

文章编号: 1001-0920(2007)01-0059-04

# 有时间窗车辆路径问题的捕食搜索算法

蒋忠中<sup>a</sup>, 汪定伟<sup>b</sup>

(东北大学 a. 工商管理学院, b. 信息科学与工程学院, 沈阳 110004)

**摘要:** 有时间窗车辆路径问题是当前物流配送系统研究中的热点问题, 该问题具有 NP 难性质, 难以求得最优解或满意解. 在建立有时间窗车辆路径问题数学模型的基础上, 设计了一种模仿动物捕食策略的捕食搜索算法. 该算法利用控制搜索空间的限制大小来实现算法的局域搜索和全局搜索, 具有良好的局部集中搜索和跳出局部最优的能力. 通过实例计算, 并与相关启发式算法比较, 取得了满意的结果.

**关键词:** 物流配送系统; 有时间窗车辆路径问题; 捕食搜索算法; 组合优化

**中图分类号:** TP29

**文献标识码:** A

## Predatory search algorithm for vehicle routing problem with time windows

JIANG Zhong-zhong<sup>a</sup>, WANG Ding-wei<sup>b</sup>

(a. College of Business Administration, b. College of Information Science and Engineering, Northeastern University, Shenyang 110004, China. Correspondent: JIANG Zhong-zhong, E-mail: neujzz@163.com)

**Abstract:** In recent logistics distribution system research, vehicle routing problem with time windows is a hot topic. However, it is hard to get an optimal or satisfactory solution because of its NP-hard complexity. Based on the mathematic model of vehicle routing problem with time windows, a predatory search algorithm is designed, which simulates animal predatory strategies. The algorithm can realize local search and global search in a way of controlling the restriction value of search space, and has better ability to search in local region and jump out of local optimal solution simultaneously. The computation results show the satisfactory performance of the designed algorithm compared with some heuristic algorithms.

**Key words:** Logistics distribution system; Vehicle routing problem with time windows; Predatory search algorithm; Combinatorial optimization

### 1 引言

随着电子商务的迅速发展, 物流配送系统的完善和优化已成为众多企业、专家和学者的研究热点<sup>[1,2]</sup>. 车辆路径问题 (VRP) 作为物流配送系统优化的重要组成部分, 自 Dantzig 等于 1959 年提出以来, 一直是运筹学和组合优化领域研究最活跃、成果最丰富的主题之一. VRP 一般描述为: 为服务于已知的一组顾客的一个车队, 设计一组开始和结束于一个中心 (配送中心) 出发点的最小费用 (距离) 路径, 每个顾客只能被服务一次, 而且一辆车服务的顾客数不能超过它的能力<sup>[3]</sup>. 有时间窗的车辆路径问题 (VRPTW) 是 VRP 的一类重要扩展, 它在 VRP 上加入了顾客访问时间窗口约束. 这一约束使得问

题的描述更贴近物流配送的现实情况, 因而对 VRPTW 的研究越来越受到人们的重视. 目前 VRPTW 已被证明是 NP 难题, 对它的求解主要集中在启发式算法上, 先后出现了求解 VRPTW 的一般启发式算法和以遗传算法为代表的智能启发式算法, 并取得了一些较好的研究成果<sup>[4,5]</sup>.

捕食搜索算法<sup>[6]</sup> (Predatory search) 是 Linhares 于 1998 年提出的一类新的仿生计算方法. 该算法模拟动物的捕食策略, 利用控制搜索空间的限制大小来实现局域搜索和全局搜索以及它们之间的转换, 具有很好的局部集中搜索和跳出局部最优的能力. 本文将捕食搜索算法应用于 VRPTW 的求解, 通过实际算例, 并与已有相关启发式算法对比,

收稿日期: 2005-09-05; 修回日期: 2006-04-19.

基金项目: 国家自然科学基金项目 (70431003, 07171056).

作者简介: 蒋忠中 (1979—), 男, 湖南祁阳人, 讲师, 博士, 从事电子商务、物流系统的建模与优化研究; 汪定伟 (1948—), 男, 江西彭泽人, 教授, 博士生导师, 从事复杂系统建模与优化、智能优化算法等研究.

取得了满意的结果.

## 2 VRPTW 的描述与模型

VRPTW 可描述为:某一配送中心为  $n$  个顾客  $C(C = \{1, 2, \dots, n\})$  进行配送服务, 顾客相应的需求量为  $d_i (i \in C)$ , 配送中心拥有  $k$  辆同种车型的车辆  $V = \{1, 2, \dots, k\}$ , 装载量为  $q$ , 同时满足以下约束:

- 1) 每个顾客仅被服务一次;
- 2) 每条车辆路径从配送中心 0 出发, 完成任务后返回配送中心 0, 将顾客和配送中心合记为  $N = \{0, 1, \dots, n\}$ ;
- 3) 每辆车服务顾客不超过它自身的装载量;
- 4) 顾客必须在其要求的时间窗口  $[a_i, b_i]$  内被服务, 服务车辆要求在  $b_i$  之前到达顾客  $i$ , 在  $a_i$  之前可以允许车辆到达顾客  $i$ , 但必须等待至时间  $a_i$  再为顾客  $i$  服务.

问题的目标函数是在满足上述约束下设计车辆路径, 使其总距离(费用)最小. 为了便于建立 VRPTW 模型, 将其他相关参数设定如下:  $w_i$  为车辆对顾客  $i$  的服务时间;  $c_{ij}$  和  $t_{ij}$  分别为顾客(或配送中心)  $i$  与  $j$  之间的距离和车辆行驶时间;  $s_{ik}$  为车辆  $k$  为顾客  $i$  开始服务的时间. 特别地, 对于配送中心  $s_{ik} = 0$ , 如果车辆  $k$  没有为顾客  $i$  服务, 则  $s_{ik}$  没有任何意义. 定义决策变量

$$x_{ijk} = \begin{cases} 1, & \text{车辆 } k \text{ 从顾客(配送中心) } i \text{ 到达 } j; \\ 0, & \text{其他.} \end{cases}$$

通过以上描述和假设, 可将 VRPTW 的数学模型表示为

$$\text{Min} \sum_{k \in V} \sum_{i \in N} \sum_{j \in N} c_{ij} x_{ijk}. \quad (1)$$

s. t.

$$\sum_{k \in V} \sum_{i \in N} x_{ijk} = 1, \quad \forall j \in C; \quad (2)$$

$$\sum_{i \in C} \sum_{j \in N} d_i x_{ijk} \leq q, \quad \forall k \in V; \quad (3)$$

$$\sum_{j \in N} x_{0jk} = 1, \quad \forall k \in V; \quad (4)$$

$$\sum_{i \in N} x_{ihk} - \sum_{j \in N} x_{hjk} = 0, \quad \forall h \in C, k \in V; \quad (5)$$

$$\sum_{i \in N} x_{i0k} = 1, \quad \forall k \in V; \quad (6)$$

$$s_{ik} + w_i + t_{ij} - T(1 - x_{ijk}) \leq s_{jk}, \quad \forall i, j \in N, k \in V; \quad (7)$$

$$a_i \leq s_{ik} \leq b_i, \quad \forall i \in N, k \in V; \quad (8)$$

$$x_{ijk} = 0 \text{ 或 } 1, \quad \forall i, j \in N, k \in V. \quad (9)$$

式(1)是车辆路径最短的目标函数; 式(2)表示每个顾客仅能被访问一次; 式(3)是车辆的装载量约束; 式(4)和(6)保证车辆从配送中心出发, 完成

配送任务后返回配送中心; 式(7)表示车辆  $k$  在从客户  $i$  向客户  $j$  行驶的过程中, 在时间  $s_{ik} + w_i + t_{ij}$  前不能到达客户  $j$ , 其中  $T$  是一个较大的数; 式(8)表示车辆为客户  $i$  服务的时间必须在其时间窗内; 式(9)为决策变量的取值.

## 3 捕食搜索算法求解 VRPTW

### 3.1 捕食搜索算法简介

动物学家在研究动物的捕食行为时发现, 尽管由于动物物种的不同而造成身体结构的千差万别, 但它们的捕食行为却惊人地相似. 动物捕食过程中在没有发现猎物和猎物的迹象时, 捕食动物在整个捕食空间沿着一定的方向以很快的速度寻找猎物. 一旦发现猎物或发现猎物的迹象, 它们就放慢步伐, 在发现猎物或有猎物迹象的附近区域进行集中的区域搜索, 以找到更多的猎物. 在搜寻一段时间没有找到猎物后, 捕食动物将放弃该集中的区域, 继续在整个捕食空间寻找猎物.

模拟动物的这种捕食策略, Linhares 于 1998 提出了一种新的仿生计算方法, 即捕食搜索算法. 算法原理如下: 捕食搜索寻优时, 先在整个搜索空间进行全局搜索, 直到找到一个较优解; 然后在较优解附近区域进行集中搜索, 直到搜索很多次也没有找到更优解, 从而放弃局域搜索; 接着再在整个搜索空间进行全局搜索; 如此循环, 直到找到最优解(或近似最优解)为止. 捕食搜索的这种策略很好地协调了局部搜索与全局搜索之间的转换: 在较差的区域进行全局搜索以找到较好的区域; 然后在较好的区域进行集中的局域搜索, 以使解得到迅速改善; 全局搜索对解空间进行广度探索, 而局域搜索可以对较好区域进行深度开发. 因为捕食搜索的局域搜索只集中在一个相对很小的区域进行, 所以搜索速度很快, 而且全局搜索可以提高搜索的质量, 使搜索避免陷入局部最优点. 目前该方法已成功应用于组合优化领域的旅行商问题(TSP)和超大规模集成电路设计问题(VLSI Layout)<sup>[7,8]</sup>. 本文尝试将捕食搜索算法应用于 VRPTW 中, 设计了求解 VRPTW 的捕食搜索算法.

### 3.2 VRPTW 的捕食搜索算法设计

#### 3.2.1 解的表达

采用基于顾客直接排列的顺序编码, 即对  $n$  个客户随机产生  $n$  个  $1 \sim n$  之间不重复的整数排列, 从而形成一个解对应的顺序编码. 例如 8 个顾客, 可编码为 8-7-6-2-3-4-5-1; 然后根据顾客的编码顺序依次将其纳入一条车辆路径中, 直到超过该车的装载量限制或对纳入的顾客服务开始时间大于其最晚服务时间时, 产生一条新的车辆路径. 依次类推, 得

到该编码对应的多条车辆路径(车辆数),即为编码对应的解.

3.2.2 邻域定义

采用逆转法实现邻域的操作,即随机选择解的两个位置,将他们之间的编码进行逆转,得到当前解的一个邻域.

3.2.3 目标值的确定

目标值  $f(x)$  由编码对应的解求得.另外,给予超出最大车辆数的解一定的惩罚,惩罚与其超出的车辆数成比例.

3.2.4 限制的选取

通过设置限制值的大小实现算法的局域和全局搜索并在它们之间进行转换是捕食搜索的重点.本文采用 Linhares 提出的目标函数值限制法,即通过多次搜索当前最好解的邻域获得一组目标函数值,选取其中较小的部分值作为局域搜索限制,较大的部分值作为全局搜索限制.

3.2.5 算法步骤

(1) 算法参数说明

$S$  为解附近邻域的搜索次数;LEVEL- NUM 为设定限制值的总数,LOCAL- LEVEL 为局域搜索限制级别的上限,GLOBAL- LEVEL 为全局搜索限制级别的下限,且三者之间满足 LEVEL- NUM > GLOBAL- LEVEL > LOCAL- LEVEL > 0; COUNTER- MAX 为每个限制下最大循环次数.

上述参数的设定目前尚无理论依据,都是针对求解问题的本身而确定的.

本文通过实验将参数大致设置如下: $S$  取问题的规模数  $L$  的 0.5 ~ 0.8 倍;LEVEL- NUM 取问题的规模数  $L$ ;LOCAL- LEVEL 的值取  $[LEVEL- NUM/t]$ , $[ ]$  表示取整数;GLOBAL- LEVEL 的值取  $\max\{LEVEL- NUM - LOCAL- LEVEL, LOCAL- LEVEL + 1\}$ ,其中  $t$  视问题的规模取值;COUNTER- MAX 取问题规模数  $L$  的 8 ~ 10 倍.

(2) 算法流程

Step 1: 随机产生一个初始解  $x$ ,令至今最好解  $x_{min} = x$ ,限制级别 Level = 0,循环次数 Counter = 0;

Step 2: 如果 Level = LEVEL- NUM,则搜索  $x$  的邻域  $S$  次,并取其最小解  $x_{n\_min}$ ,然后转 Step3,否则结束, $x_{min}$  即为求得的最优解;

Step 3: 如果  $f(x_{n\_min}) < Restriction(Level)$ ,则令  $x = x_{n\_min}$ ,然后转 Step4,否则转 Step5;

Step 4: 如果  $f(x) < f(x_{min})$ ,则令  $x_{min} = x$ ,Level = 0, Counter = 0,重新计算限制,然后转 Step2,否则转 Step5;

Step 5: 令 Counter = Counter + 1,如果 Counter > COUNTER- MAX,则令 Level = Level + 1, Counter = 0,然后转 Step6,否则转 Step2;

Step 6: 如果 Level = LOCAL- LEVEL,则令 Level = GLOBAL- LEVEL(通过限制级别 Level 的跳跃实现从局域搜索到全局搜索的转换),并转 Step 2,否则直接转 Step2.

(3) 限制(Restriction) 的计算

每当解得到改善时,即获得了一个至今最好的解,则执行以下操作,得到新的限制:

1) 搜索 LEVEL- NUM 次至今最好解的邻域,得到 LEVEL- NUM 个解的目标值;

2) 将这 LEVEL- NUM 个值与至今最好解的值按升序排列;

3) 将排列后的 LEVEL- NUM + 1 个值依次赋给限制 Restriction(0), ..., Restriction(LEVEL- NUM).

4) 取其中 Restriction(0), ..., Restriction(LOCAL- LEVEL) 部分作为算法的局域搜索限制,而 Restriction(GLOBAL- LEVEL), ..., Restriction(LEVEL- NUM) 部分作为算法的全局搜索限制.

4 计算实例与结果分析

为了便于结果分析和比较,本文选用文献[5]中的 VRPTW 例子进行试算.该问题是某一配送中心为 8 个顾客(任务)服务,配送中心与各顾客之间的距离如表 1 所示;顾客需求量分别为  $d_i$ ,服务时间(装卸货时间)为  $T_i$ ,服务时间窗口为  $[a_i, b_i]$ ,如表 2 所示.配送任务由装载量为 8 t 的车辆完成,车速为 50 km/h.要求合理安排车辆路线,使得配送总距离最短.

表 1 顾客(配送中心)之间的距离 km

距离	0	1	2	3	4	5	6	7	8
0	0	40	60	75	90	200	100	160	80
1	40	0	65	40	100	50	75	110	100
2	60	65	0	75	100	100	75	75	75
3	75	40	75	0	100	50	90	90	150
4	90	100	100	100	0	100	75	75	100
5	200	50	100	50	100	0	70	90	75
6	100	75	75	90	75	70	0	70	100
7	160	110	75	90	75	90	70	0	100
8	80	100	75	150	100	75	100	100	0

捕食搜索算法采用 Java 语言在 Windows 平台上(主频 P4/2.2 G,内存 256 M)实现.其中算法参数设置为: $S = 4$ ,LEVEL- NUM = 8,LOCAL-



表2 顾客的需求、服务时间及时间窗

顾客	需求/t	时间/h	时间窗/时刻
1	2	1	[1,4]
2	1.5	2	[4,6]
3	4.5	1	[1,2]
4	3	3	[4,7]
5	1.5	2	[3,5]
6	4	2.5	[2,5]
7	2.5	3	[5,8]
8	3	0.8	[1.5,4]

LEVEL = 2, GLOBAL-LEVEL = 6, COUNTER-MAX = 80, 不可行解惩罚比例系数为 500. 求得最优值为 910 (使用车辆 3 台, 对应车辆路径分别为 0-3-1-2-0, 0-6-4-0, 0-8-5-7-0). 而文献 [5] 利用分派启发式算法求得的最优值为 1 020 (使用车辆 3 台, 对应车辆路径分别为 0-3-1-5-0, 0-6-4-0, 0-8-2-7-0). 可见捕食搜索算法明显优于分派启发式算法.

为进一步对比分析, 将捕食搜索和基于类顺序交叉、换位变异算子的遗传算法<sup>[9]</sup> (编码相同, 交叉率 0.8, 变异率 0.05, 正比选择子代, 种群 60, 迭代次数 50) 在同一计算平台上 (硬件平台和软件平台均相同) 对上述实例随机计算 20 次, 得到相应的目标值如表 3 所示.

表3 捕食搜索和遗传算法计算结果比较

计算次数	1	2	3	4	5	6	7	8	9	10
遗传算法	945	910	910	910	970	945	910	910	945	910
捕食搜索	910	910	910	910	910	910	910	910	910	910
计算次数	11	12	13	14	15	16	17	18	19	20
遗传算法	945	910	945	910	910	970	910	970	945	945
捕食搜索	910	910	910	970	910	910	910	910	910	910

由表 3 可以看到, 捕食搜索和遗传算法均能求得问题的最优值 910. 但从计算的整体效果看, 捕食搜索更易求得问题的最优解, 20 次计算中 19 次得到了最优解, 达优率为 95%, 而遗传算法的达优率为 50%. 另外, 两算法 20 次求解的平均值分别为 913.0 和 931.3, 平均计算时间分别为 0.243 4 s 和 0.245 9 s. 因而在计算时间基本相当的情况下, 捕食搜索比遗传算法具有更好的寻优效果. 为验证上述结果的一般性, 对于随机产生的多个 VRPTW 实例, 采用相应的捕食搜索和遗传算法进行计算, 也得到了相似的结果, 在此不再赘述.

Linhares 通过研究发现, 捕食搜索算法对于局部最优解聚集于全局最优解附近的组合优化问题, 如 TSP 和二分图问题<sup>[10]</sup>, 其搜索效率很高. 这种高效率源自动物捕食行为的本质, 只有当猎物聚集时,

捕食者采用这种策略效率才高. 可以想象, 当猎物分布很扩散而且没有规律时, 捕食者进行局域搜索只能无功而返. VRPTW 本质上是一类 TSP 的扩展, 从理论上讲, 捕食搜索对于求解 VRPTW 是高效的, 而上述的实例计算结果也验证了这一点. 同时捕食搜索算法具有思路简洁、易于编程实现的特点, 因而是一种求解 VRPTW 的较好方法.

## 5 结 论

VRPTW 是 VRP 的一类重要扩展问题, 它比 VRP 更能贴近现代物流配送的实际情况, 因而对它的研究更具有应用前景. 本文在建立 VRPTW 数学模型的基础上, 针对 VRPTW 内在的 NP 难性质, 设计了一种模仿动物捕食策略的捕食搜索算法. 计算结果表明, 该算法简洁、有效, 是求解 VRPTW 这类复杂组合优化问题的一种较好方法.

## 参考文献(References)

- [1] Elliot R, Joseph P B. Physical distribution service quality in internet retailing: Service pricing, transaction attributes, and firm attributes [J]. J of Operations Management, 2004, 21(6): 651-672.
- [2] Niklids A, Fredrik S. Electronic commerce, marketing channels and logistics platforms — A wholesaler perspective [J]. European J of Operation Research, 2003, 144(2): 270-279.
- [3] Tarantilis C D. Solving the vehicle routing problem with adaptive memory programming methodology [J]. Computation and Operation Research, 2005, 32(9): 2309-2327.
- [4] 李军, 郭耀煌. 物流配送车辆优化调度理论与方法 [M]. 北京: 中国物资出版社, 2001: 113-132. (Li J, Guo Y H. Theory and method of logistics distribution vehicle scheduling [M]. Beijing: China Material Press, 2001: 113-132.)
- [5] 李军. 车辆调度问题的分派启发式算法 [J]. 系统工程理论与实践, 1999, 117(1): 27-33. (Li J. Generalized assignment heuristics for vehicle scheduling [J]. Systems Engineering Theory and Practice, 1999, 117(1): 27-33.)
- [6] Linhares A. Preying on optima: A predatory search strategy for combinatorial problems [C]. Proc of the IEEE Int Conf on Systems, Man and Cybernetics. CA: San Diego, 1998: 2974-2978.
- [7] Linhares A. State-space search strategies gleaned from animal behavior: A traveling salesman experiment [J]. Biological Cybernetics, 1998, 78(3): 167-173.
- [8] Linhares A. Synthesizing a predatory search strategy for VLSI layouts [J]. IEEE Trans on Evolutionary Computation, 1999, 3(2): 147-152.

(下转第 68 页)

一种市场经营行为,它们之间的分界线并不十分清晰.

综上所述,可得出如下结论:

1) 在动态联盟或协作制造等模式的生命周期过程中,制造任务的规划往往是过程的核心问题,制造任务是过程环节中的主要线索;

2) 制造任务的规划和制造资源的配置并不是相互孤立的,它们彼此影响,互相制约,共同构成组织形成过程的活动网络;

3) 制造任务与制造资源的配置方案形式多种多样,企业必须根据具体的市场环境和制造任务需求,具体处理问题,并没有统一的方案模式;

4) 制造任务和制造资源的配置过程不是毫无消耗的过程,市场中的企业需要考虑组建协作组织所耗费的时间、成本,以及随之而来的协作风险.

## 6 结 语

本文通过分析制造任务协作组织的形成过程,指出了制造任务的规划是协作组织组建过程的核心问题.因此,本文建立了制造任务组织图 MTOG,分析了 MTOG 图的特性和运算规则,基于 MTOG 图研究了制造任务协作组织的形成过程,并对组建过程中的各种具体问题进行了详细讨论.

## 参考文献(References)

- [1] 本杰明·古莫斯·卡瑟尔斯. 竞争的革命[M]. 广州: 中山大学出版社, 2000: 365.  
(Benjamin C C. The revolution of competition [M]. Guangzhou: Zhongshan University Press, 2000: 365.)
- [2] Burgess T F. Making the leap to agility: Defining and achieving agile manufacturing through business process redesign and business network redesign [J]. Int J of Operations and Production Management, 1994, 14(11): 23-34.
- [3] Gunasekaran A. Agile manufacturing enablers and an implementation framework [J]. Int J of Production Research, 1998, 36(5): 1223-1247.
- [4] Carnarinha L M. Towards an architecture for virtual enterprise[J]. J of Intelligent Manufacturing, 1998, 9(2): 189-199.
- [5] 战德臣,叶丹,徐晓飞,等. 动态联盟建立过程研究[J]. 计算机集成制造系统 CIMS, 1997, 8(4): 3-7.  
(Zhan D C, Ye D, Xu X F, et al. Research on the building process of virtual organization [J]. Computer Integrated Manufacturing System, 1997, 8(4): 3-7.)
- [6] Barnett W D, Presley A, Liles D H. Object-oriented business process modeling for the virtual enterprise[C]. Proc of the 4th Annual Agility Forum Conf. Atlanta, 1995: 67-69.
- [7] 蒋贵川,范玉顺,吴澄. 动态联盟建盟过程模型与分析[J]. 机械工程学报, 2002, 38(1): 58-62.  
(Jiang G C, Fan Y S, Wu C. Dynamic collaboration's constructing process model and analysis [J]. Chinese J of Mechanical Engineering, 2002, 38(1): 58-62.)
- [8] 约翰逊鲍. 离散数学[M]. 第5版. 北京: 人民邮电出版社, 2003.  
(Richard Johnsonbaugh. Discrete mathematics [M]. 5th ed. Beijing: Posts & Telecom Press, 2003.)
- [9] Douglas Tuttle, Brian Kanter. Activities: The common currency of the virtual organization[C]. Proc of the 4th Annual Agility Forum Conf. Atlanta, 1995: 211-214.
- [10] Voegtli Cinda. Real-world challenges: Using the virtual enterprise for successful product development projects[C]. IEEE Int Engineering Management Conf. Piscataway: IEEE, 1996: 698-704.
- [11] Briggs R O, De Vreede G J, Nunamaker J F. Collaboration engineering with thinklets to pursue sustained success with group support systems[J]. J of Management Information Systems, 2003, 19(4): 31-64.
- [12] 刘飞,雷琦,宋豫川. 网络化制造的内涵及研究发展趋势[J]. 机械工程学报, 2003, 39(8): 1-6.  
(Liu F, Lei Q, Song Y C. Connotation, state-of-the-art and research tendency of networked manufacturing [J]. Chinese J of Mechanical Engineering, 2003, 39(8): 1-6.)

(上接第 62 页)

- [9] 玄光男,程润伟. 遗传算法与工程设计[M]. 北京: 科学出版社, 2000: 1-22.  
(Xuan G N, Cheng R W. Genetic algorithm and engineering design [M]. Beijing: Science Press, 2000: 1-22.)
- [10] Hernandez G, Salinas L. Large scale simulations of a neural network model for the graph bisection problem on geometrically connected graphs [J]. Electronic Notes in Discrete Mathematics, 2004, 18: 151-156.