

求解带用户满意度的多目标实时 车辆路径问题的改进伊藤算法

易云飞^{1,2,3}, 蔡永乐¹, 董文永¹, 林晓东²

(1. 武汉大学计算机学院, 湖北武汉 430079; 2. 河池学院计算机与信息工程学院, 广西宜州 546300;
3. 广西高校系统控制与信息处理重点实验室, 广西宜州 546300)

摘 要: 基于对标准车辆路径问题的分析, 本文构建了一种包括交通因素、客户需求动态改变、用户满意度的多目标动态车辆路径问题模型. 针对伊藤算法在求解离散组合优化问题时效率较低、收敛性较差等缺陷, 本文以具有通用性的伊藤算法为框架, 参考蚁群算法, 设计了伊藤-蚂蚁优化算法, 并采用正交实验的方法, 分析了改进算法参数的设置问题. 为了验证改进算法的有效性, 文章对标准测试数据集中的数据进行了测试. 最后, 将标准测试数据改编成符合带用户满意度的多目标实时车辆路径问题模型的测试数据, 并用改进算法进行求解. 实验结果表明, 本文提出的问题模型和改进算法是可行的、有效的.

关键词: 动态车辆路径问题; 伊藤算法; 蚁群算法; 用户满意度

中图分类号: TP14 **文献标识码:** A **文章编号:** 0372-2112 (2015)10-2053-09

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2015.10.026

Improved ITO Algorithm for Multiobjective Real-time Vehicle Routing Problem with Customers' Satisfaction

YI Yun-fei^{1,2,3}, CAI Yong-le¹, DONG Wen-yong¹, LIN Xiao-dong¹

(1. Computer School, Wuhan University, Wuhan, Hubei 430079, China;

2. College of Computer and Information Engineering, Hechi University, Yizhou, Guangxi 546300, China;

3. Guangxi Universities Key Laboratory Breeding Base of System Control and Information Processing, Yizhou, Guangxi 546300, China)

Abstract: Based on the analysis of the standard vehicle routing problem, we proposed a multiobjective real-time vehicle routing problem model, referred as MR-VRPCS. The MR-VRPCS considers the traffic factors, customer demand dynamic change and customers' satisfaction. As we know, the ITO algorithm has low efficiency and poor convergence performance on the discrete combinatorial optimization problems. Therefore, we apply the universal framework of ITO and introduce the Ant Colony Optimization algorithm, which has got depth studied in vehicle routing problem, to design the ITO-Ant Optimization algorithm. We analyze the IAO algorithm's parameter setting problem by the method of orthogonal experiment. Finally, we use the Solomon benchmark test data to prove the effectiveness of the IAO algorithm, and adjust the standard test data to the MR-VRPCS model's, and resolve it with IAO. The experimental results show the feasibility and effectiveness of the proposed model and algorithm.

Key words: dynamic vehicle routing problem; ITO algorithm; ant colony algorithm; customers' satisfaction

1 引言

自从 1959 年 G. Dantzig 和 J. Ramser 首次提出车辆路径问题^[1] (Vehicle Routing Problem, VRP) 以来, 如何以最小的花费来为用户提供最好的服务已成为物流配送领域的一个重要研究方向^[2~6]. 很多专家和学者围绕这

个问题进行了大量的研究. 近年来, 许多研究人员从现实出发, 提出了诸如带时间窗约束的分车运输同时收发车辆路径问题^[2], 带软时间窗车辆路径问题^[6], 供应链管理中的具有模糊时间窗和客户满意度的多目标车辆路径问题^[9]等. 动态性的研究^[2,3]涵盖了客户请求、道路状况、天气等的动态变化. 一些专家和学者针对其中

收稿日期: 2014-08-12; 修回日期: 2015-05-27; 责任编辑: 蓝红杰

基金项目: 国家自然科学基金 (No. 60873114, No. 61170305); 广西自然科学基金 (No. 2013GXNSFBA019282); 广西高等学校科研项目 (No. KY2015YB254); 国家级大学生创新创业训练计划 (No. 201410605055, No. 210510605024/25); 广西混杂计算与集成电路设计分析重点实验室开放基金课题 (No. HCIC201411)

的某一方面做过研究,本文在此基础上,尽可能全面又不失重点的提出了一种带用户满意度的多目标实时车辆调度问题(Multiobjective Real-time Vehicle Routing Problem with Customers' Satisfaction)模型.近些年来,在求解 VRP 问题中最常用的就是元启发式算法,主要有粒子群算法^[4]、蚁群算法^[5]、伊藤算法^[6,7]等.但是,元启发式算法在实践的过程中是一种通用的优化器,普遍存在通用与精确、开采与勘探等方面的矛盾.本文借鉴蚁群算法的路径选择策略,以伊藤算法为框架,设计了一种伊藤-蚂蚁优化算法(ITO-Ant Optimization algorithm, IAO),并用该算法求解带用户满意度的多目标实时车辆调度问题.

2 问题分析及建模

2.1 静态车辆路径问题

车辆路径问题是对物流销售及交通运输系统的数学建模.通常,VRP 包含一个仓库点和 N 个客户点、 K 辆车,每辆车从仓库出发为 N 个客户提供服务,要得到路径长度,运输时间,车辆数等花费最少的调度方案.用无向图 $G = (V, E)$ 表示,其中, $V = \{v_0, v_1, v_2, \dots, v_n\}$ 是客户点集, $E = \{(v_i, v_j) | v_i, v_j \in V, i < j\}$ 是客户之间的边集.要求 K 辆规格相同的车,载货量均为 Q ,从仓库点 v_0 出发,为每一个客户提供服务.每个客户 v_i 具有自己的需求量 q_i .目标是找到完成配送要求的总的行驶路径最短的配送方案.该问题的目标函数定义如式(1).

$$\text{Min} \sum_{i=1}^N \sum_{j=1}^N d_{ij} \cdot \sum_{k=1}^K x_{ij}^k \quad (1)$$

其中, N 为客户数, K 为车辆数, d_{ij} 表示客户 v_i 和 v_j 之间的路径长度, x_{ij}^k 表示车辆 k 经过 v_i, v_j 之间的路径.

另外,还有其他一些限制,比如:车辆的最大载货量、最大行驶距离,客户要求的服务时间窗等.静态车辆路径问题解决的都是客户信息,车辆信息提前已知,确定调度方案后不会发生改变的情形.

2.2 动态车辆路径问题

动态车辆路径问题^[9~11]就是在静态车辆路径问题的基础上,使路径信息、客户信息、车辆信息等车辆在配送过程中可以动态改变的问题.实际应用中,由于天气变化、路况信息改变、客户需求改变等,使得初始时做出的调度决策不符合实际情况,甚至不可行.因此,动态车辆路径问题的研究更具现实意义.

动态变化事件一般包括六种:客户请求的新增、取消和修改,物流中心可用空车数、交通状况和天气的动态变化,这六种动态变化事件又分为两大类,一类为客户信息的动态变化,另一类为网络环境的动态变化.

2.3 带用户满意度的多目标实时车辆调度问题

结合服务业现状,公司不但要节省开销,更重要的

还要提供客户满意的服务,本文借鉴文献^[9]客户满意度的思想,将时间窗映射为更有意义的用户满意度,并将其作为一个优化目标,在此基础上加入客户需求动态调整,路径、天气动态变化的实时因素,使该问题模型与现实问题更契合.

2.3.1 用户满意度

本文将时间窗映射为用户的满意度,并将其作为一个优化目标,使得最终得到既节省开支又让用户满意的调度方案.通常,配送问题中,用户的满意度只有两种情况:满意和不满意,如果你按时配送客户就满意,如果你没按时配送,客户就不满意.于是映射模型如式(2).

$$\text{Customers' Satisfac} = \begin{cases} \text{满意,} & \text{if } S_i \in [B_i, E_i] \\ \text{不满意,} & \text{else} \end{cases} \quad (2)$$

其中, S_i 为车辆为客户 i 提供服务的开始时间, B_i, E_i 分别为客户的时间窗的开始和结束时间.

对于配送问题来说,满意度就是按照客户设定的时间窗为客户提供服务,如果能在客户给定的时间窗内为客户提供服务,则用户对服务满意,否则不满意.本文将满意度设定为满意的客户数与总的客户数之比.

2.3.2 多目标函数

通常车辆路径问题以寻求最短路径为目标,但最短路径存在很多局限性,比如在交通拥堵严重的大城市,最短路径也会成为最拥堵路径,从而传输时间很长.另外还需要更多关注服务质量.所以,本文以费用(车辆租借费用、行驶费用)和用户满意度作为优化目标,以期得到费用低、用户满意度高的结果.目标函数的公式化定义如式(3).

$$\text{min} Z(x) = p_1 \cdot K + p_2 \cdot \sum_{k=1}^K t_k + p_3 \cdot \sum_{i=1}^N (100 - US_i) \quad (3)$$

其中, p_1, p_2, p_3 分别表示车辆租借费用、单位时间行驶费用、用户满意度的服务费用, K 表示车辆数, t_k 表示车辆 k 的行驶时间, US_i 为客户 i 对服务的满意情况.

2.3.3 实时性

在现实世界中,由于天气因素、驾驶员自身因素、车辆因素、道路因素等,使得车辆通过某条道路的运行时间不是一成不变的,为了模拟这种效果本文将根据实时因素对各路段设定行驶速度,以便控制通过该路段的时间长度,从而反映实时变化.例如,对于交通拥堵、天气不好的路段可以设定一个低的行驶速度.

实时性的另外一点体现在客户的动态请求,即客户不是在调度方案构造前都提出请求,而是在车辆服务的过程中也允许客户提出请求,即客户点信息将根据时间的变化而动态变化.为了模拟实验,本文借鉴文献^[11]的设计方法,为每个客户增加 AVAIL. TIME 属

性,即提出请求的时间,只有在时间点大于当前时间的时候,客户才可以被考虑。

3 算法设计

3.1 算法设计思想

伊藤算法设计之初,就综合考虑了群智能算法的所有要素,以期能够给出群智能的统一算法模型.本文融合了蚁群算法的部分求解思想来改进传统的伊藤算法,以此来达到伊藤算法^[7]的通用性与蚁群算法^[11]在VRP问题求解中的精确性的统一。

伊藤算法包含微粒半径、环境温度、漂移算子和波动算子四个算子.首先,每个微粒具有自己的半径来保持自身个性,半径的大小与自身解的优劣有关,以自身个性来保证微粒多样性;其次,环境温度对微粒运动能力进行宏观控制,环境温度随迭代次数增加而降低,以保证算法逐渐趋于收敛;最后,根据微粒半径及环境温度计算微粒的运动能力即漂移和波动强度,然后分别以最优解微粒和随机得到的微粒作为吸引元进行漂移和波动运动,得到新解,并在运动过程中以期求得更优解.本文算法保留伊藤算法的微粒半径和环境温度算子,而漂移和波动算子将根据蚁群算法依信息素浓度完成路径构造的策略做融合改动,具体设计如下。

3.2 算子设计

本文算法包括维持微粒个性的半径算子,进行宏观控制的环境温度算子,及通过反馈学习^[8]不断寻优的漂移和波动算子。

3.2.1 微粒半径设计

本文参考了文献^[7]的基于排序的方法来更新半径,首先按照微粒所代表的解的适应度值进行排序,然后按照式(4)计算每个粒子的半径。

$$r(x_i) = r_{\min} + \frac{(r_{\max} - r_{\min})}{M} * i, i \in [1, N] \quad (4)$$

其中, r_{\max} 和 r_{\min} 为微粒半径的界限,按照上式得到的微粒半径按适应度值均匀分布在 r_{\max} 和 r_{\min} 之间,使得算法在运行的后期,也能保持很强的搜索能力。

3.2.2 环境温度设计

本文参考文献^[7]的温度设计方案,以模拟退火算法温度表的设计来控制环境温度变化,由于环境温度将用作 IAO 算法的宏观控制因素,于是采用线性递减的设计方式,该方式可使温度保持相对独立性,不受其他因素干扰,如式(5)。

$$T(t) = \rho * T(t-1) \quad (5)$$

其中,系数 ρ 控制温度下降速率。

3.2.3 漂移和波动算子设计

漂移算子^[7]体现出微粒的向优性,即群智能算法的开采特性.以当前最优解为吸引元,吸引其他微粒向

该最优解靠拢,以开采该最优解,并期望在开采过程中找到更优解.漂移算子由两部分构成,即漂移强度和漂移过程,根据微粒半径和环境温度计算得到微粒的漂移强度,然后在吸引元的吸引下运动以完成漂移过程。

波动算子^[7]体现出微粒的随机性,即群智能算法的勘探特性.在环境影响下随机产生一个吸引元,吸引其他微粒向随机产生的吸引元靠拢,以达到随机扰动的目的,并期望在扰动过程中保持解的多样性,避免陷入局部最优.波动算子由两部分构成,即波动强度和波动过程,根据微粒半径和环境温度计算得到微粒的波动强度,然后在随机产生的吸引元的吸引下运动以完成波动过程。

常规的群智能算法都采用上述类似的开采和勘探过程寻优,但是这样的过程化方法既不符合开采业的实际情况,也不与支撑伊藤算法的布朗运动过程相符合,所以本文从微粒运动本身出发,采纳了漂移和波动同时进行的思想,即微粒的运动不是机械的步骤性的进行(在最优吸引元吸引下漂移,然后在随机吸引元吸引下波动,分两个过程),而是在合力作用下的连续运动,即在最优吸引元和随机吸引元的共同吸引下的一次运动.这样做既符合微粒运动规律,又能通过减少对适应度值的计算有效减小算法开销。

而蚁群算法信息素的概念使漂移和波动融合进行成为可能,即漂移和波动算子同时作用于信息素浓度,然后在信息素浓度影响下构造调度方案,完成寻优。

在布朗运动中,微粒半径和环境温度决定微粒的运动能力,该运动能力并没有漂移强度和波动强度之分.本文用运动能力统一漂移和波动强度,这样既简化了伊藤算法的设计思想,也与组合优化问题相贴合.根据文^[7]的公式得到本文运动能力计算公式:

$$\delta = \frac{(e^{-\lambda r} - e^{-\lambda r_{\max}})}{(e^{-\lambda r_{\min}} - e^{-\lambda r_{\max}})} * \exp(-1/T) \quad (6)$$

其中, δ 代表运动能力, λ 为控制半径对运动能力影响的参数。

漂移和波动过程融合于信息素浓度.设计思想为:首先,漂移是指吸引元对当前微粒的吸引,于是只增加吸引元上信息素浓度.其次,波动是指受环境影响产生随机扰动,于是只从环境中随机选取若干条路径增加信息素浓度.最后,为维持总体信息素浓度含量,所有路径做信息素挥发。

具体的设计方法为:

(1)所有路径以挥发因子 ρ 进行挥发。

$$\tau(i, j) = (1 - \rho) * \tau(i, j) \quad (7)$$

其中, ρ 为挥发因子, $1 \leq i \leq n, 1 \leq j \leq n$ 。

(2)增加最优解路径上的信息素浓度。

$$\tau(i, j) = \tau(i, j) + \delta, \text{ if } e(i, j) \in \sigma' \quad (8)$$

其中, δ 代表运动能力.

(3) 增加随机选取路径上信息素浓度.

$$\tau(i, j) = \tau(i, j) + \delta, \text{ if } e(i, j) \in \sigma''' \cap \text{rand}() < p \quad (9)$$

其中, $\text{rand}()$ 为随机生成 0 到 1 的函数, p 为选择随机路径的概率, 控制波动强度.

最终信息素浓度更新策略如式(10).

$$\tau(i, j) = (1 - \rho) \cdot \tau(i, j) + \begin{cases} \delta, & \text{if } e(i, j) \in \sigma' \\ \delta, & \text{if } e(i, j) \in \sigma''' \cap \text{rand}() < p \end{cases} \quad (10)$$

其中, σ''' 为该微粒及最优解微粒均未走过的路径.

3.3 解的构造方法

借鉴蚁群算法^[11]中蚂蚁生成路径的方法, 即根据信息素浓度和距离求得各候选边的选取概率, 最后以轮盘赌的方法, 选取一个候选边作为下一步路径, 依此构造调度方案. 概率计算公式如式(11).

$$p^k(i, j) = \begin{cases} \frac{[\tau(i, j)]^\alpha [\eta(i, j)]^\beta}{\sum_{l \notin \text{tabu}_k} [\tau(i, l)]^\alpha [\eta(i, l)]^\beta}, & i \in \text{tabu}_k \cap j \notin \text{tabu}_k \\ 0, & \text{else} \end{cases} \quad (11)$$

其中, $\eta(i, j)$ 为距离的倒数, 即 $1/d_{ij}$, 称为经验知识或能见度, tabu_k 是一个禁忌表, 存储车辆 k 已经搜索过的客户需求点, α 为控制边权重在概率选择中重要程度的因子, 而 β 为控制能见度(即边长因素)在概率选择中重要程度的因子.

4 IAO 参数设置及实验结果分析

本节先将 IAO 算法应用到带时间窗的车辆路径问题求解当中, 以 Solomon 的标准测试数据集为例, 进行正交实验, 对 IAO 的各参数设置进行探讨, 并用上面的参数设置对 Solomon 算例进行求解后与文献[9]和[10]的实验结果进行对比.

4.1 正交实验设定参数

由于本文所提算法的参数较多, 本节将以正交试验法对主要参数进行实验.

4.1.1 正交试验设计

正交试验设计(Orthogonal experimental design)是研究多因素多水平的一种设计方法, 它是根据正交性从全面试验中挑选出部分有代表性的点进行试验, 这些有代表性的点具备了“均匀分散, 齐整可比”的特点.

关于环境温度的参数即初始温度、退火长度、退火率可以根据式(12)相互控制.

$$T_0 \cdot \rho^{(GEN/T_{length})} = T_N \quad (12)$$

其中, T_0 为初始环境温度, T_N 为最终环境温度, ρ 为退火率, GEN 为最大迭代次数, T_{length} 为退火长度.

关于环境温度的参数不在正交试验中做分析, 要分析的参数有: 微粒数 M 、最大迭代次数 GEN 、信息素浓度重要性因子 α 、距离重要性因子 β 、选择随机路径的概率 p , 其中, α 和 β 是一组相对值, 表示信息素浓度即经验与距离即现实所见的相对重要性, 并以 $\alpha:\beta$ 的形式作为一个参数出现.

要设定的四个参数就是正交设计中的 4 个因素, 每个因素分别取 3 个具有代表性的数值, 即 3 个水平, 每个因素的 3 个水平的取值见表 1. 如果采用全面实验的方法, 4 个参数, 每个参数 3 个水平总共有 $3^4 = 81$ 种组合方式, 若在 6 组数据上进行实验, 实验次数将达到 $81 \times 6 = 486$ 次. 本文采用正交试验的方法, 4 个参数每个参数 3 个水平, 只需 9 次实验, 正交试验设计表见表 2, 在六组数据上进行实验, 实验次数有 $9 \times 6 = 54$ 次, 实验结果见表 3.

表 1 水平因素对照表

水平因素	M	GEN	$\alpha:\beta$	p
1	20	100	5:2	0.2
2	40	300	5:3	0.3
3	60	500	5:4	0.4

表 2 正交试验设计表

序号	M	GEN	$\alpha:\beta$	p
1	20	100	5:2	0.2
2	20	300	5:3	0.3
3	20	500	5:4	0.4
4	40	100	5:3	0.4
5	40	300	5:4	0.2
6	40	500	5:2	0.3
7	60	100	5:4	0.3
8	60	300	5:2	0.4
9	60	500	5:3	0.2

4.1.2 正交试验结果分析及参数选取

本节将以带时间窗的车辆路径问题为求解目标, 对上节的正交试验设计表的各种参数组合进行实验, 最后根据实验结果给出推荐的参数组合. 带时间窗的车辆路径问题的目标函数如式(13).

$$\begin{aligned} \min Z = & p_1 \cdot K + p_2 \cdot \sum_{k=1}^K \sum_{i=0}^N \sum_{j=0}^N C_{ij} \cdot x_{ijk} \\ & + p_3 \cdot \sum_{i=1}^N \sum_{k=1}^K \max[(ET_i - s_i), 0] + p_4 \cdot \sum_{i=1}^N ST_i \end{aligned} \quad (13)$$

其中, p_1 、 p_2 、 p_3 、 p_4 分别表示车辆租借费用、单位路径行驶费用、单位时间等待费用、单位时间服务费用,其第一目标为最小化车辆使用数量,第二目标为最小化总行驶距离,第三目标为最小化总等待时间.

Solomon 的标准测试数据集包含 56 个算例来代表不同的问题模型,这些问题分为六大类,即 C1, C2, R1, R2, RC1, RC2. C 意味着聚拢,即 C 类数据中客户点比较密集;R 意味着客户点随机分布,RC 意味着问题介于 C

类问题和 R 类问题之间,部分客户点呈密集分布,部分客户点呈随机分布.另外 1 和 2 分别代表紧密型和宽松型,即时间窗口的宽度.

令车辆租借费用 $p_1 = 500$,单位路径行驶费用 $p_2 = 40$,单位时间等待费用 $p_3 = 10$,单位时间服务成本 $p_4 = 5$,以 9 组参数组合对 6 类问题的第一个实例进行实验,结果由车辆数、路径长度、总费用构成,具体如表 3 所示.

表 3 正交试验结果

序号	C101	C201	R101	R201	RC101	RC201
	11	5	24	7	19	7
1	1102.96	870.91	2121.21	1688.75	2015.32	1766.45
	100 151.06	105 096.79	114 672.31	97 792.99	102 028.43	96 082.43
	10	4	24	8	19	9
2	828.94	767.25	1958.57	1705.85	1981.67	1683.10
	83 157.47	86 702.69	109 580.41	97 863.31	101 513.29	106 101.69
	10	4	26	9	20	9
3	854.26	743.01	2070.68	1761.9	2088.58	1922.58
	84 170.79	90 424.39	116 731.01	105 452.72	106 298.46	113 484.56
	10	4	25	9	20	9
4	854.27	654.03	1995.15	1663.8	2017.67	1799.27
	84 170.79	77 058.85	113 642.80	103 798.65	103 787.64	110 845.41
	11	4	25	7	20	8
5	925.08	646.22	2075.96	1545.86	2015.01	1801.29
	90 036.01	74 973.89	115 161.47	92 594.58	104 058.98	103 704.63
	10	3	22	7	18	8
6	828.94	595.12	1823.46	1659.61	1922.08	1634.21
	83 157.47	70 403.01	101 625.03	93 434.36	97 058.98	100 633.41
	11	4	25	8	19	8
7	893.13	796.94	1996.53	1630.09	2018.45	1660.70
	91 194.73	92 843.40	112 666.36	96 861.82	102 794.54	103 920.49
	10	3	24	9	19	8
8	828.94	594.32	2007.41	1872.66	1954.38	1869.16
	83 157.47	70 381.50	110 522.43	112 856.87	100 407.24	111 726.07
	10	3	25	7	19	8
9	828.94	591.56	1940.40	1541.54	1929.43	1649.09
	83 157.47	70 162.26	111 893.27	93 037.97	99 609.25	101 935.30

表 3 中的黑体部分表示各个算例中的最优解.分析序号 6、9 和 2、7 中参数组合,发现选取随机路径的概率都是 0.2 或 0.3,没有 0.4,说明选取随机路径的概率的大小对 IAO 的性能有影响,建议取值为 0.2 ~ 0.3;序号 6、9 的迭代次数都是 500,而序号 2、7 的迭代次数为 100 和 300,说明迭代次数对 IAO 的性能有影响,建议迭代次数取值为 300 以上;除序号 2 的种群大小的取值为 20 外,其他序号的种群大小取值都是 40 或 60,表明种群大小对 IAO 的性能也有一定影响,建议取值为 40 ~ 60;信息素浓度与距离重要性比值 5:2、5:3 和 5:4 都有,说明历史经验与自身知识的相对重要性对 IAO 的性能影响不大,建议取值为 5:2 到 5:4 之间.

4.2 实验结果对比分析

实验仿真环境:Eclipse Helios Service Release 2.费用计算按 Solomon 的设置方法:每辆车的租借费用 500,单位时间等待费用 10,每辆车单位路径行驶成本 40,单位服务费用 5.算法相关参数设置如下:种群大小 $M = 50$,最大进化代数 $GEN = 500$,退火表初始温度 $T = 1000$,退火表长度 $T_{Length} = 2$,退火速率 $\rho = 0.99$,边权重重要因子 $\alpha = 5$,客户点之间距离重要性因子 $\beta = 3$,选取随机路径的概率 $p = 0.3$.实验结果如表 4 表 5 所示.

带时间窗的车辆路径问题不仅考量总的路径长度,还考量车辆数、客户等待时间、总的花费等.文[9]算法 GA 表示遗传算法,文[10]算法 PTS 表示并行禁忌

搜索算法.表 4 和表 5 中黑体表示最优结果,问题类型 R1、C1、RC1、R2、C2、RC2 对照表 4 和表 5 的实验结果如图 1~图 6.可以看出:图 4 和图 6 显示出本文算法在求解 R2 和 RC2 类问题上优于文[9]和文[10],而文[9]略优于文[10];由图 1、2、3、5 可以看出,在 R1、C1、RC1、C2 类问题求解中,本文算法优于文[10]但略差于文[9].由

于 R 代表客户点的位置随机分布,C 代表集簇分布,1 代表客户点的时间窗口紧密,2 代表客户点的时间窗口宽松,故本文算法适合求解时间窗口宽松且客户位置随机或者既随机又集簇的数据类型.此外,本文算法的运行时间全部在 50s 内,而文[9]中的运行时间在 2~7h 之间.

表 4 本文算法与文[9]、文[10]在数据集 R1,C1,RC1 上关于路径长度的对比

Instance	文[9]算法 GA	文[10]算法 PTS	本文算法 IAO	Instance	文[9]算法 GA	文[10]算法 PTS	本文算法 (IAO)	Instance	文[9]算法 GA	文[10]算法 PTS	本文算法 (IAO)
R101	1677.0	1862.84	1728.36	C101	828.94	905.23	828.94	RC101	1690.6	1874.12	1834.44
R102	1511.8	1540.33	1537.14	C102	828.94	976.10	977.69	RC102	1493.2	1686.70	1727.64
R103	1287.0	1325.31	1296.33	C103	828.06	884.98	1008.69	RC103	1331.8	1528.13	1429.70
R104	974.24	1256.27	1105.91	C104	824.78	899.03	962.77	RC104	1177.2	1322.98	1265.76
R105	1424.6	1518.12	1522.69	C105	828.94	880.51	828.94	RC105	1611.5	1753.57	1578.72
R106	1270.3	1622.13	1397.69	C106	828.94	983.27	836.69	RC106	1437.6	1637.27	1575.96
R107	1108.8	1255.54	1232.49	C107	828.94	1021.97	842.94	RC107	1222.1	1428.03	1355.92
R108	971.91	1085.22	1101.14	C108	828.94	974.99	863.81	RC108	1156.5	1307.80	1240.99
R109	1212.3	1296.14	1321.08	C109	828.94	933.25	891.74				
R110	1156.5	1286.40	1191.92								
R111	1111.9	1188.86	1217.37								
R112	1036.9	1148.08	1088.64								

表 5 本文算法与文[9]、文[10]在数据集 R2,C2,RC2 上关于路径长度的对比

Instance	文[9]算法 GA	文[10]算法 PTS	本文算法 IAO	Instance	文[9]算法 GA	文[10]算法 PTS	本文算法 (IAO)	Instance	文[9]算法 GA	文[10]算法 PTS	本文算法 (IAO)
R201	1298.8	1416.42	1137.53	C201	591.56	811.13	697.14	RC201	1206.1	1482.59	1344.57
R202	1091.22	1186.73	1010.06	C202	591.56	807.31	786.48	RC202	1201.8	1255.89	1160.33
R203	1041.0	1143.19	999.24	C203	591.56	802.43	715.62	RC203	1060.0	1198.44	1074.20
R204	1130.6	834.36	832.33	C204	599.96	812.20	667.17	RC204	901.46	908.04	866.47
R205	1087.8	1145.28	1069.81	C205	588.88	742.86	677.07	RC205	1410.3	1297.59	1170.64
R206	940.12	1078.05	936.19	C206	588.88	707.15	644.72	RC206	1194.8	1300.95	1118.96
R207	904.90	1044.99	890.53	C207	591.56	740.78	615.03	RC207	1040.6	1145.48	1024.68
R208	774.18	823.27	791.41	C208	588.32	760.37	619.79	RC208	898.50	975.21	875.54
R209	875.6	1120.02	895.93								
R210	938.58	1038.62	877.40								
R211	801.03	1071.13	842.44								

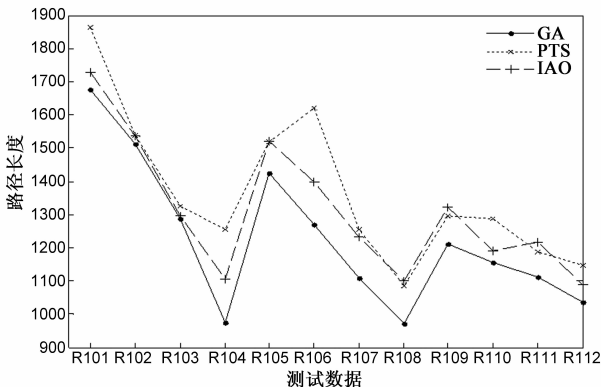


图1 数据R1上的对比

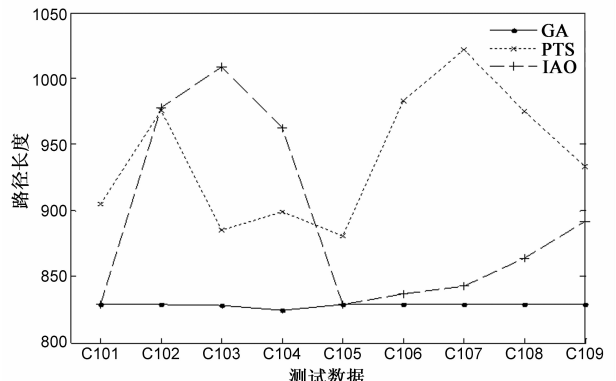


图2 数据C1上的对比

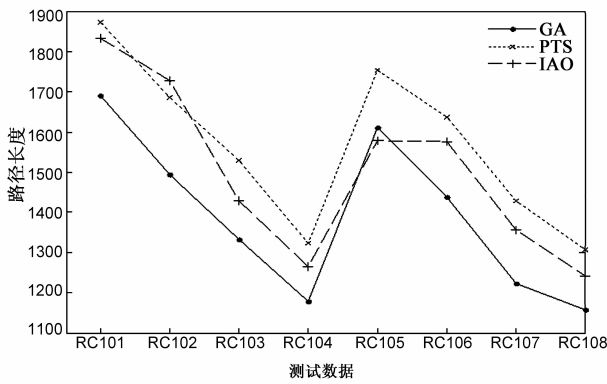


图3 数据RC1上的对比

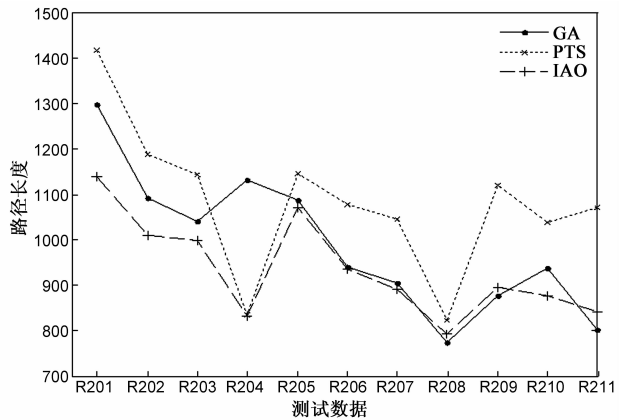


图4 数据R2上的对比

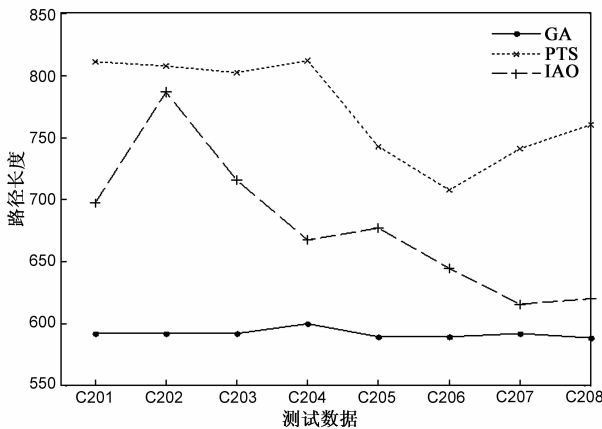


图5 数据C2上的对比

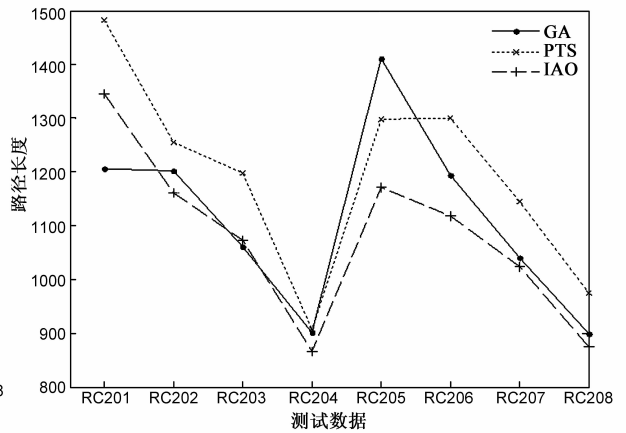


图6 数据RC2上的对比

5 求解具有用户满意度的多目标实时车辆路径问题

为了进行实验验证,需将 Solomon 的标准测试数据进行改编,以使其满足本文提出的具有用户满意度的多目标实时车辆路径问题模型.首先,根据文献[11]提出的方法,对每个客户点增加 AVAIL. TIME 属性,表示该客户提出请求的时间,该时间随机设定在 $[0, \overline{ET}_i]$ 之间 ($\overline{ET}_i = \min\{ET_i, t_{i-1}\}$, ET_i 为客户 i 设定的服务开始时间, t_{i-1} 为当前最优解中车辆从客户 i 的前面客户出发的时间),以此来达到模拟现实中客户随机提出请求的状况.另外,在 Solomon 测试数据中车辆行驶速度默认为 1,为了模拟天气、交通拥堵、驾驶员自身因素以及车辆状况对行驶速度的影响,将对每条路设置一个速度变量 $speed$, $speed = 1$ 表示这条路一切正常, $speed = 0.8$ 表示这条路天气不好或者交通拥堵或者驾驶员无心开车.

文中以 $Dyn = x\%$ 来表示数据的动态性,即以 $x\%$ 的概率将客户的 AVAIL. TIME 设定为非 0,以 $x\%$ 的概率将一条路径的行驶速度设定为 0.8.

本文对每个算例运行 10 次求平均值,再对六类问

题分别求该类问题在不同动态性下的平均值,实验结果包括平均车辆数、平均行驶时间、平均总费用和平均用户满意度,结果如图 7 ~ 图 10 所示.在图 7 中六条线条趋于水平,说明所需车辆数受动态性影响不大;在图 8、图 9 中六条线基本处于线性递增的趋势,说明随着动态性的增加,车辆的行驶时间不断增大,总费用也不断增大;在图 10 中各条线也趋于平缓,说明在动态性增加的时候也能够找到让用户满意的调度方案.

通过对上面四个图的分析,很容易看出随着动态性的增加,车辆数变化不大,而行驶时间、总花费都有所增加,但是增加值在可接受范围之内,而且最重要的是用户满意度总体保持高水平,即在动态性影响下得到了总费用增加适当、总满意度保持高水平的解,足以说明本文算法能有效求解提出的带满意度的多目标实时车辆路径问题.显然,所提出的改进伊藤算法在求解带满意度的多目标实时车辆路径问题时是可行的.

6 结论

本文在标准车辆路径问题进行分析之后提出了具有用户满意度的多目标实时车辆路径问题并建模,对算法中的各算子进行分析、改进和对比设计,给出了更

加符合布朗运动和蚂蚁寻优规律的算子设计.之后,采用正交试验法对参数设定进行实验,给出了较优的参数设定方案,并进行实验,结果表明改进算法能有效求解带时间窗的车辆路径问题.最后,对 Solomon 标准测试算例进行改编,为每个客户增加一个 AVAIL. TIME,为

每个路径增加一个 speed,使其满足本文提出的具有用户满意度的多目标实时车辆路径问题模型,并用伊藤蚂蚁群优化算法进行求解,实验结果验证了问题模型及算法的可行性.

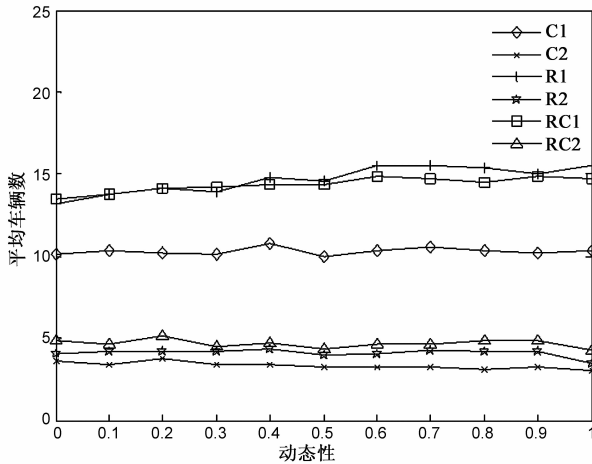


图7 车辆数随动态性变化

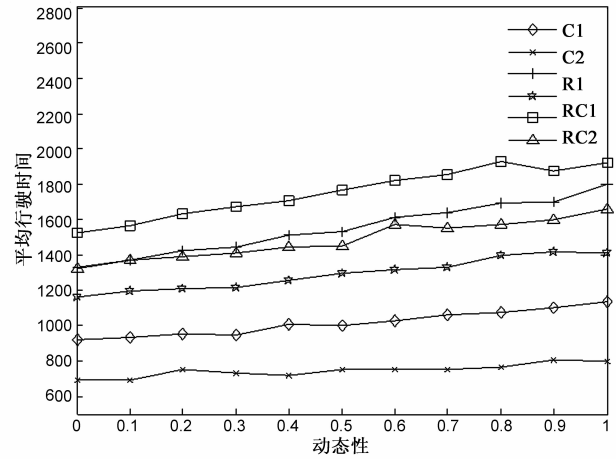


图8 行驶时间随动态性变化

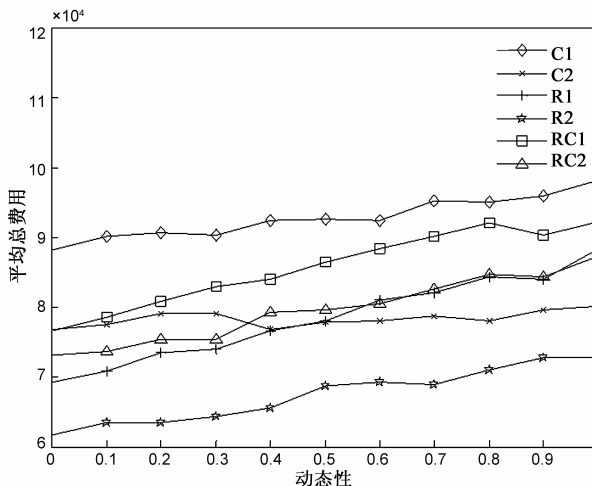


图9 总费用随动态性变化

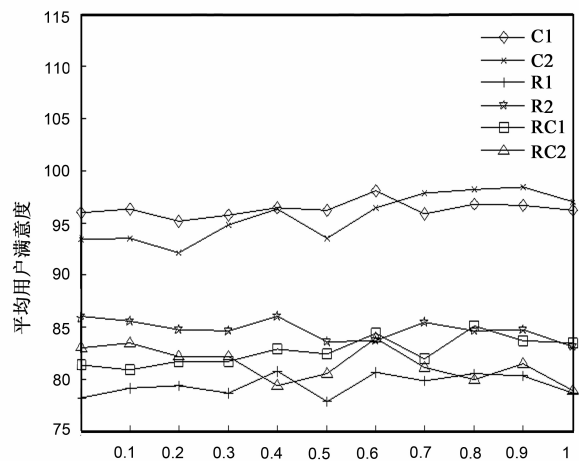


图10 用户满意度随动态性变化

本文对伊藤-蚂蚁优化算法的设计是从总体设计思路出发,而初始化方法、局部搜索策略等未做详细研究,是进一步研究的一个方向.对本文提出的具有用户满意度的多目标实时车辆路径问题模型,仅进行了对 Solomon 测试算例改编的数据的测试,下一步将结合现实中的具体问题完善该模型.

参考文献

[1] Dantzig G, Ramser J. The truck dispatching problem[J]. Management Science, 1959, 6(1): 80 - 91.
 [2] Tang Guochun, Aibing Ning, et al. A practical split vehicle routing problem with simultaneous pickup and delivery[A]. 16th International Conference on Industrial Engineering and

Engineering Management[C]. Beijing: IEEE, 2009. 26 - 30.

[3] A LKok, E W Hans, J M J Schutten. Vehicle routing under time-dependent travel times; the impact of congestion avoidance[J]. Computer & Operations Research, 2012, 39(5): 910 - 918.
 [4] S Geetha, G Poonthilir, et al. A hybrid particle swarm optimization with genetic operators for vehicle routing problem[J]. Journal of Advances in Information Technology, 2010, 1(4): 181 - 188.
 [5] Michalis Mavrovouniotis, Shengxiang Yang. Ant colony optimization with memory-based immigrants for the dynamic vehicle routing problem[A]. 2012 IEEE Congress on Evolutionary Computation[C]. Brisbane: IEEE, 2012. 1 - 8.
 [6] 易云飞, 董文永, 等. 求解带软时间窗车辆路径问题的改

进伊藤算法及其收敛性分析[J]. 电子学报, 2015, 43(4): 658 - 664.

Yi Yunfei, Dong Wenyong, et al. The improved ITO algorithm to solve the vehicle routing problem with soft time windows and its convergence analysis[J]. Acta Electronica Sinica, 2015, 43(4): 658 - 664. (in Chinese)

- [7] 董文永, 张文生, 于瑞国. 求解组合优化问题伊藤算法的收敛性和期望收敛速度分析[J]. 计算机学报, 2011, 34(4): 636 - 646.

Dong Wenyong, Zhang Wensheng, et al. Convergence and run-time analysis of ITO algorithm for one class of combinatorial optimization[J]. Chinese Journal of Computers. 2011, 34(4): 636 - 646. (in Chinese)

- [8] 喻飞, 李元香, 等. 透镜成像反学习策略在粒子群算法中的应用[J]. 电子学报, 2014, 42(2): 230 - 235.

Yu fei, Li Yuanxiang, et al. The application of a novel OBL based on lens imaging principle in PSO[J]. Acta Electronica Sinica, 2014, 42(2): 230 - 235. (in Chinese)

- [9] S FGhannadpour, S Noori, R Tavakkoli Moghaddam. Multiobjective dynamic vehicle routing problem with fuzzy travel times and customers' satisfaction in supply chain management[J]. IEEE Trans Eng Manage, 2013, 60(4): 777 - 790.

- [10] Quan XiongWen, Xu Ya. Dynamic pick-up and delivery vehicle routing problem with ready-time and deadline[A]. 32nd Chinese Control Conference[C]. Xi'an: IEEE, 2013. 2515 - 2520.

- [11] Barry van Veen, Michael Emmerich, et al. Ant colony algorithms for the dynamic vehicle routing problem with time windows[A]. 5th International Work Conference on the Interplay

Between Natural and Artificial Computation (IWINAC 2013) [C]. Mallorca: Springer Berlin Heidelberg, 2013. 1 - 10.

作者简介



易云飞 男, 1981 年出生于广西资源, 武汉大学计算机学院博士生, 副教授, 研究方向为智能计算, 机器学习等.

E-mail: gxyiyf@163.com



蔡永乐 男, 1991 年出生于河南平顶山, 武汉大学计算机学院硕士研究生, 研究方向为智能计算, 机器学习.



董文永(通信作者) 男, 1973 年生河南南阳, 武汉大学教授, 博士生导师, 长期从事系统科学、仿真与控制、演化计算、机器学习、数据挖掘等方面的研究工作.

E-mail: dwy@whu.edu.cn