

蚁群优化算法在加工工艺知识发现中的应用

王铁伟 于 金 王明海

(沈阳航空工业学院机电工程学院,沈阳 110136)

文 摘 针对加工工艺数据复杂、庞大、多变的特点,为解决加工工艺知识的匹配与推理的关键性技术,将模拟生态系统的模糊聚类算法应用于加工工艺知识库。提出了基于蚁群算法的工艺知识发现的概率查询方法,并建立加工工艺知识库的数学模型。通过输入关键词运行蚁群算法,描绘出索引地图,提取分析所需要的加工工艺序列。最后以铣加工工艺为例进行实例验证,结果表明,基于蚁群算法的工艺知识发现高效全面地优化了工艺序列。

关键词 知识发现,蚁群优化,规则,聚类,索引地图

Ant-Colony Optimization Method for Knowledge Discovery in Processing Technology

Wang Tiewei Yu Jin Wang Minghai

(School of Mechanical and Electrical Engineering, Shenyang Institute of Aeronautical Engineering, Shenyang 110136)

Abstract Processing technology is much more complex, usually large in size, and continuously changes. Aiming to solve a series of key technologies during the matching and reasoning of discovery, a clustering algorithm that imitates the ecosystem taking into account the features of processing technology data was presented. A method of processing knowledge discovery is implemented by using of an Ant-Colony clustering algorithm and probable query, based on the method, the mathematical model of processing knowledge is established. By inputting process data, Ant-Colony algorithm can be operated, the index map can be described, the process procedure can be drawn. Finally, an instance is verified based on the process information table of the milling. The result proves that the knowledge discovery based on the Ant-Colony clustering algorithm model optimizes the process sequence comprehensively and efficiently.

Key words Knowledge discovery, Ant-Colony optimization (ACO), Rule, Clustering, Index map

0 前言

在加工工艺系统开发中,有很多因素影响加工方法的选择、工艺路线的排序、定位基准的决策、加工设备的选择、切削参数的选择、切削刀具的选择、加工余量的选择等。知识库系统采用文件管理系统的模式,将各种知识以文件形式存储于计算机中。但是由于工艺数据数量巨大,知识存储结构不清晰,造成使用效率低,并且不利于查询。

工艺知识发现过程是一个多目标、多组合优化的工艺过程,常见的获取知识的方法是归纳学习法和信息融合法,但由于工艺信息案例库的复杂性和不确定性,往往使得条件属性归类可能包含于多个分类中,无法获得确定性的关联规则。本文提出基于蚁群算法的加工工艺知识系统,针对加工工艺的特点,在文献[1]的基础上,通过模拟蚂蚁群体智能来处理数据

挖掘中的查询、表示方法和检索方法等问题新的解决方法。主要包括以下两点。

(1)采用分布式并行计算和正反馈机制,与数据库相结合,将蚁群算法在聚类中的应用到加工工艺知识库系统。通过信息素的不断更新达到查询中的最优解。

(2)模拟社会性昆虫搜索机制,应用概率算法从结构中分析并提取出所需要的序列,建立加工工艺数据库模型。

1 加工工艺特征的信息模型

加工工艺特征包括材料的选择、工艺要求、刀具的选择、设备的选择等。利用加工工艺特征参数作为信息模型的基本单元,并把特征参数的描述与加工工艺的数据库相结合,通过加工对象集合进行分类。如图1所示,建立了3个基础数据库:工艺和材料参数

收稿日期:2009-09-09

基金项目:辽宁省科学技术基金(2005400610)

作者简介:王铁伟,1981年出生,硕士研究生,主要从事数控加工工艺方面的研究。E-mail:Teewell@gmail.com

库、实例库和规则库。工艺参数库的数据主要包括毛料尺寸等,主要来源于仿真与试验结合所产生的工艺设计与分析参数数据。实例库的数据来源于实际生产中已加工的成型的产品。规则库的数据则来源于加工工艺手册^[2]。规则库由语言规则以及连接词构成,对蚁群搜索起到了链接作用,基于特征参数统一的规则描述,更有利于在搜索中降低复杂工艺问题的难度,提高工艺知识的发现概率。

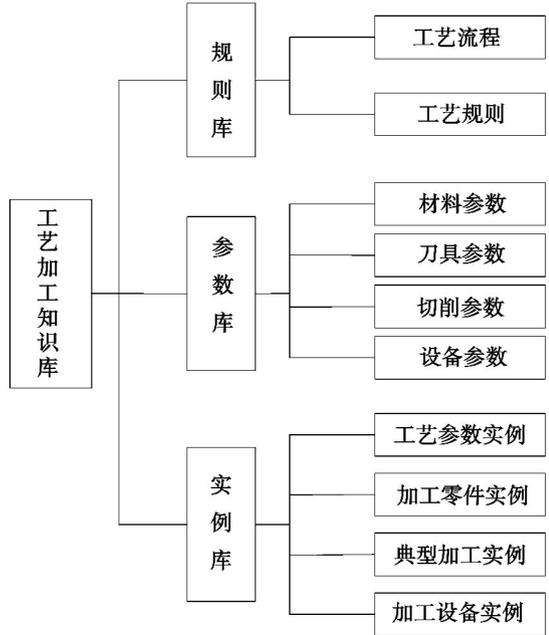


图1 知识库的结构图

Fig.1 Structure of knowledge base

2 蚁群算法的原理

人们受蚂蚁等昆虫的启发,设计了一系列蚁群算法^[3],模拟蚂蚁的群体智能来处理数据挖掘中的问题,根据数据对象的相似性度量出表达式。

首先引入如下记号:

m ——蚁群中蚂蚁数量;

$b_i(t)$ —— t 时刻位于结点 i 的蚂蚁个数, $m = \sum_{i=1}^n b_i(t)$;

d_{ij} ——两结点 i 和 j 之间的距离;

η_{ij} ——边 (i,j) 的能见度,反映由结点 i 转移到结点 j 的启发程度,这个量在蚂蚁系统的运行中不变;

τ_{ij} ——边 (i,j) 上的信息素轨迹强度;

$\Delta\tau_{ij}$ ——蚂蚁 k 在边 (i,j) 上留下的单位长度轨迹信息素量;

P_{ij}^k ——蚂蚁 k 的转移概率, j 是未访问的结点。

每只蚂蚁都具有如下的特征:

(1)在从结点 i 到结点 j 的运动过程中和在完成一次循环后,蚂蚁在路径 (i,j) 上释放一种信息素的物质,称为信息素轨迹;

(2)蚂蚁通过概率选择下一个将要访问的结点,这个概率是两个结点间距离和连接两个结点的路径上存有的轨迹量的函数;

(3)为了满足知识搜索的约束条件,在完成一次循环以前,不允许蚂蚁选择已经访问过的结点。

蚁群算法的流程如图2所示。

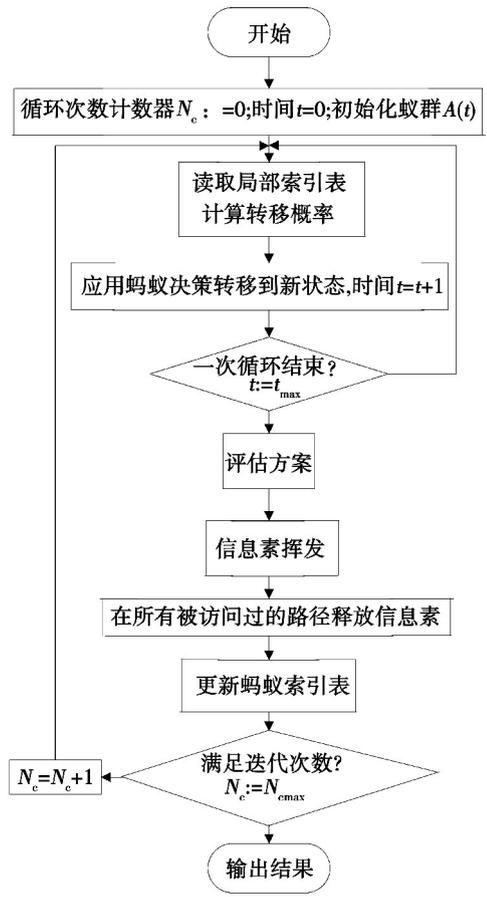


图2 蚁群优化算法流程图

Fig.2 Flow chart of Ant-Colony optimization

初始时刻,各条路径上的信息素量相等,设 $\tau_{ij}(0) = C$ (C 为常数)。蚂蚁 k ($k = 1, 2, 3, \dots, m$) 在搜索过程中根据各条路径上的信息素量决定着方向。蚂蚁系统所使用的状态转移规则被称为随机比例规则,它给出了位于结点 i 的蚂蚁 k 选择移动到结点 j 的概率。在 t 时刻,蚂蚁 k 在结点 i 选择结点 j 的转移概率 $P_{ij}^k(t)$ 为

$$P_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)}{\sum_{s \in \text{allowe } d_k} \tau_{is}^\alpha(t) \eta_{is}^\beta(t)}, & j \in \text{allowe } d_k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

其中, $\text{allowe } d_k = \{0, 1, \dots, n-1\}$ 表示 k 下一步允许选择的结点。由公式(1)可知,转移概率 $P_{ij}^k(t)$ 与 $\tau_{ij}^\alpha(t) \eta_{ij}^\beta(t)$ 成正比。 η_{ij} 为能见度因数, α 和 β 为两个参数,分别反映了蚂蚁在运动过程中所积累的信息和启发信息在蚂蚁选择路径中的重要程度。与真实蚂蚁不同的是人工蚁群系统具有记忆功能。为了满足

蚂蚁必须经过所有 n 个不同结点这个约束条件,为每只蚂蚁都设计了一个数据结构,称为索引地图(Index Map),索引地图记录了在 t 时刻蚂蚁已经走过的结点,不允许该蚂蚁在本次循环中再经过这些结点。

经过 n 个时刻,蚂蚁完成一次循环,各路径上信息素量根据公式(2)(3)调整

$$\tau_{ij}(t+1) = \rho\tau_{ij}(t) + \Delta\tau_{ij}(t, t+1) \quad (2)$$

$$\Delta\tau_{ij}(t, t+1) = \sum_{k=1} \Delta\tau_{ij}^k(t, t+1) \quad (3)$$

式中, $\Delta\tau_{ij}^k(t, t+1)$ 表示第 k 只蚂蚁在时刻 $(t, t+1)$ 留在路径 (i, j) 上的信息素量。路径越短,信息素释放的就越多; $\Delta\tau_{ij}(t, t+1)$ 表示本次循环中路径 (i, j) 的信息素量的增量。一只蚂蚁在经过路径 (i, j) 上释放的信息素量为每单位长度 Q/d_{ij} 。从而

$$\Delta\tau_{ij}^k(t, t+1) = \begin{cases} \frac{Q}{d_{ij}} & \text{若第 } k \text{ 只蚂蚁经过路径 } (i, j) \\ 0 & \text{否则} \end{cases} \quad (4)$$

文献[4-6]中在数据结构和知识发现方面进行了研究,设索引地图中由 n 个结点,其结点值分别为 d ,并用自然数编号为 $1, 2, \dots, n$ 。为了存储索引地图,首先用一个长度为 n 的一维数组 $D(1:n)$ 存放图中各数据结点的信息,再用一个 n 阶的二维数组 $R(1:n, 1:n)$ 存放图中各结点的关联信息。其中二维数组 R 称为图的关联矩阵(邻接矩阵)。在 R 中,每一个元素 $R(i, j)$ ($1 \leq i \leq n, 1 \leq j \leq n$) 的定义如下:

$$R(i, j) = \begin{cases} 1 & d_i \text{ 是 } d_j \text{ 的前结点} \\ 0 & d_i \text{ 不是 } d_j \text{ 的前结点} \end{cases} \quad (5)$$

因此关联矩阵可表示为 R 。

关联矩阵只表示了索引地图的结构,对两个结点之间的值运算需要值函数,另外还需要用求值矩阵 V 来存储。

最近邻列表 令 d_i 是从结点 i 到所有结点 j 的距离组成的数列, $j=1, \dots, n$ 且 $i \neq j$ 。令数列 d_i 以非递减的顺序排列,得到一个有序数列 d'_i , 值相等的项可以按顺序编排工序。

3 蚁群算法的在知识库中的实现

要把蚁群优化算法应用到工艺加工知识库中,需进行以下 4 个步骤^[7]: (1) 载入输入的数据,包括初始化数据; (2) 执行蚁群算法创建索引地图,索引地图显示了蚂蚁的运动搜索轨迹; (3) 处理数据挖掘中的聚类问题和评价函数; (4) 输出工序序列。本文重点描述索引地图和处理数据阶段。

3.1 蚂蚁的实现过程

在文献[8]中每只蚂蚁是一个基本计算单元,它负责构建待解问题的一条路径,并可能向它所经过的

边释放一定量的信息素 $\Delta\tau$ 。这样,每只蚂蚁必须能够: (1) 存储它至今所构建的部分路径; (2) 确定每个结点可能的相邻结点; (3) 计算并存储它构建的解的目标函数值。

3.1.1 初始化过程

设 $t:=0; N_c:=0; //t$ 时间计数器, N_c 循环次数计数器

$\tau_{ij}(t) := C; //$ 为每条路径 (i, j) 设一个轨迹强度的初始值 $\Delta\tau_{ij}=0$; 轨迹强度增量的初值设为 0

$\eta_{ij} = 1/d_{ij}; // \eta_{ij}$ 由某种启发式方法确定

$\text{inde } x_k = \emptyset; //$ 在初始阶段,索引表为空

设置 $s := 1 //s$ 为索引表的索引,将各蚂蚁的初始结点置于当前索引地图中

for $k := 1$ to n do

for $k := 1$ to $b_i(t)$ do

$\text{inde } x_k(s) = 1; //$ 以概率 $P_{ij}^k(t)$ 选择结点 $j; //$ 重复直到索引表满为止。

将选择的结点 j 加到 $\text{inde } x_k$ 中:

$$\Delta\tau_{ij}(t, t+1) := \Delta\tau_{ij}(t, t+1) + \frac{Q}{d_{ij}}$$

对于每一个路径 (i, j) , 根据方程(2)计算 $\Delta\tau_{ij}(t, t+1)$ 。

3.1.2 记录到目前为止的最优路径

If $N_c < N_{cmax}$

设置 $s := s+1$

$\text{inde } x_k(s) = 1; //$ 一次循环后蚂蚁又重新回到初始位置

设 $t := t+1$ 对于每一条路径 (i, j) , 设置 $\Delta\tau_{ij}(t, t+1) := 0$

else 输出序列

3.2 蚁群算法创建索引地图

为了把 ACO 算法应用到加工工艺知识库问题中,把它看做组合的优化问题,它通过图形来表示。把加工工艺知识库问题看作对于一定数量的规则,它就是一种涉及到用最优判据来给每个规则分配结论的方法。

将会得到一些规则与工序、刀具的选择和加工精度等之间的相似之处。每个规则所对应的可能的结论集合是不同的,而且也不可能为超过两个的规则分配一个结论。

为了构造索引地图,采取以下步骤。

3.2.1 确立规则

工艺知识表达方式应具有可解释性,因此对工艺知识的处理不能脱离工艺语言本身的固有特征,而且还要考虑灵活的数学描述处理方式。下面是对几种常见类型的工艺规则的预处理方法。

例如,一个规则 $R_i, i = 1, 2, \dots, N_r$ 由前提组合来定义为 $R_i = \text{IF } X_1 \text{ is } A_{i1} \text{ and } \dots \text{ and } X_n \text{ is } A_{in}$

Then Y is B_j (6)

其中, X_1, \dots, X_n 和 Y 分别是输入, 输出语言变量; $A_{i1} \dots A_{in}$ 和 B_j 均为由集合所定义的语言变量。

这个规则当且仅当存在

$\exists e_l = (x_1^l, \dots, x_n^l, y^l) \in E$ 使得

$$\mu_{A_{i1}}(x_1^l) \dots \mu_{A_{in}}(x_n^l) \neq 0 \quad (7)$$

才起作用。即, 在规则前提所定义子空间中, 至少存在一个满足条件的样本。具体的加工工艺数据规则预处理方法如下^[9]:

(1) 对区间型工艺规则的处理包括连续区间和离散区间的处理。典型的离散区间型工艺规则如表面粗糙度 Ra 。连续区间型的工艺样本如材料直径大小, 处理方法是将连续区间进行模糊离散化处理。区间型的工艺规则主要采用二次等价关系进行处理。

(2) 对于集值型工艺规则的处理主要是加工精度, 如 $\{\text{IT6}, \text{IT8}\}$, 集值型的工艺特征是属性值可以进行选择, 处理方法是根据集值元素逐一细化。

(3) 对于字符型工艺规则的处理主要是加工材料, 加工设备等。对于工艺规则值空缺的工艺信息决策表采用量化容差关系进行处理, 这里对这种情况暂不予考虑, 只考虑完备工艺信息决策表的情形。工艺样本间的关系是等价关系。

总之, 对工艺属性的预处理方法关键要看工艺属性的特征要求, 然后选择合适的二元关系。

3.2.2 连接规则和结论

当且仅当下面条件满足时, 规则 R_i 才和结论 $B_j (j = 1, \dots, N_c)$ 连接起来。

$\exists e_l = (x_1^l, \dots, x_n^l, y^l) \in E$ 使得

$$\mu_{A_{i1}}(x_1^l) \dots \mu_{A_{in}}(x_n^l) \mu_{B_j}(y^l) \neq 0 \quad (8)$$

其中, e_l 和 E 代表输入和输出数据集 (表示待解决问题的行为), μ 是规则的前提和输入的匹配度, 即语言变量 A_{in} 的隶属函数。

也就是, 至少在输入模糊子空间有一个样本, 被这样一个结论所包含。

3.3 评价函数

评价函数决定了解的质量, 用支持度函数和置信度函数来评估^[10]。

定义 1 序列 A 在序列数据库 D 中的支持度为

$$\sigma(A/D) = \frac{S_A}{S_B} \times 100\% \quad (9)$$

式中, σ 为支持度函数; $\sigma(A/D)$ 为序列数据库 D 中包含 A 的序列所占的百分比; S_A 为 D 中支持 A 的序列数; S_B 为 D 中的总序列数。

定义 2 数据库 D 中的序列规则是如下形式的一

种: $A \rightarrow B$, 其中 $A \subset I, B \subset I, I$ 为构成项目的集合, 且 $A \cap B = \emptyset$, 指在数据库 D 中所有支持序列 A 的序列中, 有 $c\%$ 的序列同时也支持 $B, c\%$ 称为序列规则 $A \rightarrow B$ 的置信度

$$\varphi(A \rightarrow B) = \frac{\sigma[(A \cup B)/D]}{\sigma(A/D)} \times 100\% \quad (10)$$

4 实验验证

以铣刀加工工艺规则为例, 对初始工艺规则集作数据处理, 得到表 1 所示的初始加工工艺参数表。

表 1 加工工艺参数表

Tab.1 Parameters of processing technology

序号	工艺条件, 输入规则			
	R_1	R_2	R_3	R_4
	加工精度	表面粗糙度/ μm	材料	加工直径
1	IT6	0.08 ~ 0.63	钢	中
2	IT7	0.08 ~ 0.63	铜	中
3	IT8	0.16 ~ 1.25	铝	小
4	IT9	2.5 ~ 10.00	钢	大

表 2 加工工艺决策表

Tab.2 Decision of processing technology

序号	工艺序列, 输出结论		
	B_1	B_2	B_3
	粗铣	半精铣	精铣
Y1		$B_1 \rightarrow B_3 \rightarrow B_2 \rightarrow B_3$	
Y2		$B_1 \rightarrow B_2 \rightarrow B_3 \rightarrow B_3$	
Y3		$B_1 \rightarrow B_2 \rightarrow B_3 \rightarrow B_2$	
Y4		$B_1 \rightarrow B_2 \rightarrow B_3$	

图 3 有 2 个输入变量 x_1 和 x_2 , 4 个规则 R_1, R_2, R_3 和 R_4 , 3 个输出 B_1, B_2 和 B_3 划分的一个简单工艺处理过程: (1) 每个规则对应的可能结论集合; (2) 索引图里除了 $\eta_{13}, \eta_{31}, \eta_{41}, \eta_{42}$ 为 0, 其他的 $\eta_{ij} \neq 0$; (3) 可能采取的 12 条不同的路径, 也就是, 结论组合成的索引地图。

图 3 是系统具有 4 条规则, 每个输入变量有 2 个条件, 每个输入变量有 3 个结论的一个例子。在图 3 (a) 中, 给出了对每个前提组合的结论。为了构造完全的解决方法, 一个蚂蚁反复地检查每个规则, 然后依据支持度信息 σ_{ij} 、置信度信息 φ_{ij} 和试探信息 η_{ij} 来选择可能的结论, 如图 3 (b) 所示。选择规则的次序是不相关的。在图 3 (c) 中, 可以看到在一个指定的例子里一个蚂蚁可能采取的各种路径。图 3 (d) 为第三种组合的规则决策表。

对于表面粗糙度 R_a 为离散区间, 因此 R_2 的取值范围 $t_1 \in [a_1, b_1], t_2 \in [a_2, b_2]$

(1) 如果 $a_2 \geq b_1$, 则 Y is B_1 , 如果 $a_1 \geq b_2$, 则 Y is

B_2 ;

(2) 如果 $b_2 \geq b_1, a_2 \geq a_1$, 则 Y is B_1 ;

(3) 如果 $a_1 \geq a_2, b_1 \geq b_2$, 则 Y is B_1 ;

(4) 如果 $t_1 \subseteq t_2$, 则 Y is B_1 , 反之, Y is B_2 。

可以推断例子中 Y is B_1 。

根据公式(7),

$R_1 = \text{IF } X_1 \text{ is } M \text{ and } X_2 \text{ is } S \text{ Then } Y \text{ is } B_1$
 $R_2 = \text{IF } X_1 \text{ is } S \text{ and } X_2 \text{ is } M \text{ Then } Y \text{ is } B_2$
 $R_3 = \text{IF } X_1 \text{ is } M \text{ and } X_2 \text{ is } M \text{ Then } Y \text{ is } B_2$
 $R_4 = \text{IF } X_1 \text{ is } L \text{ and } X_2 \text{ is } S \text{ Then } Y \text{ is } B_3$

因此, 从第五组合就得到的一个规则。
 相应的加工工序见表 3。

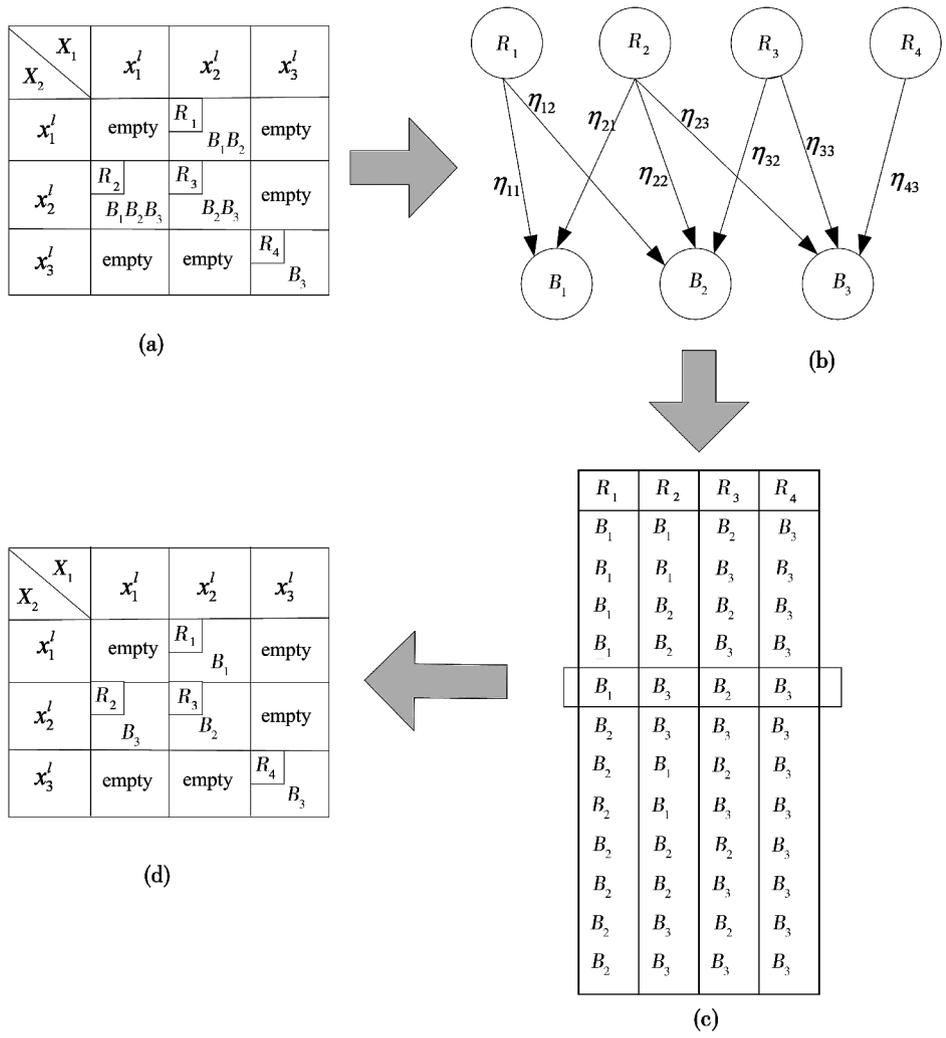


图 3 工艺处理生成的索引地图

Fig. 3 Index map of processing knowledge discovery

表 3 加工工艺工序表

Tab. 3 Process sequence of processing technology

编号	加工工序
1	粗铣→粗铣→半精铣→精铣
2	粗铣→粗铣→精铣→精铣
3	粗铣→半精铣→半精铣→精铣
4	粗铣→半精铣→精铣→精铣
5	粗铣→精铣→半精铣→精铣

所获得的加工工序规则覆盖了原始过程中工艺决策表中的所有工艺流程^[11], 将隐含的工艺数据信息中的决策信息也表达出来, 与要求的加工工艺过程

相符, 可以将其作为工艺人员的参考。

5 结论

从蚁群优化算法来解决加工工艺知识库中知识发现的的过程可以看到, 预先对工艺规则进行分类处理, 然后通过蚁群算法可以有效地避免局部极优, 快速全局地对工艺知识进行挖掘, 表达方式清晰全面, 有助于工艺人员在工艺设计工程中提高效率。

参考文献

[1] Marco Dorigo, Christian Blum. Ant colony optimization theory: A survey[J]. Theoretical Computer Science, 2005, 344: 243-278

(下转第 38 页)