

DOI:10.3969/j.issn.1673-4785.201210043

网络出版地址: <http://www.cnki.net/kcms/detail/23.1538.TP.20130409.1436.003.html>

## 基于 Agent 的专家系统推理模型

陶倩<sup>1</sup>, 马刚<sup>2</sup>, 史忠植<sup>2</sup>

(1. 上海体育学院 体育赛事研究中心, 上海 200438; 2. 中国科学院计算技术研究所 智能信息处理重点实验室, 北京 100190)

**摘要:**针对传统专家系统推理模型结构在知识获取方面适应性差的现状,从系统科学的视角,运用复杂适应系统理论,对传统专家系统的结构及运行机制进行了改进.引入 Agent 来模拟人脑中的神经元,用来承载专家系统中相互作用的知识,然后,基于 Multi-Agent 之间的相互作用来构建复杂适应的专家系统推理模型.从而,将专家系统中的知识获取机制、知识库、推理机三者统一于由 Multi-Agent 进行相互作用的复杂适应系统之中.通过设计体育赛事申办决策专家系统的原型,进行了专家系统推理模型的验证.原型运行结果表明:基于 Multi-Agent 的专家系统推理模型结构能够有效地提高专家系统知识获取的适应性.这为研究更加接近人脑智能的专家系统提供了崭新的研究思路.

**关键词:**Agent;推理模型;专家系统;体育赛事

**中图分类号:**TP391.6 **文献标志码:**A **文章编号:**1673-4785(2013)02-0135-08

## Research on the expert system reasoning model based on Agent

TAO Qian<sup>1</sup>, MA Gang<sup>2</sup>, SHI Zhongzhi<sup>2</sup>

(1. The Sports Event Research Center of Shanghai University of Sport, Shanghai 200438, China; 2. Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)

**Abstract:** The traditional expert system reasoning model structure has poor adaptability in acquiring knowledge. From the viewpoint of system science, the complex adaptive system theory is used to improve the structure and operation mechanism of a traditional expert system. Firstly, an Agent was introduced to simulate neurons in the human brain and load the knowledge interacting in the expert system reasoning model. Then an expert system reasoning model of complex adaptation was constructed based on the Multi-Agent interaction. Consequently, the knowledge acquiring mechanism, knowledge base and reasoning engine were unified into the Agents interaction in the complex adaptive expert system. Finally, by designing the expert system reasoning model prototype in decision-making of international sporting events bidding, the effectiveness of the expert system reasoning model based on Agent was verified. The results of the prototype running show that the expert system reasoning structure based on Multi-Agent model can effectively improve the adaptability of expert system knowledge acquisition. That provides a new idea for studying the expert system closer to human intelligence.

**Keywords:** Agent; reasoning model; expert system; sporting events

专家系统推理模型是对人类专家推理能力的建模,其经过 50 余年的发展,已经取得了许多重要的

进展和成果<sup>[1-9]</sup>.然而,随着研究的不断深入,人们发现,目前的专家系统推理模型并不能很好地模拟人类专家的推理能力,尤其是在对知识的获取上,还需依靠知识工程师先去理解领域专家的知识,并预先输入到专家系统的知识库中.专家系统推理模型的推理机制与知识库中的具体知识密切相关.因而,在专家系统后续运行的过程中很难进行适应性的补

收稿日期:2012-10-21. 网络出版日期:2013-04-09.

基金项目:国家“973”计划资助项目(2013CB329502);国家“863”计划资助项目(2012AA011003);国家自然科学基金资助项目(61035003, 61202212, 60933004);国家科技支撑计划资助项目(2012BA107B02).

通信作者:陶倩. E-mail: taopian\_101010@163.com.

充完善和发展创新。

目前,国内外大部分的研究成果在专家系统知识推理能力的建模上,都是采用知识获取机制、知识库、推理机这三大部件分离设计的结构。这种分离的结构是传统专家系统推理模型的重要特征。

然而,需要思考的问题是:将知识获取机制、推理机、知识库三大部件分离设计的结构是否能仿真人脑的知识获取过程呢?众所周知,人类是通过其大脑神经元中所承载的知识进行相互作用、相互关联,并构建神经网络通路来实现知识的获取及推理的。在人脑中并不存在静态的知识获取机制、知识库以及专门的推理机,也没有相互的调用关系。进一步的研究发现,人脑中的神经元作为知识的载体,其承载的知识在大脑中通过相互作用,来完成知识推理的过程<sup>[9]</sup>。神经元所承载的知识之间的相互作用是通过刺激信号来进行的,而推理过程则是通过刺激信号在神经元之间进行传递、构建动态的神经网络来完成的。人脑神经网络系统就是一个复杂的适应系统<sup>[2,10]</sup>。

根据贝塔朗菲一般系统论的定义“系统是相互联系、相互作用的诸元素的综合体”可知:虽然系统是由许多基本元素所组成的,但是,如果不考虑基本元素之间的相互作用,只是将这许多的基本元素简单地组合在一起,根本无法还原出原来的系统。

在传统专家系统推理模型的设计中,知识获取机制、知识库、推理机这三大部件是分离设计的,它们之间没有动态的相互作用,只能按照预先规划好的静态程序进行相互调用。由此可知,这三大部件的组合体不是一个真正仿真人类专家推理机制的系统。

近年来,随着人们对大脑神经网络认识的不断深入,许多研究成果运用人工神经网络技术来模拟人脑的神经网络,以构建专家系统的推理模型<sup>[6-9]</sup>。然而,这种人工神经网络推理模型是黑箱推理,其全部知识和推理过程都隐性地存储在人工神经网络的内部,很难解释其推理的过程和推理依据。

本文从系统科学的角度出发,应用复杂适应系统理论,对传统专家系统的结构及运行机制进行了改进。首先,引入 Agent 来模拟人脑中的神经元,用来承载专家系统中相互作用的知识,然后,基于 Multi-Agent 之间的相互作用构建复杂适应的专家系统推理模型<sup>[10-21]</sup>。从而,将专家系统中的知识获取机制、知识库、推理机三者统一于由 Multi-Agent 进行相互作用的复杂适应系统之中,最后,本文通过设计体育赛事申办决策专家系统的原型,进行了专家系统推理模型的验证。

## 1 多 Agent 专家系统推理模型的构建

### 1.1 模型结构

基于多 Agent 专家系统推理模型的结构与传统专家系统推理模型的结构不同,在其模型中没有明确设定推理机和知识库两大功能结构。而是仿真人脑神经系统中不同的神经功能区,分为 4 个系统功能部分:任务感知单元、任务求解单元、结果综合单元和智能引擎。其中,都是由多个 Agent 的相互作用所构成的。这些 Agent 被分别称为:任务感知 Agent、任务求解 Agent、结果综合 Agent 和控制 Agent。

感知单元中的 Agent 用于承载感知决策任务时所需的经验和知识,结果综合单元中的 Agent 用于承载对任务求解结果进行综合并输出的知识,而任务求解单元中的 Agent 则主要承载进行任务求解时所需的相关经验和知识,智能引擎用于承载对任务感知单元、任务求解单元、结果综合单元三大单元中 Agent 的运行进行控制的相关知识。

在基于 Multi-Agent 专家系统推理模型的结构中,所需的全部知识都被分散在相关的任务感知 Agent、任务求解 Agent、结果综合 Agent 和控制 Agent 中,而具体的推理逻辑就如大脑中各神经元的相互作用那样,是由相关 Agent 间的相互作用来动态决定的,同时,能够通过相关的控制 Agent 激活各 Agent 中的适应控制规则,来进行适应性改变。由此,与传统专家系统的结构相比,本文设计的结构增加了专家系统推理模型在知识获取方面的适应性。

### 1.2 模型的推理运行机制

在基于 Multi-Agent 的专家系统推理的模型中,推理过程是由智能引擎开始的。具体运行流程如下:

当某个决策任务被提交到推理模型时,将首先触发智能引擎中的相关控制 Agent 运行,控制 Agent 将发出相应的触发信号去激活感知单元中相应的任务感知 Agent。在某一时刻,可能会有多个任务感知 Agent 被同时激活,因此,控制 Agent 会根据任务感知 Agent 信用度的大小,经过相互竞争,决定哪个任务感知 Agent 被真正激活。而被激活的任务感知 Agent,将依据其承载的相应知识,进一步触发相关的其他任务感知 Agent。控制 Agent 会将任务感知 Agent 的运行轨迹记录到其结果表中,并根据感知的结果触发相关的任务求解 Agent。任务求解 Agent 同样会根据其承载的知识进一步触发相关的其他任务求解 Agent。同时,控制 Agent 将任务求解 Agent 运行的结果记录到其结果表中。进一步,控制 Agent 会根据任务求解的结果去触发结果综合 Agent。控制

Agent 也会将结果综合 Agent 综合分析的结果输出给用户. 当每次决策任务完成后, 控制 Agent 还将负责接收用户对各 Agent 运行结果的反馈信息, 并激活 Agent 中的适应控制规则.

### 1.3 Agent 结构

在基于 Multi-Agent 的专家系统推理模型中, 每个 Agent 都相当于人类大脑中的神经元, 用于承载相关的知识. Agent 结构由 4 部分构成, 即标识、感应器、规则和效应器. 规则表示 Agent 中所承载的知识, 在每个 Agent 中都有 2 类规则, 一类是求解规则, 另一类是适应控制规则; 感应器用于感应触发信号, 有激活状态和非激活状态 2 种状态; 效应器用于产生激活其他 Agent 的触发信号.

Agent 结构的形式化描述为

$Agent ::= \langle id, percep, rule, effector \rangle$ .

式中:  $id$  表示 Agent 的标识,  $percep$  表示 Agent 的感应器,  $rule$  表示 Agent 中的规则,  $effector$  表示为 Agent 的效应器.

Agent 的感应器所接受的信号可由二进制字符串表示, 即由集合  $\{0, 1, \#\}^L$  构成,  $L$  表示二进制字符串的长度,  $\#$  表示在此位置上可以接收任意字符串. Agent 感应器接收的输入信号有 2 类: 一类是求解信号, 另一类是适应控制信号.

Agent 的感应器接收触发信号后, 将根据感应规则判断自己是否被激活, 若被激活, 则进一步判断是求解信号还是适应控制信号. 如果是求解信号, 运行相应的求解规则, 并将结果输出到结果表中, 同时产生相应的触发信号; 如果是适应控制信号, 则运行相应的适应控制规则, 进行适应性进化.

### 1.4 Agent 中知识的表示

在 Agent 的结构模型中, Agent 承载的知识是用规则来表示的, 规则的形式化描述为

$Rule ::= \langle Rid, Do, Dflag, Strength, Activation \rangle$ .

式中:  $Rid$  表示规则的标识;  $Do$  表示如果满足触发条件, 需要做的工作, 且  $Do$  的内容由集合  $\{0, 1\}^K$  构成,  $K$  表示二进制字符串的长度,  $Do$  内容的具体性质则由  $Do$  属性来决定.  $Do$  中的内容有 2 种可能: 1) 产生一个要执行过程的索引, 这个执行过程既可以是一个固定的算法, 也可以是一个具有学习适应功能的程序, 而具体执行过程的详细定义, 由执行过程索引表来说明; 2) 产生具体参数的索引, 而具体参数的详细定义, 由规则参数索引表来说明.  $Dflag$  表示  $Do$  的属性, 如果  $Dflag = 0$ , 表明  $Do$  中的内容是执行过程索引表中具体过程的索引; 如果  $Dflag = 1$ , 表明  $Do$  中的内容是规则参数索引表中具体参数

的索引;  $Strength$  表示规则的信用度, 在 Agent 中, 其规则具有相应的信用度, 该信用度是在智能引擎的控制下, 由相应的信用分派算法进行赋值的;  $Activation$  表示当前规则是否被激活.

规则参数索引表中参数的形式化描述为

$Parameter ::= \langle Paid, Paras \rangle$ .

式中:  $Paid$  表示参数的索引;  $Paras$  表示具体参数的集合.

规则参数索引表的形式化描述为

$Parameter\_Table ::= \langle Parameters \rangle$ .

式中:  $Parameters$  表示参数的集合. 执行过程索引表中具体执行过程的形式化描述为:

$Process ::= \langle Proid, Pros \rangle$ .

式中:  $Proid$  表示执行过程的索引,  $Pros$  表示指向具体执行过程的指针. 执行过程索引表的形式化描述为

$Process\_Table ::= \langle Processes \rangle$ .

式中:  $Processes$  表示执行过程的集合.

### 1.5 Agent 间的信号传递

在本研究中, 当一个 Agent 需要和其他 Agent 进行相互作用的时候, 它会向系统中所有的 Agent 发出一个触发信号, 系统中的各个 Agent 在接收到这个触发信号后, 首先进行感知, 以判定自己是否将被触发. 当某个 Agent 被触发后, 该 Agent 就处于激活状态. 如果有多个 Agent 被同时激活时, 系统会根据被激活 Agent 的信用度大小, 经过相互竞争, 决定哪些 Agent 被真正激活.

## 2 基于 Multi-Agent 的体育赛事申办决策专家系统的推理模型分析

### 2.1 体育赛事的申办决策

体育赛事的申办决策是指申办主体对某项体育赛事是否申办进行选择的决策过程. 这个过程是指从申办主体有了举办某项体育赛事的意向到具体确定举办该项体育赛事, 且在进行申办签约之前的过程.

随着城市营销理念的开展, 作为城市营销实践的重要工具和载体的体育赛事越来越受到各大城市的青睐. 然而, 随着体育赛事举办规模的不断扩大, 其中申办决策所需的知识也越来越复杂. 因此, 对体育赛事申办决策专家系统可持续的知识获取能力的要求也越来越高.

目前, 申办主体对体育赛事的申办决策主要是通过通过对赛事的宏观环境条件、赛事资源条件、赛事市场竞争环境、赛事经济影响、赛事社会影响、赛事环境影响等指标体系进行效益评估, 来决定某项赛事

是否需要申办。

本文将以体育赛事的申办决策为例,设计基于 Multi-Agent 的体育赛事申办决策专家系统推理模型的原型系统。

## 2.2 推理模型分析

根据前述基于 Multi-Agent 的专家系统推理模型的理论框架,本原型系统的结构可分为4个功能部分:任务感知单元、任务求解单元、结果综合单元和智能引擎。

在原型任务感知单元中,每个感知 Agent 都代表体育赛事申办决策的某个一级理论指标,用于感知求解任务。本原型共设置了3个感知 Agent,即:体育赛事申办可行性评估的感知 Agent、赛事申办必要性评估的感知 Agent 和赛事申办风险评估的感知 Agent。这3个感知 Agent 承载了对体育赛事申办决策一级理论指标感知的知识。

在任务求解单元中首先设置了8个任务求解 Agent 来分别代表体育赛事申办决策的二级理论指标,用于对一级理论指标的求解。具体包括:体育赛事申办的宏观环境影响求解 Agent、体育赛事申办的资源条件求解 Agent、体育赛事申办的市场条件求解 Agent、体育赛事申办的竞争环境求解 Agent、体育赛事申办对经济积极影响的求解 Agent、体育赛事申办对社会积极影响的求解 Agent、体育赛事申办对环境积极影响的求解 Agent 和体育赛事申办

的风险评估 Agent。这8个任务求解 Agent 承载了对体育赛事申办决策一级理论指标的求解知识。

另外设置了29个任务求解 Agent,以代表体育赛事申办决策的三级理论指标。具体包括:举办地的政治环境、举办地的经济环境、举办地的人文环境、举办地的城市发展、举办地的自然环境、举办地对赛事的规划和布局、举办地政府的支持、城市基础设施建设、体育场馆设施建设、举办赛事所需的资金、赛事运作团队与赛事运作经验、媒体转播条件、接待条件、赛事的安保能力、志愿者队伍、赛事的品牌知名度、赛事与居民需求的契合度、赛事期间其他城市举办大型活动的可能性、赛事举办期内举办地其他的活动、体育赛事的吸引力、举办城市的其他旅游资源、体育赛事的赛程时间、体育赛事的门票价格、举办城市的物价水平、增强举办地居民自豪感、改善举办地居民生活质量、促进举办地体育发展、塑造举办地城市形象和对举办地城市环境进行治理等。这29个任务求解 Agent 承载了对体育赛事申办决策二级理论指标的求解知识。

在结果综合单元中设置了2个结果综合 Agent,一个用于承载求解结果输出的格式的知识;另一个用于承载各任务求解 Agent 求解结果综合的知识。

在智能引擎中设置了4个控制 Agent,分别用于任务感知 Agent、任务求解 Agent、结果综合 Agent 和 Agent 的适应控制。

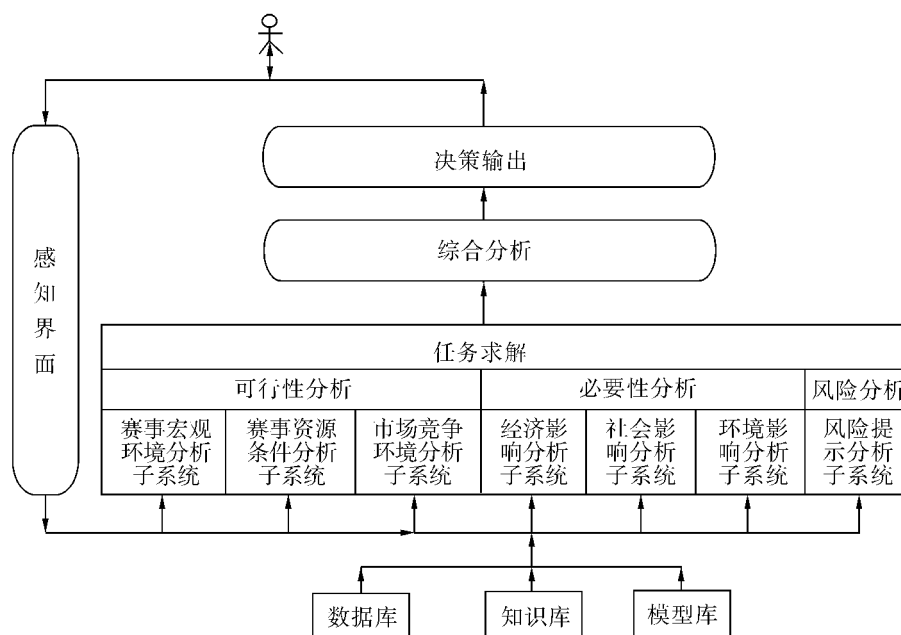


图1 决策分析逻辑图

Fig. 1 Decision analysis logic diagram

在原型运行时,任务感知 Agent 中的一级理论指标会根据需求被触发,其根据相应的知识规则进一步触发代表二级理论指标的任务求解 Agent,代表二级理论指标的任务求解 Agent 同样会根据其内的知识规则来决定具体触发哪个代表三级理论指标的任务求解 Agent. 如果专家系统还需要有四级理论指标,同样可以很方便地增加、设置代表四级理论指标的任务求解 Agent. 由此可知,Agent 中的知识规则不仅代表专家的知识,同时还决定了专家知识推理的逻辑. 这样的设计与人脑的思维方式更为接近. 而且,知识的获取也可以通过 Agent 规则的增加或修改进行. 从而,将专家系统中的知识获取机制、知识库、推理机三者统一于 Agent 的相互作用之中. 这样的设置不仅提高了专家系统推理机制的灵活性,也改进了专家系统的适应性.

如果原型中还需要增加新的理论指标系统时,只需要再增加新的感知 Agent,并同时在任务求解单元增加相应的任务求解 Agent,结果综合单元中增加新的结果综合 Agent 即可. 各 Agent 之间的相互作用是根据 Agent 中的相关规则而进行的. 整个系统决策过程的逻辑图如图 1 所示.

### 3 基于 Multi-Agent 的体育赛事申办决策专家系统推理模型原型设计

#### 3.1 原型设计

本文在原型设计中,设计了最基本的 Agent 模板类,它是基于 Multi-Agent 的体育赛事申办决策专家系统推理模型中所有 Agent 的基础类. 其他 Agent 类可以对它进行继承.

其基本结构可描述如下:

```
public class Agent { //基本 Agent 类
public int Agent_id; // Agent 标识
public Vector crules; // 规则表
public Vector signs_table; // 当前信号表
public Vector parameter; // 参数索引表
public Vector process; // 过程索引表
public Vector signs_relation; // 信号关系表
public Vector result_table; // 结果表
public Agent() { // 构造函数
set Agent initialization
}
public receive _ signal () // 接收信号
public set transform_function() // 执行规则
public self_evolution () //规则自适应
```

```
public output _ signal () // 发送触发信号
public write _ Result_table () // 写结果表
}
```

感知 Agent 类继承基本 Agent 类,用类似 Java 的语言描述如下:

```
public class sensing Agent extends Agent { // 感知 Agents
public int Agent_id; // Agent 标识
public Vector crules; //规则表
public Vector signs_table; //当前信号表
public Vector parameter; //参数索引表
public Vector process; //过程索引表
public Vector signs_relation; //信号关系表
public Vector Result_table; //结果表
public sensing Agent() { //结构函数
set sensing Agent initialization
}
public receive _ signal () // 接收信号
public set transform_function() //执行规则
public self_evolution () //规则自适应
public output _ signal () //发送触发信号
public write _ result_table () //写结果表
.....
}
```

任务求解 Agent 类也是继承基本 Agent 类,用类似 Java 的语言描述如下:

```
Public class task solving Agent extends Agent { //任务求解 Agents
public int Agent_id; // Agent 标识
public Vector crules; //规则表
public Vector signs_table; //当前信号表
public Vector parameter; // 参数索引表
public Vector process; //过程索引表
public Vector signs_relation; //信号关系表
public Vector result_table; //结果表
public task solving Agent() { //结构函数
set task solving Agent initialization
}
public receive _ signal () // 接收信号
public set transform_function() //执行规则
public self_evolution () //规则自适应
public output _ signal () //发送触发信号
public write _ result_table () //写结果表
.....
}
```

结果综合 Agent 类也是继承基本 Agent 类,用类似 Java 的语言描述如下:

```
public class results comprehensive Agent extends Agent { //结果综合 Agents
    public int Agent_id; // Agent 标识
    public Vector crules; //规则表
    public Vector signs_table; //当前信号表
    public Vector parameter; //参数索引表
    public Vector process; //过程索引表
    public Vector signs_relation; //信号关系表
    public Vector result_table; //结果表
    public results comprehensive Agent() { //构造函数
        set results comprehensive Agent initialization
    }
    public receive_signal() //接收信号
    public set_transform_function() //执行规则
    public self_evolution() //规则自适应
    public output_signal() //发送触发信号
    public write_Result_table() //写结果表
    .....
}
```

智能引擎类也是继承基本 Agent 类,用类似 Java 的语言描述如下:

```
public class intelligent engine extends Agent { //智能引擎
```

```
public int Agent_id; //智能引擎
public Vector crules; //规则表
public Vector signs_table; //当前信号表
public Vector parameter; //参数索引表
public Vector process; //过程索引表
public Vector signs_relation; //信号关系表
public Vector result_table; //结果表
public intelligent engine() { //构造函数
    set intelligent engine initialization
}
public receive_signal() //接收信号
public set_transform_function() //执行规则
public self_evolution() //规则自适应
public output_signal() //发送触发信号
public write_Result_table() //写结果表
.....
}
```

### 3.2 原型的实现与运行

图2的输出界面显示,是本文所设计的专家系统,

它既能够对体育赛事申办决策的二级理论指标进行单项指标的评估决策,也能够综合整个理论指标体系进行综合的体育赛事申办决策. 这些决策的执行不像传统专家系统,预先设置好功能模块,而是由感知 Agent 感知后通过触发相应的 Agent 来完成的.

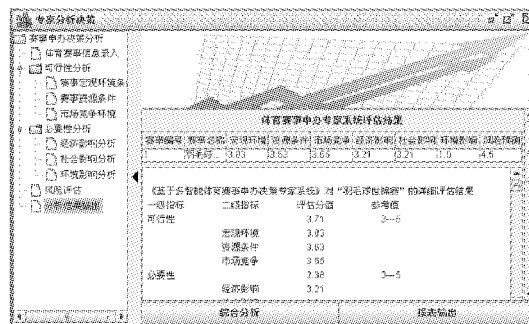


图2 决策结果输出界面 I

Fig. 2 The decision result output interface I

在图2的输出界面中,给出了体育赛事申办决策专家系统对整个理论指标体系进行综合决策后的文字性输出结果. 其中显示了专家系统对体育赛事申办决策的所有二级理论指标的评估分值,并给出了该指标适合申办决策时的理论参考值范围.

图3的输出界面则是将体育赛事申办决策专家系统综合决策的文字性输出结果用图形来显示.

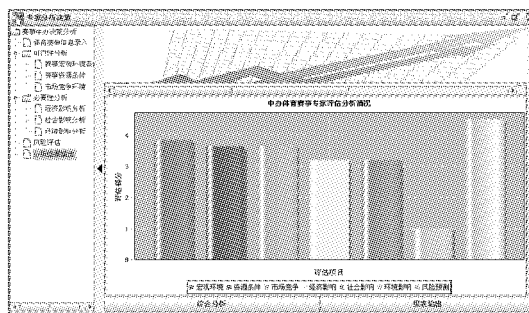


图3 决策结果输出界面 II

Fig. 3 The decision result output interface II

## 4 结束语

本文通过快速原型法,实现了基于 Multi-Agent 的体育赛事申办决策专家系统的推理模型. 运行结果表明:所设计的基于 Multi-Agent 的专家系统推理模型结构能够有效地提高专家系统的适应性. 这为研究更加接近人脑智能的专家系统提供了崭新的研究思路.

## 参考文献:

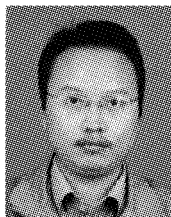
[1] 史忠植,王文杰. 人工智能[M]. 北京:国防工业出版社,

- 2007; 21-30.
- [2] 邵艳华, 张明生. 一类复杂适应系统的建模研究[J]. 计算机工程, 2012, 38(1): 253-255.  
SHAO Yanhua, ZHANG Mingsheng. Research on a class of complex adaptive system modeling[J]. Computer Engineering, 2012, 38(1): 253-255.
- [3] 韩业红. 基于粗集理论的专家系统中知识的获取、更新与推理[D]. 济南: 山东师范大学, 2007: 29-35.  
HAN Yehong. Knowledge acquisition, modification and reasoning in expert system based on rough set theory [D]. Ji'nan: Shandong Normal University, 2007: 29-35.
- [4] 钟秀琴, 符红光, 余莉, 等. 基于本体的几何学知识获取及知识表示[J]. 计算机学报, 2010, 33(1): 167-174.  
ZHONG Xiuxin, FU Hongguang, SHE Li, et al. Geometry knowledge acquisition and representation on ontology [J]. Chinese Journal of Computers, 2010, 33(1): 167-174.
- [5] 陈明亮, 李怀祖. 基于规则的专家系统中不确定性推理的研究[J]. 计算机工程与应用, 2000(5): 50-53.  
CHEN Mingliang, LI Huaizhu. Study on the non-accurate inference in expert system based on rule[J]. Computer Engineering and Application, 2000(5): 50-53.
- [6] 曹珊, 贺正洪. 基于神经网络的专家系统研究及应用[J]. 战术导弹控制技术, 2007, 56(1): 52-55.  
CHAO Shan, HE Zhenghong. Application of expert system based on artificial neural network[J]. Control Technology of Tactical Missile, 2007, 56(1): 52-55.
- [7] 徐敏, 施化吉. 基于神经网络集成的专家系统模型[J]. 计算机工程与设计, 2006, 27(7): 1216-1220.  
XU Min, SHI Huaji. Expert system model based on neural network ensembles[J]. Computer Engineering and Design, 2006, 27(7): 1216-1220.
- [8] 巩文科, 李心广, 赵洁. 基于 BP 神经网络与专家系统的故障诊断系统[J]. 计算机工程, 2007, 33(8): 199-203.  
GONG Wenke, LI Xinguang, ZHAO Jie. Fault diagnosis system based on BP neural network and expert system[J]. Computer Engineering, 2007, 33(8): 199-203.
- [9] 陈德礼, 程羽. 基于神经网络和规则的专家系统的应用研究[J]. 计算机工程与科学, 2004, 26(6): 70-75.  
CHEN Deli, CHENG Yu. Application research of the ESs based on neural networks and rules [J]. Computer Engineering & Science, 2004, 26(6): 70-75.
- [10] 陈禹. 复杂性研究的新动向—基于主体的建模方法及其启迪[J]. 系统辩证学学报, 2003, 11(1): 43-50.  
CHEN Yu. A new trend in complexity studies—Agent based modelling and its implication[J]. Journal of Systemic Dialectics, 2003, 11(1): 43-50.
- [11] 陶倩, 徐福缘, 黄平. 基于 Agent 的计算金融学建模方法研究[J]. 系统仿真学报, 2008, 20(11): 3004-3007.  
TAO Qian, XU Fuyuan, HUANG Ping. Agent-based computational finance modeling[J]. Journal of System Simulation, 2008, 20(11): 3004-3007.
- [12] 金士尧, 黄红兵, 范高俊. 面向涌现的多 Agent 系统研究及其进展[J]. 计算机学报, 2008, 31(6): 881-895.  
JIN Shiyao, HUANG Hongbing, FAN Gaojun. Emergence-oriented research on multi-agent systems and its state of arts[J]. Chinese Journal of Computers, 2008, 31(6): 881-895.
- [13] Agent based modeling[EB/OL]. [2012-08-13]. [http://www.cadrc.calpoly.edu/pdf/JSVoss\\_120400.pdf](http://www.cadrc.calpoly.edu/pdf/JSVoss_120400.pdf).
- [14] ROBERT A. Advancing the art of simulation in the social sciences[J]. Complexity Magazine, 1987, 3(2): 189-194.
- [15] CLAUDIA P W, EBENOTH E. An adaptive toolbox model: a pluralistic modelling approach for human behaviour based on observation [J]. Journal of Artificial Societies and Social Simulation, 2004, 7(1): 176-181.
- [16] BOERO R. Some methodological issues of agent based models in social sciences [EB/OL]. [2012-07-26]. [http://www.unisi.it/santachiara/aree/conf\\_phd\\_econ2003/conference\\_siena/papers/boero.pdf](http://www.unisi.it/santachiara/aree/conf_phd_econ2003/conference_siena/papers/boero.pdf).
- [17] SUN R, NAVEH I. Simulating organizational decision-making using a cognitively realistic Agent model [J]. Journal of Artificial Societies and Social Simulation, 2004, 7(3): 253-258.
- [18] TESTATSION L. Agent-based computational economics: growing economies from the bottom up[J]. Artificial Life, 2002, 8(1): 55-82.
- [19] PIERRE M J, CHRISTINE B, GABRIEL L, et al. Bio-inspired mechanisms for artificial self-organized systems[J]. Informatica, 2006, 30(1): 55-62.
- [20] WAGNER G, TULBA F. Agent-oriented modeling and Agent-based simulation [C]//Proceedings of 5th Int Workshop on Agent-Oriented Information Systems(AOIS-2003). [S.l.]: Springer-Verlag, 2003: 57-63.
- [21] MILLER J H, PAGE S E. Complex adaptive systems: an introduction to computational models of social life [M]. Princeton, USA: Princeton University Press, 2007: 25-28.

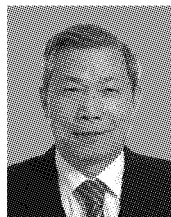
#### 作者简介:



陶倩,女,1966年生,副教授,主要研究方向为复杂适应系统建模、智能信息处理。曾主持和参与省部级科研项目,以及横向课题项目多项。发表学术论文多篇,其中2篇被EI检索。



马刚,男,1986年生,硕士研究生,主要研究方向为机器学习、数据挖掘、神经计算.曾参与国家“973”项目子课题、国家自然科学基金项目以及其他横向课题的研究,完成了多个基于多智能体专家系统的研发工作.



史忠植,男,1941年生,研究员,博士生导师.主要研究方向为智能科学、人工智能、机器学习、知识工程等.1979年、1998年、2001年均获中国科学院科技进步二等奖,1994年获中国科学院科技进步特等奖,2002年获国家科技进步二等奖.发表学术论文400余篇,出版专著15部.

## 2013 中国自动化大会暨自动化领域协同创新大会

2013 中国自动化大会暨自动化领域协同创新会议(CAC2013)将于2013年11月7-8日在湖南省长沙市召开,同时召开“自动化技术与装备”展览会,开展自动化领域“产学研”的互动交流.

### 一、征文范围

根据大会程序委员会安排,本次大会将设立8个专题、22个征文领域.热烈欢迎全国各高等院校、科研院所和企事业单位从事自动化理论与技术研究的科技工作者积极投稿,特别希望征集能反映各单位在自动化领域研究特色的学术论文.

大会专题:

专题1:智能制造装备与测控技术(负责人:王耀南,褚健)

专题2:碳效优化与环境控制(负责人:吴敏)

专题3:复杂系统建模与控制(负责人:王宏,李少远)

专题4:无人系统自主控制(负责人:沈林成)

专题5:空天飞行器控制技术(负责人:包为民)

专题6:生物信息与认知计算(负责人:胡德文,李远清)

专题7: Cyber-Physical Systems(负责人:王飞跃,关新平,陈积明)

专题8:控制系统运行安全性(负责人:周东华,阳春华)

### 二、投稿要求

1、来稿未曾公开发表过,具备真实性和原创性.论文摘要及全文请勿涉及国家秘密.

2、凡投稿论文被录用且未作特殊声明者,视为已同意授权出版.

3、论文篇幅不限,中英文均可,特别欢迎能反映本单位研究特色的长文,征稿截止日期为2013年8月31日,录用通知日期为2013年9月30日.

4、论文投稿请通过登录中国自动化学会网站(网址 <http://www.caa.org.cn>)在线投稿系统会议投稿专栏投稿.投稿时请务必注明论文投稿的专题或者征文领域.

### 三、论文出版

出版会议论文集,被IEEE Xplore检索,部分优秀论文拟推荐到国内外SCI、EI检索的重要期刊以专辑形式发表,各专题论文会后将组织出版专著.同时,大会还将讨论自动化领域在“十三五”期间重点研究方向,形成报告并向国家有关部门提交.

联系人:黄振华,余洪山,梁桥康

电话:18374966022(黄振华)、18374966028(余洪山)、18374966011(梁桥康) 0731-84573323

地址:湖南省长沙市国防科技大学机电工程与自动化学院

邮编:410073

E-mail:2013CAC@gmail.com