Задачи нахождения кратчайшего пути

Определения

- Пусть дан ориентированный взвешенный граф *G=<V,E>*

с весовой функцией $w : E \to R$

Весом пути $p = \{v_0, v_1, \ldots, v_n\}$

называется сумма весов ребер, входящих в этот путь:

$$w(p) = \sum_{i=1}^{n} w(v_{i-1}, v_i)$$

Определения

Вес кратчайшего пути из u в v равен

$$\delta(u,v) = \begin{cases} \min\{w(p) : u \to v\} & \text{если существует путь из } u \text{ в } v \\ \infty, & \text{иначе} \end{cases}$$

Определения

Кратчайший путь из u в v – это любой путь *p* из *u* и *v*, для которого

$$w(p) = \delta(u, v)$$

Варианты задач
о кратчайшем пути

- *Кратчайший путь из одной вершины*:
  Дан взвешенный граф ***G=<V,E>***
  и начальная вершина **s**.
  Требуется найти кратчайшие пути из
  s во все вершины $\boldsymbol{v \in V}$
- *Кратчайшие пути в одну вершину:*
  Дана конечная вершина ***t.***
  Требуется найти кратчайшие пути в ***t***
  из всех вершин $\boldsymbol{v \in V}$

Варианты задач о кратчайшем пути

- *Кратчайший путь между парой вершин*:
  Даны вершины ***u*** и ***v***.
  Требуется найти кратчайший путь из ***u*** в ***v***
- *Кратчайшие пути для всех пар вершин:*
  Для каждой пары вершин ***u*** и ***v*** найти кратчайший путь из ***u*** в ***v***

Варианты задач о кратчайшем пути

- Часто в задачах бывает необходимо найти не только кратчайший путь, но и сам путь.
- Для каждой вершины ***v*** будем помнить ее предшественников **п(v)**

Свойства кратчайших путей

- ***Лемма 1. (отрезки кратчайших путей являются кратчайшими)*** Пусть дан ориентированный взвешенный граф $G=<V,E>$
с весовой функцией $w : E \to R$
Если $\boldsymbol{p(v_1, v_2, \ldots, v_k)}$ – кратчайший путь из v_1 в v_k и $1 \le i \le j \le k$, то
$\boldsymbol{p_{ij}=(v_i, v_{i+1}, \ldots, v_j)}$ – кратчайший путь из v_i в v_j

Свойства кратчайших путей

- *Следствие 1*
  Пусть дан ориентированный взвешенный граф $G=\langle V,E\rangle$
  с весовой функцией $w: E \to R$
  Рассмотрим кратчайший путь ***p*** из ***s*** в ***v.***
  Пусть ***u***→***v*** – последнее ребро этого пути.
  Тогда

$$\delta(s,v) = \delta(s,u) + w(u,v)$$

Свойства кратчайших путей

- *Лемма 2*

  Пусть дан ориентированный взвешенный граф $G=<V,E>$

  с весовой функцией $w: E \to R$

  Пусть $\boldsymbol{s \in V}$

  Тогда для всякого ребра $\mathbf{(u,v) \in E}$

$$\delta(s,v) \le \delta(s,u) + w(u,v)$$

Релаксация

- Для каждого ребра $\boldsymbol{v \in V}$ будем хранить некоторое число *d[v]*, являющееся верхней оценкой веса кратчайшего пути из вершины s в v (*оценка кратчайшего пути*)

Релаксация

- Начальное значение оценки кратчайшего пути и массива π определяется следующим образом:
- ***Initialize(G,s)***

Для всех вершин $v \in V$

$$d[v] = \infty$$

$$\pi[v] = NULL$$

$$d[s] = 0$$

Релаксация

Релаксация ребра (u, v) состоит в следующем:

Значение d[v] уменьшается до

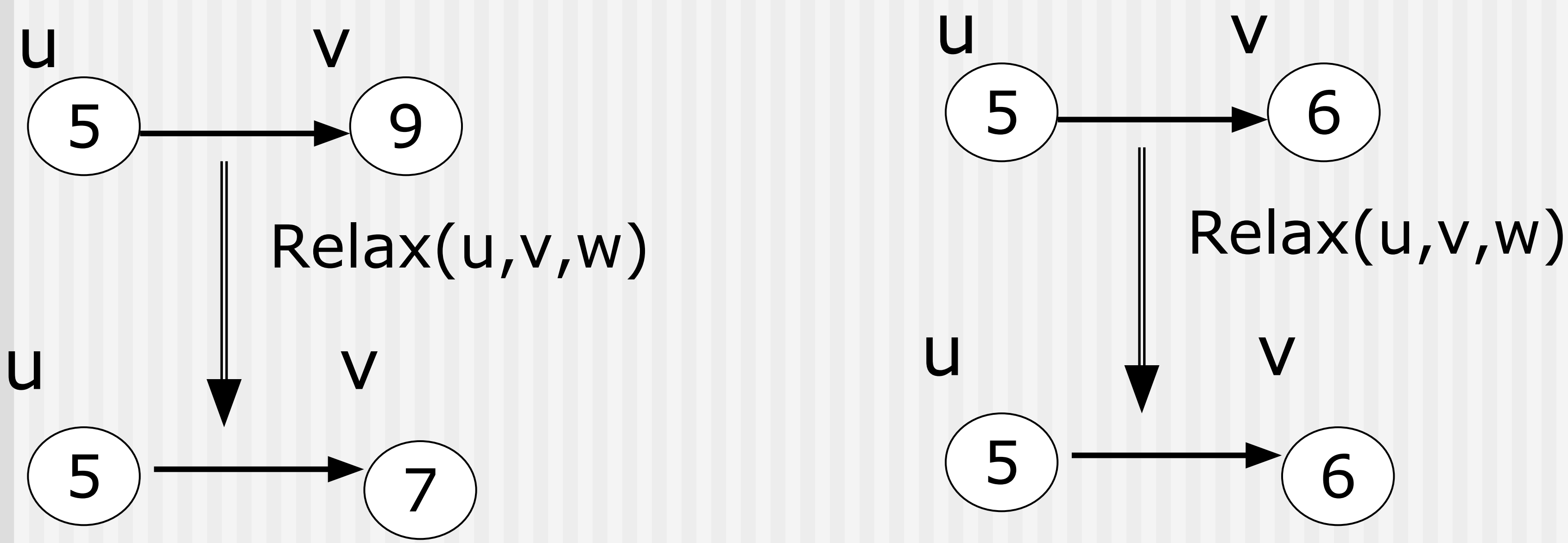
d[u]+w(u,v), если

второе значение меньше первого

При этом п(v)=u

Relax(u,v,w)

If (d[v]> d[u]+w(u,v))
d[v]=d[u]+w(u,v)
п[v]=u



В вершинах указаны оценки кратчайшего пути

Алгоритм Дейкстры

- Решает задачу о кратчайших путях из одной вершины s графа G=<V,E> с весовой функцией w до всех остальных вершин графа.
- Веса всех ребер неотрицательны

Алгоритм Дейкстры

- Алгоритм строит множество ***S*** вершин ***v***, для которых кратчайшие пути до вершины ***s*** уже известны, т. е. $d[v]=\delta(s,v)$
- На каждом шаге к множеству ***S*** добавляется та из оставшихся вершин ***u***, для которой $d[u]$ имеет наименьшее значение
- После этого проводится релаксация всех ребер, выходящих из ***u***

Алгоритм Дейкстры

- Вершины, не лежащие в множестве S, хранятся в очереди с приоритетами, определяемыми значениями функции d.
- Пусть граф представлен списками смежности
  Adj[u] –список смежных вершин u
  Q – очередь с приоритетами

Алгоритм Дейкстры

- Initialize(G,s)
  S=Ǿ
  Q=V[G]
  while Q<>Ǿ
  do u=min(Q) – выбираем вершину с наименьшим значением d[u]
  S=S U {u}
  for для всех вершин $v \in Adj[u]$
  do Relax(u,v,w)