

文章编号: 1005-8451 (2015) 08-0037-06

# 城市轨道交通列车运行图自动生成与CAD转换

王成, 蔡哲扬, 李容

(西南交通大学 信息科学与技术学院, 成都 610031)

**摘要:** 基于人工模拟法编制列车运行图, 保证每个时间段的对数满足设计要求的同时, 根据虚拟链接及调整, 使其满足列车折返约束, 之后根据列车出入库的情况进行调整, 使生成的运行图更好地符合设计需求, 能够高效地对不同线路的不同全日行车计划进行铺画。另外, 将生成的运行图转换为CAD脚本, 也减少了设计者的劳动强度。以深圳地铁6号线为例实现全日运行图的铺画, 表明该方法是有效可行的。

**关键词:** 运行图铺画; 虚拟连接; 列车出入库; 自动调整; CAD转换

**中图分类号:** U231.7 : TP39 **文献标识码:** A

## Automatic generation of train diagram and CAD conversion for Urban Transit

WANG Cheng, CAI Zheyang, LI Rong

(School of Information Science & Technology, Southwest Jiaotong University, Chengdu 610031, China)

**Abstract:** The train diagram which was drawn up by using artificial simulation method, could meet the design requirements of train numbers for each time period, satisfy the train reverse constraint with virtual link and adjustment. According to the situation of train in-out stock, the diagram was adjusted further to meet the needs of designers. Different full-time train operation plan to different line could be drawn. At the same time, the train diagram could be translated into CAD script, the labour intensity could be reduced greatly. Finally Line 6 of Shenzhen Metro was taken as an example to implement the drawing of train diagram, show the method feasibility and effectiveness.

**Key words:** draw of train diagram; virtual connection; train in-out stock; automatic adjustment; CAD conversion

列车运行图是列车运行的基础, 是轨道交通运营企业实现列车安全、正点运行, 有效组织轨道交通运输工作的重要手段, 对保证运营企业各部门有效配合起到了决定性的作用<sup>[1]</sup>。随着城市的发展, 近些年越来越多的城市进行城市轨道交通的建设。由于城市轨道交通线路众多, 通用的运行图编辑器可以达到提高时刻表运行图的编制工效、运行图的编制质量、减轻运行图编制人员的劳动强度等。

目前, 国内外对计算机编制运行图的研究方法各不相同。文献[2]以时空域滚动计算方法实现单线实用货物列车运行图计算机编制系统, 由于在城市轨道交通中线路一般为双线, 因此无法完全照搬引用; 文献[3]运用遗传算法对列车运行图初始布点进行了研究, 但由于城市轨道交通的运行环境是城市地区, 线路客流随节假日的变化波动较大, 该算法不能较高效率地满足城市轨道交通的运行需求; M.A.Odijk

根据PESP(周期事件安排问题)研究了编制一个车站周期列车时刻表的方法<sup>[4-7]</sup>, 该方法假定所有列车在车站的到站都是周期循环发生的, 但在城市轨道交通中根据旅客流量全天的运行分为高峰时段和平峰时段, 两个时段列车的周期并不相同, 该算法并不能满足不同周期同时存在的需求。在后期运行图的编辑、调整和加工方面CAD系统可以弥补本软件的弱点, 因此在计算机实现运行图绘制之后将其转换为CAD脚本可以方便设计者对运行图的后期编辑<sup>[8]</sup>。

本文通过提取人工铺画运行图的思路及经验, 在保证每个时间段开行列车对数的前提下对生成的运行图进行虚拟连接及调整, 之后根据列车出入库的情况进行进一步调整, 使运行图满足需求。

### 1 列车运行图数学模型

#### 1.1 运行线时间确定

目前, 大部分的城市轨道交通没有实行快慢车的硬件基础, 因此大部分的城市轨道交通系统不存

收稿日期: 2014-12-25

作者简介: 王成, 在读硕士研究生; 蔡哲扬, 在读硕士研究生。

在车辆的越行和会让，车辆在区间的运行时分相对固定，每个车站的站停时分也是相对固定的<sup>[9]</sup>。因此，车辆的到站时间完全取决于列车初始发车时间。假设  $t_{ij}^q$  表示列车在车站 i 到车站 j 区间的区间运行时分； $t_i^t$  表示列车在 i 站的停站时间；列车的初始发车时间为  $t_{begin}$ ；则列车到达 k 站的到站时间为  $t_k^a$  为：

$$t_k^a = \sum_{i=0}^{k-1} t_i^t + \sum_{i=1}^k t_{i-1,i}^q + t_{begin}, \quad k \in N^+, \quad i \in N^+ \quad (1)$$

由式 (1) 可进一步计算出列车的上下行、大小交路各个运行线的详细到站时间及出站时间。

## 1.2 列车运行图数学模型

在铺画列车运行图时最为关键的是列车折返时所要满足的约束条件和列车的出入库问题。根据上一小节所得到的运行线的详细数据以及以上两个约束条件可以把铺画列车运行图简化为 3 站图的问题，如图 1 所示。

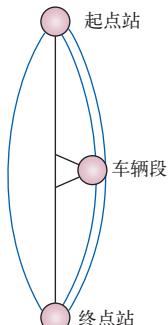


图1 线路抽象示意图

在铺画初始运行线时，初始布点的确定是极为重要的。在铺画运行图时，为了保证地铁运营时每个车站的列车追踪间隔都满足列车全日行车计划，运行图的铺画一般会提前铺画，因此若地铁的运营开始时间为  $t_f$ ；列车在车站 i 到车站 j 区间的运行时分为  $t_{i,j}^q$ ；列车在 i 站的停站时间为  $t_i^t$ ；列车运营线路为车站 m 到车站 k；列车在该线路上的运行总时间为  $t_{m,k}^z$ ，则初始发车时间  $t_{begin}$  的计算方式如下：

$$t_{begin}^z = \sum_{i=m}^{k-1} t_i^t + \sum_{i=m}^{k-1} t_{i,i+1}^q \quad (2)$$

$$t_{begin} = t_f - t_{m,k}^z, \quad i, m, k \in N^+$$

由上述方法得到第一条运行线的初始布点时间后，根据该时间段的追踪间隔进行后续运行线的铺画。

## 2 自动生成运行图的算法

### 2.1 系统流程图

(1) 得到用户提供的车站数据，区间运行数据、停站时分、站间距等线路基础数据。

(2) 根据全日行车计划在对应的时间段内进行初始运行线铺画。

(3) 对运行线进行虚拟折返连接，判断是否满足约束条件，若不满足则对所有的运行线进行平移操作，然后重复 (3)；若满足条件则继续下一步。

(4) 对所有的运行线进行连接，得到列车的出入库信息并对其进行出入库操作。

(5) 进行出去库操作后判断是否满足高峰时段行车对数，若需要则判断设定需要过渡的时间，确定过度等级，然后重复 (2) ~ (5)；若不需要则结束。

系统流程图如图 2 所示。

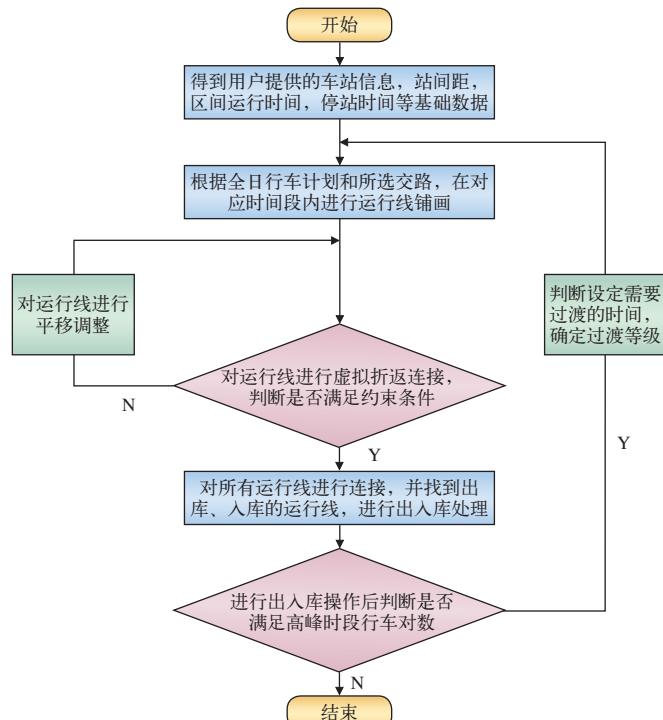


图2 系统流程图

### 2.2 运行线初始布点

铺画原则是首先满足高峰时期的开行列车对数，兼顾效率。根据用户设定的交路和全日行车计划对每个时间段进行线路的铺画，为了保证在初始运营时刻每个站的列车到发时间都满足该时间段的要求，绘制的时候需要提前进行绘制，假设一条地铁线路的

初始运营时间为 6:00，为了保证在地铁开放运营时刻每个站的列车到达时间均满足该时间段的追踪间隔，在铺画运行线时需要提前进行铺画，如图 3 所示。若下行线 1 与上行线 2 的交点在 m 站，列车追踪间隔为  $t_{zz}$ ，则运行线初始铺画的时间  $t_{iq}$  的计算方式如式 (3)。为了保证在 6:00 时起点站和终点站均有车发出，提前的时间为列车追踪间隔的整数倍。

$$t_{iq} = t_{1,m}^z - t_{1,m}^z \bmod t_{zz} + t_{zz} \quad (3)$$

同理为了保证高峰小时的对数满足条件，在高峰时段之前就应该以高峰时段的对数来进行铺画。

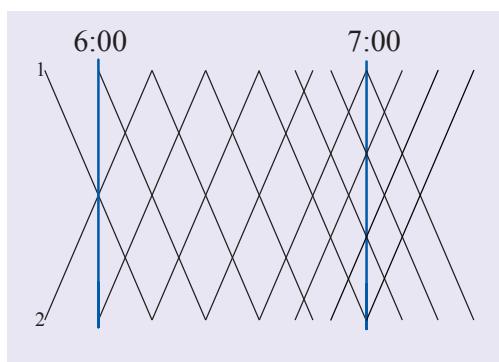


图3 运行线初始布点

### 2.3 虚拟连接

在进行完运行线的初始布点之后需要将符合折返时间约束的运行线连接起来以保证列车的合理运用，但是这样铺画的运行图大部分时候并不能满足折返时间的约束限制，会出现以下情况，因此系统需要对其进行虚拟连接，若成功则连接，若不成功则平移单个或局部运行线，使其符合折返约束。

城市轨道交通列车折返站的辅助配线型式与站台型式（如岛式站台、侧式站台）、周边环境（如临近车辆段、停车场等）等密切相关，不同的车站型式和周边环境会产生不同的折返约束<sup>[10]</sup>。假定折返模式为站后交叉渡线折返折返时间最少应为 180 s。

在虚拟连接时遇到的情况有很多种，下面介绍具体可能遇到的情况。

#### (1) 与第 1 条出站运行线的间隔大于 180 s

假设  $t_i^a$  表示第  $i$  条运行线到达折返站的时间； $t_i^d$  表示第  $i$  条运行线离开折返站的时间。如图 4 (a) 所示，若运行线 1 无连接并且运行线满足以下约束条件，则运行线 0 为入库线路；运行线 2 与运行线 1 相连接；

运行线 3 与运行线 4 相连。这种约束出现情况为高峰转低峰，所要满足的约束条件如下所示。

$$\begin{cases} t_1^a - t_0^d > 180 s \\ t_1^a - t_2^d > 180 s \\ t_3^a - t_4^d > 180 s \end{cases} \quad (4)$$

若运行线 1 无连接并且满足如图 4 (b) 所示约束条件，则运行线 0 与运行线 1 相连，所要满足的约束条件如下所示。

$$\begin{cases} t_1^a - t_0^d > 180 s \\ t_1^a - t_2^d < 180 s \end{cases} \quad (5)$$

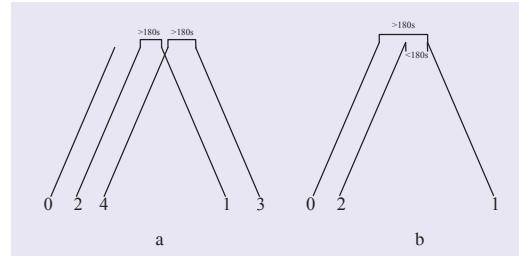


图4 第1种情况下运行线1无连接

若运行线 1 已有连接并且满足如图 5 所示约束条件，则运行线 0 与运行线 3 相连接，所要满足的约束条件如下所示。

$$\begin{cases} t_1^a - t_0^d > 180 s \\ t_1^a < t_2^d \\ t_3^a - t_2^d < 180 s \end{cases} \quad (6)$$

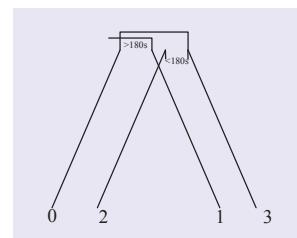


图5 第1种情况下运行线1有连接

#### (2) 与第 1 条出站运行线的间隔小于 180 s，与第 2 条出站运行线的间隔大于 180 s

如图 6 (a) 所示，若运行线 1 无连接，则运行线 0 与运行线 3 相连，运行线 1 为出库线路，所要满足的约束条件如下所示。

$$\begin{cases} t_1^a - t_0^d < 180 \text{ s} \\ t_1^a < t_2^d \\ t_3^a - t_0^d > 180 \text{ s} \end{cases} \quad (7)$$

若出现如图 6 (b) 所示情况, 则需要对一个方向的运行线进行局部平移才能满足列车在折返站的折返约束, 平移的时间为运行线 2 到站时间与运行线 1 出站时间之差, 若平移后运行线 1 到站时间与运行线 3 的出站时间之差小于 180 s 认定此类情况下的全日行车计划无法铺画。此情况的约束条件如下。

$$\begin{cases} t_1^a - t_0^d > 180 \text{ s} \\ t_1^a > t_2^d \\ t_3^a - t_0^d > 180 \text{ s} \end{cases} \quad (8)$$

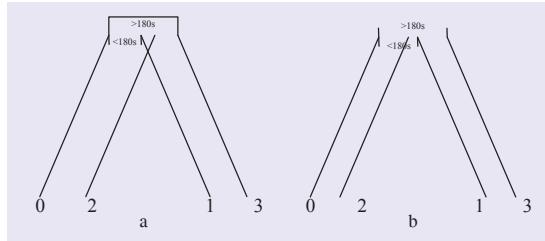


图6 第2种情况下运行线1无连接

若运行线 1 有链接, 如图 7 (a) 所示, 则运行线 2 与运行线 3 相连接, 运行线 0 为入库线路, 出现情况为高峰转低峰, 所要满足的约束条件如下所示。

$$\begin{cases} t_1^a - t_0^d < 180 \text{ s} \\ t_3^a - t_2^d > 180 \text{ s} \end{cases} \quad (9)$$

若出现图 7 (b) 所示情况, 则运行线 3 与运行线 0 相连接, 所要满足的约束条件如下所示。

$$\begin{cases} t_1^a - t_0^d < 180 \text{ s} \\ t_1^a > t_2^d \\ t_3^a - t_0^d > 180 \text{ s} \\ t_3^a - t_2^d < 180 \text{ s} \end{cases} \quad (10)$$

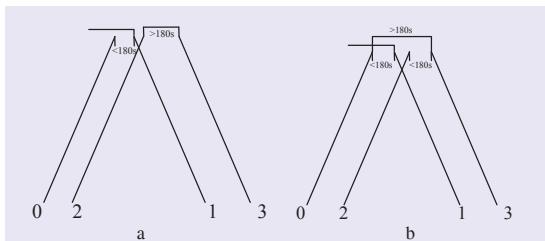


图7 第2种情况下运行线1有连接

(3) 与前两条出站运行线的间隔都小于 180 s, 与第 3 条出站运行线的间隔大于 180 s

如图 8 所示, 若遇到如下情况选择平移上行方向的运行线使其转化为情况 2 再进行判断, 平移后运行线 0 与运行线 1 的间隔刚好为 180 s, 此情况的约束条件如下。

$$\begin{cases} t_1^a - t_0^d < 180 \text{ s} \\ t_3^a - t_0^d < 180 \text{ s} \\ t_5^a - t_0^d > 180 \text{ s} \end{cases} \quad (11)$$

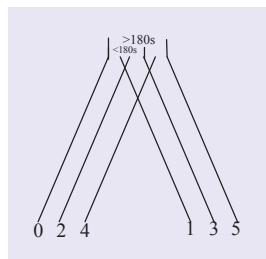


图8 第3种情况

以上为起点站折返约束的情况, 终点站折返约束的情况原理上与上方一样。

在进行判断以后, 需要进行连接的列车都会进行标示, 在判断成功以后根据这些标示对列车进行实际连接。

若折返模式为站前交叉渡线折返, 不光要满足上述条件, 另外相邻出发运行线和到达运行线一般需满足 2 min 的时间间隔。

## 2.4 列车出入库

在进行连接之后, 列车的出入库位置为起点站和终点站, 由于大部分的接轨站位置并不是设定在起点站和终点站, 因此需要进行列车的出入库操作。方法为: 遍历所有车次一遍, 对每辆车添加出入库的运行线。添加出入库运行线的时候需要考虑接轨站的位置以及发车方向, 过晚的出车和过早的抽车会影响到列车的实际开行对数。另外还要考虑接轨站存车能力的限制和同一接轨站出入库数量一致的问题。还需要根据出车的先后顺序对上下行列车。

由于线路的接轨站设置位置不同, 接轨站与正线的接轨方式分为终端接轨和中部接轨<sup>[11~14]</sup>, 因此在低峰转高峰或高峰转低峰时, 不同的接轨站位置也改变着列车的出入库位置, 这样过晚的出车和过

早的抽车会影响到列车的实际开行对数。在这种情况下可以添加高低峰过渡来对高峰时段的列车进行提前铺画或延长铺画以保证列车的出入库满足条件。

## 2.5 高低峰时段的过渡

在铺画全日运行图时,由于存在高低峰列车运行对数的差异,低峰至高峰需要从接轨站加车;高峰转低峰需要从接轨站收车。在高低峰转换时间段内会出现加车时间过晚或者抽车时间过早而导致不满足全日计划的情况。过渡的添加可以使加车提前,抽车延迟,以满足全日行车计划中的高峰段行车要求,但是在某种程度上会使低峰段的行车对数提高,从而影响效率。

本模块的主要功能就是在全日计划高低峰过渡的时间段内,满足高峰小时的行车对数的前提下,尽量兼顾效率。本模块主要流程如图9所示。

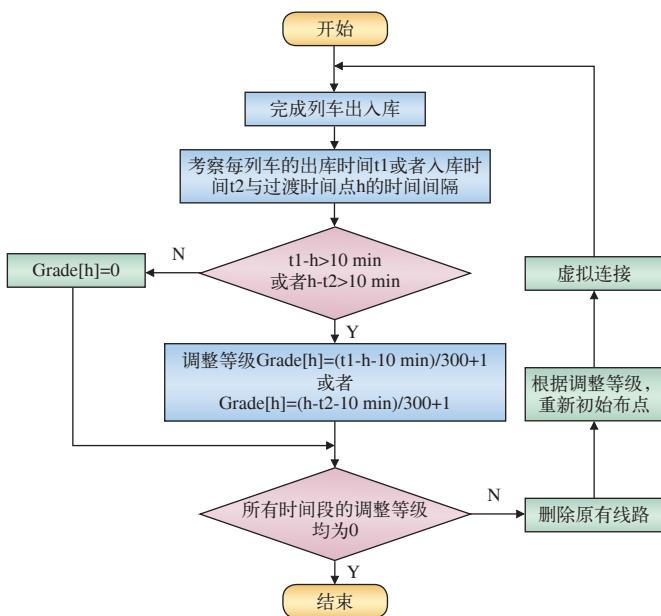


图9 过渡模块流程图

调整等级的作用是用来将高峰时间往低峰方向提前铺画一段时间,提前的时间由调整等级来决定。目前在程序中,提前的时间T和调整时间的关系如下:

$$T=m\_level[h]*300 \text{ (s)},$$

其中,  $m\_level[h]$  表示在  $h$  小时下的过渡等级。

## 3 CAD转换及系统实现

### 3.1 CAD转换

在系统生成运行图之后,为了方便设计者对其

进行进一步的编辑加工,系统可以将生成的运行图转换为CAD脚本文件导出。

在CAD脚本中“LINE a,b c,d”表示横坐标为a,纵坐标为b的点到横坐标为c,纵坐标为d的一条直线。“color 252”表示用颜色252对改线段进行绘制。“PLINE a,b c,d e,f,...”为多段线的绘制方法<sup>[15]</sup>。

根据系统生成的运行图数据以及用户输入的线路基本数据可以运用CAD脚本语言对其进行CAD脚本的转换。

### 3.2 系统实现

以深圳地铁6号线为例,表1为深圳地铁6号线的一张全日行车计划,下面根据表中的行车计划进行列车初始点铺画。

表1 深圳6号线全日行车计划

时段	对数
6:00~7:00	15
7:00~8:00	20
8:00~9:00	20
9:00~10:00	15
10:00~11:00	10
11:00~12:00	10
12:00~13:00	10
13:00~14:00	10
14:00~15:00	10
15:00~16:00	10
16:00~17:00	15
17:00~18:00	20
18:00~19:00	20
19:00~20:00	15
20:00~21:00	10
21:00~22:00	8
22:00~23:00	8
23:00~24:00	6
合计(对/日)	232

根据CAD转换规则将生成的运行图转化为CAD脚本文件,CAD运行图如图11所示。

## 4 结束语

基于人工模拟法编制列车运行图,首先保证了每个时间段的对数满足设计要求,根据虚拟及调整,使其满足

列车折返约束条件,之后根据列车出入库的情况进行进一步调整,使生成的运行图符合设计者的需求。本系统能够高效地对不同线路的不同全日行车计划进行铺画。而将生成的运行图转换为CAD脚本也极大地减少了铺画人员的劳动强度,方便运行图的后期编辑,十分具有实际意义。下一步的工作将在本系统的基础上,进行快慢车模式下运行图的自动铺画算法研究。

(下转 P45)