Grundbegriffe der Informatik

Kapitel 16: Erste Algorithmen in Graphen

Thomas Worsch

KIT, Institut für Theoretische Informatik

Wintersemester 2015/2016

Überblick

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen
- 2-Erreichbarkeit an einem Beispiel
- Matrizenmultiplikation und Matrizenaddition

Einfache Berechnung der Erreichbarkeitsrelation
- Potenzen der Adjazenzmatrix
- Erste Möglichkeit für die Berechnung der Wegematrix
- Zählen arithmetischer Operationen
- Schnellere Berechnungen der Wegematrix

Algorithmus von Warshall

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

Einfache Berechnung der Erreichbarkeitsrelation

Algorithmus von Warshall

Objekte im Rechner — Knoten, Kanten und Graphen

```
class Vertex {
   String name;
}
```

«Knoteninhalte» für uns irrelevant

```
class Edge {
   Vertex start;
   Vertex end;
}

class Graph {
   Vertex[] vertices;
   Edge[] edges;
}
```

`Vertex[] vertices;` mehr als eine Menge

`Edge[] edges;` mehr als eine Menge

Objekte im Rechner — Knoten, Kanten und Graphen

```
class Vertex {
   String name;
}

class Edge {
   Vertex start;
   Vertex end;
}

class Graph {
   Vertex[] vertices;
   Edge[] edges;
}
```

- `String name;` — «Knoteninhalte» für uns irrelevant
- `Vertex[] vertices;` — mehr als eine Menge
- `Edge[] edges;` — mehr als eine Menge

Objekte im Rechner — Knoten, Kanten und Graphen

```
class Vertex {
   String name;
}
```

«Knoteninhalte» für uns irrelevant

```
class Edge {
   Vertex start;
   Vertex end;
}
```

```
class Graph {
   Vertex[] vertices;
   Edge[] edges;
}
```

`Vertex[] vertices;` mehr als eine Menge

`Edge[] edges;` mehr als eine Menge

Objekte im Rechner — Knoten, Kanten und Graphen

```
class Vertex {
   int id;
}
```

Annahme: Wertebereich $\mathbb{Z}_{|V|}$

```
class Edge {
   Vertex start;
   Vertex end;
}

class Graph {
   Vertex[] vertices;
   Edge[] edges;
}
```

`Vertex[] vertices;` mehr als eine Menge

`Edge[] edges;` mehr als eine Menge

Adjazenzlisten — eine andere Repräsentation für Graphen

```
class Vertex {
   int id;
   Vertex[] neighbors;
}

class Edge {
   Vertex start;
   Vertex end;
}

class Graph {
   Vertex[] vertices;
   Edge[] edges;
}
```

Feldlänge = Knotengrad

Inzidenzlisten

```
class Vertex {
   int id;
   Edge[] incoming;
   Edge[] outgoing;
}
```

Feldlänge = Eingangsknotengrad

Feldlänge = Ausgangsknotengrad

```
class Edge {
   Vertex start;
   Vertex end;
}

class Graph {
   Vertex[] vertices;
   Edge[] edges;
}
```

Variante von Adjazenzlisten (auf dem Weg zu Matrizen)

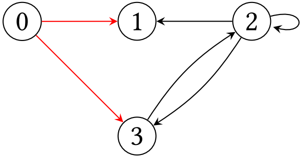
```
class Vertex {
   int id;
   boolean[] is_connected_to;
}
```

Feldlänge = $|V|$

$$u.\texttt{is\_connected\_to}[v.\mathrm{id}] = \begin{cases} \texttt{true} & \text{falls } (u, v) \in E \\ \texttt{false} & \text{falls } (u, v) \notin E \end{cases}$$

```
class Graph {
   Vertex[] vertices;
}
```

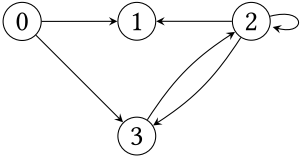
Variante von Adjazenzlisten für einen Beispielgraph



Objekt u für Knoten 0

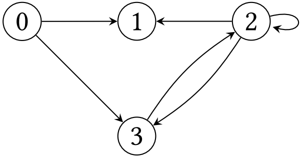
u.id	u.is_connected_to			
	0	1	2	3
0	false	true	false	true

Variante von Adjazenzlisten für einen Beispielgraph



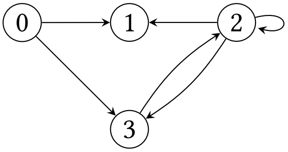
u.id	u.is_connected_to 0	1	2	3
0	false	true	false	true
1	false	false	false	false
2	false	true	true	true
3	false	false	true	false

Variante von Adjazenzlisten für einen Beispielgraph



u.id	u.is_connected_to 0	1	2	3
0	false	true	false	true
1	false	false	false	false
2	false	true	true	true
3	false	false	true	false

Adjazenzmatrix eines gerichteten Graphen



$G = (V, E)$ gerichtet mit n Knoten
$n \times n$-Matrix A mit

$$A_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in E \\ 0 & \text{falls } (i,j) \notin E \end{cases}$$

ungerichtetes $U = (V, E) \rightsquigarrow G = (V, E_g)$

Beispiel

$$\begin{array}{c c} & \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

Repräsentation von Relationen durch Matrizen

endliche Menge M mit n Elementen

binäre Relation $R \subseteq M \times M$

repräsentiert durch $n \times n$-Matrix $A(R)$:

$$(A(R))_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in R \quad \text{d. h. also } iRj \\ 0 & \text{falls } (i,j) \notin R \quad \text{d. h. also } \neg(iRj) \end{cases}$$

verschiedene Relationen $\leftrightarrow$ verschiedene Matrizen

Wegematrix eines Graphen

Erreichbarkeitsrelation E^* als Matrix W

$$W_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in E^* \\ 0 & \text{falls } (i,j) \notin E^* \end{cases}$$

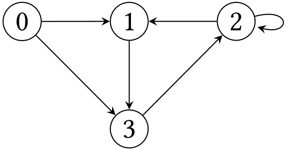
$$= \begin{cases} 1 & \text{falls es in } G \text{ einen Pfad von } i \text{ nach } j \text{ gibt} \\ 0 & \text{falls es in } G \text{ keinen Pfad von } i \text{ nach } j \text{ gibt} \end{cases}$$

algorithmisches Problem:

- gegebene Probleminstanz: Adjazenzmatrix eines Graphen
- gesucht: zugehörige Wegematrix des Graphen

Wegematrix — ein Beispiel

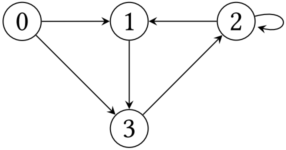
$$W_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in E^* \\ 0 & \text{falls } (i,j) \notin E^* \end{cases}$$



$$A = \begin{array}{c c} & \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

Wegematrix — ein Beispiel

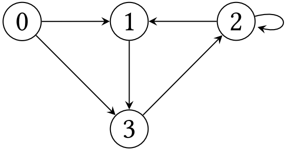
$$W_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in E^* \\ 0 & \text{falls } (i,j) \notin E^* \end{cases}$$



$$A = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} \qquad W = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{pmatrix} & & & \\ & & & \\ & & & \\ & & & \end{pmatrix} \end{array}$$

Wegematrix — ein Beispiel

$$W_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in E^* \\ 0 & \text{falls } (i,j) \notin E^* \end{cases}$$

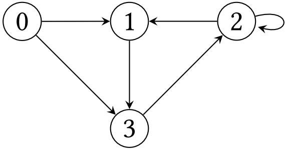


$$A = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

$$W = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{pmatrix} \end{array}$$

Wegematrix — ein Beispiel

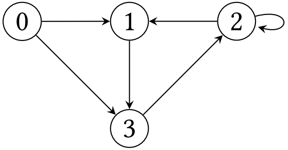
$$W_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in E^* \\ 0 & \text{falls } (i,j) \notin E^* \end{cases}$$



$$A = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} \qquad W = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 1 \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{pmatrix} \end{array}$$

Wegematrix — ein Beispiel

$$W_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in E^* \\ 0 & \text{falls } (i,j) \notin E^* \end{cases}$$

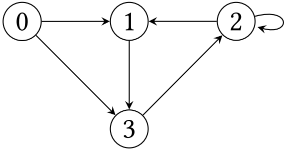


$$A = \begin{array}{c|c} & \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

$$W = \begin{array}{c|c} & \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & & \\ 0 & & 1 & \\ 0 & & & 1 \end{pmatrix} \end{array}$$

Wegematrix — ein Beispiel

$$W_{ij} = \begin{cases} 1 & \text{falls } (i,j) \in E^* \\ 0 & \text{falls } (i,j) \notin E^* \end{cases}$$



$$A = \begin{array}{c c} & \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} \qquad W = \begin{array}{c c} & \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix} \end{array}$$

Was ist wichtig

Das sollten Sie mitnehmen:

- Repräsentation von Relationen als Matrizen
- z. B. Kantenrelation eines Graphen: Adjazenzmatrix

Das sollten Sie üben:

- zu gegebenem Graphen die Adjazenzmatrix hinschreiben
- zu gegebener Adjazenzmatrix den Graphen hinmalen
- z. B. für irgendwelche „speziellen“ Graphen und Matrizen

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

- 2-Erreichbarkeit an einem Beispiel
- Matrizenmultiplikation und Matrizenaddition

Einfache Berechnung der Erreichbarkeitsrelation

Algorithmus von Warshall

Wo sind wir?

Repräsentation von Graphen im Rechner

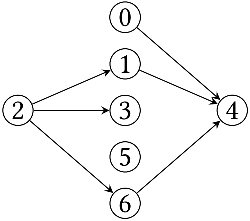
Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

- 2-Erreichbarkeit an einem Beispiel
- Matrizenmultiplikation und Matrizenaddition

Einfache Berechnung der Erreichbarkeitsrelation

Algorithmus von Warshall

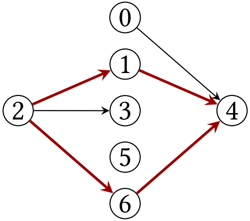
2-Erreichbarkeit an einem Beispiel



$$\begin{array}{c c} & \begin{array}{ccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} & \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

gesucht: Pfade der Länge 2
von Knoten 2 zu Knoten 4

2-Erreichbarkeit an einem Beispiel



$$\begin{array}{c|ccccccc} & 0 & 1 & 2 & 3 & 4 & 5 & 6 \\ \hline 0 & \multicolumn{7}{l}{} \end{array}$$

$$\begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} \begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

gesucht: Pfade der Länge 2
von Knoten 2 zu Knoten 4

hinsehen: $(2, 1, 4)$ und $(2, 6, 4)$

Systematische Suche nach Pfaden im Beispiel

alle Pfade?

prüfe für *alle* Knoten $k \in V$:

- Ist $(2, k, 4)$ ein Pfad?

Systematische Suche nach Pfaden im Beispiel

alle Pfade?

prüfe für *alle* Knoten $k \in V$:

- Ist $(2, k, 4)$ ein Pfad?
- Ist $(2, k) \in E$ und $(k, 4) \in E$?

Systematische Suche nach Pfaden im Beispiel

alle Pfade?

prüfe für *alle* Knoten $k \in V$:

- Ist $(2, k, 4)$ ein Pfad?
- Ist $(2, k) \in E$ und $(k, 4) \in E$?
- Ist $A_{2k} = 1$ und $A_{k4} = 1$?

Systematische Suche nach Pfaden im Beispiel

alle Pfade?

prüfe für *alle* Knoten $k \in V$:

- Ist $(2, k, 4)$ ein Pfad?
- Ist $(2, k) \in E$ und $(k, 4) \in E$?
- Ist $A_{2k} = 1$ und $A_{k4} = 1$?
- Ist $A_{2k} \cdot A_{k4} = 1$?

Systematische Suche nach Pfaden im Beispiel

alle Pfade?

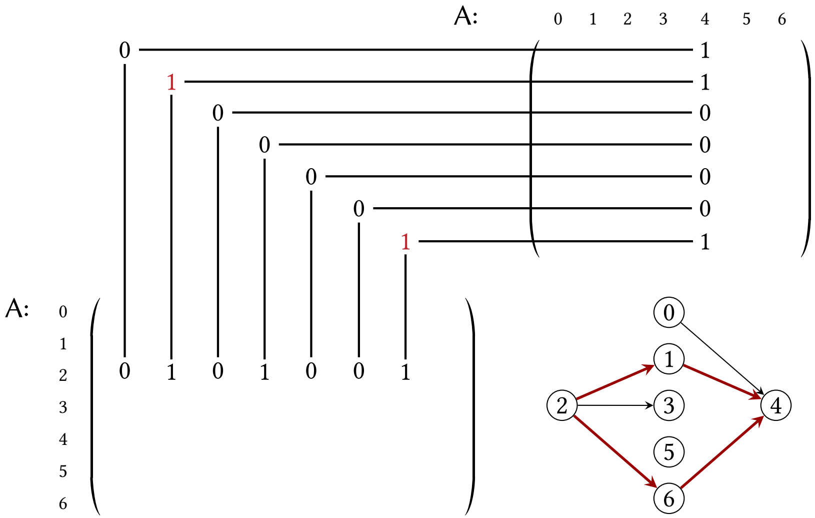
prüfe für *alle* Knoten $k \in V$:

- Ist $(2, k, 4)$ ein Pfad?
- Ist $(2, k) \in E$ und $(k, 4) \in E$?
- Ist $A_{2k} = 1$ und $A_{k4} = 1$?
- Ist $A_{2k} \cdot A_{k4} = 1$?

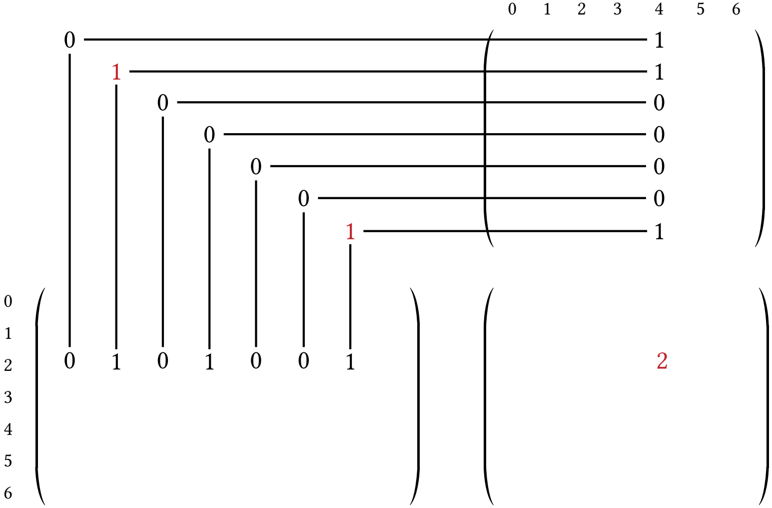
durchlaufe sequenziell für alle k

- gleichzeitig alle A_{2k} und alle A_{k4}, d. h.
- *Zeile* für Knoten 2 und *Spalte* für Knoten 4

Systematische Suche nach Pfaden



Zählen der Pfade im Beispiel



Zählen der Pfade im Beispiel (2)

$$P_{24} = \sum_{k=0}^{6} A_{2k} \cdot A_{k4}$$

$$\begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{pmatrix} & & & & 1 & & \\ & & & & 1 & & \\ & & & & 0 & & \\ & & & & 0 & & \\ & & & & 0 & & \\ & & & & 0 & & \\ & & & & 1 & & \end{pmatrix} \end{array}$$

$$\begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} \begin{pmatrix} & & & & & & \\ & & & & & & \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \end{pmatrix} \quad \begin{pmatrix} & & & & & & \\ & & & & & & \\ & & & & 2 & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \end{pmatrix}$$

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

- 2-Erreichbarkeit an einem Beispiel
- Matrizenmultiplikation und Matrizenaddition

Einfache Berechnung der Erreichbarkeitsrelation

Algorithmus von Warshall

Matrizenmultiplikation

es sei

- A eine $\ell \times n$-Matrix
- B eine $n \times m$-Matrix

Produkt $C = A \cdot B$ die $\ell \times m$-Matrix mit

$$C_{ij} = \sum_{k=0}^{n-1} A_{ik} \cdot B_{kj}$$

im Allgemeinen $A \cdot B \neq B \cdot A$

- falls überhaupt beides definiert, also $\ell = m$

Algorithmus für Matrizenmultiplikation

zunächst die naheliegende Möglichkeit

- demnächst: es geht auch anders!

ℓ Zeilen

m Spalten

$$C_{ij} = \sum_{k=0}^{n-1} A_{ik} \cdot B_{kj}$$

$\textbf{for}\ i \leftarrow 0\ \textbf{to}\ \ell - 1\ \textbf{do}$
 $\textbf{for}\ j \leftarrow 0\ \textbf{to}\ m - 1\ \textbf{do}$
  $C_{ij} \leftarrow 0$
  $\textbf{for}\ k \leftarrow 0\ \textbf{to}\ n - 1\ \textbf{do}$
   $C_{ij} \leftarrow C_{ij} + A_{ik} \cdot B_{kj}$
  $\textbf{od}$
 $\textbf{od}$
$\textbf{od}$

Einheitsmatrizen

$$\begin{matrix} 1 & 0 & & \cdots & & 0 \\ 0 & 1 & 0 & & & \\ & 0 & 1 & 0 & & \vdots \\ \vdots & & 0 & 1 & 0 & \\ & & & 0 & 1 & 0 \\ 0 & & \cdots & & 0 & 1 \end{matrix}$$

Einheitsmatrix I quadratische Matrix mit

$$\mathrm{I}_{ij} = \begin{cases} 1 & \text{falls } i = j \\ 0 & \text{falls } i \neq j \end{cases}$$

für jede $m \times n$-Matrix A gilt

$$\mathrm{I} \cdot A = A = A \cdot \mathrm{I}$$

- linke Einheitsmatrix $m \times m$
- rechte Einheitsmatrix $n \times n$

Potenzen quadratischer Matrizen

$$A^0 = \mathsf{I}$$
$$\forall n \in \mathbb{N}_0 : A^{n+1} = A^n \cdot A$$

Quadrierte Adjazenzmatrix

A^2 einer Adjazenzmatrix A

$$(A^2)_{ij} = \sum_{k=0}^{n-1} A_{ik} A_{kj}$$

$A_{ik} A_{kj} = 1$
$\iff A_{ik} = A_{kj} = 1$
$\iff$ Kanten von i nach k und von k nach j existieren
$\iff (i, k, j)$ ein Pfad der Länge 2 von i nach j ist.

und 0 sonst.

für $k_1 \neq k_2$ sind (i, k_1, j) und (i, k_2, j) verschieden

$(A^2)_{ij}$ ist Anzahl der Pfade der Länge 2 von i nach j

Matrizenaddition

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} + \begin{pmatrix} 10 & 20 \\ 30 & 40 \end{pmatrix}$$

$$= \begin{pmatrix} 11 & 22 \\ 33 & 44 \end{pmatrix}$$

A und B zwei $m \times n$-Matrizen

Summe $C = A + B$ ist $m \times n$-Matrix mit

$$C_{ij} = A_{ij} + B_{ij}$$

stets $A + B = B + A$

neutrales Element: Nullmatrix 0, überall Nullen

algorithmisch:

for $i \leftarrow 0$ **to** $m - 1$ **do**
  for $j \leftarrow 0$ **to** $n - 1$ **do**
    $C_{ij} \leftarrow A_{ij} + B_{ij}$
  od
od

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

Einfache Berechnung der Erreichbarkeitsrelation
- Potenzen der Adjazenzmatrix
- Erste Möglichkeit für die Berechnung der Wegematrix
- Zählen arithmetischer Operationen
- Schnellere Berechnungen der Wegematrix

Algorithmus von Warshall

Berechnung von E^* — die naheliegende Idee

Benutze

$$E^* = \bigcup_{i \in \mathbb{N}_0} E^i$$

um die Wegematrix zu berechnen.

Berechnung von E^* — die naheliegende Idee

Benutze

$$E^* = \bigcup_{i \in \mathbb{N}_0} E^i$$

um die Wegematrix zu berechnen.

Probleme:

- Was kann man gegen das unendlich tun?

Berechnung von E^* — die naheliegende Idee

Benutze

$$E^* = \bigcup_{i \in \mathbb{N}_0} E^i$$

um die Wegematrix zu berechnen.

Probleme:

- Was kann man gegen das unendlich tun?
- Woher kommen die Matrizen für die Relationen E^i?

Berechnung von E^* — die naheliegende Idee

Benutze

$$E^* = \bigcup_{i \in \mathbb{N}_0} E^i$$

um die Wegematrix zu berechnen.

Probleme:

- Was kann man gegen das unendlich tun?
- Woher kommen die Matrizen für die Relationen E^i?
- Welcher Matrizen-Operation entspricht die Vereinigung?

Beseitigung der *unendlichen* Vereinigung

Existiert Pfad in G von Knoten i nach Knoten j ?

Sei

- $G = (V, E)$ mit $|V| = n$
  - nur *endlich* viele Knoten
- $p = (i_0, i_1, \dots, i_k)$ ein Pfad mit $i_0 = i$ und $i_k = j$.

Beseitigung der *unendlichen* Vereinigung

Existiert Pfad in G von Knoten i nach Knoten j ?

Sei

- $G = (V, E)$ mit $|V| = n$
  - nur *endlich* viele Knoten
- $p = (i_0, i_1, \ldots, i_k)$ ein Pfad mit $i_0 = i$ und $i_k = j$.

wenn $k \geq n$, dann

- enthält p Zyklus von x nach x
- Weglassen ergibt kürzeren Pfad von i nach j

Beseitigung der *unendlichen* Vereinigung

Existiert Pfad in G von Knoten i nach Knoten j ?

Sei

- $G = (V, E)$ mit $|V| = n$
  - nur *endlich* viele Knoten
- $p = (i_0, i_1, \ldots, i_k)$ ein Pfad mit $i_0 = i$ und $i_k = j$.

wenn $k \geq n$, dann

- enthält p Zyklus von x nach x
- Weglassen ergibt kürzeren Pfad von i nach j

wiederhole, solange Pfadlänge $\geq n$

Ergebnis: Pfad mit Länge $\leq n - 1$ von i nach j

Beseitigung der *unendlichen* Vereinigung (2)

eben begründet

Lemma

Für jeden gerichteten Graphen $G = (V, E)$ mit n Knoten gilt:

$$E^* = \bigcup_{i \in \mathbb{Z}_n} E^i$$

Beseitigung der *unendlichen* Vereinigung (2)

eben begründet

Lemma

Für jeden gerichteten Graphen $G = (V, E)$ *mit* n *Knoten gilt:*

$$E^* = \bigcup_{i \in \mathbb{Z}_n} E^i$$

längere Pfade schaden nicht

Lemma

Für jeden gerichteten Graphen $G = (V, E)$ *mit* n *Knoten gilt:*

$$\forall k \geq n-1 : E^* = \bigcup_{i \in \mathbb{Z}_k} E^i$$

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

Einfache Berechnung der Erreichbarkeitsrelation
- Potenzen der Adjazenzmatrix
- Erste Möglichkeit für die Berechnung der Wegematrix
- Zählen arithmetischer Operationen
- Schnellere Berechnungen der Wegematrix

Algorithmus von Warshall

Potenzen der Adjazenzmatrix haben eine Bedeutung

Lemma

Es sei G ein gerichteter Graph mit Adjazenzmatrix A.
Für alle $k \in \mathbb{N}_0$ gilt:

$(A^k)_{ij}$ *ist die Anzahl der Pfade der Länge k in G von i nach j.*

- Beweis durch vollständige Induktion.
- Induktionsschritt fast wie im Fall $k = 2$.

Signum-Funktion

$$\mathrm{sgn} : \mathbb{R} \to \mathbb{R} : \mathrm{sgn}(x) = \begin{cases} 1 & \text{falls } x > 0 \\ 0 & \text{falls } x = 0 \\ -1 & \text{falls } x < 0 \end{cases}$$

bei Matrizen komponentenweise

$$\mathrm{sgn} : \mathbb{R}^{m \times n} \to \mathbb{R}^{m \times n} \quad \text{mit} \quad (\mathrm{sgn}(M))_{ij} = \mathrm{sgn}(M_{ij})$$

Matrizendarstellung für E^k — sgn(A^k) tut es

Korollar

Es sei G ein gerichteter Graph mit Adjazenzmatrix A.
Für alle $k \in \mathbb{N}_0$ gilt:

1. $$\operatorname{sgn}((A^k)_{ij}) = \begin{cases} 1 & \text{falls in } G \text{ ein Pfad der Länge } k \\ & \quad \text{von } i \text{ nach } j \text{ existiert} \\ 0 & \text{falls in } G \text{ kein Pfad der Länge } k \\ & \quad \text{von } i \text{ nach } j \text{ existiert} \end{cases}$$
2. Matrix $\operatorname{sgn}(A^k)$ repräsentiert die Relation E^k.

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

Einfache Berechnung der Erreichbarkeitsrelation

- Potenzen der Adjazenzmatrix
- Erste Möglichkeit für die Berechnung der Wegematrix
- Zählen arithmetischer Operationen
- Schnellere Berechnungen der Wegematrix

Algorithmus von Warshall

Vereinigung von Relationen

Relationen $R \subseteq M \times M$ und $R' \subseteq M \times M$ repräsentiert durch Matrizen A und A'.

dann:

$$\begin{aligned}
(i,j) \in R \cup R' &\leftrightarrow (i,j) \in R \vee (i,j) \in R' \\
&\leftrightarrow A_{ij} = 1 \vee A'_{ij} = 1 \\
&\leftrightarrow A_{ij} + A'_{ij} \geq 1 \\
&\leftrightarrow (A + A')_{ij} \geq 1 \\
&\leftrightarrow \operatorname{sgn}(A + A')_{ij} = 1
\end{aligned}$$

$R \cup R'$ wird durch $\operatorname{sgn}(A + A')$ repräsentiert

Eine erste Formel für die Wegematrix — es gibt auch noch andere …

Lemma

Es sei G ein gerichteter Graph mit Adjazenzmatrix A. Dann gilt für alle $k \geq n-1$:

- *Die Matrix $sgn(\sum_{i=0}^{k} A^i)$ repräsentiert die Relation E^*.*
- *Mit anderen Worten:*

$$W = sgn\left(\sum_{i=0}^{k} A^i\right)$$

ist die Wegematrix des Graphen G.

Beweis

noch zu überlegen

$\bigcup_{i=0}^{n-1} E^i$ durch $\mathrm{sgn}(\sum_{i=0}^{k} \mathrm{sgn}(A^i))$ repräsentiert

- leichte Verallgemeinerung des Falles $R \cup R'$

„innere“ Anwendungen von sgn dürfen fehlen

- Wenn alle Matrixeinträge ≥ 0 sind, gilt:

$$\mathrm{sgn}(\mathrm{sgn}(M) + \mathrm{sgn}(M'))_{ij} = \mathrm{sgn}(M + M')_{ij}$$

Einfachster Algorithmus für die Wegematrix

$\langle$*Matrix A sei die Adjazenzmatrix*$\rangle$
$W \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $M \leftarrow \mathbf{I}$
  for $j \leftarrow 1$ **to** i **do**
    $M \leftarrow M \cdot A$
  od

  $W \leftarrow W + M$

od
$W \leftarrow \mathrm{sgn}(W)$

Einfachster Algorithmus für die Wegematrix

$\langle$*Matrix A sei die Adjazenzmatrix*$\rangle$
$W \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $M \leftarrow \mathsf{I}$
  for $j \leftarrow 1$ **to** i **do**
    $M \leftarrow M \cdot A$
  od
  $\{ M = A^i \}$
  $W \leftarrow W + M$
  $\{ W = \sum_{k=0}^{i} A^k \}$
od
$W \leftarrow \operatorname{sgn}(W)$
$\{$ W ist die Wegematrix $\}$

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

Einfache Berechnung der Erreichbarkeitsrelation

- Potenzen der Adjazenzmatrix
- Erste Möglichkeit für die Berechnung der Wegematrix
- Zählen arithmetischer Operationen
- Schnellere Berechnungen der Wegematrix

Algorithmus von Warshall

Was ist der «Aufwand» eines Algorithmus?

Anzahl Codezeilen?

Entwicklungszeit?

Anzahl Schritte?

- nicht immer gleich

benötigter Speicherplatz?

- nicht immer gleich

vorläufig: Anzahl arithmetischer Operationen

- später anders

Wieviele elementare Operationen für Matrizenaddition?

for $i \leftarrow 0$ **to** $m-1$ **do**
  for $j \leftarrow 0$ **to** $n-1$ **do**
    $C_{ij} \leftarrow A_{ij} + B_{ij}$
  od
od

$m \cdot n$ Additionen

für $n \times n$-Matrizen: n^2

Wieviele elementare Operationen für Multiplikation?

for $i \leftarrow 0$ **to** $\ell - 1$ **do**
  for $j \leftarrow 0$ **to** $m - 1$ **do**
    $C_{ij} \leftarrow 0$
    for $k \leftarrow 0$ **to** $n - 1$ **do**
      $C_{ij} \leftarrow C_{ij} + A_{ik} \cdot B_{kj}$
    od
  od
od

$\ell \cdot m \cdot n$ Additionen und
$\ell \cdot m \cdot n$ Multiplikationen

für $n \times n$-Matrizen: $2n^3$
kleine Variante: $2n^3 - n^2$

Achtung: es geht auch mit weniger Operationen!

Wieviele elementare Operationen für Wegematrix?

$W \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $M \leftarrow \mathrm{I}$
  for $j \leftarrow 1$ **to** i **do**
    $M \leftarrow M \cdot A$
  od
  $W \leftarrow W + M$
od
$W \leftarrow \operatorname{sgn}(W)$

Aufwand:

$$\left(\sum_{i=0}^{n-1} i\right) \cdot (2n^3 - n^2) + n \cdot n^2 + n^2$$

$$= n^5 - \frac{3}{2}n^4 + \frac{3}{2}n^3 + n^2$$

Wieviele elementare Operationen für Wegematrix?

$W \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $M \leftarrow \mathrm{I}$
  for $j \leftarrow 1$ **to** i **do**
    $M \leftarrow M \cdot A$
  od
  $W \leftarrow W + M$
od
$W \leftarrow \mathrm{sgn}(W)$

Aufwand:

$$\left(\sum_{i=0}^{n-1} i\right) \cdot (2n^3 - n^2) + n \cdot n^2 + n^2$$

$$= n^5 - \frac{3}{2}n^4 + \frac{3}{2}n^3 + n^2$$

Wieviele elementare Operationen für Wegematrix?

$W \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $M \leftarrow I$
  for $j \leftarrow 1$ **to** i **do**
    $M \leftarrow M \cdot A$
  od
  $W \leftarrow W + M$
od
$W \leftarrow \operatorname{sgn}(W)$

Aufwand:

$$\left(\sum_{i=0}^{n-1} i\right) \cdot (2n^3 - n^2) + n \cdot n^2 + n^2$$
$$= n^5 - \frac{3}{2}n^4 + \frac{3}{2}n^3 + n^2$$

Wieviele elementare Operationen für Wegematrix?

$W \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $M \leftarrow \mathrm{I}$
  for $j \leftarrow 1$ **to** i **do**
    $M \leftarrow M \cdot A$
  od
  $W \leftarrow W + M$
od
$W \leftarrow \mathrm{sgn}(W)$

Aufwand:

$$\left(\sum_{i=0}^{n-1} i\right) \cdot (2n^3 - n^2) + n \cdot n^2 + n^2$$

$$= n^5 - \frac{3}{2}n^4 + \frac{3}{2}n^3 + n^2$$

Wieviele elementare Operationen für Wegematrix?

$W \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $M \leftarrow \mathrm{I}$
  for $j \leftarrow 1$ **to** i **do**
    $M \leftarrow M \cdot A$
  od
  $W \leftarrow W + M$
od
$W \leftarrow \operatorname{sgn}(W)$

Aufwand:

$$\left(\sum_{i=0}^{n-1} i\right) \cdot (2n^3 - n^2) + n \cdot n^2 + n^2$$

$$= n^5 - \frac{3}{2}n^4 + \frac{3}{2}n^3 + n^2$$

Wieviele elementare Operationen für Wegematrix?

$W \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $M \leftarrow I$
  for $j \leftarrow 1$ **to** i **do**
    $M \leftarrow M \cdot A$
  od
  $W \leftarrow W + M$
od
$W \leftarrow \operatorname{sgn}(W)$

Aufwand:

$$\left(\sum_{i=0}^{n-1} i\right) \cdot (2n^3 - n^2) + n \cdot n^2 + n^2$$
$$= n^5 - \frac{3}{2}n^4 + \frac{3}{2}n^3 + n^2$$

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

Einfache Berechnung der Erreichbarkeitsrelation
- Potenzen der Adjazenzmatrix
- Erste Möglichkeit für die Berechnung der Wegematrix
- Zählen arithmetischer Operationen
- Schnellere Berechnungen der Wegematrix

Algorithmus von Warshall

Wiederverwendung —
auch bei Zwischenergebnissen eine gute Sache

$W \leftarrow 0$
$M \leftarrow \mathsf{I}$
for $i \leftarrow 0$ **to** $n-1$ **do**
  $W \leftarrow W + M$
  $M \leftarrow M \cdot A$
od
$W \leftarrow \operatorname{sgn}(W)$

für A^i kann man A^{i-1} wiederverwenden

Aufwand:

$$n \cdot (n^2 + (2n^3 - n^2)) + n^2 = 2n^4 + n^2$$

Es geht noch besser —
erst mehr denken und dann weniger rechnen

Schon vergessen?

$$\forall k \geq n-1 : E^* = \bigcup_{i=0}^{k} E^i$$

Wie kann ein $k > n-1$ helfen?

Es geht noch besser — erst mehr denken und dann weniger rechnen

Schon vergessen?

$$\forall k \geq n-1 : E^* = \bigcup_{i=0}^{k} E^i$$

Wie kann ein $k > n-1$ helfen?

- statt $n-1$ wähle kleinste Zweierpotenz $k = 2^m \geq n$
- also $m = \lceil \log_2 n \rceil$
- finde Matrix F mit $W = F^{2^m} = (\cdots((F^2)^2)\cdots)^2$
- nur noch $m = \lceil \log_2 n \rceil$ Matrizenmultiplikationen

Es geht noch besser — erst mehr denken und dann weniger rechnen

Schon vergessen?

$$\forall k \geq n-1 : E^* = \bigcup_{i=0}^{k} E^i$$

Wie kann ein $k > n-1$ helfen?

- statt $n-1$ wähle kleinste Zweierpotenz $k = 2^m \geq n$
- also $m = \lceil \log_2 n \rceil$
- finde Matrix F mit $W = F^{2^m} = (\cdots((F^2)^2)\cdots)^2$
- nur noch $m = \lceil \log_2 n \rceil$ Matrizenmultiplikationen

Preisfrage: Wie sieht F aus?

Es geht noch besser — erst mehr denken und dann weniger rechnen

Schon vergessen?

$$\forall k \geq n-1 : E^* = \bigcup_{i=0}^{k} E^i$$

Wie kann ein $k > n-1$ helfen?

- statt $n-1$ wähle kleinste Zweierpotenz $k = 2^m \geq n$
- also $m = \lceil \log_2 n \rceil$
- finde Matrix F mit $W = F^{2^m} = (\cdots((F^2)^2)\cdots)^2$
- nur noch $m = \lceil \log_2 n \rceil$ Matrizenmultiplikationen

Preisfrage: Wie sieht F aus?

Wähle Matrix F für Relation $E^0 \cup E^1 = \mathrm{I}_V \cup E$.

Es geht noch besser (2)

$$
\begin{aligned}
F &= E^0 \cup E^1 \\
F^2 &= (E^0 \cup E^1) \circ (E^0 \cup E^1) = E^0 \cup E^1 \cup E^1 \cup E^2 \\
&= E^0 \cup E^1 \cup E^2 \\
F^4 &= (F^2)^2 = (E^0 \cup E^1 \cup E^2) \circ (E^0 \cup E^1 \cup E^2) \\
&= \dots \\
&= E^0 \cup E^1 \cup E^2 \cup E^3 \cup E^4
\end{aligned}
$$

Induktion: Für alle $m \in \mathbb{N}_0$ gilt:

$$F^{2^m} = \bigcup_{i=0}^{2^m} E^i$$

Es geht noch besser (3)

$F \leftarrow \mathsf{I} + A$
$m \leftarrow \lceil \log_2 n \rceil$
for $i \leftarrow 1$ **to** m **do**
  $F \leftarrow F \cdot F$
od
$W \leftarrow \operatorname{sgn}(F)$

Berechnung von $\lceil \log_2 n \rceil$ aus n
benötigt höchstens $\lceil \log_2 n \rceil$ Operationen

Aufwand:

$$n^2 + \lceil \log_2 n \rceil + \lceil \log_2 n \rceil \cdot (2n^3 - n^2) + n^2$$
$$= \lceil \log_2 n \rceil \cdot 2n^3 + \cdots$$

- «besser» als $2n^4$
- nächstes Kapitel: präzise ungenau sein

Was ist wichtig

Das sollten Sie mitnehmen:

- die naheliegende Idee nicht immer die beste
- Denken, Mathematik, Kreativität, Einfach-mal-drüber-schlafen, …

Das sollten Sie üben:

- Aufwandsabschätzungen bei (ineinander geschachtelten) Schleifen
- auch mal verrückte Ideen ausprobieren

Wo sind wir?

Repräsentation von Graphen im Rechner

Berechnung der 2-Erreichbarkeitsrelation und Rechnen mit Matrizen

Einfache Berechnung der Erreichbarkeitsrelation

Algorithmus von Warshall

Algorithmus von Warshall — noch ein bisschen schneller

for $i \leftarrow 0$ **to** $n-1$ **do**
  for $j \leftarrow 0$ **to** $n-1$ **do**

$$W_{ij} \leftarrow \begin{cases} 1 & \text{falls } i = j \\ A_{ij} & \text{falls } i \neq j \end{cases}$$

  od
od

for $k \leftarrow 0$ **to** $n-1$ **do**
  for $i \leftarrow 0$ **to** $n-1$ **do**
    for $j \leftarrow 0$ **to** $n-1$ **do**
      $W_{ij} \leftarrow \max(W_{ij}, W_{ik} \cdot W_{kj})$
    od
  od
od

Algorithmus von Warshall — noch ein bisschen schneller

for $i \leftarrow 0$ **to** $n-1$ **do**
  for $j \leftarrow 0$ **to** $n-1$ **do**

$$W_{ij}^{[0]} \leftarrow \begin{cases} 1 & \text{falls } i = j \\ A_{ij} & \text{falls } i \neq j \end{cases}$$

  od
od

for $k \leftarrow 0$ **to** $n-1$ **do**
  for $i \leftarrow 0$ **to** $n-1$ **do**
    for $j \leftarrow 0$ **to** $n-1$ **do**

$$W_{ij}^{[k+1]} \leftarrow \max(W_{ij}^{[k]}, W_{ik}^{[k]} \cdot W_{kj}^{[k]})$$

    od
  od
od

Algorithmus von Warshall — noch ein bisschen schneller

$$
\begin{array}{l}
\textbf{for}\ \ i \leftarrow 0\ \ \textbf{to}\ \ n-1\ \ \textbf{do} \\
\quad \textbf{for}\ \ j \leftarrow 0\ \ \textbf{to}\ \ n-1\ \ \textbf{do} \\
\quad\quad W_{ij}^{[0]} \leftarrow \begin{cases} 1 & \text{falls } i = j \\ A_{ij} & \text{falls } i \neq j \end{cases} \\
\quad \textbf{od} \\
\textbf{od}
\end{array}
$$

$$
\begin{array}{l}
\textbf{for}\ \ k \leftarrow 0\ \ \textbf{to}\ \ n-1\ \ \textbf{do} \\
\quad \textbf{for}\ \ i \leftarrow 0\ \ \textbf{to}\ \ n-1\ \ \textbf{do} \\
\quad\quad \textbf{for}\ \ j \leftarrow 0\ \ \textbf{to}\ \ n-1\ \ \textbf{do} \\
\quad\quad\quad W_{ij}^{[k+1]} \leftarrow \max(\,W_{ij}^{[k]},\ W_{ik}^{[k]} \cdot W_{kj}^{[k]}\,) \\
\quad\quad \textbf{od} \\
\quad \textbf{od} \\
\textbf{od}
\end{array}
$$

Pfad $p = (v_0, v_1, \ldots, v_{m-1}, v_m)$ der Länge m

- $v_1, \ldots, v_{m-1}$ Zwischenknoten
- $m \leq 1$: keine Zwischenknoten

Algorithmus von Warshall — noch ein bisschen schneller

$$
\begin{array}{l}
\textbf{for } i \leftarrow 0 \textbf{ to } n-1 \textbf{ do} \\
\quad \textbf{for } j \leftarrow 0 \textbf{ to } n-1 \textbf{ do} \\
\qquad W_{ij}^{[0]} \leftarrow \begin{cases} 1 & \text{falls } i = j \\ A_{ij} & \text{falls } i \neq j \end{cases} \\
\quad \textbf{od} \\
\textbf{od}
\end{array}
$$

$$
\begin{array}{l}
\textbf{for } k \leftarrow 0 \textbf{ to } n-1 \textbf{ do} \\
\quad \textbf{for } i \leftarrow 0 \textbf{ to } n-1 \textbf{ do} \\
\qquad \textbf{for } j \leftarrow 0 \textbf{ to } n-1 \textbf{ do} \\
\qquad\quad W_{ij}^{[k+1]} \leftarrow \max(W_{ij}^{[k]}, W_{ik}^{[k]} \cdot W_{kj}^{[k]}) \\
\qquad \textbf{od} \\
\quad \textbf{od} \\
\textbf{od}
\end{array}
$$

Pfad $p = (v_0, v_1, \ldots, v_{m-1}, v_m)$ der Länge m

- $v_1, \ldots, v_{m-1}$ Zwischenknoten
- $m \leq 1$: keine Zwischenknoten

$W_{ij}^{[k]} = 1 \longleftrightarrow$

- Pfad von i nach j und
- nur Zwischenknoten mit Nummern $x < k$ benutzt

Algorithmus von Warshall — noch ein bisschen schneller

```
for i ← 0 to n−1 do
   for j ← 0 to n−1 do
      W_ij^[0] ← { 1     falls i = j
                 { A_ij  falls i ≠ j
   od
od
```

$$W_{ij}^{[0]} \leftarrow \begin{cases} 1 & \text{falls } i = j \\ A_{ij} & \text{falls } i \neq j \end{cases}$$

```
for k ← 0 to n−1 do
   for i ← 0 to n−1 do
      for j ← 0 to n−1 do
         W_ij^[k+1] ← max(W_ij^[k], W_ik^[k] · W_kj^[k])
      od
   od
od
```

$$W_{ij}^{[k+1]} \leftarrow \max(W_{ij}^{[k]}, W_{ik}^{[k]} \cdot W_{kj}^{[k]})$$

Pfad $p = (v_0, v_1, \ldots, v_{m-1}, v_m)$ der Länge m

- $v_1, \ldots, v_{m-1}$ Zwischenknoten
- $m \leq 1$: keine Zwischenknoten

$W_{ij}^{[k]} = 1 \longleftrightarrow$

- Pfad von i nach j und
- nur Zwischenknoten mit Nummern $x < k$ benutzt

Zum Aufwand des Algorithmus von Warshall

drei ineinander geschachtelte Schleifen

deren jeweiliger Rumpf n-mal durchlaufen wird

„irgendwie ungefähr" n^3 Operationen …

Zusammenfassung

Repräsentationen von Graphen im Rechner

Berechnung der Wegematrix

- mit vielen oder weniger Operationen
- Algorithmus von Warshall kommt mit weniger Operationen aus als alle unsere vorherigen Versuche