

通用型 CAPP 专家系统中 知识表示和推理机制的研究*

杨莉 何志均

(浙江大学 CAD&CG 国家重点实验室, 杭州 310027)

(浙江大学计算机系, 杭州 310027)

摘要 CAPP 作为 CAD 与 CAM 的中间桥梁, 是 CIMS 必不可少的重要环节. 本文针对开发通用性 CAPP 专家系统过程中, 产品零件表示和工艺规程设计这两个关键问题, 根据事物之间的相似性原理, 提出了面向对象的产品零件和工艺规程的知识表示方法, 以满足 CAPP 系统对产品零件信息表示的完整性、通用性、逻辑可分解性、动态性和相容性等要求, 并在此基础上提出了 CAPP 专家系统的混合推理机制, 包括单继承、多继承、非单调继承、类比推理等.

关键词 计算机辅助工艺过程设计, 知识表示, 推理机制, 专家系统.

为了提高产品质量和生产率, 以面对越来越激烈的市场竞争, 制造业越来越重视计算机集成制造系统(CIMS)的应用. 计算机辅助工艺过程设计(CAPP)作为 CAD 和 CAM 的中间桥梁, 是 CIMS 必不可少的重要环节. CAPP 通过改变工艺设计过程中传统的手工劳动性质, 从而大幅度地提高工艺设计效率和工艺规程规划, 加快生产准备工作的进度. 目前, 国际上发展比较成熟的 CAPP 系统主要采用派生式方法, 在这样的系统中只能利用一般的、不变的知识, 只适用于常规的普通零件, 不能为新零件设计工艺规划过程且难以优化和移植. 随着人工智能和专家系统技术的发展和推广应用, 导致了 CAPP 专家系统的涌现. 专家系统技术的引入能够提高 CAPP 系统适应环境变化的能力, 增强它的柔性, 它不仅能模拟有经验工艺人员的工艺规程设计过程, 实现工艺规程的自动设计, 而且能根据不同的应用环境、不同类型的工厂方便地更新和修正 CAPP 专家系统的工艺数据库和知识库, 使之具有一般 CAPP 系统无法比拟的通用性和适应性^[1].

然而, 独立开发一个实用的 CAPP 专家系统需要花费大量的人力和物力, 为了避免重复劳动, 解决的途径是构造工艺过程自动化的开发环境, 即建造 CAPP 专家系统外壳. 但目前现有的专家系统都是针对诊断型领域, 不能体现面向 CAPP 的规划问题的特点. 为了解决上述问题, 我们开展了通用型 CAPP 专家系统开发工具 Z-ESPP 的研究. 本文的第 1 节

* 本文 1994-01-03 收到

本文得到国家自然科学基金和浙江省自然科学基金的资助, 并得到浙江大学 CAD&CG 国家重点实验室开放基金的支持. 作者杨莉, 女, 29 岁, 副教授, 主要研究领域为人工智能, 神经网络, 智能 CAD、CAPP 等. 何志均, 71 岁, 教授, 博士生导师, 主要研究领域为人工智能, 专家系统, CIMS 等.

本文通讯联系人: 杨莉, 北京 100080, 国家智能计算机研究开发中心

着重介绍 Z-ESPP 的知识表示模型 KRM. 第 2 节探讨在 KRM 模型基础上的问题求解机制, 包括单继承、多继承、非单调继承、类比相似推理等. 第 3 节给出了全文的一个总结.

1 KRM 知识表示模型

一般情况下, 专家系统是由 3 大功能模块组成: 知识库、推理机和解释界面. 知识表示作为专家系统的核心, 有什么样的表示机制, 对应于什么样的推理机制和获取机制. 在 Z-ESPP 系统中, 针对产品零件信息表示的完整性、通用性、逻辑可分解性、动态性和相容性等特点, 采用面向对象(OO)的知识表示方法. 用 OO 进行知识表示, 可以把相类似的零件规划过程聚类, 在同一对象类下, 子对象可继承父对象的内容(如特征、规划过程等), 也可处理例外情况(例外情况的处理, 可通过定义局部和全局变量来进行, 也可加标志符来处理)^[2]. 另外, 在我们所定义的对象类中, 可引入异常操作来处理同一类对象下的各个不同部分. 此外, 针对 CAPP 工艺规划知识的错综复杂、难以精确划分成层次结构等特点, 在对象表示中引入方法作为对象的特殊侧面, 来定义该对象类接受到消息时要做的动作. 这里, 方法用逻辑语言中的子句进行描述, 其中子句头与消息选择模式进行匹配. 方法子句可以是非确定的, 即可通过回溯来寻求消息目标的多重解. 因而, 同其它知识表示方法相比, KRM 模型既具有面向对象语言的模块化结构、信息隐藏、数据抽象、知识共享、动态链接等特性, 又具有逻辑语言语义清晰、表达能力强和自动推理的特点. 下面我们详细介绍 KRM 模型.

1.1 KRM 中对象的表示

在 KRM 中, 每一个对象均有一对象标识符, 它可以在对象创建时由用户赋予一个有意义的值, 也可由系统唯一地产生. 另外, 在 KRM 中, 类也定义为一个对象. 对象的定义如下:

```

<obj> ::= object
    <name>,
    <super_obj>,
    <ch_obj>,
    <meta_obj>,
    <obj_vari>,
    <def_vari>,
    <if_needed>,
    <if_added>,
    <normal_method>
endobject.

<super_obj> ::= [],
<super_obj> ::= [<name> | <super_obj>],
<ch_obj> ::= [],
<ch_obj> ::= [<name> | <ch_obj>],
<meta_obj> ::= [],
<meta_obj> ::= [<name> | <meta_obj>],
<def_vari> ::= [],
<def_vari> ::= [<property, value> | <def_vari>],
<obj_vari> ::= [],
<obj_vari> ::= [<obj> | <obj_vari>],

```

```

(obj_vari)::=[(stru_vari)|(obj_vari)].
(stru_vari)::=[],
(stru_vari)::=[(property,value)],
(stru_vari)::=[(property),(property,value)*].
(if_needed)::=[],
(if_needed)::=(method)*.
(if_added)::=[],
(if_added)::=(method)*.
(normal_method)::=[],
(normal_method)::=(method)*.

```

其中 name、property、value 均为原子(字符串),name 表示对象名,property 表示特性名,value 表示特性值,super_obj 指明类/超类关系,ch_obj 实现对象之间的 is_a 关系,meta_obj 表示类/原类关系,def_vari 表示在大多数情况下,该对象特性的取值.以后,如果知识库出现不一致或矛盾时,将对此特性进行修正.method 相当于逻辑程序中的子句,形式为 $P_1 \vee P_2 \vee P_3 \vee \dots \vee P_m: -Q_1 \wedge Q_2 \wedge Q_3 \wedge \dots \wedge Q_n$. (P_i, Q_j 均为项),“*”表示可重复多次.这里值得注意的是:method 在 Z_ESPP 中是做为字符串读入的,故系统首先提供了一个小型解释器,对 method 进行解释后再执行.method 的含义为:当前提 Q_1, Q_2, \dots, Q_n 在 Z_ESPP 的推理过程中被满足(一致化),则顺序执行结论 P_1, P_2, \dots, P_m . 在 KRM 中,对象类、元对象、对象实例均用对象方式进行表示,因而对象还应该具有两方面的功能:①存放实例对象可继承的性质和方法;②创建新对象和实例对象.

1.2 对象创建

对象可以通过知识获取界面静态地描述,也可以在系统运行过程中动态地创建.对象创建包括对象类创建和实例创建.向一个已存在的对象发送创建的消息,则创建该对象的局部特性(包括具体的特性及值定义,子对象和方法的创建.因特性也可定义为对象和方法).Z_ESPP 中定义函词 create_obj 来完成这一动作:

```

create_obj((obj_name),(obj_vari),(def_vari),(if_needed),(if_added),
(normal_method)).

```

其中各语义项的含义与 1.1 节对象定义中的含义一致.

1.3 消息发送

在 Z_ESPP 中,请求对象运行的方法是向对象发送消息,消息的形式为:

```

send(object,P).

```

其中二元谓词 send 作为 Z_ESPP 的基本函词,在系统的解释器中进行定义和解释.它的含义为:在对象 object 定义的环境中证明目标 P,若证明成功,则该目标为真,回答“yes”,并返回一致化的结果,否则回答“no”.这里 P 为逻辑程序设计中的项.值得注意的是:在 Z_ESPP 中,把消息发送解释为目标证明(一致化过程),使得目标参数即可作为方法操作的输入参数,又可作为输出参数,具有灵活的操作调用方式.另外,方法可以有重定义,因此可用回溯来寻找正确的求解方法.

1.4 一致化算法

在 Z_ESPP 中,整个零件的工艺规划过程,可看成为目标证明的一致化过程.因

Z-ESPP是用 C 语言在 SUN 工作站上实现,故有必要简单介绍一下 Z-ESPP 中的一致化算法(z-mgu),在 z-mgu 中用到的有关项、原子、置换、复合置换的概念可见于一般的数理逻辑书中,本文不再介绍.

定义1.1. 令 S 是项的非空有穷集 $\{E_1, \dots, E_n\}$,我们记 $S\theta = \{E_1\theta, \dots, E_n\theta\}$,当且仅当 $E_1\theta = E_2\theta = \dots = E_n\theta$ 时, $S\theta$ 为单元素集,此时 θ 称为 S 的一致化置换.若对 S 的任一一致化置换 δ ,存在一个置换 γ ,使 $\delta = \theta\gamma$ (γ 依赖于 δ),则 θ 称为 S 的最一般的一致化置换,记为 z-mgu.

定义1.2. 设 S 为项的有穷非空集, S 的分歧集定义如下:在 S 的所有项中,找出对应符号不全相同的第一个符号(自左边算起),以该符号的位置为起点的子项的集合称为 S 的分歧集.

下面我们给出 Z-ESPP 的一致化算法如下:

输入:项的有穷集 S

输出:若 S 有 z-mgu,输出一个 z-mgu,否则报告 S 不可一致化.

步骤如下:

(1) $k \leftarrow 0; \delta_0 \leftarrow \epsilon;$

(2) 若 $S\delta_k$ 是单元素集,则输出 z-mgu δ_k 结束;否则计算 $S\delta_k$ 的分歧集 D_k ;

(3) 若 D_k 中有一个变量 v 和一个项 t ,且 v 不出现在 t 中,则 $\delta_{k+1} \leftarrow \delta_k\{v/t\}; k \leftarrow k+1$ 转 (2),否则输出“ S 不可一致化”结束.

2 Z-ESPP 的问题求解机制

上节我们已详细介绍了 Z-ESPP 的知识表示模型 KRM,在 KRM 基础上,可构造 Z-ESPP 的知识库.本节根据 Z-ESPP 的知识表示方式,构造 Z-ESPP 的问题求解模型,它包括单继承、多继承、非单调继承、类比推理等,现分别对它们进行介绍.

2.1 继承

继承关系包含实例/类(即 is-a)关系和子类/超类 super(即 a-kind-of 或 a-part-of).在 Z-ESPP 中,a-part-of 表示产品零件中整体与部分的关系.如在一个圆柱上钻一个深孔,深孔与圆柱有 a-part-of 关系,若深孔与圆柱表面的光洁度相同,则深孔的光洁度可通过 a-part-of 指针继承圆柱表面的光洁度.在一般的 CAPP 系统中,存在大量的 a-part-of 关系,由此也可看出在 Z-ESPP 中定义 a-part-of 关系的重要性)关系两个方面.实例对象接收到一个消息,它首先通过 is-a 指针访问所属类,寻找证明消息的方法,若该类中没有方法定义时,再用类的 super 指针访问类的超类.

2.2 非单调继承

当子类中的某些特性不能继承其父类中相同特性的值时,或者说子类不具有父类中的某些特征,我们说子类相对于父类来说出现了例外的情况.已往的面向对象系统通过代码重载来获得例外情况的处理^[3].

在 Z-ESPP 中,当出现例外情况时,我们提供 undo(求解无效)、MM(模型维护)操作来进行解决.为此,在 Z-ESPP 中引入了求解无效的概念,即为每个对象提供一个约束单元,

由该对象中说明语句:constrain_object(X)来定义。“X”对该对象及其超类所产生的求解结论进行选择,放弃在“X”或“X”的超类中说明的求解结果. 即当对象“O”证明出一个结论 S, 系统就询问“O”的约束对象“X”, 如果 S 在“X”中可证, 则自动置 S 无效, 否则 S 是“O”的一个有效结论.

2.3 多重继承

在 Z_ESPP 中, 多重继承用回溯来支持. 一个对象可以有多个 super 指针, 这时该对象的继承链有多条, 多重继承通过回溯依次在这些继承链上搜索来实现. 定义 super 指针的顺序决定了继承链的搜索顺序. 一般来说, 多重继承主要用来让对象可以继承两个不同类的方法.

2.4 类比推理

问题输入

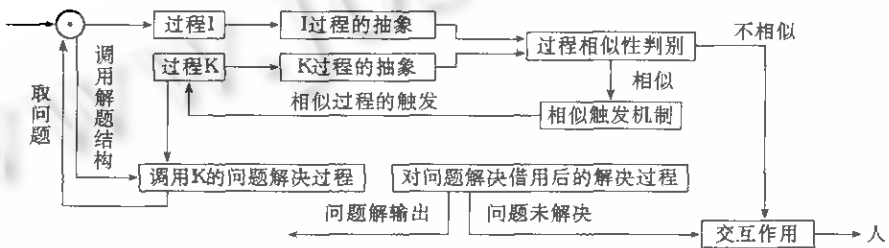


图1 一种相似问题求解系统

因为机械制造业中的大多数零件具有相似的工艺规划过程, 因此, 在 Z_ESPP 系统中引入类比推理机制不但能够避免很多重复劳动, 而且可以提高系统的运行性能和效能. 下面, 我们给出一种机制来反映这个过程^[4]:

在图1过程 I 中有一个问题要解决, 经过示例的过程 I 与过程 K 的“抽象”得到属性矢量的“相似”, 则触发 K 过程并启用 K 解决问题的过程以对过程 I 中的问题进行解决. 但这样做存在着3种可能性:

- (1) 问题就此解决, 过程终结;
- (2) 问题并未解决, 自动反馈系统将启动第2个相似过程, 如能解决, 则终结. 否则对第2步做递归直到问题解决;
- (3) 问题全然不能解决或部分不能解决, 则通过人——机交互系统, 借助人的智慧进行解决.

相似(类比)推理的核心是, 对不同域中的不同事物, 过程 I、K 的“相似性抽取”及“相似性判别”是怎样做出的, 下面, 我们对此进行详细的介绍.

2.4.1 相似性的比较

类比是根据两事物的相似性来进行的, 这是一个不精确的推理过程, 所以要通过相似度的计算来对类比的可靠性程度给出度量, 从而确定所要选择的类比源. 下面我们给出对象之间的相似性度量如下:

1. 正规化

设被分类的对象为 A_1, A_2, \dots, A_n , 考虑的因素(属性)为 B_1, B_2, B_m , 因此 A_i 可由 m 个属

性数据进行描述,记 A_i 所对应的 m 个数据为:

$$(x'_{i1}, x'_{i2}, \dots, x'_{im}) \quad (i=1, 2, \dots, n),$$

对因素 B_k 可测得 n 个数据为:

$$x'_{1k}, x'_{2k}, \dots, x'_{nk} \quad (k=1, 2, \dots, m).$$

(1)求平均值

$$\bar{x}'_k = \frac{1}{n} \sum_{i=1}^n x'_{ik} \quad k=1, 2, \dots, m.$$

(2)求标准差

$$s_k = \left(\frac{1}{n} \sum_{i=1}^n (x'_{ik} - \bar{x}'_k)^2 \right)^{1/2} \quad k=1, 2, \dots, m.$$

(3)求标准化差

$$x_{ik} = \frac{x'_{ik} - \bar{x}'_k}{s_k} \quad i=1, 2, \dots, n, \quad k=1, 2, \dots, m.$$

或者,令

$$x_{ik} = \frac{x'_{ik} - x'_{kmin}}{x'_{kmax} - x'_{kmin}} \quad i=1, 2, \dots, n, \quad k=1, 2, \dots, m.$$

其中 x'_{kmax}, x'_{kmin} 分别为 $x'_{1k}, x'_{2k}, \dots, x'_{nk}$ 中的最大值、最小值. 这样处理之后,就得到了 A_i 所对应的标准化数据组

$$(x_{i1}, x_{i2}, \dots, x_{im}) \quad (i=1, 2, \dots, n).$$

2. 标 定

标定,即是要求出被分类对象间相似程度的统计量 $r_{ij} (1 \leq i, j \leq n)$,从而确定出相似矩阵 $R = (r_{ij})_{n \times n}$.

由1所得到的 A_i 和 A_j 的标准化数据组按下列某种方法求得 A_i 和 A_j 的相似程度 $r_{ij} (1 \leq i, j \leq n)$.

(1)欧氏距离法

$$r_{ij} = \left(\frac{1}{m} \sum_{k=1}^m (x_{ik} - x_{jk})^2 \right)^{1/2}$$

(2)最大最小法

$$r_{ij} = \frac{\sum_{k=1}^m (x_{ik} \wedge x_{jk})}{\sum_{k=1}^m (x_{ik} \vee x_{jk})}$$

(3)算术平均最小法

$$r_{ij} = \frac{\sum_{k=1}^m (x_{ik} \wedge x_{jk})}{\frac{1}{2} \sum_{k=1}^m (x_{ik} + x_{jk})}$$

(4)绝对值指数法

$$r_{ij} = e^{-\sum_{k=1}^m |x_{ik} - x_{jk}|}$$

3. 相似划分

在 Z_ESPP 中,可采用等价矩阵划分方法或相似矩阵划分方法,下面分别进行介绍.

1. 等价矩阵划分方法

众所周知,在经典数学中将对象进行分类是依据经典的等价关系,相应地在 Z_ESPP

中是根据模糊等价关系来进行相似划分的. 因为, 一般所考察的对象通常是有限个, 因此这时的相似划分便是以等价矩阵为基础. 为此, 我们先给出如下定理:

定理 2.1. 设 $R \in IL_{n \times n}$ 是一等价矩阵, 若 $0 \leq \lambda \leq \mu \leq 1$, 则按 R_μ 所分出的每一类必是按 R_λ 所分出的某一类的子类 (此亦称 R_μ 的分法比 R_λ 的分法细).

$$\begin{aligned} \text{证明: } \mu r_{ij} = 1 &\Leftrightarrow r_{ij} \geq \mu \\ &\Leftrightarrow r_{ij} \geq \lambda \\ &\Leftrightarrow \lambda r_{ij} = 1, \end{aligned}$$

即若 A_i, A_j 按 R_μ 归为一类, 则按 R_λ 分类时, A_i 和 A_j 也必归为一类.

当 λ 从 1 降至 0, 分类则由细变粗, 逐步归并, 形成一个动态聚类相似图.

分类原则: 在 λ 水平上, A_i 与 A_j 同类 $\Leftrightarrow \lambda r_{ij} = 1$.

2. 相似矩阵划分方法

先将相似矩阵按平方法改造为等价矩阵, 然后根据改造后的等价矩阵按第一种方法再作聚类.

2.4.2 相似性方法转换^[5]

对于目标问题, 如果我们找到了它的相似问题, 那么下面我们要做的就是如何使源问题的求解方法能应用于目标问题的求解.

要把一个问题的求解方法应用到另一个问题的求解, 不仅要有它们的相似性关系, 而且必须在两个问题之间建立相关映射关系.

1. 相关映射

在 Z-ESPP 系统中, 问题及有关的概念是用属性来表征的, 如果两个问题有了相似性关系, 它们的属性间就存在一种映射关系, 而且不仅在其属性间可建立起一一对应的关系, 通过属性间的比较, 我们还能得到属性是否相似等有关信息, 这样通过相关映射, 两问题的属性之间可建起一个相关的对应表, 然后就可以进行类比方法的转换了.

2. 类比方法转换

方法转换一直是一个难以实现的问题, 其关键就在于不好从那一方面着手. 因此, 我们在 Carbonell 变换空间的基础上, 提出了对类比方法可进行的 4 种变换: 参数修正、方法补充、方法删减、方法替换. 这 4 种转换是通过在相关映射过程中得到的对应表来实现的. 我们知道, 不同的问题反应在描述它们的属性上是不同的, 而每个问题的求解方法是与其属性相关联的. 因而, 通过属性间的关系, 我们就能够将源问题的求解方法变换成适用于目标问题求解的方法.

(1) 参数修正: 如果源问题求解方法中所用的参数与目标问题的某个参数相对应, 且具有相似关系, 那么就用目标问题的参数替换源问题的参数.

(2) 方法补充: 如果目标问题中的某个属性在源问题中找不到与之相对应的属性, 这时有可能要补充与目标问题中此属性相关的方法.

(3) 方法删减: 在源问题中存在某种属性并有与之相关的求解方法, 但在目标问题中却没有转换成对应的属性, 那么在源问题的解题方法转换成目标问题的求解方法时, 这一相关的方法将要删除.

(4) 方法替换: 由于存在这样一种情况: 源问题的某一属性与目标问题的某一属性是相

对应的,但它们之间没有相似的关系,此时源问题与此属性相关的求解方法可能适用于目标问题的求解,也可能不适合目标问题的求解,当不适合时,将源问题与此属性相关的求解方法替换成适合于目标问题求解的新方法.

2.4 Z-ESPP 问题求解机制的特点

在 Z-ESPP 的问题求解机制中,由于提供了类比推理机制,从而可以实现传统派生式系统中的完全匹配搜索之外,也可处理相似零件的工艺规划过程. 这样一来,一方面可以引入相似比较和相似推理机制(类比推理机制),来寻找已有典型零件的规划过程,避免重复劳动和节约人力、物力,提高经济效益. 另一方面,由于 Z-ESPP 系统中同时融合了派生式和创成式工艺规划系统中所采用的成熟技术,可增强系统中推理机制的柔性,使之能力更强.

3 总 结

本文详细讨论了面向通用型 CAPP 专家系统开发工具 Z-ESPP 中知识表示和问题求解机制的实现方案. Z-ESPP 作为“产品模型为基础的 CAD/CAPP/CAM 系统”的一个重要组成部分已在 SUN 工作站上用 C 语言加以实现,并在杭州齿轮箱厂等单位进行了实际的推广和应用. 实际应用表明,本文所提出的知识表示模型 KRM 可以有效地解决一般 CAPP 系统中所存在的产品零件信息表示的完整性、通用性、逻辑可分解性、动态性和相容性等问题,且所提出的问题求解机制由于同时融合了派生式和创成式工艺规划系统中所采用的成熟技术,从而增强了系统推理机制的柔性,使之能力更强. 因此,本文所提出的思想和方法可作为通用型 CAPP 专家系统开发工具的核心模块,来满足国内机械制造行业对创成式 CAPP 系统日趋增长的需要.

参 考 文 献

- 1 Tien Chien Chang. Expert process planning for manufacturing. Addison—Wesley Publishing Company, Inc., 1990.
- 2 Tell E R. Object—oriented programming for artificial intelligence. Addison—Wesley Inc., 1989.
- 3 Lenat D.B. CYC: toward programming on common sense. Comm. of ACN, 1990.
- 4 杨莉,钟毓,何志均. Z-ESPP 的实现机制. 计算机研究与发展, 1994, 31(4):54—60.
- 5 杨莉. 知识库系统支撑环境的研究. 工学博士学位论文,国防科技大学研究生院,1992.

THE RESEARCH OF KNOWLEDGE REPRESENTATION AND INFERENCE MECHANISM ON GENERAL CAPP EXPERT SYSTEM

Yang Li and He Zhijun

(State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310027)
(Department of Computer Science, Zhejiang University, Hangzhou 310027)

Abstract The representation and the process planning of the machine parts are of key importance in general CAPP (Computer Aided Process Planning) expert system constructing tool. This paper proposes the knowledge representation model KRM of object-oriented(OO) machine part and process planning according to the theory of similarity among objects to satisfy the requirements of completeness, generalization, logical hierarchical segmentation, dynamic and consistency etc. of the machine part information representation in CAPP system, and present the hybrid inference mechanism of CAPP expert system based on KRM, including inheritance, multi-inheritance, nonmonotonic inheritance and analogical reasoning.

Key words Computer aided process planning, knowledge representation, inference mechanism, expert system.

中国计算机学会1994、1995年国际学术活动计划

会议名称	时间	地点	主办单位	联系人
The 3rd Pacific RIM International Conference on Artificial Intelligence 第三届太平洋地区国际人工智能会议	8.16—8.18	北京	CCF CAA	史忠植 2565533—419 Fax. 2567724
世界华人计算机科学技术及产业研讨会	8.28—9.1	北京	CCF CAS	李光华 2560911
HKICC(GZ)	10.3—10.4	广州	CCF HKCS	王介生 2565533—818
ICA'94展览会	10.6—10.10	上海	CCF SCS B/I	徐桂珍 3726055
国际软件测试和软件可靠性研讨会	10.18—10.20	北京	IEEE SE 专委 CCF 容错专委	闵应骅 2565533—836
The 19th Computer Software & Applications Conference 第19届国际计算机软件及应用会议	1995.10	北京	IEEE CS CCF	朱明远 6756956(O) 4919213(F)
ICYCS'95 第四届国际青年计算机学术会议	1995.8	北京	CCF 主办 智能中心承办	白硕 2565533—162
第四届 CAD & CG 国际学术会议	1995.10	武汉	CCF 主办 华中理工大学 (430074) 承办	华中理工大学 机一系 刘健