

Федеральное агентство по образованию  
Государственное образовательное учреждение  
высшего профессионального образования  
**ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ УПРАВЛЕНИЯ**

Институт информационных систем управления  
кафедра экономической кибернетики

**ЦЕЛЕВОЕ ПРОГРАММИРОВАНИЕ  
В СИСТЕМЕ Linear Program Solver**

Москва 2010

## **Содержание**

1. Краткий обзор методов целевого программирования.....	3
3. Тестирование методов ЦП, реализованных в системе LiPS.....	8

## 1. Краткий обзор методов целевого программирования

Целевое программирование (ЦП), зародившись первоначально в качестве приложения обычного линейного программирования в работах Чарнса и Купера, приобрело популярность в 60-е и 70-е годы. В настоящее время ЦП - это важная область многокритериальной оптимизации.

Идея ЦП заключается в том, чтобы установить некоторый уровень достижения целей по каждому критерию. Целевое программирование отличается от обычного линейного программирования следующими особенностями:

1. Концептуализацией (пониманием) критериев как целей.
2. Приписыванием приоритетов и/или весов достижению отдельных целей.
3. Присутствием переменных-невязок, являющихся мерой отклонения от целевых (или пороговых) уровней сверху и снизу.
4. Минимизацией взвешенных сумм переменных отклонений с целью найти решения, наилучшим образом удовлетворяющие целям.

Обычно точка, удовлетворяющая сразу всем целям, не является допустимой. Поэтому мы стараемся найти допустимую точку, которая достигает всех целей «как можно лучше». Методы отыскания таких точек, использующие конструкции, связанные с приоритетами и/или весами, и составляют предмет целевого программирования.

В целевом программировании рассматриваются два основных подхода к решению задач: *весовая модель* и *модель с приоритетами*.

### 1.1. Метод весовых коэффициентов

Особенности весовой модели:

1. Целевая функция представляет собой взвешенную сумму переменных нежелательных отклонений.
2. Каждая цель порождает одно целевое ограничение, кроме случая задания диапазона, когда возникает два целевых ограничения.
3. В формулировке нужно использовать только переменные отклонений (невязки), которые соответствуют нежелательным отклонениям.

4. Задачи целевого программирования с весами можно решать, используя обычные методы ЛП.

Рассмотрим задачу ЦП:

$$\text{Цель } \{c^1 x = z_1\}, z_1 \geq t_1;$$

$$\text{Цель } \{c^2 x = z_2\}, z_2 = t_2;$$

$$\text{Цель } \{c^3 x = z_3\}, z_3 \leq t_3;$$

$$\text{при } x \in S$$

Формулировка задачи в форме весовой модели имеет следующий вид:

$$\begin{aligned} & \min \{w_1 \cdot d_1 + w_2 \cdot d_2^+ + w_2 \cdot d_2^- + w_3 \cdot d_3\} \\ & \left. \begin{aligned} & c^1 x + d_1 \geq t_1; \\ & c^2 x - d_2^+ + d_2^- = t_2; \\ & c^3 x - d_3 \leq t_3; \end{aligned} \right\} \text{целевые ограничения} \\ & \text{при } x \in S \end{aligned}$$

Задача, записанная в таком виде, может быть решена стандартным пакетом ЛП.

## 1.2. Метод назначения приоритетов

В приоритетном (лексикографическом) целевом программировании цели группируются по приоритетам. Цели с высшим (первым) уровнем приоритета считаются бесконечно важными по сравнению с целями со следующим (вторым) уровнем приоритета, а цели со вторым уровнем приоритета — бесконечно важными по сравнению с целями с третьим уровнем приоритета, и т. д.

Рассмотрим задачу ЦП с приоритетами:

$$\text{Цель } \{c^1 x = z_1\}, P_1(z_1 \leq t_1);$$

$$\text{Цель } \{c^2 x = z_2\}, P_2(z_2 \geq t_2);$$

$$\text{Цель } \{c^3 x = z_3\}, P_3(z_3 = t_3).$$

$$\text{при } x \in S$$

где  $P_j$  указывает цели с уровнем приоритета  $j$ .

Перепишем задачу в лексикографической форме:

$$\begin{aligned} & \text{lex min } \{d_1, d_2, (d_3^+ + d_3^-)\} \\ & \left. \begin{aligned} & c^1 x - d_1 \leq t_1; \\ & c^2 x + d_2 \geq t_2; \\ & c^3 x - d_3^+ + d_3^- = t_3; \end{aligned} \right\} \text{целевые ограничения} \\ & \text{при } x \in S \end{aligned}$$

Для решения этой задачи с помощью обычных систем ЛП могут потребоваться целых три этапа оптимизации.

На первом этапе решаем задачу:

$$\begin{aligned} & \min \{d_1\} \\ & c^1 x - d_1 \leq t_1; \\ & \text{при } x \in S \end{aligned}$$

Если в этой задаче есть альтернативные оптимумы, то мы формулируем и решаем задачу второго этапа:

$$\begin{aligned} & \min \{d_2\} \\ & c^1 x \leq t_1 + d_1^*; \\ & c^2 x + d_2 \geq t_2; \\ & \text{при } x \in S \end{aligned}$$

где  $d_1^*$  - оптимальное значение переменной  $d_1$ , найденное на первом этапе. Если в задаче второго этапа есть альтернативные оптимумы, то формулируем и решаем задачу третьего этапа:

$$\begin{aligned} & \min \{d_3^+ + d_3^-\} \\ & c^1 x \leq t_1 + d_1^*; \\ & c^2 x \geq t_2 - d_2^*; \\ & c^3 x - d_3^+ + d_3^- = t_3; \\ & \text{при } x \in S \end{aligned}$$

Любое решение задачи третьего этапа определяет лексикографический минимум в задаче ЦП с приоритетами.

Может случиться, что нам не нужно будет решать столько задач оптимизации, сколько имеется уровней приоритета. Число этапов может быть сокращено, как только на каком-то этапе нам встретится единственное решение. Таким образом, нежелательное следствие приоритетного подхода состо-

ит в том, что цели низших уровней могут и не иметь шанса влиять на финальное решение.

Целевое программирование с приоритетами часто критиковалось за его негибкость, состоящую в том, что высшие уровни приоритета не «терпят» никаких компромиссов с нижними уровнями. Одна из возможностей избежать этой негибкости — использовать *метод последовательных уступок*.

### 1.3. Метод последовательных уступок

Этот метод вносит элемент интерактивности в процесс решения задач ЦП с приоритетами. В качестве примера воспользуемся ранее рассмотренной задачей. После решения ЛП задачи первого приоритета требуется обновить целевое ограничение:

$$c^1 x \leq t_1 + d_1^* + \Delta_1$$

где  $d_1^*$  - оптимальное значение переменной  $d_1$ , найденное на первом этапе;  $\Delta_1$  - уступка, определяемая ЛПР.

Очевидно, что успешность применения такого подхода зависит от правильности выбора величин  $\Delta$ , которые назначаются пользователем перед очередным этапом решения задачи ЦП.

### 1.4. Минимаксное целевое программирование

Если какой-то уровень приоритета включает отклонения от нескольких целей, то один из возможных способов постановки задачи - *минимизация максимального отклонения*, что требует введения специальной минимаксной переменной  $\alpha$ . Критерий на рассматриваемом уровне приоритета приобретает вид  $\min\{\alpha\}$ . При этом нужно выписать столько дополнительных ограничений, сколько имеется переменных отклонения на рассматриваемом уровне приоритета.

Рассмотрим задачу ЦП с приоритетами:

$$\begin{aligned}
&\text{Цель } \{c^1 x = z_1\}, P_1(z_1 = t_1); \\
&\text{Цель } \{c^2 x = z_2\}, P_2(z_2 \geq t_2); \\
&\text{Цель } \{c^3 x = z_3\}, P_2(z_3 \geq t_3); \\
&\text{Цель } \{c^4 x = z_4\}, P_2(z_4 \geq t_3). \\
&\text{при } x \in S
\end{aligned}$$

На первом этапе минимизируются отклонения для первого уровня приоритетов:

$$\begin{aligned}
&\min\{d_3^+ + d_3^-\} \\
&c^3 x - d_3^+ + d_3^- = t_3; \\
&\text{при } x \in S
\end{aligned}$$

Далее на втором этапе, составляется и решается задача минимизации максимального отклонения:

$$\begin{aligned}
&\min\{\alpha\} \\
&c^1 x = t_1 + (d_1^+)^* - (d_1^-)^*; \\
&c^2 x + d_2 \geq t_2; \\
&c^3 x + d_3 \geq t_3; \\
&c^4 x + d_4 \geq t_4; \\
&d_2 \leq \alpha \\
&d_3 \leq \alpha \\
&d_4 \leq \alpha \\
&\text{при } x \in S
\end{aligned}$$

В этой формулировке  $\alpha$  - минимаксная переменная, которая в процессе минимизации на втором уровне приоритета минимизирует наибольшее из отклонений  $d_2$ ,  $d_3$  или  $d_4$ .

### 1.5. Масштабирование целевых показателей

На решение задач целевого программирования большое влияние оказывают масштабы, в которых измеряются целевые показатели. Если эти масштабы для разных целей существенно отличаются, это может значительно исказить решение задачи ЦП. Для того чтобы избежать подобных негативных явлений, применяются методы шкалирования, предназначенные для приведения «пространства целей» к единому масштабу измерения. Простым и в

то же время эффективным методом масштабирования является так называемое «сбалансированное шкалирование» (equilibration scaling).

Пусть  $C$  – матрица, строками которой являются коэффициенты целевых ограничений;  $g$  – вектор столбец, составленный из правых частей целевых ограничений. При применении сбалансированного шкалирования, каждая строка масштабируется таким образом, чтобы наибольший по модулю коэффициент этой строки был равен 1. Для этого каждую строку нужно умножить на величину, обратную максимальному по модулю элементу данной строки.

$$\begin{aligned}\hat{C} &= R \cdot C; \\ \hat{g} &= R \cdot g\end{aligned}$$

$$R = \begin{pmatrix} \frac{1}{r_1} & & & & \\ & \ddots & & & \\ & & \frac{1}{r_i} & & \\ & & & \ddots & \\ & & & & \frac{1}{r_n} \end{pmatrix}$$

$$r_i = \max_j \{C_{i,j}\}$$

## 2. Тестирование методов целевого программирования, реализованных в рамках оптимизационной среды LiPS

На простом примере мы проиллюстрируем использование методов целевого программирования на базе оптимизационной среды LiPS.

*Формулировка задачи.* Новое рекламное агентство, в котором работают 10 рекламных агентов, получило контракт на рекламу нового продукта. Агентство может провести рекламную акцию на радио и телевидении. В следующей таблице приведены данные о количестве людей, охватываемых тем или иным видом рекламы, стоимость этой рекламы и количество необходимых рекламных агентов. Все данные отнесены к одной минуте рекламного времени.



	Радио	Телевидение
Рекламная аудитория (млн. чел.)	4	8
Стоимость (тыс. долл.)	8	24
Количество рекламных агентов	1	2

Реклама на радио и телевидении должна охватить не менее 45 миллионов человек (рекламная аудитория), но контракт запрещает использовать более 6 минут рекламы на радио. Рекламное агентство может выделить на этот проект бюджет, не превышающий 100 000 долл. Сколько минут рекламного времени агентство должно купить на радио и сколько на телевидении?

Обозначим через  $x_1$  и  $x_2$  количество минут рекламного времени, купленного соответственно на радио и телевидении, и составим следующую модель ЦП:

$$\begin{aligned} \max \{ & 4 \cdot x_1 + 8 \cdot x_2 \geq 45 \} \quad (\text{условие по рекламной аудитории}) \\ \min \{ & 8 \cdot x_1 + 24 \cdot x_2 \leq 100 \} \quad (\text{условие по бюджету}) \\ & x_1 + 2 \cdot x_2 \leq 10; \quad (\text{ограничение по рекламным агентам}) \\ & x_1 \leq 6. \quad (\text{ограничение на рекламу по радио}) \end{aligned}$$

Для решения задачи целевого программирования в системе LiPS необходимо выполнить последовательность шагов:

- 1) Создать модель в табличной или текстовой форме;
- 2) Запустить мастер решения моделей ЦП;
- 3) Выбрать метод решения;
- 4) Провести спецификацию целей

Рассмотрим каждый из перечисленных этапов подробнее:

1) Для создания модели в табличной форме следует воспользоваться меню *Файл – Создать – Табличную модель*.

В появившемся окне необходимо заполнить параметры задачи: число переменных, ограничений и целевых функций. В рассматриваемом нами примере имеется 2 переменных, 2 ограничения и 2 целевых функции.

Параметры задачи

Параметры задачи

Число переменных: 2

Число ограничений: 2

Число целевых функций: 2

Направление оптимизации

☒ Максимизация

☐ Минимизация

OK Cancel

Далее необходимо ввести модель задачи в табличной форме:

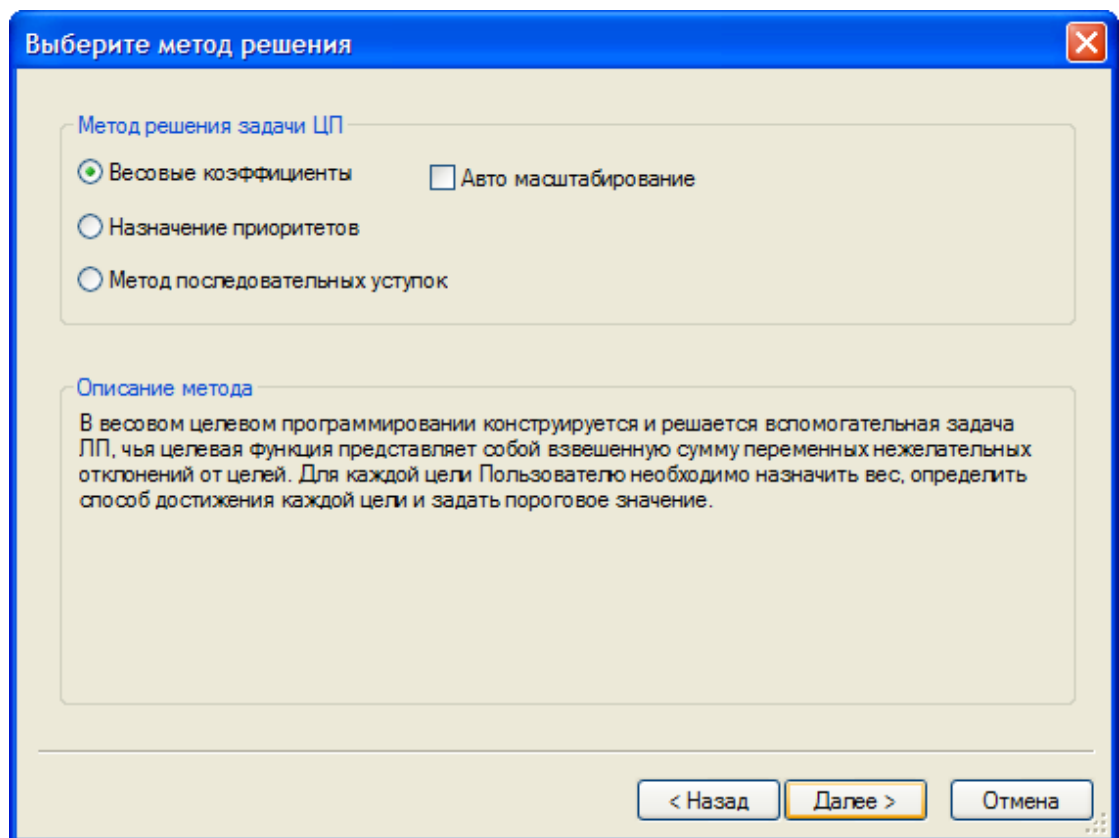
	x1	x2		RHS
Аудитория	4	8	->	MAX
Бюджет	8	24	->	MIN
Ограничение 1	1	2	<=	10
Ограничение 2	1		<=	6

Для создания модели в текстовой форме следует воспользоваться меню *Файл – Создать – Текстовую модель*. Далее необходимо ввести модель, соблюдая требования формата данных LPX, описанного в документации к системе LiPS. Наша модель в формате LPX имеет вид:

```
max: 4*x1 + 8*x2;
min: 8*x1 + 24*x2;

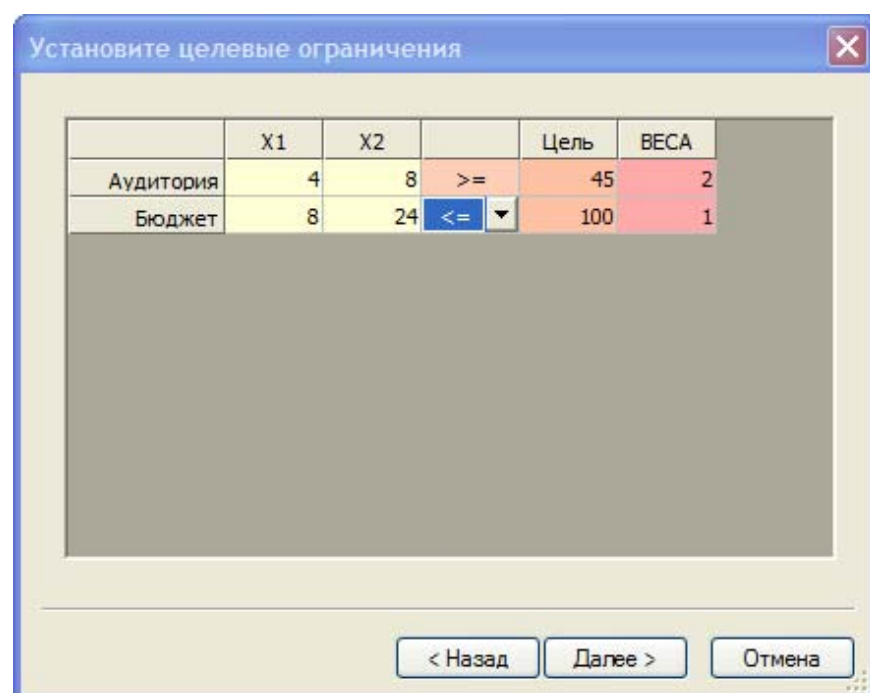
c1: x1 + 2*x2 <= 10;
c2: x1 <= 6;
```

3) Запустить мастер решения моделей ЦП: меню *LiPS – Решить модель*.



4) В появившемся окне можно выбрать метод решения задач целевого программирования и указать программе провести автоматическое масштабирование матрицы целевых показателей.

5) Выбрав метод весовых коэффициентов и нажав Далее, мы перейдем к следующему этапу, на котором для каждой цели необходимо назначить вес (приоритет), определить способ ее достижения и задать пороговое значение.



6) Нажимая кнопку Далее, запускается процесс решения, по окончании которого выдается ответ.

Результаты

>> Оптимальное решение НАЙДЕНО

\*\*\* РЕЗУЛЬТАТЫ \*\*\*

переменная	Значение
x1	5
x2	2.5

Название	цель	Факт	Отклонение	Вес
Аудитория	45	40	5	2
Бюджет	100	100	0	1

\*\*\* COST Range \*\*\*

Variable	Current COST	Min COST	Max COST	Reduced Cost
x1	0	0	1/6	0
x2	0	-1/3	0	0
delta_1	2	0	+infinity	0
delta_2	1	0	+infinity	-1

< Назад    Готово    Отмена

При решении задачи методом весовых коэффициентов в качестве ответа выводится 3 таблицы: в первой представлены переменные и их оптимальные значения; во второй – фактические значения целевых показателей и их отклонения от заданных пороговых значений.

Используя кнопку **Назад** всегда можно вернуться к предыдущим экранам и поэкспериментировать с настройками.

При использовании метода последовательных уступок процесс решения состоит из последовательности шагов, в ходе которых пользователю будет предложено вводить уступки после оптимизации соответствующих целей.

Например, в случае нашей задачи будет сначала выведена таблица промежуточных результатов по оптимизации цели первого приоритета (*Аудитория*)

Результаты

>> Оптимизация целей приоритета < 1 >  
 >> Оптимальное решение НАЙДЕНО

\*\*\* ЦЕЛИ \*\*\*

Название	Цель	Факт	Отклонение	Приоритет
Goal1	45	40	5	< 1 >
Goal2	100	120	20	2

\*\*\* ПЕРЕМЕННЫЕ \*\*\*

переменная	Значение
x1	0
x2	5

< Назад    Далее >    Отмена

После нажатия кнопки Далее пользователю предлагается ввести значение уступки для цели первого приоритета:

Установите значения уступок

	x1	x2		Значе...	Уступка
Goal1	4	8	=	40	3

< Назад    Далее >    Отмена

Нажимая Далее, мы получаем окончательный ответ:

Результаты

>> Оптимизация целей приоритета < 2 >  
>> Оптимальное решение НАЙДЕНО

\*\*\* ЦЕЛИ \*\*\*

Название	Цель	Факт	Отклонение	Приоритет
Goal1	45	37	8	1
Goal2	100	100	0	< 2 >

\*\*\* ПЕРЕМЕННЫЕ \*\*\*

переменная	Значение
x1	2.75
x2	3.25

< Назад   Готово   Отмена