

城市主干道红绿灯配时优化与仿真程序

黄亚坤 丁润泽 赵治文 吴铁鹰

(南京航空航天大学能源与动力学院 江苏 南京 210016)

【摘要】城市路口设置红绿灯的作用是使交通流在路口能够有序地通过,而城市主干道则是城市交通的主要承担者。但传统的上下行同步红绿灯常常导致“绿灯无车红灯堵”、主干道堵车越来越严重等现象,车辆的能耗和尾气排放无谓增加,造成资源浪费。本文介绍一种采用“十字路口不同步红绿灯配时”方案,使得“城市主干道不间断通行”、总车辆等待时间短,以疏通城市车辆交通、达到节能减排目的。为此,我们编写了面向对象模拟软件,通过建模和仿真,比较分析了主干道以及单行道分别采用错时红绿灯和同步红绿灯时,交通通行能力、能耗和排放。程序仿真证明:主干道错时红绿灯系统令道路通行能力显著提高,使得汽车能耗和尾气排放相应减少,对改善交通和城市环境具有较大价值。

【关键词】面向对象;红绿灯配时;减排;模拟仿真

1. 研制背景及意义

城市交通管理要比航空和铁路运输管理复杂得多。因为航班、列车的运行时间、出发地、终点、甚至当前位置都可以知道,而城市交通,被管理的目标有各种机动车、非机动车、还有行人,关键是他们个体的出行精确数据无法得到,这就给准确的数值模拟带来了很大的困难。大部分研究中都采用了连续数学函数建模的方式,甚至用于规划模拟整个城市的交通管理。目前,世界上有一些相关模拟软件,如美国的schwerdtfeger。国内交警部门引进后发现完全不适用于国内情况。在说中国所有大城市的车辆拥有量过大,任何全城模拟管理算法都很难有较好的效果。所以,我们认为:应该综合治理城市中的主干道的交通,主干道流量大,如果通畅,车辆无用排放总量就小。

本项目基于主干道承担城市主要交通流量的原则,针对城市的主干道建立交通流量模型,通过主干道交通流量调节支路交通流量;同时采用各个十字路口红绿灯错时算法和设置单行道配合主干道,使所有参与该干道的车辆遇红灯等待的时间总和较小,并提高道路利用率。配合计算机科学理论,形成市内主干道交通管理、错时红绿灯、单行道合理设置等理论,通过对数据的统计和分析建立面向对象微观仿真模型,从而编写出相应的仿真程序,进而验证了该方案具有减少等待时间、使流量最大化的效果。

2. 设计方案

将一个城市分成 n 个模块,分别用主干道和单行道进行连接。通过设计错时红绿灯配时,达到城市主干道不间断通行的目标。以主干道为核心,提高主干道的通流量,从而缓解整个城市的交通拥堵问题。主要实施手段是软件模拟:通过对主干道的调研、优化,采用微观交通仿真模型,以车辆为点、道路为轴,进行算法设计及软件编写。

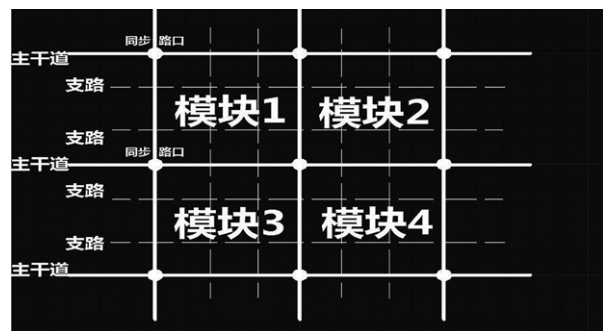


图 2.1 n 个模块井字形分布

2.1 道路说明

程序中的道路为城市主干道。此类道路多为双向多车道(上下行道路间有较宽的隔离带),在此等价于平行但方向相反的两个单行道。

单行道有多个十字交叉路口,主干道上的车流量要远远大于的支路车流量。

以南京市仙林大道为例,建立如图 2.1 所示坐标系:

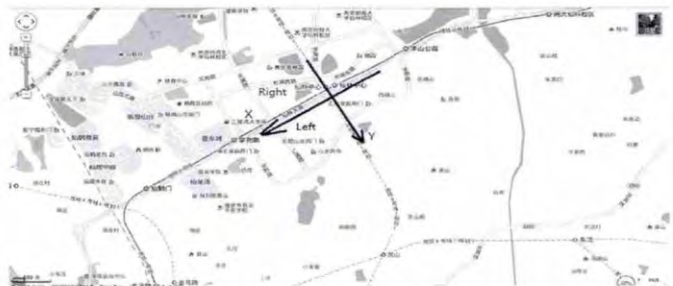


图 2.2 南京市仙林大道

2.2 车辆简化

程序会在各路口按规定时间生成车辆;由支路驶出和驶入主干道的车流量近似相等,因此忽略转弯。

2.3 主干道

设计时主干道按井字形分布。针对同一条主干道,对比红绿灯错时和同步时的车辆等待时间。下图所示为主干道红

绿灯示意图,各红绿灯分别控制不同方向的车流。

2.4 延迟时间

以两条主干道相交的十字路口为起点,即不同模块的交点处,设此处红绿灯同步且红绿灯的时间比为 $a_0:b_0=1$,然后分别根据双向车流量以及到该路口到下个路口的距离进行错时调节:到达第一个红绿灯 H_1 时,假设其红绿灯的时间比为 $a_1:b_1$,一次进行推算,直到第 n 个红绿灯 H_n 的时间比再次恢复同步,即 $a_n:b_n=1$ 。这样做可以实现不间断通行(即“绿波原理”),以达到使车流量最大的目的。详细计算过程如下:

我们假设每一个路口距起点路口距离分别为 $X_1、X_2、X_3……X_n$; 车辆恒定速度 v ; 车辆到达各路口耗时依次为 $\Delta T_1、\Delta T_2、\Delta T_3……\Delta T_n$, 则车辆到达第一个路口耗时 $\Delta T_1=X_1/v$; 车辆到达第二个路口耗时 $\Delta T_2=X_2-X_1/v$; 以此类推,车辆到达第 n 个路口耗时 $\Delta T_n=X_n-X_{n-1}/v$; 总计 $\Delta T=\Delta T_i=X_n/v$ 。

因为往返的情况相同,可知当距离为 $2X_n$ 时,往返正好同步,即模块交点处同步。

以仙林大道为例,我们对道路以及现行交规进行改善。在模块交点处十字路口的红绿灯同步计时,而其间的所有的十字路口上下行红绿灯错时。由于上下行车辆到达同一路口所走的路程不相同,但主干道车速均匀且相等,故上下行车辆到达同一个十字路口所经历的时间不同,从而产生时间差 ΔT 。利用这时间差 ΔT ,实现支路车辆进入和驶出路口隔离带处的等待区,使支路车辆分步穿过主干道,牺牲支路车辆的通畅,借此保证主干道的畅通,从而实现主干道通流最大化和等待时间最小化。

2.5 支路调节模式

2.5.1 支路车辆过多

针对上下班高峰期和周末道路车辆过多的情况:在上下班以及周末时间,直接设置管理支路横穿的两个红绿灯为永久红灯,禁止通行。使支路车辆混入主干道并在下一个十字路口左转通行,从而错十字路口穿过主干道。

2.5.2 支路车辆少

当车辆少的时候,支路车辆主要利用主干道绿波通行的车流间隔通行。即主干道为不间断通行,支路车辆依靠主干道红灯时间分两次通行。

3. 软件设计与工作原理

面向对象设计思想比较适合我们的现实世界,我们的世界就是由各种实际的或抽象对象(Objects)组成的,因此面向对象编程语言被广泛地用于软件开发。

在本项目的软件设计中我们根据对实际情况的观察、分析,用类(Class)较真实地描述了城市道路交通的各个单元。

3.1 类的设计

3.1.1 路 CRoad

也就是所要研究的主干道,它由若干个路口组成,每个路口有它的坐标(图 3.1)。这些坐标值也就定义了该干道的长度,同时坐标系也用于给车辆定位。虽然,从数学的观点,这只是一维坐标,但它可以模拟弯曲的主干道,也就是坐标指标是相对于干道上基准点的距离。

3.1.2 路口 CIntersection

路口的红绿灯可以设置为图 2.3 的形式,也可以改为 4 组(无中间等待区)或 3 组(单行主干道)。由于数据较为简单,红绿灯的描述就直接在路口类内。

路口数据结构除了含有多组红绿灯外,还定义了路口的车道数。根据我们的观察,一般正常行驶的情况下,每秒钟每条道平均只能过 1 辆车。因此,路口就可以模拟道路的通量。如果进入路口的车辆太多,以至于绿灯时无法全部通过,有些车辆只好原地等待,等待时间将被加入最后结果。

3.1.3 车辆 CVehicle

车辆在模拟中主要的数据为位置、行驶方向、进出口,当然,还可以车辆的红灯等待时间,最后这个数字被累积到该干道总的车辆红灯等待时间中,并用于判断配时方案的好坏。

3.2 道路信息的输入形式:从文件中提取。

Road.cpp 中 BOOL CRoad::Construct(CStringstrInputFile) 读取输入文件

环陵路	5780	0	0	120	60	2	0	3	2	
金马路	4800	0	0	120	60	0	20	3	2	
青马北路	4500	0	0	120	60	0	20	3	2	
赛马路	4150	0	0	120	60	0	20	3	2	
马土路	2730	0	0	120	60	0	20	3	2	
燕西线	2280	0	0	120	60	0	20	3	2	
仙隐南路	1780	0	0	120	60	0	20	3	2	
学则路	1250	0	0	120	60	0	20	3	2	
文澜路	800	0	0	120	60	0	20	3	2	
学津路	450	0	0	120	60	0	20	3	2	
学海路	0	0	0	120	60	0	20	3	2	
学典路	-500	0	0	120	60	2	0	3	2	

图 3.2 输入文件 Road.dat

图 3.2 为输入文件格式。其中第一列为十字路口名;第二列为坐标;第三列为上行方向路口绿灯开启时间与坐标原点(学海路)绿灯开启时间的延迟;第四列为下行方向路口绿灯开启时间与坐标原点(学海路)绿灯开启时间的延迟;第五列为主方向红绿灯周期时间;第六列为主方向绿灯时间;第七列为该路口主干到车辆进入时间间隔;第八列该路口次干道车辆进入间隔时间;第九列为路口主方向车道数,也是每秒钟的通量,用于在绿灯亮时,最多可以过多少辆车;第十列是横向路的道数。

坐标原点(学海路)是主要交叉路口,也就是说,上下

行红绿灯在这一点是同步的。从理论上,存在下一个同步路口,是的红绿灯再次同步。这个路口就是图 2.1 中的另一个网格点。

城市里车流量随时间段可分为高峰低谷,在星期里还有周末工作日之分。不同时间段,流量不同,最佳配时方案也应该不同。

3.3 算法设计

从现实世界中,车辆、行人和红绿灯的运行基本上是相互独立的,每个个体有他的思维,理论上软件模拟应该用多线程(Multi-threading)的方法。计算机将这些对象独立模拟,它们按照自己的规律运行,同时要考虑之间的关系,而这些关系是随机的。从此可见,多线程软件编写难度非常大。

本项目考虑到实际情况,选用了单线程方法。但是在计算过程中,每个对象也是按自己的规律单独模拟,当然也要考虑它们之间的关系,如红灯停绿灯行。程序中这些对象的动作是靠秒触发来进行的,也就是通过时间秒的循环,对所有对象触发计算,使其能够在这一秒正确地完成自己该做的动作。

程序从数据文件读入数据,在建立了道路后,通过模拟一大批车(算例中为 1 千辆),根据设定的各路口流量,逐辆进行模拟,当所有车辆都驶出路口时,计算结束,从而得到总红灯等待时间。

这一算法,没有复杂的数学公式,只有上述简单公式。这些公式完全可以近似描述我们日常交通各对象的行为。编程难度也大大减小,计算结果具有统计意义。

3.4 程序模拟

3.4.1 将输入文件第三、四列按行车速度和路段距离配时,程序中的行车速度是 $m_nMPerS = 14$,约 50km/h。目前全 0 的设置意味着个路口红绿灯是同时的。

3.4.2 最后两列数据可以调整主干道和单行次干道流量比值。注意:同一横道上的绿灯可能不同时亮,所以穿越车辆只能在中间隔离带路口停下等待另一边绿灯。一次横穿车辆不宜过多,以免造成路口堆积。

3.4.3 现在程序模拟总共 1000 辆车进入,后驶出该路段的情况。当所有的车都驶离时,模拟结束。

3.4.4 程序还考虑了控制流量。实际上每分钟通过的车辆是有限的,路越宽、可通过的车越多。通量代表了道路在单位时间可通过车的最大辆数,当车流量大于该值时,意味着堵车了。程序中用车道数来代表通量,也就是说,通量为 1 辆/秒/道。

3.5 输出信息:输出总等待时间和,为屏幕输出

采用控制变量的方法对红绿灯的配时比例与主干道车流量做了一系列模拟,通过程序在南京仙林大道进行 1000 辆车,在各路口按规定时间生成车辆,对等的拐弯用直行等价,得出模拟结果如下:

表 3.3 车辆进入主干道时间间隔 2s/ 辆,红绿灯周期 120s,配时比例变化

车辆进入主干道时间间隔 2s/ 辆,红绿灯周期 120s,配时比例变化						
红绿灯配时比(绿/红)		40/80	50/70	60/60	70/50	80/40
主干道	优化前(s)	155980	139500	88080	63527	52558
	优化后(s)	25921	26411	27173	31241	34783
单行道	优化前(s)	99516	91956	64864	52030	47560
	优化后(s)	17431	17606	18525	20514	23148

结论:车辆进入主干道时间间隔/红绿灯周期不变,配时比例(绿/红)越大,该方案的减少等待时间效果降低。说明此优化设计当主干道车辆流畅时,方案效果降低,符合实际道路情况。

表 3.4 车辆进入主干道时间间隔变化,红绿灯周期 120s,配时比例 1/1

车辆进入主干道时间间隔变化,红绿灯周期 120s,配时比例 1/1						
车辆进入主干道时间间隔(s)		1	2	3	4	10
主干道	优化前(s)	108836	88080	74020	64921	42075
	优化后(s)	27507	27173	27879	28736	29928
单行道	优化前(s)	88062	64864	54399	47672	33160
	优化后(s)	18997	18525	17837	17801	17199

结论:红绿灯周期、配时比例不变,车辆进入主干道时间间隔越大,该方案的减少等待时间效果降低。说明此优化设计随着主干道车辆减少,即当主干道车辆流畅时,方案效果降低,符合实际道路情况。

综述:由上述仿真数据可知,在一定范围内,交通越拥堵,此方案越有效。但当车辆少到一定程度时,仿真出的等待时间差趋近于定值,说明当主干道通畅的情况下,我们方案效果存在极限值。由模拟结果看出,本仿真程序基本符合实际情况。

4. 创新点及应用

4.1 创新点

1. 考虑到我国大城市的车辆数量大等特点,我们提出通过主干道红绿灯配时方案,来减少车辆的遇红灯等待总时间,达到减少无用的汽车排放,同时也是整个城市交通较通常。

2. 提供了一面向对象软件,可用于城市主干道红绿灯配时

模拟,以确定较好的配时方案。算例证明,改程序模型与算法较符合实际实际情况,有较好的实用性。

4.2 应用前景

由于采用了面向对象编程,该软件很容易进一步改进,以更适合现实的交通实际情况,提高模拟精度。另外,使用该软件,几乎无需在路上改变或添加任何硬件,只需变换各

路口的配时时间。而且还可以做到按日、按时间段配时,优化城市交通,减少无用的汽车排放污染。另外,复杂的城市交通网络可以简化成图 2.1 的主干道和支路,一个城市只要主干道通畅(大流量),那么整个城市交通就通畅,总体无用汽车排放就少。本软件可以用于反覆地分别模拟调整各主干道配时,以达到较好地治理整个城市交通的目的。

【参考文献】

- [1] 曹树志. 智能红绿灯对城市交通节能减排效果的仿真. Computer Knowledge and Technology. No.13,May 2011
- [2] 周慧虹. 科学设置红绿灯关乎节能减排. 光明日报. 2009.11.09 09:25
- [3] 周文军. 发展智能交通缓解南京交通拥堵. China Academic Journal Electronic Publish House. 2012年第11期上
- [4] 熊建平. 交通拥堵的治本之道
- [5] 魏明杨方延等. 交通仿真的发展及现状研究. Vol.15 No.8 Aug 2003 Journal of System Simulation

【作者简介】

黄亚坤(1992年9月--)男 籍贯:宁夏中卫市 学历:本科在读 专业:飞行器动力工程

(上接第24页)

法,利用塔身结构如顶部螺栓、塔顶端角钢两端等测出其投影即可判断。在此不再赘述。

四、交叉跨越距离测量

线路交叉跨越的测量方法一般的方法和步骤如下:找到线路交叉跨越点并在下方立尺,经纬仪安放在大约线路交叉跨越线的中线的垂直线上,读尺,折算出水平距离,然后对准上、下导(地)线,分别读出垂直角,然后计算: $H=L*(\tan \alpha - \tan \beta)$ 。

但在实际操作中,时常遇到交叉跨越点落在水塘或其它人很难到达的地方这就有必要用其它方法。

首先假定一个三角形,第一个点A为交叉跨越点的投影,第二个点B为上方交叉跨越线的投影线上的仪器观测点,第三个点C为下方交叉跨越线的投影线上的仪器观测点。

方法及步骤:

1:安放经纬仪于B点,对准上方交叉跨越线调至重合并归零,量仪高、做标记,对下方导线读垂直角,对C测水平角、长度、高差。

2:同样,安放经纬仪于C点,对准上方交叉跨越线调至

重合并归零,量仪高做标记,对上方导线读垂直角,对B测长度,高差。见表

仪器位置	仪高	垂直角	水平角	长度	高差
B站	1.3	79°20'	60°20'	92米	-3.1米
C站	1.3	77°25'	50°10'	92米	0.5米

根据上述条件和利用正弦定理推算出 $AC=85.3$ 米, $AB=75.4$ 米。

设C站高程为0米 则上方交叉跨越线高程为 $1.3+85.3*\tan(90^\circ-77^\circ25')=20.3$ 米,下方交叉跨越线高程为 $0.5+1.3+75.4*\tan(90^\circ-79^\circ20')=16$ 米,交叉跨越线距离为4.3米。本例是上下线分别为110V、10V线路,故符合规范。

结束语

把技术、技能、规范三者有机地结合起来是职业线路测量人员必备的素质。用光学经纬仪测量工作量大,同时特别在测量长度时要注意其精度要求。现在测量普遍使用全站仪,换算简单,速度快,精度高,尽量在测量中推广应用。

【参考文献】

- [1]《架空送电线路施工手册》中国电力出版社 2002
- [2]《送电线路》中国水利水电出版社 2007