

基于聚类的城市共同配送海量订单调度问题研究^①

谢天保¹, 赵 萌¹, 雷西玲²

¹(西安理工大学 经济与管理学院, 西安 710054)

²(西安理工大学 计算机科学与工程学院, 西安 710054)

摘要: 针对云物流环境下城市共同配送海量订单调度难的问题, 本文提出基于订单聚类的调度算法. 首先针对单中心多车辆调度问题, 提出基于单亲遗传的优化调度算法; 在此基础上综合考虑城市配送中心的地理位置、车辆及配送点的地理位置、货物的种类、需求量, 提出采用蚁群算法构建基于配送中心的海量订单聚类、优化调度算法.

关键词: 共同配送; 海量订单; 聚类; 蚁群算法; 调度

引用格式: 谢天保, 赵萌, 雷西玲. 基于聚类的城市共同配送海量订单调度问题研究. 计算机系统应用, 2017, 26(7): 232-237. <http://www.c-s-a.org.cn/1003-3254/5837.html>

Research on Massive Orders Scheduling Problem of Urban Joint Distribution Based on Clustering

XIE Tian-Bao¹, ZHAO Meng¹, LEI Xi-Ling²

¹(School of Economics and Management, Xi'an University of Technology, Xi'an 710054, China)

²(School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710054, China)

Abstract: In order to solve the problem of massive orders scheduling in the cloud logistics environment, this paper proposes a scheduling algorithm based on orders clustering. Firstly, aiming at the single center multi vehicle scheduling problem, an optimal scheduling algorithm based on the single parent genetic algorithm is proposed. On this foundation, considering the location of the city distribution centers, the vehicles and the distribution points, the type and the demand of goods, an order clustering model based on distribution center is built by using the ant colony algorithm.

Key words: joint distribution; massive orders; clustering; ant colony algorithm; scheduling

1 引言

随着信息技术的高速发展, 我国的城市物流配送正在进行着变革, 传统的物流配送已经不能满足现代城市物流发展的需求, 加快物流信息的共享互通、降低物流成本、统筹优化社会物流资源、提高城市物流配送效率是新环境下城市物流配送亟需解决的问题. 而云物流是基于云计算强大的通信能力、运算能力和匹配能力^[1], 建立物流信息共享平台, 实现智能匹配、智能组合、智能管理物流服务等先进功能^[2]. 基于云物流的物流信息共享平台, 是对整个物流网络中的信息进行收集、处理、传递、共享的集中地, 其主要思想就是: 在物流信息充分共享的前提下, 对供货方和收货

方之间交易的海量物流订单、各类企业和第三方物流企业现有的零散物流资源进行整合, 依靠云计算技术对这些信息进行处理和挖掘, 实行科学合理的订单规划和车辆调度进行管理, 实施共同配送, 使得共同配送信息平台的各方参与者都能获取更多的利益, 从而吸引更多企业的物流订单和物流资源加入到平台中, 形成一个平台开放、资源共享、终端无限的网络. 因此, 研究云物流下城市共同配送海量订单调度问题, 有效缓解城市交通压力、减少城市污染, 对提高城市物流管理水平、改善城市居民的生存环境具有重要的意义, 也对云物流信息共享平台的实际建设与应用具有推进作用.

^① 基金项目: 陕西省教育厅人文社科重点研究基地科研计划项目(15JZ039)

收稿时间: 2016-10-28; 收到修改稿时间: 2016-11-30

物流配送车辆优化调度问题最早由DnaZtig和Ramser于1959年首次提出^[3]。车辆优化调度问题根据时间特征的差异可分为车辆路径规划问题VRP(Vehicle Routing Problem)、车辆调度问题VSP (Vehicle Scheduling Problem)以及有时间窗的车辆路径问题VRPTW (Vehicle Routing Problem with Time Windows)^[4], 属于一个NP-hard问题, 其模型建立针对一个配送中心而且只有当其规模较小时, 才能求其精确解。而对于多中心协同配送调度问题, 在以往的国内外研究中^[5-11], 文献[5-6]首先将多车场问题转化为单车场问题, 用分解算法处理多车场配送问题; 文献[7]提出了一种基于网络流模型最优解的启发式算法, 来探讨多车场满载货运车辆的优化调度问题; 文献[8]运用了禁忌搜索法解决车辆配送路线问题; 文献[9]采用了C-W节约算法对有时间窗约束的非满载车辆调度进行路线安排; 文献[10]提出用神经网络算法来求解车辆配送路线优化问题; 文献[11]建立了一种基于最小配送费用的数学模型来研究多车场多车型车辆调度问题, 并借助遗传算法对模型进行求解。云物流模式下城市共同配送涉及到海量订单、众多配送中心以及各种类型车辆调度, 再加上多种约束条件(时间窗约束、类别), 显然是一个复杂的NP-Hard问题, 同时考虑到云物流模式下的订单配送具有多频次、少批量、多品种、零散化、个性化、订单动态性、时效性高、随机性强等特点, 这些特点无疑增加了云物流海量订单调度模型求解的复杂性, 为此本文提出基于多中心聚类结合单中心多车辆调度算法实现海量订单的优化调度, 根据物流订单的特性(物流中心空间距离、货物类型等)采用聚类分析^[12], 分别计算各中心(类)的配送成本, 以上过程结合改进蚁群算法对海量订单反复迭代聚类分析, 最终可求取城市海量物流订单的最优调度方案, 便于云物流分布式调度的实现。

2 基于聚类的海量订单共同配送调度模型

假设某城市现有物流中心 M 个, 用变量 $m(m=1, 2, \dots, M)$ 表示, 考虑到城市居民需求的动态性, 这些物流配送中心在城市区域均匀分布, 其坐标为 (X_m, Y_m) , 每个物流中心拥有多个不同类型的运输车辆, 平均车速 V , 物流中心的地理信息、及车辆信息通过云服务收集于城市共同配送云物流平台。在某一时刻(通常周期性处理), 城市物流平台收集到 N 个订单, 货物配送点随机

分布于城市覆盖区域, 根据 M 个配送中心的地理位置信息、各自的车辆信息、配送点地理位置信息, 如何从 M 个配送中心选择出若干个配送中心及车辆负责海量订单的配送任务, 合理调度车辆资源、提高运输车辆的配载率, 缓解城市交通拥堵, 降低城市物流配送成本是本文研究的目标。

2.1 问题分析及变量说明

在云物流城市共同配送模式下, 配送的总成本主要包括固定成本和运输成本。固定成本包括车辆动用成本和人力资源成本等, 在基于时间窗约束车辆调度^[13]的数学模型中, 这里采用硬时间窗, 即必须在规定时间段送达。下面首先对模型中的变量进行说明, 然后分析城市共同配送活动的成本。

(1) 配送中心信息描述:

$m(m=1, 2, \dots, M)$ 表示配送中心编号;

$(X_m, Y_m)(m=1, 2, \dots, M)$ 表示配送中心 m 的地理坐标;

$f(m, k)(m=1, 2, \dots, M, k=1, 2, \dots, K)$ 表示配送中心 m 的车辆编号;

$g(f(m, k))(m=1, 2, \dots, M, k=1, 2, \dots, K)$ 表示 m 配送中心第 k 辆车的最大载重量;

$C(f(m, k))(m=1, 2, \dots, M, k=1, 2, \dots, K)$ 表示 m 配送中心第 k 辆车固定成本(包括人力成本);

$UC(f(m, k))(m=1, 2, \dots, M, k=1, 2, \dots, K)$ 表示 m 配送中心第 k 辆车每公里消耗成本。

(2) 配送点信息描述:

$n(n=1, 2, \dots, N, N \gg M)$ 表示配送点编号;

$(X_n, Y_n)(n=1, 2, \dots, N)$ 表示配送点 n 的地理坐标;

$Pg(n)(n=1, 2, \dots, N)$ 表示配送点 n 所需货物的重量;

$Tp(n)(n=1, 2, \dots, N)$ 表示配送点 n 所需货物的类别, 不同类别的货物不能同车, 例如蔬菜和农药;

$st(n)(n=1, 2, \dots, N)$ 表示配送点 n 要求送货的最早时间;

$et(n)(n=1, 2, \dots, N)$ 表示配送点 n 要求送货的最迟时间;

$tt(n)(n=1, 2, \dots, N)$ 表示配送点 n 卸货消耗时间;

$D(i, j)(i=1, 2, \dots, N, j=1, 2, \dots, N)$ 表示配送点 i 和配送点 j 之间的距离;

$DHS(i, m)$ 表示配送点 i 和配送中心 m 之间的距离。

(3) 决策变量:

a) 配送点*i*是否为配送中心*M*的第*k*车辆*f(m, k)*的第一个配送点.

$$HS(i, f(m, k)) = \begin{cases} 1 & \text{是} \\ 0 & \text{否} \end{cases} \quad (1)$$

b) 配送点*i*是否为配送中心*M*的第*k*车辆*f(m, k)*的最后一个配送点.

$$HE(i, f(m, k)) = \begin{cases} 1 & \text{是} \\ 0 & \text{否} \end{cases} \quad (2)$$

c) 紧前配送点*i*是和紧后配送点*j*是否由配送中心*M*的第*k*车辆*f(m, k)*完成.

$$HIJ(i, j, f(m, k)) = \begin{cases} 1 & \text{是} \\ 0 & \text{否} \end{cases} \quad (3)$$

(4) 中间变量: 决策变量确定后, 即可计算出中间变量.

$Sst(f(m, k), i)$ 表示*m*中心的车辆*k*到达配送点*i*的实际时间:

$$Sst(f(m, k), i) = Sst(f(m, k), j) + tt(j) + D(i, j) / V \quad (4)$$

$SD(f(m, k))$ 表示*m*中心车辆*k*行使的总路程:

$$SD(f(m, k)) = \sum_{i=1}^N HS(i, f(m, k)) * DHS(i, m) + \sum_{i, j=1}^N HIJ(i, j, f(m, k)) * D(i, j) + \sum_{i=1}^N HE(i, f(m, k)) * DHS(i, m) \quad (5)$$

$U(f(m, k))$ 用于判别*m*中心的车辆*k*是否被调用:

$$U(f(m, k)) = \begin{cases} 1 & SD(f(m, k)) > 0 \\ 0 & SD(f(m, k)) = 0 \end{cases} \quad (6)$$

2.2 模型目标函数的确定

首先分析动用车辆的固定成本, 固定成本主要包括人力成本(驾驶员和装卸工)、车辆管理成本和折旧成本等. VFC 表示完成整个物流订单所有物流中心动用车辆的固定成本:

$$VFC = \sum_{m=1}^M \sum_{k=1}^{mk} U(f(m, k)) * C(f(m, k)) \quad (7)$$

其次分析货物配送的运输成本, 运输成本主要包括车辆的燃油费、路桥费用等, 其中 $SD(f(m, k))$ 表示*m*中心第*k*辆车的行驶距离, $UC(f(m, k))$ 表示该辆车单位行使距离的消耗成本. VTC 表示总的运输成本:

$$VTC = \sum_{m=1}^M \sum_{k=1}^{mk} SD(f(m, k)) * UC(f(m, k)) \quad (8)$$

模型的目标函数为总成本 $SUMC=VFC+VTC$ 最小. 约束条件:

(1) 任一配送点*i*, 只能由一辆车完成配送. 由2.1中的(3)节内容看出, 任一配送点*i*只能由某配送中心*m*第*k*辆车完成, 如为车辆的第一个配送点时, 决策变量 $HS(i, f(m, k))=1$; 如为车辆的最后一个配送点时, 决策变量 $HE(i, f(m, k))=1$; 否则为中间节点, 只能为某一个节点*j*的紧前节点, 因此 $\sum_{i=1}^N HIJ(i, j, f(m, k)) = 1$.

$$\forall i \sum_{m=1}^M \sum_{k=1}^{mk} (HS(i, f(m, k)) + HE(i, f(m, k)) + \sum_{j=1}^N HIJ(i, j, f(m, k))) = 1 \quad (9)$$

(2) 每个客户服务时间必须在约束的时间窗内. $lt(i)$ 和 $ut(i)$ 为配送节点*i*所要求的货物到达的最早时间和最迟时间, $Sst(f(m, k), i)$ 为车辆实际达到节点*i*的时间点.

$$lt(i) \leq Sst(f(m, k), i) \leq ut(i) \quad (10)$$

(3) 保证配送中心的车辆经过一系列的配送任务最终回到原点. 对任一车辆*k*, 如其承担某配送点*i*为出发点时, 必有另一配送点*j*为其最终配送点, 完成最终配送任务后, 返回配送中心.

$$\forall m, \forall k \text{ if } \sum_{i=1}^N HS(i, f(m, k)) = 1 \text{ then } \sum_{j=1}^N HE(j, f(m, k)) = 1 \quad (11)$$

(4) 对于*m*中心任一车辆*k*, 其负责的配送点的需求货物总重量之和不能超出车辆*k*的最大载重量 $g(f(m, k))$.

$$\forall m, \forall k \sum_{i=1}^N HS(i, f(m, k)) * pg(i) + HE(i, f(m, k)) * pg(i) + \sum_{j=1}^N HIJ(i, j, f(m, k)) * pg(i) \leq g(f(m, k)) \quad (12)$$

(5) 假如配送点*i*和配送点*j*由同车配送, 货物类别差级应不大于*L*, 即同车货物类别不冲突, 例如食物和农药、或带刺激性的气味货物不能同车, *L*的取值取决于货物具体的编码方案.

$$|tp(i) - tp(j)| \leq L \quad (13)$$

3 基于聚类分析的海量订单共同配送调度模型求解

在实际的配送过程中, 考虑到车辆的载重限制、行驶距离最小、返回原配送中心等约束条件, 可以断

定配送车辆负责完成的配送点必将分布配送的周围附近,这就提示我们针对众多物流中心、海量订单的配送问题可以通过聚类划分成多个单中心的物流配送问题,降低问题的求解规模,然后启发式优化算法各个求解,最后通过反复迭代聚类分析结合优化算法可求出全局最优解。

3.1 基于单亲遗传的单中心车辆调度问题求解

假设某城市均匀分布着 m 个物流中心,要获取整个城市海量订单优化调度方案(多中心配送),首先解决单中心车辆调度优化问题,下面以单配送中心多配送点车辆调度问题为例,采用单亲遗传算法求解进行说明。

定义1. 虚拟配送点,形式类同于一般配送点,其货物需求量为零,与配送中心及其他配送点的距离为零,仅为染色体基因交叉而定义。

(1) 染色体编码采用整数编码,染色体中基因由配送点编号、配送中心编号和虚拟配送点组成。如图1所示,图中的 s 为配送中心编号, v 为虚拟配送点,相邻两个 s 节点代表一辆车的运输路径。

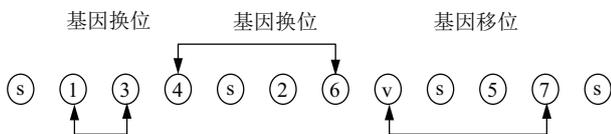


图1 染色体编码示意图

这个编码方案优点是染色体确定后,很容易计算出决策变量 $HS(i, f(m, k))$ 、 $HE(i, f(m, k))$ 、 $HLJ(i, j, f(m, k))$, 并根据公式(4)、(5)、(6)计算出中间变量 $Sst(f(m, k), i)$ 和 $SD(f(m, k))$ 。

(2) 初始染色体的生成,针对类中的配送点,可随机生成若干组染色体。

(3) 计算染色体中各基因的换位概率,虚拟配送点的换位概率为1,其他配送节点换位概率以配送节点3为例说明。

$$p_3 = \frac{D(1,3)+D(3,4)}{DHS(s,1)+D(1,3)+D(3,4)+DHS(s,4)} \quad (14)$$

(4) 针对种群中染色体,随机产生两个换位节点及随机数 $p(0, 1)$, 如果两个节点的换位概率均大于 p , 实施基因换位; 否则重新产生换位节点和随机数 p 。

(5) 根据车辆时间窗约束条件公式(10)和载重约束条件(12)检查每条路径的可行性,剔除不满足约束条件的染色体。

(6) 根据公式(7)和(8)计算配送中心总成本 $SUMC$, 包括车辆固定成本 VFC 和车辆运输成本 VTC , 及 m 中心的总成本 $Z(m)$ 。

(7) 如果连续几次最优总成本 $SUMC$ 不再降低,即获取最优解,算法结束。否则转(8)。

(8) 选择出种群中总成本 $SUMC$ 最少的若干染色体最为新种群,转步骤(3)。

3.2 基于聚类分析的海量订单调度算法

配送中心的运输能力决定了它能够负责周围配送点的多少,因此可根据配送中心的运输能力,选择周围配送点的聚类半径,结合蚁群算法^[15]实施海量订单的优化调度。具体算法如下:

(1) 初始化参数 α, β , 准备 N 组蚂蚁, 每组 m 个蚂蚁随机分布于 m 个配送中心(聚类中心)。

(2) 针对每个聚类中心 i , 按照搜索半径 r 范围, 确定范围内的配送点 V_i , 要求 V_i 的并集为所有需求点。初始化蚂蚁的禁忌表 $Tuba_i = V_i$, 令 $\eta_{ij} = 1/dhs(i, j)$, $dhs(i, j)$ 为配送点 j 到配送中心 i 的距离。

(3) 针对每只蚂蚁 k , 随机选取搜索范围内的配送点 j , 按照概率最大转移规则, 分配配送点 j 于配送中心 i 中, 令 $s(i, j) = 1$, 并将配送点 j 输入禁忌表 $Tuba_i$ 。

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha * [\eta_{ij}]^\beta}{\sum_{j=1}^N [\tau_{ij}(t)]^\alpha * [\eta_{ij}]^\beta}, & j \in allowed_k \\ 0, & j \in allowed_k \end{cases} \quad (15)$$

其中: $\tau_{ij}(t)$ 为 t 时刻, 配送点 j 到配送中心 i 路径的信息素。

(4) 当一组蚂蚁搜索完成后, 所有的配送点分配完毕, 按照3.1节的算法, 计算各配送中心的配送成本 $Z(m)$, 然后计算总成本 $SumC = \sum Z(m)$ 。

(5) 记录蚂蚁 k 经过配送点 j 和配送中心 i 路径的信息增量。

$$\Delta\tau_{i,j}^k = 1/Sum \quad (16)$$

(6) 当 N 组蚂蚁全部完成搜索, 转(7), 否则转(3)。

(7) 更新蚂蚁分组配送点到配送中心路径上的信息量, 并记录最好的分类结果。

$$\tau_{i,j}(t+1) = \rho * \tau_{i,j}(t) + \Delta \quad (17)$$

$$\Delta\tau = \sum_{k=1}^N \tau_{i,j}^k \quad (18)$$

(8) 如连续几次 $SumC$ 不再降低, 算法结束; 否则转(2), 直至总成本不再降低。

4 实验分析

云物流城市共同配送车辆调度问题优化研究过程中,为了节省计算资源,且更加清晰直观的说明所要研究的问题,选取了37个配送点和6个配送中心作为研究对象,针对配送点订单信息的三个属性进行聚类调度分析,包括:横坐标、纵坐标、订单类别(共包含十个等级,不同等级代表不同种类的货物;级别相差越大,种类差别越大,本文假定相差2个级别,不能同车装载),车辆行驶单公里成本1元.假定这些订单和配送中心分布在100*100的区域内,具体订单数据和配送中心信息如表1,表2所示.

针对3.2节中的聚类算法, $\alpha=1.9$ 反映了配送点被聚类到任一配送中心的历史信息强度, α 过小,历史信息重视不够,算法收敛较慢; α 过大,容易陷入局部最优化. $\beta=1.2$ 表示启发式信息(配送点到配送中心的距离DHS)受重视的程度,显然 β 过大,聚类结果受距离DHS影响较大,容易陷入局部最优.共设置50组蚂蚁,每组蚂蚁数37只,经过239次迭代获取调度方案如表3和图2所示.

表1 订单信息表

订单序号	横坐标	纵坐标	类别	需求量(吨)	时间窗下限	时间窗上限
1	8	20	9	1.3	0.4	0.9
2	18	30	5	2.7	0.3	1.1
3	28	25	4	0.3	0.4	1.2
4	37	39	5	3.0	0.5	1.6
5	45	16	4	2.3	0.2	0.6
6	60	21	9	1.3	0.4	1.6
7	73	42	7	1.9	1.2	0.5
8	76	62	4	1.7	0.4	0.7
9	85	32	8	1.0	0.6	0.9
10	92	53	8	3.0	0.8	1.2
11	31	61	3	1.7	0.3	1.2
12	16	62	1	1.0	0.2	0.7
13	49	51	1	2.3	0.1	0.6
14	60	71	4	1.3	0.3	0.5
15	57	62	8	1.3	0.9	1.4
16	80	93	1	1.6	0.6	2.6
17	92	80	2	1.5	0.8	1.2
18	83	61	1	2.7	0.6	0.9
19	85	72	7	1.3	0.2	0.8
20	95	99	8	2.7	0.4	0.8
21	19	50	3	2.7	0.3	0.9
22	11	91	8	1.0	0.3	0.7
23	72	3	4	0.7	0.6	1.1
24	33	2	6	1.3	0.2	1.5
25	74	33	4	2.3	0.2	0.8
26	65	51	6	2.5	0.6	1.6
27	86	47	3	1.3	0.5	1.1
28	63	90	3	0.7	0.5	0.8
29	98	39	8	0.7	0.7	1.5
30	70	99	4	1.0	0.8	1.3
31	53	34	2	1.3	0.1	0.6
32	53	84	1	2.7	0.9	1.4
33	4	33	4	1.0	0.1	0.8
34	78	18	4	0.3	0.1	0.5
35	20	84	5	0.3	0.2	0.6
36	45	63	5	1.3	0.1	0.4
37	63	5	9	1.3	0.3	1.3

表2 配送中心信息表

中心编号	横坐标	纵坐标	车辆数	载重量(吨)
1	75	77	2	10
2	19	72	3	10
3	22	49	4	10
4	82	51	2	10
5	50	78	4	10
6	62	16	3	10

表3 车辆调度最优路径

配送中心	所需车辆	车辆最优路径	配载率	行驶距离(公里)
① (75, 77)	2	①-18-8-17-16-30-①	85%	25.24
		①-28-32-14-13-27-①	83%	34.03
③ (22, 49)	2	③-3-2-1-33-21-③	80%	22.37
		③-12-35-22-11-36-4-③	83%	35.40
④ (82, 51)	2	④-15-26-7-6-9-④	80%	31.20
		④-29-10-19-20-④	77%	33.41
⑥ (62, 16)	1	⑥-31-5-24-37-23-34-25-⑥	95%	37.53

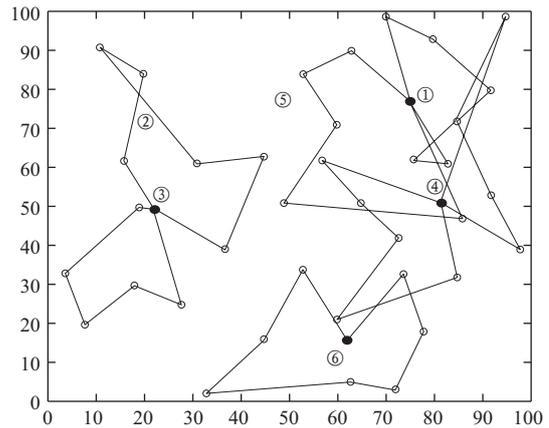


图2 聚类及车辆优化调度路线图

分析表3不难看出,6个配送中心经聚类优化后,选择其中四个参与配送任务.配送中心1,3和4各出2辆车,中心6出1辆车,车辆行驶路径、配载率、行使距离见表3,各车配载率较高,均在75%以上.

如上所示,图2展示了6个配送中心、37个配送点的空间位置、以及7辆车的负责配送的行驶路径.配送中心②和⑤不参与配送任务.经过实心点(配送中心①③④⑥)的封闭曲线为各车辆的行驶路线.配送中心③和⑥负责的配送点不存在运送货物种类冲突问题,车辆路径基本符合路径最短原则,配送中心①和④负责配送点存在物品类别冲突,类别冲突的货物不能同车配送,同时为了满足各配送点时间窗约束,车辆路径

出现交叉,并未满足路径最短原则,这与实际情况相符,并不影响总体配送成本降低。

5 结语

针对云物流环境下城市共同配送海量订单难以调度问题,本文提出基于订单聚类结合单中心多车辆优化调度的迭代算法,对海量订单调度问题进行了探讨研究。单中心多车辆调度模块由各中心服务器完成,物流订单聚类分析由云物流中心系统已完成,实现分布式调度,以便快速求出车辆调度最优解。

参考文献

- 1 贡祥林,杨蓉. “云计算”与“云物流”在物流中的应用. 中国流通经济, 2012, 26(10): 29–33. [doi: 10.3969/j.issn.1007-8266.2012.10.006]
- 2 张水旺,胡小建. 云物流概念模型及其运作机理研究. 科技管理研究, 2015, 35(19): 186–190, 196. [doi: 10.3969/j.issn.1000-7695.2015.19.035]
- 3 田冉,孙林夫,唐慧佳,等. 多车场物流协同运输调度问题研究. 计算机工程与应用, 2015, 51(21): 230–236. [doi: 10.3778/j.issn.1002-8331.1408-0158]
- 4 王天成. 物流配送车辆优化调度问题概述. 物流工程与管理, 2013, 35(8): 29–30.
- 5 杭省策,李怀祖. 多车场车流分配的广义指派模型及其分解算法. 西安交通大学学报, 1997, (12): 111–116.
- 6 郭耀煌,李军. 车辆优化调度问题的研究现状评述. 西南交通大学学报, 1995, 30(4): 376–382.
- 7 张明善,唐小我. 多车场满载货运车辆优化调度的网络流算法. 系统工程学报, 2002, 17(3): 216–220.
- 8 Bianchessi N, Righini G. Heuristic algorithms for the vehicle routing problem with simultaneous pick-up and delivery. Computers & Operations Research, 2007, 34(2): 578–594.
- 9 李作秋,王国林. 一种有时间窗约束的非满载车辆调度问题中的启发式算法研究. 公路交通科技, 2006, 23(7): 147–149.
- 10 Golden BL, Raghavan S, Wasil EA. The Vehicle Routing Problem: Latest Advances and New Challenges. US: Springer, 2008.
- 11 马宇红,姚婷婷,张芳芳. 多车场多车型车辆调度问题及其遗传算法. 数学的实践与认识, 2014, 44(2): 107–114.
- 12 王骏,王士同,邓赵红. 聚类分析研究中的若干问题. 控制与决策, 2012, 27(3): 321–328.
- 13 张建强,方卫国. 有时间窗约束车辆路径问题的改进遗传算法. 计算机工程与应用, 2010, 46(32): 228–231. [doi: 10.3778/j.issn.1002-8331.2010.32.063]