

基于图论的路网交通检测器之布点

林培群, 徐建闽

(华南理工大学 土木与交通学院, 广东 广州 510640)

摘要: 为获取各路段的交通流量, 将路网检测器布点问题转变成寻求有向图的流控制子图的问题. 首先将任意路网抽象为有向图, 定义弧的度表征路段的重要性, 证明完全有向回路图(CCG)的若干结论后给出CCG最小流控制子图的获取算法, 同时给出有向图非回路部分的流控制子图获取方法, 进而提出能在任意路网上进行检测器优化布点的完整算法. 算例选取广州火车站附近的路网, 结果验证了所提出的方法的有效性.

关键词: 检测器布点; 路网; 交通流量; 图论; 流控制子图; 完全有向回路图

中图分类号: U491 **文献标识码:** A

Layout of traffic detectors in road network based on graph theory

LIN Pei-qun, XU Jian-min

(School of Civil Engineering and Transportation, South China University of Technology, Guangzhou Guangdong 510640, China)

Abstract: To determine the traffic flow on every road-section, we convert the road network detector layout into the flow control sub-graph(FCSG) in a directed graph. Firstly, a road network is abstracted as a directed graph, and the edge-degree measure is defined as the importance of a road-section. After proving several statements about the completely cyclic graph(CCG), we develop an algorithm for determining the minimal FCSG from a CCG. Meanwhile, a method for determining the FCSG from an acyclic graph is also given. Finally, the integrated algorithm for determining the detector layout in the road network is put forward. A numerical example employing the road network around Guangzhou Eastern Railway Station is given to demonstrate the effectiveness of the proposed method.

Key words: detector layout; road network; traffic flow; graph theory; flow control sub-graph; completely cyclic graph

1 引言(Introduction)

道路交通信息采集的准确性、全面性、实时性制约着智能运输系统(ITS)的应用效果^[1]. 在构建固定式交通信息采集系统的时候, 人们总会遇到一个重要的基本问题, 即如何进行检测器布点. 现有研究中, 姜桂艳^[2]等提出通过若干原则来定性指导检测器布点. 伍建国^[3]等所应用的相关分析法、张航^[4]等所应用的聚类分析法在实施之前都必须有一个初始的检测器布置方案. 在已知极点转移概率的基础上, YANG^[5]、EHLERT^[6]等利用整数线性规划模型求解检测器的布点方案, 该方法的目标是获取OD数据而非路段的实时流量. 采用图论模型来描述路网检测器布点问题是现有方法中最为科学的, 美国德州大学的GU^[7,8]等人利用ZHANG^[9]对于有向图的结论提出一种基于破圈法^[10]的检测器布点方法, 具有一定的理论价值, 但该方法不能解决非平衡节点的问题, 而其基本假设认为对于有向图 G , 将

其分为两部分 X 和 Y 则 $X \rightarrow Y$ 的流量与 $Y \rightarrow X$ 的流量相等, 显然不符合实际. 总的来说, 国内外研究者对交通检测器布点方法的研究还不是很多, 相关问题并没有得到很好的解决.

本文基于图论方法, 研究基础是 GU^[7,8]、ZHANG^[9]等人的结论, 目标是建立一种能够获取任意路网各路段流量, 具有普遍适应性和可操作性的检测器优化布点方法.

2 交通检测器布点方法(Traffic detector layout method)

2.1 路网图模型(Graph models of road network)

定义 1 一个有向图可用 $G = (V, E)$ 来表示, 其中 $V \neq \emptyset$ 称为顶点集, V 的元素称为顶点或节点, E 称为弧或有向边集, 对于 $\forall e \in E$ 均从顶点开始到顶点结束.

可将任意路网转换为有向图, 如图1所示.

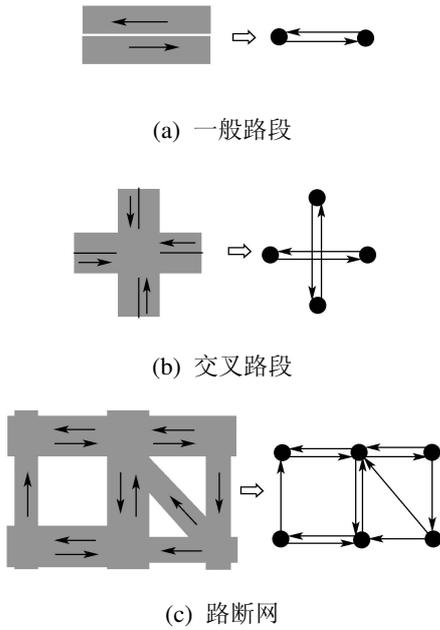


图1 道路对象的图模型

Fig. 1 Graph models of road objects

道路上每个交通流方向为1个路段,下文所述的“路段”都假设其上没有交通产生、吸引点,如果有必要考虑这些交通产生、吸引点的影响,则可将路段进一步细分.此外,由于本文的主要研究对象是整个路网的情况,因此将交叉口当成一个节点,而且总是考虑最复杂的情况,即在对应路段存在的情况下,进入节点(交叉口)的交通流可以直行、左转、右转以及掉头.此外,本文所研究的图均为简单图,即不存在平行边和环^[10].

2.2 有向图及其性质(Directed graph and its properties)

定义 2 一个有向图 $G = (V, E)$, 其中 $V \neq \emptyset$ 称为顶点集, $\forall v \in V$ 不是孤立顶点, $E \neq \emptyset$ 称为弧集, 对于 $\forall e(o, d) \in E$, 其中 o, d 为 e 的起、迄点, 如果从 d 出发, 能够在 G 中找到一条路径到达 o , 则称 G 为完全有向回路图.

从物理含义上看, 城市道路相互交接, 大部分路段均能找到一个回路.

定理 1 对于任意有向图 $G = (V, E)$, 当且仅当 $\forall e \in E$ 均在有向圈上, G 是完全有向回路图.

证 充分性方面, 由于 $\forall e(o, d) \in E$ 均在有向圈上, 因此如果沿着这个圈, $d \rightarrow o$ 一定能够实现. 必要性方面, 如果 G 是完全有向回路图, 那么 $d \rightarrow o$ 能够实现, 也即存在一条有向路径, 可以不重复任意节点而从 d 到 o , 记这个路径为 L , 则 $L \cup \{e\} \subseteq G$ 是一个圈. 证毕.

定义 3 设 $G = (V, E)$ 为有向图, 称完全有向回路图 $G_C = (V_C, E_C) \subseteq G$ 是 G 的完全有向回路子图, 记 G 的完全有向回路子图的集合为 $R_C(G)$. 如果 $|E_C| = \max \{|E(Y)| | Y \in R_C(G)\}$, 则 G_C 是 G 的最大完全有向回路子图, 此时如果 $T = G - G_C \neq \emptyset$, 则称 T 是 G 的最大非回路子图.

定理 2 对于任意有向图 $G = (V, E)$, 如果 G_C 是 G 的最大完全有向回路子图, 则 G_C 具有唯一性.

证 假设 $G_T \neq G_C$ 也是 G 的最大完全有向回路子图, 则 $\exists e \in E(G_T)$, 且 $e \notin E(G_C)$, 而由于 $e \in E(G_T)$, 所以 e 能在 G 中找到一个回路(圈), 记该回路为 $C(e)$, 此时 $G_C \cup C(e)$ 是 G 的有向回路子图, 且 $|E(G_C \cup C(e))| \geq |E(G_C \cup \{e\})| > |E(G_C)|$, 这与 G_C 是 G 的最大完全有向回路子图矛盾, 所以 G_C 具有唯一性. 证毕.

定义 4 设 $G = (V, E)$ 为有向图, 以顶点 v 为起点的弧的数量称为 v 的出度, 记为 $d^-(v)$, 以顶点 v 为迄点的弧的数量称为 v 的入度, 记为 $d^+(v)$. $d^-(v)$, $d^+(v)$ 统称为顶点 v 的度, 记为 $d(v)$, 则:

$$d(v) = d^+(v) + d^-(v). \tag{1}$$

在交通系统中, 节点的度越大, 代表与该节点相接的路段越多, 在没有其他信息的情况下该节点的重要性也越大.

定义 5 设 $G = (V, E)$ 为有向图, 对于 $\forall e \in E$, 记其起点为 v_e^o , 迄点为 v_e^d , 并记 e 的度为 $d_E(e)$, 则:

$$d_E(e) = d(v_e^o) + d(v_e^d). \tag{2}$$

以弧代表路段, 则路段的度越大, 代表与该路段相接的其他路段越多, 在没有其他信息的情况下该路段的重要性也越大, 因此可用路段的度代表布置检测器的优先权.

定理 3 设 $G = (V, E)$ 为有向图, 若 $|E| = m$, 则对于 $\forall e \in E$, 有

$$2 \leq d_E(e) \leq 2m. \tag{3}$$

证 对于 $\forall e \in E$, 由于 G 中无环, 故有 $1 \leq d(v_e^o) \leq m, 1 \leq d(v_e^d) \leq m$, 将其代入式(2), 可得式(3).

证毕.

定理 4 设 $G = (V, E)$ 为有向图, $C \subseteq G$ 是一个圈, 对于 $\forall e(o, d) \in E$, 如果 $\{o, d\} \subseteq V(C)$, 则 $C \cup \{e\}$ 是 G 的完全有向回路子图.

证 由定义2可知, 证明定理4只需要证明由 d 出发可达 o . 由于 C 是 G 中的一个圈, 因此 $V(C)$ 中任意两点间均是有向可达的. 证毕.

由定理4很容易推广到下面更具普遍性的结论.

推论 1 设 $G = (V, E)$ 为有向图, $C \subseteq G$ 是一个圈, G 中的一条路径 $R(u, v)$ 以 u, v 为起讫点, 如果 $\{u, v\} \subseteq V(C)$, 则 $C \cup R$ 是 G 的完全有向回路子图.

定理 4 与推论 1 有利于提高弧分类算法的速度, 可见下文的算法 3.

定义 6 设 $G = (V, E)$ 为有向图, 对于 $\forall e \in E$ 记其流(flow)为 $f(e)$, 对于 $\forall v \in V$, 记以 v 为起点的弧的集合为 $E^-(v)$, 以 v 为讫点的弧的集合为 $E^+(v)$, 则 $\sum_{e \in E^-(v)} f(e)$ 称为顶点 v 的流出流, $\sum_{e \in E^+(v)} f(e)$ 称为顶点 v 的流入流. 图 G 中 $\forall v \in V$ 可分为 2 种类型:

1) 流入流与流出流相等, 则

$$\sum_{e \in E^+(v)} f(e) = \sum_{e \in E^-(v)} f(e). \quad (4)$$

2) 流入流与流出流不相等, 则

$$\sum_{e \in E^+(v)} f(e) \neq \sum_{e \in E^-(v)} f(e). \quad (5)$$

考虑实际应用需要, 简化处理过程:

如果 $d^+(v) > 0$ 且 $d^-(v) > 0$, 则认为式(4)成立, v 是一个平衡点; 如果 $d^+(v) > 0$ 且 $d^-(v) = 0$, 称 v 为纯吸引点, 如果 $d^-(v) > 0$ 且 $d^+(v) = 0$, 称 v 为纯产生点, 纯吸引点与纯产生点统称为非平衡点.

如果与非平衡点 p 相接只有 1 条弧 l , 则称 l 为 p 的单流路, 此时如果 l 的起点(讫点)为 p , 称 l 为 p 的单流出(入)路.

文献[7,8]中的方法需要所有节点均满足式(4), 这在实际应用中受到一定的限制.

定理 5 对于满足定义 2 的完全有向回路图 $G = (V, E)$, $\forall v \in V$ 均满足式(4).

证 $\forall v \in V$, 由定义 2 可知从 v 出发经过某条路径可以回到 v , 则 $\exists e \in E$ 以 v 为起点, 且 $\exists b \in E$ 以 v 为讫点, 此时 v 符合定义 6 的第 1 种顶点类型.

证毕.

定义 7 设 $G = (V, E)$ 为有向图, $H \subseteq G$, 如果由 $\{f(b) | b \in E(H)\}$ 能确定 $\{f(e) | e \in E\}$, 则称 H 是 G 的流控制子图, 记 G 的流控制子图的集合为 $F(G)$, 如果 $|E(H)| = \min \{|E(P)| | P \in F(G)\}$, 则称 H 为 G 的最小流控制子图, 此时:

$$H = \arg \min_{P \in F(G)} (|E(P)|). \quad (6)$$

路网的交通检测器布点本质上是一个寻找流控制子图的过程.

算法 1 对于满足定义 2 的完全有向回路图 $G = (V, E)$, 可按照以下算法得到 G 的流控制子图 H . 算

法如下:

Step 1 $H = \emptyset$.

Step 2 循环执行: 如果 $G \neq \emptyset$, 选择 $\forall e \in E(G)$, 令 $H = H \cup \{e\}$, $G = G - \{e\} - Z$, 其中 Z 是 $G - \{e\}$ 的基本图的割边集.

算法 1 基于破圈法的思想, 所获得的弧集是 E 的真子集. 算法中 Step 2 的每一步在选择弧的时候带有任意性, 即 e 可选 $\forall b \in (\{e\} \cup Z)$, 因此该算法得到的流控制子图具有多种可能. 假如 $C_1 \cup C_2 \cup \dots \cup C_n = G$, 其中 C_i 是圈, 当 G 中任意弧仅属于某个圈时(即当 $i \neq j$ 时, $C_i \cap C_j = \emptyset$), 那么按照算法 1 能获得的流控制子图的数量为:

$$|F(G)| = \prod_{i=1}^n |E(C_i)|. \quad (7)$$

定理 6 对于满足定义 2 的完全有向回路图 $G = (V, E)$, 按照算法 1 得到的流控制子图是 G 的最小流控制子图.

证明过程可见文献[7]. 最小流控制子图能用于交通检测器的最优布点, 但是由于所得到的方案具有多种可能, 且选取弧时随意性较大, 因此有必要予以改进. 下文算法 4 采用弧的度来衡量路段的优先顺序, 可以克服这种随意性.

定理 7 对于满足定义 2 的完全有向回路图 $G = (V, E)$, 如果 G 的基本图是连通的, 当且仅当 $T = G - H$ 的基本图是 G 的基本图的生成树, H 是 G 的最小流控制子图.

充分性和必要性的证明可见文献[8]. 值得注意的是, Kruskal 等算法^[11~14]能够得到 G 的基本图的生成树, 因此在某些情况下(例如知道弧的权求最小生成树)可用 Kruskal 等算法代替算法 1.

推论 2 对于满足定义 2 的完全有向回路图 $G = (V, E)$, 按照算法 1 产生的最小流控制子图为 H , 则 $T = G - H$ 是 G 的生成森林, 且 T 的集合为 G 的所有生成森林.

证 $\forall v \in V$, 假如有向图 $G = G_1 \cup G_2 \cup \dots \cup G_n$, 且当 $i \neq j$ 时 $G_i \cap G_j = \emptyset$, 则 $\forall G_i \subseteq G$ 是一个完全有向回路图(否则不符合定义 2), 此时采用算法 1 分别获取 G 的每一部分的流控制子图 H_1, H_2, \dots, H_n , 若 $T_1 = G_1 - H_1, T_2 = G_2 - H_2, \dots, T_n = G_n - H_n$, 则由定理 7 的必要性可知 T_i 可能是 G_i 的任何一种生成树, 因此 $T = T_1 \cup T_2 \cup \dots \cup T_n$ 可能是 G 的任何一种生成森林. 证毕.

定理 8 对于满足定义 2 的完全有向回路图 $G = (V, E)$, $T \subseteq G$ 是 G 的基本图的生成森林, 则有:

1) $H = G - T$ 是 G 的最小流控制子图.

2) 如果 G 的基本图是连通的, 则 T 的基本图是 G 的基本图的生成树, 有

$$|H| = |E| - |V| + 1. \quad (8)$$

3) 如果 $G = G_1 \cup G_2 \cup \dots \cup G_n$ 且当 $i \neq j$ 时 $G_i \cap G_j = \emptyset$, 那么 T 含有 n 棵树, 此时有

$$|H| = |E| - |V| + n. \quad (9)$$

证 由推论2可得到1); 2)可参考文献[8]; 3)是更具一般性的结论, 证明如下: 如按算法1, 分别求取 G_1, G_2, \dots, G_n 每一部分的最小流控制子图, 则

$$\begin{aligned} |H| &= \sum_{i=1}^n (|E(G_i)| - |V(G_i)| + 1) = \\ &= \sum_{i=1}^n (|E(G_i)|) - |V| + n = \\ &= |E| - |V| + n. \end{aligned}$$

证毕.

2.3 非回路图的处理(Processing for acyclic graph)

如将任意有向图 $G = (V, E)$ 的最大完全有向回路子图 G_C 除去, 那么非回路子图 $G - G_C$ 是一个有向无圈图(directed acyclic graph, 简称为DAG), 其基本图是一个森林. 在一般的城市道路网络中, $E(G_C)$ 包括 E 大部分的弧, 采用贪婪策略^[15], 规定凡与 G_C 直接相连的弧必须设置检测器.

算法2 对于有向图的非回路子图 $T \subseteq G$, 按照以下算法布置检测器:

Step 1 令能确定流大小的弧集 $S = \emptyset$, 设置检测器的弧集 $I = \emptyset$.

Step 2 选择所有与 G 的最大完全有向回路子图 G_C 直接相接的弧集 K , 令 $I = I \cup K$, 则不能确定流量的弧集 $T = T - K$.

Step 3 选择所有连接两个非平衡节点的弧集 J , 令 $I = I \cup J, T = T - J, S = I$.

Step 4 循环: 根据 $\sum_{e \in E^+(v)} f(e) = \sum_{e \in E^-(v)} f(e)$, 修改 S, T , 如果本次循环后 $|S|$ 不增加则跳出循环.

Step 5 如果 $T = \emptyset$, 算法结束.

Step 6 选择与某平衡点 v 相接、还无法确定流的任意弧 e 设置检测器, 令 $I = I \cup \{e\}, S = S \cup \{e\}, T = T - \{e\}$, 转到Step 4.

2.4 弧分类算法(Edge classification algorithm)

采用算法1, 2对有向图进行检测器布点, 必须先区分哪些弧在有向圈中、哪些不在. 由定义2可知, 判断弧是否在有向圈中可以考察从其讫点出发能

否到达其起点, 为此本文在深度优先搜索(DFS)算法^[16]的基础上, 结合定理1、定理4的结论提出改进算法. 通过该算法, 可将所有弧分为回路弧与非回路弧两种, 其中前者的集合组成最大有向回路子图, 适合算法1, 后者的集合组成最大非回路子图, 适合算法2.

算法3 回路弧与非回路弧分类算法的C语言伪代码如下:

```
int edge[ ]; '还没确定的弧
int edgeC[ ]; '能够找到回路的弧
int edgeT[ ]; '不能找到回路的弧
int t = 0;
Node *p;
bool get;
do while(t < n)
    p=edge[i].p2; 'p1, p2是弧i的起、讫点
    get=FALSE;
    do
```

```
    If(p==edge[i].p1)
```

找到一个圈, 将圈上的每条弧记入edgeC[]中,

```
    修改edge[], t, get=TRUE,
```

```
    根据定理4、推论1修改edgeC[], edge[], t
```

```
    退出里层do-while
```

```
    If(p!=edge[i].p1)
```

```
    if(可以继续搜索)
```

```
        p往下
```

```
    else
```

```
        p往上
```

```
    If(!get && p==edge[i].p2)
```

```
        将edge[i]放入edgeT[], 修改edge[], t
```

```
        退出里层do-while
```

```
    while(p!= edge[i].p1 )
```

算法3也可以用于检测割边集^[10~12], 这对算法1是个必要的辅助. 算法3中每个节点均为树结构, 该结构有利于索引, 如图2所示. 在进行关系描述时, 可采用以下邻接关系矩阵:

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}.$$

矩阵中1代表有连接的弧, 0代表没有. 将矩阵中取1的值用权值代替, 0用 ∞ 代替就变为邻接权矩阵.

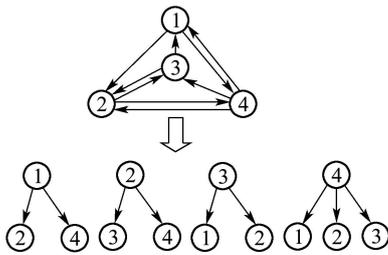


图 2 节点的树型数据结构
Fig. 2 Tree model of vertex

2.5 任意有向图流控制子图获取算法(Algorithm for getting flow control sub-graph)

对于完全有向回路图 G , 采用算法1所获得的最小流控制子图有多种可能; 对于非回路图, 算法2 Step 6在选择弧时也具有随意性. 为解决解不唯一的问题, 本文结合弧的物理含义提出一种有效的解决方法, 即根据定义5按照弧的度的大小对 E 进行排序, 得到一个新的有向弧集 $E_d = \{e_1, e_2, \dots, e_n\}$, 此时对于任意弧 $e_i (i \in \{1, 2, \dots, n\})$, i 越小代表弧越重要. 对于算法1、算法2, 在选取弧时, 总是在剩下的弧集中选择最重要的, 然后更新 E_d 直到 $E_d = \emptyset$.

算法 4 综合算法1, 2, 3, 任意有向图流控制子图的获取步骤如下:

Step 1 计算有向图 $G = (V, E)$ 中 $\forall e \in E$ 的度, 并按度从大到小重新排列 E 的元素, 得到一个新的有向边集 $E_d = \{e_1, e_2, \dots, e_n\}$.

Step 2 按照算法3从图 G 中得到最大完全有向回路子图 G_C .

Step 3 令 $H = \emptyset, K = G_C$.

Step 4 循环: 当 $E(K) \neq \emptyset$, 按照在 E_d 中的先后顺序选择一条弧 $e \in E(K)$, 使 $H = H \cup \{e\}, K = K - C_H(e)$, 其中 $C_H(e) = \{e\} \cup Z, Z$ 是 $K - \{e\}$ 的基本图的割边集.

Step 5 令 $T = G - G_C, E_T = E(T)$ 代表不能确定流量的弧集. 对于所有与 G_C 直接相交的弧的集合 $E_{G_C}^{dc}$, 令 $H = H \cup E_{G_C}^{dc}$, 使 $E_T = E_T - E_{G_C}^{dc}$.

Step 6 对于所有连接两个非平衡点的弧的集合 $E_{G_C}^m$, 令 $H = H \cup E_{G_C}^m$, 使 $E_T = E_T - E_{G_C}^m$, 此后令 $E_X = E - E_T$ 代表能确定流量(通过检测或推算)的弧集.

Step 7 循环: 若 $E_X = E$, 结束算法, 否则对于所有平衡节点 v , 按照 $\sum_{e \in E^+(v)} f(e) = \sum_{e \in E^-(v)} f(e)$ 判断是否有弧能够由 E_X 确定, 并相应修改 E_T, E_X , 直到本次循环不能使 $|E_X|$ 增加.

Step 8 当 $E_T \neq \emptyset$, 在 E_T 中寻找优先级最高的弧 b , 令 $H = H \cup \{b\}$, 修改 $E_T = E_T - \{b\}$,

$E_X = E_X \cup \{b\}$, 退到Step7.

算法的最终目标是使 $E_X = E$, 即使路网中任意路段的流量均能确定; 搜索结果 H 就是检测器布点方案.

3 算例(An example)

选取广州市火车东站周边的道路作为考察对象, 并严格按照实际交通组织方式建立路段(见图3(b)). 值得注意的是, 两个交叉口之间的道路如果能够双向通行, 则必须分为两个路段, 对于所有交叉口, 总是按照最复杂的情况处理, 即如果对应路段存在的话允许车辆直行、左转、右转、掉头. 按照算法4, 将主要处理过程图形化, 如图3(c)~(h). 图中节点标有“+”号的为纯吸引点, 标有“-”号的为纯产生点, 其他为平衡点.

对于如图3(e)所示的最大完全有向回路子图, 检测器布点方法是: 然后循环按照优先顺序选取1个路段, 然后删除剩余图的基本图的割边集, 如果剩余部分为空则跳出循环. 此过程最后选取的路段是

$$\{e_1, e_2, e_3, e_4, e_5, e_6, e_8, e_9, e_{10}, e_{12}, e_{13}, e_{15}, e_{19}, e_{21}\}.$$

对于如图3(f)所示的非回路部分, 检测器布点方法是: 选取与最大有向回路子图 G_C 相接的路段布置检测器, 然后循环{按照式(4)寻找能够由其他路段确定流量的路段, 寻找不到则跳出}. 然后再循环{如果所有路段流量均能确定则退出, 否则按照编号顺序选取1个路段, 每选取一个路段, 循环[按照式(4)寻找能够由其他路段确定流量的路段, 寻找不到则跳出]}. 此部分最后选取的路段是: $\{e_{27}, e_{28}, e_{29}\}$. 此时整个检测器布点结束, 需要布置检测器的弧是

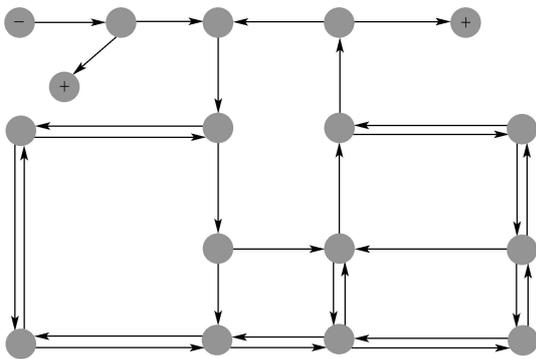
$$\{e_1, e_2, e_3, e_4, e_5, e_6, e_8, e_9, e_{10}, e_{12}, e_{13}, e_{15}, e_{19}, e_{21}, e_{27}, e_{28}, e_{29}\}.$$



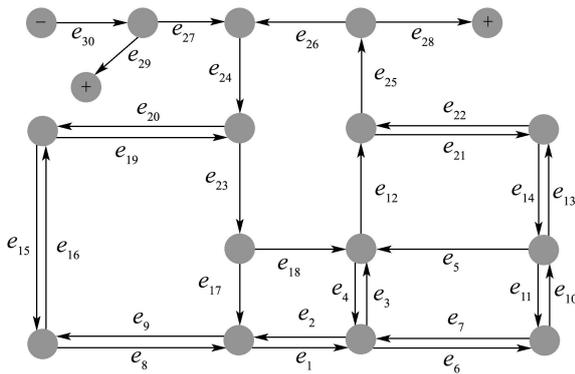
(a) 基本地图(www.google.com)



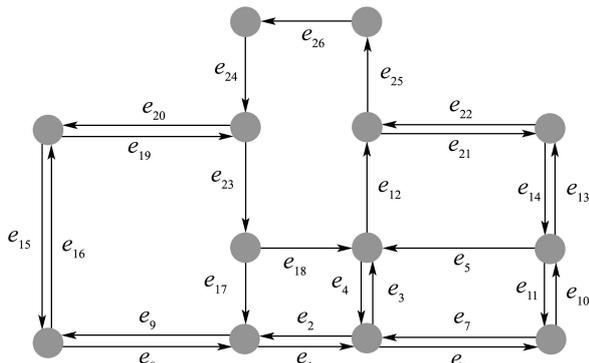
(b) 抽取主要的路网



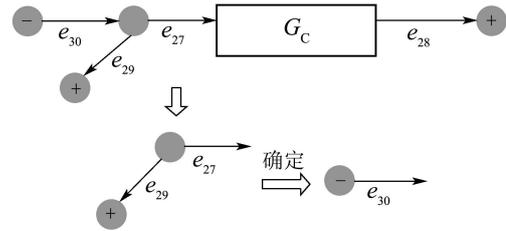
(c) 路网的图抽象



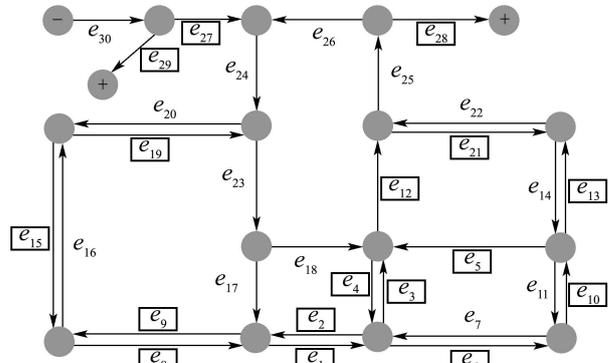
(d) 各路段按度大小排列优先顺序



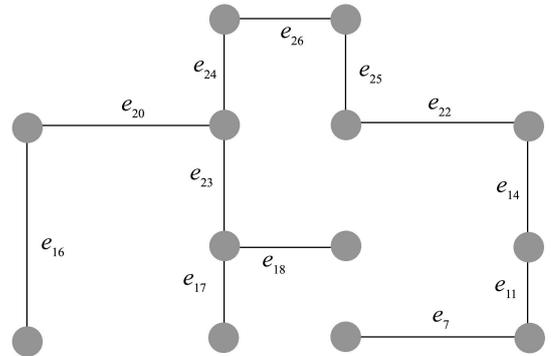
(e) 最大完全有向回路子图



(f) 非回路子图



(g) 最终检测器布置方案(编号带矩形框的弧)



(h) 完全有向回路子图中不布置检测器的部分构成其基本图的生成树

图3 检测器布点过程

Fig. 3 Detector layout process

结果分析: 在30条弧组成的路网中, 选取其中17条设置检测器就可以获得整个路网的流量数据, 而由26条弧组成的回路子图需要选取14条弧. 检测器布置率: 整个路网为56.7%, 回路子图为53.8%, 非回路子图为75%, 可见回路子图具有更低的布置率. 实际路网中, 数量众多的城市道路大部分都在回路子图中, 而城市对外高速公路、普通公路虽在非回路子图中, 但数量较少, 因此本文方法如果应用在城市道路中具有较高的效率.

4 结论(Conclusion)

为了解决任意路网的检测器布点问题, 本文首先定义了完全有向回路图、弧的度、有向图的流控制子图等概念, 其次证明了若干具有应用指导意义的

结论, 并在此基础上分别针对有向图的回路子图、非回路子图提出获取其流控制子图的算法, 最后给出能在任意路网上进行检测器优化布点的完整算法。

算例说明了本文所提出的方法的有效性和可操作性。当然, 实际路网中对交通参数的检测不仅仅是流量, 而即便是流量, 还具有复杂的时空变化情况, 在更多约束条件下获取检测器的布点方案是下一阶段的研究内容。

参考文献(References):

- [1] ORTIZ A G, AMIN S M, WOOTTON J R. Intelligent transportation systems-enabling technologies[J]. *Mathematical and Computer Modelling*, 1995, 22(4): 11 – 81.
- [2] 姜桂艳. 道路交通状态判别技术与应用[M]. 北京: 人民交通出版社, 2004.
(JIANG Guiyang. *Technology and Application of Road Traffic Status Identification*[M]. Beijing: China Communications Press, 2004.)
- [3] 伍建国, 王峰. 城市道路交通数据采集系统检测器优化布点研究[J]. 公路交通科技, 2004, 21(2): 88 – 98.
(WU Jianguo, WANG Feng. Study on optimal layout of traffic detector for traffic data collection system in urban area[J]. *Journal of Highway and Transportation Research and Development*, 2004, 21(2): 88 – 98.)
- [4] 张航, 田冬军, 曾宪宝, 等. 城市道路交通检测器优化配置方法研究[J]. 武汉理工大学学报(交通科学与工程版), 2008, 32(6): 1161 – 1164.
(ZHANG Hang, TIAN Dongjun, ZENG Xianbao, et al. Study on the method of optimizing the allocation of city traffic detector[J]. *Journal of Wuhan University of Technology (Transportation Science & Engineering)*, 2008, 32(6): 1161 – 1164.)
- [5] YANG H, ZHOU J. Optimal traffic counting locations for origin-destination matrix estimation[J]. *Transportation Research(Part B)*, 1998, 32(2): 109 – 126.
- [6] EHLERT A, BELL M G H, GROSSO S. The optimization of traffic count locations in road networks[J]. *Transportation Research(Part B)*, 2006, 40(6): 460 – 479.
- [7] GU W Z, JIA X D. On a traffic control problem[C] // *Proceedings of the 8th International Symposium on Parallel Architectures, Algorithms and Networks*. Piscataway, America: Institute of Electrical and Electronics Engineers, 2005: 510 – 1515.
- [8] GU W Z, HSU D F, JIA X D. On a network sensing problem[J]. *Journal of Interconnection Networks*, 2006, 7(1): 63 – 73.
- [9] ZHANG C Q. *Integer Flows and Cycle Covers of Graphs*[M]. New York, America: Marcel Dekker, Inc., 1997.
- [10] CHARTRAND G, ZHANG P. *Introduction to Graph Theory*[M]. New York, America: McGraw-Hill Science, Engineering, Math, 2004.
- [11] MARTINS P, SOUZA M C. VNS and second order heuristics for the min-degree constrained minimum spanning tree problem[J]. *Computers & Operations Research*, 2009, 36(11): 2969 – 2982.
- [12] ATAJAN T, YONG X, INABA H. Further analysis of the number of spanning trees in circulant graphs[J]. *Discrete Mathematics*, 2006, 306(22): 2817 – 2827.
- [13] MAKINO K, UNO Y, IBARAKI T. On minimum edge ranking spanning trees[J]. *Journal of Algorithms*, 2001, 38(2): 411 – 437.
- [14] SALAMON G, WIENER G. On finding spanning trees with few leaves[J]. *Information Processing Letters*, 2008, 105(5): 164 – 169.
- [15] VINCE A. A framework for the greedy algorithm[J]. *Discrete Applied Mathematics*, 2002, 121(1): 247 – 260.
- [16] SUR S, SRIMANI P K. A depth-first search routing algorithm for star graphs and its performance evaluation[J]. *Mathematical and Computer Modeling*, 1994, 19(9): 35 – 52.

作者简介:

林培群 (1980—), 男, 博士, 讲师, 研究方向为智能交通系统、模式识别、图像处理, E-mail: pqlin@scut.edu.cn;

徐建闽 (1960—), 男, 教授, 博士生导师, 研究方向为智能交通系统、交通信息工程及控制, E-mail: aujmxu@scut.edu.cn.