МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Государственное образовательное учреждение высшего образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

МАТЕМАТИЧЕСКИЕ МЕТОДЫ ИССЛЕДОВАНИЯ СИСТЕМ

Учебно – м*етодическое пособие по курсу «Математические методы исследования систем» для выполнения практических, лабораторных работ и проведения самостоятельной работы для студентов ВУЗа*

Томск
2018

Пособие составлено в соответствии с тематикой практических, лабораторных работ и самостоятельной работы по курсу «Математические методы исследования систем». Пособие содержит темы и содержание практических и лабораторных работ, методические указания к их проведению.

Для преподавателей, аспирантов, студентов и магистрантов.

СОСТАВИТЕЛЬ: Е.А. Шельмина, В.Г. Спицын

СОДЕРЖАНИЕ

Раздел 1. Практические работы

Практическая работа №1
Математические методы и их применение при принятии управленческих решений

Математическая модель системы - это ее отображение в виде совокупности уравнений, неравенств, логических отношений, графиков. Таким образом, модель - это условный образ системы, созданный для упрощения ее исследования, получения о ней новых знаний, анализа и оценки пронимаемых решений в конкретных или возможных ситуациях.

Выпуклость графика функции. Точки перегиба. Системы линейных уравнений

При исследовании поведения функции и формы ее графика полезно установить, на каких интервалах график функции обращен выпуклостью вверх, а на каких - выпуклостью вниз.

Определение. График функции $y=f(x)$, $x€(a;b)$ называется выпуклым вверх (вогнутым вниз) на интервале (a,b), если график расположен ниже (точнее не выше) любой своей касательной. Сама функция $f(x)$ также называется выпуклой вверх (вогнутой вниз).

Определение. График функции $y=f(x)$, $x€(a;b)$ называется выпуклым вниз (вогнутым вверх) на интервале (a,b), если он расположен выше (точнее не ниже) любой своей касательной. Сама функция $f(x)$ также называется выпуклой вниз (вогнутой вверх).

На интервале выпуклости вверх (вогнутости вниз) производная функции убывает.

На интервале выпуклости вниз (вогнутости вверх) производная функции возрастает.

Достаточное условие выпуклости графика функции. Если на интервале $(a;b)$ дважды дифференцируемая функция $y=f(x)$, имеет отрицательную (положительную) вторую производную, то график функции является выпуклым вверх (вниз).

Точка графика непрерывной функции $f(x)$, в которой существует касательная и при переходе через которую график функции меняет направление выпуклости, называется точкой перегиба.

Необходимое условие существования точки перегиба. Если дифференцируемая функция $y=f(x)$, $x€(a;b)$, имеет непрерывные производные до второго порядка включительно на интервале $(a;b)$ и точка $(x_0,f(x_0))$, где $x_0€ (a;b)$ является точкой перегиба графика функции $f(x)$, то $f''(x_0)=0$.

Достаточное условие существования точки перегиба. Если функция $y=f(x)$, $x€(a;b)$ дважды дифференцируема на интервале $(a;b)$ и при переходе через $x_0€(a;b)$, вторая производная $f''(x_0)=0$ меняет знак, то точка кривой с абсциссой $x=x_0$ является точкой перегиба.

Задания

1. Найти промежутки выпуклости (вогнутости) и точки перегиба графиков следующих функций:
   a) $y = x^3 - 3x^2 + 2x + 1$;
   b) $y = 3x^2 - x^3$;
   c) $y = x + x^{\frac{5}{3}}$;
   d) $y = \ln(1 + x^2)$.
2. Исследовать на экстремум следующие функции двух переменных:

a) $y = x_1^2 - 2x_1x_2 + 2x_2^2 - x_1 - 2x_2$;

b) $y = x_1^2 - x_2^2$;

c) $y = x_1^2 - 4x_1x_2 + x_2^2$;

Практическая работа №2
Линейное программирование и теория двойственности

Линейное программирование (планирование) - математический метод поиска максимума или минимума линейной функции при наличии ограничений в виде линейных неравенств или уравнений.

С каждой задачей линейного программирования связана другая задача, называемая двойственной по отношению к исходной.

1. Если первая задача имеет размеры m×n (m ограничений с n неизвестными), то вторая – размеры n×m.

2. Каждому ограничению исходной задачи соответствует переменная двойственной задачи, называемая двойственной переменной. Каждой переменной исходной задачи соответствует ограничение двойственной задачи; так, если в исходной системе имеется три переменных, то двойственная задача также должна иметь три ограничения.

3. Матрица коэффициентов при двойственных переменных в ограничениях двойственной задачи является транспонированной матрицей коэффициентов при переменных, состоящих в ограничениях. Так, если в исходной задаче имеется два ограничения, то матрица их коэффициентов А и ее транспонированный аналог АТ (столбцы превращаются в ряды) имеют вид:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix},$$

$$A^T = \begin{pmatrix} a_{11}\, a_{21} \\ a_{12}\, a_{22} \\ a_{13}\, a_{23} \end{pmatrix}.$$

4. Если в исходной задаче ограничения имеют знаки неравенств типа (≤) *меньше*, то в двойственной они изменяются на противоположные - типа *больше* (≥).

5. Правые части ограничений в двойственной задаче равняются коэффициентам при переменных целевой функции в исходной задаче, а коэффициенты при двойственных переменных в целевой функции двойственной задачи равняются правым частям ограничений исходной задачи.

6. Максимизация целевой функции исходной задачи заменяется минимизацией целевой функции двойственной задачи.

Задания

1. Решить геометрически и проверить на компьютере (MS Excel или OpenOffice Calc) следующие задачи:
   a. Найти максимальное значение целевой функции:

$$F = 2x_1 - 6x_2$$

при ограничениях

$$x_1 + x_2 \geq 2,$$
$$-x_1 + 2x_2 \leq 4,$$
$$x_1 + 2x_2 \leq 8,$$
$$x_1 \geq 0, x_2 \geq 0.$$

b. Найти минимальное значение целевой функции:

$$F = 2x_1 - x_2$$

при ограничениях:

$$x_1 + x_2 \geq 4,$$
$$-x_1 + 2x_2 \leq 2,$$
$$x_1 + 2x_2 \leq 10,$$
$$x_1 \geq 0, x_2 \geq 0.$$

2. Решить задачи симплексным методом, составить задачи двойственные данным и найти их решения:

1. $F = x_1 + x_2 \to \max$

при ограничениях

$$x_1 - x_2 \leq -2,$$
$$x_1 - 2x_2 \geq -13,$$
$$3x_1 - x_2 \leq 6,$$
$$x_1 \geq 0, x_2 \geq 0.$$

2. $Z = 10y_2 - 3y_3 \to \min$

при ограничениях

$$-2y_1 + y_2 - y_3 \geq 1,$$
$$y_1 + 2y_2 - y_3 \geq 3,$$
$$y_1 \geq 0, y_2 \geq 0, y_3 \geq 0$$

3. $F = x_1 + x_2 \to \max$

при ограничениях

$$x_1 - 4x_2 - 4 \leq 0,$$
$$3x_1 - x_2 \geq 0,$$
$$x_1 + x_2 - 4 \geq 0,$$
$$x_i \geq 0, x_{2i} \geq 0.$$

Практическая работа №3
Задачи многокритериальной оптимизации

Задачу многокритериальной оптимизации можно сформулировать следующим образом:

$$Z(\overline{X}) = \langle Z_1(\overline{X}), Z_2(\overline{X}), \ldots, Z_m(\overline{X}) \rangle \to \max,$$
$$\overline{X} \in Q.$$

Здесь i-ый частный критерий обозначен через $Z_i(\overline{X})$, где $\overline{X}$ - допустимое решение, а область допустимых решений – через Q.

В процессе решения задач многокритериальной оптимизации предварительно проводят экспертные оценки, как самих критериев, так и взаимоотношений между ними, и на основе этих оценок выбирают один из следующих методов решения:

- оптимизация одного признанного наиболее важным критерия, остальные критерии при этом играют роль дополнительных ограничений;
- упорядочение заданного множества критериев и последовательная оптимизация по каждому из них с помощью *метода последовательных уступок;*
- сведение многих критериев к одному введением экспертных весовых коэффициентов для каждого из критериев таким образом, что более важный критерий получает более высокий вес.

Решение задачи многокритериальной оптимизации должно принадлежать пересечению множеств оптимальных решений всех однокритериальных задач, которое обычно оказывается пустым, поэтому рассматривается обычно множество эффективных решений – оптимальных по Парето. Критерий оптимальности итальянского экономиста В. Парето применяется при решении таких задач, когда оптимизация означает улучшение одних показателей при условии, чтобы другие не ухудшились.

Определение. Вектор X^* называется эффективным (оптимальным по Парето) решением задачи многокритериальной оптимизации, если не существует такого вектора $\overline{X} \in Q$, что

$$Z_i(\overline{X}) \geq Z_i(\overline{X}^*), i = \overline{1,m},$$

причём хотя бы для одного значения i имеет место строгое неравенство.

Множество допустимых решений, для которых невозможно одновременно улучшить все частные показатели эффективности (т.е. улучшить хотя бы один из них, не ухудшая остальных), принято называть областью Парето, или областью компромиссов, а принадлежащие ей решения – эффективными.

Задания

1. Решить задачу трёхкритериальной оптимизации:

$$Z_1 = -x_1 + 2x_2 \to \max,$$
$$Z_2 = 2x_1 + x_2 \to \max,$$
$$Z_3 = x_1 - 3x_2 \to \max,$$

$$\left.\begin{array}{l} x_1 + x_2 \leq 6 \\ 1 \leq x_1 \leq 3 \\ 1 \leq x_2 \leq 4 \end{array}\right\}.$$

Практическая работа №4
Нелинейное программирование

Из методов математического программирования наиболее сложными считаются методы нелинейного программирования. Задача нелинейного программирования формулируется так же, как и общая задача оптимального программирования:

$$F = f(x_1, x_2, \ldots, x_n) \to \max(\min);$$

$$g_i(x_1, x_2, \ldots, x_n)(\le, =, \ge)d_i, i = \overline{1, m};$$

$$a_j \ge x_j \ge b_j, j = \overline{1, n}.$$

Здесь a_j и b_j — нижнее и верхнее предельно допустимые значения x_j, причём целевая функция F и (или) хотя бы одна из функций g_i являются нелинейными. Если в ограничениях стоит знак равенства, то их называют ещё уравнениями связи.

У произвольной задачи нелинейного программирования отсутствует некоторые или все свойства задачи линейного программирования. В задачах линейного программирования экстремум целевой функции находится только на границе, что не является справедливым для нелинейных задач. Поэтому наибольшее или наименьшее значение функции без учета того, где находится такое значение (внутри заданного интервала или на его границе), называют не экстремумом, а оптимумом. Оптимум — более широкое понятие, чем экстремум. Если экстремум есть не у всех функций, то в практических задачах оптимум существует всегда, причем он может быть как локальным, так и глобальным.

К задачам оптимизации в нелинейном программировании относятся задачи безусловной и условной оптимизации.

Задачами безусловной оптимизации называются такие, в которых задается лишь одна целевая функция *F*, без указания ограничений и граничных условий (эти задачи носят теоретический характер, так как на практике граничные условия задаются всегда). В этих задачах понятия оптимума и экстремума совпадают, и для нахождения оптимума применяют методы нахождения экстремума.

Задачами условной оптимизации называются задачи, в которых кроме целевой функции задаются некоторые дополнительные условия и ограничения. Ограничения могут быть заданы в виде, как уравнений, так и неравенств, при этом введение ограничений либо не влияет на оптимум, либо ухудшает его, подтверждая тем самым вывод, сделанный для задач линейного программирования, что введение дополнительных условий не улучшает оптимального решения, а в ряде случаев приводит к несовместности.

Аналитические методы решения задач безусловной оптимизации заключаются в поиске экстремума функции *F*, который находится в точке с координатами, определяемыми в результате решения следующей системы уравнений:

$$\partial F / \partial x_1 = 0; \partial F / \partial x_2 = 0, \ldots, \partial F / \partial x_n = 0.$$

Эта точка называется стационарной точкой. Для получения достаточных условий существования экстремума следует определить в стационарной точке знак дифференциала второго порядка функции $F(X)=f(x_1,x_2,\ldots,x_n)$:

$$d^2 f(X) = \sum_{i=1}^{n} \sum_{j=1}^{n} f''_{x_i x_j}(X) \Delta x_i \Delta x_j .$$

В стационарной точке X^0 функция $f(X)$ имеет максимум, если $d^2f(X^0)<0$, и минимум, если $d^2f(X^0)>0$, при любых Δx_i и Δx_j, не обращающихся в нуль одновременно.

Вектор столбец, составленный из частных производных функции $F=f(x_1,x_2,\ldots,x_n)$:

$$\Delta f(x_j) = \begin{vmatrix} \partial f / \partial x_1 \\ \partial f / \partial x_2 \\ \dots \\ \partial f / \partial x_n \end{vmatrix},$$

называется градиентом функции $f(x_j)$. Градиент показывает направление наискорейшего увеличения функции. Вектор, противоположенный градиенту, называют антиградиентом, он показывает направление скорейшего уменьшения функции. Численной мерой градиента, как и любого вектора, является модуль, определяемый как:

$$\left|\Delta f(x_j)\right| = \left[(\partial f / \partial x_1)^2 + (\partial f / \partial x_2)^2 + \ldots + (\partial f / \partial x_n)^2\right]^{1/2} = \left[\sum_{j=1}^{n} (\partial f / \partial x_j)^2\right]^{1/2}.$$

Если обозначить модуль градиента через

$$\varepsilon_M = \left[\sum_{j=1}^{n} (\partial f / \partial x_j)^2\right]^{1/2},$$

то признаком экстремума является условие $\varepsilon_M = 0$.

Наиболее простая задача условной оптимизации имеет вид:

$$F = f(x_j) \to \min;\, a_j \le x_j \le b_j;\, j = \overline{1,n},$$

(если решается задача поиска максимума функции F, то это всегда равнозначно поиску минимума функции – F). При решении такой задачи применятся тот же алгоритм, что и в задачах безусловной оптимизации. Для нахождения стационарной точки также решается система уравнений. Если при решении этой системы какое - либо из значений переменной x_j выходит на (или за) нижнюю границу, то в качестве оптимального принимается значение x_j равное a_j, и поиск продолжается по остальным переменным; аналогично осуществляется поиск для верхней границы.

Пусть в общем случае задача условной оптимизации имеет вид:

$$\left.\begin{aligned} &F = f(x_j) \to \min; \\ &g(x_j) = 0; \\ &a_j \le x_j \le b_j,\, j = \overline{1,n}. \end{aligned}\right\}$$

Известен ряд достаточно сложных методов решения подобных задач. Один из таких методов называется методом штрафных функций; суть его заключается в следующем. От задачи условной оптимизации переходят к такой задаче, в которой минимизируется новая целевая функция, включающая, кроме заданной целевой функции $f(x_j)$, заданные ограничения $g(x_j)$. Новая целевая функция записывается следующим образом:

$$\Phi(X) = f(X) + \Psi(g(X)) \to \min(\max),$$

где $\Psi(g(X))$ - штрафная функция.

Штрафная функция может иметь следующий вид:

$$\Psi(g_i(x_j)) = M\sum_{i=1}^{m} g_i^2(x_j),$$

где М — большое число (например, если ожидаемое оптимальное значение целевой функции порядка единиц, то можно принять М= 1000).

Если x_j находится в области допустимых решений, то выполняются все ограничения и, значит, $g_i(x_j)=0$. При этом штрафная функция равняется нулю, и новая целевая функция оказывается равной заданной.

Если x_j, не находится в области допустимых решений, то не выполняются ограничения. Значит, $g_i^2(x_j) > 0$, а большое число М приводит к тому, что в новой целевой функции, штрафная функция оказывается существенно больше заданной целевой функции *f(x_j)*. Поэтому в ходе минимизации благодаря большому градиенту в первую очередь уменьшается штрафная функция, пока она не станет равной нулю (т. е. пока значения x_j не войдут в область допустимых решений или пока не будет найдено допустимое решение), а далее начинается процесс минимизации заданной целевой функции.

Задания

1. Решить задачу условной оптимизации методом штрафных функций:

$$\left.\begin{aligned} &F = x_2 \to \min; \\ &x_2 = 2(x_1 - 2)^2 + 1; \\ &(x_1 - 3)^2 + (x_2 - 0{,}5)^2 \geq 4; \\ &0 \leq x_1 \leq 4; 0 \leq x_2 \leq 3. \end{aligned}\right\}$$

Практическая работа №5
Целочисленное программирование

Задачи оптимизации, решением которых должны быть целые числа, называют задачами целочисленного (дискретного) программирования. В том случае, если ограничения и целевая функция задачи представляют собой линейные зависимости, задачу называют целочисленной задачей линейного программирования, если же хотя бы одна зависимость является нелинейной, задачу называют целочисленной задачей нелинейного программирования.

Огромное количество экономических задач носит дискретный, чаще всего целочисленный характер, что связано с физической неделимостью многих элементов расчёта. В ряде случаев такие задачи решаются обычными методами, например, симплекс- методом, с последующим округлением до целых чисел. Такой метод оправдан, когда отдельная единица составляет очень малую часть всего объёма (например, товарных запасов); в противном случае он может внести значительные искажения в действительно оптимальное решение. Поэтому разработаны специальные методы решения целочисленных задач, среди которых можно выделить два направления: методы отсечения (отсекающих плоскостей) и комбинаторные методы.

Метод отсекающих плоскостей (известный как метод Гомори) состоит в построении дополнительных ограничений и применении двойственного симплекс-метода. Представление о комбинаторных методах даёт широко используемый на практике метод ветвей и границ.

С помощью метода ветвей и границ решаются задачи целочисленного программирования, в которых как целевая функция, так и функции в системе ограничений являются линейными. В общем виде любая из этих задач может быть записана следующим образом: найти максимум функции

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

при условиях:

$$\sum_{j=1}^{n} a_{ij} x_j = b_i (i = \overline{1,m}), x_j \geq 0,$$

$$x_j \equiv 0 (j = \overline{1,n}).$$

Как и при решении задачи методом Гомори, находится оптимальный план задачи без учёта целочисленности переменных с помощью симплекс-метода. Пусть им является план X_0. Если среди компонент этого плана нет дробных чисел, то найдено искомое оптимальное решение задачи. Если среди компонента плана X_0 имеются дробные числа, то X_0 не удовлетворяет условию целочисленности и необходимо осуществить упорядоченный переход к новым планам, пока не будет найдено решение задачи, причём для всякого последующего плана $X - F(X_0) \geq F(X)$. Так как у плана X_0 имеются дробные числа, то пусть, например, это будет компонента x_{i0}. Тогда в оптимальном целочисленном плане её значение будет, по крайней мере, либо меньше, либо равно ближайшему меньшему целому числу K_{i0}, либо больше, либо равно ближайшему целому числу $K_{i0}+1$. Для нахождения этих чисел, решаются симплекс-методом следующие две задачи линейного программирования:

$$(I)\begin{cases} F = \sum\limits_{j=1}^{n} c_j x_j \to \max, \\ \sum\limits_{j=1}^{n} a_{ij} x_j = b_i (i = \overline{1,m}), \\ x_{i0} \leq K_{i0}, \\ x_j \geq 0 (j = \overline{1,n}); \end{cases}$$

$$(II)\begin{cases} F = \sum\limits_{j=1}^{n} c_j x_j \to \max, \\ \sum\limits_{j=1}^{n} a_{ij} x_j = b_i (i = \overline{1,m}), \\ x_{i0} \geq K_{i0} + 1, \\ x_j \geq 0 (j = \overline{1,n}); \end{cases}$$

Здесь возможен один из следующих четырёх случаев:

1. Одна из задач неразрешима, а другая имеет целочисленный оптимальный план. Тогда этот план и значение целевой функции на нём и дают решение исходной задачи.
2. Одна из задач неразрешима, а другая имеет оптимальный план, среди компонент которого есть дробные числа. Тогда рассматривается вторая задача и в её оптимальном плане выбирается одна из компонент, значение которой равно дробному числу, и строятся две задачи, аналогичные задачам (I) и (II).
3. Обе задачи разрешимы. Одна из задач имеет целочисленный оптимальный план, а в оптимальном плане другой задачи есть дробные числа. Тогда вычисляются значения целевой функции на этих планах и сравниваются между собой. Если на целочисленном оптимальном плане значение целевой функции больше или равно её значению на плане, среди компонент которых есть дробные числа, то данный целочисленный план является оптимальным для исходной задачи, и он, вместе со значением целевой функции на нём даёт искомое решение. Если значение целевой функции больше на плане, среди компонент которого есть дробные числа, то следует взять одно из таких чисел и для задачи, план которой рассматривается, необходимо построить две задачи, аналогичные задачам (I) и (II).
4. Обе задачи разрешимы, и среди оптимальных планов обеих задач есть дробные числа. Тогда вычисляются значения целевой функции на этих планах и сравниваются между собой. Далее рассматривается та из задач, для которой значение целевой функции больше, и в её оптимальном плане выбирается одна из компонент, значение которой равно дробному числу, и строятся две задачи, аналогичные задачам (I) и (II).

Таким образом, описанный выше итерационный процесс может быть представлен в виде некоторого дерева, на котором исходная вершина отвечает оптимальному плану X_0, а каждая соединённая с ветвью вершина отвечает оптимальным планам задач (I) и (II). Каждая из этих вершин имеет свои ветвления. На каждом шаге выбирается та вершина, для которой значение целевой функции наибольшее. Если на некотором шаге будет получен план, имеющих целочисленные компоненты, и значение функции на нём будет больше или равно значений функции в других возможных для ветвления вершинах, то данный план является оптимальным планом исходной задачи целочисленного программирования и значение целевой функции на нём является оптимальным.

Метод ветвей и границ имеет более простую логическую схему расчётов, чем метод Гомори и более удобен при решении с использованием компьютера. Схему расчёта задач на основе метода ветвей и границ часто называют принятием решений на дереве возможных вариантов.

Задания

1. Методом ветвей и границ найти решение задачи, состоящей в определении максимального значения функции:

$$F = 2x_1 + x_2$$

при условиях

$$\begin{cases} 6x_1 + 4x_2 + x_3 = 24, \\ 3x_1 - 3x_2 + x_4 = 9, \\ -x_1 + 3x_2 + x_3 = 3, \end{cases}$$

$$x_j \geq 0, x_j \equiv 0 (j = \overline{1,5})$$

Практическая работа №6
Сетевые модели

Сетевой моделью (другие названия: сетевой график, сеть) называется экономико-математическая модель, отражающая комплекс работ (операций) и событий, связанных с реализацией некоторого проекта, в их логической и технологической последовательности и связи. Анализ сетевой модели, представленной в графической или табличной (матричной) форме, позволяет, во-первых, более чётко выявить взаимосвязи этапов реализации проекта и, во-вторых, определить наиболее оптимальный порядок выполнения этих этапов в целях, например, сокращения сроков выполнения всего комплекса работ. Таким образом, методы сетевого моделирования относятся к методам принятия оптимальных решений.

Математический аппарат сетевых моделей базируется на теории графов. Графом называется совокупность двух конечных множеств: множества точек, которые называются вершинами (отображаются кружочками, точками и др.), и множества пар вершин, которые называются ребрами (дугами, соединяющими вершины графа). Если рассматриваемые пары вершин являются упорядоченными, т.е. на каждом ребре задаётся направление, то граф называется ориентированным; в противном случае – неориентированным. Последовательность неповторяющихся рёбер, ведущая от некоторой вершины к другой, образует путь. Граф называется связным, если для любых двух его вершин существует путь, их соединяющий, в противном случае граф называется несвязным. В экономике чаще всего используются два вида графов: дерево и сеть. Дерево представляет собой связный граф без циклов, имеющий исходную вершину (корень) и крайние вершины; пути от исходной вершины к крайним вершинам называются ветвями. Сеть – это ориентированный конечный связный граф, имеющий начальную вершину (источник) и конечную вершину (сток). Таким образом, сетевая модель представляет собой граф вида «сеть».

С помощью сетевого планирования и управления (СПУ) решаются различные оптимизационные экономические задачи, связанные с пространственным перемещением объектов, временным исполнением работ субъектами и др. Объектом управления в СПУ являются коллективы исполнителей, располагающих определёнными ресурсами и выполняющих определённый комплекс операций, который призван обеспечить достижение намеченной цели, например, разработку нового изделия. Основой СПУ является сетевая модель (СМ), в которой моделируется совокупность взаимосвязанных работ и событий отражающих процесс достижения определённой цели. Она может быть представлена в виде графика или таблицы.

Основные понятия сетевой модели: событие, работа и путь.

Работа характеризует материальное действие, требующее использования ресурсов, или логическое, требующее лишь взаимосвязи событий. При графическом представлении изображается стрелкой, которая соединяет два события. Она обозначается парой заключённых в скобки чисел (i,j), где i – номер события, из которого работа выходит, а j – номер события, в которое она входит. Работа не может начаться раньше, чем свершится событие, из которого она выходит. Каждая работа имеет предельную продолжительность t(i,j). Например, запись t(2,5)=4 означает, что работа (2,5) имеет продолжительность 4 единицы. К работам относятся также такие процессы, которые не требуют ни ресурсов, ни времени выполнения. Они заключаются в установлении логической взаимосвязи работ и показывают, что одна из них непосредственно зависит от другой; такие работы называются фиктивными и в графике изображаются пунктирными стрелками.

Событиями называются результаты выполнения одной или нескольких работ. Они не имеют протяжённости во времени. Они не имеют протяжённости во времени. Событие свершается в тот момент, когда оканчивается последняя из работ, входящая в него. События обозначаются одним числом и при графическом представлении СМ изображаются кружком (или иной геометрической фигурой), внутри которого проставляется его порядковый номер (i=1,2,…,N). В СМ имеется начальное событие (с номером 1), из которого работы только выходят, и конечное событие (с номером N), в которое работы только входят.

Путь – это цепочка следующих друг за другом работ, соединяющих начальную и конечную вершины, например, в приведённой выше модели путями являются L_1=(1,2,3,7,10,11), L_2=(1,2,4,6,11) и др. Продолжительность пути определяется суммой продолжительностей составляющих его работ. Путь, имеющий максимальную длину, называют критическим и обозначают $L_{кр}$, а его продолжительность – $t_{кр}$. Работы, принадлежащие критическому пути, называются критическими. Их несвоевременное выполнение ведёт к срыву всего комплекса работ.

СМ имеет ряд характеристик, которые позволяют определить степень направленности выполнения отдельных работ, а также всего их комплекса и принять решение о перераспределении ресурсов. Однако перед расчётом СМ следует убедиться, что она удовлетворяет следующим основным требованиям:

1. События правильно пронумерованы, т.е. для каждой работы (i,j) i<j. При невыполнении этого требования необходимо использовать алгоритм перенумерации событий, который заключается в следующем:

- нумерация событий начинается с исходного события, которому присваивается №1;

- из исходного события вычёркиваются все исходящие из него работы (стрелки), и на оставшейся сети находят событие, в которое не входит ни одна работа, ему и присваивается №2;

- затем вычёркивают работы, выходящие из события №2, и вновь находят событие, в которое не входит ни одна работа, и ему присваивают №3, так продолжается до завершающего события, номер которого должен быть равен количеству событий в сетевом графике.

- если при очередном вычёркивании работ одновременно несколько событий не имеют, входящих в них работ, то их нумеруют очередными номерами в произвольном порядке.

2.Отсутствуют тупиковые события (кроме завершающего его), т.е. такие, за которыми не следует хотя бы одна работа.

3. Отсутствуют события (за исключением исхода его), которым не предшествует хотя бы одна работа.

4. Отсутствуют циклы, т.е. замкнутые пути, соединяющие событие с ним же самим.

Для событий рассчитывают три характеристики: ранний и поздний срок совершения события, а также его резерв.

Ранний срок свершения события определяется величиной наиболее длительного отрезка пути от исходного до рассматриваемого события, причём $t_P(1)= 0$, и $t_P(N)= t_{KP}(L)$:

$$t_P(j) = \max_i \{t_P(i) + t(i,j)\};\ j = \overline{2, N}.$$

Поздний срок свершения события характеризует самый поздний допустимый срок, к которому должно свершится событие, не вызывая при этом срыва срока конечного события:

$$t_П(i) = \min_i \{t_П(j) - t(i,j)\};\ i = \overline{2, N-1}.$$

Этот показатель определяется «обратным ходом», начиная с завершающего события, с учётом соотношения $t_П(N)=t_P(N)$.

Все события, за исключением событий, принадлежащих критическому пути, имеют резервы $R(i)$:

$$R(i) = t_П(i) - t_P(i).$$

Резерв показывает, на какой предельно допустимый срок можно задержать наступление события, не вызывая при этом увеличения срока выполнения всего комплекса работ.

Для всех работ (i,j) на основе ранних и поздних сроков свершения всех событий, можно определить показатели:

ранний срок начала –

$$t_{PH}(i,j) = t_P(i),$$

ранний срок окончания –

$$t_{PO}(i,j) = t_P(i) + t(i,j),$$

поздний срок окончания –

$$t_{ПO}(i,j) = t_П(j),$$

поздний срок начала -

$$t_{ПH}(i,j) = t_П(j) - t(i,j),$$

полный резерв времени –

$$R_П(i,j) = t_П(j) - t_P(i) - t(i,j),$$

независимый резерв времени -

$$R_H(i,j) = \max\{0, t_P(j) - t_П(i) - t(i,j)\} =$$
$$= \max\{0, R_П(i,j) - R(i) - R(j)\}.$$

Полный резерв времени показывает, на сколько можно увеличить время выполнения конкретной работы при условии, что срок выполнения всего комплекса работ не изменится.

Независимый резерв времени соответствует случаю, когда все предшествующие работы заканчиваются в поздние сроки, а все последующие начинаются в ранние сроки. Использование этого резерва не влияет на величину резервов времени других работ.

Путь характеризуется двумя показателями – продолжительностью и резервом. Продолжительность пути определяется суммой продолжительностей составляющих его работ. Резерв определяется как разность между длинами критического и рассматриваемого путей. Из этого определения следует, что работы, лежащие на критическом пути, и сам критический путь имеют нулевой резерв времени. Резерв времени пути показывает, насколько может увеличиться продолжительность работ, составляющих данный путь, без изменения продолжительности общего срока выполнения всех работ.

Наиболее распространенными на практике сетевыми моделями и задачами являются – задача коммивояжера, задача поиска кратчайшего пути, задача о распределении потоков в сетях.

Задание

1. Постройте сетевую модель программы опроса общественного мнения, которая включает разработку (A; 1 день) и распечатку анкет (B; 0,5 дня), прием на работу (C; 2 дня) и обучение (D; 2 дня) персонала, выбор опрашиваемых лиц (E; 2 дня), рассылку им анкет (F; 1 день) и анализ полученных данных (G; 5 дней).

Раздел 2. Лабораторные работы

Весь необходимый для выполнения лабораторных работ теоретический материал приведен в Разделе 1 и в источниках из списка литературы.

В рамках выполнения лабораторных работ необходимо решить задачи из лабораторных и подготовить отчет.

Лабораторная работа №1
Математические методы и их применение при принятии управленческих решений

Задание 1. Запрограммировать решение следующей задачи.
Для заданной функции F(x) найти точки экстремума:

a) $y = x_1^2 - 2x_1x_2 + 2x_2^2 - x_1 - 2x_2$;

b) $y = x_1^2 - x_2^2$;

c) $y = x_1^2 - 4x_1x_2 + x_2^2$;

Лабораторная работа №2
Линейное программирование и теория двойственности

Задание 1. Запрограммировать решение приведенной ниже задачи. Для решения использовать симплекс-метод.

Задача. Предприятие выпускает 2 вида продукции и использует 3 типа основного оборудования: токарное, фрезерное, шлифовальное. Затраты времени на изготовление единицы продукции для каждого из типов оборудования приведены в таблице 1. В ней указаны общий фонд рабочего времени каждого из типов оборудования, а также прибыль от реализации одного изделия данного вида. Определить такой объем выпуска каждого из изделий, при котором общая прибыль от их реализации максимальна.

Таблица 1.

Тип оборудования	Затраты времени на единицу продукции вида		Общий ресурс рабочего времени
	1	2	
Токарное	1	3	300
Фрезерное	2	1	180
Шлифовальное	1	-	80
Прибыль	2	3	

Задание 2. Составить задачу, двойственную к задаче из задания №1.

Лабораторная работа №3
Задачи многокритериальной оптимизации

Задание. Решить задачу многокритериальной оптимизации и запрограммировать это решение:

$$\begin{cases} 3x_1 + x_2 \le 12 \\ 5x_1 - x_2 \le 15 \\ 0 \le x_1, -10 \le x_2 \le 10 \end{cases}$$

$$Z_1(x) = 3x_1 + 6x_2 \to \max, \delta_1 = 32$$

$$Z_2(x) = -3x_1 + 4x_2 \to \min, \delta_2 = 50\%$$

$$Z_3(x) = -x_1 + 2x_2 \to \max$$

Лабораторная работа №4
Целочисленное программирование

Задание 1. Решить задачу целочисленного программирования методом Гомори, методом ветвей и границ и запрограммировать это решение:

$$\begin{cases} 6x_1 + 3x_2 \ge 12 \\ x_1 - x_2 \le 4 \\ x_1 \ge 0 \end{cases}$$

$$f(x) = x_1 - 4x_2 \to \max$$

Раздел 3. Самостоятельная работа

3.1. Проработка лекционного материала по темам лекций.
3.2. Подготовка к практическим работам по темам из Раздела 1.
3.3. Оформление отчетов по лабораторным работам.

Список литературы

1. Голубева Н.В. Математическое моделирование систем и процессов [Электронный ресурс] : учеб. пособие — Электрон. дан. — Санкт-Петербург : Лань, 2016. — 192 с., https://e.lanbook.com/book/76825, дата обращения: 13.05.2018

2. Пантелеев, А.В. Методы оптимизации в примерах и задачах [Электронный ресурс] : учеб. пособие / А.В. Пантелеев, Т.А. Летова. — Электрон. дан. — Санкт-Петербург : Лань, 2015. — 512 с., https://e.lanbook.com/book/67460, дата обращения: 13.05.2018