

具有模糊预约时间的 VRP 混合遗传算法

张建勇¹, 李 军², 郭耀煌³

(1. 南开大学国际商学院, 天津 300071; 2. 天津职业大学经济管理学院, 天津 300402;
3. 西南交通大学经济管理学院, 成都 610031)

摘要: 在对具有模糊预约时间的多对多货物收发情况下的车辆路径问题进行简单描述的基础上, 构建了该问题的多目标数学规划模型, 提出了解决该问题的一种基于插入启发式算法、并用修正的推—碰—掷过程进行改进的混合遗传算法, 最后, 给出了该问题的一个计算实例, 并与改进的 Solomon 插入启发式算法进行了比较.

关键词: 模糊预约时间; 车辆路径问题; 混合遗传算法

中图分类号: U116.2 **文献标识码:** A **文章编号:** 1007 - 9807(2005)03 - 0064 - 08

0 引言

车辆路径问题 (vehicle routing problem, VRP) 是指为服务于已知的一组顾客的一个车队, 设计一组开始和结束于一个中心出发点的最低费用路径. 每个顾客只能被服务一次, 而且, 一个车辆服务的顾客数不能超过它的能力^[1].

带有时间窗的车辆路径问题 (vehicle routing problem with time windows, VRPTW) 是车辆路径问题的一类重要拓展. 在 VRPTW 中, 对每项任务的服务都必须在一定的时间范围内开始, 如车辆到达时间早于该任务可以开始的最早时刻, 则车辆必须在该任务处等待; 车辆到达时间晚于任务可以开始的最晚时间, 则任务必须被推迟进行或被拒绝.

在过去的几十年中, 各国学者对带有时间窗的车辆路径优化问题进行了大量研究, 提出了解决该问题的大量启发式、亚启发式算法^[2~10]. 但是, 在许多实际的应用中, 时间窗并不能真正反映顾客的时间偏好. 顾客可能更偏好于服务在时间窗内的某一时点 (段) 开始, 而服务开始早于或晚于该时点 (段), 顾客的满意度都会降低. 在这种情

况下, Cheng 和 Gen^[11] 提出用模糊预约时间 (fuzzy due time) 的概念来代替时间窗的概念, 并研究了单收或单发的情况下的具有模糊预约时间的车辆路径优化问题. 本文在对模糊预约时间的概念进行简单介绍的基础上, 进一步研究具有模糊预约时间的多对多货物收发情况下的车辆路径问题, 并提出解决该问题的一种混合遗传算法.

1 问题描述

1.1 模糊预约时间

模糊预约时间是一种反映顾客偏好的时间窗, 它可以表示为关于时间 t 的凸模糊数, 顾客的满意度函数由该模糊数的隶属度函数 $\mu_i(t_i)$ 表示, 如图 1 所示. 模糊预约时间应包含可容忍的时间范围 $[ET_i, LT_i]$ 和期望服务时间 DT_i 两部分. 当顾客在期望服务时间 DT_i 被服务时, 顾客满意度最大 (满意度 μ 设为 1); 否则, 顾客满意度随着服务时间与期望时间之间的差距的增大而降低. 若服务时间落在可容忍时间范围之外 ($t_i < ET_i$ 或 $t_i > LT_i$), 则顾客不满意 ($\mu = 0$).

收稿日期: 2003 - 05 - 26; 修订日期: 2005 - 03 - 24.

基金项目: 国家自然科学基金资助项目 (70071028); 南开大学科研启动经费资助项目.

作者简介: 张建勇 (1975 -), 男, 山西和顺人, 博士.

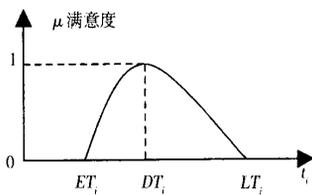


图 1 模糊预约时间
Fig. 1 Fuzzy due time

1.2 具有模糊预约时间的多对多货物收发情况下的 VRP 描述

具有模糊预约时间的多对多货物收发情况下的 VRP 一般可描述为：设有 m 项货物运输任务，分别用编号 $1, 2, \dots, m$ 表示，任务 i ($i = 1, 2, \dots, m$) 要求从始点(发点) u_i 运送数量 g_i 的货物到终点(收点) v_i ，始点和终点的模糊预约时间分别为 S_i 和 D_i ，任务由车场(用编号 0 表示)发出的车辆来完成。已知车辆的运输能力为 q ，且 $q \geq g_i$ 。求满足一定目标的车辆行驶路径计划。

1.3 模型建立

1) 基本思路

为解决问题方便起见，本文将每一任务 i 的发点和收点作为两个顾客来处理，其编号分别为 i 和 $m + i$ ，两顾客的需求量分别为 g_i 和 $-g_i$ ，每一顾客的模糊预约时间分别对应原任务的 S_i 和 D_i 。从而将含 m 项运输任务的、具有模糊预约时间的多对多货物收发情况下的 VRP 问题转化为含有 $2m$ 个顾客的、满足一定条件的具有模糊预约时间的单收或单发情况下的 VRP 问题来求解，这些条件包括：

顾客 i 和 $m + i$ 必须由一辆车在一次服务过程中服务；

对顾客 $m + i$ 的服务必须在顾客 i 之后进行。

2) 符号说明

为构造数学模型，定义常量和变量如下：

常量

- m —— 运输任务数；
- $2m$ —— 转换为单收或单发情况下的顾客数；
- n —— 车辆数；
- q —— 车辆运载能力；
- g_i —— 顾客 i 的需求量；
- d_{ij} —— 由顾客 i 直接行驶到顾客 j 的行驶

距离；

$w_i(t_i)$ —— 当车辆在顾客 i 处的开始服务时间为 t_i 时，车辆在该顾客处的等待时间；

$\mu_i(t_i)$ —— 当车辆在顾客 i 处的开始服务时间为 t_i 时，顾客 i 的满意度。

变量

t_i —— 顾客 i 的开始服务时间；

$x_{ik} = \begin{cases} 1 & \text{点 } i \text{ 的由车辆 } k \text{ 服务} \\ 0 & \text{否则} \end{cases}$

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

3) 目标

在综合考虑企业运输成本最小化和顾客满意度最大化两方面因素的基础上，具有模糊预约时间的车辆路径问题的优化目标定为以下 4 个方面：

$$\text{最小化车辆使用数 } \min \sum_{j=1}^{2m} \sum_{k=1}^n x_{0jk}$$

$$\text{最小化总行驶距离 } \min \sum_{k=1}^n \sum_{i=1}^{2m} \sum_{j=1}^{2m} d_{ij} x_{ijk}$$

$$\text{最小化平均等待时间 } \min \frac{1}{2m} \sum_{i=1}^{2m} w_i(t_i)$$

设某一车辆 k 由顾客 i 直接行驶到顾客 j ，车辆在 i 处的开始服务时间、服务时间以及车辆从 i 到 j 的行驶时间如图 2 所示，则车辆 k 在顾客 j 处的等待时间 w_j 可按下式确定

$$w_j = t_j - (t_i + T_i + t_{ij})$$

其中： w_j 表示车辆在顾客 j 处的等待时间； t_j 表示车辆在顾客 j 处的开始服务时间 ($[ET_j, LT_j]$ 范围内的某一时间点)； T_i 表示车辆在顾客 i 处的服务时间； t_{ij} 表示车辆 k 由顾客 i 处直接行驶到顾客 j 处所用时间。

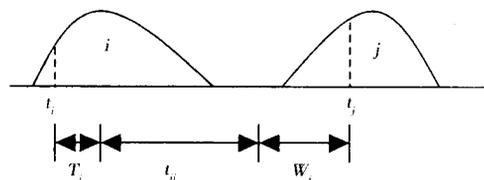


图 2 等待时间的确定

Fig. 2 Waiting time

最大化顾客平均满意度

$$\max \frac{1}{2m} \sum_{i=1}^{2m} \mu_i(t_i)$$

4) 约束条件

综合满意度约束、能力约束以及取值约束等, 可给出如下约束条件:

约束 1 $\mu_i(t_i) \geq \mu_i$; $i = 1, 2, \dots, 2m$. 保证顾客的满意度不至于过低而失去顾客, 其中 μ_i 由决策者给出.

约束 2 $\sum_k x_{ik} = 1$; $i = 1, 2, \dots, 2m$. 此式保证每个顾客只能由一辆车服务.

约束 3 $x_{ik} = 0$ 或 1 ; $i = 0, 1, \dots, 2m \quad \forall k$. 该式为 x_{ik} 的取值约束.

约束 4 $x_{ik} = x_{m+i, k}$; $i = 1, 2, \dots, m \quad \forall k$. 保证顾客 i 和 $m+i$ 由同一车辆服务.

约束 5 $t_i < t_{m+i}$; $i = 1, 2, \dots, m$. 对顾客 $m+i$ 的服务必须在 i 之后进行.

约束 6 $x_{ijk} = x_{jki}$ $j = 0, 1, \dots, 2m \quad \forall k$;

$x_{ijk} = x_{ikj}$ $i = 0, 1, \dots, 2m \quad \forall k$. 此两式为变量 x_{ijk} 与 x_{ikj} 的关系约束.

约束 7 $\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq |S| - 1 \quad S \subseteq \{1, 2, \dots, 2m\} \quad \forall k$. 此式为支路消去约束.

约束 8 $x_{ijk} = 0$ 或 1 ; $i, j = 0, 1, \dots, 2m \quad \forall k$. 变量 x_{ijk} 的取值约束.

约束 9 $x_{ijk} = 1 \Rightarrow t_i + T_i + t_{ij} \leq t_j$. 一条线路上两邻接任务存在的条件.

约束 10 能力约束. 对于任一既定的车辆路径安排, 设分配给该车辆的顾客 (包括发点顾客和收点顾客) 按服务顺序排列为 x_1, x_2, \dots, x_h , 则能力约束可表示为

$$\sum_{i=1}^k g_{x_i} \leq q; \quad k = 1, 2, \dots, h$$

该式保证任意时刻车辆的运载量都不超过其最大运载能力.

2 混合遗传算法

本文采用自然数编码, 构造如下的求解具有模糊预约时间的多对多货物收发情况下的 VRP 的混合遗传算法.

2.1 构造染色体, 产生初始可行种群

初始可行种群可按如下步骤产生:

1) 生成 m 项任务的随机排列;

2) 对该任务排列对应车辆进行聚类 (即将任务安排给车辆), 产生一可行染色体;

3) 对产生的染色体进行改进, 确定每一顾客的最佳服务时间;

4) 重复以上过程, 直至产生给定的 $pop. size$ 个可行染色体.

其中, 染色体按如图 3 所示方式表示, 即将顾客/服务时间/车辆列表用于染色体表达, 它表达了与服务时间和指定车辆有关的顾客排列, 其基因是一个顺序的三元组 (顾客, 服务时间, 车辆).

顾客 x_1	顾客 x_2	...	顾客 x_m
服务时间 t_{s1}	服务时间 t_{s2}	...	服务时间 t_{sm}
车辆 v_{s1}	车辆 v_{s2}	...	车辆 v_{sm}

图 3 染色体表达方式

Fig. 3 Representation of chromosomes

2.1.1 任务排列对应于车辆的聚类方法

1) 聚类步骤

任务排列对应车辆的聚类可按如下步骤进行:

步骤 1 新开一辆车, 将任务排列最左侧的任务所对应发点顾客和收点顾客分配给该车辆 (发点顾客位于收点顾客之前), 并计算各自的最早开始服务时间. 最后, 从任务排列中删除该任务.

步骤 2 找出当前任务排列最左侧任务所对应的发点顾客在当前车辆的可行插入位置, 若无可行插入位置, 转步骤 1; 否则, 找出所有可行插入位置, 转步骤 3.

步骤 3 将该发点顾客插入到目前可行插入位置中的最佳插入位置, 并在该插入点后寻找目前任务所对应的收点顾客的可行插入位置, 若有可行插入位置, 则将其插入到最佳可行插入位置, 从任务排列中删除该任务, 转步骤 4; 否则, 转步骤 5.

步骤 4 检查是否所有任务都安排完毕, 若是, 生成一可行染色体, 当前任务排列对应于车辆的聚类结束; 否则, 转步骤 2.

步骤 5 从当前发点顾客插入点删除该插入顾客, 并从其可行插入位置集合中删除该位置. 检查可行插入位置集合是否为空, 若是, 转步骤 1;

否则, 转步骤 3.

其中, 可行插入位置与最佳插入位置的确定方法如下:

2) 可行插入位置的确定

时间可行位置的确定

为保证任一顾客 i 的满意度都不小于 μ_i , 令 $ET_i = \min\{t / \mu_i(t) = \mu_i\}, LT_i = \max\{t / \mu_i(t) = \mu_i\}$, 即截去可容忍时间范围内顾客满意度小于 μ_i 的时间段.

对于已部分建立的可行车辆路径 i_1, i_2, \dots, i_p , 若在其第 k 个位置之后 (i_k 和 i_{k+1} 之间) 插入一顾客 h , 则插入 h 后, 车辆到达 h 的时间将确定, 同时原车辆路径中的第 $k+1$ 到第 p 个顾客的开始服务时间也将部分或全部发生改变.

令 $Y_L h$ 表示插入后车辆到达 h 的时间, 则

$$Y_L h = \sum_{s=1}^{k-1} t_{i_s, i_{s+1}} + \sum_{s=1}^k T_{i_s} + \sum_{s=1}^k w_{i_s} + t_{i_k, h} \quad (1)$$

其中: t_{ij} 表示车辆由 i 直接行驶到 j 所用时间; T_i 表示车辆在 i 处的服务时间; w_i 表示车辆在 i 处的等待时间.

假设旅行时间满足三角不等式, 且令 $td_{i_{k+1}}$ 表示设插入 h 后, 车辆到达 i_{k+1} 的时间比原线路上到达时间的延迟量, 则

$$td_{i_{k+1}} = t_{i_k} + T_{i_k} + t_{i_k, h} + w_h + T_h + t_{h, i_{k+1}} - (t_{i_k} + T_{i_k} + t_{i_k, i_{k+1}}) = t_{i_k, h} + w_h + T_h + t_{h, i_{k+1}} - t_{i_k, i_{k+1}} \quad (2)$$

由于原线路上车辆等待时间对插入 h 后产生的延迟时间的“吸收”, 车辆在 i_s 处由于 h 的插入而造成的到达时间延迟量为

$$td_{i_s} = \begin{cases} td_{i_{k+1}} - \sum_{j=k+1}^{s-1} w_{i_j} & \text{如果 } td_{i_{k+1}} - \sum_{j=k+1}^{s-1} w_{i_j} > 0 \\ 0 & \text{否则} \end{cases} \quad (3)$$

$s = k + 2, \dots, p$

设原线路上车辆到达 i_s 的时间为 $Y_L i_s$, 则使该点不违反模糊预约时间约束 ($t_{i_s} \leq LT_{i_s}$) 的达到时间最大允许推迟量为 $LT_{i_s} - Y_L i_s$.

若某插入位置为一时间可行插入位置, 则插入顾客以及插入点之后顾客的开始服务时间一定不能

违反模糊预约时间的约束, 因此可得以下定理:

定理 1 在已部分建立的可行车辆路径 i_1, i_2, \dots, i_p 上, 在其第 k 个位置之后 (i_k 和 i_{k+1} 之间) 插入一顾客 h 时, 时间可行的充分必要条件是

$$Y_L h \leq LT_h, \text{ 且 } td_{i_s} \leq LT_{i_s} - Y_L i_s, \quad s = k + 1, \dots, p.$$

运载能力可行位置的确定

类似于时间可行位置的确定, 为保证车辆在任何时刻的运载量都不超过其最大运载能力, 可得如下运载能力可行性插入定理:

定理 2 在已部分建立的可行车辆路径 i_1, i_2, \dots, i_p 上, 在其第 k 个位置之后 (i_k 和 i_{k+1} 之间) 插入一顾客 h 时, 运载能力可行的充分必要条件是

$$\sum_{s=1}^k g_{i_s} + g_h \leq q, \text{ 且 } \sum_{s=1}^k g_{i_s} + g_h + \sum_{r=k+1}^z g_{i_r} \leq q, \quad z = k + 1, k + 2, \dots, p.$$

可行插入位置的确定

综合时间可行和运载能力可行两方面的内容, 可得:

定理 3 在已部分建立的可行车辆路径 i_1, i_2, \dots, i_p 上, 在其第 k 个位置之后 (i_k 和 i_{k+1} 之间) 插入一顾客 h 时, 可行的充分必要条件是

$$Y_L h \leq LT_h, \sum_{s=1}^k g_{i_s} + g_h \leq q, \text{ 且 } td_{i_z} \leq LT_{i_z} - Y_L i_z, \sum_{s=1}^k g_{i_s} + g_h + \sum_{r=k+1}^z g_{i_r} \leq q, \quad z = k + 1, \dots, p.$$

3) 最佳插入位置的确定

所谓最佳插入位置是指在所有可行插入位置中, 插入后可使所有顾客的综合满意度最大的插入位置. 最佳插入位置可按如下步骤确定:

步骤 1 找出待插入顾客在当前路径中的所有可行插入位置集合, 并设顾客综合满意度为 0;

步骤 2 计算将顾客插入当前可行插入位置集合中最左侧位置后所有顾客的综合满意度, 若计算出的综合满意度大于原综合满意度, 则设顾客综合满意度为当前计算出的满意度, 并设当前插入位置为最佳插入位置, 并从可行插入位置集合中删除该位置, 转步骤 3;

步骤 3 检查可行插入位置集合, 若集合非空, 则转步骤 2, 否则, 目前计算出的最佳插入位置即为该顾客在当前路径中的最佳插入位置, 过程结束.

2.1.2 可行车辆路径安排的最佳服务时间确定

通过 2.1.1 可得一可行染色体,在这一可行染色体中,每一顾客的开始服务时间都是其最早可开始服务时间.有必要在一定范围内对其服务时间进行调整,找出不改变染色体可行性条件下的、使顾客综合满意度最大化的顾客最佳服务时间.本文在对 Cheng 和 Gen^[11] 提出的推—碰—掷过程进行进一步的改进和拓展的基础上,提出对任意凸预约时间模糊数均实用的、解决多对多

货物收发情况下模糊 VRP 问题的改进推—碰—掷过程.

由前面可行染色体的生成可以看出,每一可行染色体可能包含多个车辆路径计划,而每一车辆路径计划又可分为一个或多个任何两相邻顾客间均没有等待时间的紧路径.如图 4 所示,假设一车辆从车场出发顺序服务 1,2,3,4,5 五个顾客后返回车场,每个顾客的开始服务时间如图所示,则该路径计划可分为如图所示两个紧路径.

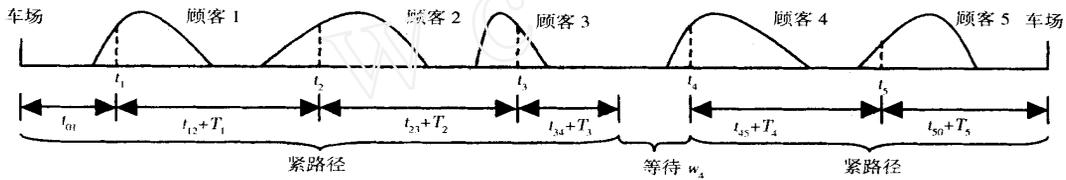


图 4 车辆路径计划图例

Fig.4 Example of vehicle routing

对任一车辆路径计划,确定顾客最佳服务时间的改进推—碰—掷过程进行步骤可归纳如下:

步骤 1 找出当前车辆路径计划最前面的紧路径,确定其不可推部分,并将其从当前紧路径中扔掉(“掷”出);

步骤 2 检查当前车辆路径计划中的顾客集合,如非空(即仍有顾客未被“掷”出),转步骤 3; 否则,过程结束;

步骤 3 确定当前紧路径(经过“掷”之后)的最优整体前推量 t_o ,比较 t_o 与当前紧路径与下一紧路径间等待时间长度 w 的大小,若 t_o 小于 w ,前推 t_o ,当前紧路径成为一个时间发生变化的新紧路径,返回步骤 1; 若 t_o 等于 w ,前推 t_o ,当前紧路径碰下一紧路径形成新的紧路径,返回步骤 1;

其中,任一紧路径中不可推部分的确定以及最优整体前推量的确定方法如下:

1) 紧路径中不可推部分的确定

不可推部分是指对该部分的任意前推(即延迟一段时间进行)都会导致该路径中顾客综合满意度的降低,或者造成该路径时间上的不可行,或者由于该部分的前推造成的当前紧路径中顾客综合满意度的提高量不如仅对该部分之后的顾客进行前推产生的顾客综合满意度的提高量大.

设一紧路由顾客 i_1, i_2, \dots, i_p 顺序组成,并令 $\mu_{i_k}(t_{i_k})$ 表示顾客 i_k 在时刻 t_{i_k} 时的满意度函数斜率,则当前紧路径中的不可推部分可按如下定

理确定.

定理 4 若 $\min_{s=1}^k \mu_{i_s}(t_{i_s}) < 0 \mid k = 1, 2, \dots, p$ 存在,且该最小值由 $\mu_{i_s}(t_{i_s})$ 产生,则 i_1, i_2, \dots, i_k 构成当前紧路径的不可推部分. 若 $\min_{s=1}^k \mu_{i_s}(t_{i_s}) < 0 \mid k = 1, 2, \dots, p$ 不存在,即所有 $\mu_{i_s}(k = 1, 2, \dots, p)$ 都大于 0,则该紧路径可整体前推.(证明略)

2) 最优整体前推量的确定

若某紧路径 i_1, i_2, \dots, i_p 可整体前推,即所有 $\mu_{i_s}(k = 1, 2, \dots, p)$ 都大于 0,则由于预约时间间的约束和下一紧路径开始时间的限制,当前紧路径的最大允许前推量为

$$= \min\{\min\{LT_{i_k} - t_{i_k} \mid k = 1, 2, \dots, p\}, w\} \quad (4)$$

其中, w 为当前紧路径与下一紧路径间的等待时间,如图 4 所示,两紧路径间的等待时间 w 为 w_4 .

令整体前推 t 后顾客的综合满意度为

$\mu(t) = \min_{s=1}^p \mu_{i_s}(t_{i_s} + t)$, 其斜率为 $\mu(t) = \min_{s=1}^p \mu_{i_s}(t_{i_s} + t)$, 则最优整体前推量可按如下定理确定:

定理 5 若某紧路径 i_1, i_2, \dots, i_p 可整体前推, 即所有 $\mu_{i_s}^k (k = 1, 2, \dots, p)$ 都大于 0, 且 $\mu(t) = \sum_{s=1}^p \mu_{i_s}^k (t_{i_s} + t) = 0$, 则在不违背预约时间约束以及不影响下一紧路径的条件下, 该紧路径最优整体前推量 t_o 在 t 时等于 t , 在 $t >$ 时等于 t . (证明略)

2.2 适值计算

适值根据原始目标函数值计算, 一个染色体的目标加权和表示如下:

$$g(ch_k) = \frac{1}{v_{m \times}} \frac{v_k}{d_{m \times}} + \frac{2}{d_{m \times}} \frac{d_k}{w_{m \times}} + \frac{3}{w_{m \times}} \frac{w_{ki}}{4 \left(1 - \frac{1}{2^m} \sum_{i=1}^4 \mu_i(t_i)\right)}$$

$i = 1, \dots, 4 \quad \mu_i = 0 \quad i = 1, 2, 3, 4$

其中, ch_k 表示第 k 个染色体, v_k 表示染色体 ch_k 中的车辆使用总数, $v_{m \times}$ 表示当前种群中染色体的最大车辆使用数; d_k 表示染色体 ch_k 的行驶距离, $d_{m \times}$ 表示当前种群中染色体的最大行驶距离; w_{ki} 表示染色体 ch_k 中车辆在顾客 i 处的等待时间, $w_{m \times}$ 表示当前种群中染色体的最大总等待时间; $\mu_i (i = 1, \dots, 4)$ 为权重系数. 则一个染色体的适值可计算如下:

$$eval(ch_k) = \frac{g(ch_{m \times}) - g(ch_k)}{g(ch_{m \times}) - g(ch_{min})}$$

其中, ch_{max}, ch_{min} 分别表示当前种群中目标加权和最大与最小的染色体.

2.3 判断停止进化条件

判断迭代的代数是否为要求代数 $Max-gen$, 若是, 停止进化, 选择性能最好的染色体所对应的路径集合作为原 VRP 问题的优化解输出.

2.4 自然选择

用赌盘法作选择机制, 并用精选法嵌入其中,

以保证最优个体可以生存到下一代, 并克服采样的随机误差.

2.5 杂交

在每代种群中, 以一定的交叉概率 p_c 进行交叉重组, 本文采用文献 [12] 中提出的一种新颖杂交算子进行杂交操作. 该交叉算子的基本做法是: 对任意给定的两个染色体 A 和 B , 随机产生两个交叉点, 将第一个染色体的交叉段移到对方染色体的首部得到新个体 $A1$ 和 $B1$, 消去相同项后得到两个新个体 $A2$ 和 $B2$. 该算子最大的特点是当两个父串相同时, 仍然能产生新的个体, 从而减弱了对群体多样性的要求, 能够有效避免传统遗传算法“早熟收敛”的缺点.

2.6 染色体变异

对每代种群以一定的变异率 p_m 进行染色体变异.

经过上述步骤可产生新一代种群, 然后, 对新一代种群中每一染色体按与 2.1 中相同的方法重新分配车辆以保证可行性, 并返回 2.3 循环.

3 实验分析

计算实验由 10 项任务组成, 每项任务的发点位置坐标和收点位置坐标以及其需求量都随机产生, 模糊预约时间则为随机生成的三角模糊数, 遗传算法基本参数设置见表 1.

表 1 遗传算法基本参数设置

Table 1 Basic setting of genetic algorithm parameters

Pop-size	Max-gen	p_c	p_m
50	400	0.3	0.03

基于上述设置, 表 2 给出了不同权重设置情况下的最终决策结果. 从表中可以看出, 车辆使用数与行驶距离密切相关, 即随着车辆使用数的增加, 总行驶距离也基本呈增加趋势; 而顾客满意度目标与其他所有目标都相对立, 即顾客满意度的增加是以运输企业成本的增加为代价的.

表 2 不同权重设置对最终决策结果的影响

Table 2 Results with varied weights

权重设置				决策结果			
1	2	3	4	v	d	w	μ
0.85	0.05	0.05	0.05	3	2 062.35	102.40	0.624
0.05	0.85	0.05	0.05	4	1 727.44	325.87	0.853
0.05	0.05	0.85	0.05	5	2 133.27	49.35	0.746
0.05	0.05	0.05	0.85	8	2 451.46	354.61	0.958
0.50	0.10	0.10	0.30	5	2 101.18	186.34	0.869

这里进一步将所提出的混合遗传算法与修正的推—碰—挪过程改进的Solomon插入启发式算法^[5]进行对比,表3给出了6次随机实验的对比结果

(权重值 w_1 、 w_2 、 w_3 、 w_4 分别设为 0.5、0.1、0.1 和 0.3)。从表3中的结果可以看出,本文所提混合遗传算法明显优于改进的Solomon插入启发式算法。

表3 混合遗传算法与Solomon插入启发式算法的比较

Table 3 Comparison between genetic algorithm and solomon insertion algorithm

实验	方法	μ	v	d	w	实验	方法	μ	v	d	w
1	GA	0.837	5	2 207.32	188.74	4	GA	0.734	3	1 766.26	97.72
	IH	0.662	6	2 516.75	101.53		IH	0.503	4	2 018.27	75.24
2	GA	0.794	4	1 952.04	146.55	5	GA	0.856	4	1 977.35	72.32
	IH	0.482	6	2 107.62	70.24		IH	0.677	4	2 139.52	79.49
3	GA	0.861	4	2 573.10	248.26	6	GA	0.883	5	2 185.41	197.76
	IH	0.700	4	2 794.66	187.43		IH	0.695	6	2 548.64	92.48

注: GA 表示本文所提混合遗传算法; IH 表示改进的Solomon插入启发式算法

4 结束语

在对具有模糊预约时间的多对多货物收发情况下的车辆路径问题进行简单描述的基础上,本文构建了该问题的多目标数学规划模型,提出了解决该问题的一种基于插入启发式算法的、并用

修正的推—碰—挪过程进行改进的混合遗传算法,最后,给出了该问题的一个计算实例。

本文研究了具有模糊预约时间的多对多货物收发情况下的车辆路径问题,更复杂的模糊车辆调度问题可能具有模糊旅行时间、模糊顾客需求时间、模糊费用系数以及动态特征等等,进一步的研究应该从这些方面入手,寻找解决这些问题的启发式或亚启发式算法。

参考文献:

- [1] Bodin L, Golden B, Assad A, et al. Routing and scheduling of vehicles and crews: The state of the art[J]. Computer and Operation Research, 1983, 10: 62—212.
- [2] Chiang W C, Russell R A. Simulated annealing metaheuristics for the vehicle routing problem with time windows[J]. Annals of Operations Research, 1996, 63: 3—27.
- [3] Chiang W C, Russell R A. A reactive tabu search metaheuristic for the vehicle routing problem with time windows[J]. INFORMS Journal on Computing, 1997, 9: 417—430.
- [4] Potvin J Y, Rousseau J M. An exchange heuristic for routing problems with time windows[J]. Journal of the Operational Research Society, 1995, 46: 1433—1446.
- [5] Solomon M M. Algorithms for the vehicle routing and scheduling problems with time window constraints[J]. Operations Research, 1987, 35: 254—265.
- [6] Gendreau Badeau P, Guertin M F, et al. A parallel tabu search heuristic for the vehicle routing problem with time windows[J]. Transportation Research-C 5, 1997. 109—122.
- [7] Potvin J Y, Kervahut T, Garcia B L, et al. The vehicle routing problem with time windows—part I: Tabu search[J]. INFORMS Journal on Computing, 1996, 8: 158—164.
- [8] Potvin J Y, Bengio S. The vehicle routing problem with time windows—part II: Genetic search[J]. INFORMS Journal of Computing, 1996, 8: 165—172.
- [9] 袁庆达, 杜文, 周再玲. 带软时间窗的混合车队车辆路线问题的模型和算法研究[J]. 西南交通大学学报, 2001, 4: 401—406.
Yuan Qingda, Du Wen, Zhou Zailing. Model and algorithms for mixed fleet vehicle routing problem with soft time windows[J]. Journal of Southwest Jiaotong University, 2001, 4: 401—406. (in Chinese)

- [10]郭耀煌,李 军. 车辆优化调度[M]. 成都:成都科技大学出版社,1994. 44—60.
Guo Yaohuang, Li Jun. Vehicle Routing Problem[M]. Chengdu: Press of Chengdu Science and Technology University, 1994. 44—60. (in Chinese)
- [11]Cheng R, Gen M. Vehicle routing problem with fuzzy due-time using genetic algorithms[J]. Japanese Journal of Fuzzy Theory and Systems, 1995, 7(5): 1050—1061.
- [13]张丽萍,柴跃廷. 车辆路径问题的改进遗传算法[J]. 系统工程理论与实践, 2002, 8: 79—84.
Zhang Liping, Chai Yueting. Improved genetic algorithm for vehicle routing problem[J]. System Engineering Theory and Practice, 2002, 8: 79—84. (in Chinese)
- [14]玄光南,程润伟. 遗传算法与工程设计[M]. 北京:科学出版社,2000. 244—256.
Xuan Guangnan, Chen Runwei. Genetic Algorithm and Engineering Design[M]. Beijing: Science Press, 2000. 244—256. (in Chinese)
- [15]Teodorovic D. Fuzzy set theory applications in traffic and transportation[J]. Europe Journal of Operation Research, 1994, 74: 379—390.
- [16]李敏强,张志强,寇纪淞. 关于杂合遗传算法的研究[J]. 管理科学学报, 1998, 1(1): 64—67.
Li Minqiang, Zhang Zhiqiang, Kou Jisong. Studies on hybrid genetic algorithms[J]. Journal of Management Sciences in China, 1998, 1(1): 64—67. (in Chinese)

Hybrid genetic algorithm to vehicle routing problem with fuzzy due-time

ZHANG Jian-yong¹, LI Jun², GUO Yao-huang³

1. International Commercial College, Nankai University, Tianjin 300071, China;
2. School of Economics and Management, Tianjin Professional College, Tianjin 300402, China;
3. School of Economics and Management, Southwest Jiaotong University, Chengdu 610031, China

Abstract: The traditional vehicle routing problem with time windows (VRPTW) is expanded to the situation, where the time window is replaced by fuzzy due time which can represent the preferences of the customers. After a simple description of the fuzzy dial-a-ride problem, a multi-objective mathematical model for the problem is built up. Then, an insertion heuristic-based hybrid genetic algorithm is proposed to this kind of problem. In this algorithm, the modified push-bump-throw procedure is employed to handle the fuzzy nature of the problem. Finally, an example is presented, the superiority of the genetic algorithm is proved through the comparison of the genetic algorithms with improved Solomon insert heuristic algorithm.

Key words: fuzzy due time; vehicle routing problem; hybrid genetic algorithm