

第3章 决策支持系统

3.1 决策支持系统结构

在决策支持系统的发展历程中先后提出过多种系统结构,如三部件结构、三系统结构、三库结构和四库结构等。但影响最大的结构形式是前两个,即 1980 年 R. H. Sprague 提出的三部件结构和 1981 年 R. H. Bonczek 等人提出的三系统结构。三部件结构强调模型部件在决策支持系统中的作用;而三系统结构强调知识系统在决策支持系统中的作用,该结构与人工智能的专家系统很接近。以模型部件为主体的三部件结构更能代表决策支持系统。在三部件的基础上增加知识部件就形成了智能决策支持系统,这已成为大家的共识。下面对这两种结构形式进行详细讨论。

3.1.1 决策支持系统的三部件结构

决策支持系统的三部件结构如图 3.1 所示。

决策支持系统(DSS)是三部件即三个子系统的有机结合,这三个部件是对话部件(人机交互系统)、数据部件(数据库管理系统和数据库)、模型部件(模型库管理系统和模型库)。

这种结构是为了达到 DSS 的目标要求而建立的。DSS 吸取了运筹学模型辅助决策的优点,集成多个模型组合形成方案,提高辅助决策的能力。通过对话部件的人机交互,形成人机共同完成辅助决策,这样,它实质上是属于解决半结构化决策问题的系统。可见,DSS 是有广泛前途的发展领域,在国际上已形成了一个学科领域。

下面介绍决策支持系统各组成部分的功能。

1. 对话部件

对话部件是决策支持系统与用户之间的交互界面,用户通过“人机交互系统”控制实际决策支持系统的运行。决策支持系统既需要用户输入必要的信息(用于控制)和数据(用于计算),同时要向用户显示运行的情况以及最后的结果。对话部件包括如下几方面的功能。

1) 提供丰富多彩的显示和对话形式

目前,计算机中几种常见的人机界面技术有 3 种:菜单和窗口,命令语言和自然语言,以及多媒体和可视化技术。

目前,计算机上的软件都提供了丰富的窗口和菜单等界面。窗口和菜单用于引导用户逐级进入系统,用户只需按照菜单提示,按动几个选择键(或点击鼠标)即可操纵和使用系

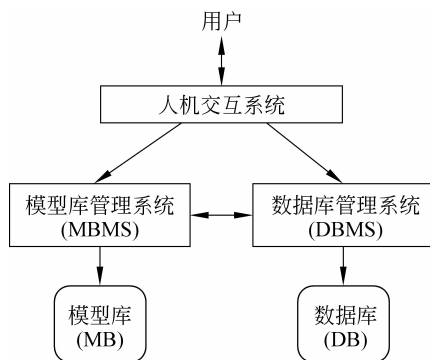


图 3.1 决策支持系统三部件结构

统。若用命令语言来操作,可以脱离菜单的固定模式,适用范围更宽,且能更有效地控制系统的运行。对于不熟悉计算机的人员,使用自然语言更方便,但对计算机技术就要求更高,需采用人工智能技术,如自然语言理解和问题分析等技术。

20 世纪 90 年代发展起来的多媒体技术极大地丰富了人机交互的内容。图形、图像、声音和视频的组合使计算机更接近现实世界。可视化技术是计算机的数据及处理过程,用直观的图形来表示,大大增加了计算机内部数据及数据处理的透明度。

2) 输入输出转换

系统对输入的数据和信息要转换成系统能够理解和执行的内部表示形式。当系统运行结束后,应该把系统的输出结果按一定的格式显示或打印给用户。

3) 控制决策支持系统的有效运行

决策支持系统是三部件的有机结合体。对话部件需要将模型部件和数据部件进行有机集成(包括部件接口)形成系统,并控制 DSS 的有效运行,在对话部件中一般要通过组合模型部件和数据部件的集成语言所编制的 DSS 控制程序来完成。

对话部件容易被人简单地理解为人机交互。实质上,对话部件在 DSS 中主要任务是组合模型部件和数据部件形成 DSS,并控制 DSS 的运行。人机交互只是 DSS 运行中的表现形式。

2. 数据部件

数据部件包括数据库和数据库管理系统。经过几十年的发展,技术趋于成熟,已经有比较成熟的数据库组织方法和数据库管理系统。数据库系统的结构如图 3.2 所示。

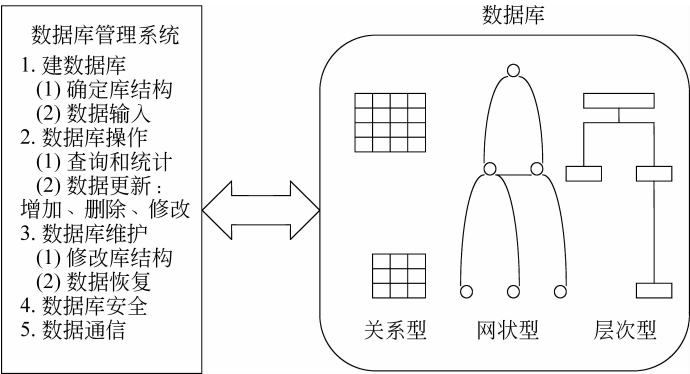


图 3.2 数据库系统结构

1) 数据库存储的组织形式

数据库用来存储大量数据,一般组织成易于进行大量数据操作的形式,典型的数据组织模式有关系模式、层次模式和网络模式等。数据库由数据库管理系统来管理和维护。

2) 数据库管理系统的功能

数据库管理系统具有数据库建立、删除、修改、维护、查询、浏览、排序、索引、统计、安全和通信等功能。

3) 数据库管理语言体系

数据库管理系统提供了一套语言体系,供用户使用数据库并提供与高级语言的接口,这套语言体系一般由两个部分构成:

(1) 数据库定义语言(DDL):对数据库中数据的组成形式,如数据存储模式(数据类型、长度和小数点位置等)、数据依赖关系(如关键字)等进行定义。

(2) 数据库操作语言(DML):对数据库中的数据进行操作,包括数据库的建立和维护、数据字典的建立和维护、数据查询和检索等。

3. 模型部件

模型部件由模型库和模型库管理系统组成。

1) 模型库

模型库用来存放模型,模型不同于数据之处主要有以下两方面:

(1) 模型的表示。它总是以某种计算机程序形式表示的,如语句、程序(源程序和目标程序)甚至对象等。模型一般表示为两部分:一部分对模型进行说明(静态形式),如模型名称、相关的计算机程序名称、模型的输入输出数据的说明等,另一部分为模型程序文件(源程序和目标程序)。

(2) 模型的动态形式。它以程序形式运行(目标程序),完成输入、输出、计算等处理。这种形式是一种可运行的文件。

2) 模型库管理系统

模型库管理系统用于管理模型库,为了适应模型的静态表示与动态运行的特点,模型库管理系统有两方面的功能,一个是类似数据库管理系统的静态管理功能,另一个是模型的动态(运行)管理功能。

(1) 模型库的静态管理

为了有效地管理模型库,一般要建立模型字典,在模型字典中包括模型名、模型程序名、模型功能说明和模型所需数据说明等。模型的静态管理功能如下:

① 模型字典的管理。模型字典的组织形式一般采用数据库形式。在数据库的各个属性项中存放模型名和模型程序文件名等。模型字典的管理可以采用数据库管理。

② 模型文件管理。模型文件一般包括模型算法程序文件、模型功能说明文件以及模型数据说明文件等,它是模型的主体。模型的各文件是通过不同的方式产生的,算法程序文件是由高级语言按照算法过程编制的。功能说明文件是利用编辑程序按文本形式输入的。数据说明文件是模型程序与实际数据之间的接口文件,通过它提取模型所需要的数据。

模型各文件的内容虽不相同,但都以文件形式存放,对文件的管理可以按照计算机操作系统对文件管理的方式进行管理,也可交给操作系统直接管理。

③ 模型字典和模型文件的统一管理。模型字典是模型文件的索引,它们都是模型的组成部分。对模型的管理既包括对模型字典的管理,又包括对模型文件的管理,且两者要统一进行管理。例如,对模型的增加,既要增加该模型的各个文件,也要增加该模型的字典记录。

(2) 模型的动态管理

对模型的动态管理也称为运行管理,它是把模型看做一个活动的实体进行的动态管理,

它的功能如下：

① 控制模型的运行。模型不但可以单独运行,还可以组合运行。运行控制机构必须能够提供顺序、选择和循环这 3 种基本的运行控制机制。

② 模型与数据库部件之间的接口。在模型运行时,规定输入输出数据的来源及去向,并同数据库管理系统进行数据交换。

(3) 模型库管理系统的语言体系

同数据库管理系统一样,模型库管理系统也有一个语言体系,这个语言体系应包括如下两个方面：

① 模型管理语言。定义模型的有关字典属性,如名称、功能、参数、程序构成以及与其他模型的关系等。

② 模型操作语言。执行模型,与数据库之间的动态数据交换,模型之间的运行控制等。

(4) 模型库管理系统的特定功能

模型在计算机中通常表现为程序的形式,用计算机语言编制的模型程序,分为源程序和目标程序,这两种状态与计算机的编辑功能和语言的编译功能相连。有必要把编辑功能和编译功能以特定的形式纳入模型库管理系统中去。

3.1.2 决策支持系统的三系统结构

1981 年 Bonczek 等人提出了决策支持系统(DSS)的三系统结构形式,即它由语言系统(LS)、知识系统(KS)和问题处理系统(PPS) 3 个部分组成。该结构形式如图 3.3 所示。



图 3.3 决策支持系统三系统结构图

1. 语言系统

早期的 DSS 提出：语言系统的功能是把自然语言转化为机器能够理解的形式,以及把机器对问题的解答或者系统内部的其他信息转化为自然语言的相应形式向用户输出。

把自然语言用于 DSS 中,这是人们的理想。技术发展到今天,自然语言处理技术仍未成熟,用自然语言来描述决策问题是将来的事。

目前,计算机语言有三大类：数值计算语言(也称高级程序设计语言),如 C、Java 等;数据库语言,如 MySQL、Oracle 等;智能语言,如 PROLOG、LISP 等。

决策支持系统涉及模型计算和数据库操作,故用于决策支持系统的语言应该是数值计算语言和数据库语言的结合。目前,市场上仍然没有这两类语言合一的语言系统,但有这两类语言的接口语言,如 ODBC、ADO 等。决策支持系统语言实质上是数值计算语言和数据库语言通过接口语言进行集成的语言,用这种集成语言来描述决策问题。

2. 问题处理系统

问题处理系统是对描述的决策问题进行识别、分析和求解的过程。

问题处理系统必须具有明确地识别问题的能力,它能把问题的陈述转化为相应的可执行的操作方案,它能够对问题作比较透彻的分析,确定什么时候问题陈述已经变成了详细的

过程说明,什么时候执行什么,什么时候得到问题的解答。

除了问题识别之外,问题分析能力也是问题处理系统应该具备的重要能力。这是一个在模型、知识、数据和用户之间反复交互的过程。最简单的情况是只在模型和数据之间进行交互,在管理科学/运筹学(MS/OR)领域有大量的计算程序和软件包可以完成这样的工作。最困难的分析过程是在模型、知识、数据和用户四者之间的交互,应该把用户和系统紧密地结合在一起。

由于自然语言处理还未成熟,目前采用的 DSS 语言是数值计算语言、数据库语言以及接口语言的集成语言。问题处理系统是在以集成语言描述的决策问题的基础上,实现决策问题的分析和求解。把问题求解结果反馈给用户。

3. 知识系统

知识系统包含决策问题领域中的大量事实和相关知识。最基本的知识系统由数据文件或数据库组成。数据库的一条记录表示一个事实,按一定的组织方式进行存储。

更广泛的知识是对问题领域的规律性描述。这种描述用定量方式表示为数学模型。数学模型一般用方程、方法等形式描述客观规律性。这种形式的知识可以称为过程性知识。

随着人工智能技术的发展,对问题领域的规律性知识用定性方式描述,一般表现为产生式规则。除了数理逻辑中的公式和微积分公式等精确知识外,一般表现为经验性知识,它们是非精确知识,这样就大大提高了解决问题的能力。

决策支持系统的这种结构形式有以下特点。

1) 强调语言系统

利用计算机对决策问题求解、实现决策支持是需要通过计算机语言来完成的。人类所使用的自然语言在计算机上使用是将来的理想。目前计算机语言种类很多,仍属于“上下文无关文法”,离自然语言相差较远。为了有效地进行问题求解,一般在计算机的输入和输出方面采取简化的自然语言以及有效的人机交互环境来帮助人的理解和使用。

可以认为,语言系统是利用计算机语言来形式化描述决策问题和知识系统中的知识,目前在计算机中,决策支持系统语言是数值计算语言、数据库语言以及它们的接口语言的集成语言,它使决策支持系统能在计算机上实现。

2) 强调问题处理系统的重要性

问题处理系统对语言系统所描述的决策问题进行分析和求解。不同的决策问题,需要进行的问题处理也是不相同的。如何解决实际决策问题在计算机中的求解是问题处理系统的关键所在。在问题求解时要利用知识系统中的知识,按问题的求解途径进行计算,得到问题的解答。

3) 强调知识系统的决策支持作用

知识系统在决策支持中体现了定性辅助决策的效果。知识系统作为决策支持系统的重要组成部分,强调了知识系统的决策支持作用。知识表示形式有多种,数据可以看成是事实性知识,模型是过程性知识,规则是产生式知识。这些知识都能为解决决策问题提供服务。在三系统结构中,把数据、模型和规则统一视为问题处理系统服务的知识。

由于该决策支持系统的结构中包含知识系统,从而体现了智能的特点。

3.1.3 决策支持系统的基本结构

3.1.3.1 决策支持系统的结构比较

决策支持系统有多种结构形式,但主要是以上两种基本结构形式:

(1) 以对话(人机交互)、模型和数据三部件组成 DSS。

(2) 以语言系统(LS)、问题处理系统(PPS)和知识系统(KS)三系统组成 DSS。

从宏观上看它们相差很大。需要进行认真分析,找出它们的共性,提出合理的统一结构形式,以利于 DSS 的开发和发展。

1. 对三部件结构的看法

三部件结构具有如下优点:

(1) 明确了三部件的结构以及它们之间的关系,数据部件和模型部件由于它们的差异,分别存储和管理,便于人们对 DSS 的深刻理解。它们之间存在接口关系和集成关系。接口关系即模型调用数据库中数据时需要通过接口去调用,直接存取是得不到数据的。集成关系表示将模型部件和数据部件以及人机交互集成起来才能形成 DSS。明确了三部件结构和它们的关系后,将便于决策支持系统的设计和关键技术的解决。

(2) 便于和其他系统的区别。它和管理信息系统(MIS)的区别在于 DSS 多了模型部件。它和专家系统(ES)的区别在于,DSS 中是以“模型、数据”部件进行数值计算为主体的系统,而 ES 是以定性知识进行推理为主体的系统。

三部件结构具有如下不足:

没有突出 DSS 的问题处理特性。问题处理系统是解决决策问题的核心,它虽然用到模型和数据,但对不同的 DSS,问题处理是大不相同的。该结构也没有强调语言系统,DSS 语言有它自身特点,既需要数据库处理能力(由数据库语言来完成),也需要模型计算能力(由数值计算语言来完成),即 DSS 语言是两类语言的组合语言。作为三部件结构,可以理解为 DSS 的语言系统和问题处理系统是隐含在人机交互系统中的。

2. 对三系统结构的看法

三系统结构具有如下优点:

(1) 突出了问题处理系统(PPS)的重要性。在设计和开发 DSS 时,应该重点考虑决策问题的处理。

(2) 明确了语言系统(LS)在人机交互中的作用。人机交互是要通过语言系统来完成的。决策问题的形式化也要用语言系统来描述。

(3) 强调了知识的作用。利用知识辅助决策是决策支持系统的一个重要方面,这为决策支持系统向智能方面发展提出了宏观方向。

三系统结构具有如下不足:

(1) 在三系统结构中将数据和模型都统一放入知识系统中,忽略了数据库系统和模型库系统之间的区别和相互关系。模型和数据相差甚远。数据是静态的,它被用来显示或者

参与运算。模型在静态时是一个程序,由数值计算语言编制而成,分为源程序和目标程序。它们占据一定的空间,比单个数据占据的空间大多了;模型在动态时是需要运行的,模型的目标程序要完成对大量数据的加工。模型和数据作为知识的两种不同形式放在一个知识系统中是不合适的,既不利于人们对知识系统的理解,也不利于对知识系统的开发。

(2) 三系统结构接近专家系统。三系统结构中知识系统体现了智能的内容。但如何利用知识,在问题处理系统中只是客观地说明利用知识来解决实际决策问题,这是不够的。知识有不同的类型,如数据、模型和规则等不同知识,它们的表现形式不一样,其存储方式和使用方式都相差很大,在设计和开发它们时,需要对各类知识分别处理,只用一个“知识系统”来概括它们是太粗糙了。这既不利于人们对三系统结构在智能效果上的理解,也不利于该结构的系统开发。如果将知识系统中的知识局限为规则知识,问题处理系统对知识的利用采用推理方式,该系统结构可以认为是专家系统。而专家系统概念比决策支持系统概念提出要早,并且也更成熟,有明确的系统开发方法。可见,用三系统结构来定义决策支持系统的结构就显得不合适了。

3.1.3.2 决策支持系统的结构统一为基本结构

对上述两种基本结构形式的分析可知,用三部件结构来代表 DSS 更合适一些。它能明显地突出 DSS 的特点。在三系统结构中,把数据和模型统一在知识系统中,这不利于对 DSS 的认识和开发。数据与模型,不但本质上不同,而且它们之间存在接口,这在系统开发中是不能忽略的。如果知识系统中再增加规则知识,它又不同于数据和模型,它们决不能组织和存放在一个库中,应该分别建立数据库、模型库和知识库。对不同的库又必须建立相应的库管理系统,分别对不同的库进行管理,各库之间存在着接口问题。为便于 DSS 开发,应该把数据、模型和知识分别建立各自的库和相应的库管理系统。知识库涉及智能,将在第 4 章详细讨论。在不考虑智能方面时,DSS 的基本结构采用三部件结构就很合适。

三部件结构中的最大弱点在于人机交互部件太简化。该部件应该是三系统中将问题处理系统、语言系统和人机交互系统进行综合集成的综合部件。把人机交互部件改为问题综合与人机交互系统即综合部件更合适一些,它将多模型组合运行、大量数据库的存取和人机交互综合为一个整体,形成实际的决策支持系统。

决策支持系统的基本结构形式如图 3.4 所示。

问题综合与人机交互系统(综合部件)可理解为对实际决策问题的处理与人机交互的综合作用。

在决策支持系统出现之前,组合多模型辅助决策早已出现,具体做法是对各模型编制程序在计算机中运行,模型之间的关联由人来完成,即由人来完成模型的组合。对模型间的数值计算和数据处理,只能由人在计算机外进行。因为每个模型本身不考虑它与其他模型之间的联结问题,这项工作只能由人来完成。在出现决策支持系统之后,这种模型间的处理应由问题综合与人机交互系统部件来完成。

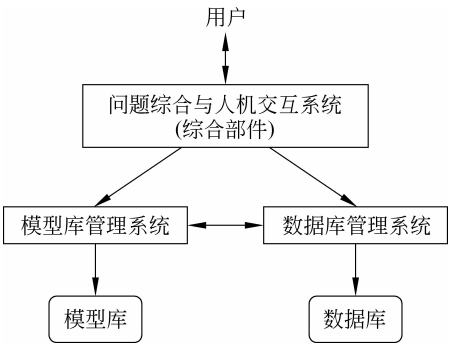


图 3.4 决策支持系统的基本结构

解决了这个问题才能使多模型的组合运行在计算机中自动进行。多模型的组合形成了系统的方案,能解决更复杂的问题,多模型组合的自动运行为改变方案中的模型和数据带来了方便。在系统方案中采用不同的模型或数据的组合将形成不同的方案,故决策支持系统使半结构化问题的解决成为可能。

为使决策支持系统有效自动运行,它对语言系统的功能要求比较高,即它应具有调用模型运行能力、数据库存取能力、数值运算能力、数据处理能力和人机交互能力 5 种综合能力,称它为决策支持系统语言(DSS 语言),它不同于数值计算语言(如 C、Java 等),它还要有很强的数据处理能力。DSS 语言应是两类语言(数值计算语言和数据库语言)的综合。

决策支持系统语言是使原来不能在计算机上实现的问题,即多模型组合辅助决策问题(半结构化问题),能在计算机帮助下完成。

可见,决策支持系统是技术进步的产物。

3.2 数据库系统

3.2.1 数据库系统结构与应用

3.2.1.1 数据库系统组成

数据库系统一般由 4 部分组成,如图 3.5 所示。

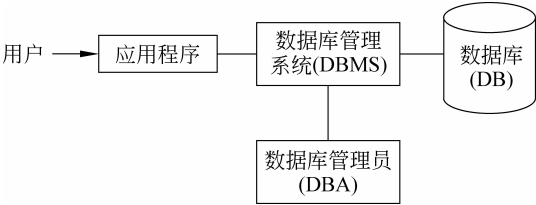


图 3.5 数据库系统组成

1. 数据库

数据库是按一定结构组织在一起的相关数据的集合。一般来说,数据库的容量是很大的,数据按一定的组织结构存放,以便查询利用,数据库中的数据没有冗余,能提供多种应用服务,且数据的存储方式和位置相对地独立于使用它们的程序。

2. 数据库管理系统

数据库管理系统(DBMS)是一组能完成描述、管理和维护数据库等功能的程序系统。它按照一种公用的和可控制的方法完成插入新数据,修改和检索所有数据的操作。

3. 数据库管理员

数据库管理员(DBA)负责设计、建立、维护和管理数据库。

4. 用户和应用程序

它们的操作是以数据库中的数据为基础进行的。每个用户只用到数据库中的一部分数据,不同用户使用的数据以多种方式重叠。

数据库系统的核心是数据库管理系统和数据库,因此也经常把这个核心称为数据库系统。

数据库的重要特点是实现不同用户对数据库的共享。

3.2.1.2 数据库管理系统

数据库管理系统实现了数据的统一管理。数据库管理系统能对数据库中的数据进行查询、增加、删除、修改和维护等统一的管理。

数据库管理系统是在操作系统的基础上对数据库进行统一的管理和控制。数据库管理系统有 4 项主要功能。

1. 描述数据库

描述数据库中的数据的逻辑结构、存储结构、语义信息和保密要求等。

2. 管理数据库

- (1) 控制整个数据库系统的运行。
- (2) 执行数据检索、插入、删除和修改等操作。
- (3) 控制用户的并发性访问。
- (4) 检验数据的安全、保密与完整性。

3. 维护数据库

- (1) 控制数据库初始数据的装入。
- (2) 记录工作日志。
- (3) 修改、更新数据库。
- (4) 重新组织数据库。
- (5) 恢复出现故障的数据库。

4. 数据通信

组织数据的传输。

从系统的角度看,数据库管理系统是由完成上述各功能的许多子系统(程序模块)构成的语言处理系统,其核心是数据库语言和语言处理系统。包括总控子系统、维护子系统和用户接口等。数据库语言通常由两部分组成。

(1) 数据描述语言(DDL)及其解释程序

数据描述语言用于描述数据库数据的结构,此结构称为数据库模式。解释程序将所描述的各项内容从源形式转换成目标形式,即数据库模式的内部表示(存储数据形式)。

(2) 数据操作语言(DML)及其解释程序

数据操作语言供用户存储、检索和修改数据库中的数据之用。该语言由一系列语句组成。解释程序分析数据操作语言的语句,将其分解为数据库基本操作指令并解释执行。

3.2.1.3 数据库应用

1. 数据库查询

数据库查询是数据库中最基本、最常用的操作,也是辅助决策的重要手段。一般的数据库查询功能(如数据库列查询、条件查询和连接查询等)由数据库管理系统提供。更复杂的查询功能需要开发者编制相应的查询程序来完成。

1) 数据库列查询

选择数据库中的全部列或部分列的操作称为投影操作。按列查询包括以下几种:

- (1) 查询指定的列,如查询全体学生的籍贯。
- (2) 查询全部列,如查询全部学生的详细情况。
- (3) 指定条件的查询,如查询“籍贯=湖南”的学生。

2) 条件查询

按指定的查询条件进行查询。

查询条件包括比较大小、指定范围、指定集合、字符匹配、空值和多重条件等。

- (1) 比较大小的查询,即利用关系符=、>、<、≥、≤和≠等建立的条件进行查询。

例如:

- 查询所有年龄在 20 岁以下的学生姓名及其年龄。
- 查询考试成绩不及格的学生姓名。

- (2) 指定范围的查询,即查询属性值在(或不在)指定范围内的元组。例如:

查询考试成绩为良好(80~89 分)的学生姓名。

- (3) 指定集合的查询,即查询属性值属于指定集合的元组。例如:

查询所有计算机系学生的姓名和性别。

- (4) 字符匹配的查询,即查询指定的属性值与<字符串>相匹配的元组。例如:

查询所有姓王的学生的姓名和性别。

- (5) 涉及空值的查询,即查询指定属性值是空值的元组。例如:

查询英语课没有考试成绩的学生姓名。

- (6) 多重条件查询,是用逻辑运算符 AND 和 OR 连接多个查询条件的查询。例如:

查询计算机系年龄在 20 岁以下的学生姓名。

3) 组合查询

在多个属性中,对所需要的属性输入查询条件并进行多条件的任意组合的查询,是一种功能更强的查询方式,也将给用户提供更强的辅助决策能力。

例如,组合查询条件为

- ①起止日期;②物资编号;③发物仓库;④收物单位;⑤调拨分类;⑥地理区。

用户可按需要任意选择条件项,分别输入条件后实现多条件组合查询。

完成这种查询需要根据用户选择的条件生成组合查询语句,并嵌入到查询程序中完成组合查询工作。

例如,查询 2012 年 1—12 月 5 号仓库发给某商店的电冰箱的情况。

2. 数据项表达式查询

在数据库中有一种特殊的查询任务,需要得到某些数据项进行数值计算(表达式计算)后的结果。数据库中数据项之间的各种联系系统称为数据项关系,表示运算关系的式子称为数据项表达式(以下简称项表达式)。

例如,在区域经济发展长期规划和年度计划的制定过程中,需要大量反映国民经济发展的指标,这些指标之间存在着密切的联系。有些指标需要由其他指标(在数据项中)计算得出(生成),并且计算方法多种多样。

1) 数据项表达式公式实例

区域经济发展规划中用于指标生成的表达式如下:

$$\textcircled{1} \text{ 人均社会总产值(元/人)} = \text{社会总产值(万元)} / \text{总人口数(人)} * 10000$$

$$\textcircled{2} \text{ 人均国民收入(元/人)} = \text{国民收入(万元)} / \text{总人口数(人)} * 10000$$

$$\textcircled{3} \text{ 人均国民生产总值(元/人)} = \text{国民生产总值(万元)} / \text{总人口数(人)} * 10000$$

$$\textcircled{4} \text{ 物耗率(\%)} = (\text{社会总产值(万元)} - \text{国民收入(万元)}) / \text{社会总产值(万元)} * 100$$

$$\textcircled{5} \text{ 产值利税率(\%)} = \text{工业利润税金(万元)} / \text{工业总产值(万元)} * 100$$

$$\textcircled{6} \text{ 固定资产投资效果(\%)} = (\text{当年工业总产值(万元)} - \text{去年工业总产值(万元)}) / \text{固定资产投资额(万元)} * 100$$

$$\textcircled{7} \text{ 人均年末储蓄额(元/人)} = \text{城乡居民储蓄额(万元)} / \text{总人口(人)} * 10000$$

$$\textcircled{8} \text{ 收入支出比(\%)} = \text{财政收入(万元)} / \text{财政支出(万元)} * 100$$

$$\textcircled{9} \text{ 社会商品零售额增长率(\%)} = (\text{当年社会商品零售额(万元)} - \text{去年社会商品零售额(万元)}) / \text{去年社会商品零售额(万元)} * 100$$

$$\textcircled{10} \text{ 农村人均纯收入(元/人)} = \text{农村经济纯收入(万元)} / \text{农村总人口(人)} * 10000$$

2) 数据项表达式设计

这些数据项表达式计算的特点如下:

(1) 表达式的形式是任意变化的。

(2) 表达式的计算是临时进行的。

这种对数据项表达式计算的查询不是查询语句所能够完成的,必须专门编制程序来完成这种特殊的查询。编制一个对不同形式的表达式的一个统一的、通用的识别和解释执行程序需要利用编译技术来实现。

数据项表达式包含基本运算、函数、变量和常数。

(1) 基本运算符: +、-、*、/、↑(幂)。

(2) 函数: $\ln(x)$, $\exp(x)$, $\sin(x)$, $\cos(x)$, $\max(x, y)$, $\min(x, y)$, $\arctan(x)$, ...。

(3) 变量: 变量代表某一指标,为了方便数据项表达式计算时对指标的查找,变量用指标的编码来表示。

(4) 常数: 整数、实数。

在计算机语言系统的编译系统中,对所有项表达式的识别和解释执行的具体过程是:先对表达式进行词法分析,得出表达式的组成单词,如“人均社会总产值(元/人)”、“=”等单词。再进行语法分析,将单词构成句子,如整个表达式,生成表达式的目标语言。在计算机中它可以是逆波兰式中间语言或者是机器语言。该编译程序能识别和执行任何表达式并进行计算。

3.2.2 数据库系统在决策支持系统中的作用

3.2.2.1 数据通过模型与方案计算产生辅助决策信息

数据库系统在决策支持系统中为模型与方案提供数据。模型与方案对数据进行加工计算后产生辅助决策信息。

在 2.3.1 节的线性规划模型实例中,通过模型计算,得出生产玻璃门(x)和铝框窗(y)的最优计划分别为:

$$x = 2, \quad y = 6$$

按此信息生产就可以得到最大利润。

在 2.3.2 节对某县粮食产量的预测模型实例中,通过多个模型计算,得出该县 2013 年的粮食产量的预测值是 14~17.5 亿公斤。该县按预测值信息可以制定发展规划。

在 2.4.2 节对橡胶产品的多目标规划模型实例中,通过模型计算,得出橡胶配方的三个原料(x_1 、 x_2 、 x_3)分别为

$$x_1 = 50.7275, \quad x_2 = 25, \quad x_3 = 1.8968$$

在生产中,按此信息配方制成的橡胶产品,其性能就能够较理想地达到用户提出的性能指标要求。

以上实例都说明了数据通过模型计算能产生有效的辅助决策信息。

但要说明的是,计算产生的信息是否准确,取决于该模型与方案是否正确地反映客观现实。在 2.3.1 节中,线性规划的正确性取决于模型参数(两产品单位利润系数、A、B、C 工厂生产新产品时间等)是否正确;在 2.3.2 节中,预测模型的正确性取决于各个模型公式是否准确;在 2.4.2 节中,橡胶产品优化方案计算产生的信息的正确性取决于多元线性回归模型与多目标规划模型的准确性,这两个模型的准确性又取决于模型方程的正确性与模型中的数据正确性。

由于以上模型与方案都与现实情况有一定的误差,模型与方案计算出的信息的准确性也存在一定的误差。在 2.3.1 节中,利用 What-If 分析来弥补产品单位利润的变化范围;在 2.3.2 节中,利用多个模型的计算产生预测值的区间,再由人来最后确定合理的预测值;在 2.4.2 节中,利用两个模型组合成方案,进行试生产的逼近方法,解决橡胶配方问题。

3.2.2.2 数据是模型组合的基础

每个数学模型都需要对大量数据进行加工,这些数据可以看成是模型的输入数据。数学模型的计算结果也是数据,但这些数据是更有价值的信息,它们是数学模型的输出数据。

对于一个较复杂的问题,靠单个模型是不够的,要多个模型组合起来,共同辅助决策。模型之间的组合一般是通过数据来实现的,即一个模型的输出数据是另一个模型的输入数据,或者是一个模型输出数据经过加工处理后成为另一个模型的输入数据。

例如,线性规划模型中约束方程的系数数据的获取,除人工按经验给出外,更多的是靠大量数据经过线性回归模型计算求出。

在 2.4.2 节的实例中,性能与原料之间的线性方程系数是多元线性回归模型的输出,也是线性规划模型约束方程系数的输入。此时,该数据可看成是模型之间组合的基础。

3.3 模型库系统

模型库系统是决策支持系统的核心部件,通过模型或者模型的组合来辅助决策是决策支持系统的中心思想。模型库系统由模型库和模型库管理系统所组成。

3.3.1 模型库

3.3.1.1 模型库的概念

模型是对客观事物的一种抽象描述,人们通过模型来对客观事物进行理解 and 处理。用模型来辅助决策已经是人们的共识。利用计算机对模型的使用经历了 3 个阶段。

1. 模型程序

模型在计算机中的实现主要是编制模型程序。模型程序是利用计算机语言来描述模型的算法过程。一般介绍模型的算法是人工算法。适合于人工进行计算。这种人工算法并不能直接搬到计算机上来实现,这是由于计算机的局限性造成的。必须把人工算法转换成计算机算法才能在计算机上进行计算,求出结果。计算机算法是建立在计算机语言基础上的。计算机语言(满足 2 型和 3 型文法)离人类的自然语言(满足 0 型和 1 型文法)相差甚远。设计和编制计算机程序必须先设计数据的存储结构(如变量、数组、线性表、树、图、文件和数据库等),再用计算机语言设计算法,利用计算机语言的语句以及语句的组合(顺序、选择和循环)形成计算机程序。例如对运输问题采用表上作业法,其 3 个步骤为:求基本解;解位势方程及检验解;调整解。其位势方程的求解,人工算法和计算机算法本质一样,但算法的实现相差很大。

计算机程序又分为源程序和目标程序。源程序是用计算机语言编写的,便于人阅读和修改;目标程序是机器语言(二进制指令代码)形式、便于计算机操作和运算。由源程序转换成目标程序是通过编译程序来完成的。不同的计算机语言有不同的编译程序。目前,对数学模型采用的计算机语言有 FORTRAN、Pascal、C 和 Ada 等。

在计算机上利用模型程序的运算实现辅助决策能够起到很好的效果。

2. 模型程序包

为了减少人们重复编制模型程序的工作,出现了模型程序包。由专人编制各种模型程

序组成程序包。用户调用相应的模型程序,输入所需要的数据,就可以在计算机上计算出该模型的运行结果。

模型程序包的特点如下:

(1) 模型程序包组织结构简单。程序包中的模型程序有的是分散存放的程序文件(由操作命令来调用),有的是通过多级菜单连接起来(通过菜单的选择项来调用的)。

(2) 各模型程序相对独立。各模型程序是分别挂在菜单的底层,各模型程序之间除了和菜单连接外,没有其他任何联系。

(3) 每个模型程序的数据是各自封闭的。每个模型程序使用时都要输入实际问题的数据。一个模型程序的数据不可能使用另一个模型的数据。

(4) 程序包是适合于模型间无关系的组织结构形式。程序包只适合各模型独立使用,不适合多模型的组合。

目前的模型程序包主要以数学模型为主体,如运筹学软件包等。

3. 模型库

模型库是将众多的模型按一定的结构形式组织起来,通过模型库管理系统对各个模型进行有效的管理和使用。

模型库像数据库一样,是一个共享资源。模型库中的模型可以重复使用,即可以被不同系统所调用,避免了冗余。通过模型库可以将多个模型组合起来构成更大的模型。这样模型库就比模型程序包具有更强的使用能力。

模型库的模型主要是数学模型,它是管理科学和运筹学中研究的模型,它们在辅助决策中起到了很明显的作用。另外还有数据处理模型、图形图像模型、报表模型、智能模型等。多种类型的模型不但扩充了辅助决策的能力,而且不同类型的模型组合起来,能适应更广泛的决策问题。

3.3.1.2 模型库中模型的种类和表示

1. 数学模型

数学模型是辅助决策中用得最多,使用范围最广泛的模型。数学模型的表示形式有方程形式、算法形式和程序形式。

1) 方程形式

数学模型的方程形式是建立变量之间的关系。例如,管理科学中应用的最广泛的线性规划模型的方程形式是由目标函数取极值(极大或极小)以及多个约束方程共同组成的。

方程形式是数学模型的一种数学结构形式。它建立了模型中变量的相互关系,反映了事物的规律性,具有高度的概括性。数学方程形式的直观性便于人们掌握事物的内在本质。

方程形式便于理解但不利于计算。对模型的介绍和解释说明一般用方程形式。

2) 算法形式

模型的算法是用一系列演算步骤表示模型的数学求解过程。当代入实际问题的数据,经过算法步骤的演算就可求出模型的理想结果(人工算法)。运筹学中的各个模型都给出了

具体的求解算法,可以通过手算就可以求出模型的解。

例如,线性规划模型是通过单纯形法进行求解的。输入目标函数中各目标变量的系数,确定目标的极值(极大或极小),输入多个约束方程的系数、约束值以及约束关系(小于、等于、大于等),通过单纯形法的演算步骤,就能求出理想的目标值和变量值。模型表示的算法形式能够计算出结果,很实用,但不直观。对模型的运用一般采用模型的算法形式。

3) 程序形式

模型在计算机中的求解是利用计算机语言按模型的算法步骤编制模型程序,在计算机中进行计算。

利用计算机语言编制模型程序需要将人工算法变换成计算机算法。由于人具有很强的形象思维和逻辑思维,而计算机不具有,故很多人工算法不能直接编制成程序,需要编制程序人员付出很大的劳动来完成模型程序的编制。

数学模型程序一般是利用数值计算语言来编制的。数值计算语言有 C、Pascal、Ada 和 Java 等,它们均具有较强的计算能力,如数组运算、链表使用、过程调用、循环与递归等。由于算法比较复杂,在进行数值计算中具有一定的计算精度限制,连续多次的数值计算会造成误差的增大,因此误差的控制是模型程序要解决的问题。

2. 数据处理模型

数据处理主要是对数据库中的数据的数据处理。数据处理模型是完成一定任务的数据处理过程。它与数学模型的不同在于它不需要复杂的计算,如矩阵运算(相乘、求逆等)、方程求解、递归迭代等计算,而是对数据库中的数据进行数据处理。数据处理的特点是处理的数据量很大。数据处理模型完成的基本工作为对数据的选择、投影、旋转、排序和统计等。一部分数据处理模型如统计模型具有辅助决策作用,更多的数据处理模型是完成两个数学模型之间的数据处理工作。

数据处理模型一般采用数据库语言来编制数据处理过程的程序。数据库语言有 Oracle、SQL Server 和 MySQL 等。

3. 图形图像模型

图形图像模型用于人机交互,使计算机更形象、更直观地将数据表现给用户。可以认为它属于人机交互模型。

图形模型一般以向量数据形式表示或以绘图程序形式表示。向量数据形式表示的特性直接可以显示在屏幕上。而以绘图程序形式表示的图形在显示时需要运行该程序,使它在屏幕上画出来。

图像模型是以点阵数据形式表示的。图像的数据文件一般存储量很大。图像要求愈清晰,色彩愈丰富,数据量愈大。

4. 报表模型

报表是数据处理的主要输出手段。它可以看成是人机交互的一种输出形式,也可以看成是数据处理的结果。由于报表的大量使用和报表格式的种类繁多,我们把它也作为一种

类型的模型。

报表模型是以程序形式表示的。通过程序描述报表的格式,数据取自数据库,运行报表程序能在打印机上输出各种类型的报表。报表程序一般用数据库语言编写。而一般报表工具却是用数值计算机语言(如 Pascal、C 语言等)编写,工具程序需要完成任意表格的生成,以及数据项表达式的识别和求解等复杂运算。它存取数据库中数据时,需要解决好接口问题,因为这些语言不能直接对数据库进行操作。

5. 智能模型

在人工智能中,应用最为广泛的是专家系统(关于专家系统将在 4.1 节详细说明)。它是以知识推理形式达到人类专家解决问题的能力。它是区别于数学模型、数据处理模型的一种重要辅助决策手段,我们称它为智能模型。在智能模型中,需要利用递归技术。为了便于智能模型的编制,出现了人工智能语言。20 世纪 60 年代出现了 LISP 语言,它是一种表处理语言,具有很强的递归功能。20 世纪 80 年代 PROLOG 语言形成热潮,它不但有很强的递归功能,还隐含了一个深度优先搜索的推理机制。它推动了专家系统的发展。近年来,由于知识推理要将数值计算和数据处理相结合,又兴起了用 C 语言编制智能模型的趋势。

智能模型是以智能程序形式表示的,它处理的对象是知识库。知识不同于数据,也不同于数学模型的方程和算法。专家系统用得最多的知识是产生式规则,以“if 条件 then 结论”形式表示。知识库是由大量的产生式规则知识和事实知识组成的,由知识库管理系统来管理。专家系统是一个独立系统,3.1.2 节中的三系统结构的决策支持系统和专家系统相似,利用知识辅助决策。

3.3.2 模型库的组织 and 存储

模型库的组织 and 存储是模型库的重要问题。模型库的组织形式与模型的表示形式有关。

模型库中除智能模型外,模型都以程序形式或数据文件表示,程序和数据都以文件存储。但程序又有源程序和目标程序,这样一个模型至少有两个文件。如果对模型进行文字说明,包括模型的方程形式以及算法的自然语言描述,将形成模型的说明文件。如果对模型的输入数据和输出数据进行说明,又将形成模型的数据描述文件,这样一个模型将对应 4 个文件。对这些文件需要建立一个文件库。对大量模型统一组织和存储,建立一个字典库来索引描述对应的模型文件就很有必要。这样,模型库由字典库和文件库两者组成。

3.3.2.1 字典库

模型字典库需要对模型的名称、编号和模型的文件等进行说明。

1. 字典库的作用

(1) 字典是模型文件的索引。每个模型都有 4 个文件。模型的数量多,相应的文件就更多。为了方便各模型与模型文件的联系,建立索引是非常必要的。

(2) 字典便于对模型的分类。随着技术的发展,模型将会愈来愈多,目前应用成熟的模

型已经相当多了,对模型分类很有必要。例如预测模型多达 200 多种,对预测模型进行分类就很有必要。按时间分类有短期预测、中期预测和长期预测。按预测结果和按限制条件等都能分类。对模型分类,首先要对模型字典分类。

(3) 字典方便了对模型的查询和修改。由于字典是模型文件的索引,要查询模型文件,通过索引能迅速地查找到所需要的模型。同时,也方便了对模型文件的修改。对模型文件的修改主要是对模型算法、参数以及有关模型说明的修改。修改工作一般包括增加、删除和更新等内容。

2. 字典库的组织结构

字典库的组织结构一般有文本形式、菜单形式和数据库形式等。

(1) 文本形式:模型字典内容用文本形式进行存储。这种形式把所有模型内容都以文字形式进行说明,存入文本文件中。这种形式的模型字典只能起查询作用。

(2) 菜单形式:模型字典用一个层次式的菜单来表示。菜单中的各项内容联系到各模型的模型文件,这样就把模型字典和模型文件联系上了,可以通过模型字典(菜单)运行模型文件(模型目标程序文件)和查询模型文件(模型源程序文件和模型说明文件)。

模型软件包一般采取这种形式。

(3) 数据库形式:模型字典的内容按照关系数据库的组织形式存放。按照模型分类就可以分别建立不同的字典库,一个库存放一类模型,每个模型是一个记录,每个记录中包含模型的编号、名称和各种模型文件名等数据项,这样字典库实质上是数据库,需要把它和有关模型文件本身联系起来。

这种组织存储形式便于模型的分类、查询和修改。决策支持系统一般采取这种形式。

3.3.2.2 模型文件库

模型文件是模型的主体。源程序和目标程序是主要的模型文件,一个模型至少有 2~4 个模型文件,模型库中大量的模型就对应着大量的模型文件。对这些模型文件,如何存储以及如何调用是一个关键问题。

1. 模型文件的存储方式

1) 直接在计算机操作系统管理下存储

这种方式是最简单和省事的。计算机操作系统对于数据文件和程序文件都是以文件形式统一存储和管理,操作系统按文件的大小以及存储空间中的空位决定该文件的存放位置,在文件目录中记录了该文件的起始地址。

采用这种存储方式,所有的模型文件的存储位置是杂乱的,它们和磁盘中其他文件混杂在一起。

2) 建立子目录存储模型文件

为了区分模型文件和磁盘上的其他文件,可以利用建立子目录的方法,把模型文件都建立在子目录下。具体可采用建立一个子目录存入所有的模型文件和建立多个子目录分开存放不同的模型文件两种方式,显然后者更好一些。建立子目录又有两种形式:

(1) 按模型分类建子目录。每类模型(如预测模型类)建一个子目录,该类中的模型的所有模型文件都存放在此子目录下。这样,模型文件库和模型字典库一一对应。在此子目录下的模型文件既含源程序文件又含目标程序文件以及有关的说明文件和数据描述文件。

(2) 按模型文件的类别建立子目录。模型文件类至少有两类(源文件和目标文件)到四类(增加说明文件和数据描述文件)。这样,建立 2~4 个子目录,分别存放各类的模型文件,即所有模型的源文件放在一个子目录下,所有模型的目标文件放在另一个子目录下。

采用这种方式,如果模型作为商品时只提供目标文件库而不提供源文件库就比较容易,即只需取出目标文件子目录下的所有模型文件即可。

2. 模型文件的调用

模型文件特别是目标程序文件的调用(即模型的运行)是模型库的另一个重要问题。模型文件的调用是与模型文件的存储方式直接有关的。调用模型文件首先要按它的存储路径找到该文件,然后,再启动该文件。

在操作系统下启动某程序文件,直接运行该文件名即可。在计算机各种语言中启动某文件需要利用此语言中运行某文件的命令。如 Pascal 语言有 EXEC 命令用来启动某文件运行。

对模型文件的运行,一般应该通过模型字典库,沿着模型文件的存储路径找到具体的模型文件,再启动它运行。

3.3.3 模型库管理系统

3.3.3.1 基本概念

模型库管理系统(Model Base Management System, MBMS)类似于数据库管理系统(DBMS)。数据库系统为了解决数据冗余和数据独立性问题,用一个管理系统(即 DBMS)来统一管理所有的数据,从而实现了数据的共享,同时,使数据的完整性和安全性等问题,也得到了相应的解决。

模型管理技术经历了 3 个发展阶段。

(1) 程序文件。这一阶段,模型算法以程序文件形式存放在计算机的存储器中,由操作系统的文件管理系统统一管理。程序文件分为源程序文件和目标程序文件两种,以程序文件名的后缀来区分。例如,用 Pascal 语言编写的源程序后缀为. PAS,目标程序文件名的后缀为. EXE。使用程序时,启动目标程序文件名即可。模型程序一般由用户自己编制,并且是现用现编。

(2) 模型软件包。这一阶段,对大量的较通用的模型由专职人员预先编制好模型程序,统一挂在交互菜单下。用户不用自己再编模型程序,而是在人机交互菜单的驱动下调用所需要的模型。模型之间是相对独立的,模型使用的数据仍放在各自的数据文件中。这比模型程序文件前进了一步,它是大量模型的简单组织机构。

(3) 模型库管理系统。这一阶段,建立了大量模型的有效组织机构。模型不是简单地挂在菜单下面,而是在管理系统下有效地对模型进行存储、修改、查询和调用。模型之间

不仅可以相互独立,而且可以互相组合。模型使用的数据就可以统一存放在数据库中,不同的模型可以存取同一个数据。这比模型软件包又前进了一步。

模型库管理系统是随决策支持系统的需要而发展起来的。它使模型管理技术提高到一个新水平。

模型不是简单的数据,它是一个程序文件或数据文件。这样模型库比数据库就要复杂得多。数据库技术目前已经成熟,而且还在继续发展,微机 and 小型机上的数据库管理系统的功能也在逐步地提高。但模型库技术处于发展阶段,还未有成熟软件。

模型能辅助决策,模型的组合辅助决策的能力更强,因此,模型的组合是模型库的新的研究内容。模型的组合形成了新的模型,相当于新模型的生成,称为新模型的建立。

建立模型本身就是一个困难的工作。对单一的问题建立单模型,对复杂的问题则需要用多个模型,并且将这些模型组合成一个大模型。对数学模型,建立模型实质上是建立数学方程。人工建模取决于人的经验,建立模型的效果则因人而异。用计算机来建模则要利用机器学习中的机器发现技术(计算机自动建立模型的数学方程)来实现,这是人工智能的一个研究领域。随着科学技术的发展,可以通过机器学习技术来进行计算机建模。

目前,计算机建模主要表现在两方面:

(1) 在已有的数学模型结构中确定变量,并生成方程的系数。

例如,线性规划问题的模型生成主要包括以下几个工作:

- ① 目标函数中变量个数的选取以及对应系数的生成。
- ② 约束方程中变量的选取、变量系数的生成以及约束值的生成。
- ③ 决定约束方程的个数。

这样,线性规划方程的具体形式就建立起来了,再利用线性规划的算法程序对它进行求解。

这种建模是将已成熟的模型应用到实际问题中去。例如,将线性规划模型应用到农业中去。这种建模也要做大量的工作。

(2) 选择基础模型并将它们组合成大模型。

在模型库中已经有了一些基本模型,在分析实际问题后,选择多个基本模型作为基础模型,根据模型间的逻辑关系将这些基础模型组合成实际问题的大模型。一般的逻辑关系有“与”(AND)和“或”(OR)。但是,组合模型形成复杂大模型应该利用程序设计中的三种组织结构的嵌套结构形式。这三种结构如下:

- (1) 顺序结构,模型间按前后顺序依次执行。
- (2) 选择结构,模型间按选择条件决定执行哪个模型。
- (3) 循环结构,一个模型或多个模型的多次反复执行。

这种建模实际是建立实际的决策支持系统。决策支持系统是多个模型的组合以形成决策方案。这种建模称为决策支持系统建模。

3.3.3.2 模型库管理系统的功能

模型库管理系统的主要功能有 3 个方面:模型的存储管理、模型的运行管理以及支持模型的组合。

1. 模型的存储管理

模型的存储管理包括模型的表示、模型存储组织结构以及模型的查询和维护等。

1) 模型的表示

模型的表示与模型自身的特点有关。

(1) 数学模型：在计算机中都是以数值计算语言的程序形式表示，在它数据后，执行程序就能得出结果。程序在计算机中仍是以文件形式存储。为区别于其他形式的文件，称它为程序文件。

(2) 数据处理模型：它是对大量数据库数据进行选择、投影、旋转、排序和统计等处理。它以数据库语言的程序形式表示，仍为程序文件。

(3) 图形图像模型：它是利用大量有灰度、有颜色的点阵组成的图像。例如人像是一个数据文件。

(4) 报表模型：它是有一定格式的结构，中间填入数值、文字等数据。它是由报表打印程序表示的。它在接收到要输出的数据后，将数据和报表框架一起形成报表在打印机上输出。它仍是一个程序文件。

用于辅助决策的模型主要是以上 4 种(智能模型以后再讨论)。不管是哪种，在计算机中都是文件形式，具体表示为程序文件或数据文件。

2) 模型存储的组织结构

模型表示为文件形式。如何组织存储就是一个很重要的问题。在模型数量少时，一般存放在计算机外存中，由操作系统中的文件系统进行管理，具体的组织管理方式是：在开始时顺序存放输入的各种文件，以后就按空位存放新输入的文件。这种存储组织方式以文件为单位，不过问文件的内容。文件的读取则是通过文件目录来找到文件的位置，再进行读取。对于大量的模型文件的存储，由操作系统来管理是不合适的，因为大量的存储空间既存储这个系统的文件，也存储那个系统的文件，这些文件都混杂地存储在一起，不利于单个系统中文件的独立管理。这样，就需要重新组织存储。

一个模型库的存储组织存储结构形式可以借鉴操作系统的方法来设计，即建立一个模型文件字典和模型文件库，在模型字典中指明模型文件的存储路径。在模型文件的存储上，把那些关系密切的或者类型相同的存放在一起，便于查找和存取。

组织存储结构形式还可以借鉴数据库的组织形式，建立多个库，在一个库中存放同类型的模型或者经常在一起使用的模型。例如，预测模型库存放用于预测的各种模型，优化模型库存放用于优化的各种模型。模型库的概念不能和数据库的概念相混，数据是一个基本单位，固定长度而且长度都很小；而一个模型则是一个文件，长度很大且不固定。数据库的形式只能用于模型字典库上，而模型文件仍以文件的形式存储。模型字典库的组织结构虽然与数据库的结构形式相同，但存放的内容不再是数据而是模型文件名。

模型库的组织存储结构包含两部分。第一部分是模型字典库，它类似于数据库的组织结构形式，但存储的是模型文件名。第二部分是模型文件库，它是模型的主体，具有文件的形式，按文件方式存储。在模型字典库中应该指明模型文件的存取路径。