北方交通大学学报
1992年 第16卷 第1期 JOURNAL OF NORTHERN JIAOTONG UNIVERSITY Vol.16 No.1 1992

中国邮路问题的0-1规划解法

廖业元
(工业与建筑管理工程系)

摘 要：在用“奇偶点图上作业法”求解“中国邮路问题”时，需检查图中的每一个回路。当图中回路较多时，检查不便且易出错。针对此，本文建立了求解“中国邮路问题”的0-1规划模型，并给出了算例。

关键词：0-1规划；最短路；连通图/中国邮路问题；奇偶点图上作业法
分类号：O221.4

1 建立模型

“一个邮递员从邮局出发递送邮件，必须至少一次地走过他管辖范围内的每一条街道，然后再返回邮局．在此前提下，如何选择递送路线，以便走尽可能少的路程”．这个问题称为“中国邮路问题”．许多实际问题可以归纳为此模型．“中国邮路问题”是在图论中一个既与欧拉图又与最短通路有关的问题．

“中国邮路问题”用图论的术语来描述就是：把邮递员管辖的街道视为非负赋权无向网络（即赋权无向图）N，网络中的一条边 e 表示一条街道，其权值 $\omega(e)$ 即为街道的长度．在此网络N中，找到一条闭链 Q^* 包含N中的每条边至少一次，而且具有最小的长度．亦即有：

$$\omega(Q^*)=\min\{\omega(Q)=\sum\omega(e)\,|\,Q\text{ 为 N 中一条包含 N 全部边的闭链}\}$$

若网络N是一个连通欧拉图，根据求欧拉环游算法，求出一条闭的欧拉环游即为所求的最短邮递路线路 Q^*．

若网络N是一个连通M图（只有两个顶点的度数为奇数），则存在一条欧拉链，包含N中的所有边．此时所要求的最短邮递路线 Q^* 即为这样的一条欧拉链和在N中这条欧拉链的两个端点间的最短通路组成．

但在一般情况下，无向网络N既非欧拉图，又非M图，其奇数度数的顶点个数多于2个时，此时所要求的邮递路线必须要有更多重复的边．在这种情况下，利用所谓的“奇偶点图上作业法”可求出所需路线．其方法如下：

本文收到日期 1991-06-29

（设网络 N 中奇阶顶点为 $2q$ 个）

1．把 N 中的 $2q$ 个奇顶点分成 q 对，每对顶点间取一条初等链，该 q 条初等链中边的全体作平行的添加边作为初始可行解 F；

2．若 F 中有平行边，则删除偶数条平行边；

3．对 N 中任一回路 C，检查是否全部满足：

$$\omega(C_1)\leqslant\frac{1}{2}\omega(C)$$

其中：$C_1=\{e\mid e\in C,\ e$ 有相应添加边 $e'\in F\}$

若全部满足上式，则 F 为最优解，在网络 $N'=N\cup F$ 中运用求欧拉环游算法可得欧拉环游 Q，即为最短邮递路线。否则将 C_1 在 F 中相应的添加边置换成 $C-C_1$ 所相应的添加边。再进行第 2 步。最终可求出最优解。

从上面的“奇偶点图上作业法”中可以看出，在进行上述的第 3 步时，要检查 N 中的每一个回路，而这不是很容易的。在一个“日”字型的网络中，有 3 个回路；而在“田”字型的网络中，则有 13 个回路。稍为复杂一点的网络，回路会更多，检查起来就很不方便，而且易出错。同时在上述的回路调整过程中，某一个回路调整合格后，还可能影响本已合格的回路。因此“奇偶点图上作业法”还不够理想。针对此种方法之不足，下面提出另一种求解最短邮递路线的方法。

设无向网络：$N=(V,\ E)$ 有 $2q$ 个（$q\geq 2$）奇顶点，

$$V=\{V_1,V_2,\cdots,V_n\}$$

$$E=\{e_{ij}=(V_i,V_j)\mid V_i,V_j\in V\}$$

$$\text{边权：}\ \omega_{ij}=\begin{cases}\omega(e_{ij}), & \text{当 } e_{ij}\in E\ \text{（表示街道长度）}\\ M, & \text{当 } e_{ij}\notin E\ \text{（}M\text{ 为充分大正数）}\end{cases}$$

我们知道，要想在具有 $2q$ 个奇顶点的网络上找到一条包含网络中所有边的闭链 Q，必须把 $2q$ 个奇顶点分成 q 对，在每对顶点间最短路上边重复一次，所得网络可得到欧拉环游。如果某次划分 $2q$ 个奇顶点成 q 对后按最短路重复边所得闭链 Q^* 具有最小长度，其充要条件是：

1）每条边重复一次；2）在 N 的每个回路上，有重复边的长度和不超过回路长的一半。

从而可以看出，找最短邮递路线的问题，也就是一个适当地配对奇顶点的问题。于是我们可以通过下述方法来求解。

1.1 求网络中任意两顶点最短通路长度

用 R．W．Floyd 算法，求出网络中任意两顶点间的最短通路长度。方法是从矩阵 $D^{(0)}$ 出发，依次构造矩阵 $D^{(1)},D^{(2)},\cdots,D^{(n)}$。

其中：$D^{(0)}=[\omega_{ij}]$，ω_{ij} 为边 $e_{ij}=(V_i,V_i)$ 的权值（即街道长）。

$D^{(k)}=[d_{ij}^{(k)}]$，元素 $d_{ij}^{(k)}$ 表示从 V_i 到 V_j 而中间点仅属于 V_1 到 V_k 的 k 个点的所有通路中的最短通路长。

已知 $D^{(k-1)}=[d_{ij}^{(k-1)}]$，则 $D^{(k)}=[d_{ij}^{(k)}]$ 定义如下：

$$d_{ij}^{(k)}=\min\{d_{ij}^{(k-1)},\ d_{ik}^{(k-1)}+d_{kj}^{(k-1)}\}.$$

从 $k=1$ 开始，直至 $k=n$ 时停止。这时 $D^{(n)}$ 的元素 $d_{ij}^{(n)}$ 就是 V_i 到 V_j 的最短通路长。

1.2 从 $D^{(n)}$ 中列出任意两奇顶点距离

设 $2q$ 个奇阶顶点为 $V_1^1, V_2^1, \cdots, V_{2q}^1$，从矩阵 $D^{(n)}$ 中可得到如下任意两奇顶点间的距离。

$$
\begin{array}{c|cccc}
 & V_2^1 & V_3^1 & \cdots & V_{2q}^1 \\
\hline
V_1^1 & l_{12} & l_{13} & \cdots & l_{1,2q} \\
V_2^1 & & l_{23} & \cdots & l_{2,2q} \\
\vdots & & & \vdots & \vdots \\
V_{2q-1} & & & & l_{2q-1,2q}
\end{array}
$$

其中：l_{ij} 为 $D^{(n)}$ 中 V_i^1 到 V_j^1 的最短通路长。据此，可得到配对 $2q$ 个顶点的 0-1 规划算法。

1.3 0-1规划求解模型

设变量 x_{ij} 表示顶点 V_i^1 和 V_j^1 的配对关系

$$
x_{ij}=\begin{cases}1, & V_i^1 \text{ 和 } V_j^1 \text{ 配对时} \\ 0, & V_i^1 \text{ 和 } V_j^1 \text{ 不配对时}\end{cases}
$$

则有 $$\min z=\sum_{j>i}^{2q}\sum_{i=1}^{2q-1} l_{ij}x_{ij} \qquad (1)$$

$$
\begin{cases}
\sum_{j=2}^{2q} x_{1j}=1 & (2) \\
\sum_{i=1}^{2q-1} x_{i,2q}=1 & (3) \\
\sum_{i=1}^{k-1} x_{ik}+\sum_{j>k}^{2q} x_{kj}=1 \quad (k=2,\cdots,2q-1) & (4) \\
x_{ij}=0 \text{ 或 } 1, \quad (i=1,2,\cdots,2q-1;\ j=2,3,\cdots,2q)
\end{cases}
$$

式(1)是目标函数，表示在网络中重复边的总和最小；式(2)、式(3)、式(4)表示任一顶点只能与其他一个顶点有匹配关系。

1.4 模型的解

求解上述模型，可以得到其中的 q 个变量的值为1。据此可以对 $2q$ 个顶点进行匹配，并知道了每个顶点对间的最短通路长，找出这样的最短通路，把它们相应的边作为添加边，就可求得网络 N 上的最短邮递路线。

在上面的 0-1 规划求解中第1步要求任意两顶点间的最短距离，似乎使问题复杂化了，但实际利用 R. W. Floyd 算法，其工作量是不大的，而且可以借助计算机来完成。在后续的模型中，只与奇顶点有关系，比“奇偶点图上作业法”检查回路方法要优越得多。

2　应用举例

下面我们给出一个 0-1 规划求解算例。

对如下网络 N，求其中国邮路。

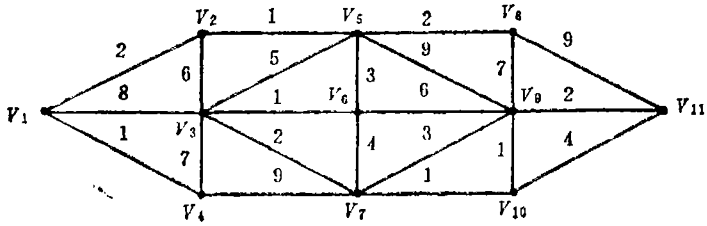

图 1　网络 N

其中 $V_1,V_2,V_4,V_5,V_7,V_8,V_{10},V_{11}$ 为奇顶点。

〔求解〕：

1．据网络图 N 可得 $D^{(0)}$ 如下：

$$D^{(0)}=\begin{bmatrix}
0 & 2 & 8 & 1 & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
2 & 0 & 6 & \infty & 1 & \infty & \infty & \infty & \infty & \infty & \infty \\
8 & 6 & 0 & 7 & 5 & 1 & 2 & \infty & \infty & \infty & \infty \\
1 & \infty & 7 & 0 & \infty & \infty & 9 & \infty & \infty & \infty & \infty \\
\infty & 1 & 5 & \infty & 0 & 3 & \infty & 2 & 9 & \infty & \infty \\
\infty & \infty & 1 & \infty & 3 & 0 & 4 & \infty & 6 & \infty & \infty \\
\infty & \infty & 2 & 9 & \infty & 4 & 0 & \infty & 3 & 1 & \infty \\
\infty & \infty & \infty & \infty & 2 & \infty & \infty & 0 & 7 & \infty & 9 \\
\infty & \infty & \infty & \infty & 9 & 6 & 3 & 7 & 0 & 1 & 2 \\
\infty & \infty & \infty & \infty & \infty & \infty & 1 & \infty & 1 & 0 & 4 \\
\infty & \infty & \infty & \infty & \infty & \infty & \infty & 9 & 2 & 4 & 0
\end{bmatrix}$$

其中：$d_{ij}^{(0)}$ 表示两顶点间边（V_i,V_j）的长度。

从 $D^{(0)}$ 出发，根据 R．W．Floyd 算法，可得到 $D^{(11)}$ 如下：

$$D^{(11)}=\begin{array}{c}
\\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ V_7 \\ V_8 \\ V_9 \\ V_{10} \\ V_{11}
\end{array}
\begin{array}{c}
\begin{array}{ccccccccccc} V_1 & V_2 & V_3 & V_4 & V_5 & V_6 & V_7 & V_8 & V_9 & V_{10} & V_{11} \end{array} \\
\begin{bmatrix}
0 & 2 & 7 & 1 & 3 & 6 & 9 & 5 & 11 & 10 & 13 \\
2 & 0 & 5 & 3 & 1 & 4 & 7 & 3 & 9 & 8 & 11 \\
7 & 5 & 0 & 7 & 4 & 1 & 2 & 6 & 4 & 3 & 6 \\
1 & 3 & 7 & 0 & 4 & 7 & 9 & 6 & 11 & 10 & 13 \\
3 & 1 & 4 & 4 & 0 & 3 & 6 & 2 & 8 & 7 & 10 \\
6 & 4 & 1 & 7 & 3 & 0 & 3 & 5 & 5 & 4 & 7 \\
9 & 7 & 2 & 9 & 6 & 3 & 0 & 8 & 2 & 1 & 4 \\
5 & 3 & 6 & 6 & 2 & 5 & 8 & 0 & 7 & 8 & 9 \\
11 & 9 & 4 & 11 & 8 & 5 & 2 & 7 & 0 & 1 & 2 \\
10 & 8 & 3 & 10 & 7 & 4 & 1 & 8 & 1 & 0 & 3 \\
13 & 11 & 6 & 13 & 10 & 7 & 4 & 9 & 2 & 3 & 0
\end{bmatrix}
\end{array}$$

其中：$d_{ij}^{(11)}$表示两顶点 V_i 与 V_j 的最短通路长度。

2. 从 $D^{(11)}$ 可得到任意两奇顶点间的距离

$$
\begin{array}{c}
 \\ V_1 \\ V_2 \\ V_4 \\ V_5 \\ V_7 \\ V_8 \\ V_{10}
\end{array}
\begin{array}{c}
\begin{array}{ccccccc} V_2 & V_4 & V_5 & V_7 & V_8 & V_{10} & V_{11} \end{array} \\
\begin{bmatrix}
2 & 1 & 3 & 9 & 5 & 10 & 13 \\
 & 3 & 1 & 7 & 3 & 8 & 11 \\
 & & 4 & 9 & 6 & 10 & 13 \\
 & & & 6 & 2 & 7 & 10 \\
 & & & & 8 & 1 & 4 \\
 & & & & & 8 & 9 \\
 & & & & & & 3
\end{bmatrix}
\end{array}
$$

分别用 $V_1^1, V_2^1, \cdots, V_8^1$ 代表奇顶点 $V_1, V_2, V_4, \cdots, V_{11}$，来构造下述模型。

3. 构造 0-1 规划模型

$$x_{ij} = \begin{cases} 1\text{，表示顶点 } V_i^1\text{，} V_j^1 \text{ 配对} \\ 0\text{，其他} \end{cases}$$

$$
\begin{aligned}
\min z = {} & 2x_{12} + 1x_{13} + 3x_{14} + 9x_{15} + 5x_{16} + 10x_{17} + 13x_{18} \\
& + 3x_{23} + 1x_{24} + 7x_{25} + 3x_{26} + 8x_{27} + 11x_{28} \\
& + 4x_{34} + 9x_{35} + 6x_{36} + 10x_{37} + 13x_{38} \\
& + 6x_{45} + 2x_{46} + 7x_{47} + 10x_{48} \\
& + 8x_{56} + 1x_{57} + 4x_{58} \\
& + 8x_{67} + 9x_{68} \\
& + 3x_{78}
\end{aligned}
$$

$$
\begin{cases}
x_{12} + x_{13} + x_{14} + x_{15} + x_{16} + x_{17} + x_{18} = 1 \\
x_{12} + x_{23} + x_{24} + x_{25} + x_{26} + x_{27} + x_{28} = 1 \\
x_{13} + x_{23} + x_{34} + x_{35} + x_{36} + x_{37} + x_{38} = 1 \\
x_{14} + x_{24} + x_{34} + x_{45} + x_{46} + x_{47} + x_{48} = 1 \\
x_{15} + x_{25} + x_{35} + x_{45} + x_{56} + x_{57} + x_{58} = 1 \\
x_{16} + x_{26} + x_{36} + x_{46} + x_{56} + x_{67} + x_{68} = 1 \\
x_{17} + x_{27} + x_{37} + x_{47} + x_{57} + x_{67} + x_{78} = 1 \\
x_{18} + x_{28} + x_{38} + x_{48} + x_{58} + x_{68} + x_{78} = 1 \\
x_{ij} = 0 \text{ 或 } 1, i = 1, 2, \cdots, 7, \quad j = 2, \cdots, 8
\end{cases}
$$

4. 求解上述 0-1 规划，得最优解

$$
\begin{cases}
x_{13} = 1 \\
x_{24} = 1 \\
x_{57} = 1 \\
x_{68} = 1 \\
x_{ij} = 0 \quad \text{（其他）}
\end{cases}
$$

即奇顶点 V_1^1 与 V_3^1、V_2^1 与 V_4^1，V_5^1 与 V_7^1，V_6^1 与 V_8^1 配对，亦即 V_1 与 V_4，V_2 与 V_5，

V_7与V_{10}，V_8与V_{11}配对，分别取其长度为1，1，1，9的通路加边即可求得中国邮路。即在如下网络 N′ 中求欧拉环游即为网络 N 所示的邮路问题的最优解。

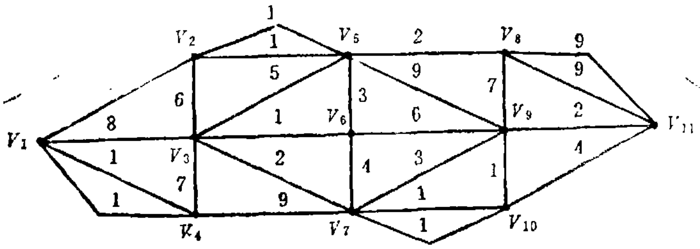

图 2　网络 N′

参　考　文　献

1　滕传琳．管理运筹学．北京：中国铁道出版社，1987．251～259

2　王朝瑞．图论．北京：高等教育出版社，1983．62～64

3　李尚．钱颂迪．运筹学．北京：清华大学出版社，1982．313～316

The 0–1 Programming Model of Chinese Postline Problem

Liao Yeyuan

(Department of Industrial and Constructional Engineering)

Abstract: It is required to check every circuit of a graph in order to find its Chinese postline by using operaton methcd on an odd-even vertex graph. The more circuits in the graph, the more complex to check and mcre mistakes likely to make. Therefore, this paper presents a 0-1 programming model solving the Chinese postline problem and an application example.

Key words: 0-1 programming; shortest path; connected graph/Chinese postline problem; operation method on an odd-even vertex graph