求解多车型校车路径问题的混合集合划分的GRASP算法

侯彦娥[1] 孔云峰[2] 党兰学[1]
(河南大学计算机与信息工程学院 河南 开封 475004)[1]
(河南大学黄河中下游数字地理技术教育部重点实验室 河南 开封 475004)[2]

摘 要 针对不同规划场景下具有不同优化目标的多车型校车路径问题(HSBRP),提出一种混合集合划分(SP)的贪婪随机自适应(Greedy Randomized Adaptive Search Procedure,GRASP)算法。根据GRASP算法寻优过程中产生的路径信息构建SP模型,然后使用CPLEX精确优化器对SP模型进行求解。为了适应不同类型的HSBRP问题,改进GRASP的初始解构造函数得到一个可行解,并将其对应的路径放入路径池;在局部搜索过程中应用多种邻域结构和可变邻域下降(VND)来提升解的质量,同时在路径池中记录在搜索过程中得到提升的路径和在每次迭代中得到局部最好解的路径信息。使用基准测试案例进行测试,实验结果表明在GRASP算法中,混合SP能够有效地提高算法的求解性能和稳定性,并且该算法能适应不同优化目标下车型混合和车辆数限制两类HSBRP的求解;与现有算法的比较结果再次验证了所提算法的有效性。

关键词 多车型校车路径问题,集合划分,贪婪随机自适应,混合元启发

中图法分类号 TP301.6 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.04.040

Greedy Randomized Adaptive Search Procedure Algorithm Combining Set Partitioning for Heterogeneous School Bus Routing Problems

HOU Yan-e[1] KONG Yun-feng[2] DANG Lan-xue[1]
(College of Computer and Information Engineering,Henan University,Kaifeng,Henan 475004,China)[1]
(Key Laboratory of Geospatial Technology for the Middle and Lower Yellow River Regions,Ministry of Education,Henan University,Kaifeng,Henan 475004,China)[2]

Abstract In practice of school bus route planning,there are a variety of planning applications with different optimization objectives under the types of school buses constraints. This paper dealt with a class of heterogeneous school bus routing problem(HSBRP) with different objectives. A greedy randomized adaptive search procedure(GRASP) algorithm combining set partition(SP) procedure was proposed in this paper. First,the routes generated in the execution of GRASP are used to build the set partition model,and then the model is solved by the CPLEX optimization software. To keep the algorithm suitable for different HSBRP problems,the initialization solution generation procedure of GRASP is adapted for these problems to obtain a valid solution,and the routes of this initialization solution are put into the route pool. In the local search phase,the many neighborhood operators and variable neighborhood descent procedure are executed for improving the solution. At the same time,the routes of the solution that is improved and the best local optimization in every iteration are both put into the route pool. The test results on the benchmark datasets show that the SP procedure of the proposed algorithm can improve the quality and stability of the algorithm. The proposed algorithm can effectively solve two types of HSBRP with different objectives,and it is effective when compared with the existing HSBRP algorithms.

Keywords Heterogeneous school bus routing problem,Set partitioning,Greedy randomized adaptive search procedure,Hybrid meta-heuristic

1 引言

校车路径规划是一项极其复杂的工作,涉及到学区道路状况、学生、学校以及校车等多种因素。受校车运营公司的资金配置、学生分布状况、道路等级等条件的影响,使用多种类型的校车完成校车路径规划更加符合实际应用的需求。多车

到稿日期:2017-02-15 返修日期:2017-05-26 本文受国家自然科学基金项目(41401461)资助。

侯彦娥(1980-),女,博士,副教授,主要研究方向为智能算法、空间优化,E-mail:houyane@henu.edu.cn;**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;**党兰学**(1980-),男,博士,副教授,主要研究方向为智能算法、空间分析与优化,E-mail:danglx@foxmail.com(通信作者)。

型校车路径问题（Heterogeneous School Bus Routing Problem，HSBRP）是研究如何使用一组校车容量、固定成本和单位可变成本各异的校车完成学生接送工作并达到特定目标的组合优化问题[1]。HSBRP 是校车路径问题（School Bus Routing Problem，SBRP）的一种重要变体，也属于车辆路径问题（Vehicle Routing Problem，VRP）的应用范畴。根据每种类型的校车数目是否受限，HSBRP 分为车型混合和车辆数限制两类。前者侧重于确定校车的车型组合和数量，后者则是在一组车型类型确定、每种类型的校车数量固定的情况下规划路径。在问题的求解难度上，求解车辆数限制 HSBRP 比求解车型混合 HSBRP 更加困难。这两类 HSBRP 均是求解难度极高的 NP-hard 问题[2-3]。

国内外学者针对 HSBRP 开展了相关研究。Ke[4] 使用 Xpress 优化软件精确求解了以总成本为优化目标的 HSBRP 模型，但最大仅能求解包含 20 个站点的 HSBRP 问题。Ripplinger[5] 使用一种两阶段算法求解学生总乘车时间最小的 HSBRP，该算法首先使用扫描法完成解的构造，然后再使用禁忌算法来提升解。文献[6]采用改进的基于位置的构造启发算法来求解以总运营里程为优化目标的 HSBRP 问题。文献[7]研究了农村地区的 HSBRP 问题，提出了一种多阶段的构造启发算法，综合使用学生分配到站点、车辆分配到学校、交换学生等步骤和车型调整策略来完成问题的求解。Park 等[8] 采用一种后启发算法完成不同类型校车的分配，并尽可能地减少校车的使用数量。随着 HSBRP 问题的规模和复杂性的不断增加，元启发算法也逐渐应用于 HSBRP 的求解。文献[9]设计了一种贪婪随机自适应算法（GRASP）来求解总成本最小的车型混合 HSBRP 问题，其根据算法搜索过程中解的质量自适应地选择阈值参数的取值，从而提升解的质量。文献[10]研究了每种车型车辆数目限制的 HSBRP，使用混合变邻域下降搜索的迭代局部搜索算法来规划总成本最小时的校车路径。以上文献中的算法大多考虑的是特定场景的 HSBRP 问题，并且优化目标较为单一。而规划校车路径时，每种类型的校车的数量可能是未知的，需要确定的车型组合和数量亦可能是固定的，校车也可能是购买或者租赁而来的，因此在不同规划场景下优化目标亦不相同。再者，在 VRP 领域涌现出了一些针对一类 VRP 问题的求解算法，比如自适应大规模邻域搜索[11]、迭代局部搜索算法[12-13]、基因算法[14] 等，而在 HSBRP 领域关于此类算法的研究较少。

集合划分模型（Set Partitioning，SP）是 VRP 问题的一种模型表示方法。有些学者[12-13,15-16] 在元启发算法中使用混合 SP 模型的精确求解来提高算法的求解质量。Rochat 等[15] 在求解带容量约束的 VRP 时，在禁忌算法中生成问题的部分解并记录部分解的路径信息，然后构建 SP 模型并精确求解；作者将 SP 过程作为提升算法质量的一种后优化技术。Alvarenga 等[16] 针对以最小车辆数为第一目标的带时间窗的 VRP 问题，提出一种将基因算法和 SP 混合的元启发算法，该算法使用基因算法生成多种路径信息，随后借助 SP 模型精确求解；其实验结果表明了所提算法的有效性。Subramanian 等[12] 将迭代局部搜索算法和 SP 混合用于求解多车型车辆路径问题，在求解过程中仅使用局部发现最好解的路径信息构造 SP 模型；而后作者又将其应用到一类 VRP 问题的求解中[13]，实验结果证明在元启发算法混合 SP 后能够充分利用模型求解的优势，提升算法的求解质量。

基于以上分析，本文针对车型混合和车辆数限制两种规划场景下的 HSBRP 问题，兼顾不同的优化目标，提出了一种混合 SP 的 GRASP 算法（GRASP_SP）进行统一求解。首先，使用 GRASP 算法产生多样性的初始解，借助可变邻域下降[17] 和多个邻域结构对解进行改进，并记录执行过程中产生的路径；然后，基于记录的路径信息构建 SP 模型并进行精确求解；最后，在测试案例上进行实验以验证算法的有效性。

2　算法设计

2.1　算法的基本描述

本文设计的 GRASP_SP 混合元启发算法分为两个阶段：首先使用 GRASP 算法进行寻优，将每次迭代过程中初始解和局部最好解对应的路径，以及在局部搜索过程中改进的路径记录到路径池中；然后，在 GRASP 算法执行完毕后，根据路径池中的路径构造问题的 SP 模型，借助优化器软件进行求解。算法的基本流程如下：

1）初始化最大迭代次数 $MaxIter$、迭代变量 $t=0$、路径池 $RoutePool=\varnothing$、全局最好解 $S^*=\varnothing$、GRASP 算法发现的全局最好解 $S_g=\varnothing$ 以及算法的其他参数。

2）使用初始解构造函数构造初始解 S_0，并将 S_0 对应的路径信息放入 $RoutePool$。

3）应用多种邻域结构对 S_0 进行改进，若解有所改进，则将涉及操作的两条路径信息放入 $RoutePool$。在可变邻域下降搜索结束后得到局部最好解 S_c，将 S_c 对应的所有路径信息放入 $RoutePool$ 中，同时更新 S_g。

4）若 $t>MaxIter$，跳转到步骤（5）；否则，t 加 1，跳转到步骤 2）继续执行。

5）根据 $RoutePool$ 中的路径信息构造 HSBRP 的 SP 模型，并生成相应的模型文件。

6）调用 IBM CPLEX 优化软件提供的编程接口，设置最大执行时间和 tolerances 等参数并对 SP 模型进行求解，得到优化器求解的解 S_p。

7）将 S_g 和 S_p 中的较优者作为问题的最好解 S^*，并返回 S^*。

2.2　初始解的构造

标准的 GRASP 算法在构造初始解时，使用的路径数目是不受限制的。但对于车辆数限制的 HSBRP 问题而言，初始解最多使用的路径数目是当前所有可用车型数量之和。为此，本文对标准的 GRASP 算法的初始解构造方法进行改进，使其能够适应车型混合和车辆数限制这两类 HSBRP 问题的初始解构造。初始解构造的基本流程如下：

1）初始化候选列表（CL），CL 由所有乘车站点组成；并设初始解 S_c 为空。

2）当求解问题的类型为车型混合的 HSBRP 时，直接跳转到步骤 3）。当求解问题的类型为车辆数限制的 HSBRP 时，首先构造 v 条路径 $0-s$，其中 0 代表场站，s 代表学校，v 是当前所有可用车型之和；然后，循环执行步骤 3），直到 v 条

路径全部变为 $0-k-s$ 的形式(k 为站点)时,跳转到步骤 4)。

3)从 CL 中随机选择一个站点 i 插入到路径 $0-s$ 中,得到路径 $0-i-s$,给此路径分配能够满足站点 i 需求的最小车型,并将其放入当前解 S_c 中,然后将 i 从 CL 中移除。

4)计算 CL 中所有站点能够插入到当前解 S_c 中的最小成本。若有节点不能插入到 S_c 中,则设其成本为一个非常大的正数。按照成本值对节点进行升序排列,将插入成本值在 $[C_{\min}, C_{\min}+\alpha(C_{\max}-C_{\min})]$ 之间的站点放入受限候选列表(RCL),其中 $C_{\min}$ 和 $C_{\max}$ 分别表示最小成本和最大成本,$a\in[0,1]$,是阈值参数。

5)从 RCL 中随机选择一个站点 j,寻找其插入成本最小的位置。若能够找到站点 j 插入的最佳位置,则将 j 插入到当前解 S_c 中,并从 CL 中移除 j。对于车型混合 HSBRP 问题,若找不到满足条件的站点 j,则构造一条新路径 $0-j-s$,并为新路径分配能够满足站点 j 需求的最小车型,将新路径放入当前解 S_c 中,并且从 CL 中移除 j。对于车辆数限制 HSBRP 问题,直接跳转至步骤 1)重新执行。

6)若 CL 为空,返回构造解 S_c;否则,根据求解问题的类型决定跳转步骤。对于车型混合 HSBRP 问题,跳转到步骤 3)继续执行;对于车辆数限制 HSBRP 问题,跳转到步骤 4)继续执行。

2.3 局部搜索

局部搜索通过可变邻域下降[17]而完成,同时借助 5 个路径间邻域结构和 1 个路径内邻域结构完成搜索。受问题约束的限制,路径间邻域结构仅能操作学生乘车站点,并且约定路径内邻域结构仅在路径间操作的路径有改进时才能使用。这 6 个邻域结构定义如下:

1)shift(1,0):将一条路径 r_1 上的一个站点 i 移动到另外一条路径 r_2 的站点 k 之后,要求 k 不能是学校站点。

2)shift(2,0):将一条路径 r_1 上相邻的两个站点即 $i-j$ 移动到另外一条路径的 r_2 站点 k 之后,k 不能是学校站点。

3)swap(1,1):将路径 r_1 上的站点 i 和路径 r_2 上的站点 j 进行交换,交换完成后得到两条新路径。

4)swap(1,2):将路径 r_1 上的站点 i 和路径 r_2 上的一条边 $u-v$ 进行交换。

5)swap(2,2):将路径 r_1 上的一条边 $i-j$ 和路径 r_2 上的一条边 $u-v$ 进行交换。

6)2-opt:在同一条路径上移除两条不相邻的边,然后再添加两条新边,构成一条新的路径。当两条不相邻的边中分别包含场站或者学校站点时,直接对路径中的学生站点进行逆序得到一条新路径。

2.4 路径池的生成和更新

路径池是一组路径的集合,它是构造集合划分模型的基础。算法开始时路径池为空,随着算法的执行,将逐渐产生的路径信息加入到路径池内。添加到路径池的路径包括初始解对应的路径信息、进行邻域搜索时得到提升的路径信息以及每次具有局部最好解的路径信息。算法执行完毕时,再将路径池中的所有路径信息以特定的格式写入到文件中。

路径池中的每一条路径均由场站、一个或多个学生乘车站点和学校站点组成。此外,每条路径还有其使用的车型、负载、路径长度、行驶时间、路径成本等附属信息。使用面向对象的技术将路径池中的路径抽象成一个类 *RTour*,该类包含路径编号、车型编号、路径负载、路径长度、总成本、路径站点序列等属性。每一条路径都是 *RTour* 类的一个对象。路径池 *RoutePool* 则表示 *RTour* 对象的集合,针对 *RoutePool* 能够实现 *RTour* 的遍历和更新。

路径池在算法初始化时被创建,在算法执行过程中进行更新。路径池更新的一般步骤如下:

1)将要添加到路径池中的路径转化为 *RTour* 类的对象。创建一个 *RTour* 类的对象 rt,使用当前路径的信息对该对象 rt 的属性进行赋值。

2)判定对象 rt 在路径池中是否已经存在。两个 *RTour* 对象只有当其车型、实际负载、路径长度、总成本以及路径上的节点序列全部相同时,才被认为是相同的对象。

3)若 rt 在路径池中不存在,则将 rt 放入到路径池中;否则不进行任何操作。

该路径池更新步骤将在算法执行过程中被反复调用,直到算法循环结束。最终得到用于构造 SP 模型的所有路径信息。

2.5 SP 模型的创建和求解

假设 R 是 HSBRP 问题中所有车型对应的所有可能路径的集合,C 是所有乘车站点的集合,M 是由车型容量和成本各异的校车类型组成的集合。对于车型为 $k(k\in M)$ 的校车,设其容量为 Q_k,固定成本为 f_k,单位可变成本为 v_k,最大数量为 n_k。由此可定义:$R_j\subseteq R$ 是包含乘车站点 $j(j\in C)$ 的所有路径的集合,$R_k\subseteq R$ 是使用车型 k 的所有路径的集合($k\in M$)。c_r 表示 R 中路径 r 的总成本,表示为 $c_r=f_k+d_r*v_k$,其中 d_r 是路径 r 的长度,k 为路径 r 使用的车型。定义决策变量 x_r,当路径 r 包含在问题的解中时 $x_r=1$,否则 $x_r=0$。因此,HSBRP 的 SP 模型定义如下:

$$\min \sum_{r\in R} c_r x_r \tag{1}$$

$$\text{s.t.} \sum_{r\in R_j} x_r=1, j\in C \tag{2}$$

$$\sum_{r\in R_k} x_r\leqslant n_k, k\in M \tag{3}$$

$$x_r\in\{0,1\} \tag{4}$$

目标函数(1)是最小化路径的总成本。在默认情况下,优化目标包含固定成本和可变成本。当 $f_k=0$ 时,优化目标仅包含可变成本;当 $f_k\neq 0$ 且 $v_k=1$ 时,优化目标为固定成本。约束(2)表示 C 中的每一个乘车站点必须包含在唯一一条路径上。约束(3)表示每种类型的校车的可用车辆数目不能超过 n_k,此约束条件仅对车辆数限制的 HSBRP 有效;约束(4)限定了决策变量 x_r 的取值。

根据 HSBRP 的 SP 模型定义和路径池 *RoutePool* 中的路径信息,创建 CPLEX 支持的.*lp* 格式的模型文件。此外,为了便于在进行模型求解后能够快速得到解的路径信息,还需要将路径池 *RoutePool* 中的每条路径以特定格式写入到文本文件中。一个案例对应一个存放所有路径信息的文本文件,一条路径对应文本文件中的一行记录。

求解 SP 模型时,在算法中调用 CPLEX ILOG 提供的

ILOG. Conert. dll和 ILOG. CPLEX. dll 中的类来完成案例模型文件的加载、参数设置和问题求解等基本操作。精确求解模型后，可以得到解的目标值和决策变量值等于 1 的路径编号的集合。若要输出通过模型求解得到的路径信息，可以查找案例文件对应的路径文件，在路径文件内找到对应路径编号的记录，取出其对应的站点序列及其他附属信息，然后将这些信息组合。

3　实验与分析

GRASP_SP 算法使用 Visual Studio 2010 环境中的 C# 实现，测试环境为 PC 机，配置为 Intel® core i7-4790 3.60 GHz、8 GB 内存、Windows 7 64 位操作系统。算法的参数设置如下：迭代次数为 100，邻域大小为 min{30，n}，参数 a 的取值与文献[9]相同。SP 模型调用 ILOG CPLEX 12.6 优化软件提供的 C# 编程接口进行求解。CPLEX 优化器的参数设置为 $mipgap=10^{-6}$，运算时间限定为 60 s，其余参数设定为默认值。测试案例取自文献[9-10]设计的 HSBRP 测试案例，问题规模为 17～75，车型有 2～3 种；案例包含乘车站点和学校集中分布的 C01－C06 以及乘车站点和学校随机分布的 R01－R06。约定每个案例运行 10 次。

3.1　实验结果

为了说明算法混合集合划分(SP)过程后的优势，在本文设计的算法框架内分别运行不带 SP 过程的 GRASP 算法和带 SP 过程的 GRASP_SP 算法，以测试这两种算法在车型混合和车辆数限制这两种 HSBRP 问题上的运行效果。

针对车型混合 HSBRP 问题，在校车可变成本按距离(即每英里)取值和不同优化目标的情况下，使用文献[9]设计的测试案例对 GRASP 算法和 GRASP_SP 算法进行比较。表 1 给出了 GRASP 和 GRASP_SP 算法在优化目标为总成本、固定成本和可变成本 3 种情况下的运算结果。其中，C_1 和 C_2 分别表示 GRASP 和 GRASP_SP 算法发现的最好解；T_1 和 T_2 代表两种算法相应的平均运行时间；gz，gf 和 gv 分别表示 GRASP_SP 算法相对于 GRASP 算法在 3 种优化目标下的提升程度(%)。

表 1　车型混合 HSBRP 3 种优化目标下两种算法的比较

Table 1　Comparison of two algorithms for fleet and mix HSBRP with three different optimization objectives

案例	总成本					固定成本					可变成本				
	C_1	C_2	T_1/s	T_2/s	gz/%	C_1	C_2	T_1/s	T_2/s	gf/%	C_1	C_2	T_1/s	T_2/s	gv/%
C01	41244.50	39704.29	5.05	81.25	3.73	40505.07	39478.89	4.91	81.62	2.53	756.90	697.15	5.29	23.60	7.89
C02	32348.43	31123.66	1.25	4.01	3.79	32222.34	30997.11	1.23	3.83	3.80	435.87	422.69	1.23	2.27	3.02
C03	21772.12	21268.22	0.83	1.26	2.31	21188.84	21188.68	0.80	1.26	0.00	368.22	362.94	0.81	1.61	1.43
C04	17882.99	17582.47	0.43	0.82	1.68	17520.16	17516.58	0.43	0.79	0.02	277.99	277.21	0.41	0.67	0.28
C05	59738.32	56751.53	6.26	51.71	5.00	60047.66	56527.92	6.40	51.64	5.86	730.53	691.02	6.07	24.82	5.41
C06	18776.70	18776.70	0.26	0.31	0.00	18704.37	18704.37	0.26	0.32	0.00	270.90	270.90	0.23	0.28	0.00
R01	25018.54	21998.65	1.60	10.48	12.07	24737.89	21892.57	1.52	11.34	11.50	388.87	376.57	1.62	5.84	3.16
R02	32097.12	30202.44	1.92	6.10	5.90	32017.27	30113.72	2.00	8.38	5.95	263.31	261.01	1.77	3.58	0.87
R03	41651.77	41618.89	2.88	78.30	0.08	42909.22	41376.46	2.81	86.94	3.57	637.14	617.00	3.73	17.38	3.16
R04	26846.46	26028.24	1.01	2.02	3.05	26375.27	25961.19	1.01	2.19	1.57	331.24	326.78	1.05	1.73	1.35
R05	31522.37	29806.23	1.98	6.81	5.44	31422.89	29698.02	1.93	6.55	5.49	416.15	395.50	1.99	5.74	4.96
R06	28583.40	27373.75	2.19	70.25	4.23	28507.06	27266.76	2.12	70.75	4.35	368.52	366.25	2.09	8.76	0.62
平均	31456.89	30186.26	2.14	26.11	3.94	31346.50	30060.21	2.12	27.13	3.72	437.14	422.09	2.19	8.02	2.68

由表 1 可知，GRASP_SP 算法在总成本、固定成本和可变成本 3 种优化目标下均优于 GRASP 算法。与 GRASP 算法相比，GRASP_SP 在 3 种优化目标下分别平均提升了 3.94%，3.72% 和 2.68%；单个案例分别最大提升了 12.07%，11.50% 和 7.89%。因为混合了 SP 模型的求解过程，所以 GRASP_SP 算法相比 GRASP 算法花费了更多的运算时间。

表 2 给出了在 3 种不同优化目标下 GRASP 和 GRASP_SP 算法在每个案例上求解的标准差系数。其中，D_1 和 D_2 分别表示 GRASP 和 GRASP_SP 针对案例运行 10 次的标准差系数；标准差系数为 0 表示算法运行 10 次的结果相同，用黑体表示。

由表 2 可知，GRASP_SP 算法在优化总成本、固定成本和可变成本 3 种优化目标下比 GRASP 算法更加稳定。在 3 种优化目标下，GRASP_SP 算法在所有案例上的平均标准差系数分别为 0.27%，0.29% 和 0.36%，并且部分案例的标准差系数为 0。这表明混合了 SP 精确求解过程的 GRASP_SP 算法能够充分利用精确求解的优势，提升算法的稳定性。

表 2　针对车型混合 HSBRP 问题两种算法在 3 种优化目标下的稳定性比较

Table 2　Stability comparison of two algorithms for fleet and mix HSBRP with three different optimization objectives

(单位：%)

案例	总成本		固定成本		可变成本	
	D_1	D_2	D_1	D_2	D_1	D_2
C01	1.39	**0**	1.91	**0**	1.72	**0**
C02	1.51	**0**	1.78	**0**	1.66	**0**
C03	2.00	1.71	1.04	1.72	1.73	0.74
C04	1.87	**0**	1.18	0.63	3.59	2.58
C05	2.10	0.36	1.59	0.24	1.85	0.09
C06	3.51	**0**	3.71	0.01	0.64	0.78
R01	4.77	0.11	2.00	0.17	3.21	**0**
R02	2.91	**0**	2.80	**0**	2.39	**0**
R03	2.10	1.09	1.54	0.75	2.84	0.08
R04	1.33	**0**	2.78	**0**	1.32	**0**
R05	3.52	**0**	2.75	**0**	1.64	0.06
R06	1.66	0.02	1.20	0.01	3.16	**0**
平均	2.39	0.27	2.02	0.29	2.15	0.36

针对车辆数限制的 HSBRP 问题，分别使用 GRASP 和 GRASP_SP 算法求解文献[10]设计的测试案例。测试时考

虑总成本和可变成本两种优化目标，统计两种算法得到的最好解、平均运行时间和标准差系数，结果如表 3 所列。其中 C_1 和 C_2、T_1 和 T_2 的含义同表 1，d_1 和 d_2 分别表示 GRASP 和 GRASP_SP 算法的标准差系数，g_1 和 g_2 分别表示在优化目标为总成本和可变成本时 GRASP_SP 算法相对于 GRASP 算法的改进程度。

表 3 对于车辆数限制的 HSBRP 问题两种算法在两种优化目标下的比较

Table 3 Comparison of two algorithms for fixed HSBRP with two different optimization objectives

案例	总成本							可变成本						
	C_1	C_2	T_1/s	T_2/s	d_1/%	d_2/%	g_1/%	C_1	C_2	T_1/s	T_2/s	d_1/%	d_2/%	g_2/%
C01	41759.49	39727.96	47.22	120.41	3.65	0.01	4.86	746.88	701.75	35.55	53.13	14.71	0.01	6.04
C02	32534.85	32533.43	7.35	10.24	4.23	**0**	0	452.86	433.43	5.20	6.17	3.36	**0**	4.29
C03	22275.18	22272.00	2.88	9.24	1.57	0.03	0.01	373.65	372.00	2.01	3.04	2.98	**0**	0.44
C04	18297.01	18297.01	6.01	4.84	0	**0**	0	297.01	297.01	3.05	3.64	0.69	0.21	0
C05	60256.86	56751.53	55.16	120.09	4.51	0.24	5.82	750.08	700.49	35.08	65.67	4.77	0.01	6.61
C06	18776.70	18776.70	0.49	0.35	7.32	**0**	0	278.73	275.06	0.28	0.43	1.77	2.31	1.32
R01	25928.46	23407.59	1.91	6.47	3.63	1.90	9.72	427.39	394.91	1.38	12.21	4.46	0.12	7.60
R02	32677.09	30299.61	3.65	4.82	7.44	**0**	7.28	271.92	264.56	2.17	6.33	2.52	**0**	2.71
R03	43640.87	43127.43	5.15	78.73	4.87	0.01	1.18	651.42	627.12	3.89	27.02	5.04	0.01	3.73
R04	28238.49	26432.46	7.18	7.03	4.94	0.03	6.40	341.33	327.90	4.43	6.25	2.01	0.26	3.93
R05	33817.14	31198.45	10.38	59.89	6.73	0.02	7.74	408.19	398.45	9.05	10.29	6.52	0.18	2.39
R06	28786.83	28082.87	6.76	35.18	3.42	**0**	2.45	435.53	377.71	4.07	8.64	1.58	0.43	13.28
平均	32249.08	30908.92	12.85	38.11	4.36	0.19	3.79	452.92	430.87	8.85	16.90	4.20	0.29	4.36

由表 3 可知，针对车辆数限制的 HSBRP 问题的总成本和可变成本两种优化目标，GRASP_SP 算法仍然优于 GRASP 算法。在求解质量上，GRASP_SP 算法在总成本和可变成本两种优化目标下比 GRASP 算法分别平均提升了 3.79%和 4.36%；单个案例的最大提升分别为 9.72%和 13.28%。在算法稳定性方面，GRASP_SP 算法的平均标准差系数比 GRASP 算法更小，平均仅为 0.19%和 0.29%；并且，部分案例的标准差系数达到了 0(用黑体表示)，这说明 GRASP_SP 算法的稳定性更好。由于 SP 模型的创建和求解，因此 GRASP_SP 算法的运算时间比 GRASP 算法有所增加。但 GRASP_SP 算法在求解质量和稳定性方面有较大的提升。

综合表 1－表 3 的实验结果可知：1)混合了 SP 过程之后，GRASP_SP 算法的求解质量和稳定性有较大的提升。这是因为 GRASP_SP 能够从全局的角度收集搜索过程中产生的路径信息；再者，SP 模型的求解借助精确优化器完成，能够充分利用精确求解的优势，提升了算法的求解质量和稳定性。在 GRASP_SP 中混合 SP 过程后使得其整体运算时间有所增加，但亦能在合理的时间内完成模型的求解。2)GRASP_SP 算法能够适应车型混合和车辆数限制这两类 HSBRP 问题以及不同优化目标问题的求解，并且算法的求解性能和稳定性较好。

3.2 算法比较

为了验证 GRASP_SP 算法的优势，在车型混合和车辆数限制两类 HSBRP 问题上分别将其与现有的 HSBRP 求解算法进行比较。

针对车型混合 HSBRP 问题，将 GRASP_SP 算法与文献[9]提出的参数自适应选择的 SGRASP 算法进行比较。由于文献[9]中每种类型的校车可变成本按时间取值并且优化目标为总成本，因此此处 GRASP_SP 算法仅优化总成本目标。GRASP_SP 算法和 SGRASP 算法的比较结果如表 4 所列。其中，*Best* 和 *Avg* 分别代表案例获得的最好解和平均解；*Time* 表示案例运行 10 次的平均运行时间；*Dev* 表示标准差系数；gap_1 和 gap_2 表示 GRASP_SP 算法相对于 GRASP 算法的最好解和平均解的提升程度；SGRASP 算法的结果取自文献[9]。

表 4 GRASP_SP 算法与 SGRASP 算法的比较

Table 4 Comparison of GRASP_SP algorithm and SGRASP algorithm

案例	SGRASP			GRASP_SP				改进程度(%)	
	Best	*Avg*	*Time*/s	*Best*	*Avg*	*Dev*/%	*Time*/s	gap_1	gap_2
C01	42189.19	42215.18	27.58	41108.59	41117.61	0.01	79.95	2.56	2.60
C02	31975.16	32044.06	7.06	31970.41	31970.41	**0**	2.79	0.01	0.23
C03	22012.48	22457.89	4.49	22004.27	22185.94	1.67	1.27	0.04	1.21
C04	18147.20	18166.60	2.55	18147.14	18147.14	**0**	0.85	0	0.11
C05	59632.50	60486.33	32.55	58253.61	58352.01	0.26	50.10	2.31	3.53
C06	19329.76	19332.05	1.42	19329.76	19330.26	0.36	0.31	0	0.01
R01	22884.98	23964.69	7.99	22795.53	22795.53	**0**	9.37	0.39	4.88
R02	31269.18	31732.41	8.01	30806.71	30806.71	**0**	5.05	1.48	2.92
R03	42929.33	43658.14	13.98	42867.90	43220.21	1.40	77.58	0.14	1.00
R04	26692.28	27044.31	6.19	26684.23	26684.61	0.01	1.86	0.03	1.33
R05	30641.66	31374.29	9.20	30618.08	30618.31	**0**	5.83	0.08	2.41
R06	28280.67	29072.89	9.96	28120.54	28143.81	0.08	69.70	0.57	3.20
平均	31332.03	31795.74	10.92	31058.90	31114.38	0.32	25.39	0.63	1.95

表 4 的结果表明 GRASP_SP 算法优于 SGRASP 算法，能够获得更好的平均解和最好解。与 SGRASP 算法相比，GRASP_SP 算法在最好解和平均解上平均提升了 0.63%和 1.95%；并且算法的稳定性较好，平均标准差系

数较小（仅为0.32%），案例 C02，C04，R01，R02 和 R05 的标准差系数达到 0。

对于车辆数限制的 HSBRP 问题，使用 GRASP_SP 算法进行求解，并与文献[10]提出的 HILS 算法进行比较。在优化目标为总成本且每辆校车的可变成本按距离取值时，两种算法的比较结果如表 5 所列。其中，$Best$，Avg，Dev，$Time$ 的含义同表 1；gap_1 和 gap_2 分别代表 GRASP_SP 算法相对于 HILS 算法在最好解和平均解上的改进程度。

从表 5 可以看出，GRASP_SP 算法在求解质量和稳定性方面均优于文献[10]的 HILS 算法。相对于 HILS 算法，GRASP_SP 算法在最好解和平均解上分别平均提升了 1.15%和 1.07%；而且针对单个案例 C01，R01 和 R02，算法提升较多，最好解和平均解最多分别提升了 6.10% 和 5.79%。在算法稳定性方面，GRASP_SP 算法在整体上拥有较小的标准差系数，平均标准差系数仅为 0.19%；除案例 R01 外，其余案例的标准差系数均比较小，这也表明 GRASP_SP 算法能在大部分案例上取得较小的标准差系数，算法的整体稳定性较高。

表 5　GRASP_SP 算法与 HILS 算法的比较

Table 5　Comparison of GRASP_SP algorithm and HILS algorithm

案例	HILS				GRASP_SP				改进程度/%	
	$Best$	Avg	Dev/%	$Time$/s	$Best$	Avg	Dev/%	$Time$/s	gap_1	gap_2
C01	40232.22	40622.15	0.98	13.21	39727.96	39730.22	0.01	120.41	1.25	2.20
C02	32534.25	32537.39	0.01	1.62	32533.43	32533.44	0	10.24	0	0.01
C03	22272.97	22276.01	0.02	1.05	22272.00	22273.15	0.03	9.24	0	0.01
C04	18297.01	18297.90	0.02	1.88	18297.01	18297.01	0	4.84	0	0
C05	57235.62	57797.38	1.41	17.19	56751.53	57165.19	0.24	120.09	0.85	1.09
C06	18776.70	18777.96	0.01	0.19	18776.70	18776.70	0	0.35	0	0.01
R01	23427.37	23513.21	0.49	1.39	23407.59	23300.78	1.90	6.47	6.10	0.90
R02	32084.35	32161.79	0.72	1.43	30299.61	30300.84	0.002	4.82	5.56	5.79
R03	43138.28	43147.55	0.01	3.27	43127.43	43129.16	0.01	78.73	0.03	0.04
R04	26432.51	26713.57	0.59	2.61	26432.46	26439.10	0.03	7.03	0.00	1.03
R05	31209.32	31440.92	0.78	2.99	31198.45	31199.31	0.02	59.89	0.03	0.77
R06	28085.49	28376.32	1.94	2.49	28082.87	28083.82	0.001	35.18	0.01	1.03
平均	31143.84	31305.18	0.58	4.11	30908.92	30935.73	0.19	38.11	1.15	1.07

表 4 和表 5 的实验结果表明，针对车型混合和车辆数限制两类 HSBRP 问题，GRASP_SP 算法表现出比现有 HSBRP 算法更好的求解性能和稳定性；尽管 GRASP_SP 算法的运算时间有所增加，但其性能和稳定性得到了提升。

3.3　路径池构造方式对算法的影响

在路径池的构造过程中，通常将局部最好解对应的路径信息[13]或局部搜索过程中产生的路径信息放入到路径池中[15]。为了保证通过求解 SP 模型能够得到可行的解，有时还需要将初始解对应的路径信息放入到路径池中。由于 SP 模型的创建依赖于路径池中的路径信息，因此路径池的构造方式会对 SP 模型的求解产生影响。

在 GRASP_SP 算法的执行过程中，初始解、局部搜索（LS）和局部最好解都有对应的路径信息；并且，在局部搜索过程中仅记录路径目标值有改进的路径。根据路径池中路径信息的来源，路径可能来自初始解、LS 路径和局部最好解三者之一或者它们的组合。由此可以得到路径池构造的 7 种方式，排除掉仅使用初始解构造路径的无效方式以及有可能产生不可行解的 LS 路径方式，最终得到 5 种有效方式：

1）局部最好解：仅包含每次迭代过程中产生的局部最好解对应的路径；

2）初始解和 LS 路径：组合初始解和局部搜索过程中产生的路径；

3）初始解和局部最好解：组合初始解和每次迭代局部最好解对应的路径；

4）LS 路径和局部最好解：组合局部搜索和每次迭代局部最好解对应的路径；

5）初始解、LS 路径和局部最好解：组合初始解、LS 和每次迭代局部最好解对应的路径。这也是本文采用的方式。

接下来主要分析 5 种路径池构造方式对 GRASP_SP 算法的求解质量和运算时间的影响。在相同的测试条件下应用不同的路径池构造方式时，使用 GRASP_SP 算法求解车型混合和车辆数限制两种 HSBRP 问题的多种优化目标。算法的优化性能、SP 模型的大小及其平均执行时间如表 6 所列。其中，fg_1，fg_2，fg_3 是 GRASP_SP 算法在求解车型混合 HSBRP 问题时相对于文献[9]在总成本、固定成本和可变成本 3 个优化目标上的平均提升程度；hg_1 和 hg_2 代表 GRASP_SP 算法求解车辆数限制 HSBRP 问题时相对于文献[10]在总成本和可变成本两个优化目标上的平均提升程度；SPf_v，SP_f 和 SP_v 表示优化目标为总成本、固定成本和可变成本时 SP 模型的平均大小（SP 模型的大小由路径池中的路径条数衡量，保留到整数条路径）；T_{fv}，T_f 和 T_v 代表优化目标为总成本、固定成本和可变成本时算法的平均执行时间。

根据表 6 的实验结果可以得出以下结论：1）当路径池中包含初始解、LS 路径和局部最好解等对应的路径信息时，GRASP_SP 算法的求解性能最好。2）在路径池构造过程中加入 LS 路径信息能够提升算法的求解质量；在 3 种包含 LS 路径信息的路径池构造方式中，GRASP_SP 算法在求解车型混合和车辆数限制 HSBRP 两类问题上均表现出良好的求解性能。这是因为在局部搜索过程中将质量较好的路径放入到路径池中能够提升路径池中路径的质量；再者，在局部搜索过程中产生了大量的路径，增加了路径池中路径的多样性。3）针对 HSBRP 问题，在路径池中仅使用局部最好解对应的路径信息并不能有效地提高 GRASP_SP 算法的求解质量。4）针对同一类 HSBRP 问题，在路径池中放入的路径数目越多，

SP 模型越大，其运算时间也越长。车辆数限制 HSBRP 比车型混合 HSBRP 多了一个每种车型的最大车辆数数目约束，尽管其 SP 模型比具有相同优化目标的车型混合 HSBRP 模型的小，但其问题求解比较复杂，因此仍需要较多的运算时间。

表 6 采用不同的路径池构造方式时求解两类 HSBRP 问题的算法性能、SP 模型大小和平均执行时间的比较

Table 6 Comparisons of performance, SP model size and mean execution time of GRASP_SP algorithm with different construction methods of route pool for two kinds of heterogeneous school bus routing problem

路径池的构成方式	车型混合 HSBRP									车辆数限制 HSBRP					
	fg_1	fg_2	fg_3	SP_{fv}	SP_f	SP_v	T_{fv}	T_f	T_v	hg_1	hg_2	SP_{fv}	SP_v	T_{fv}	T_v
局部最好解	0.092	0.201	0.952	445	446	411	3.09	5.14	2.65	0.018	0.713	540	521	14.72	11.57
初始解和 LS 路径	0.218	0.558	1.093	10319	10380	9652	24.28	25.08	7.26	0.649	0.723	8233	7718	27.99	14.75
初始解和局部最好解	0.093	0.558	1.091	995	997	971	6.03	7.77	2.81	0.019	0.733	911	889	16.69	12.82
LS 和局部最好解	0.220	0.559	1.093	10035	10125	9398	26.36	26.93	7.97	0.650	0.759	8055	7514	28.08	14.02
初始解、LS 路径和局部最好解	0.221	0.560	1.099	10395	10423	9708	27.21	27.70	8.13	1.153	0.759	8412	7803	31.27	14.75

结束语 针对不同应用场景和规划目标的多车型校车路径问题，本文设计了一种贪婪随机自适应(GRASP)算法和集合划分(SP)混合的元启发算法(GRASP_SP)进行统一求解。首先利用 GRASP 算法在寻优过程中产生的路径信息构造问题的集合划分模型，然后再使用精确优化器求解创建的 SP 模型。对于车型混合和车辆数限制两类多车型校车路径问题，GRASP_SP 算法能够获得较好的求解性能，并且其稳定性较好。实验结果表明，在 GRASP 元启发算法中混合集合划分过程能够在适当增加运算时间的基础上提高算法的求解质量和稳定性。通过测试不同的路径池构造方式，从所得结果发现在路径池中混合局部搜索产生的路径能够提高算法的求解质量；此外，SP 模型的大小取决于路径池中的路径数目，但其求解难度还与问题类型相关。下一步计划将此算法应用到涵盖单校和多校、单车型和多车型等多种约束的 SBRP 问题求解中。

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] PARK J, KIM B I. The school bus routing problem: A review [J]. European Journal of Operational Research, 2010, 202(2): 311-319.

[3] DANG L X, CHEN X P, KONG Y F. Review of School Bus Routing Problem: Concept, Model and Optimization Algorithms [J]. Journal of Henan University (Natural Science), 2013, 43(6): 682-691. (in Chinese)
党兰学，陈小潘，孔云峰. 校车路径问题模型及算法研究进展[J]. 河南大学学报(自然科学版)，2013，43(6)：682-691.

[4] KE X. School bus selection, routing and Scheduling [D]. Canada, Windsor: University of Windsor, 2005.

[5] RIPPLINGER D. Rural school vehicle routing problem [J]. Transportation Research Record: Journal of the Transportation Research Board, 2005, 1992(1): 105-110.

[6] DE SOUZA L, SIQUEIRA P H. Heuristic Methods Applied to the Optimization School Bus Transportation Routes-A Real Case [C]// 23rd International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems. 2010: 247-256.

[7] THANGIAH S R, FERGANY A, WILSON B, et al. School Bus Routing in Rural School Districts[C]// Proceedings of the 9th International Conference on Computer-Aided Scheduling of Public Transport. Spring, 2008: 209-232.

[8] 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(1): 204-213.

[9] HOU Y E, DANG L X, KONG Y F, et al. GRASP Algorithm with Parameter Selection Mechanism for Heterogeneous Fleet School Bus Routing Problem [J]. Computer Science, 2016, 43(8): 233-239. (in Chinese)
侯彦娥，党兰学，孔云峰，等. 求解多车型校车路径问题的带参数选择机制的 GRASP 算法[J]. 计算机科学，2016，43(8)：233-239.

[10] HOU Y E, KONG Y F, DANG L X, et al. Model and Algorithm for Heterogeneous Fixed Fleet School Bus Routing Problem [J]. Computer Science, 2016, 43(12): 234-240. (in Chinese)
侯彦娥，孔云峰，党兰学，等. 车辆数限制的多车型校车路径问题模型及算法研究[J]. 计算机科学，2016，43(12)：234-240.

[11] PISINGER D, ROPKE S. A general heuristic for vehicle routing problems[J]. Computer & Operations Research, 2007, 34(8): 2403-2435.

[12] SUBRAMANIAN A, PENNA P H V, UCHOA E, et al. A hybrid algorithm for the Heterogeneous Fleet Vehicle Routing Problem [J]. European Journal of Operational Research, 2012, 221: 285-295.

[13] SUBRAMANIAN A, UCHOA E, OCHI L. A hybrid algorithm for a class of vehicle routing problems [J]. European Journal of Operational Research, 2013, 40: 2519-2531.

[14] VIDAL T, CRAINIC T, GENDREAU M, et al. A unified solution framework for multi-attribute vehicle routing problems [J]. European Journal of Operational Research, 2014, 234: 658-673.

[15] ROCHAT Y, TAILLARD E D. Probabilistic diversification and intensification in local search for vehicle routing [J]. Journal of Heuristics, 1995, 1: 147-167.

[16] ALVARENGA G, MATEUS G, DE TOMI G. A genetic and set partitioning two-phase approach for the vehicle routing problem with time windows [J]. Computers and Operations Research, 2007, 34(6): 1561-1584.

[17] HANSEN P, MLADENOVIC N. Variable Neighborhood Search: Principles and Applications [J]. European Journal of Operations Research, 2001, 130(3): 449-467.