ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.17, No.3, March 2006, pp.364−370
DOI: 10.1360/jos170364

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

求解多目标最小生成树问题的改进算法*

陈国龙[1,2+], 郭文忠[2], 涂雪珠[2], 陈火旺[1]

[1](国防科学技术大学 计算机学院,湖南 长沙 410073)
[2](福州大学 数学与计算机科学学院,福建 福州 350002)

An Improved Algorithm to Solve the Multi-Criteria Minimum Spanning Tree Problem

CHEN Guo-Long[1,2+], GUO Wen-Zhong[2], TU Xue-Zhu[2], CHEN Huo-Wang[1]

[1](School of Computer, National University of Defense Technology, Changsha 410073, China)
[2](Institute of Mathematics and Computer Science, Fuzhou University, Fuzhou 350002, China)
+ Corresponding author: Phn: +86-591-87893316, Fax: +86-591-83713866, E-mail: cgl@fzu.edu.cn, http://www.fzu.edu.cn

Chen GL, Guo WZ, Tu XZ, Chen HW. An improved algorithm to solve the multi-criteria minimum spanning tree problem. *Journal of Software*, 2006,17(3):364−370. http://www.jos.org.cn/1000-9825/17/364.htm

Abstract: The multi-criteria minimum spanning tree (mc-MST) problem is a typical NP-hard problem. An algorithm to enumerate the set of Pareto optimal spanning trees on some mc-MST instances was put forward by Zhou and Gen, but it does not guarantee returning all the Pareto optimal solutions. To settle this problem, an improved algorithm is developed and also proved to be able to find all the true Pareto optimal solutions in this paper. The new algorithm adds some conditions in the elimination of subtrees. Simulation results show that the new algorithm can find all the Pareto optimal solutions and also show that the new algorithm has potential usage in practice.
Key words: minimum spanning tree; Pareto optimal

摘 要: 多目标最小生成树问题是典型的NP问题,Zhou和Gen提出了一种用于计数多目标最小生成树问题的所有非劣最优最小生成树的算法,但该算法无法保证能够找到所有非劣最优最小生成树.针对此问题,提出一种改进的计数算法,并定性说明改进算法能够找到问题的所有非劣最优最小生成树.改进算法在进行子树剔除时增加了一些条件.模拟实验结果表明,改进后的计数算法能够找到所有的非劣最优解.这也说明该算法具有应用的潜力.
关键词: 最小生成树;非劣最优解
中图法分类号: TP301 **文献标识码**: A

最小生成树问题(minimum spanning tree,简称MST)是构造一个带权图的最小代价生成树,在网络优化中具有举足轻重的作用.目前,图论中已有一些经典的求解最小生成树算法,如Kruskal算法、Prim算法等[1].经典的

* Supported by the National Natural Science Foundation of China under Grant No.60172017 (国家自然科学基金); the Natural Science Foundation of Fujian Province of China under Grant No.A0410010 (福建省自然科学基金); the Natural Science Foundation of Fujian Provincial Educational Commission of China under Grant No.K03012 (福建省教育厅科技三项科研项目)

Received 2004-04-15; Accepted 2005-03-10

最小生成树算法,其计算复杂度为多项式时间,但当给 MST 增加约束条件或问题目标为多个时,相应的问题称为多目标最小生成树问题(multi-criteria MST,简称 mc-MST),而且问题本身也变成了一个 NP 完全问题.因此,近年来许多启发式算法应运而生,其中最具代表性的算法是 Zhou 和 Gen[2]所提出的算法.

文献[2]中的算法是以随机生成的 10 个 50 顶点的双目标 mc-MST 问题作为算法的性能测试问题,并将测试结果与所提出的计数图的所有最小生成树的方法得到的"所有非劣最小生成树"相比较,测试结果称算法找到了问题的所有非劣最优解.实际上,Zhou 和 Gen 提出的计数算法所求解到的非劣最优解是不精确的,这个问题 Knowles 在其博士论文中进行了反证[3],然而 Knowles 也没有给出精确的算法.为此,我们对 Zhou 和 Gen 提出的计数算法作了改进,并定性说明了该改进算法能够找到问题的所有非劣最优解.

1 mc-MST 问题

1.1 MST

设有一个无向图 $G=(N,E,W)$,其中 $W=\sum_{e\in E} w_e$ 为权函数,若树 $T=(N,E_T,W_T)$包含了图 G 的所有顶点,则 T 称为 G 的一个生成树或支撑树,$W_T=\sum_{e\in E_T} w_e$ 为 T 的权.图 G 的生成树并不唯一,权最小的生成树相应地称为 G 的最小生成树.在实际工程问题中,最终可以转化为求解图的最小生成树的那类问题称为 MST 问题.

1.2 Mc-MST

设有一个连通的无向图 $G=(V,E)$,其中 $V=\{v_1,v_2,\ldots,v_n\}$是一个有限集合,代表图 G 的顶点,$E=\{e_{1,2}, e_{1,3},\ldots,e_{i,j},\ldots,e_{n-1,n}\}$, $e_{i,j}=\begin{cases}1, & \text{if } v_i,v_j\text{之间有边}\\ 0, & \text{otherwise}\end{cases}$ $(i=1,2,\ldots,n-1,j=i+1,\ldots,n)$,为图 G 的边集合.若边 $e_{i,j}$存在,则有 m 个值为正数的属性与之相对应,用 $w_{i,j}=\{w_{i,j}^1,w_{i,j}^2,\ldots,w_{i,j}^m\}$ 表示.实际问题中,$w_{i,j}^k\ (k=1,2,\ldots,m)$ 可以是距离、代价等.

令 $x=(x_{1,2}x_{1,3}\ldots x_{i,j}\ldots x_{n-1}x_n)$, $x_{i,j}=\begin{cases}1, & \text{if } e_{i,j}=1\text{且被选中}\\ 0, & \text{otherwise}\end{cases}$,$i=1,2,\ldots,n-1;j=i+1,\ldots,n$ 表示图 G 的一棵生成树,X 为所有 x 的集合,则 mc-MST 问题描述为:

$$\left.\begin{array}{l}\min\ f_1(x)=\sum w_{i,j}^1 x_{i,j};\\ \min\ f_2(x)=\sum w_{i,j}^2 x_{i,j};\\ \quad\vdots \qquad\qquad\qquad i=1,2,\ldots,n-1,\ j=i+1,\ldots,n\\ \min\ f_m(x)=\sum w_{i,j}^m x_{i,j};\\ \text{s.t. } x\in X\end{array}\right\}\tag{1}$$

其中 $f_i(x)$为问题中需要最小化的第 i 个目标.

2 Zhou 和 Gen 提出的计数算法及其分析

Zhou 和 Gen 提出了一种求解多目标最小生成树问题的遗传算法,该算法中的染色体编码采用 Prüfer 数编码,而选择策略则是用 Srinivas 和 Deb 提出的个体非劣性排序方法[4].为了评价该算法的有效性,Zhou 和 Gen 相应地给出了一种计数算法,用于计数 mc-MST 中的所有非劣最优解.因篇幅所限,Prüfer 编码过程和计数算法过程省略,具体可见参考文献[2].

定义 1[5]. Prüfer 给出了 n 个节点的生成树和长度为 $n-2$ 的串之间的一个对应关系.而习惯地用数字来标志这些节点,所形成的数字串称为 Prüfer 数.

定义 2[4]. 非劣性排序过程的基本思想是:在选择过程中,采用随机选择方法加强较优的个体和小生境方法维持较优的个体的子种群的稳定性.相应地,称采用此思想的方法为非劣性排序方法.

定义 3(优劣性,dominance/inferiority)[6]. 称目标向量解 $u=(u_1,u_2,\ldots,u_n)$优于 $v=(v_1,v_2,\ldots,v_n)$,记为 $up<v$,如果 u 部分小于 v,也就是 $\forall i\in\{1,2,\ldots,n\}, u_i\le v_i \wedge \exists i\in\{1,2,\ldots,n\}, u_i \prec v_i$.反过来,称目标向量解 v 劣于 u,记为 $vp<u$.

定义 4(非劣最优解,Pareto optimal)[6]. 决策变量 $x_{u\in R^n}$ 称为多目标问题的非劣最优解,当且仅当不存在决策变量 $x_{v\in R^n}$,使得相应的目标向量 $v=f(x_{v\in R^n})=(v_1,v_2,\ldots,v_n)$ 优于 $u=f(x_{u\in R^n})=(u_1,u_2,\ldots,u_n)$.

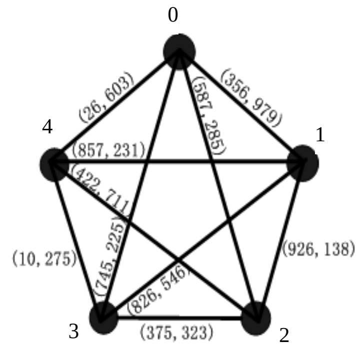

Fig.1 A graph of five vertexes

图 1 含有 5 个顶点的完全图

文献[2]中,作者利用计数算法计数一些较小规模的 mc-MST 问题,并用文中所提出的遗传算法求解到的非劣最优生成树的数目与用计数算法找到的非劣最优生成树数目的比例来评价算法的性能.然而,Knowles 在其博士论文[3]中举例反证了该计数过程的不精确性,其中的一个反例及分析如图 1 和表 1 所示.

Knowles 的反例有力地说明了文献[2]提出的计数一个无向连通图 G 的所有非劣最优生成树的方法不但不能保证找到所有非劣最优解,而且找到的生成树还可能不是非劣最优的.我们经过研究发现,文献[2]的计数方法不能保证找到的是所有非劣最优解,问题出在子树剔除上.例如,T_1 和 T_2 为两棵 2 条边的生成子树,如果 $T_1<T_2$,并不能就此认为由 T_2 派生的生成树就一定不会是非劣最优生成树,从而把 T_2 从子树集合中删除掉.

Table 1 The procedure and result for algorithm of Zhou and Gen

表 1 Zhou 和 Gen 方法的求解过程

0-1(356,979)**			
0-2(587,285)	0-1(943,1264)**		
	0-3(1332,510)*		
	0-4(613,888)*		
	1-2(1513,423)	0-3(2258,648)	0-4(2284,1251)**
			1-4(3115,879)*
			2-4(2680,1359)**
			3-4(2268,923) (1)
		0-4(1539,1026)*	
		1-3(2339,969)**	
		1-4(2370,654)**	
		2-3(1888,746)*	
		2-4(1935,1134)**	
	2-3(962,608)*		
	2-4(1009,996)**		
0-3(745,225)	0-1(1101,1204)**		
	0-2(1332,510)**		
	0-4(771,828)**		
	1-3(1571,771)**		
	2-3(1120,548)**		
	3-4(755,500)	0-1(1111,1479)*	
		0-2(1342,785)	0-1(1698,1764)*
			1-2(2268,923)*
			1-3(2168,1331)*
			1-4(2199,1016)*
		1-3(1581,1046)**	
		1-4(1612,731)	0-2(2199,1016)*
			1-2(2538,869) (2)
			2-3(1987,1054)R
			2-4(2034,1442)**
		2-3(1130,823)	0-1(1486,1802)*
			1-2(2056,961) (3)
			1-3(1956,1369)*
			1-4(1987,1054) (4)
		2-4(1177,1211)**	
0-4(26,603)	0-1(382,1582)**		
	0-2(613,888)**		
	0-3(771,828)*		
	1-4(883,834)**		
	2-4(448,1314)**		
	3-4(36,878)	0-1(392,1857)	0-2(979,2142) (5)
			1-2(1318,1995)*
			2-3(767,2180) (6)
			2-4(814,2568)**

续表 1

		0-2(623,1163)	0-1(979,2142)R
			1-2(1549,1301) (7)
			1-3(1449,1709)*
			1-4(1480,1394)*
		1-3(862,1424)**	
		1-4(893,1109)	0-2(1480,1394)*
			1-2(1819,1247) (8)
			2-3(1268,1432) (9)
			2-4(1315,1820)**
		2-3(411,1201)	0-1(767,2180)R
			1-2(1337,1339) (10)
			1-3(1237,1747) (11)
			1-4(1268,1432)R
		2-4(458,1589)**	

3 算法的改进

由上所述,我们根据生成树的特点,在进行两棵生成子树的Pareto优劣性比较时,增加了一些条件,只有两棵生成子树满足以下条件时才将其中的劣解剔除,否则认为不能从两棵生成子树之间的 Pareto 优劣性关系来确定由它们生长成的最终生成树之间的Pareto优劣关系.

引理. 假设无向连通图 $G=(V,E)$.其中,$V=\{v_1,v_2,\ldots,v_n\}$,$E=\{e_1,e_2,\ldots,e_l\}$,S为应用文献[2]中的计数算法产生的生成子树集合,其中的每一个生成子树 s 具有 k 条边$(1\le k<n-1)$,对于任意的 $s_1,s_2\in S,s_1=(V_1,E_1),s_2=(V_2,E_2),V_{12}=V_1\cap V_2,E_{12}=E_1\cap E_2,V^*_{12}$为与$E_{12}$相连的顶点集,如果$s_1<s_2$且满足以下条件之一,则由$s_2$生长成的生成树一定不会Pareto优于由$s_1$生长成的生成树.

Ⅰ. $V_1=V_2$,两子树具有相同的顶点集合;

Ⅱ. $k=1$,两子树都只有一条边且有一个共同顶点;

Ⅲ. s_1,s_2为同一子树增加一条边后的生成子树或者 $V_{12}=V^*_{12}$ 且 s_1,s_2只有一个非共同顶点.

证明:Ⅰ. 令 $V_{12}=V_1=V_2=\{v_1,v_2,\ldots,v_i\}$, $\overline{V}_{12}=V-V_{12}$,根据文献[2]的计数算法过程,每次都是添加分别与 V_{12}, $\overline{V}_{12}$ 中的一个顶点相连的边.s_1,s_2具有相同的顶点集合,如果能够添加边 $v_pv_q,p\in\{i+1,i+2,\ldots,n\},q\in\{1,2,\ldots,i\}$到 s_2 中不构成回路,则同样能够将边 v_pv_q 添加到 s_1 中.

因为 $V_1=V_2=\{v_1,v_2,\ldots,v_i\}$,

所以 $\forall s_2'=(V_2',E_2'),V_2'=V_2\cup\{v_q\},E_2'=E_2\cup\{v_pv_q\}$,

$\exists s_1'=(V_1',E_1'),V_1'=V_1\cup\{v_q\},E_1'=E_1\cup\{v_pv_q\},p\in\{1,2,\ldots,i\},q\in\{i+1,i+2,\ldots,n\}$,

又因为 $s_1\prec s_2$,所以 $s_1'\prec s_2'$.

同理,对任一由 s_2' 添加一条边后生成的子树一定劣于由 s_1' 再添加该条边后生成的子树.

Ⅱ. 不妨设 $s_1=(V_1,E_1)=(\{v_1,v_2\},\{v_1v_2\}),s_2=(V_2,E_2)=(\{v_1,v_3\},\{v_1v_3\})$.

由s_2不断生长成生成树的过程为:每次添加一条分别与 $V_2,\overline{V}_2$ 中的一个顶点相连的边到E_2中,同时把与该边相连的$\overline{V}_2$中的那个顶点添加到V_2中再从$\overline{V}_2$中删除,假设增加顶点v_2前s_2的边集为$E_2,V_2=\{v_1\}\cup V',V'=\{v_3\}\cup V'',V''\subseteq(V-\{v_1,v_2,v_3\})$.

(1) 如果正要添加的边为 v_1v_2,即 $s_2'=(V_2',E_2')=(V_2\cup\{v_2\},E_2\cup\{v_1v_2\})$.显然, $\exists s_1'=(V_1',E_1')$ $V_1'=V_2',E_1'=E_1\cup E_2=\{v_1v_2\}\cup E_2=E_2'$,即 $s_1'=s_2'$.

(2) 如果正要添加的边为 $v_pv_2,p\in V'$

1) 若 v_2 到 v_3 的路不经过 v_1,即 $s_2'=(V_2',E_2')=(V_2\cup\{v_2\},E_2\cup\{v_1v_2\})$,$p\in V'$,则根据生成树的性质,可知 $\exists s_1'=(V_1',E_1')=(V_2',E_1\cup(E_2-\{v_1v_3\}))$.又因为 $s_1\prec s_2$,所以 $s_1'\prec s_2'$.

2) 若v_2到v_3的路经过v_1,不妨假设v_2到v_1的路径上经过的顶点依次为$v_2\to v_p\to\ldots\to v_4\to v_1$,令$s_3=(V_3,E_3)=(\{v_1,v_4\},\{v_1v_4\})$.

如果 $s_1\prec s_3$,则 $\exists s_1'=(V_1',E_1')=(V_2',E_1\cup(E_2-\{v_1v_4\}))$,使得 $s_1'\prec s_2'$;

否则 $\exists s_3' = (V_3', E_3') = (V_2', E_3 \cup (E_2 - \{v_1 v_4\}) = (V_2', E_2'))$,即该生成子树可由 s_3 派生出来.

III. 令 $V_1=V_{12}\cup\{v_p\}, V_2=V_{12}\cup\{v_q\}, V_{12}=\{v_1,v_2,\ldots,v_s\}, E_1=E_{12}\cup\{v_{s_1} v_p\}, E_2=E_{12}\cup\{v_{s_2} v_q\}$.同样,由 s_2 不断生长成生成树的过程为:每次添加一条分别与 $V_2,\overline{V}_2$ 中的一个顶点相连的边到 E_2 中,同时把与该边相连的 $\overline{V}_2$ 中的那个顶点添加到 V_2 中再从 $\overline{V}_2$ 中删除,则向 V_2 增加顶点 v_p 后, $s_2' = (V_2', E_2') = (V_s \cup \{v_q, v_p\} \cup V_2'', E_2')$, $V_2'' \subseteq (V - V_1 - \{v_p\})$,根据生成树的性质,新增加的边的另一端点必定不属于 V_s,因此, $\exists s_1' = (V_1', E_1') = (V_2', (E_2' - \{v_q v_{s2}\}) \cup \{v_p v_{s1}\})$.

又因为 $s_1 \prec s_2$,所以 $s_1' \prec s_2'$.

本文所提出的计数图的所有 Pareto 最优生成树的改进算法具体步骤是:

步骤 1. 任意选择一个顶点作为出发点,不妨假设选择 v_1 作为开始顶点,检查所有与 v_1 相连的边,将这些边中的 Pareto 最优边加入到子树集 S 中;

步骤 2. for each subtree $s \in S$;

逐一检查所有与 s 中的任一顶点相连的边 e_i,如果加入边 e_i 不会使 s 存在回路,则把由 s 和 e_i 构成的新的一棵子树放入 S^*中;

步骤 3. 对于 S^*中任意两棵生成子树 s_1,s_2,如果$(s_1<s_2)\&\&((V_1=V_2)||(k=1,|V_s|=1)||(V_s=V_{es},|V_1|-|V_s|=1))$,则把 s_2 从 S^*中删除;

步骤 4. $S \leftarrow S^*, S^* \leftarrow \varnothing$;

步骤 5. 如果 S 中的子树边数小于 $n-1$,则转步骤 2,否则返回 S.

Knowles 列举了一些例子反证了文献[2]中计数算法的不精确性,图 1 所示的是其中一个例子.为了进一步说明本文改进算法的有效性,本文给出文献[2]中计数算法和本文改进算法求解图 1 的实验结果,见表 1、表 2.

表中"*"表示该子树劣于其他子树派生出来的子树,而"**"表示了该子树劣于同一子树派生出来的子树.从求解结果我们可以看出,Zhou 和 Gen 的方法得到 11 个解;本文方法找到了 12 个解.比较两个表中的解,显然 Zhou 和 Gen 的方法只找到了 10 个真正的非劣最优解,其中第 4 个解并不是非劣最优解,而本文算法找到了 12 个解,可用穷举法验证这 12 个解都是非劣最优解,而且找到了全部非劣最优解.经典的最小生成树算法的算法复杂度为多项式时间,而多目标最小生成树却是一个 NP 完全问题,即对于一个完全图 G,过 n 个顶点的树的数目为 n^{n-2},其算法复杂度相应为 $O(n^{n-2})$.

为此,本文改进算法的目的是修正 Zhou 和 Gen 的计数算法所存在的问题,以期取代原有算法作为一种评价其他启发式算法性能的工具.

Table 2 The procedure and result for the new algorithm

表 2 本文改进算法的求解过程

0-1(356,979)**			
0-2(587,285)	0-1(943,1264)**		
	0-3(1332,510)*		
	0-4(613,888)*		
	1-2(1513,423)	0-3(2258,648)	0-4(2284,1251)**
			1-4(3115,879)*
			2-4(2680,1359)**
			3-4(2268,923) (1)
		0-4(1539,1026)	0-3(2284,1251)*
			1-3(2365,1572)**
			2-3(1914,1349)**
			3-4(1549,1301)R
		1-3(2339,969)**	
		1-4(2370,654)**	
		2-3(1888,746)	0-4(1914,1349)**
			1-4(2745,977)*

续表 2

			2-4(2310,1457)**
			3-4(1898,1021) (2)
		2-4(1935,1134)**	
	2-3(962,608)	0-1(1318,1587)**	
		0-4(988,1211)**	
		1-2(1888,746)R	
		2-4(1384,1319)**	
		1-3(1788,1154)**	
		3-4(972,883)	0-1(1328,1862)*
			1-2(1898,1021)*
			1-3(1798,1429)*
			1-4(1829,1114) (3)
	2-4(1009,996)**		
0-3(745,225)	0-1(1101,1204)**		
	0-2(1332,510)**		
	0-4(771,828)**		
	1-3(1571,771)**		
	2-4(1120,548)**		
	3-4(755,500)	0-1(1111,1479)*	
		0-2(1342,785)	0-1(1698,1764)*
			1-2(2268,923)*
			1-3(2168,1331)*
			1-4(2199,1016)*
		1-3(1581,1046)**	
		1-4(1612,731)	0-2(2199,1016)*
			1-2(2538,869) (4)
			2-3(1987,1054)*
			2-4(2034,1442)**
		2-3(1130,823)	0-1(1486,1802)*
			1-2(2056,961) (5)
			1-3(1956,1369)*
			1-4(1987,1054)*
		2-4(1177,1211)**	
0-4(26,603)	0-1(382,1582)**		
	0-2(613,888)**		
	0-3(771,828)*		
	1-4(883,834)**		
	2-4(448,1314)**		
	3-4(36,878)	0-1(392,1857)	0-2(979,2142) (6)
			1-2(1318,1995)*
			2-3(767,2180) (7)
			2-4(814,2568)**
		0-2(623,1163)	0-1(979,2142)R
			1-2(1549,1301) (8)
			1-3(1449,1709)*
			1-4(1480,1394)*
		1-3(862,1424)**	
		1-4(893,1109)	0-2(1480,1394)*
			1-2(1819,1247) (9)
			2-3(1268,1432) (10)
			2-4(1315,1820)**
		2-3(411,1201)	0-1(767,2180)R
			1-2(1337,1339) (11)
			1-3(1237,1747) (12)
			1-4(1268,1432)R
		2-4(458,1589)**	

4 结束语

本文首先在 Knowles 的分析基础上指出了 Zhou 和 Gen 提出的计数算法所存在的问题,继而提出了一种用于计数多目标最小生成树问题的所有非劣最优最小生成树的改进算法,并定性证明了该算法能够找到问题的所有非劣最优最小生成树.最后对改进算法进行测试,测试结果表明该算法是有效和可行的.

References:

[1] Yan WM, Wu WM. Data Structure. 2nd ed. Beijing: Tsinghua University Press, 1992. 168–174 (in Chinese).

[2] Zhou GG, Gen M. Genetic algorithm approach on multi-criteria minimum spanning tree problem. European Journal of Operational Research, 1999,114(1):141–152.

[3] Knowles JD. Local-Search and hybrid evolutionary algorithms for Pareto optimization [Ph.D. Thesis]. University of Reading, RG6 6AY, 2002.

[4] Srinivas N, Deb K. Multiobjective optimization using non-dominated sorting in genetic algorithms. Evolutionary Computation, 1995,2(3):221–248.

[5] Gottlieb J, Julstrom BA, Raidl GR, Rothlauf F. Prüfer numbers: A poor representation of spanning trees for evolutionary search. In: Spector L, Goodman E, Wu A, Langdon W, Voigt HW, Gen M, Sen S, Dorigo M, Pezeshk S, Garzon M, Burke E, eds. Proc. of the 2001 Genetic and Evolutionary Computation Conf. San Francisco: Morgan Kaufmann Publishers, 2000. 343–350.

[6] Xie T, Chen HW, Kang LS. Evolutionary algorithms of multi-objective optimization problems. Chinese Journal of Computers, 2003,26(8):997–1003 (in Chinese with English abstract).

附中文参考文献:

[1] 严蔚敏,吴伟民.数据结构.第 2 版,北京:清华大学出版社,1992.168–174.

[6] 谢涛,陈火旺,康立山.多目标优化的演化算法.计算机学报,2003,26(8):997–1003.

陈国龙(1965－),男,福建莆田人,博士,副教授,主要研究领域为信息安全,计算机网络,智能算法.

涂雪珠(1977－),女,硕士生,主要研究领域为智能算法.

郭文忠(1979－),男,硕士,主要研究领域为智能优化算法,计算机网络.

陈火旺(1936－),男,教授,博士生导师,中国工程院院士,主要研究领域为软件工程,人工智能.