

全路径 Logit 交通分配模型的求解方法^{*}

李 军, 聂佩林, 余 志
(中山大学工学院, 广东 广州 510275)

摘 要: 分析了求解 Logit 模型的现有算法的不足, 根据路网的连通特征提出了一种新的求解全路径 Logit 分配模型的迭代算法。给出了算法和 Logit 模型的等价性证明, 并和经典的 Dial 算法进行了比较。

关键词: Logit 分配模型; Dial 算法; 迭代法

中图分类号: U116.2 **文献标识码:** A **文章编号:** 0529-6579 (2004) 05-0124-03

Dial 通过合理路径的定义提出了一种有效求解 Logit 分配模型的算法^[1]。然而由于 Dial 算法对“有效路径”的定义过于严格, 导致分配结果中旅行成本较低的线路不被使用, 旅行成本较高的线路反而被使用的现象, 限制了 Dial 算法在实际中的应用^[2]。而 Bell 和 Akamatsu 提出全路径 Logit 模型的求解需要对高阶矩阵求逆^[3-5], 难以进行大型网络的计算。

本文首先回顾了 Dial 算法和 Bell Akamatsu 模型, 分析了两者的优缺点, 在此基础上提出了一种改进的 Dial 算法来求解全路径 Logit 模型。最后, 本文给出了计算实例, 并和经典的 Dial 算法进行比较, 分析两种算法的优缺点。

1 Logit 分配模型的算法回顾

Logit 分配模型要求出行者选择某条路径 p 的概率同该路径的阻抗的负指数幂成正比, 即

$$\Pr(p) = \frac{\exp(-\theta \cdot c_p)}{\sum_{i \in P} \exp(-\theta \cdot c_i)} \quad (1)$$

式中, $\Pr(p)$ 为出行者选择路径 p 的概率; $\theta > 0$ 为模型参数; P 为所有从起点到终点 s 的路径的集合。由于路径数目的巨大, 很难利用定义来直接求解 Logit 模型。Dial 提出了一种有效的算法来计算这一问题, 其出发点是通过判断一条路径是否为“合理路径”来排除大量的“不合理路径”, 在所定义的合理路径的集合中建立满足式 (1) 的递推公式, 从而达到避免路径枚举的目的。用 $r(i)$ 和 $s(i)$ 表示点 i 到起点 r 和终点 s 的最短路的阻抗, Dial 算法的“合理路径”的定义要求路段 $(i \rightarrow j)$ 满足

$$r(i) < r(j) \text{ 和 } s(i) > s(j) \quad (2)$$

不满足条件 (2) 的路段流量将为 0, 即该路段被

从网络中删除。研究表明, Dial 算法的“合理路径”的定义过于严格, 许多实际中被使用的路段的流量为 0, 而最短路上分配的流量过多。

由于 Dial 算法存在上述缺陷, Bell 提出了两种计算方法^[3]。第 1 种计算方法为近似算法, 其计算结果与计算的顺序有关, 其结果不具备惟一性和稳定性, 且缺乏理论依据证明计算结果和 Logit 模型的定义相符。Bell 的第 2 种方法假设网络的权重矩阵 W 的求和序列收敛并通过矩阵求逆来计算 Logit 模型。Akamatsu 后来证明 Bell 这一计算方法和包含所有路径的 Logit 模型是等价的, 并建立了一个仅包含路段流量变量的模型。Bell Akamatsu 模型没有排除任何路径, 所有无限循环的回路均包括在计算中, 是一种全路径 Logit 分配模型^[4,5]。对于一个大型网络而言, Bell Akamatsu 模型缺乏有效的计算方法, 因此有必要寻求新的算法。

2 改进的 Dial 算法

改进算法的出发点是将 Dial 算法“合理路径”这一约束去掉, 使得新的路径集合包含了所有的可能路径。新算法路段 $(i \rightarrow j)$ 的相似性 (likelihood) 由下式计算:

$$L_{(i \rightarrow j)} = \begin{cases} 0 & j = r \text{ 或 } i = s \\ \exp[\theta \cdot (r(j) - r(i) - t_{(i \rightarrow j)})] & \text{其它} \end{cases} \quad (3)$$

式中, $t_{(i \rightarrow j)}$ 为路段 $(i \rightarrow j)$ 的阻抗。这里将终点为 r 的路段及起点为 s 的路段的相似性定义为 0, 使得从起点 r 出发用户不再回到起点 r , 同时保证到达终点 s 的用户不再离开终点。

新算法定义路段 $(i \rightarrow j)$ 的权重为

^{*} 收稿日期: 2004-02-25

基金项目: 广东省科技厅攻关资助项目 (2002B60112); 中山大学青年教师科研启动基金资助项目

作者简介: 李军 (1968 年生), 男, 博士, E-mail: stsljun@zsu.edu.cn

$$w_{(i \rightarrow j)} = \begin{cases} L_{(i \rightarrow j)} & i = r \\ L_{(i \rightarrow j)} \sum_k w_{(k \rightarrow i)} & i \neq r \end{cases} \quad (4)$$

式 (4) 没有给出计算顺序, 因此式 (4) 实际上给出的一个 n 维方程组, 这里 n 为网络的路段数。可以通过解方程组得到每一路段的权重。

通过路段 $(i \rightarrow j)$ 的流量的计算和 Dial 算法类似, 由下式计算

$$x_{(i \rightarrow j)} = \begin{cases} \frac{q_{rs} \circ w_{(i \rightarrow j)}}{\sum_k w_{(k \rightarrow j)}} & j = s \\ \sum_l x_{(j \rightarrow l)} \frac{w_{(i \rightarrow j)}}{\sum_k w_{(k \rightarrow j)}} & j \neq s \end{cases} \quad (5)$$

式中, q_{rs} 为起点 r 到终点 s 之间的交通需求。式 (5) 也没有给出计算顺序, 同样需要求解方程组来得到路段流量。

式 (3)、(4) 和 (5) 与 Logit 模型的等价性证明和 Dial 所采用的方法类似, 下面给出具体的证明。首先注意路段 $(i \rightarrow j)$ 的相似性和出行者在点 i 选择路段 $(i \rightarrow j)$ 的概率成正比, 因此出行者选择某条路径 p 的概率和路径上所有路段的相似性的乘积成正比, 也就是说用户选择的路径 p 的概率为

$$\Pr(p) = K \prod_{(i \rightarrow j)} (L_{(i \rightarrow j)})^{\delta_{(i \rightarrow j), p}} \quad (6)$$

式中如果 $(i \rightarrow j)$ 在路径 p 上则 $\delta_{(i \rightarrow j), p} = 1$, 否则 $\delta_{(i \rightarrow j), p} = 0$; K 为常数。将式 (3) 代入式 (6) 得到

$$\Pr(p) =$$

$$\begin{aligned} & K \prod_{(i \rightarrow j)} \exp[\theta \circ (r(j) - r(i) - t_{(i \rightarrow j)} \circ \delta_{(i \rightarrow j), p})] = \\ & K \exp\left[\sum_{(i \rightarrow j)} \theta \circ (r(j) - r(i) - t_{(i \rightarrow j)} \circ \delta_{(i \rightarrow j), p})\right] = \\ & K \exp\left[\theta \left((r(s) - r(r) - \sum_{(i \rightarrow j)} \delta_{(i \rightarrow j), p} \circ t_{(i \rightarrow j)})\right)\right] = \\ & K \exp[\theta(r(s) - c_p)] \end{aligned} \quad (7)$$

由于所有路径被选择的概率之和为 1, 即

$$\sum_p \Pr(p) = \sum_p K \exp[\theta(r(s) - c_p)] = 1 \quad (8)$$

故有

$$K = \frac{1}{\sum_p \exp[\theta(r(s) - c_p)]} \quad (9)$$

代入 (7) 路径 p 被选择的概率为

$$\Pr(p) = \frac{\exp(-\theta c_p)}{\sum_l \exp(-\theta c_l)} \quad (10)$$

这和 Logit 模型的定义是相同的, 由于新算法没有对路径做任何限制, 因此有效路径包括所有可能的路径, 其中也包括了所有有限和无限循环的回路, 这表明是 (3)、(4)、(5) 定义了一个所有路径上

的 Logit 模型, 也就是前面所描述的 Bell-Akamatsu 模型。

现实世界中连接交叉路口的路段数目一般有限, 远远少于网络中路段的数目。例如一个互通的十字路口连接的路段数只有 8 个, 式 (4) 和 (5) 所表示的方程组事实上非常稀疏, 其特点使得式 (4) 和 (5) 非常适合用迭代法来求解。

为了加速算法的收敛, 新算法采用了和 Dial 算法相同的结点计算顺序。下面给出利用迭代法求解的详细步骤。

和 Dial 算法一样, 新算法首先计算权重。其求解过程为:

步骤 1 设定权重的收敛判别标准 ϵ_1 和最大迭代次数 M_1 , 设 $m = 0$, 并将所有路段的权重设为 0, 即 $w_{(i \rightarrow j)}^{(m)} = 0$;

步骤 2 将网络中的节点 i 从起点 r 依照 $r(i)$ 进行升序排列, 并按照顺序对每一个节点 i 根据下式计算 $w_{(i \rightarrow j)}^{(m+1)}$;

$$w_{(i \rightarrow j)}^{(m+1)} = \begin{cases} L_{(i \rightarrow j)} & i = r \\ L_{(i \rightarrow j)} \sum_k w_{(k \rightarrow j)}^{(m)} & i \neq r \end{cases} \quad (11)$$

步骤 3 判断 $w_{(i \rightarrow j)}^{(m+1)}$ 是否满足条件

$$\sqrt{\frac{\sum (w_{(i \rightarrow j)}^{(m+1)} - w_{(i \rightarrow j)}^{(m)})^2}{\sum (w_{(i \rightarrow j)}^{(m)})^2}} < \epsilon_1 \quad (12)$$

如果满足, 计算停止, 得到各路段的权重; 否则判断 m 是否大于, 如果 $m > M_1$, 计算停止; 否则设 $m = m + 1$, 转向步骤 3。

路段流量的求解过程和权重求解过程类似, 如下所示:

步骤 1 设定流量的收敛判别标准 ϵ_2 和最大迭代次数 M_2 , 设 $m = 0$, 并将所有路段的流量设为 0, 即 $x_{(i \rightarrow j)}^{(m)} = 0$;

步骤 2 将网络中的节点 i 从终点 s 依照 $r(i)$ 进行逆序排列, 并按照顺序对每一个节点 i 根据下式计算根据下式计算 $x_{(i \rightarrow j)}^{(m+1)} = 0$:

$$x_{(i \rightarrow j)}^{(m+1)} = \begin{cases} \frac{q_{rs} \circ w_{(i \rightarrow j)}}{\sum_k w_{(k \rightarrow j)}} & j = s \\ \sum_l x_{(j \rightarrow l)}^{(m)} \frac{w_{(i \rightarrow j)}}{\sum_k w_{(k \rightarrow j)}} & j \neq s \end{cases} \quad (13)$$

步骤 3 判断 $x_{(i \rightarrow j)}^{(m+1)}$ 是否满足条件

$$\sqrt{\frac{\sum (x_{(i \rightarrow j)}^{(m+1)} - x_{(i \rightarrow j)}^{(m)})^2}{\sum (x_{(i \rightarrow j)}^{(m)})^2}} < \epsilon_2 \quad (14)$$

如果满足, 结束; 否则判断 m 是否大于 M_2 , 如果 m

> M_2 , 计算停止; 否则设 $m = m + 1$, 转向步骤 3。

3 计算实例

本节给出 2 个计算实例, 第 1 个为不包含回路的网路, 另 1 个为包含回路的网路。第 1 个例子的交通网络如图 1 (a) 所示, 图中表明每条路段的阻抗。假设起点为 1, 终点为 9, 交通需求为 1 000 单位, 并假设 Logit 参数 $\theta = 1.0$ 。计算结果如图 1 (b) 所示, 括号中为 Dial 算法的流量^[6]。由于没有回路的存在, 新算法得到和枚举法完全相同的结果; 而 Dial 算法没有能够给出正确的结果。这表明新算法计算无环网络时可以得到和理论值一样的结果, 这和前面的分析是一致, 也是新算法比 Dial 算法的优越之处。

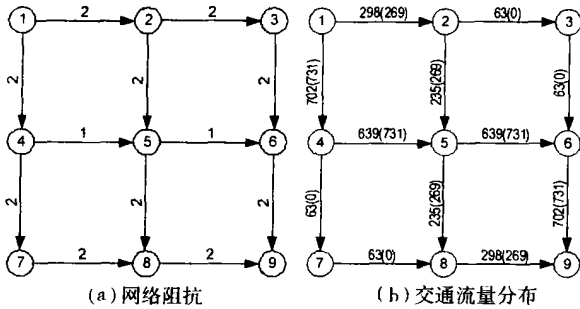


图 1 无环网络

Fig. 1 Network without loop

第 2 个实例在第 1 个网络上修改而成, 将联接节点之间单向路段变成双向的, 修改后的网络和计算结果如图 2 所示, 括号中的数字为枚举法所计算的不含回路的结果。由于回路的存在使得连接节点 4、5、6 的路段流量偏大, 表明出行者会在这些路段上绕圈。这虽然和实际有所出入, 与全路径模型的假设却是一致的。通过对 θ 的不同值的计算表明, 在误差设为 1% 时, 迭代次数约为 10 次。由于回路的存在, 当 θ 变小时, 收敛速度会变慢, 而当 θ 小于特定值时, 算法会发散, 这和 Song 的论述是一致的^[7]。

4 结 论

本文提出的迭代算法可避免 Dial 算法所带来

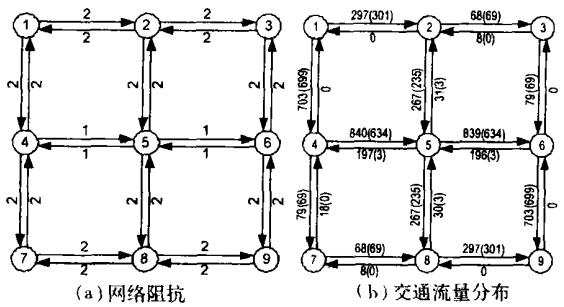


图 2 有环网络

Fig. 2 Network with loops

的路径选择中的不合理现象, 同时不需要进行路径的枚举和高阶矩阵的求逆, 计算量较小。在交通网络不存在回路的情况下, 新算法可以得到和理论值完全相同的结果; 对有环网络, 在一般的 Logit 参数范围内, 也可以得到合理的结果。全路径模型假定出行者会选择无穷环路, 和现实世界不完全相符。将来的研究课题包括如何排除环路的影响, 以及将该算法扩展到计算可变成本的 Logit 分配模型。

参考文献:

[1] DIAL R B. A probabilistic multipath traffic assignment model which obviates path enumeration [J] . Transportation Research, 1971, 5: 83—111.
[2] LI J, KAWAKAMI S. Which path is reasonable: A paradox in Dial's algorithm[A] . Proceedings of JSCE 3th International Summer Conference, 2001.
[3] BELL G. H. Alternatives to Dial's LOGIT assignment algorithm [J] . Transportation Research, 1995, 29B: 287—296.
[4] AKAMATSU T. Cyclic flows, Markov process and stochastic traffic assignment [J] . Transportation Research, 1996, 30B: 369—386.
[5] AKAMATSU T. Decomposition of path choice entropy in general transport networks [J] . Transportation Science, 1997, 31: 349—362.
[6] SHEFFI Y. Urban transportation network [M] . Massachusetts Institute of Technology, 1984.
[7] WONG S C. On the convergence of Bell's logit assignment formulation [J] , Transportation Research, 1999, 33B: 609—616.

Solving All path Logit Assignment Problem by Iteration Method

LI Jun, NIE Pei lin, YU Zhi

(School of Engineering, Sun Yat-sen University, Guangzhou 510275, China)

Abstract: The deficiency of existing algorithms for logit assignment model is discussed , and a new iteration method is proposed for all path logit assignment problem. The equivalence between new algorithm and logit assignment model is proved. Two numerical examples are presented, and comparison between new algorithm and Dial's algorithm is given.

Key words: Logit traffic assignment; Dial's algorithm, iteration method