

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Локтионова Оксана Геннадьевна
Должность: проректор по учебной работе
Дата подписания: 25.09.2023 23:07:00
Уникальный программный ключ:
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра биомедицинской инженерии



Практикум по дисциплине «Методы оптимизации и принятия проектных решений»

для студентов направления 12.04.04 «Биотехнические системы и
технологии»

Курск 2023

УДК 004+007+53.05+76.03+76.13.15+519.711.3
Составитель: М.В. Артёменко

Рецензент

Доктор технических наук, профессор И. Е. Чернецкая

Практикум по дисциплине «Методы оптимизации и принятия проектных решений» / Юго-Зап. гос. ун-т; сост.: М.В. Артеменко. Курск, 2023. 51 с.

Практикум содержит методические рекомендации для выполнения практических работ по дисциплине «Методы оптимизации и принятия проектных решений». Указывается порядок и объем практических работ и структура отчета. Приведены краткие теоретические сведения по темам условной минимизации, условия оптимизации Куна-Таккера, метод штрафных функций, принятие оптимальных решений в условиях неопределенности, поведение типовых функций принадлежности, построение решающих правил, структура и поддержка САПР и АСППР

Предназначено для студентов направления 12.04.04 «Биотехнические системы и технологии».

Текст печатается в авторской редакции

Подписано в печать _____ . Формат 60x84 1/16.
Усл.печ.л. _____. Уч.-изд.л. _____. Тираж _____ экз. Заказ: _____. Бесплатно.
Юго-Западный государственный университет.
305040, г.Курск, ул. 50 лет Октября, 94.

Содержание

Модуль 1. Условная минимизация, метод множителей Лагранжа, условия оптимальности Куна-Таккера.....	5
Модуль 2: Метод штрафных функций и барьерный метод	8
Модуль 3. Принятие решений в условиях неопределенности. Исследование поведения типовых функций принадлежности.....	14
Модуль 4. Разработка плана реализации управленческого решения	18
Модуль 5. Поведение типовых функций принадлежности.	19
Модуль 6. Построение решающих правил.....	29
Модуль 7: Типовая структура и методы принятия решений в САПР.....	36
Модуль 8. Типовая структура автоматизированной системы поддержки принятия решений	40

Модуль 1. Условная минимизация, метод множителей Лагранжа, условия оптимальности Куна-Таккера

1 Условная минимизация

Системы алгебраических уравнений часто представляют условия минимума или стационарности для некоторых функционалов (условия Эйлера), возникающих в задачах вариационного исчисления и в теории оптимизации процессов и конструкций. Задачи о минимизации функционалов составляют предмет теории математического (линейного и нелинейного) программирования.

Сначала рассмотрим задачи минимизации линейных функционалов. Несмотря на линейность рассматриваемых функционалов их минимизация является существенно нелинейной задачей, так как множество (область) допустимых решений определяется набором некоторых ограничений типа равенств и неравенств. Например, вес летательного аппарата представляется суммой весов его составных частей. Аргументами функционала веса являются параметры геометрии, удельные веса материалов и тому подобные характеристики, каждая из которых имеет определенные границы изменения, зависящие возможно от значений других параметров и от существующих технологий а авиапромышленности.

Минимум функционала может быть неединственным, задача его поиска является нетривиальной и, конечно же, не сводится к решению системы линейных уравнений.

Другой пример приложения теории линейного программирования дает задача поиска равнопрочных конструкций, все составляющие которых имеют одинаковый запас прочности, т.е. отношение предела прочности к максимальному напряжению при эксплуатации. Такие функционалы как правило не имеют аналитического представления и определяются алгоритмически по результатам решения вспомогательных краевых задач. Каноническая формулировка задачи линейного программирования имеет вид: найти $x \mid \min_x (c^T * x)$ при ограничениях $A \cdot x = b$, $x \geq 0$ где c^T - заданный вектор размерности n , n - размерность вектора

неизвестных x , A является матрицей m на n и $m < n$, m – число ограничений. Записанная выше в предикатах (в конструкциях математической логики) формулировка читается так: найти x такой, что на нем достигается минимум целевого линейного функционала $(c^T * x)$ при ограничениях $A \cdot x = b$ и положительных $x \geq 0$. В формулировке задачи вертикальная черта заменяет слова «такой что» или «обеспечивающий».

Каждое из ограничений равенств определяет $(n-1)$ -мерную плоскость, пересечение которой с областью допустимых значений неизвестных Q , определяемой неравенствами ($x \geq 0$), дает выпуклый $(n-1)$ -мерный многогранник G . Минимальное значение целевого функционала достигается в некоторой вершине многогранника G (при вырождении оно может достигаться во всех точках ребра (линии пересечения граней) или грани многогранника). Для решения задачи линейного программирования достаточно найти координаты вершины с наименьшим значением целевого функционала $(c^T * x)$. В практических задачах количество вершин многогранника G так велико, то просмотр их перебором даже с использованием ЭВМ невозможен. Поэтому разработаны специальные численные методы решения задач линейного программирования. Наиболее распространенным является симплекс-метод. Идея симплекс-метода состоит в следующем. Вначале рассматривается некоторая вершина многогранника G и все ребра, выходящие из этой вершины. Далее при поиске минимума перемещаются вдоль того из ребер, по которому функция убывает, и попадают в следующую вершину. Находят выходящие из нее ребра и повторяют процесс. Когда приходят в такую вершину, в которой вдоль всех выходящих из нее ребер функция возрастает, то минимум найден. Отметим, что, выбирая одно ребро, исключают из рассмотрения вершины, лежащие на остальных траекториях. В результате количество рассматриваемых вершин резко сокращается.

Методы минимизации нелинейных функционалов общего вида при наличии ограничений составляют предмет теории нелинейного программирования. Постановка общей задачи нелинейного программирования имеет вид: найти $x \mid \min_x F(x)$

при ограничениях $c_i(x) \geq 0$, $i=1,2,\dots,m$, где размерность вектора неизвестных x равна n и $n > m$, m – число ограничений. При $m=0$ имеем задачу безусловной минимизации, в противном случае решаем задачу условной минимизации. Иногда ограничения можно учесть явно. Это можно сделать, например за счет специальной замены переменных, при которой ограничения выполняются автоматически. Например, в задачах о течениях несжимаемой жидкости ограничение, выражающее условие несжимаемости (равенство дивергенции скорости нулю), можно учесть явно с самого начала, записывая уравнения Навье-Стокса в переменных функция тока – завихренность. Более общие методы неявного учета ограничений сводят задачу условной минимизации к последовательности задач безусловной минимизации. Такие методы называются также методами преобразования. Основоположителем методов преобразования является американский математик Курант (1943). Среди множества методов преобразования наиболее употребительными являются метод множителей Лагранжа, метод штрафных функций и барьерный метод [3].

Модуль 2: Метод штрафных функций и барьерный метод

В методе штрафных функций вместо отыскания решения x^* исходной задачи условной минимизации функционала $F(x)$ рассматривается модифицированная задача безусловной минимизации функционала

$$T(x,s) = F(x) + F(c(x),s)$$

где s - вектор управляющих параметров, а F – положительная штрафная функция, регулируемая вектором s . Безусловный локальный минимум функционала T по x обозначается $x(s)$. Различные методы преобразования отличаются выбором штрафного функционала F и последовательности управлений $s^{(k)}$, обеспечивающих сходимость $x(s^{(k)})$ к x^* при $k \rightarrow \infty$.

В методе штрафных функций приближенные решения могут не принадлежать допустимому множеству пробных решений, то есть ограничения типа равенств выполняются с погрешностью, которая постепенно стремится к нулю с ростом k .

Примером метода штрафных функций в механике несжимаемой жидкости является введение искусственной сжимаемости и поиск давления в виде $p = -\lambda \nabla \cdot u$, $\lambda \gg 1$ - коэффициент штрафа.

В методе барьерных функций для выполнения ограничений типа неравенств функция F подбирается так, чтобы на границе допустимого множества “построить барьер”, препятствующий нарушению ограничений в процессе безусловной минимизации функции T по x , и чтобы точки $x(s^{(k)})$ сходились к x^* изнутри допустимого множества R . Проще говоря в этом методе чем сильнее нарушение ограничений, тем больше значение минимизируемого функционала, то есть нарушать ограничения можно, но невыгодно.

Конкретные примеры штрафных и барьерных функций приводятся далее при рассмотрении способов учета ограничений в задачах о несжимаемых средах, в контактных задачах, в задачах построения сеток и управления подвижными сетками [3].

2 Метод множителей Лагранжа

Обозначим множество номеров активных ограничений (обращающихся в нуль на решении x^*) символом J . Допустимое множество значений

$$R = \{ x: c_i(x) \geq 0 \ (i = 1, \dots, m) \}$$

где записанное следует читать так: множество R состоит из элементов x таких, что $c_i(x) \geq 0$ для $i=1, \dots, m$.

Если x^* является решением задачи нелинейного программирования, то найдется вектор λ^* размерности m такой, что

$$g(x^*) - \sum_{i=1}^m \lambda_i^* a_i(x^*) = 0$$

$$\lambda_i^* \geq 0, i \in J$$

$$\lambda_i^* = 0, i \notin J$$

где через g и a_i обозначены градиенты функций F и c_i . Этот результат, известный как теорема Куна-Таккера, распространяется и на случай ограничений типа равенств, только в этом случае соответствующие множители Лагранжа λ_i^* могут иметь произвольный знак. При дополнительном требовании независимости градиентов функций ограничений, активных на x^* , множители Лагранжа определяются однозначно.

Учет ограничений с помощью множителей Лагранжа носит название метода множителей Лагранжа. Хорошим примером этого способа решения являются расчеты несжимаемых сред, в которых давление играет роль множителя Лагранжа для условия не сжимаемости [3].

Рассмотрим общую задачу оптимизации, содержащую несколько ограничений в виде равенств: минимизировать $f(x)$ при ограничениях $g_j(x)=0, j=1, \dots, k$.

Эта задача может быть решена как задача безусловной оптимизации, полученная путем исключения из целевой функции k независимых переменных с помощью заданных равенств. Наличие ограничений в виде равенств фактически позволяет уменьшить размерность исходной задачи с n до $n-k$. В качестве иллюстрации рассмотрим следующий пример.

Минимизировать $f(x) = x_1 * x_2 + x_3$ при ограничении: $g(x) = x_1 + x_2 + x_3 - 1 = 0$. Исключив переменную x_3 с помощью уравнения $g(x) = 0$, получим оптимизационную задачу с двумя переменными без ограничений: $f(x_1, x_2) = x_1 * x_2 + (1 - x_1 - x_2)$.

Метод исключения переменных применим лишь в тех случаях, когда уравнения, представляющие ограничения, можно разрешить относительно некоторого конкретного набора независимых переменных. При наличии большого числа ограничений в виде равенств, процесс исключения переменных становится весьма трудоемкой процедурой. Кроме того, возможны ситуации, когда уравнения не удастся разрешить относительно переменной. В частности, если в приведенном примере ограничения $g(x)=0$ задать в виде $g(x)=x_1^2 \cdot x_3^2 + x_2 \cdot x_3^2 + x_2^{-1} \cdot x_1$, то получить аналитическое выражение какой-либо из переменных через другие не представляется возможным. Таким образом, при решении оптимизационных задач, содержащих сложные ограничения в виде равенств, целесообразно использовать метод множителей Лагранжа. С помощью этого метода находят необходимые условия, позволяющие идентифицировать точки оптимума в задачах оптимизации с ограничением в виде равенств. При этом задача с ограничением в виде равенств, преобразуется в эквивалентную задачу безусловной оптимизации.

Рассмотрим задачу, имеющую несколько ограничений в виде равенств: минимизировать $f(x)$ при ограничениях $g_j(x)=0$, при $j=1,2,\dots,k$.

В соответствии с методом множителей эта задача преобразуется в следующую задачу безусловной оптимизации:

минимизировать $L(x, \vartheta) = f(x) - \sum_{j=1}^k \vartheta_j \cdot g_j$,

где $L(x, \vartheta)$ - функция Лагранжа, ϑ_j - множители Лагранжа.

На знак ϑ_j никаких требований не накладывается.

Приравниваем частные производные $L(x, \vartheta)$ по x к нулю, получаем следующую систему n уравнений с n неизвестными:

$$\frac{\partial L(x, \vartheta)}{\partial x_1} = 0, \quad \dots, \quad \frac{\partial L(x, \vartheta)}{\partial x_n} = 0.$$

Если найти решение приведенной выше системы в виде функций вектора оказывается затруднительным, то можно расширить систему путем включения в нее ограничений в виде равенств:

$$g_1(x)=0,$$

$$g_2(x) = 0,$$

.....

$$g_k(x) = 0.$$

Решение расширенной системы, состоящей из $n+k$ неизвестными, определяет стационарную точку функции L .

Пример Минимизировать $f(x) = x_1^2 + x_2^2$ при ограничении $g(x) = 2x_1 + x_2 - 2 = 0$.

Соответствующая задача безусловной оптимизации записывается в следующем виде.

Минимизировать $L(x, \vartheta) = x_1^2 + x_2^2 - \vartheta \cdot (2 \cdot x_1 + x_2 - 2)$

Приравняв две компоненты градиента L к нулю, получим:

$$\frac{\partial L}{\partial x_1} = 2 \cdot x_1 - 2 \cdot \vartheta = 0, \quad x_1^* = \vartheta;$$

$$\frac{\partial L}{\partial x_2} = 2 \cdot x_2 - \vartheta = 0, \quad x_2^* = \frac{\vartheta}{2}.$$

Поставив полученные значения x_1^* и x_2^* в уравнение $2x_1 + x_2 = 2$,

получим, $2 \cdot \vartheta + \frac{\vartheta}{2} = 2$, т.е. $\vartheta = \frac{4}{5}$. Следовательно, $x_1^* = \frac{4}{5}, x_2^* = \frac{2}{5}$. [2]

3 Условия оптимальности Куна-Таккера

Условная оптимизация, то есть решение задачи

$$\begin{cases} f(x) \longrightarrow \inf, \\ x \in X \end{cases}$$

где X – некоторое непустое множество, называемое допустимым.

Причем будем считать, что X определяется ограничениями в виде неравенств и равенств нулю некоторых достаточно гладких функций.

$$\begin{cases} f(x) \longrightarrow \inf, \\ g_i(x) \leq 0, \quad i = 1, \dots, m \\ h_j(x) = 0, \quad j = 1, \dots, l \end{cases} \quad (3.1)$$

Ясно, что тут допустимое множество

$$X = \{x \in \mathbb{R}^n \mid g_i(x) \leq 0, h_j(x) = 0, \quad i = 1, \dots, m, \quad j = 1, \dots, l\}.$$

Если ограничения-неравенства в некоторой точке x обращаются в равенства, то соответствующие ограничения называют активными в данной точке, а совокупность всех таких индексов $R(x)$ – множеством индексов активных ограничений

$$R(x) = \{i \in \{1, \dots, m\} \mid g_i(x) = 0\}.$$

Методы безусловной оптимизации базировались на необходимом условии экстремума (равенство градиента нулю), поэтому вначале сформулируем необходимые условия экстремума для задачи (3.1).

Теорема 1. Пусть $x^* \in X$ и градиенты $g'_i(x^*)$, $h'_j(x^*)$ при $i \in R(x^*)$, $j=1, \dots, l$ образуют линейно независимую систему векторов, тогда для того, чтобы точка x^* была решением задачи (3.1) необходимо, чтобы нашлись $\lambda_i \geq 0$, $i = 1, \dots, m$ и μ_j , $j = 1, \dots, l$ произвольного знака, такие что

$$\begin{cases} f'(x^*) + \sum_{i=1}^m \lambda_i g'_i(x^*) + \sum_{j=1}^l \mu_j h'_j(x^*) = 0, \\ \lambda_i g_i(x^*) = 0, \quad i = 1, \dots, m. \end{cases}$$

Аналогично можно сформулировать эти условия и для частных случаев задачи (3.1).

$$\begin{cases} f(x) \longrightarrow \min, \\ g_i(x) \leq 0, \quad i = 1, \dots, m. \end{cases} \quad (3.2)$$

Тут допустимое множество

$$X = \{x \in \mathbb{R}^n \mid g_i(x) \leq 0, \quad i = 1, \dots, m\}.$$

Теорема 2. Пусть $x^* \in X$ и градиенты $g'_i(x^*)$ при $i \in R(x^*)$ образуют линейно независимую систему векторов, тогда для того, чтобы точка x^* была решением задачи (3.2) необходимо, чтобы нашлись $\lambda_i \geq 0$, $i = 1, \dots, m$, такие что

$$\begin{cases} f'(x^*) + \sum_{i=1}^m \lambda_i g'_i(x^*) = 0, \\ \lambda_i g_i(x^*) = 0, \quad i = 1, \dots, m. \end{cases}$$

Если же все ограничения имеют вид равенств, то приходим к известным из курса математического анализа условиям.

$$\begin{cases} f(x) \longrightarrow \min, \\ h_j(x) = 0, \quad j = 1, \dots, l. \end{cases} \quad (3.3)$$

Здесь допустимое множество

$$X = \{x \in \mathbb{R}^n \mid h_j(x) = 0, \quad j = 1, \dots, l\}.$$

Теорема 3. Пусть $x^* \in X$ и градиенты $h'_j(x^*)$ при $j=1, \dots, l$ образуют линейно независимую систему векторов, тогда для того, чтобы точка x^* была решением задачи (3.3) необходимо, чтобы нашлись $\mu_j, j = 1, \dots, l$ произвольного знака, такие что

$$f'(x^*) + \sum_{j=1}^l \mu_j h'_j(x^*) = 0 \quad [1].$$

Пример. Рассмотрим простой пример

$$\begin{cases} f(x) = x^2 \longrightarrow \min, \\ g(x) = x - 1 \leq 0. \end{cases}$$

Ясно, что минимум здесь достигается в нуле. Попробуем получить эту точку с помощью условий Куна-Таккера. Выписывая эти условия, получаем

$$\begin{cases} 2x + \lambda = 0, \\ \lambda(x - 1) = 0, \\ \lambda \geq 0. \end{cases}$$

Здесь возможны два случая

$$\begin{cases} 2x + \lambda = 0, \\ \lambda = 0, \\ x - 1 \leq 0, \end{cases} \quad \text{либо} \quad \begin{cases} 2x + \lambda = 0, \\ x - 1 = 0, \\ \lambda \geq 0. \end{cases}$$

В первой из этих двух систем условие $x-1 \leq 0$ необходимо, чтобы точка оставалась допустимой (принадлежала допустимому множеству X). Множество решений второй системы пусто. Из первой же получаем $x=0, \lambda=0$ [4].

Модуль 3. Принятие решений в условиях неопределенности. Исследование поведения типовых функций принадлежности.

1. Понятие, сущность и виды управленческих решений

Принятие решений является функцией каждого человека в личной жизни и функцией руководителя любого уровня в процессе управления.

Одним из основных отличительных признаков управленческого решения является то, что оно принимается при наличии назревшей проблемы. А поскольку такие проблемы возникают при управлении любым объектом постоянно, то функция принятия решений заключается в постоянном решении в процессе управления различными задачами.

Задача принятия решений направлена на определение наилучшего варианта действий для достижения поставленных целей.

Принятие решений - это процесс, который начинается с возникновения проблемной ситуации и заканчивается выбором решения - действия по устранению проблемной ситуации

Выполнение функций управления можно представить в виде последовательно сменяющихся друг друга этапов действий:

- сбора, обработки и анализа информации о состоянии объекта управления и системы управления им;
- определения цели функционирования и выработки управленческого решения;
- доведения решения до исполнителя;
- реализации решения и изменений в системе.

Существуют разные подходы и особенности управленческих решений: научные, социально-политические, организационно-технологические, социально-психологические.

Научный подход к принятию решения предполагает наличие соответствующей теории, совокупность практических рекомендаций, вытекающих из теории и опыта ее применения, комплексное использование всех средств для принятия решения обучение теории и практике принятия решений. Руководитель более объективно способен оценивать проблемную ситуацию, учитывать имеющиеся ресурсы и ограничения, формулировать и

анализировать варианты решений, выбирать из них оптимальное решение и предвидеть его возможные последствия.

Социально-политическое направление предполагает исследования социальной сущности решений, определяемые типом общественно-экономической системы, ролью и функциями государства в управлении объектами экономики.

Организационно-технологическое направление - исследование методов и технологий подготовки и принятия решений. Организационно-технологические особенности управленческих решений представляют собой последовательность принятия решений и определяются средствами, методами их обоснования.

Психологическое направление - исследование мыслительной деятельности человека, роль мотивов его поведения, эмоции и воля в процессе принятия и реализации решения.

Социально-психологические особенности управленческих решений не исключают интуицию, выявление и анализ проблем, определение целей их решения, формирование и оценка альтернативных вариантов решений. Выполнить их могут только руководитель и специалист, которые обладают необходимыми знаниями и опытом.

При классификации решений вид решения определяется исходя из характера задачи принятия решений. Общепринятыми критериями деления задач принятия решений и соответственно решений являются следующие:

- степень определенности информации;
- количество лиц, принимающих решение;
- степень охвата объекта управления;
- длительность действия и характер целей;
- содержание задачи принятия решений;
- количество целей.

По длительности действия и характеру целей решения могут быть:

- Тактические решения, как правило, обеспечивают реализацию стратегических задач. По времени они не превышают одного года.

– Оперативные решения связаны с осуществлением текущих целей и задач. По времени они рассчитаны на период, не превышающий месяца.

По признаку содержания задачи принятия решений различают: экономические, организационные, технические, технологические, политические и другие решения.

По признаку количества целей выделяют: одноцелевые и многоцелевые решения.

По частоте принятия решений выделяют: одноразовые (случайные) и повторяющиеся (рутинные) решения.

При рассмотрении процессов принятия решений следует учитывать следующие два момента:

– принимать решения, как правило, сравнительно легко, но принять хорошее решение трудно;

– принятие решения - это психологический процесс, поэтому неудивительно, что способы, используемые руководителем для принятия решений, варьируются от спонтанных до высоко логических.

1. Технологии принятия решения в условиях неопределенности

Неопределенность - это неполнота или недостоверность информации об условиях реализации решения, наличие фактора случайности или противодействия. Принятие решения в условиях неопределенности означает выбор варианта решения, когда одно или несколько действий имеют своим следствием множество частных исходов, но их вероятности совершенно не известны или не имеют смысла .

Источниками неопределенности ожидаемых условий в развитии предприятия могут служить поведение конкурентов, персонала организации, технические и технологические процессы и изменения конъюнктурного характера. Условия могут подразделяться на социально-политические, административно-законодательные, производственные, коммерческие, финансовые. Таким образом, условиями, создающими неопределенность, являются воздействия факторов внешней к внутренней среды организации. Решение принимается в условиях неопределенности, когда невозможно оценить вероятность

потенциальных результатов. В итоге вероятность определенного последствия невозможно предсказать с достаточной степенью достоверности. Сталкиваясь с неопределенностью, руководитель может использовать две основные возможности:

- попытаться получить дополнительную релевантную информацию и еще раз проанализировать проблему. Этим часто удается уменьшить новизну и сложность проблемы. Руководитель сочетает эту дополнительную информацию и анализ с накопленным опытом, способностью к суждению или интуицией, чтобы придать ряду результатов субъективную или предполагаемую вероятность;

- действовать в точном соответствии с прошлым опытом, суждениями или интуицией и сделать предположение о вероятности событий.

При принятии управленческого решения в общем случае необходимо:

- спрогнозировать будущие условия, например, уровни спроса;
- разработать список возможных альтернатив
- оценить окупаемость всех альтернатив;
- определить вероятность каждого условия;
- оценить альтернативы по выбранному критерию решения.

Одно из главных правил управленческой деятельности гласит: не избегать риска, а предвидеть его, стремясь снизить до возможно более низкого уровня. Это требует грамотного управления рисками, т.е. своевременного предвидения, заблаговременного выявления неопределенностей и их последствий на деятельность организации для разработки и реализации УР по их уменьшению.

Для анализа риска необходима быстрая и достоверная информация. В условиях более жесткой конкуренции, вызванной глобализацией рынков, победу будут одерживать не крупные предприятия над малыми, а динамичные над медленно реагирующими на изменение обстановки. Второй этап управления риском - выявление риска.

Модуль 4. Разработка плана реализации управленческого решения

Принятое решение требует чёткого плана его реализации. При организации исполнения плана мероприятий должно быть предусмотрено:

- доведение содержания работ до исполнителей;
- сроки выполнения определённых мероприятий;
- необходимые ресурсы;
- ожидаемые конкретные результаты;
- форма контроля исполнения;
- ответственные за контроль службы и лица;
- формы стимулирования участников мероприятий;
- формы ответственности за ненадлежащее выполнение работ;
- порядок делегирования полномочий.

Существует целый ряд причин, препятствующих применению рациональной модели в процессе принятия решения:

- Часто менеджеры не знают, что проблема существует. Они либо перегружены, либо проблема хорошо скрыта от них.
- Не представляется возможности собирать вокруг проблемы всю имеющуюся информацию по техническим или стоимостным причинам.
- Ограничения во времени заставляют принимать не лучшие решения.
- Во многих случаях рассматриваются не все альтернативы, а при их оценке и выборе трудно учесть качественные факторы.
- Выполнения решения не многими менеджерами связывается с самим решением, что позволяет проблеме продолжать развиваться.

Модуль 5. Поведение типовых функций принадлежности.

Существует множество других функций принадлежности нечетких множеств, заданных как композиции вышеупомянутых базовых функций (двойная гауссова, двойная сигмоидальная и т.п.), либо как комбинации по участкам возрастания и убывания (сигмоидально-гауссова, сплайн-треугольная и т.п.).

Функция принадлежности $\mu_A x$ – это некоторая не вероятностная субъективная мера нечеткости, которая определяется в результате опроса экспертов о степени соответствия элемента x понятию, формализуемому нечетким множеством A . В отличие от вероятностной меры, нечеткая мера является численной оценкой лингвистической неопределенности, связанной с неоднозначностью и расплывчатостью категорий человеческого мышления.

При построении функции принадлежности $\mu_A x$ с каждым нечетким множеством A ассоциируется некоторое свойство, признак или атрибут R , который характеризует некоторую совокупность объектов X . Чем большей, конкретный объект $x \in X$ обладает этим свойством R , тем ближе к соответствующему, значению $\mu_A x$. Если элемент $x \in X$, определенно обладает этим свойством R , то $\mu_A x = 1$, если же $x \in X$ определенно не обладает этим свойством R , то $\mu_A x = 0$. Существуют прямые и косвенные методы построения функций принадлежности.

Прямые методы целесообразно использовать для измеримых свойств, признаков и атрибутов (скорость, время, температура, давление и т.п.). При использовании прямых методов зачастую не требуется абсолютно точного поточечного задания $\mu_A x$. Как правило, бывает достаточно зафиксировать вид функции принадлежности и характерные точки, по которым дискретное представление функции принадлежности аппроксимируется непрерывным аналогом – наиболее подходящей типовой функцией принадлежности.

Косвенные методы (метод парных сравнений) используются в тех случаях, когда отсутствуют измеримые свойства объектов в рассматриваемой предметной области. В силу специфики рассматриваемых задач при построении

нечетких систем автоматического управления, как правило, применяются прямые методы. В свою очередь, в зависимости от числа привлеченных к опросу экспертов как прямые, так и косвенные методы делятся на одиночные и групповые. Наиболее грубую оценку характеристических точек функции принадлежности можно получить путем опроса одного эксперта, который просто задает для каждого значения $x \in X$ соответствующее значение $\mu_A(x)$.

Метод относительных частот. Пусть имеется m экспертов, n_1 из которых на вопрос о принадлежности элемента $x \in X$ нечеткому множеству A отвечают положительно. Другая часть экспертов $n_2 = m - n_1$ отвечает на этот вопрос отрицательно. Тогда принимается $\mu_A(x) = \frac{n_1}{n_1 + n_2} = \frac{n_1}{m}$.

Для непрерывного представления нечеткой переменной используем какую-нибудь из П-образных функций принадлежности, например, Гауссову. Из множества гауссовых функций $\text{gaussmf}(x, \sigma, c) = \exp\left(-\frac{x - c}{\sigma}\right)^2$ через характерные точки функции принадлежности: точку перехода $\mu_A(3) = 0,5$ и максимум $\mu_A(5) = 1$; проходит функция с параметрами $\sigma = 1,7$, $c = 5$. В качестве альтернативного метода перехода от дискретного ряда точек к непрерывному заданию функции принадлежности можно предложить поиск параметров Гауссовой функции принадлежности, максимально близко аппроксимирующей дискретный ряд по критерию СКО (рис.4.1).

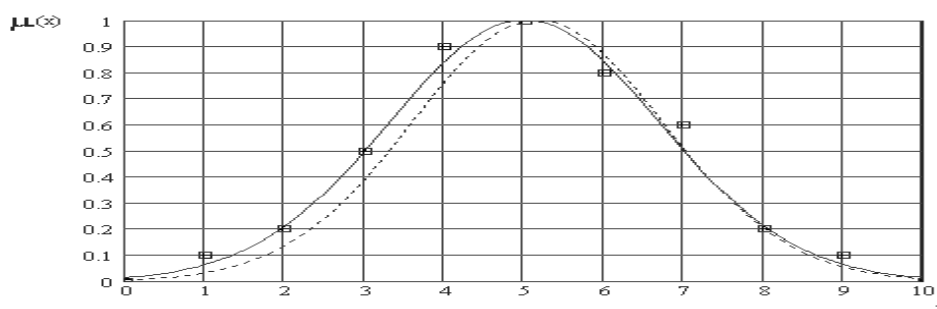


Рис. 4.1 Дискретный ряд по критерию СКО

В структуре управленческой деятельности руководителя - бизнесмена можно выделить несколько важнейших элементов - функции прогнозирования, планирования, организации, мотивирования, контроля. Все они по-своему важны для

успешной деятельности предпринимателя, но многие специалисты в области менеджмента полагают, что наиболее существенным элементом среди этих аспектов работы руководителя является функция принятия управленческих решений. Это связано с тем, что большинство других функций руководитель может делегировать своим подчиненным, но если он делегирует функцию принятия решений, то сразу теряет бразды правления и возможность управлять ситуацией, автоматически переставая быть руководителем. Если говорить об определениях, то решением называют выбор одной из ряда альтернатив в процессе достижения поставленных целей.

В своей реальной работе предпринимателю каждый день приходится иметь дело с информацией и принимать множество решений, которые касаются самых различных сфер его деятельности. Иногда ошибки в этом процессе могут быть легко исправлены, но иногда они могут оказаться фатальными для его бизнеса. Однако, хотя все понимают, что эффективная работа руководителя немислима без умения принимать качественные решения, этому умению предприниматели, как правило, не обучены, и вынуждены обучаться этому умению самостоятельно через процедуру проб и ошибок. Таким образом, исследования в области теории и практики принятия решений имею.

Нечеткие множества – это частный случай нечисловых данных.

Пусть A - некоторое множество. Подмножество B множества A характеризуется своей характеристической функцией:

$$\mu_B(x) = \begin{cases} 1, & x \in B, \\ 0, & x \notin B. \end{cases} \quad (1)$$

Что такое нечеткое множество? Обычно говорят, что нечеткое подмножество C множества A характеризуется своей функцией принадлежности $\mu_C: A \rightarrow [0,1]$. Значение функции принадлежности в точке x показывает степень принадлежности этой точки нечеткому множеству. Нечеткое множество описывает неопределенность, соответствующую точке x – она одновременно и входит, и не входит в нечеткое множество C . За входение - $\mu_C(x)$ шансов, за второе, т.е. за то, что точка не входит в множество, $(1 - \mu_C(x))$ шансов.

Если функция принадлежности $\mu_C(x)$ имеет вид (1) при некотором B , то C есть обычное (четкое) подмножество A . Таким образом, теория нечетких множеств является более общей или хотя бы не менее общей математической дисциплиной, чем обычная теория множеств, поскольку обычные множества – частный случай нечетких. Соответственно можно ожидать, что теория нечеткости как целое обобщает классическую математику.

По степени общности обычная математика и нечеткая математика эквивалентны. Однако для практического применения, например, в теории принятия решений описание и анализ неопределенностей с помощью теории нечетких множеств весьма плодотворны.

Теория нечеткости является обобщением интервальной математики. Действительно, функция принадлежности:

$$\mu_B(x) = \begin{cases} 1, & x \in [a, b], \\ 0, & x \notin [a, b] \end{cases}$$

задает интервальную неопределенность – про рассматриваемую величину известно лишь, что она лежит в заданном интервале $[a, b]$. Тем самым описание неопределенностей с помощью нечетких множеств является более общим, чем с помощью интервалов.

Л.А. Заде рассматривал теорию нечетких множеств как аппарат анализа и моделирования гуманистических систем, т.е. систем, в которых участвует человек. Его подход опирается на предпосылку о том, что элементами мышления человека являются не числа, а элементы некоторых нечетких множеств или классов объектов, для которых переход от «принадлежности» к «непринадлежности» не скачкообразен, а непрерывен. В настоящее время методы теории нечеткости используются почти во всех прикладных областях, в том числе при управлении предприятиями, качеством продукции и технологическими процессами, при описании предпочтений и оптимизации различных процессов, в том числе диагностического характера.

Л.А. Заде использовал термин «fuzzy set» (нечеткое множество). На русский язык термин «fuzzy» переводится как нечеткий, размытый, расплывчатый, и даже как пушистый и туманный.

В качестве примера аппарата теории нечетких множеств рассмотрим определения теоретико-множественных операций над нечеткими множествами. Пусть C и D - два нечетких подмножества A с функциями принадлежности $\mu_C(x)$ и $\mu_D(x)$ соответственно. Пересечением $C \cap D$, произведением CD , объединением $C \cup D$, отрицанием \bar{C} , суммой $C+D$ называются нечеткие подмножества A с функциями принадлежности, соответственно:

$$\mu_{C \cap D}(x) = \min(\mu_C(x), \mu_D(x)), \quad \mu_{CD}(x) = \mu_C(x)\mu_D(x), \quad \mu_{\bar{C}}(x) = 1 - \mu_C(x),$$

$$\mu_{C \cup D}(x) = \max(\mu_C(x), \mu_D(x)), \quad \mu_{C+D}(x) = \mu_C(x) + \mu_D(x) - \mu_C(x)\mu_D(x), \quad x \in A.$$

Теория нечетких множеств в определенном смысле сводится к теории случайных множеств. Однако при решении прикладных задач вероятностно-статистические методы и методы теории нечеткости

Законы де Моргана для нечетких множеств. Законами же Моргана называются следующие тождества алгебры множеств:

$$\overline{A \cup B} = \bar{A} \cap \bar{B}, \quad \overline{A \cap B} = \bar{A} \cup \bar{B}. \quad (2)$$

$$\overline{A \cup \bar{B}} = \bar{A} \cap B, \quad \overline{A \cap \bar{B}} = \bar{A} \cup B, \quad (3)$$

$$\overline{A+B} = \bar{A}\bar{B}, \quad \overline{A\bar{B}} = \bar{A}+B. \quad (4)$$

Тождества (3) и (4) называются *законами де Моргана для нечетких множеств*.

Дистрибутивный закон для нечетких множеств. Некоторые свойства операций над множествами не выполнены для нечетких множеств. Так, $A+A \neq A$, за исключением случая, когда A – «четкое» множество (т.е. функция принадлежности принимает только значения 0 и 1).

Носителем нечеткого множества A называется совокупность всех точек $y \in Y$, для которых $\mu_A(y) > 0$.

Нечетким логическим выводом (fuzzy logic inference) называется

получение заключения в виде нечеткого множества, соответствующего текущим значениям входов, с использованием нечеткой базы знаний и нечетких операций.

В общем случае нечеткий вывод решения происходит за три (или четыре) шага:

1) *этап фаззификации.* С помощью функций принадлежности всех термов входных лингвистических переменных и на основании задаваемых четких значений из универсумов входных лингвистических переменных определяются степени уверенности в том, что выходная лингвистическая переменная принимает конкретное значение.

2) *этап непосредственного нечеткого вывода.* На основании набора правил - нечеткой базы знаний - вычисляются значения истинности для предпосылок всех правил на основании конкретных нечетких операций, соответствующих конъюнкции или дизъюнкции термов в левой части правил. В большинстве случаев это либо максимум, либо минимум из степеней уверенности термов, вычисленных на этапе фаззификации, который применяется к заключению каждого правила. Используя один из способов построения нечеткой импликации, получается нечеткая переменная, соответствующая вычисленному значению степени уверенности в левой части правила и нечеткому множеству в правой части правила.

Основным способом построения нечеткой импликации является способ Kleene-Dienes: пусть A и B - нечеткие высказывания и μ_A, μ_B — соответствующие им функции принадлежности. Импликация $A \Rightarrow B$ определяется формулами:

$$A \Rightarrow B \equiv \bar{A} \vee B,$$

$$\mu_{A \Rightarrow B}(x, y) = \max(1 - \mu_A(x); \mu_B(y)).$$

Однако, это не единственное обобщение оператора импликации. В таблице приведены другие интерпретации этого понятия, наиболее часто применяемые на практике.

Larsen	$\mu_{A \Rightarrow B}(x, y) = \mu_A(x) \mu_B(y)$
Lukasiewicz	$\mu_{A \Rightarrow B}(x, y) = \min(1; 1 - \mu_A(x) + \mu_B(y))$
Mamdani	$\mu_{A \Rightarrow B}(x, y) = \max(\mu_A(x); \mu_B(y))$
Kleene-Dienes-Lu	$\mu_{A \Rightarrow B}(x, y) = 1 - \mu_A(x) + \mu_A(x) \mu_B(y)$

3) *этап композиции (агрегации, аккумуляции)*. Все нечеткие множества, назначенные для каждого термина каждой выходной лингвистической лингвистической переменной, объединяются вместе, и формируется единственное нечеткое множество - значение для каждой выводимой лингвистической переменной. Обычно используются функции MAX или SUM.

4) *этап дефазификации (необязательный)*. Используется тогда, когда полезно преобразовать нечеткий набор значений выводимых лингвистических переменных к точным. В теории нечетких множеств процедура дефазификации аналогична нахождению характеристик положения (математического ожидания, моды, медианы) случайных величин в теории вероятности. Имеется достаточно большое количество методов перехода к точным значениям.

Простейшим способом выполнения процедуры дефазификации является выбор четкого числа, соответствующего максимуму функции принадлежности. Однако пригодность этого способа распространяется лишь на одноэкстремальные функции принадлежности. Для многоэкстремальных функций принадлежности часто используются следующие методы дефазификации:

1) COG (Center Of Gravity) – «центр тяжести». Физическим аналогом этой формулы является нахождение центра тяжести плоской фигуры, ограниченной осями координат и графиком функции принадлежности нечеткого множества.

2) MOM (Mean Of Maximums) – «центр максимумов». При использовании метода центра максимумов требуется найти среднее арифметическое элементов универсального множества, имеющих максимальные степени принадлежности.

3) First Maximum – «первый максимум» - максимум функции принадлежности с наименьшей абсциссой.

Конкретные способы синтеза интегральных функций принадлежности определяется из характера решаемой задачи, ее целей и предметной области.

В связи с этим, на этапе разведочного анализа результатов мониторинга или в процессе диагностики целесообразно проводить исследование зависимости результирующей функции принадлежности от значений функции принадлежностей аргументов в нее входящих.

Порядок выполнения исследований:

1. Изучить теоретические сведения - теорию нечетких множеств, принципы выбора и построения функций принадлежности, методы вычисления интегральной функции принадлежности.
2. На абстрактном носителе x синтезировать функции принадлежности согласно Таблице 1.
3. Согласно Таблице 2 составить алгоритм вычисления интегральной функции принадлежности в случае применения N решающих правил обладающих определенными функциями принадлежностями.
4. Оценить значения скорости и ускорения сходимости интегральной функции к единице в зависимости от количества решающих правил и выбранной функции принадлежности.
5. Проанализировать полученные результаты с точки зрения: сходимости, применения определенных функций принадлежности и методов вычисления интегральной функции, количества итераций в процессе вычисления.
6. Сделать выводы.

Таблица 1. – Типовые функции принадлежности

Название функции	Вид функции
S	$s(x; a, b, c) = \begin{cases} 0, & \text{для } x \leq a \\ 2 \cdot \left(\frac{x-a}{c-a} \right)^2, & \text{для } a \leq x \leq b \\ 1 - 2 \cdot \left(\frac{x-c}{c-a} \right)^2, & \text{для } b \leq x \leq c \\ 1, & \text{для } x \geq c \end{cases}$
π	$\pi(x; b, c) = \begin{cases} s(x; c-b, c-b/2, c), & \text{для } x \leq c \\ 1 - s(x; c, c+b/2, c+b), & \text{для } x \geq c \end{cases}$

γ	$\gamma(x; a, b) = \begin{cases} 0, & \text{для } x \leq a \\ \frac{x-a}{b-a}, & \text{для } a \leq x \leq b \\ 1, & \text{для } x \geq b \end{cases}$
t	$t(x; a, b, c) = \begin{cases} 0, & \text{для } x \leq a \\ \frac{x-a}{b-a}, & \text{для } a \leq x \leq b \\ \frac{c-x}{c-b}, & \text{для } b \leq x \leq c \\ 0, & \text{для } x \geq c \end{cases}$
L	$L(x; a, b) = \begin{cases} 1, & \text{для } x \leq a \\ \frac{b-x}{b-a}, & \text{для } a \leq x \leq b \\ 0, & \text{для } x \geq b \end{cases}$

Таблица 2 Интегральные функции принадлежности (пример)

№	$T(a,b)$ – «пересечение»	$S(a,b)$ – «объединение»	параметры
1	$\text{Min}(a,b)$	$\text{Max}(a,b)$	
2	$a \cdot b$	$a+b-a \cdot b$	
3	$\text{MAX}(a+b-1,0)$	$\text{Min}(a+b,1)$	
4	$\begin{cases} a, & b = 1 \\ b, & a = 1 \\ 0, & \text{иначе} \end{cases}$	$\begin{cases} a, & b = 0 \\ b, & a = 0 \\ 0, & a, b \neq 0 \end{cases}$	
5	$\frac{a \cdot b}{\gamma + (1-\gamma) \cdot (a+b-a \cdot b)}$	$\frac{a+b-(2-\gamma) \cdot a \cdot b}{1-(1-\gamma) \cdot (a \cdot b)}$	$\gamma > 0$
6	$\frac{a \cdot b}{\max(a, b, \alpha)}$	$\frac{a+b-a \cdot b - \min(a, b, 1-\alpha)}{\max(1-a, 1-b, \alpha)}$	$\alpha \in [0,1]$
7	$1 - \sqrt[p]{(1-a)^p + (1-b)^p - (1-a)^p \cdot (1-b)^p}$	$\sqrt[p]{a^p + b^p - a^p \cdot b^p}$	$p > 0$
8	$\left[1 + \lambda \sqrt[\lambda]{\left(\frac{1}{a} - 1\right)^\lambda + \left(\frac{1}{b} - 1\right)^\lambda} \right]^{-1}$	$\left[1 + \lambda \sqrt[\lambda]{\left(\frac{1}{a} - 1\right)^{-\lambda} + \left(\frac{1}{b} - 1\right)^{-\lambda}} \right]^{-1}$	$\lambda > 0$
9	$\max\left(1 - \sqrt[p]{(1-a)^p + (1-b)^p}, 0\right)$	$\min\left(\sqrt[p]{a^p + b^p}, 1\right)$	$p \geq 1$
10	$\log_w \left(1 + \frac{(w^a - 1) \cdot (w^b - 1)}{w - 1} \right)$	$1 - \log_w \left(1 + \frac{(w^{1-a} - 1) \cdot (w^{1-b} - 1)}{w - 1} \right)$	$w = \min(1, 1-a; 1, 1-b)$
12	$\max\left(\frac{a+b-1+\lambda \cdot a \cdot b}{1+\lambda}, 0\right)$	$\min(a+b+\lambda \cdot a \cdot b, 1)$	$\lambda \geq -1$

Контрольные вопросы:

1. Чем отличается нечеткое множество от четкого?
2. Что такое функция принадлежности?
3. Как получить интегральное значение функции принадлежности в случае анализа нескольких нечетких множеств?
4. Что такое носитель функции принадлежности?
5. Когда рекомендуется применять аппарат теории нечетких множеств при построении диагностических правил?
6. В чем заключаются алгоритмы нечеткого вывода Мамдани, Лоренса, Цукамото?
7. Что такое лингвистическая переменная?
8. Каким образом осуществляется коррекция интегральной функции принадлежности в случае неадекватно быстрого приближения ее значения к 1?

Модуль 6. Построение решающих правил

Под диагностическими правилами понимается процедура вывода заключения о соотношении состояния анализируемого объекта или процесса к определенному классу или области на основании временно-пространственной регистрации существенных характеристик.

Любой объект (процесс) с точки зрения диагностики подвергается анализу со стороны исследователя, который, как правило, априори знает, какие существенные характеристики ему следует регистрировать для решения диагностической задачи. То есть, в этом случае, исследователь уже владеет набором диагностических правил, которые либо опровергают, либо подтверждают выдвинутую им рабочую гипотезу о состоянии объекта. Так как о каждом состоянии объекта может выдвигаться различное количество гипотез, то, следовательно, диагностические правила каждой из них не должны в случае объединения поглощать друг друга, и, вообще говоря, должны иметь минимальное количество пересечений как по регистрируемым параметрам, так и по диапазонам их изменений.

В общем случае диагностическое правило имеет вид, например, продукции: если значение $P=P_0$, то состояние $S=S_0$.

$$P=F(S, t, dS), \quad (1)$$

где S - состояние; t - время; dS - диагноз изменения характеристик состояния.

Если зависимость (1) достаточно хорошо идентифицирована (с заданной степенью точности или неопределенности), то нетрудно построить эксперто-диагностическую систему продукционного типа с указанием исследователю технологии реализации необходимой информации для достаточно достоверной диагностики гипотетического состояния.

Рассмотрим **логический** механизм синтеза правила (1).

1 этап. Организация мониторинга состояния заданной глубины и полноты.

2 этап. Выделение множества ортогональных и информативных признаков с точки зрения вариативности. То есть, с одной стороны, селектируем сильно коррелированные

характеристики, с другой стороны, отбираем те из них, вариативность которых (отношение дисперсии к среднему значению) выше определенного порогового уровня (например 10%).

3 этап. Кодлируем состояния (лучше в двоичном коде): с учителем - то есть исследователь знает состояния, без учителя - выполняется кластер-анализ и задаются состояния или вводится пороговый принцип. Таким образом, получаем значения «логической» функции $Y=(Y_{i1}, Y_{i2}, Y_{il})$. Если состояний не много, то рекомендуется применять унитарное кодирование с минимизацией Хемингового расстояния соседних состояний.

4 этап. Кодлируем значение признакового пространства, следующим образом (во всех случаях рекомендуется унитарный код). По каждому оставленному признаку выделяем определенный набор состояний, как попадание значения признака в определенный диапазон. Диапазон определяется либо:

1) Экспертом, исходя из его знаний и жизненного опыта.

2) Исследователем, по анализу частоты распределений значений и личного опыта. При достаточно небольшом количестве признаков анализ гистограммы рекомендуется проводить визуально, наблюдая все признаки одновременно (в концепции системный подход).

3) Автоматически (с применением ЭВМ) по следующему алгоритму.

Исследователь задает количество состояний по каждому признаку n_i (каждое из них кодруется, желательно в унитарном коде). Определяется медиана M_0 и дисперсия G_0 . Определяется удельное отклонение как $G_y = G_0/(n_i-1)$. В качестве первого диапазона (состояния) выбирается величина внутри диапазона $M_0 \pm G_0$. Все значения X_i попавшие в данный диапазон кодруются определенным состоянием S_0 . Величина n_i декрементируется и повторяется описанный процесс над «оставшимися» данными. Так продолжается до тех пор, пока n_i не станет равно 0 и всем оставшимся значениям будет присвоено состояние S_n . Граничные значения $M_0 \pm G_{y0}$ либо включаются в одно из состояний, либо, что более оптимально, кодруются знаком переходной функции.

5 этап. Определяем функциональные зависимости между полученными булевыми функциями (парные и множественные) и парное Хемингово расстояние. Те признаки, у которых это расстояние равно нулю, селектируются путем оставления одного из них с наибольшей вариативностью.

Явный вид логической зависимости между булевыми переменными X_k , $k=1, m$ определяются следующим образом. На первом шаге проверяются условия независимости: поскольку каждая булева функция может иметь два значения истинности, то m булевых функций может образовывать 2^m комбинаций значений истинности. Согласно определению m -булевых функций независимы, если в совокупности при всех возможных значениях аргументов они могут принимать 2^m комбинаций значений истинности. Т.е. для проверки независимости необходимо вычислить их изображающие числа и проверить, образуют ли они полный набор чисел. Если да, то функции независимы, в противном случае - зависимы.

На втором шаге в базисе булевых функций выписывают в последовательные строки изображающие числа и определяют какие числа отсутствуют в наборе столбцов (повторяющиеся значения чисел считают один раз). Столбцы набора представляют собой комбинации значений истинности функций X_1, \dots, X_m , при которых соответствующие элементарные произведения составленные из X_1, \dots, X_m истинны.

Таким образом, если идентифицируется зависимость:

$$F(X_1, \dots, X_n) = 1 \quad (3),$$

то, следовательно, имеющиеся в наборе столбцы указывают номера тех колонок базиса в (X_1, \dots, X_n) , которые совпадают с номерами изображающего числа $\#F(X_1, \dots, X_m)$, на которых функция F истинна.

Например, пусть задан протокол мониторинга трех логических функций:

X_1	11001010
X_2	10101100
X_3	11001100

Выпишем последовательно все столбцы в этом наборе изображающих чисел как строки и укажем справа их десятичные значения:

111=7, 101=5, 010=2, 000=0, 111=7, 110=6, 001=1, 000=0

Видно, что десятичные эквиваленты 3 и 4 отсутствуют, а это означает, что по отношению и в (X_1, X_2, X_3) изображающее число связи $F(X_1, X_2, X_3) = 1$ имеет вид $\#F(X_1, X_2, X_3) = 1$.

Минимизируя полученную функцию, получаем:

$$\#F = X_1 X_3 + \bar{X}_2 \bar{X}_3 + X_1 X_2 = 1$$

Проверяем:

X1	X2	\bar{X}_3	$\bar{X}_1 X_3$	$X_2 X_3$	$\bar{X}_1 X_2$	F
1	1	1	0	1	0	1
1	0	1	0	0	1	1
0	1	0	1	0	0	1
0	0	0	1	0	0	1
1	1	1	0	1	0	1
0	1	1	0	1	0	1
1	0	0	0	0	1	1
0	0	0	1	0	0	1

Таким образом, определяется как логические функции связаны между собой.

6 этап. Идентифицируем логические функции $Y=F(X)$ - парная зависимость и/или $Y=F(\{X\})$ (4) - множественная зависимость. Заметим, что возможен вариант отсутствия тех или иных функциональных зависимостей.

7 этап. Переходим от полученных булевских функций либо к продукционным диагностическим правилам, либо к схмотехническому решению идентификационного диагностического устройства. Однако, второй вариант менее устойчив и мобилен в случае достаточно быстрого изменения окружающей среды, приводящего к изменению в функционировании анализируемого объекта (системы, процесса), а, следовательно, и вида идентифицированных функций.

Как и во множественном регрессионном анализе, при синтезе зависимостей (4) для получения более строгого результата (минимизации пересечений понятий в диагностических, классификационных правилах каждого состояния) рекомендуется руководствоваться правилом максимальной организации (независимости) факторного пространства. Для этого необходимо добиться максимальной независимости X между собой, т.е. в идеале не должно

существовать функциональных зависимостей между X_i . Т.е., если на пятом этапе идентифицируется $F(x)=1$ (3), то необходимо изменить множество X : либо путем исключения переменных (по критерию вариативности), что чревато в общем случае, потерей информации; либо изменить кодирование вводимых сигналов путем уменьшения количества состояний и/или изменения (экспертным путем) диагностических классов состояний. При достаточно мощной вычислительной технике и сравнительно небольшом размере факторного пространства (до 100 признаков) эти проблемы могут быть решены переборным путем. В противном случае, следует применять методы целенаправленного случайного поиска.

Как и в регрессионном анализе возможно формирование продукционных диагностических правил с учетом фактора запаздывания.

Порядок выполнения работы.

1. Самостоятельно изучите теоретический материал.
2. Согласно номеру варианта (равен порядковому номеру студента в журнале группы) N сформируйте протокол мониторинга наблюдения за процессом X , следующим образом:

Индекс переменной X | Индекс протокола Z (см. таблицу 1)

1	mod(n,8)+1
2	mod(n,8)+2
3	mod(n,8)+3
4	mod(n,8)+4

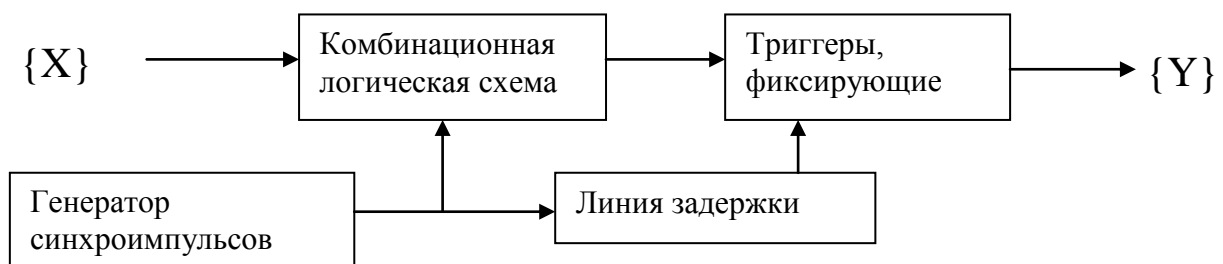
3. Выберите в качестве выходной величины, определяющей состояние процесса переменную мониторинга с индексом 4.
4. Зададитесь числом состоянием по Y - 3, X_1 -5, X_2 - 3, X_3 -2.
5. Определите диапазон изменений состояний, причем для Y_1, X_1 , - автоматически по дисперсии (здесь и далее рекомендуется использовать интегрированные среды типа EXCEL, STATISTICA), X_2 -- экспертным путем анализа гистограммы, X_3 - экспериментальным путем заданием одного порога (<, >=).
6. Закодируйте состояние по X , Y - т.е. получите характеристические числа (вид) булевых функций X_b и Y_b .
7. Проанализируйте взаимозависимость между булевскими факторами X и сформируйте наиболее ортогональное факторное

пространство (векторы которого в наименьшей степени зависимы между собой).

8. Идентифицируйте в минимальном виде (с помощью карт Карно) логические функции $Уб=F(Xб)$.

9. Перейдите от булевского представления к логико-семантическому и сформулируйте диагностические правила продукционного типа.

10. Составьте схемотехническое решение диагностических правил по следующей структуре:



Рассчитайте скважность синхроимпульсов и их характер для обеспечения устойчивой работы схем, считая время срабатывания любого логического элемента 10мс, время срабатывания триггера 50мс.

11. Оформите отчет с указанием последовательности своих действий, необходимых комментариев и выводов.

Контрольные вопросы:

1. Что определяет решающее правило?
2. Какие типы решающих правил применяют в диагностическом процессе при обработке результатов мониторинга?
3. В чем заключается логический способ синтеза решающего правила?
4. Каким образом осуществляется бинарное кодирование признаков пространства при синтезе логических решающих правил?
5. Как формулируется решающее правило продукционного типа?
6. Как осуществляется семантическое описание решающего правила?
7. Каким образом реализуется схемотехническая реализация решающего правила на определенной электронной базе?
8. Как проверяется качество применения решающего правила?

Таблица 1. Результаты регионального мониторинга (начало)

год	Всего родилось	Всего заболело	Врожденные пороки (ВП)	асфаксия	Умерл о всего	Умер ло от ВП	Умерл о от асфик сии
1	1657	90	22	6	17	4	5
2	2081	170	24	9	32	3	9
3	2173	201	20	25	32	7	5
4	2676	198	41	17	34	5	5
5	2557	191	51	47	21	3	10
6	2522	586	83	78	23	1	3
7	2893	252	30	19	31	3	7
8	2956	270	45	8	32	6	5
9	2650	197	38	12	25	3	6
10	3036	213	42	36	36	4	12
11	3165	230	37	32	21	2	5
12	3181	218	61	42	27	8	3
13	2930	216	65	58	18	1	6
14	2491	202	41	55	20	2	5
15	2964	185	37	39	25	2	1
16	2425	290	65	87	22	2	7
17	2432	238	50	53	19	3	4
18	2388	196	34	65	19	0	9
19	2290	197	34	58	22	4	9
20	2995	193	45	27	28	5	7

Год	Ревм. пораж. сердца	Инфаркт миокарда	Гипертан. болезнь	стенокардия	Септический эндокартит	летальность
1	107	8	171	30	5	18
2	147	4	151	38	9	20
3	146	22	124	42	4	17
4	122	27	145	56	10	20
5	104	37	134	83	8	23
6	77	37	110	33	11	23
7	82	24	156	38	8	20
8	104	37	100	40	4	13
9	88	21	87	45	35	18
10	75	17	111	41	6	2
11	44	28	71	29	3	17
12	56	4	120	38	8	12
13	44	16	100	31	7	7
14	44	12	104	28	5	24
15	57	15	97	35	15	10
16	32	17	84	33	14	5
17	36	15	116	40	20	16
18	44	20	176	44	2	28
19	51	23	157	37	11	13
20	48	18	162	43	17	17

Модуль 7: Типовая структура и методы принятия решений в САПР

Система автоматизированного проектирования, САПР, CAD - автоматизированная система, реализующая информационную технологию выполнения функций проектирования, представляет собой организационно-техническую систему, предназначенную для автоматизации процесса проектирования, состоящую из персонала и комплекса технических, программных и других средств автоматизации его деятельности.

САПР - организационно-техническая система, входящая в структуру проектной организации и осуществляющая проектирование при помощи комплекса средств автоматизированного проектирования (КСАП).

Система автоматизированного проектирования (САПР) – это организационно-техническая система, состоящая из комплекса средств АП, взаимодействующего с подразделениями проектной организации, и выполняющая АПР.

Система автоматизированного проектирования (САПР) – комплекс средств автоматизации проектирования, взаимосвязанных с необходимыми подразделениями проектной организации или коллективом специалистов (пользователей системы), выполняющий автоматизированное проектирование.

В понятие САПР входят и такие системы как CAD, CAM, CAE.

CAD-системы (computer-aided design) — компьютерная поддержка проектирования, предназначенная для решения конструкторских задач и оформления конструкторской документации (более привычно они именуются системами автоматизированного проектирования — САПР).

CAM-системы (computer-aided manufacturing) — компьютерная поддержка изготовления, предназначенная для проектирования обработки изделий на станках с числовым программным управлением (ЧПУ) и выдачи программ для этих станков. CAM-системы еще называют системами технологической подготовки производства.

САЕ- системы (computer-aided engineering) — поддержка инженерных расчетов представляющая собой применение обширного класса систем, каждая из которых позволяет решать определенную расчетную задачу (группу задач), начиная от расчетов на прочность, анализа и моделирования тепловых процессов до расчетов гидравлических систем и машин, расчетов процессов литья. В САЕ-системах также используется трехмерная модель изделия. САЕ-системы еще называют системами инженерного анализа [2].

Основная функция САПР - осуществление АПР на всех или отдельных стадиях проектирования объектов и их составных частей на основе применения математических и других моделей, автоматизированных проектных процедур и средств ВТ. Функционирование САПР должно обеспечивать получение проектных решений, выполненных в заданной форме. Проектное решение – описание объекта проектирования или его составной части, достаточное для рассмотрения и принятия заключения об окончании проектирования или путях его продолжения. Результатом проектирования в САПР является удовлетворяющая заданным требованиям совокупность законченных проектных решений, необходимых для создания объекта проектирования.

Большой объем информации, с одной стороны, позволяет получить более точные расчеты и анализ, с другой — превращает поиск решений в сложную задачу. Неудивительно, что первичный анализ данных был переложен на компьютер. В результате появился целый класс программных систем, призванных облегчить работу людей, выполняющих анализ (аналитиков). Такие системы принято называть системами поддержки принятия решений — СППР (DSS, Decision Support System).

СППР — это системы, обладающие средствами ввода, хранения и анализа данных, относящихся к определенной предметной области, с целью поиска решений.

СППР (Decision Support System, DSS) - компьютерная *автоматизированная* система, целью которой является помощь людям, принимающим решение в сложных условиях для полного и объективного анализа предметной деятельности.

Система поддержки принятия решений предназначена для поддержки многокритериальных решений в сложной информационной среде. При этом под многокритериальностью понимается тот факт, что результаты принимаемых решений оцениваются не по одному, а по совокупности многих показателей (критериев) рассматриваемых одновременно. Информационная сложность определяется необходимостью учета большого объема данных, обработка которых без помощи современной вычислительной техники практически невыполнима.

Система поддержки решений СППР решает две основные задачи:

- выбор наилучшего решения из множества возможных (оптимизация),
- упорядочение возможных решений по предпочтительности (ранжирование).

Также все АСППР представляют собой интерактивный человеко-машинный комплекс ЛПР-ЭВМ, который реализует тот или иной метод принятия речи с активным участием ЛПР. Этот комплекс служит для усиления интеллекта ЛПР при решении трудно формализуемых задач проектирования, управления и др. АСППР помогает ЛПР преодолевать трудности, связанные с процессом принятия решений.

1. принятие размерности, т.е. большое число альтернатив
2. многокритериальность решаемых задач
3. ограничение по используемым ресурсам
4. неполноту информации о создаваемой системе
5. неполноту информации о внешней среде
6. жесткие требования на время решения задачи и др.

АСППР реализует новую информационную технологию решения задач, а именно: ЛПР задает схему решения задачи, а система доводит ее до конечного решения. При этом человек контролирует процесс решения задачи и вносит соответствующие коррективы.

Типовая структура САПР

В соответствии с ГОСТ 23501.101-87, в структуре САПР выделяют следующие элементы: КСАП САПР — комплекс

средств автоматизации проектирования САПР, подсистемы САПР, как элемент структуры САПР, возникают при эксплуатации пользователями КСАП подсистем САПР.

КСАП-подсистемы САПР — совокупность ПМК, ПТК и отдельных компонентов обеспечения САПР, не вошедших в программные комплексы, объединённая общей для подсистемы функцией.

ПТК — программно-технические комплексы компоненты обеспечения ПТК САПР; ПМК — программно-методические комплексы компоненты обеспечения ПМК САПР компоненты обеспечения САПР, не вошедшие в ПМК и ПТК. Совокупность КСАП различных подсистем формируют КСАП всей САПР в целом.

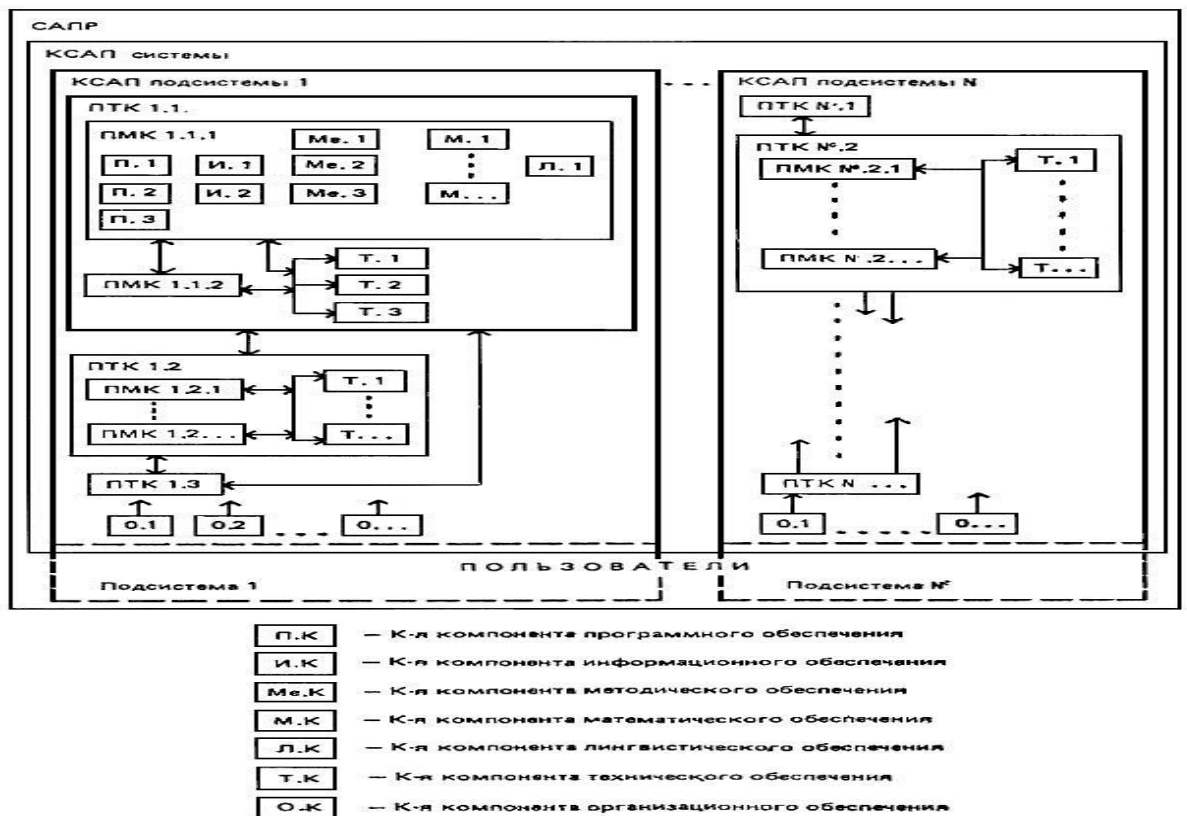


Рисунок 1 Структурная схема САПР

Модуль 8. Типовая структура автоматизированной системы поддержки принятия решений

Выделяют четыре основных структурных компонента автоматизированной СППР:

- информационные хранилища данных;
- средства и методы извлечения, обработки и загрузки данных (ETL);
- многомерная база данных и средства анализа OLAP;
- средства Data Mining.

Хранилища данных

Хранилище данных — это предметно-ориентированная, интегрированная, вариантная по времени, не разрушаемая совокупность данных, предназначенная для поддержки принятия управленческих решений.

Хранилище данных — ориентированная на поддержку управленческих решений автоматизированная система, состоящая из организационной структуры, технических средств, базы или совокупности базы данных (БД) и ПО, которое выполняет, как правило, следующие функции:

- извлечение данных из разрозненных источников, их трансформация и загрузка в хранилище;
- администрирование данных и хранилища;
- извлечение данных из хранилища, аналитическая обработка и представление данных конечным пользователям.

Многомерная база данных и средства анализа OLAP;

Технология комплексного многомерного анализа данных получила название OLAP. OLAP - это ключевой компонент организации хранилищ данных. Концепция OLAP была описана в 1993 году Эдгаром Коддом, известным исследователем баз данных и автором реляционной модели данных.

Термин OLAP - это понятие для различных технологий, включая системы поддержки принятия решений, Business Intelligence и управленческие информационные системы.

Средства и методы извлечения, обработки и загрузки данных (ETL)

Подсистема ETL (Extract, Transform, Load) корпоративного хранилища данных обеспечивает выполнение одного из

основных процессов в управлении хранилищем данных, который включает в себя:

- извлечение данных из внешних систем источников (Extract);
- преобразование (трансформацию) и очистку данных – приведение данных к структурам модели данных и к заданному качеству данных (Transform);
- загрузку данных в область оперативного и постоянного хранения данных хранилища данных (Load).

Data Mining

Data Mining - это процесс поддержки принятия решений, основанный на поиске в данных скрытых закономерностей (шаблонов информации).

Data Mining - это процесс обнаружения в сырых данных ранее неизвестных, нетривиальных, практически полезных и доступных интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности.

Суть и цель технологии Data Mining можно охарактеризовать так: это технология, которая предназначена для поиска в больших объемах данных неочевидных, объективных и полезных на практике закономерностей.

Неочевидных - это значит, что найденные закономерности не обнаруживаются стандартными методами обработки информации или экспертным путем.

Объективных - это значит, что обнаруженные закономерности будут полностью соответствовать действительности, в отличие от экспертного мнения, которое всегда является субъективным.

Практически полезных - это значит, что выводы имеют конкретное значение, которому можно найти практическое применение.

Таким образом, нами были рассмотрены такие понятие как: САПР, автоматизированной системы поддержки принятия решений и их типовая структура.

Техническое обеспечение САПР

Техническое обеспечение САПР включает в себя различные технические средства, которые используются для выполнения

автоматизированного проектирования, а именно ЭВМ, сетевое оборудование, периферийные устройства, а также оборудование некоторых вспомогательных систем поддерживающих проектирование.

Требования, которые предъявляют к техническому обеспечению:

1) обеспечение выполнения всех необходимых проектных процедур, для которых имеется соответствующее ПО;

2) обеспечение взаимодействия между ЭВМ и проектировщиками, а также поддержка интерактивного режима работы;

3) обеспечение взаимодействия между членами коллектива, занятых общим проектом. В результате общая структура ТО САПР представляет сеть узлов, которые связаны между собой средой передачи данных. Узлами (станциями данных) являются рабочие места проектировщиков, часто называемые автоматизированными рабочими местами (АРМ) или рабочими станциями (WS — Workstation), ими могут быть также большие ЭВМ, отдельные измерительные и периферийные устройства.

Среда передачи данных представлена каналами передачи данных, которые состоят из линий связи и коммутационного оборудования. В каждом узле выделяют оконечное оборудование данных (ООД), которое выполняет определенную работу по проектированию, и аппаратуру окончания канала данных (АКД), которое предназначено для связи ООД со средой передачи данных (ООД можно представить персональным компьютером, а в качестве АКД представим вставляемую в компьютер сетевую плату).

Канал передачи данных — это средство двустороннего обмена данными, которое состоит из АКД и линии связи. Линия связи — часть физической среды, используемая для обеспечения распространения сигналов в определенном направлении.

Существуют следующие виды вычислительных сетей: 1) Локальная вычислительная сеть (ЛВС), или LAN (Local Area Network), имеет линию связи, к которой подключаются все узлы сети. При этом топология соединений узлов может быть

шинная, кольцевая, звездная. Применяется в небольших проектных организациях (1-10 компьютера), которые размещены на малых расстояниях; 2) Корпоративная сеть. Состоит из ряда ЛВС, называемых подсетями, и средства связи ЛВС между собой. В эти средства имеются коммутационные серверы (блоки взаимодействия подсетей). Применяется в более крупных по масштабам проектных организациях (10-100 и более компьютеров), относящихся к разным проектным и управленческим подразделениям и размещенных в помещениях одного или нескольких зданий; 3) Территориальная сеть (WAN — Wide Area Network). Та же корпоративная, но здания организации значительно удалены друг от друга (вплоть до разных городов). Здесь выделяют магистральные каналы передачи данных (магистральную сеть), имеющие достаточную протяженность, и каналы передачи данных, связывающие ЛВС с магистральной сетью, называемые абонентской линией. [2] Взаимодействие станции (узла сети) со средой передачи данных для обмена информацией с другими станциями называют доступом к сети. Управление таким доступом к сети осуществляется установлением последовательности, в которой станции получают доступ к среде передачи данных.

Метод доступа бывает:

1) случайный - основан на контроле наличия электрических колебаний (несущей) в линии передачи данных и устранении конфликтов, возникающих в случае попыток одновременного начала передачи двумя или более станциями, путем повторения попыток захвата линии через случайный отрезок времени;

2) детерминированный метод (например, маркерный) - основан на передаче полномочий (право инициировать определенные действия, динамически предоставляемые объекту) передающей станции с помощью специального информационного объекта, называемого маркером.

Программное обеспечение САПР

Программное обеспечение разделяют на следующие классы:

1) Общесистемное ПО. В данную группу обычно относят ОС (обеспечивают управление аппаратной частью компьютера и

прикладными программами, а также их взаимодействием между собой и пользователем), сетевые ПО типовых телекоммуникационных услуг (обеспечивают обработку, передачу и хранение данных в сети), сервисное ПО (необходимо для выполнения вспомогательных операций обработки данных или обслуживания компьютеров). Возможности вычислительных систем определяются больше ОС, чем техническими устройствами. ОС организует одновременное решение различных задач на ЭВМ, динамическое распределение каналов передачи данных и внешних устройств между задачами, планирование потоков задач и последовательность их решения с учетом установленных критериев, динамическое распределение памяти вычислительного комплекса. ОС требует для своей работы определенных ресурсов: процессора, внешней и основной памяти;

2) Системная среда. Основой системной среды является система управления проектными данными, инструментальные средства разработки и сопровождения ПО, интеллектуальные средства поддержки принятия проектных и управленческих решений;

3) Прикладное ПО. Представлено программно-методическими комплексами, пакетами прикладных программ для выполнения проектных и бизнес-процедур и пакетами прикладных программ общего назначения. ППП общего назначения наиболее распространен, что обусловлено обширным применением средств компьютерной техники во всех сферах деятельности, созданием автоматизированных информационных систем различных предметных областей;

4) Специальное ПО. Отличительной особенностью являются высокие требования к технической части системы обработки данных, наличие библиотек встроенных функций, объектов, интерфейсов с графическими системами и базами данных.

Информационное обеспечение САПР

Банки данных - это совместное использование СУБД и Базы Данных. База Данных - это поименованная совокупность данных, организованная по определённым законам, предусматривающим

общие принципы описания, управления и хранения данных независимо от прикладных программ.[3] Принципами построения Банков данных является интеграция данных и централизация управления. База данных состоит из записей, являющиеся наименьшими единицами обмена между ВЗУ и ОЗУ. Записи состоят из полей - минимальные единицы обработки данных. Физическая организация БД – это совокупность методов и средств размещения данных на машинных носителях, определяющая методы доступа к данным. Итогом разработки физической организации являются сами файлы баз данных и файлы поисковых структур. Из множества поисковых структур чаще используются цепные и линейные списки, инвертированные и индексные файлы.

СУБД – это совокупность программ и языковых средств, предназначенные для создания, ведения и использования баз данных.

Среди общих требований к СУБД отмечают: обеспечение целостности данных (их полноты и достоверности), защита данных от несанкционированного доступа и от искажений вследствие возникающих сбоев аппаратуры, удобство пользовательского интерфейса. Банк данных в САПР является важной обслуживающей подсистемой, который выполняет функции информационного обеспечения и имеет ряд особенностей. В нем хранятся как редко изменяемые данные (справочные данные, архивы, типовые проектные решения), так и сведения о текущем состоянии различных версий выполняемых проектов. Обычно, Банк Данных работает в многопользовательском режиме, с его помощью осуществляется информационное взаимодействие различных подсистем САПР.

Математическое обеспечение САПР

Моделирование это экспериментально – наблюдательный метод, эксперименты проводятся не с реальным объектом, а с его моделью (материальный объект, система математических зависимостей, программа или алгоритм, имитирующие функции или структуру исследуемого объекта). Принято выделять

материальное моделирование и идеальное моделирование. К материальному моделированию относят аналоговое и физическое моделирование. Идеальное моделирование носит теоретический характер и основано на мысленной аналогии. Различают два типа идеального моделирования – знаковое и интуитивное. При знаковом моделировании в качестве модели используют графы, схемы, методы и формулы.

Важнейшим видом знакового моделирования является математическое моделирование, при этом модель формируется на языке математики и изучается математическими методами.

Математическое моделирование можно разделить по некоторым признакам:

1) по характеру отображаемых свойств объектов различают структурные модели, предназначены для отображения структурных свойств объекта (топографические, геометрические), и функциональные модели, которые отображают протекание физических и информационных процессов в технологических системах при их функционировании (механические, гидравлические). Обычно представлены системой уравнений, которое описывает фазовые переменные, внутренние, внешние и выходные параметры;

2) по уровню иерархии различают микро (используется для описания дифференциальных уравнений с частными производными), макро (используется для алгебраических и дифференциальных уравнений), функционально-логический (используется аппарат передаточных функций, аппарат математической логики и конечных автоматов) и системный (используются системы массового обслуживания, сети Петри);

3) по способу представления свойств объекта бывают аналитические модели, алгоритмические модели и имитационные модели;

4) по способу получения разделяют на теоретические и эмпирические модели.

Получение элементов математических моделей состоит из следующих операций:

- 1) выбор свойств объекта, подлежащие отражению в модели. Основано на анализе возможных применений модели, также определяется степень универсальности ММ;
- 2) сбор исходной информации о выбранных свойствах объекта;
- 3) синтез структуры ММ;
- 4) расчет числовых значений параметров ММ;
- 5) оценка точности и адекватности ММ;
- 6) реализация функциональных ММ на ЭВМ (получение рабочей программы анализа в виде последовательности элементарных действий (логических и арифметических операций), реализуемых ЭВМ).

Лингвистическое обеспечение САПР

Совокупность языков, используемых в САПР для представления информации о проектируемых объектах, процессе и средствах проектирования, а также для осуществления диалога проектировщик-ЭВМ и обмена данными между техническими средствами САПР называется лингвистическим обеспечением.

Языки проектирования ориентированы на пользователей – проектировщиков и предназначены для эксплуатации САПР. Эта группа языков делится на: входные (служат для задания исходных данных или формирования проблемы), внутренние (служат для представления информации об объектах и процессах проектирования во внутренних кодах ЭВМ), базовые (предназначенный для представления дополнительных сведений к первичному описанию объекта проектирования, проектных решений, описаний проектных процедур), выходные (обеспечивают оформление результатов проектирования в текстовом или графическом виде, в форме, удовлетворяющей требованиям его дальнейшего применения).[2]

Языки программирования необходимы для создания программного обеспечения при разработке САПР. Выделяют следующие уровни языков программирования:

- 1) машинные (низкий уровень) - языки программирования, воспринимаемые аппаратной частью компьютера (машинные коды);

2) машинно - ориентированные (низкий уровень) - языки программирования, которые отражают структуру конкретного типа компьютера;

3) машинно-независимые языки высокого уровня.

Языки низкого уровня, требуют указания мелких деталей процесса обработки данных. Языки высокого уровня имитируют естественные языки, используя некоторые слова разговорного языка и общепринятые математические символы. Языки высокого уровня делятся на: процедурные, логические и объектно-ориентированные.

Для эффективного функционирования лингвистического обеспечения САПР необходимо, чтобы оно обладало свойством "открытости", поэтому в составе инструментальных средств САПР целесообразно иметь синтаксически ориентированный языковой процессор, предназначенный для разработки, как новых языков взаимодействия, так и для модификации уже используемых. Этим средством являются языковые процессоры, предназначены для организации взаимодействия проектировщиков с САПР на основе проблемно-ориентированных языков и выполняющие следующие основные функции: осуществление синтаксического и семантического анализа; выдача диагностических сообщений об ошибках пользователя; формирование последовательности выполнения программных моделей, реализующих заданную проектную процедуру; передача данных, содержащихся в предложениях, проектным процедурам.

Языковой процессор представляет собой транслятор, который переводит исходный текст, написанный на языке взаимодействия, в эквивалентный ему выходной текст. Транслятор в зависимости от типа выходного текста разделяют на компилятора, интерпретаторы и препроцессоры.

Компилятор - транслятор, формирующий выходной текст в виде объектной программы на машинном языке или языке Ассемблера. В этом случае трансляция и выполнение скомпилированной программы разделяются по времени.

Интерпретатор - для некоторого языка воспринимает исходный текст. Интерпретатор отличается от компилятора тем, что он не

порождает объектную программу, которая затем должна выполняться, и непосредственно выполняет ее сам. Интерпретатор по сравнению с компилятором имеет ряд преимуществ: сообщения об ошибках передавать пользователю легче в терминах исходной программы; версия программы на промежуточном языке обычно оказывается компактнее, чем машинный код, выдаваемый компилятором; изменение части исходной программы не требует перекомпиляции всей программы.

Методическое обеспечение САПР

Создание комплекса средств автоматизации проектирования сопровождается разработкой различных видов документов, определяющих состав, содержание и функциональное назначение всех видов обеспечения САПР.

Наряду с этим выделяется особо комплекс работ, связанный с созданием методического обеспечения.

Под методическим обеспечением понимается совокупность документов, устанавливающих состав и правила отбора и эксплуатации средств обеспечения автоматизированного проектирования.

При разработке методического обеспечения подготавливаются следующие документы: пояснительная записка; задание на строительные, электротехнические, санитарно-технические и другие подготовительные работы (при необходимости); план мероприятий подготовки предприятия-пользователя по вводу в действие САПР; план размещения технических средств САПР с разработкой планировок размещения технических средств, линий связи между ними; технико-экономические показатели САПР; комплекс эксплуатационных документов.

Основным документом является пояснительная записка, которая содержит следующие разделы: общие положения; назначение и область применения; описание процесса проектирования; общее описание системы; описание видов обеспечения; описание подсистемы; связь САПР с другими системами проектирования и управления производственными подразделениями.

Для предприятия-пользователя, обеспечивающего функционирование САПР после ее внедрения, разрабатывается специальный комплект эксплуатационных документов, в состав которых входят: описание САПР; инструкции по эксплуатации комплекса средств автоматизации проектирования.

Общее описание САПР содержит назначение, состав и взаимодействие элементов комплекса средств при автоматизации проектирования и последовательность действий при автоматизированном проектировании.

Инструкция по эксплуатации содержит правила эксплуатации службой САПР этого комплекса. Сюда входят следующие виды документов: руководство по использованию; описание применения; руководство системного программиста; руководство программиста; руководство оператора; описание входного языка; контрольные примеры.[3,4]

7 Организационное обеспечение САПР

Совокупность документов, определяющих состав проектной организации и ее подразделений, функциональные связи между ними, порядок подготовки и прохождения проектов, необходимых для автоматизированного проектирования называется организационным обеспечением.

Организационное обеспечение включает два основных вида документов: положение о службе САПР и программы подготовки специалистов-пользователей САПР с перечнем необходимых пособий и методических материалов.

Положение о службе САПР содержит функции и структуру службы САПР, а также определяет взаимоотношения подразделений этой службы.

При внедрении САПР на предприятии необходима организация при отделе главного конструктора (технолога) специализированного подразделения (сектора или отдела) по САПР.

Специализированные подразделения, являющиеся пользователями САПР, должны выполнять следующие виды работ:

- разрабатывать перспективные планы развития и внедрения САПР;
- готовить предложения по составу технических средств необходимых для внедрения САПР;
- получать у разработчиков программное обеспечение для автоматизации проектно-конструкторских работ;
- организовывать централизованное хранение документации по программному обеспечению и самих программ на машинных носителях, поддерживая их в работоспособном состоянии;
- получать у разработчиков системы и в родственных организациях, уже внедривших САПР, массивы справочно-нормативной информации, необходимой для машинного выполнения проектных работ, адаптировать этот вид обеспечения;
- получать ГОСТы и ОСТы и другие руководящие документы по САПР, разрабатывать стандарты предприятия и инструкции, устанавливающие порядок внедрения автоматизированного проектирования;
- организовывать обучение сотрудников работе САПР;
- оценивать эффективность использования САПР.