МЕЖРЕГИОНАЛЬНАЯ
АКАДЕМИЯ УПРАВЛЕНИЯ ПЕРСОНАЛОМ

МАУП

**В. Р. Кулян, Е. А. Юнькова,
А. Б. Жильцов**

МАТЕМАТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ
(с элементами информационных технологий)

*Учебное пособие для студентов
нематематических специальностей вузов*

2-е издание, стереотипное

Киев 2003

ББК 22.18я73
К90

Рецензенты: *Ф. Г. Гаращенко, д-р техн. наук, проф.*
В. С. Рогоза, д-р техн. наук, проф.

Одобрено Ученым советом Межрегиональной Академии управления персоналом (протокол № 6 от 30.09.02)

Кулян В. Р. и др.
К90 Математическое программирование (с элементами информационных технологий): Учеб. пособие для студ. нематемат. спец. вузов / В. Р. Кулян, Е. А. Юнькова, А. Б. Жильцов. — К.: МАУП, 2003. — 2-е изд., стереотип. — 124 с.: ил. — Библиогр.: с. 120.

ISBN 966-608-333-7

В учебном пособии содержатся основные сведения курса “Математическое программирование”. Теоретический материал достаточно полно проиллюстрирован примерами решения задач. Приведены начальные сведения об использовании информационных технологий при решении задач математического программирования.

Для студентов нематематических специальностей вузов.

ББК 22.18я73

ISBN 966-608-333-7

ВВЕДЕНИЕ

При изучении сложных процессов и явлений, когда проведение экспериментов требует значительных затрат или вообще невозможно, применяется моделирование. ***Модель*** — это специально создаваемый объект, на котором воспроизводятся вполне определяемые характеристики исследуемого объекта с целью его изучения, а ***моделирование*** — способ отражения рассматриваемых характеристик изучаемого объекта.

Модели могут быть реализованы с помощью некоторых физических (например, аэродинамическая труба для изучения обтекания воздухом крыла самолета или тренажеры для летчиков, водителей) и абстрактных объектов, описанных выражениями искусственного языка (например, математические выражения различных физических законов).

Математическое моделирование является наиболее совершенным и наиболее эффективным методом моделирования. Естественно, результаты исследования такой модели имеют практический интерес, если модель вполне соответствует (адекватна) рассматриваемому явлению. Для более полного описания действительности приходится строить более сложные и точные математические модели.

Экономическая наука давно использует модели. Одна из первых экономических моделей — модель воспроизводства Ф. Кене — относится к 1758 г. Совершенствование экономико-математических моделей привело к дальнейшему развитию моделирования в экономике. И сейчас ни одна экономическая теория не обходится без математического описания современных экономических процессов.

Одним из наиболее практически важных вопросов экономики является построение хозяйственного плана на разных уровнях экономической системы — от цеха до всего хозяйства страны. Вопросы нахождения наилучшего в определенном смысле, или, как говорят, оптимального, плана играют особую роль в жизни общества: одни и те же затраты могут давать различный экономический эффект в зависимости от принимаемых экономических решений.

В работе всякого производства есть некоторые объективные характеристики, например технологические показатели (параметры) оборудования. Изменять их в процессе производства, как правило, не допускается. Поэтому эти параметры называются *неуправляемыми*. Те же показатели, которые зависят от субъективных решений, например объемы сырья, запущенного в обработку, или количество конечной продукции, запланированное на выпуск, называются *управляемыми параметрами*.

Цель производства, сформулированная в виде некоторого функционального соотношения, включающего в себя как управляемые, так и неуправляемые параметры, называется ***функцией цели***.

Для того чтобы выбрать наилучший план хозяйствования, функцию цели необходимо оптимизировать, т. е. выбрать такие значения управляемых параметров, чтобы целевая функция достигла либо максимального (если речь идет о прибыли), либо минимального (если речь идет о себестоимости продукции) значения.

Однако нельзя запустить в производство сырья больше, чем имеется на предприятии, и нельзя выпустить больше продукции, чем позволяют наличные сырьевые, технологические, финансовые либо другие ресурсы. Другими словами, выбирать наилучший план необходимо на ограниченном множестве управляемых переменных, которое иначе называется ***допустимым множеством решений***.

Выраженные через управляемые переменные целевая функция и ограничения составляют математическую модель задачи оптимизации. Всякий набор значений переменных, удовлетворяющий ограничениям, определяет *допустимый план*, а тот из них, на котором достигается максимум (минимум) целевой функции, определяет *оптимальный план.*

Рассмотрим некоторые из задач планирования и управления, математические модели которых сводятся к оптимизационным задачам или так называемым задачам математического программирования.

Определение наилучшего состава смеси

Иногда такая задача называется задачей о выборе диеты. Пусть известно содержание питательных веществ в различных продуктах питания. Известна также калорийность единицы каждого вида продуктов. Требуется выбрать рацион — набор и количество продуктов — так, чтобы каждое питательное вещество содержалось в нем в необходимом количестве и суммарная калорийность диеты была минимальной.

К задачам о смеси относится и следующая. Бензины разных сортов получают смешиванием различных нефтепродуктов. Заданные показатели качества бензина (октановое число, степень очистки и др.) должны выдерживаться максимально точно. Но исходные нефтепродукты имеют различные технические характеристики (например, северная и южная нефть существенно отличаются по качественному составу), и от того, какие нефтепродукты смешиваются, зависит рентабельность производства. В данной задаче требуется построить такой план смешивания нефтепродуктов, который обеспечивал бы максимальную рентабельность производства и позволял получать бензины заданных сортов в необходимых пропорциях.

Задача об оптимальном плане выпуска продукции

Пусть некоторое предприятие выпускает продукцию заданного ассортимента. Затраты определенного вида ресурсов на выпуск одного изделия из указанного ассортимента, а также полные объемы наличных ресурсов являются фиксированными. Прибыль, получаемая предприятием при изготовлении и реализации единицы каждого вида продукции, известна (постоянная величина).

Предприятию требуется составить такой план выпуска продукции, который был бы технологически осуществим по имеющимся ресурсам всех видов, удовлетворял бы заданным ограничениям на выпуске каждого вида продукции и в то же время приносил наибольшую общую прибыль.

Оптимизация межотраслевых потоков

Пусть имеется несколько отраслей хозяйства, каждая из которых производит только один специфический вид продукции, причем каждый произведенный вид продукции используется (в частности, в нулевом количестве) в производстве во всех остальных отраслях.

Требуется найти такие возможные в заданных условиях объемы производства каждой отрасли и такой план выпуска конечной продукции, при котором максимизируется общая стоимость произведенного конечного продукта.

Транспортная задача

В простейшем варианте эта задача возникает, когда речь идет о рациональной перевозке некоторого однородного продукта от производителей к потребителям. Предполагается, что потребителям безразлично, откуда, из каких пунктов производства будет поступать продукт, лишь бы он поступал в нужном объеме. Однако от того, насколько рациональным будет прикрепление пунктов потребления к пунктам производства, существенно зависит объем транспортной работы. В этой связи естественно возникает задача о наиболее рациональном прикреплении, правильном направлении перевозок груза, при котором потребности удовлетворяются, а затраты на транспортировку минимальны.

Простейшая задача размещения

Пусть в известных пунктах имеются или могут быть размещены предприятия, производящие некоторый продукт. Этот продукт потребляется в других известных пунктах.

Известны затраты на производство единицы продукта и возможный максимальный объем производства во всех пунктах производства, а также затраты на транспортировку из пунктов производства в пункты потребления.

Требуется так выбрать места расположения новых предприятий, объемы производства в них и план перевозок, чтобы суммарные затраты на производство и транспортировку всего необходимого объема продукта были минимальны.

Задача может быть сведена к обычной транспортной задаче, в которой к затратам на транспортировку добавлены затраты на производство в пункте отправления.

Другие виды оптимизационных задач

Приведем типичные классы задач:

1) **управление запасами** (с увеличением запасов увеличиваются расходы на хранение, но при этом уменьшаются потери из-за возможной их нехватки);
2) **распределение ресурсов** (для определенных наборов работ необходимо так распределить ресурсы, чтобы получить наибольшую прибыль при выполнении этих работ либо минимизировать потери, связанные с неполным обеспечением ресурсами);
3) **ремонт и замена оборудования** (работающее оборудование со временем изнашивается, устаревает и подлежит замене, поэтому желательно определить наилучшие сроки восстановительного ремонта и момент замены оборудования модернизированным);
4) **массовое обслуживание** (имеет место в организациях, обслуживающих очереди заявок или требований, например на телефонных станциях, в ремонтных мастерских, билетных кассах и т. п.; здесь задача состоит в том, чтобы минимизировать суммарные ожидаемые потери от несвоевременного обслуживания заявок или простоев оборудования);
5) **календарное планирование** (позволяет составить такое расписание для загрузки оборудования, чтобы суммарная продолжительность комплекса завершенных работ была минимальной);
6) **сетевое планирование и управление** (имеет место при выполнении сложных и дорогостоящих объектов, когда необходимо согласование сроков завершения отдельных комплексов работ и моментов запуска операций всего комплекса);
7) **выбор маршрута** (при проектировании коммуникаций или трубопроводов необходимо выбирать наилучшее их расположение, чтобы оптимизировать потоки в сетях);
8) **комбинированные задачи** (содержат несколько типичных задач одновременно).

Разработка моделей и их типы

Остановимся подробнее на формализации моделей.

По степени соответствия оригиналу модели делятся на *изоморфные*, которые строго соответствуют оригиналу (используются, как правило, для простых систем), и *гомоморфные*, отражающие лишь определенные свойства оригинала (таковыми являются математические модели).

При построении математических моделей исходят из следующих принципов:

1) изучаются и анализируются причинно-следственные связи;
2) используются аналогии;
3) проводятся эксперименты (если это возможно) для выявления существенных переменных.

Управляемые переменные (значения которых можно изменять в определенных пределах) обозначим $x_1, x_2, \dots, x_n$ и объединим их в вектор $\boldsymbol{x} = (x_1, \dots, x_n)$. Неуправляемые переменные (значения которых изменять нельзя) запишем в вектор $\boldsymbol{y} = (y_1, \dots, y_m)$. Размерности векторов при этом могут быть разными, т. е. $n \neq m$.

Пусть функция, зависящая как от управляемых, так и неуправляемых переменных, $E = f(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_m)$, или $E = f(\boldsymbol{x}, \boldsymbol{y})$, является показателем качества или эффективности системы.

Набор функций $g_i(\boldsymbol{x}, \boldsymbol{y})$, $i = 1, 2, \dots, m$ задает соотношения между показателями функционирования системы, а неравенства $g_i(\boldsymbol{x}, \boldsymbol{y}) \leq b_i$, $i = 1, 2, \dots, m$ задают ограничения на все имеющиеся в системе ресурсы.

Тогда в наиболее общем случае задача оптимизации формулируется следующим образом: *найти такое значение вектора $\boldsymbol{x} = (x_1, x_2, \dots, x_n)$, при котором целевая функция $E = f(\boldsymbol{x}, \boldsymbol{y})$ достигает экстремального (максимального или минимального) значения и при этом удовлетворяются все существующие ограничения на вектор управляемых переменных.*

В математическом представлении эта задача имеет такой вид: *найти*

$$\operatorname{extr} E = f(\boldsymbol{x}, \boldsymbol{y}) \tag{1}$$

при ограничениях

$$g_i(\boldsymbol{x}, \boldsymbol{y}) \leq b_i, \qquad i = 1, 2, \dots, m. \tag{2}$$

Для нахождения оптимального решения задачи (1), (2) используют различные методы теории оптимальных решений или так называемого математического программирования.

Математическое программирование — область математики, разрабатывающая теорию и численные методы решения многомерных экстремальных задач с ограничениями, т. е. задач на экстремум функции многих переменных с ограничениями на область изменения этих переменных. В отличие от классической теории экстремальных задач, которая является частью математического анализа, основное внимание уделяется задачам, в которых активно участвуют ограничения на область изменения переменных.

В зависимости от вида целевой функции и ограничений *экономико-математические модели делятся на виды*:

1) если функции $f(\boldsymbol{x}, \boldsymbol{y})$ и $g_i(\boldsymbol{x}, \boldsymbol{y})$, $i = 1, \ldots, m$ линейны относительно $\boldsymbol{x}$, то имеем задачи ***линейного программирования***;
2) если хотя бы одна из этих функций нелинейна относительно $\boldsymbol{x}$, то имеем задачи ***нелинейного программирования***;
3) в ***динамическом программировании*** целевая функция $f(\boldsymbol{x}, \boldsymbol{y})$ имеет специальную структуру (представляет собой сумму или произведение функций, зависящих от различных аргументов);
4) ***стохастическое программирование*** используется в том случае, если вектор неуправляемых переменных $\boldsymbol{y}$ случаен (в этих задачах рассматривается математическое ожидание функции цели при вероятностных ограничениях);
5) если на управляемые переменные накладывается ограничения целостности (переменные $\boldsymbol{x}$ являются целыми либо натуральными числами), то используются методы ***дискретного программирования***;
6) ***эвристическое программирование*** используется в том случае, если строгими методами не удается получить оптимальное решение, но с помощью эвристических подходов можно найти решение, близкое к оптимальному.

Цель настоящего пособия — привести подробные постановки перечисленных задач и основные методы их решения.

1. ЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

Типичной задачей линейного программирования (ЗЛП) является задача определения оптимального ассортимента. Рассмотрим более подробно ее математическую модель.

Пусть на предприятии имеется m видов ресурсов в количествах $b_1, b_2, \ldots, b_m$. Технологически предприятие в состоянии выпускать n видов различных изделий, причем норма расхода ресурса j-го вида ($j = 1, 2, \ldots, m$) на единицу i-го изделия (т. е. изделия под номером i) является величиной заданной, обозначим ее a_{ij}. Эффективность выпуска единицы изделия i-го наименования, т. е. прибыль предприятия после его изготовления и реализации, является величиной известной и характеризуется показателем c_i.

Задача состоит в том, чтобы определить план выпуска изделий (оптимальный ассортимент), при котором суммарный показатель эффективности принимает наибольшее значение.

Пусть x_i — количество единиц изделия i-го вида. В этой задаче именно переменные x_i являются управляемыми, а все остальные — фиксированными и наперед заданными, т. е. неуправляемыми.

Функция цели вида

$$E = f(\boldsymbol{x}) = c_1x_1 + c_2x_2 + \cdots + c_nx_n = \sum_{i=1}^{n} c_ix_i$$

должна быть максимизирована за счет наилучшего выбора значений $x_1, x_2, \ldots, x_n$. Однако выбор должен осуществляться так, чтобы не было перерасхода имеющихся ресурсов, т. е. выполнялись ограничения

$$\sum_{i=1}^{n} a_{ij}x_i \le b_j, \qquad j = 1, 2, \ldots, m.$$

Поскольку бессмысленно говорить об отрицательных объемах выпуска изделий, совершенно естественными в этой задаче будут так называемые прямые ограничения на управляемые переменные:

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

Таким образом, окончательно задача будет иметь такой вид: *найти*

$$\max z = \sum_{i=1}^{n} c_i x_i \tag{1.1}$$

при ограничениях

$$\sum_{i=1}^{n} a_{ij}x_i \le b_j, \qquad j = 1, 2, \ldots, m; \tag{1.2}$$

$$x_1 \ge 0,\ x_2 \ge 0,\ \ldots,\ x_n \ge 0. \tag{1.3}$$

В данном случае функция цели $f(\boldsymbol{x}) = \sum\limits_{i=1}^{n} c_i x_i$ и ограничения $\sum\limits_{i=1}^{n} a_{ij}x_i \le b_j$ являются линейными относительно управляемых переменных, т. е. $x_1, x_2, \ldots, x_n$ входят в указанные выражения в первой степени с некоторыми постоянными коэффициентами. Следовательно, имеем типичную ЗЛП.

В наиболее общем виде математически поставленная ЗЛП формулируется так: требуется найти такие значения $x_1^0, x_2^0, \ldots, x_n^0$ переменных $x_1,\ x_2, \ldots, x_n$, которые удовлетворяют соотношениям вида

$$\begin{array}{c} a_{11}x_1 + \ a_{12}x_2 + \cdots + \ a_{1n}x_n\ R_1\ b_1, \\ a_{21}x_1 + \ a_{22}x_2 + \cdots + \ a_{2n}x_n\ R_2\ b_2, \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n\ R_m b_m \end{array} \tag{1.4}$$

и при этом определяют экстремальное (наибольшее или наименьшее) значение функции

$$f(x_1, x_2, \ldots, x_n) = c_1x_1 + c_2x_2 + \cdots + c_nx_n \tag{1.5}$$

по сравнению с ее значениями при всех других наборах переменных $x_1, x_2, \ldots, x_n$, удовлетворяющих систему ограничений. Здесь R_1, $R_2, \ldots, R_m$ — один из знаков "$\leq$", "$=$" или "$\geq$"; a_{ij}, b_j, c_i — заданные действительные числа.

Любой упорядоченный набор $(x_1, x_2, \ldots, x_n)$ значений переменных, удовлетворяющих всем ограничениям (1.4) (т. е. любое решение системы линейных уравнений и(или) неравенств (1.4)), называется *допустимым решением*, или *планом задачи*. Множество всех допустимых решений называется *допустимым множеством задачи*. Допустимое решение $x_1^0, x_2^0, \ldots, x_n^0$, при котором целевая функция (1.5) принимает максимальное (минимальное) значение, называется *оптимальным решением*, *оптимальным планом* или *просто решением рассматриваемой ЗЛП*.

Геометрическая интерпретация задачи линейного программирования

В простейшем случае, когда ЗЛП содержит только две переменные, легко получить ее геометрическую интерпретацию и решить задачу графически. Разумеется, случай двух переменных не имеет особого практического значения, однако поясняет основные свойства ЗЛП.

Пусть эффективность производства изделий наименований 1 и 2 составляет соответственно $c_1 = 2$; $c_2 = 5$. Кроме того, предприятие располагает тремя видами ресурсов в объемах $b_1 = 400$; $b_2 = 300$; $b_3 = 500$, причем технология производства такова, что ресурсы первого вида расходуются на производство изделий первого наименования, второго — на производство изделий второго наименования и третьего — на производство изделий обоих наименований и при этом в равных долях. Модель задачи такова: *найти*

$$\max z = 2x_1 + 5x_2$$

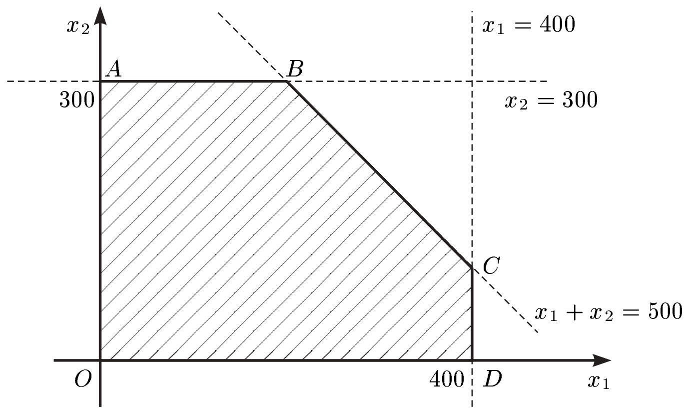
при

$$x_1 \leq 400, \quad x_2 \leq 300, \quad x_1 + x_2 \leq 500$$

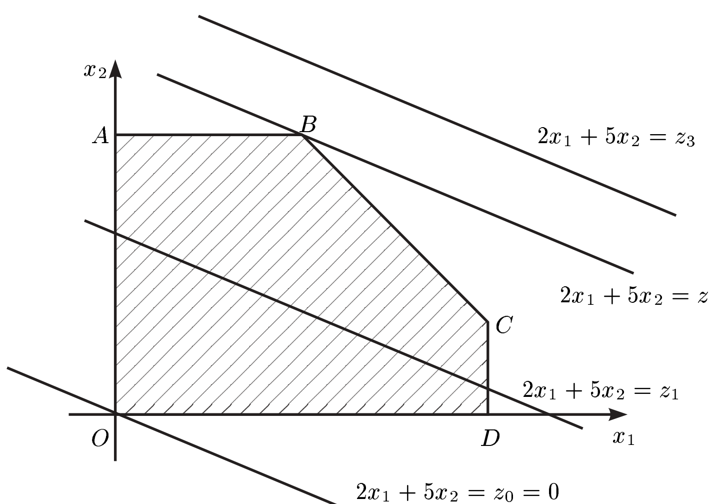
и, естественно,

$$x_1 \geq 0, \quad x_2 \geq 0.$$

Построив на плоскости x_1Ox_2 графики функций всех ограничений, получим следующий график.



Таким образом, допустимая область решений данной ЗЛП представляет собой многоугольник $OABCD$. График целевой функции $f(x_1, x_2) = 2x_1 + 5x_2$ есть множество прямых линий вида $2x_1 + 5x_2 = \text{const}$, среди которых нужно выбрать такую, чтобы значение const было максимальным. Этого можно достичь параллельным сдвигом линии $2x_1 + 5x_2$ в сторону увеличения const. Очевидно, прямая, проходящая через точку B, и будет искомой.



Рассмотрим прямые $2x_1 + 5x_2 = z_1$; $2x_1 + 5x_2 = z_2$ и $2x_1 + 5x_2 = z_3$. Очевидно, имеет место соотношение $z_1 < z_2 < z_3$, т. е. прямая, проходящая через точку B, будет наилучшей, так как для всех линий, расположенных ниже, выполняется $z_1 < z_2$. Третья прямая хотя и соответствует большей константе $z_3 > z_2$, но не имеет с допустимой областью решений общих точек. Таким образом, единственная точка, принадлежащая одновременно прямой $f(x_1, x_2) = z_2$ и допустимой области решений, т. е. точка B, будет оптимальным решением поставленной ЗЛП. Координаты этой точки $(200; 300)$ легко определить на графике.

В такой геометрической интерпретации становятся очевидными основные свойства ЗЛП, которые сформулированы в следующей теореме.

Теорема 1. *Если целевая функция принимает максимальное (минимальное) значение в некоторой точке допустимого множества R, то это значение она принимает в крайней точке множества R. Если целевая функция достигает максимума (минимума) более чем в одной точке, то она принимает это же значение в их выпуклой комбинации.*

Это означает, что решение ЗЛП следует искать лишь в крайних точках допустимого множества решений.

Естественным в этом случае кажется следующий подход. Можно найти каким-либо способом все вершины допустимого множества (доказано, что их число всегда конечно) и, сравнив между собой значения целевой функции в них, выбрать наилучшее. Но такой путь решения ЗЛП, даже с относительно небольшим количеством ограничений и неизвестных, практически неосуществим, так как процесс отыскания вершин весьма трудоемкий, а вершин многогранника может оказаться очень много. Но если осуществлять целенаправленный перебор вершин, то процесс вычислений значительно сократится, так как вершины с заведомо худшим значением целевой функции будут исключены при расчетах. Для определения момента окончания вычислений используют критерий завершения процедуры. Именно такими свойствами обладает симплекс-метод решения ЗЛП (предложен в 1939 г. математиком И. Данцигом), или иначе — метод последовательного улучшения плана. Процедура вычислений в нем гарантирует “улучшение” значения целевой функции в каждой последующей вершине по сравнению с предыдущей, а также выделяет ситуации, когда найдена “наилучшая” вершина с оптимальным значением целевой функции или же когда целевая функция не ограничена на допустимом множестве решений.

Прежде чем переходить к изложению собственно симплекс-метода, необходимо вспомнить несколько положений линейной алгебры, которые лежат в основе вычислений этого метода.

В зависимости от ограничений в условиях исходной задачи различают несколько форм представления ЗЛП. Если ограничения заданы в виде равенств

$$\begin{aligned} a_{11}x_1 + \ a_{12}x_2 + \cdots + \ a_{1n}x_n &= b_1, \\ a_{21}x_1 + \ a_{22}x_2 + \cdots + \ a_{2n}x_n &= b_2, \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m \end{aligned} \tag{1.6}$$

и неизвестные удовлетворяют прямым ограничениям

$$x_1 \geq 0,\ x_2 \geq 0,\ \dots,\ x_n \geq 0, \tag{1.7}$$

то такая форма ЗЛП называется *канонической*.

Если в ограничениях (1.6) используются не только равенства, но и неравенства вида “$\leq$”, “$\geq$”, а прямые ограничения (1.7) накладываются лишь на часть переменных, то такая форма ЗЛП называется *общей*.

Стандартная форма ЗЛП

Существуют стандартные приемы перехода от произвольной ЗЛП к любому специальному ее виду. Как правило, это достигается введением дополнительных переменных, или расширением пространства. Рассмотрим, например, способы перехода к каноническому виду ЗЛП.

Пусть исходная ЗЛП представлена не в канонической форме:

$$c_1x_1 + c_2x_2 + \cdots + c_nx_n \to \min\ (\max); \tag{1.8}$$

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1, \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n &= b_k; \end{aligned} \tag{1.9}$$

$$
\begin{array}{c}
a_{k+1,1}x_1 + a_{k+1,2}x_2 + \cdots + a_{k+1,n}x_n \le b_{k+1}, \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
a_{m1}x_1 + \quad a_{m2}x_2 + \cdots + \quad a_{mn}x_n \le b_m, \\
x_1 \ge 0, \ \dots, \ x_r \ge 0, \quad r \le n.
\end{array} \tag{1.10}
$$

Для того чтобы от ограничений-неравенств (1.9) перейти к строгим равенствам, в неравенства $k+1, k+2, \dots, m$ вводят дополнительные переменные $x_{n+1} \ge 0$; $x_{n+2} \ge 0, \dots$; $x_{n+(m-k)} \ge 0$, которые в целевую функцию входят с нулевыми коэффициентами:

$$
\begin{array}{l}
a_{k+1,1}x_1 + a_{k+1,2}x_2 + \cdots + a_{k+1,n}x_n + x_{n+1} \qquad = b_{k+1}, \\
a_{k+2,1}x_1 + a_{k+2,2}x_2 + \cdots + a_{k+2,n}x_n + x_{n+2} \qquad = b_{k+2}, \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
\quad a_{m1}x_1 + \quad a_{m2}x_2 + \cdots + \quad a_{mn}x_n + x_{n+(m-k)} = b_m.
\end{array}
$$

Переменные, для которых не требуется условие неотрицательности, формально заменяются разностью двух положительных переменных:

$$
x_{r+1} = x'_{r+1} - x''_{r+1}, \ x_{r+2} = x'_{r+2} - x''_{r+2}, \ \dots, \ x_n = x'_n - x''_n,
$$

где $x'_{r+1} \ge 0, \ x''_{r+1} \ge 0, \ \dots, \ x'_n \ge 0, \ x''_n \ge 0.$

Пример. Привести к каноническому виду задачу

$$
\begin{array}{c}
2x_1 - 3x_2 + 5x_3 - \ x_4 \ \to \ \max, \\
x_1 + 2x_2 - \ x_3 + \ x_4 \ \le \ 2, \\
2x_1 - 3x_2 - 2x_3 + 5x_4 \ = \ 4, \\
3x_1 + \ x_2 + 3x_3 - \ x_4 \ \le \ 3, \\
x_1 \ge 0, \qquad x_3 \ge 0.
\end{array}
$$

Вводим две дополнительные переменные $x_5 \ge 0$ и $x_6 \ge 0$ в первом и третьем ограничении, а переменные x_2 и x_4, для которых не требуется неотрицательность, заменяем соответственно двумя разностями: $x_2 = x'_2 - x''_2$; $x_4 = x'_4 - x''_4$, где $x'_2 \ge 0$; $x''_2 \ge 0$; $x'_4 \ge 0$; $x''_4 \ge 0$.

Таким образом, получаем следующую задачу канонического вида:

$$\begin{array}{rcl}
2x_1 - 3(x_2' - x_2'') + 5x_3 - (x_4' - x_4'') + 0\cdot x_5 + 0\cdot x_6 & \to & \max,\\
x_1 + 2(x_2' - x_2'') - x_3 + (x_4' - x_4'') + x_5 & = & 2,\\
2x_1 - 3(x_2' - x_2'') - 2x_3 + 5(x_4' - x_4'') & = & 4,\\
3x_1 + (x_2' - x_2'') + 3x_3 - (x_4' - x_4'') + x_6 & = & 3,
\end{array}$$
$$x_1 \ge 0;\ x_2' \ge 0;\ x_2'' \ge 0;\ x_3 \ge 0;\ x_4' \ge 0;\ x_4'' \ge 0;\ x_5 \ge 0;\ x_6 \ge 0.$$

Иногда необходимо заменить задачу максимизации эквивалентной ей задачей минимизации, или наоборот. Для этого изменяют знак целевой функции, оставляя неизменными ограничения исходной ЗЛП: $\max f(x_1, x_2, \ldots, x_n) = \min[-f(x_1, x_2, \ldots, x_n)]$. В этом легко убедиться на примере функции одной переменной при $a \le x_1 \le b$.

Обоснование симплекс-метода решения ЗЛП

Пусть задача линейного программирования приведена к каноническому виду

$$\begin{array}{rcl}
c_1x_1 + c_2x_2 + \cdots + c_nx_n & \to & \max,\\
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n & = & b_1,\\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n & = & b_2,\\
\multicolumn{3}{c}{\dotfill}\\
a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n & = & b_m,
\end{array}$$
$$x_1 \ge 0,\quad x_2 \ge 0,\quad \ldots,\quad x_n \ge 0.$$

Такое представление ЗЛП, где все уравнения записаны в развернутом виде, называется координатной формой записи. В векторной форме эта ЗЛП выглядит так:

$$(\boldsymbol{c}, \boldsymbol{x}) \to \max,$$
$$A_1x_1 + A_2x_2 + \cdots + A_nx_n = A_0,$$
$$\boldsymbol{x} \ge 0.$$

Здесь $\boldsymbol{c}$ — вектор коэффициентов при неизвестных в целевой функции $\boldsymbol{c} = (c_1, c_2, \ldots, c_n)$; $\boldsymbol{x}$ — вектор переменных $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$; A_j — m-мерный вектор-столбец коэффициентов при x_j,

$$A_j = \begin{pmatrix} a_{1j} \\ a_{2j} \\ \dots \\ a_{mj} \end{pmatrix}, \qquad j = 1, 2, \dots, m;$$

A_0 — m-мерный вектор-столбец свободных членов, стоящих в правой части уравнения,

$$A_0 = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_m \end{pmatrix}.$$

Пусть количество ограничений меньше количества неизвестных, т. е. $m < n$. Случай $m = n$ не представляет интереса, поскольку тогда система является крамеровской, т. е. имеет единственное решение и это решение — точка $\boldsymbol{x}^0 = (x_1^0, x_2^0, \dots, x_n^0)$ — является (в силу единственности) оптимальным.

Для случая $m<n$ интерес представляют только те допустимые решения, у которых m координат отличны от нуля (положительны), а остальные являются нулями. Для того чтобы такие решения описывали вершины допустимой области, необходимо и достаточно, чтобы векторы при соответствующих нулевых компонентах составляли линейно независимую систему. Такая система векторов называется базисом, а соответствующее допустимое решение — опорным.

Пример. Пусть задана система ограничений

$$\begin{aligned} 2x_1 + x_2 + 3x_3 \quad\quad &= 2, \\ x_1 - \quad\quad 2x_3 + x_4 &= 1, \\ x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad x_4 &\geq 0. \end{aligned}$$

Здесь $m = 2$; $A_1 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$; $A_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$; $A_3 = \begin{pmatrix} 3 \\ -2 \end{pmatrix}$; $A_4 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$; $A_0 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$.

Рассмотрим векторы $\boldsymbol{x}^{(1)} = (0, 2, 0, 1)$; $\boldsymbol{x}^{(2)} = (1, 0, 0, 0)$; $\boldsymbol{x}^{(3)} = \left(\frac{1}{2}, 1, 0, \frac{1}{2}\right)$. Легко проверить, что они являются допустимыми решениями. Выделим среди них опорные. В векторе $\boldsymbol{x}^{(1)}$ вторая и четвертая координаты нулевые, а соответствующие им векторы

$A_2 = = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ и $A_4 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ линейно независимы. Следовательно, $\boldsymbol{x}^{(1)}$ — опорное решение.

Для $\boldsymbol{x}^{(2)}$ только первая координата отлична от нуля, а вектор $A_1 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$ ненулевой, следовательно, соответствующая система линейно независима и $\boldsymbol{x}^{(2)}$ — опорное решение.

Решение $\boldsymbol{x}^{(3)}$ заведомо не является опорным, поскольку имеет три (больше чем m) положительные координаты, а всякие три двумерных вектора являются линейно зависимыми.

Замечание. Векторы $A_1, A_2, \ldots, A_m$ называются линейно независимыми, если равенство $\alpha_1 A_1 + \alpha_2 A_2 + \cdots + \alpha_m A_m = 0$ возможно только при $\alpha_1 = \alpha_2 = \cdots = \alpha_m = 0$.

Пример. $A_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$; $A_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Тогда

$$\alpha_1 A_1 + \alpha_2 A_2 = \begin{pmatrix} \alpha_1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

лишь при $\alpha_1 = \alpha_2 = 0$.

Таким образом, линейно независимая система векторов — это минимальный набор векторов, каждый из которого не может быть получен как линейная комбинация остальных векторов. В самом деле, никакими линейными преобразованиями нельзя получить вектор A_1 из A_2 и наоборот. Но любой двумерный вектор есть их линейная комбинация, а именно: произвольный вектор $\begin{pmatrix} a \\ b \end{pmatrix}$, где a, b — произвольные действительные числа, можно представить в виде $\begin{pmatrix} a \\ b \end{pmatrix} = aA_1 + bA_2$. Такая запись вектора называется разложением по базису $\{A_1, A_2\}$.

Пример. Есть набор векторов

$$A_1 = \begin{pmatrix} 1 \\ 0 \\ 3 \end{pmatrix}; \quad A_2 = \begin{pmatrix} 2 \\ 2 \\ 1 \end{pmatrix}; \quad A_3 = \begin{pmatrix} -4 \\ -1 \\ 3 \end{pmatrix};$$

$$A_4 = \begin{pmatrix} 3 \\ -1 \\ 0 \end{pmatrix}; \quad A_5 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.$$

Известно, что для размерности $m = 3$ базисными могут быть только три вектора. Пусть это будут A_2, A_3, A_5. (Матрица B, составленная из этих векторов, имеет определители, отличные от нуля вплоть до третьего порядка.) Тогда координаты любого вектора в данном базисе можно найти по формуле

$$\begin{pmatrix} x_{1l} \\ x_{2l} \\ \dots \\ x_{nl} \end{pmatrix} = B^{-1} \begin{pmatrix} a_{1l} \\ a_{2l} \\ \dots \\ a_{nl} \end{pmatrix},$$

где $\begin{pmatrix} a_{1l} \\ a_{2l} \\ \dots \\ a_{nl} \end{pmatrix} = A_l$ — любой небазисный вектор; $B = \{A_2, A_3, A_5\}$ — матрица, составленная из базисных векторов-столбцов, расположенных в порядке их следования в базисе (называется базисной); $\begin{pmatrix} x_{1l} \\ x_{2l} \\ \dots \\ x_{nl} \end{pmatrix}$ — вектор-столбец искомых координат, т. е. $A_l = A_2 x_{1l} + A_3 x_{2l} + A_5 x_{3l}$.

Здесь $B = \begin{pmatrix} 2 & -4 & 1 \\ 2 & -1 & 1 \\ 1 & 3 & 1 \end{pmatrix}$, а обратная к ней матрица $B^{-1} = \frac{1}{3}\begin{pmatrix} -4 & 7 & -3 \\ -1 & -1 & 0 \\ 7 & 10 & 6 \end{pmatrix}$. Тогда искомые координаты для A_1 принимают значения

$$\begin{pmatrix} x_{11} \\ x_{21} \\ x_{31} \end{pmatrix} = B^{-1} A_1 = \frac{1}{3}\begin{pmatrix} -4 & 7 & -3 \\ -1 & -1 & 0 \\ 7 & 10 & 6 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \\ 3 \end{pmatrix} =$$

$$= \begin{pmatrix} -13/3 \\ -1/3 \\ 25/3 \end{pmatrix} = \frac{1}{3}\begin{pmatrix} -13 \\ -1 \\ 25 \end{pmatrix}.$$

Аналогично

$$\begin{pmatrix} x_{14} \\ x_{24} \\ x_{34} \end{pmatrix} = B^{-1} A_4 = \begin{pmatrix} -19/3 \\ -4/3 \\ 31/3 \end{pmatrix} = \frac{1}{3}\begin{pmatrix} -19 \\ -4 \\ 31 \end{pmatrix}.$$

Для базисных векторов координатные векторы не вычисляют, поскольку они всегда являются единичными (составлены из всех нулей и одной единицы на соответствующей позиции), а именно:

$$\begin{pmatrix} x_{12} \\ x_{22} \\ x_{32} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}; \quad \begin{pmatrix} x_{13} \\ x_{23} \\ x_{33} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}; \quad \begin{pmatrix} x_{15} \\ x_{25} \\ x_{35} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

Если в исходном базисе A_3 заменить вектором A_1, то линейная независимость набора сохранится, т. е. $\{A_2, A_1, A_5\}$ также будет базисом, но координаты небазисных векторов при разложении по новому базису, естественно, изменятся.

Алгоритм симплекс-метода

Пусть задана каноническая невырожденная ЗЛП

$$\begin{array}{rcl}
c_1x_1 + \quad c_2x_2 + \cdots + \quad c_nx_n & \to & \max, \\
a_{11}x_1 + \ a_{12}x_2 + \cdots + \ a_{1n}x_n & = & b_1, \\
a_{21}x_1 + \ a_{22}x_2 + \cdots + \ a_{2n}x_n & = & b_2, \\
\multicolumn{3}{c}{\dotfill} \\
a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n & = & b_m, \\
\multicolumn{3}{c}{x_1 \geq 0,\ x_2 \geq 0,\ \ldots,\ x_n \geq 0.}
\end{array}$$

Предполагаем, что ранг матрицы $A = (a_{ij})$ равен m, т. е. все ограничения-равенства задачи линейно независимы.

Пусть известно какое-нибудь опорное решение $\boldsymbol{x}^{(1)} = \Big(x_1^{(1)}, x_2^{(1)}, \ldots$ $\ldots, x_n^{(1)}\Big)$ этой задачи, т. е. решение, которое имеет m положительных и $n - m$ нулевых координат. Векторы, соответствующие ненулевым координатам, являются линейно независимыми и составляют исходный базис.

Чтобы решить данную задачу или убедиться в том, что она неразрешима, необходимо выполнить предписания в указанном порядке.

1. Разложить по исходному базису все небазисные векторы, т. е. найти все коэффициенты x_{kj}, $k = \overline{1, m}$; $j = \overline{1, n}$, по формуле

$$(x_{1j}, x_{2j}, \ldots, x_{mj})^T = B^{-1}(a_{1j}, a_{2j}, \ldots, a_{mj})^T,$$

где B — базисная матрица.

Для вектора A_0 разложение известно: его компоненты совпадают с ненулевыми компонентами заданного опорного решения, т. е. $x_{i0} = x_s^{(1)}$, где s — номера положительных координат исходного опорного решения.

2. Для каждого j $(j = \overline{1,n})$ вычислить оценку, так называемую симплекс-разность:

$$\Delta_j = z_j - c_j = \sum c_{s_k} x_{kj} - c_j,$$

где c_j — коэффициенты при неизвестных переменных x_j в целевой функции; c_{s_k} — коэффициенты целевой функции при базисных переменных (всего m); x_{kj} — компоненты соответствующего вектора A_j. Если все коэффициенты c_{s_k} объединить в m-мерный вектор $\boldsymbol{c}$, а коэффициенты x_{kj} обозначить как вектор-столбец A_j, то симплекс-разность Δ_j можно записать через скалярное произведение векторов в виде

$$\Delta_j = (\boldsymbol{c}, A_j) - c_j.$$

Если все $\Delta_j \geq 0$, то процесс решения задачи окончен. Рассматриваемое опорное решение оптимально, а оптимум целевой функции

$$z_{\max} = (\boldsymbol{c}, A_0).$$

Если есть $\Delta_j < 0$, то переходим к следующему предписанию.

3. Выяснить, существует ли хотя бы одна отрицательная оценка Δ_j такая, что координаты x_{kj} вектора A_j в заданном базисе будут отрицательны. Если такая оценка Δ_j есть, то процесс решения задачи окончен. Целевая функция задачи не ограничена (сверху) на допустимом множестве. В противном случае переходим к следующему предписанию.

4. Выбрать наименьшую оценку из $\Delta_j < 0$ (пусть это будет Δ_s); вычислить отношение x_{k0}/x_{ks} для всех k, для которых $x_{ks} > 0$, и найти минимальное из этих отношений (пусть это будет $\theta_s = x_{r0}/x_{rs}$).

5. Перейти к новому опорному решению, базис которого получается заменой вектора A_r в предыдущем базисе вектором A_s. Координаты всех векторов $A_0, A_1, \ldots, A_n$ в новом базисе вычисляются по таким основным формулам:

$$x'_{rj} = \theta_j = \frac{x_{rj}}{x_{rs}}; \quad x_{kj} = x_{kj} - \theta_j x_{ks},$$
$$j = \overline{0,n}, \quad k = \overline{1,m}, \quad k \neq r.$$

6. Перейти к выполнению предписания 2 и, если потребуется, выполнить дальнейшие предписания, имея в виду новый базис — новые значения x_{kj}.

Каждый переход к новому базису (а следовательно, к новым значениям величин x_{kj}) называется шагом, или, чаще, итерацией симплекс-метода. Доказано, что за конечное число шагов (итераций) процесс вычислений закончится либо на п. 2 (найдено оптимальное решение), либо на п. 3 (установлено, что оптимального решения нет).

Примечание. Для выполнения п. 1 необходимо знать некоторое опорное решение и матрицу B^{-1}, обратную базисной матрице B. Если исходная задача имеет достаточно большое количество ограничений, то для определения начального базиса, т. е. проверки линейной независимости векторов, необходимо вычислять определители матриц достаточно большой размерности, что вместе с определением обратной матрицы B^{-1} потребует значительного объема вычислений. Поэтому на практике, как правило, удобнее начинать вычисления с опорного решения, которому соответствует единичный базис. Если такого опорного решения у задачи нет, то по определенным правилам ее сводят к другой задаче, имеющей очевидное опорное решение с единичным базисом.

Так, если до преобразования к каноническому виду задача имела ограничения типа "$\leq$", то дополнительные переменные, переводящие эти неравенства в строгие равенства, используются в качестве базисных. Если изначально ограничение имело вид равенства, то в его левую часть можно ввести искусственную переменную, которая отличается от дополнительной тем, что в целевую функцию входит не с нулевым, а с достаточно большим коэффициентом M ($M \gg 0$ для задачи минимизации и $M \ll 0$, если решается задача максимизации). Для ограничений типа "$\geq$" требуются как дополнительные, так и искусственные переменные. Дополнительная переменная, позволяющая преобразовать неравенство в строгое равенство, в этом случае имеет знак "$-$", тогда соответствующая ей компонента опорного плана будет отрицательной, что недопустимо при такой постановке задачи. Поэтому искусственная переменная помогает избежать подобной неувязки.

При ручном счете вычисления удобно оформлять в виде таблиц, каждую из которых называют полной симплекс-таблицей, соответствующей данному опорному решению. Приведем пример такой таблицы:

№ п/п	Базис	$c_{\text{баз}}$	A_0	c_1 / A_1	c_2 / A_2	$\cdots$ / $\cdots$	c_s / A_s	$\cdots$ / $\cdots$	c_n / A_n	θ
1	A_{i_1}	c_{i_1}	x_{10}	x_{11}	x_{12}	$\cdots$	x_{1s}	$\cdots$	x_{1n}	$\dfrac{x_{10}}{x_{1s}}$
2	A_{i_2}	c_{i_2}	x_{20}	x_{21}	x_{22}	$\cdots$	x_{2s}	$\cdots$	x_{2n}	$\dfrac{x_{20}}{x_{2s}}$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
r	A_{i_r}	c_{i_r}	x_{r0}	x_{r1}	x_{r2}	$\cdots$	x_{rs}	$\cdots$	x_{rn}	$\dfrac{x_{r0}}{x_{rs}}$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
m	A_{i_m}	c_{i_m}	x_{m0}	x_{m1}	x_{m2}	$\cdots$	x_{ms}	$\cdots$	x_{mn}	$\dfrac{x_{m0}}{x_{ms}}$
$m+1$			z	Δ_1	Δ_2	$\cdots$	Δ_s	$\cdots$	Δ_m	

Здесь z — значение целевой функции при данном опорном решении,

$$z = (\boldsymbol{c}, \boldsymbol{x}) = c_{i_1}x_{10} + c_{i_2}x_{20} + \cdots + c_{i_m}x_{m0};$$

симплекс-разности

$$\Delta_j = z_j - c_j = \sum_{k=1}^{m} c_{i_k}x_{k0} - c_j.$$

Вектор A_s с наименьшей отрицательной симплекс-разностью Δ_s назовем ведущим (разрешающим). В последнем столбце запишем отношения положительных координат разрешающего вектора A_s к соответствующим координатам вектора A_0.

Базисный вектор A_{i_r} с наименьшим значением θ будем выводить из базиса. Строка, соответствующая этому вектору, а также элемент x_{rs} назовем ведущими (разрешающими).

Пример. Найти максимум функции

$$2x_1 - x_2 + 3x_3 + 2x_4$$

при условиях

$$\begin{aligned} 2x_1 + 3x_2 - x_3 + 2x_4 + x_5 &= 4, \\ x_1 - 2x_2 + 5x_3 - 3x_4 - x_6 &= 1, \\ 4x_1 + 10x_2 + 3x_3 + x_4 + x_7 &= 8, \\ x_j \geq 0, \quad j = 1, 2, \ldots, 7, & \end{aligned}$$

если известно опорное решение данной задачи $(1; 0; 0; 0; 2; 0; 4)$.

Очевидно, ранг матрицы

$$A = (A_1, A_2, A_3, A_4, A_5, A_6, A_7) = \begin{pmatrix} 2 & 3 & -1 & 2 & 1 & 0 & 0 \\ 1 & -2 & 5 & -3 & 0 & -1 & 0 \\ 4 & 10 & 3 & 1 & 0 & 0 & 1 \end{pmatrix}$$

равен 3. Данное опорное решение невырожденное, поскольку имеет три положительные координаты. Базис этого опорного решения составляют векторы A_1, A_5, A_7, т. е.

$$B = (A_1, A_5, A_7) = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 0 & 0 \\ 4 & 0 & 1 \end{pmatrix}; \qquad B^{-1} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -2 & 0 \\ 0 & -4 & 1 \end{pmatrix}.$$

Разложение небазисных векторов A_2, A_3, A_4, A_6 по исходному базису в сокращенной форме имеет вид

$$\begin{pmatrix} x_{12} & x_{13} & x_{14} & x_{16} \\ x_{22} & x_{23} & x_{24} & x_{26} \\ x_{32} & x_{33} & x_{34} & x_{36} \end{pmatrix} = B^{-1}(A_2, A_3, A_4, A_6) =$$

$$= \begin{pmatrix} 0 & 1 & 0 \\ 1 & -2 & 0 \\ 0 & -4 & 1 \end{pmatrix} \begin{pmatrix} 3 & -1 & 2 & 0 \\ -2 & 5 & -3 & -1 \\ 10 & 3 & 1 & 0 \end{pmatrix} = \begin{pmatrix} -2 & 5 & -3 & -1 \\ 7 & -11 & 8 & 2 \\ 18 & -17 & 13 & 4 \end{pmatrix}.$$

Если во второе уравнение-ограничение ввести искусственную переменную $x_8 \geq 0$, то матрица A увеличится на один столбец A_8, но базисная матрица $B = (A_5, A_7, A_8)$ будет единичной, следовательно, $B^{-1} = E$ (единичная матрица) и коэффициенты x_{ij} небазисных векторов пересчитываться не будут.

Поменяв порядок следования ограничений, получим

$$A_0 = \begin{pmatrix} 4 \\ 8 \\ 1 \end{pmatrix}; \qquad A = \begin{pmatrix} 2 & 3 & -1 & 2 & 1 & 0 & 0 & 0 \\ 4 & 10 & 3 & 1 & 0 & 0 & 1 & 0 \\ 1 & -2 & 5 & -3 & 0 & -1 & 0 & 1 \end{pmatrix};$$

$$B = (A_5, A_7, A_8) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

В этом случае опорное решение задается через координаты вектора A_0 в виде

$$\boldsymbol{x}^{(1)} = (0; 0; 0; 0; 4; 0; 8; 1).$$

Заполним первую симплекс-таблицу, соответствующую опорному решению $\boldsymbol{x}^{(1)}$.

№ п/п	Базис	$c_{\text{баз}}$	A_0	2	-1	3	2	0	0	0	$-M$	θ
				A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	
1	A_5	0	4	2	3	-1	2	1	0	0	0	$\frac{4}{-1}$
2	A_7	0	8	4	10	3	1	0	0	1	0	$\frac{8}{3}$
3	A_8	$-M$	1	1	-2	5	-3	0	-1	0	1	$\frac{1}{5}$
			$-M$	$-M-2$	$2M+1$	$-5M-3$	$3M-2$	0	M	0	0	

$z = (c_{\text{баз}}, A_0) = 0 \cdot 4 + 0 \cdot 8 + (-M) \cdot 1 = -M,$

$\Delta_1 = z_1 - c_1 = (c_{\text{баз}}, A_1) - c_1 = 0 \cdot 2 + 0 \cdot 4 - M \cdot 1 - 2 = -M - 2,$

$\Delta_2 = z_2 - c_1 = (c_{\text{баз}}, A_2) - c_2 = 0 \cdot 3 + 0 \cdot 10 - M \cdot (-2) + 1 = 2M + 1,$

$\Delta_3 = z_3 - c_3 = (c_{\text{баз}}, A_3) - c_3 = 0 \cdot (-1) + 0 \cdot 3 - M \cdot 5 - 3 = -5M - 3,$

$\Delta_4 = z_4 - c_4 = (c_{\text{баз}}, A_4) - c_4 = 0 \cdot 2 + 0 \cdot 1 - M \cdot (-3) - 2 = 3M - 2,$

$\Delta_6 = z_6 - c_6 = (c_{\text{баз}}, A_6) - c_6 = 0 \cdot 0 + 0 \cdot 0 - M \cdot (-1) - 0 = M.$

В задаче максимизации $c_8 \ll 0$, т. е. $M \gg 0$, следовательно, отрицательными будут симплекс-разности Δ_1 и Δ_3, причем $\Delta_3 > \Delta_1$. Вектор A_3 имеет две положительные координаты и, следовательно, может быть ведущим (разрешающим), т. е. будет вводиться в базис. Последний столбец в таблице — вектор θ — составлен из отношений координат вектора A_0 к соответствующим координатам ведущего вектора A_2, т. е. $\overline{\theta}_0 = \left(-\frac{4}{1}; \frac{8}{3}; \frac{1}{5}\right)$. Поскольку отрицательные координаты этого вектора в рассмотрение не принимаются, а минимальной среди положительных будет последняя компонента, то это означает, что ведущей будет строка, стоящая на третьем месте ($r = 3$), т. е. соответствующая A_8. Так как переменная x_8 изначально была искусственной переменной, после первой итерации ее можно исключить из таблицы.

Переход к новому опорному решению осуществляется методом исключения Жордана—Гаусса. Перерасчет элементов матрицы $A = \{A_0, A_1, \ldots, A_7\}$ осуществляется так, чтобы на месте вновь введенного столбца A_3 получился единичный вектор с единицей на третьей позиции ($r = 3$). Для этого все элементы ведущей строки делят на ведущий элемент x_{33} и на его месте получают “1”. Остальные элементы в ведущем векторе-столбце исключают, т. е. за счет подходящего умножения ведущей строки и сложения с соответствующими строками получают “0” на всех позициях, кроме третьей. Например, чтобы исключить элемент $x_{13} = -1$, первую строку складывают с третьей (ведущей) без дополнительного умножения, так как $-1 + 1 = 0$, а чтобы исключить элемент $x_{23} = 3$, ведущую строку предварительно умножают на (-3). Как отмечалось, этот процесс представляется основными формулами симплекс-метода:

$$x'_{rj} = \theta_j = \frac{x_{rj}}{x_{rs}}, \qquad k \neq r,$$

$$x'_{kj} = x_{kj} = \theta_j x_{ks}.$$

Для вычисления симплекс-разностей также можно использовать подобную формулу, а именно:

$$\Delta'_j = \Delta_j - \frac{x_{rj}}{x_{rs}} \Delta_s.$$

В результате вычислений получаем последовательность таблиц, последней из которых будет та, у которой либо все симплекс-разности будут положительными, либо ведущий вектор (соответствующий отрицательной симплекс-разности) не будет иметь положительных компонент. В первом случае окончательное решение будет получено в столбце A_0, т. е. положительные компоненты, соответствующие базисным переменным, и нулевые, соответствующие небазисным переменным, составят оптимальное решение. Во втором случае, оптимального решения не существует, поскольку целевая функция не является ограниченной на допустимом множестве решений.

№ п/п	Базис	$c_{баз}$	A_0	2	-1	3	2	0	0	0	θ
				A_1	A_2	A_3	A_4	A_5	A_6	A_7	
1	A_5	2	21/5	11/5	16/5	0	7/5	1	1/5	0	
2	A_7	0	37/5	17/5	56/5	0	14/5	0	3/5	1	
3	A_3	3	1/5	1/5	$-2/5$	1	$-3/5$	0	$-1/5$	0	

Модифицированный симплекс-метод

При решении ЗЛП на ЭВМ (а это, как правило, задачи большой размерности) существенным является количество вычислений на каждой итерации, поскольку это сказывается на экономии памяти, времени и точности вычислений.

Анализ симплекс-метода показал, что для вычисления оценок Δ_j не требуется пересчета всей матрицы $A=\{A_0, A_1, \dots, A_n\}$. Установлено, что $\Delta_j^{(l)}$ (l — номер итерации) можно вычислить так:

$$\Delta_j^{(l)} = \left(c_{\text{баз}}, A_j^{(l)}\right) - c_j = \left(c_{\text{баз}}, B^{-1}(l)A_j\right) - c_j = \left(u^{(l)}, A_j\right) - c_j,$$

где $c_{\text{баз}} = (c_{l1}, \dots, c_{lm})$ — коэффициенты целевой функции при базисных переменных (третий столбец симплекс-таблицы); $u^{(l)} = (u_{l1}, \dots, u_{lm}) = c_{\text{баз}}B^{-1}(l)$. Координаты u_{li} вектора $u^{(l)}$ называются симплекс-множителями.

Кроме того, существуют простые формулы перехода от $B^{-1}(l)$ к $B^{-1}(l+1)$, т. е. от обратной к базисной матрице l-й итерации к матрице $B^{-1}(l+1)$, обратной к базисной матрице $(l+1)$-й итерации. Это известные формулы прямоугольника (метод Жордана—Гаусса)

$$b_{kj}^{(l+1)} = \begin{cases} b_{kj}^{(l)} - \dfrac{x_{rj}^{l}}{x_{rs}} b_{ks}, & r \neq j, \\ \dfrac{b_{rj}}{x_{rs}}, & r = k. \end{cases} \qquad (*)$$

Алгоритм модифицированного симплекс-метода

1. Вычисляем симплекс-разность $\Delta_j = uA_j - c_j$, где $u = c_{\text{баз}}B^{-1}$ — вектор симплекс-множителей; B^{-1} — матрица, обратная к базисной. Если $\Delta_j \geq 0$, $j = \overline{1,n}$, то конец вычислений. Базисное решение оптимально. Если существует j такое, что $\Delta_j < 0$, то переходим к следующему шагу.

2. Выбираем k такое, что $\Delta_k < 0$ (обычно наименьшее). Вычисляем новые векторы $\alpha_s = B^{-1}A_s$ и $\beta = B^{-1}b$. Если вектор α_k не имеет положительных компонент, т. е. $\alpha_k \leq 0$, то конец вычислений. Целевая функция не ограничена сверху на допустимом множестве. Если существуют i такие, что $\alpha_{ik} > 0$, то переходим к следующему шагу.

3. Вычисляем $\theta = \beta_i/\alpha_{ik}$ для всех $\alpha_{ik} > 0$ и среди них выбираем минимальное. Пусть это θ, стоящее на r-м месте. Вектор A_s вводится в базис, а вектор A_r выводится из базиса. Матрицу $B^{-1}(l)$ на l-й итерации вычисляем по матрице $B^{-1}(l-1)$ (на предыдущей итерации) по формулам $(*)$ "правила прямоугольника". Для канонической ЗЛП $B^{-1}(0) = E$ — единичная матрица.

Теория двойственности и двойственные оценки линейных оптимизационных моделей

Пусть некоторый вид продукции может производиться n различными технологическими способами, которые различаются количеством всех производственных ресурсов (различные виды сырья, полуфабрикатов, оборудования, труда и т. д.), используемых за единицу времени, и количеством произведенной при этом готовой продукции. Различаться эти данные могут за счет использования оборудования разных марок, различных пропорций взаимозаменяемого сырья и т. п. Пусть общее количество используемых факторов равно m. Тогда j-я технология характеризуется $(m+1)$-мерным вектором

$$A_j = \begin{pmatrix} a_{1j} \\ a_{2j} \\ \cdots \\ a_{mj} \\ c_j \end{pmatrix},$$

где a_{ij} — количество i-го ресурса, затрачиваемого при единичной интенсивности данной технологии; c_j — количество получаемой при этом продукции.

Предполагается, что все ресурсы имеются в определенных количествах b_i, $i = \overline{1,m}$. Задача заключается в определении интенсивностей, с которыми должны использоваться различные технологии, чтобы суммарное количество готовой продукции было максимальным.

Запишем математическую модель данной задачи. Пусть $x_1, x_2, \ldots$ $\ldots, x_n$ — искомые интенсивности. Будем считать, что затраты ресурсов и объем выпуска готовой продукции пропорциональны соответствующей интенсивности. Тогда общее количество продукции выразится суммой

$$c_1x_1 + c_2x_2 + \cdots + c_nx_n,$$

а используемый в различных технологиях i-й ресурс не должен превышать имеющиеся запасы b_i, т. е. для всех $i = 1, 2, \ldots, m$ должно выполняться неравенство

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \leq b_i.$$

Все x_j $(j = 1, 2, \ldots, n)$ в соответствии с их смыслом неотрицательны, т. е. $x_j \geq 0$, $j = \overline{1, n}$. Таким образом, получается ЗЛП:

$$\begin{array}{rcrcccrcl} c_1x_1 & + & c_2x_2 & + & \cdots & + & c_nx_n & \to & \max, \\ a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & \leq & b_1, \\ a_{21}x_1 & + & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & \leq & b_2, \\ \multicolumn{9}{c}{\dotfill} \\ a_{m1}x_1 & + & a_{m2}x_2 & + & \cdots & + & a_{mn}x_n & \leq & b_m, \\ \multicolumn{9}{c}{x_j \geq 0, \quad j = 1, 2, \ldots, n.} \end{array}$$

Чтобы объективно соизмерить общие затраты с результатом, необходимо определить удельные оценки y_i $(i = \overline{1, m})$ всех задействованных в производстве ресурсов. Тогда функция

$$b_1y_1 + b_2y_2 + \cdots + b_my_m$$

есть суммарная оценка всех ресурсов в тех их количествах $b_1, b_2, \ldots$ $\ldots, b_m$, которые есть на предприятии.

В свою очередь, удельные оценки задействованных ресурсов в зависимости от технологии удовлетворяют неравенствам

$$\begin{array}{rcrcccrcl} a_{11}y_1 & + & a_{21}y_2 & + & \cdots & + & a_{m1}y_m & \geq & c_1, \\ a_{12}y_1 & + & a_{22}y_2 & + & \cdots & + & a_{m2}y_m & \geq & c_2, \\ \multicolumn{9}{c}{\dotfill} \\ a_{1n}y_1 & + & a_{2n}x_2 & + & \cdots & + & a_{mn}y_m & \geq & c_n. \end{array}$$

Здесь используется знак “$\geq$”, поскольку стоимость произведенного продукта c_j не может превышать его производственные затраты: нельзя получить что-то из ничего. Таким образом, получается задача, двойственная к первоначальной. Если первую задачу можно интерпретировать (трактовать) как задачу производственного планирования с одним продуктом, m производственными факторами и

n технологиями, то двойственную к ней задачу — как задачу отыскания удельных оценок, позволяющих соизмерять затраты с результатом при различных технологических способах производства.

Что касается общей теории двойственности в линейном программировании, то существуют определенные правила перехода от прямой ЗЛП к двойственной к ней и наоборот, а также доказаны так называемые теоремы двойственности, устанавливающие взаимосвязь двойственных ЗЛП.

Сформулируем эти правила и теоремы. Пусть имеется некоторая каноническая ЗЛП

$$\begin{array}{rcl} c_1x_1 + \ \ c_2x_2 + \cdots + \ \ c_nx_n & \to & \max, \\ a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n & = & b_1, \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n & = & b_2, \\ \multicolumn{3}{c}{\dotfill} \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n & = & b_m, \\ \multicolumn{3}{c}{x_j \geq 0, \quad j = 1, 2, \ldots, n.} \end{array} \tag{1.11}$$

Тогда двойственная к ней задача строится по очень простому правилу. Свободные члены исходной задачи (1.11) превращаются в коэффициенты целевой функции двойственной задачи, векторы-столбцы коэффициентов при неизвестных — в векторы-строки коэффициентов при неизвестных в двойственной задаче, а коэффициенты целевой функции задачи (1.11) — в свободные члены вновь формулируемой задачи. Кроме того, задача на максимум заменяется задачей на минимум, ограничения-уравнения заменяются неравенствами типа "$\geq$", а ограничения на знак переменных вообще отсутствуют, т. е. имеем задачу, двойственную к исходной:

$$\begin{array}{rcl} b_1y_1 + \ \ b_2y_2 + \cdots + \ \ b_my_m & \to & \min, \\ a_{11}y_1 + a_{21}y_2 + \cdots + a_{m1}y_m & \geq & c_1, \\ a_{12}y_1 + a_{22}y_2 + \cdots + a_{m2}y_m & \geq & c_2, \\ \multicolumn{3}{c}{\dotfill} \\ a_{1n}y_1 + a_{2n}x_2 + \cdots + a_{mn}y_m & \geq & c_n. \end{array} \tag{1.12}$$

При этом задачи (1.11) и (1.12) называются несимметричной парой двойственных задач.

Пример. Известна задача максимизации

$$
\begin{aligned}
3x_1 + 2x_2 - x_3 \qquad\quad &\to \max,\\
2x_1 - 2x_2 + x_3 - 3x_4 &= 8,\\
x_1 + 4x_2 - 2x_3 + x_4 &= 4,\\
2x_1 - x_2 - x_3 \qquad\quad &= 2,\\
x_j \geq 0, \quad j = 1, 2, 3, 4.&
\end{aligned}
$$

Двойственная к ней задача по указанному правилу будет иметь вид

$$
\begin{aligned}
8y_1 + 4y_2 + 2y_3 &\to \min,\\
2y_1 + y_2 + 2y_3 &\geq 3,\\
-2y_1 + 4y_2 - y_3 &\geq 2,\\
y_1 - 2y_2 - y_3 &\geq -1,\\
-3y_1 + y_2 \qquad &\geq 0.
\end{aligned}
$$

Допустим, задана ЗЛП в симметричной форме

$$
\begin{aligned}
c_1x_1 + c_2x_2 + \dots + c_nx_n &\to \max,\\
a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &\leq b_1,\\
a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &\leq b_2,\\
\dots\dots\dots\dots\dots\dots\dots\dots&\dots\dots\\
a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &\leq b_m,\\
x_j \geq 0, \quad j = 1, 2, \dots, n.&
\end{aligned}
\tag{1.13}
$$

Преобразовав эту задачу к каноническому виду и применив указанное правило, получим двойственную к ней задачу

$$
\begin{aligned}
b_1y_1 + b_2y_2 + \dots + b_my_m &\to \min,\\
a_{11}y_1 + a_{21}y_2 + \dots + a_{m1}y_m &\geq c_1,\\
a_{12}y_1 + a_{22}y_2 + \dots + a_{m2}y_m &\geq c_2,\\
\dots\dots\dots\dots\dots\dots\dots\dots&\dots\dots\\
a_{1n}y_1 + a_{2n}y_2 + \dots + a_{mn}y_m &\geq c_n,\\
y_i \geq 0, \quad j = 1, 2, \dots, m.&
\end{aligned}
\tag{1.14}
$$

Задача (1.14) имеет ту же симметричную форму, что и задача (1.13). Такая пара задач называется симметричной парой взаимно двойственных задач.

Заметим, что в общем случае понятие двойственности взаимное, т. е. задача, двойственная к двойственной, совпадает с исходной. Поэтому чаще речь идет не о прямой и двойственной задачах, а именно о паре двойственных задач.

Кроме того, свойства этих задач тесно взаимосвязаны. Установлено, что значения целевой функции в задаче максимизации (у нас есть задача (1.11)) при любом ее допустимом решении $\boldsymbol{x}$ не превышают значения целевой функции задачи минимизации (у нас (1.12)) при любом ее допустимом решении $\boldsymbol{y}$.

Если для каких-нибудь $\boldsymbol{x}_0$ и $\boldsymbol{y}_0$ значения целевых функций совпадают, т. е. $(\boldsymbol{c}, \boldsymbol{x}_0) = (\boldsymbol{y}_0, \boldsymbol{b})$, то $\boldsymbol{x}_0$ и $\boldsymbol{y}_0$ — оптимальные решения соответствующих задач.

На основании этого свойства доказаны следующие теоремы двойственности.

Теорема 2. *Если одна из задач (1.11) или (1.12) имеет оптимальное решение, то и другая также имеет оптимальное решение, причем их оптимумы (значения целевой функции) совпадают.*

Если целевая функция одной из задач не ограничена сверху (снизу) на своем допустимом множестве, то и другая не имеет допустимых решений.

Теорема 3 (двойственный критерий оптимальности). *Допустимые решения $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ и $\boldsymbol{y} = (y_1, y_2, \ldots, y_m)$ двойственных задач соответственно (1.11) и (1.12) тогда и только тогда являются оптимальными решениями этих задач, когда для каждого j: $j = 1, 2, \ldots, n$ выполняется равенство*

$$(a_{1j}y_1 + a_{2j}y_2 + \cdots + a_{mj}y_m - c_j)\, x_j = 0, \quad j = 1, 2, \ldots, n. \tag{1.15}$$

Если рассматривается пара симметричных двойственных задач (1.13), (1.14), то должны выполняться еще и равенства

$$(a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n - b_i)\, y_i = 0, \quad i = 1, 2, \ldots, m. \tag{1.16}$$

Условия (1.15), (1.16) называются условиями дополняющей нежесткости.

Задача (1.11), как отмечалось, называется общей задачей производственного планирования с m ингредиентами и n технологиями. Задача (1.12), двойственная к ней, позволяет найти неотрицательные удельные оценки всех ингредиентов, измеряемые в тех же единицах, в которых измеряется доход (убыток) каждой технологии.

Эти оценки должны быть такими, чтобы суммарная оценка каждой технологии при единичной интенсивности не была меньше соответствующего “дохода”. Наибольший интерес представляет такой допустимый вариант, при котором суммарная оценка всех ингредиентов минимальная.

При такой постановке задач теоремы двойственности получают определенный экономический смысл, а именно: первая теорема двойственности утверждает, что если одна из задач имеет оптимальное решение, то и двойственная к ней задача также разрешима. Это означает, что если определен максимальный возможный доход производства, то всегда можно найти суммарную оценку всех ингредиентов, причем эти оценки обеих задач будут совпадать между собой. Если максимальный возможный доход определить нельзя (целевая функция не ограничена сверху), то нельзя найти и подходящие оценки используемых ингредиентов.

Оптимальные оценки, полученные как решения двойственной ЗЛП, позволяют не только сопоставлять затраты с результатами, но и оценивать степень влияния изменения правых частей ограничений задачи на величину оптимума. Если задача решена, то, естественно, интересно и важно знать, как повлияет на величину оптимума увеличение (или уменьшение) того или иного ресурса, насколько сильно он “реагирует” на изменение запасов отдельных ресурсов.

Свойства объективно обусловленных двойственных оценок широко используются при экономическом анализе результатов решения ЗЛП и при выработке соответствующих рекомендаций. Здесь необходимо отметить, что с изменением условий задачи, вообще говоря, изменяются и оценки. Это означает, что рекомендации должны предваряться исследованиями по устойчивости оценок, т. е. заранее необходимо знать, какие пределы изменения параметров задачи (у нас коэффициенты a_{ij}, b_i, c_j, $i = \overline{1,m}$; $j = \overline{1,m}$) не вызовут значительного изменения полученных оптимальных оценок.

2. ТРАНСПОРТНАЯ ЗАДАЧА

Среди ЗЛП особое место занимает транспортная задача (ТЗ). Это обусловлено, с одной стороны, практической значимостью последней. С другой стороны, ТЗ имеет такие специфические особенности, которые позволяют существенно упростить общие методы решения применительно к этой конкретной задаче.

Транспортная задача формулируется следующим образом. Имеется m поставщиков $P_1, P_2, \ldots, P_m$ однородной продукции. Максимальные объемы поставок продукции заданы и равны a_i, $i = 1, 2, \ldots, m$. Эта продукция используется n потребителями. Объемы потребностей заданы и равны b_j, $j = 1, 2, \ldots, n$. Стоимость перевозки единицы продукции от i-го поставщика к j-му потребителю известна для всех $i = 1, 2, \ldots, m$ и $j = 1, 2, \ldots, n$ и равна c_{ij}. Требуется установить такие объемы перевозок x_{ij} от каждого поставщика P_i $(i = 1, 2, \ldots, m)$ к каждому потребителю Q_j $(j = 1, 2, \ldots, n)$, чтобы суммарные затраты на перевозки были минимальными и потребности всех потребителей были удовлетворены (если это возможно).

Математическая модель задачи имеет вид

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \to \min; \tag{2.1}$$

$$\begin{aligned} &\sum_{j=1}^{n} x_{ij} \le a_i, \qquad i = 1, 2, \ldots, m; \\ &\sum_{i=1}^{m} x_{ij} \ge b_j, \qquad j = 1, 2, \ldots, n, \end{aligned} \tag{2.2}$$

$$x_{ij} \ge 0, \qquad i = 1, 2, \ldots, m, \qquad j = 1, 2, \ldots, n.$$

Если, кроме того, выполняется условие

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j,$$

то имеем так называемую закрытую (сбалансированную) ТЗ.

Основные свойства закрытой ТЗ

1. Задача в любом случае допустима и разрешима.
2. Среди уравнений-ограничений (2.2) лишь $m + n - 1$ линейно независимых.
3. Если в ТЗ все числа a_i $(i = 1, 2, \ldots, m)$ и b_j $(j = 1, 2, \ldots, n)$ целые, то хотя бы одно оптимальное решение задачи целочисленное.

При решении ТЗ, как и любой ЗЛП, базисные решения — вершины допустимого многогранника — играют особую роль. В силу общих определений и с учетом второго свойства можно заключить, что невырожденное базисное решение ТЗ содержит ровно $m + n - 1$ ненулевых перевозок x_{ij}. В вырожденном случае базисное решение ТЗ содержит менее $m + n - 1$ ненулевых перевозок.

Данные ТЗ и вычисления, связанные с решением задачи, заносят в транспортную таблицу. От обычной симплекс-таблицы она отличается тем, что в ней заполнены лишь те клетки (i, j), где перевозки $x_{ij} > 0$. Такие клетки называются занятыми, остальные — свободными. Очевидно, что для невырожденного случая занятых клеток будет $m + n - 1$.

Исходная транспортная таблица имеет такой вид.

	Q_1	Q_2	$\cdots$	Q_n	a_i
P_1	c_{11}	c_{12}	$\cdots$	c_{1n}	a_1
P_2	c_{21}	c_{22}	$\cdots$	c_{2n}	a_2
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
P_m	c_{m1}	c_{m2}	$\cdots$	c_{mn}	a_m
b_j	b_1	b_2	$\cdots$	b_n	$\sum\limits_i a_i = \sum\limits_j b_j$

В последнем столбце таблицы приведена информация о возможностях поставщиков, в последней строке — о потребностях потребителей. В остальных клетках помещена стоимость перевозок c_{ij}, $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$ и, кроме того, в $m + n - 1$ клетках также записываются значения $x_{ij} > 0$.

Для ТЗ в силу ее специфики найти начальное базисное решение проще, чем для обычной ЗЛП. Приведем два возможных, широко используемых на практике метода построения начального базисного решения ТЗ.

Метод северо-западного угла. Заполняют ненулевыми перевозками транспортную таблицу начиная с ее верхней левой (северо-западной) клетки. При этом распределяют продукцию первого поставщика, максимально удовлетворяя первого потребителя, затем второго и т. д. до полного распределения продукции первого поставщика. Если в каком-либо пункте Q_1 спрос не был удовлетворен, то поставки осуществляются от второго поставщика, затем третьего и т. д. до полного удовлетворения спроса. Разумеется, если потребности в пунктах $Q_1, Q_2, \ldots$ выполнены или полностью распределена продукция поставщиков $P_1, P_2, \ldots$, то соответствующие потребители и поставщики уже не участвуют в распределении. Занятые клетки транспортной таблицы имеют характерную ступенчатую форму (начало “ступенек” находится в верхнем левом углу таблицы, конец — в нижнем правом углу). Сумма перевозок в i-й строке равна a_i, сумма перевозок в j-м столбце — b_j.

Очевидно, полученный таким образом план перевозок образует допустимое решение ТЗ.

Пример 1. Продукцию поставщиков P_1, P_2, P_3 с возможностями $\boldsymbol{a} = \{10, 15, 7\}$ необходимо распределить между потребителями Q_1, Q_2, Q_4 с потребностям $\{3; 5; 10; 14\}$.

Воспользуемся следующей транспортной таблицей.

	Q_1	Q_2	Q_3	Q_4	a_i
P_1					$a_1 = 10$
P_2					$a_2 = 15$
P_3					$a_3 = 7$
b_j	$b_1 = 3$	$b_2 = 5$	$b_3 = 10$	$b_4 = 14$	

После распределения продукции по методу северо-западного угла получим решение ТЗ.

	Q_1	Q_2	Q_3	Q_4	a_i
P_1	3	5	2		$a_1 = 10$
P_2			8	7	$a_2 = 15$
P_3				7	$a_3 = 7$
b_j	$b_1 = 3$	$b_2 = 5$	$b_3 = 10$	$b_4 = 14$	

Заметим, что в рассмотренном методе не использовалась матрица C транспортных издержек. Поэтому начальное допустимое базисное решение, построенное по методу северо-западного угла, обычно далеко от оптимального. В предлагаемом далее методе начальное решение строится с использованием информации о транспортных издержках.

Метод минимального элемента. Идея метода состоит в том, чтобы максимально загрузить перевозками коммуникации с минимальными транспортными издержками. Здесь в отличие от предыдущего метода таблицу заполняем вразброс начиная с наиболее "дешевых" клеток.

Метод реализуется следующим образом. На первом шаге определяем минимальный элемент матрицы C. Пусть это будет $c_{i_1 j_1}$. Полагаем $x_{i_1 j_1} = \min(a_{i_1}, b_{j_1})$. Возможны два случая: 1) $a_{i_1} \le b_{j_1}$ (возможности a_{i_1} поставщика P_{i_1} меньшие, чем потребности b_{j_1} потребителя Q_{j_1}; 2) $a_{i_1} > b_{j_2}$ (наоборот). Тогда в первом случае нулями заполняется строка i_1, кроме элемента j_1, а во втором — столбец j_1, кроме элемента i_1, и в дальнейшем в вычислениях не участвует поставщик P_{i_1} (в случае 1) или потребитель Q_{j_1} (в случае 2). На втором шаге из матрицы C вычеркиваем соответствующую строку или столбец. Получаем матрицу $C^{(1)}$. Полагаем также

$$a_i^{(1)} = \begin{cases} a_i, & i \ne i_1, \\ a_i - x_{i_1 j_1}, & i = i_1; \end{cases}$$

$$b_j^{(1)} = \begin{cases} b_j, & j \ne j_1, \\ b_j - x_{i_1 j_1}, & j = j_1. \end{cases}$$

Второй шаг — выполнение описанных на первом шаге операций применительно к матрице $C^{(1)}$ и величинам $a_i^{(1)}$ и $b_j^{(1)}$. Следующие шаги продолжаются до полного заполнения транспортной таблицы (или вычеркивания строк (столбцов) матрицы C).

Пример 2. Пусть для рассматриваемой далее задачи с $\boldsymbol{a} = \{10; 15; 7\}$ и $\boldsymbol{b} = \{3; 5; 10; 14\}$ матрица транспортных издержек имеет вид

$$C = \begin{pmatrix} 2 & 4 & 3 & 7 \\ 3 & 6 & 3 & 5 \\ 2 & 2 & 5 & 4 \end{pmatrix}.$$

Тогда построение начального базисного решения по методу минимального элемента выполняется следующим образом.

	Q_1	Q_2	Q_3	Q_4	a_i	$a^{(1)}$	$a^{(2)}$	$a^{(3)}$	$a^{(4)}$	$a^{(5)}$
P_1	3		7		10	7	7	0	0	0
P_2			3	12	15	15	15	15	12	0
P_3		5		2	7	7	2	2	2	0
b_j	3	5	10	14						
$b^{(1)}$	0	5	10	14						
$b^{(2)}$	0	0	10	14						
$b^{(3)}$	0	0	3	14						
$b^{(4)}$	0	0	0	14						
$b^{(5)}$	0	0	0	0						

Вычеркивания в матрице C выполняют в следующем порядке:

$$C = \begin{array}{ccccccccc}
2^{1} & & 4 & \text{—} & 3^{3} & \text{—} & 7 & \text{—} & \text{3-й шаг}\\
| & & | & & & & & & \\
3 & & 6 & & 3^{4} & & 5 & \text{—} & \text{6-й шаг}\\
| & & | & & | & & & & \\
2 & & 2^{2} & & 5 & & 4 & \text{—} & \text{5-й шаг}\\
| & & | & & | & & & & \\
\text{1-й шаг} & & \text{2-й шаг} & & \text{4-й шаг} & & & &
\end{array}$$

Предложенный метод является некоторой модификацией метода северно-западного угла и, как и последний, приводит к начальному допустимому базисному решению. Поскольку при этом используется матрица транспортных издержек, указанное решение довольно близко к оптимальному.

Метод потенциалов решения транспортной задачи

Пусть u_i и v_j — потенциалы пункта соответственно P_i и Q_j ($i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$). Назовем величину $c_{ij} - (v_j - u_i) = \Delta_{ij}$ относительной оценкой переменной x_{ij}. Тогда базисное решение $x = \|x_{ij}\|$, $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$ оптимально тогда и только тогда, когда существуют потенциалы u_i и v_j такие, что относительные оценки $\Delta_{ij} = 0$ для базисных (занятых) клеток транспортной таблицы и $\Delta_{ij} \geq 0$ для небазисных (свободных) клеток.

Сущность метода потенциалов состоит в следующем.

1. В качестве первого приближения к оптимальному решению берется любое начальное базисное решение (построенное методами северо-западного угла, минимального элемента или любым другим методом).

2. Определяются потенциалы u_i и v_j так, чтобы в каждой базисной клетке выполнялось условие $\Delta_{ij} = 0$ или, что то же самое, $v_j - u_i = c_{ij}$, $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$. Заметим, что такая система содержит $m+n-1$ уравнений (по количеству базисных клеток) и $m + n$ переменных. Поэтому одну из переменных задают произвольно (например, полагаем $u_1 = 0$), остальные находят из указанной системы.

3. Вычисляются относительные оценки Δ_{ij} для небазисных (свободных) клеток. Если для всех этих клеток $\Delta_{ij} \geq 0$, то проверяемое базисное решение оптимально. Если же существуют небазисные клетки, для которых $\Delta_{ij} < 0$, то исследуемое базисное решение можно улучшить, вводя в число базисных одну из указанных клеток (обычно ту, для которой относительная оценка $\Delta_{ij} < 0$ минимальна). Пусть это (i_0, j_0). Рассмотрим вопрос о том, какая из базисных клеток выводится из числа базисных. Из клеток транспортной таблицы образуем цикл (замкнутую цепочку) так, чтобы он содержал небазисную клетку, а остальные клетки были базисными. Пометим клетки цикла следующим образом: небазисную — знаком "+", остальные (базисные) — знаком "−" или "+", но так, чтобы соседние по строке или по столбцу клетки имели разные знаки.

Увеличим теперь объем перевозок, стоящих в клетках со знаком "+", на величину θ, а объем перевозок, стоящих в клетках со знаком "−", уменьшим на ту же величину θ. Очевидно, если в качестве θ выбрать минимум среди отрицательных клеток, то в результате такого перераспределения грузов одна из базисных клеток станет свободной, а клетка (i_0, j_0) войдет в базис.

4. С помощью описанной процедуры перераспределения грузов по циклу клетка (i_0, j_0) вводится в число базисных, а базисная клетка, которая становится свободной, очевидно, из базиса выводится. Полученное новое базисное решение проверяется на оптимальность (п. 2, 3) и т. д., пока для текущего базисного решения не будет выполнен критерий оптимальности.

В невырожденном случае метод потенциалов приводит к оптимальному решению ТЗ за конечное количество шагов. В вырожденном случае, если занятых клеток меньше, чем $m + n - 1$, возможно зацикливание, для предупреждения которого необходимы дополнительные меры.

Пример 3. Для рассмотренного ранее примера построим оптимальное решение методом потенциалов. В качестве начального опорного решения выберем решение, построенное методом северо-западного угла.

	Q_1	Q_2	Q_3	Q_4	a_i	u_i
P_1	3	5	2		10	
P_2			8	7	15	
P_3				7	7	
b_j	3	5	10	14		
v_j						

Для базисных клеток (1;1), (1;2), (1;3), (2;3), (2;4), (3;4) вычислим потенциалы u_i, v_j $(i = 1, 2, 3;\ j = 1, 2, 3, 4)$ используя соотношения $v_j - u_i = c_{ij}$.

Имеем

$$\begin{aligned} v_1 - u_1 &= 2, & v_4 - u_2 &= 5,\\ v_3 - u_1 &= 3, & v_3 - u_2 &= 3,\\ v_2 - u_1 &= 4, & v_4 - u_3 &= 4. \end{aligned}$$

В системе 6 уравнений и 7 неизвестных u_1, u_2, u_3, v_1, v_2, v_3 и v_4, поэтому одну из неизвестных переменных необходимо доопределить. Пусть $u_1 = 0$, тогда $v_1 = 2$, $v_2 = 4$, $v_3 = 3$, $u_2 = 0$, $v_4 = 5$, $u_3 = 1$.

Заносим эти данные в таблицу и вычисляем относительные оценки $\Delta_{ij} = c_{ij} - (v_j - u_i)$ для небазисных клеток:

$$\begin{aligned}
\Delta_{14} &= c_{14} - (v_4 - u_1) = 7 - 5 + 0 = 2,\\
\Delta_{21} &= 1 - (v_1 - u_2) = 3 - (2 - 0) = 1,\\
\Delta_{22} &= 6 - (v_2 - u_2) = 6 - (4 - 0) = 2,\\
\Delta_{31} &= 2 - (2 - 1) = 1,\\
\Delta_{32} &= 2 - (4 - 1) = -1,\\
\Delta_{33} &= 5 - (3 - 1) = 3.
\end{aligned}$$

Относительные оценки Δ_{ij} помещаем в левом верхнем углу клеток транспортной таблицы. В правом верхнем, как и прежде, размещены значения c_{ij}, а в центре — план перевозок.

	Q_1	Q_2	Q_3	Q_4	a_i	u_i
P_1	3^{2}	5^{4}	2^{3}	2 7	10	0
P_2	1 3	2 6	8^{3}	7^{5}	15	0
P_3	1 2	−1 2	3 5	7^{4}	7	1
b_j	3	5	10	14	32	
v_j	2	4	3	5		

Единственной клеткой, для которой $\Delta_{ij} < 0$, является клетка (3;2). Строим цикл, включающий эту клетку (остальные клетки цикла должны быть базисными), и помечаем его вершины знаками "+" и "−". Определяем $Q = \min(5, 8, 7) = 5$ и, перезагружая клетки ТЗ по указанному циклу, получаем новое базисное решение.

	Q_1	Q_2	Q_3	Q_4	a_i	u_i
P_1	3^{2}	0 4	7^{3}	2 7	10	0
P_2	1 3	2 6	3^{3}	12^{5}	15	0
P_3	1 2	5^{2}	3 5	2^{4}	7	1
b_j	3	5	10	14	32	
v_j	2	4	3	5		

Вычисляя, как и ранее, потенциалы для нового базисного решения, а затем относительные оценки Δ_{ij} небазисных переменных, убеждаемся, что все $\Delta_{ij} \geq 0$, а значит, полученное базисное решение

$$x^* = \begin{pmatrix} 3 & 0 & 7 & 0 \\ 0 & 0 & 3 & 12 \\ 0 & 5 & 0 & 2 \end{pmatrix}$$

является оптимальным.

Многие практические задачи, связанные с планированием перевозок, не укладываются в рамки рассмотренной задачи, так как осложнены дополнительными ограничениями или вообще существенно сложнее по постановке.

Например, понятно, что транспортные магистрали, связывающие источники продукции с потребителями, имеют ограниченные (в ограниченный промежуток времени) пропускные способности. Эти ограничения могут не позволить реализовать найденный оптимальный план перевозок. Поэтому приходится включать в задачу ограничения, связанные с пропускными способностями магистралей. Тогда математическая модель задачи принимает вид

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \to \min;$$

$$\sum_{j=1}^{n} x_{ij} = a_i, \qquad i = 1, 2, \ldots, m;$$

$$\sum_{i=1}^{m} x_{ij} = b_j, \qquad j = 1, 2, \ldots, n;$$

$$0 \leq x_{ij} \leq d_{ij}.$$

Здесь d_{ij} — пропускная способность магистрали (i, j), т. е. максимальный объем продукции, который можно перевезти по ней за рассматриваемый промежуток времени.

Такая задача может оказаться неразрешимой, если сумма всех магистралей, ведущих к j-му потребителю, меньше его потребностей, т. е. $\sum\limits_{i=1}^{m} d_{ij} < b_j$.

Для решения задач с ограниченными пропускными способностями применяется специальный метод потенциалов, позволяющий за конечное количество итераций найти оптимальное решение либо установить неразрешимость такой задачи.

Исходная ТЗ может быть осложнена еще наличием промежуточных транспортных узлов, в которых обрабатываются грузы (например, перевалка на другой вид транспорта), или отсутствием транспортных магистралей между отдельными потребителями и поставщиками. Для таких задач также имеются более или менее эффективные методы решения.

Принципиально меняется характер модели задачи даже в простейшем случае, если критерием оптимальности выбрано время, необходимое для обеспечения всех потребителей нужным объемом продукции, и требуется минимизировать это время. Такая задача называется ТЗ по критерию времени. Математическая модель этой задачи оказывается нелинейной (за счет нелинейности целевой функции). Если вместо матрицы стоимостей перевозок $C = (c_{ij})$ задана матрица $T = (t_{ij})$, где t_{ij} — время перевозки, и предполагается, что t_{ij} не зависит от объема перевозимой продукции, то целевая функция примет вид

$$\max t_{ij} \to \min .$$

Здесь $\max t_{ij}$ означает максимальное из t_{ij}, соответствующих ненулевым перевозкам (для $x_{ij} = 0$ считаем $t_{ij} = 0$) рассматриваемого допустимого плана перевозок. Очевидно, за время $\max t_{ij}$ будут осуществлены все перевозки. Требуется выбрать допустимый план, для которого это время будет минимальным.

Решение такой задачи можно свести к последовательному решению нескольких ЗЛП. Имеется также алгоритм, позволяющий постепенно улучшать исходный опорный план по правилам, напоминающим метод потенциалов.

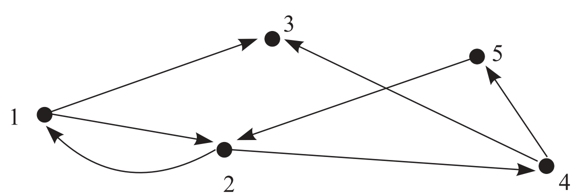
Часто требуется составлять рациональный план перевозок не одного, а нескольких видов продукции. Отказ от предположения, что рассматриваются перевозки однородного вида продукции, значительно усложняет задачу. И еще более сложной является задача, когда требуется распределить перевозки между разными видами транспорта, учитывая, что стоимость перевозки единицы продукции от i-го поставщика к j-му потребителю зависит не только от вида продукции, но и от вида транспорта, на котором она перевозится.

Такие задачи называются многоиндексными ТЗ. Увеличение количества индексов у переменных резко увеличивает количество расчетов, необходимых для решения задачи, и делает невозможным применение простых методов для ее решения.

В заключение отметим, что многие задачи транспортного типа можно решать методами, основанными на математическом анализе свойств соответствующих им сетей (графов), т. е. условного графического изображения транспортных узлов и коммуникаций. Сетевые методы решения транспортных (и не только транспортных) задач часто более эффективны, чем методы их решения в рассмотренной матричной постановке.

Характерным примером решения ТЗ на сетке является задача отыскания потоков в сетях с некоторыми оптимальными свойствами.

Графическое изображение транспортных узлов и коммуникаций (так называемый граф) характеризуется набором вершин и дуг (упорядоченных пар вершин, изображающих направления перевозок между пунктами).



Например, граф $\Gamma = \{I, U\}$ описывается множеством вершин $I = \{1; 2; 3; 4; 5\}$ и множеством дуг $U = \{(1; 2), (1; 3), (2; 1), (2; 4), (4; 3), (4; 5), (5; 2)\}$. Такой граф называется ориентированным (направленным). Если направление движения не указано, то дуги называются ребрами, а граф — неориентированным. На "транспортном" языке дуга называется коммуникацией, последовательность взаимосвязанных вершин (цепь) — маршрутом, путь — направленным маршрутом и т. д.

С понятием графа тесно связано понятие сети. Сетью называется граф, элементам которого поставлены в соответствие некоторые параметры: d_i — вершинам, r_{ij} — дугам. На практике величины d_i часто интерпретируются как объемы производства ($d_i > 0$) или потребления ($d_i < 0$) некоторого однородного продукта; r_{ij} — ограничения на пропускные способности коммуникации, связывающей пункты i

и j. Параметром дуги может быть также стоимость c_{ij} перевозки продукции из пункта i в пункт j.

Понятие потока, который вводится на графе, обобщает понятие допустимых транспортных перевозок. Как и физический аналог потока жидкости в сообщающихся сосудах, поток на графе обладает свойствами сохранения и непрерывности. В зависимости от критериев оптимальности на сети ставятся задачи об оптимальном потоке (если необходимо минимизировать стоимость перевозок), о кратчайшем пути (если из многих вариантов маршрута необходимо выбрать оптимальный) о максимальном потоке (если необходимо максимизировать весь объем перевозок) и др.

Более подробно с методами решения транспортных матричных и сетевых задач можно ознакомиться в [3].

3. ЦЕЛОЧИСЛЕННЫЕ И ДИСКРЕТНЫЕ ЗАДАЧИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

Среди практически важных задач отыскания условного экстремума линейной функции важное место занимают задачи с требованием целочисленности всех или части переменных.

Прежде чем формально определить задачи целочисленного линейного программирования, рассмотрим примеры.

Задача о назначениях. Имеется n видов различных работ и n кандидатов для их выполнения. Предполагается, что каждый кандидат назначается на один и только один вид роботы. Обозначим c_{ij} доход, получаемый при назначении i-го кандидата на j-й вид работы. Необходимо распределить кандидатов на работу так, чтобы суммарный доход от произведенных назначений был максимальным.

Положим

$$x_{ij} = \begin{cases} 1, & \text{если } i\text{-й кандидат назначен на } j\text{-ю работу;} \\ 0 & \text{в противном случае.} \end{cases}$$

Тогда математическая модель задачи о назначениях примет следующий вид:

максимизировать

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

при ограничениях

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, 2, \ldots, n; \qquad \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, 2, \ldots, n,$$

$(x_{ij} = 0 \quad \text{либо} \quad x_{ij} = 1)$.

Задача о ранце. Имеются различные предметы n наименований. Путешественник, собираясь в поход, должен упаковать некоторые из них в ранец. Ранец имеет m ограничений по весу, объему, линейным размерам и т. п. Пусть a_{ij} — i-я характеристика ($i = 1, 2, \ldots, m$) предмета j-го наименования ($j = 1, 2, \ldots, n$); b_i — ограничения по весу, объему, линейным размерам и т. д. Пусть x_j — количество предметов j-го наименования, запланированных к загрузке в ранец, а c_j — полезность одного предмета j-го наименования. Тогда математическая постановка задачи приобретает следующий вид:
максимизировать

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

при ограничениях

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

$$x_j \geq 0 \text{ — целое}, \qquad j = 1, 2, \ldots, n.$$

Заметим, что в первоначальной формулировке данной задачи речь шла не о рюкзаке, а о загрузке бомбардировщиков различных типов бомбовым запасом с целью максимизации суммарного эффекта данной системы боевых операций.

Задача о выборе типа судов (или других видов транспортных средств). Речное пароходство обслуживает n маршрутов с примерно постоянным количеством пассажиров на каждом. Расходы пароходства для обслуживания каждого из m его судов определяются содержанием обслуживающего персонала на каждом судне и расходом горючего, доходы — количеством проданных билетов.

Возникает задача о выборе для каждого маршрута судов таких типов и в таких количествах, чтобы перевезти всех пассажиров и максимизировать прибыль пароходства.

Пусть по j-му маршруту за сезон перевозится $b_j^{(1)}$ человек. Перевозку пассажиров по этому маршруту можно осуществлять m различными типами судов. Для каждого i-го типа судов ($i = 1, 2, \ldots, m$) известны следующие характеристики:

1) $a_i^{(1)}$ — грузоподъемность (количество мест);

2) $a_i^{(2)}$ — численность обслуживающего персонала;

3) $a_i^{(3)}$ — расход горючего за сезон;

4) c_{ij} — прибыль за сезон от использования i-го судна по j-му маршруту.

Необходимо выбрать парк судов для каждого маршрута, при котором максимизируется прибыль пароходства при соблюдении ограничений: общий расход горючего за сезон не может превышать b_3 единиц, а общая численность обслуживающего персонала — b_2 человек.

Теперь приведем формальное определение задачи целочисленного программирования (в стандартной форме): найти вектор $\boldsymbol{x} = (x_1, \ldots, x_n)$, минимизирующий линейную функцию

$$F(x) = \sum_{j=1}^{n} c_j x_j$$

и удовлетворяющий системе ограничений

$$\sum_{j=1}^{n} a_{ij} x_j = b_i, \qquad i = 1, 2, \ldots, m;$$

$$x_j \geq 0, \qquad x_j \text{ — целое}, \quad j = 1, 2, \ldots, n.$$

Приведенная формулировка отличается от обычной ЗЛП лишь условием целочисленности, которое накладывается на переменные $\boldsymbol{x}$. Это требование приводит к таким специфическим особенностям, что иногда эту задачу относят к нелинейному программированию.

Если целочисленность накладывается лишь на часть переменных, то задача называется задачей частично-целочисленного программирования. Если же переменные могут принимать лишь конечное количество значений (не обязательно целое), то говорят о задаче дискретного программирования.

Кажется, целочисленная задача решается проще, чем обычная ЗЛП, в силу того, что ее допустимое множество решений есть конечное множество точек. Но этим не удается воспользоваться при решении практических задач. Например, 64 переменных $\boldsymbol{x}$, принимающих значение лишь нуль или единица, дадут множество точек 2^{64}.

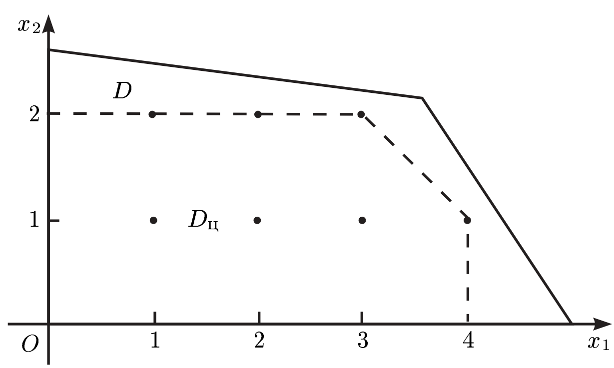
Часто при первом ознакомлении с целочисленной ЗЛП возникает естественное желание решить эту задачу без учета целочисленности $\boldsymbol{x}$, а затем округлить результат до ближайшего целого. Но такой непосредственный подход может увести далеко от действительного оптимума.

Пример 1. В задаче

$$\begin{aligned}
x_1 - 3x_2 + 3x_3 &\to \max,\\
2x_1 + x_2 - x_3 &\le 4,\\
4x_1 - 3x_2 &\le 2,\\
-3x_1 + 2x_2 + x_3 &\le 3,\\
x_j \ge 0; \quad x_j \text{ — целое}; \ j &= 1, 2, 3.
\end{aligned}$$

Решением является точка $x^*_{\text{ц}} = (2; 2; 5)$, а без требования целочисленности решение ЗЛП имеет вид $x^* = (1/2; 0; 9/2)$. Очевидно, никакие варианты округления решения вспомогательной ЗЛП не дают даже допустимого решения исходной задачи целочисленного линейного программирования.

Геометрически области допустимых решений обычной ЗЛП и целочисленной задачи можно представить так, как показано далее.



Но если в обычной ЗЛП перейти от допустимой области D к области $D_{\text{ц}}$, решение такой задачи будет целочисленным.

К сожалению, непосредственный переход от задачи целочисленного программирования к ЗЛП практически неосуществим, поскольку построить область $D_{\text{ц}}$ очень трудно. Эту задачу можно решить в некотором смысле частично, последовательно урезая область D с помощью так называемых отсекающих плоскостей.

Методы отсечения

Идея методов отсечения состоит в следующем. Решается ЗЛП, полученная из целочисленной задачи отбрасыванием условия целочисленности. Если решение вспомогательной ЗЛП целочисленное, то оно же является и решением исходной задачи. Если оптимальное решение вспомогательной задачи не целочисленное, то к решенной ЗЛП добавляют новое линейное ограничение; оно удовлетворяет целочисленным решениям исходной целочисленной задачи, но не удовлетворяет полученному нецелочисленному решению первоначальной ЗЛП. Указанное дополнительное линейное ограничение определяет некоторую отсекающую плоскость и называется *правильным отсечением*. Новые правильные отсечения добавляют к первоначальной вспомогательной ЗЛП до тех пор, пока на некотором шаге не будет получено целочисленное решение вспомогательной задачи, являющееся, очевидно, оптимальным решением исходной задачи целочисленного линейного программирования.

Одним из методов отсечения является **метод Гомори**.

Решается задача полностью целочисленного линейного программирования

$$\begin{gathered}\sum_{j=1}^{n} c_j x_j \to \max,\\ a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1,\\ \dots\dots\dots\dots\dots\dots\dots\dots\\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m,\end{gathered} \tag{3.1}$$

где $\boldsymbol{x} = (x_1, x_2, \dots, x_n)$ — целочисленный вектор.

Наряду с ней рассматривается вспомогательная ЗЛП

$$\begin{gathered}\sum_{j=1}^{n} c_j x_j \to \max,\\ a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1,\\ \dots\dots\dots\dots\dots\dots\dots\dots\\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m,\\ x_1, x_2, \dots, x_n \geq 0,\end{gathered} \tag{3.2}$$

полученная из предыдущей отбрасыванием условия целочисленности вектора $\boldsymbol{x}$. Задача (3.2) решается симплекс-методом. Пусть на последней итерации ограничения этой задачи приняли вид

$$\begin{array}{rl} x_1 + \cdots + \alpha_{1,m+1}x_{m+1} + \cdots + \alpha_{1n}x_n & = \beta_1, \\ x_2 + \cdots + \alpha_{2,m+1}x_{m+1} + \cdots + \alpha_{2n}x_n & = \beta_2, \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\ x_m + \alpha_{m,m+1}x_{m+1} + \cdots + \alpha_{mn}x_n & = \beta_m. \end{array}$$

Как известно, оптимальным решением задачи (3.2) будет вектор $\boldsymbol{x}^* = (\beta_1, \ldots, \beta_m; 0; \ldots, 0)$. Пусть существует номер i такой, что β_i — нецелое (в противном случае целочисленный вектор $\boldsymbol{x}^*$ является оптимальным решением и задачи (3.1)).

Обозначим, как обычно, $[z]$ и $\{z\}$ соответственно целую и дробную части числа z. Заметим, что если β_i — нецелое число, то $\{\beta_i\} \neq 0$.

Согласно общей идее метода отсечения необходимо перейти к решению новой вспомогательной ЗЛП, у которой наряду с исходными ограничениями рассматривается дополнительное ограничение, осуществляющее правильное отсечение. Рассмотрим ограничение

$$\{\beta_i\} - \sum_{j=m+1}^{n} \{\alpha_{ij}\} x_j \leq 0, \tag{3.3}$$

которое соответствует β_i такому, что $\{\beta_i\} \neq 0$. Имеет место следующее утверждение.

Теорема. *Дополнительное линейное ограничение (3.3) является правильным отсечением для задачи целочисленного линейного программирования (3.1).*

Пример 2. Решить целочисленную ЗЛП

$$\begin{array}{c} -x_1 + x_2 - 2x_3 \to \min, \\ x_1 + \qquad 5x_3 = 4, \\ x_2 + 5x_3 = 1, \\ x_j \geq 0; \quad x_j \text{ — целое; } j = 1, 2, 3. \end{array}$$

Решаем вспомогательную ЗЛП, полученную из исходной отбрасыванием условия целочисленности x_j, $j = 1, 2, 3$. Вычисления выполняем в виде симплекс-таблицы.

№ п/п	$c_{\text{баз}}$	Базис	A_0	-1	1	-2	θ
				A_1	A_2	A_3	
0	-1	x_1	4	1	0	5	4/5
	1	x_2	1	0	1	$\boxed{5}$	1/5
		Δ		0	0	-2	
1	-1	x_1	3	1	-1	0	
	2	x_3	1/5	0	1/5	1	
		Δ		0	2/5	0	

На последнем шаге ограничения вспомогательной ЗЛП имеют вид

$$\begin{aligned} x_1 - \quad x_2 \qquad\quad &= 3, \\ \frac{1}{5}x_2 + x_3 \ &= \frac{1}{5}. \end{aligned}$$

По второму ограничению строим правильное отсечение, определяемое соотношением (3.3), а именно:

$$\frac{1}{5} - \frac{1}{5}x_2 \le 0 \quad \text{или} \quad x_2 \ge 1.$$

Строим новую вспомогательную ЗЛП, добавляя ограничение $x_2 \ge 1$. Чтобы ограничение “канонизировать”, используем дополнительную, переменную x_4. Получаем задачу

$$\begin{aligned} -x_1 + \quad x_2 - 2x_3 \qquad &\to \ \min, \\ x_1 - \quad x_2 \qquad\qquad &= \ 3, \\ \frac{1}{5}x_2 + \quad x_3 \qquad &= \ \frac{1}{5}, \\ - \quad x_2 + \qquad\quad x_4 \ &= -1. \end{aligned}$$

Так как последняя компонента в векторе свободных членов является отрицательной, полученную ЗЛП решаем двойственным симплекс-методом. Вычисления выполняем в виде симплекс-таблицы.

№ п/п	$c_{баз}$	Базис	A_0	A_1	A_2	A_3	A_4
0	-1	x_1	3	1	-1	0	0
	2	x_3	1/5	0	1/5	1	0
	0	x_4	-1	0	$\boxed{-1}$	0	1
		Δ		0	2/5	0	0
1	-1	x_1	4	1	0	0	-1
	2	x_3	0	0	0	1	1/5
	1	x_2	1	0	1	0	-1

В отличие от обычного симплекс-метода в двойственном ведущей будет строка с отрицательным свободным членом, а ведущим будет столбец, соответствующий положительной симплекс-разности.

Вычисления в двойственном симплекс-методе заканчиваются после того, как в столбце свободных членов A_0 не остается отрицательных элементов.

Таким образом, на первом шаге двойственного симплекс-метода получено оптимальное целочисленное решение $\boldsymbol{x}^* = (4; 1; 0; 0)$ вспомогательной ЗЛП, следовательно, решением целочисленной задачи будет $\boldsymbol{x}^* = (4; 1; 0)$.

Заметим, что в практических вычислениях сходимость этого метода очень медленная; реализация приведенного алгоритма с повышением размерности затрудняется.

Метод ветвей и границ

Для решения задач дискретного (и, в частности, целочисленного) линейного программирования широко применяется метод ветвей и границ. Это один из комбинаторных методов, который реализуется в виде направленного перебора вариантов решений оптимизационной задачи указанного типа.

Идея метода такова. Вычисляется нижняя (для задачи минимизации) оценка целевой функции $F(\boldsymbol{x})$ на допустимом множестве решений D (причем способ вычисления границы для каждой конкретной задачи решается особо).

Множество D разбивается на непересекающиеся подмножества. Осуществляется пересчет оценок. В дальнейшем на ветвление выбирается подмножество с наименьшей оценкой, поскольку в нем естественно искать минимум в первую очередь. Подмножества, в которых оценка строго больше предыдущей, в дальнейшем не рассматриваются.

Проверяется оптимальность вновь полученных решений. Для этого используется очевидный признак оптимальности, суть которого в следующем. Пусть D состоит из объединения подмножеств $D^{k,l}$, $l = \overline{1,r}$, а $\boldsymbol{x}^{k,l}$ — оптимальное решение ЗЛП на соответствующем подмножестве $D^{k,l}$. Если при этом

$$F(\boldsymbol{x}^{k,\nu}) = z(D^{k,\nu}) \leq z(D^{K,l}), \qquad l = \overline{1,r},$$

то решение $\boldsymbol{x}^{k,\nu}$ оптимальное. (При решении задачи максимизации знак последнего неравенства меняется на противоположный.)

Применительно к задаче целочисленного линейного программирования процедура метода ветвей и границ имеет следующий вид.

Рассматривается частично целочисленная ЗЛП:
минимизировать

$$F(\boldsymbol{x}) = \sum_{j=1}^{n} c_j x_j \tag{3.4}$$

при ограничениях

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \qquad i = \overline{1,m}, \tag{3.5}$$

$$x_j \text{ — целые; } j = \overline{1,n_1};\ n_1 \leq n;\ 0 \leq x_j \leq d_j;\ j = \overline{1,n}. \tag{3.6}$$

Некоторые из чисел d_j могут равняться $+\infty$. Предполагается, что многогранное множество, заданное этими ограничениями, ограничено.

Как и при решении подобных задач методами отсечения, рассматриваем вспомогательную ЗЛП, полученную из исходной отбрасыванием условия целочисленности. Пусть $\boldsymbol{x}^0 = (x_1^0, \ldots, x_n^0)$ — оптимальное решение указанной вспомогательной ЗЛП. Если вектор $\boldsymbol{x}$ удовлетворяет условию целочисленности, он является оптимальным решением исходной задачи. В противном случае величина $z(D^0) = F(x_0)$ дает нижнюю оценку целевой функции $F(\boldsymbol{x})$ на множестве D, заданном ограничениями (3.5) и (3.6) ($D^0 = D$).

Пусть некоторая координата вектора $\boldsymbol{x}^0$ не является целочисленной. В этом случае проводим ветвление множества D^0 на подмножества $D^{1,1}$ и $D^{1,2}$, добавляя к исходным ограничениям, задающим D^0, ограничения соответственно $x_j \le [x_j^0]$ и $x_j \ge [x_j^0]+1$, где $[\,\cdot\,]$ — целая часть числа. Затем решаем новые вспомогательные ЗЛП, соответствующие подмножествам $D^{1,1}$ и $D^{1,2}$, и вычисляем оценки $F(\boldsymbol{x})$ на этих множествах и т. д.

При дальнейшем ветвлении естественно выбрать множество $D^{k,\nu}$ с наименьшей оценкой целевой функции. Процесс продолжается до тех пор, пока не будет получено оптимальное решение, удовлетворяющее условию целочисленности. Не исключено, что ветвления придется проводить до тех пор, пока все решения ЗЛП (если они допустимы) не будут удовлетворять условию целочисленности. В этом случае выбирают решение, которому соответствует минимальное значение $F(\boldsymbol{x}^{k,\nu})$.

Заметим, что для полностью целочисленной ЗЛП в случае, когда коэффициенты c_j $(j=\overline{1,n})$ целевой функции $F(x)$ целые, оценка z может быть заменена на более сильную оценку

$$z =]F(\boldsymbol{x}^{k,\nu})[,$$

где $]\cdot[$ — наименьшее целое число, не меньшее u ($]u[=\min\limits_{n\ge u} n$; n — целое). Например, $]1,5[=2$; $]-1,5[=-1$.

Пример 3. Решить задачу

$$F(\boldsymbol{x}) = -x_1 - 3x_2 \to \min$$

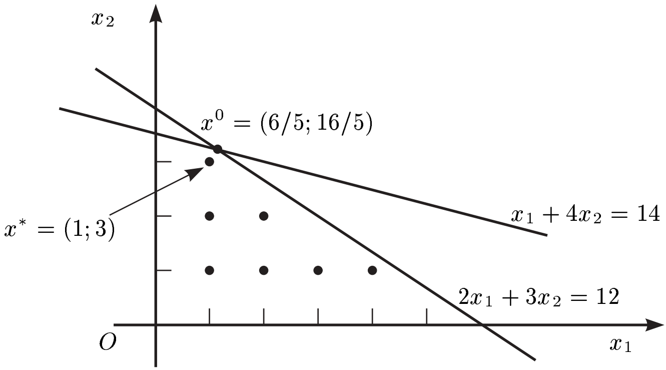
при ограничениях

$$\begin{aligned} x_1 + 4x_2 &\le 14, \\ 2x_1 + 3x_2 &\le 12, \\ x_1 \ge 0;\ x_2 \ge 0;\ &x_1, x_2 \text{ — целые.} \end{aligned}$$

Обозначим D область, заданную ограничениями задачи.

1. Решаем вспомогательную ЗЛП (без условия целочисленности переменных x_1 и x_2):

$$\begin{aligned} F(\boldsymbol{x}) = -x_1 - 3x_2 &\to \min, \\ x_1 + 4x_2 &\le 14, \\ 2x_1 + 3x_2 &\le 12, \\ x_1 \ge 0; \quad x_2 &\ge 0. \end{aligned} \tag{1}$$



Решение задачи (1) имеет вид

$$\boldsymbol{x}^0 = (6/5; 16/5); \qquad F(\boldsymbol{x}^0) = -54/5.$$

Проводим ветвление множества D^0: $D^0 = D = D^{1,1} \cup D^{1/2}$, где

$$D^{1,1} = \{\boldsymbol{x} : \ \boldsymbol{x} \in D^0, \ x_1 \le [6/5] + 1\};$$
$$D^{1,2} = \{\boldsymbol{x} : \ \boldsymbol{x} \in D^0, \ x_1 \ge [6/5] + 1 + 2\}.$$

Вычисляем границу $z(D^0) = F(\boldsymbol{x}^0)$:

$$F(\boldsymbol{x}^0) =]-54/5[=]-10 - 4/5[= -10.$$

2. Решаем вспомогательные ЗЛП:

$$F(\boldsymbol{x}) = -x_1 - 3x_2 \to \min, \qquad x \in D^{1,1}; \tag{2}$$

$$F(\boldsymbol{x}) = -x_1 - 3x_2 \to \min, \qquad x \in D^{1,2}. \tag{3}$$

Решения задач (2) и (3) имеют вид соответственно

$$\boldsymbol{x}^{1,1} = (1, 13/4); \qquad F(\boldsymbol{x}^{1,1}) = -43/4;$$

$$\boldsymbol{x}^{1,2} = (2, 8/3); \qquad F(\boldsymbol{x}^{1,2}) = -10,$$

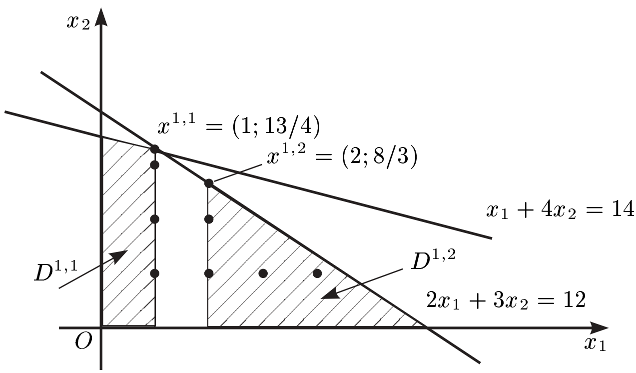
значит,

$$z\left(D^{1,1}\right) =]-43/4[= -10; \quad z\left(D^{1,2}\right) =]-10[= -10.$$

Проводим ветвление множества $D^{1,1}$: $D^{1,1} = D^{2,1} \cup D^{2,2}$, где

$$D^{2,1} = \{\boldsymbol{x} :\ \boldsymbol{x} \in D^{1,1},\ x_2 \le [13/4] = 3\};$$
$$D^{2,2} = \{\boldsymbol{x} :\ \boldsymbol{x} \in D^{1,1},\ x_2 \ge [13/4] + 1 = 4\}.$$

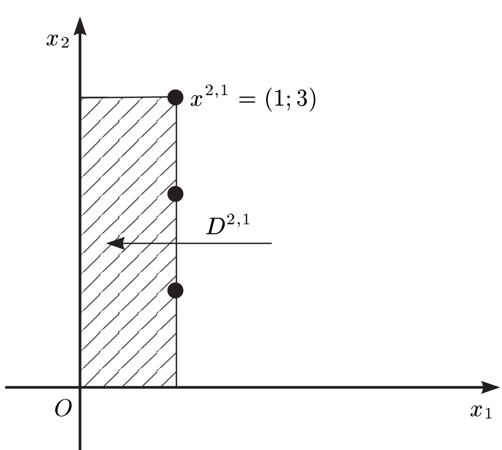
Полагаем $D^{2,3} = D^{1,2}$.



Решаем вспомогательные ЗЛП:

$$F(\boldsymbol{x}) = -x_1 - 3x_2 \to \min, \qquad x \in D^{2,1}; \tag{4}$$
$$F(\boldsymbol{x}) = -x_1 - 3x_2 \to \min, \qquad x \in D^{2,2}. \tag{5}$$

Решение задачи (4) имеет вид $\boldsymbol{x}^{2,1} = (1; 3)$; $F(\boldsymbol{x}^{2,1}) = -10$.



Задача (5) решения не имеет, поскольку множество $D^{2,2}$ пустое.

Полагаем $z(D^{2,1}) =]-10[= -10$; $z(D^{2,2}) = +\infty$. Анализируя решение ЗЛП, связанных с областями $D^{2,1}$, $D^{2,2}$ и $D^{2,3}$, и используя признак оптимальности $F(\boldsymbol{x}^{2,1}) \le z(D^{2,l})$, $l = \overline{1,3}$, заключаем, что оптимальное решение исходной целочисленной задачи имеет вид

$$\boldsymbol{x}^* = (1;3); \qquad F(\boldsymbol{x}^*) = -10.$$

Схематически решение выглядит так.

1.	$\boxed{\begin{array}{c} \boldsymbol{x} \in D^0,\ \boldsymbol{x}^0 = (6/5;16/5) \\ F(\boldsymbol{x}^0) = -54/5;\ z(F) = -10 \end{array}}$		
2.	$\boxed{\begin{array}{c} \boldsymbol{x} \in D^{1,1},\ \boldsymbol{x}^{1,1} = (1;13/4) \\ F(\boldsymbol{x}^{1,1}) = -43/4;\ z(F) = -10 \end{array}}$		$\boxed{\begin{array}{c} \boldsymbol{x} \in D^{1,2},\ \boldsymbol{x}^{1,2} = (2;8/3) \\ F(\boldsymbol{x}^{1,2}) = -10;\ z(F) = -10 \end{array}}$
3.	$\boxed{\begin{array}{c} \boldsymbol{x} \in D^{2,1}, \\ \boldsymbol{x}^{2,1} = (1;3) \\ F(\boldsymbol{x}^{2,1}) = -10, \\ z(F) = -10 \end{array}}$	$\boxed{\begin{array}{c} \boldsymbol{x} \in D^{2,2}, \\ D^{2,2} = \emptyset \\ z = +\infty \end{array}}$	$\boxed{\begin{array}{c} \boldsymbol{x} \in D^{2,3},\ D^{2,3} = D^{1,2}, \\ \boldsymbol{x}^{2,3} = (2;8/3) \\ F(\boldsymbol{x}^{2,3}) = -10, \\ z(F) = -10 \end{array}}$

В заключение отметим, что часто использовать целочисленные модели не следует, особенно когда речь идет о массовом выпуске гаек, болтов, гвоздей и т. п., где округлять результат вполне правомерно, в отличие от случая, когда выпускается мелкосерийное уникальное оборудование, техника (станки, самолеты и т. п.).

4. НЕЛИНЕЙНОЕ ПРОГРАММИРОВАНИЕ

Рассмотрим несколько примеров экономических задач, приводящих к задаче нелинейного программирования (ЗНЛП).

1. Минимизация издержек производства. Пусть необходимые затраты для производства x единиц продукта A выражаются следующим образом:

$$K(x) = ax^K, \qquad x > 0, \quad a > 0, \quad K > 1.$$

Здесь общие издержки начиная с некоторого x_0 увеличиваются быстрее, чем количество произведенных единиц продукта A (прогрессия издержек). Очевидно, что это экономически нежелательная ситуация.

Если издержки производства продукта A определены формулой

$$K(x) = ax^K, \qquad x > 0, \quad a > 0, \quad 0 < K < 1,$$

то общие затраты увеличиваются медленнее, чем количество произведенных единиц продукта A (регрессия издержек). Эта ситуация (эффект крупносерийного производства) встречается чаще всего.

Наконец, в редко встречающемся случае регрессии издержек справедливо соотношение

$$K(x) = a - bx^K, \qquad x > 0, \quad a, b, K > 0.$$

Если в общем случае затраты на производство x_j единиц продукта A_j $(j = 1, 2, \ldots, n)$ на некотором предприятии заданы в виде

$$K(x_1, x_2, \ldots, x_n) = \sum_{j=1}^{n} a_j, \qquad x_j > 0, \quad j = 1, 2, \ldots, n,$$

где a_j, $K_j > 0$, а продукты A_j должны быть произведены в количестве не менее b_j $(j = 1, 2, \ldots, n)$ и удовлетворять условию на ассортимент $\sum\limits_{j=1}^{n} d_j x_j \geq d$, где $d_j, d > 0$, то минимизация издержек на предприятии означает, что требуется найти минимум нелинейной функции $K(\boldsymbol{x})$ на множестве $G = \{(x_1, x_2, \ldots, x_n) : \sum\limits_{j=1}^{n} d_j x_j \geq d,$ $x_j \geq b_j,\ j = 1, 2, \ldots, n\}$.

2. Оптимальный выбор факторов на основе технологической производственной функции. Пусть выпуск продукции предприятия (в стоимостном или натуральном выражении) зависит от некоторых факторов (точнее их значений $x_1, x_2, \ldots, x_n$) следующим образом(технологическая производственная функция):

$$y = a x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n},$$

где $a, \alpha_1, \alpha_2, \ldots, \alpha_n$ — известные числовые параметры, которые определяются на основе статистических данных. Пусть затраты, связанные с использованием единицы каждого j-го фактора, равны c_j $(j = 1, 2, \ldots, n)$. Требуется добиться максимально возможного объема выпуска продукции y при условии, что общая стоимость факторов не превышает c.

Математическая модель задачи имеет такой вид: найти такие значения переменных $x_1, x_2, \ldots, x_n$, чтобы при $x_j \geq 0$ (количество факторов не может быть отрицательным) и $\sum\limits_{i=1}^{n} c_i x_i \leq c$ (ограничения на общие затраты на все факторы) получить максимально возможный объем выпускаемой продукции:

$$\max a x_1^{\alpha_1} x_2^{\alpha_2} \ldots x_n^{\alpha_n}.$$

В наиболее общем виде эти и подобные задачи можно сформулировать в терминах ЗНЛП, а именно: найти максимальное или минимальное значение функции $f(\boldsymbol{x})$ на заданном множестве M решений $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$. Для простоты изложения предположим, что множество M определяется конечной системой равенств и неравенств, хотя могут быть и другие способы задания этого множества.

Итак, рассмотрим следующую задачу: пусть заданы функции переменных $f(\boldsymbol{x})$, $\varphi_i(\boldsymbol{x})$, $i = 1, 2, \ldots, m$, $\psi_j(\boldsymbol{x})$, $j = 1, 2, \ldots, p$, зависящие от многих переменных, т. е. $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$. Найти

$$\min f(\boldsymbol{x})$$

при ограничениях

$$\varphi_i(\boldsymbol{x}) \le 0, \qquad i = 1, 2, \ldots, m;$$
$$\psi_j(\boldsymbol{x}) = 0, \qquad j = 1, 2, \ldots, p.$$

Если все функции f, φ_i, ψ_j линейные, то соответствующая задача является ЗЛП. Если хотя бы одна из перечисленных функций нелинейная, то соответствующая задача является ЗНЛП.

В частности, в приведенных примерах нелинейными являются целевые функции $K(\boldsymbol{x})$ и $y = f(x_1, x_2, \ldots, x_n)$; ограничения-неравенства в обоих случаях линейны; ограничения-равенства отсутствуют.

Существует несколько сравнительно простых ЗНЛП.

1. Классические задачи оптимизации. Эти задачи состоят в нахождении минимума (максимума) функции $f(\boldsymbol{x})$, $\boldsymbol{x} = (x_1, x_2, \ldots$ $\ldots, x_n)$ при ограничениях-равенствах $g_i(\boldsymbol{x}) = 0$, $i = 1, 2, \ldots, m$; $m < n$. Случай $m = 0$ соответствует задаче без ограничений. Существенным для этих задач является требование гладкости (наличие у функций f и g_i непрерывных частных производных по меньшей мере до второго порядка включительно).

Классические задачи оптимизации могут быть решены (хотя бы в принципе) классическими методами с использованием аппарата дифференциального исчисления. Однако для решения практических задач эти методы не всегда подходят в силу громоздких вычислений.

2. Задачи с нелинейной целевой функцией $f(\boldsymbol{x})$ и линейными ограничениями

$$\varphi_i(x) = \sum_{j=1}^{n} a_{ij}x_j - b_j \le 0, \qquad i = 1, 2, \ldots, m.$$

3. Задачи квадратичного программирования. В этих задачах целевая функция представляет собой квадратичную функцию

$$f(\boldsymbol{x}) = \sum_{j=1}^{n} c_j x_j + \sum_{k=1}^{n}\sum_{j=1}^{n} d_{kj} x_k x_j,$$

а ограничения имеют линейный вид:

$$\varphi_i(x) = \sum_{j=1}^{n} a_{ij}x_j - b_i \le 0, \qquad i = 1, 2, \ldots, m.$$

4. Задачи выпуклого программирования. В этих задачах целевая функция $f(\boldsymbol{x})$ и функции ограничений $g_i(\boldsymbol{x})$, $i = 1, 2, \ldots, m$ представляют собой выпуклые (вогнутые) функции.

5. Задачи с сепарабельной целевой функцией

$$f(\boldsymbol{x}) = f_1(x_1) + f_2(x_2) + \cdots + f_n(x_n)$$

и аналогичными функциями ограничений $g_i(\boldsymbol{x})$.

Как отмечалось, наиболее распространенный симплекс-метод решения ЗЛП дает точное решение за конечное количество шагов. Нелинейные задачи можно решать приближенными методами, которые позволяют находить некоторые решения оптимизационной задачи с заданной точностью.

Геометрическая интерпретация задачи нелинейного программирования

Для геометрического решения задачи математического программирования существенным является понятие о линиях уровня целевой функции.

Определение. *Множество точек* $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ *составляет линию уровня функции* $f(\boldsymbol{x})$, *если они удовлетворяют соотношению* $f(\boldsymbol{x}) = \mathrm{const}$. *Причем различным константам соответствуют различные линии уровня.*

Например, для линейной функции $f(\boldsymbol{x}) = ax_1 + bx_2$ линиями уровня будут параллельные прямые $ax_1 + bx_2 = \mathrm{const}$, где const определяет сдвиг относительно начала координат. Для квадратичной функции $f(\boldsymbol{x}) = x_1^2 + x_2^2$ линии уровня суть концентрические окружности различных радиусов $r = \sqrt{\mathrm{const}}$ с центром в начале координат. Для функции $f(\boldsymbol{x}) = (x_1 - a)^2 + (x_2 - b)^2$, очевидно, линиями уровня будут также концентрические окружности, но с центром в точке (a, b), а если рассматривать функцию $f(\boldsymbol{x}) = c(x_1 - a)^2 + d(x_2 - b)^2$, где $c > 0$, $d > 0$, то для нее линиями уровня будут концентрические эллипсы также с центром в точке (a, b).

Пример 1. Пусть $\boldsymbol{x} = (x_1, x_2) \in E_2$ и область допустимых решений D задается ограничениями

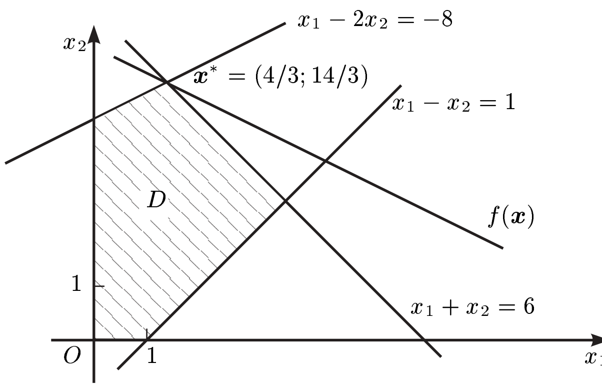
$$x_1 + x_2 \le 6, \qquad x_1 - 2x_2 \ge -8,$$
$$x_1 - x_2 \le 1, \qquad x_1 \ge 0, \quad x_2 \ge 0.$$

Найти $\max\limits_{x \in D} f(\boldsymbol{x}) = 0,5x_1 + 2x_2$.

Оптимальным решением этой задачи будет

$$\boldsymbol{x}^* = (4/3; 14/3); \quad f(\boldsymbol{x}^*) = 10.$$

Ту же задачу можно решить графически.



Пример 2. Найти минимум функции

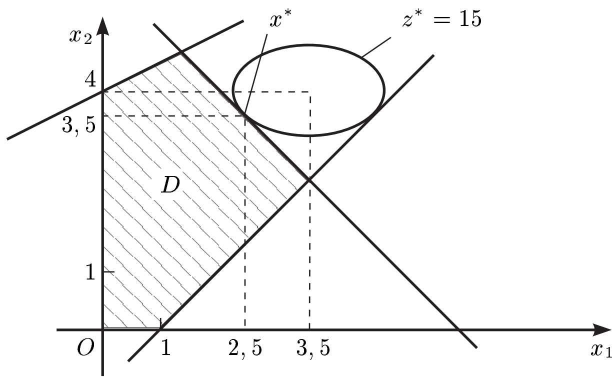
$$f(\boldsymbol{x}) = 10(x_1 - 3,5)^2 + 20(x_2 - 4)^2$$

при ограничениях, заданных в предыдущем примере.

Рассматриваемая задача представляет собой ЗНЛП с нелинейной (квадратичной) функцией цели и линейными ограничениями. Решение этой задачи, как видно из рисунка, сводится к отысканию x_1^*, x_2^* и $z^* = f(x_1^*, x_2^*)$, удовлетворяющих системе уравнений

$$x_1^* + x_2^* = 6,$$
$$f(\boldsymbol{x}^*) = z^* = 10(x_1^* - 3,5)^2 + 20(x_2^* - 4)^2,$$
$$x_2^* - 4 = 0,5(x_1^* - 3,5).$$

Эта система получена из очевидных с геометрической точки зрения фактов: оптимальное решение представляет собой точку $(x_1^*; x_2^*)$, лежащую на прямой $x_1 + x_2 = 6$ и линии уровня $f(\boldsymbol{x}^*) = z^*$;



в точке $(x_1^*; x_2^*)$ прямая $x_1 + x_2 = 6$ касается линии уровня $f(\boldsymbol{x}^*) = z^*$, откуда

$$-1 = \frac{dx_2}{dx_1} = -\frac{\partial z/\partial x_1}{\partial z/\partial x_2} = -\left.\frac{20(x_1 - 3,5)}{40(x_2 - 4)}\right|_{x_1 = x_1^*,\ x_2 = x_2^*}.$$

Решая приведенную выше систему уравнений, получаем $\boldsymbol{x}^* = (2,5; 3,5)$; $z^* = 15$.

Заметим, что оптимальное значение целевой функции не достигается в вершине допустимой области.

Классический метод оптимизации нелинейных задач с помощью множителей Лагранжа

Не обладая достоинствами ЗЛП, нелинейные оптимизационные задачи требуют привлечения более сложных методов анализа. Такие свойства функций, задействованных в постановке задач, как ограниченность, непрерывность, дифференцируемость, выпуклость и др., существенно влияют на выбор методов решения ЗНЛП.

В первую очередь следует выделить условия, при которых целевая функция достигает экстремальных значений. Известно, что непрерывная функция на замкнутом ограниченном множестве по крайней мере один раз достигает минимального и максимального значений. Причем точки экстремума являются либо стационарными, либо точками границы, либо точками, где функция недифференцируема.

Напомним, что для функции $f(x)$ стационарными являются точки, в которых ее производная равна нулю, т. е. $f'(x) = 0$. Для функции многих переменных $f(\boldsymbol{x}) = f(x_1, x_2, \ldots, x_n)$ первой производной будет вектор, составленный из частных производных по всем переменным:

$$\nabla f(\boldsymbol{x}) = \left\{ \frac{\partial f(\boldsymbol{x})}{\partial x_1}, \frac{\partial f(\boldsymbol{x})}{\partial x_2}, \ldots, \frac{\partial f(\boldsymbol{x})}{\partial x_n} \right\}.$$

Этот вектор называется градиентом функции. Чтобы определить, являются ли стационарные точки точками экстремума, необходимо исследовать в них вторую производную. Для функции $f(\boldsymbol{x}) = f(x_1, x_2, \ldots, x_n)$ это будет матрица Гессе вида

$$H = \left\| \frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} \right\|, \qquad i, j = 1, 2, \ldots, n.$$

Индексы i, j указывают место элементов в матрице (i — номер строки; j — номер столбца). Так как производная не зависит от порядка дифференцирования, то, очевидно, матрица H — является симметричной, т. е. выполняется равенство

$$\frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j} = \frac{\partial^2 f(\boldsymbol{x})}{\partial x_j \partial x_i}.$$

Определители, стоящие на главной диагонали матрицы H, обозначим

$$\Delta_1 = |f_{11}|;\ \Delta_2 = \begin{vmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{vmatrix}, \ \ldots,\ \Delta_n = \begin{vmatrix} f_{11} & f_{12} & \cdots & f_{1n} \\ f_{21} & f_{22} & \cdots & f_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ f_{n1} & f_{n2} & \cdots & f_{nn} \end{vmatrix}.$$

Здесь $f_{ij} = \dfrac{\partial^2 f(\boldsymbol{x})}{\partial x_i \partial x_j}$ — вторые частные производные функции $f(x_1,\, x_2, \ldots, x_n)$, вычисленные в точке $\boldsymbol{x}^0 = (x_1^0, x_2^0, \ldots, x_n^0)$.

Если определители имеют один знак (а именно положительные), то $\boldsymbol{x}^0$ — точка относительного (локального) минимума. Если же $\Delta_1,\, \Delta_2, \ldots, \Delta_n$ поочередно меняют знак начиная с минуса ($\Delta_1 < 0$, $\Delta_2 > 0$, $\Delta_3 < 0$ и т. д.), то $\boldsymbol{x}^0$ — точка относительного максимума. До тех пор, пока исследования проводятся в отдельных точках, речь идет именно о локальных (местных) экстремумах. Проанализировав всю допустимую область решений, можно выделить среди локальных экстремумов наибольший и наименьший, которые будут называться глобальными.

Пример 3. Пусть $f(\boldsymbol{x}) = f(x_1, x_2) = 10x_1 + 20x_2 + x_1x_2 - 2x_1^2 - 2x_2^2$ определена на всем пространстве R^2. Для вычисления относительного экстремума этой функции имеем два уравнения

$$\frac{\partial f}{\partial x_1} = 10 + x_2 - 4x_1 = 0, \qquad \frac{\partial f}{\partial x_2} = 20 + x_1 - 4x_2 = 0,$$

решив которые, найдем $x_1^0 = 4$, $x_2^0 = 6$.

Матрица Гессе для этой функции имеет вид

$$H = \begin{Vmatrix} -4 & 1 \\ 1 & -4 \end{Vmatrix} \quad \text{и} \quad \Delta_1 = -4 < 0, \quad \Delta_2 = \begin{vmatrix} -4 & 1 \\ 1 & -4 \end{vmatrix} = 15 > 0.$$

Следовательно, точка $\boldsymbol{x}^0 = (4; 6)$ является точкой относительного максимума. Поскольку $f(\boldsymbol{x})$ в области определения других стационарных точек не имеет, то $\boldsymbol{x}^0$ является также точкой глобального максимума.

Для задачи с ограничениями классический метод поиска условного экстремума состоит в следующем.

1. Находят множество $S_1(\boldsymbol{x})$ всех стационарных точек $f(\boldsymbol{x})$ внутри допустимого множества D, которые исследуют на минимум (максимум) и определяют среди них наименьший (наибольший).

2. Исследуют точки границы $S_2(\boldsymbol{x})$ и находят те из них, где функция $f(\boldsymbol{x})$ достигает минимума (максимума). Для этого выбирают произвольную границу, например $g_1(\boldsymbol{x}) = 0$. Если это уравнение можно решить относительно компонент точки $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, то подставив их в выражение для целевой функции, приходят к безусловной задаче оптимизации, т. е. к задаче без ограничений. Такой прямолинейный подход требует значительных вычислительных затрат и применим лишь в простейших случаях, а именно: при небольшом количестве ограничений $g_i(\boldsymbol{x}) = 0$, $i = = 1, 2, \ldots, m$, причем таких, что каждое уравнение $g_i(\boldsymbol{x}) = 0$ легко может быть разрешено относительно всех переменных $x_1, x_2, \ldots, x_n$, т. е. можно получить явные функции вида

$$x_k = \psi_k(x_1, \ldots, x_{k-1}, x_{k+1}, \ldots, x_n), \qquad k = 1, 2, \ldots, n.$$

В частности, это возможно, если функции $g_i(\boldsymbol{x})$ линейны. В других ситуациях для вычисления точек экстремума функции $f(\boldsymbol{x})$ на границе допустимого множества применяются более эффективные методы.

Метод Лагранжа используется для решения ЗНЛП следующего вида:
найти
$$f(\boldsymbol{x}) \to \min \tag{4.1}$$
на ограниченном множестве
$$D = \{x: \ g_i(\boldsymbol{x}) = 0, \ i = 1, 2, \ldots, m\}. \tag{4.2}$$

Если функции $f(\boldsymbol{x})$, $g_i(\boldsymbol{x})$ непрерывно дифференцируемы, то для задачи (4.1)–(4.2) строится функция Лагранжа
$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i g_i(\boldsymbol{x}).$$

Пусть в некоторой точке $\boldsymbol{x}^0 = (x_1^0, x_2^0, \ldots, x_n^0)$ существует вектор $\boldsymbol{\lambda}^0 = (\lambda_1^0, \lambda_2^0, \ldots, \lambda_m^0)$ такой, что выполняется необходимое условие экстремума функции $L(\boldsymbol{x}, \boldsymbol{\lambda})$ вида
$$\frac{\partial L(\boldsymbol{x}^0, \boldsymbol{\lambda}^0)}{\partial x_j} = 0, \qquad j = 1, 2, \ldots, n;$$
$$\frac{\partial L(\boldsymbol{x}^0, \boldsymbol{\lambda}^0)}{\partial \lambda_i} = 0, \qquad i = 1, 2, \ldots, m,$$
тогда $\boldsymbol{x}^0$ является решением задачи (4.1) с ограничениями (4.2).

Пример 4. Имеется два способа производства некоторого продукта. Пусть x_1, x_2 — количества, произведенные соответствующим способом. Издержки производства зависят от произведенных количеств x_1 и x_2 следующим образом:
$$H_1(x_1) = a_0 + a_1 x_1 + a_2 x_1^2, \qquad a_0, a_1, a_2 > 0;$$
$$H_2(x_2) = b_0 + b_1 x_2 + b_2 x_2^2, \qquad b_0, b_1, b_2 > 0.$$

За некоторый промежуток времени необходимо произвести c единиц продукции (т. е. $c = x_1 + x_2$), распределив ее для двух способов так, чтобы минимизировались общие издержки.

В терминах общей постановки задача имеет такой вид:
найти
$$f(\boldsymbol{x}) = H_1(x_1) + H_2(x_2) \to \min$$

при условии

$$g_1(\boldsymbol{x}) = c - x_1 - x_2 = 0.$$

При этом функция Лагранжа приобретает вид

$$L(x_1, x_2, \lambda) = a_0 + a_1 x_1 + a_2 x_1^2 + b_0 + b_1 x_2 + b_2 x_2^2 + \lambda(c - x_1 - x_2),$$

откуда

$$\frac{\partial L}{\partial x_1} = a_1 + 2a_2 x_1 - \lambda = 0,$$
$$\frac{\partial L}{\partial x_2} = b_1 + 2b_2 x_2 - \lambda = 0,$$
$$\frac{\partial L}{\partial \lambda} = c - x_1 - x_2 = 0.$$

Решив эту систему, находим количества

$$x_1^0 = \frac{b_2}{a_2 + b_2} c + \frac{b_1 - a_1}{2(a_2 + b_2)}; \qquad x_2^0 = \frac{a_2}{a_2 + b_2} c - \frac{b_1 - a_1}{2(a_2 + b_2)}.$$

Элементы выпуклого анализа

При изучении задач математического программирования важное значение имеет такое свойство функции, как выпуклость (вогнутость). Задачи, в которых целевая функция является выпуклой (вогнутой) и допустимое множество решений — выпуклым множеством, называются задачами выпуклого программирования (ЗВП). Преимущество таких задач заключается в том, что всегда существует минимум (максимум) целевой функции и он является абсолютным (глобальным).

Определение 1. *Множество X называется выпуклым, если любые две его точки можно соединить отрезком, который полностью принадлежит этому множеству или, что то же самое, для любых его точек $\boldsymbol{x}^1$, $\boldsymbol{x}^2$ и произвольного числа λ: $0 < \lambda < 1$ точка $\boldsymbol{y} = \lambda \boldsymbol{x}^1 + (1 - \lambda)\boldsymbol{x}^2$ также принадлежит этому множеству.*

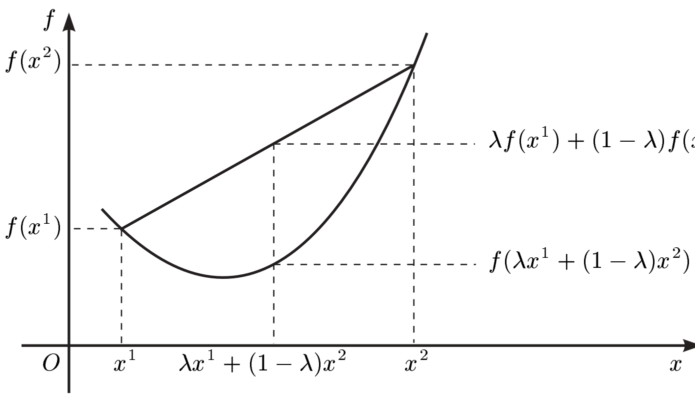
Выпуклыми множествами являются, например, отрезок, луч, прямая, шар.

Определение 2. *Функция $f(\boldsymbol{x})$, определенная на некотором выпуклом множестве X, называется выпуклой, если для любых то-*

чек $\boldsymbol{x}^1$ *и* $\boldsymbol{x}^2$ *из множества* X *и произвольного числа* λ*:* $0 < \lambda < 1$ *выполняется неравенство*

$$f\left(\lambda \boldsymbol{x}^1 + (1-\lambda)\boldsymbol{x}^2\right) \le \lambda f(\boldsymbol{x}^1) + (1-\lambda) f(\boldsymbol{x}^2).$$

В частности, для функции одной переменной это неравенство означает, что отрезок, соединяющий произвольную пару точек ее графика, ни в каких других точках этот график не пересекает.



Если функция $f(\boldsymbol{x})$ выпуклая, то $-f(\boldsymbol{x})$ — вогнутая. Если функция $g(\boldsymbol{x})$ выпуклая, то множество W, заданное неравенством $g(\boldsymbol{x}) \le$ ≤ 0, является выпуклым. Пересечение выпуклых множеств, т. е. множество, заданное системой ограничений вида $W = \{\boldsymbol{x} : g_i(\boldsymbol{x}) \le 0,$ $i = 1, 2, \ldots, m\}$, где все функции $g_i(\boldsymbol{x})$ выпуклые, является либо пустым, либо выпуклым и замкнутым множеством.

В наиболее общей постановке задача выпуклого программирования имеет такой вид:
найти

$$\min f(\boldsymbol{x}) \tag{4.3}$$

при ограничениях

$$g_i(\boldsymbol{x}) \le 0, \qquad i = 1, 2, \ldots, m, \tag{4.4}$$

где функции $f(\boldsymbol{x})$ и $g_i(\boldsymbol{x})$, $i = 1, 2, \ldots, m$ — выпуклые функции многих переменных $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$.

Если множество допустимых решений ЗВП (4.3), (4.4) пустое, то задача не имеет решений, а если оно неограниченное, то не существует оптимальное решение. На замкнутом ограниченном множестве задача выпуклого программирования обладает следующим очень важным свойством, которое облегчает поиск решения.

Теорема. *Любой локальный минимум задачи выпуклого программирования является глобальным.*

Приведем теорему Куна — Таккера, которую можно рассматривать как обобщение теоремы двойственности в линейном программировании, а множители Лагранжа $\{\lambda_i, i = 1, 2, \ldots, m\}$ являются допустимыми решениями двойственной ЗЛП.

Теорема Куна — Таккера. *Пусть множество $W = \{\boldsymbol{x} : g_i(\boldsymbol{x}) \le 0,$ $i = 1, 2, \ldots, m\}$ имеет внутренние точки, т. е. выполняется условие Слейтера: существует $\boldsymbol{x} \in W$, что $g_i(\boldsymbol{x}) < 0$ для всех $i = 1, 2, \ldots, m$. Тогда для того чтобы $\boldsymbol{x}^*$ было оптимальным решением задачи (4.3)–(4.4), необходимо и достаточно, чтобы существовал неотрицательный m-мерный вектор $\boldsymbol{\lambda}^* = \{\lambda_i^*\}_{i=1}^m$ такой, что пара $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$ является седловой точкой функции Лагранжа*

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i g_i(\boldsymbol{x}), \qquad \boldsymbol{x} \in E_n, \quad \lambda_i \ge 0,$$

т. е. выполняются неравенства

$$f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i g_i(\boldsymbol{x}) \ge f(\boldsymbol{x}^*) + \sum_{i=1}^{m} \lambda_i g_i(\boldsymbol{x}^*) \ge f(\boldsymbol{x}^*) + \sum_{i=1}^{m} \lambda_i^* g_i(\boldsymbol{x}^*),$$
$$\lambda_i \ge 0, \quad i = 1, 2, \ldots, m.$$

Наиболее общие методы решения нелинейных задач оптимизации

Если в некоторых случаях классические методы анализа нелинейных оптимизационных задач не дают приемлемых результатов, то применяются численные методы решения, которые представляют собой вычислительные процедуры, постепенно “улучшающие” значение целевой функции. Наиболее эффективные методы разработаны для решения ЗВП, но и для других классов нелинейных задач существующие численные методы позволяют за конечное количество шагов получить достаточно хороший результат.

С помощью численных методов решения ЗНЛП из любой точки допустимой области можно приблизиться к некоторому локальному (а для ЗВП — глобальному) минимуму целевой функции. Поскольку эти вычисления носят циклический характер (многократно повторяющиеся однотипные действия), применять ЭВМ особенно эффективно.

Если задача математического программирования не имеет ограничений, т. е. отсутствуют условия (4.2), то поиск экстремума ведется на всем пространстве R^n (напомним, что R^1 — числовая ось; R^2 — плоскость и т. д.).

Удачно выбирая направление поиска, можно с достаточной точностью прийти к экстремальной точке за конечное количество шагов (итераций). В частности, для дифференцируемой целевой функции градиент указывает направление наибольшего возрастания функции, антиградиент (градиент с противоположным знаком) — направление убывания функции (*градиентные методы*). Если к тому же целевая функция имеет и вторые производные по всем переменным, то направление движения можно уточнить с помощью матрицы Гессе (вторых производных) (*метод Ньютона*). Для недифференцируемых функций перспективные направления выбираются среди направлений координатных осей (*метод покоординатного спуска*) или направлений, полученных случайным образом (*метод случайного поиска*). Наличие ограничений существенно усложняет методы решения нелинейных оптимизационных задач. Для того чтобы вычислительные процедуры не выходили за пределы допустимой области, необходимо решить вспомогательные задачи. Например, если направление градиента (антиградиента) является недопустимым (выходит за пределы допустимой области), требуется операция проектирования градиента на допустимую область, а это достаточно сложная задача, особенно при нелинейных ограничениях (*метод проекции градиента*).

Если поиск ведется в некоторых возможных направлениях, не выходящих за пределы допустимой области, то для выбора наилучших среди них необходимо решить вспомогательную ЗЛП (*метод возможных направлений*).

Идея *методов штрафных и барьерных функций* заключается в том, чтобы определенным образом включить ограничения в целевую функцию и рассматривать вместо исходной задачи задачу без ограничений, решить которую намного проще.

Квадратичное программирование

Одним из наиболее простых и важных классов ЗНЛП являются задачи квадратичного программирования, которые заключаются в минимизации (максимизации) квадратичной функции при линейных ограничениях. В векторной форме эта задача имеет такой вид:
найти

$$\min[(D\boldsymbol{x}, \boldsymbol{x}) + (\boldsymbol{c}, \boldsymbol{x})]$$

при ограничениях

$$A\boldsymbol{x} \leq \boldsymbol{b},$$

где D — симметричная матрица размерности $n \times n$; $\boldsymbol{x}$ — неизвестный n-мерный вектор; $\boldsymbol{c}$, $\boldsymbol{b}$ — векторы соответственно n- и m-мерные; A — матрица ограничений размерности $m \times n$.

Например, для матриц

$$D = \begin{pmatrix} 1 & 2 \\ 2 & 3 \end{pmatrix}, \qquad A = \begin{pmatrix} -1 & 1 \\ 2 & 0 \\ 1 & -1 \end{pmatrix}$$

и векторов $\boldsymbol{c} = (4, 2)$, $\boldsymbol{b} = (2, 3, 1)$, $\boldsymbol{x} = (x_1, x_2)$ в координатной форме задача будет иметь такой вид:
найти

$$\min(x_1^2 + 4x_1x_2 + 2x_2^2 + 4x - 1 + 2x_2)$$

при ограничениях

$$\begin{aligned} -x_1 + x_2 &\leq 2, \\ 2x_1 \quad\quad &\leq 3, \\ x_1 - x_2 &\leq 1, \end{aligned}$$

так как

$$D\boldsymbol{x} = \begin{pmatrix} 1 & 2 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_1 + 2x_2 \\ 2x_1 + 3x_2 \end{pmatrix};$$

$$(D\boldsymbol{x}, \boldsymbol{x}) = (x_1 + 2x_2)x_1 + (2x_1 + 3x_2)x_2 = x_1^2 + 4x_1x_2 + 3x_2^2;$$

$$A\boldsymbol{x} = \begin{pmatrix} -1 & 1 \\ 2 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -x_1 + x_2 \\ 2x_1 \\ x_1 - x_2 \end{pmatrix}.$$

Способы решения задачи квадратичного программирования во многом определяются видом матрицы D: если D — неотрицательно определенная матрица, то это ЗВП, и любой локальный минимум этой задачи будет ее решением; если D — неположительно определенная матрица, то имеем задачу вогнутого программирования; при этом может быть большое количество неэквивалентных локальных минимумов, но глобальный минимум, если он существует, достигается в одной из вершин допустимого многогранного множества M, определенного системой ограничений $A\boldsymbol{x} \leq \boldsymbol{b}$. Большинство известных процедур точного решения задач вогнутого программирования в той или иной степени предполагает перебор вершин множества M. Эти методы по структуре напоминают методы дискретного программирования.

Еще более сложный случай, когда собственные числа матрицы D имеют разные знаки. Это фактически многоэкстремальная задача, но глобальный минимум может не достигаться в вершине многогранника M. В этом случае решение задачи, если оно существует, является стационарной точкой некоторой грани многогранного множества M (включая и нульмерные грани-вершины). Этих граней конечное множество, и нахождение стационарной точки квадратичной формы сводится к решению системы линейных уравнений, т. е. и в этом случае решение можно получить за конечное количество операций.

5. ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

Часто возникают ситуации, когда решение можно принять только в несколько приемов — шаг за шагом. В этом случае необходимо привлекать аппарат динамического программирования.

Одна из таких типичных задач — распределение капитальных вложений, когда на каждом этапе (например, ежегодно) решается задача распределения. Требуется установить такое распределение капитальных ресурсов на каждом этапе, при котором суммарная прибыль за все этапы будет максимальной.

Решение задач динамического программирования базируется на использовании принципа оптимальности, сформулированного математиком Р. Беллманом: *если некоторая последовательность решений оптимальна, то отдельные последующие решения внутри ее оптимальны по отношению к предыдущим решениям* или, другими словами, *оптимальная стратегия решения многоэтапной задачи состоит из оптимальных решений для каждого отдельного этапа.* В соответствии с этим принципом задачу решают начиная с последнего этапа. Затем переходят к решению предыдущего этапа, отбрасывая возможные альтернативы, которые противоречат уже отработанному решению.

Первыми к появлению динамического программирования привели динамические задачи управления запасами (ЗУЗ). Они составляют один из наиболее многочисленных классов экономических задач. Правильное и своевременное определение оптимальной стратегии управления запасами, а также их нормативного уровня позволяет высвободить значительные оборотные средства, замороженные в виде запасов, что в конечном итоге повышает эффективность используемых ресурсов.

Пример 1. Пусть x — количество капитальных вложений, которые необходимо распределить между двумя отраслями. Количество капитальных средств y, вложенное в первую отрасль, за год приносит прибыль $g(y) = 0,8y$. Доля капитальных средств $x-y$, вложенных во вторую отрасль, приносит за год доход $h(x-y) = 0,5(x-y)$. К концу года средства, вложенные в первую отрасль, составят $a(y) = 0,3y$, во вторую — $b(x-y) = 0,6(x-y)$. По истечении каждого года оставшиеся капитальные вложения заново распределяются по отраслям. Необходимо установить распределение так, чтобы суммарная прибыль за три года была максимальной.

Итак, сначала ищем величину y_2 — количество средств, вложенных в течение третьего года, затем y_1 и y_0 (соответственно второго и первого годов). В соответствии с принципом оптимальности, зная зависимость от величин y_1 и y_0, а также x_2 — количество ресурсов, полученных к началу третьего года, необходимо наилучшим образом использовать доступное количество ресурсов, т. е. решить задачу

$$f_1(x_2) = \max_{0 \le y_2 \le x_2} \{g(y_2) + b(x_2 - y_2)\}$$

или

$$f_1(x_2) = \max_{0 \le y_2 \le x_2} \{0,8y_2 + 0,5(x_2 - y_2)\} = \max_{0 \le y_2 \le x_2} \{0,5x_2 + 0,3y_2\}.$$

Очевидно, максимум возможен при $x_2 = y_2$.

Следовательно, все ресурсы на последнем этапе нужно направить в первую отрасль. При этом будет получена прибыль $f_1(x_2) = 0,8x_2$. Теперь найдем y_1. Рассмотрим двухшаговый процесс — последний и предпоследний этапы. Необходимо наилучшим образом использовать ресурсы x_1, не интересуясь предыдущим решением. Максимальный суммарный доход на последних двух этапах

$$f_2(x_1) = \max_{0 \le y_1 \le x_1} \{g(y_1) + h(x_1 - y_1) + f_1[a(y_1) + b(x_1 - y_1)]\},$$

где $g(y_1) + h(x_1 - y_1)$ — прибыль на предпоследнем этапе при y_1; $f_1[a(y_1) + b(x_1 - y_1)]$ — максимальная прибыль на последнем этапе.

Необходимо выбрать $y_1 \in [0, x_1]$ такое, чтобы $f_2(x_1)$ была максимальной.

По известному f_1 получаем

$$f_2(x_1) = \max_{0 \le y_1 \le x_1} \{0,8y_1 + 0,5(x_1 - y_1) + f_1[0,3y_1 + 0,6(x_1 - y_1)]\} =$$
$$= \max_{0 \le y_1 \le x_1} \{0,5x_1 + 0,3y_1 + 0,8[0,6x_1 - 0,3y_1]\} =$$
$$= \max_{0 \le y_1 \le x_1} \{0,98x_1 + 0,06y_1\}.$$

Отсюда $f(x_1) = 1,04x_1$; $y_1 = x_1$.

Значит, на предпоследнем этапе все капитальные вложения также должны быть направлены в первую отрасль.

Аналогично для первого этапа

$$f_3(x) = \max_{0 \le y \le x} \{g(y) + h(x - y) + f_2[a(y) + b(x - y)]\} =$$
$$= \max_{0 \le y \le x} \{0,8y + 0,5(x - y) + 1,04[0,3y + 0,6(x - y)]\} =$$
$$= \max_{0 \le y \le x} \{1,124x - 0,012y\}.$$

Здесь максимум достигается при $y = 0$. Следовательно, наибольший суммарный доход на всех этапах $f_3(x) = 1,124x$. На первом этапе все ресурсы необходимо направить во вторую отрасль: $y = 0$.

Таким образом, получена оптимальная стратегия $y = 0$; $y_1 = x_1$; $y_2 = x_2$.

Первый этап: все ресурсы — во вторую отрасль; прибыль составит $h(x) = 0,5x$. К концу года останется $b(x) = 0,6x$ капитальных вложений, которые будут направлены в первую отрасль и дадут прибыль $g(0,6x) = 0,8 \cdot 0,6x = 0,48x$. Остаток ресурсов при этом $a(0,6x) = 0,3 \cdot 0,6x = 0,18x$.

Прибыль на третьем этапе $g(0,18x) = 0,8 \cdot 0,18x = 0,144x$; тогда суммарная прибыль составит $0,5x + 0,48x + 0,144x = 1,124x$.

Формально задача динамического программирования имеет такой вид:

найти

$$\max_{x_1,\dots,x_n} z = \sum_{j=1}^{n} f_j(x_j)$$

при условиях

$$\sum a_j x_j \le b, \qquad a_j > 0, \quad x_j \ge 0, \quad j = 1, 2, \dots, n.$$

Целевая функция задачи является суммой функций от одной переменной. Такая функция называется сепарабельной, или аддитивной.

6. ЗАДАЧИ И МЕТОДЫ СТОХАСТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Детерминированные модели математического программирования часто оказываются неадекватными реальным процессам в экономике. Это объясняется прежде всего неточностью и вероятностным характером тех показаний и ограничений, которые вкладываются в модели. Планирование на основе наиболее оптимистических прогнозов, факторов, обусловливающих производство и потребление, зачастую оказывается несостоятельным из-за отсутствия резервов для коррекции неувязок, которые возникают, если какое-либо звено в цепи производства дает меньше, чем ожидалось по оптимистичному прогнозу. Намного целесообразнее планировать по усредненным показателям, если их разброс достаточно велик. И в этом случае реальные значения показателей могут значительно отличаться от средних значений, и предложенный план окажется непригодным.

Для решения задач планирования в условиях неопределенности разработаны специальные математические модели и методы — ***стохастическое программирование***.

Задача. Пусть предприятие выпускает m видов продукции, используя n видов покупных изделий и материалов; при этом заявка на поставку этих изделий оформляется до того, как точно известны заказы потребителей на продукцию, производимую предприятием, имеется лишь вероятностный прогноз спроса потребителей. После того как спрос потребителей уточняется, составляется план производства с учетом следующих возможностей:

1) допускается дополнительная закупка покупных изделий по высоким ценам;

2) допускается отказ от части заявленных закупок, связанный с определенным штрафом;

3) спрос потребителей может быть удовлетворен не полностью, но при этом предприятие также несет дополнительные потери в виде штрафов.

Пусть a_{ij} — норма расхода изделия (материала) j-го типа на производство единицы продукции i-го вида; x_j — количество заявленного (ресурса, материала) j-го типа; c_j — его цена; y_j — количество дополнительно покупаемого ресурса; d_j — его цена; z_j — величина части ресурса j-го типа, от которой отказываются; f_j — штраф за отказ от единицы ресурса; w_i — планируемый выпуск продукции i-го типа; $\{\xi_i\}_{i=1}^m$ — случайный вектор спроса на продукцию, производимую предприятием; b_i — доход от реализации единицы продукции i-го типа; g_i — штраф за недопоставку единицы ресурса i-го типа.

Подобная задача решается в два этапа. На первом этапе выбирается вектор $\{x_j\}_{j=1}^n$ — заявки на необходимые производству ресурсы, на втором этапе при известном $\{\xi_i\}_{i=1}^m$ выбираются переменные $\{y_j\}_{j=1}^n$; $\{z_j\}_{j=1}^n$; $\{w_i\}_{i=1}^m$ — определяющие способы использования заказанных материалов и план производства.

Такие задачи оптимального планирования называются *двухэтапными задачами стохастического программирования.* Чтобы построить адекватную описанной задаче математическую модель, рассматривают сначала задачу второго этапа при фиксированном плане первого этапа $\{x_j\}_{j=1}^n$. Построенная здесь модель зависит не только от $\{x_j\}$, но и от случайного вектора $\{\xi_i(\omega)\}$. В этом случае необходимо рассматривать математическое ожидание (M_ω) функции-критерия, т. е.

$$\Phi(\{x_j\}) = M_\omega \varphi(\{x_j\},\ \{\xi_i(\omega)\}).$$

Задача первого этапа состоит в том, чтобы выбрать план $\boldsymbol{x} = \{x_j\}$, позволяющий максимизировать $\Phi(\boldsymbol{x})$. Хотя подобная задача и является ЗВП, однако неявный вид функции $\Phi(\boldsymbol{x})$ (как математического ожидания) создает значительные трудности в вычислении. Только если множество значений случайного вектора конечно, та двухэтапная линейная задача стохастического программирования сводится к ЗЛП специального вида.

В общем случае линейная стохастическая двухэтапная задача формулируется следующим образом.

Заданы матрица A размера $m \times n_1$, семейство матриц D_ξ размера $n \times n_2$, зависящих от случайного параметра ξ, вектор $\boldsymbol{c}$ и семейства векторов $\boldsymbol{c}_\xi^1$, $\boldsymbol{b}_\xi$ размерности соответственно n_1, n_2 и m. Для каждого фиксированного ξ рассматривается задача

$$\min(\boldsymbol{c}_\xi^1, \boldsymbol{y})$$

при ограничениях

$$D_\xi \boldsymbol{y} \geq \boldsymbol{b}_\xi - A\boldsymbol{x}.$$

Пусть оптимальное решение этой задачи $\boldsymbol{y}_\xi^*(\boldsymbol{x})$ существует при любом ξ. Требуется определить

$$\min_{\boldsymbol{x}\geq 0}[(\boldsymbol{c}, \boldsymbol{x}) M_\xi(\boldsymbol{c}_\xi^1, \boldsymbol{y}_\xi^*(\boldsymbol{x}))],$$

где ξ — случайный параметр; M_ξ — символ математического ожидания.

Пусть $M_\xi(\boldsymbol{c}_\xi^1, \boldsymbol{y}_\xi^*(\boldsymbol{x})) = F(\boldsymbol{x})$. Доказано, что функция $F(\boldsymbol{x})$ является выпуклой при $\boldsymbol{x} \geq 0$. Таким образом, линейная стохастическая двухэтапная задача сводится к задаче минимизации выпуклой функции

$$f(\boldsymbol{x}) = (\boldsymbol{c}, \boldsymbol{x}) + F(\boldsymbol{x})$$

при ограничении $\boldsymbol{x} \geq 0$.

Если количество реализаций случайного вектора ξ конечно, то получаем ЗЛП блочной структуры.

К ЗНЛП сводятся также задачи стохастического программирования, в которых решение принимается за один этап без учета конкретной реализации случайного параметра ("состояния природы"). как правило, в таких задачах целевая функция и ограничения понимают в некотором усредненном по случайному параметру смысле. Основная трудность при решении таких задач состоит в том, что целевую функцию и функцию ограничений в явном виде можно получить лишь в исключительных случаях, а можно только непосредственно вычислить значения этих функций и их градиентов при фиксированных $\boldsymbol{x}$ и ω.

Важным подклассом являются задачи с вероятностными ограничениями. Такие модели используются в тех случаях, когда можно потребовать не абсолютного выполнения тех или иных ограничений с определенной вероятностью.

Наиболее простой и реальный путь решения задач стохастического программирования связан с применением *методов обобщенных стохастических градиентов.* Эти методы являются разновидностью метода случайного поиска, когда направление движения на каждой итерации определяется с помощью некоторого случайного или псевдослучайного процесса.

7. МАТРИЧНЫЕ ИГРЫ

Матричная игра предполагает наличие двух игроков (конкурентов) J_1, J_2 и строится по следующим правилам. Первый игрок выбирает один вариант из m возможных. Пусть это будет вариант i ($i = 1, 2, \dots, m$). Второй игрок может выбрать один из n вариантов, например j ($j = 1, 2, \dots, n$). Варианты i и j каждый игрок выбирает самостоятельно. Результаты всех возможных ходов обоих игроков заносятся в так называемую платежную матрицу, или матрицу выигрышей (второго игрока) $C = \{c_{ij}\}$. Смысл элементов этой матрицы состоит в том, что c_{ij} указывает сумму выигрыша второго игрока при выборе вариантов i и j игроками соответственно J_1 и J_2. Если $c_{ij} > 0$, то первый игрок платит второму, а при $c_{ij} < 0$ наоборот.

Пример 1 (угадывание монет). Каждый из двух игроков выбирает определенную сторону монеты, затем игроки одновременно называют свой выбор. Если выбраны разные стороны, то первый игрок платит второму одну денежную единицу, если же выбраны одинаковые стороны, то второй платит первому одну денежную единицу. Для этой игры платежная матрица имеет вид $C = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$.

Пусть в векторе $\boldsymbol{x} = (x_1, \dots, x_n)$ содержатся вероятности выбора первым игроком каждого из m доступных ему вариантов, причем $\sum\limits_{i=1}^{m} = 1$, $0 \le x_i \le 1$, а в векторе $\boldsymbol{y} = (y_1, \dots, y_n)$ — вероятности выбора каждого из n вариантов, доступных второму игроку, т. е. имеют место соотношения $\sum\limits_{j=1}^{n} y_j = 1$, $0 \le y_j \le 1$.

Если для некоторой стратегии $\boldsymbol{x} = (x_1, \dots, x_i, \dots, x_m)$ выполняется $x_i = 1$, а остальные $x_k = 0$ ($k = 1, 2, \dots, m$, $k \ne i$), то эта стратегия называется i-й чистой стратегией игрока J_1. Аналогично определяется j-я чистая стратегия игрока J_2. В остальных случаях стратегии называются смешанными.

Оптимальные чистые стратегии

Рассмотрим матричную игру двух лиц с платежной матрицей $C = \{c_{ij}\}$; $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$ с точки зрения второго игрока. Так как второй игрок выбирает j-й вариант, это означает, что все возможные его выигрыши — в j-м столбце матрицы C, т. е. это $(c_{1j}, \ldots, c_{mj})$. Для того чтобы сделать выигрыш максимальным, он должен выбрать такой вариант j, чтобы максимизировать наименьший элемент в этом столбце. При этом гарантированный выигрыш второго игрока (нижняя цена игры)

$$\underline{v} = \max_{1 \le j \le n} \min_{1 \le i \le m} c_{ij}.$$

Эту же игру можно рассмотреть с точки зрения игрока J_1. Очевидно, если C является матрицей выигрыша второго игрока, то матрица $-C = \{-c_{ij}\}, i = 1, 2, \ldots, m; j = 1, 2, \ldots, n$ — матрицей выигрышей первого игрока. Рассуждая, как и ранее, приходим к выводу, что гарантированный выигрыш первого игрока

$$\max_{1 \le i \le m} \min_{1 \le j \le n} (-c_{ij}) = -\min_{1 \le i \le m} \max_{1 \le j \le n} c_{ij}.$$

При этом второй игрок получит наибольшее

$$\overline{v} = \min_{1 \le i \le m} \max_{1 \le j \le n} c_{ij}.$$

Величина $\overline{v}$ представляет собой верхнюю цену игры. Итак, игрок J_2 может гарантировать по меньшей мере $\underline{v}$, а игрок J_1 может помешать ему получить больше $\overline{v}$. Если окажется, что $v = \overline{v} = \underline{v}$ или

$$v = \min_i \max_j c_{ij} = \max_j \min_i c_{ij}, \qquad (*)$$

то игрок J_2 должен понять, что он может получить v, а его противник помешает ему получить больше v. Поэтому числа i^* и j^* такие, что в соотношении $(*)$ выполняется $v = (c_{i^* j^*})$; естественно назвать оптимальными решениями игроков соответственно J_1 и J_2. В этом случае v — цена игры, а сама игра допускает решение в чистых стратегиях. Векторы

$$\underbrace{(\overbrace{0,\ldots,0,1}^{i^*},0,\ldots,0)}_{m}, \qquad \underbrace{(\overbrace{0,\ldots,0,1}^{j^*},0,\ldots,0)}_{n}$$

называются оптимальными чистыми стратегиями игроков соответственно J_1 и J_2.

Оказывается, что соотношение $(*)$ выполняется не для каждой игры, определяемой платежной матрицей C. В этой связи не всякая матричная игра разрешима в чистых стратегиях. Например, для игры "угадывание монет" заданной платежной матрицей является $C = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$; нетрудно видеть, что

$$\underline{v} = \max_j \min_i c_{ij} = -1, \quad \overline{v} = \min_i \max_j c_{ij} = 1, \quad \underline{v} \neq \overline{v},$$

и поэтому данная игра неразрешима в чистых стратегиях.

При ответе на вопрос о разрешимости матричной игры в чистых стратегиях основное значение имеет следующее утверждение.

Теорема. *Матричная игра двух лиц с платежной матрицей $C = \{c_{ij}\}$, $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$ разрешима в чистых стратегиях (выполняется соотношение $(*)$) тогда и только тогда, когда матрица C имеет седловую точку. При этом если (i^*, j^*) — седловая точка матрицы C, то цена игры $v = c_{i^*j^*}$.*

Если платежная матрица C имеет седловую точку, то говорят, что сама игра имеет седловую точку.

Напомним, что (i^*, j^*) — седловая точка матрицы C, если для всех указанных i и j

$$c_{i^*j} \le c_{i^*j^*} \le c_{ij^*}.$$

Пример 2. Матрица C выигрышей игрока J_2 имеет такой вид:

$$\begin{array}{cc|c} & & \max\limits_j c_{ij} \\ C = & \begin{pmatrix} 50 & 27 & 64 \\ 50 & 5 & 90 \\ 18 & 9 & 12 \\ 25 & 95 & 20 \end{pmatrix} & \begin{array}{c} 64 \\ 90 \\ 18 \\ 95 \end{array} \\ \hline \min\limits_i c_{ij} & \begin{array}{ccc} 18 & 5 & 12 \end{array} & \end{array}$$

В дополнительной нижней строке вычисляются величины $\min\limits_i c_{ij}$, а в дополнительном правом столбце — величины $\max\limits_j c_{ij}$ соответственно для каждого фиксированного j и i.

Для рассматриваемой игры соотношение $(*)$ выполняется при $i^* = 3$, $j^* = 1$; при этом цена игры $v = 18$. Если игрок J_1 отступит от своей оптимальной стратегии, соответствующей $i^* = 3$, а игрок J_2 будет придерживаться своей оптимальной стратегии $j^* = 1$, то первый игрок проиграет более 18 ед. Аналогично если игрок J_2 отступит от своей оптимальной стратегии, а игрок J_1 будет придерживаться своей оптимальной стратегии, то второй игрок выиграет менее 18 ед.

Оптимальные смешанные стратегии

При отсутствии седловой точки в платежной матрице (например, как в игре “угадывание монет”), очевидно, ни одному из игроков не следует постоянно применять одну и ту же чистую стратегию. Здесь естественно определить понятие оптимальной стратегии в классе смешанных стратегий.

Пусть векторы $\boldsymbol{x} = (x_1, \ldots, x_m)$, $0 \le x_1 < 1$, $\sum\limits_{i=1}^{m} x_i = 1$; $\boldsymbol{y} = (y_1, \ldots, y_n)$, $0 \le y_i < 1$, $\sum\limits_{j=1}^{n} y_j = 1$, $i = 1, 2, \ldots, m$ суть смешанные стратегии игроков соответственно первого и второго, а матричная игра определяется платежной матрицей $C = \{c_{ij}\}$, $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, n$. Тогда математическое ожидание платежа игрока J_1 игроку J_2 (средний выигрыш игрока J_2) определяется очевидным образом:

$$\mathcal{F}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x} C \boldsymbol{y} = \sum_{i=1}^{m} \sum_{j=1}^{n} x_i x_{ij} y_j. \tag{7.1}$$

Как и для случая чистых стратегий, игрок J_1 может обеспечить себе средний проигрыш не более

$$\min_{\boldsymbol{x}} G(\boldsymbol{x}) = \min_{\boldsymbol{x}} [\max_{\boldsymbol{y}} \mathcal{F}(\boldsymbol{x}, \boldsymbol{y})], \tag{7.2}$$

а игрок J_2 может обеспечить себе средний выигрыш не менее

$$\max_{\boldsymbol{y}} H(\boldsymbol{y}) = \max_{\boldsymbol{y}} [\min_{\boldsymbol{x}} \mathcal{F}(\boldsymbol{x}, \boldsymbol{y})]. \tag{7.3}$$

Задачи (7.2) и (7.3) представляют собой задачи поиска гарантированных смешанных стратегий игроками соответственно первым и вторым.

Если для некоторых фиксированных смешанных стратегий $\boldsymbol{x}^*$ и $\boldsymbol{y}^*$ и для всех остальных смешанных стратегий $\boldsymbol{x}$, $\boldsymbol{y}$ выполняется

$$\mathcal{F}(\boldsymbol{x}^*, \boldsymbol{y}) \leq \mathcal{F}(\boldsymbol{x}^*, \boldsymbol{y}^*) \leq \mathcal{F}(\boldsymbol{x}, \boldsymbol{y}^*),$$

т. е. $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ является седловой точкой функции $\mathcal{F}(\boldsymbol{x}, \boldsymbol{y})$, то доказано, что

$$\mathcal{F}(\boldsymbol{x}^*, \boldsymbol{y}^*) = \min_{\boldsymbol{x}} \max_{\boldsymbol{y}} \mathcal{F}(\boldsymbol{x}, \boldsymbol{y}) = \max_{\boldsymbol{y}} \min_{\boldsymbol{x}} \mathcal{F}(\boldsymbol{x}, \boldsymbol{y}). \tag{7.4}$$

Таким образом, если функция $\mathcal{F}(\boldsymbol{x}, \boldsymbol{y})$, представляющая собой средний выигрыш второго игрока и определяемая соотношением (7.1), имеет седловую точку, то вводятся следующие определения.

Если существует седловая точка $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ функции $\mathcal{F}(\boldsymbol{x}, \boldsymbol{y})$, то ее компоненты $\boldsymbol{x}^*$ и $\boldsymbol{y}^*$ называются соответственно оптимальными стратегиями игроков J_1 и J_2, а $\mathcal{F}(\boldsymbol{x}^*, \boldsymbol{y}^*)$ — ценой игры. При этом также говорят, что матричная игра имеет решение в смешанных стратегиях.

Доказано, что поиск седловой точки $(\boldsymbol{x}^*, \boldsymbol{y}^*)$ функции $\mathcal{F}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x}C\boldsymbol{y}$ связан с решением двойственных ЗЛП; кроме того, эта точка всегда существует, а это означает, что любая матричная игра разрешима в смешанных стратегиях. Эти результаты сформулируем в виде теорем.

Теорема 1. *Задачи (7.2) и (7.3) игроков J_1 и J_2 эквивалентны следующим ЗЛП:*

$$\begin{gathered} x_{m+1} \to \min, \\ \sum_{i=1}^{m} c_{ij} x_i \leq x_{m+1}, \qquad j = 1, 2, \ldots, n, \\ \sum_{i=1}^{m} x_i = 1, \qquad x_i \geq 1, \quad i = 1, 2, \ldots, m; \end{gathered} \tag{7.5}$$

$$
\begin{gathered}
y_{m+1} \to \max, \\
\sum_{j=1}^{n} c_{ij} y_j \le y_{m+1}, \qquad i = 1, 2, \ldots, m, \\
\sum_{j=1}^{n} y_j = 1, \qquad y_j \ge 1, \quad j = 1, 2, \ldots, n.
\end{gathered}
\tag{7.6}
$$

Теорема 2 (о минимаксе). *Любая матричная игра имеет решение в смешанных стратегиях.*

Пример 3. Рассмотрим игру "угадывание монет" с платежной матрицей $C = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$. Пусть $\boldsymbol{x} = (x_1, x_2)$; $\boldsymbol{y} = (y_1, y_2)$ — смешанные стратегии игроков соответственно J_1 и J_2. Средний выигрыш второго игрока

$$
\mathcal{F}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x} C \boldsymbol{y} = y_1(-x_1 + x_2) + y_2(x_1 - x_2).
$$

Задачи (7.2), (7.3) приобретают вид

$$
\begin{gathered}
\max_{\boldsymbol{y}} [y_1(-x_1 + x_2) + y_2(x_1 - x_2)] \to \min_{\boldsymbol{x}}; \\
\min_{\boldsymbol{x}} [y_1(-x_1 + x_2) + y_2(x_1 - x_2)] \to \max_{\boldsymbol{y}},
\end{gathered}
$$

где

$$
\begin{array}{llll}
\boldsymbol{x} = (x_1, x_2), & x_1 \ge 0, & x_2 \ge 0, & x_1 + x_2 = 1; \\
\boldsymbol{y} = (y_1, y_2), & y_1 \ge 0, & y_2 \ge 0, & y_1 + y_2 = 1.
\end{array}
$$

Задачи (7.5), (7.6) записываются в виде

$$
\begin{array}{cc}
x_3 \to \min, & y_3 \to \max, \\
-x_1 + x_2 \le x_3, & -y_1 + y_2 \ge y_3, \\
x_1 - x_2 \le x_3, & y_1 - y_2 \ge y_3, \\
x_1 \ge 0, \quad x_2 \ge 0; & y_1 \ge 0, \quad y_2 \ge 0.
\end{array}
$$

Решая эти двойственные ЗЛП (например, с помощью симплекс-метода), получаем оптимальные смешанные стратегии $\boldsymbol{x}^* = (1/2; 1/2)$; $\boldsymbol{y}^* = (1/2; 1/2)$ игроков соответственно J_1 и J_2. При этом цена игры $\mathcal{F}(\boldsymbol{x}^*, \boldsymbol{y}^*) = 0$.

Как отмечалось в случае чистых стратегий, если один из игроков придерживается своей оптимальной стратегии, а второй отступает от своей оптимальной стратегии, то последний (отступающий) игрок либо выигрывает меньше, либо проигрывает больше. Аналогичные действия игроков в случае смешанных стратегий имеют результат, сформулированный в следующей теореме.

Рассматривается оптимальная смешанная стратегия $\boldsymbol{x}^* = (x_1^*, \ldots \ldots, x_m^*)$ игрока J_1. Назовем i-ю стратегию игрока активной (существенной), если $x_i^* > 0$. Аналогично определяются активные стратегии игрока J_2.

Теорема (об активных стратегиях). *Если игрок J_2 придерживается своей оптимальной стратегии $\boldsymbol{y}^*$, то его средний выигрыш*

$$\mathcal{F}(\boldsymbol{x}, \boldsymbol{y}^*) = \boldsymbol{x}C\boldsymbol{y}^*$$

остается неизменным и равным цене игры $v = \mathcal{F}(\boldsymbol{x}^*, \boldsymbol{y}^*)$ независимо от стратегии игрока J_1, если только первый (J_1) игрок не выходит за пределы своих активных стратегий (пользуется любой из них в чистом виде или смешивает их в любых пропорциях).

Разумеется, теорему можно переформулировать, меняя местами игроков J_1 и J_2.

Пример 4. Игра определяется матрицей $C = \begin{pmatrix} -2 & 1 \\ 1 & -1 \end{pmatrix}$, не имеющей седловой точки. Следовательно, ее решение нужно искать в классе смешанных стратегий. Пусть $\boldsymbol{y}^* = (y_1^*, y_2^*)$ — оптимальная смешанная стратегия игрока J_2. В силу сказанного она удовлетворяет системе уравнений

$$\begin{aligned} v &= -2y_1^* + y_2^*, \\ v &= \quad\ y_1^* - y_2^*, \\ & y_1^* + y_2^* = 1, \end{aligned}$$

откуда $\boldsymbol{y}^* = (2/5; 3/5)$, $v = -1/5$.

Аналогично находим оптимальную смешанную стратегию игрока J_1: $\boldsymbol{x}^* = (2/5; 3/5)$.

Приложение

ИСПОЛЬЗОВАНИЕ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ ПРИ РЕШЕНИИ ЗАДАЧ МАТЕМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Для многих задач математического программирования предусматривается графическое решение. В случае графического решения задач в плоскости целесообразно использовать программу *GRAN1* (авторы М. И. Жалдак, Ю. В. Горошко). Возможности работы с программой *GRAN1* подробно описаны в [4]. Здесь рассмотрим лишь те детали работы с программным средством, которые необходимы для решения задач в контексте изучаемого материала.

Название программы *GRAN1* продиктовано ее предназначением — графический анализ функций (*Graphic Analysis*). Для работы с программой необходимо наличие файла *GRAN1.exe*, а в случае использования в работе контекстной помощи — наличие также файла *GRAN1.hlp*. Суммарный объем памяти, необходимый для работы с программой, не превышает 240 Кбайт. После запуска на выполнение файла *GRAN1.exe* на экране появляется изображение, показанное на рис. 1.

Экран условно разделен на шесть частей. Верхнюю часть занимает основное меню. Каждая его команда позволяет обратиться к определенному набору команд (подменю). Центральное место занимает окно "*Графік*", предназначенное для представления графиков введенных в рассмотрение функций.

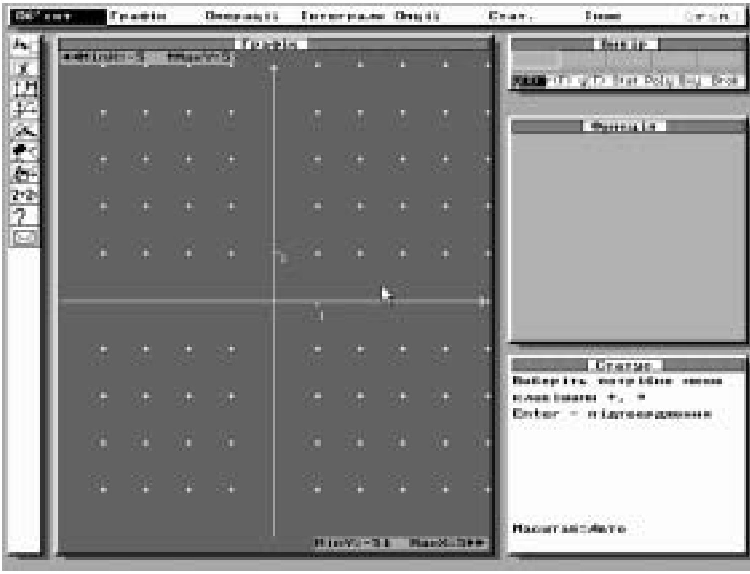


Рис. 1

Окно “*Вибір*” отображает количество введенных в рассмотрение функций, их названия, а также тип функциональной зависимости. Одновременно можно работать не более чем с пятью функциями. Каждая из них может быть задана с помощью одного из следующих типов функциональных зависимостей:

- явное задание функции, т. е. зависимость между переменными X и Y представлена в виде $y = y(x)$;
- параметрическое задание, т. е. зависимость между переменными x и y задается с использованием параметра t: $x = x(t)$, $y = y(t)$;
- полярное представление функции, т. е. зависимость задана в полярных координатах в виде $r = r(F)$, где r — полярный радиус точки на плоскости; F — полярный угол;
- неявное задание функции, т. е. зависимость между переменными X и Y представлена в виде $G(x, y) = 0$;
- табличное задание функции.

Программа *GRAN1* предусматривает возможность работы со статистической выборкой и объектом типа “*Ламана*”. Для задания типа функциональной зависимости используют команду “*Встановити тип*” пункта основного меню “*Опції*”.

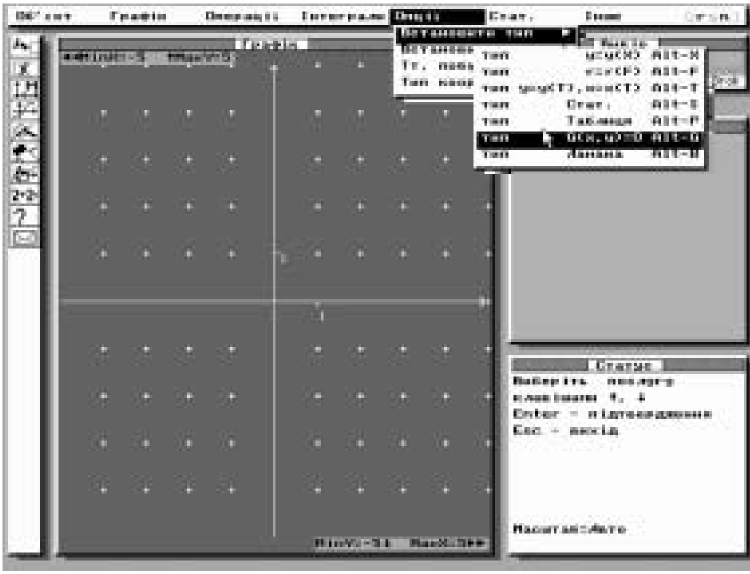


Рис. 2

После того как указан тип функциональной зависимости при обращении к команде “*Новий об’єкт*”, которая находится в подменю команды главного меню “*Об’єкт*”, пользователю предлагается ввести функцию согласно указанному типу.

Пример. Пусть необходимо ввести в рассмотрение функцию $3x+$ $+7y = 4$. Преобразуем выражение, описывающее функцию, к виду $G(x, y) = 0$ (неявное представление функции):

$$3x + 7y - 4 = 0. \qquad (*)$$

Выберем из основного меню пункт “*Опції*”, далее — пункт “*Встановити тип*”, а затем “*тип* $G(x, y) = 0$” (рис. 2).

В основном меню выбираем пункт “*Об’єкт*” и далее “*Новий об’єкт*”. После этого вводим левую часть выражения $(*)$, используя клавиатуру либо панель калькулятора, который предлагается программой (рис. 3).

Чтобы ввести выражение с помощью панели калькулятора, необходимо поставить курсор на соответствующую кнопку калькулятора и нажать клавишу “*Пропуск*”. Заканчивается ввод выражения вы-

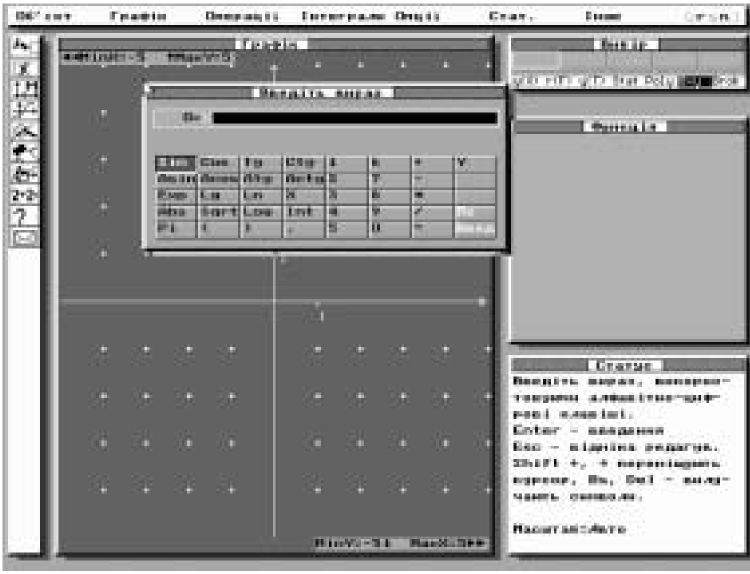


Рис. 3

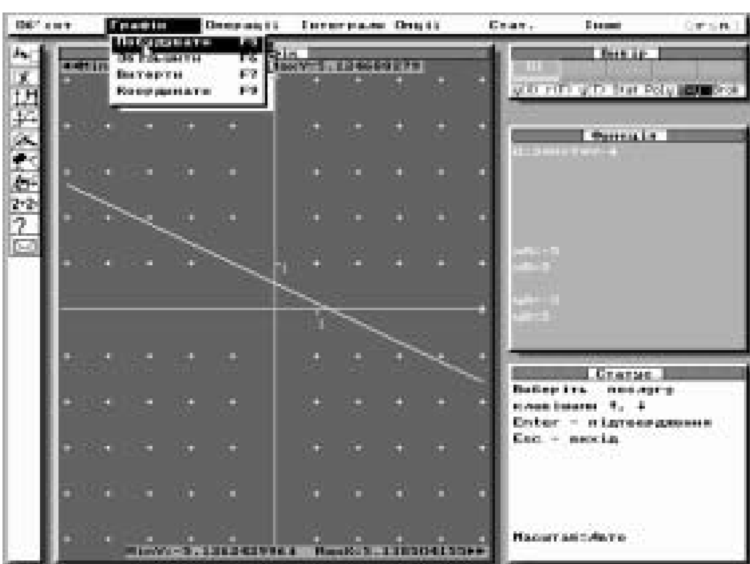


Рис. 4

бором на панели калькулятора кнопки “*Введення*”. Удобно вводить выражения с помощью манипулятора “мышь”.

Введенное выражение отображается в окне “*Функція*”. Чтобы получить изображение графика заданной функции в окне “*Графік*”, нужно обратиться к пункту “*Графік*” основного меню и выбрать подпункт “*Побудувати*”. Аналогичным будет результат при нажатии клавиши *F5* (рис. 4).

В нижнем правом углу экрана расположено окно “*Статус*”, которое содержит информацию, о текущем моменте работы с программой.

В левой части экрана расположен набор пиктограмм, которые позволяют пользователю ускорить работу с программой. Следует отметить, что текущее действие при работе с *GRAN1* отменяется нажатием клавиши *Esc*, а одни и те же функциональные клавиши в разных ситуациях могут иметь разное назначение.

Некоторые возможности программы и ее особенности разберем непосредственно, рассмотрев примеры решения задач с использованием *GRAN1*.

Графическое решение некоторых задач линейного и нелинейного программирования

Использовать программное средство *GRAN1* при решении ЗЛП можно и целесообразно в таких случаях:

1) если ограничений-неравенств не более четырех;
2) переменных в ограничениях-неравенствах не более двух.

Задача 1. Построить многоугольник решения ЗЛП, заданный ограничениями-неравенствами:

$$2x + 5y \le 10, \tag{1}$$
$$5x + 2y \le 10, \tag{2}$$
$$x + y \le 2,5, \tag{3}$$
$$x \ge 0, \tag{4}$$
$$y \ge 0. \tag{5}$$

Установим тип функциональной зависимости — неявно заданная функция, выбирая последовательно команды “*Опції*” — “*Встановити тип*” — “*тип* $G(x, y) = 0$”.

Запишем ограничения в виде $G(x,y) \geq 0$:

$$-2x - 5y + 10 \geq 0, \tag{1'}$$
$$-5x - 2y + 10 \geq 0, \tag{2'}$$
$$-x - y + 2,5 \geq 0, \tag{3'}$$
$$x \geq 0, \tag{4'}$$
$$y \geq 0. \tag{5'}$$

Введем в рассмотрение пять функций $G1$–$G5$ указанного типа, выбирая последовательно опции “*Об’єкт*” — “*Новий об’єкт*”. Выражения, которые находятся в левой части ограничений (1′)–(5′), вводятся либо непосредственно с клавиатуры, либо с использованием калькулятора на экране. После ввода каждой из функций в окне “*Вибір*” появляется обозначение для введенной функции, а в окне “*Функція*” — введенные выражения.

Все функции должны быть активными. Признаком того, что функция активная, является подчеркнутое обозначение функции в окне “*Вибір*”. Если обозначение в окне “*Вибір*” не подчеркнуто, т. е. функция неактивная, ее необходимо сделать активной. Для этого нужно из главного меню выбрать опцию “*Об’єкт*”, далее “*Вибір*”, установить курсор в окне “*Вибір*” с помощью стрелок вправо-влево на обозначение функции и нажать клавишу пропуск. (Аналогично отменяется активность функции.)

После того как введены в рассмотрение все функции, причем все они отмечены как активные, необходимо построить графики. Как отмечалось, для этого используют команды “*Графік*” — “*Побудувати*”. В окне “*Графік*” появляется изображение пяти графиков (рис. 5). Очевидно, для дальнейшего решения необходимо изменить отрезки наблюдения по координатным осям. Для этого используют команды “*Опції*” — “*Встановити масштаб*” — “*Масштаб корист. Alt-U*” (рис. 5′). На запрос ввести $XA = -1$; $XB = 6$; $YA = -1$; $YB = 6$. Возможно, для большей наглядности целесообразно использовать команды “*Об’єкт*” — “*Змінити відрізок*”, поскольку изначально были установлены $XA = YA = -5$; $XB = YB = 5$. Далее из главного меню необходимо выбрать пункт “*Операції*”, а из него — “*Система нерів.* $G(x,y)$ >= 0”. После обращения к данной услуге в окне “*Графік*” заштриховывается область из точек, координаты которых удовлетворяют все неравенства (1′)–(5′) (рис. 6). Заштрихованная область является многоугольником решений исходной задачи.

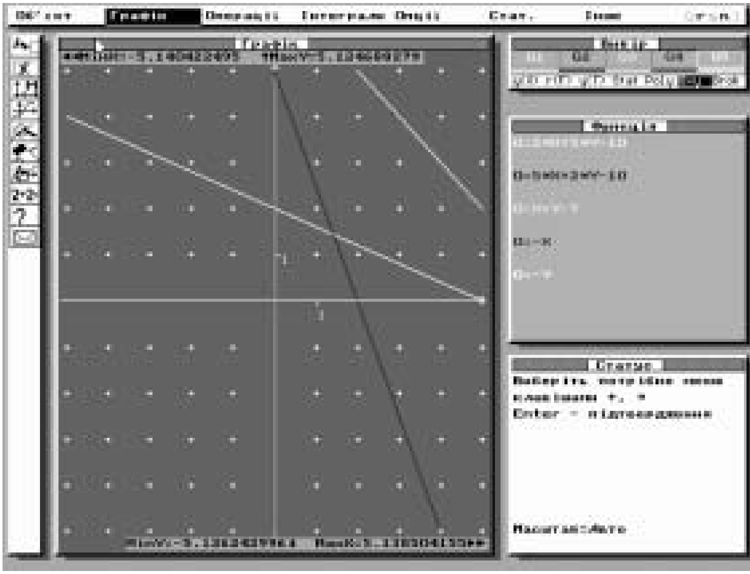


Рис. 5

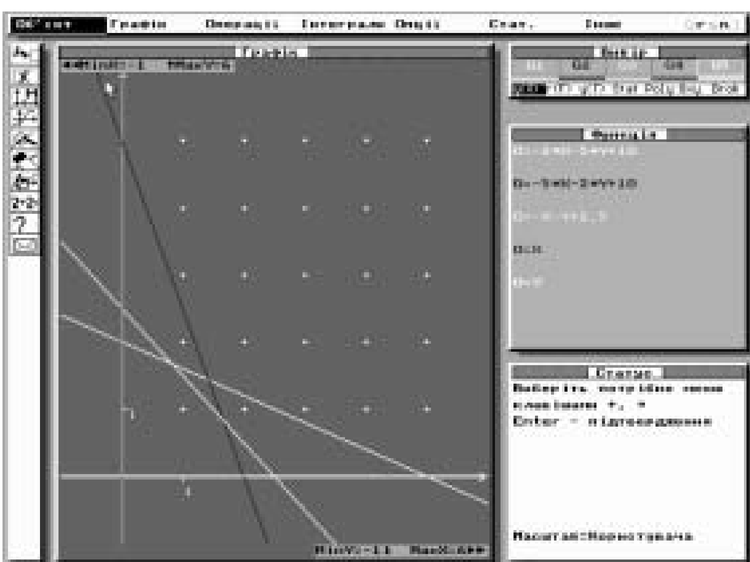


Рис. 5′

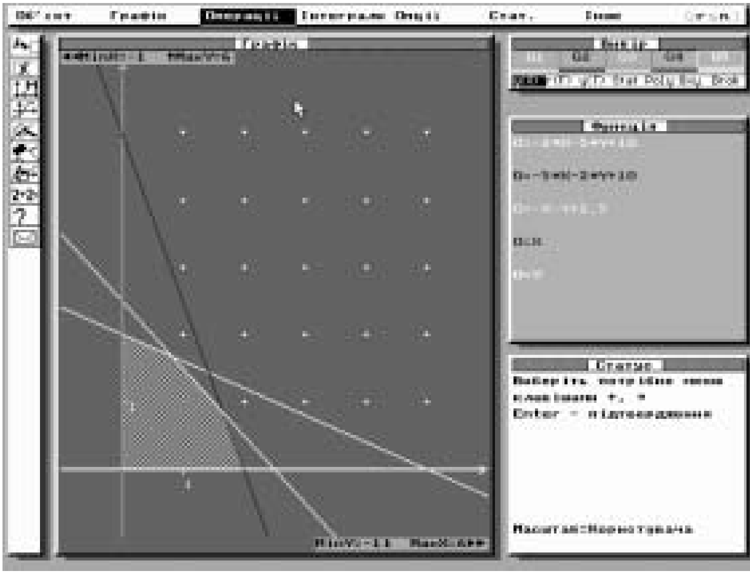


Рис. 6

Задача 2. Решить графически ЗЛП: найти максимальное значение линейной функции $Z = 12x_1 + 16x_2$ при ограничениях

$$\begin{aligned} 5x_1 + 4x_2 &\leq 20, \\ 16x_1 + 4x_2 &\leq 48, \\ 3x_1 + 8x_2 &\leq 30, \\ x_1 \geq 0, \quad x_2 &\geq 0. \end{aligned}$$

Количество ограничений-неравенств не превышает трех ($x_1 \geq 0$; $x_2 \geq 0$ не учитываем), а переменных в записи целевой функции две, т. е. можно использовать программу *GRAN1* для графического решения задачи. Перепишем ограничения-неравенства в виде $G(x, y) \geq 0$:

$$\begin{aligned} 20 - 5x_1 - 4x_2 &\geq 0, \\ 48 - 16x_1 - 4x_2 &\geq 0, \\ 30 - 3x_1 - 8x_2 &\geq 0. \end{aligned}$$

Заменим переменные x_1 на x, x_2 на y и введем в рассмотрение функции

$$
\begin{aligned}
G1(x,y) &= 20 - 5x - 4y,\\
G2(x,y) &= 48 - 16x - 4y,\\
G3(x,y) &= 30 - 3x - 8y.
\end{aligned}
$$

Для этого сначала установим необходимый тип функции: выберем команду главного меню "*Опції*" и далее "*Встановити тип*" — "*Тип* $G(y,t) = 0$". После того как тип $G(x,y)$ выбран с использованием последовательности команд "*Об'єкт*" — "*Нова функція*", задаем функции $G1(x,y)$; $G2(x,y)$; $G3(x,y)$.

На запрос системы относительно границ по координатным осям $(xA; XB; yA; yB)$ в данной задаче целесообразно принять предложенные изначально значения $xA = -5$; $xB = 5$; $yA = -5$; $yB = 5$.

Графики введенных функций строим с помощью команд "*Графік*" — "*Побудувати*" (рис. 7).

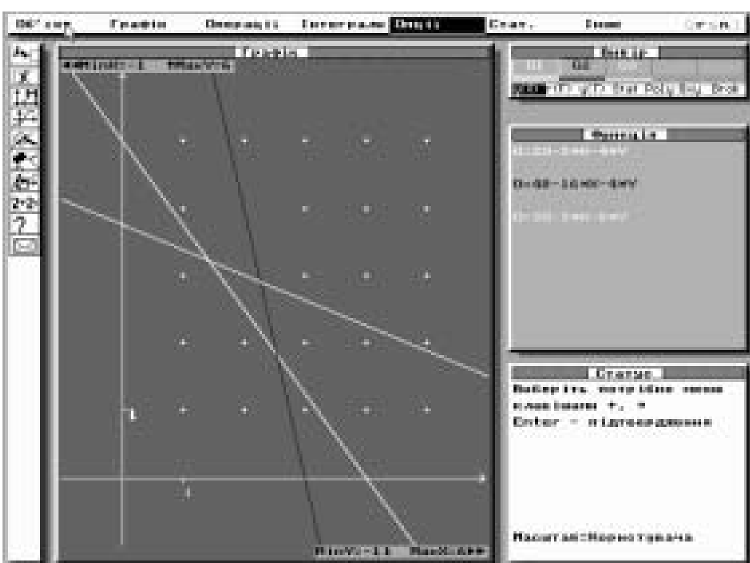


Рис. 7

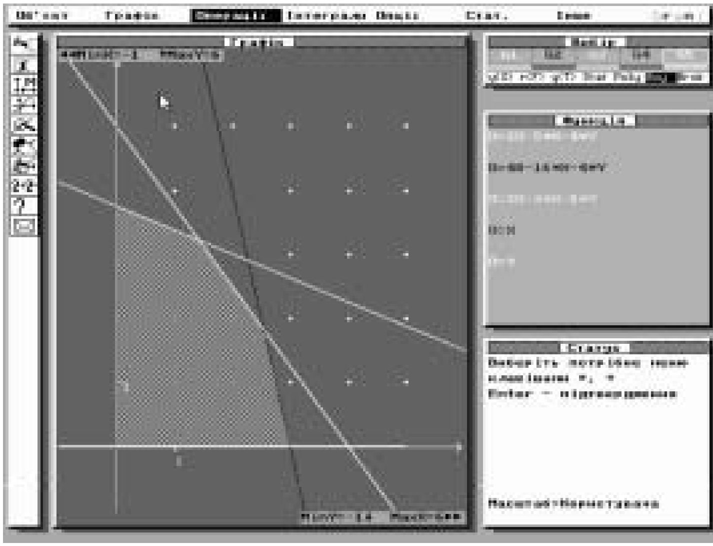


Рис. 7′

Если ввести в рассмотрение функции $G4 = x$ и $G5 = y$, а затем использовать команду “*Операції*” — “*Система нерів.* $G(x, y) >= 0$”, получим многоугольник решений данной задачи (рис. 7′).

Для дальнейшей работы удалим из рассмотрения функции $G4$ и $G5$. Для этого необходимо поставить курсор в окне “*Вибір*” на одну из функций и нажать клавишу $F8$ (или “*Об’єкт*” — “*Вилучити*”); то же самое следует повторить для второй функции.

Введем в рассмотрение функцию $G4(x, y) = 12x + 16y$, которая соответствует заданной целевой функций Z. Полученный на экране график функции $G4$ соответствует значению целевой функции $z_0 = 0$, т. е. $0 = 12x + 16y$ (рис. 8).

В силу того, что данная задача предусматривает нахождение максимального значения целевой функции на заданных ограничениях, то естественно следующий график построить для функции, соответствующей значению $Z_1 > Z_0$, например $15 = 12x + 16y$. Для этого нужно с помощью команды “*Об’єкт*” — “*Змінити функцію*” заменить функцию $G4(x, y)$ на функцию $G4(x, y) = 12x + 16y - 15$. Напомним, что курсор в окне “*Вибір*” должен находиться на функции $G4$. Внеся изменения в запись функции $G4$ и построив график измененной функции $G4$ (рис. 9), отметим, что значение $Z = 15$ не является оптимальным решением при заданных ограничениях.

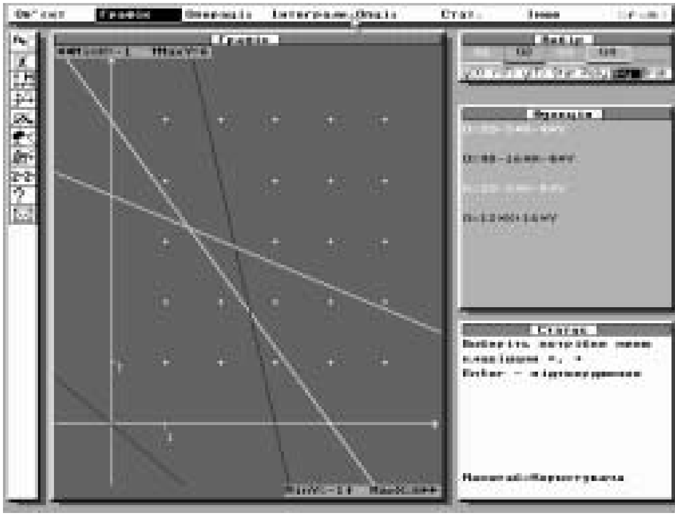


Рис. 8

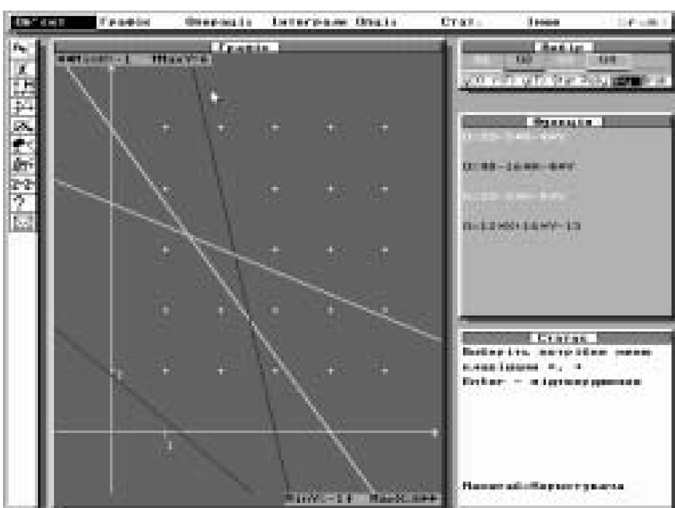


Рис. 9

Увеличивая значения Z, повторяя процедуру изменения функции и построения графика, а также учитывая, что целевая функция может достигать оптимального значения в вершинах многоугольника решений, подбираем такое значение $Z'_{\text{опт}}$, при котором график функции $G5$ пройдет через точку пересечения графиков функций $G1$ и $G3$. Очевидно, что подобранное таким образом значение является приближенным. Для более точного подбора значения $Z'_{\text{опт}}$ можно использовать команды “*Графік*” — “*Збільшити*”. При этом необходимо в окне “*Графік*” выделить участок для увеличения — область пересечения графиков функций $G1$ и $G3$ (рис. 10), а далее, используя команду “*Змінити функцію*”, продолжить подбор значения $Z'_{\text{опт}}$ (рис. 11). Используя описанный способ, получаем значение целевой функции $Z_{\text{опт}} = 68,55$.

Для того чтобы найти не только оптимальное значение $Z_{\text{опт}}$, но и оптимальный план, на котором это значение достигается, необходимо с помощью команды “*Графік*” — “*Координати*” отыскать координаты точек пересечения графиков функций $G1$ и $G3$. Более точные значения координат точки пересечения можно получить, используя команду “*Графік*” — “*Збільшити*”. Оптимальный план будет таким (рис. 12):

$$X_{\text{опт}}(1,42;3,22).$$

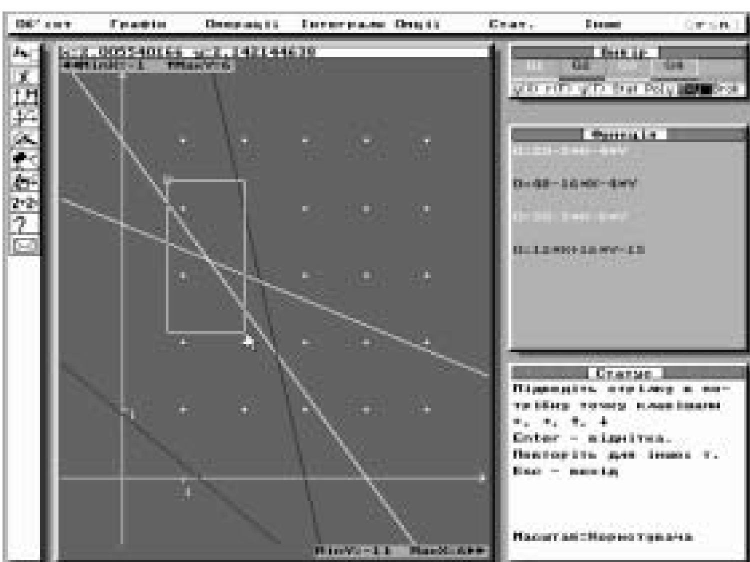


Рис. 10

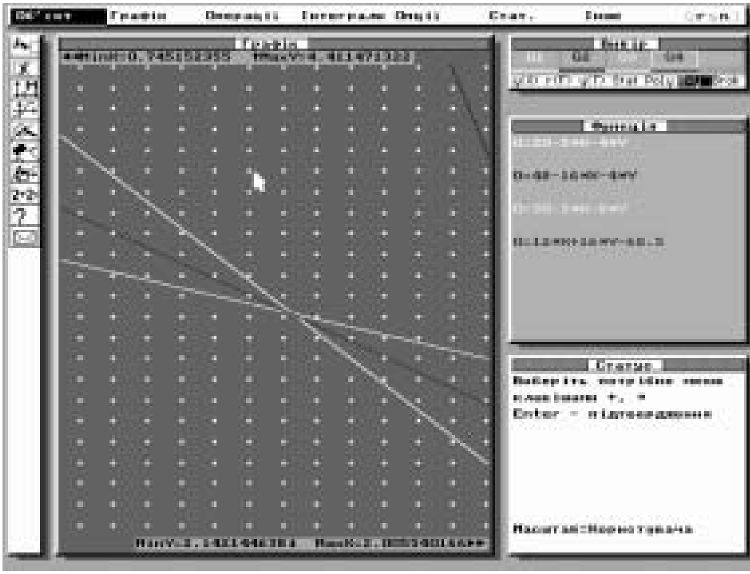


Рис. 11

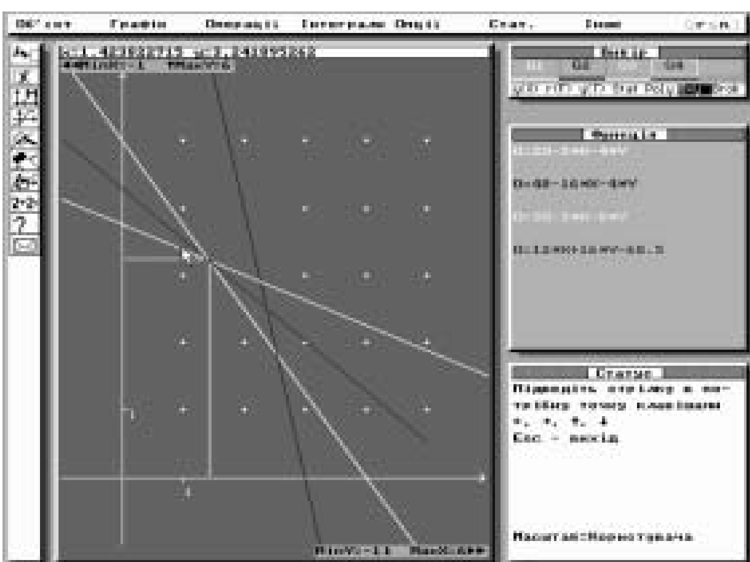


Рис. 12

Следует напомнить, что полученные таким образом значения компонент оптимального плана являются приближенными. Однако с помощью команд "*Збільшити*" и "*Координати*" можно добиться достаточно высокой точности результата.

Следует отметить, что при решении задачи в *GRAN1* часто возникает необходимость использовать не автоматически предложенный системой масштаб, а так называемый *Масштаб пользователя*, что позволяет рациональнее разместить графики на экране. Напомним, что установить желаемые границы по осям координат позволяют команды "*Опції*" — "*Встановити масштаб*" — "*Масштаб корист.*".

Задача 3. Найти максимальное и минимальное значения функции $Z = x_1^2 + x_2^2 - 4x_1 - 4x_2 + 8$ при ограничениях

$$\begin{cases} x_1 + x_2 \leq 7; \\ x_1 \geq 0; \\ x_2 \geq 0. \end{cases}$$

Введем в рассмотрение функции $G1(x, y) = 7 - x - y$, $G2(x, y) = x$, $G3(x, y) = y$ соответственно ограничениям-неравенствам. Используя известные команды, получаем многоугольник решений данной задачи (рис. 13).

Если преобразовать выражение для Z, то легко увидеть, что уравнение $Z = (x_1 - 2)^2 + (x_2 - 2)^2$ описывает окружность радиуса $\sqrt{Z}$ с центром в точке $(2; 2)$.

Введем в рассмотрение функцию $G4(x, y) = (x-2)^2 + (y-2)^2$ и построим ее график. Графиком функции $G4$ является точка с координатами $(2; 2)$. Использовав команду "*Змінити функцію*", внесем изменения в запись функции $G4$. Пусть $G4(x, y) = (x-2)^2 + (y-2)^2 - 1$. Графиком $G4(x, y)$ является окружность радиуса 1 с центром в точке $(2; 2)$ (рис. 14). Будем увеличивать радиус окружности до касания окружности и прямой. Последний вариант для функции $G4$ предлагаем такой: $G4(x, y) = (x - 2)^2 + (y - 2)^2 - 4,53$, т. е. радиус окружности равен $\sqrt{4,53}$, что соответствует значению целевой функции $Z_{\max} = 4,53$ (рис. 15). При использовании команд "*Збільшити*" и "*Змінити функцію*" можно добиться более точного результата. Очевидно, что отыскание наименьшего значения функции Z не требует никаких построений, так как оно достигается в точке с координатами $(2; 2)$ области решений и равно нулю, т. е. $Z_{\min} = 0$.

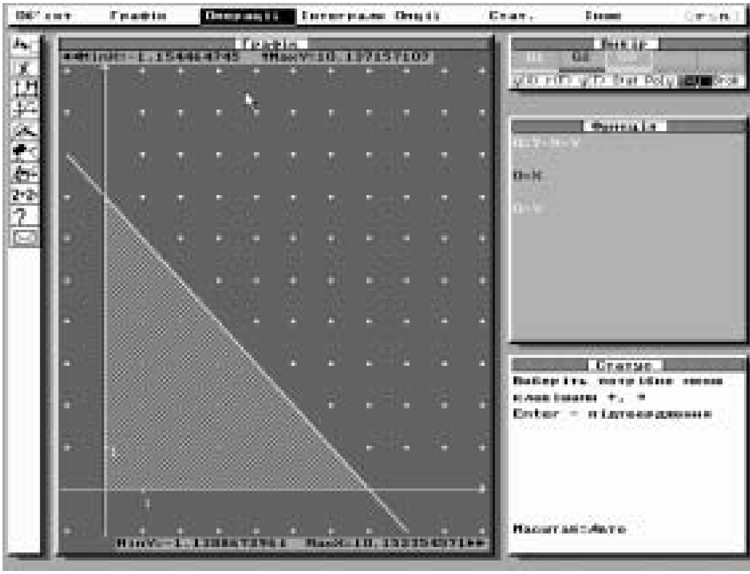


Рис. 13

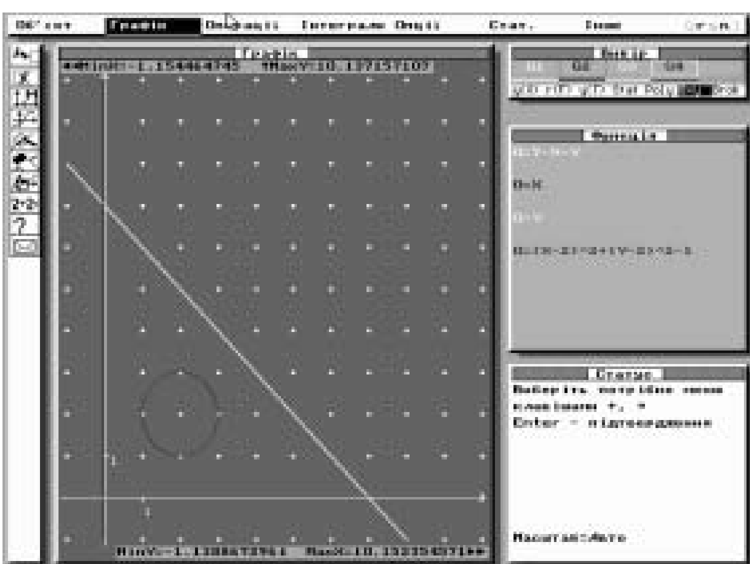


Рис. 14

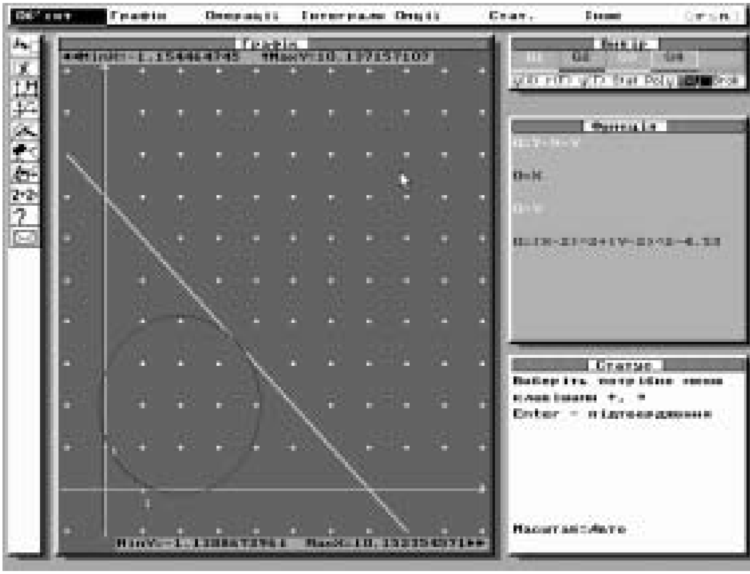


Рис. 15

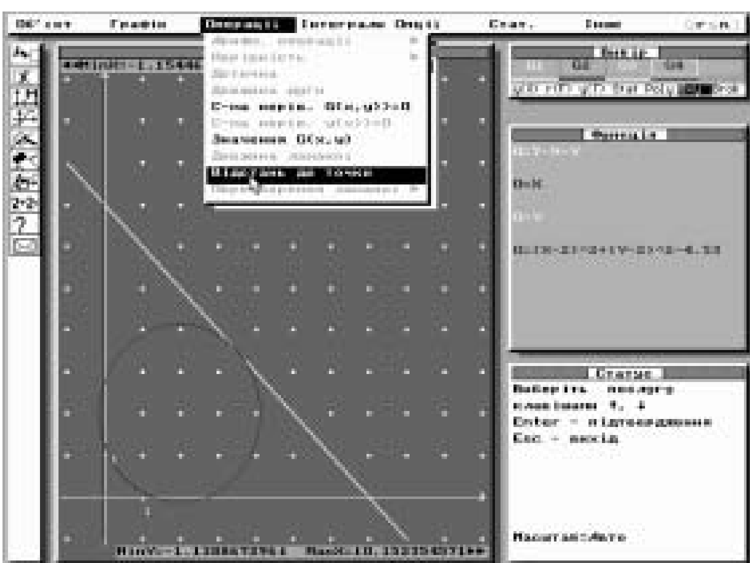


Рис. 16

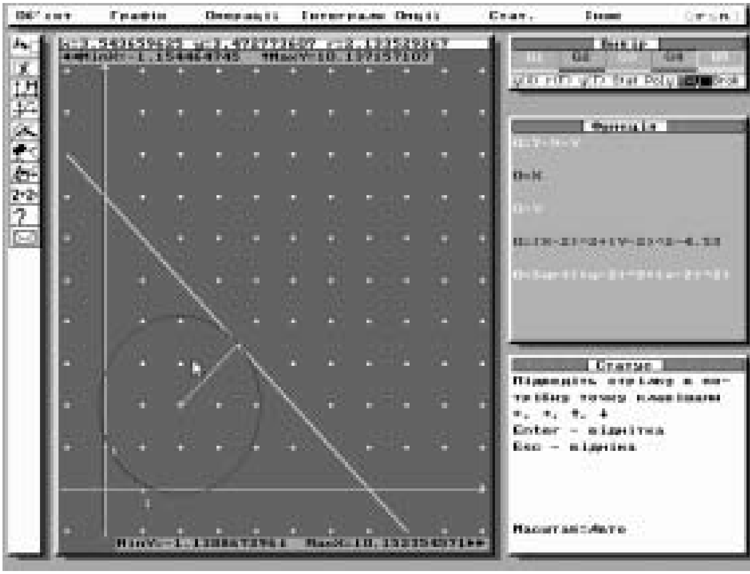


Рис. 17

Если графиком целевой функции является окружность с известным центром, то с помощью программы *GRAN1* решить задачу можно другим способом. После того как многоугольник решения задачи построен, из главного меню выбирают команду "*Операції*", а далее "*Відстань до точки*" (рис. 16). Координаты точки $(2; 2)$ можно вводить как с клавиатуры, так и с помощью манипулятора "мышь". Заданная точка отмечена на экране крестиком, а с помощью "мыши" нужно указать точку, от которой вычисляется расстояние до заданной точки. Очевидно, что для данной задачи точку на прямой $G1(x, y) = 7 - x - y$ отмечаем так, чтобы прямая $7 - x - y = 0$ и прямая, проходящая через две заданные точки ($(2; 2)$, а также отмеченная точка на прямой) были перпендикулярны (рис. 17). В верхней части окна "*Графік*" отображаются координаты точки, в которой целевая функция достигает наибольшего значения, и расстояния между заданными точками. В данной задаче $x \approx 3,54$; $y \approx 3,47$; $r \approx 2,13$, т. е. $Z_{\max} = r^2$. Окончательно $Z_{\max} \approx 2,13^2 \approx 4,54$. Как видим, ответы, полученные при разных способах решения, практически равны.

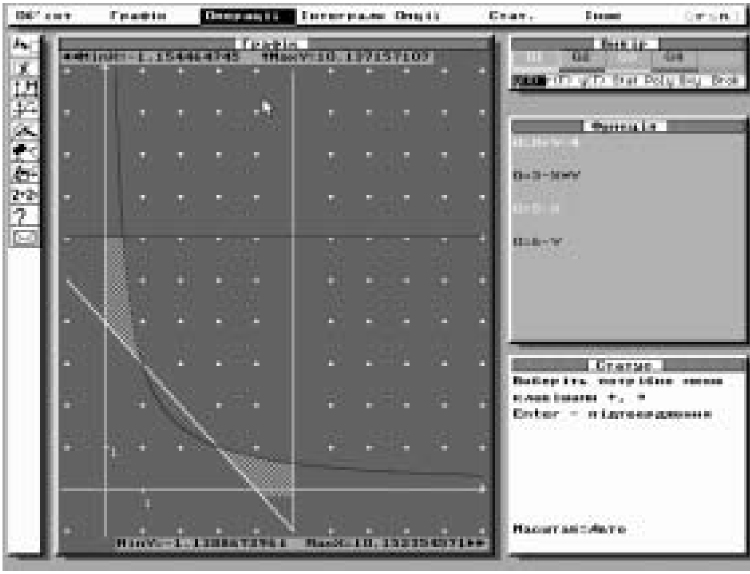


Рис. 18

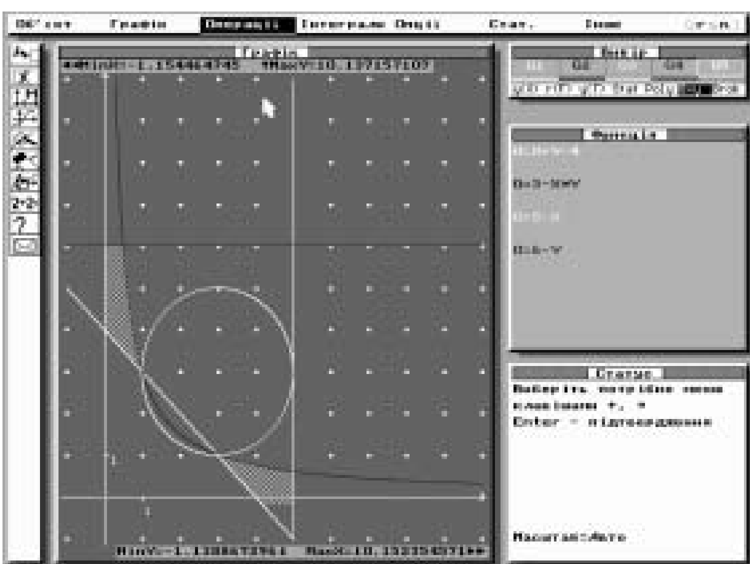


Рис. 19

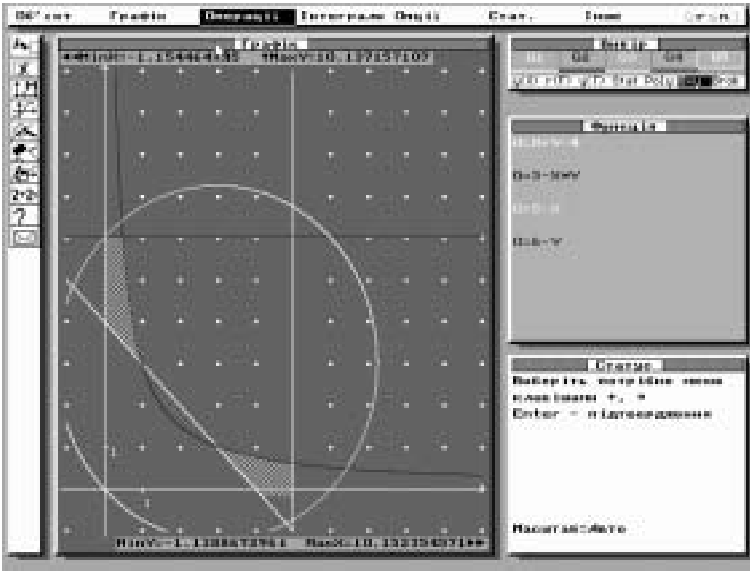


Рис. 20

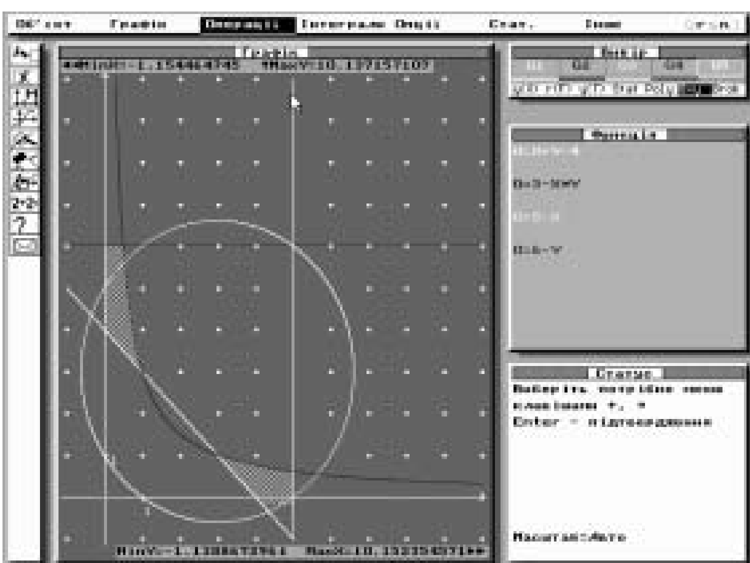


Рис. 21

Задача 4. Найти наибольшее и наименьшее значения функции $Z = x_1^2 - 6x_1 + x_2^2 - 6x_2 + 18$ при ограничениях

$$\begin{cases} x_1 + x_2 \geq 4; \\ x_1 x_2 \leq 3; \\ 0 \leq x_1 \leq 5; \\ 0 \leq x_2 \leq 6. \end{cases}$$

Обозначим x_1 через x, а x_2 — через y и введем в рассмотрение функции $G1(x, y) = x + y + 4$; $G2(x, y) = 3 - xy$; $G3(x, y) = 5 - x$; $G4(x, y) = 6 - y$. Используя команды "*Операції*" — "*Система нерів.* $G(x, y) >= 0$", получаем многоугольник решений данной задачи (рис. 18). Область решений задачи состоит из двух частей.

Функцию Z можно записать в виде $Z = (x_1 - 3)^2 + (x_2 - 3)^2$. Данное уравнение задает окружность радиуса $\sqrt{Z}$ с центром в точке с координатами $(3; 3)$. Используя команды "*Операції*" — "*Відстань до точки*", заключаем, что задача имеет два локальных минимума $Z_{\min 1}$ и $Z_{\min 2}$ в точках соответственно $(3; 1)$ и $(1; 3)$, и два локальных максимума $Z_{\max 1}$ и $Z_{\max 2}$ в точках с координатами соответственно $(0; 6)$ и $(5; 0)$. При этом $Z_{\min 1} = Z_{\min 2} = 4$; $Z_{\max_1} = 18$; $Z_{\max_2} = 13$.

Точность получения значений $Z_{\max}$ и $Z_{\min}$ достигается использованием команд "*Графік*" — "*Збільшити*". Значения $Z_{\max}$ и $Z_{\min}$ довольно легко можно получить, введя в рассмотрение функцию $G5(x, y) = (x - 3)^2 + (y - 3)^2$ и подобрав радиус окружности с помощью команд "*Об'єкт*" — "*Змінити функцію*" (рис. 19–21).

Решение некоторых вычислительных задач математического программирования

При аналитическом решении ряда задач математического программирования довольно часто возникает необходимость выполнения громоздких вычислений как над числами, так и над массивами чисел (матрицами, векторами и т. д.). Ускорить выполнение вычислительных операций и уменьшить вероятность появления ошибок при решении задач позволяет использование различных программных средств. К таким программам, в частности, относятся программное средство *DERIVE* и известная программа *Excel* из пакета *Microsoft Office*. Безусловно, существуют и другие программные средства математического направления, такие как *Mathcad*, *Mathlab*, *Eureka*, но остановимся на программах *DERIVE* и *Excel*. Выбор этот не случаен. Программное средство *DERIVE* довольно распространено среди пользователей ПК (так как существуют версии под *DOS* и *Windows*), достаточно полно описано в специальной литературе, удобно в использовании и широко по реализуемым возможностям. Табличный процессор *Excel* известен пользователям и не требует дополнительного времени на изучение, так как является стандартным продуктом пакета *Microsoft Office.*

Программа *DERIVE* предназначена для решения ряда математических задач в символьном виде. К таким задачам относятся: упрощение выражений, выполнение арифметических операций, разложение на множители, нахождение пределов (как в точке, так и на бесконечности), вычисление производных, интегралов, решение уравнений, выполнение действий над матрицами (векторами) и др. Программой предусмотрена возможность построения графиков функций на плоскости и изображения поверхностей в трехмерном пространстве. Следует отметить, что графические возможности программы *GRAN1* (относительно случая на плоскости) значительно шире, чем у программы *DERIVE.*

Для работы с программой *DERIVE* (для *Windows*) необходимо иметь файл *DFW.EXE.* После активизации последнего на экране появляется изображение (рис. 22). Изначально устанавливается активным алгебраическое окно, о чем свидетельствует надпись в верхней левой части экрана. Возможна работа и в графических режимах *2D-PLOT Windows*; *3D-PLOT Windows*, которые соответствуют двух- и трехмерному пространству.

В верхней части экрана постоянно находится основное меню программы, состоящее из набора команд (стандартное оформление программных продуктов для *Windows*). Для работы с основным меню используется, как правило, манипулятор “мышь”. Для выбора команды из основного меню можно воспользоваться “*горячей клавишей*” — подчеркнутой буквой в названии команды (например, *L* для выбора команды *so<u>L</u>ve*) — либо пиктограммой.

Выражение для работы можно ввести с клавиатуры (или загрузить из файла), а также с помощью панели с набором символов, предлагаемой в программе. Каждое введенное выражение отображается в алгебраическом окне под определенным номером. Назначение команд основного меню будем разбирать по ходу решения конкретных задач.

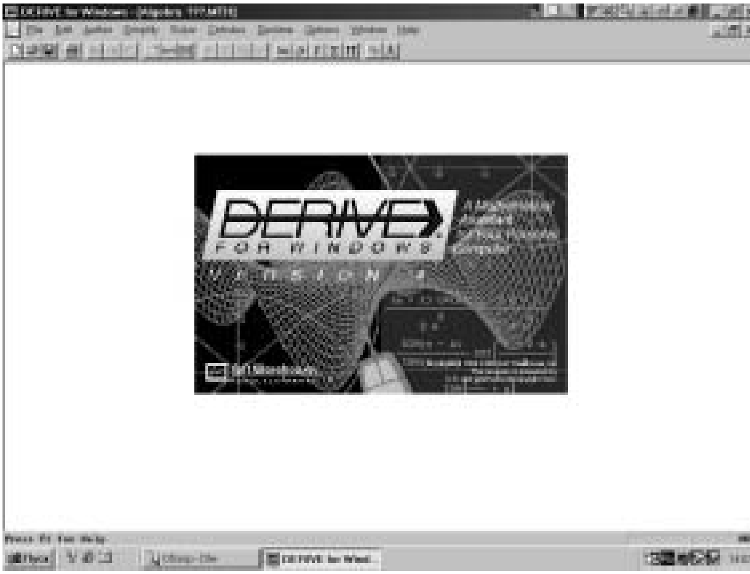


Рис. 22

Задача 1. Для функции $F(x, y, z, k, m)$ найти частные производные $\dfrac{\partial F}{\partial x}, \dfrac{\partial F}{\partial y}, \dfrac{\partial F}{\partial z}, \dfrac{\partial F}{\partial k}, \dfrac{\partial F}{\partial m}$, если $F = xy+yz+k(x+y-2)+m(y+z-2)$.

Введем в рассмотрение функцию F. Для этого из основного меню выбираем команду *Author* (щелчок правой клавиши мыши на слове *Author*). Из дополнительного меню выбираем *Expression* ...и далее вводим с клавиатуры выражение для функции F. После того как выражение введено, из основного меню выбираем команду *Calculus*. После обращения к услуге *Calculus* на месте главного меню появляется подменю выбранной команды:

Limit ...
Differentiate ...
Taylor series ...
Integrate ...
Sum ...
Produkt ...
Vector ...

Для вычисления производных используем команду *Differentiate*, вслед за выбором которой появляется запрос относительно выражения, которое необходимо дифференцировать, переменной, по которой нужно продифференцировать, и порядка производной.

В ответ на запрос необходимо ввести номер выражения, под которым была записана функция — номер 1, переменную дифференцирования *variable* — X и порядок производной (*order*) — 1.

После этого под номером 2 в алгебраическом окне будет записано

$$\frac{d}{dx}(xy + yz + k(x + y - 2) + m(y + z - 2)).$$

Далее выберем команду основного меню *Simplify*, а из появившегося подменю — *Expand*. При обращении к команде *Expand* на экране появляется окно — запрос относительно номера выражения, к которому необходимо применить команду *Expand*, переменных, используемых в выбранном выражении, и вида подачи выражения после выполнения действий (*Trivial*, *ScuareFree*, *Rational*, *Redical*).

После ответа на все запросы в окне запросов необходимо выбрать кнопку *Expand*.

Под номером 3 в алгебраическом окне будет записан результат дифференцирования F по переменной x, а именно $k+y$. Аналогично находят оставшиеся частные производные (рис. 23).

Если после ответов на запросы относительно переменной дифференцирования и порядка производной выбирается кнопка *Simplify*, а не *Ok*, то в алгебраическом окне записывается не выражение $\dfrac{dF}{dS}$ (где S — переменная дифференцирования), а сразу результат выполнения дифференцирования. На рис. 23 в алгебраическом окне под номерами #8 и #9 записаны результаты дифференцирования функции F по переменным соответственно k и m.

Замечание. С помощью команды *CALCULUS* программы *DERIVE* можно находить пределы (одно- и двусторонние), интегралы (неопределенные и определенные), раскладывать функцию в ряд Тейлора (с заданным количеством первых членов разложения), находить суммы типа $\sum\limits_{i=a}^{k} S$ (S — некоторое выражение; i — параметр суммирования; a, k — пределы суммирования соответственно нижний и верхний) и произведения типа $\prod\limits_{i=a}^{k} P$ (рис. 24).

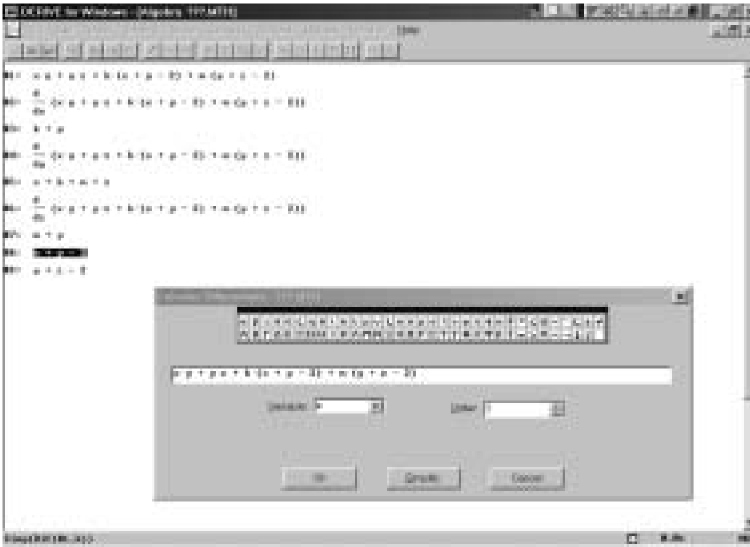


Рис. 23

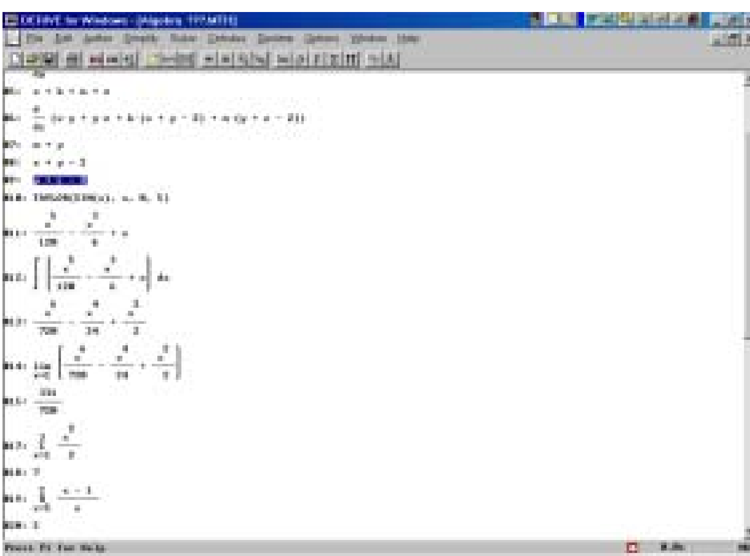


Рис. 24

Задача 2. Решить систему уравнений матричным методом:

$$\begin{cases} x_1 + 2x_2 - x_3 = 4; \\ 2x_1 - x_2 + x_3 = 12; \\ 3x_1 + 2x_2 + 3x_3 = 26. \end{cases}$$

Известно, что решением системы будет матрица-столбец

$$X = A^{-1}B,$$

где $B = \begin{pmatrix} 4 \\ 12 \\ 26 \end{pmatrix}$; A^{-1} — матрица, обратная к матрице

$$A = \begin{pmatrix} 1 & 2 & -1 \\ 2 & -1 & 1 \\ 3 & 2 & 3 \end{pmatrix}; \quad X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$ — столбец неизвестных.

Введем в рассмотрение матрицы A и B, используя команду основного меню *Author-Matrix*.

После того как количество строк (*Rows*) и столбцов (*Column*) введено, появляется запрос относительно элементов матрицы. В ответ необходимо ввести соответствующее количество выражений (чисел), которые определяют элементы матрицы. После этого в алгебраическом окне появляется под соответствующим номером (в данном случае под номером 1) матрица A. Аналогично в рассмотрение вводится матрица-столбец B (рис. 25).

Далее необходимо найти матрицу A^{-1}, обратную к матрице A. Для этого из основного меню выбираем команду *Author-Expression...* В рабочей строке окна *Author-Expression* вводим выражение (#1)^(−1), где (#1) — номер выражения, которое соответствует матрице A. Заканчиваем ввод выбором кнопки *Simplify*.

Под номером 3 в алгебраическом окне записана матрица A^{-1}. Далее, используя команду *Author-Expression*, перемножаем матрицы A^{-1} и B. Для этого в рабочей строке окна *Author-Expression* вводим (#3) · (#2).

Для наглядности решения закончим ввод $A^{-1}B$ выбором кнопки "*Ok*". Тогда в алгебраическом окне под номером 4 запишется умножение матриц A^{-1} и B. Далее используем команду *Simplify*. Очевидно, что выбрав после ввода выражения $A^{-1}B$ кнопку *Simplify*, под

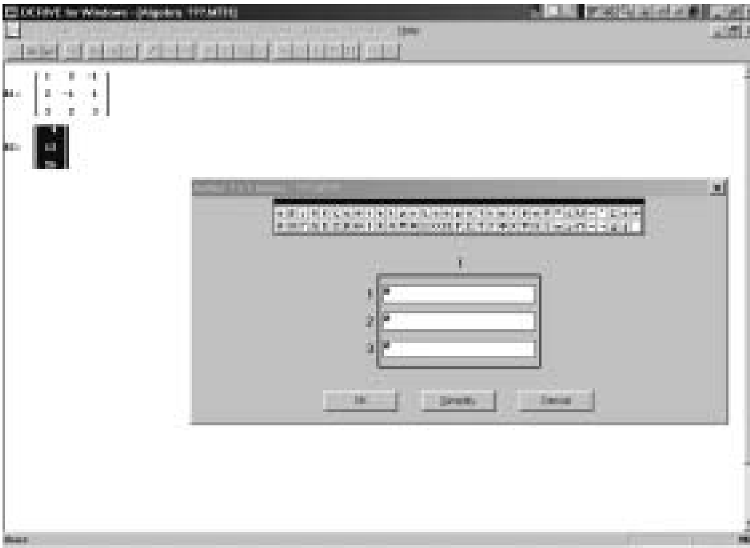


Рис. 25

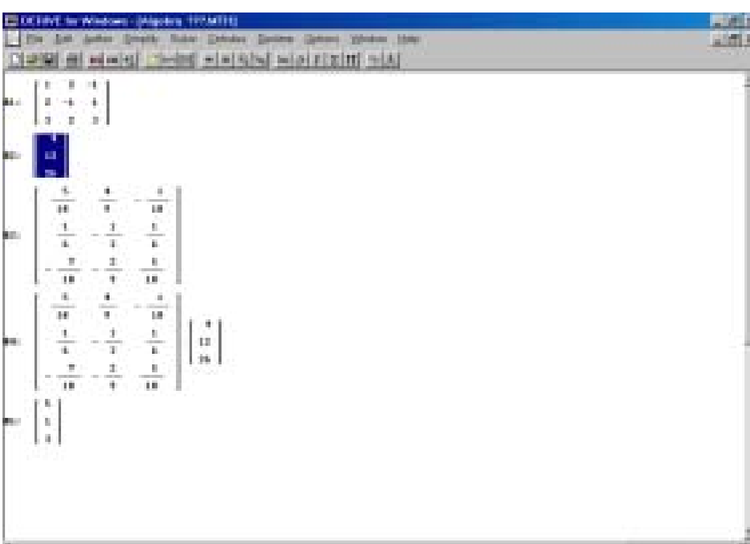


Рис. 26

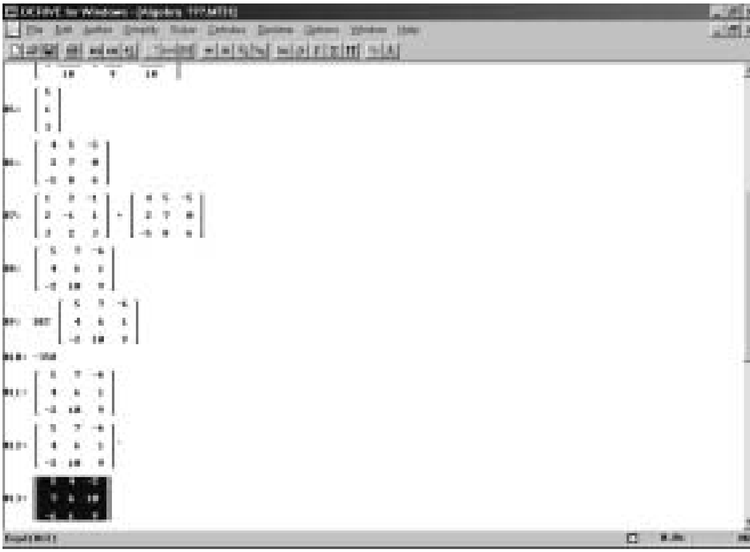


Рис. 27

номером 4 сразу можно было получить результат умножения матриц A^{-1} и B (рис. 26).

Программа *DERIVE* позволяет выполнять транспонирование матриц, сложение и вычитание матриц одинаковой размерности, вычислять определитель квадратной матрицы и др. (рис. 27).

Рассмотрим пример решения одной из задач вычислительного характера с использованием программы *Excel*.

Задача 3. В таблице приведены данные об исполнении баланса за отчетный период (в условных денежных единицах).

Отрасль	Потребление		Объем	
	1	2	конечного продукта	валового выпуска
1	120	180	240	540
2	280	50	85	415

Вычислить объем валового выпуска по каждой отрасли, если конечный продукт первой отрасли должен увеличиться в 2 раза, а второй отрасли — на 30 %.

Решение. Пусть x_{ij} — объем продукции i-й отрасли, потребляемой j-й отраслью в процессе производства $(i, j = 1, 2)$; x_i — общий (валовой) объем продукции i-й отрасли $(i = 1, 2)$.

Тогда $x_{11} = 120$; $x_{12} = 180$; $x_{21} = 280$; $x_{22} = 50$; $x_1 = 540$; $x_2 = 415$.

По условию задачи конечный продукт первой отрасли должен увеличиться в 2 раза, т. е. с 240 до 480, а второй отрасли — на 30 %, т. е. с 85 до 110,5. Вектор конечного продукта $Y = \begin{pmatrix} 480 \\ 110,5 \end{pmatrix}$.

Коэффициенты прямых затрат a_{ij} $(i, j = 1, 2)$ вычисляются по формуле $a_{ij} = \dfrac{x_{ij}}{x_j}$ $(i, j = 1, 2)$:

$$a_{11} = \frac{120}{540} \approx 0,22; \qquad a_{12} = \frac{180}{415} \approx 0,43;$$

$$a_{21} = \frac{280}{540} \approx 0,52; \qquad a_{22} = \frac{50}{415} \approx 0,12.$$

Матрица прямых затрат $A = \begin{pmatrix} 0,22 & 0,43 \\ 0,52 & 0,12 \end{pmatrix}$ имеет неотрицательные элементы и удовлетворяет критерию продуктивности:

$$\max\{0,22 + 0,52;\ 0,43 + 0,12\} = \max\{0,74;\ 0,55\} = 0,74 < 1.$$

Поэтому для каждого вектора конечного продукта Y можно найти необходимый объем валового выпуска X по формуле

$$X = (E - A)^{-1} Y.$$

Очевидно, что отыскать матрицу $E - A$ легко:

$$E - A = \begin{pmatrix} 0,78 & -0,43 \\ -0,52 & 0,88 \end{pmatrix}.$$

Матрицу полных затрат $S = (E - A)^{-1}$ вычислим, используя программу *Excel*.

Введем в рассмотрение матрицу $E - A$ (диапазон $A1$:$B2$).

Проверим, существует ли для матрицы $E - A$ обратная матрица. Для этого выполним такие действия:

1) переместим курсор в ячейку $D1$;
2) из главного меню выберем команду “*Вставка*” — “*Функция*”;
3) из предложенных категорий выберем “*Математические*”, а из предложенных функций — “*МОПРЕД*”;

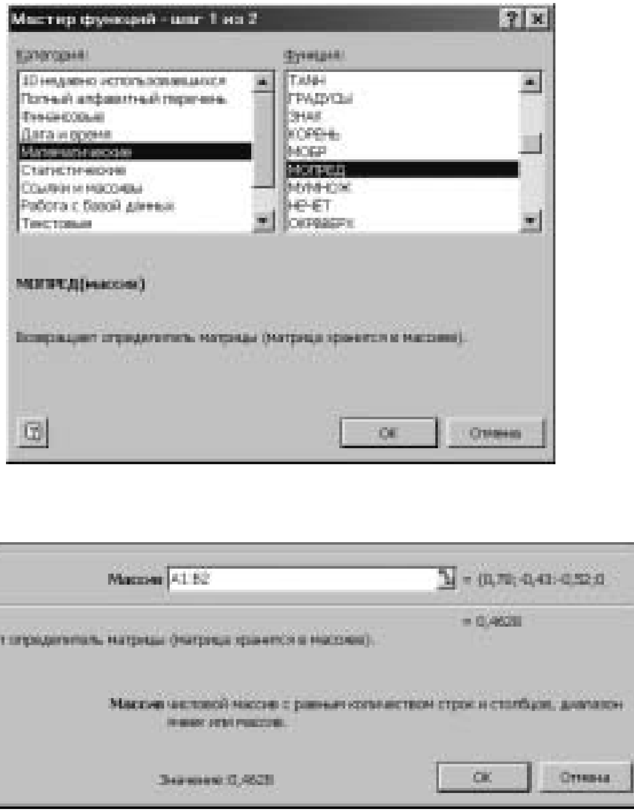


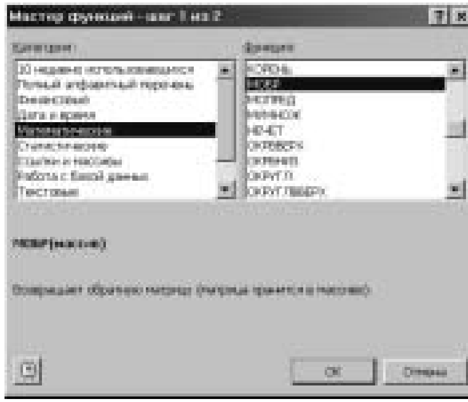
Рис. 28

4) на запрос системы вводим, используя “мышь”, диапазон $A1{:}B2$ и выбираем “*Ок*” как окончание ввода (рис. 28).

В ячейку $D1$ заносится число, равное значению определителя матрицы $E - A$. Так как полученное значение определителя (0,4628) не равно нулю, обратная матрица $S = (E-A)^{-1}$ существует. Вычислим матрицу S.

Для этого выполним такие действия:

1) выделим, используя клавиши *Shift*+, диапазон $A4{:}B5$ для размещения матрицы $(E - A)^{-1}$;
2) выберем из главного меню команду “*Вставка*” — “*Функция*”; из предложенного набора категорий выберем “*Математические*”, из набора функций — “*МОБР*”;



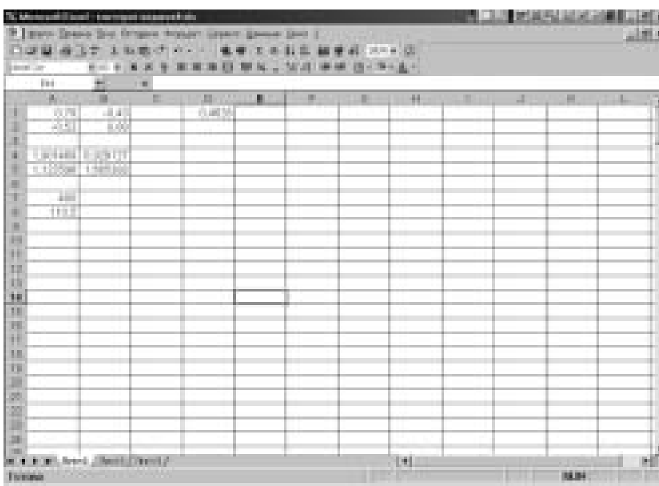


Рис. 29

3) в ответ на запрос введем диапазон $A1{:}B2$ (с помощью “мыши”);
4) закончим выполнение команды комбинацией клавиш *Shift* + + *Ctrl* + *Enter* (обязательно).

В диапазоне $A4{:}B5$ записана матрица $S = (E - A)^{-1}$ (рис. 29).

Используя формулу $X = (E - A)^{-1} \cdot Y$, можно отыскать вектор X валового выпуска. Для этого в диапазон $A7{:}A8$ введем матрицу Y и перемножим матрицы $(E - A)^{-1}$ и Y, используя функцию умножения матриц “*МУМНОЖ*”.

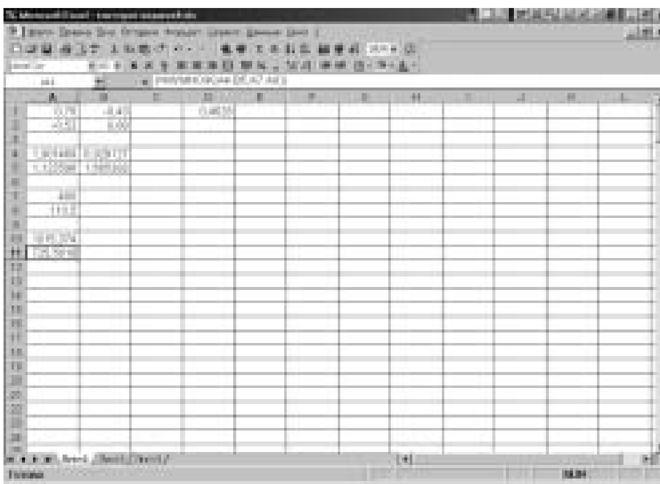


Рис. 30

Очевидно, что результатом умножения матриц размерностей 2×2 и 2×1 является матрица-столбец размерности 2×1, поэтому перед обращением к функции "*МУМНОЖ*" необходимо выделить диапазон клеток, в котором предполагается разместить результат умножения (в данном случае $A10 : A11$). На запрос системы относительно множителей необходимо ввести диапазоны $A4 : B5$ и $A7 : A8$, в которых располагаются матрицы $(E - A)^{-1}$ и Y. Полученный в диапазоне $A10 : A11$ вектор $X = \begin{pmatrix} 1015,374 \\ 725,5618 \end{pmatrix}$ является ответом задачи (рис. 30), т. е. объем валового выпуска в первой отрасли нужно увеличить с 540 до 1015,4 усл. ден. ед., а во второй — до 725,6 усл. ден. ед.

СПИСОК ИСПОЛЬЗОВАННОЙ И РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. *Акулич И. Л.* Математическое программирование в примерах и задачах. — М.: Высш. шк., 1986.
2. *Вагнер Г.* Основы исследования операций: В 3 т. — М.: Мир, 1973.
3. *Гольштейн Е. Г., Юдин Д. Б.* Задачи линейного программирования транспортного типа. — М.: Наука, 1969.
4. *Жалдак М. І.* Комп'ютер на уроках математики: Посібник для вчителів. — К.: Техніка, 1997.
5. *Зайченко Ю. П.* Исследование операций. — К.: Выща шк., 1984.
6. *Карманов В. Г.* Математическое программирование. — М.: Наука, 1975.
7. *Костарчук В. М., Вивальнюк Л. М.* Векторні простори і розв'язування задач лінійного програмування. — К.: Рад. шк., 1972.
8. *Ляшенко И. Н.* Линейное и нелинейное программирование. — К.: Выща шк., 1975.
9. *Математическое* программирование / Ю. Н. Кузнецов и др. — М.: Высш. шк., 1980.
10. *Попов Ю. Д.* Линейное и нелинейное программирование: Учеб. пособие. — К.: Изд-во КГУ, 1988.

СОДЕРЖАНИЕ

У навчальному посібнику містяться основні відомості курсу “Математичне програмування”. Теоретичний матеріал досить повно проілюстрований прикладами розв’язання задач. Наведено початкові відомості про використання інформаційних технологій при розв’язанні задач математичного програмування.

Для студентів нематематичних спеціальностей вузів.

Навчальне видання

Кулян Віктор Романович
Юнькова Олена Олександрівна
Жильцов Олексій Борисович

МАТЕМАТИЧНЕ ПРОГРАМУВАННЯ
(з елементами інформаційних технологій)

Навчальний посібник для студентів нематематичних спеціальностей вузів

2-ге видання, стереотипне

(Рос. мовою)

Відповідальний редактор *М. В. Дроздецька*
Редактор *Л. В. Логвиненко*
Коректори *А. А. Тютюнник, Л. В. Сазонова*
Комп’ютерне верстання *В. М. Бойко, А. В. Цебренко*
Оформлення обкладинки *О. А. Линник*

Підп. до друку 18.08.03. Формат $60 \times 84/_{16}$. Папір газетний. Друк офсетний.
Ум. друк. арк. 7,20. Обл.-вид. арк. 6,0. Тираж 6000 пр. Зам. № 03-01

Міжрегіональна Академія управління персоналом (МАУП)
03039 Київ-39, вул. Фрометівська, 2, МАУП

Свідоцтво про внесення до Державного реєстру суб’єктів видавничої справи ДК № 8 від 23.02.2000

АТЗТ “Видавничий центр “ДрУк”
03057 Київ-57, вул. Довженка, 10

Свідоцтво ДК № 108 від 4.07.2000