению задач раскроя:

1. Раскрой делает ЭВМ.
1. Лицо принимающее решение (ЛПР) принимает несколько различных вариантов раскроя, а ЭВМ выбирает лучший из них.

Подход 1. Раскрой делает ЭВМ

Постановка задачи.

1. В качестве *показателя эффективности* целесообразно взять число комплектов K, которое можно получить из заданного числа заготовок. Возможна другая постановка - качестве показателя эффективности взять число заготовок Z, которое необходимо иметь, чтобы получить заданное цисло комплектов.
1. В качестве управляемых *переменных* задачи следует взять x_A и x_B - соответственно, число деталей A и B, получаемых из одной заготовки; каждая в год.

3. *Целевая функция*:

$W_1 = K(\rho)$ max;

или

$W_2 = Z(\rho)$ min.

4. *Ограничения*:

4.1. По длине заготовки, м:

$2x_A + 1{,}25x_B \leq 6{,}5.$

4.2. По комплектности, шт.:

$k x_A - 2K \geq 0;$

$k x_B - 3K \geq 0,$

где

k - число заготовок.

4.3. Областные ограничения и требования целочисленности:

$x_A \geq 0,\ x_B \geq 0,\ K \geq 0$ - целые.

При решении задачи в первой постановке (k=50)

$$\begin{cases} W(x) \to \min(\max); \\ g_j(x) \le 0, j = 1, \ldots, J; \\ h_k(x) = 0, k = 1, \ldots, K; \\ x_i^L \le x_i \le x_i^U, i = 1, 2, \ldots, N \end{cases}$$

на ЭВМ получено: x_A=2; x_B=2; K=33. Это означает, что если из одной заготовки выкраивать две 2 м детали и две 1,25 м детали, то максимальное количество комплектов будет 33.

Подход 2. ЛПР принимает несколько различных вариантов раскроя, а ЭВМ выбирает лучший из них

Сырье может раскраиваться на заготовки различными способами - вариантами (картами) раскроя, которые сводятся в специальную таблицу. В этой таблице указывают требуемое число заготовок каждого вида и величина отходов для каждого варианта раскроя. Допустим, в нашем примере ЛПР разработал 4 варианта, которые приведены в табл. 2. Отметим, что вариант предложенный ЭВМ в первом подходе присутствует в таблице как одна из альтернатив.

Таблица 4.1.

Длина заготовки, м	Варианты раскроя				Число деталей в комплекте
	1	2	3	4	
2	3	2	1	0	2
1,25	0	2	3	5	3
Отходы	0,5	0	0,75	0,25	-
Число заготовок	x_1	x_2	x_3	x_4	-

Постановка задачи.

1. В качестве *показателя эффективности* целесообразно взять число комплектов K, которое можно получить из заданного числа заготовок. Возможны другие постановки - взять число заготовок Z, которое необходимо иметь, чтобы получить заданное число комплектов или отходы О.
1. В качестве управляемых *переменных* задачи следует взять:

x_1 - число заготовок, раскраиваемых по 1 варианту;
x_2 - число заготовок, раскраиваемых по 2 варианту;
x_3 - число заготовок, раскраиваемых по 3 варианту;
x_4 - число заготовок, раскраиваемых по 4 варианту.

3. *Целевая функция*:

W_1 = K(ρ) max;
или
W_2 = Z (ρ) min;
или
W_3 = O = 0,5 x_1 + 0 x_2 + 0,75 x_3 + 0,25 x_4 (ρ) min.

4. *Ограничения*:

4.1. По числу деталей, получаемых в результате раскроя по всем вариантам:
$x_1 + x_2 + x_3 + x_4 = k$,
где

k - число заготовок.

4.2. По комплектности, шт.:
$3 x_1 + 2 x_2 + x_3 - 2 K \ge 0$;
$2 x_2 + 3 x_3 + 5 x_4 - 3 K \ge 0$;
4.3. Областные ограничения и требования целочисленности:
$x_A \ge 0$, $x_B \ge 0$, $K \ge 0$ - целые.

При решении задачи в первой постановке (k=50)

$$\begin{cases} W_1 = K \to \max; \\ x_1 + x_2 + x_3 + x_4 = 50; \\ 3 \cdot x_1 + 2 \cdot x_2 + x_3 - 2 \cdot K \ge 0; \\ 2 \cdot x_2 + 3 \cdot x_3 + 5 \cdot x_4 - 3 \cdot K \ge 0; \\ x_1 \ge 0, \ldots, x_4 \ge 0, K \ge 0 \text{ - целые,} \end{cases}$$

на ЭВМ получен результат, представленный в табл. 4.2.

Таблица 4.2.

Переменная	Решения				
	1	2	3	4	5
x_1	0	0	0	1	0
x_2	41	41	42	40	40
x_3	0	1	0	1	2
x_4	9	8	8	8	8
K	41	41	41	41	41

Из таблицы 4.2 видно, что существует пять равноценных варианта раскроя, которые приводят к получению 41 комплекта из 50 заготовок. Если данный результат сравнить с результатом, полученным по первому подходу (33 комплекта из тех же самых 50 заготовок), то получаем выигрыш в 8 комплектов. Следовательно, для лучшего использования сырья целесообразно использовать сочетание различных вариантов раскроя.

Подход, при котором выбирается из ряда разработанных ЛПР вариантов раскроя наилучшее их сочетание, имеет широкое применение в практике раскроя не только линейного, но и плоскостного.

4.5. Постановка задачи дискретного программирования

Под задачей дискретного программирования (ДП) понимается целочисленная задача, в которой все или некоторые переменные должны принимать не любые целые значения.

Особый интерес к задачам ДП вызван тем, что во многих практических задачах необходимо находить целочисленное решение ввиду дискретности ряда значений искомых переменных. Например, мебельная фабрика выпускает диваны, кресла и стулья. Требуется определить, сколько можно изготовить спинок диванов, подлокотников кресел и ножек стульев, если для их изготовления требуются заданные ресурсы, чтобы доход был максимален. При этом следует иметь ввиду, что выпуск спинок может принимать любое целое значение, подлокотники изготавливаются парами, а ножки должны быть кратными четырем.

Рассмотренную задачу распределения ресурсов с учетом требования дискретного значения переменных в общем виде можно записать так:

$$\begin{cases} W = \sum_{i=1}^{n} c_i x_i \to \max(\min); \\ \sum_{i=1}^{n} a_{ji} \le b_j (j = 1, 2, ..., m); \\ x_i = \sum_{k=1}^{k_i} d_{jk} \delta_{jk} (i = 1, 2 ..., n); \\ \sum_{k=1}^{k_i} \delta_{jk} = 1, \end{cases}$$

где

$d_{i1}, d_{i2}, ..., d_{ik}$ - дискретные значения, которые может принимать переменная x_i.

$$\delta_i = \begin{cases} 1, \text{если i вариант принят;} \\ 0, \text{если i вариант не принят.} \end{cases}$$

Данная постановка отличается от задачи распределения ресурсов линейного программирования

$$\begin{cases} W = \sum_{i=1}^{n} c_i x_i \to \max(\min); \\ \sum_{i=1}^{n} a_{ji} x_i \le b_j (j = 1,2,\ldots,m); \\ d_i \le x_i \le D_i; \\ i = 1,2,\ldots,n, \end{cases}$$

появлением булевых переменных и увеличением числа ограничений. На практике к задачам с булевыми переменными можно свести значительное число самых различных задач. Там, где есть варианты, из которых надо выбирать, задачу можно решать с помощью булевых переменных.

Выделяют два метода решения задач с булевыми переменными:

1. *Метод полного перебора*. Алгоритм метода заключается в следующем:

- в специальной таблице заполняются все варианты сочетаний значений $\delta_1, \delta_2, \ldots, \delta_k$;
- определяются значения левых частей ограничений и целевой функции и записываются в таблицу;
- вычеркиваются варианты, в которых нарушается по крайней мере одно ограничение;
- из оставшихся вариантов принимается тот, в котором целевая функция принимает максимальное (минимальное) значение.

Решаются как обычные задачи *целочисленного программирования*, т.е. методом ветвей и границ. При этом на каждую переменную накладывается два ограничения: они должны быть в пределах $0 \le \delta_i \le 1$; δ_i должны быть целыми.

Использованная и рекомендуемая литература:

1. Исследование операций в экономике: Учеб. пособие для вузов /Н.Ш. Кремер, Б А Путко, И.М. Тришин, М.Н. Фридман; Под ред. проф. Кремера. - М: ЮНИТИ, 2002. - 407 с.
2. Зайченко Ю.П. Дослідження операцій. [Текст]: – К. Видавничий дім «Слово». – 2006. – 816 с
3. Таха Х.А. [Текст]: Введение в исследование операций, 7-е изд. – М.: Издательский дом «Вильямс», 2005. – 912 с.