

一种求解混载校车路径的启发式算法

党兰学 王震 刘青松 孔云峰

(河南大学环境与规划学院 开封 475004)

摘要 对一个区域内多所学校进行校车路径规划时,允许校车混载不同学校的学生能显著地减少校车数量,从而降低运营成本。已有学者针对允许混载的校车路径问题(SBRP)提出了启发式算法,但这些算法对邻域解的搜索不够全面,在缩减路径方面仍有较大的提升空间。提出了一种以记录更新法(record-to-record travel, RRT)为基础的启发式算法。该算法从初始解出发,利用求解有时间窗装卸问题(PDPTW)时使用的算子搜索邻域解,逐步优化校车路径数目。与现有算法相比,该算法扩展了求解混载 SBRP 的启发策略,能够在全局范围内对校车路径进行优化,从而获得所需校车较少的路径规划方案。实验结果验证了该算法的有效性。

关键词 校车路径问题,混载,有时间窗装卸问题,记录更新法

中图分类号 TP18 **文献标识码** A

Heuristic Algorithm for Solving Mixed Load School Bus Routing Problem

DANG Lan-xue WANG Zhen LIU Qing-song KONG Yun-feng

(College of Environment and Planning, Henan University, Kaifeng 475004, China)

Abstract Mixed load school bus routing problem (SBRP) for multiple schools in a city or county, which allows students from different schools to get on same bus at the same time, aims to reduce the number of school buses needed and total operational costs. Several algorithms for solving mixed load SBRP have been developed since 1990s. However, due to the complexity of mixed load SBRP, the neighborhood solutions are not fully explored in these approaches. This paper proposed a new heuristic method for mixed load SBRP using a record-to-record travel (RRT) algorithm and neighborhood operators of pickup and delivery (PDPTW) to minimize the number of required buses. Test results show the effectiveness of this algorithm. Compared with the existing SBRP solutions, this algorithm expands the neighborhood search strategy, and is capable of finding better solutions using fewer buses.

Keywords School bus routing problem (SBRP), Mixed load, Pickup and delivery with time window (PDPTW), Record-to-record travel (RRT)

1 引言

为中小学学生提供安全、高效的校车服务是教育主管部门的一项重要职能,也是我国义务教育发展面临的新要求。然而如何设计校车路径以减少运营成本是地方教育主管部门面临的难题之一。校车服务需要考虑的因素众多:不仅要满足校车容量限制、学生与学校时间窗限制、最大乘车时间限制等各种约束,同时要求尽可能减少校车数量和运行成本。因此,校车路径问题(SBRP)的本质是在满足校车服务各种规定约束的条件下,兼顾运营成本和服务水平的车辆路径优化问题。

自 Newton 和 Thomas^[1]首次提出 SBRP 后,众多学者一直在探索相关的数学模型、优化算法及其应用。从现实的应用中发现,减少校车的数量可以显著地降低成本。为了减少校车数量,提高校车的使用效率,允许混载多个学校学生的校

车服务应运而生,针对混载校车路径问题(混载 SBRP)的研究也逐渐受到学者们的关注。

混载 SBRP 属于典型的 NP 难题,精确方法仅适用于小规模的应用。文献[2-4]针对混载 SBRP 的特点,分别给出了启发式的求解算法。混载校车路径上站点的调整都是成对的(学生站点和对应的学校站点),站点的移动比较复杂。现有算法使用的启发策略在邻域解搜索方面并不充分,解算结果具有一定的局限性。相对于只服务于一所学校的单校校车路径问题(单校 SBRP),目前混载 SBRP 在线路优化方面仍有较大的提升空间。

本文以记录更新法(record-to-record travel, RRT)^[5]为基础,提出一种求解混载 SBRP 的算法。该算法扩展了混载 SBRP 的启发搜索策略,利用求解带有时间窗装卸问题(PDPTW)时使用的邻域算子^[6-8],在路径间、路径内重新安排站点(对)的位置启发搜索邻域解,从而优化校车路径。与现

到稿日期:2012-09-18 返修日期:2012-12-19 本文受国家自然科学基金项目(41201402),省部共建河南大学科研基金项目(SBGJ090605)资助。
党兰学(1980-),男,博士生,CCF 会员,主要研究方向为智能算法、空间分析与优化,E-mail:danglx@foxmail.com;王震(1976-),男,博士生,讲师,主要研究方向为空间分析与优化;刘青松(1987-),男,硕士生,主要研究方向为空间优化及算法设计;孔云峰(1967-),男,博士,教授,博士生导师,主要研究方向为空间分析与应用、GIS 分析与设计,E-mail:yfkong@henu.edu.cn(通信作者)。

有的算法^[2,4,31]相比,该算法在启发过程中能够生成更多可供选择的邻域解,从而得到需要更少车辆数目的校车路径方案。

2 相关工作

SBRP 是车辆路径问题(VRP)的一个重要分支和应用领域^[9,10],它在 VRP 的基础上,增加了与学生相关的因素,比如学生与学校的对应关系、学生最大乘车时间或最长乘车距离等^[11]。SBRP 的约束条件通常包括:校车容量、最大乘车距离或时间、学校时间窗口、学生最早乘车时间等。其优化目标主要有:最少化校车数量、最小化车辆总的运行距离或时间、最小化学生总的乘车距离或时间等。根据校车服务学校的数量,SBRP 可分为单校 SBRP 和多校 SBRP^[12,13],依据是否允许不同学校学生搭乘同一校车,又将多校 SBRP 分为不允许混载和允许混载两种类型^[2,4,12]。

已发表的 SBRP 研究文献多属于单校 SBRP^[14]。针对单个学校的 SBRP,可以通过构造带有时间窗和容量约束的车辆路径问题模型^[15],对单个学校的校车线路、乘车时间、时间窗口等进行统筹。相关的模型和算法比较成熟,基本能够进行较为有效的交通服务规划。国内学者针对单个学校校车路径问题也进行了相关的研究,以服务于学前儿童^[16-18]、中小学学生上学^[17,18]或教职员工上班^[19,20]。张苗等^[18]的研究涉及了多个学校校车路径规划的问题,但每辆车仅服务于单个学校,本质上仍属于单校 SBRP。

不允许混载的多校 SBRP 研究,将路径生成和车辆调度问题结合起来,根据运行成本、学校开学的时间窗等约束条件来分配车辆,从而使校车能够服务于多个学校^[21,22]。求解思路一般是先对单个学校生成校车路径段,然后在车辆和路径段之间进行指派。这种类型的校车服务虽然可以针对多个学校,但是必须严格地按先后顺序进行,只有将一个学校的学生送达学校后才可以开始服务另一个学校的学生。

允许混载的校车可以同时搭载不同学校的学生,按学校开学先后顺序将学生送至对应的学校。文献^[2-4,12,13,23,24]对混载 SBRP 进行了研究,其中文献^[2-4]给出了启发式算法。在时间窗口事先确定且不考虑换乘车辆的前提下,Braca 等^[2]讨论了纽约市曼哈顿区 73 所学校校车路径与时间安排,采用随机的基于位置启发(LBH)算法产生可行的校车线路,在迭代运算过程中,检验约束条件。RLBH 算法^[2]是逐条有序地生成路径,对先插入到路径上的站点不再依据后续站点重新调整,也只有当一条路径服务学生人数少于 11 时才会在路径间进行站点调整以达到各路径服务能力的均衡。de Souza 等^[3]进一步改进 LBH 算法,从最远的学生站点构造线路,用于巴西 Parana 州 10 个城市的校车交通安排。Park 等^[4]提出了多阶段的方法,首先针对每个学校独立地生成路径,然后利用匈牙利算法将车辆指派到各条路径,生成不允许混载的服务于多个学校的校车路径,最后尝试将路径合并生成允许混载的校车路径。Park 的算法能在满足校车服务约束的前提下显著地减少校车数量。但该算法在最后的提高阶段要求一次必须将一条路径上的所有站点并入其他路径,如果不能,则撤销。在合并路径过程中没有在路径内或路径间对站点(对)的位置再进行调整优化。

总体来看,混载 SBRP 的研究相对较少,仍处在不断地探索中。现有算法邻域解的搜索有限,并没有在全局范围内进

行线路优化。很多用来解决 VRP 的方法并没有体现在混载 SBRP 的应用中,详细的文献回顾可以参考文献^[2,14,25]。

3 混载校车路径问题模型

SBRP 通常分为 5 个子问题:数据准备、站点选择、路径生成、学校开学时间调整和车辆调度^[14]。除了数据准备外,其他 4 个问题都属于 NP 难问题,一般学者都是选择其中一个问题进行研究,即使对多个进行研究也是按顺序分步进行。研究问题的范围和关注角度不同,提出的约束和目标会有差异。本文的研究属于路径生成,目标是 minimized 校车数量。

在研究校车路径生成时考虑的约束条件通常包括校车容量、学生最大乘车时间、学校开学时间等,体现了校车服务方、学生和学校等各方的约束。优化的目标一般是从运营成本和服务水平两个方面来考虑,主要体现在以下 3 个方面:(1)所用车辆数最少;(2)所有车辆运行的总里程最短;(3)学生乘车的总时间最少。

现实世界中 SBRP 问题非常复杂,对其进行定义时通常会做一些限定和抽象。混载 SBRP 允许不同学校学生乘坐同一辆校车,一条路径可能存在多个学校站点,交替路过学生站点和学校站点,上下车交替进行。PDPTW 在收集的过程中同时发送,因此通常将站点分为收集站点和发送站点。若将学生站点看作收集站点(上车站点) P^+ ,学校站点看作发送站点(下车站点) P^- ,则混载 SBRP 与 PDPTW 模型比较接近。

在多个场站构成的集合 D 内,一共有 K 辆校车,分别位于各自的场站 $d(k)$,拥有不同的容量 Q_k 。每条路径上的车辆 k 从场站 $d(k)$ 出发,经过一系列学生站点和学校站点之后,最终返回到 $d(k)$ 。场站的位置可以根据实际情况来确定,比如专门的停车场或学校,甚至可以是司机的家庭居住地。

任意两个站点 i 和 j 间的运行距离和时间分别为 c_{ij} 和 t_{ij} ,在现实世界中,可以使用 GIS 的网络分析工具计算获得。本文所用的实验数据集中, c_{ij} 使用两点间的曼哈顿距离 $(|x_i - x_j| + |y_i - y_j|)$,而 t_{ij} 则由 c_{ij} 除以车辆的平均速度计算得到。

表 1 数学模型中使用的符号

参数	
P^+	学生站点集合
P^-	学校站点集合
P	学生站点和学校站点的并集,即 $P^+ \cup P^-$
D	场站集合
V	所有站点,包括学生站点、学校站点和场站,即 $P \cup D$
$s(i)$	表示学生站点 i 所对应的学校站点, $i \in P^+, s(i) \in P^-$
K	车辆集合
$d(k)$	第 k 辆车所在的场站, $k \in K$
Q_k	第 k 辆车的容量, $k \in K$
R	学生的最大乘车时间
c_{ij}	站点 i 到站点 j 的行驶距离, $i, j \in V$
t_{ij}	站点 i 到站点 j 的行驶时间, $i, j \in V$
e_i, l_i	站点的最早到达时间和最晚到达时间, $i, j \in V$
n_i	站点 i 上(下)车的学生人数, $i \in P$
t_i	站点 i 所需要的服务时间(上车或下车), $i \in P$
决策变量	
x_{ijk}	如果第 k 辆车经过站点 i 到 j 则为 1,否则为 0
y_{ik}	如果第 k 辆车访问站点 i 则为 1,否则为 0
T_{ik}	第 k 辆车到达站点 i 的时间
L_{ik}	第 k 辆车访问站点 i 后车上的学生人数

每个站点都有最早开始服务时间 e_i 和最晚开始服务时间 l_i ,即时间窗 $[e_i, l_i]$ 。学校具有严格的时间窗,校车到达学

校的时间应该处于时间窗内;学生站点的时间窗则相对宽松,以场站开放时间为最早时间 e_i ,以对应学校的 l_i 为最晚时间。校车路径优化的目标是运营成本最小,但是为了保证校车的服务质量,限定了学生乘车时间不超过一个固定的值 R 。表 1 列出模型中所用到的参数和决策变量。

利用上述参数和决策变量,在文献[2,4,10]的基础上,定义本文所研究的混载 SBRP 的混合整型规划模型。

$$\min \sum_{j \in P^+} \sum_{k \in K} x_{d(k),j} \quad (1)$$

$$\text{s. t. } \sum_{k \in K} y_{ik} = 1, i \in P^+ \quad (2)$$

$$\sum_{j \in V} x_{ijk} - \sum_{j \in V} x_{jik} = 0, i \in P, k \in K \quad (3)$$

$$\sum_{j \in V} x_{jik} - \sum_{j \in V} x_{js(i)k} = 0, i \in P^+, k \in K \quad (4)$$

$$T_{ik} + t_i + t_{v(i)} \leq T_{s(i)k} \quad (5)$$

$$\sum_{j \in P} x_{d(k),j} = \sum_{j \in P} x_{jd(k)} = 1, k \in K \quad (6)$$

$$L_{ik} = 0, i \in D, k \in K \quad (7)$$

$$n_i \leq L_{ik} \leq Q_k, i \in P^+, k \in K \quad (8)$$

$$0 \leq L_{s(i)k} \leq Q_k - n_i, i \in P^+, k \in K \quad (9)$$

$$x_{ijk} (L_{ik} + n_j - L_{jk}) = 0, j \in P, k \in K \quad (10)$$

$$T_{s(i)k} - T_{ik} \leq R, i \in P^+ \quad (11)$$

$$e_i \leq T_i \leq l_i, i \in V \quad (12)$$

$$x_{ijk} \in \{0, 1\}, i, j \in P, k \in K \quad (13)$$

$$y_{ij} \in \{0, 1\}, i, j \in P \quad (14)$$

函数(1)以最小化校车数量为优化目标。约束(2)限定所有的学生站点都必须由其中一辆车提供服务,并且只允许被访问一次。约束(3)确保一辆车驶入站点 i 后离开站点 i 。约束(4)保证经过学生站点的车辆必须经过该站点所属的学校。约束(5)为访问顺序限定,要求校车必须先访问学生站点然后才能访问它对应的学校。约束(6)保证所有车辆从场站出发,最后又回到场站。约束(7)规定校车从场站出发或者回到场站时容量为 0,即出发和结束时车上都没有学生。约束(8)和(9)对校车经过路径上一个站点对时车上的学生数进行了限定,约束(8)表示任何车辆 k 经过学生站点 i 后人数的变化,而约束(9)表示车辆经过学校站点时车上学生数量的变化。约束(10)表示的是连续站点间容量的变化关系。需要说明的是如果站点 j 为学校, n_j 为负值,表示所有应该到该站下车的学生数。约束(11)为最大乘车时间限制,任何学生从上车站点到学校的时间都不超过 R 。约束(12)为时间窗限制,到达任何站点的时间都必须在时间窗内。约束(13)和(14)将决策变量限定为 0 或 1。

4 算法设计

4.1 邻域搜索算子

混载 SBRP 算法的复杂性主要体现在站点的各种操作都是成对的,它们的访问有先后顺序,每次对学生站点的移动都要考虑其对应的学校。顺序要求导致 CW 算法中的路径逆序操作不能使用, VPR 优化中常用的 One-point move、Two-point move、Two-opt move、Or-opt move、Three-opt move、Cross-exchange 等邻域操作^[26],大部分都需做相应修改才能使用。站点成对操作的特性与 PDPTW 相似。Nanry 等^[6]利用适应性禁忌搜索^[27]算法求解 PDPTW,定义了 3 种邻域算子,分别是单个站点对插入(Single Paired Insertion, SPI)、路径间站点对交换(Swapping pairs between routes, SBR)、路径内插入(Within route insertion, WRI),它们对应的操作如图 1

一图 3 所示^[6,7]。PDPTW 的后续研究^[7,8,28]也使用了这些算子。本文提出的启发式算法使用 SPI、SBR 和 WRI 这 3 个邻域操作算子对路径进行优化。

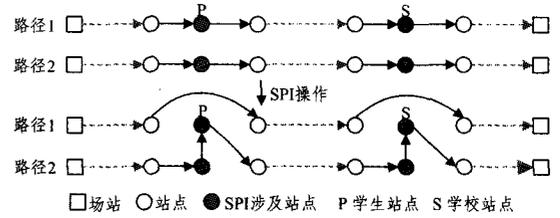


图 1 SPI 操作

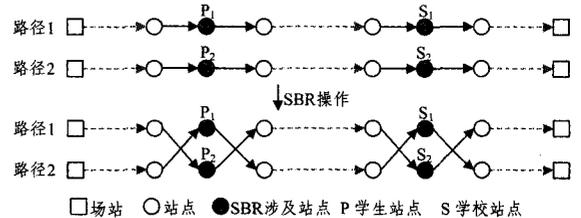


图 2 SBR 操作

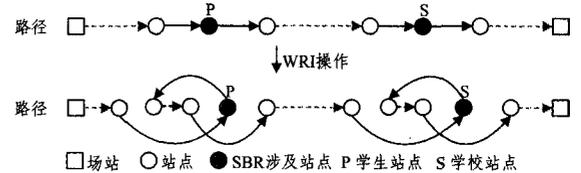


图 3 WRI 操作

SPI 尝试将一个站点对(学生站点,学校站点)从一条路径移到另外一条路径,这种移动能够减少路径数目,从而最大限度地提高目标函数。移动一个站点对到其他路径时,首先要找到适合学生站点的插入位置,然后寻找合适的位置插入学校站点,插入时必须确保其满足时间窗、容量、最大乘车时间等约束。PDPTW 中上车站点和下车站点是一一对应的关系,而混载 SBRP 中乘车站点与学校站点是多对一的关系。应用邻域操作算子时要注意学生站点与学校站点之间的对应关系,比如图 1 执行 SPI 时是否将学校 S 从路径 1 中移除,还需要检测路径 1 中除站点 P 之外是否还有到 S 下车的站点,如果有,则需要保留 S ,线路 2 中如果已有对应的学校站点 S ,则无需再次插入。

SBR 对现有两个不同路径上的站点对进行交换。通过一段时间的 SPI 移动达到局部最优,无法再找到能够改进目标函数的 SPI 移动。而 SBR 通过交换点对来改变搜索策略以发现新的可行解。

WRI 是通过在路径内调整学生站点和学校站点来优化单条路径,站点调整可以单独进行也可以成对进行,但调整必须在满足各种约束的前提下进行。虽然 WRI 不能减少路径,但 WRI 通过改变路径上的站点顺序可以使单条路径达到最优,为继续进行 SPI、SBR 等操作提供可能。学生站点的时间窗很大,因此在同一条路径上对站点的前后移动会得到较多的其他可行解,从而确定更合理的方案。

假定 S 是混载 SBRP 的一个实例, R 是 S 的一个可行解,通过点对的移动可以得到 R 的邻域解。SPI、SBR 和 WRI 邻域操作算子在算法中分别用 $N_{SPI}(p, R)$, $N_{SBR}(p, R)$, $N_{WRI}(p, R)$ 来表示, p 为移动的站点对中的学生站点,对应的学校

站点通过 school(p)可得到。

4.2 算法框架

本文的算法从初始解出发,以 RRT 作为基础,通过邻域操作算子优化线路,算法整体上描述如下:

- (1) 读入数据文件,初始化 SBRP 实例;
- (2) 将每个学生站点和它所对应的学校连接起来构成一条路径,所有路径的集合即为后续优化所用的初始解;
- (3) 调用启发式算法 RouteMinimize_RRT 改进初始解;
- (4) 输出路线方案及相关辅助信息。

算法在启发过程中需要一个合适的评价函数来评估是否可以接受产生的邻域解。该算法以最小化校车数目为目标,过度地关注总里程会使得减少车辆数目变得困难,因此采用的评估函数为:

$$Cost(R) = \alpha|R| + \beta\left(-\sum_{r \in R} |r|^2\right) + \gamma\sum_{r \in R} c(r)$$

式中, $|R|$ 表示解包含的路径数, $|r|^2$ 为其中一条路径上服务学生站点的数量, $c(r)$ 为路径 r 的行驶距离。 $\alpha > \beta > \lambda$, 即 3 个量是按照字典序依次进行评价。 $|r|^2$ 增大意味着解决方案中存在某些路径正向服务更多的站点的方向发展,更接近于路径的合并减少^[26,28]。

4.3 减少路径数的 RRT 算法

RRT 算法由 Dueck 提出^[5],对 TSP 实例测试取得了很好的效果。文献[26,29,30]等将 RRT 算法应用到 VRP 当中,已有研究显示了 RRT 在路径优化问题中的良好表现。RRT 算法属于模拟退火算法(SA)的确定性变种^[30],同 SA 一样有一个多次的循环,区别在于不是通过初始温度和冷却系数来控制,而是直接设定一个固定的循环次数。为避免陷入局部最优,SA 在迭代过程中根据概率函数来接受没有提高的新解 R^* 。RRT 用 Record 记录当前最优解,设定一个偏差系数 deviation,当 $Cost(R^*) < (1 + deviation) \times Record$ 时即可接受新解 R^* ,只有 $Cost(R^*) < Record$ 时将 Record 更新为新解。本文中仅将 deviation 作用于评价函数中的路径长度上。

搜索一个站点所有邻域的时间复杂度会随着问题规模增加而急剧增大。直觉上距离过远的两个站点连接降低运行成本的可能性较小,因此可以通过限定邻域表长度来提高计算速度,已有研究证明了这种策略的有效性^[26,29,30]。

RRT 算法的实现如算法 1 所示。第 1 行从 SBRP 实例 S 中构建学生站点列表 PList;第 2 行对 Record 赋初始值;第 3—8 行循环执行邻域操作,在此阶段由 deviation 控制允许接受没有提高(路径不减少)的解,称之为上坡(uphill)阶段。第 5 行调用 SPI 函数改进当前解,第 6 行和第 7 行分别执行 WRI、SBR 操作进行局部优化。第 10—12 行再次执行 SPI、WRI、SBR 邻域操作,与第 3—8 行循环不同的是此时称之为下坡(downhill)阶段,只接受 Cost 值得到提高的解。第 4 行和第 9 行对 PList 随机重新排列。

算法 1 启发式算法 RouteMinimize_RRT

输入: SBRP 实例 S、初始解 R、循环次数 M、邻域表长度 L、偏差系数 deviation。

输出: 路径方案。

- (1) PList = GetPickupStop(S);
- (2) Record = Cost(R);

- (3) for(i=0; i<M; i++) {
- (4) permutation(PList);
- (5) R = SPI(R, PList, L, deviation);
- (6) R = WRI(R, PList, 0);
- (7) R = SBR(R, PList, 0);
- (8) }
- (9) permutation(PList);
- (10) R = SPI(R, PList, L, 0);
- (11) R = WRI(R, PList, 0);
- (12) R = SBR(R, PList, 0);
- (13) return R;

算法 2 是 SPI 邻域搜索及记录更新的算法,主要是通过一个循环,对 PList 中的每个学生站点依次寻找最优的插入位置。第 2 行 $N_{SPI}(R, p, L)$ 在当前解 R 内搜索站点 p 的最佳插入位置,邻域的搜索范围限定在 L 的长度内。第 3 行限定了新解的接受条件,第 6 行判断是否需要更新 Record。

算法 2 邻域搜索及记录更新算法 SPI

输入: 路线方案 R、学生站点列表 PList、邻域表长度 L、偏差系数 deviation。

输出: 路径方案。

- (1) for(each p in PList) {
- (2) $R^* = N_{SPI}(R, p, L)$;
- (3) if($Cost(R^*) < (1 + deviation) \times Record$)
- (4) $R = R^*$;
- (5) }
- (6) if($Cost(R) < Record$)
- (7) Record = Cost(R);
- (8) Return R;

WRI 和 SBR 的实现与 SPI 类似。WRI 限定在一条路径内搜索邻域,而 SBR 执行次数较少,所以 WRI 和 SBR 并没有用 L 来限定邻域列表。

4.4 算法分析

在上节中给出的算法中,每次迭代过程利用 SPI、SBR 和 WRI 算子搜索邻域解,因此这 3 种算子的执行情况决定了整个算法的复杂度。为了方便分析算法的时间复杂度,可以认为学生站点和学校是一对一的,即为每个学生站点虚拟一个学校站点,且不考虑邻域表长度 L 的限制。假设路径方案中需要 K 辆校车为 N 个站点服务,RRT 算法的循环次数为 O(M)。以 $T(K, N)$ 为局部搜索算子总的复杂度,则算法的时间复杂度为 $O(M) \cdot T(K, N)$ ^[7]。

在平均情况下,每条路径上有 $\lceil \frac{N}{K} \rceil$ 个站点,即 $\lceil \frac{N}{2K} \rceil$ 个站点对。算法执行时首先将所有的站点对形成一个列表 PList,其中包含了 $\frac{N}{2}$ 个站点对。

SPI 操作首先从 PList 中选择一个站点对,共有 $\frac{N}{2}$ 种选择。选定的站点对有 K-1 条路径可以作为移动目标。移动点对到其他路径上时,需要分别确定站点 p 和对应学校站点 school(p) 的插入位置,各有 $O\left(\lceil \frac{N}{K} \rceil\right)$ 种可能。因此 SPI 操作的时间复杂度为 $O\left(\frac{N}{2} \cdot K \cdot \lceil \frac{N}{K} \rceil \cdot \lceil \frac{N}{K} \rceil\right)$,即 $O\left(\frac{N^3}{K}\right)$,这与文献[6]的分析基本是一致的。文献[7]分析的复杂度高达

$O(\frac{N^3}{K^2})$,主要原因在于对路径内任意两个站点都组合成站点对尝试邻域搜索,其实站点对的关系在原始数据中已经是固定的,没必要做大量的随机组合。

与 SPI 一样, SBR 操作同样是先从 $PList$ 中选择一个站点对,共有 $\frac{N}{2}$ 种选择。选定的站点对可以和其他 $K-1$ 条路径上的站点对进行交换,每条路径上可用来进行交换的站点对有 $O(\lfloor \frac{N}{2K} \rfloor)$ 个。SBR 选择好交换的点对以后,并不去搜索对方路径内的最优插入方案,而直接交换双方的位置^[6]。因此 SBR 的时间复杂度为 $O(\frac{N}{2} \cdot K \cdot \frac{N}{2K})$,即 $O(N^2)$ 。

WRI 操作从 $PList$ 中选择一个站点对也有 $\frac{N}{2}$ 种可能,但选定的站点对只能在其路径内重新安排位置。调整站点对时需在其路径内分别为 p 和 $school(p)$ 确定一个新位置,各有 $O(\lfloor \frac{N}{K} \rfloor)$ 种可能,因此 WRI 的时间复杂度为 $O(\frac{N}{2} \cdot \lfloor \frac{N}{K} \rfloor \cdot \lfloor \frac{N}{K} \rfloor)$,即 $O(\frac{N^3}{K^2})$ 。

$T(K, N) = \max\{\frac{N^3}{K}, N^2, \frac{N^3}{K^2}\}$,由于 K 总是小于 N ,得到 $T(K, N) = O(\frac{N^3}{K})$ 。所以总的时间复杂度为 $O(M) \cdot T(K, N) = O(M \cdot \frac{N^3}{K})$ 。

5 实验结果及分析

为了测试本文设计算法的有效性,利用文献[4,31]提供的网址下载 SBRP 数据集运行程序。该数据集专门针对研究混载 SBRP 设计的,在长宽均为 32.2km(20 英里)的范围内生成了若干学校和学生站点。RSRB 类测试案例中学校 and 乘车站点分布是随机的, CSCB 类测试案例中学校 and 乘车站点则是相对集中地分布在若干个积聚区。表 2 给出各个测试案例的基本情况,体现了问题的规模,数据集的具体描述可以参考文献[4]。为了便于对比,问题约束的设置与文献[4]一致,校车容量限制为 66,最大乘车时间分别设置为 2700s(45min)和 5400s(90min)。车辆的平均行驶速度为 32.2km/h(20 英里/小时)。各个站点的服务时间^[2,4]为:

$$t_i = \begin{cases} 19.0 + 2.6n_i, & \text{站点 } i \text{ 为乘车站点} \\ 29.0 + 1.9n_i, & \text{站点 } i \text{ 为学校} \end{cases}$$

式中, n_i 为在站点 i 上车或下车的学生数, t_i 的单位为 s。

表 2 测试案例基本情况

实例	学校个数	站点个数	学生人数	实例	学校个数	站点个数	学生人数
RSRB01	6	250	3409	CSCB01	6	250	3907
RSRB02	12	250	3670	CSCB02	12	250	3204
RSRB03	12	500	6794	CSCB03	12	500	6813
RSRB04	25	500	6805	CSCB04	25	500	7541
RSRB05	25	1000	13765	CSCB05	25	1000	16996
RSRB06	50	1000	12201	CSCB06	50	1000	18232
RSRB07	50	2000	26912	CSCB07	50	2000	27594
RSRB08	100	2000	31939	CSCB08	100	2000	27945

RRT 算法的参数设置在参考 Li^[30]解决大规模 VRP 问题的基础上,通过实验,确定循环次数 $M=30$ 。偏差系数设

置需要考虑 Cost 函数值的规模,这里设置 $deviation=10^{-5}$ 。RSRB 类型案例站点随机分布,设置邻域大小 $L=150$; CSCB 类型案例站点属于集聚性分布,一定距离内的邻居站点较多,设置 $L=300$,站点数小于 300 时以站点数为准。

表 3 和表 4 分别给出 RSRB 和 CSCB 的测试结果(黑体表示优于其他算法的结果)。RRT 算法采用 Visual Studio 2010 环境中的 VC++ 实现,测试环境为 PC 机,配置为 Intel Core i5-2400 3.1G CPU, 2G RAM, 操作系统为 Windows 7。 V_{RRT} 、 V_{Park} 、 V_{Braca} 分别是 RRT(5 次实验结果取最优值)、Park^[4,31]和 Braca^[2]算法得到的车辆路径数目, T_{RRT} 、 T_{Park} 、 T_{Braca} 是对应的运行时间,其中 V_{Park} 、 V_{Braca} 、 T_{Park} 、 T_{Braca} 是由文献^[31]给出的数据。

表 3 RSRB 案例测试结果

实例	最大乘车时间/s	车辆数			运算时间/s		
		V_{RRT}	V_{Park}	V_{Braca}	T_{RRT}	T_{Park}	T_{Braca}
RSRB01	2700	28	30	36	39.6	0.9	26.6
RSRB02	2700	26	29	34	58.3	0.9	32.2
RSRB03	2700	55	56	77	164.9	3.2	123.8
RSRB04	2700	55	59	78	190.7	2.2	121.1
RSRB05	2700	93	98	118	529.7	5.0	574.7
RSRB06	2700	82	89	106	747.3	5.8	695.5
RSRB07	2700	149	154	201	2365.5	40.7	3296.1
RSRB08	2700	156	157	212	2826.0	18.2	3455.6
RSRB01	5400	26	27	31	38.6	0.9	46.2
RSRB02	5400	23	23	29	55.9	0.7	43.3
RSRB03	5400	47	47	61	180.7	3.4	166.8
RSRB04	5400	42	46	60	205.2	2.1	183.7
RSRB05	5400	78	79	86	570.5	5.6	866.9
RSRB06	5400	66	72	81	805.4	5.8	760.8
RSRB07	5400	132	134	163	2593.3	43.1	4422.7
RSRB08	5400	134	142	186	2912.3	26.4	5728.4
平均	—	74.5	77.6	97.4	892.7	10.3	1284.0

表 4 CSCB 案例测试结果

实例	最大乘车时间/s	车辆数			运算时间/s		
		V_{RRT}	V_{Park}	V_{Braca}	T_{RRT}	T_{Park}	T_{Braca}
CSCB01	2700	29	30	39	40.8	1.0	26.9
CSCB02	2700	27	30	38	73.7	0.9	31.1
CSCB03	2700	51	55	69	390.4	4.4	154.3
CSCB04	2700	57	62	84	364.4	1.7	109.0
CSCB05	2700	108	116	157	1124.5	10.6	560.8
CSCB06	2700	110	120	138	1263.7	10.4	627.8
CSCB07	2700	161	175	229	5137.3	12	4733.3
CSCB08	2700	160	175	226	5434.9	24.1	6125.7
CSCB01	5400	24	24	33	57.8	1.1	77.0
CSCB02	5400	21	22	26	80.9	0.8	82.4
CSCB03	5400	40	41	50	293.4	3.9	544.4
CSCB04	5400	41	43	67	443.7	1.7	395.0
CSCB05	5400	93	97	134	1152.6	9.9	1269.6
CSCB06	5400	96	102	110	1334.2	13.2	1411.6
CSCB07	5400	130	133	179	5455.0	16.7	8266.2
CSCB08	5400	137	141	174	5562.2	27.8	5943.8
平均	—	80.3	85.4	109.6	1763.1	8.8	1897.4

从表 3 和表 4 可以看出, RRT 算法求出的路径数总体上明显优于 Park 和 Braca 的算法。Braca 算法的主要缺陷在于其贪婪的特性,在逐条生成路径的过程中尽最大可能将剩余站点合并到当前构建的路径中,算法后期未访问的站点大多是距离学校较远的站点,互相难以合并。Park 算法将大规模问题按学校划分为若干个小规模的问题,分别求解后再合并路径,算法具有很强的时间优势。但是 Park 算法将问题分解后,后期的合并要求整条合并,并没有在全局的范围内对路径

进行优化。本文的RRT算法使用SPI、WRI、SBR 3种邻域操作算子,在全局范围内求解大规模的混载SBRP问题,能够更进一步减少路径数。校车路径优化不是实时的需求^[2],一般仅在每个学期开学前或需要变更路径时执行,因此RRT算法的执行时间是可以接受的。

结束语 本文在分析SBRP研究现状的基础上,指出了现有求解混载SBRP算法在邻域搜索上的不足。针对混载SBRP设计了一个以RRT为基础的启发式求解算法,其引入了求解PDPTW时采用的SPI、WRI、SBR 3种邻域操作算子,在路径间和路径内调整站点,扩展了邻域解的启发策略。

实验表明本文提出的启发式算法是有效的,与现有算法相比,能够优化得到需要车辆数更少的路径方案。后续将对RRT算法参数的设置与数据特征之间的关系进行研究,同时利用真实的数据进一步测试算法的效果。

参 考 文 献

- [1] Newton R M, Thomas W H. Design of school bus routes by computer[J]. Socio-Economic Planning Sciences, 1969, 3(1): 75-85
- [2] Braca J, Bramel J, Posner B, et al. A computerized approach to the New York City school bus routing problem[J]. IIE Transactions, 1997, 29: 693-702
- [3] de Souza L V, Siqueira P H. Heuristic Methods Applied to the Optimization School Bus Transportation Routes-A Real Case[C]// IEA/AIE 2010, Part II. Berlin: Springer Verlag, 2010: 247-256
- [4] Park J, Tae H, Kim B-I. A post-improvement procedure for the mixed load school bus routing problem[J]. European Journal of Operational Research, 2012, 217: 204-213
- [5] Dueck G. New optimization heuristics: The great deluge algorithm and the record-to-record travel[J]. J. Comput. Phys., 1993, 104: 86
- [6] Nanry W, Barnes J. Solving the pickup and delivery problem with time windows using reactive tabu search[J]. Transportation Research Part B, 2000, 34: 107-21
- [7] Li H, Lim A. A metaheuristic for the pickup and delivery problem with time windows[C]// 13th IEEE International Conference on Tools with Artificial Intelligence (ICTAI). 2001: 160-167
- [8] Lau H, Liang Z. Pickup and delivery with time windows: algorithms and test case generations[C]// Proceedings of the 13th IEEE Conference on tools with Artificial Intelligence (ICTAI). 2001: 333-340
- [9] Golden B, Assad A, Levy L, et al. The fleet size and mix vehicle routing problem[J]. Computers and Operations Research, 1984, 11(1): 49-66
- [10] Toth P, Vigo D. The vehicle routing problem [M]. Philadelphia: Society for Industrial and Applied thematics philadelphia, 2002
- [11] Schittkat P, Sörensen K, Sevaux M, et al. A matheuristic for the school bus routing problem[R]. Research paper, 2009
- [12] Bodin L D, Berman L. Routing and scheduling of school buses by computer[J]. Transportation Science, 1979, 13(2): 113-129
- [13] Bodin L D, Golden B, Assad A, et al. Routing and scheduling of vehicles and crews: the state of the art[J]. Computers and Operations Research, 1983, 10: 63-211
- [14] Park J, Kim B-I. The school bus routing problem: A review[J]. European Journal of Operational Research, 2010, 202: 311-319
- [15] Li L O Y, Fu Z. The school bus routing problem: a case study [J]. Journal of Operation Research Society, 2002, 53: 552-558
- [16] 郭强, 李育安, 郭耀煌. 社区儿童接送服务车辆的线路优化[J]. 西南交通大学学报, 2006(4): 486-490
- [17] 许文龙, 李小娟, 宫辉力, 等. 校车最优路径规划算法[J]. 地理空间信息, 2011(4): 67-68
- [18] 张苗. 基于双层规划的多目标校车路径优化研究[D]. 成都: 西南交通大学, 2008
- [19] 张富, 朱泰英. 校车站点及线路的优化设计[J]. 数学的实践与认识, 2012(4): 141-146
- [20] 张玉兵, 吴霄翔, 任意. 校车安排问题[J]. 高等数学研究, 2011(1): 122-125
- [21] Fügenschuh A. Solving a school bus scheduling problem with integer programming[J]. European Journal of Operational Research, 2009, 193: 867-884
- [22] Kim B-I, Kim S, Park J. A school bus scheduling problem[J]. European Journal of Operational Research, 2012, 218: 577-585
- [23] Chen D, Kallsn H A, Snider R C. School bus routing and scheduling: an expert system approach[J]. Computers and Industrial Engineering, 1988, 15: 179-183
- [24] Spada M, Bierlaire M, Liebling T M. Decision-aiding methodology for the school bus routing and scheduling problem[J]. Transportation Science, 2005, 39: 477-490
- [25] Desrosiers J, Ferland J A, Rousseau J-M, et al. An overview of a school busing system[M]// Jaiswal N K. Scientific Management of Transport Systems. North-Holland, Amsterdam, 1981: 235-243
- [26] Groër C, Golden B, Wasil E. A library of local search heuristics for the vehicle routing problem[J]. Math. Prog. Comp., 2010, 2: 79-101
- [27] Carlton W B. A tabu search approach to the general vehicle routing problem[D]. Texas: University of Texas at Austin, 1995
- [28] Bent R, Van Hentenryck P. A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows [J]. Computers & Operations Research, 2006, 33: 875-893
- [29] Golden B L, Wasil E A, Kelly J P, et al. The impact of metaheuristics on solving the vehicle routing problem: algorithms, problem sets, and computational results[M]// Crainic T, Laporte G. Fleet management and logistics. Boston, MA: Kluwer, 1998: 33-56
- [30] Li F, Golden B, Wasil E. Very large-scale vehicle routing: New test problems, algorithms, and results[J]. Computers & Operations Research, 2005, 32: 1165-1179
- [31] Park J, Tae H, Kim B-I. Corrigendum to "Post-improvement procedure for the mixed load school bus routing problem[J/OL]. <http://dx.doi.org/10.1016/j.ejor.2012.10.050>, 2012-11-28