

空间决策支持系统模型库系统研究

黄跃进¹ 反伟胜¹ 朱云龙²

(1. 杭州市规划设计院 310006; 2. 中国科学院沈阳自动化研究所 110015)

摘要: 文章探讨了面向对象的模型设计、管理. 提出将模型描述、源程序、目标文件等作为一个框架的存储组织方式, 讨论了基于 agent 的模型自动生成、运行、修改方法.

关键词: 空间决策支持系统, 模型, 对象, 主体, 模型库管理系统

中图分类号: TP13

文献标识码: B

1 前言

空间决策支持系统(SDSS)是地理信息系统(GIS)的一个重要发展方向, 有的专家干脆将 GIS 定义为空间决策支持系统. 但是, 大多数 GIS 以数据库软件系统为驱动核心, 模型在系统中处于从属地位. GIS 提供的主要是原始数据和有限的低层生成信息, 在辅助决策过程中 GIS 只能提供数据级支持, 而不能提供实质性的决策方案, 难以求解复杂的结构化较差的空间决策问题.

SDSS 是面向空间问题领域的决策支持系统(DSS), 主要用于求解难于具体描述和模拟的结构较差的空间问题. SDSS 能快速地决策者准确提供决策所需的数据、信息的背景材料、图形图像及报表, 帮助决策者明确决策目标. 建立修改决策模型和进行空间复合运算提供各种被选方案, 并对各种方案进行评价.

模型库系统是 SDSS 的核心, 模型库管理系统功能是: 快速简便地构造新模型, 通过数据库将若干模型连接起来构成合成模型, 对模型进行分类和维护, 方便地实现对模型的建立、修改、维护、连接和使用.

SDSS 模型库管理系统的研究是当前 GIS, DSS 等领域十分活跃的研究领域. 1980 年, Blanning 提出了关于模型库的概念, 并用模型库查询语言(MQL)来管理模型; 1984 年, Dolk 等提出了基于框架和知识表达的模型抽象技术; 1987 年, Geoffrion 设计了一套结构化模型构造语言, 首次将结构化程序设计思路植入模型生成问题; 1988 年, Muhana 等又将系统论的概念用于模型管理系统; 1993 年, Ting-Peng Liang 将同类推理知识学习方法融进了模型管理系统; K. M. Vanhee 建立了基于模型概念的模型运行环境系统. 1996 年, Wesseling 设计了动态模型语言来支持空间数据结构.^[1]

2 面向对象的模型库

模型是对客观事物的一种抽象描述, 人们通过对模型的认识来增加对复杂问题的理解和处理. 模型库中的模型可分为数学模型、数据处理模型、图形图像模型、报表模型, 智能模型等.

模型库是将众多的模型将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. 通常模型库由字典库和文件库组成, 利用字典库对模型的名称、编号、模

型的文件进行说明,模型文件中主要是源程序文件和目标程序文件.模型文件以文件形式直接存放在外存的某一个目录下.显然采用这种方式存在诸多如模型共享、安全等问题.利用面向对象的模型设计,面向对象的模型库组织和存储可以很好的解决上述问题.

2.1 面向对象的模型设计

可以采用下述形式建立面向对象的模型,对于已有模型程序文件也可以用相似的方式改造为面向对象形式.

Class Example : public Model

```
{
  数据部分
  (1) 输入参数表
  (2) 输出参数表
  方法部分
  (1) 构造函数与析构函数
  (2) 取出与存储操作
  (3) 模型执行方法
  (4) Static Build()
  (5) 其它模型方法
}
```

模型之间的调用通过对象提供的接口方法实现,所以修改已投入应用的模型时,不应改变已有接口方法名称,以免影响其他模型正常工作,可以参考部件对象模型(Component Object Model, COM)来开发模型

2.2 模型在模型库中的组织和存储

模型库中的模型事实上是人们认识事物的知识,所以可以用知识表示方法来表示模型,并采用人工智能方法来搜索、运行、管理模型并辅助建模人员进行模型的组合.

在各种知识表示方法中,框架表示法是一种比较接近于人类思维形式的知识表示方法,常被用来表示哪些比较复杂且具有一定结构性的领域知识.采用这种方法,知识表示的基本单位是框架,领域知识通过框架的层次结构来描述,框架的可嵌套性是构造框架层次结构的基础,



图1 三类框架关系

同时也体现了框架之间的继承关系.现将模型库中模型分为二类框架,一种框架用来表示一个单独的模型,第二类用来表示复杂合成模型,为了进行模型的组合引入第三类框架用来连接模型库中已有的模型成为合成模型.

连接框架有三种形式实现结构化程序设计对应的顺序、循环、分支结构功能.框架之间的关系见图1所示.

单一模型框架和合成模型框架的结构如图2所示.

连接框架相应的结构如图3(三种结构统一表示).

通常,GIS中的信息分为空间信息和属性信息,相应的有空间数据库和属性数据库,一般认为采用分库存储信息是不合理的.而采用面向对象的空间数据模型和面向对象数据库可以

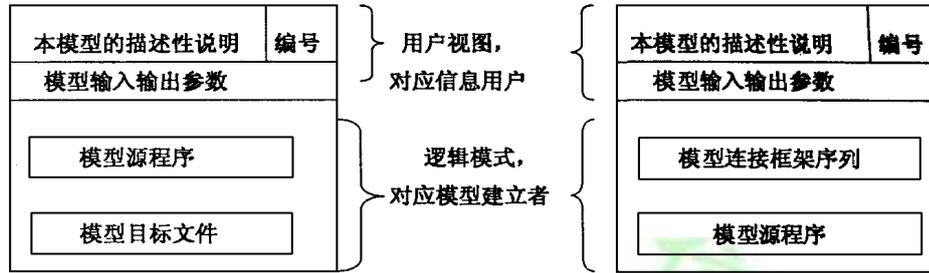


图 2 模型对应框架结构示意图

将一个对象的空间信息、属性信息存放在同一库中. 显然空间决策问题中的模型和模型库也应采用面向对象的方法表示比较合适.



图 3 连接框架结构示意图

2.3 框架的面向对象表示

采用面向对象方法表示时, 框架可以用类来描述, 实体用类的对象来描述, 框架的槽用相应类的属性来描述.^[2]

2.3.1 框架的数据结构定义

无对应实体的框架数据结构可表示为

```
struct frame {
    char    框架名
    slot-node 指向槽链的指针
}
```

槽链中每个槽节点可表示为

```
struct slot- node {
    char    槽名
    slot-node 指向下一槽结点的指针
}
```

对于有对应实体的框架的数据结构中, 应包含一个由该框架的诸实体所构成的一个实体链.

2.3.2 类的定义

在对类层次结构进行类定义时, 类的一个属性描述框架中的一个槽, 如果框架的某一个槽是另外一个子框架, 则应将此子框架的类定义为某父框架的类的友类, 并按公有方式派生, 如下例.

```
Class student : public people {
    Char-slot    * name
    Char-slot    * sex
    Int-slot     * age
Public
```

```

Student()
~Student()
void set-slot-value()
void show-slot-value()
Student * get-slot-value()

```

2.3.3 组合模型的框架表示

通过连接框架将模型连接成合成模型,合成模型指向第一个连接框架,整个合成模型通过指针和连接框架连成一体,合成框架的链结构见图 4.

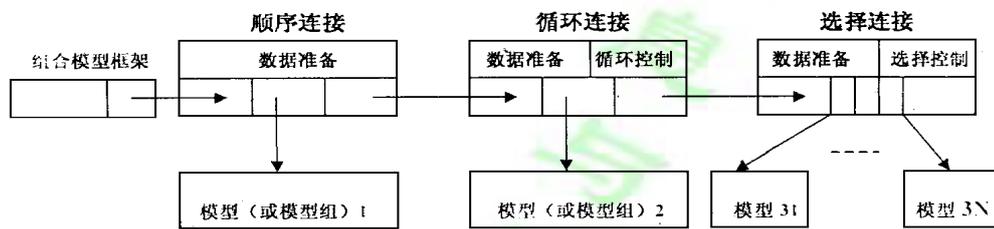


图 4 合成模型框架连接示意

为了方便的实现各模型框架的连接,框架之间的数据传递通过数据库来完成.

3 基于 agent 模型生成、运行和修改

模型库管理系统(Model Management System, MBMS)主要功能为:模型的存储管理、模型的运行、支持模型的组合.模型库管理系统类似于数据库的管理系统(DBMS),数据库系统是为解决数据冗余和数据独立性问题,用一个管理系统来统一管理所有的数据,从而实现数据的共享.同时,对于数据的完整性、安全性等问题,也得到相应的解决,显然模型库管理系统比一般数据库系统更复杂.

模型库管理系统的各种功能的实现,一般是由模型库管理系统语言体系来实现,根据模型库的特点,模型库管理系统语言体系分为模型管理语言(Model Management Language, MML),模型运行语言(Model Run Language, MRL),数据接口语言(Data Interface Language, DIL).

本文提出一种基于 agent 的模型库管理系统来实现模型库的智能管理,克服单纯采用模型库管理系统语言体系带来的困难.

Yoav Shoham 理论中定义了 agent:如果一个实体可以被视为包含了诸如信念、能力、选择、承诺等心态,这个实体便可以称为 agent.通常认为一个 agent 需具有下述部分或全部特性:自治性、交互性、协作性、可通信性、长寿性、自适应性、个性等.

3.1 模型构造 agent

为利用模型构造 agent 来可视的组合生成复合模型利用图标来表示模型库中已有的模型.通过在窗口中拖动模型对应的图标形成新的模型.模型的建立由使用人员和 agent 一起协作完成,agent 的作用是按使用人员的意图自动生成新的模型.

模型构造 Agent 内含用户与计算机进行人机对话的接口,接受用户输入后 agent 自动检

查使用人员用图标组合成的组合模型的合法性, 通过人机对话的接口与使用人员交互, 并可根据需要可将可视模型翻译成模型库管理系统语言代码。

进一步, 可以由多 agent 构成自动生成模型系统, 多 agent 系统结构见图 5 所示。图中, 由交互 agent 接收用户需求解的问题信息, 并为用户显示结果。问题分解 agent 将用户问题分解为多个系统所能求解的子问题。综合 agent 将各子问题 agent 的结果进行综合和评价, 从而产生问题的解。控制 agent 负责将各子问题分配给相应的子问题求解 agent。同时, 各 agent 的信息都传递到该 agent 的黑板中, 由控制 agent 控制信息的交换和各 agent 之间的通讯。各子问题 agent 根据内部知识选择相应的模型 agent。子问题 agent 除了具有问题求解推理能力, 同时还具有通讯和学习能力, 子问题 agent 可以通过通讯相互合作解决问题。^[3]模型 agent 则是在传统 CASE 的基础上封装相应的推理, 通讯和学习等能力, 是一个具有自主性的智能实体对象类。

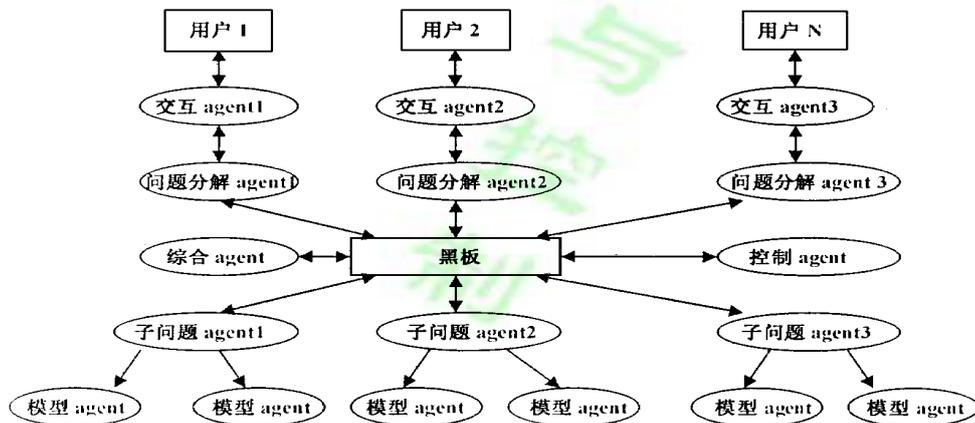


图 5 模型自动组合多 agent 关系示意

3.2 模型运行 agent

模型运行的第一步必须准确的选择相应的模型, 由于模型已采用知识表示法——框架表示法来表示, 在框架中有模型的描述(包括模型的适用条件)。所以可以将用户对需决策问题的描述用框架表示, 通过推理得到最接近的模型。

模型的运行不同于一般程序的运行, 尤其是复合模型以及空间求解问题采用 agent 来控制模型的运行是合适的。图 6 所示认知 agent 用来识别运行模型。

3.3 模型的修改

通常模型的修改不是一件容易的事, 模型的修改涉及源程序、目标文件、模型字典等改变。可以利用 agent 将修改后的源程序调用编译系统进行编译, 并替换旧的目标文件, 对于复合模型的修改同合成模型建立过程类似。

Agent 指出所需要修改的地方及修改可能影响到的模型, 并将修改内容存回。

4 实验系统和结论

实验系统采用的软件平台见图 7 所示, GIS 平台采用价格性能比较好的 Mapinfo。模型库

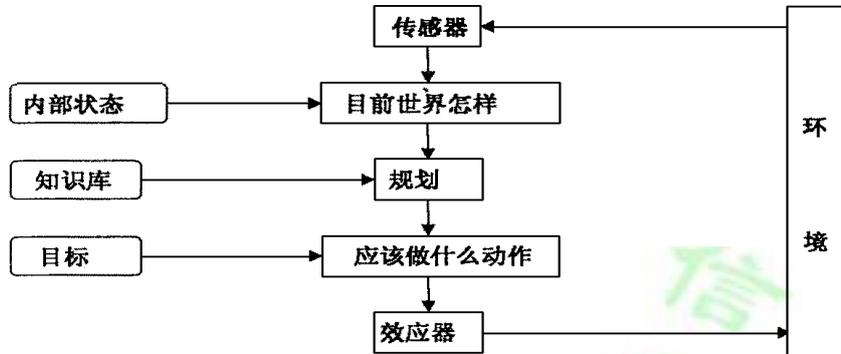


图 6 模型运行 agent 内部结构

采用 Notes 数据库代替, Notes 数据库以非结构化数据(主要为文档)为存储对象,各种形式的计算机文件都可以以附件方式存放在数据库中. 采用 VB 作为主要的开发工具, 因为 VB 具备编程简便、可以方便地操纵数据库和进行数值计算.



图 7 实验系统软件平台

根据文中所述方法进行土地管理及决策支持系统实验软件开发, 以实现土地管理及相应的辅助决策工作. 实验系统的重点是城市合理用地决策. 城市合理用地规模决策主要涉及经济水平、人口规模、生态环境、城区建设、郊区建设、区位条件等因素. 分别建立上述因素的模型(计算机程序). 最后用地规模的决策采用可能-满意度原理. 显然, 只有各因素的模型计算后才能用可能-满意度原理得出用地规模. 所以采用循序框架将各模型框架相连形成用地规模的决策模型. 直接引用资料[4]提供的数据进行计算, 结果显示采用框架表示模型是适宜的. 采用 agent 管理模型实现模型库的智能管理是决策支持系统的一个非常重要的问题. 由于时间关系这部分实验尚未完成, 有待进一步的研究和实验.

参 考 文 献

- 1 David A B. A Framework for the Integration of Geographical Information Systems and Modelbase Management, GEOGRAPHICAL INFORMATION SCIENCE, 1997, 11(4) (下转第 229 页)