

基于约简矩阵和 C4.5 决策树的故障诊断方法

徐 曩 张 斌

(桂林电子科技大学 计算机与信息安全学院, 广西 桂林 541000)

摘要: 现阶段由于智能装备的结构和功能不断完善,机械故障预兆和故障特征也不断复杂化,导致了故障诊断难度大大增加。由于机器学习和数据挖掘技术的不断革新,基于数据挖掘的故障诊断系统快速发展,提高了故障诊断效率,减少了因诊断延迟造成的损失。对此,提出一种基于辨识约简矩阵的决策树故障诊断方法,实现了故障样本决策表的高效生成并保证诊断的正确性。首先采用粗糙集的决策树方法建立故障诊断决策表,然后离散化处理特征数据;接着采用可辨别矩阵约简算法进行属性约简,删除冗余信息,形成精简的决策表;最后使用 C4.5 算法构造出最终决策树,并用该方法与直接使用 C4.5 算法所生成决策树进行对比分析。实验结果表明,该方案有一定的容错能力,并且是一种快速、可靠的故障诊断方法。

关键词: 粗糙集; 可辨别矩阵; C4.5 算法; 决策树; 故障诊断

中图分类号: TP393

文献标识码: A

文章编号: 1673-629X(2018)02-0040-05

doi: 10.3969/j.issn.1673-629X.2018.02.010

A Fault Diagnosis Method Based on C4.5 Decision Tree and Reduction Matrix

XU Zhao ZHANG Bin

(School of Computer and Information Security, Guilin University of Electronic Technology, Guilin 541000, China)

Abstract: Due to the continuous improvement of the structure and function of intelligent equipment at present, the mechanical failure warning and fault features are also becoming increasingly complicated, resulting in a great increase in fault diagnosis. With the continuous innovation of machine learning and data mining technology, the fault diagnosis system based on data mining is developing rapidly, which improves the efficiency of fault diagnosis and reduces the loss caused by the delay of diagnosis. For this, we put forward a decision tree fault diagnosis scheme based on identification reduction matrix, which can effectively extract the fault samples and ensure the correctness of the diagnosis. Firstly, the decision tree method based on rough set is used to establish the fault diagnosis decision table, and then the characteristic data are discretized. Then the algorithm of the discernible matrix reduction is used for attribute reduction, deleting redundant information, and forming a simplified decision table. Finally, the final decision tree is constructed by C4.5 algorithm and is compared with the decision tree generated by C4.5 algorithm. The experiments show that the scheme, with a certain fault tolerance, is a fast and reliable fault diagnosis method.

Key words: rough set; discernible matrix; C4.5 algorithm; decision tree; fault diagnosis

0 引言

工业 4.0 环境下,智能设备通过多种传感器采集实时参数产生大量的数据。利用数据挖掘进行故障诊断就是将状态参数进行关联、分类、推理,得到决策树^[1],并利用决策树对智能设备进行故障诊断。相关研究表明:工业机械遇到故障时,技术人员需要花 70%~90% 的时间来寻找故障原因和部位,而故障维修只占 10%~30% 的时间;在维修成本方面,预测维修成本只占故障后维修成本的 40%^[2]。

机械设备结构功能的逐渐增强,使得现阶段故障诊断的难度不断提高。文献[3]是基于知识库逻辑理论对规则进行处理,但没有给出规则间的相互关系,诊断效率不高。文献[4]是基于知识库逻辑理论,利用粗糙集直接从故障样本中得到诊断规则,效率不高;并且在缺乏属性间的关系时,必须对整个规则库遍历搜索,而且当缺失某些关键信息时诊断结果会受到影响。文献[5]结合了故障树与模糊推理用于动态系统的故障诊断;文献[6]利用粗糙集研究了故障树基本事件

收稿日期: 2017-02-25

修回日期: 2017-06-27

网络出版时间: 2017-11-15

基金项目: 国家自然科学基金(11562006)

作者简介: 徐 曩(1991-),男,硕士研究生,研究方向为工业监控;张

斌,硕士,副教授,研究方向为计算机网络与应用技术、计算机控制技术、网络安全。

网络出版地址: <http://kns.cnki.net/kcms/detail/61.1450.TP.20171115.1425.028.html>

的排序问题。当系统比较复杂时,这类方法的搜索过程也变得非常复杂,诊断结果的准确性也不高。

为解决上述问题,文中结合可辨识矩阵与决策树的特点,提出基于可辨识矩阵约简和 C4.5 决策树的故障诊断方法,并对其进行了验证。

1 约简决策树故障诊断方法

首先确定故障诊断的训练样本集,接着对训练样本集里的连续属性进行离散化处理,得到离散样本集。然后用可辨识矩阵约简算法对离散样本集进行属性约简,并根据聚类比选择出最佳约简矩阵,得到最佳约简样本集。再用 C4.5 算法对最佳约简样本集进行分析,生成最终决策树。最后通过测试样本集验证决策树的故障诊断准确性。

训练样本需要全面反映设备的情况,这样的样本经过训练后生成的决策才能对新的数据做出正确的判断。以转子磨粉机为例,有转子不平衡、转子不对中、油膜振荡三种常见机械故障。由历史振动波形测量数据建立如表 1 所示的样本训练集 $T = \langle U, C, D \rangle$ 。其中, U, C 和 D 是非空有限集。 U 是故障状态域; $C = (C_1, C_2, \dots, C_7)$ 表示 7 种故障条件属性集, C_1 表示 $0.01 \sim 0.40f$ 、 C_2 表示 $0.41 \sim 0.50f$ 、 C_3 表示 $0.51 \sim 0.99f$ 、 C_4 表示 $1f$ 、 C_5 表示 $2f$ 、 C_6 表示 $3 \sim 5f$ 、 C_7 表示大于等于 $5f$; $D = (F_1, F_2, F_3)$ 是故障决策属性,其中 F_1 为不平衡、 F_2 为不对中、 F_3 为油膜振荡,且 $C \cap D = \emptyset$; $X = (X_1, X_2, \dots, X_9)$ 表示 9 个故障样本。

表 1 训练样本集 T

U	C_1	C_2	C_3	C_4	C_5	C_6	C_7	D
X_1	0.105 6	0.146 2	0.117 6	0.956 9	0.006 6	0.044 2	0.050 3	F_1
X_2	0.046 0	0.022 9	0.085 2	0.589 2	0.079 7	0.298 5	0.031 8	F_1
X_3	0.110 7	0.006 1	0.114 3	0.961 7	0.032 3	0.050 7	0.024 4	F_1
X_4	0.096 0	0.037 6	0.060 3	0.501 2	0.341 5	0.131 3	0.110 6	F_2
X_5	0.146 8	0.012 8	0.078 6	0.582 9	0.555 3	0.031 8	0.126 2	F_2
X_6	0.130 7	0.107 7	0.138 3	0.575 2	0.424 8	0.120 6	0.076 5	F_2
X_7	0.025 5	0.219 7	0.077 7	0.230 9	0.050 9	0.108 6	0.123 9	F_3
X_8	0.211 8	0.211 5	0.020 6	0.358 3	0.075 5	0.026 6	0.066 8	F_3
X_9	0.395 7	0.522 5	0.109 7	0.198 4	0.064 5	0.103 4	0.086 3	F_3

2 连续属性离散化算法

连续属性的离散化是非常必要的,因为粗糙集理论以及多种离散型数据挖掘算法只能解决离散数据,因此需要对训练样本集进行离散化处理。离散化方法主要包括两大类:无监督的离散化方法和有监督的离散化方法。基于熵的离散化方法就是在处理数据时考虑了条件属性和决策属性之间的依赖关系。

假设样本集有 N 个故障样本 $X_j(j = 1, 2, \dots, N)$, M 个故障条件属性, K 个故障类型。应用一种基于最

小熵原理的离散算法将样本集中的连续属性 C_i 进行离散化处理^[7]。步骤如下:

步骤 1: 把连续属性值 $C_{iN}(N = 1, \dots, j, \dots, N)$ 按照升序排列,记排列后的属性值为 $C'_{iN}(N = 1, \dots, j, \dots, N)$ 。计算排序后每相邻两个属性值的平均值,并记作 $\bar{C}'_{i1}, \dots, \bar{C}'_{ij}, \dots, \bar{C}'_{i(N-1)}$, 其中 $\bar{C}'_{ij} = (C'_{ij} + C'_{i(j+1)}) / 2, j = 1, 2, \dots, N - 1$ 。

步骤 2: 离散分类数 $r = 3$ 。

步骤 3: 对于每一个 $\bar{C}'_{ij}(j = 1, 2, \dots, N - 1)$ 都可以将 $[C'_{i1}, C'_{iN}]$ 分成两个区间 $[C'_{i1}, \bar{C}'_{ij}]$ 和 $[\bar{C}'_{ij}, C'_{iN}]$, 两个子区间分别记作 p 区间和 q 区间; 根据式 (1) ~ (5) 得到 \bar{C}'_{ij} 在 $[C'_{i1}, C'_{iN}]$ 中所对应的信息熵 $S(\bar{C}'_{ij})(j = 1, 2, \dots, N - 1)$, PRI_i 作为信息熵最小的 \bar{C}'_{ij} 。

$$S(\bar{C}'_{ij}) = p(\bar{C}'_{ij}) S_p(\bar{C}'_{ij}) + q(\bar{C}'_{ij}) S_q(\bar{C}'_{ij}) \quad (1)$$

$$S_p(\bar{C}'_{ij}) = - \sum_{k=1}^K p_k(\bar{C}'_{ij}) \ln p_k(\bar{C}'_{ij}) \quad (2)$$

$$S_q(\bar{C}'_{ij}) = - \sum_{k=1}^K q_k(\bar{C}'_{ij}) \ln q_k(\bar{C}'_{ij}) \quad (3)$$

$$p_k(\bar{C}'_{ij}) = (n_k(\bar{C}'_{ij}) + 1) / (n(\bar{C}'_{ij}) + 1) \quad (4)$$

$$q_k(\bar{C}'_{ij}) = (N_k(\bar{C}'_{ij}) + 1) / (N(\bar{C}'_{ij}) + 1) \quad (5)$$

其中, $p(\bar{C}'_{ij})$ 与 $q(\bar{C}'_{ij})$ 指全部故障在 p 区间和 q 区间上的概率; k 是故障类型; $p_k(\bar{C}'_{ij})$ 和 $q_k(\bar{C}'_{ij})$ 分别是 k 类故障在 p 区间和 q 区间上的条件概率; $n_k(\bar{C}'_{ij})$ 和 $N_k(\bar{C}'_{ij})$ 是 k 类故障在 p 区间和 q 区间上的样本数量; $n(\bar{C}'_{ij})$ 和 $N(\bar{C}'_{ij})$ 是 k 类故障在 p 区间和 q 区间上的样本总数; N 是 $[C'_{i1}, C'_{iN}]$ 的样本总数。

步骤 4: 对于各个 $\bar{C}'_{ij}(\bar{C}'_{ij} < \text{PRI}_i)$, 按照式 (1) ~ (5) 再次计算在区间 $[C'_{i1}, \text{PRI}_i]$ 内的信息熵, 并且使 SEC1_i 等于信息熵最小的 \bar{C}'_{ij} ; 对于 $\bar{C}'_{ij}(\bar{C}'_{ij} > \text{PRI}_i)$, 按照式 (1) ~ (5) 再次计算在区间 $[\text{PRI}_i, C'_{iN}]$ 内的信息熵, 并且使 SEC2_i 等于信息熵最小的 \bar{C}'_{ij} 。

步骤 5: 根据最大隶属度原则, 将样本集中的连续属性值按照式 (6) 进行离散化处理, 处理过后的条件属性值记为 $\text{Cd}_{iN}(N = 1, 2, \dots)$:

$$\text{Cd}_{ij} = \begin{cases} 1, & C_{ij} \leq P1_i \\ 2, & P1_i < C_{ij} < P2_i \\ 3, & C_{ij} > P2_i \end{cases} \quad (6)$$

其中, $P1_i$ 和 $P2_i$ 是连续属性离散化过程中的两个断点值, 其中 $P1_i = (\text{SEC1}_i + \text{PRI}_i) / 2, P2_i = (\text{SEC2}_i + \text{PRI}_i) / 2, j = 1, 2, \dots, N$ 。

将样本训练集 T 中连续属性进行离散化处理后

得到离散样本集 T' 。

3 属性约简及最佳约简优选

粗糙集理论^[8]可以在缺乏先验知识的前提下,以分类能力为基础很好地进行模糊数据的分类和处理。粗糙集的关键内容是属性约简,能够有效地删除样本集中的冗余信息,形成精简的规则库,提高规则库的读取使用效率^[9]。文中使用可辨别矩阵作为样本集属性的约简算法。

假设 $T' = (U, C', D)$, 其中包括 N 个经过离散化处理的样本 $Y_j(j = 1, 2, \dots, N)$, M 个条件属性和 K 个故障类型。基于可辨别矩阵约简^[10]和最大集群比^[11]原则,文中采用如下算法对 T' 进行约简并选择出聚类率最高的属性组合。

步骤 1: 按照式 (7) 计算出 T' 的可辨别矩阵 $C = (c_{ij})_{N \times N}$ 。

$$C(i, j) = \begin{cases} \{a_k \in A \mid a_k(y_i) \neq a_k(y_j), D(y_i) \neq D(y_j)\} \\ 0, D(y_i) = D(y_j) \end{cases} \quad (7)$$

步骤 2: 对辨识矩阵 C 中所有值为非空集合的元素 $C_{ij}(C_{ij} \neq 0, C_{ij} \neq \emptyset)$ 构建出对应的析取逻辑表达式,记作 $L_{ij} = \bigcup_{a_i \in C_{ij}} a_i$ 。

步骤 3: 把步骤 2 中得到的全部析取逻辑表达式进行合取运算,记作 $L' = \bigcap_i L_i$ 。

步骤 4: 输出属性约简结果。

步骤 5: 按照式 (8) 计算步骤 5 中得到的所有约简集的聚类比。最佳约简集即为聚类比 R_a 最大的一组。

$$R_a = (N - N_R) / N - 1 \quad (8)$$

其中, N 为规则表中离散化的样本数目; N_R 为 T' 约简后的论域个数。

用步骤 1 对离散样本集 T' 进行处理,得到对应的可辨别矩阵,如表 2 所示。

表 2 对应于 T' 的可辨别矩阵

	1	2	3	4	5	6	7	8	9
1	\emptyset								
2	0	\emptyset							
3	0	0	\emptyset						
4	$C_2, C_3, C_4, C_5, C_6, C_7$	C_1, C_5, C_6, C_7	C_3, C_4, C_5, C_7	\emptyset					
5	C_2, C_3, C_4, C_5, C_7	C_1, C_5, C_6, C_7	C_3, C_4, C_5, C_6, C_7	0	\emptyset				
6	C_2, C_4, C_5, C_6, C_7	C_1, C_3, C_5, C_6, C_7	C_4, C_5, C_7	0	0	\emptyset			
7	C_1, C_3, C_4, C_6, C_7	C_2, C_4, C_6, C_7	C_1, C_2, C_3, C_4, C_7	C_1, C_2, C_4, C_5, C_7	C_1, C_2, C_4, C_5, C_6	$C_1, C_2, C_3, C_4, C_5, C_7$	\emptyset		
8	C_3, C_4, C_7	$C_1, C_2, C_3, C_4, C_6, C_7$	C_2, C_3, C_4, C_6, C_7	C_2, C_3, C_4, C_5, C_6	C_2, C_3, C_4, C_5, C_7	C_2, C_3, C_4, C_5, C_6	C_2, C_3, C_4, C_5, C_7	0	\emptyset
9	C_1, C_2, C_4, C_6, C_7	$C_1, C_2, C_3, C_4, C_6, C_7$	C_1, C_2, C_4, C_7	C_1, C_2, C_3, C_4, C_5	$C_1, C_2, C_3, C_4, C_5, C_6, C_7$	C_1, C_2, C_4, C_5	0	0	\emptyset

根据步骤 2~4 进一步对 T' 各个属性进行约简,可得: $\{C_3, C_4, C_7\}, \{C_2, C_4, C_6, C_7\}, \{C_1, C_2, C_4, C_7\}, \{C_1, C_5, C_6, C_7\}, \{C_4, C_5, C_7\}, \{C_1, C_2, C_4, C_5\}, \{C_2, C_3, C_4, C_5, C_7\}$, 其中 $\{C_3, C_4, C_7\}$ 的聚类率最大,所以把 $\{C_3, C_4, C_7\}$ 作为最佳约简集,从而得到最优约简诊断决策表 T'' ,如表 3 所示。

4 C4.5 故障决策树算法

4.1 C4.5 算法

C4.5 算法^[12]是目前最具影响的决策树算法之一,以信息增益率作为确定分支属性的标准,更好地处理算法在使用信息增益选择分支属性时偏向于取值较多的属性的缺陷^[13]。核心思想是通过分析训练集的

表 3 最优约简诊断决策表 T''

U	C_3	C_4	C_7	D
X_1	3	3	1	F_1
X_2	2	2	1	F_1
X_3	3	3	1	F_1
X_4	2	2	2	F_2
X_5	2	2	3	F_2
X_6	3	2	2	F_2
X_7	2	1	3	F_3
X_8	1	1	2	F_3
X_9	3	1	2	F_3

数据,在数据集上递归地建立一个决策树,该决策树由一组规划来表示,尤其是被识别的模式有连续特征值

属性时,甚至是在缺失某些重要特征属性时,C4.5 算法也是有效的。该算法用已知故障类别的样本得到顺序测试特征,直到所有特征被正确归类。在得到决策树之后,只需要判断部分的条件属性,即可得到正确的故障类型^[14]。C4.5 算法使用后剪枝方法简化决策树,从而提高多特征下的辨识效果。过程如下:

步骤 1: 计算所有属性的信息增益率,选择具有最大信息增益率的属性作为根节点的分支属性。

(1) 设数据集为 T , $|T|$ 作为数据集 T 的样本数,共有 k 个类。 $|C_i|$ 是 T 中属于类别 C_i 的样本数,先验概率可以表示为 $P_i = |C_i|/|T|$,并且对 T 分类的信息熵为:

$$I(|C_1|, |C_2|, \dots, |C_k|) = - \sum_{i=1}^k P_i \log_2 P_i \quad (9)$$

(2) 设属性 $A_k (k = 1, 2, \dots, m)$ 有 q 个不同取值 $\{a_{1k}, a_{2k}, \dots, a_{qk}\}$,描述属性 A_k 将 T 划分为 q 个子集 $\{T_1, T_2, \dots, T_q\}$ 。 $|T_j|$ 是在子集 T_j 中的样本数, $|C_j^i|$ 是 T_j 属于 C_i 的样本数,则 A_k 划分 T 得到的信息熵为:

$$E(A_k) = \sum_{j=1}^q \frac{|T_j|}{|T|} I(|C_j^1|, |C_j^2|, \dots, |C_j^k|) \quad (10)$$

$$I(|C_j^1|, |C_j^2|, \dots, |C_j^k|) = - \sum_{i=1}^k P_{ij} \log_2 P_{ij} \quad (11)$$

并由式 (9) ~ (11) 可以得到按照 A_k 划分数据集的信息增益为:

$$\text{Gain}(A_k) = I(|C_1^1|, |C_1^2|, \dots, |C_1^k|) - E(A_k) \quad (12)$$

其中 A_k 的信息熵为:

$$\text{Split}(A_k) = - \sum_{k=1}^m \frac{|T_k|}{|T|} \log_2 \frac{|T_k|}{|T|} \quad (13)$$

最终由式 (12)、(13) 得到 A_k 的信息增益率为:

$$\text{Gain_Ratio}(A_k) = \frac{\text{Gain}(A_k)}{\text{Split}(A_k)} \quad (14)$$

步骤 2: 不断利用上述方法建立树的分支,将信息增益率最大的属性作为子节点,直到完成决策树的创建为止。

4.2 生成故障诊断决策树

利用 C4.5 算法对 T^m 进行分析,计算可得 C_4 的信息增益率为 0.8,同理可得 C_3 和 C_7 的信息增益率分别为 0.18 和 0.6。因为 C_4 的信息增益率最大,所以将 C_4 作为根节点建立决策树,并且在每个节点使用 C4.5 算法进行分析,最后生成的决策树如图 1 所示。使用 C4.5 算法直接对 T^m 进行分析,得到的决策树如图 2 所示。并且用离散化测试样本对最终决策树和 C4.5 算法生成的决策树进行效果验证。

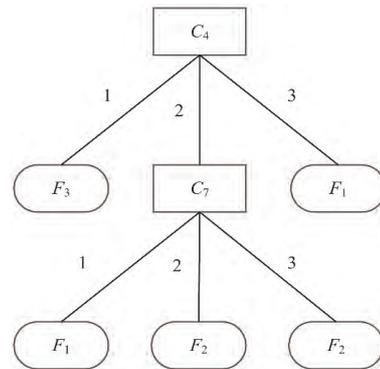


图 1 约简的决策树(文中算法)

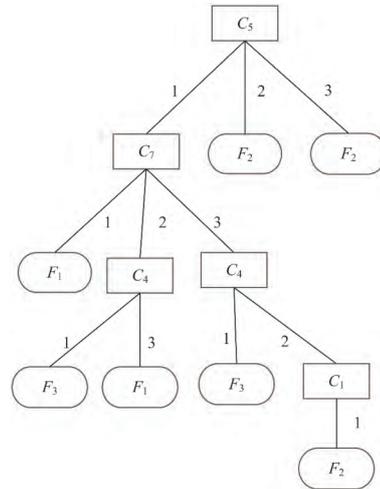


图 2 C4.5 算法决策树

5 实验结果及分析

(1) 决策树构建速度的比较。

5 次使用相同训练数据集得到决策树的时间,可以得出文中方法生成决策树的平均时间比直接使用 C4.5 算法快约 31%。

(2) 决策树复杂度的比较。

首先,直接使用 C4.5 算法生成的决策树(图 2)所需要的故障特征向量为 4 维,而文中方法(图 1)仅需故障特征向量为 2 维。证明文中方法在保证的诊断准确性的同时,有效地减少了故障特征数据获取的工作量,并且降低了决策树的复杂度。

(3) 故障诊断准确率的比较。

使用两个决策树对相同故障数据进行准确性测试,如图 3 所示。其中对于 F_1 故障类型的诊断准确率基本相同,达到了 95.3%。C4.5 算法对于 F_2 故障类型诊断正确率为 95.3%,而文中方法的诊断准确率达到 96%。C4.5 算法对于 F_3 故障类型的诊断正确率为 96.5%,而文中方法的诊断准确率达到 97.8%。两种方法的准确率出现了较为明显的差异。其次,文中方法可以准确识别故障类型,说明将粗糙集和决策树结合的方法不仅减少了规则库搜索的步骤,同时具有很

强的容错能力。

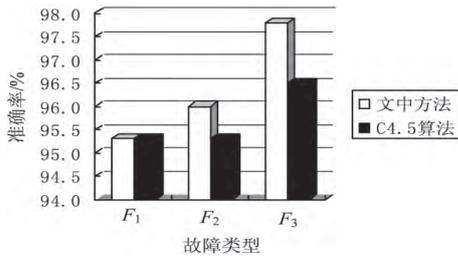


图 3 故障诊断准确率

综上所述可以证明,文中方法不仅在降低最终决策树复杂度的基础上提高了决策树的构建速率,还提高了对部分故障类型的诊断准确率。

6 结束语

应用可辨识矩阵约简算法和决策树结合的方法能有效地对条件属性进行约简,消除冗余信息,并减少决策树的节点数,降低决策树复杂度,最终高效率地构建决策树,实现对机械快速准确的故障识别。然而,数据挖掘技术是建立在海量的数据样本的基础上^[15],文中的故障样本数据是比较典型的故障类别数据,不排除存在具有两种甚至更多的故障特征数据。当样本数据更加充足时,决策规则库也将有更强的工程实用性。

参考文献:

[1] 陈绍伟,王聪,赵帅.决策树算法在电路故障诊断中的应用[J].计算机工程与应用,2013,49(12):233-236.

[2] GILLUM M N, ARMIJO C B. Optimizing the frequency of the rotary knife on a roller gin stand [J]. Hydrobiologia, 2000, 43(4): 809-817.

[10] CHETTY S, ADEWUMI A O. Comparison study of swarm intelligence techniques for the annual crop planning problem [J]. IEEE Transactions on Evolutionary Computation, 2014, 18(2): 258-268.

[11] MEHRABIAN A R, LUCAS C. A novel numerical optimization algorithm inspired from weed colonization [J]. Ecological Informatics, 2006, 1(4): 355-366.

[12] 苏守宝,汪继文,张铃,等.一种约束工程设计问题的入侵性杂草优化算法[J].中国科学技术大学学报,2009,39(8):885-893.

[13] KARIMKASHI S, KISHK A A. Invasive weed optimization and its features in electromagnetic [J]. IEEE Transactions on Antennas and Propagation, 2010, 58(4): 1269-1278.

[14] 张勋才.自组装DNA计算模型的研究及应用[D].武汉:华中科技大学,2009.

[15] SEDIGHY S H, MALLAHZADEH A R, SOLEIMANI M, et

[3] 王庆,巴德纯,王晓冬.智能故障诊断的粗糙决策模型[J].东北大学学报:自然科学版,2005,26(1):80-83.

[4] 束洪春,孙向飞,司大军.基于粗糙集理论的配电网故障诊断研究[J].中国电机工程学报,2001,21(10):73-77.

[5] CHANG S Y, LIN C R, CHANG C T. A fuzzy diagnosis approach using dynamic fault trees [J]. Chemical Engineering Science, 2002, 57(15): 2971-2985.

[6] KHOO L P, TOR S B, LI J R. A rough set approach to the ordering of basic events in a fault tree for fault diagnosis [J]. International Journal of Advanced Manufacturing Technology, 2001, 17(10): 769-774.

[7] 贾智皓,刘方.服务于粗糙集信息处理的数据离散化技术[J].数字技术与应用,2014(11):77-79.

[8] 马文萍,黄媛媛,李豪,等.基于粗糙集与差分免疫模糊聚类算法的图像分割[J].软件学报,2014,25(11):2675-2689.

[9] 李明祥.基于粗糙集理论的数据挖掘方法的研究[D].青岛:山东科技大学,2003.

[10] 武志峰,吉根林.一种基于决策矩阵的属性约简及规则提取算法[J].计算机应用,2005,25(3):639-642.

[11] 滕书华,鲁敏,杨阿锋,等.基于一般二元关系的粗糙集加权不确定性度量[J].计算机学报,2014,37(3):649-665.

[12] 徐鹏,林森.基于C4.5决策树的流量分类方法[J].软件学报,2009,20(10):2692-2704.

[13] 李学明,李海瑞,薛亮,等.基于信息增益与信息熵的TFIDF算法[J].计算机工程,2012,38(8):37-40.

[14] 黄爱辉.决策树C4.5算法的改进及应用[J].科学技术与工程,2009,9(1):34-36.

[15] 云玉屏.基于C4.5算法的数据挖掘应用研究[D].哈尔滨:哈尔滨理工大学,2008.

(上接第39页)

(ABC) algorithm [J]. Journal of Global Optimization, 2007, 39(3): 459-471.

al. Optimization of printed Yagi antenna using invasive weed optimization (IWO) [J]. IEEE Antennas and Wireless Propagation Letters, 2010, 9(1): 1275-1278.

[16] ZHANG X, WANG Y, CUI G, et al. Application of a novel IWO to the design of encoding sequences for DNA computing [J]. Computers and Mathematics with Applications, 2009, 57(11-12): 2001-2008.

[17] MARYAM Q, MAHDI S. Solving quadratic assignment problem (QAP) using invasive weed optimization algorithm [J]. Journal of Industrial Engineering, 2011, 45: 113-125.

[18] 张氢,陈丹丹,秦仙蓉,等.杂草算法收敛性分析及其在工程中的应用[J].同济大学学报:自然科学版,2010,38(11):1689-1693.

[19] 刘燕,焦永昌,张亚明,等.混合入侵杂草算法用于阵列天线波束赋形[J].西安电子科技大学学报,2014,41(4):94-99.

[20] 黄霞,叶春明,曹磊.一种混沌变异的入侵杂草优化算法及性能仿真[J].系统仿真学报,2016,28(8):1732-1739.