Лекция 3. **Применение линейного программирования в математических моделях**

Содержание лекции:

Литература

- Экономико-математические методы и прикладные модели: Учеб. пособие для вузов / Под ред. *В.В. Федосеева*. — 2-е изд. М.: ЮНИТИ-ДАНА, 2005. — глава 2.
- *Вентцель Е.С.* Исследование операций: Задачи, принципы, методология. М.: Высшая школа, 2001.
- *Канторович Л.В.* Экономический расчёт наилучшего использования ресурсов. М.: Изд-во АН СССР, 1960.
- *Светлов Н.М., Светлова Г.Н.* Построение и решение оптимизационных моделей средствами программ MS Excel и XA: Методические указания для студентов экономического факультета / РГАУ – МСХА имени К.А. Тимирязева. М., 2005. http://svetlov.timacad.ru/umk1/xa_1.doc

3.1. Принцип оптимальности в планировании и управлении

- Принцип оптимальности предполагает следующее:
  - наличие определённых ресурсов
  - наличие определённых технологических возможностей
  - цель хозяйственной деятельности
    - извлечение прибыли
    - удовлетворение потребностей
    - предотвращение угрозы
    - накопление знаний
    - и т.д.
- Суть принципа:
  - планировать хозяйственную деятельность таким образом, чтобы при имеющихся ресурсах и технологиях *не существовало* способа достичь цели в большей степени, чем это предусматривает план
- В полной мере этот принцип может быть реализован только с помощью экономико-математических моделей

3.2. Задача линейного программирования

Линейная целевая функция

$$\max (\min)\ c_1 x_1 + c_2 x_2 + \ldots + c_n x_n$$

Линейные ограничения

$$a_{11} x_1 + a_{12} x_2 + \ldots + a_{1n} x_n \ \{\le, =, \ge\}\ b_1,$$
$$a_{21} x_1 + a_{22} x_2 + \ldots + a_{2n} x_n \ \{\le, =, \ge\}\ b_2,$$
$$\ldots \ \ldots \ \ldots$$
$$a_{m1} x_1 + a_{m2} x_2 + \ldots + a_{mn} x_n \ \{\le, =, \ge\}\ b_m,$$

$$x_j \ge 0,\ j = 1 \ldots n$$

Условия неотрицательности переменных

- Это **развёрнутая** форма записи

3.2. Задача линейного программирования

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

Линейная целевая функция

$$\sum_{j=1}^{n} a_{ij} x_j = b_i, i = 1 \ldots m$$

$$x_j \geq 0, j = 1 \ldots n; b_i \geq 0, i = 1 \ldots m$$

Линейные ограни-чения

Условия неотрицательности переменных

Любую ЗЛП можно записать в каноническом виде (ограничения – равенства, свободные члены неотрицательны, решается на максимум)

- Это **каноническая** форма записи

3.2. Задача линейного программирования

Линейная целевая функция

$$\max \mathbf{c}\mathbf{x}$$

$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

$$\mathbf{x} \geq \mathbf{0};\ \mathbf{b} \geq \mathbf{0}$$

Линейные ограни-чения

Условия неотрицательности переменных

- Это **матричная** форма записи
  - Она тождественна канонической форме

3.2. Задача линейного программирования

$$\max\ (\min)\ \mathbf{c}\mathbf{x}$$

$$\mathbf{A}\mathbf{x}\ \{\le, =, \ge\}\ \mathbf{b}$$

$$\mathbf{x} \ge \mathbf{0}$$

Линейная целевая функция

Условия неотрицательности переменных

Линейные ограни-чения

- Это **стандартная** форма записи

3.2.

- *Любой* вектор **x**, удовлетворяющий *ограничениям* и *условиям неотрицательности* (безотносительно к целевой функции), называется допустимым решением
  - Если допустимых решений не существует, говорят, что система ограничений несовместна
- Областью допустимых решений (ОДР) называется множество, включающее *все допустимые решения* данной ЗЛП
- Допустимое решение **x***, доставляющее наибольшее значение целевой функции среди всех допустимых решений данной ЗЛП, называется оптимальным решением
  - часто его называют просто решением ЗЛП

3.2.

ЗЛП может:

- не иметь ни одного оптимального решения
  - допустимой области не существует – система ограничений не совместна

    Компактная запись

    $z = \max(x_1+x_2 | x_1+5x_2 \le 1,\ x_1+x_2 \ge 5,\ x_1 \ge 0,\ x_2 \ge 0)$
  - допустимая область существует, но не ограничивает целевую функцию

    $z = \max(2x_1+x_2 | 0.1x_1+0.1x_2 \ge 5,\ x_1 \ge 0,\ x_2 \ge 0)$
- иметь одно оптимальное решение

  $z = \max(x_1+x_2 | 0.1x_1+0.2x_2 \le 5,\ x_1 \ge 0,\ x_2 \ge 0)$

  $x_1=50,\ x_2=0;\ z = 50$
- иметь бесконечно много оптимальных решений

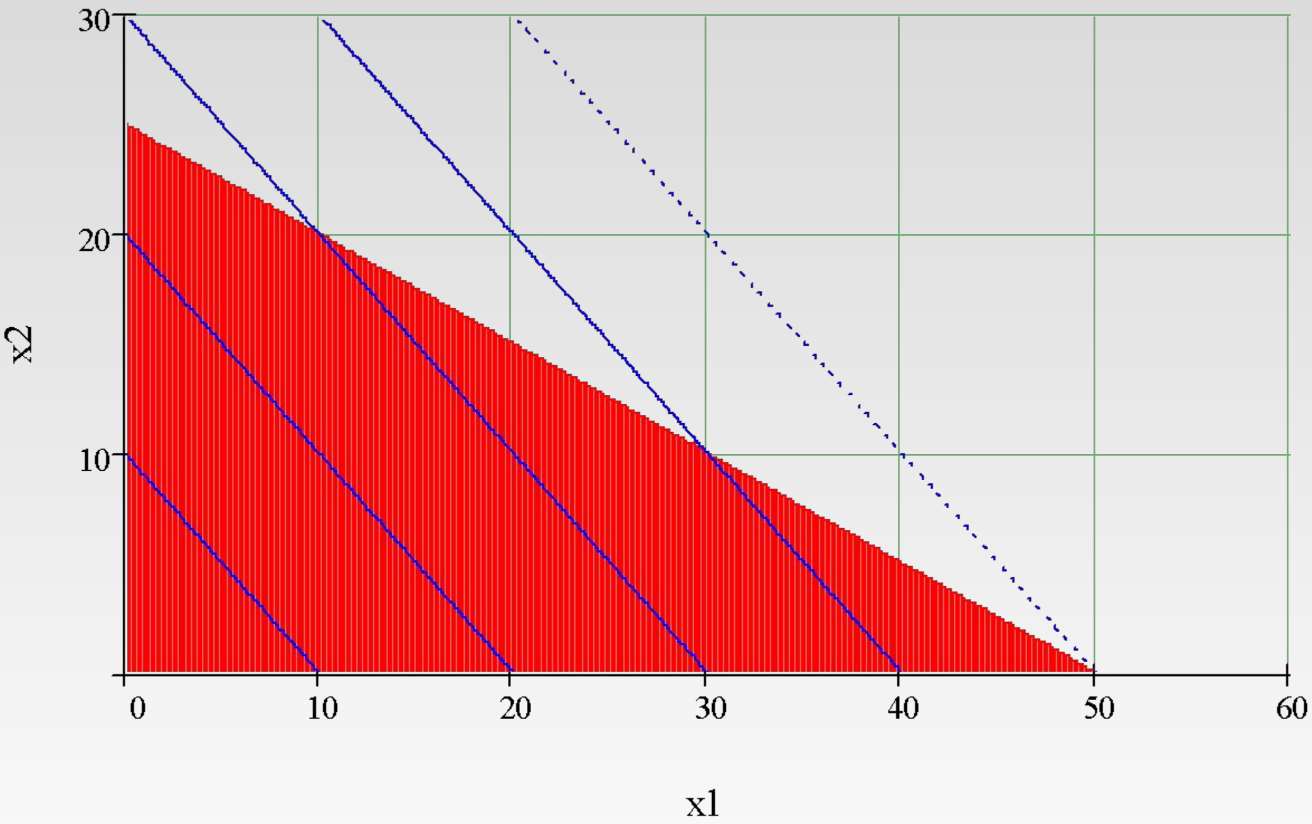
  $z = \max(x_1+x_2 | 0.1x_1+0.1x_2 \le 5,\ x_1 \ge 0,\ x_2 \ge 0)$

  $x_1=50,\ x_2=0;\ z = 50 \ldots x_1=0,\ x_2=50;\ z = 50$

3.2.

$$z = \max(x_1 + x_2 \mid 0.1x_1 + 0.2x_2 \le 5,\ x_1 \ge 0,\ x_2 \ge 0)$$

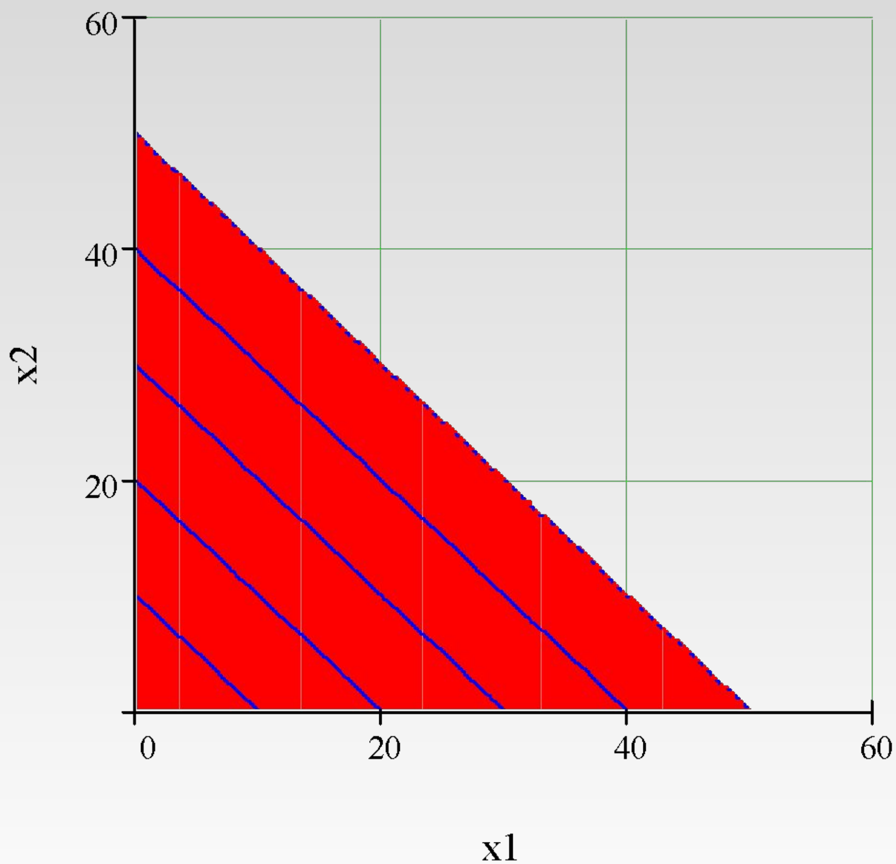
$x_1 = 50$, $x_2 = 0$; $z = 50$



3.2.

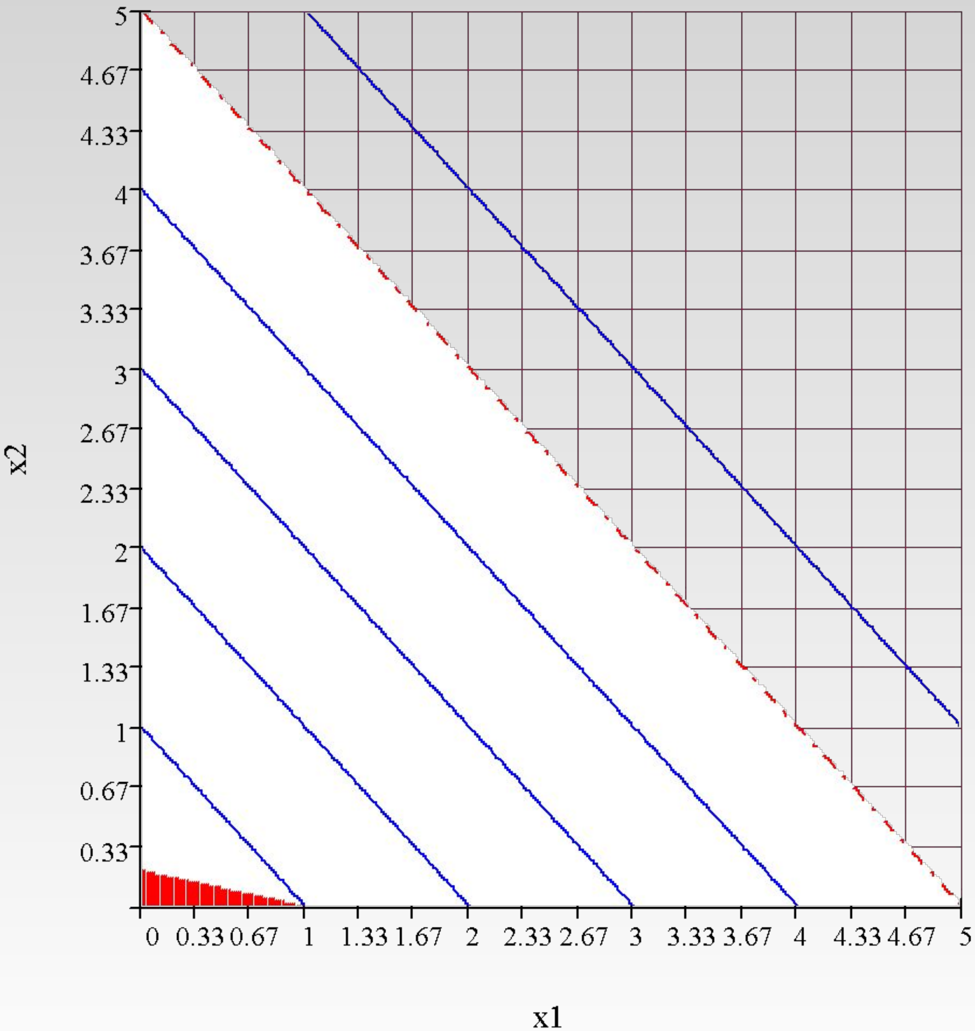
$$z = \max(x_1 + x_2 | 0.1x_1 + 0.1x_2 \le 5,\ x_1 \ge 0,\ x_2 \ge 0)$$

$x_1 = 50$, $x_2 = 0$; $z = 50$ … $x_1 = 0$, $x_2 = 50$; $z = 50$



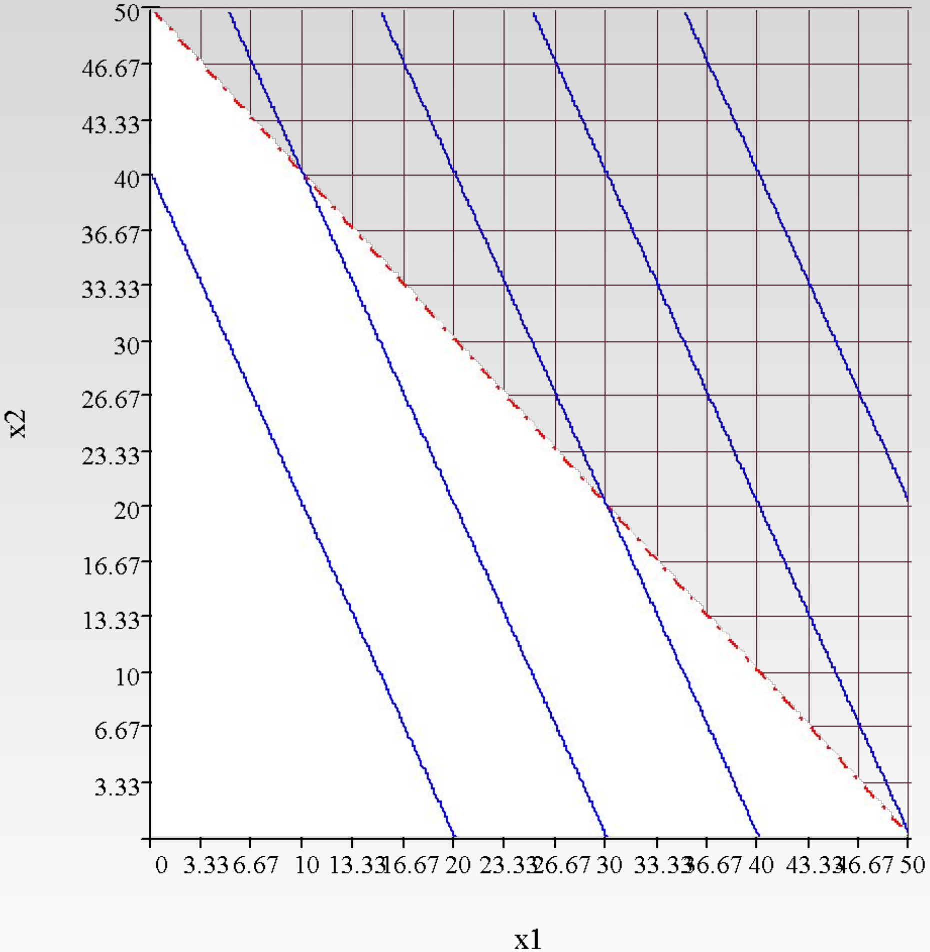
3.2.

$z = \max(x_1 + x_2 | x_1 + 5x_2 \leq 1,\ x_1 + x_2 \geq 5,\ x_1 \geq 0,\ x_2 \geq 0)$



Несовместность системы ограничений

3.2. $z = \max(2x_1+x_2 | 0.1x_1+0.1x_2 \geq 5,\ x_1 \geq 0,\ x_2 \geq 0)$



Неограниченность целевой функции

3.3. Симплексный метод

- Исходные условия применения симплексного метода
  1. ЗЛП записана в канонической форме
  2. Её ограничения линейно независимы
  3. Известно *опорное решение*, в котором:
     - имеется не более *m* ненулевых переменных
       - задача содержит *n* переменных и *m* ограничений
     - все ограничения выполняются
  4. *m* переменных, называемых <u>базисными</u> (среди которых все ненулевые) выражены через:
     - *n*–*m* переменных, называемых <u>свободными</u> (каждая равна нулю)
     - свободный член ограничения
  5. Результат этой процедуры записан в <u>начальную</u> (первую, исходную) <u>симплексную таблицу</u>

3.3.

$z = \max(x_1 + x_2 | 0.1x_1 + 0.2x_2 \le 5,\ x_1 - 2x_2 \le 75,\ x_1 \ge 0,\ x_2 \ge 0)$

$x_1 = 50,\ x_2 = 0;\ z = 50$

Каноническая форма:

$\max x_1 + x_2$

$0.1x_1 + 0.2x_2 + x_3 = 5$

$x_1 - 2x_2 + x_4 = 75$

$x_1 \ge 0,\ x_2 \ge 0,\ x_3 \ge 0,\ x_4 \ge 0$

	$j = 1$	$j = 2$	$j = 3$ $i = 1$	$j = 4$ $i = 2$	b
$c_j,\ c_i \Rightarrow$	1	1	0	0	0
$i = 1$	0.1	0.2	1	0	5
$i = 2$	1	-2	0	1	75

3.3.

- Разрешающий столбец:
  - столбец с наибольшим положительным c_j
    - если положительного c_j нет, <u>достигнут оптимум</u>
- Разрешающая строка:
  - для всех положительных a_{ij} в выбранном столбце считаем b_i/a_{ij}
    - если положительных нет, ц.ф. не ограничена
  - выбираем строку, где это значение минимально

В таблице выделены жирным шрифтом

	$j=1$	$j=2$	$j=3$ $i=1$	$j=4$ $i=2$	b
c_j, $c_i \Rightarrow$	1	1	0	0	0
$i=1$	***0.1***	***0.2***	***1***	***0***	5
$i=2$	***1***	***-2***	***0***	***1***	75

3.3.

- Выполняем *обыкновенные жордановы исключения* во всей таблице:
- для строк $i \neq i'$: $a_{ij\text{нов}} = a_{ij} - a_{i'j}a_{ij'}/a_{i'j'}$, где i' и j' – координаты выбранных (разрешающих) строки и столбца
- для строки $i = i'$: $a_{ij\text{нов}} = a_{ij}/a_{i'j'}$

	$j=1$ $i=1$ (50)	$j=2$ (0)	$j=3$ (0)	$j=4$ $i=2$ (25)	b
c_j, $c_i \Rightarrow$	0	-1	-10	0	-50
$i=1$	***1***	***2***	***10***	***0***	50
$i=2$	***0***	***-4***	***-10***	***1***	25

Положительных c_j больше нет – достигнут оптимум (в больших задачах для этого требуются тысячи итераций)

3.3.

- Опорное решение может быть получено по следующей процедуре:

1. Выбираем произвольный набор базисных переменных и выражаем их через свободные
2. Если строк с отрицательными свободными членами нет – опорное решение получено; иначе – п.3.
3. Одну из таких строк выбираем в качестве вспомогательной целевой функции и проводим по ней процедуру решения ***на минимум***, используя алгоритм симплекс-метода
   - Если в качестве разрешающей выбирается строка с отрицательным свободным членом, то разрешающий элемент тоже должен быть отрицательным
     - для *всех* a_{ij} в выбранном столбце считаем b_i/a_{ij}
     - наименьшее положительное значение этого отношения указывает разрешающую строку
       - если положительных нет, выбираем другую строку с отрицательным свободным членом в качестве вспомогательной целевой функции
       - если таковых не находится, опорных решений не существует (целевая функция не ограничена множеством допустимых решений)
4. Если оптимум достигнут при отрицательном свободном члене – система ограничений несовместна; иначе – п.5
5. Как только достигнуто положительное значение свободного члена, переходим к п.2.

3.3.

В некоторых случаях алгоритм симплексного метода может зацикливаться.

Пути преодоления этой проблемы описаны в рекомендуемой литературе.

3.4. Экономические приложения линейного программирования

Матрица потребности в ресурсах для

Объёмы невоспроизводимых ресурсов

Матрица затрат (+) и выпуска (-) ресурсов при единичном объёме производства в

Матрица выпуска (+) конечной продукции

Вектор объёмов потребления каждого вида конечной продукции при единичном (стандартном) уровне удовлетворения потребностей

Основная задача нар

планирования

$\mathbf{x} = (x_j)$ – объёмы производс

(т, шт., м^3 и т.д.

y – объём удовлетворе

Целевая функция: max y

Балансы невоспроизводимых ресурсов: $\mathbf{A}_1\mathbf{x} \le \mathbf{b}$

Балансы воспроизводимых ресурсов: $\mathbf{A}_2\mathbf{x} \le \mathbf{0}$

Баланс продукции: $\mathbf{A}_3\mathbf{x} \ge y\mathbf{c}$

$\mathbf{x} \ge \mathbf{0}$, $y \ge 0$.

3.4. Экономические приложения линейного программирования

Основная задача произв…

планирования

$\mathbf{x} = (x_j)$ – объёмы реализации п…

(т, шт., м^3 и т.д.)

$\mathbf{y} = (y_k)$ – объёмы закупки ресур…

(т, шт., м^3 и т.д.)

Целевая функция: max

Балансы ресурсов: $\mathbf{A}_1\mathbf{x} \le \mathbf{y} + \mathbf{b}_1$

(например, работники, производственные помещения, оборудование, сырьё, электроэнергия и т.п.)

Выполнение обязательств: $\mathbf{A}_2\mathbf{x} \ge \mathbf{b}_2$

(например, налог на имущество, возврат инвестиционного кредита и т.п.)

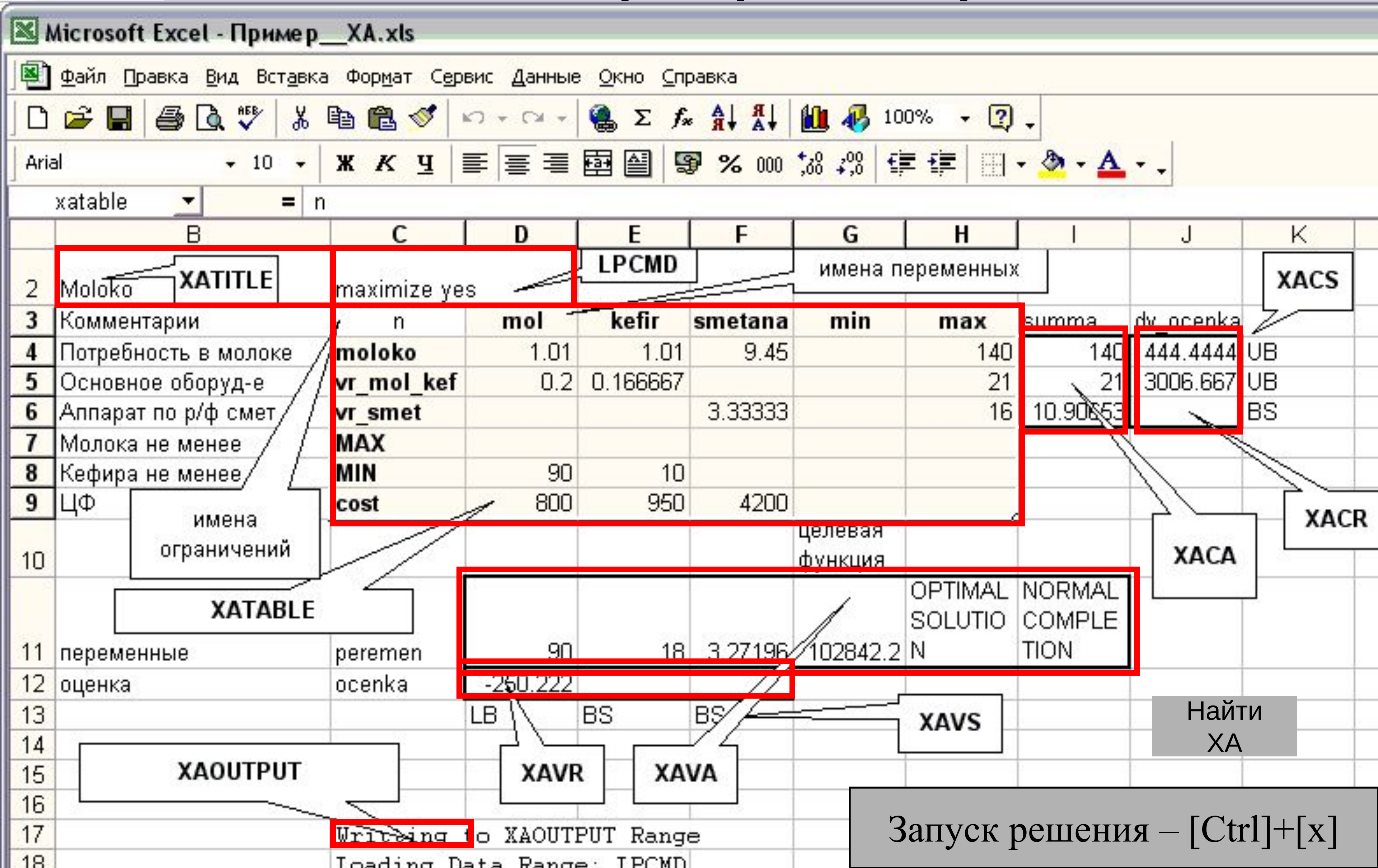
$\mathbf{x} \ge \mathbf{0}, \mathbf{y} \ge \mathbf{0}.$

Вектор цен

Объёмы обязательств, имеющихся у предприятия и учитываемых при оптимальном планировании (выполнение которых зависит от составленного плана)

реализации единицы продукции каждого вида

3.5. Программное обеспечение линейного программирования



Запуск решения – [Ctrl]+[x]

3.5.

- Два способа установки ХА
  - Если есть права доступа к каталогу C:\WINDOWS
    - копируем туда файлы CXA32.DLL и CAXA32.DLL
  - Иначе
    - копируем файлы CXA32.DLL и CAXA32.DLL в ту папку, в которой решаем модель
    - после вызова файла модели нажимаем кнопку

      Найти ХА

      и указываем расположение любого из этих файлов
      - это действие повторяется при каждом вызове Excel
- Антивирус Касперского блокирует выполнение ХА
    - При первом вызове программы следует в ответ на предупреждение антивируса дать ему указание разрешать выполнение данной программы