

Лекция 22

Виды поддержки проектных решений

Знания в СППР делятся на фактические (декларативные), процедурные и управляющие. Последние два типа образуют модели, взаимодействие которых с декларативными знаниями, называемыми далее просто данными, приводит к выработке решений.

Организация знаний в системе осуществляется с помощью некоторого формализма, который включает в себя совокупность структур, служащих для представления знаний и механизма вывода, позволяющего использовать знания для обеспечения принятия решений, объяснения выводов, порожденных системой, решения конкретных задач. Приведем примеры наиболее известных формализмов.

1. Логика кловов Хорна: знания представляются в терминах отношений и логических связей. Механизм вывода организован по принципу поиска в глубину с возвратом.
2. Правила продукций: знания представляются в терминах соотношений, связывающих ситуации и действия. Механизм вывода основан на оценке ситуации по некоторому условию и выполнении соответствующего этой оценке действия.

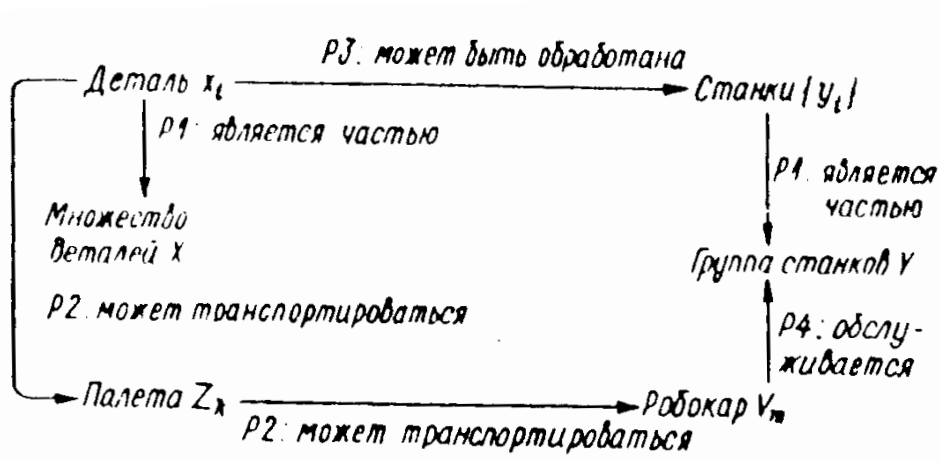


Рис. 4.10. Пример семантической сети

3. Семантические сети: знания представляются посредством сетей и правил перевода в них. Механизм вывода основан на построении путей на сетях, соответствующих решаемой задаче.

Представление данных с помощью указанных формализмов, а также с использованием фреймов широко освещено в литературе [25, 41, 62]. Существуют способы интеграции знаний в их рамках, созданы языки, так или иначе на них ориентированные.

Пример представления данных с помощью семантической сети дан на рис. 4.10. При описании технологического процесса будем изображать участвующие в нем объекты вершинами сети, а существующие между этими

объектами связи — дугами. Имена этих дуг ($P_1 - P_4$) отражают характер (тип) связей.

Использование семантических сетей может оказаться целесообразным при описании базы данных, а также при исследовании конкретной задачи: содержательная трактовка зависимостей, как это видно на представленном примере, позволяет описать проектную ситуацию, а манипулирование с данными, фигурирующими в сети, получить требуемые по ходу процесса проектирования факты.

Аналогичные возможности имеют и другие методы представления знаний. Важно отметить, однако, что представлять необходимо и модели самого процесса организации поддержки решений. В общем случае этот процесс не определяется полностью объектами, составляющими содержание задачи, т. е. не сводится к организации вычислений на моделях, представляющих объект проектирования. К этим моделям процесс должен сходиться в результате некоторой довольно сложной процедуры анализа проблемной ситуации. Сама процедура, тем не менее, может быть построена на системе взаимодействующих семантических сетей, основанных на тех же принципах, с расширенным множеством типов объектов и связей.

Пусть требуется определить, например, возможные технологические маршруты обработки детали x_i с учетом межоперационной транспортировки. Исходной информацией может служить технологическая карта, представляемая СТСО детали, и сеть, описывающая варианты транспортировки (сеть вида приведенной на рис. 4.10). Соотношение сетей (СТСО это тоже семантическая сеть) позволяет сделать заключение о том, что деталь x_i может быть обработана на станках $\{y_i\}$, обслуживаемых робокарами V_m , предназначенными для транспортировки налет Z_k .

Выбор способа представления знаний определяется их структурой и составом, инструментальными средствами, задачами и ситуациями, характерными для проектируемой СППР. В ряде случаев знания могут быть представлены тем же способом, что и модели. Показательный пример представление технологических знаний и моделей в системах поддержки проектирования автоматизированных производств (СПИ АП). Основой соответствующего формализма может служить понятие модифицированной структурно-технологической схемы обработки (СТСОдеталей).

СТСО (ГОСТ 14.416-83) представляет собой фактически семантическую сеть с графом $\Gamma = (U, I)$, где U - множество вершин технологических операций; I - множество дуг, определяющих их очередность. Модификация СТСО осуществляется добавлением к ней (как к графу ассоциированного объекта) множества X требований к выполнению операций, поставленного в соответствие множеству U по правилу

$$u_i \rightarrow x_i; \quad u_i \in U; \quad x_i \in X.$$

Введение такой структуры обусловлено удобством манипулирования с соответствующими понятиями в процессе решения задач анализа и выбора

состава оборудования. Аналогичным образом формируется описание процесса обработки данных в информационно-управляющей системе (ИУС). С точки зрения представления знаний структура — это семантическая сеть с двумя видами вершин, представляющих собой операции и требования к их выполнению.

Преобразование сетей в процессе проектирования ГПС сводится к операциям объединения и вложения графов, факторизации, стягивания и дробления вершин, которые являются в определенном смысле базовыми при разработке соответствующих вычислительных процедур (табл. 4.1). Основные требования при преобразованиях — сохранение очередности

Реализация возможностей СППР, рассмотренных в гл. 4, связывается в настоящее время с данными, моделями и средствами поддержки их взаимодействия с пользователем (парадигма «диалог данные - модель»). Эти компоненты появились, как подмечено в работе [120], в результате эволюции данных и моделей в процессе развития программных средств.

Данные первоначально использовались в виде файлов непосредственно в составе программы. Затем появились файловые системы с возможностью обращения к ним из нескольких программ. Следующий этап эволюции — появление систем управления базами данных, обеспечивших полное разделение программ и данных. Затем были разработаны языки запросов и генераторы отчетов, позволившие обращаться к базам данных специалистам непрограммистам, т. е. конечным пользователям.

Аналогичный процесс эволюции претерпели модели. Сначала это были математические модели вида уравнений-неравенств, затем стали возможными компьютерные вычисления на этих моделях. Новый этап развития моделей — появление собственно компьютерных моделей, когда компьютерная программа сама стала моделью, а не средством вычисления характеристик модели. Пример системы имитационного моделирования, системы символьных вычислений и т. д. Усложнение исследовательских задач привело к появлению пакетов программ моделирования моделирующих систем (системы статистического анализа и др.). Наконец, ориентация на конечного пользователя привела к созданию средств интерактивного моделирования.

Можно констатировать сходимость обоих процессов эволюции к указанным компонентам СППР (рис. 5.1). Рассмотрим некоторые особенности таких компонентов.

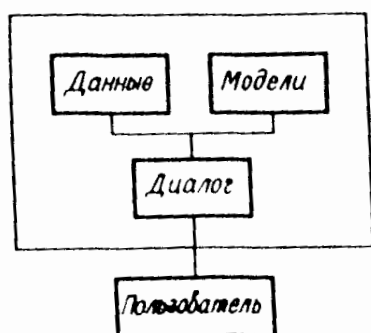


Рис. 5.1. Концепция СППР: данные и модели — основа системы

Базы данных. В основе построения баз данных лежит структурирование данных. Структурирование — основное понятие, которое выделяет базу данных из класса файловых систем, имеющих распространение и использование, в частности в САПР. Обычно различают логическое и физическое описание, т. е. представление данных с точки зрения

вычислительной системы. И в том, и в другом случаях вводят следующие понятия:

элемент данных — наименьшая единица данных, имеющая идентификатор;

сегмент данных — совокупность элементов данных;

логическая запись — совокупность элементов и сегментов данных;

файл — совокупность экземпляров логических записей;

физическая запись — элементарная единица данных, которая может быть записана или считана одной командой ввода-вывода ЭВМ;

набор данных — совокупность физических записей. В этой терминологии *база данных* может быть определена как совокупность различных типов записей и отношений между элементами, записями и сегментами данных.

Из рассмотренного определения базы данных следует, что до ее проектирования и использования информация об объектах, которая должна храниться в базе данных, должна быть предварительно структурирована. Другими словами, предварительно требуется, чтобы каждый представленный в базе данных объект имел определенную совокупность описывающих его атрибутов. Кроме того, требуется, чтобы существовали правила, позволяющие сопоставить эти атрибуты.

Структурирование данных связано с использованием одной из трех моделей данных: реляционной, иерархической и сетевой. Особенности и способы реализации этих моделей хорошо освещены в литературе [22, 60]. В силу концептуальной прозрачности и простоты, а также возможности использования теоретико-множественных операций и реляционной алгебры [40] реляционная модель наиболее популярна, в том числе в системах поддержки решений. Для этого типа модели созданы развитые языки высокого уровня, позволяющие формулировать запросы на данные, представленные средствами реляционной модели, т. е. в виде таблиц. Извлечение данных в СПР осуществляется при этом очень просто, с использованием фактически трех элементарных операторов: ВЫБОР, ОБЪЕДИНЕНИЕ, ПРОЕКЦИЯ.

На языке запросов SQL СУБД System R запрос имеет вид:

SELECT (список атрибутов)

FROM (отношение)

WHERE (условие)

Например, запрос «Перечислить наименование деталей, входящих в состав изделия N» реализуется следующим образом:

SELECT наименование, часть

FROM изделие

WHERE изделие

На известном языке манипулирования данными QBE (Query-By-Example) запрос имеет табличную форму.

Так, содержание запроса предыдущего примера можно проиллюстрировать таблицей

ИЗДЕЛИЕ –N	ДЕТАЛЬ -ЧАСТИ -N	НАЗВАНИЕ-ЧАСТИ
------------	------------------	----------------

...

Таблица заполняется соответствующими содержанию запроса константами и переменными. Ответ на запрос формируется системой в форме такой же таблицы.

Языки запросов, выступая в ряде случаев в виде обычных языков программирования, предоставляют пользователю дополнительные возможности арифметических вычислений, использования команд присваивания, печати и др. Однако пользователь лишь в редких случаях имеет возможность ограничиться применением языков запросов СУБД и СУБМ. Это объясняется прежде всего сложностью решаемой задачи, недостаточной квалификацией пользователя как программиста. Для поиска самого характера запроса ему часто требуется помощь, организация которой не менее сложна, чем осуществление поиска в базах данных.

Модели. Рассмотрим теперь проблемы использования моделей в СППР. Вообще говоря, нужно различать два типа моделей: модели *исследуемых объектов* (процессов) и модели *принятия решений*, связанные с организацией их поиска. Модель второго типа реализуется по двухуровневой схеме. Подмодель первого уровня строится непосредственно на базе модели

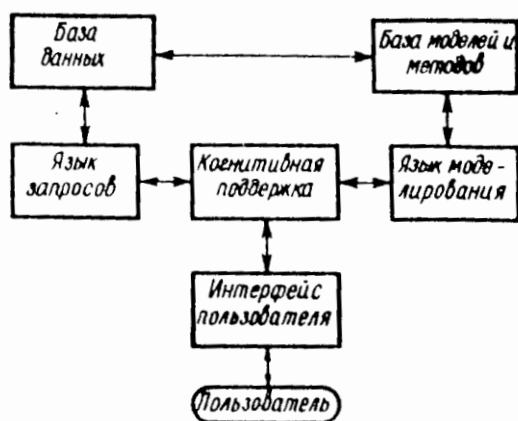


Рис. 5.2. Блок когнитивной поддержки в СППР

объекта, это прежде всего оптимизирующие стратегии экспериментирования с объектом, а также вычисления, связанные с реализацией процессов типа “цель — поиск” или процессов с использованием конструкций “что — если”. Второй уровень связан с реализацией указанной в п. 4.1 технологии поддержки принятия решения, т. е. с реализацией схемы: [Постановка проблемы] → [Анализ проблемы] → [Выбор]. Модели второго уровня реализуются специальными

средствами, например блоком когнитивной поддержки (рис. 5.2). Модели объекта можно условно разделить на два класса.

Первый класс — *формальные модели*, выраженные, как правило, средствами математики. Это модели, описывающие структурные и динамические свойства проектируемых объектов, их экономические характеристики и т. д. Модели данного класса выражают взгляд на объект или явление извне, когда они рассматриваются как черный ящик.

В противоположность этому модели второго класса, *на процессах или правилах*, не опираются на представление явлений уравнениями. Они основаны на понятиях «объект» (сущность), «атрибут», «сценарий». С объектом ассоциируется совокупность атрибутов. В качестве объекта может

выступать, например, совокупность моделей станков в ГПС, атрибутами служат: при описании технологической структуры системы — паспортные данные станков, при описании динамики ~ параметры, характеризующие динамические свойства системы (характеристики деталиеопераций, производительность транспортных средств и т. д.). Сценарий содержит правила преобразования объектов и условия, при которых эти преобразования осуществляются. Реализовать сценарий можно двумя способами: декларативным, например средствами логики предикатов, и процедурным, используя соответствующие языки программирования, например язык Симула-67.

Языки моделирования, подобные Симуле-67, используются для имитации функционирования проектируемой системы (изделия).

Рассмотрим кратко применение имитационного моделирования на примере проектирования ГПС. Основные этапы исследования системы следующие (рис. 5.3). Сначала на основе анализа варианта состава технологического комплекса ГПС определяют элементарные объекты, которые будут являться компонентами имитационной модели. Затем находят необходимые атрибуты: данные о номенклатуре деталей, продолжительности их обработки, технологическом маршруте и т. д. Затем на основании указанной информации формируются сама модель (на том языке, который предусмотрен используемой системой имитационного моделирования) и массивы данных, обеспечивающие возможность проведения процесса моделирования. После этого проверяют модель на ее адекватность моделируемой производственной системе.

Следующие этапы связаны непосредственно с планированием и осуществлением имитационных экспериментов, по их результатам оценивают искомые параметры проектируемой ГПС.

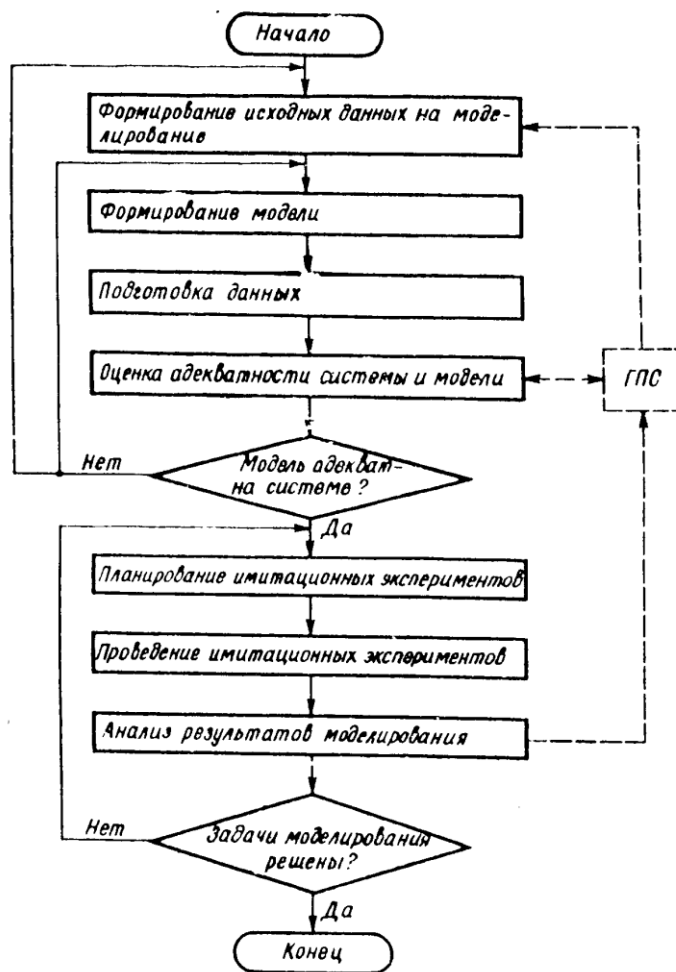


Рис. 5.3. Алгоритм исследования ГПС с помощью имитационного моделирования

Для конкретизации схемы проведения имитационного моделирования рассмотрим процесс представления системы в ЭВМ, характерный для системы имитационного моделирования Симула-67, использующей язык событий.

Компоненты ГПС (станки, накопители, роботы и т. д.) характеризуются набором атрибутов (табл. 5.1).

Атрибуты участвуют в формировании процессов взаимодействия компонентов ГПС. Результатом взаимодействия являются события. Примеры событий: деталь загружается в станок, деталь сходит со станка, робот берет палету с транспортного устройства и т. п.

Другими объектами, с которыми оперирует система моделирования, являются действия (деталь ждет, станок обрабатывает деталь и т. д.) и отношения. Отношения отражают прежде всего порядок выполнения технологических операций в ГПС. Кроме того, сюда же относится назначение деталей к накопителям, распределение приоритетов при обработке деталей и т. д. Действия изменяют атрибуты системы и, следовательно, ее состояние.

Если порядок следования событий заранее непредсказуем, то они реализуются с помощью введения случайных процессов (обычно простейших). Подготовка данных к моделированию включает в себя анализ

исходных требований и формирование указанных массивов. Эта часть работы с трудом поддается автоматизации и практически целиком ложится на плечи проектировщиков.