

# 一类经验性知识解释器的设计与实现

袁晓征,毛宁,陈庆新,陈晨

(广东工业大学机电工程学院,广东广州 510090)

**摘要:**以“问题-原因-解决方案”来表达的这类经验性知识,常用“大概”或“很可能”等感性词语修饰,具有模糊性和不确定性的特点,且解决方案与问题和原因高度关联,该特点适合用产生式规则来表示。为便于计算机处理,产生式规则往往由计算机能识别的某种编码组成,而这类编码用户无法解读,需将其“翻译”成人类易于理解的自然语言。文中探讨了如何表示这类经验性知识,提出了一种模板解释方法,采用 MVC(Model-View-Controller, MVC)架构、基于 VS2005 + SQL Server2000 平台实现了该解释器。该解释器已在某模具厂的改模知识归纳系统中应用且取得了良好的效果。

**关键词:**经验性知识;模板;解释器;MVC架构

**中图分类号:**TP311

**文献标识码:**A

**文章编号:**1673-629X(2009)10-0242-05

## Design and Realization of a Type of Interpreter for Experiential Knowledge

YUAN Xiao-zheng, MAO Ning, CHEN Qing-xin, CHEN Chen

(Department of Mechanical Engineering, Guangdong University of Technology, Guangzhou 510090, China)

**Abstract:** The experiential knowledge expressed by the way of “problems—causes—solutions” takes on the characteristics of fuzziness and uncertainty that are caused by the use of the perceptual words such as “probably”, “likely” and so on. The solution of it is highly related to the causes, which suits the production rule. For the sake of processing in computer, production rule is usually constituted of certain codes recognizable by the computer but not for users, so these rules should be “translated” into a language understandable for human beings. Aiming to this goal, this article first discusses the way how to express the experiential knowledge, and then puts forward a interpreter on the basis of the template, at last points out how to realize the interpreter on the basis of MVC (Model-View-Controller) framework and the platform of VS2005 + SQL server2000. This interpreter has now been applied in a die shop and achieved a satisfactory result.

**Key words:** experiential knowledge; template; interpreter; MVC

## 0 引言

以“问题-原因-解决方案”来表达的这类经验性知识很多,例如,在医疗诊断过程中,病人向医生描述病情,医生通过病人的描述和身体表现出来的症状进行原因分析而得出治疗方案<sup>[1]</sup>;在故障诊断中,专家对故障症状进行观察和分析,推断故障位置和产生的原因,从而得出排除故障的解决方案<sup>[2,3]</sup>;在模具制造领域,修(改)模是一个必不可少的环节,修(改)模工程师

通过问题的表现形式及发生的位置进行原因分析,从而得出修(改)模工艺方案<sup>[4]</sup>等等,这类知识属于隐性知识<sup>[5,6]</sup>。

这类经验性知识有其自身的特点:首先,该类知识具有模糊性故无法加以明确表述且多为感性知识;其次,该类知识与具体问题高度关联;最后,这类知识主要通过实践进行积累和完善从而被知识所属领域的专家所掌握<sup>[7]</sup>。传统上,这类经验性知识的传承是通过师傅带徒弟的方式进行,师傅针对具体问题,发表自己的观点和意见,徒弟通过不断观摩和个人的不断实践,并经历各种失败与成功的方式逐渐完善的。

这类经验性知识往往只被少数专家所掌握,因此将随着专家的流失而流失,给需要拥有这类知识的企业带来不可估量的损失,正是认识到该类经验性知识的重要性,一些企业认识到可以通过计算机将这些经验性知识管理起来,以实现知识的重用、共享和积累。

收稿日期:2009-02-12;修回日期:2009-05-09

基金项目:国家自然科学基金项目(50775042);国家 863/CIMS 探索项目(2007AA04Z1A8);国家支撑计划项目(2006BAF01A43)

作者简介:袁晓征(1978-),男,硕士研究生,研究方向为制造业信息化、知识工程、知识管理;毛宁,教授,硕士生导师,研究方向为制造业信息化、知识工程、知识管理、工业工程、CIMS 下的质量控制等;陈庆新,教授,博士生导师,研究方向为敏捷制造、并行工程等。

这类经验性知识往往是专家根据自己的感觉来表达的,这种表达随意且不规范,计算机无法识别。所以首先要解决的就是这类知识如何表达,其次就是如何将其“翻译”成人类自然语言,为人们所应用。文中分析了这类经验性知识的表示方法,然后提出基于模板解释方法,并设计了七种模板,实现了对这类经验性知识的解释。最后采用 MVC 架构、基于 VS2005(C#) + SQL Server2000 平台实现了该解释器。

## 1 以“问题-原因-解决方案”来表达的经验性知识的表示方法

以“问题-原因-解决方案”这种方式来表达的经验性知识,由于解决方案与问题和原因高度关联,所以适合用产生式规则来表示这类知识。

因为这类经验性知识中产生问题的原因具有模糊性和不确定性,可以给原因加入隶属度因子来表达这种模糊不确定性,隶属度因子的取值在 0 到 1 之间,越接近于 1,表示原因与问题的确定性程度越高。

这类经验性知识中针对问题的解决方案可能有多个,到底采用那一个解决方案,我们认为可以通过置信度来体现。置信度的取值在 0 到 1 之间,置信度的值越高,解决方案的可信度越好,即采用该方案能够解决问题的可能性越高。

这样在传统产生式规则的基础上加入了模糊表示的能力<sup>[8]</sup>,即应用“IF(问题,原因 with 隶属度) THEN (解决方案 with 置信度)”来表示这类经验性知识。

笔者认为问题一般可由问题类型、问题位置、问题程度三个主要特征组成;原因可由原因类型和原因程度两个主要特征组成;解决方案可由操作类型和操作对象两个主要特征组成。如:“产品分型面有批锋,可能是行位过切加工造成,请检查确定后烧焊前模”,其中批锋可以看作是问题的类型,分型面可以看作是问题的位置,过切加工可以看作是原因的类型,而烧焊和前模可分别看作操作的类型和对象。

这七个主要特征可以有各自的特征值,它的具体形式根据这类经验性知识的领域不同会有差异。例如,在模具的修(改)模知识中,问题类型特征的特征值可以有批锋、多纹等;问题位置特征的特征值可以有分型面、产品表面等。为了便于计算机识别,需要给特征和特征值进行编码,表示成“特征.特征值编码”的形式,用编码 PT 表示问题类型,编码 PP 表示问题位置,编码 PS 表示问题程度,编码 RT 表示原因类型,编码 RS 表示原因程度,编码 OO 表示操作对象,编码 OT 表示操作类型;特征值可用数字编码来表示,所以,以“问题-原因-解决方案”来表达的经验性知识的产生

式规则为:

IF([(问题位置.特征值) and (问题类型.特征值) and (问题程度.特征值)]

AND [(原因类型.特征值) and (原因程度.特征值) with  $\alpha$ ],

AND /OR[(问题位置.特征值) and (问题类型.特征值) and (问题程度.特征值)])

AND [(原因类型.特征值) and (原因程度.特征值) with  $\alpha$ ],

AND /OR [……]

THEN [(操作类型.特征值) and (操作对象.特征值) with  $\beta$ ]

AND /OR [(操作类型.特征值) and (操作对象.特征值) with  $\beta$ ],

AND /OR […with  $\beta$ ] (1)

“THEN”之前(称为前件)主要由问题类型(PT)、问题位置(PP)、问题程度(PS)、原因类型(RT)、原因程度(RS)5 个部分组成;“THEN”之后(称为后件)主要由操作对象(OO)和操作类型(OT)2 个部分组成; $\alpha$  表示隶属度; $\beta$  表示置信度。

例如:在模具制造过程中,存在这样的修改模知识:“产品表面有流痕,可能是加工时增加材料过多,检查确定后可能采用烧焊分型面去除流痕”,按照上述(1)的表示方法为:

IF (<PT.1.1.1.1> AND <PP.1.1> AND <RT.2.1> AND <RS.1> with 0.6)

THEN (<OO.2.1.5> AND <OT.1.1>) with 0.6

其中:编码 PT.1.1.1.1 表示流痕;编码 PP.1.1 表示产品表面;编码 RT.2.1 表示增加材料;编码 RS.1 表示过多;编码 OO.2.1.5 表示分型面;编码 OT.1.1 表示烧焊。隶属度  $\alpha=0.6$ ;置信度  $\beta=0.6$

在计算机内部存储的格式为:

<PT.1.1.1.1>^<PP.1.1>^<RT.2.1>^<RS.1> with 0.6 → <OO.2.1.5>^<OT.1.1> with 0.6

## 2 基于模板的解释器设计

由知识在计算机内部的存储形式可知,如果直接把知识的存储形式展示给用户,用户或知识工程师是无法解读这些“知识”含义的。为了让用户能够正确理解上述产生式规则表达的含义,不但需要将类似于“PT.1.1.1.1”这样的编码“翻译”成其对应的特征值,还要符合规则的逻辑结构,以人类能理解的自然语言陈述给用户。由于特征值属于离散值,需要添加相应的修饰文本才能组合成形式语言,这一特点适合采用模板的方式来描述这类经验性知识。

由于这类经验性知识常常用“大概”或“很可能”等感性词语来修饰,具有模糊性的特点,解释还需要处理模糊性修饰词。针对这类经验性知识的特点,为了方便起见,分别用 RW1 表示隶属度  $\alpha$  的修饰词, RW2 表示置信度  $\beta$  的修饰词,如表 1 所示。

表 1 隶属度、置信度的关键词及对应值

$\alpha$	隶属度( $\alpha$ )关键词与对应值		
RW1		很可能	可能
值	1	0.8	0.6
$\beta$	置信度( $\beta$ )关键词与对应值		
RW2	必须采用	最好采用	可以采用
值	1	0.8	0.6

若采用上述式(1)的知识表达方式,可用如下修饰词(黑体部分):

1) [(问题位置.特征值) and (问题类型.特征值) and (问题程度.特征值)] and [(原因类型.特征值) and (原因程度.特征值) with  $\alpha$ ]: 在(问题位置.特征值)有(问题类型.特征值)(问题程度.特征值)问题,原因 **RW1** 是(原因类型.特征值)(原因程度.特征值);

2) AND /OR: 而且还 / 或者;

3) [(操作类型.特征值) and (操作对象.特征值) with  $\beta$ ]: **RW2**(操作类型.特征值)(操作对象.特征值)解决。 (2)

但这类知识在表达过程中往往并不完整。有时只是简单说明问题类型,就需要解决方案。观察这类知识的表示特点,其后件用来说明对问题的处理方法,也就是问题的求解答案,是知识的核心部分,采用了 2 个特征:操作类型和操作对象,2 个特征缺少任何一个都不能描述处理方法,两者都不能为空。而前件中的 5 个特征是用来描述问题及其原因的,其中 PS 用来说明问题 PT 的程度,而 RS 用来说明原因 RT 的程度,所以起关键作用的只有 PP、PT 和 RT 三个特征,这三个特征至少会出现一个,即有  $C_3^1 + C_3^2 + C_3^3 = 3 + 3 + 1 = 7$  种可能性,也就是说需要建立 7 个模板。

首先分析 PP、PT 和 RT 都同时取出的情况,由于 PS、RS 只起修饰 PT、RT 的作用,其值不影响模板的解释效果,所以可得模板与上述式(2)等价,将其定义为模板 1(修饰文本用黑体表示,下同):

在(问题位置.特征值)有(问题类型.特征值)(问题程度.特征值)问题,原因 **RW1** 是(原因类型.特征值)(原因程度.特征值),**RW2**(操作类型.特征值)(操作对象.特征值)解决。

例如: IF(<PT.1.1.1.1> AND <PP.1.1> AND <RT.2.3> AND <RS.1> with 0.6) THEN (<OO.2.1.5> AND <OT.2.1.1>) with 0.6

其中,特征编码的含义为:PP.1.1:产品表面;PT.1.1.1.1:流痕;RT.2.3:表面粗糙度;RS.1:过大;OT.2.1.1:雕刻;OO.2.1.5:分型面。

调用模板 1 后的解释结果为:

在产品表面有流痕问题,原因可能是表面粗糙度过大,可以采用雕刻分型面解决。

其次分析在三个关键特征 PP、PT 和 RT 中只取两个的情况,三种组合方法可得出三个模板:

第一种情况,如果在 PP、PT、RT 中取出的是 PP 和 PT,即知道某问题位置上产生什么问题类型,但不知道问题类型是由什么问题原因引起的,所以模板只需要说明某位置上产生了什么问题即可得出模板 2:

由于(问题位置.特征值)有(问题类型.特征值)(问题程度.特征值),**RW2**(操作类型.特征值)(操作对象.特征值)解决。

例如: IF(<PT.1.1.2.2> AND <PP.1.1>) THEN(<OO.2.1.2.2> AND <OT.1.1>) with 0.8

其中,特征编码含义为:PP.1.1:产品表面;PT.1.1.2.2:错纹;OT.1.1:烧焊;OO.2.1.2.2:标识。

调用模板 2 后的解释结果为:由于产品表面有错纹,最好采用烧焊标识解决。

第二种情况,如果在 PP、PT、RT 中取出的是 PP 和 RT,即知道某位置上因为某原因产生了不确定的问题类型,这时往往需要该类知识领域的专家通过原因和产生的位置进行判断从而得出解决方案,模板只需要说明问题产生的位置和问题产生的原因即可,模板 3 如下:

在(问题位置.特征值),**RW1** 由于(原因类型.特征值)(原因程度.特征值),**RW2**(操作类型.特征值)(操作对象.特征值)解决。

例如: IF(<PP.2.1> AND <RT.1.5> AND <RS.2> with 0.6) THEN(<OO.2.1.2.2> AND <OT.2.1.1>) with 0.6

特征值含义为:PP.2.1:流道面;RT.1.5:浇口、流道设计;RS.2:不合理;OT.2.1.1:雕刻;OO.2.1.2.2:标识。

解释结果为:在流道面,可能由于浇口、流道设计不合理,可以采用雕刻标识解决。

第三种情况,如果在 PP、PT、RT 中取出的是 PT 和 RT,即知道某问题原因引起了某问题类型,但不知道问题类型发生在什么位置。在实际应用这类产生式规则的时候,同样需要该领域的专家根据问题原因及问题类型来判断可能发生的位置。在对产生式规则的解释中,同样只需要把原因类型和问题类型描述清楚,

即可得出模板 4:

**RW1** 因为(问题原因.特征值)(原因程度.特征值),所以有(问题类型.特征值)(问题程度.特征值)问题,**RW2**(操作类型.特征值)(操作对象.特征值)解决。

例如:

IF(<PT.1.1.4.2> AND <RT.2.1> AND <RS.3>with 1)THEN(<OO.2.1.5> AND <OT.2.1.3>) with 1

特征值含义为:PT.1.1.4.2:批锋;RT.2.1:材料加工量增加;RS.3:过少;OT.2.1.3:打磨;OO.2.1.5:分型面。

解释结果为:因为材料加工量增加过少,所以有批锋问题,必须采用打磨分型面解决。

最后分析 3 个关键特征中只取出一个有的情况:在这种情况下有 3 种不同的取法,所以有三种不同的可能性,分别讨论如下:

第一种可能,如果在 PP、PT、RT 中取出的仅有 PP,而没有 RT 和 PT,也就是只知道问题出现的位置,而不知道如何用自然语言描述在这个位置上出现的问题类型或根本不知道是什么问题类型,同时也不知道是什么原因引起的,但该类经验性知识的专家知道这种情况如何解决。

模板 5 如下:

由于(问题位置.特征值)问题,**RW2**(操作类型.特征值)(操作对象.特征值)解决。

例如:IF(<PP.1.1.1>)THEN(<OO.2.1.5> AND <OT.2.1.3>) with 0.6

特征值含义为:PP.1.1.1:分型面;OT.2.1.3:打磨;OO.2.1.5:分型面。

解释结果为:由于分型面问题,可以采用打磨分型面解决。

第二种可能,如果在 PP、PT、RT 中取出的仅有 PT,也就是知道是什么问题类型,但问题类型发生的位置和原因不确定,与第一种可能一样,该类经验性知识的专家同样知道这种情况一般应该如何解决。

模板 6 如下:

有(问题类型.特征值)(问题程度.特征值)问题,**RW**(操作类型.特征值)(操作对象.特征值)解决。

例如:IF(<PT.1.1.6>)THEN(<OO.2.1.5> AND <OT.2.1.3>) with 0.8

特征值含义为:PT.1.1.6:气穴(泡);OT.2.1.3:打磨;OO.2.1.5:分型面。

解释结果为:有气穴(泡)问题,最好采用打磨分型面解决。

第三种可能,如果在 PP、PT、RT 中取出的仅有 RT,即只知道问题原因而问题位置和问题类型不确定,可设计模板 7 如下:

**RW1** 由于(原因类型.特征值)(原因程度.特征值)原因,**RW**(操作类型.特征值)(操作对象.特征值)解决。

例如:IF(<RT.1.3.1>with 0.8)THEN(<OO.1.4.1.1> AND <OT.3.1>) with 0.8

特征值含义为:RT.1.3.1:顶出结构设计;OT.3.1:更换;OO.1.4.1.1:顶针。

解释结果为:很可能由于顶出结构设计原因,最好采用更换顶针解决。

对这类产生式规则的解释,需要上述七类模板结合(2)式中的第二条一起使用才能达到更好的解释效果,例如下列产生式规则:

IF(<PT.1.1.1.1>AND<PP.1.1>AND[(<RT.2.2>AND<RS.1>with0.6)OR(<RT.2.3>AND<RS.1>)])

THEN(( [<OO.2.1.5>AND<OT.1.1>] with 0.8) OR([<OO.2.1.2.2>AND <OT.2.1.3>] with 0.6))

其中,特征值含义为:PT.1.1.1.1:流痕;PP.1.1:产品表面;RT.2.2:走刀量;RS.1:过大;RT.2.3:表面粗糙度;OO.2.1.5:分型面;OT.1.1:烧焊;OO.2.1.2.2:标识;OT.2.1.3:打磨。

这条产生式规则所体现的就是不同的原因引起了相同的问题类型,其解决方案也不同。其解释结果为:在产品表面有流痕问题,原因很可能是走刀量过大或者表面粗糙度过大,最好采用烧焊分型面或者可以采用打磨标识解决。

### 3 系统实现

系统的实现采用数据模型-视图-控制器(Model-View-Controller, MVC)三层架构,视图是数据模型的表示,它提供用户交互界面。使用多个包含单显示页面的用户部件,复杂的 Web 页面可以展示来自多个数据源的内容;控制器是控制和协调每个用户跨越多个请求的处理,控制机制以集中的方式进行管理;数据模型是业务流程/状态的处理以及业务规则的制定,业务流程的处理过程对其它层来说是黑箱操作,模型接受视图请求的数据,并返回最终的处理结果。MVC 三层架构组件类型的关系和功能如图 1 所示。

开发平台采用 VS2005(C#),视图用 ASP.net 来实现,控制器采用 C# 实现,模型用 SQL SERVER 2000。解释系统的视图部分主要提供两种方式的操

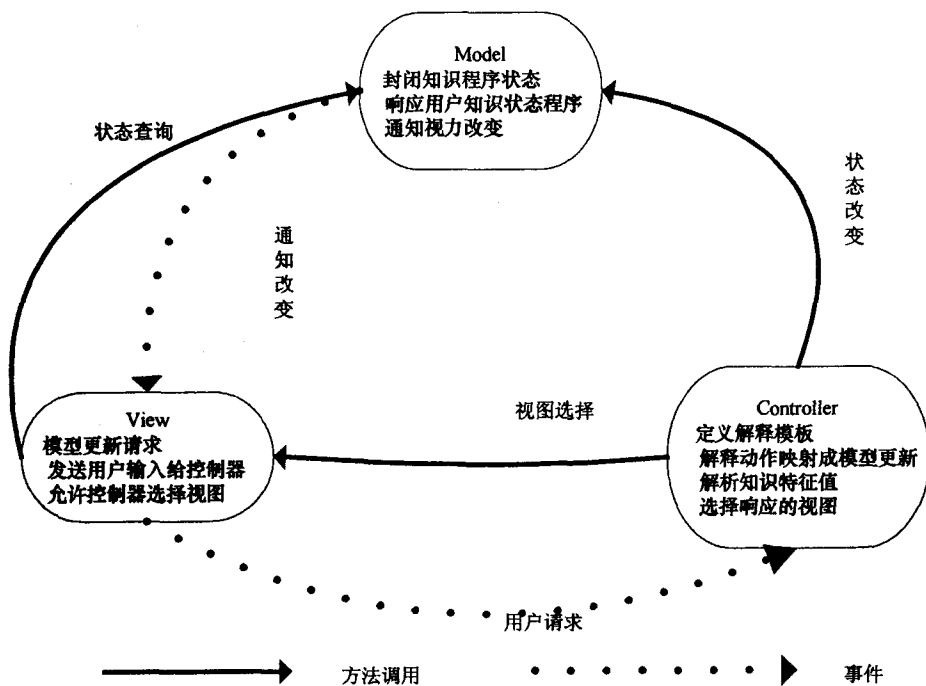


图 1 MVC 架构组件类型的关系和功能

作:用户选择的解释和系统整个知识库的解释。同时,用户可输入改模问题查询问题的解决方案;控制器部分主要定义解释模板和实现选择模板的功能、解释知识特征值和响应视图,为了方便模板的添加与修改,用公共函数实现模板的定义与选择;模型主要实现知识的查询和知识特征值的解析,知识的查询采用存储过程来实现,特征值的解析应用用户自定义函数来实现以方便调用。系统部分解释结果如图 2 所示。

## 解释结果管理:

编号:67

规则知识:if(PT.1.1.1.1)and (PP.1.1)and (RT.2.3)with0.6)then(00.2.1.5)and(OT.2.1.1)with0.6

问题位置:PP.1.1:产吕表面;问题类型:PT.1.1.1:流痕;问题程度:;原因类型:RT.2.3:表面粗糙度;

原因程度:RS.1:过大;操作类型:OT.2.1.1:雕刻;操作对象:OO.2.1.5:分型面

解释结果:在产品表面有流痕问题,原因可能是表面粗糙度过大,可以采用雕刻分型面解决。

图 2 解释结果

## 4 结束语

文中所设计开发的解释器已成功应用于改模知识归纳系统,并在揭阳某模具厂成功使用,而且在此基础上进一步研究基于问题特征的检索技术,通过计算问题相似度向用户提供相似问题解决方案,从而在此基础上向用户推荐问题的最佳解决方案,为提高改模效率,降低改模成本提供进一步的支持和有益的帮助。

## 参考文献:

- [1] 梁嘉骅. 医疗诊断专家系统的新思想与新方法[J]. 系统工程学报,1999,14

(1):83-89.

- [2] 关惠玲. 设备故障诊断专家系统原理及实践[M]. 北京:机械工业出版社,2000.
- [3] Daoliang, Fu Z, Duan Yanqing. Fish - Expert: a Web - based expert system for fish disease diagnosis[J]. Expert Systems with Application, 2002, 23(3): 121 - 130.
- [4] 陈 晨, 毛 宁, 陈庆新. 注塑模改模工艺表达及知识建模[J]. 模具制造, 2008(5): 13 - 15.
- [5] Nanaka, Takeuchi H. The knowledge creating company: How Japanese companies create the dynamics of innovation[M]. Oxford City: Oxford University Press, 1995.
- [6] 陈梓展, 陈 晨, 毛 宁, 等. 基于企业知识形成过程的会议系统研究[J]. 计算机技术与发展, 2007, 17(11): 187 - 189.
- [7] Bhatt G D. Organizing in the Knowledge Development Cycle [J]. Journal of Knowledge Management, 2000, 4(1): 15 - 26.
- [8] 吴 钊, 项经猛. 一种专家系统模糊解释器的设计与实现 [J]. 计算机工程, 2005, 31(3): 155 - 157.

(上接第 241 页)

IEEE International Symposium. Munich: [s. n.], 2007: 560 - 574.

- [3] Steel C, Nagappan R, Lai R. Core security patterns: best practices and strategies for J2EE, web services, and identity management[M]. NJ: Prentice hall, 2006.
- [4] 徐永祥. 统一用户管理系统的设计[J]. 计算机工程, 2003, 29(8): 120 - 123.
- [5] 孙宏波, 颜 波, 黄必清, 等. 面向复杂信息系统的用户管

理策略[J]. 计算机集成制造系统, 2004, 10(12): 105 - 113.

- [6] 宋晓婵, 刘连忠. 异构操作系统环境的统一用户管理的研究[J]. 计算机技术与发展, 2008, 18(4): 150 - 152.
- [7] 黄叶珏, 陈 勤. Web 网站统一口令认证系统的设计与实现[J]. 计算机技术与发展, 2007, 17(6): 169 - 171.
- [8] 袁晓舟, 范菲雅, 马登哲. 应用服务提供商平台统一用户管理模型的研究与实现[J]. 计算机集成制造系统, 2006, 12(12): 1972 - 1977.