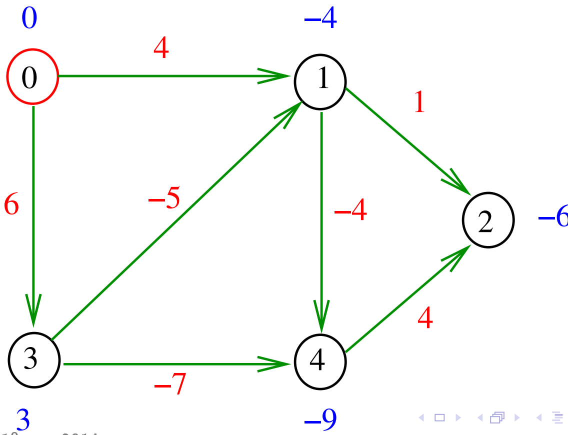
Mais Potenciais

Potenciais

Um **potencial** é um vetor y indexado pelos vértices do digrafo tal que

$y[w] - y[v] \leq$ `G->adj[v][w]` para todo arco `v-w`

Exemplo:

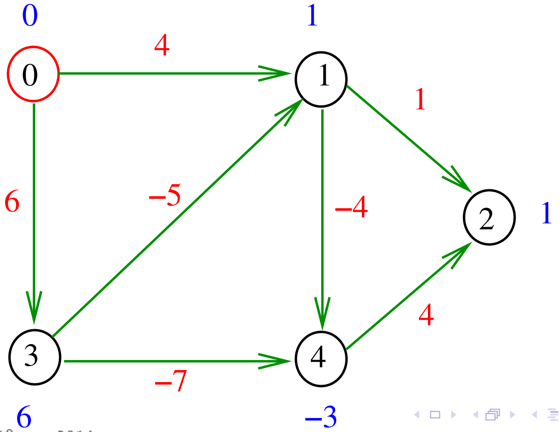


Potenciais

Um **potencial** é um vetor y indexado pelos vértices do digrafo tal que

$$y[w] - y[v] \leq \texttt{G->adj}[v][w] \text{ para todo arco } v\text{-}w$$

Exemplo:

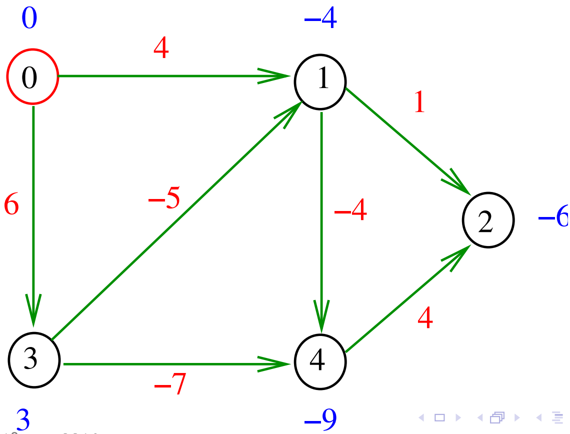


Potenciais

Um **potencial** é um vetor y indexado pelos vértices do digrafo tal que

$$y[w] \le y[v] + \texttt{G->adj}[v][w] \text{ para todo arco } v\text{-}w$$

Exemplo:

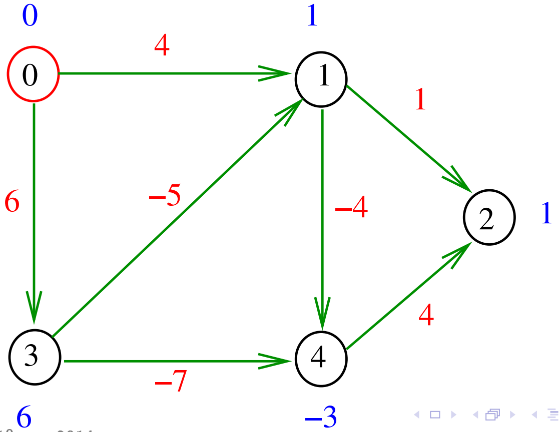


Potenciais

Um **potencial** é um vetor y indexado pelos vértices do digrafo tal que

$y[w] \le y[v] + \texttt{G->adj}[v][w]$ para todo arco v-w

Exemplo:

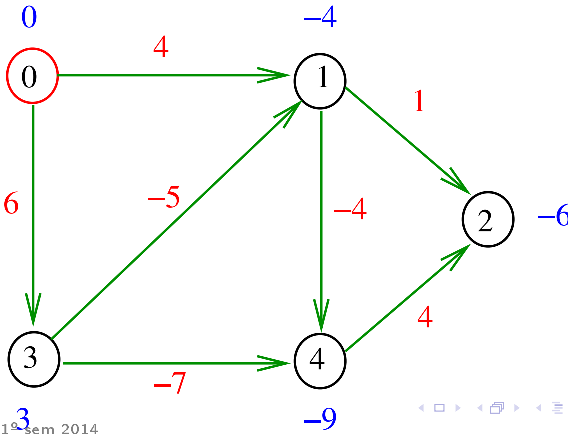


Propriedade dos potencias

Lema da dualidade. Se $\mathtt{P}$ é um caminho de $\mathtt{s}$ a $\mathtt{t}$ e $\mathtt{y}$ é um potencial, então

$$\mathtt{custo}(\mathtt{P}) \geq \mathtt{y}[\mathtt{t}] - \mathtt{y}[\mathtt{s}]$$

Exemplo:

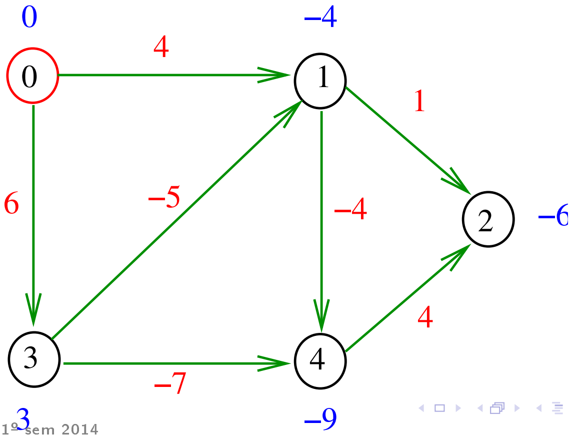


Propriedade dos potencias

Lema da dualidade. Se P é um caminho de s a t e y é um potencial, então

$$\texttt{custo}(P) + y[s] \geq y[t]$$

Exemplo:



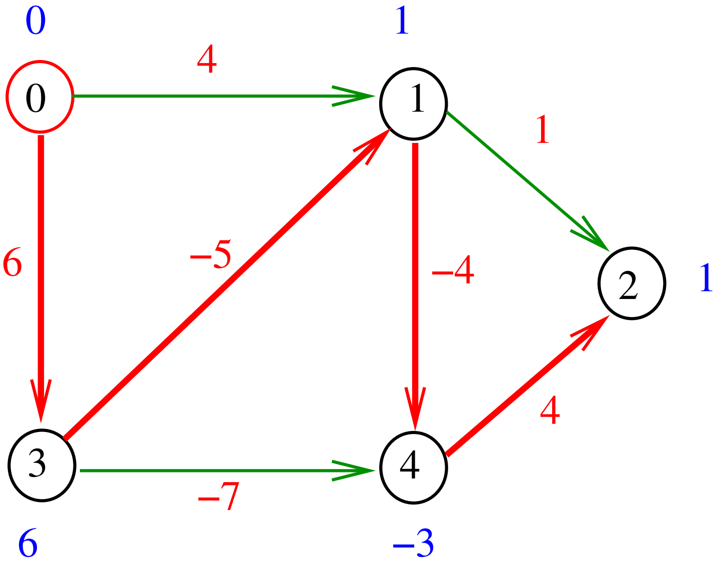
Conseqüência

Se P é um caminho de s a t e y é um potencial tais que

$$\texttt{custo}(P) = y[t] - y[s],$$

então P é um caminho **mínimo** e y é um potencial tal que $y[t] - y[s]$ é **máximo**

Exemplo



Teorema da dualidade

Da propriedade dos -potenciais (lema da dualidade) e da correção de `dijkstra` concluímos o seguinte:

> Se s e t são vértices de um digrafo com custo **não-negativos** nos arcos e t está ao alcance de s então
>
> $$\min\{\mathtt{custo}(\mathtt{P}) : \mathtt{P} \text{ é um } \mathtt{s}\ \mathtt{t}\text{-caminho}\}$$
> $$= \max\{\mathrm{y}[\mathtt{t}] - \mathrm{y}[\mathtt{s}] : \mathrm{y} \text{ é um potencial}\}.$$

Algoritmo Floyd-Warshall

S 21.3

Problema dos caminhos mínimos entre todos os pares

Problema: Dado um digrafo com custo nos arcos, determinar, para cada par de vértices `s`, `t` o custo de um caminho mínimo de `s` a `t`

Esse problema pode ser resolvido aplicando-se `V` vezes o algoritmo Bellman-Ford

O consumo de tempo dessa solução é $O(\mathtt{V}^2\mathtt{A})$.

Um algoritmo mais eficiente foi descrito por Floyd, baseado em uma idéia de Warshall.

O algoritmo supõe que o digrafo não tem ciclo negativo

Problema dos caminhos mínimos entre todos os pares

Problema: Dado um digrafo com custo nos arcos, determinar, para cada par de vértices `s`, `t` o custo de um caminho mínimo de `s` a `t`

Esse problema pode ser resolvido aplicando-se `V` vezes o algoritmo Bellman-Ford

O consumo de tempo dessa solução é $\mathrm{O}(\mathtt{V}^2\mathtt{A})$.

Um algoritmo mais eficiente foi descrito por Floyd, baseado em uma idéia de Warshall.

O algoritmo supõe que o digrafo não tem ciclo negativo

Problema dos caminhos mínimos entre todos os pares

Problema: Dado um digrafo com custo nos arcos, determinar, para cada par de vértices s, t o custo de um caminho mínimo de s a t

Esse problema pode ser resolvido aplicando-se V vezes o algoritmo Bellman-Ford

O consumo de tempo dessa solução é $O(V^2 A)$.

Um algoritmo mais eficiente foi descrito por Floyd, baseado em uma idéia de Warshall.

O algoritmo supõe que o digrafo não tem ciclo negativo

Programação dinâmica

$0, 1, 2, \ldots, \mathtt{V}{-}1$ = lista dos vértices do digrafo

$\mathtt{custo}[\mathtt{k}][\mathtt{s}][\mathtt{t}]$ = menor custo de um caminho de $\mathtt{s}$ a $\mathtt{t}$ usando vértices internos em $\{0, 1, \ldots, \mathtt{k}{-}1\}$

Recorrência:

$$\begin{aligned} \mathtt{custo}[0][\mathtt{s}][\mathtt{t}] &= \mathtt{G\text{-}>adj}[\mathtt{s}][\mathtt{t}] \\ \mathtt{custo}[\mathtt{k}][\mathtt{s}][\mathtt{t}] &= \min\{\mathtt{custo}[\mathtt{k}\text{-}1][\mathtt{s}][\mathtt{t}], \\ &\quad \mathtt{custo}[\mathtt{k}\text{-}1][\mathtt{s}][\mathtt{k}\text{-}1] + \mathtt{custo}[\mathtt{k}\text{-}1][\mathtt{k}\text{-}1][\mathtt{t}]\} \end{aligned}$$

Se o digrafo não tem ciclo negativo acessível a partir de $\mathtt{s}$, então $\mathtt{custo}[\mathtt{V}][\mathtt{s}][\mathtt{t}]$ é o menor custo de um **caminho simples** de $\mathtt{s}$ a $\mathtt{t}$

Programação dinâmica

$0, 1, 2, \ldots, \mathtt{V}{-}1$ = lista dos vértices do digrafo

$\mathtt{custo}[\mathtt{k}][\mathtt{s}][\mathtt{t}]$ = menor custo de um caminho de
s a t usando vértices internos em $\{0, 1, \ldots, \mathtt{k}{-}1\}$

Recorrência:

$$\begin{aligned}
\mathtt{custo}[0][\mathtt{s}][\mathtt{t}] &= \mathtt{G\text{->}adj}[\mathtt{s}][\mathtt{t}] \\
\mathtt{custo}[\mathtt{k}][\mathtt{s}][\mathtt{t}] &= \min\{\mathtt{custo}[\mathtt{k}\text{-}1][\mathtt{s}][\mathtt{t}], \\
&\quad \mathtt{custo}[\mathtt{k}\text{-}1][\mathtt{s}][\mathtt{k}\text{-}1] + \mathtt{custo}[\mathtt{k}\text{-}1][\mathtt{k}\text{-}1][\mathtt{t}]\}
\end{aligned}$$

Se o digrafo não tem ciclo negativo acessível a partir de s, então $\mathtt{custo}[\mathtt{V}][\mathtt{s}][\mathtt{t}]$ é o menor custo de um **caminho simples** de s a t

```
void floyd_warshall (Digraph G){
    Vertex s, t ; double d ;
    for (s =0; s < G->V ; s++)
        for (t =0; t < G->V ; t++)
            custo[0][s][t] = G->adj[s][t];
    for (k =1; k <=G->V ; k++)
        for (s =0; s < G->V ; s++)
            for (t =0; t < G->V ; t++){
                custo[k][s][t]=custo[k-1][s][t];
                d =custo[k-1][s][k-1]
                          +custo[k-1][k-1][t];
                if (custo[k][s][t] > d)
                    custo[k][s][t] = d ;
            }
        }
}
```

Consumo de tempo

O consumo de tempo da função
`floyd_warshall1` é $O(V^3)$.

```
void floyd_warshall (Digraph G){
    Vertex s, t ; double d ;
    for (s =0; s < G->V ; s++)
        for (t =0; t < G->V ; t++)
            cst[s][t] = G->adj[s][t];
    for (k =1; k <= G->V ; k++)
        for (s =0; s < G->V ; s++)
            for (t =0; t < G->V ; t++){
                d =cst[s][k-1]+cst[k-1][t];
                if (cst[s][t] > d)
                    cst[s][t] = d ;
            }
        }
}
```

Relação invariante

No início de cada iteração da linha 5 vale que

$\mathtt{cst}[\mathtt{s}][\mathtt{t}] = \mathtt{custo}[\mathtt{k}][\mathtt{s}][\mathtt{t}] =$ o menor custo de um caminho de $\mathtt{s}$ a $\mathtt{t}$ usando vértices internos em

$$\{0, 1, \ldots, \mathtt{k}-1\}$$

Novo resumo

função	consumo de tempo	observação
DAGmin	$O(V + A)$	digrafos acíclicos custos arbitrários
dijkstra	$O(A \lg V)$	custos ≥ 0, min-heap
	$O(V^2)$	custos ≥ 0, fila
bellman-ford	$O(V^3)$ $O(VA)$	digrafos densos digrafos esparsos
floyd-warshall	$O(V^3)$	digrafos sem ciclos negativos

O problema SPT em digrafos com ciclos negativos é NP-difícil.