

# 车辆数限制的多车型校车路径问题模型及算法研究

侯彦娥<sup>1,2</sup> 孔云峰<sup>1</sup> 党兰学<sup>2</sup> 谢毅<sup>1,2</sup>

(河南大学黄河中下游数字地理技术教育部重点实验室 开封 475004)<sup>1</sup>

(河南大学计算机与信息工程学院 开封 475004)<sup>2</sup>

**摘要** 为适应校车路径规划中校车有多种车型且每种车型数量受限的需求,建立车辆数限制的多车型校车路径问题(HFSBRP)的数学模型,并提出一种迭代局部搜索算法进行求解。该算法借助邻域随机选择的变邻域下降搜索(VND)算法完成局部提升。局部提升过程中,首先调整车型,然后再混合使用缩减路径数和提高车辆利用率的邻域解接受策略以提高算法的寻优能力,为保证解的多样性,允许接受一定偏差范围内的邻域解。此外,为避免算法过早陷入局部最优,设计了多点交换和移动的扰动规则。基于国际基准测试案例进行模型验证和算法测试,实验结果表明了模型的正确性和算法的有效性。

**关键词** 多车型校车路径问题,车辆数限制,迭代局部搜索,随机邻域选择

**中图分类号** TP301.6,U492.4 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.12.043

## Model and Algorithm for Heterogeneous Fixed Fleet School Bus Routing Problem

HOU Yan-e<sup>1,2</sup> KONG Yun-feng<sup>1</sup> DANG Lan-xue<sup>2</sup> XIE Yi<sup>1,2</sup>

(Key Laboratory of Geospatial Technology for the Middle and Lower Yellow River Regions, Ministry of Education,

Henan University, Kaifeng 475004, China)<sup>1</sup>

(College of Computer and Information Engineering, Henan University, Kaifeng 475004, China)<sup>2</sup>

**Abstract** In practice of school bus route planning, the bus fleet usually consists of a limited number of buses with different capacities, purchase costs and operation costs. However, the heterogeneous fixed fleet school bus routing problem (HFSBRP) has not been well investigated. In this paper, we introduced a mathematical model for HFSBRP and also proposed an iterated local search (ILS) algorithm to optimize the total cost. The ILS is combined with a variable neighborhood descent (VND) algorithm with random neighborhood selection. In local search, the bus type for one or more routes will be adjusted to reduce the costs. Two acceptance rules are used to accept the solution while satisfying the rules. In addition, some worst solutions within the scope of cost deviation are accepted to keep the diversification of the search. Moreover, a perturbation mechanism with multiple points swap or shift is used to avoid local optima. The experimental results demonstrate the correctness and effectiveness of the proposed model.

**Keywords** Heterogeneous school bus routing problem, Fixed fleet, Iterated local search, Random variable neighborhood search

## 1 引言

随着我国义务教育的不断发展,为中小學生提供校车服务成为学校和教育主管部门的一项新要求。如何合理规划校车路径并尽可能地降低运营成本是地方教育主管部门面临的一个难题。与规划校车路径紧密相关的校车路径问题(School Bus Routing Problem, SBRP)即是在满足既定约束条件下,合理地规划校车线路将学生从乘车站点送到学校(或从学校送回乘车站点),并达到特定优化目标的组合优化问题<sup>[1]</sup>。SBRP属于车辆路径问题(Vehicle Routing Problem, VRP)的应用范畴<sup>[1]</sup>,因增加了与学生相关的约束条件(比如

最大乘车时间、学校时间窗等),使SBRP求解比经典VRP更加复杂。与VRP一样,SBRP也是一类求解难度极高的NP-hard问题<sup>[2,3]</sup>。SBRP自提出后,国内外学者对其进行了大量的研究,大部分集中在单一车型<sup>[4-7]</sup>。然而在实际的服务管理中,受地理位置、道路等级、学生分布情况等因素的影响,提供校车服务的车队通常由一组不同类型的车辆组成,这些车辆的装载容量、购置成本和单位行驶成本各不相同。因此,研究使用多种车型的校车路径问题更符合实际应用需求。

多车型校车路径问题(Heterogeneous School Bus Routing Problem, HSBRP)是SBRP的一种变体,与单一车型的SBRP的不同之处在于每种类型的校车拥有不同的容量、最

到稿日期:2015-10-21 返修日期:2016-02-05 本文受国家自然科学基金项目:大规模混载校车路径问题多目标优化算法研究(41401461),河南省教育厅自然科学重点项目(15A520009)资助。

侯彦娥(1980-),女,博士生,讲师,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,E-mail:yfkong@henu.edu.cn(通信作者);党兰学(1980-),男,博士,讲师,主要研究方向为智能算法、空间分析与优化;谢毅(1983-),男,博士生,讲师,主要研究方向为空间优化。

大允许行驶时间、固定成本和可变成本等特征<sup>[2]</sup>。根据每种类型的校车可用数目是否限制,HSBRP 又可分为车辆数不限制的 HSBRP 和车辆数限制的 HSBRP。前者主要用来确定不同类型校车的组合,后者则是在已知车型组合的情况下确定路径;在求解难度上,后者比前者求解更加困难。HSBRP 与多车型车辆路径问题(Heterogeneous VRP, HVRP)<sup>[8]</sup>在问题描述上类似,但仍存在差异,主要体现在学生最大乘车时间的约束、学校时间窗限制、站点服务时间与站点人数相关以及站点与学校的对应关系等。因 HSBRP 问题的约束和模型的特殊性,求解 HVRP 成熟的算法并没有很好地用于 HSBRP 的求解<sup>[2]</sup>。

现有 SBRP 研究中,仅有少量文献研究了 HSBRP<sup>[9-13]</sup>。文献[9]以学生总乘车时间为优化目标研究了农村地区的 SBRP,提出一种扫描法和禁忌算法混合的两阶段启发式算法(RRH)进行问题求解。其使用随机生成的小规模人工案例进行了测试,实验结果表明 RRH 优于节约法和基于位置的随机启发(RLBH)算法。文献[10]针对运营里程、学生步行距离和车辆成本等多个目标的 HSBRP,分别建立车流、单商品流、二商品流和多商品流等 4 个数学模型,然后使用商业软件 Xpress 分别求解,因此对比了 4 种模型的求解结果。由于使用精确算法进行求解,因此仅能求解 20 个站点的测试案例。文献[11]在学校、场站、车型等多个约束条件下,设计了一个构造启发式算法求解农村地区的 SBRP。算法包括学生分配到站点、车辆分配到学校、估算学生插入成本和交换学生等多个步骤进行解的构造和提升,在提升过程中使用“两小换一大”的策略调整车型。此外,他还对农村 SBRP 问题涉及的各种评价指标进行了讨论。文献[12]以总运营里程最小为优化目标,设计了一种自适应的基于位置的启发算法(ALBH)进行求解。车辆安排时以大车优先,所有路径构造完成之后再根据车的实际载重调整车型。文献[13]以车辆数为优化目标,设计了一个“后启发”算法进行求解,求解过程中对不同车型的校车进行分配,并不考虑不同车型的固定成本和可变成本的差异。

以上文献虽然对 HSBRP 进行了研究,但仍存在不足:1)目标函数中没有同时考虑车辆的固定成本和可变成本。因不同类型的校车购置成本及单位行驶成本不同,单纯以车辆数或总运营里程进行衡量并不能反映实际运营成本。2)没有考虑车辆数限制的车型的使用。实际应用中,受现有资源和资金等约束,校车运营车队中每种车型的车辆数目是固定的,因此路径安排时不能超过每种车型校车的最大数量限制。3)求解算法以精确算法和简单启发算法为主。HSBRP 因增加了车型的限制,使其求解更加困难,精确算法仅能求解小规模案例。简单启发算法以构造启发式和简单禁忌算法为主,而求解 NP-hard 难题较为有效的元启发算法在 HSBRP 上的研究成果较少。

迭代局部搜索算法(Iterated Local Search, ILS)是一种基于邻域搜索的元启发算法,其借助扰动跳出局部最优。ILS 具有简单有效、易与其他算法混合等优点,在求解车辆路径等问题上表现出良好的性能<sup>[14,15]</sup>。可变邻域下降(Variable Neighborhood Descent, VND)是变邻域搜索的一种,它能够在给定的若干个邻域上快速寻优,通常与其他算法混合<sup>[16]</sup>。研究表明,混合使用多个元启发算法能够充分利用不同算法的优势,进而产生更好的解<sup>[14]</sup>。

基于此,本文以车辆的固定成本和可变成本为优化目标,

建立车辆数限制的多车型校车路径问题(Heterogeneous Fixed Fleet School Bus Routing Problem, HFSBRP)的数学模型,并提出一种迭代局部搜索算法(HILS)进行求解。HILS 在局部搜索过程中混合使用随机邻域选择的 VND 算法进行局部寻优;并针对 HFSBRP 的问题特征,设计车型调整策略、邻域解接受策略和当前解扰动机制等提升优化目标。在国际基准案例上进行了测试,实验结果证明了 HILS 算法的有效性。

## 2 问题描述及定义

HFSBRP 涉及的主要因素包括场站、校车、乘车站点、学生和道路网络等。本文以上学情况下的 HFSBRP 问题进行研究(放学问题可以转化为上学问题进行解决)。HFSBRP 可以描述为:校车服务场站有一组车辆类型不同的校车提供校车服务,校车从场站出发经过乘车站点将学生送到学校,每个乘车站点均有若干名学生,并且校车在站点停留一段时间以便学生能够上车。目标是在考虑校车固定成本和校车行驶成本的情况下,找到总运营成本最小的路径安排。约束条件为:所有的校车必须从同一个场站出发,到达学校之后即完成任务配送;每个乘车站点必须被访问且只能访问一次;任意时刻校车上的学生人数之和不能超过校车的容量;学生在校车上的乘坐时间不能超过最大乘车时间;每种类型的校车具有不同的容量、固定成本和单位可变成本,且使用的每种车型的校车数目不能超过其车型最大数量的限制。

根据以上描述,定义符号和决策变量如下: $G=(V, E)$ 为配送网络; $V$ 是节点的集合, $V=\{0, 1, 2, 3, \dots, n, n+1\}$ ,其中 0 代表场站,设  $C=\{1, 2, 3, \dots, n\}$  代表乘车站点, $n+1$  代表学校; $E$  是任意两个节点之间边的集合,即  $E=\{(i, j) | i, j \in V, i \neq j\}$ ;  $d_{ij}$  表示  $E$  中一条边  $(i, j)$  的距离,且  $d_{ij}=d_{ji}$ ;  $q_i$  为乘车站点  $i$  的乘车学生的人数;  $t_i$  为乘车站点  $i$  的服务时间,与站点的乘车人数有关,  $t_0=0$ ;  $M$  为校车类型集合,  $M=\{1, 2, 3, \dots, m\}$ ;  $Q_k$  为  $k$  类型的校车的容量,  $k=1, 2, 3, \dots, m$ ;  $f_k$  为  $k$  类型的校车的固定成本;  $v_k$  为  $k$  类型的校车的单位可变成本;  $n_k$  为  $k$  类型的校车的最多车辆数;  $T$  为学生的最大乘车时间;  $x_{ijk}$  表示校车  $k$  访问站点  $i$  后是否直接访问站点  $j$ ,若是则  $x_{ijk}=1$ ,否则  $x_{ijk}=0$ ;  $y_{ik}$  表示校车  $k$  到达站点  $i$  时的累积乘车人数,  $y_{0k}=0$ ;  $z_{ik}$  表示校车  $k$  离开站点  $i$  的累计行驶时间(按逆序进行计算,  $z_{(n+1)k}=0$ )。

借助以上符号和决策变量定义,建立基于校车类型的 HFSBRP 混合整型线性规划(MIP)模型:

$$\text{Min} \sum_{j \in C} \sum_{k \in M} f_k x_{0jk} + \sum_{i \in V \setminus \{n+1\}} \sum_{j \in V \setminus \{0, i\}} \sum_{k \in M} v_k d_{ij} x_{ijk} \quad (1)$$

$$\text{s. t.} \sum_{j \in V \setminus \{i, n+1\}} \sum_{k \in M} x_{ijk} = 1, \forall i \in C, k \in M \quad (2)$$

$$\sum_{i \in V \setminus \{n+1\}} x_{ipk} - \sum_{j \in V \setminus \{0\}} x_{pj k} = 0, \forall p \in C, k \in M \quad (3)$$

$$\sum_{i \in V \setminus \{0\}} x_{0ik} \leq n_k, \forall k \in M \quad (4)$$

$$y_{ik} \leq Q_k, \forall i \in V, k \in M \quad (5)$$

$$y_{ik} + q_i - y_{jk} \leq M_1(1 - x_{ijk}), \forall i \in V \setminus \{n+1\}, j \in V \setminus \{0, i\}, k \in M \quad (6)$$

$$z_{ik} \leq T, \forall i \in V \setminus \{0\}, k \in M \quad (7)$$

$$z_{jk} + t_{ij} + t_i - z_{ik} \leq M_2(1 - x_{ijk}), \forall i \in V \setminus \{n+1\}, j \in V \setminus \{i\}, k \in M \quad (8)$$

$$x_{ijk} \in \{0, 1\}, \forall i \in V \setminus \{n+1\}, j \in V \setminus \{0, i\}, k \in M \quad (9)$$

$$y_{ik} \in \{0, 1, 2, \dots\}, \forall i \in V, k \in M \quad (10)$$

$$z_{ik} \in \{0, 1, 2, \dots\}, \forall i \in V, k \in M \quad (11)$$

函数(1)表示优化目标为总运营成本(即车辆的固定成本和可变成本)最小。约束(2)限定每一个学生站点都必须由一

辆车提供服务,并且只允许被访问一次。约束(3)保证若一辆校车访问某一个学生站点之后,该校车必须离开。约束(4)保证每种车型的校车不能超过其最大车辆数。约束(5)保证任何时刻,乘坐的学生人数不能超过车辆的容量。约束(6)表示两个站点间的累积关系。若校车  $k$  访问站点  $i$  后立即访问站点  $j$  ( $x_{ijk}=1$ ),校车容量满足  $(y_k + q_j)x_{ijk} = y_k$ ; 其他情况下 ( $x_{ijk}=0$ ),则满足  $(y_k + q_j)x_{ijk} < y_k$ 。因此采用非线性不等式  $(y_k + q_j)x_{ijk} \leq y_k$  表示站点间的容量积累关系,引入一个较大的正整数  $M_1$  将其转换为线性不等式(6)。约束(7)表示所有学生的最大乘车时间不能超过最大值  $T$ 。约束(8)表示两个连续站点的校车运行时间积累,同式(6)通过引入较大整数  $M_2$ ,将非线性不等式  $(z_k + t_{ij} + t_i)x_{ijk} \leq z_k$  转换为线性不等式。约束(6)和约束(8)均能够消除不经过车场和学校闭合回路。约束(9)~约束(11)限定决策变量  $x_{ijk}$ ,  $y_k$  和  $z_k$  的取值。

基于车型构造混合整型规划模型,减少了决策变量的个数,能在一定程度上减少模型的计算复杂度;另外,按逆序计算校车行驶时间,能够简单而清晰地表达学生最大乘车时间约束条件,校车行驶时间可在模型求解后再计算。

### 3 算法设计

#### 3.1 算法基本描述

本文设计的 HILS 算法采用标准的 ILS 算法框架,并对其改进。首先,通过增加外层循环提供多个初始解,然后在内层循环中使用局部搜索和扰动等进行寻优。为保证每次迭代从不同的初始解开始,随机选择一种初始解构造方法。局部搜索过程中,借助一种改进的 VND 算法进行解的提升,并通过扰动方法跳出局部最优。经过若干次外层迭代之后,得到算法的全局最好解。HILS 的算法流程如下。

1. 初始化外层迭代次数  $outerIter$ 、参数  $\beta$ 、搜索规则  $rules$ 、全局最好解  $S_g$ , 设置迭代变量  $t=0$ 。
2. 根据车型信息,构造问题的初始解  $S_0$ 。
3. 使用初始解  $S_0$  和参数  $\beta$  确定内层循环次数  $inner$ ; 设置内层迭代变量  $m=0$ , 当前解  $S_c$ , 局部最优解  $S_b$ , 置  $S_c = S_b = S_0$ 。
4. 使用改进的 VND 算法对当前解  $S_c$  进行局部提升得到解  $S_c'$ , 将  $S_c'$  和  $S_b$  二者中的更优者作为新的局部最优解  $S_b$ 。
5. 使用扰动规则对  $S_c'$  进行扰动,并将扰动后的解赋给  $S_c$ 。
6. 执行  $m++$ ; 若  $m < inner$ , 转到步骤 4 继续执行。
7. 若局部最优解  $S_b$  优于全局最好解  $S_g$ , 则  $S_g = S_b$ 。
8.  $t++$ ; 若  $t < outerIter$ , 跳转到步骤 2。
9. 输出全局最好解  $S_g$ , 算法结束。

#### 3.2 初始解构造

因每种车型有最大车辆数限制,且有最大乘车时间的约束,使得使用节约法、扫描法等经典的初始解构造方法并不容易得到可行的初始解。为此,本文设计一种基于最廉价成本的插入构造方法,提供串行插入策略和并行插入两种策略。具体构造过程如下。

- 1) 初始化所有站点的集合  $U$ , 设初始解  $S = \emptyset$ 。
- 2) 根据当前所有的车型总数目  $v$ , 构造  $v$  条  $0-s$  的路径, 并为每条路径分配车型, 然后将其放入  $S$ 。其中,  $0$  和  $s$  分别表示场站和学校。
- 3) 针对当前  $S$  中的每一条路径, 从  $U$  中选择 1 个节点  $i$  插入到路径中, 构成  $0-i-s$  形式的路径。循环执行此步骤, 直到形成  $v$  条  $0-i-s$  为止, 并将选中的  $v$  个站点从  $U$  中移除。具体插入时先将站点按照容量降序排列, 然后随机选择

$2/3$  的大站点放入车型容量较大的路径上, 然后再随机选择其他节点放入其他路径中, 插入过程中保证每条路径都是  $0-i-s$  的形式, 且不违反车型容量约束。

4) 随机选择一种插入策略, 从  $U$  中选择待插入到  $S$  中的站点  $j$  及其目标路径  $dr$ 。假设  $\Delta f(k, r)$  代表将站点  $k$  以最廉价的方式插入到  $S$  中的路径  $r$  时增加的成本值,  $\Delta f(k, r) = \infty$  时表示  $k$  不能插入到路径  $r$  中。二元组  $(j, dr)$  表示选中的站点  $j$  及其要插入的目标路径  $dr$ 。

① 并行插入策略: 考虑  $S$  中的所有路径, 寻找  $U$  中能够以最廉价成本插入到  $S$  中的站点  $j$  和  $dr$ 。此时, 二元组  $(j, dr) \leftarrow \min\{\Delta f(k, r)\}, k \in U, r \in S$ 。

② 串行插入策略: 首先针对  $S$  中的每一条路径  $r$ , 计算插入到  $r$  上的最廉价成本及其对应的站点等信息。使用二元组  $(m, r)$  表示, 则  $(m, r) \leftarrow \min\{\Delta f(k, r)\}, k \in U$ 。然后, 从这  $v$  个二元组中找出插入成本最小的二元组  $(j, dr)$ , 二元组  $(j, dr) \leftarrow \min\{\Delta f(m, r)\}, m \in U, r \in S$ 。

根据找到的二元组  $(j, dr)$ , 将站点  $j$  插入目标路径  $dr$  中, 同时将  $j$  从  $U$  中移除。若  $\min\{\Delta f(k, r)\} = \infty$ , 表示不存在这样的二元组, 跳转到步骤 1)。

5) 若  $U$  不为空跳转到步骤 4) 继续执行, 直到初始解  $S$  构造完成。

由于问题的约束较紧, 得到可行解比较困难。尽管该初始解构造算法需要经过多次尝试(实验验证不超过 30 次)才能获得初始解, 但该算法克服了常见经典算法无法获得可行解的局限性。另外, 步骤 3) 中站点的随机选择, 以及步骤 4) 中插入策略的随机选择, 使所构造的初始解具有随机性, 从而有利于增强邻域解的多样性。

#### 3.3 邻域结构设计

邻域操作是完成局部搜索的重要步骤, 本文使用了 6 个路径间邻域操作方法和 1 个路径内 2-opt 操作。为了便于描述, 假设路径  $r_1$  为  $0-a-b-c-d-e-s$ , 路径  $r_2$  为  $0-w-x-y-z-s$ 。其中,  $0$  和  $s$  是场站和学校, 并要求操作的节点不能为场站或学校。邻域结构主要包括:

- 1) shift(1, 0)。将一条路径上的节点移动到另外一条路径上。例如, 将  $r_1$  上的点  $b$  移动到  $r_2$  上的  $w$  之后, 路线  $r_1$  和  $r_2$  将变为  $0-a-c-d-e-s$  和  $0-w-b-x-y-z-s$ 。
- 2) swap(1, 1)。在两条不同路径上实现两个点的交换。例如, 将  $r_1$  上的点  $b$  与  $r_2$  上的  $x$  交换之后, 路线  $r_1$  和  $r_2$  将变为两条新路径  $0-a-x-c-d-e-s$  和  $0-w-b-y-z-s$ 。

3) shift(2, 0)。将一条路径两个连续的节点移动到另外一条路径上。例如, 将  $r_1$  上的  $b$  和  $c$  移动于  $r_2$  上的  $x$  之后, 路线  $r_1$  和  $r_2$  将变为  $0-a-d-e-s$  和  $0-w-x-b-c-y-z-s$ 。

4) swap(1, 2)。将一条路径上一个点与另外一条路径上的一条边进行交换。例如, 将  $r_1$  上的  $c$  与  $r_2$  上的边  $x-y$  交换之后, 路线  $r_1$  和  $r_2$  将变为  $0-a-b-x-y-d-e-s$  和  $0-w-c-z-s$ 。

5) swap(2, 2)。在两条不同路径上实现两条边的交换。例如, 将  $r_1$  上的  $c-d$  与  $r_2$  上的边  $x-y$  交换之后, 路径  $r_1$  和  $r_2$  将变为  $0-a-b-x-y-e-s$  和  $0-w-c-d-z-s$ 。

6) cross。每条路径上分别移除一条边, 然后再增加两条边, 实现路径段间的交叉。例如, 移除  $r_1$  上的  $c-d$  和  $r_2$  上的

边  $x-y$  之后,增加新边  $c-y$  和  $x-d$ ,此时路线  $r_1$  和  $r_2$  将变为  $0-a-b-c-y-x-s$  和  $0-w-x-d-e-s$ 。

7)2-opt。在一条路径内移除两条不相邻的边,而后将其两条边之间的路径反向,之后再添加两条新边,构成一条新的路径。例如,将  $r_1$  的边  $0-a$  和  $d-e$  移除,同时将  $a-b-c-d$  逆向为  $d-c-b-a$ ,再增加两条边  $0-d$  和  $a-e$ ,此时  $r_1$  变为新路径  $0-d-c-b-a-e-s$ 。2-opt 操作仅在路径间邻域操作有提升时,才使用其对参与路径间操作的两条路径进行提升。

### 3.4 邻域随机选择的 VND 算法

ILS 算法中局部搜索通常使用多个局部邻域操作顺序执行或 VND 来实现。本文使用一种邻域随机选择的 VND 算法(RVND)完成局部搜索。RVND 与常规 VND 的不同之处在于:每次随机选择一个邻域进行操作,若当前解有所提升,则继续使用此算子;当使用该邻域操作不能再对解进行提升时,将其从邻域列表中移除。迭代执行,直到邻域列表为空。RVND 算法的执行步骤如下。

1. 初始化邻域集合  $NL = \{N_1, N_2, \dots, N_{kmax}\}$ , 邻域下标  $m$ , 当前解  $S_c$ 。  
此处  $NL$  是 3.3 节定义的 6 种路径间操作算子的集合。
2. 随机选择一个邻域  $N_m (1 \leq m \leq kmax)$ 。
3. 使用  $N_m$  对  $S_c$  进行提升,找到此邻域内的最好解  $S_c^* (S_c^* \in N_m (S_c))$ 。
4. 若  $S_c^*$  优于  $S_c$ ,或者  $S_c^*$  与  $S_c$  的成本值在一定偏差范围内,置  $S_c = S_c^*$  同时跳转到步骤 3;否则将  $N_m$  从  $NL$  中移除。
5. 若  $NL$  不为空,跳转到步骤 2 继续执行;否则执行结束返回  $S_c$ 。

$NL$  中每个邻域结构执行的过程中,均要尝试进行车型调整。因每种车型的车辆数固定,因此车型能否调整成功取决于能否找到可用车型以及可用车型数。车型调整策略定义如下:若当前路径上的车型不能满足容量需求,尝试使用较大的车型,寻找增加成本最小的车型;当车型容量满足需求,尝试使用容量较小的车型,寻找成本下降最多的车型;无论哪种车型调整均要保证每种车型的使用数量不能超过车型车辆数的限制。若找不到合适的可用车型,则不予进行车型调整。车型调整的好处是尽可能地减少路径上的总成本,并为后续搜索提供多样性的解。

### 3.5 邻域解接受策略

邻域解接受策略能够通过定义的策略决定是否接受邻域解,它能使算法沿着目标值下降的方向进行搜索。本文设计缩减路径数目(MNS)和提高路径满载率(VUS)两种邻域解接受策略,以尽可能地降低总成本。

#### (1)MNS 邻域解接受策略

由于 3.2 节的初始解构造方法中使用了所有可用的车型,尽管这样能够得到可行的初始解,但是总成本过高,尤其是固定成本太高。由于 HFSBRP 的优化目标为固定成本和可变成本,而固定成本总是大于其可变成本,因此快速降低其固定成本能够较快地降低总成本。因此在搜索前期,缩减路径数目、降低路径上车辆的固定成本有助于更好地优化目标。因此,MNS 定义如下:

$$Eval(S) = \alpha|S| + \beta(-\sum_{r \in S} |r|^2) + \gamma \sum_{r \in S} c(r) \quad (12)$$

其中,  $|S|$  表示路径解  $S$  包含的路径数,  $|r|$  为其中一条路径上站点的数量,  $c(r)$  表示路径的成本。系数  $\alpha \gg \beta \gg \gamma$ , 表示按照字典序依次进行评价。 $\sum_{r \in S} |r|^2$  增大意味着解决方案中存

在某些路径正向服务更多站点的方向发展,有利于路径的合并从而减少路径数<sup>[17]</sup>。

#### (2)VUS 邻域解接受策略

若在优化过程中仅考虑路径的缩减可能会过早陷入局部最优,从而导致倾向使用较大的车型,此时可以使用 VUS 策略。以参与邻域操作的两条路径  $r_1$  和  $r_2$  为例,VUS 定义如下:

$$Eval(S) = \min\{|Q_m - D_m|, |Q_l - D_l|\} - \min\{|Q_k - D_k|, |Q_l - D_l|\} \quad (13)$$

其中,  $r_1$  和  $r_2$  进行邻域操作前的车型分别为  $k$  和  $l$ , 对应的车型容量为  $Q_k$  和  $Q_l$ , 实际载重容量为  $D_k$  和  $D_l$ ; 进行邻域操作后的车型分别为  $m$  和  $n$ , 对应的车型容量为  $Q_m$  和  $Q_n$ , 实际载重容量为  $D_m$  和  $D_n$ 。若  $Eval(s) < 0$ , 表示操作后至少有一条路径上车辆的满载率提高,此时接受此邻域解;若  $Eval(s) = 0$ , 表示满载率没有变化,此时仅接受总成本降低的操作,否则不予接受。

### 3.6 扰动规则

为避免 HILS 算法过早地陷入局部最优,本文设计多点移动和多点交换两种扰动规则对当前局部最优解进行扰动。多点移动是针对每个随机选择的节点,在其邻域内查找不同路径间能够移动的位置完成移动操作;而多点交换则是寻找其邻域内能够进行交换的节点完成交换操作。扰动时不考虑总成本的变化,仅考虑是否能够得到可行的解。扰动时随机选择一种扰动规则进行操作。

### 3.7 算法复杂度

假设 HILS 算法的外层迭代次数为  $M$ , 问题规模为  $N$ , 内层迭代次数为  $L$ , 其中  $L$  与问题规模和总车型的数量  $v$  有关, 即  $L = N + \beta * v (\beta, v$  均为常数)。按照一次迭代过程分析算法的时间复杂度。一次迭代要执行初始解构造、邻域搜索和扰动等步骤。初始解构造最多尝试次数为  $P (P$  为常数), 初始解中共有  $v$  条路径, 平均每条路径上有  $\lceil \frac{N}{v} \rceil$  点, 将  $N$  个点插入到路径上的时间复杂度为  $O(N * N/v) = O(N^2/v)$ , 由此初始解构造函数的时间复杂度为  $P * O(N^2/v)$ 。邻域搜索 VND 中包含 6 个邻域间操作算子, 其时间复杂度均为  $O(N^2)$ <sup>[14]</sup>。当路径间操作算子有提升时继续执行 2-opt 算子, 故一个邻域操作最长执行复杂度为  $O(N^2 + N^2)$ , 所以 VND 的最长执行时间的复杂度为  $O(6 * 2 * N^2)$ 。多点移动和多点交换扰动时相当于执行多次 shift(1,0) 和 swap(1,1), 因此扰动的复杂度仍为  $O(N^2)$ 。由此可知, HILS 算法的复杂度为  $M * (N + \beta * v) [P * O(N^2/v) + O(12 * N^2) + O(N^2)] \approx M * P * O(N^3)$ 。

## 4 实验与结果分析

### 4.1 测试案例

由于 HSBRP 问题尚无国际基准测试案例, 因此在文献 [13] 设计的 SBRP 基准测试案例的基础上扩充车型信息, 以便进行测试。根据站点和学校分布的情况, SBRP 案例分为随机分布 RSRB 和集中分布 CSCB 两类。假定车辆的平均行驶速度为 29.333333 英尺/秒 (20 英里/小时)。学生站点的服务时间为  $t_i = 19.0 + 2.6q_i$ , 其中  $q_i$  为在站点  $i$  上车的学生数,  $t_i$  的单位为秒 (s)。最大乘车时间 (MRT) 设定为 2700s。

距离计算方式使用曼哈顿距离,并以英尺进行表示。

在文献[13]的基础上设计两组测试案例。第一组案例是由 RSRB02 和 CSCB02 中站点数小于或等于 15 的案例组成,编号为 S01—S08。第二组基准测试案例由 RSRB01 和 CSCB01 中的学校案例组成,并将 RSRB01 中 6 个案例编号为 R01—R06, CSCB01 中 6 个案例编号为 C01—C06。两组

测试案例使用与文献[13]相同的原始数据,在此基础上设计车型信息,其车型信息如表 1 所列。其中,  $Q_k$  是类型为  $k$  的车辆载重能力,  $k \in \{A, B, C\}$ ;  $f_k$  表示类型为  $k$  的车辆的固定成本,  $v_k$  代表类型为  $k$  的车辆行驶每英里的可变成本,  $n_k$  是车型  $k$  的最大数目。假设当车型  $Q_A < Q_B$  时,  $f_A < f_B$  且  $v_A < v_B$ 。

表 1 20 个测试案例车型信息

案例	站点数	学生数	车型 A				车型 B				车型 C			
			$Q_A$	$f_A$	$v_A$	$n_A$	$Q_B$	$f_B$	$v_B$	$n_B$	$Q_C$	$f_C$	$v_C$	$n_C$
S01	5	50	12	1000	0.6	1	20	2000	0.9	2	40	3000	1.8	1
S02	10	83	20	1200	1.2	1	30	2000	1.5	2	30	1500	1.5	1
S03	9	49	10	800	0.6	1	20	1000	1.2	2	66	1800	1.8	2
S04	12	212	40	1200	1.2	2	50	1500	1.5	1	40	1600	1.5	2
S05	7	29	10	1000	0.6	2	20	1500	1.2	2				
S06	13	136	10	500	0.6	1	30	1200	1.2	3				
S07	7	80	20	1200	1.2	1	40	2400	1.5	2				
S08	6	131	40	1500	1.2	2	66	1800	1.5	1				
C01	70	887	27	1000	1.0	5	54	2500	1.5	12	72	3000	1.7	9
C02	35	674	40	2200	1.2	4	60	2700	1.3	6	72	3000	1.6	4
C03	30	492	30	1200	1.0	3	60	2500	1.3	3	70	3000	1.4	4
C04	23	402	30	1000	1.1	2	50	2200	1.3	5	60	2500	1.4	2
C05	75	1116	40	2500	1.2	4	60	3000	1.4	16	70	3500	1.5	6
C06	17	336	30	2000	1.1	2	40	2500	1.3	3	60	3000	1.5	4
R01	38	569	40	2000	1.1	1	50	2200	1.3	5	70	2500	1.4	6
R02	40	557	30	2400	1.0	2	50	3000	1.2	5	70	3500	1.5	6
R03	51	794	40	2500	1.2	2	50	3000	1.5	6	70	3500	1.7	9
R04	35	427	30	1800	1.0	5	45	2500	1.2	5	60	3200	1.5	3
R05	42	550	40	2600	1.0	5	60	3200	1.3	6	70	3500	1.6	2
R06	44	512	30	1800	1.0	4	40	2500	1.2	3	60	3000	1.4	6

#### 4.2 算法测试与比较

HILS 算法使用 Visual Studio 2010 环境中的 C# 来实现,在 PC 机上进行测试,机器配置为 Intel core i7-4790 3.60 GHz, 8GB 内存、Windows 7 64 位操作系统。算法的各种参数设置如下:外层迭代次数为 5;邻域大小  $nbsize = \min\{n, 30\}$ ;内层循环次数为  $n + \beta \times v$ , 参数  $\beta$  的取值为 5(实验测试的最好值);偏差值系数为  $10^{-5}$ ;扰动操作尝试 50 次,扰动节点个数取问题规模的 20%。每个案例随机运行 10 次。

##### 4.2.1 算法测试

为评估算法的性能,分别使用 IBM CPLEX 12.6 优化软

件和 HILS 求解表 1 中的案例。针对表 1 中的每一个案例,构建第 2 节定义的数学模型。CPLEX 与 HILS 的运行环境相同,模型中参数  $M_1 = 200$ ,  $M_2 = 20000$ ; CPLEX 的参数  $mipgap = 10^{-10}$ , 运算时间限定为 7200s。CPLEX 能够得到可行解的案例结果和相应 HILS 的运算结果,如表 2 所列。表中“\*”标注的目标值表示达到 CPLEX 求解的最优解; Best, Average, Worst 分别代表最好解、平均解和最差解; SDV 表示标准差系数(%), Fleet 和 CUR 表示最好解对应的车型组合和总的车辆利用率; Time 表示 CPLEX 的执行时间(s), AvgTime 表示 HILS 的平均运算时间(s)。

表 2 CPLEX 和 HILS 案例的运算结果

案例	CPLEX				HILS					
	Best	Time(s)	Best	Average	Worst	SDV(%)	Fleet	CUR(%)	AvgTime(s)	
S01	5062.57*	0.16	5062.57*	5062.57*	5062.57*	0.00	1A2B0C	96.15	0.01	
S02	7135.77*	4.16	7135.77*	7135.77*	7135.77*	0.00	0A2B1C	83.00	0.03	
S03	3386.92*	1.08	3386.92*	3387.26	3388.64	0.01	1A1B1C	81.67	0.02	
S04	6469.90*	13.18	6469.90*	6470.60	6472.24	0.11	1A1B2C	95.50	0.06	
S05	3567.46*	0.14	3567.46*	3567.46*	3567.46*	0.00	2A1B0C	72.50	0.02	
S06	6261.82*	112.1	6261.82*	6262.44	6264.92	0.01	1A2B2C	90.67	0.09	
S07	6122.95*	0.22	6122.95*	6122.95*	6122.95*	0.00	1A2B0C	80.00	0.01	
S08	4894.26*	0.17	4894.26*	4894.26*	4894.26*	0.00	2A1B0C	89.73	0.01	
C01	—	7202.68	40232.22	40622.15	40737.37	0.98	5A3B9C	93.86	13.21	
C02	35178.81	7279.02	32534.25	32537.39	32539.85	0.01	3A5B4C	95.20	1.62	
C03	—	7202.96	22272.97	22276.01	22278.48	0.02	2A3B4C	94.62	1.05	
C04	18297.01	7202.18	18297.01	18297.90	18300.40	0.02	2A5B2C	93.49	1.88	
C05	—	7227.83	57235.62	57797.38	58736.53	1.41	0A13B5C	98.76	17.19	
C06	18776.70*	2894.93	18776.70*	18777.96	18780.90	0.01	2A1B4C	98.82	0.19	
R01	—	7203.81	23427.37	23513.21	23713.97	0.49	1A4B5C	96.44	1.39	
R02	—	7204.73	32084.35	32161.79	32679.01	0.72	2A2B6C	96.03	1.43	
R03	—	7203.91	43138.28	43147.55	43166.08	0.01	2A2B9C	98.02	3.27	
R04	—	7205.03	26432.51	26713.57	27130.95	0.59	5A3B3C	91.83	2.61	
R05	—	7203.87	31209.32	31440.92	31836.48	0.78	3A5B2C	98.21	2.99	
R06	—	7205.93	28085.49	28376.32	30112.89	1.94	4A1B6C	98.46	2.49	
平均	—	4118.41	20831.39	20928.27	21146.09	0.36	—	92.15	2.45	

从表 2 的实验结果可知:1)CPLEX 能够求解小规模案例(S01-S08),并能获得最优解。这些案例有 5-13 个学生站点,使用 3-5 辆校车,需要 0.16-112.1s 的计算时间。2)对于案例 R01-R06 和 C01-C06,CPLEX 能够找到 C06 的最优解,另外 C02 和 C04 两个案例能够在 7200s 内找到可行解,而其余案例均无法找到一个可行解。1)、2)进一步可知本文基于车型构建模型,并按逆序计算车辆行驶时间是可行的;同时也说明该模型的数学求解相当困难。3)HILS 算法在 CPLEX 能够找到最优解的案例上均发现了最优解,而对于 CPLEX 仅能发现可行解的 C02 案例,HILS 相对于 CPLEX 提升了 7.52%。对于 CPLEX 找到最优解的案例,HILS 能够在较短的时间找到最优解。HILS 算法在所有案例上的平均车辆利用率为 92.15%,部分案例(C05,C06,R03,R05 和 R06)的车辆利用率在 98%以上。与此同时,HILS 在所有案例上的标准差系数平均为 0.36%,算法相对比较稳定。

#### 4.2.2 算法比较

为验证 HILS 算法的性能,将其与另外 3 种 ILS 算法进行比较。1)局部搜索中邻域算子按照固定顺序执行的 ILS 算法,记为 ILS\_FLS;2)局部搜索中邻域算子随机选择的 ILS 算法,记为 ILS\_RLS;3)局部搜索使用标准 VND 的 ILS 算法,记为 ILS\_BVND。分别使用这 3 种算法求解表 1 中的案例,每种算法与 HILS 设置相同的参数,随机运行 10 次,统计其最好解和平均运算时间。对于小规模案例 R01-R08,3 种算法均能在较短的时间内找到最优解,此处仅给出案例 R01-R06 和 C01-C06 的统计结果,如表 3 所列。其中  $TC_1, TC_2, TC_3$  和  $TC_4$  分别代表 ILS\_FLS, ILS\_RLS, ILS\_BVND 和 HILS 找到的最好解。 $T_1, T_2, T_3$  和  $T_4$  表示相应的平均运算时间(s)。 $g_1, g_2$  和  $g_3$  分别是 HILS 相对于 ILS\_FLS, ILS\_RLS 和 ILS\_BVND 3 种算法的改进程度。

表 3 HILS 与另外 3 种 ILS 算法的运算结果比较

案例	ILS_FLS		ILS_RLS		ILS_BVND		HILS		Gap(%)		
	$TC_1$	$T_1$	$TC_2$	$T_2$	$TC_3$	$T_3$	$TC_4$	$T_4$	$g_1$	$g_2$	$g_3$
C01	40727.01	12.99	40732.60	13.63	40724.49	13.38	40232.22	13.39	1.21	1.23	1.21
C02	32546.98	1.44	32539.79	1.54	32536.70	1.71	32534.25	1.77	0.04	0.02	0.01
C03	22278.79	0.78	22276.43	1.31	22273.22	0.84	22272.97	1.17	0.03	0.02	0.001
C04	18297.01	1.48	18297.01	1.50	18297.65	1.80	18297.01	2.10	0.00	0.00	0.003
C05	57732.87	14.96	57730.70	19.07	57735.35	15.51	57235.62	22.27	0.86	0.86	0.87
C06	18776.77	0.16	18776.70*	0.16	18776.70*	0.19	18776.70*	0.20	0.0003	0.00	0.00
R01	23718.18	1.33	23714.06	1.34	23701.01	1.50	23427.37	1.48	1.23	1.21	1.15
R02	32103.08	1.45	32209.98	1.50	32098.61	1.50	32084.35	1.56	0.06	0.39	0.04
R03	43182.64	3.22	43174.66	3.31	43146.55	3.15	43138.28	3.27	0.10	0.08	0.02
R04	27150.34	2.85	26432.69	3.17	26432.63	3.19	26432.51	2.57	2.64	0.001	0.0004
R05	31215.12	2.63	31228.18	2.87	31216.04	2.75	31209.32	2.78	0.02	0.06	0.02
R06	28122.24	2.23	28116.34	1.89	28090.96	2.38	28085.49	2.80	0.13	0.11	0.19
平均	31320.92	3.79	31269.10	4.27	31252.49	3.99	31143.84	4.61	0.53	0.33	0.28

由表 3 可知,HILS 在所有案例上拥有最好的平均解,且在所有案例上均找到了最好解。相比 ILS\_FLS, ILS\_RLS 和 ILS\_BVND, HILS 对于单个案例最大提升 2.64%, 1.23% 和 1.21%; HILS 在所有案例上平均提升 0.53%, 0.33% 和 0.28%。在运算时间上,3 种 ILS 算法基本相当。ILS\_BVND 和 HILS 的求解质量优于 ILS\_FLS 和 ILS\_RLS,原因在于局部搜索过程中混合了 VND 算法,能够充分发挥 VND 的优势,这也说明混合算法优于单一算法。在 VND 中邻域的排序会影响到其求解性能,HILS 中邻域随机排序能够在搜索过程中增加邻域选择的多样性,为后续得到更好的解提供了可能,故而 HILS 优于邻域按大小固定排序的 ILS\_BVND。

#### 4.3 邻域解接受策略对算法的影响

为评估邻域解接受策略对算法的影响,测试不同邻域解接受策略对 HILS 算法的影响。在本文算法框架下,分别使用 MNS 策略、VUS 策略以及本文算法使用的混合策略(MNS+VUS)对表 1 中 R01-R06 和 C01-C06 的案例进行求解,统计 3 种策略下算法的平均解、发现最好解的数目和总执行时间等信息,结果如表 4 所列。

表 4 不同邻域解接受策略的求解效果对比

评价标准	MNS	VUS	MNS+VUS
平均解	31577.20	31385.28	31143.84
发现最好解数目	2	4	12
总运算时间(s)	45.88	46.56	48.58

由表 4 可知,使用 MNS+VUS 混合策略在平均解、发现

最好解数目上优于单独使用 MNS 策略或 VUS 策略,在总运算时间上基本相当。VUS 策略优于 MNS 策略,原因在于 VUS 侧重于提高车辆的利用率,本质上也能减少路径数目;MNS 策略只关注路径数目的减少,有可能倾向于使用大车以减少路径数,进而导致总成本很难减少。MNS+VUS 的混合策略能够考虑到车型与成本之间的关系,在缩减路径数的同时尽可能地提高车辆的利用率,从而能够在搜索过程中发现更好的解。

**结束语** 本文针对车辆数限制的多车型校车路径问题(HFSBRP)定义其数学模型,并提出一种迭代局部搜索算法进行求解。该算法在标准 ILS 基础上增加外层循环提供多个初始解,并使用随机选择邻域的 VND 算法进行局部提升;针对求解问题的特征,使用车型调整策略和两种邻域解接受策略进一步提升算法的寻优能力。在国际基准测试案例的实验结果表明:建立的 HFSBRP 模型是正确的;在 ILS 的局部搜索过程中,使用随机邻域选择的 VND 算法比使用邻域固定顺序、邻域随机选择和标准 VND 的求解质量更高;混合使用两种邻域解接受策略比使用单一策略在寻优能力上更有优势。下一步研究将尝试设计求解 HFSBRP 问题的其他元启发算法,例如粒子群算法、蚁群算法等。

#### 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 Lan-xue, Chen Xiao-pan, Kong Yun-feng. 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] Zhang Fu, Zhu Tai-ying. Optimization Design for the School Bus Stations and Routing [J]. *Mathematics in Practice and Theory*, 2012, 42(4): 141-146 (in Chinese)  
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, 42(4): 141-146
- [5] Kinable J, Spijksma F C R, Berghe V G. School bus routing-a column generation approach [J]. *International Transactions in Operational Research*, 2014(21): 453-478
- [6] Xu Wen-long, Li Xiao-juan, Gong Hui-li, et al. An algorithm for school bus optimal path planning [J]. *Geospatial Information*, 2011, 9(4): 67-71 (in Chinese)  
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, 9(4): 67-71
- [7] Dang Lan-xue, Hou Yan-e, Kong Yun-feng. Spatio temporal Neighborhood Search for Solving Mixed-load School Bus Routing Problem [J]. *Computer Science*, 2015, 42(4): 221-225 (in Chinese)  
党兰学, 侯彦娥, 孔云峰. 时空相关的混载校车路径问题邻域搜索 [J]. *计算机科学*, 2015, 42(4): 221-225
- [8] Kusuma S, Anan M, Gerrit K J, et al. Heterogeneous VRP Review and Conceptual Framework [J]. *Lecture Notes in Engineering and Computer Science*, 2014, 2210(1): 1052-1059
- [9] Ripplinger D. Rural school vehicle routing problem [J]. *Transportation Research Record; Journal of the Transportation Research Board*, 2005, 1992: 105-110
- [10] Ke X. School bus selection, routing and Scheduling [D]. Canada, Windsor; University of Windsor, 2005
- [11] 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
- [12] 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, 6093: 247-256
- [13] 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
- [14] Vidal T, Crainic T G, Gendreau M, et al. Heuristics for multi-attribute vehicle routing problems: A survey [J]. *European Journal of Operational Research*, 2013, 231(1): 1-21
- [15] Penna P H V, Subramanian A, Ochi L S. An iterated local search heuristic for the heterogeneous fleet vehicle routing problem [J]. *Journal of Heuristics*, 2013, 19(2): 201-232
- [16] Hansen P, Mladenovic N. Variable Neighborhood Search; Principles and Applications [J]. *European Journal of Operations Research*, 2001, 130(3): 449-467
- [17] Dang Lan-xue, Wang Zhen, Liu Qing-shong, et al. Heuristic Algorithm for Solving Mixed Load School Bus Routing Problem [J]. *Computer Science*, 2013, 40(7): 248-253 (in Chinese)  
党兰学, 王震, 刘青松, 等. 一种求解混载校车路径的启发式算法 [J]. *计算机科学*, 2013, 40(7): 248-253

(上接第 208 页)

目占用比和平均评分因子的改进相似度计算方法, 得到一种新的协同过滤算法。仿真结果表明, 该算法有效地缓解了传统相似度计算引起的推荐结果不准确的问题; 将该算法应用到电影推荐系统中, 提高了电影推荐质量。

### 参考文献

- [1] Goldberg D, Nichols D, Oki B M, et al. Using Collaborative Filtering to Weave an Information Tapestry [J]. *Communications of the ACM*, 1992, 35(12): 61-70
- [2] Herlocker J, Konstan J A, RIED J. An Empirical Analysis of Design Choices in Neighborhood-Based Collaborative Filtering Algorithms [J]. *Information Retrieval Journal*, 2002, 5(4): 287-310
- [3] Breese J S, Heckerman D, Kadie C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering [C] // *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*. 1998: 43-52
- [4] Shang M S, Zhang Z K, Zhou T, et al. Collaborative filtering with diffusion-based similarity on tripartite graphs [J]. *Physica A; Statistical Mechanics and its Applications*, 2010, 389(6): 1259-1264
- [5] Resnick P, Iacovou N, Suchak M. Grouplens: an open architecture for collaborative filtering of net news [C] // *Proceedings of ACM CSCW 94 Conference on Computer-Supported Cooperative Work*. 1994: 175-186
- [6] Wang J, de Vries A P, Reinders M J. Unifying user-based and item-based collaborative filtering approaches by similarity fusion [C] // *Proceeding of SIGIR*. 2006
- [7] Li Hua, Wang Gen-long, Gao Min. A novel similarity calculation for collaborative filtering [C] // *Proceeding of the 2013 International Conference on Wavelet Analysis and Pattern Recognition*. 2013: 14-17
- [8] Wang Wei-jie, Yang Jing, He Liang. An improved Collaborative Filtering based on item similarity modified and common ratings [C] // *International Conference on Cyberworlds*. 2012: 213-235
- [9] Lee H C, Lee Y J. A study on the improved collaborative filtering algorithm for recommender system [C] // *The 5th International Conf. on Software Engineering Research, Management and Applications*. 2007: 297-304
- [10] Zhu Xu-zhen, Tian Hui, Cai Shi-min. Personalized recommendation with corrected similarity [J]. *Journal of Statistical Mechanics Theory & Experiment*, 2014, 2014(7)
- [11] Sun Hai-feng, Gan Ming-xin, Liu Xin, et al. Review on dominating websites for movie recommender systems [J]. *Journal of Computer Application*, 2013, 33(S2): 119-124 (in Chinese)  
孙海峰, 甘明鑫, 刘鑫, 等. 国外电影推荐系统网站研究与评述 [J]. *计算机应用*, 2013, 33(S2): 119-124