第 42 卷 第 6A 期
2015 年 6 月
计 算 机 科 学
Computer Science
Vol. 42 No. 6A
June 2015

基于相对细化量的粗糙集属性约简算法

徐 婕 郭 明
(湖北大学计算机与信息工程学院 武汉 430062)

摘 要 属性约简是指将信息表中不影响决策或者分类的多余属性去掉,是粗糙集理论研究中的一个核心内容。现已证明寻找信息表的最小约简是一个 NP-hard 问题。目前提出的启发式算法一般没有同时考虑算法完备性和数据噪音这两个方面。在分析了属性的重要性是与其细化能力相关的基础上,提出使用相对细化量作为启发式信息的属性约简算法 REDA。该算法能解决数据中的噪音问题,且从理论上和实例中证明是完备和有效的,而且求得最小约简的可能性要高于其它基于属性重要性的算法。实验也表明,REDA 是一个高效的属性约简算法,能够有效地降低后继工作的时间和空间复杂度。

关键词 粗糙集,属性约简,启发式,完备,相对细化量
中图法分类号 TP181 文献标识码 A

Attribute Reduction Algorithm Based on Relative Refinement Capacity

XU Jie GUO Ming
(School of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China)

Abstract Attribute reduction is one of the key problems in the research field of rough set theory, which can eliminate unnecessary attributes of the information table. However, the number of possible subsets is always very large when the number of attributes N is large, because there are 2^N subsets for N attributes. Hence exhaustively examining all subsets of attributes for finding the minimal reduction is a NP-hard problem. Many heuristic algorithms are proposed to solve this difficulty. But most of them have not considered two problems at the same time which are the completeness and the data noise. An efficient algorithm named REDA that can process data noises was proposed in this paper. Firstly, the feature that the significance of attribute is related to the property of its refinement was analyzed. Then the relative refinement capacity was employed as the heuristic information. The completeness and the correctness of REDA were proved from the theoretic analysis. The examples show the minimal attribute reduction can be found by REDA in some cases with a higher probability than that gained by other approaches. Finally, the experiment indicates REDA is an efficient and complete attribute reduction algorithm, which can reduce the temporal and spatial complexity for the next work.

Keywords Rough set, Attribute reduction, Heuristic, Complete, Relative refinement capacity

1 引言

粗糙集作为一种处理不精确、不确定与不完全数据的数学工具,最初由 Pawlak 于 1982 年提出[1]。近年,其已经广泛应用于机器学习与知识发现、数据挖掘、决策支持与分析等领域。

属性约简是粗糙集理论的核心内容之一。它的目的是要求在不改变分类或决策能力的前提下,用尽量少数目的属性集合将论域中的对象分开,删除多余属性,从而降低后继工作的时间、空间复杂度。但是,已经证明求出所有最小约简是一个 NP-hard 问题[2],通常只能采取启发式算法来解决。

目前,用于解决属性约简的启发式算法有 3 类:(1)基于属性的重要性[3-7];(2)基于区分矩阵[8-10];(3)基于遗传算法[11,12]。

文献[3,4]分别使用知识量和熵作为衡量属性重要性的指标,可以处理数据中的噪音问题,但不是完备的算法,不能保证一定能得到最小约简。

文献[5]使用正区域在论域中所占的比例作为衡量属性重要性的指标,是一个完备、高效的算法。但存在 3 个问题:第一,不能解决数据噪音;第二,与本文提出的算法相比,其启发性信息不能很好地体现各个属性的细化能力,因此在某些情况下,虽然能得到最小约简,但不是数目较少的一个;第三,如果信息表中各属性的正区域与核为空,算法不能正常执行。

目前,大部分约简算法是不完备[3,4,6-12]的,不能保证一定可以得到 Pawlak 约简[7],或者不能处理数据的噪音[5,8-12]。本文提出的算法使用相对细化量作为衡量属性重要性的指标,能解决数据的噪音问题,且被证明是完备和高效的。

本文第 2 节介绍粗糙集的基本概念;第 3 节给出算法的

本文受国家自然科学基金青年项目(61403132)资助。
徐 婕(1975—),女,副教授,主要研究方向为数据挖掘、机器学习、计算机网络,E-mail:frangipani@hubu.edu.cn;郭 明(1986—),男,硕士生,主要研究方向为数据挖掘。

理论依据、设计思想以及完备性说明，第4节列出实验结果和分析，最后给出结论和未来工作。

2 粗糙集的基本概念

下面对本文用到的粗糙集的基本概念进行简单描述，相关精确定义请参见文献[1,13,14]。

定义1 一个信息系统 S 可以表示为 $S=\langle U,Q,V,f\rangle$，其中，U 是对象的集合，即论域；Q 是属性集合；$V=\bigcup_{q\in Q}V_q$，V_q 表示属性 q 的值域；$f:U\times Q\to V$ 是一个信息函数，它指定 U 中每一个对象 x 的属性值，即对 $x\in U,q\in Q$，有 $f(x,q)\in V_q$。

定义2 在信息系统 S 中，对于每个属性子集 $P\subseteq Q$，可以定义一个不可区分关系 $IND(P)$：$IND(P)=\{(x,y)\in U\times U:\forall p\in P,p(x)=p(y)\}$，显然 $IND(P)$ 是一个等价关系，对象 x 在属性集 P 上的等价类定义为 $[x]_{IND(P)}=\{y:y\in U,yIND(P)x\}$。

3 以相对细化量为启发的约简算法

3.1 相对细化量的概念及性质

属性约简的目的就是用最少数目的属性子集 P 将论域 U 中的对象分开，所得结果和属性集合 Q 在论域 U 上操作的结果相同。这就需要逐步求出最能细化对象的属性子集，本文引入相对细化量来衡量这种细化程度，细化程度越高的属性子集其重要性也越大。

规则1 如果 $P\subseteq Q,U/IND(P)=\{U_1,U_2,\cdots,U_m\}$，则对 $U/IND(Q)$ 中的各等价类做一次调整，得到 $U/IND(Q)=\{U_1',U_2',\cdots,U_m',\cdots,U_n'\}$，调整规则为：对于 $1\leqslant i\leqslant m$，等价类 U_i' 和等价类 U_i 中的第一个元素相同（本文所有经过细化的等价类均按照规则1进行调整）。

定理1 规则1是可行的。

证明：$\because P\subseteq Q$，对于 $\forall U_i'\in U/IND(Q)(1\leqslant i\leqslant n)$，$\exists U_j\in U/IND(P)(1\leqslant j\leqslant m)$，使得 $U_i'\subseteq U_j$，即每个 U_i' 都可以看成是某个 U_j 分裂而得，则对于每个 U_j 中的第一个元素，一定有某个 U_i' 的第一个元素相对应，即规则1是可行的。证毕。

定义3 如果 $P\subseteq Q,U/IND(P)=\{U_1,U_2,\cdots,U_m\}$，其中各等价类的元素数目为 $|U_1|,|U_2|,\cdots,|U_m|$，$U/IND(Q)=\{U_1',U_2',\cdots,U_m',\cdots,U_n'\}$，其中等价类的顺序按规则1进行了调整且各等价类的元素数目为 $|U_1'|,|U_2'|,\cdots,|U_m'|,\cdots,|U_n'|$，则将 $D(P|Q)$ 记为 P 相对于 Q 的相对细化量且 $D(P|Q)=\sum_{i=1}^{m}(|U_i|-|U_i'|)^2+\sum_{i=m+1}^{n}|U_i'|^2$。

相对细化量有如下性质：(1) $D(Q|Q)=0$；(2) 如果 $U/IND(P)=U/IND(Q)$，则 $D(P|Q)=0$。对以上的性质进行说明：属性子集不能细化自己，即子集 Q 相对于子集 Q 的细化程度是最高的，因此 $D(Q|Q)=0$；同理，如果 $U/IND(P)=U/IND(Q)$，属性子集 P 不能细化子集 Q，即子集 P 相对于子集 Q 的相对细化量为0，则 $D(P|Q)=0$。

文中提到的属性的相对细化量如果没有特殊说明，均指属性相对于整个属性集合的相对细化量。

定义4 如果 $U/IND(P)=\{U_1,U_2,\cdots,U_m\}$，$U/IND(P^{(1)})=\{U_1,U_2,\cdots,U_{m-1}\cup U_m\}$，则称 $U/IND(P)$ 为 $U/IND(P^{(1)})$ 的一次细化，$U/IND(P^{(1)})$ 为 $U/IND(P)$ 的一次合并。

引理1 如果 $P\subseteq Q,U/IND(P)=\{U_1,U_2,\cdots,U_m\}$，$U/IND(P^{(1)})=\{U_1,U_2,\cdots,U_{m-1}\cup U_m\}$，$U/IND(Q)=\{U_1',U_2',\cdots,U_m',\cdots,U_n'\}$，则 $D(P^{(1)}|Q)-D(P|Q)>0$，即一次细化后的相对细化量减小。

证明：设 $U/IND(P)$ 中各等价类的元素数目为 $|U_1|,|U_2|,\cdots,|U_m|$，$U/IND(Q)$ 中各等价类的元素数目为 $|U_1|',|U_2'|,\cdots,|U_m'|,\cdots,|U_n'|$，则 $U/IND(P^{(1)})$ 中各等价类的元素数目为 $|U_1|,|U_2|,\cdots,|U_{m-1}|+|U_m|$。

$$D(P^{(1)}|Q)=\sum_{i=1}^{m-2}(|U_i|-|U_i'|)^2+(|U_{m-1}|+|U_m|-|U_{m-1}'|)^2+\sum_{i=m}^{n}|U_i'|^2$$

$$D(P|Q)=\sum_{i=1}^{m}(|U_i|-|U_i'|)^2+\sum_{i=m+1}^{n}|U_i'|^2$$

$$\begin{aligned}&D(P^{(1)}|Q)-D(P|Q)\\&=(|U_{m-1}|+|U_m|-|U_{m-1}'|)^2+|U_m'|^2-(|U_{m-1}|-|U_{m-1}'|)^2-(|U_m|-|U_m'|)^2\\&=2|U_m|(|U_{m-1}|-|U_{m-1}'|)+2|U_m||U_m'|\end{aligned}$$

$\because U_i'$ 由 U_i 分裂而得，即有 $|U_i|\geqslant|U_i'|>0,1\leqslant i\leqslant m$

$\therefore D(P^{(1)}|Q)-D(P|Q)>0$。证毕。

定理2 对于 $\forall q\in Q\wedge q\notin P$，有 $D(P|Q)-D(P\cup\{q\}|Q)\geqslant 0$。

证明：$\because q\in Q\wedge q\notin P,\therefore P\subset P\cup\{q\}\subseteq Q$，即 $U/IND(P\cup\{q\})$ 就是对 $U/IND(P)$ 进行了 $n(n\geqslant 0)$ 次细化，由引理1可得 $D(P|Q)-D(P\cup\{q\}|Q)\geqslant 0(n=0$ 时等号成立）。证毕。

由定理2可知，细化程度越高的属性集合，相对细化量越小，这样的属性子集也越重要，在属性约简的过程中被优先选取。

定理3 给定信息表 $S=\langle U,Q,V,f\rangle$，$P\subseteq Q$，P 是 Q 的一个约简，对于 $\forall p\in P$，当且仅当：$D(P|Q)=0,D(P-\{p\}|Q)-D(P|Q)>0$。

证明：1）必要性

如果 P 是 Q 的一个约简，那么 P 和 Q 将论域 U 分成相同的等价类，每个等价类中的元素和元素数目相同。由定义3可知：$D(P|Q)=0$；并且对于 $\forall p\in P,P-\{p\}$ 将不是 Q 的约简，由定理2可知：$D(P-\{p\}|Q)-D(P|Q)>0$。

2）充分性

如果 $P\subseteq Q,D(P|Q)=0,D(P-\{p\}|Q)-D(P|Q)>0$ 成立，假设 P 不是 Q 的一个约简：

①P 中不包含 Q 的一个约简并且 $P\subseteq Q$，因此 $U/IND(Q)$ 是 $U/IND(P)$ 的 n 次细化，由定理2和相对细化量的性质可得：$D(P|Q)-D(Q|Q)>0\Rightarrow D(P|Q)>0$。

与已知矛盾，假设错误。

②P 中包含 Q 的一个约简，因此 $U/IND(Q)=U/IND(P)$，由相对细化量的性质可得：$D(P|Q)=0$，但同时一定 $\exists p\in P$，使得 $D(P-\{p\}|Q)-D(P|Q)=0$，与已知矛盾，假设错误。

综上所述，命题得证。

定理2和定理3保证了属性约简算法的正确性，是本算法的理论依据。

3.2 属性约简算法

算法 REDA：基于相对细化量的属性约简算法

输入：一个信息表 S=(U,Q,V,f) 以及噪音参数 β

输出：信息表的一个属性约简 RED

初始化：RED=Φ；n=1；P=Q

1. 求出属性集合 Q 中各属性 q_i 的等价类 U/IND(q_i) 及集合 Q 的等价类 U/IND(Q)，算出各属性相对于属性集合 Q 的相对细化量；
2. 求出 $r_n \in P$，使得 D(RED∪{r_n}|Q) 最小；
3. RED←RED∪(r_n)，P←P−(r_n)，D(RED|Q)=D(RED∪{r_n}|Q)，n=n+1；
4. 如果 D(RED|Q)>β，转 2；否则，转 5；
5. 从 RED 首部开始，由前至后，对每个属性 r_i(i=1,…,n−2) 进行判断：若 U/IND(RED)=U/IND(RED−{r_i})，则从 RED 中去掉 r_i；
6. 算法结束，输出结果 RED。

计算某个属性子集 P 的等价类的一般方法是：对论域 U 中的每个对象进行两两比较，比较它们对 P 中的每个属性取值是否相同，若相同，则属于同一个等价类。因此，步骤 1 的时间复杂度为 $O(|Q||U|^2)$。改进的方法：首先按照 P 对论域 U 进行排序，然后对排序后的论域 U 扫描一遍，即可以得到等价类，此时步骤 1 的时间复杂度可以降为 $O(|Q||U|\log|U|)$。

求 $D(RED\cup\{r_n\}|Q)$ 首先要求出 $RED\cup\{r_n\}$ 的等价类，再求出相对细化量，因此步骤 2 的时间复杂度为 $O(|Q||U|\log|U|)$。

步骤 2—步骤 4 最多循环 $|Q|$ 次，因此总的时间复杂度为 $O(|Q|^2|U|\log|U|)$。在最差的情况下，步骤 5 的时间复杂度为 $O(|Q|^2|U|\log|U|)$。

综上所述，算法 REDA 的时间复杂度为 $O(|Q|^2|U|\log|U|)$。

如果按照文献[15]对算法进行改进，采用计数排序方法对论域 U 进行排序，而不是快速排序，REDA 算法时间复杂度可以降低为：$O(|Q|^2|U|)$。

为了说明本算法是完备的，首先给出相关定义。

定义 5[7]　一个 Pawlak 约简必须满足独立性条件：对于一个不为空的集合 $P\subseteq Q$，如果 $\forall p\in P, U/IND(P)\neq U/IND(P-\{p\})$，则称 P 具有独立性，是一个 Pawlak 约简。

定义 6[7]　如果一个算法一定能找到 Pawlak 约简，则这个算法相对于 Pawlak 约简是完备的。

定理 4　算法 REDA 相对于 Pawlak 约简是完备的。

证明：本算法所求的 RED 中，细化能力最强的属性排在最前，并且强度依次减弱；但存在这样的属性 q_i，它的细化能力很强，使得 $D(RED\cup\{q_i\}|Q)$ 最小，优先被选中，但在逐步求解的过程中，其细化能力可能分散地包含在后继所选的若干个属性中，最后两个属性的细分能力没有其他后继属性包含，所以不用检查。因此求出 RED 后，可将多余属性省去，保证 RED 是最小约简。步骤 5 保证 $\forall r_i\in RED, U/IND(RED)\neq U/IND(RED-\{r_i\})$，根据定义 5、定义 6 可知本算法是完备的。证毕。

本算法根据 β 的设置，还可以处理数据噪音。当 $\beta=0$ 时，即为经典的属性约简。

3.3　实例

举例说明算法 REDA 处理过程与文献[3,5]（分别称为算法 Reduce-β 和算法 γ）中提出的算法的比较。本文对算法 γ 略加修改，使其能够用于非决策系统，默认每个对象的决策属性值均不相同。

给定两个信息表：表 1、表 2，分别用算法 REDA、算法 Reduce-β 和算法 γ 进行属性约简。

表 1　信息表 1

U	a	b	c	d
X1	1	1	1	1
X2	2	1	2	2
X3	3	2	2	3
X4	1	1	1	1
X5	2	1	2	2
X6	1	2	2	1
X7	2	2	2	1
X8	3	2	2	4
X9	1	2	2	1
X10	2	2	2	1
X11	4	3	3	1
X12	4	3	4	1

表 2　信息表 2

U	a	b	c	d	e
X1	1	1	1	1	1
X2	1	2	2	2	1
X3	1	3	3	2	1
X4	1	1	1	1	1
X5	1	2	2	2	1
X6	2	1	1	1	2
X7	2	2	1	2	1
X8	2	3	1	3	1
X9	2	1	1	1	2
X10	2	2	1	2	1
X11	1	4	1	4	2
X12	2	4	1	5	2

对于表 1：

$$U/IND(Q)=\{\{X1,X4\},\{X2,X5\},\{X3\},\{X6,X9\},\{X7,X10\},\{X8\},\{X11\},\{X12\}\}$$

算法 REDA：

步骤 1　算出各属性的相对细化量，找出值最小的属性 a 加入到 RED（如表 3）；

步骤 2　分别计算{a,b},{a,c},{a,d}的相对细化量，找出值最小的属性子集{a,b}，将属性 b 加入到 RED（如表 4）；

步骤 3　计算{a,b,c},{a,b,d}的相对细化量，找出值最小的属性子集{a,b,c}，将属性 c 加入到 RED（如表 5）；

步骤 4　将最后一个属性 d 加入 RED；

步骤 5　扫描 RED 中各个属性，得到 U/IND(RED)=U/IND(RED−{b})，删去属性 b；

步骤 6　输出最后结果 RED={a,c,d}。

算法 Reduce-β 得到：$RED=\{a,b,c,d\}$。

算法 γ 得到：$RED=\{a,c,d\}$。

由此可知，算法 Reduce-β 是不完备的。

对于表 2：

$$U/IND(Q)=\{\{X1,X4\},\{X2,X5\},\{X3\},\{X6,X9\},\{X7,X10\},\{X8\},\{X11\},\{X12\}\}$$

算法 REDA：$RED=\{a,b\}$。

算法 Reduce-β 得到：$RED=\{a,b\}$。

算法 γ 得到：$RED=\{c,d,e\}$。

虽然 3 个算法得到的都是最小约简，但是在算法 γ 中的启发式信息不能很好地表现各个属性的细化能力，得出的最小约简的属性数目大于前 2 个算法。因此，算法 REDA 能更好地减小后继工作的时间和空间复杂度。

表 3　步骤 1 属性子集相对细化量

属性子集	相对细化量
a	20
b	44
c	46
d	46

表 4　步骤 2 属性子集相对细化量

属性子集	相对细化量
a,b	4
a,c	10
a,d	10

表 5　步骤 3 属性子集相对细化量

属性子集	相对细化量
a,b,c	2
a,b.d	2

表 6　步骤 4 属性子集相对细化量

属性子集	相对细化量
a,b,c,d	0

4　实验结果和分析

选用 UCI 机器学习数据库[16]中的信息表 Flags 在 PC 机上进行实验。实验分成两个部分：第一部分分别采用算法 REDA、算法 Reduce-β 和算法 γ 进行属性约简，判断各算法是否完备；第二部分分别采用上述 3 种算法计算获得约简所需要的时间，从而比较各算法的效率。

4.1　完备性测试

Flags 中有 193 个对象和 30 个属性。首先进行数据预处理，删去不同数值的数目过大的属性 1 (*name*)、4 (*area*)、5(*population*)，然后将非数字的属性 18、29、30 的值按照 *red*=1，*green*=2，*blue*=3，*orange*=4，*gold*=5，*white*=6，*black*=7，*brown*=8 转化为数字类型。应用 REDA 约简算法，对 Flags 进行属性约简，每个单独属性的相对细化量如表 7 所列。

表 7　各属性初次相对细化量

属性	相对细化量	属性	相对细化量	属性	相对细化量
2 Landmass	6906	13 Blue	24896	22 Quarters	28324
3 Zone	12548	14 Gold	18696	23 Sunstars	31106
6 Language	5846	15 White	18632	24 Crescent	28656
7 Religion	7352	16 Black	18696	25 Triangle	15430
8 Bars	25615	17 Orange	23426	26 Icon	33416
9 Stripes	14266	18 Mainhue	22674	27 Animate	28424
10 Colours	9488	19 Circles	28306	28 Text	23232
11 Red	28324	20 Crosses	8298	29 Topleft	7692
12 Green	31106	21 Saltires	27598	30 Botringht	8849

其中属性 6 *Language* 的相对细化量最小，首先被选入 *RED* 中。同理选入其他属性，被选入的属性子集的相对细化量如表 8 所列。

表 8　最终被选中属性的相对细化量

加入属性	最小相对细化量	加入属性	最小相对细化量
6	5846	9	48
29	1329	18	30
10	287	3	6
23	178	28	2
30	84	12	0

最后约简得：

$RED=\{6,29,10,23,30,9,18,3,12,28\}$，将原来信息表 30 维属性降到 10 维，大大减小了后继识别 Flag 的工作复杂度。

采用算法 γ：

得到 $RED=\{9,10,6,30,10,7,3,18,12,13,28\}$，虽然此算法也是完备的属性约简算法，但属性维数为 11。

采用算法 Reduce-β：

得到 $RED=\{6,29,10,7,13,9,2,15,22,18,12,26\}$，其中属性 10 为冗余属性，即这个约简并不是 Pawlak 约简，因此 Reduce-β 不是完备的算法。

4.2　效率比较

分别采用算法 Reduce-β、算法 γ、算法 REDA 进行效率测试，实验结果如表 9 所列。从表 9 可以看出，算法 REDA 的效率略优于算法 γ，明显优于算法 Reduce-β。

表 9　3 种算法效率的比较

	算法 Reduce-β	算法 γ	算法 REDA
数据集	Flags		
实例数	193		
原数据数目	30		
约简后属性数目	12	11	10
是否 Pawlak 约简	否	是	是
执行时间(s)	0.15	0.06	0.04

结束语　本文用相对细化量作为衡量属性细化能力的指标，并以此作为启发式信息，提出了一种高效、完备并且具有较好抗噪性的属性约简算法。实验表明，本算法能高效地获得属性集合的约简，能更加有效地减小后继工作的时间和空间复杂度。如何解决属性约简的增量问题，如何使用粗糙集进行聚类等，都是下一步需要研究的问题。

参 考 文 献

[1] Pawlak Z. Rough set[J]. International Journal of Computer and Information Science, 1982, 11(4): 341-356

[2] Wong S K M, Ziarko W. On optimal decision rules in decision tables[J]. Bulletin of Polish Academy of Sciences, 1985, 33(11/12): 693-696

[3] 徐燕，怀进鹏，王兆其. 基于区分能力大小的启发式约简算法及其应用[J]. 计算机学报，2003，26(1)：97-103

[4] 王莎莎，江峰，王文鹏. 基于相对决策熵与加权相似性的粗糙集数据补齐方法[J]. 计算机科学，2014，41(2)：245-248

[5] 刘少辉等. Rough 集高效算法的研究[J]. 计算机学报，2003，26(5)：524-529

[6] 张迎春，王宇新，郭禾. 基于有序差别集和属性重要性的属性约简[J]. 计算机科学，2011，38(10)：243-247

[7] 叶东毅. Jelonek 属性约简算法的一个改进[J]. 电子学报，2000，28(12)：81-82

[8] 张颖淳，苏伯洪，曹娟. 基于粗糙集的属性约简在数据挖掘中的应用研究[J]. 计算机科学，2013，8(40)：223-226

[9] 周建华，徐章艳，章晨光. 改进的差别矩阵的快速属性约简算法[J]. 小型微型计算机系统，2014，35(4)：831-834

[10] 张长胜. 基于决策表的区分矩阵增量属性约简算法[J]. 计算机工程与应用，2012，48(35)：110-113

[11] 马希骜，王国胤，于洪. 决策域分布保持的启发式属性约简方法[J]. 软件学报，2014，25(8)：1761-1780

(下转第 114 页)

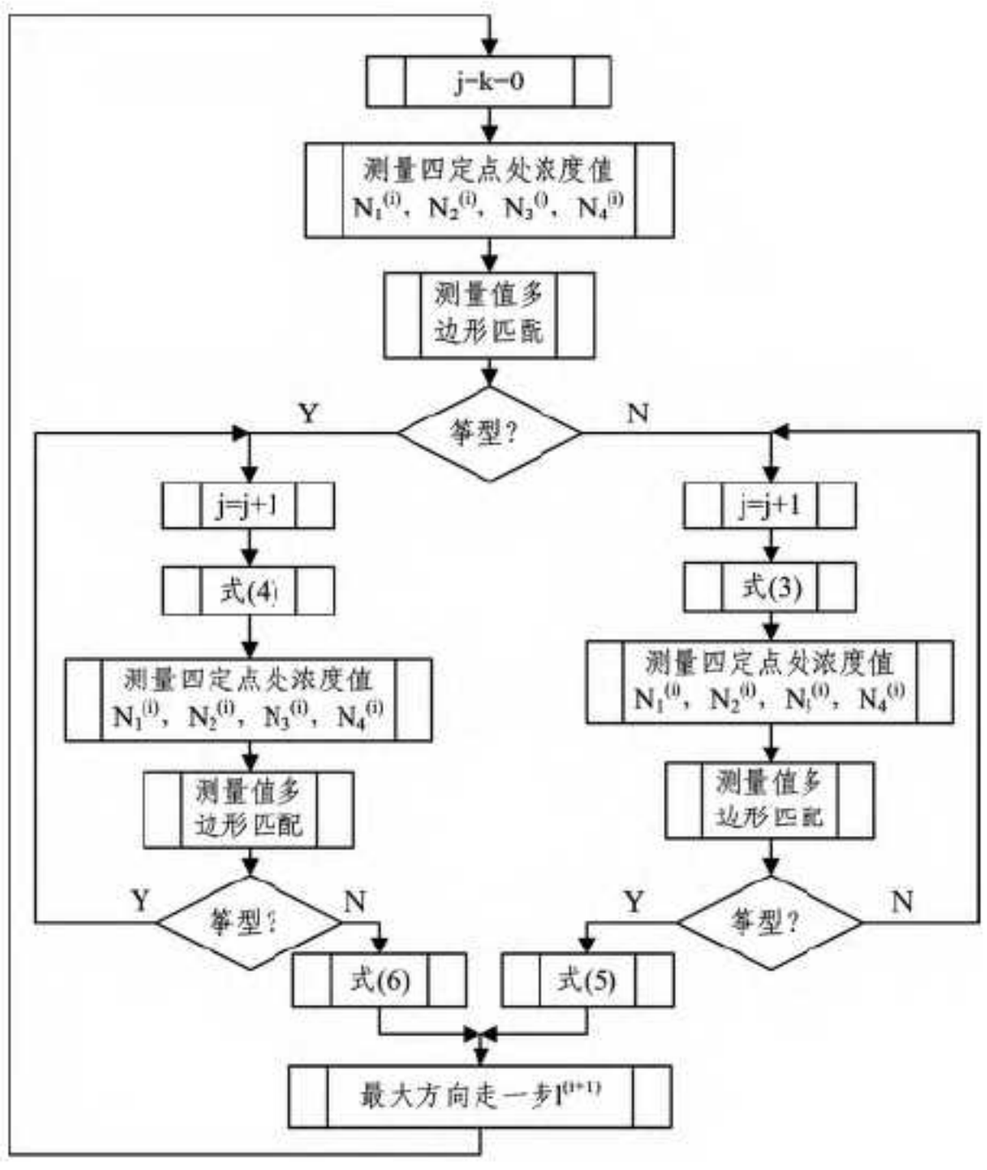

图 6　动态调整测距和步长的算法描述

3　改进策略的效果

考虑行走机构不一致性和地表不平坦影响下的搜索路径效果如图 7 所示。

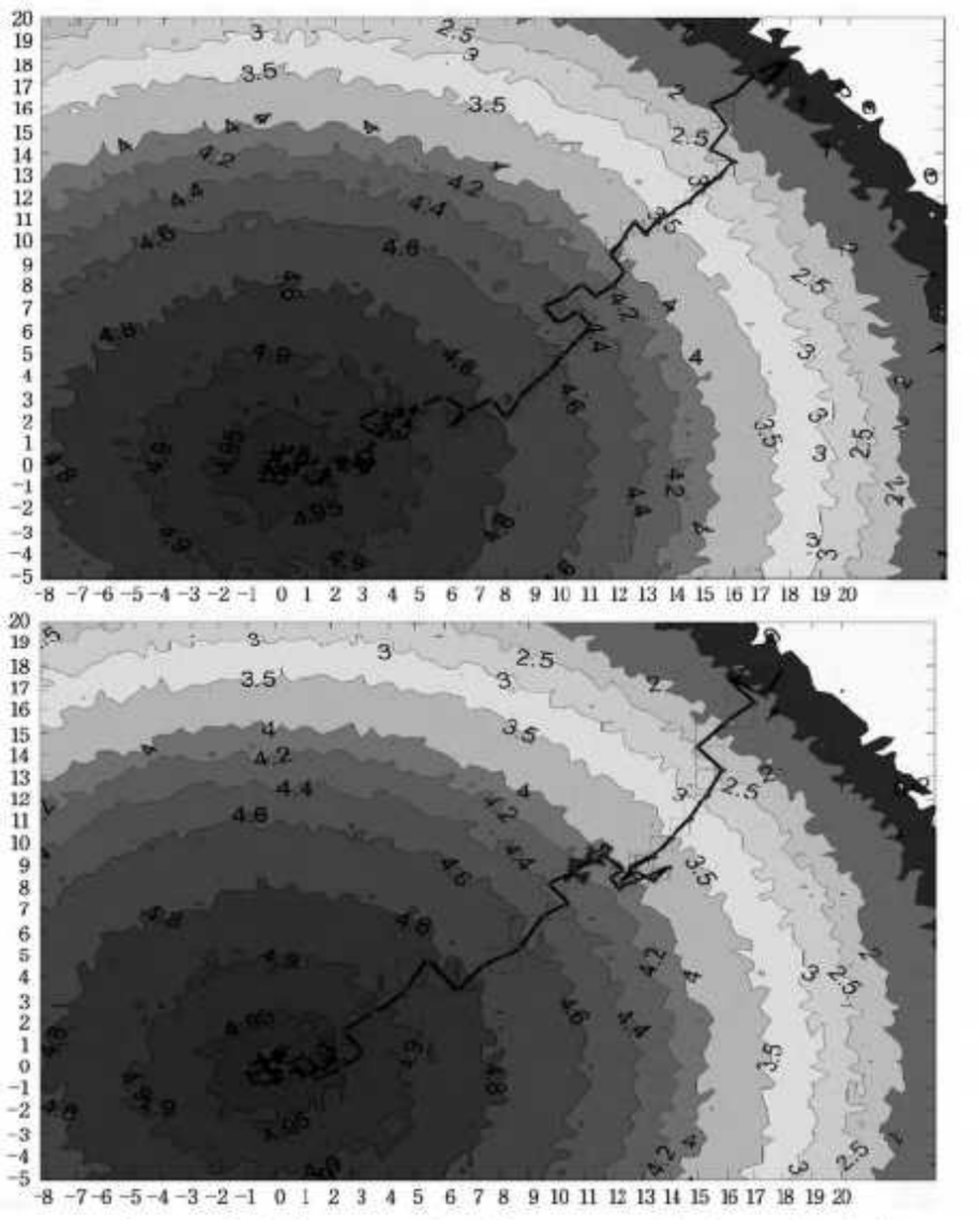

图 7　考虑行走机构不一致性和地表不平坦影响下的搜索路径

设浓度 $a=0.03698, b=0.17815$，取初始步长为 1，初始探测半径为 0.5，$d_{inc}=0.1, d_{dec}=0.05$，分别依照改进前和改进后的策略进行气体源搜索的仿真。在平坦地表情况的结果是：改进前的收敛比例是 40% 左右，而改进后的收敛比例达到 92% 左右。考虑机器人本体行走机构的不一致性和不平坦地表，在行进中随机加入左右转向的 20° 角以内偏差，以模拟其效果，结果表明收敛性能甚至收敛速度不受什么影响（见图 7）。

结束语　对四点探测最大方向跟踪策略的改进策略是：基于 4 个等距探测值的分布四边形的形状特征，调整探头测距，在探测值四边形满足特殊条件即筝型的情况下，参考上一次测距和上一步步长，对步长做出调整，并按新的步长向着最大探测值方向行走一步。仿真表明：改进策略对原有策略在气味烟羽发现、气味跟踪、气体源定位 3 个主要性能上均有明显改进。当然，改进策略对硬件实现的要求是机器人携带的探头的臂长应具有可伸缩功能。

参 考 文 献

[1] Holland O, Melhuish C. Some Adaptive Movements of Animates with Single Symmetrical Sensor[C] // From Animals to Animates 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior. Cambridge: MIT Press, 1996: 55-64

[2] Russell R A. Chemical Source Location and the RoboMole Project [C] // Australasian Conference on Robotics and Automation, CD-ROM Proceedings. 2003

[3] Lilienthal A, Reimann D, Zell A. Gas Source Tracing with a Mobile Robot Using an Adapted Moth Strategy[C] // Proceedings of American Mathematical Society 2003. Berlin, New York: Springer, 2003: 150

[4] Lilienthal A, Dukett T. Building Gas Concentration Gridmaps with a Mobile Robot[J]. Robotics and Autonomous Systems, 2004, 48(1): 3-16

[5] 孟庆浩，李飞. 主动嗅觉研究现状[J]. 机器人，2006，28(1)：89-96

[6] 孟庆浩，李飞，张明路，等. 湍流烟羽环境下多机器人主动嗅觉实现方法研究[J]. 自动化学报，2008，34(10)：1281-1290

[7] 李俊彩，孟庆浩，梁琼. 基于进化梯度搜索的机器人主动嗅觉仿真研究[J]. 机器人，2007，29(3)：234-238

[8] 张小俊，张明路，孟庆浩，等. 一种基于动物捕食行为的机器人气味源定位策略[J]. 机器人，2008，30(3)：265-272

[9] 蒋萍，孟庆浩，曾明，等. 融合机器人视/嗅觉信息的室内气体源识别[J]. 高技术通讯，2011，21(8)：867-872

[10] 王阳，孟庆浩，李腾，等. 室内通风环境下基于模拟退火算法的单机器人气味源定位[J]. 机器人，2013，35(3)：283-291

[11] 王俭，季剑岚，陈卫东. 基于行为特征的机器人变步长气味源搜索算法[J]. 系统仿真学报，2009，24(17)：5427-5430，5435

[12] 唐波，王俭，季剑岚. 一种机器人气味源跟踪算法的性能分析[J]. 苏州科技学院学报：工程技术版，2008，21(2)：68-71

(上接第 97 页)

[12] 焦娜. 基于相容粗糙集的改进的基因特征选择方法[J]. 计算机科学，2013，40(Z6)：125-128，140

[13] 张文修，等. Rough 集理论与方法[M]. 北京：科学出版社，2001

[14] 王国胤. Rough 集理论和知识获取[M]. 西安：西安交通大学出版社，2001

[15] 徐章艳，刘作鹏，杨炳儒，等. 一个复杂度为 $\max(O(|C||U|), O(|C|2|U/C|))$ 的快速属性约简算法[J]. 计算机学报，2006，29(3)：391-399

[16] Newman D J, Hettich S, Blake C L, et al. UCI repository of machine learning databases[OL]. http://www.ics.uci.edu/~mlearn/MLRepository.html