# 用蚁群算法求解需求可拆分车辆路径问题*

隋露斯， 唐加福， 潘震东， 刘树安

东北大学 信息学院 流程工业综合自动化教育部重点实验室，沈阳，110004
E-mail:suilusi@163.com

**摘 要**:传统的车辆路径问题是组合优化中一个典型的 NP 难题，都假设条件客户需求不可拆分，但在实际的物流运作中，有时通过需求的拆分可以更好的降低运输成本。本文尝试用蚁群算法来求解需求可拆分车辆路径问题，给出了基于整数规划的描述方法；针对模型的特点，对蚁群算法进行了改进，设计了一种针对需求可拆分车辆路径问题的新的蚁群优化算法（Ant Colony Optimization-ACO）。并对需求可拆分的车辆路径问题（SDVRP）问题进行仿真试验，将其结果与传统 VRP 作比较，得到了满意的结果。

**关键字**:可拆分车辆路径问题，蚁群优化算法，物流

# Ant Colony Optimization Algorithm to Solve Split Delivery Vehicle Routing Problem

SUI Lu-si, TANG Jia-fu, PAN Zhendong and LIU Shu-an

Key Lab of Integrated Automation of Process Industry of MOE, Northeastern University, Shenyang, 110004, China
E-mail:suilusi@163.com

**Abstract**: As a combinatorial optimization problem, vehicle routing problem (VRP) is a typical NP-hard problem; an assumption that the demand of customers can not be split is given to the traditional VRP formulation. However, the transportation cost can be reduced by means of splitting the demand of customers in practical logistics operation. This paper solved the split delivery vehicle routing problem (SDVRP) by means of the ant colony algorithm. The integer programming model of the problem is given in this paper, and a corresponding improved ant colony optimization algorithm (ACO) based on characteristic of the problem is designed to solve the problem. Some simulation experiments on this algorithm and in comparison with the traditional VRP are conducted, which show that the proposed ACO algorithm is effective and the SDVRP can reduce more total costs than traditional VRP.

**Key Words**: the split delivery vehicle routing problem (SDVRP) ant colony algorithm (ACO) logistic

## 1．引言（INTRODUCTION）

车辆调度是物流管理中最重要的部分。随着社会的发展以及消费者对服务质量要求的不断提高，高效的车辆调度寻找出一条最优的车辆行驶路线，不但有利于降低物流中货物库存的费用，缩短商品上市周期，而且逐渐成为物流成败的关键。研究车辆路径问题，进行车辆路线优化，提高物流效率、降低物流成本和提高服务质量对于促进经济健康稳定的发展具有重要意义。

传统的车辆路径问题（Vehicle Routing Problem，简称 VRP）最早于 1959 年由 Dantig 和 Ramser[1]提出，并一直是网络最优化问题中最基本的问题之一，由于其应用的广泛性和经济上的重大价值，多年以来始终受到国内外学者的广泛关注。该问题一般可以定义为：对一系列装货点和（或）卸货点，组织适当的行车路线，使得车辆按照一定的顺序通过他们，在满足一定的约束条件（如货物的需求量、发送量、交发货时间、车辆容量或载重限制、行驶里程限制、时间限制等）下，达到一定的目标（如路程最短、费用最少、时间尽量少、使用车辆尽量少等）[2]。

目前车辆路径问题的研究主要分为两种：非满载和满载。这两种是根据需求量和车辆载重量来划分的，而且目前大多数人研究的是非满载的情况。非满载的车辆路径问题研究的是每个客户点的需求量仅由一辆车来完成，也就是说需求是不可拆分的。但是在实际情况中，往往会出现部分任务或者全部任务的需求量都较大的情况，此时如果仍然要一辆车来完成任务的话必然会使车辆的空载率提高，从而造成车辆资源的浪费。因此，研究需求可拆分的车辆路径问题有很好的现实意义，而且越来越多的学者开始研究这个问题。

## 2．SDVRP 问题的整数规划模型（INTEGER PROGRAMMING MODEL of SDVRP）

具有拆分的车辆路径问题 SDVRP(Split Delivery VRP) [7]是 Dror 和 Trudeau 在 1989 年引入到车辆路径问题中的。与传统 VRP 的不同，SDVRP 能用两辆

* 基金项目：国家自然科学基金 （70625001，70721001）、教育部新世纪优秀人才支持计划项目（NCET-04-0280）。

或多辆车对一个客户进行服务。因此，很多情况下，SDVRP 能够明显减少配送车辆数和缩短运行总路径。当客户的平均需求超过车辆最小运载力的 10%时（Dror et tal.,1994.），这种优势更明显。

SDVRP 就是最少化所有车辆的行驶路径的距离，并满足下面的限制：

（1）客户点的货物由一辆或多辆车完成配送。

（2）每辆车从配送中心出发，最后回到配送中心。

（3）每辆车一次配送的客户货物总量小于车的最大运载量。

（4）每条总路径的长度不超过车辆一次配送的最大行使距离限制。

由于 SDVRP 具有客户可分的宽松条件，因此 SDVRP 能够在很大程度上实现车辆的满载，减少车辆的数量，有效寻求更短的路径。

为方便起见，本文假设对客户的需求拆分是整数拆分， 事实上，SDVRP 的整数假设完全符合实际，如现实生活中的人员运输、机场的航空集装箱运输、大型牲畜运输等都可以运用整数规划模型[4]。这种情况下，SDVRP 可以表示为赋权图 $\boldsymbol{G}=(V,E)$，点集 $V=\{0,1,2,...,N\}$ 和边集 $E=\{(i,j) \quad i,j\in V, i\neq j\}$，在点集中，0 表示配送中心，其它点表示 N 个客户，点 $i(i=0,1,...,N)$ 表示为配送中心和所有的客户点，在边集中，对应每条边 $(i,j)$，非负值 $c_{ij}$ 表示从客户 i 到点 j 的距离，$q_i$ 代表客户点 i 的货物需求量并且 $q_0=0$，Q 代表车辆的最大运载量。K 表示车辆的总数量，U 为完成配送的最小车辆数，

其中 $U=\left\lceil \dfrac{\sum_{i=1}^{N} q_i}{Q} \right\rceil$，$\lceil\ \rceil$ 表示向上取整数。

定义变量如下：

$$y_{ki}=\begin{cases}1 & \text{客户点}i\text{的任务由车辆}k\text{完成}\\ 0 & \text{否则}\end{cases}$$

$$x_{ijk}=\begin{cases}1 & \text{车辆k从点i 行驶到}j\\ 0 & \text{否则}\end{cases}$$

$w_{ik}$=车辆 $k$ 在客户 i 点的配送量 $(w_{0k}=0)$

基于此，SDVRP 可以描述为如下整数规划模型：

$$min\,Z=\sum_i\sum_j\sum_k c_{ij}x_{ijk} \qquad (1)$$

s.t.

$$w_{ik}\le q_i y_{ik} \qquad \forall i,\forall k \qquad (2)$$

$$\sum_{k=1}^{K}\sum_{i=0}^{N} x_{ijk}\ge 1 \qquad j=0,1,...,N \qquad (3)$$

$$\sum_{i}^{N} x_{ipk}-\sum_{j}^{N} x_{pjk}=0 \qquad p=0,1,...,N\text{；}k=1,2,...,K \qquad (4)$$

$$\sum_{k=1}^{K} w_{ik}=q_i \qquad i\in V \qquad (5)$$

$$\sum_{i=1}^{N} w_{ik}\le Q \qquad k=1,2,...,K \qquad (6)$$

$$w_{ik}>0 \qquad w_{ik}=1,2,...,Q\text{-}1\text{；}\ k=1,2,...,K \qquad (7)$$

$$\sum_{i\in s}\sum_{j\in s} x_{ijk}\le |S|-1 \qquad 2\le |S|\le n-1 \qquad (8)$$

$$x_{ijk}=0\text{或}1 \qquad i,j=0,1,...,n; \qquad \forall k \qquad (9)$$

$$y_{ki}=0\text{或}1 \qquad i=0,1,...,n; \qquad \forall k \qquad (10)$$

在整数规划模型中，式（1）为优化的目标，表示求车辆总成本最低，即路线总长度最短；式（2）表示第 $k$ 辆车只为访问的客户服务，配载量小于等于客户货物需求，式（3）表示每个点至少车辆要经过一次，式（4）表示流量守恒，即进入某点的车辆数与离开这点的车辆数相等；式（5）表示客户点所有的需求任务必须完成，式（6）表示车辆载重约束，即某条路线上车辆的总运货量不大于车辆的载重量限制，式（7）限制客户需求为整数拆分，式（8）消除了支路限制条件。

## 3．求解 SDVRP 问题的蚁群算法(ANT COLONY ALGORITHM of SDVRP)

### 3.1 算法描述(Algorithm Description)

蚁群算法（Ant Colony Algorithm）是 20 世纪 90 年代 首先由意大利科学家 M.Dorigo[3]等人提出来的，是一种模拟自然界蚂蚁觅食行为的新型的模拟进化算法，它通过信息素（pheromone）的积累和更新来寻找最优解[8]。蚂蚁有能力在没有任何提示下找到从巢穴到食物源的最短路径，并且能随环境的变化而变化，适应性地搜索新的路径产生新的选择，其根本原因就是：蚂蚁觅食时，会在所经过路线上留下信息素，其他蚂蚁可以并且习惯追踪此信息素爬行。在确定位置的食物和蚁穴之间，路线越近，蚂蚁重复爬行的次数就越多，由于每只蚂蚁每经过一次都要释放信息素，这样重复次数多的路线由于其信息素浓度较大就更容易被其他蚂蚁选中，这样整个蚁群就由开始的多路线爬行逐渐集中到最短的路线上爬行，使路线得到优化选择[5]。

为了模拟实际蚂蚁的行为[6]，引进如下记号：

设共有 m 只蚂蚁在配送中心和货物需求点组成的图 $\boldsymbol{G}=(V,E)$ 中运动，$c_{ij}(i,j=1,2,...N)$ 表示城市 i 和城市 j 之间的距离，$\tau_{ij}(t)$ 表示 t 时刻在城市 i 和城市 j 连线上残留的信息素。初始时刻，各条路径上信息素相等，设 $\tau_{ij}(0)=C$（C 为常数）。蚂蚁 k(k=1,2,…,m)在运动过程中根据各条路径上的信息素决定转移方向。$p_{ij}^{k}(t)$ 表示在 t 时刻蚂蚁 k 由城市 i 转移到城市 j 的概率：

$$p_{ij}^{k}=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\alpha}(\eta_{ij})^{\beta}}{\sum\limits_{k\notin tabuk}[\tau_{ik}(t)]^{\alpha}(\eta_{ik})^{\beta}}, j\in allowed_k \\ 0 \quad\quad \text{其他}\end{cases} \quad (11)$$

其中：$\eta_{ij}$ 为先验知识或称为能见度，在这里 $\eta_{ij}$ 为蚂蚁 k 由城市 i 转移到城市 j 的启发信息，一般取 $\eta_{ij}=1/c_{ij}$，$\alpha$ 为在路径 ij 上残留信息的重要程度；$\beta$ 为启发信息的重要程度；$allowed_k=\{N-tabu_k\}$，$tabu_k(k=1,2,...,m)$ 用以记录蚂蚁 k 当前所走过的城市，称为禁忌表（下一步不允许选择的城市）。集合 $tabu_k$ 随着进化过程作动态调整。

经过n个时刻，所有蚂蚁都完成了一次周游。它们本次周游的禁忌表将满，此时应清空，将当前蚂蚁所在城市置入 $tabu_k$，准备下一次周游。这时，计算每一只蚂蚁所走过的路径 $L_k$，并保存最短路径

$L_{k\,min}(L_{k\,min}=min\,L_k,k=1,...,m)$。

随着时间的推移，以前留下的信息逐渐消逝，用参数 $\rho$ 表示信息素的持久性，蚂蚁完成一次循环以后，各路径上的信息素要根据式(12)作调整：

$$\tau_{ij}(t+\Delta t)=\rho\tau_{ij}(t)+\Delta\tau_{ij},\ \Delta\tau_{ij}=\sum_{1}^{m}\Delta\tau_{ij}^{k} \quad (12)$$

$\Delta\tau_{ij}^{k}$ 表示第k只蚂蚁在本次循环中留在路径ij上的信息素，$\Delta\tau_{ij}$ 表示本次循环中路径ij上的信息素的增量。此处的 $\Delta\tau_{ij}$ 根据信息素更新策略的不同，有3种不同的算法模型[9]。

（1）Ant- cycle 模型

$$\Delta\tau_{ij}^{k}=\begin{cases}C/L_k, & \text{第k只蚂蚁在t和t+1时刻经过ij}\\ 0, & \text{其他}\end{cases}$$

其中C是常量，信息素增量只与蚂蚁k的循环路线 $L_k$ 和C有关系。而和具体的 $c_{ij}$ 无关。

（2） Ant- quantity 模型

$$\Delta\tau_{ij}^{k}=\begin{cases}C/c_{ij}, & \text{第k只蚂蚁在t和t+1时刻经过ij}\\ 0, & \text{其他}\end{cases}$$

其中C是常量，信息素的增量与城市ij之间的距离有关。

（3）Ant- density模型

$$\Delta\tau_{ij}^{k}=\begin{cases}C, & \text{第k只蚂蚁在t和t+1时刻经过ij}\\ 0, & \text{其他}\end{cases}$$

其中C是常量，信息素只增加一个固定值，与城市i和城市j之间的距离无关。

最后，设置最大迭代次数NC，记录NC次中 $L_{k\,min}$ 的最小值为最终的最优路径 $L_{min}$。

### 3.2 SDVRP 的蚁群算法设计(The Design of the Ant Colony Algorithm of SDVRP )

#### 3.2.1 蚁群算法设计(The Design of the Ant Colony Algorithm)

求解SDVRP的蚁群算法建立在求解传统VRP的蚁群算法的基础上，根据SDVRP问题的本身的特点做一些修改。

在SDVRP中，客户的货物量可以分派给两辆车或多辆车进行配送。因此，在蚁群算法中，蚂蚁的一次迭代构造解路径的过程中，需要对客户的货物量进行分割。正常情况下，蚂蚁选择下一个客户进行移动时，若所有满足限制的客户按选择机制选择下一个客户，当所有满足限制的客户选择完成后，蚂蚁返回配送中心。但在SDVRP问题中，若车辆还有运载能力，这时不是返回配送中心开始新一轮配送，而是从剩下的客户中选择一个客户进行分割，将客户的一部分货物分给车辆进行配送。

在分割时，为了达到最短路径的最优，蚂蚁按照一定的规则选择客户对客户进行分割。分割过程叙述如下。

和传统VRP蚁群算法一样，如果 $allowed_k$ 不为空，依据转移概率 $p_{ij}$ 式（11）进行客户选择，服务完这些点，$allowed_k$ 中为所有剩余货物量小于车辆当前的剩余运载量的客户。当 $allowed_k$ 为空时，如果车辆的剩余运载量大于零，这时 $allowed_k$ 变成了所有未被服务完的点，按照下面规则式（13）从 $allowed_k$ 中选则一个客户进行配送。

$$min\{c_{ij}+c_{j0}\} \quad j\in allowed_k \quad (13)$$

这个规则就是保证车辆到选中的客户且返回配送中心的距离最短。选中客户后对选中客户进行货物需求更新，然后返回配送中心开始。

#### 3.2.2 算法步骤(Algorithm steps)

Step 1 系统初始化：初始化距离 $c_{ij}$，设置初始迭代次数nc=0，设置 $\tau_{ij}(0)=C$，（C为环境初始残留信息素）

Step 2 判断 $tabu_k$ 表是否已满，是则转Step 7 。否则转Step 3

Step 3 增加一辆车，

Step 4 根据轮盘赌规则选取可选城市，若所选城市的需求量小于等于车辆剩余的运载能力，则全部装车，将该城市列入 $tabu_k$ 和 $tabucar_k$ 中转Step 5；若所选城市的需求量大于车辆剩余的运载能力，转Step 6

Step 5 车向前行驶一步，判断 $tabu_k$ 表是否已满，是则转Step 7 。否则转Step 4

Step 6 若所选城市的需求量大于车辆剩余的运载能力，按照规则式（13）从剩余城市中选择距离最短的城市，将车装满转Step 2

Step 7 记录历史最优解

Step 8 对信息素进行更新，并清空 $tabu_k$ 表和 $tabucar_k$ 表，进行跌代，直到最大跌代次数停止。

## 4. 仿真结果与分析(SIMULATION RESULTS AND ANALYSIS )

为了描述现实情况的复杂性和不确定性，城市点的坐标采用随机产生的方式，客户点的需求量也采用随机产生的方式。设车辆的最大运载能力为 500，以下试验的基本参数设置为迭代次数 NC=1000，$\alpha$ =2，$\beta$ =3，$\rho$ =0.7，C=1。下面以 15 个客户点（配送中心的坐标为原点）为例详细比较传统 VRP 和 SDVRP。随即产生 15 个客户点，总需求量为 4881，数据如表 1 所示。

表 1 15 个客户点的基本信息

| 客户点 | 横坐标 | 纵坐标 | 需求量 |
|---|---|---|---|
| 1 | 32 | 41 | 468 |
| 2 | 96 | 9 | 335 |
| 3 | 7 | 58 | 1 |
| 4 | 97 | 87 | 170 |
| 5 | 26 | 21 | 225 |
| 6 | 23 | 100 | 479 |
| 7 | 52 | 31 | 359 |
| 8 | 76 | 43 | 463 |
| 9 | 74 | 17 | 465 |
| 10 | 72 | 104 | 206 |
| 11 | 40 | 99 | 146 |
| 12 | 8 | 16 | 282 |
| 13 | 27 | 38 | 328 |
| 14 | 78 | 69 | 462 |
| 15 | 46 | 16 | 492 |

经过计算后 15 个客户点的最优路径结果如表 2 所示，表 2 的结果表明在一定情况下使用 SDVRP 可以在减少运输车辆数的同时缩短最优路径，

表 2 15 个客户点的最优路径结果

| 传统 VRP 的路径表<br>城市（需求量） | | SDVRP 的路径表<br>城市（需求量） | |
|---|---|---|---|
| 0 ——12（282）—11（146）——3（1） | | 0 ——12（282）—3（1）——11（146）—13（71） | |
| 0 ——15（492） | | 0 ——15（492）—5（8） | |
| 0 ——5（225）—10（206） | | 0 ——9（465）—5（35） | |
| 0 ——1（468） | | 0 ——5（182）—13（257）——1（61） | |
| 0 ——7（359） | | 0 ——1（407）—7（93） | |
| 0 ——13（328）—4（170） | | 0 ——8（463）—7（37） | |
| 0 ——9（465） | | 0 ——14（462）—7（38） | |
| 0 ——2（335） | | 0 ——2（335）—7（165） | |
| 0 ——8（463） | | 0 ——7（26）—4（170）——10（206）—6（98） | |
| 0 ——14（462） | | 0 ——6（381） | |
| 0 ——6（479） | | | |
| 总车辆数 | 最短路径 | 总车辆数 | 最短路径 |
| 11 | 1990 | 10 | 1836 |

经过多次试验还可以证明客户点的数量越多，SDVRP 越能显示出它的优越性，从而节约的资源越多，如表 3 所示。

表 3 在城市点较多情况下的传统 VRP 和 SDVRP 的比较

| 客户点数 | 传统 VRP | | SDVRP | |
|---|---|---|---|---|
| | 最短路径 | 最小车辆数 | 最短路径 | 最小车辆数 |
| 50 | 5936 | 33 | 5489 | 30 |
| 70 | 8044 | 43 | 7172 | 39 |
| 90 | 11090 | 55 | 10034 | 50 |
| 100 | 13180 | 61 | 10952 | 54 |
| 200 | 25198 | 113 | 20580 | 104 |

面对一个传统的 VRP，如果客户允许进行需求拆分，那么是否在路径上安排和在车辆调度中进行需求拆分是摆在决策者面前的一个主要问题，图 1 和图 2 以 80 个顾客点为例，不考虑客户的需求分布，假设所有客户的需求 $q_i$ 都是相等的，物流中心有足够的车辆来进行服务，每辆车的满载量 Q 都是 500，下面的实验数据对比了分别用传统 VRP 和 SDVRP 求解问题的最短路径和最少车辆数。

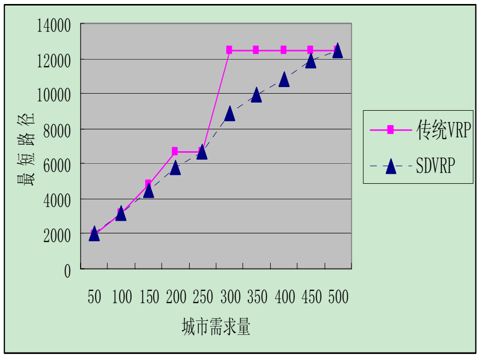


图 1 传统 VRP 和 SDVRP 的最短路径比较

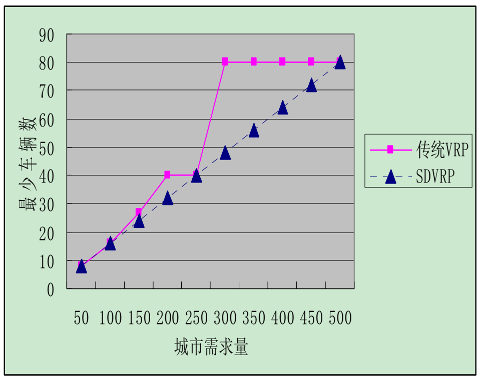


图 2 传统 VRP 和 SDVRP 的最少车辆数比较

从图 1 和图 2 可以看出，当 $(Q+1)/2 < q_i < 2(Q+1)/3$ 时[10]，两种模型的差值最大，所以在这个时候采用 SDVRP 模型对于车辆数和路径的节约最为显著。

## 5．结语(CONCLUSION)

本文主要研究了需求可拆分的物流车辆路径问题，给出了他们的问题描述和数学模型，根据问题本身的特点，应用蚁群算法对问题进行求解与仿真，并将 SDVRP 与 VRP 问题的结果作比较，得到了较好的结论。可见研究需求可拆分的车辆路径问题具有一定的实际价值，在一定情况下对客户的需求进行拆分，能有效的节约车辆资源、缩短最优路径。

## 参考文献

[1] Dantzig G. and Ramser J., “The truck dispatching problem,” Management Science, 1959(6), pp.80.

[2] 李军，郭辉煌. 物流配送——车辆优化调度理论与方法. 中国物资出版社，2001. 6.

[3] Dorigo M. and Gambardella L M, “Ant Algorithms for Discrete Optimization,” Artificial Life, 1999, 5(3), 137–172.

[4] 赵元，客户可分的车辆路径问题研究，上海交通大学硕士论文，2006.

[5] 廖兴新，蚂蚁算法在 TSP 问题中的应用和研究. 四川大学硕士论文，2006（5）.

[6] 柳林 朱建荣，基于混合蚂蚁算法的物流配送路径优化问题研究，计算机工程与应用，2006，203－205.

[7] Ho S C and Haugland D, “A tabu search heuristic for the vehicle routing problem with time windows and split deliveries,” Computers & Operations, 2004, 31, 1947-1964.

[8] Cordón O, Herrera F and Stützle T, “A Review on the Ant Colony Optimization Metaheuristic: Basis, Models and New Trends,” Mathware and oft Computing, 2002, 9.

[9] 张宏达，郑全弟. 基于蚁群算法的 TSP 的仿真与研究. 航空计算技术，2005, 04, 108-111.

[10] 谢毅，需求可拆分的车辆路线问题研究，同济大学硕士论文，2006.