图论及其应用

北京邮电大学理学院

Ch2 最短路问题

Ch2 主要内容

- 最短路问题
- Dijkstra 算法
- Bellman-Ford算法
- Floyd-Warshall算法
- 最短路问题的应用

2.3 Floyd-Warshall算法

- 一点到多点最短路：**Dijkstra、Bellman-Ford**
- 有负权（没有负圈），若求所有点之间的最短路，怎么求？

  v次调用**Bellman-Ford**？

- $O(v^4)$ 或者至少$O(v^2\epsilon)$，复杂度高
- **Floyd-Warshall**算法，**1962**年

  适用多点对多点的最短路

算法原理

$$\begin{cases} u_{i,j}^{(1)} = w_{i,j} & i,j=1,\cdots,\upsilon \\ u_{i,j}^{(k+1)} = \min\{u_{i,j}^{(k)}, u_{i,k}^{(k)} + u_{k,j}^{(k)}\} & i,j,k,=1,\cdots,\upsilon \end{cases}$$

标号 $u_{i,j}^{(k)}$（$k=1,2,\cdots,\upsilon$）表示第 k 次迭代得到的顶点 v_i 到顶点 v_j（$1\le i,j\le\upsilon$）的临时标号（表示从 v_i 到 v_j 不通过顶点 $v_k,v_{k+1},\cdots v_\upsilon$ 的最短路或最短有向路路长）

最后得到的 $u_{i,j}^{(\upsilon+1)}$ 就是从顶点 v_i 到顶点 v_j 的最终路长；$w_{i,j}$ 表示顶点 v_i 到 v_j 的边或有向弧的权重;）

6

算法的正确性

定理 2-2 在式子（2-4）中，标号$u_{i,j}^{(k)}$（$i,j=1,\cdots,\upsilon$，$k=1,2,\cdots,\upsilon$）是不通过顶点$v_k,v_{k+1},\cdots v_\upsilon$（$v_i$，$v_j$除外）时从顶点$v_i$到顶点$v_j$的最短路（或有向路）路长。

证明：对k用归纳法。当$k=1$时，显然成立。假设结论对k时成立，下面考虑$k+1$时的情况。

从顶点v_i到顶点v_j且不通过顶点$v_{k+1},v_{k+2},\cdots v_\upsilon$的最短（有向）路有两种可能：

（1）该最短路不经过顶点v_k，则根据归纳假设，此最短路路长就是$u_{i,j}^{(k)}$；

（2）该最短路经过顶点v_k，则该最短路路长为由顶点v_k分开的两条子路的最短路路长之和，即$u_{i,k}^{(k)}+u_{k,j}^{(k)}$。在这两种可能中取最小值，正好就是$u_{i,j}^{(k+1)}$。

根据定理 2-2，当$k=\upsilon+1$时一定有$u_{i,j}^{(k)}$就是最短路路长$u_{i,j}$，即Floyd-Warshall算法一定在第υ步迭代后收敛，所以Floyd-Warshall算法是正确的。

算法复杂性

Floyd-Warshall 算法主要计算量是一个三重循环，最外层循环是对 k，循环 υ 次，里面分别是对所有顶点 i 和 j，所以其计算量至多为 $O(\upsilon^3)$。

算法

$\mathrm{Pr}_{i,j}^{(k)}$ 表示在第 k 次迭代时从顶点 i 到顶点 j 的当前最短路中第一条弧的头端点

最后根据最终的二维数组 $\mathrm{Pr}_{i,j}^{(\upsilon+1)}$，采用正向追踪的方式得到最短路。

（1）. $k=1$，对所有顶点 i 和 j，令 $u_{i,j}^{(1)}=w_{i,j}$； $\mathrm{Pr}_{i,j}^{(1)}=j$。

（2）. 对于所有顶点 i 和 j，若 $u_{i,j}^{(k)}\le u_{i,k}^{(k)}+u_{k,j}^{(k)}$，则令 $u_{i,j}^{(k+1)}=u_{i,j}^{(k)}$， $\mathrm{Pr}_{i,j}^{(k+1)}=\mathrm{Pr}_{i,j}^{(k)}$；否则令 $u_{i,j}^{(k+1)}=u_{i,k}^{(k)}+u_{k,j}^{(k)}$， $\mathrm{Pr}_{i,j}^{(k+1)}=\mathrm{Pr}_{i,k}^{(k)}$。

（3）. 如果 $k=\upsilon$，结束；否则令 $k=k+1$，转步骤（2）.

注：

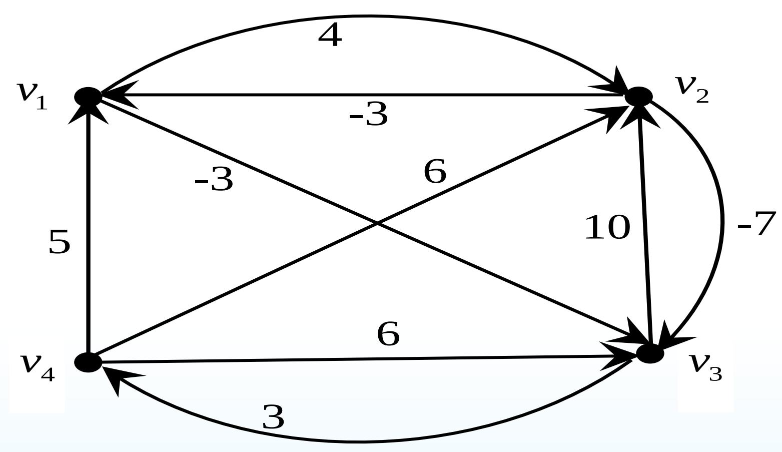
- **Floyd-Warshall**算法要将υ次关于k迭代进行完，才能保证得到所有正确的结果
- 适当修改 $\mathrm{Pr}_{ij}^{(k)}$ 的取值，也可以用**Floyd-Warshall**算法求出各种希望得到的路线（多条最短有向路，次最短有向路等）。但是会有额外的计算量。
- 也可以用于判断图（有向图）是否含有负圈

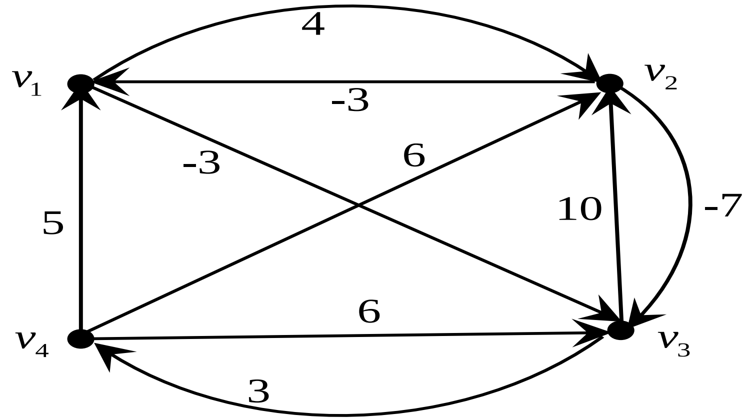
对任意的$k(1 \le \mathrm{k} \le \upsilon)$，如果图中没有负圈，则$u_{i,i}^{(k)} = 0 \ (1 \le i \le \upsilon)$；

如果在某次迭代时（存在$k_0 (1 \le \mathrm{k}_0 \le \upsilon)$）发现某个顶点（$v_j \ (1 \le j \le \upsilon)$）有$u_{j,j}^{(k_0)} < 0$，则说明图中有负圈。

例2-4：

- 求下面的赋权图（图2-10）中所有顶点之间的最短有向路的路长及路线。





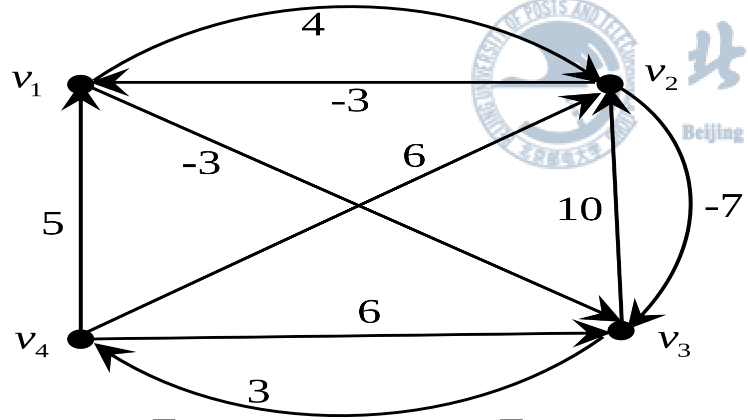
解：记最短路长矩阵为U，最短路矩阵为Pr，行和列都按照顶点v_1,v_2,v_3,v_4的顺序存放有关信息。则初始值为：

$$U^{(1)}=\begin{bmatrix}0 & 4 & -3 & \infty\\ -3 & 0 & -7 & \infty\\ \infty & 10 & 0 & 3\\ 5 & 6 & 6 & 0\end{bmatrix}\qquad \mathrm{Pr}^{(1)}=\begin{bmatrix}1 & 2 & 3 & 4\\ 1 & 2 & 3 & 4\\ 1 & 2 & 3 & 4\\ 1 & 2 & 3 & 4\end{bmatrix}$$

$$\boxed{u_{i,j}^{(k+1)}=\min\{u_{i,j}^{(k)},u_{i,k}^{(k)}+u_{k,j}^{(k)}\}\quad i,j,k,=1,\cdots,\upsilon}$$

第一次迭代后得到：

$$U^{(2)}=\begin{bmatrix}0 & 4 & -3 & \infty\\ -3 & 0 & -7 & \infty\\ \infty & 10 & 0 & 3\\ 5 & 6 & 2 & 0\end{bmatrix}\qquad \mathrm{Pr}^{(2)}=\begin{bmatrix}1 & 2 & 3 & 4\\ 1 & 2 & 3 & 4\\ 1 & 2 & 3 & 4\\ 1 & 2 & 1 & 4\end{bmatrix}$$



第一次迭代后得到：

$$U^{(2)} = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ \infty & 10 & 0 & 3 \\ 5 & 6 & 2 & 0 \end{bmatrix} \qquad \mathrm{Pr}^{(2)} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 1 & 4 \end{bmatrix}$$

第二次迭代后得到：

$$U^{(3)} = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ 7 & 10 & 0 & 3 \\ 3 & 6 & -1 & 0 \end{bmatrix} \qquad \mathrm{Pr}^{(3)} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 2 & 2 & 3 & 4 \\ 2 & 2 & 2 & 4 \end{bmatrix}$$

第二次迭代后得到：

$$U^{(3)} = \begin{bmatrix} 0 & 4 & -3 & \infty \\ -3 & 0 & -7 & \infty \\ 7 & 10 & 0 & 3 \\ 3 & 6 & -1 & 0 \end{bmatrix} \qquad \Pr^{(3)} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 2 & 2 & 3 & 4 \\ 2 & 2 & 2 & 4 \end{bmatrix}$$

第三次迭代后得到：

$$U^{(4)} = \begin{bmatrix} 0 & 4 & -3 & 0 \\ -3 & 0 & -7 & -4 \\ 7 & 10 & 0 & 3 \\ 3 & 6 & -1 & 0 \end{bmatrix} \qquad \Pr^{(4)} = \begin{bmatrix} 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 3 \\ 2 & 2 & 3 & 4 \\ 2 & 2 & 2 & 4 \end{bmatrix}$$

第四次迭代后得到：

$$U^{(5)} = \begin{bmatrix} 0 & 4 & -3 & 0 \\ -3 & 0 & -7 & -4 \\ 6 & 9 & 0 & 3 \\ 3 & 6 & -1 & 0 \end{bmatrix} \qquad \Pr^{(5)} = \begin{bmatrix} 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 3 \\ 4 & 4 & 3 & 4 \\ 2 & 2 & 2 & 4 \end{bmatrix}$$

最后得到的最短路长为$U^{(5)}$所示，最短路线根据$\mathrm{Pr}^{(5)}$正向追踪可得表 2-1：

终点 / 起点	v_1	v_2	v_3	v_4
v_1		v_1v_2	v_1v_3	$v_1v_3v_4$
v_2	v_2v_1		v_2v_3	$v_2v_3v_4$
v_3	$v_3v_4v_2v_1$	$v_3v_4v_2$		v_3v_4
v_4	$v_4v_2v_1$	v_4v_2	$v_4v_2v_3$	

$$U^{(5)}=\begin{bmatrix}0 & 4 & -3 & 0\\ -3 & 0 & -7 & -4\\ 6 & 9 & 0 & 3\\ 3 & 6 & -1 & 0\end{bmatrix}\qquad \mathrm{Pr}^{(5)}=\begin{bmatrix}1 & 2 & 3 & 3\\ 1 & 2 & 3 & 3\\ 4 & 4 & 3 & 4\\ 2 & 2 & 2 & 4\end{bmatrix}$$

北京邮電大學 Beijing University of Posts and Telecommunications

习题2-3

- 用Floyd-Warshall算法求下图中所有顶点之间的最短路路线及距离（弧旁的数字表示一条弧的距离）：