文章编号: 1001-0920 (2016) 04-0623-06　　DOI: 10.13195/j.kzyjc.2015.0116

一类标准矩形网络节点间最短路径的求解方法

刘宏志[1,2], 高立群[1], 欧阳海滨[1]

(1. 东北大学 信息科学与工程学院，沈阳 110004；2. 辽宁工程技术大学 电气与控制工程学院，辽宁 葫芦岛 125105)

摘　要: 针对常见的交通道路最短路径问题, 提出标准矩形网络的概念, 分析其节点间最短路径的性质, 并在此基础上给出一种新颖的最短路径求解算法. 该算法利用标准矩形网络的几何性质, 简化了搜索方向和步长的判断, 同时指出常见的交通道路网络一般均可以整体或部分化为标准矩形网络. 与常见的求取最短路径的Dijkstra、Floyd、ACO、A* 等算法进行仿真实验比较, 实验结果表明, 对于大规模标准矩形道路网络, 所提出算法具有更好的寻优精度、稳定性和寻优速度.

关键词: 最短路径；标准矩形网络；交通道路；搜索方向

中图分类号: TP391　　**文献标志码:** A

A kind of solving method for the shortest path between standard rectangular network nodes

LIU Hong-zhi[1,2], *GAO Li-qun*[1], *OUYANG Hai-bin*[1]

(1. College of Information Science and Engineering，Northeastern University，Shenyang 110004，China；2. Faculty of Electrical and Control Engineering，Liaoning Technical University，Huludao 125105，China．Correspondent：LIU Hong-zhi，E-mail：19048851@qq.com)

Abstract: In view of the shortest path problem of the common traffic road, the standard rectangular network concept is proposed, the properties of the shortest path between nodes are analyzed, and a novel shortest path algorithm based on the standard rectangular network(SRNSP) is presented. By using the geometric properties of the standard rectangular network, the search direction and step judgment are simplified. Meanwhile, it is pointed out that some or even all of the common traffic road networks can be converted into standard rectangular networks. Compared with the common Dijkstra, Floyd, ACO and A* algorithms for solving the shortest path problem, the experiment results show that the proposed algorithm has better optimization accuracy, stability and searching speed for the large-scale standard rectangular road network.

Keywords: shortest path；standard rectangular network；traffic road；search direction

0 引　　言

最短路径分析是资源分配、路线设计和分析等优化问题的基础, 在生产实际、国防建设、航空航天、交通物流等领域中, 许多优化问题都可以转化为最短路径问题. 最短路径不仅意味着空间距离的长短, 还可以引申到其他度量, 如时间、金钱、资源等. 最短路径起源于图论, 旨在寻找图中两点之间的最短路径, 或距离最短, 或费时最少, 或花费最小. 在计算机图形学的基础上, 交通道路网络中最短路径问题得到了大量研究和发展[1-6]. 1959 年, Dijkstra[7]首先提出了最短路径问题的经典Dijkstra算法, 此后延伸出许多新的最短路径算法, 如Floyd算法[8]、A* 算法[9], 接着蚁群算法[10]、粒子群[11]、遗传算法[12]等智能优化算法应运而生. Dijkstra算法是一种盲目搜索算法, 搜索效率不高, 不便于解决某些复杂问题. Floyd算法可以计算出任意两个节点之间的最短距离, 代码编写简单, 但时间复杂度较高, 不适合计算大量数据. A* 算法是常见的启发式搜索算法之一, 具有较快的搜索速度, 但易陷入局部最优. 虽然蚁群、粒子群、遗传等智能优化算法在最短路径的求解中获得了某些成功, 但计算量较大, 限制了其在大规模网络中的应用.

本文首先针对一类特殊网络给出标准矩形网络的定义, 根据标准矩形网络的特点, 提出两节点间最短路径的求解方法, 称为SRNSP算法; 然后, 进行算

收稿日期: 2015-01-26；**修回日期:** 2015-07-25.

基金项目: 国家自然科学基金项目(61403174).

作者简介: 刘宏志(1977−), 男, 讲师, 博士生, 从事数学建模与智能优化理论的研究；高立群(1949−), 男, 教授, 博士生导师, 从事模式识别、智能优化理论等研究.

法分析, 并指出许多非标准矩形网络可以转化为标准矩形网络; 最后, 进行了求取交通道路网络最短路径的仿真对比实验, 仿真结果验证了所提出方法的有效性和优越性.

1 一类标准矩形网络

从理论上讲, 利用穷举法可以准确得到有限节点网络中任意两个节点间的最短路径, 但是当节点数量众多时, 计算量会变得非常巨大, 导致穷举法失效. 在这种情况下, 人们只能借助于智能优化方法寻找最短路径, 由于网络结构的复杂性, 对于具有大量节点的一般网络, 即使运用智能优化方法也相当困难, 不过对于某些具有特殊结构的网络问题会变得简单. 下面定义一种特殊的网络, 并对其最短路径问题加以研究.

定义1 考虑具有 $m\times n$ 个节点的矩形网络 $W(m,n)$, 节点 $x_{i,j}$ 与 $x_{i+k,j}$ 和 $x_{i,j+s}$ 间的边长分别为 $d(x_{i,j},x_{i+k,j})$ 和 $d(x_{i,j},x_{i,j+s})$, $1\leqslant i<m$, $i+k\leqslant m$, $1\leqslant j<n$, $j+s\leqslant n$, 若满足

$$\begin{aligned}&d(x_{i,j},x_{i+k,j})<\\&d(x_{i,j},x_{i,j+s})+d(x_{i,j+s},x_{i+k,j+s})+\\&d(x_{i+k,j+s},x_{i+k,j}),\end{aligned}\tag{1}$$

$$\begin{aligned}&d(x_{i,j},x_{i,j+s})<\\&d(x_{i,j},x_{i+k,j})+d(x_{i+k,j},x_{i+k,j+s})+\\&d(x_{i+k,j+s},x_{i,j+s}),\end{aligned}\tag{2}$$

则称该网络为标准矩形网络.

具有 $m\times n$ 个节点的标准矩形网络有4个角点, 每个角点与2条边(路径)相连; 具有 $2(m+n-4)$ 个边点, 每个边点与3条边(路径)相连; 其余 $(m-2)\times(n-2)$ 个节点为内点, 每个内点均与4条边(路径)相连. 式(1)和(2)对于构成网络的每个矩形作了进一步限制, 规定其必须满足3边之和大于第4边. 正是由于这种限制, 使得其最短路径的求取变得容易.

定理1 在标准矩形网络 $W(m,n)$ 中, 从节点 $x_{i,j}$ 出发到节点 $x_{k,s}(k>i,s>j)$ 的最短路径由 $(k-i+s-j+1)$ 个节点间的 $(k-i+s-j)$ 条小矩形边组成.

证明 用数学归纳法证明. 不失一般性, 设 $n\geqslant m$, $i=j=1$. 当 $k=i+1$, $s=j+1$, 即 $k+s\in\{4\}$ 时, 定理成立.

下面假设当 $k+s\in\{4,5,\cdots,N\}$ 时定理成立, 证明 $k+s\in\{4,5,\cdots,N,N+1\}$ 时, 定理仍然成立. 记 $d^*(x_{i,j},x_{k,s})$ 为点 $x_{i,j}$ 到 $x_{k,s}$ 的最短路径, 当 $k+s=N+1$ 时, 由动态规划原理可知, 点 $x_{1,1}$ 到 $x_{k,N+1-k}$ 的最短路径长度属于集合

$$\begin{aligned}&\omega_1=d(x_{1,1},x_{k,N+1-k})=\\&d^*(x_{1,1},x_{k,t})+d(x_{k,t},x_{k,N-k+1}),\\&t=1,2,\cdots,N+1-k,\ t\leqslant n,\ k\leqslant m;\end{aligned}$$

或

$$\begin{aligned}&\omega_2=d(x_{1,1},x_{N+1-s,s})=\\&d^*(x_{1,1},x_{t,s})+d(x_{t,s},x_{N+1-s,s}),\\&t=1,2,\cdots,N+1-s,\ t\leqslant m,\ s\leqslant n.\end{aligned}$$

当最短路径长度属于 ω_1 时, 根据假设可知 $d^*(x_{1,1},x_{k,t})$ 由 $k+t-2$ 条小矩形边组成, 注意到对于任意的 $t=1,2,\cdots,N+1-k$, $t\leqslant n$, $1\leqslant p\leqslant m-k$, 有

$$\begin{aligned}&d(x_{k,t},x_{k,s})\leqslant\\&d(x_{k,t},x_{k+p,t})+d(x_{k+p,t},x_{k+p,s})+d(x_{k+p,s},x_{k,s}).\end{aligned}$$

可知

$$\begin{aligned}&d^*(x_{k,t},x_{k,N-k+1})=\\&d(x_{k,t},x_{k,t+1})+d(x_{k,t+1},x_{k,t+2})+\cdots+\\&d(x_{k,N-k},x_{k,N-k+1}),\end{aligned}$$

其中 $d^*(x_{k,t},x_{k,N-k+1})$ 由 $N-k+1-t$ 条小矩形边组成. 进而可知路径 $d(x_{1,1},x_{k,N+1-k})$ 由 $w=k+t-2+N-k+1-t=N-1=(k-1)+((N+1-k)-1)$ 条相互连接的小矩形边组成.

同理可证, 当最短路径长度属于 ω_2 时, 定理仍然成立. □

定理2 在标准矩形网络 $W(m,n)$ 中, 从节点 $x_{i,j}$ 出发寻找到 $x_{k,s}(k>i,s>j)$ 的最短路径, 途经节点 $x_{p,q}$ 时, 如果 $p<k$, $q<s$, 则只有两种选择, 向节点 $x_{p,q+1}$ 前进, 或向节点 $x_{p+1,q}$ 前进; 如果 $p=k(q=s)$, 则只有一种选择, 向节点 $x_{p,q+1}$ (或 $x_{p+1,q}$) 前进.

证明 对于最短路径中所经过的节点 $x_{p,q}$, 应有

$$d^*(x_{i,j},x_{k,s})=d^*(x_{i,j},x_{p,q})+d^*(x_{p,q},x_{k,s}).$$

当 $p<k$, $q<s$ 时, 在节点 $x_{p,q}$ 最多有4种选择, 分别为向节点 $x_{p,q+1}$、$x_{p+1,q}$、$x_{p-1,q}$ 和 $x_{p,q-1}$ 前进. 如果向节点 $x_{p-1,q}$ 行进, 则到达终点的路径至少为 $(p-i)+(q-j)+1+(k-(p-1)+(s-q))=k-i+s-j+2$, 由定理1可知, 此路径并非最短路径. 同理可知, 如果向节点 $x_{p,q-1}$ 行进, 则到达终点的路径也不是最短路径. 因此在这种情况下, 只有向节点 $x_{p,q+1}$ 或者向节点 $x_{p+1,q}$ 前进两种选择.

当 $p=k$ 时, 如果向节点 $x_{p+1,q}$ (或 $x_{p,s+1}$) 前进, 则到达终点的路径至少为 $(k-i)+(q-j)+1+(k+(1-k)+(s-q))=k-i+s-j+2$, 由定理1可知, 此路径并非最短路径.

同理可知, 当 $q=s$ 时, 经节点 $x_{p,s+1}$ 到达终点 $x_{k,s}$ 的路径也不是最短路径. □

由于标准矩形网络中两节点间最短路径具有上

述性质, 求解最短路径问题变得非常简单. 当 m 和 n 不大时, 可以利用穷举法求得精确的最短路径; 当 m 和 n 稍大时, 可利用 Dijkstra、Floyd、A* 等常规算法; 当 m 和 n 很大时, 需要借助智能优化算法.

2　标准矩形网络节点间最优路径的求取

实际上, 一般最短路径的精确求解可以通过穷举法或借助于动态规划完成, 但其计算量随节点数目的增加呈指数增加, 形成所谓 NP 维数灾难. 为解决这一问题, 人们提出了多种寻优方法, 但这些方法仍存在一些问题, 如计算量较大、容易陷入局部最优、难以达到寻优精度等. 对于标准矩形网络, 这一问题的求解相对简单些, 其原因在于从任意节点 $x_{i,j}$ 出发到同行节点 $x_{i,j+t}$ (或同列节点 $x_{i+w,j}$) 的最佳路径是直接前往, 而不是绕行.

不失一般性, 同时为了描述方便, 下面以寻找节点 $x_{1,1}$ 到 $x_{m,n}$ 的最短路径为例对算法进行说明. 定义 y_i 为行进中所走过的第 i 段小矩形边的边长, Ω 为网络中节点的集合. 由定理 1 和定理 2 可知, $y_1 \in \{d(x_{1,1}, x_{1,2}),\ d(x_{1,1}, x_{2,1})\}$. 记 $y_i = d(x_{u,t}, x_{v,w})$, 则有

$$y_{i+1} \in \{d(x_{v,w}, x_{v+1,w}),\ d(x_{v,w}, x_{v,w+1})\},\quad v < m,\ w < n; \tag{3}$$

$$y_{i+1} = d(x_{v,w}, x_{v,w+1}) = d(x_{m,w}, x_{m,w+1}),\quad v = m; \tag{4}$$

$$y_{i+1} = d(x_{v,w}, x_{v+1,w}) = d(x_{v,n}, x_{v+1,n}),\quad w = n. \tag{5}$$

于是从节点 $x_{1,1}$ 到 $x_{m,n}$ 的最短路径问题可描述为

$$\min_{x_{i,j}\in\Omega} f(y_1, y_2, \cdots, y_{m+n-2}) = \min_{x_{i,j}\in\Omega} \sum_{k=1}^{m+n-2} y_k. \tag{6}$$

根据标准矩形网络的性质, 采用 SRNSP 算法求解上述问题, 具体流程如下.

Step 1: 设定初始参数. 种群规模 N, 最大迭代次数 G, 初始搜索概率 $P_1 = 0.5$.

Step 2: 生成初始解向量. 将 N 个维数为 $D = m + n - 2$ 的正数向量作为每一代的路径长度向量, 记第 g 代的第 i 条路径分段长度向量为

$$y_i^g = [y_{i,1}^g, y_{i,2}^g, \cdots, y_{i,D}^g]^{\mathrm{T}},\quad i = 1, 2, \cdots, N,\ g = 1, 2, \cdots, G.$$

令 $g = 1$, 取随机数 $F_1 = \mathrm{rand}(0, 1)$, 则有

$$y_{i,1}^1 = \begin{cases} d(x_{1,1}, x_{1,2}),\ F_1 \geqslant P_1; \\ d(x_{1,1}, x_{2,1}),\ F_1 < P_1. \end{cases}$$

按式 (3) ∼ (5) 得到第 1 代解向量

$$y_i^1 = [y_{i,1}^1, y_{i,2}^1, \cdots, y_{i,D}^1]^{\mathrm{T}},\ i = 1, 2, \cdots, N.$$

Step 3: 计算对应解向量 y_i^g 的目标函数值 (即对应的路径长度)

$$J_i^g(y_i^g) = \sum_{j=1}^{D} y_{i,j}^g,\ i = 1, 2, \cdots, N.$$

记 y_{best}^g 为第 g 次搜索得到的最优解, 对应的路径长度为 J_{best}^g.

Step 4: 判断终止条件. 若 $g = G + 1$, 则停止运算, 得到最优解 y_{best}^G, 否则转至 Step 5.

Step 5: 在集合 $\Delta = \{2, 3, \cdots, m + n - 2\}$ 中等概率随机选取 $T \in \Delta$, $F_g = \mathrm{rand}(0, 1)$. 从第 T 个位置开始判断当前解与最优解路径的相对位置, 使当前解向最优解位置方向靠近. 设 y_i^g 和 y_{best}^g 的第 j 个分量分别为 $y_{i,j}^g = d(x_{u,t}, x_{v,w})$ 和 $y_{\mathrm{best},j}^g = d(x_{a,b}, x_{c,d})$, $j = 2, 3, \cdots, m+n-2$, 搜索概率 $P = 1-(J_{\mathrm{best}}^g)/(J_i^g)$, 构造解向量

$$\tilde{y}_{i,j}^{g+1} = \begin{cases} y_{i,j}^g,\ j < T; \\ d(x_{v,w}, x_{v+1,w}),\ v < c,\ F_g \leqslant P,\ j \geqslant T; \\ d(x_{v,w}, x_{v,w+1}),\ v < c,\ F_g > P,\ j \geqslant T; \\ d(x_{v,w}, x_{v,w+1}),\ v > c,\ F_g \leqslant P,\ j \geqslant T; \\ d(x_{v,w}, x_{v,w+1}),\ v > c,\ F_g > P,\ j \geqslant T; \\ d(x_{v,w}, x_{v,w+1}),\ v = m,\ j \geqslant T; \\ d(x_{v,w}, x_{v+1,w}),\ w = n,\ j \geqslant T; \\ d(x_{v,w}, x_{v,w+1}),\ v = c,\ F_g \leqslant 0.5,\ j \geqslant T; \\ d(x_{v,w}, x_{v+1,w}),\ v = c,\ F_g > 0.5,\ j \geqslant T. \end{cases}$$

Step 6: 进行贪婪选择. 若

$$J_i^g = \sum_{i=1}^{D} y_i^g > \sum_{i=1}^{D} \tilde{y}_i^{g+1},$$

则取 $y_i^{g+1} = \tilde{y}_i^{g+1}$, 否则取 $y_i^{g+1} = y_i^g$.

Step 7: $g = g + 1$, 转至 Step 3.

此算法具有两个显著特点: 一是运算简单, 二是易于实现. 整个算法中只需人为确定种群个数 N 和最大迭代次数 G 两个参数, 在目前所有智能优化算法中是所需参数最少的. 事实上, 在各种智能优化算法中, 一般都含有一些需要人为确定的参数, 这些参数的取值对寻优性能和结果具有很大的影响, 对于不同问题, 往往需要赋予不同的数值. 遗憾的是, 对于需要解决的问题, 人们是无法给出确定的数值, 只能通过自适应的手段加以解决, 此算法中第 3 个参数 P 便是通过自适应的方法赋值的.

3　算法分析

本文算法是针对求取标准矩形网络中两节点间最短路径而设计的, 具有明显的针对性. 现有各类算

法中很难确保寻优中每次所得到的解向量都是可行解, 因此每次均需要进行解的判断, 无疑增加了算法的复杂度. 本文算法很好地解决了这一问题, 每次寻优中所得到的解向量都是可行解.

每次寻优均是在连接起点与终点间由最少小矩形边所组成的路径集合中进行搜索, 不仅保证了所得解的可行性, 而且大大缩小了搜索范围, 提高了搜索效率. 寻优中采取贪婪准则, 确保种群中每个个体在每一次寻优中所得的解均不劣于其历史解, 从而得到的整个种群最好解均不劣于种群历史最好解, 这样可以保证算法的收敛性. 算法中可行路径集合中路径受到较为苛刻的限制, 使得每条路径所对应的长度间差异并非特别显著. 因此, 搜索概率 P 值不会很大, 搜索具有很强的随机性, 可以在很大程度上避免陷入局部最优.

4 网络的转化

前文定义的标准矩形网络是一类特殊的网络, 有些网络虽然不是标准矩形网络但是可以转化为标准矩形网络, 尤其在处理道路交通问题时, 道路网络往往可以全部或部分转化为标准矩形网络, 其原因在于道路网络通常满足如下几何性质: n 边形的任意一边边长小于其他各边边长之和.

网络内部某一节点若存在 4 条以上路径, 则可以将其分成相互间距离为零的两个节点, 使得分开后每个节点所连接路径数小于等于 4 (对于道路网络几乎没有与 8 条以上路径相连接的节点); 若所连接路径小于 4, 则可以增加虚拟节点和虚拟边将其转换为矩形网络, 然后通过定义各条虚拟边长 (路径) 使其满足标准矩形网络的要求. 网络的边点和角点也可以作类似处理. 例如, 图 1 为一个不规范网络, 可以转换为如图 2 所示的等价标准矩形网络.

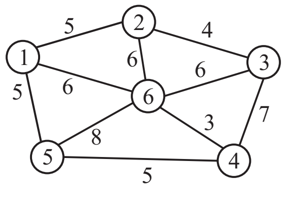

图 1 不规范网络

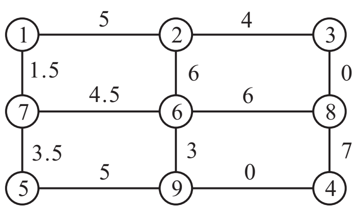

图 2 等价的标准矩形网络

5 仿真实验

在硬件环境为 Intel(R) Core(TM)2 Quad CPU, Q9400@2.66 GHz, 3.50 GB RAM 的计算机上使用 Matlab 7.5.0 编程进行相关仿真实验. 这里的最短路径问题属于离散优化问题, 当前主流算法中只有蚁群算法是针对离散优化问题的, 其他算法主要针对连续优化问题, 如果用来处理离散问题, 则需要对问题或算法作一定的转化, 这样不仅会增加问题的复杂性, 而且会影响算法的通用性. 因此, 将本文算法与蚁群算法、Dijkstra、Floyd、A* 算法进行仿真比较.

实验中, Dijkstra、Floyd、A* 无需参数设置: 蚁群算法中, 参数 $\alpha = 1$, $\beta = 5$, $\rho = 0.9$, 迭代次数为 100, 蚂蚁个数为 30; SRNSP 算法中, $N = 20$, $G = 20$.

5.1 仿真实验 1

在研究城市道路网时, 通常将各条道路的交叉点作为节点, 根据其坐标画出网络, 在每条边上表明相邻节点间距离. 由于实际问题的复杂性, 所得到的网络结构图往往不是前面所定义的标准矩形网络, 但是可以将其部分或全部转化为标准矩形网络.

青年大街是沈阳市的一条主要交通干道, 其部分路段的道路网络节点如图 3 所示. 各路段的权值为该路段的距离, 单位为 m, 各路段的权值均标示在路段上. 显然, 其道路网络并不是标准矩形网络, 可以通过引入虚拟节点和虚拟路径, 将其转化为等价的标准矩形网络 (见图 4), 基于图 4 解决图 3 中的最短路径问题.

仿真实验中, 分别求取图 3 中节点 1 到节点 32 的最短路径和节点 30 到节点 6 的最短路径, 前者的实际最短路径为 1 500, 后者的实际最短路径为 951. 各算法均运行 20 次, 具体结果见表 1 和表 2. 由表 1 和表 2 可见, 本文方法和 Floyd 算法在 20 次寻优中均得到了实际最短路径, 不仅具有良好的寻优精度, 也具有很好的稳定性; Dijkstra、A* 方法虽然稳定性也较好, 但均未能得到实际最短路径; ACO 算法在节点 1 到节点 30 的 20 次寻优中不但没有找到实际最优解, 而且所得最短路径的平均值为 1 615.5, 方差为 32.68, 在节点 30 到节点 6 的 20 次寻优中虽能够找到最优解, 所得最短路径的平均值为 954.3, 方差为 5.78, 但其稳定性较差. 从寻优时间来看, 本文算法慢于 Dijkstra 和 Floyd 算法, 快于 A* 算法, 显著快于 ACO 算法.

5.2 仿真实验 2

仿真实验 1 结果表明, 对于节点较少的网络, 本文算法和 Floyd 算法搜索性能大致相当. 为了比较本文算法对于节点较多的网络最短路径寻优性能, 针对图 5 具有 100 个节点的网络节点求取最短路径, 相邻节点间距离标示在路段上 (单位: 千米), 实验结果见表 3. 由表 3 可见, Dijkstra、A* 算法未能搜索到最优

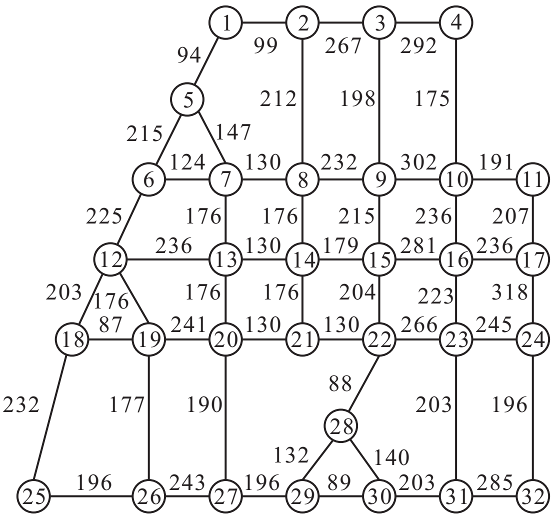

图 3　沈阳市青年大街附近道路节点

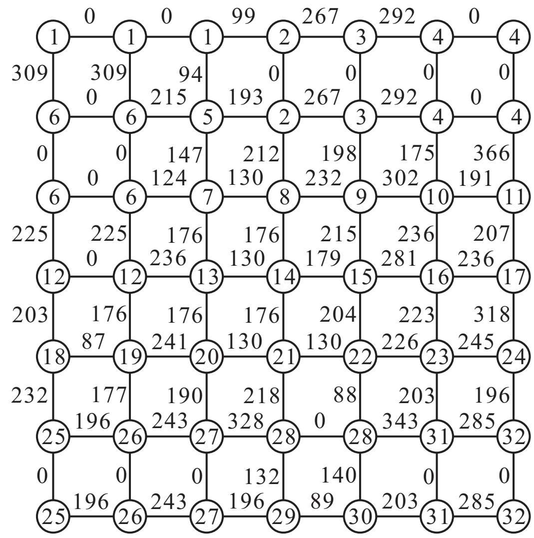

图 4　沈阳市青年大街附近标准矩形网络节点

表 1　节点 1 到节点 32 间最优路径的仿真结果

算法	最好解	最差解	平均值	方差	最优路径	平均时间/s
Dijkstra	1 577	1 577	1 577	0	1→2→8→14→15→22→23→24→32	0.006 99
Floyd	1 500	1 500	1 500	0	1→2→8→14→21→22→23→24→32	0.001 22
A*	1 556	1 556	1 556	0	1→5→7→13→20→27→29→30→31→32	0.033 12
ACO	1 560	1 650	1 615.5	32.68	1→5→7→13→20→21→22→23→24→32	12.403
SRNSP	1 500	1 500	1 500	0	1→2→8→14→21→22→23→24→32	0.008 2

表 2　节点 30 到节点 6 间最优路径的仿真结果

算法	最好解	最差解	平均值	方差	最优路径	平均时间/s
Dijkstra	964	964	964	0	30→28→22→21→14→8→7→6	0.007 14
Floyd	951	951	951	0	30→29→27→20→13→7→6	0.001 99
A*	1 133	1 133	1 133	0	30→28→22→15→9→8→7→6	0.036 3
ACO	951	964	954.3	5.78	30→29→27→20→13→7→6	9.748
SRNSP	951	951	951	0	30→29→27→20→13→7→6	0.012 1

表 3　节点 12 到节点 87 间最优路径的仿真结果

算法	最好解	最差解	平均值	方差	最优路径	平均时间/s
Dijkstra	38	38	38	0	12→2→3→4→5→6→16→26→36→46→56→57→67→77→87	0.076 6
Floyd	34	34	34	0	12→22→32→42→43→44→45→46→56→57→67→77→87	0.031 3
A*	43	43	43	0	12→22→32→42→52→62→72→82→83→84→85→86→87	0.094 1
ACO	34	40	36.7	1.35	12→22→32→42→43→44→45→46→56→57→67→77→87	130.82
SRNSP	34	34	34	0	12→22→32→42→43→44→45→46→56→57→67→77→87	0.021

解, 所搜到的最好路径大于 Floyd、ACO 和 SRNSP 算法; ACO 方法虽然搜到了最优解, 但 20 次搜索得到的解的平均值和方差均大于 Floyd 和 SRNSP 算法, 特别是平均搜索时间远远大于 Floyd 和 SRNSP 算法; Floyd 和 SRNSP 算法在最优解、最差解、平均值和方差方面表现最好, 在寻优时间上 SRNSP 方法明显优于 Floyd 和其他算法. 对比实验 1 和实验 2 的结果可见, 当网络节点数量由 32 增加到 100 时, SRNSP 算法和 Floyd 算法的平均寻优时间由 6.72 和 6.08 下降到 0.67. 由于篇幅所限, 没有对更大规模的标准矩形网络进行仿真实验, 但不难看出, 对于标准矩形网络, 随着节点数量的增加, SRNSP 方法在求解最短路径的寻优速度优势会显著增大.

6　结　论

本文针对常见的交通道路中的最短路径问题, 提出了标准矩形网络的概念, 对标准矩形网络的路径特点进行了分析, 指出某些非标准矩形网络可以通过增加虚拟节点和虚拟路径的方法得到等价的标准矩形网络. 针对标准矩形网络给出了一种新的最短路径求解算法 SRNSP, 并与 Dijkstra、Floyd、ACO、A* 算法进行了仿真对比实验. 实验结果表明, SRNSP 算法不仅具有良好的寻优精度和稳定性, 而且具有明显的寻优速度优势, 是求解大规模标准矩形网络节点间最短路径的一种极为有效的工具. SRNSP 的缺欠在于不能直接用于处理一般网络的最短路径问题, 应用时需要将一般网络转化为等价的标准矩形网络.

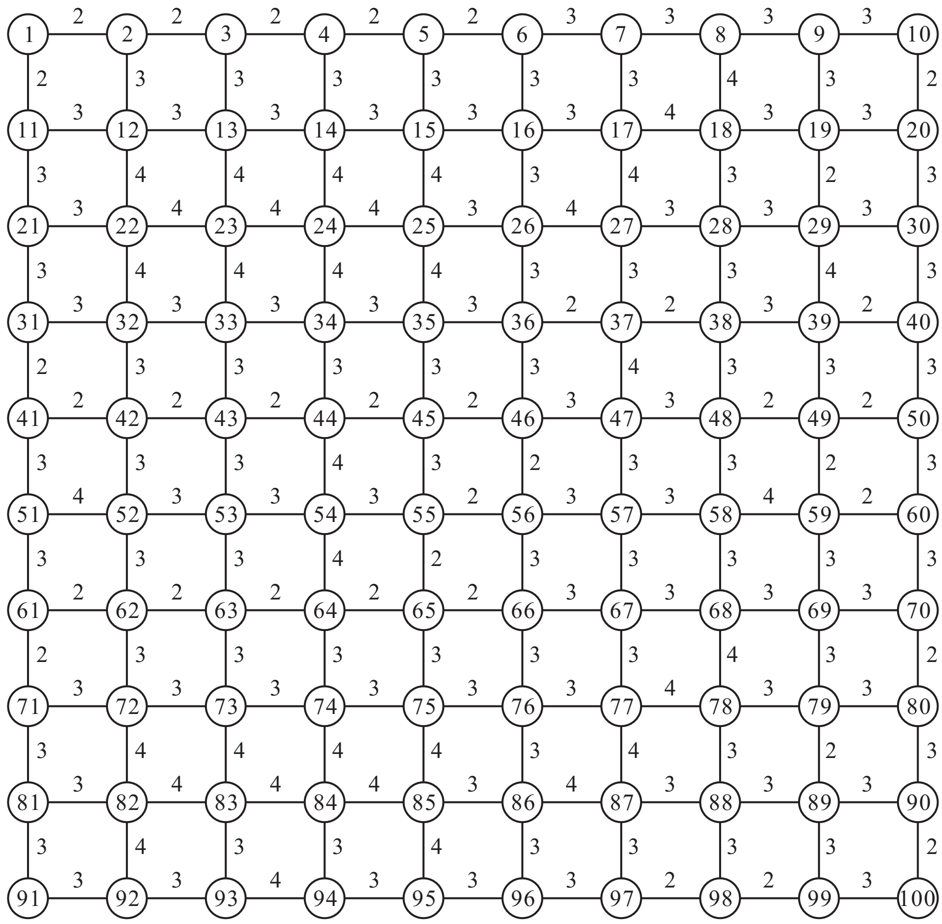

图 5 100 个节点无向带权网络节点

参考文献(References)

[1] Yu F, Li Y J, Wu T J. A temporal ant colony optimization approach to the shortest path problem in dynamic scale-free networks[J]. Physica A: Statistical Mechanics and its Applications, 2010, 389(3): 629-636.

[2] Du W B, Wu Z X, Cai K Q. Effective usage of shortest paths promotes transportation efficiency on scale-free networks[J]. Physica A: Statistical Mechanics and its Applications, 2013, 392(17): 3505-3512.

[3] López D, Lozano A. Techniques in multimodal shortest path in public transport systems[J]. Transportation Research Procedia, 2014, 3(1): 886-894.

[4] Han D H, Kim Y D, Lee J Y. Multiple-criterion shortest path algorithms for global path planning of unmanned combat vehicles[J]. Computers & Industrial Engineering, 2014, 71(1): 57-69.

[5] Duque D, Lozano L, Medaglia A L. An exact method for the biobjective shortest path problem for large-scale road networks[J]. European J of Operational Research, 2015, 242(3): 788-797.

[6] Liu L Z, Yang J H, Mu H B, et al. Exact algorithms for multi-criteria multi-modal shortest path with transfer delaying and arriving time-window in urban transit network[J]. Applied Mathematical Modelling, 2014, 38(9): 2613-2629.

[7] Dijkstra E W. A note on two problems in connexion with graphs[J]. Numerische Mathematik, 1959, 1(1): 269-271.

[8] Floyd R W. Algorithm 97: Shortest path[J]. Communications of the ACM, 1962, 5(6): 345.

[9] Hart P E, Nilsson N J, Raphael B. A formal basis for the heuristic determination of minimum cost paths[J]. IEEE Trans on Systems Science and Cybernetics, 1968, 4(2): 100-107.

[10] Colorni A, Dorigo M, Maniezzo V. Distributed optimization by ant colonies[C]. Proc of the 1st European Conf on Artificial Life. Paris: Elsevier Publishing, 1991, 142: 134-142.

[11] Mohemmed A W, Sahoo N C, Geok T K. Solving shortest path problem using particle swarm optimization[J]. Applied Soft Computing, 2008, 8(4): 1643-1653.

[12] Hassanzadeh R, Mahdavi I, Mahdavi-Amiri N, et al. A genetic algorithm for solving fuzzy shortest path problems with mixed fuzzy arc lengths[J]. Mathematical and Computer Modelling, 2013, 57(1): 84-99.

(责任编辑：郑晓蕾)