

基于分流率的路径交通量求解算法*

李 峰 王书宁

清华大学自动化系, 北京 100084

摘要 为求解交通分配问题, 提出一种基于分流率的路径交通量求解方法. 该方法通过求解一个基于分流率的交通分配模型确定了满足用户平衡准则的路径交通量. 文中首先给出分流率的定义, 并根据该定义建立一个以分流率为求解变量的交通分配模型, 然后证明该模型等价于传统的以路径交通量为变量的交通分配模型. 新建模型是一带等式约束和不等式约束的非线性规划, 它的变量个数是网络中终点个数和路段个数之积的线性函数, 且比以路径交通量为变量的交通分配模型要少很多. 文中通过引入网络中节点和有向路段的平均成本的定义简单有效地解决了新建模型消除等式约束后梯度难以求解的问题, 并以此给出新建模型的求解算法. 该算法以近似 Newton 方向到约束空间上的投影作为下降方向, 用不精确搜索技术确定搜索步长. 文中从理论上证明了算法的可行性和收敛性, 并通过算例阐述了算法的有效性. 计算结果表明: 该方法可在理想的时间内求解大规模网络的交通分配问题; 在计算时间、计算精度和所需存储内存上比其他能够确定出路径交通量的求解算法都要优越; 在相同的计算精度下计算时间仅仅是其他算法的三分之一, 甚至更小.

关键词 交通分配 分流率 基于分流率算法 路径交通量 用户平衡

交通分配是城市交通控制和诱导的前提, 是城市交通网络收费系统的基础, 是城市交通系统规划和评估的关键. 它在已知城市交通网络的基础上, 将网络中所有 OD(起点-终点)对间的交通量通过一定的准则分配到各个路段, 并统计出相应路段的交通量^[1]. 当前被普遍采用的分配准则是 Wardrop^[2]第一准则和第二准则: 用户平衡准则和系统最优准则, 与这两个准则相对应的交通分配分别称为用户平衡交通分配和系统最优交通分配. 由于系统最优交通分配可以通过变换转化为用户平衡交通分配^[3], 本文只考虑用户平衡交通分配问题.

为求解用户平衡交通分配问题, 通常将其描述成如下的两种模型: 一种是以路径交通量为变量的

凸数学规划^[4], 另一种则是以路段交通量为变量的凸数学规划^[5]. 通过求解这两种模型都能获得满足用户平衡准则的路段交通量. 通常将与上述两种模型相对应的求解算法分别称为交通问题的基于路径算法和基于路段算法. 基于路径算法是以直接求解路径交通量为目的, 基于路段算法则是以直接求解路段交通量为目的. 常用的基于路径算法有聚集单纯形分解法(DSD)^[6], 投影梯度法(GP)^[7,8]和共轭投影梯度法(CGP)^[9]; 而常用的基于路段算法有 FrankWolfe(FW)^[5]算法及其改进算法^[10-14]. 实际上, 除上述求解算法之外还存在另外一种交通分配问题的求解算法, 即文献 [15] 中给出的基于起点算法. 该算法也是以求解路径交通量为目的, 同基

2005-07-22 收稿, 2005-11-10 收修改稿

* 国家自然科学基金(批准号: 60374061)和国家“九七三”(批准号 2002CB312200)资助项目

E-mail: lifeng00@mails.tsinghua.edu.cn

于路径算法不同,它是间接获得路径的交通量,是将原始网络上的交通分配问题转化成基于起点的不带环子网上的交通分配问题来求解的。

尽管基于路段算法可有效确定出路段交通量,但不能求解出路径交通量,而路径交通量比路段交通量能提供更多有价值的信息^[16-18],所以人们对能够确定出路径交通量的算法更有兴趣,即基于路径算法和基于起点算法比基于路段算法更受到关注。基于路径算法和基于起点算法能够确定出满足用户平衡准则的路径交通量,但它们仍然存在一定的缺陷。基于路径算法在每次迭代过程中需标记一定数量的路径,且随着迭代的进行需标记的路径数量会越来越大,因此它所需存储的内存相对较大,对于大型交通网络来讲,其所需存储内存就更大,这将导致基于路径算法的计算时间过长。基于起点算法虽然可以避免存储大量的路径变量,但不能在每次迭代过程中简单有效地确定出目标函数的 Hessian 矩阵,同样会使得该算法的计算时间较长。大量的实验表明基于路径算法的计算时间主要花费在路径交通量的存储和计算上,而基于起点算法的计算时间则主要花费在 Hessian 矩阵的确定上,两者相比,在相同的计算环境下后者所需计算时间较少^[19]。

综上,本文将提出一种更为有效的路径交通量求解方法。首先建立一个基于分流率的交通分配模型,并证明该模型等价于传统的以路径交通量为变量的交通分配模型,然后再给出新建模型的求解算法。在给出新建模型的求解算法过程中,首先消除新建模型的等式约束,并通过引入网络中节点和有向路段平均成本的定义简单有效地解决了新建模型消除等式约束后梯度难以求解的问题,再以近似 Newton 方向到约束空间上的投影方向作为算法的下降方向,最后采用不精确搜索技术确定搜索步长。算例表明该方法适合于大规模网络的交通分配模型的求解,其所需存储的内存和计算时间明显优于基于路径算法和基于起点算法,在相同的计算精度下其计算时间仅仅是基于路径算法和基于起点算法的三分之一,甚至更小。

1 基于分流率的交通分配模型

文献 [4] 中将城市交通网络 $G=(\mathcal{N}, \mathcal{A})$ 上用户平衡交通分配问题描述成如下的以路径交通量为变量的凸数学规划,

$$\begin{aligned} \min \mathcal{F} &= \sum_{a \in \mathcal{A}} \int_0^{x_a} c_a(\omega) d\omega, \\ \text{s. t. } \sum_{r \in \mathbb{R}_{ij}} f_{ij}^r &= q_{ij}, \quad 1 \leq i, j \leq N, \\ x_a &= \sum_{i=1}^N \sum_{j=1}^N \sum_{r \in \mathbb{R}_{ij}} f_{ij}^r \delta_{ij}^{r,a}, \quad \forall a \in \mathcal{A}, \\ f_{ij}^r &\geq 0, \quad 1 \leq i, j \leq N, r \in \mathbb{R}_{ij} \end{aligned} \quad (1)$$

其中 $\mathcal{N} = \{1, 2, \dots, N\}$ 表示 G 中所有节点的集合(包括交叉口节点、起点和终点); \mathcal{A} 表示 G 中所有有向路段的集合; x_a 表示有向路段 a 上的交通量; $c_a(\omega)$ 是有向路段 a 的成本函数(不失一般性,假设其为 ω 的正的单调递增函数,文中将采用 BPR 函数作为各个路段的成本函数^[20]); \mathbb{R}_{ij} 表示从起点 $i \in O$ 到终点 $j \in D$ 的所有路径的集合(O 和 D 分别表示网络 G 中起点和终点集合;为叙述方便,文中假设 $O=D=\mathcal{N}$),特别地,当 $i=j$ 时令 $\mathbb{R}_{ij} = \emptyset$; f_{ij}^r 表示 OD 对 ij 的路径 r 上的交通量; $f = (\dots, f_{ij}, \dots)$, $f_{ij} = (\dots, f_{ij}^r, \dots)$ 为(1)式的变量; q_{ij} 是 OD 矩阵 $Q = \{q_{ij}\}_{N \times N}$ 的元素,它表示从起点 i 到终点 j 的交通量,当 $i=j$ 时 $q_{ij} = 0$; $\delta_{ij}^{r,a}$ 为路段和路径间的指标因子,定义如下:

$$\delta_{ij}^{r,a} = \begin{cases} 1, & \text{如果路段 } a \text{ 在 } OD \text{ 对 } ij \text{ 的路径 } r \text{ 上,} \\ 0, & \text{否则.} \end{cases}$$

虽然满足用户平衡准则的路径交通量中可能存在带环路径,但通过求解以路径交通量为变量的交通分配模型(1)所获得的路径交通量中不可能存在带环路径。因此在建立用户平衡交通分配模型时,完全可以不考虑网络中的带环路径。故本文中不再考虑带环路径。前文已指出求解(1)式的算法主要有基于路径算法和基于起点算法,但这两种算法都存在一定的缺陷。实际上导致这些缺陷的本质原

因是当城市交通网络规模较大时模型(1)的变量个数非常庞大, 庞大的变量个数导致算法难以求解, 且计算时间较长. 为解决上述问题, 本文将建立一个基于分流率的数学模型, 并用其描述用户均衡交通分配问题. 在建立该模型之前, 先给出分流率的定义.

定义 1 网络 G 中的任意一有向路段 $a = [a_i, a_n] \in \mathcal{A}$ 关于终点 $n \in D$ 的分流率 α_a^n 定义为该路段上流向终点 n 的交通量 x_a^n 同该有向路段的尾节点 a_i 处流向终点 n 交通量 $g_{a_i}^n$ 间的比率. 其中 a_i 和 a_n 分别表示有向路段 a 的尾节点和头节点, 且

$$g_{a_i}^n = \sum_{j=1}^N \sum_{r \in S_m, a_i \in r} f_m^r, \quad \forall n \in D, \quad (2)$$

其中 S_m 表示网络 G 中 OD 对 jn 间所有不带环路径的集合.

显然根据定义 1 可知,

$$\begin{aligned} \sum_{a \in A(i)} \alpha_a^n &= 1, 1 \leq i, n \leq N, i \neq n, \\ 0 \leq \alpha_a^n &\leq 1, \forall a \in \mathcal{A}, \forall n \in D, \end{aligned} \quad (3)$$

其中 $A(i)$ 表示网络 G 中所有尾节点为节点 i 的有向路段集合. 网络 G 中任一 OD 对 in 的路径 $r \in S_m$ 上的交通量 f_m^r 可描述为

$$f_m^r = q_{in} \prod_{a \subseteq r} \alpha_a^n, \quad \forall r \in S_m, 1 \leq i, n \leq N. \quad (4)$$

因此,

$$\begin{aligned} x_a &= \sum_{n=1}^N x_a^n(a) = \sum_{n=1}^N \sum_{i=1}^N \sum_{r \in S_m} f_m^r \delta_m^{r,a} = \\ &= \sum_{n=1}^N \sum_{i=1}^N \sum_{r \in S_m} q_{in} \prod_{a \subseteq r} \alpha_a^n \delta_m^{r,a}. \end{aligned} \quad (5)$$

引理 1 在不考虑带环路径的网络 G 中, 满足(3)式的 α , 必将满足

$$\sum_{r \in S_m} \prod_{a \subseteq r} \alpha_a^n = 1, 1 \leq i, n \leq N, i \neq n.$$

证明 对于任意的起点 i 和终点 $n (i \neq n)$, 用数学归纳法来证明该结论. 对 S_m 中的路径个数进行归纳. 显然当 S_m 中只有一条路径时,

$$\alpha_a^n = 1, \quad \forall a \subseteq r, r \in S_m.$$

结论显然成立.

假设当 S_m 中的路径个数小于等于 m 时结论成立, 下面来证明当路径的个数为 $m+1$ 时结论也成立. 因为

$$\sum_{r \in S_m} \prod_{a \subseteq r} \alpha_a^n = \sum_{a \in A(i)} \alpha_a^n \left\{ \sum_{r \in S_m, i a \subseteq r} \prod_{b \subseteq r, b \neq a} \alpha_b^n \right\},$$

且根据假设有

$$\sum_{r \in S_m, i a \subseteq r} \sum_{b \subseteq r, b \neq a} \alpha_b^n = 1,$$

所以

$$\sum_{r \in S_m} \prod_{a \subseteq r} \alpha_a^n = \sum_{a \in A(i)} \alpha_a^n \left\{ \sum_{r \in S_m, i a \subseteq r} \sum_{b \subseteq r, b \neq a} \alpha_b^n \right\} = 1.$$

定理 1 在不考虑带环路径的情况下, 按照下式给出的以路段分流率为变量的数学规划, 等价于传统的以路径交通量为变量的用户平衡交通分配模型(1).

$$\min \mathcal{F} = \sum_{a \in \mathcal{A}} \int_0^{x_a(a)} c_a(\bar{\omega}) d\bar{\omega},$$

$$\text{s. t. } \sum_{a \in A(i)} \alpha_a^n = 1, 1 \leq i, n \leq N, i \neq n, \quad (6)$$

$$\alpha_a^n \geq 0, \quad \forall a \in \mathcal{A}, 1 \leq n \leq N,$$

其中 $\alpha = (\alpha^1, \alpha^2, \dots, \alpha^N)$, $\alpha^n = (\dots, \alpha_a^n, \dots)$, $a \in \mathcal{A}$ 是(6)式的变量.

证明 根据引理 1 以及(4)式, 可知满足(6)式约束条件的 α 必有

$$\sum_{r \in S_{ij}} f_m^r = \sum_{r \in S_{ij}} q_{ij} \prod_{a \subseteq r} \alpha_a^j = q_{ij}, \quad \forall r \in S_{ij}, 1 \leq i, j \leq N$$

考虑到网络中任意路径上的交通量一定是非负

的, 因此由满足(6)式约束条件的 α 及(4)式所确定出的路径交通量将满足模型(1)的约束条件.

根据分流率的定义可知, 由满足(1)式约束条件的路径交通量及下式所确定出的 α 也满足模型(6)的约束条件.

$$\alpha_a^n = \frac{x_a^n}{g_{a_i}^n} = \frac{\sum_{i=1}^N \sum_{r \in S_m} f_m^r \delta_m^{r,a}}{\sum_{j=1}^N \sum_{r \in S_{m_i}, a_i \in r} f_j^r}, \forall a \in \mathcal{A}, \forall n \in D.$$

这说明在不考虑带环路径的情况下, 模型(6)和(1)的约束条件可以相互转换. 因为模型(6)和(1)的目标函数具有相同的形式, 所以在不考虑带环路径的情况下(6)式等价于传统的用户平衡交通分配模型(1). 结论成立.

从模型(1)和模型(6)的变量个数来看, 模型(1)的变量个数是网络 G 中所有路径的个数, 是网络 G 中节点个数的指数函数, 而模型(6)的变量个数为 $M \cdot (N-1)$, 其中 M 表示网络 G 中所有有向路段的个数. 因此模型(6)比模型(1)变量的个数明显少很多. 这也为模型的进一步求解提供了方便.

2 基于分流率的交通分配模型求解算法

模型(6)是一个带线性等式约束和不等式约束的非线性规划. 考虑到求解此类规划的一些算法(如 FrankWolfe 算法、既约梯度法、投影梯度法等)并不特别适合该模型的求解, 本文提出一种新的求解算法——基于分流率算法. 它是一种迭代算法, 基本步骤如下:

- 步 1: 算法初始化, 给定 α 的初始值.
- 步 2: 收敛性判断, 判断 α 是否满足收敛性条件, 若满足则输出路径交通量, 否则转步 3.
- 步 3: 主循环, 确定下降方向及搜索步长, 更新 α 并转步 2.

显然, 该算法的关键是如何确定下降方向和搜索步长. 它的好坏直接决定了算法计算时间和收敛性. 好的下降方向和搜索步长可使算法计算时间短, 收敛性好; 反之则计算时间长, 收敛性差. 在

算法的每次迭代过程中, 本文首先消除模型(6)的等式约束, 并求解出该模型消除等式约束后的梯度, 再以此确定出近似 Newton 方向, 最后用近似 Newton 方向到约束空间上的投影方向作为算法的下降方向; 同时用不精确搜索技术求解搜索步长. 利用这两项技术, 文中所给算法的收敛性好, 且计算时间比基于路径算法和基于起点算法要短.

2.1 梯度的确定

消除模型(6)的等式约束及确定该模型消除等式约束后的梯度, 是本算法的核心, 也是该算法的难点所在. 该步骤直接决定了算法的计算时间. 为简单有效地确定出模型(6)消除等式约束后的梯度, 本文特给出如下方法:

第一步: 对网络中的任意一终点 n , 用网络中所有关于终点 n 的分流率非零的有向路段组成子网 SN_n^0 , 并求解出子网 SN_n^0 中各个节点 i 到终点 n 的最大成本 $C_i^n = \sum_{r \in S_m[SN_n^0]} c_r$, 再将 \mathcal{A} 中所有满足 $C_{a_i}^n > C_{a_h}^n$ 的有向路段 $a = [a_i, a_h]$ 添加到子网 SN_n^0 中, 以获得子网 SN_n .

其中: $S_m[SN_n^0]$ 表示子网 SN_n^0 从节点 i 到终点 n 的所有路径集合, $c_r = \sum_{a \in r} c_a$ 为路径 r 的成本.

定理 2 按照第一步中所给方法确定的子网 SN_n 是不带环的, 且 SN_n 的节点集合为 \mathcal{N} .

证明 显然 SN_n^0 是不带环的, 且有 SN_n^0 的节点集合为 \mathcal{N} ; 从而有 SN_n 的节点集合也为 \mathcal{N} . 因此只需证明经过向 SN_n^0 中添加有向路段之后, 所确定的 SN_n 是不带环的即可. 用反证法来证明该结论. 反设 SN_n 是带环的, 也就是说 SN_n 中存在一段路径 $[v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_m = v_1]$. 因为 SN_n^0 是不带环的, 路径 $[v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_m = v_1]$ 中的所有路段可分成两部分: 一部分是属于 SN_n^0 , 另一部分不属于 SN_n^0 , 且这两部分都是非空的. 不失一般性假设 $[v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_m]$ 属于 SN_n^0 , $[v_m \rightarrow v_{m+1} \rightarrow \dots \rightarrow v_m = v_1]$ 不属于 SN_n^0 . 因此 $C_{v_1}^n \geq C_{v_2}^n \geq \dots \geq C_{v_m}^n > C_{v_{m-1}}^n > \dots > C_{v_m}^n = C_{v_1}^n$, 矛盾. 这说明 SN_n 是不带环的.

由定理2可知,子网 SN_n 是不带环的,因此可按照如下的规则重新定义子网 SN_n 中各个节点 i 的顺序 $o_n(i)$

$$\forall a = [a_i, a_h] \in SN_n \Rightarrow o_n(a_i) > o_n(a_h),$$

其中 o_n 是一映射: $o_n: \mathcal{N} \rightarrow \{1, 2, \dots, N\}$.

第二步:重新确定出子网 SN_n 中各个节点 i 的顺序 $o_n(i)$,并按该顺序求解出网络中各节点 i 的平均成本 σ_i^n 以及各有向路段 a 的平均成本 μ_a^n . 其中, σ_i^n 和 μ_a^n 分别为

$$\begin{aligned} \sigma_i^n &= \sum_{r \in S_{i,n}} c_r \cdot \prod_{a \subseteq r} \alpha_a^n, \quad 1 \leq i, n \leq N, i \neq n, \\ \mu_a^n &= \sum_{r \in S_{a,n}, a \subseteq r} c_r \cdot \prod_{b \subseteq r, b \neq a} \alpha_b^n. \end{aligned}$$

显然有

$$\sigma_i^n = \sum_{a \in \mathcal{A}, a_i=1} \alpha_a^n \cdot \sum_{r \in S_{i,n}, a \subseteq r} c_r \cdot \prod_{a' \subseteq r, a' \neq a} \alpha_{a'}^n = \sum_{a \in \mathcal{A}, a_i=1} \alpha_a^n \cdot \mu_a^n, \quad (7)$$

$$\mu_a^n = c_a + \sigma_{a_h}^n. \quad (8)$$

第三步:对 SN_n 中任意一节点 $i (i \neq n)$, 将集合 $A(i)$ 中的有向路段分成两个子集 B_i^n 和 NB_i^n , 使得 B_i^n 中有且仅有一个元素,且满足下式,

$$\mu_b^n \leq \mu_a^n, \quad \forall b \in B_i^n, \forall a \in NB_i^n.$$

将 B_i^n 中的有向路段称为节点 i 处关于终点 n 的基路段,并记为 b_i^n ; 将 NB_i^n 中的所有有向路段称为节点 i 处关于终点 n 的非基路段; 将基路段和非基路段相对应的变量分别称为基变量和非基变量,并分别记为 $\alpha_{b_i^n}^n$ 和 $\alpha_a^n, \forall a \in NB_i^n$; 用 $NB^n = \bigcup_{i \in \mathcal{N}, i \neq n} NB_i^n$ 表示网络 G 中关于终点 n 的所有非基路段集合. 因此根据(3)式可将 α 描述成

$$\frac{\partial x_{a'}}{\partial \hat{\alpha}_a^n} = \frac{\partial x_{a'}}{\partial \alpha_a^n} - \frac{\partial x_{a'}}{\partial \alpha_{b_i^n}^n}$$

$$\alpha_{b_i^n}^n(\hat{\alpha}) = 1 - \sum_{a \in NB_i^n} \hat{\alpha}_a^n, \quad \forall i \in \mathcal{N} \setminus \{n\}, \forall n \in D, \quad (9a)$$

$$\alpha_a^n(\hat{\alpha}) = \hat{\alpha}_a^n, \quad \forall a \in NB_i^n, \forall n \in D, \quad (9b)$$

其中 $\hat{\alpha} = (\hat{\alpha}^1, \hat{\alpha}^2, \dots, \hat{\alpha}^N), \hat{\alpha}^n = (\dots, \hat{\alpha}_a^n, \dots), a \in NB^n$. 从而模型(6)可以转化成

$$\begin{aligned} \min \mathcal{F} &= \sum_{a \in \mathcal{A}} \int_0^{x_a(\hat{\alpha})} c_a(\bar{\omega}) d\bar{\omega}, \\ \text{s. t. } \hat{\alpha}_a^n &\geq 0, \quad \forall a \in NB^n, \forall n \in D. \end{aligned} \quad (10)$$

第四步:根据下式确定出模型(10)目标函数 \mathcal{F} 关于非基变量 $\hat{\alpha}_a^n (\forall a \in NB^n, n \in D)$ 的偏导.

$$\frac{\partial \mathcal{F}}{\partial \hat{\alpha}_a^n} = g_{a_i}^n \cdot (\mu_a^n - \mu_{b_i^n}^n), \quad \forall a \in NB^n, n \in D. \quad (11)$$

(11)式由如下的定理保证.

定理3 模型(6)的目标函数 \mathcal{F} 关于非基变量 $\hat{\alpha}_a^n (\forall a \in NB, \forall n \in D)$ 的偏导数为

$$\frac{\partial \mathcal{F}}{\partial \hat{\alpha}_a^n} = g_{a_i}^n \cdot (\mu_a^n - \mu_{b_i^n}^n), \quad \forall a \in NB^n, n \in D.$$

证明 根据(2)和(4)式可知,

$$\begin{aligned} g_i^n &= \sum_{j=1}^N \sum_{r \in S_{j,i}, i \in r} q_{j,n} \prod_{a \subseteq r} \alpha_a^n = \\ &= \sum_{j=1}^N q_{j,n} \cdot \sum_{r \in S_{j,i}, a \subseteq r} \prod_{a \subseteq r} \alpha_a^n \cdot \sum_{r' \in S_{i,n}, a \subseteq r'} \prod_{a \subseteq r'} \alpha_a^n = \\ &= \sum_{j=1}^N q_{j,n} \cdot \sum_{r \in S_{j,i}, a \subseteq r} \prod_{a \subseteq r} \alpha_a^n, \quad \forall i, n \in \mathcal{N}, i \neq n. \end{aligned}$$

为了叙述方便,令 $\chi_{j \rightarrow i}^n = \sum_{r \in S_{j,i}} \prod_{a \subseteq r} \alpha_a^n, \forall n \in D, 1 \leq i, j \leq N, j \neq i$.

因此

$$\begin{aligned}
 &= \sum_{i=1}^N \sum_{\substack{r \in S_{m_i} \\ a' \subseteq r; a \subseteq r}} q_m \cdot \prod_{\substack{e \subseteq r; \\ e \neq a}} \alpha_e^n - \sum_{i=1}^N \sum_{\substack{r \in S_{m_i} \\ a' \subseteq r; b_{a_i}^n \subseteq r}} q_m \cdot \prod_{\substack{e \subseteq r; \\ e \neq b_{a_i}^n}} \alpha_e^n = \\
 &\begin{cases} \sum_{j=1}^N q_{j_n} \cdot \sum_{r \in S_{j_n}} \prod_{a \subseteq r} \alpha_a^n, & a' = a \\ - \sum_{j=1}^N q_{j_n} \cdot \sum_{r \in S_{j_n}} \prod_{a \subseteq r} \alpha_a^n & a' = b_{a_i}^n \\ 0, & a'_i = a_i, a' \neq a, a' \neq b_{a_i}^n \\ \sum_{i=1}^N \left\{ \sum_{\substack{r \in S_{m_i} \\ a' \subseteq r; a \subseteq r}} q_m \prod_{\substack{e \subseteq r; \\ e \neq a}} \alpha_e^n - \sum_{\substack{r \in S_{m_i} \\ a' \subseteq r; b_{a_i}^n \subseteq r}} q_m \prod_{\substack{e \subseteq r; \\ e \neq b_{a_i}^n}} \alpha_e^n \right\}, & \text{其他} \end{cases} \\
 &= \begin{cases} g_{a_i}^n, & a' = a \\ -g_{a_i}^n, & a' = b_{a_i}^n \\ 0, & a'_i = a_i, a' \neq a, a' \neq b_{a_i}^n \\ \alpha_{a_i}^n g_{a_i}^n (\chi_{a_h \rightarrow a_i}^n - \chi_{b_{a_i}^n \rightarrow a_i}^n), & o_n(a_i) > o_n(a'_i) \\ 0, & o_n(a_i) < o_n(a'_i). \end{cases}
 \end{aligned}$$

所以

$$\frac{\partial \mathcal{F}}{\partial \alpha_a^n} = g_{a_i}^n \cdot (\mu_a^n - \mu_{b_{a_i}^n}^n), \forall a \in NB^n, n \in D.$$

$$\begin{aligned}
 \frac{\partial \mathcal{F}}{\partial \alpha_a^n} &= \sum_{a' \in \mathcal{A}} \frac{\partial \mathcal{F}}{\partial x_{a'}} \cdot \frac{\partial x_{a'}}{\partial \alpha_a^n} = \sum_{a' \in \mathcal{A}} c_{a'} \cdot \frac{\partial x_{a'}}{\partial \alpha_a^n} = \\
 &g_{a_i}^n \cdot \left[c_a - c_{b_{a_i}^n} + \sum_{\substack{a' \in SN_{a_i} \\ o_n(a_i) > o_n(a'_i)}} ca' \cdot \alpha_{a'}^n \cdot \right. \\
 &\left. (\chi_{a_h \rightarrow a_i}^n - \chi_{b_{a_i}^n \rightarrow a_i}^n) \right]. \tag{12}
 \end{aligned}$$

又因为

$$\begin{aligned}
 \mu_a^n &= c_a + \sum_{r \in S_{a_h}, a \subseteq r} c_r \cdot \prod_{a' \subseteq r} \alpha_{a'}^n = \\
 &c_a + \sum_{r \in S_{a_h}, a \subseteq r} \prod_{a' \subseteq r} \alpha_{a'}^n \cdot \sum_{e \subseteq r} c_e = \\
 &c_a + \sum_{\substack{a' \in SN_{a_i} \\ o_n(a_i) > o_n(a'_i)}} ca' \cdot \chi_{a_h \rightarrow a_i}^n \cdot \alpha_{a'}^n \cdot \chi_{a_i \rightarrow n}^n = \\
 &c_a + \sum_{\substack{a' \in SN_{a_i} \\ o_n(a_i) > o_n(a'_i)}} c_{a'} \cdot \chi_{a_h \rightarrow a_i}^n \cdot \alpha_{a'}^n, \\
 &\forall a \in \mathcal{A}, \forall n \in D.
 \end{aligned}$$

从而

2.2 下降方向及搜索步长的确定

根据(12)式可知, 目标函数关于 $\alpha_a^n (\forall a \in NB, \forall n \in D)$ 的二阶导数为

$$\begin{aligned}
 \frac{\partial^2 \mathcal{F}}{\partial \alpha_a^{n^2}} &= \sum_{a' \in \mathcal{A}} \left[\frac{\partial^2 \mathcal{F}}{\partial x_{a'}^2} \cdot \left(\frac{\partial x_{a'}}{\partial \alpha_a^n} \right)^2 + \frac{\partial \mathcal{F}}{\partial x_{a'}} \cdot \left(\frac{\partial^2 x_{a'}}{\partial \alpha_a^{n^2}} \right) \right] = \\
 &\sum_{a' \in \mathcal{A}} c_{a'}' \cdot \left(\frac{\partial x_{a'}}{\partial \alpha_a^n} \right)^2 = \\
 &c_a' \cdot g_{a_i}^{n^2} + c_{b_{a_i}^n}' \cdot g_{a_i}^{n^2} + \sum_{\substack{a' \in SN_{a_i} \\ o_n(a_i) > o_n(a'_i)}} c_{a'}' \cdot \alpha_{a'}^{n^2} \cdot g_{a_i}^{n^2} \cdot \\
 &(\chi_{a_h \rightarrow a_i}^n - \chi_{b_{a_i}^n \rightarrow a_i}^n)^2, \tag{13}
 \end{aligned}$$

其中 $c_a' (\forall a \in \mathcal{A})$ 表示有向路段 a 的成本函数的一次导数.

显然(13)式在实际计算过程中很难十分精确地确定, 为此本文将用其下界 $g_{a_i}^{n^2} \cdot (c_a' + c_{b_{a_i}^n}')$ 来近似替代. 从而可得到如下的近似 Newton 方向,

$$\Delta d_a^n = \frac{\mu_{b_{a_i}}^n - \mu_a^n}{(c'_a + c'_{b_{a_i}}) \cdot g_{a_i}^n}, \forall a \in NB^n, n \in D.$$

为了提高算法的收敛性和确保每次迭代过程中所有变量均满足约束条件, 本文用近似 Newton 方向到约束空间上的投影作为算法的下降方向, 并采用如下的不精确搜索技术确定搜索步长:

$$\Delta \alpha_a^n = \begin{cases} \max\{-\alpha_a^n, \lambda \cdot \Delta d_a^n\}, & g_{a_i}^n > 0 \\ -\alpha_a^n, & g_{a_i}^n = 0 \end{cases}, \forall a \in NB^n, n \in D, \quad (14a)$$

$$\Delta \alpha_a^n = 0, \quad \forall a \in \mathcal{A}/SN_n, \quad (14b)$$

$$\Delta \alpha_{b_i}^n = - \sum_{a \in A(i); a \neq b_i^n} \Delta \alpha_a^n, \quad \forall i \in \mathcal{N}, \forall n \in D, i \neq n, \quad (14c)$$

其中

$$\lambda = \max\{2^{-m} (m = 0, 1, 2, \dots) \mid \mathcal{F}(\alpha + \Delta \alpha) < \mathcal{F}(\alpha)\}.$$

2.3 算法具体步骤

据上述, 模型(6)的求解算法——基于分流率算法的具体步骤如下:

步1 初始化. 对网络 G 中所有终点 $n \in D$, 根据各个路段的自由流成本确定出 G 中各节点到终点 n 的最短路集, 用 SN_n 表示由这些最短路集所组成的子网; 并根据最短路集进行 0-1 分配, 同时确定出节点 i 到终点 n 的交通量 $g_i^n (1 \leq i \leq N, i \neq n)$; 再根据下式确定出 α 的初始值 $\alpha_k (k=1)$,

$$\alpha_{a_k}^n = \begin{cases} 1, & \forall a \in SN_n \\ 0, & \forall a \in \mathcal{A} \setminus SN_n. \end{cases} \quad (15)$$

步2 收敛性判定. 对一给定的正小数 ϵ , 如果 $\frac{\mathcal{F}(\alpha_k - 1) - \mathcal{F}(\alpha_k)}{\mathcal{F}(\alpha_k)} \leq \epsilon, k \geq 2$, 则根据(4)式确定出网络中所有非零交通量路径的路径交通量, 否则转步3.

步3 主循环. 按照如下步骤更新 α_k , 并确定出当前网络中各个路段的交通量, 交通成本, 交通成本导数以及节点 i 到终点 n 的交通量 $g_i^n (\forall i, n \in D$ 且

$i \neq n)$; 然后转步2.

(1) 对终点 n 令其从 1 到 N , (i) 从子网 SN_n 中删除所有到终点 n 的路段交通量为零的有向路段; (ii) 求解此时子网 SN_n 中各个节点 i 到终点 n 的最大成本 C_i^n ; (iii) 将 \mathcal{A} 所有满足 $C_{a_i}^n > C_{a_n}^n$ 的有向路段 $a = [a_i, a_n]$ 添加到子网 SN_n 中; (iv) 确定出此时子网 SN_n 各个节点 i 的顺序 $o_n(i)$; 并按照该顺序求解出 $\mu_a^n (\forall a \in SN_n)$; (v) 对 SN_n 中任一节点 $i (i \neq n)$, 确定出 $A(i)$ 的基路段 b_i^n , 使得 $\mu_{b_i}^n = \min_{a \in A(i)} \mu_a^n$; 并以此确定出 NB^n .

(2) 根据(14)式确定出 $\Delta \alpha_k$,

(3) 令 $\alpha_{k+1} = \alpha_k + \Delta \alpha_k; k = k + 1$; 并更新当前网络上各个路段上的交通量, 交通成本, 交通成本导数以及节点 i 处到终点 n 的交通量 $g_i^n (\forall i, n \in D$ 且 $i \neq n)$.

2.4 收敛性证明

定理4 对基于分流率算法中任意的步 Π 前确定出的 α_k 及其相对应的 $SN_n (\forall n \in D)$, 均存在 $\hat{\lambda} > 0$, 使得 $0 < \lambda \leq \hat{\lambda}$ 满足下式,

$$\mathcal{F}(\alpha_k + \Delta \alpha_k) < \mathcal{F}(\alpha_k),$$

其中 $\Delta \alpha_k$ 是由(14)式所确定的.

证明 因为当 $\lambda \rightarrow 0$ 时有

$$\Delta \hat{\alpha}_{a_k}^n = \Delta \alpha_{a_k}^n = \frac{\lambda \cdot (\mu_{b_{a_i}}^n - \mu_a^n)}{(c'_a + c'_{b_{a_i}}) \cdot g_{a_i}^n}, \forall a \in NB^n, n \in D.$$

所以

$$\begin{aligned} \mathcal{F}(\alpha_k + \Delta \alpha_k) - \mathcal{F}(\alpha_k) &= \mathcal{F}(\hat{\alpha}_k + \Delta \hat{\alpha}_k) - \mathcal{F}(\hat{\alpha}_k) = \\ &= \sum_{n \in D} \sum_{a \in NB^n} \Delta \hat{\alpha}_{a_k}^n \cdot \frac{\partial \mathcal{F}}{\partial \alpha_{a_k}^n} + o(\lambda) = \\ &= \sum_{n \in D} \sum_{a \in NB^n} \Delta \hat{\alpha}_{a_k}^n \cdot g_{a_i}^n \cdot (\mu_a^n - \mu_{b_{a_i}}^n) + o(\lambda) = \\ &= \sum_{n \in D} \sum_{a \in NB^n} \frac{\lambda \cdot (\mu_{b_{a_i}}^n - \mu_a^n)}{(c'_a + c'_{b_{a_i}})} \cdot (\mu_a^n - \mu_{b_{a_i}}^n) + o(\lambda) = \\ &= \sum_{n \in D} \sum_{a \in NB^n} \frac{-\lambda \cdot (\mu_{b_{a_i}}^n - \mu_a^n)^2}{(c'_a + c'_{b_{a_i}})} + o(\lambda) \leq 0. \end{aligned}$$

这说明一定存在 $\hat{\lambda} > 0$, 使得当 $0 < \lambda \leq \hat{\lambda}$ 时有 $\mathcal{F}(\alpha +$

$\Delta\alpha) < \mathcal{F}(\alpha)$.

定理 5 基于分流率算法收敛于模型(6)的极值点.

定理 6 若 α^* 是由基于分流率算法确定出来的, 则由该 α^* 及(4)式求解出的路径交通量满足交通分配的用户平衡条件.

上述定理中, 定理 4 保证了算法的可行性; 定理 5 和定理 6 则保证了算法的收敛性和有效性. 定理 5 和定理 6 的证明参见附录.

3 算例

从基于分流率算法的整个求解过程来看, 算法中需要求解网络中各个 OD 对的最短路一次, 确定网络中各个 OD 对的所有非零交通量路径一次, 并在每次迭代过程中求解出分流率, 网络中各个路段的交通量、交通成本、成本导数、各个节点的平均成本以及目标函数关于分流率的导数. 由于存储最短路和非零交通量路径所花内存相对较少, 且每次迭代过程中存储上述变量所需内存仅是网络中路段个数的线性函数, 因此同基于路径算法和基于起点算法相比, 基于分流率算法所需存储内存相对较少. 以文中提到的北京市网络为例, 基于分流率算法所需存储内存是 1.45 MB, 而基于路径算法和基于起点算法所需存储内存则分别是: 65 MB 和 2.14 MB.

为进一步说明基于分流率算法的有效性和优越性, 下面将通过实际的计算算例对该算法和其他能够求解路径交通量的算法(包括基于路径算法和基于起点算法)进行比较. 考虑到共轭投影梯度法是基于路径算法中最有效的算法, 文中将只针对如下的 3 个算法进行比较: 基于分流率算法(SRA), 共轭投影梯度法(CGP)和基于起点算法(OBA). 文中所有实验运行的硬件环境为: CPU: Intel P4 2000 MHz, 内存: 512 MB, 操作系统: Windows XP, 所有程序均采用 C# 编写. 所有算法将采用相同的数据结构来存储网络拓扑和网络数据以及相同的收敛准则; 并按如下步骤来对上述各个算法进行比较.

(1) 利用改进的 FW 算法^[13]以尽可能高的收敛准则求解原始的交通分配问题, 将得到的解认为是原始交通分配问题的理想最优解 ISO, 并记其相应的目标函数值为 \mathcal{F}_{ISO} . 采用该算法对原始交通分配

问题求解的主要原因是该方法所需存储内存较少, 且在实际计算过程中能找到较为理想的最优解.

(2) 由于交通分配问题存在唯一的路段交通量解^[1], 且目标函数值是有下界的, 文中每一个算法将采用如下的收敛准则

$$\epsilon = \frac{\mathcal{F}_k - \mathcal{F}_{ISO}}{\mathcal{F}_{ISO}}, \quad (16)$$

其中 \mathcal{F}_k 表示各个算法第 k 次迭代时的目标函数值. 当 ϵ 小于一个给定的小数时(如 1.0×10^{-n} , $n=0, 1, 2, \dots$), 算法结束.

文中将采用如下的网络对各个算法进行比较, 这些网络的基本数据见表 1.

表 1 测试网络基本数据

| 网络 | 起点 | 终点 | 节点 | 路段 | OD 对 |
|-------------|-----|-----|-----|-----|------|
| Sioux Falls | 24 | 24 | 24 | 76 | 528 |
| 北京 | 135 | 135 | 135 | 450 | 1482 |
| 网络 1 | 100 | 100 | 100 | 180 | 500 |
| 网络 2 | 225 | 225 | 225 | 420 | 1125 |
| 网络 3 | 400 | 400 | 400 | 760 | 2000 |

其中网络 1、网络 2 和网络 3 均是格网络, 分别为 10×10 , 15×15 和 20×20 的格网络.

图 1—4 表明在对实际网络的求解过程中, 文中所给的算法无论是在计算时间上, 还是在计算精度上都比其他算法要好. 且在相同的计算精度下, 文中所给算法的计算时间仅是其他算法的三分之一左右, 甚至更小.

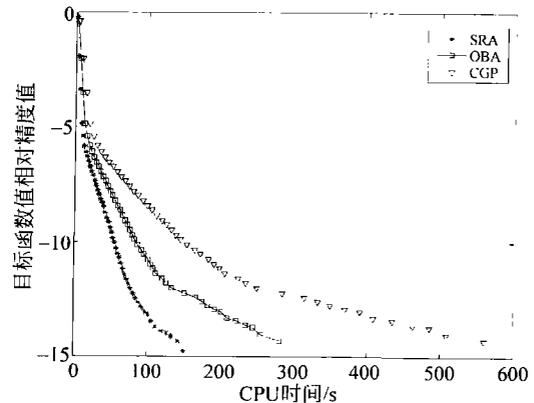


图 1 ϵ —CPU 时间 (Sioux Falls)

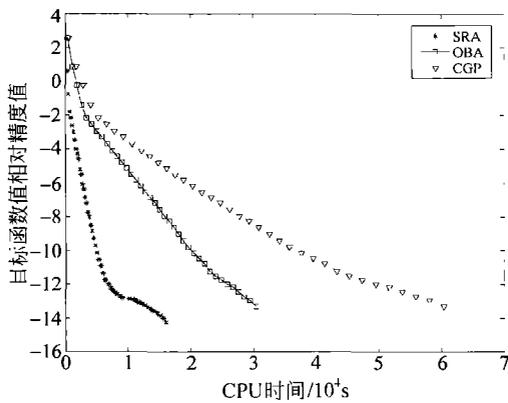


图2 ε -CPU时间(北京)

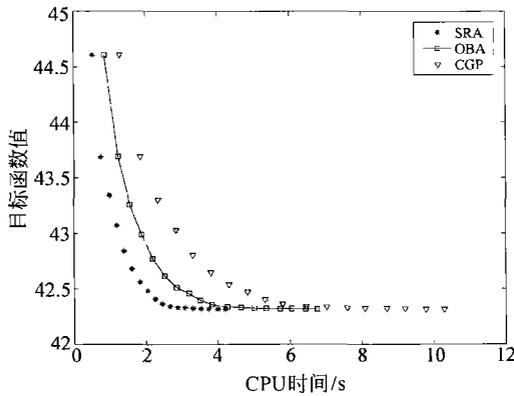


图3 目标函数值—CPU时间(Sioux Falls)

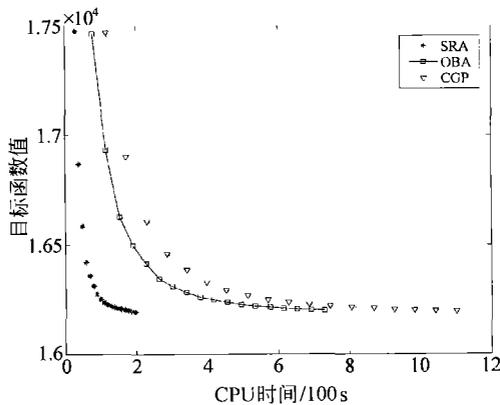


图4 目标函数值—CPU时间(北京)

表2中给出了不同精度下,各个算法对于随机网络1,2和3的计算时间.该表进一步说明了在相同的计算精度下,文中所给算法在计算时间上明显优于其他算法.

表2 计算时间 (单位:h)

| 网络 | 算法 | 相对精度 | | | | |
|-----|-----|------|-------|-------|--------|-----------------------|
| | | 0.1 | 0.01 | 0.001 | 0.0001 | 1.0×10^{-10} |
| 网络1 | CGP | 0.58 | 2.13 | 11.78 | 17.95 | 55.65 |
| | OBA | 0.24 | 1.17 | 6.40 | 9.73 | 27.12 |
| | SRA | 0.08 | 0.31 | 0.85 | 1.55 | 6.25 |
| 网络2 | CGP | 4.93 | 17.38 | 54.91 | 110.22 | >360 |
| | OBA | 2.60 | 9.96 | 28.79 | 53.21 | >360 |
| | SRA | 0.51 | 1.58 | 3.91 | 7.14 | 68.61 |
| 网络3 | CGP | 7.08 | 20.86 | 65.64 | 130.35 | >360 |
| | OBA | 3.63 | 11.86 | 33.18 | 64.53 | >360 |
| | SRA | 1.06 | 2.15 | 5.11 | 9.48 | 101.65 |

4 结论

本文考虑了用户平衡交通分配的求解问题,给出了一种基于分流率的路径交通量求解方法.该方法是通过对一个基于分流率的交通分配模型确定路径交通量.文中首先建立基于分流率的交通分配模型,然后给出该模型的求解算法,并从理论上证明该算法的可行性和收敛性,最后通过算例说明算法的有效性.计算结果表明:文中所给方法适合于大规模网络的交通分配的求解问题;且在计算时间、计算精度和所需存储内存上比基于路径算法和基于起点算法都要优越;在相同的计算精度下计算时间仅仅是这些算法的三分之一,甚至更小.

附录

定理5的证明.考虑到模型(10)的K-T条件为

$$\frac{\partial \mathcal{F}}{\partial \alpha_a^n} = g_{a_i}^n \cdot (\mu_a^n - \mu_{b_{a_i}}^n) \geq 0, \forall a \in NB^n, n \in D,$$

$$\alpha_a^n \cdot \frac{\partial \mathcal{F}}{\partial \alpha_a^n} = \alpha_a^n \cdot g_{a_i}^n \cdot (\mu_a^n - \mu_{b_{a_i}}^n) = 0, \forall a \in NB^n, n \in D.$$

从而模型(6)的K-T条件为

$$g_{a_i}^n \cdot (\mu_a^n - \mu_{b_{a_i}}^n) \geq 0, \forall a \in NB^n, n \in D, \quad (A1a)$$

$$\alpha_a^n \cdot g_{a_i}^n \cdot (\mu_a^n - \mu_{b_{a_i}}^n) = 0, \forall a \in NB^n, n \in D. \quad (A1b)$$

因此要想证明基于分流率算法收敛于模型(6)的极值点,只需证明,该算法收敛时所确定出的 α^* 满足(A1a, b)式即可.

根据定理4可知,对任意的 $k, k=1, 2, \dots$ 存在 $\hat{\lambda}_k > 0$ 使得

$$\mathcal{F}(\alpha_k) - \mathcal{F}(\alpha_{k+1}) \geq \sum_{n \in D} \sum_{a \in NB^n} \frac{\hat{\lambda}_k \cdot (\mu_{a_i}^n - \mu_a^n)^2}{(c'_a + c'_{b_{a_i}})},$$

从而

$$\begin{aligned} \mathcal{F}(\alpha_1) - \mathcal{F}(\alpha_{k+1}) &= \sum_{i=1}^k \mathcal{F}(\alpha_i) - \mathcal{F}(\alpha_{i+1}) \\ &\geq \sum_{i=1}^k \sum_{n \in D} \sum_{a \in NB^n} \frac{\hat{\lambda}_k \cdot (\mu_{a_i}^n - \mu_a^n)^2}{(c'_a + c'_{b_{a_i}})}, \end{aligned}$$

因为 $\mathcal{F}(\alpha)$ 是有界的, 所以 $\mathcal{F}(\alpha_1) - \mathcal{F}(\alpha_{k+1})$ 也是有界的. 因

此正项级数 $\sum_{i=1}^k \sum_{n \in D} \sum_{a \in NB^n} \frac{\hat{\lambda}_k \cdot (\mu_{a_i}^n - \mu_a^n)^2}{(c'_a + c'_{b_{a_i}})}$ 是收敛的. 从而当 $k \rightarrow \infty$ 时有

$$\sum_{n \in D} \sum_{a \in NB^n} \frac{\hat{\lambda}_k \cdot (\mu_{a_i}^n - \mu_a^n)^2}{(c'_a + c'_{b_{a_i}})} \rightarrow 0,$$

因此

$$\lim_{k \rightarrow \infty} \hat{\lambda}_k = 0, \tag{A2}$$

和

$$\mu_{a_i}^n = \mu_a^n, \forall a \in NB^n, n \in D, \tag{A3}$$

两式中至少有一个成立.

当(A2)式成立时, $\{\alpha_k, k=1, 2, \dots\}$ 有极限, 算法将收敛于其极限点 α^* . 从而有

$$\begin{aligned} \Delta \alpha_a^{n*} &= 0, \forall a \in NB^n, n \in D, \\ \mu_a^n - \mu_{b_{a_i}}^n &\geq 0, \forall a \in NB^n, n \in D, \end{aligned}$$

因此只需证明 α^* 满足式(A1b)即可. 用反证法来证明, 反设至少存在一终点 n 及其相应的有向路段 $a \in NB^n$ 使得(A1b)不成立, 即

$$\alpha_a^{n*} \cdot \frac{\partial \mathcal{F}}{\partial \alpha_a^{n*}} = \alpha_a^{n*} \cdot g_{a_i} \cdot (\mu_a^n - \mu_{b_{a_i}}^n) > 0,$$

从而存在 $\lambda^* > 0$ 使得

$$\Delta \alpha_a^{n*} = \frac{\lambda^* \cdot (\mu_{b_{a_i}}^n - \mu_a^n)}{(c'_a + c'_{b_{a_i}}) \cdot g_{a_i}} > 0.$$

矛盾. 这说明 α^* 是目标函数的极值点.

当(A3)式成立时, (A1a, b)式显然成立; 因此基于分流率算法收敛于模型(6)的极值点. 定理6的证明. 根据定理

4可知, α^* 是目标函数的极值点, 则其必满足

$$\mu_a^n - \mu_{b_{a_i}}^n \geq 0, \forall a \in NB^n, n \in D, \tag{A4a}$$

$$\alpha_a^{n*} \cdot g_{a_i} \cdot (\mu_a^n - \mu_{b_{a_i}}^n) = 0, \forall a \in NB^n, n \in D, \tag{A4b}$$

$$\alpha_a^{n*} = 0, \forall a \in \mathcal{A} \setminus SN_n. \tag{A4c}$$

其中 $NB^n, SN_n (n \in D)$ 均是算法1收敛时确定出的. 由算法1中 SN_n 的确定方法可知, 算法1收敛时 $NB^n, SN_n (n \in D)$ 是惟一不变的. 因此当算法收敛时有

$$\alpha_a^{n*} \neq 0, \forall a \in SN_n, n \in D. \tag{A5}$$

从而根据(4)式可知, 当算法1收敛时, 网络中任一 OD 对 $in (i \neq n)$ 间的所有被占用(交通流量非零)的路径均属于 SN_n , 而所有未被占用的路径不属于 SN_n .

因此只需证明 SN_n 中任一 OD 对 $in (i \neq n)$ 间所有路径的成本均相等, 且小于原始网络 G 中该 OD 对的所有未被占用路径的成本即可.

首先来证明 SN_n 中任一 OD 对 $in (i \neq n)$ 间所有路径的成本均相等(为方便起见将其记为 c_i^n). 用数学归纳法来证明, 对 SN_n 中节点 i 的拓扑顺序 $o_n(i)$ 进行归纳.

当 $o_n(i) = 2$ 时, SN_n 中 OD 对 in 间有且仅有一条路径, 结论自然成立.

假设当 $o_n(i) \leq k$ 时结论成立, 也就是说, 当 $o_n(i) \leq k$ 时, 有

$$\begin{aligned} \sigma_i^n &= \sum_{r \in S_{in}^n} c_r \cdot \prod_{a \subseteq r} \alpha_a^{n*} = \sum_{r \in S_{in}^n} c_r \cdot \prod_{a \subseteq r} \alpha_a^{n*} = \\ c_i^n &\cdot \sum_{r \in S_{in}^n} \prod_{a \subseteq r} \alpha_a^{n*} = c_i^n \cdot \sum_{r \in S_{in}^n} \prod_{a \subseteq r} \alpha_a^{n*} = \\ c_i^n. \end{aligned} \tag{A6}$$

下面将证明当 $o_n(i) = k+1$ 时结论也成立. 由(A4a-c)和(A5)式可知

$$\mu_a^n = \mu_{b_i}^n, \forall a \in A(i) \cap NB^n, \tag{A7}$$

且根据假设有

$$\mu_a^n = c_a + \sigma_{a_h}^n = c_a + c_{a_h}^n, \forall a \in A(i) \cap NB^n, \tag{A8}$$

$$\mu_{b_i}^n = c_{b_i}^n + \sigma_{b_i h}^n = c_{b_i}^n + c_{b_i h}^n. \tag{A9}$$

因此当 $o_n(i) = k+1$ 时结论成立.

下面将证明 SN_n 中任一 OD 对 $in (i \neq n)$ 间的路径成本 c_i^n , 小于原始网络 G 中该 OD 对的所有未被占用路径的成本. 采用反证法来证明, 反设 G 中至少存在一条路径 $\bar{r} = [i = v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_m = n]$ 使得

$$c_r < c_i^n, \hat{r} \notin S_m[SN_n]. \quad (A10)$$

因此 \hat{r} 上至少存在一个有向路段不属于 SN_n . 不失一般性假设 \hat{r} 上只有有向路段 $\hat{a} = [v_k, v_{k+1}] \notin SN_n$. 从而

$$\begin{aligned} c_r &= c_r' + c_a + c_r'', \\ c_i^n &= c_r' + c_{v_k}^n, \end{aligned}$$

其中 $r' = [i=v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_k]$, $r'' = [v_{k+1} \rightarrow v_{k+2} \rightarrow \dots \rightarrow v_m = m]$. 则根据(A10)式可知

$$c_a + c_r'' < c_{v_k}^n. \quad (A11)$$

又由算法1中 SN_n 的确定方法可知,

$$C_{v_k}^m \leq C_{v_{k+1}}^m,$$

且

$$\begin{aligned} C_{v_k}^m &= c_{v_k}^n, \\ C_{v_{k+1}}^m &= c_r'', \end{aligned}$$

从而有

$$c_{v_k}^n \leq c_r''.$$

与(A11)式矛盾. 因此由该 α^* 确定出的路径交通量满足交通分配的用户平衡条件.

参 考 文 献

- 1 Yosef Sheffi. Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods. Englewood cliffs, N. J: Prentice-Hall Inc, 1984
- 2 Wardrop J G. Some theoretical aspects of road traffic research. Proceedings of the Institute of Civil Engineers, Part II, 1952, 325—378
- 3 陆化谱, 宿 良, 史其信, 等著. 交通规划理论与方法. 北京: 清华大学出版社, 1998
- 4 Beckmann M, McGuire C B, Winster C B. Studies in the Economics of Transportation. New Heaven, CT: Yale University Press, 1956
- 5 Leblanc L J, Morlok E K, Pierskalla W. An efficient approach to solving the road network equilibrium traffic assignment problem. Transportation Research, 1975, 9(4): 309—318
- 6 Larsson T, Patriksson M. Simplicial decomposition with disaggregated representation for the traffic assignment problem.

- Transportation Science, 1992, 26(1): 4—17
- 7 Bertsekas D. On the Goldstein-Levitin-Polyak gradient projection method. IEEE Transaction on Automatic Control, 1976, 21(2): 174—183
- 8 Jayakrishnan R, Tsai W K, Prashker J N. Faster path-based algorithm for traffic assignment. Transportation Research Record, 1994, 1443(1): 75—83
- 9 Lee D H, Yu N, Chen A. A conjugate gradient projection algorithm for the traffic assignment problem. Mathematical and Computer Modeling, 2003, 37(7/8): 863—878
- 10 Leblanc L J, Helgason R V, David E B. Improved efficiency of the Frank Wolfe algorithm for convex network problems. Transportation Science, 1985, 19(4): 445—462
- 11 Masao Fukushima. A modified frankwolfe algorithm for solving the traffic assignment problem. Transportation Research Part B, 1984, 18(2): 169—177
- 12 Anders Weintraub, Carmen Ortiz. Accelerating convergence of the Frank Wolfe algorithm. Transportation Research Part B, 1985, 19(2): 113—122
- 13 Lee D H, Yu N. Accelerating strategies and computational studies of the Frank Wolfe algorithm for the traffic assignment problem. Transportation Research Record, 2001, 1771(1): 97—105
- 14 Gao Z Y, Lam W H K, Wong S C, et al. The convergence of equilibrium algorithms with non-monotone line search technique. Applied Mathematics and Computation, 2004, 148(1)1—13
- 15 Hillel B G. Origin-based algorithm for the traffic assignment problem. Transportation Science, 2002, 36(4): 398—417
- 16 Rossi T F, McNeil S, Hendrickson C. Entropy model for consistent impact fee assessment. Journal of Urban Planning and Development/ASCE, 1989, 115(1): 51—63
- 17 Larsson T, Lundgren J, Parkriksson M, et al. Most likely traffic equilibrium route flows analysis and computation. Report LiTH-MAT-R-1999-05, Department of Mathematics. Linköping Institute of Technology, Linköping, Sweden
- 18 Hillel B G H, Boyce D. Route flow entropy maximization in origin-base traffic assignment. Proceedings of the 14th International Symposium on Transportation and Traffic Theory, Jerusalem, Israel, 1999
- 19 李 峰, 王书宁. 基于终点的路径交通量确定方法. 清华大学学报(自然科学版), 2006, 46(1): 37—40
- 20 美国交通研究委员会专题报告 209 号. 任福田, 等译. 道路通行能力手册. 北京: 中国建筑工业出版社, 1991