4. Greedy Algorithms II

- Dijkstra’s algorithm demo
- improved Dijkstra’s algorithm demo
4. Greedy Algorithms II

- Dijkstra’s algorithm demo
- improved Dijkstra’s algorithm demo
Dijkstra’s algorithm demo

- Initialize $S = \{s\}$ and $d(s) = 0$.
- Repeatedly choose unexplored node $v$ which minimizes

\[\pi(v) = \min_{e = (u,v): u \in S} d(u) + \ell_e,\]

add $v$ to $S$; set $d(v) = \pi(v)$. 

![Graph diagram showing nodes and edges with weights](image-url)
Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $d(s) = 0$.
- Repeatedly choose unexplored node $v$ which minimizes

$$\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,$$

add $v$ to $S$; set $d(v) = \pi(v)$. 

![Diagram of Dijkstra's algorithm process](attachment:diagram.png)
Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $d(s) = 0$.
- Repeatedly choose unexplored node $v$ which minimizes
  
  $$\pi(v) = \min_{e = (u,v) : u \in S} d(u) + \ell_e,$$

  add $v$ to $S$; set $d(v) = \pi(v)$. 

\[\begin{array}{c}
0 \\
4 \\
8 \\
16 \\
\end{array}\]

\[\begin{array}{c}
0 + 16 = 16 \\
4 + 3 = 7 \\
0 + 8 = 8 \\
\end{array}\]
Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $d(s) = 0$.
- Repeatedly choose unexplored node $v$ which minimizes

\[
\pi(v) = \min_{e = (u,v) : u \in S} d(u) + \ell_e,
\]

add $v$ to $S$; set $d(v) = \pi(v)$. 

![Graph diagram with nodes and edges labeled with distances.](image)
Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $d(s) = 0$.
- Repeatedly choose unexplored node $v$ which minimizes

$$
\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,
$$

add $v$ to $S$; set $d(v) = \pi(v)$. 

![Graph diagram with edge weights and distances](image-url)
Dijkstra’s algorithm demo

- Initialize $S = \{s\}$ and $d(s) = 0$.
- Repeatedly choose unexplored node $v$ which minimizes

$$\pi(v) = \min_{e = (u,v) : u \in S} d(u) + \ell_e,$$

add $v$ to $S$; set $d(v) = \pi(v)$.
Dijkstra’s algorithm demo

- Initialize $S = \{s\}$ and $d(s) = 0$.
- Repeatedly choose unexplored node $v$ which minimizes

$$
\pi(v) = \min_{e = (u,v) : u \in S} d(u) + \ell_e,
$$

add $v$ to $S$; set $d(v) = \pi(v)$. 

![Graph](image_url)
Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $d(s) = 0$.
- Repeatedly choose unexplored node $v$ which minimizes

$$
\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,
$$

add $v$ to $S$; set $d(v) = \pi(v)$. 

![Diagram of a graph with nodes and edges labeled with weights]
4. Greedy Algorithms II

- Dijkstra’s algorithm demo
- improved Dijkstra’s algorithm demo
• Initialize $S = \{ s \}$ and $\pi(s) = 0$.
• Repeatedly choose $u \notin S$ with minimum $\pi(v)$.
  - for each edge $(u, v)$ leaving $u$, set $\pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \}$
  - add $u$ to $S$
Improved Dijkstra’s algorithm demo

• Initialize $S = \{ s \}$ and $\pi(s) = 0$.
• Repeatedly choose $u \notin S$ with minimum $\pi(v)$.
  - for each edge $(u, v)$ leaving $u$, set $\pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \}$
  - add $u$ to $S$
Improved Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $\pi(s) = 0$.
- Repeatedly choose $u \notin S$ with minimum $\pi(v)$.
  - for each edge $(u, v)$ leaving $u$, set $\pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \}$
  - add $u$ to $S$
Improved Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $\pi(s) = 0$.
- Repeatedly choose $u \not\in S$ with minimum $\pi(v)$.
  - for each edge $(u, v)$ leaving $u$, set $\pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \}$
  - add $u$ to $S$
Improved Dijkstra’s algorithm demo

- Initialize $S = \{s\}$ and $\pi(s) = 0$.
- Repeatedly choose $u \notin S$ with minimum $\pi(v)$.
  - for each edge $(u, v)$ leaving $u$, set $\pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \}$
  - add $u$ to $S$
Improved Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $\pi(s) = 0$.
- Repeatedly choose $u \not\in S$ with minimum $\pi(v)$.
  - for each edge $(u, v)$ leaving $u$, set $\pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \}$
  - add $u$ to $S$
Improved Dijkstra’s algorithm demo

- Initialize \( S = \{ s \} \) and \( \pi(s) = 0 \).
- Repeatedly choose \( u \notin S \) with minimum \( \pi(v) \).
  - for each edge \((u, v)\) leaving \( u \), set \( \pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \} \)
  - add \( u \) to \( S \)
Improved Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $\pi(s) = 0$.
- Repeatedly choose $u \notin S$ with minimum $\pi(v)$.
  - for each edge $(u, v)$ leaving $u$, set $\pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \}$
  - add $u$ to $S$
Improved Dijkstra’s algorithm demo

- Initialize $S = \{ s \}$ and $\pi(s) = 0$.
- Repeatedly choose $u \not\in S$ with minimum $\pi(v)$.
  - for each edge $(u, v)$ leaving $u$, set $\pi(v) = \min \{ \pi(v), \pi(u) + \ell(u, v) \}$
  - add $u$ to $S$