

# 并行工程中 MAS 决策的协商机制及其评价<sup>\*</sup>

段 华<sup>1,2</sup> 曾庆田<sup>2</sup> 韩丛英<sup>1,2</sup> 廉文娟<sup>2</sup>

(上海交通大学应用数学系 上海 200040)<sup>1</sup> (山东科技大学信息科学与工程学院 青岛 266510)<sup>2</sup>

**摘 要** 并行工程中,多 Agent 智能决策是一个核心组成部分。Agent 间决策结果的不一致性会影响整个并行工程的决策结果,需要采取适当的协商机制调节 Agent 之间的矛盾。本文给出了可信度优先、权威性优先、相邻影响、彼此影响等四种 Agent 协商机制以及相应的决策公式,并对它们进行了量化比较,分析了各自的优越性和适应性情形。最后给出了基于协商的 Agent 分组决策的一致性策略。

**关键词** 并行工程,知识管理,智能决策,Agent,协商机制

## Cooperation Strategies for Multi-Agent Decisions in Concurrent Engineering

DUAN Hua<sup>1,2</sup> ZENG Qing-Tian<sup>2</sup> HAN Cong-Ying<sup>1,2</sup> LIAN Wen-Juan<sup>2</sup>

(Department of Mathematics, Shanghai Jiaotong University, Shanghai 200040)<sup>1</sup>

(College of Information Sci. & Tech., Shandong University of Science and Technology, Qingdao 266510)<sup>2</sup>

**Abstract** In concurrent engineering, the inconsistency of the multi-Agent decisions can influence the results of the whole engineering. We introduce four inconsistency processing strategies for the multi-Agent decisions, which are reliability-first, authority-first, border-influence, and all-influence. The decision formulae of these four strategies are presented, and their decision results are compared by listing the histogram, with which their advantages and adaptability are shown. The consistency decision strategy of the Agent group is given finally.

**Keywords** Concurrent engineering, Knowledge management, Intelligent decision, Agent, Cooperation strategy

## 1 引言

并行工程(Concurrent Engineering)<sup>[1,2]</sup>是为了尽可能地缩短产品上市时间(time-to-market),同时兼顾质量、成本等诸多因素而采取的设计哲理。并行工程通过多学科团队、过程分析和改进、产品数据管理、协同工作环境、自动化的设计支持工具及信息集成等方法 and 工具集,来促进多学科、跨部门的信息共享和协作,使得在产品开发过程的早期,就尽可能地考虑下游的因素(如:可制造性、可维护性等),以减少工程更改,缩短开发周期,实现产品开发的一次成功和设计成果的优化。

知识管理是并行工程中的一个核心内容,基于知识的智能决策活动又是知识管理的主要组成部分。知识支持的任务通常是由多个智能 Agent 分布式处理,然后通过协同等方式完成并做出决策。在决策过程中,对于同一个问题,不同的 Agent 常常会给出互不相同的结论,即为多 Agent 之间决策的不一致性。当决策不一致时,系统所给定的结论取决于推理机的选择策略,因此必须给出适当的协商机制有效解决 Agent 之间的冲突,协调它们之间的矛盾以便给出正确有效的决策。

对于并行工程的过程管理和过程仿真,国内已有多方面的研究<sup>[2,3]</sup>。文[3]从并行工程运行模式出发,在 Petri 网基础上引入面向对象技术,针对并行工程的并发特性,讨论了基于并行工程的产品开发过程的面向对象 Petri 网<sup>[4~6]</sup>模型,很好地继承了面向对象方法的封装、分类、继承等优点,同时又具有 Petri 网的图形易于描述系统并行性、不确定性等特征,

为并行工程过程管理的仿真、调度和监控提供了统一模型。文[7]对并行设计基于约束的一致性检测算法进行了研究,建立了 Petri 网模型。分析了模型的性质,给出了适用于并行设计冲突检测的算法,并且通过具体应用实例进行了验证,说明了方案的可行性。本文着重讨论 AgentGroup 内的一致性处理问题,给出可信度优先、权威性优先、相邻影响、彼此影响等四种协商策略并给出相应的决策公式。对四种策略进行了对比,指出了各自的优势和适应情形,并给出了 AgentGroup 最终决策的方法。

## 2 并行工程环境下知识管理环境的体系结构

从逻辑上来看,并行工程中知识管理环境的体系结构可以分为 3 层<sup>[1]</sup>,如图 1 所示。

(1)对象层:包括各种具体的知识,它们分别可以以不同的形式存在,如:文档、数据库、过程实例、具体的领域知识等。在众多的知识表示方法中,领域知识的表示通常是基于规则的。对于领域问题的处理以及领域故障的诊断,具体的领域知识发挥着重要的作用,在整个并行工程中占据着重要的位置。本文将主要讨论对象层的领域知识的管理和验证问题。

(2)描述层:为对象层的知识提供描述的框架,这层的知识表示通常是基于本体<sup>[8~10]</sup>的。企业本体描述了知识的背景,包括组织、过程、活动等的表达;信息本体描述的知识的元模型,包括知识存在的形式、地点、可靠性等方面的内容;领域本体描述了对对象层领域知识的具体内容,包括关键字、词组等内容相关的方面,为知识的自动获取提供可能。

(3)应用层:为用户提供面向活动的自动的知识支持机

<sup>\*</sup> 本课题得到国家自然科学基金(No. 60274063 和 No. 10171055)和山东省中青年科学家科研奖励基金(02BS069)的资助。

制。知识支持的任务通常由多个智能 Agent 来协同完成,并集成在 workflow 系统的任务分配工具中。

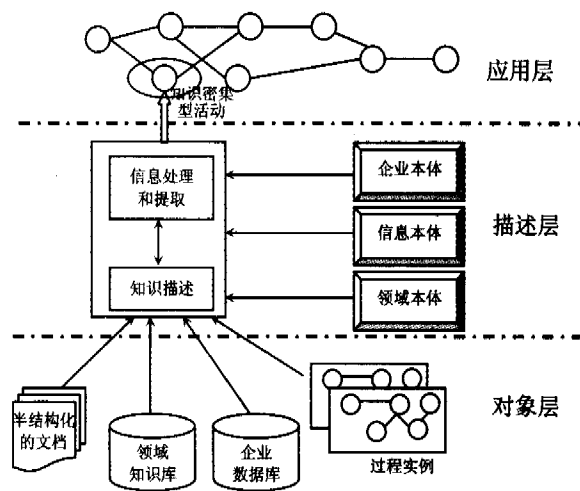


图1 面向并行工程的知识管理的三层结构

知识支持的任务通常是由多个智能 Agent 分布式处理,然后通过协同等方式完成并做出决策。在决策过程中,对于同一个问题,不同的 Agent 常常会给出不一致的结论,我们把这种现象称为多 Agent 之间决策的不一致性。

在并行工程中,把多 Agent 对同一个问题的各种结论综合成一个代表整个协作系统最终结论的方法,是多 Agent 协作系统、分布式知识系统和分布式人工智能等领域研究的一个十分关键的课题。这个过程,通常包括下列步骤:

- (1)对参与协作的每个 Agent 的结论用领域知识进行评价,排除错误结论。
- (2)定义 Agent 决策间的相似度,根据 Agent 结论间的一致性程度进行分组得到 AgentGroup,每个 AgentGroup 内各个 Agent 间的结论相差最近,即形成派别组。
- (3)分别对每个 AgentGroup 内的各个 Agent 进行协商,处理它们的矛盾,做出一致性决策。
- (4)对不同的 AgentGroup 之间进行组间协调,给出最终的决策结果。

在并行工程中,这四个过程构成了智能决策不可分割的整体。我们在文[11]中讨论了基于 Petri 网的领域知识库建模与校验方法,可以很好地辅助于单个 Agent 决策结论正确性判断。根据不同的划分策略,可以得到不同的派别组划分算法。AgentGroup 内的一致性处理和 AgentGroup 间的组间协调是两个关键性的问题。

### 3 AgentGroup 内协商机制

#### 3.1 与决策相关的 Agent 描述

在整个并行工程中,不同的 Agent 具有不同的分工,所以不同 Agent 的功能描述不尽相同,同时不同的 Agent 之间需要通讯、交互、协作等,因此每个 Agent 都有若干个属性描述。

在本文中,我们只给出与决策相关的几个属性描述,下面是 Agent 的本体(Ontology)定义:

Defontology Agent:决策 Agent

- 属性:所属组别:
  - 类型 字符串;
  - 值域 AgentGroup 名称
- 属性:权威性

- :类型 实数
- :值域 介于 0 和 1 之间的实数
- 属性:断言结论
  - :类型 布尔值
  - :值域 T 或 F
- 属性:断言可信度
  - :类型 实数
  - :值域 介于 0 和 1 之间的实数

上述本体给出了所有参与智能决策的 Agent 的公共属性。其中,“所属组别”说明该 Agent 所属于的 AgentGroup,其取值是由派别组划分算法所决定的。本文采用最简单的划分算法,同一个 AgentGroup 各个 Agent 对同一断言具有相同的取值,并且断言可信度值全部落在相同的某个区间。权威性用于说明该 Agent 在整个智能决策中的可信度,取值越大,权威性越大。“断言结论”是指该 Agent 对某个问题所做出的真假值判断。断言可信度用于说明该 Agent 所做出的断言的可靠性。

表 1 给出了同一个 AgentGroup 内 4 个 Agent 对某个问题决策时的状态。

表 1 AgentGroup 内不同 Agent 状态描述

所属组别	Agent 名称	权威性	断言结论	断言可信度
AgentGroup1	Agent1	0.5	T	0.9
AgentGroup1	Agent2	0.8	T	0.7
AgentGroup1	Agent3	0.8	T	0.3
AgentGroup1	Agent4	0.9	T	0.6
AgentGroup1	Agent5	0.6	T	0.4

由表 1 可以看出,虽然同一 AgentGroup 内的各个 Agent 对同一问题的断言结论相同,但是不同 Agent 具有不同的权威性和不同的断言可信度,所以对系统仍然很难给出最后一致的决策。

#### 3.2 AgentGroup 内协商策略

为了便于问题的说明以及公式的描述,对于第  $i$  个 Agent,我们用  $W_i$  表示它的权威性, $AS_i$  表示断言结论, $GF_i$  表示断言可信度。

由前面的分析可知:两个 Agent 属于同一 AgentGroup 当且仅当:

- (1) $AS_i = AS_j$ ;
- (2) $\alpha \leq CF_i \leq \beta$  且  $\alpha \leq CF_j \leq \beta$ ,其中  $[\alpha, \beta]$  为根据领域知识指定的可信度区间。

不失一般性,假定组 AgentGroup 内有  $n$  个 Agent,即  $AgentGroup = \{Agent_i | 1 \leq i \leq n\}$ ,且  $Agent_i$  的权威性、断言结论和断言可信度分别用  $W_i$ ,  $AS_i$  和  $CF_i$  表示。

下面我们分别给出用于 AgentGroup 内不一致性处理的策略的几种不同策略。

(1)可信度优先机制 AgentGroup 内各个 Agent 的  $W_i$  比较接近时,找出  $CF_i$  最大的  $Agent_i$ ,调整 AgentGroup 内其它 Agent 的可信度,使之接近于  $CF_i$ 。具体调整方法为:

$$CF'_j = \frac{(W_i \times CF_i + W_j \times CF_j)}{W_i + W_j}, \text{其中 } CF'_j \text{ 为调整后的 } Agent_j \text{ 的可信度, } 1 \leq j \leq n (i \neq j)。$$

可信度优先策略是对 AgentGroup 之断言(即组中任一 Agent 对同一问题之断言)很有可能存在的较为乐观的观点。这种方法在很多时候有较广泛的应用,如在故障诊断时,为了尽可能地减少故障发生的可能,尽管只有少数 Agent 比

较肯定地坚持可能会发生故障,也往往要进行预防和控制。

例如:按照可信度优先策略,Agent1 作为调整的基准。根据  $CF'_i$  的计算公式,调整后的各个 Agent 的决策情况如表 2 所示,因为 AgentGroup 和断言的取值都不会改变,在下面的举例中,我们不再明确给出这两个属性。

表 2 可信度优先策略调整后 AgentGroup 内各 Agent 的状态描述

Agent 名称	权威性	断言可信度	调整后的断言可信度
Agent1	0.5	0.9	0.9
Agent2	0.8	0.7	0.8
Agent3	0.8	0.3	0.5
Agent4	0.9	0.6	0.9
Agent5	0.6	0.4	0.6

(2)权威性优先机制 在 AgentGroup 中找出权威性最高(即  $W_i$  最大)的  $Agent_i$ ,调整 AgentGroup 内其它 Agent 的可信度,以  $CF_i$  为基准,调整其他 Agent 的断言可信度。具体调整方法为:

$$CF'_j = \frac{(W_i \times CF_i + W_j \times CF_j)}{W_i + W_j}, \text{其中 } CF'_j \text{ 为调整后的 } Agent_j \text{ 的可信度, } 1 \leq j \leq n (i \neq j).$$

权威性优先策略是根据  $W_i$  的值确定了 Agent 之间的优先级别。 $W_i$  值越大,  $Agent_i$  的决策权就越高,其他 Agent 必须向其看齐。

根据权威性优先策略,Agent4 作为调整的基准。表 1 中的各个 Agent 调整后的决策状态如表 3 所示。

表 3 权威性优先策略调整后 AgentGroup 内各 Agent 的状态描述

Agent 名称	权威性	断言可信度	调整后的断言可信度
Agent1	0.5	0.9	0.7
Agent2	0.8	0.7	0.6
Agent3	0.8	0.3	0.4
Agent4	0.9	0.6	0.6
Agent5	0.6	0.4	0.5

虽然可信度优先策略和权威性优先策略两者的调整公式相同,但是由于选择的 Agent 基准不同,由表 2 和表 3 可以看出调整后的结果也不尽相同。

前两种策略均是以某一个 Agent 为核心进行调整的,而在实际工程中,Agent 之间通常是相互影响、相互协作等,AgentGroup 内决策不一致性的调整更多的应该关注 Agent 与 Agent 之间的相互影响。

(3)相邻影响机制 同一个 AgentGroup 内部的各个 Agent 的权威性相差一般不会十分悬殊,否则将会影响 Agent 在协作过程中的参与性。从人类的思考过程来看,人们对于问题做出判断的结果也往往最容易收到近似结果的影响。根据这个特点,我们来调整各个 Agent 的决策。

首先将 AgentGroup 内部的各个 Agent 按照断言可信度进行排序(升序或降序均可),不妨设按降序排序后的结果为:  $Agent_{i_1}, Agent_{i_2}, \dots, Agent_{i_m}$ , 对应的断言可信度  $CF_{i_1}, CF_{i_2}, \dots, CF_{i_m}$ , 权威性为  $W_{i_1}, W_{i_2}, \dots, W_{i_m}$ 。

根据排序后的结果,  $Agent_k$  最容易受  $Agent_{i(k-1)}$  和  $Agent_{i(k+1)}$  的影响 ( $2 \leq k \leq m-1$ )。由于  $CF_{i_1}, CF_{i_2}, \dots, CF_{i_m}$  是按照降序排列的,因此  $Agent_k$  最容易受  $Agent_{i(k-1)}$  的正面影响(可信度增加),受到  $Agent_{i(k+1)}$  的负面影响(可信度降低)。当  $k=1$  时,  $Agent_{i_1}$  只受  $Agent_{i_2}$  的负面影响;当  $k=m$  时,

$Agent_{i_m}$  只受  $Agent_{i(m-1)}$  的正面影响。

具体的调整方法为:

$$CF'_k = \frac{W_{i(k-1)} \times CF_{i(k-1)} + W_k \times CF_k + W_{i(k+1)} \times CF_{i(k+1)}}{W_{i(k-1)} + W_k + W_{i(k+1)}}, \text{其中}$$

中  $CF'_k$  为调整后的可信度 ( $2 \leq k \leq m-1$ )。

$$\text{当 } k=1 \text{ 时, } CF'_{i_1} = \frac{W_{i_1} \times CF_{i_1} + W_{i_2} \times CF_{i_2}}{W_{i_1} + W_{i_2}}; \text{当 } k=m \text{ 时,}$$

$$CF'_{i_m} = \frac{W_{i(m-1)} \times CF_{i(m-1)} + W_m \times CF_m}{W_{i(m-1)} + W_m}.$$

根据相邻影响策略,表 1 中的各个 Agent 调整后的决策状态如表 4 所示。

表 4 相邻影响策略调整后 AgentGroup 内各 Agent 的状态描述

Agent 名称	权威性	断言可信度	调整后的断言可信度
Agent1	0.5	0.9	0.8
Agent2	0.8	0.7	0.7
Agent3	0.8	0.3	0.3
Agent4	0.9	0.6	0.6
Agent5	0.6	0.4	0.4

(4)相互影响机制 根据相互影响策略,人类专家易于接受相近观点的影响,但这并不意味着相差较大的观点之间就不发生影响。实际上,同一个 AgentGroup 中每个 Agent 都对其它 Agent 施加影响。通常情况下,权威性大的 Agent 较难接受比之权威小的 Agent 的影响;反之权威性小的 Agent 较易接受权威性比它大的 Agent 的影响。相互影响策略强调 AgentGroup 中每个协作伙伴的之间的协调作用。

具体的调整处理策略为:

$$CF'_i = W_i \times \frac{\sum_{1 \leq j \leq n, j \neq i} W_j \times CF'_j}{\sum_{1 \leq j \leq n, j \neq i} W_j}, \text{其中 } (1 \leq i \leq n).$$

根据相互影响策略,表 1 中的各个 Agent 调整后的决策状态如表 5 所示。

表 5 相互影响策略调整后 AgentGroup 中各 Agent 的状态描述

Agent 名称	权威性	断言可信度	调整后的断言可信度
Agent1	0.5	0.9	0.3
Agent2	0.8	0.7	0.4
Agent3	0.8	0.3	0.5
Agent4	0.9	0.6	0.6
Agent5	0.6	0.4	0.3

### 3.3 四种协商机制的效果比较

前面我们给出了 Agent 之间的四种协商机制,下面我们比较这四种不同机制的调整效果,最终结果如图 2 所示。

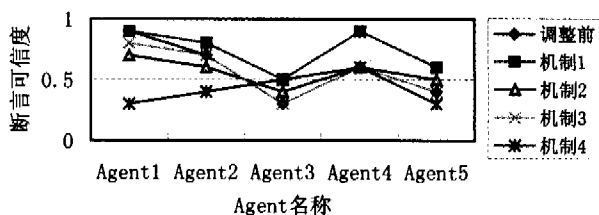


图 2 四种协商机制效果比较

从图 2,我们可以看出:

(1)机制 1(可信度优先机制)调整后,所有 Agent 的可信度都会提高。此机制的原则是“宁可信其有不可信其无”,多用于为减少工程风险等情形而提供决策的可信度。

(2)机制 2(权威性优先机制)可以很好地协调各个 Agent 之间的差异,可信度高的降低,可信度低的提高,使得决策结果基本趋于一致。很显然,这种机制有利于 Agent 之间的协调,可以很好地解决 Agent 之间的矛盾。

(3)机制 3(相邻影响机制)调整的效果不是很明显,基本上保持了调整前的状态。该机制可以很好地保持系统的稳定性。

(4)机制 4(彼此影响机制)与权威性优先机制类似,但是由于 Agent 的权威性  $W_i$  参与了最终的决策,使得决策的结

果较权威性优先策略有所降低。如果去掉 Agent 的各自权威性参数,会提高每个 Agent 断言的可信度。

### 4 基于协商的 AgentGroup 决策

AgentGroup 内各 Agent 的断言可信度值经过上述四种影响策略处理后,都朝着代表本组的某种统一看法的方向发生了变化。再将修改了的各个可信度值适当组合,形成代表本组观点的统一可信度值,这一过程称为 AgentGroup 决策,图 3 给出了决策过程的描述。

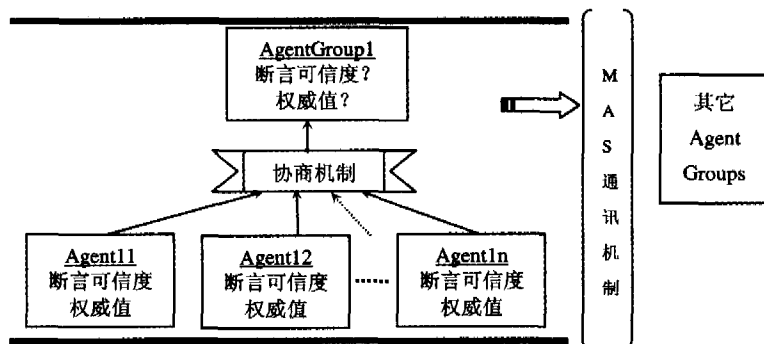


图 3 AgentGroup 协商决策机制

我们采用加权平均方法组合这些可信度值。设经过某种策略处理后,各个 Agent 的断言可信度的取值分别为:  $CF'_1, CF'_2, \dots, CF'_n$ , 组 AgentGroup 决策的可信度为:  $CF = \frac{\sum_{i=1}^n W_i \times CF'_i}{\sum_{i=1}^n W_i}$ , 其中  $(1 \leq i \leq n)$ 。

下面我们给出四种调整策略所决定的 AgentGroup 最终决策的情况,如表 6 所示。

表 6 四种策略下的 AgentGroup 决策情况

	直接决策	策略 1	策略 2	策略 3	策略 4
AgentGroup 断言可信度	0.6	0.7	0.6	0.6	0.4

根据表 6,我们可以给出四种结果的比较图,如图 4 所示。

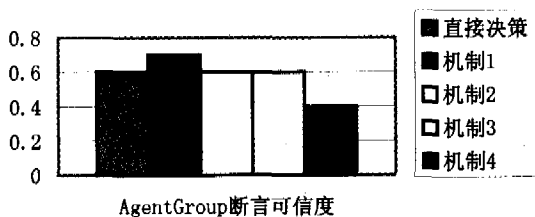


图 4 四种策略决定的组内决策情况比较

由图 3 组内决策的最终结果可以看出,4 种决策方法所得到的组内决策结果基本上一致。策略 4 下的 AgentGroup 的决策降低了可信度,是由于 Agent 的权威性  $W_i$  参与了最终的决策  $W_i$ ,如果去掉参数,其决策的结果接近于其他情况。

### 5 问题与讨论

并行工程中,多 Agent 决策结果的不一致性会影响整个并行工程的决策结果。本文给出了可信度优先、权威性优先、相邻影响、彼此影响等四种协商机制用于处理 Agent 之间的矛盾,并给出相应的决策公式,并对它们进行了量化比较和分析。

下面的一些问题有待于继续讨论和研究:

(1)四种协商机制的可靠性。我们没法从理论上证明各自的正确性,只能分析它们的调整效果和各自的适应情形。在实践中,需要根据决策的实际需要,选择相应的调整策略,并通过实验检测其可靠性。

(2)最高权威性和可信度都不唯一的情况。所给出的例子中,最高权威性和可信度都不唯一的情况都是唯一的。假如不唯一,根据本文的协商策略 Agent 之间会产生争吵,所以对它们的优先级选择需要提高相应的控制策略。

(3)本文只讨论了组内不一致性的协商问题。关于 Agent 的分组算法,以及 AgentGroup 与 AgentGroup 之间的协调机制等问题还需要进一步的研究。

### 参考文献

- 熊光楞,王昕. 面向并行工程的知识管理研究. 高技术通讯, 2002,6: 34~42
- 许维胜,吴启迪. 并行工程过程研究. 同济大学学报,2001,29(2): 185~190
- 李秀,姜澄宇,王宁生. 基于并行工程的产品开发过程建模. 机械设计与制造工程,2001,29(1):34~37
- 袁崇义. Petri 网原理与应用. 北京:电子工业出版社,1998
- 曾庆田,吴哲辉. Petri 网的进程网系统. 计算机学报[J],2002,25(12):1308~1315
- 曾庆田,吴哲辉. 基于库所指标的 Petri 网分解方法. 计算机科学[J],2002,29(4):15~17
- 陈梦,王新平,宿英新. 并行设计中冲突的一致性检测算法. 北京科技大学学报,2002,24(4):479~482
- McGuinness D L, Fikes R, Rice J, Wilder S. An Environment for Merging and Testing Large Ontologies. In: Proc. of the Seventh Intl. Conf. on Principles of Knowledge Representation and Reasoning (KR2000). Breckenridge, Colorado, USA
- Noy N F, Musen M A. SMART: Automated Support for Ontology Merging and Alignment. In: Proc. of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management, Banff, Canada, July 1999
- Sure Y, Staab S, Angele J, Wenke D, Maedche A. OntoEdit: Guiding ontology development by methodology and inferencing. Prestigious Applications of Intelligent Systems (PAIS), ECAI 2002, Lyon, France, 2002
- 段华,曾庆田,贺国平. 面向并行工程的知识库校验模型与分析方法研究. 系统工程理论与实践,2004,24(3): 79~86