

Research on Optimization Model of Empty Car Distribution among Regional Railways Based on Freight Service Network

Hanxiao Jiao^{1,2}, Xiqi Wei^{1,2}, Dongyue Liang¹, Weihua Yang^{1*}

¹School of Mathematics, Taiyuan University of Technology, Jinzhong Shanxi

²Research Institute of Information Technology, Tsinghua University, Beijing

Email: 15735219674@163.com, 1052958064@qq.com, 1033621435@qq.com, *yangweihua@tyut.edu.cn

Received: Nov. 10th, 2019; accepted: Nov. 15th, 2019; published: Nov. 22nd, 2019

Abstract

This paper studies the problem of empty car distribution among regional railways. We have not seen any work on effective models and optimization algorithms for this problem. We construct an integer programming optimization model that considers the trade-off between the cost and distributing speed. The model dynamically selects assembly stations and plans car assembly paths according to the distribution of empty car. In this paper, a hybrid genetic algorithm for the above model is designed, and the minimum cost maximum flow algorithm is used to calculate the fitness value of individuals in the population, thus obtaining the high-accuracy solution. Numerical experiments show that under the premise of known freight service cost, the method adopted in this paper can save over 35% of the assembly cost on average.

Keywords

Empty Railcar Distribution, Integer Programming, Genetic Algorithm, Minimum Cost Maximum Flow

基于货运服务网络的铁路局车辆排空优化模型研究

焦含笑^{1,2}, 魏夏琦^{1,2}, 梁东岳¹, 杨卫华^{1*}

¹太原理工大学数学学院, 山西 晋中

²清华大学信息技术研究院, 北京

Email: 15735219674@163.com, 1052958064@qq.com, 1033621435@qq.com, *yangweihua@tyut.edu.cn

*通讯作者。

收稿日期：2019年11月10日；录用日期：2019年11月15日；发布日期：2019年11月22日

摘要

本文研究铁路空车调配过程中的车辆局外排空问题。针对已有研究对于铁路局之间的车辆排空问题缺乏有效模型与优化算法的问题，本文构建了综合考虑车辆排空成本与排空速度的整数规划优化模型，该模型能够依据空车分布情况动态选择车辆集结地、规划车辆集结路径。本文设计了针对上述模型的混合遗传算法，利用最小费用最大流算法计算种群中个体的适应度值，从而得到具有高精确性的解。通过数值实验表明，在货运服务费用已知的前提下，本文所采用的方法平均节约空车集结成本35%以上。

关键词

空车调配，整数规划，遗传算法，最小费用最大流算法

Copyright © 2019 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

货运列车由提供动力的机车与装载货物的车辆组成。一方面，车辆的种类众多，如棚车、敞车、罐车等等；另一方面，任意类型车辆的使用循环周期是一致的：空车 - 送达装车地装车 - 重车 - 送达目的地卸车 - 空车。当车辆在货物目的地卸空之后，将空车由卸车地运送至装车地进行重新装车的过程被称为空车调配。为空车指定装车地的决策过程是空车调配的核心问题。按照装车地归属的铁路局进行划分，空车调配可分为局内排空与局外排空两部分。本文主要研究空车调配中的局外排空问题，即空车所在地与装车地分属不同铁路局的情况。我国铁路部门将整个铁路网络(不包含城市轨道交通部分)进行划分，由18个铁路局分别管辖。每一个铁路局都可以被看作是一个独立运营的铁路公司。空车在进行局内排空时调度较为灵活，既可以与重车混编，也可以编开整列空车直达列车，而进行局外排空时一般只能通过编开空车直达列车送达外局。目前铁路局对于向局外排空的通行做法是，指定若干车站作为集结地，依据集结地的选择对全局的货运站进行划分，使得每个货运站均属于某个集结地的集结范围。当该货运站产生需要排空的车辆时，将这些车辆运送到对应的集结地，同时在集结地将到达的车辆编入空车直达列车向局外排空。这样做的不足之处在于没有考虑实际空车分布情况，一方面车辆运送成本较高，另一方面如果属于某集结地的集结范围内空车较少，则这些空车的排空速度将会明显降低。因此，我们面临的挑战是：铁路局应当如何高效且低成本地向局外排空。这个问题被称之为铁路局车辆排空问题。

本文将构建关于铁路局车辆排空问题的整数规划模型。这里引入服务网络：货运服务网络指为货物提供服务的网络，每个服务有对应的服务起点和终点，以及使用该服务的花费。空车从产生地经由货运服务网络运送至集结地。在该问题中铁路局管辖界内铁路网络是已知的。这个网络可以被一个有向赋权图来表示，图中的顶点表示货运站，边表示运输服务，边上的权重表示运输成本。例如，图中存在一条由顶点 u 指向 v 的有向边 (u, v) ，其权重为 w_{uv} ，则意味着铁路提供从 u 出发到达 v 的运输服务，该服务运送一辆车辆从 u 抵达 v 的成本为 w_{uv} 。显然，这个图是强连通的，即从任意顶点出发的车辆可以通过一串连续的首尾相接的运输服务抵达任意终点。我们设每个顶点上现有的需要向局外排空的空车数量亦为

已知量。铁路局车辆排空问题的主要决策内容包含两点：1) 选择哪些车站作为车辆集结地？2) 任意车站上的空车应当选择哪条服务路径到达集结地？在解决铁路局车辆排空问题时，我们希望以最小代价且最迅速地完成车辆排空。但是，成本最小与速度最快通常是矛盾的：快速通常意味着较高的代价，反之，低成本则无法保证速度。因此，我们在解决这个问题时需要寻求二者之间的平衡。车辆排空的成本主要产生于将车辆送至集结地的过程中。这里我们以车辆在集结过程中的运输成本最小化作为铁路局车辆排空优化模型的优化目标。车辆排空过程的快慢主要由集结到同一集结地的车辆数决定。集结到某一集结地的车辆越多，则编成空车直达列车所需的时间越短，意味着排空速度越快；反之则排空速度越慢。为了保证车辆排空速度，我们设置一个集结参数作为解的约束条件，使得求得的所有集结地所集结的车辆数都不低于这个集结参数。明显地，当集结参数取值为零时将会得到集结成本最小的解。集结参数取值越大，每个集结地所集结的车辆将越多，相应的车辆排空速度就越快，同时会造成集结成本升高。综上所述，我们可以通过调整集结参数来控制车辆排空成本与排空速度之间的平衡。

国内外学者对于空车调配问题做出了大量研究。Gorman [1]综述了铁路空车调配问题。很多学者针对客户需求，研究以客户满意度为主要优化目标的空车调配问题。Heydari 等人[2]建立了基于客户的多时段需求的空车调配模型，并且利用拉格朗日启发式算法与分枝定界算法求解该模型。温鑫[3]针对客户实际需求，建立了综合考虑车种代用与需求时间窗要素的铁路空车调配优化模型，并利用遗传算法求解模型。程学庆[4]综合考虑铁路经济效益最大化、货主满意度最大化和空重车调配路径最合理化目标，构建了铁路空车调配多目标优化模型。另一类研究则主要考虑了铁路运输能力对于空车调配的约束。程学庆等人[5]建立了基于时间窗和区段空车运输能力约束的空车调配优化模型。Spieckermann 等[6]将空车调配问题看作一个调度问题，以最小化空车走行费用建模，并设计求解模型的贪婪算法。林柏梁等人[7]综合考虑空车供需平衡与路段通过能力约束，建立了空车调配优化模型。朱海豹[8]考虑了列车开行频次所产生的影响，提出了一种最短路径未知且路段通过能力有限的空车调配优化模型。Holmberg 等人[9]考虑列车的到站和发站时间，建立了关于空车调配问题的带列车运力约束的多商品流优化模型。进一步的，Holmberg 等人[10]研究了具有路径成本固定的多商品流问题，并提出了一种基于拉格朗日启发式算法的混合算法，该算法采用了对偶次梯度搜索和原始启发式算法，计算结果表明该方法能较好地估计最优目标函数值的下界。亦有学者通过将时间变量划分为较短的时间段，研究多阶段动态空车调配问题。梁栋等人[11]采用动态规划方法提出了铁路局管内空车调配的多阶段策略优化模型。苏婕[12]利用时空网络图将空车调配问题转化为的运输问题(Transportation Problem)，通过依次计算每个时段内的运输问题最终得到原问题的解。目前已知的关于空车调配的研究几乎全部基于空车可以与重车混合编组的前提条件。通过前文对于铁路局车辆排空的流程描述可知，这些研究均不适用于铁路局之间的车辆排空问题。因此，本文的研究拓展了关于空车调配问题的研究领域，同时对于提升铁路车辆使用效率、降低运输成本具有实际意义。

本文的主要贡献有以下两点：

1. 通过对铁路局车辆排空流程与优化需求的分析，综合考虑排空成本与排空速度二者之间的平衡，本文建立了关于铁路局车辆排空问题的优化模型，该模型根据实际的空车分布情况动态选择车辆集结地、规划车辆集结方式；

2. 针对模型设计了以遗传算法为元启发式算法，以最小费用最大流算法计算种群中个体适应度的混合遗传算法，该算法对于铁路局车辆排空优化模型具有较高精确度。

本文主要内容安排如下：第二节主要描述关于铁路局车辆排空问题的整数规划优化模型；第三节主要叙述关于上述模型的算法设计；第四节主要描述数值实验设计与计算结果；第五节是关于本文研究的结论总结以及对于未来研究方向的展望。

2. 模型

2.1. 问题分析

排空问题主要需要决策集结站点如何选取以及空车的集结方式, 针对这样的需求, 已知的是货运服务网络和空车的分布。由于成本主要产生于集结过程, 因此将集结成本作为优化目标, 这里的成本是空车在运送至集结地过程中使用的服务网络成本。同时考虑到排空速度, 本文引入集结参数, 将每个集结地保有一定数量的空车作为约束条件。若该参数越大, 则编成列车发出的速度越快, 但相应地会增加成本。集结成本与排空速度两个因素相互制约, 本文通过建立车辆排空模型得到平衡两者的优化结果。

2.2. 符号说明

- G : 货运服务网络;
- V : 图中顶点集, 表示车站集合;
- E : 图中边集, 表示运输服务;
- W : 边上权重, 表示货运服务费用;
- a : 顶点上权重, 表示各车站上的空车数量;
- B : 给定常数, 集结参数, 表示集结站点上要求的空车数量下限;
- C : V 的子集, 表示集结站点集合;
- S : 表示除集结车站的其余车站集合;
- i : 表示除集结车站外的其余车站;
- j : 表示集结站点;
- x_{ij} : 决策变量; 表示 S 中 i 点送往 C 中 j 点的空车数量;
- N : 表示正整数集。

2.3. 符号说明

根据空车分布不同, 动态选取空车集结站点, 以集结过程中运送空车成本之和最小为目标, 并保证每个集结站点空车数量大于给定下限。模型输入为: 货运服务网络 G (假设任意两点之间空车均选择最短路径到达, 因此货运服务网络为实际路网抽象成的完全图), 各站空车数量 a , 集结参数 B 。输出为: 最优集结站点集合 C , 空车集结方案。构建铁路局车辆排空问题的整数规划优化模型如下:

$$\min \sum_i \sum_j w_{ij} \cdot x_{ij} \quad (1)$$

$$\text{s.t. } \sum_i x_{ij} + a_j \geq B, \forall j \in C, \quad (2)$$

$$\sum_j x_{ij} = a_i, \forall i \in S, \quad (3)$$

$$x_{ij} \in N. \quad (4)$$

模型中: 式(1)为目标函数, 为所有从车站 i 到车站 j 的空车数量乘以 i 到 j 的单位运送成本表示运送空车成本最小; 式(2)为从所有不同车站 i 发出到达 j 的空车数加上 j 站上的空车数之和大于 B , 表示各集结站点上空车数量的下限约束; 式(3)为 S 中任意一个车站 i , 送往 j 站的所有空车数之和等于该车站上的空车数量。式(4)为决策变量约束。

3. 算法

3.1. 最小费用最大流算法

最小费用最大流算法[13]是一类应用在网络中的组合优化算法, 网络的每条边上都有容量和费用两个

属性, 如何选择流向和相应的流量, 得到最小费用下的可行流是最小费用最大流算法主要解决的问题。具体为: 在初始网络 W 上, 取初始流为零流, 构造增广费用网络图, 求出网络中从起点到终点的最短路, 如果不存在, 输出最小费用最大流; 如果存在, 通过在原网络中得到相应的增广路, 对可行流进行调整, 调整后得到新的可行流, 重复以上过程直到找到网络中的最小费用最大流。

3.2. 遗传算法

遗传算法[14]是美国 J. Holland 教授于 1975 年提出的一种借鉴于生物进化规律而来的随机搜索算法, 它模拟了生物的进化过程, 通过选择、交叉以及变异等操作, 种群经过若干代进化后, 可以达到最优(或近最优)的状态。已广泛用于生产调度问题[15]、函数优化[16]、自适应控制[17]等方面。算法首先编码产生初始种群, 通过计算种群中个体的适应度值选择出适应度高的个体。利用交叉和变异操作产生新的种群, 重新对新的种群进行评价, 直到满足停止条件, 得出优化结果。

3.3. 最小费用最大流 - 遗传混合算法

(1) 编码

本文选择二进制编码。每个解是由 0 和 1 组成的符号串, 简单易行。在车辆排空问题中 1 代表选择该点作为集结站点, 0 代表不被选为集结站点。将所有车站编号, 例如, 解(0 1 0 1 0 0)代表一共有 6 个车站, 其中第 2 个车站和第 4 个车站被选为空车的集结站点。

(2) 初始化种群

初始种群指随机生成一定数量的个体形成一个初始群体, 以作为进化的基础。每个个体代表一个解。问题中假设所有站上待分配的车辆总数为 Q , 要求分配后每个集结站上最少达到的空车数量为 B , 因此最终选取的空车集结站点数最多为 $\lceil Q/B \rceil$, 即集结站点数量有上限。为了缩小算法的搜索空间, 在生成初始种群时, 控制每个个体中基因为 1 的数量不超过 $\lceil Q/B \rceil$, 直接生成符合条件的种群可以减少搜索无用个体, 加快进化效率。

根据车站数 m , 建立一个 $m \times px$ 的 0~1 (初始种群)矩阵:

$$pop = x_m, u = 1, 2, \dots, px, v = 1, 2, \dots, m$$

(3) 适应度值计算

适应度函数是评价个体好坏的标准, 一般以目标函数或者改造目标函数作为适应度函数。本文研究问题的目标是选取集结站点后空车集结过程的花费最小, 因此可以根据空车集结过程花费规定个体的适应度值计算方法。

根据实际背景, 服务网络可以看作是由集结车站和其余车站组成的完全二部图, 集结过程可以看作是其余车站向集结车站运送空车, 使得成本最小并且满足集结站点对空车数量的需求。这一过程可以利用图论中最小费用最大流的思想。因此本文采用最小费用最大流算法实现其余站点到集结站点的分配过程。适应度值的确定如下:

步骤 1 对于每个种群中的每个个体 x_m , 利用最小费用最大流算法, 计算每个个体在网络流中的费用, 即对应的集结站点选取方式下空车的集结成本 w_m 。其中最小费用最大流算法的实现参数如下:

$N = (V, E, C, W)$ 是以 V 为顶点集, E 为弧集, C 为弧上的容量, W 为弧上货运服务费用的网络。增加一个源点 s 和汇点 t , 源点和所有的供点相连, 容量为点上的空车数 a_i , 费用为 0, 汇点和所有集结站点对应的点相连, 容量为 $B - a_i$, 费用为 0, 供点和集结站点之间的容量设为 ∞ , 费用为对应服务的运输成本, 即经过此条服务运送一辆空车的成本。根据上述参数设置, 这里最小费用最大流算法计算出的是满足集结站点上最低空车数对应的费用和空车分配方案, 如果车站上还有未集结空车, 将其按照成本最小

的原则分配到相应的集结车站。整个过程的成本由单位运送费用乘以运送空车数。

步骤 2 计算种群中最大成本值 w_{\max} ，每个个体的适应度值为：

$$w'_{uv} = w_{\max} - w_{uv}$$

(4) 遗传算子的确定

遗传算子包括选择、交叉和变异。本文选择轮盘赌输法进行选择，并且加入精英保留策略。每个个体进入下一代的概率等于其适应度值与整个种群中个体适应度值之和的比例，每一代中最好的个体不参与交叉和变异，直接保留到下一代。使得最优个体不会被破坏。

交叉是通过两个个体进行部分基因的互换，从而产生新的个体。本文选择单点交叉。由于个体中 1 的个数即集结站点的个数有上限，通过交叉操作时可能会破坏这样的特点，因此在交叉时对此进行调整。如果经过交叉操作后产生的个体中基因为 1 的数量超过上限，则需要做以下的边界吸收操作：检查个体中 1 的数目，如果超过上限，则产生一个随机数，作为重新调整为 0 的个数，并通过随机产生调整的位置将该位置上的 1 变为 0。

变异是指在一定概率下，个体发生基因突变的情况。本文采取单点变异的方式，随机指定个体中某个基因发生变异。和交叉类似，在此过程中也要进行边界吸收操作。

(5) 停止条件

本文将最大迭代次数作为停止条件。

4. 数值实验

本节的数值实验均使用 MATLAB 语言实现，其中遗传算法参数设置为：种群规模为 30，交叉概率为 0.6，变异概率为 0.01，迭代次数为 300 次。

4.1. 数值实验 1

在 4 个服务网络上利用本文提出的算法进行数值实验，并与枚举法进行对比。服务费用矩阵见附录 (表 A1)，计算结果见表 1。

Table 1. Results of genetic algorithm and enumeration

表 1. 遗传算法和枚举法结果

图规模(车站数)	各站空车数	集结参数	遗传算法计算结果	枚举法计算结果
4	9, 5, 4, 6	10	1, 2	1, 2
5	11, 7, 9, 8, 5	12	1, 2, 3	1, 2, 3
8	10, 6, 7, 8, 9, 11, 13, 9	15	3, 4, 5, 6	3, 4, 5, 6
9	17, 16, 14, 12, 9, 10, 11, 15, 16	20	2, 3, 4, 5, 8, 9	2, 3, 4, 5, 8, 9

由表 1 可以看出，本文使用最小费用最大流 - 遗传算法计算得出的最优近似解和枚举法得出的精确值相同，遗传算法作为一种随机搜索算法，本文通过引入最小费用最大流算法，提高了计算结果的精度。

4.2. 数值实验 2

本节以沈阳铁路局为例，利用区段中心优化法[18]对简化后 51 个车站(见附录：图 A1)的排空问题进行计算。货运服务费用与距离、车站级别等有关，我们综合这些影响因素确定各服务上的费用。随机生成 20 组空车分布(各站上的空车数量)见附录(表 A2)，分别使用本文提出的排空方法和现有的固定空车集结站点的排空方式对以上 20 组数据计算并作对比，结果见表 2。

Table 2. Comparison of experimental results
表 2. 实验结果对比

数据	现有方法				本文方法				成本对比		
	集结站点空车数量			成本	集结站点空车数量			成本			
1	31	43	45	1020335	9	20	26	30	31	543760	46.7%
	2924	267	76		663	663	600	600	741		
2	31	43	45	920092	7	16	28	31	44	495247	46.17%
	2874	287	80		600	692	704	645	600		
3	31	43	45	989275	2	16	36	40	41	564496	42.94%
	2641	385	127		648	600	632	673	600		
4	31	43	45	1112841	7	17	28	41	47	519864	53.28%
	3112	283	59		794	741	719	600	600		
5	31	43	45	949385	2	17	30	40	47	516277	45.62%
	2659	303	115		600	600	600	677	600		
6	31	43	45	1035154	14	15	17	31	39	600686	41.97%
	2976	338	48		833	628	600	701	600		
7	31	43	45	1088642	8	17	20	35	44	553947	49.12%
	2782	322	3		707	600	600	600	600		
8	31	43	45	890168	9	17	28	30	31	446208	49.87%
	2812	218	121		614	600	665	600	672		
9	31	43	45	915997	2	12	17	30	35	523668	42.83%
	2926	239	54		600	600	600	600	819		
10	31	43	45	1146525	7	17	31	35	46	561440	51.03%
	3019	283	14		719	682	600	715	600		
11	31	43	45	983900	9	17	20	22	28	560060	43.1%
	2724	426	33		672	600	600	711	600		
12	31	43	45	1001288	7	26	31	40	46	481566	51.9%
	2970	237	3		600	669	600	698	643		
13	31	43	45	943713	7	17	20	39	49	533978	43.41%
	2655	467	5		600	600	600	600	727		
14	31	43	45	129526	2	17	20	22	30	562907	50.16%
	3006	258	15		624	692	660	703	600		
15	31	43	45	994333	9	17	35	47	49	584492	41.22%
	2878	349	32		649	600	810	600	600		
16	31	43	45	917953	7	12	26	28	42	536937	41.51%
	2566	354	108		600	600	600	628	600		
17	31	43	45	1059555	7	12	26	30	40	489855	53.77%
	2825	361	16		664	600	728	600	610		
18	31	43	45	910668	2	17	30	40	47	512141	43.76%
	2853	260	91		607	624	691	682	600		
19	31	43	45	818793	8	12	26	36	40	522625	36.17%
	2485	447	127		600	600	600	600	659		
20	31	43	45	917835	7	17	22	31	35	477062	48%
	2771	259	33		600	600	600	600	663		

由表 2 可以看出, 利用本文设计的算法优化出了合理的空车集结站点。经过分配, 每个集结站点上的空车数量达到设定的下限, 可以满足集结站点上空车快速编成列车发出的需求; 相比于现有的固定集结站点的集结方式, 降低了空车的集结成本, 改变了空车数量分布差别较大的情况。

5. 结论与展望

本文主要研究铁路空车调配过程中的车辆局外排空问题。在已知货运服务网络的基础上, 本文构建了综合考虑车辆排空成本与速度的整数规划优化模型, 并设计了针对上述模型的混合最小费用最大流算法的遗传算法, 通过数值实验说明运用该算法能够得到具有高精度性的解。进一步的, 本文通过数值实验表明与固定空车集结点的做法相比本文所采用的方法平均节约空车集结成本 35%以上。在研究车辆排空问题的同时考虑空、重车流的配合与路网运能资源合理运用是本领域的重要研究方向。

参考文献

- [1] Gorman, M.F. (2015) Empty Railcar Distribution. In: Patty, B., Ed., *Handbook of Operations Research Applications at Railroads*, International Series in Operations Research & Management Science, Vol. 222, Springer, Boston, 177-189. https://doi.org/10.1007/978-1-4899-7571-3_7
- [2] Heydari, R. and Melachrinoudis, E. (2017) A Path-Based Capacitated Network Flow Model for Empty Railcar Distribution. *Annals of Operations Research*, **253**, 773-798. <https://doi.org/10.1007/s10479-016-2259-4>
- [3] 温鑫. 我国铁路空车调整计划优化研究[D]: [硕士学位论文]. 兰州: 兰州交通大学, 2017.
- [4] 程学庆. 铁路空车调配综合优化模型及求解[J]. 中国铁道科学, 2012, 33(6): 117-121.
- [5] 程学庆, 陆一新, 尹传忠, 等. 基于时间窗的铁路空车调配优化模型及求解[J]. 中国铁道科学, 2007, 28(6): 113-116.
- [6] Spieckermann, S. and Voß, S. (1995) A Case Study in Empty Railcar Distribution. *European Journal of Operational Research*, **87**, 586-598. [https://doi.org/10.1016/0377-2217\(95\)00232-4](https://doi.org/10.1016/0377-2217(95)00232-4)
- [7] 林柏梁, 乔国会. 基于线路能力约束下的铁路空车调配迭代算法[J]. 中国铁道科学, 2008, 29(1): 93-96.
- [8] 朱海豹. 基于路段的空车调配优化模型研究[D]: [硕士学位论文]. 北京: 北京交通大学, 2008.
- [9] Holmberg, K., Joborn, M. and Lundgren, J.T. (1998) Improved Empty Freight Car Distribution. *Transportation Science*, **32**, 163-173. <https://doi.org/10.1287/trsc.32.2.163>
- [10] Holmberg, K., Joborn, M. and Melin, K. (2008) Lagrangian Based Heuristics for the Multicommodity Network Flow Problem with Fixed Costs on Paths. *European Journal of Operational Research*, **188**, 101-108. <https://doi.org/10.1016/j.ejor.2007.04.029>
- [11] 梁栋, 林柏梁, 严贺祥, 等. 车种代用情况下的铁路空车调配研究[J]. 铁道学报, 2005, 27(4): 1-5.
- [12] 苏婕. 铁路车流推算方法研究[D]: [硕士学位论文]. 成都: 西南交通大学, 2018.
- [13] 王海英, 黄强, 李传涛. 图论算法及其 MATLAB 实现[M]. 北京: 北京航空航天大学出版社, 2010: 126-130.
- [14] 韩瑞峰. 遗传算法原理与应用实例[M]. 北京: 兵器工业出版社, 2010: 49-71.
- [15] 陶泽, 谢里阳, 郝长中, 等. 基于混合遗传算法的车间调度问题的研究[J]. 计算机工程与应用, 2005, 41(18): 19-22.
- [16] 王悦. 遗传算法在函数优化中的应用研究[J]. 电子设计工程, 2016, 24(10): 74-76.
- [17] 吴瑞节, 孙鹏, 孙昱, 等. 自适应量子遗传算法在指挥控制结构设计中的应用[J]. 计算机应用研究, 2017, 34(7): 2045-2048.
- [18] 果鹏文, 林柏梁, 余洋. 大规模路网上空车调配的区段中心优化法[J]. 中国铁道科学, 2001, 22(2): 122-128.

附录

Table A1. A service cost matrix

表 A1. 服务费用矩阵

车站	1	2	3	4	5	6	7	8	9
1	0	4	5	3	6	9	4	7	3
2	4	0	4	3	2	4	5	6	7
3	5	4	0	6	6	7	4	3	8
4	3	3	6	0	5	4	5	3	7
5	6	2	6	5	0	3	6	7	8
6	9	4	7	4	3	0	4	3	4
7	4	5	4	5	6	4	0	2	5
8	7	6	3	3	7	3	2	0	7
9	3	7	8	7	8	4	5	7	0

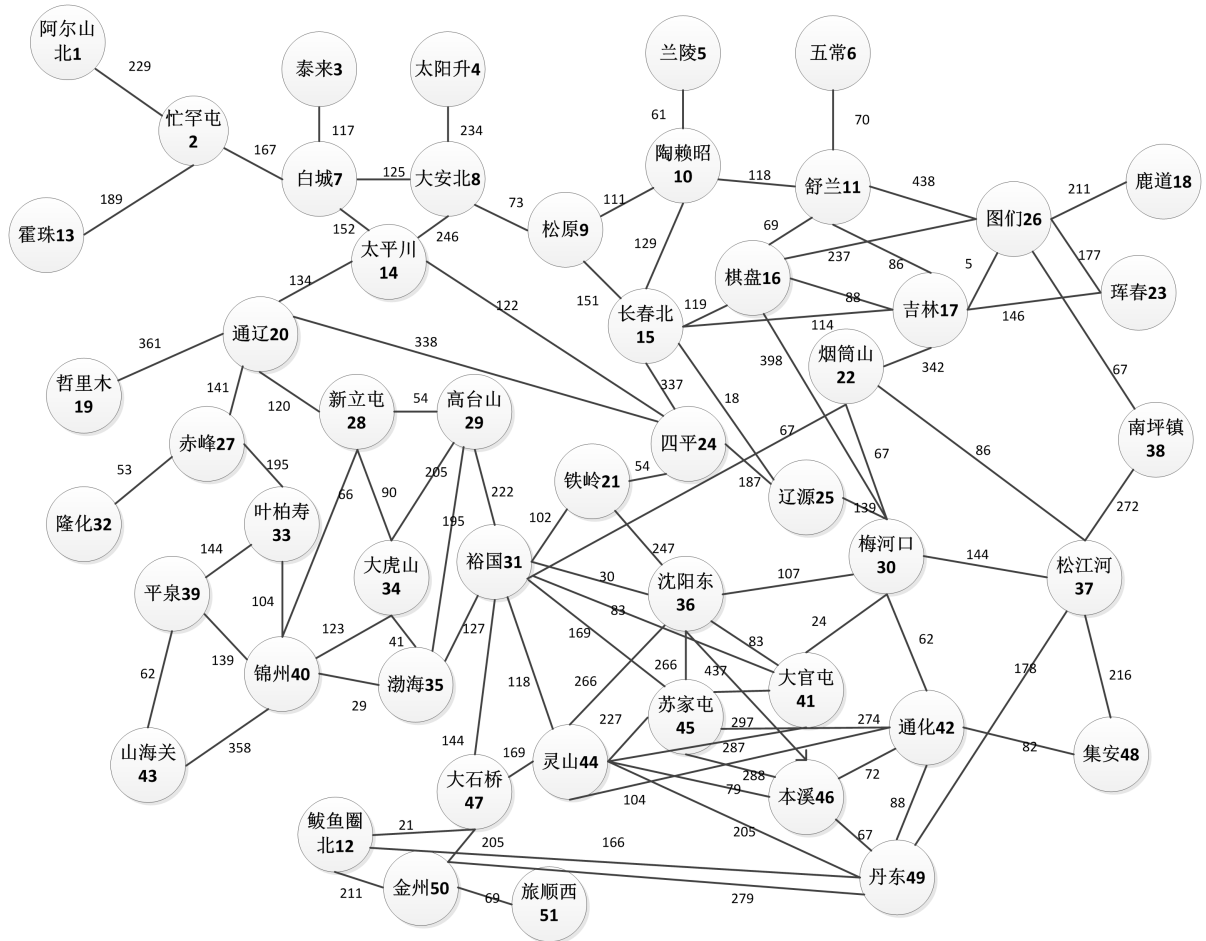


Figure A1. Network map of Shenyang railway bureau

图 A1. 沈阳铁路局路网图

Table A2. Empty train distribution at each station
表 A2. 各站空车分布

数据	各站空车数量(辆)
1	40, 62, 30, 21, 85, 53, 33, 94, 103, 4, 71, 49, 75, 30, 17, 127, 5, 87 112, 91, 123, 72, 51, 89, 30, 120, 110, 5, 84, 32, 71, 99, 12, 95 76, 53, 83, 8, 43, 12, 103, 100, 3, 49, 76, 101, 106, 41, 93, 119, 19
2	5, 114, 27, 59, 15, 71, 121, 29, 55, 77, 95, 13, 5, 91, 34, 100, 69 104, 99, 18, 103, 114, 7, 122, 102, 65, 30, 87, 78, 126, 31, 5, 72, 75 128, 20, 119, 36, 122, 34, 9, 73, 58, 54, 80, 124, 12, 24, 42, 43, 45
3	89, 74, 6, 74, 93, 67, 29, 58, 19, 98, 111, 58, 127, 96, 95, 129, 121 22, 3, 127, 100, 22, 26, 30, 15, 30, 50, 127, 82, 3, 130, 20, 130, 83 14, 6, 35, 38, 75, 10, 43, 20, 110, 5, 127, 55, 27, 73, 90, 49, 32
4	81, 16, 118, 90, 34, 125, 99, 81, 110, 125, 121, 94, 48, 101, 7, 127, 82 53, 9, 54, 61, 21, 72, 13, 30, 81, 27, 109, 87, 41, 110, 60, 48, 108 57, 42, 80, 46, 100, 12, 60, 89, 48, 76, 59, 63, 50, 45, 55, 8, 121
5	62, 79, 69, 48, 42, 65, 108, 17, 35, 33, 42, 103, 78, 37, 113, 3, 20 79, 79, 76, 21, 127, 111, 72, 25, 83, 28, 56, 111, 54, 49, 38, 129, 50 64, 42, 26, 17, 46, 25, 112, 119, 62, 3, 115, 81, 31, 17, 51, 28, 96
6	120, 109, 65, 28, 94, 16, 12, 4, 130, 110, 110, 104, 19, 115, 47, 121, 57 129, 65, 56, 69, 93, 7, 107, 116, 27, 4, 90, 26, 16, 28, 103, 26, 109 129, 41, 89, 25, 109, 42, 66, 4, 96, 81, 48, 56, 55, 37, 49, 37, 64
7	62, 101, 19, 78, 123, 70, 32, 101, 125, 81, 10, 30, 116, 51, 109, 50, 10 57, 46, 90, 7, 5, 114, 59, 65, 103, 90, 73, 103, 67, 13, 48, 58, 17 112, 24, 77, 81, 27, 66, 64, 13, 99, 41, 3, 124, 71, 13, 4, 71, 34
8	3, 17, 118, 70, 99, 6, 101, 17, 5, 61, 67, 54, 11, 19, 38, 86, 102 112, 63, 115, 111, 50, 55, 31, 127, 31, 48, 105, 18, 54, 120, 33, 64, 7 86, 70, 44, 109, 11, 80, 123, 118, 62, 125, 121, 15, 14, 58, 25, 23, 49
9	107, 47, 48, 35, 15, 88, 63, 9, 44, 48, 12, 102, 22, 38, 85, 80, 43 49, 21, 45, 78, 71, 24, 62, 130, 130, 24, 93, 127, 61, 27, 65, 7, 89 130, 37, 118, 4, 77, 130, 17, 38, 66, 62, 54, 120, 70, 7, 62, 119, 119
10	70, 74, 87, 62, 8, 118, 102, 120, 20, 32, 98, 49, 121, 63, 87, 115, 51 88, 115, 18, 21, 130, 128, 33, 17, 4, 20, 93, 20, 36, 96, 67, 52, 42 61, 112, 78, 114, 120, 97, 31, 123, 24, 13, 14, 45, 30, 32, 19, 96, 50
11	66, 102, 4, 6, 129, 15, 90, 112, 129, 128, 5, 60, 6, 95, 110, 12, 126 94, 113, 49, 26, 64, 7, 40, 53, 40, 30, 33, 66, 3, 113, 115, 84, 55 67, 5, 110, 17, 95, 123, 10, 93, 102, 11, 33, 50, 64, 14, 114, 56, 39

Continued

12	41, 26, 96, 55, 116, 51, 98, 30, 71, 9, 118, 109, 29, 14, 21, 122, 71 69, 86, 19, 13, 82, 86, 46, 110, 95, 53, 130, 33, 69, 37, 48, 37, 111 61, 119, 24, 52, 108, 32, 104, 79, 95, 23, 5, 86, 124, 105, 126, 75, 13
13	18, 116, 62, 54, 37, 49, 128, 38, 61, 36, 56, 55, 4, 25, 128, 105, 61 84, 119, 13, 16, 57, 15, 44, 27, 38, 97, 11, 8, 29, 33, 55, 112, 105 61, 119, 24, 52, 108, 32, 104, 79, 95, 23, 5, 86, 124, 105, 126, 75, 13
14	7, 124, 120, 42, 102, 26, 37, 61, 125, 11, 45, 73, 108, 27, 40, 80, 118, 118, 113, 107, 130, 21, 88, 56, 128, 82, 79, 105, 23, 70, 9, 71, 11, 6, 10, 70, 121, 29, 49, 62, 41, 103, 48, 9, 15, 128, 45, 11, 72, 75, 28
15	47, 52, 107, 4, 49, 6, 19, 17, 93, 108, 86, 79, 26, 26, 55, 13, 67 108, 128, 71, 48, 63, 117, 31, 89, 122, 66, 36, 95, 54, 32, 18, 60, 41 130, 120, 62, 38, 89, 63, 5, 46, 116, 56, 32, 57, 97, 57, 80, 84, 94
16	14, 28, 3, 16, 19, 76, 38, 25, 120, 47, 55, 87, 112, 34, 100, 68, 26 27, 105, 15, 66, 53, 23, 119, 7, 120, 77, 7, 49, 5, 7, 71, 123, 90 29, 81, 92, 87, 43, 44, 99, 84, 40, 48, 108, 35, 31, 18, 119, 121, 117
17	64, 70, 105, 34, 126, 88, 54, 77, 99, 22, 19, 123, 47, 38, 130, 59, 112 121, 10, 29, 24, 16, 36, 57, 42, 112, 57, 13, 59, 7, 52, 106, 56, 18 17, 26, 104, 125, 114, 52, 120, 34, 28, 28, 16, 76, 103, 119, 80, 16, 62
18	109, 17, 85, 57, 24, 73, 25, 4, 14, 96, 69, 111, 24, 121, 39, 96, 105 63, 5, 76, 16, 45, 27, 5, 56, 5, 34, 84, 130, 66, 110, 7, 76, 111 74, 40, 97, 90, 61, 18, 101, 28, 82, 79, 91, 39, 96, 83, 52, 86, 102
19	57, 57, 36, 56, 4, 93, 6, 97, 55, 45, 55, 109, 29, 56, 112, 4, 8 85, 15, 45, 22, 7, 41, 30, 15, 16, 120, 10, 130, 43, 90, 25, 88, 126 75, 123, 40, 50, 114, 128, 55, 48, 100, 105, 127, 60, 59, 47, 27, 108, 6
20	42, 70, 24, 80, 36, 86, 91, 98, 60, 13, 32, 119, 22, 108, 71, 130, 13 59, 16, 126, 3, 102, 107, 114, 13, 54, 36, 105, 58, 119, 26, 36, 21, 20 114, 77, 73, 21, 112, 82, 47, 68, 54, 12, 33, 18, 26, 33, 56, 9, 118
