UE19CS251
Design and Analysis of Algorithms
Unit 5: Limitations of Algorithmic Power and Coping with the Limitations

All Pairs Shortest Path (Floyd's Algorithm)

PES University

Outline

Concepts covered

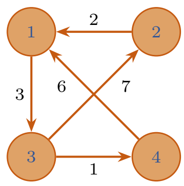
- All Pairs Shortest Path (Floyd's Algorithm)
  - Definition
  - Algorithm
  - Example

1 Problem Definition

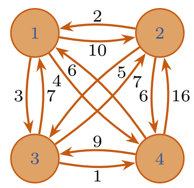
- Given an undirected or directed graph, with weighted edges, find the shortest path between every pair of vertices
  - Dijkstra's algorithm found shortest paths from given vertex to remaining $n-1$ vertices ($\Theta(n)$ paths)
  - Current problem is to find the shortest path between every pair of vertices ($\Theta(n^2)$ paths)
- Solution approach is similar to the transitive closure approach: Compute transitive closure via sequence of $n \times n$ matrices $R^{(0)}, \ldots, R^{(k)}, \ldots, R^{(n)}$ where $R^{(k)}[i,j] = 1$ iff there is nontrivial path from i to j with only first k vertices allowed as intermediate vertices
- Compute all pairs shortest paths via sequence of $n \times n$ matrices $D^{(0)}, \ldots, D^{(k)}, \ldots, D^{(n)}$ where $D^{(k)}[i,j]$ is the shortest path from i to j with only first k vertices allowed as intermediate vertices

2 Example

- Example of all pairs shortest paths:



$$\begin{array}{cc} & \begin{array}{cccc}1 & 2 & 3 & 4\end{array} \\ \begin{array}{c}1\\2\\3\\4\end{array} & \begin{bmatrix}0 & \infty & 3 & \infty \\ 2 & 0 & \infty & \infty \\ \infty & 7 & 0 & 1 \\ 6 & \infty & \infty & 0\end{bmatrix}\end{array}$$



$$\begin{array}{cc} & \begin{array}{cccc}1 & 2 & 3 & 4\end{array} \\ \begin{array}{c}1\\2\\3\\4\end{array} & \begin{bmatrix}0 & 10 & 3 & 4 \\ 2 & 0 & 5 & 6 \\ 7 & 7 & 0 & 1 \\ 6 & 16 & 9 & 0\end{bmatrix}\end{array}$$

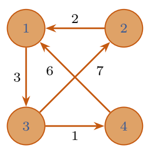
3 Algorithm

Transitive Closure (Floyd's Algorithm)

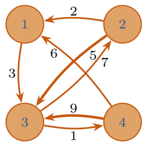
procedure FLOYD$(()A[1\ldots n, 1\ldots n])$
    ▷ **Input:** Weight matrix A of a graph with no negative length cycles
    ▷ **Output:** Distance matrix of shortest paths
    $D \leftarrow W$
    for $k \leftarrow 1$ **to** n **do**
        for $i \leftarrow 1$ **to** n **do**
            for $j \leftarrow 1$ **to** n **do**
                $D[i, j] \leftarrow \min(D[i, j], D[i, k] + D[k, j])$
    return D

- Complexity: $\Theta(n^3)$

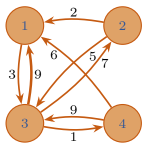
4 Example



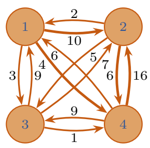
$$\begin{array}{cc} & \begin{array}{cccc}1 & 2 & 3 & 4\end{array} \\ \begin{array}{c}1\\2\\3\\4\end{array} & \begin{bmatrix}0 & \infty & 3 & \infty \\ 2 & 0 & \infty & \infty \\ \infty & 7 & 0 & 1 \\ 6 & \infty & \infty & 0\end{bmatrix}\end{array}$$



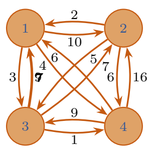
$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & \infty & 3 & \infty \\ 2 & 2 & 0 & \mathbf{5} & \infty \\ 3 & \infty & 7 & 0 & 1 \\ 4 & 6 & \infty & \mathbf{9} & 0 \end{array}$$



$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & \infty & 3 & \infty \\ 2 & 2 & 0 & 5 & \infty \\ 3 & \mathbf{9} & 7 & 0 & 1 \\ 4 & 6 & \infty & 9 & 0 \end{array}$$



$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & \mathbf{10} & 3 & \mathbf{4} \\ 2 & 2 & 0 & 5 & \mathbf{6} \\ 3 & 9 & 7 & 0 & 1 \\ 4 & 6 & \mathbf{16} & 9 & 0 \end{array}$$



$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 10 & 3 & 4 \\ 2 & 2 & 0 & 5 & 6 \\ 3 & \mathbf{7} & 7 & 0 & 1 \\ 4 & 6 & 16 & 9 & 0 \end{array}$$

5 Think About It

- Give an example of a graph with negative weights for which Floyd's algorithm does not yield the correct result
- Enhance Floyd's algorithm so that shortest paths themselves, not just their lengths, can be found