

基于剩余装载能力的蚁群算法求解同时送取货车车辆路径问题

张 涛¹, 田文馨², 张玥杰³, 刘士新⁴

(1. 上海财经大学 信息管理与工程学院, 上海 200433; 2. 上海金融学院 现代教育中心信息化办公室, 上海 201209;

3. 复旦大学 计算机科学技术学院, 上海市智能信息处理重点实验室, 上海 200433;

4. 东北大学 信息科学与工程学院, 辽宁 沈阳 110004)

摘要: 建立了带车辆最大行程约束的同时送取货车车辆路径问题的混合整数规划模型; 采用了基于排序的蚂蚁系统和最大最小蚂蚁系统的信息素更新策略; 设计了基于车辆剩余装载能力的启发信息策略, 可在满足车辆负载的限制下, 提高车辆的负载利用率; 并在改进阶段使用了节点交换的局部搜索策略, 以提高算法收敛速度. 仿真结果表明本文算法能够在可接受的计算时间内得到满意解.

关键词: 系统工程; 同时送取货的车辆路径问题; 蚁群系统; 混合整数规划

中图分类号: O223 **文献标识码:** A

A residual-loading-capacity-based ant colony system for the vehicle routing problem with simultaneous delivery and pickup

ZHANG Tao¹, TIAN Wen-xin², ZHANG Yue-jie³, LIU Shi-xin⁴

(1. School of Information Management and Engineering, Shanghai University of Finance and Economics, Shanghai 200433, China;

2. Informatization Office in Center of Modern Education Technology, Shanghai Finance University, Shanghai 201209, China;

3. School of Computer Science, Shanghai Key Laboratory of Intelligent Information Processing, Fudan University, Shanghai 200433, China;

4. School of Information Science and Engineering, Northeastern University, Shenyang Liaoning 110004, China)

Abstract: The vehicle routing problem with simultaneous delivery and pickup (VRPSDP) is studied under capacity constraint and maximum distance constraint; and the mixed integer programming model is built. To deal with the fluctuation in vehicle load, we propose an ant colony system (ACS) approach by combining the pheromone updating strategy of rank-based version of the ant system (ASRank) with the MAX-MIN ant system (MMAS). A heuristic factor based on the residual loading capacity is also designed to improve the vehicle loading rate. Additionally, a local search strategy of node-exchange is used in the process of tour improvement to accelerate the searching. Finally, numerical results show that the algorithm provides the desirable solution with high convergence rate.

Key words: system engineering; vehicle routing problem with simultaneous delivery and pickup; ant colony system; mixed integer programming

1 引言(Introduction)

VRPSDP问题由Min^[1]于1989年首次提出, 解决了图书发送与回库问题. Dethloff^[2]首次从逆向物流的角度来研究VRPSDP问题, 建立了VRPSDP的数学模型; Tang和Galvão^[3]提出了具有车辆最大行程约束的VRPSDP问题的数学模型, 并使用禁忌搜索算法以及局部优化算法进行求解.

Anily^[4]证明了VRPB(vehicle routing problem with backhauls)是NP难题, 而当每个顾客的送货需求量或取货需求量两者之一取值为0时, VRPB就

成为VRPSDP问题的一个特例, 因此VRPSDP也是NP难题. 蚁群算法是一种并行的元启发式算法, 全局搜索能力强. Wade和Salhi^[5]利用ACS解决了VRPBM(mixed vehicle routing problem with backhauls)问题; Reimann等^[6]利用基于ACS的算法解决了带时间窗的VRPB问题; Çatay和Gökçe^[7]将改进了的ACS算法用于解决有能力约束的VRPSDP问题, 其中对启发式因子的设计引入了节约式启发算法. 研究表明, 蚂蚁之间通过信息交流加速了进化过程, 具有较强的发现较好解的能力, 因此本文选择蚁群算

收稿日期: 2007-10-22; 收修改稿日期: 2008-12-03.

基金项目: 国家自然科学基金资助项目(70501018, 60773124, 70771020); 上海市自然科学基金资助项目(08ZR1407400); 上海财经大学211工程三期重点学科建设项目.

法作为求解策略。

VRPSDP问题本身的最大复杂性就是在于容易造成车辆超载,为了解决此问题,本文提出基于剩余装载能力的蚁群算法(RLC-ACS, ant colony system based on residual loading capacity),设计了基于剩余装载能力的启发式信息来克服这一难点。为了提高算法的收敛效果,本文在改进阶段使用了局部搜索策略。

2 数学模型(Mathematical model)

引入下面的符号体系:

C : 客户的集合, $C = \{1, 2, 3, \dots, n\}$;

V : 车辆的集合;

A : 顶点的集合, $A = \{0\} \cup C$, $\{0\}$ 代表仓库;

Q : 车辆的负载能力限制;

L : 车辆的行驶距离;

c_{ij} : 客户 i 与客户 j 之间的距离($i \in A, j \in A, (i \neq j); c_{ii} = \infty, (i \neq 0); c_{00} = 0$);

d_i : 客户 i 的送货需求量;

p_i : 客户 i 的取货需求量;

q_{ijk} : 车辆 k 访问完 i 后在访问 j 之前的负载量;

令决策变量 x_{ijk} 定义如下,其中, $k \in V; i, j \in A$.

$$x_{ijk} = \begin{cases} 1, & \text{车辆} k \text{从客户} i \text{到客户} j, \\ 0, & \text{否则.} \end{cases}$$

由如上假设,建立VRPSDP问题的模型如下:

$$\min \sum_{k \in V} \sum_{i \in A} \sum_{j \in A} c_{ij} \cdot x_{ijk}. \quad (1)$$

约束:

$$\sum_{i \in A} \sum_{j \in A} c_{ij} x_{ijk} \leq L, \forall k \in V, \quad (2)$$

$$\sum_{k \in V} \sum_{i \in A} x_{ijk} = 1, \forall j \in C, \quad (3)$$

$$\sum_{i \in A} x_{ijk} - \sum_{i \in A} x_{jik} = 0, \forall j \in A, \forall k \in V, \quad (4)$$

$$\sum_{i \in A} x_{0ik} \leq 1, \forall k \in V, \quad (5)$$

$$q_{0jk} = \sum_{i \in A} \sum_{j \in A} x_{ijk} \cdot d_j, \forall j \in C, \forall k \in V, \quad (6)$$

$$\sum_{k \in V} \sum_{i \in A} q_{ijk} - d_j = \sum_{k \in V} \sum_{i \in A} q_{jik} - p_j, \forall j \in A, \quad (7)$$

$$q_{i0k} = \sum_{i \in A} \sum_{j \in A} x_{ijk} \cdot p_j, \forall i \in C, \forall k \in V, \quad (8)$$

$$q_{ijk} \leq Q, \forall i, j \in A, \forall k \in V, \quad (9)$$

$$q_{ijk} \geq 0, \forall i, j \in A, \forall k \in V, \quad (10)$$

$$\max(c_{j0} + c_{0j}) \leq L \leq \sum_{j \in C} (c_{0j} + c_{j0}), \quad (11)$$

$$\forall j \in C, \quad (11)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq |S| - 1, \forall S \subseteq C, \forall k \in V. \quad (12)$$

其中: 目标(1)表示最小化车辆总行程; 约束(2)保证每辆车的行驶距离都不能超过其最大路程限制; 约束(3)保证每个顾客都要被服务一次且仅一次; 约束(4)保证当一辆车到达某顾客 j , 则离开该顾客的仍然是这辆车, 即顾客 j 只被同一辆车服务; 约束(5)保证每辆车只能使用一次; 约束(6)保证车辆从仓库出发时的载货量等于所服务的客户的送货需求量之和; 约束(7)表示车辆在经过顾客 j 的前后路段上的载货量的变化等式; 约束(8)保证车辆回到仓库的载货量为已服务的客户的取货需求量之和; 约束(9)保证每辆车的任何路段上的负载都不得超过车辆的最大负载限制; 约束(10)保证 q_{ijk} 取非负数. 约束(11)保证车辆可以满足最远路段客户需求; 约束(12)消除子回环。

3 算法设计(Algorithm designing)

蚁群算法是由意大利学者Dorigo首先提出的一种仿生行为算法^[8], M. Dorigo和Gambardella^[9]又提出一种修正的蚁群算法, 并称之为蚁群系统(ant colony system, ACS). 为了克服蚁群算法收敛慢、容易出现停滞现象、算法的运算时间长等缺点, Stutz和Hoos^[10]提出了最大最小蚂蚁系统MMAS; Bullnheimer等^[11]提出了基于等级的蚂蚁系统ASRank; 冯远静提出了一种自适应蚁群算法^[12].

3.1 伪随机概率选择规则(Pseudo-random probability selection rule)

蚂蚁在客户 i 选择客户 j 采用伪随机比率选择规则, 由公式(13)和公式(14)决定:

$$j = \begin{cases} \arg \max_{j \in M_i} \{[\tau(i, j)]^\alpha \cdot [\eta_1(i, j)]^\beta \cdot [\eta_2(i, j)]^\gamma\}, \\ \text{如果 } q < q_0, \\ u, \text{ 否则.} \end{cases} \quad (13)$$

$$P(i, u) = \begin{cases} \frac{[\tau(i, u)]^\alpha \cdot [\eta_1(i, u)]^\beta \cdot [\eta_2(i, u)]^\gamma}{\sum_{j \in M_i} [\tau(i, j)]^\alpha \cdot [\eta_1(i, j)]^\beta \cdot [\eta_2(i, j)]^\gamma}, \\ \text{如果 } u \in M_i, \\ 0, \text{ 否则.} \end{cases} \quad (14)$$

其中: q_0 是一个常数, $q_0 \in (0, 1)$, q 为随机生成的数, $q \in (0, 1)$; M_i 代表蚂蚁在客户 i 处选择下一客户时的可选客户集, $P(i, u)$ 是蚂蚁选择 u 作为下一个要访问的客户的概率。

3.2 信息素局部更新规则(Pheromone local update rules)

信息素局部更新是指对于所有蚂蚁, 每当蚂蚁从客户 i 转移到客户 j 后, 路径 ij 上的信息素量按公式(15)进行局部更新, 其中 $\rho \in (0, 1)$ 为可调参数, 代

表信息素的挥发系数:

$$\tau(i, j) = (1 - \rho) \cdot \tau(i, j) + \rho \cdot \tau_0, \quad (15)$$

其中: $\tau_0 = 1/(n \cdot T_{nn})$, T_{nn} 是按照最近邻域法构造的初始可行解中车辆所用的总行程.

3.3 信息素全局更新策略(Pheromone global update rules)

本文设计的蚁群算法的全局信息更新采用了ASRank的信息素更新方式,即在完成一次循环后,将蚂蚁找到的所有路径按路径长度length从小到大的顺序排列,即 $\text{length}^1 \leq \text{length}^2 \leq \dots \leq \text{length}^m$, m 代表蚂蚁数,并根据最短的rankant条路径长度分别给rankant只蚂蚁的路径赋予不同的权重,路径越短权重越大.全局最优解的权重为 w ,第 r 个最优解的权重为 $\max\{0, w - r\}$.全局更新按公式(16)更新rankant只蚂蚁的路径上各边的信息素:

$$\tau(i, j) = (1 - \rho)\tau(i, j) + \sum_{r=1}^{w-1} (w - r) \cdot \Delta\tau_{ij}^r + w \cdot \Delta\tau_{ij}^{gb}, \quad (16)$$

其中: $\Delta\tau_{ij}^r = 1/\text{length}^r$, $\Delta\tau_{ij}^{gb} = 1/\text{length}^{gb}$, $\rho \in (0, 1)$.为了避免搜索时出现停滞现象,采用MMAS算法的方式,各路径上的信息素量被限制在范围 $[\tau_{\min}, \tau_{\max}]$, $\tau_{\max} = n\tau_0/\rho$, $\tau_{\min} = \tau_{\max}/(2n)$.

3.4 启发式信息(Heuristic function)

Casco等^[13]首先提出基于车辆装载能力的插入准则, Nagy和Salhi^[14]又对插入准则进一步进行了改造. Dethloff提出了“剩余装载能力”这一概念.节点 i 的剩余装载能力指的是从仓库到插入到节点 i 之后的那个节点之间的路径上,车辆的最大允许的额外装载的能力.剩余装载能力包括剩余送货能力 $Rd(i)$ 以及剩余取货能力 $Rp(i)$.

本文借将车辆剩余装载能力引入到蚁群算法的启发信息中,用 Ld_i 表示车辆访问过节点 i 后的装载量, CLd_i 表示车辆访问过节点 i 之后的剩余可装载量.车辆剩余装载能力的计算是对下一个访问节点以及当前路径已访问节点集的一个顺次递归计算,其过程表达如下:

$$Ld_0 = \sum_{s \in T} d_s + d_j, \quad (17)$$

$$Ld_j = Ld_0 - \sum_{s \in T \cup \{j\}} d_s + \sum_{s \in T \cup \{j\}} p_s, \quad (18)$$

$$CLd_j = Q - Ld_j, \quad (19)$$

$$Rd(0) = Q - Ld_0 = CLd_0, \quad (20)$$

$$Rd(j) = \min\{Rd(PRI(j), Q - Ld_j)\} = \min\{Rd(PRI(j), CLd_j)\}, \quad (21)$$

$$Rp(j) = Q - Ld_j = CLd_j, \quad (22)$$

$$TU(i) = Rd(i) + Rp(i), \quad (23)$$

其中 $PRI(j)$ 表示节点 j 的立即前驱.在此基础上,本文将启发信息设计为:

$$\eta(i, j) = \frac{TU(j)}{c_{ij} \cdot Q}, \quad (24)$$

从而总信息公式如下:

$$\text{total}(i, j) = [\tau(i, j)]^\alpha \cdot [\eta(i, j)]^\beta. \quad (25)$$

4 实验结果(Computational results)

为了验证本文算法对VRPSDP问题求解的有效性,进行了实例仿真,测试数据采用了Dethloff算例(客户数均为50,由Tang和Galvão两位学者提供原始数据).实验采用C++语言编程,在PentiumIII计算机上运行.

经过多次试验,利用本文设计的蚁群算法在进行求解以上的算例时,发现参数组合 $\alpha = 1$, $\beta = 0.2$, $q_0 = 0.9$ 在规定迭代次数内可以较快的找到较好解,是一组较为可行的参数.

图1用曲线方式对比了本文设计的RLC_ACS算法与3种不同的算法对于40组Dethloff算例的计算结果;这3种算法分别是: Dethloff的RCRS算法、Çatay B & Gökçe El^[7]的节约ACO(ant colony optimization)算法、Tang和Galvão^[3]的禁忌搜索(tabu search, TS)算法.

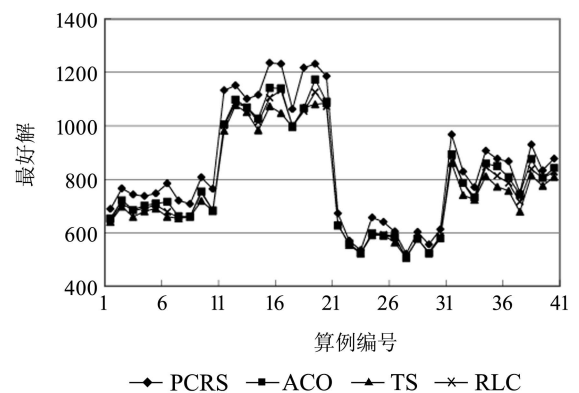


图1 Dethloff算例的结果对比

Fig. 1 Comparison results for the Dethloff Instances

根据图1和实验结果,本文设计的RLC_ACS算法与RCRS算法的求解结果相比,平均改进率是13.56;而与节约ACO算法的求解结果相比,平均改进率为1.04;与禁忌搜索算法的求解结果基本相当.从计算时间上来看,RLC_ACS的计算时间明显比Tang和Galvão的禁忌搜索算法长,尽管如此

该计算时间也是可以接受的. 从车辆数上来看, RLC_ACS所求解的车辆使用数与禁忌搜索算法的结果是相同的. RLC_ACS算法在蚁群算法中引入一些改进的启发式算法的原理, 使之更加适合VRPSDP问题的特性, 可以有效地提高蚁群算法用于求解VRPSDP问题的性能.

图2是对Dethloff的SCA3_0这一组算例, 采用本文RLC_ACS算法求得的车辆负载量曲线图(除最后一辆车以外).

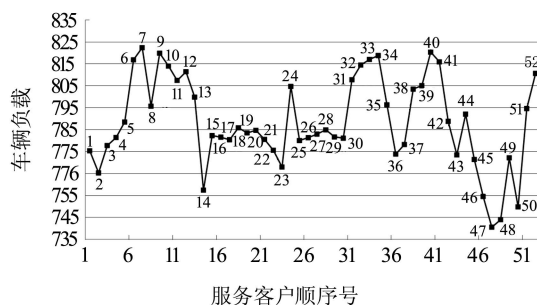


图2 车辆负荷情况变化曲线

Fig. 2 Curve of the vehicle load

从图2中可以看出, 每辆车的当前负载量呈现波动变化. 从总体上看, 车辆的最小负载量为740.5, 最大负载量为822.4, 平均负载量大约为785, 与车辆负载限制823.7相差38左右. 可见在满足最大负载能力的约束下, 车辆在途的负载量一直持续了较高的水平, 使车辆空间使用率得到了有效的利用, 也说明了对负载量波动这一难点的有效解决, 可见本文提出的RLC_ACS算法中对启发信息的重新构造方法在原理上是可行的, 在运算结果中也获得了良好效果.

5 结论(Conclusion)

本文设计了解决VRPSDP问题的基于剩余装载能力的蚁群算法(RLC_ACS), 结合了ASRank和MMAS的信息素更新规则, 并将Dethloff提出的车辆剩余装载能力的概念引入到蚁群算法的启发信息中, 以剩余装载能力作为路径安排中的考虑因素, 提高了路径安排中车辆对客户访问的自由度. 本文还设计了基于交换操作的局部优化算法, 对蚁群算法的每一次迭代中得到的最好解进行优化, 提高了蚁群算法的寻优能力. 实验结果表明, 本文设计的RLC_ACS算法对于求解VRPSDP问题有很好的求解性能, 在可接受的计算时间内RLC_ACS算法能够得到与Tang和Galvão相近的满意解.

参考文献(References):

- [1] MIN H. The multiple vehicle routing problems with simultaneous delivery and pick-up points[J]. *Transportation Research A*, 1989, 23(5):

377 – 386.

- [2] DETHLOFF J. Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up[J]. *OR Spektrum*, 2001, 23(1): 79 – 96.
- [3] TANG MONTANÉ FA, GALVÃO RD. A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service[J]. *Computer & Operations Research*, 2006, 33(3): 595 – 619.
- [4] ANILY S. The vehicle-routing problem with delivery and back-haul options[J]. *Naval Research Logistics*, 1996, 43(3): 415 – 434.
- [5] WADE A, SALHI S. An ant system algorithm for the mixed vehicle routing problem with backhauls[M] // *Metaheuristics: computer decision-making*. Netherlands: Kluwer Academic Publishers, 2004, 699 – 719.
- [6] REIMANN M, DOERNER K, HARTL R F. Insertion based ants for vehicle routing problems with backhauls and time windows[C] // *ANTS: 2002 Lecture Notes in Computer Sciences*. Berlin: Springer, 2002: 135 – 148.
- [7] ÇATAY B, GÖKÇE E I. An ant colony system approach for the capacitated vehicle routing problem with simultaneous delivery and pick-up[R]. Working Paper, Tuzla, Istanbul: Sabanci University, 2005.
- [8] DORIGO M, MAINEZO V, COLONI A. Positive feedback as a search strategy[R]. Technical Report 91-016, Milano, Italy: Dipartimento di Elettronica, Politecnico di Milano, 1991.
- [9] DORIGO M, GAMBARDELLA L M. Ant colonies for the travelling salesman problem[J]. *BioSystems*, 1997, 43(2): 73 – 81.
- [10] STUTZLE T, HOOS H. Max-min ant system and local search for the traveling salesman problem[C] // *Proceedings of IEEE-ICEC-EPS'97, IEEE International Conference on Evolutionary Computation and Evolutionary Programming Conference*. New York: IEEE, 1997: 309 – 314.
- [11] BULLNHEIMER B, HARTL R F, STRAUSS C. A new rank based version of the ant system: a computational study[J]. *Central European Journal for Operation Research and Economics*, 1999, 7(1): 25 – 28.
- [12] 冯远静, 冯祖仁, 彭勤科. 一类自适应蚁群算法及其收敛性分析[J]. *控制理论与应用*, 2005, 22(5): 713 – 717.
(FENG Yuanjing, FENG Zhuren, PENG Qinke. Adaptive ant colony optimization algorithms and its convergence[J]. *Control Theory & Applications*, 2005, 22(5): 713 – 717.)
- [13] CASCO D, GOLDEN B, WASIL E. Vehicle routing with backhauls: models, algorithm, and case studies[M] // *Vehicle Routing: Methods and Studies*. Amsterdam: North-Holland, 1988: 121 – 147.
- [14] NAGY G, SALHI S. Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries[J]. *European Journal of Operational Research*, 2005, 162(1): 126 – 141.

作者简介:

张涛 (1970—), 男, 博士, 副教授, 主要研究方向为物流建模与优化、生产计划与调度、智能优化算法, E-mail: taozhang@mail.shufe.edu.cn;

田文馨 (1981—), 女, 硕士, 主要研究方向为供应链与物流、智能优化算法;

张明杰 (1973—), 女, 博士, 副教授, 主要研究方向为智能优化算法、机器学习, E-mail: yjzhang@fudan.edu.cn, 本文通讯作者;

刘士新 (1968—), 男, 博士, 教授, 主要研究方向为物流建模与优化、智能优化算法.