

分段蚁群算法在运输调度问题中的应用

师 凯,蔡延光,邹谷山,王 涛

(广东工业大学 自动化学院,广东 广州 510090)

摘要:在基本蚁群算法的基础上提出了分段蚁群算法,通过改进选择策略和信息素更新机制以求解一类运输调度问题,实验证明结果理想。

关键词:启发式算法;蚁群算法;运输调度

中图分类号: TP301

文献标识码: A

文章编号: 1007-7162(2006)01-0071-06

运输调度问题(Vehicle Routing and Scheduling Problems, VRP)有着强烈的应用背景,在工农业生产和社会生活中随处可见。例如,生产型企业的成品、零件和在制品的配送;商业流通型企业的商品配送;公共事业的城市客运车辆的配置和调度;城市垃圾清理;通讯网络中的信息传输等。然而,我国现在的物流业和发达国家有较大的差距。为了提高我国物流业的竞争力,首要任务是提高物流运输管理能力,而提高运输调度水平则是其中的关键。鉴于其重要性,自 VRP 提出以来,国内外学者集思广益,设计了多种求解算法,如贪婪算法,树搜索法,动态规划法,分支定界法等传统算法;还有遗传算法,模拟退火算法,禁忌搜索法及蚁群算法等智能算法。由于其在求解最短路线问题上的先天优势,本文拟用蚁群算法求解一类运输调度问题,并针对选择策略和信息素更新机制进行改进,克服了基本蚁群算法的某些弊病,从而可以较好地解决大规模运输调度问题。

1 VRP 模型的建立

先给出运输调度问题的文字描述:已知需求点(客户)的位置坐标和货物需求量,一个车队(有多个车辆)从一个供应点(配送中心)出发,每个需求点只被一辆车访问,且该车所访问需求点的需求量总和不能超过车辆的负载能力;应如何安排车辆的行走路线使得总路线最短。要求:每辆车运输完后后回到出发点(供应点)。

设供应点有 K 辆车,每辆车的载重为 Q_k ($k = 1, 2, \dots, K$),且每辆车每次送货最大距离不超过 D_k ;需求点个数为 L ,每个需求点的需求量为 q_i ($i = 1, 2, \dots, L$);需求点 i 到 j 的距离为 d_{ij} ($i, j = 0, 1, 2, \dots, L$, 其中 $i = 0$ 或 $j = 0$ 表示该点为供应点);第 k 辆车访问的需求点个数为 n_k ($n_k = 0$ 表示未使用第 k 辆车);用集合 R_k 表示第 k 辆车的行驶路线, r_{ki} 代表 R_k 中一个需求点,它在路线 R_k 中的顺序为 i , r_{k0} 表示供应点。借鉴并改进文献[1]的数学模型,有:

$$\min Z = \sum_{k=1}^K \left[\sum_{i=0}^{n_k-1} d_{r_{ki}r_{k(i+1)}} + d_{r_{kn_k}r_{k0}} \text{sign}(n_k) \right], \quad (1)$$

收稿日期: 2005-03-03

基金项目:国家自然科学基金(60374062);广东省科技计划项目(2004B10101038);广东省自然科学基金项目(04009488)

作者简介:师凯(1980-),男,在读研究生,主要研究方向为物流管理、组合优化。

$$\text{s. t. } \sum_{i=1}^{n_k} q_{r_{ki}} \leq Q_k, \quad (2)$$

$$\sum_{i=0}^{n_k-1} d_{r_{ki}r_{k(i+1)}} + d_{r_{kn_k}r_{k0}} \text{sign}(n_k) \leq D_k, \quad (3)$$

$$0 \leq n_k \leq L, \quad (4)$$

$$\sum_{k=1}^K n_k = L, \quad (5)$$

$$R_k = \{r_{ki} \mid r_{ki} \in \{1, 2, \dots, L\}, i = 1, 2, \dots, n_k\}, \quad (6)$$

$$R_{k_1} \cap R_{k_2} = \phi, \forall k_1 \neq k_2, \quad (7)$$

$$\text{sign}(n_k) = \begin{cases} 1 & (n_k) \geq 1, \\ 0 & \text{其它} \end{cases} \quad (8)$$

其中,式(1)为目标函数;式(2)保证每条路径上各需求点的需求量之和不超过汽车的载重量;式(3)保证每条配送路径的长度不超过汽车一次配送的最大行驶距离;式(4)表明每条路径上的需求点数不超过总需求点数;式(5)表明每个需求点都得到配送服务;式(6)为每条路径的需求点的组成;式(7)限制每个需求点仅能由一辆汽车送货;式(8)表明当第 k 辆汽车服务的客户数大于等于 1 时,该车参加了配送,此时取 $\text{sign}(n_k) = 1$,当第 k 辆汽车服务的客户数小于 1 时,表示未使用该车,取 $\text{sign}(n_k) = 0$.

2 运输调度问题的基本蚁群算法

蚁群算法是继遗传算法、模拟退火算法、禁忌搜索算法等智能算法迅速发展和应用之后提出的一种新型的模拟进化算法.该算法首先由意大利学者 M. Dorigo 等人提出并用于求解 TSP 问题^[2].昆虫学家发现,单个蚂蚁的智能水平低下,充其量不过是沿着另一只蚂蚁走过的路径行走;但蚂蚁一旦集结成群,便表现出高度的组织性和协调一致性,并能解决一系列单个蚂蚁无法完成的问题.因此,科学工作者把蚂蚁的这种群体行为定义为“群集智能”(swarm intelligence).蚁群算法便是源于大自然中蚂蚁觅食行为的研究.研究表明,蚂蚁在走过的路线上会留下一一种被称作信息素(pheromone)的物质,蚂蚁之间就是通过这种物质进行信息传递的.一旦后来的蚂蚁感知到同一蚁群前面蚂蚁遗留的信息素,就会受这种信息素的影响以较大概率选择信息素浓度大的路线.随着蚂蚁数量的增多,该路线上的信息素浓度就会不断增大.这种自催化行为是蚂蚁能够迅速找到最短路线的生理特点,从而也是蚁群算法研究问题的主要机制.

基本蚁群算法主要由选择策略,信息素局部更新,局部搜索机制,信息素全局更新组成^[3].

2.1 选择策略

在运输调度问题中,转移概率 P_{ij}^k 定义为

$$P_{ij}^k = \frac{\tau_{ij}^\alpha(t) \eta_{ij}^\beta}{\sum_{j \in \text{allowed}} \tau_{ij}^\alpha(t) \eta_{ij}^\beta},$$

式中 $\tau_{ij}(t)$ 表示在某次循环 t 中边 (i, j) 上的遗留信息浓度; η_{ij} 表示相应边上的路径启发信息,在该例中常取 $\eta_{ij} = 1/d_{ij}$, d_{ij} 表示边 (i, j) 之间的距离; α, β 是权重参数; $j \in \text{allowed}$ 表示还未访问过的需求点.

2.2 信息素局部更新

每只蚂蚁局部更新过程表示为

$$\Delta \tau_{ij} \leftarrow \Delta \tau_{ij} + \Delta \tau_{ij}^k.$$

2.3 局部搜索机制

m 只蚂蚁搜索完一个循环后将得到 m 个候选解,在这些解的邻域运用某些局部搜索算法以加速局部最优解的得出.常用方法有 2-opt 法、爬山法等.

2.4 信息素全局更新

表示为

$$\tau_{ij}(t+1) = (1 - \rho) \tau_{ij}(t) + \Delta \tau(t),$$

式中 $\rho \in (0, 1)$ 是信息素挥发速度, $1 - \rho$ 表示信息素的残留度;初始时刻 τ_{ij} 是一个常数 C ; $\Delta \tau(t)$ 是边 (i, j) 上信息素的增量,即 m 只蚂蚁本次循环在边上留下的信息素之和.

3 分段蚁群算法

在运用基本蚁群算法求解时发现,基本蚁群算法存在某些缺陷.如当需求点数目较大时,整个算法的搜索时间较长,得出的最优解不稳定,严重限制了该算法的优势;另外,搜索过程中易陷入局部最优解,即在达到规定迭代次数之前很长时间内不再发生变化,搜索停滞.针对这些问题,国内外学者提出了一些行之有效的改进算法,而本文则另外提出一种新的改进方式,如下所述.

3.1 选择策略的分段

将整个迭代过程分为三段:初始阶段以较大概率选择信息素浓度较大的边,从而加速初期算法的收敛;中期适当降低对当前最优解的选择概率,并相应增加对其邻域的选择概率,使其跳出局部最优解;后期再恢复初期速度,使整个迭代过程加速.设 P_{ij}^k 为改进后的转移概率, p_{ij}^k 为基本蚁群算法转移概率, $\arg \max(\tau)$ 为各条边中信息素最大的边, $nc_{\max}$ 为最大迭代次数, nc_0 和 nc_1 分别为介于 1 和 $nc_{\max}$ 之间的迭代次数,且 $nc_0 < nc_1$.则改进后的选择策略可改进为

1) 在算法初始阶段 $(1, nc_0)$ 以较大概率选择信息素浓度大的边,加速初期算法收敛.

$$P_{ij}^k = \begin{cases} \frac{1 + p_{ij}^k}{2} & \tau_{ij} = \arg \max_{\tau_{ij} \in \tau}(\tau), \\ \frac{p_{ij}^k}{2} & \text{其它.} \end{cases}$$

2) 为了防止陷入局部最优解,中期 (nc_0, nc_1) 适当降低对当前最优解的选择概率,并相应增加对其邻域的选择概率.

$$P_{ij}^k = \begin{cases} \frac{p_{ij}^k}{2} & \tau_{ij} = \arg \max_{\tau_{ij} \in \tau}(\tau), \\ \frac{1 + p_{ij}^k}{2} & \text{其它.} \end{cases}$$

3) 后期 $(nc_1, nc_{\max})$ 恢复初期速度.转移概率同 1).

3.2 信息素挥发速度的改进

在基本蚁群算法中,尽管各边的信息素浓度差别较大,但它们的挥发速度却是相同的.而实际情况则是信息素浓度越大,挥发速度越快.可依据信息素浓度与挥发速度的同向关系调整信息素.因此,可以将信息素全局更新公式表示为

$$\tau_{ij}(t+1) = (1 - \rho(\tau_{ij})) \tau_{ij}(t) + \Delta \tau(t),$$

式中 $\rho(\tau_{ij})$ 不是一个常数,而是一个位于 $(0, 1)$ 的以 τ_{ij} 为自变量的函数. $\rho(\tau_{ij})$ 随自变量 τ_{ij} 的增大而增大. 本算法中取 $\rho(\tau_{ij})$ 为一正比例函数.

3.3 局部搜索机制

在蚂蚁得出某条路径时,采用 3-opt 法进一步完善此候选解. 这是因为尽管 3-opt 法的复杂度比 2-opt 法的复杂度增加 n 倍,但对本文算法而言精度大大提高,故采用 3-opt 法.

3.4 算法步骤

根据上面的讨论,可给出解决运输调度问题的改进蚁群算法的步骤.

步骤 1:初始化 α, β 等参数,置 $\Delta \tau \leftarrow 0$,迭代次数 $nc \leftarrow 0$.

步骤 2:将 m 只蚂蚁置于初始点,并将初始点置于当前解集中.

步骤 3:对每只蚂蚁 k ,计算 p_{ij}^k ,并在不同阶段选取相应的 P_{ij}^k 和剩余载重量选择下一顶点 j ,并将 j 置于当前解集中.

步骤 4:依式 $\Delta \tau_{ij} \leftarrow \Delta \tau_{ij} + \Delta \tau_{ij}^k$ 进行信息素局部更新.

步骤 5:应用 3-opt 法完善局部搜索,计算各蚂蚁的目标函数,并记录局部最优解.

步骤 6:依式 $\tau_{ij}(nc+1) = (1 - \rho(\tau_{ij})) \tau_{ij}(nc) + \Delta \tau(nc)$ 进行信息素全局更新.

步骤 7:置 $\Delta \tau_{ij} \leftarrow 0$, $nc \leftarrow nc + 1$.

步骤 8:若 $nc < nc_{\max}$,则转步骤 2.

本算法用 java 编程实现,并在 Windows xp 系统环境下运行通过.

4 实例分析

这里引用文献[3]的一个常用的运输调度问题为例来测试算法. 有 8 个需求点和一个供应点的系统,各个需求点对供应点的需求量为 $q_i (i = 1, 2, \cdots, 8)$ (单位为 t). 供应点只有两辆车用于配送,每辆车的载重皆为 8 t,已知供应点与各个需求点之间的距离(单位为 km)如表 1 所示(其中 0 表示中心仓库),要求合理安排车辆的行驶路线,使总运输里程最小.

表 1 各点间的距离与需求量

									km
需求点 d_{ij}	0	1	2	3	4	5	6	7	8
0	0	4	6	7.5	9	20	10	16	8
1	4	0	6.5	4	10	5	7.5	11	10
2	6	6.5	0	7.5	10	10	7.5	7.5	7.5
3	7.5	4	7.5	0	10	5	9	9	15
4	9	10	10	10	0	10	7.5	7.5	10
5	20	5	10	5	10	0	7	9	7.5
6	10	7.5	7.5	9	7.5	7	0	7	10
7	16	11	7.5	9	7.5	9	7	0	10
8	8	10	7.5	15	10	7.5	10	10	0
需求量 q_i/t	-	1	2	1	2	1	4	2	2

本例中,取 $\alpha = 1, \beta = 3, Q = 10$ (Q 为求解某条边上信息素增量用到的一个常数), $nc_{\max} = 200$, $nc_0 = 20$, $nc_1 = 100$. 通过 10 次计算,全部找到最优解:

第一辆车:0→1→3→5→8→9;
第二辆车:0→4→7→6→2→9;
总里程:65.5 km.

而遗传算法和节约法得到的最优解分别为 67.5 和 79.5. 故本文算法优于遗传算法和节约法. 另外,通过对文献[4]中提供的几种规模的运输调度问题进行求解,并与文献[1]和[5]所得出的解进行比较,结果(如表 2 所示)发现本文所提出的分段蚁群算法对于规模较小的问题得出的解优势不明显,但对于规模越大的问题解越优,基本适用于目前规模日益增大的运输调度问题.

表 2 结果对比 km

问题		扫描法	分配法	树搜索法	遗传算法	蚂蚁算法	本文蚁群算法
序号	规模						
1	50	532	524	534	521.8	564.2	582.4
2	100	851	833	832.9	840.5	840.7	841.9
3	150	1 079	1 014	1 088.6	1 060.9	1 032.5	1 029.5
4	199	1 389	1 386	1 461.2	1 461.2	1 375.4	1 367.0

5 结束语

蚁群算法具有很强的鲁棒性和发现较好解的能力,是一种基于合作的无监督机制的并行算法. 蚁群算法并不孤立,它可以同其他多种启发式算法或智能算法相结合,获得更优的研究结果和更大的研究范围. 本文提出的改进蚁群算法从例证中可以看出是有效的,实用的;但同时,还有较多需要进一步解决的问题. 比如算法中某些参数(如 α, β)的搭配,现在只能通过试凑得出最佳方案,并且不同类的运输调度问题对参数也很敏感. 运输调度问题还可以衍生出很多更为复杂也更贴近实际应用的问题,如多个供应/需求点,多车队,时间窗^[6],多重运输需求^[7-8],多种运输环节,多重交通网络,多重约束条件,客户需求多样性等,这都是蚁群算法今后需要突破的重要方向之一.

参考文献:

[1]姜大立,杨西龙,杜文,等. 车辆路径问题的遗传算法研究[J]. 系统工程理论与实践,1999(6):40-45.
[2]Colormi A, Dorigo M, Maniezzo V. Distributed Optimization by Ant Colonies[A]. Proc. 1st European conf. Artificial Life [C],Pans,France:Elsevier. 1991. 134-142.
[3]Breiman L. Bagging predictors[J]. Machine Learning,1996(24):123-140.
[4]Christofides N,Mingozzi A,Toth P. The Vehicle Routing Problem Combinational Optimization[M]. New York:Johnly Wiley,1979.
[5]刘云忠,宣慧玉. 蚂蚁算法在车辆路径问题中的应用研究[J]. 信息与控制,2004,33(2):249-252.
[6]蔡延光,钱积新,孙优贤. 带时间窗的多重运输调度问题的自适应 Tabu Search 算法[J]. 系统工程理论与实践,2000(12):42-50.
[7]蔡延光,钱积新,孙优贤. 多重运输调度问题的模拟退火算法[J]. 系统工程理论与实践,1998(10):11-15.
[8]蔡延光,钱积新,孙优贤. 多重运输调度问题基于双表的并行表搜索算法[J]. 系统工程理论与实践,1998(11):20-26.

Application of Subsection Ant Colony Algorithm to Vehicle Routing Problems

SHI Kai ,CAI Yan-guang ,ZOU Gu-shan ,WANG Tao

(Faculty of Automation ,Guangdong University of Technology ,Guangzhou 510090 ,China)

Abstract : A subsection ant colony algorithm on the base of the original ant colony algorithm is put forward to solve a kind of VRP. And it is improved in two parts of selection strategy and information update. The result of some experiments is encouraging.

Key words : heuristic algorithm ; ant colony system ; vehicle routing problems

(上接第 70 页)

参考文献:

- [1]孙青,庄奕琪,王吉锡.电子元器件可靠性工程[M].北京:电子工业出版社,2002.
- [2]王少萍.工程可靠性[M].北京:北京航空航天大学出版社,2000.
- [3]周玉芬,高锡俊,李建华.环境应力筛选加速系数[J].航空精密制造技术,2003,5(4):38-41.
- [4]MIL-STD-883D,美国军用微电子器件试验方法和程序[S].
- [5]GB 548A-96,军用微电子器件试验方法和程序[S].

The Reliable Stress Screening of Electronic Components

YANG Shao-hua¹,WU Fu-gen¹,HUANG Rui-yi²

(1. Engineering and Experiment Center ,Guangdong University of Technology ,Guangzhou 510090 ,China ;

2. The Fifth Research Institute (Electronics) of MII ,Guangzhou 510610 ,China)

Abstract : In this paper ,the failure models of electronic components under a variety of stresses are introduced ,as well as its corresponding reliable stress screening methods. Meanwhile ,the paper mainly emphasizes on the environmental stress screening methods ,accelerated factor and environmental factor ,which provide proper guidance for the electronic products to choose reasonable stress level in reliable stress screening. Finally ,by giving an engineering test example ,the effects of the reliable stress screening are illustrated.

Key words : components ; EES ; accelerated factor ; environmental factor