第 7 期
2009 年 7 月
电 子 学 报
ACTA ELECTRONICA SINICA
Vol. 37 No. 7
July 2009

基于“矩阵乘法”的网络最短路径算法

邓方安，雍龙泉，周 涛，刘丽华
(陕西理工学院数学系，陕西汉中 723001)

摘 要： 网络最短路径问题可以作为许多实际应用问题的模型，但传统的求解算法其迭代过程复杂. 本文描述了基于矩阵乘法的最短路算法，其时间复杂度与 Dijkstra 算法相同. 在给定的一个网络图中，在不改变网络图中的最短路的条件下，删除“多余”的结点或边，可以达到简化网络图和提高求解速度的目的，从而降低计算复杂性. 最后，研究了该方法在最短路径问题和旅行商问题中的应用. 实例表明，这种算法与传统的动态规划技术相比，具有运算简便、易于理解的优点.

关键词： 矩阵乘法；最短路问题；约简原则；旅行商问题
中图分类号： TP301.6 **文献标识码**： A **文章编号**： 0372-2112 (2009) 07-1594-05

Shortest Path Problem Algorithm in Network Based on Matrix Multiplication

DENG Fang-an，YONG Long-quan，ZHOU Tao，LIU Li-hua
(Department of Mathematics, Shaanxi University of Technology, Hanzhong, Shaanxi 723001, China)

Abstract： Shortest Path Problem in network can be acted as a model for many application problems, but the iteration process of the conventional solution algeorithm is complex. In this paper, the shortest path algorithm based on the matrix multiplication in a weighted-graph are described, and its time complexity is as same as Dijkstra algorithm. In a given network graph, this algorithm delete spare nodes or edges and reach the goal that reduced network graph and improved speed of solving problem under the condition of unchanged shortest parth. Finally, we give examples, e. g. Traveling Salesman Problem, shortest path problem, to illistrate its advanteges, possessing merits of simple operation and easy understanding, compared with dynamic programming technology.

Key words： matrix multiplication; shortest path problem; reduced principle; traveling salesman problems

1 引言

日常生活和生产中许多问题都可以用一个网络来描述. 例如，交通网络，计算机网络，工程进度网络，生物信息网络和互联网等[1～4]. 对于诸如最短路径问题、最小费用问题，最大流问题等经典网络优化问题的解决，除了比较成熟的动态规划[5]技术外，一些新的进化算法，包括蚁群算法、人工神经网络和遗传算法被提出[6,7]. 传统的 Bellman 动态优化算法可以求解最短路问题，但这种算法需要很大的空间，最好和最坏的情况下的时间差不多相同. 虽然它在规模比较小的时候，可以很好的解决问题，但是，随着问题规模的扩大，Bellman 动态优化算法就显得无能为力. 文献[8]扩展了线性代数中的矩阵运算，定义了矩阵和与积的概念及运算，通过这种方法，可以把求最短路转化为矩阵的运算，计算简便、有效. 文献[10]用实例证明了著名的 Dijkstra 算法在时间依赖的网络上不能有效地求解最短路径问题，给出了时间依赖的网络的定义和模型，给出一种实用反馈式神经网络来求解时间依赖的网络的最短路径问题. 并用模拟实验验证了它在不同的网络更新时间区间上收敛速度的稳定性. 结果是神经网络求解非 NP-难解类优化问题的一种新尝试. 本文介绍了基于矩阵乘法运算的最短路算法，并考虑了该算法的时间复杂度，研究了它的应用.

2 算法的描述

动态规划是求解许多重要应用问题的关键技术，可以高效地解决许多用贪心算法或分支定界方法无法解决的问题. 但该方法比较抽象，难于理解. 下面利用“矩阵乘积”运算求解最短路问题. 不难看出：新方法比动态规划方法运算简便，容易理解.

例 1 图 1 表示从起点 A 到终点 E 之间各点的距离，求 A 到 E 的最短路径.

用动态规划方法不难求得从起点 A 到终点 E 的距

收稿日期：2008-03-28；修回日期：2009-03-22
基金项目：国家自然科学基金(No. 60875034)

离为 19,从 A 到 E 的最短路径为 A→B_2→C_1→D_1→E.

下面举例来说明“矩阵乘积”运算求解最短路问题的方法. 我们把求 A→E 的最短路分解为四个阶段 A→B→C→D→E 来求解. 每一个阶段可以用一个矩阵来表示,我们称之为权矩阵;而相邻阶段的最短路用权矩阵乘积运算来计算. 如 A→C 可以看成 A→B 的权矩阵和 B→C 的权矩阵的乘积,这里的矩阵乘法和普通矩阵乘积运算的区别是:普通矩阵乘积其对应元素是相应元素乘积的代数和,这里把元素相乘改为相加,元素的代数和改为取小运算,如果不同层节点间没有连接,则视它们之间的距离为无穷大. 如果是求极大,改为取大运算,此时如果不同层节点间没有连接,则视它们之间的距离为零.

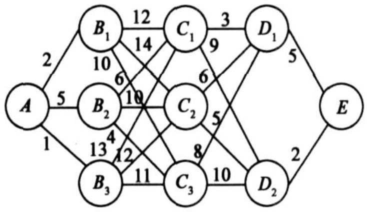

图1

以图 1 为例来解释这个算法.

A→B 的权矩阵为:

$$\begin{array}{cccc} & B_1 & B_2 & B_3 \\ A & (2 & 5 & 1) \end{array}$$

B→C 的权矩阵为:

$$\begin{array}{c} \\ B_1 \\ B_2 \\ B_3 \end{array}\begin{array}{c} \begin{array}{ccc} C_1 & C_2 & C_3 \end{array} \\ \begin{pmatrix} 12 & 14 & 10 \\ 6 & 10 & 4 \\ 13 & 12 & 11 \end{pmatrix} \end{array}$$

则 A→C 的权矩阵为:

$$(2,5,1)\begin{pmatrix} 12 & 14 & 10 \\ 6 & 10 & 4 \\ 13 & 12 & 11 \end{pmatrix} = (11,13,9)$$

A→D 的权矩阵为:

$$(11,13,9)\begin{pmatrix} 3 & 9 \\ 6 & 5 \\ 8 & 10 \end{pmatrix} = (14 \wedge 19 \wedge 17, 20 \wedge 18 \wedge 19) = (14,18)$$

最后得到 A→E 的最短路:$(14,18)\begin{pmatrix} 5 \\ 2 \end{pmatrix} = 19$. 因此,A→E 的最短路为 19,其最短路径为 A→B2→C1→D1→E.

下面我们给出基于“矩阵乘法”求解最短路的算法:

第一阶段:计算出图中从起始点到终点最短路的长度.

step1　划分出该网络图中的层次关系(网络划分为 N 层,起点为第一层,终点为第 N 层);

step2　依次给出从第 i 层到第 $i+1$ 层的权矩阵($i=1,2,\cdots,N-1$);(若第 i 层有 m 个顶点;第 $i+1$ 层有 n 个顶点,则从第 i 层到第 $i+1$ 层的权矩阵为 $m*n$ 阶).

step3　按照我们定义的矩阵乘法计算出最短路的数值.

第二阶段:寻找最短路所经过的中间点.

(利用第一阶段中 step2 的数据)计算出从第 i 层到终点的最短路,对比与 i-1 层到终点的最短路,从而确定出第 i 层上最短路所经过的顶点($i=2,\cdots,N-1$).

注:该算法的复杂度与 Dijsktra 算法的复杂度相同,都为 $O(n^2)$,具有运算简便、易于理解的优点.

3　算法的应用

下面我们给出该算法在网络最短路问题和旅行商问题中的应用.

例 2　如图 2 所表示的网络,连接各点的线段上的数字表示它们之间的弧长. 我们从 A 出发要走到目的地 G,问经过哪些点,走什么路线使得总路程最短.

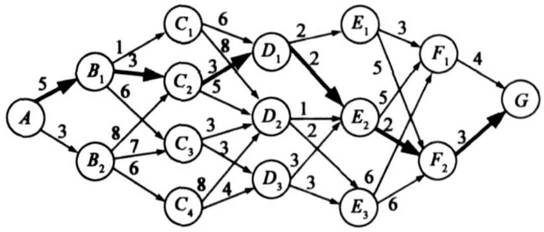

图2

文献[9]给出用动态规划求解的方法,这里用前面介绍的“矩阵乘法”来求解. 由于

$$(5,3)\begin{pmatrix} 1 & 3 & 6 & \infty \\ \infty & 8 & 7 & 6 \end{pmatrix}\begin{pmatrix} 6 & 8 & \infty \\ 3 & 5 & \infty \\ \infty & 3 & 3 \\ \infty & 8 & 4 \end{pmatrix}\begin{pmatrix} 2 & 2 & \infty \\ \infty & 1 & 2 \\ \infty & 3 & 3 \end{pmatrix}\begin{pmatrix} 3 & 5 \\ 5 & 2 \\ 6 & 6 \end{pmatrix}\begin{pmatrix} 4 \\ 3 \end{pmatrix} = 18.$$

因此该网络最短路长度为 18,且其最短路径为:

A→B1→C2→D1→E2→F2→G.

注意:在用矩阵乘积确定最短路的过程中,如有多条路径的长度都取相同值(最小值),则这个网络存在多条最短路.

例 3　旅行商问题(TSP)是计算机算法中的一个经

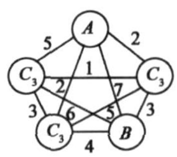
图3

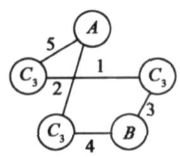
图4

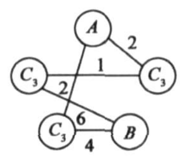
图5

典的难解问题,已归为 NP 一完备问题类. 围绕这个问题有各种不同的求解方法,已有的算法如动态规划法、分支限界法、回溯法等,这些精确式方法都是指数级 $O(2^n)$ 的,根本无法解决目前的实际问题,贪心法是近似方法,而启发式算法不能保证得到的解是最优解,甚至是较好的解释.

旅行商问题一般表述为:已知 n 个城市之间的相互距离,现有一个推销员必须遍访这 n 个城市,并且每个城市只能访问一次,最后又必须返回出发城市. 如何安排他对这些城市的访问次序,可使其旅行路线的总长度最短(这里以 $n=5$ 为例) ?

(1) 若旅行商由 A 出发,先到 B 城市,采用矩阵算法求解如下:

$$(2)\ (3,5,1)\begin{pmatrix} 4 & 6 & \infty & \infty & \infty & \infty \\ \infty & \infty & 4 & 3 & \infty & \infty \\ \infty & \infty & \infty & \infty & 6 & 3 \end{pmatrix}$$

$$\cdot\begin{pmatrix} 3 & \infty & \infty & \infty & \infty & \infty \\ \infty & 3 & \infty & \infty & \infty & \infty \\ \infty & \infty & 6 & \infty & \infty & \infty \\ \infty & \infty & \infty & 6 & \infty & \infty \\ \infty & \infty & \infty & \infty & 4 & \infty \\ \infty & \infty & \infty & \infty & \infty & 4 \end{pmatrix}\begin{pmatrix} 5 \\ 2 \\ 5 \\ 7 \\ 2 \\ 7 \end{pmatrix}=15$$

最短路径为:A→B→E→C→D→A.

(2) 若旅行商由 A 出发,先到 C 城市,采用矩阵算法求解如下:

$$(7)\ (3,4,6)\begin{pmatrix} 5 & 1 & \infty & \infty & \infty & \infty \\ \infty & \infty & 5 & 3 & \infty & \infty \\ \infty & \infty & \infty & \infty & 1 & 3 \end{pmatrix}$$

$$\cdot\begin{pmatrix} 3 & \infty & \infty & \infty & \infty & \infty \\ \infty & 3 & \infty & \infty & \infty & \infty \\ \infty & \infty & 1 & \infty & \infty & \infty \\ \infty & \infty & \infty & 1 & \infty & \infty \\ \infty & \infty & \infty & \infty & 5 & \infty \\ \infty & \infty & \infty & \infty & \infty & 5 \end{pmatrix}\begin{pmatrix} 5 \\ 2 \\ 5 \\ 2 \\ 2 \\ 2 \end{pmatrix}=16$$

最短路径为:A→C→B→E→D→A. 但该路径长度大于(1) 中所求最短路径,因此从全局来看,旅行商从 A 出发先到 C 城市这一走法无全局最短路径.

(3) 若旅行商由 A 出发,先到 D 城市,采用矩阵算法求解如下:

$$(2)\ (5,4,3)\begin{pmatrix} 3 & 1 & \infty & \infty & \infty & \infty \\ \infty & \infty & 3 & 6 & \infty & \infty \\ \infty & \infty & \infty & \infty & 1 & 6 \end{pmatrix}$$

$$\cdot\begin{pmatrix} 6 & \infty & \infty & \infty & \infty & \infty \\ \infty & 6 & \infty & \infty & \infty & \infty \\ \infty & \infty & 1 & \infty & \infty & \infty \\ \infty & \infty & \infty & 1 & \infty & \infty \\ \infty & \infty & \infty & \infty & 3 & \infty \\ \infty & \infty & \infty & \infty & \infty & 3 \end{pmatrix}\begin{pmatrix} 5 \\ 7 \\ 5 \\ 2 \\ 7 \\ 2 \end{pmatrix}=15$$

从上述计算中可以发现有 2 条路径均最短,分别为:A→D→C→B→E→A 和 A→D→C→E→B→A.

(4) 若旅行商由 A 出发,先到 E 城市,采用矩阵算法求解如下:

$$(5)\ (1,6,3)\begin{pmatrix} 3 & 5 & \infty & \infty & \infty & \infty \\ \infty & \infty & 3 & 4 & \infty & \infty \\ \infty & \infty & \infty & \infty & 5 & 4 \end{pmatrix}$$

$$\cdot\begin{pmatrix} 4 & \infty & \infty & \infty & \infty & \infty \\ \infty & 4 & \infty & \infty & \infty & \infty \\ \infty & \infty & 5 & \infty & \infty & \infty \\ \infty & \infty & \infty & 5 & \infty & \infty \\ \infty & \infty & \infty & \infty & 3 & \infty \\ \infty & \infty & \infty & \infty & \infty & 3 \end{pmatrix}\begin{pmatrix} 2 \\ 7 \\ 2 \\ 2 \\ 7 \\ 2 \end{pmatrix}=15$$

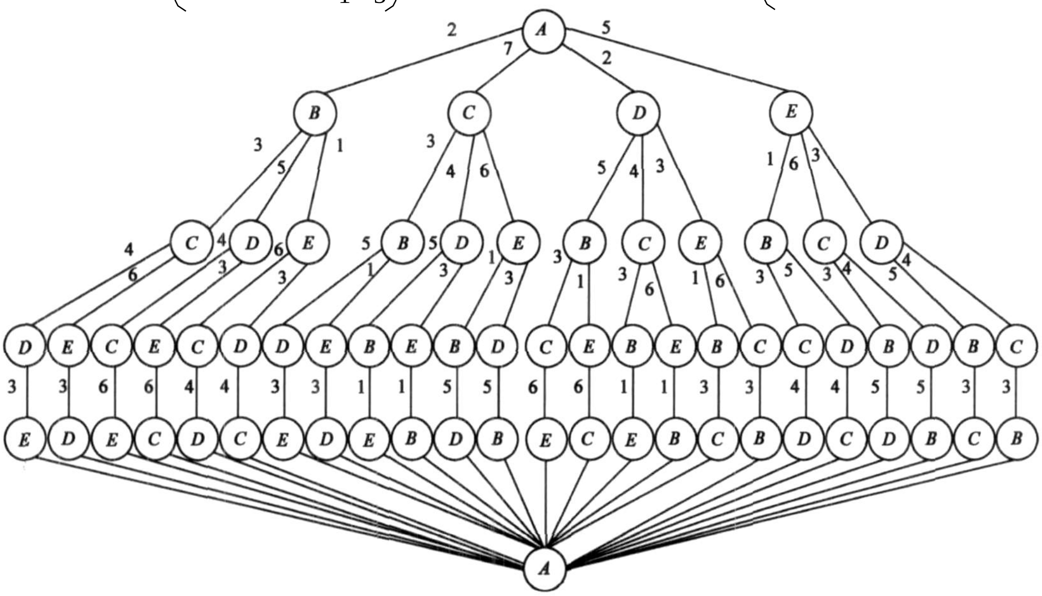

图6

最短路径为:A→E→B→C→D→A.

4 网络图的简化

在求解网络图的最短路问题中,适当地删除结点或边,可以简化图,这样可以简化计算复杂度,提高求解速度.简化的基本原则是不改变原来网络图中的最短路.

在文献[7]中,用遗传算法求得图 8 中从 S 点到 D 点的最短路径.这里我们给出一种新算法.即在求最短路的过程中,可以把某一层中不可能用到的结点或边删除.

删除结点的原则如下:

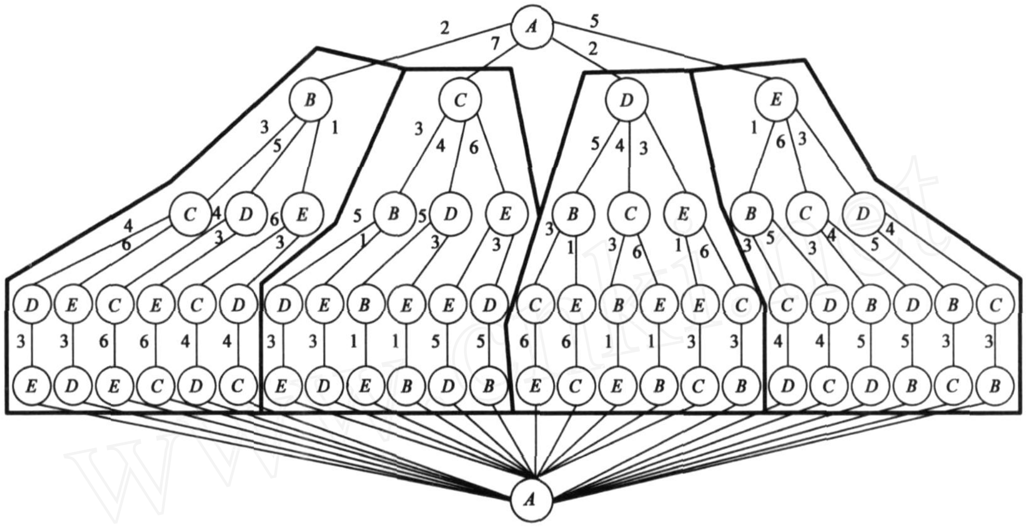

图7

(1) 从始点出发不过该结点,通过其他结点完全可以到达下一层的所有各个结点;

(2) 从始点出发过该结点到下一层的路径最长.

删除同层结点间的边的原则如下:

从上一层结点到下一层的结点不必通过某两结点之间的边也可以到达,且通过该边的路程是较长的,则这样的边可以删除.

如图 8 中,结点 5 可以删除,因为通过 5 到达下一层的路径明显多于其他路径.同时结点 3,4 之间的边以及 2,3 之间的边都可以删除.删除以后的网络图如下:

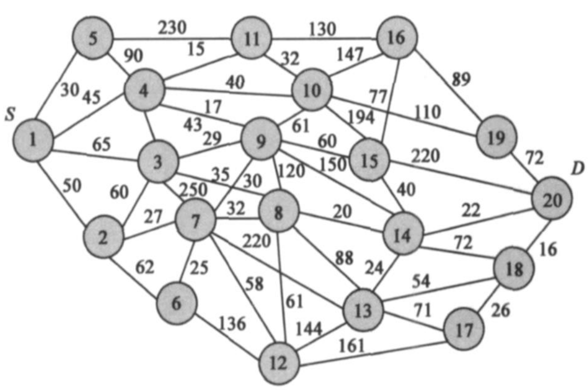

图8

下面用矩阵乘法求关于图 9 的最短路.在图 9 中,

$$(45,65,50)\begin{pmatrix}15 & 40 & 17 & \infty & \infty & \infty\\ \infty & \infty & 29 & 35 & 250 & \infty\\ \infty & \infty & \infty & \infty & 27 & 62\end{pmatrix}$$

$$\cdot\begin{pmatrix}130 & \infty & \infty & \infty & \infty\\ 147 & 194 & \infty & \infty & \infty\\ \infty & 60 & 150 & \infty & \infty\\ \infty & \infty & 20 & 88 & 61\\ \infty & \infty & \infty & 220 & 58\\ \infty & \infty & \infty & \infty & 136\end{pmatrix}\begin{pmatrix}89 & \infty & \infty\\ 14 & 220 & \infty\\ \infty & 72 & \infty\\ \infty & 54 & 71\\ \infty & \infty & 161\end{pmatrix}\begin{pmatrix}72\\ 16\\ \infty\end{pmatrix}$$

$$=(60,85,62,100,77,112)$$

$$\cdot\begin{pmatrix}130 & \infty & \infty & \infty & \infty\\ 147 & 194 & \infty & \infty & \infty\\ \infty & 60 & 150 & \infty & \infty\\ \infty & \infty & 20 & 88 & 61\\ \infty & \infty & \infty & 220 & 58\\ \infty & \infty & \infty & \infty & 136\end{pmatrix}\begin{pmatrix}89 & \infty & \infty\\ 14 & 220 & \infty\\ \infty & 72 & \infty\\ \infty & 54 & 71\\ \infty & \infty & 161\end{pmatrix}\begin{pmatrix}72\\ 16\\ \infty\end{pmatrix}$$

$$=(190,122,120,188,135)\begin{pmatrix}89 & \infty & \infty\\ 14 & 220 & \infty\\ \infty & 72 & \infty\\ \infty & 54 & 71\\ \infty & \infty & 161\end{pmatrix}\begin{pmatrix}72\\ 16\\ \infty\end{pmatrix}$$

$$=208\wedge 208\wedge\infty=208.$$

从而有两条路径的长度为 208,但由于图中存在跨层的结点连接,还需要计算跨层的结点间的路径长度,因此它们不一定是最短路.不难看出另有一条从结点 14 到 20 的边,长度 22,故又可以得到另一条路的长度计算如下:

$$(45,65,50)\begin{pmatrix}15 & 40 & 17 & \infty & \infty & \infty\\ \infty & \infty & 29 & 35 & 250 & \infty\\ \infty & \infty & \infty & \infty & 27 & 62\end{pmatrix}\cdot\begin{pmatrix}130 & \infty & \infty & \infty & \infty\\ 147 & 194 & \infty & \infty & \infty\\ \infty & 60 & 150 & \infty & \infty\\ \infty & \infty & 20 & 88 & 61\\ \infty & \infty & \infty & 220 & 58\\ \infty & \infty & \infty & \infty & 136\end{pmatrix}(22)$$

$$=(60,85,62,100,77,112)\begin{pmatrix}130 & \infty & \infty & \infty & \infty\\ 147 & 194 & \infty & \infty & \infty\\ \infty & 60 & 150 & \infty & \infty\\ \infty & \infty & 20 & 88 & 61\\ \infty & \infty & \infty & 220 & 58\\ \infty & \infty & \infty & \infty & 136\end{pmatrix}(22)$$

$=(190,122,120,188,135)(22)=212\wedge 144\wedge 142\wedge 210\wedge 157=142$. 其路径是:$1\to 3\to 8\to 14\to 20$.

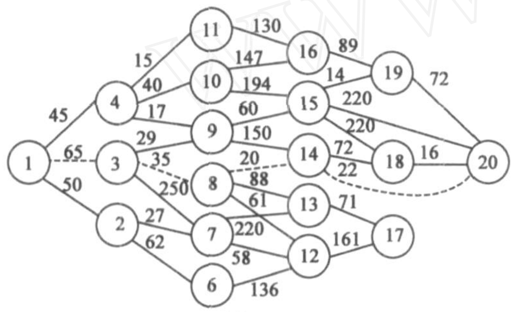

图9

5 结束语

最短路径问题一直是计算机科学、运筹学、地理信息科学等学科的一个研究热点,经典的图论与不断完善的计算机数据结构及算法的有效结合使得新的最短路径算法不断涌现.近年来,随着对组合优化问题的深入研究,很多进化算法,包括蚁群算法、人工神经网络和遗传算法被用于求解有关最短路径问题.本文基于“矩阵乘法”运算,提出了一个求解网络中最短路径的算法,该算法与 Dijsktra 算法具有相同的时间复杂度,并把该算法应用于求解最短路径问题和旅行商问题,实例表明该方法具有运算简便、易于理解的优点,既可以作为教学内容讲授给学生,也可以提供给科研工作者直接使用.

参考文献:

[1] 陈洁,陆锋.交通网络最短路径标号算法的实现与效率分析[J].中国图象图形学报,2005,10(9):1134 - 1138.

[2] Azaron,A..Bicriteria shortest path in networks of queues[J].Applied Mathematics and Computation ,2006,182 (1):434 - 442.

[3] BRAESS Dietrich, NAGURNEY Anna, WAKOLBINGER Tina. On a paradox of traffic planning[J]. Transportation science,2005,39(4):446 - 450.

[4] Dominique Feillet, Pierre Dejax, Michel Gendreau. Traveling Salesman Problems with Profits[J]. Transportation Science,2005,39(2):188 - 205.

[5] Wayne L. Winston. Operations Research[M]. Duxbury Pr, 1994.

[6] 刘建强,许雯,刘粉林等.一种最短路问题的遗传算法求解[J].数学实践与认识,2007,37(17):53 - 58.

[7] 赵建宏,杨建宇,雷维礼.一种新的最短路径算法[J].电子科技大学学报,2005,34(6):778 - 781.

[8] 詹棠森,张三强,唐敏.用矩阵和积求最短路的一种新算法[J].数学的实践与认识,2006,136(19):170 - 172.

[9] 王晟,李乐民.一种改进的多约束最佳路径算法研究[J].电子学报,2004,32(4):529 - 535.
WANG Sheng,LI Le-min. An enhanced algorithm for multiple constraints optimal path calculation[J]. Acta Electronica Sinica,2004,32(4):529 - 535. (in Chinese)

[10] He Hong ,Zhu Daming ,Ma Shao-han. A New Algorithm for the Shortest Paths Computationby Neural Networks on Time dependent Networks[J]. Fudan University (Natural Science), 2004,43(5):714 - 716.

作者简介:


邓方安 男,1963 年生于陕西宁强县,教授,博士,主要研究方向为粗糙集理论及应用、粒计算等.
E-mail:dengfangans@126.com


雍龙泉 男,1980 年出生于陕西洋县,讲师,硕士,主要研究方向为优化理论与算法设计及其应用等.E-mail:yonglongquan@163.com


周 涛 男,1977 年出生于宁夏同心县,回族,副教授,在读博士,主要研究方向为模式识别、数据挖掘、软计算理论等.
E-mail:zhout123@gmail.com