



## Graphentheoretische Definitionen

Ein Graph ist im Folgenden immer multikantenfrei und gerichtet.

Seine Knotenmenge heißt  $V$ , die Kantenmenge  $E$ .

Seine Kanten sind mit Gewichtsfunktion  $\beta$  gewichtet.

$\beta(u,v) \geq 0$  für alle  $(u,v) \in E$

$\delta(\text{Weg})$  gibt die Länge des Weges an = Summe der Kantengewichte

$d(u,v)$  = Länge des kürzesten Weges von  $u$  nach  $v$ .

## Der Dijkstra Algorithmus (Greedy - Algorithmus)

Wir konstruieren schrittweise den kürzeste - Wegebaum von einem Startknoten  $s$  aus.

0.Schritt :  $B_0 := \{s\}$ .  $d(s,s) := 0$ .

Die Menge aller kürzesten Wege von einem Knoten  $u$  hat Baumstruktur.



$i$  : Schritt : Sei  $B_i$  der kürzeste - Wege - Baum nach dem  $i$  ten Schritt.

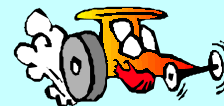
Im  $i + 1$  ten Schritt wird der Knoten  $v_{i+1} \notin B_i$  in den  $B_{i+1}$  aufgenommen, der zu  $s$  den  $i + 1$  kürzesten Abstand hat.

Die Menge  $N_i := \{v \in V \mid \exists (u,v) \in E \text{ mit } u \in B_i, v \notin B_i\}$  heißt *Nachbarschaftsliste*.

Wir suchen also den Knoten  $v_{i+1}$  der der folgenden Bedingung genügt :

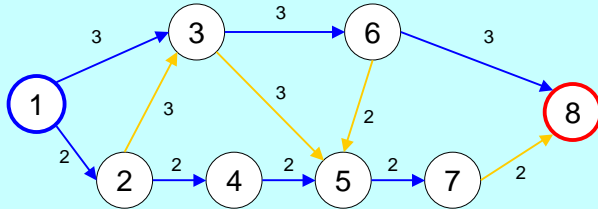
$$d(s, v_{i+1}) := \min \{d(s, u) + \beta(u, v) \mid (u, v) \in E, u \in B_i, v \in N_i\}$$

$$B_{i+1} := B_i \cup \{v_{i+1}\}$$

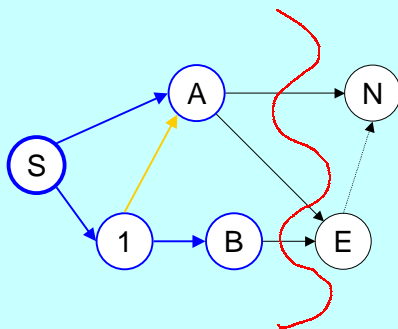


Beweis geht indirekt mit vollständiger Induktion. Der Aufwand des Algorithmus liegt bei  $O((|V| + |E|) \cdot \log |V|)$  wenn die Nachbarschaftsliste als Heap organisiert wird.

## Minimierung des Abstandes zur Wurzel



$$6+3 = d(1,6) + \beta(6,8) < d(1,7) + \beta(7,8) = 8+2$$



$$d(S,B) + \beta(B,E) + \delta(E,N) < d(S,N) \text{ (Annahme)}$$

$$= d(S,A) + \beta(A,N)$$

$$\leq d(S,B) + \beta(B,E) \text{ (Algorithmus)}$$

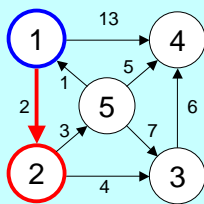
Widerspruch zu  $\delta(A,B) \geq 0$

Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

5

isi

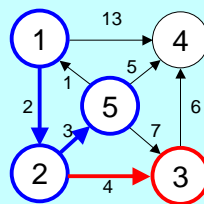
## Beispiel für die Anwendung des Dijkstraalgorithmus



$$\beta(1,2) = 2$$

$$< \beta(1,4) = 13$$

$$d(1,2) := 2$$



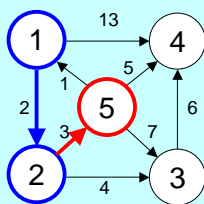
$$d(1,2) + \beta(2,3) = 6$$

$$< d(1,5) + \beta(5,4) = 10$$

$$< d(1,5) + \beta(5,3) = 12$$

$$< \beta(1,4) = 13$$

$$d(1,3) := 6$$

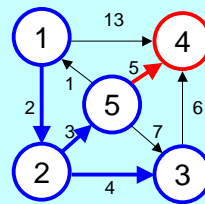


$$d(1,2) + \beta(2,5) = 5$$

$$< d(1,2) + \beta(2,3) = 6$$

$$< \beta(1,4) = 13$$

$$d(1,5) := 5$$



$$d(1,5) + \beta(5,4) = 10$$

$$< d(1,3) + \beta(3,4) = 12$$

$$< \beta(1,4) = 13$$

$$d(1,4) := 10$$

Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

6

isi

## Definition für k - kürzeste Wege

Gegeben sei ein Graph  $G := (V, E)$  sowie ein Startknoten  $s$  und ein Zielknoten  $z$ .

Ein Weg  $w$  heißt *zyklenfrei*:  $\Leftrightarrow$  kein Knoten kommt in  $w$  mehrfach vor.

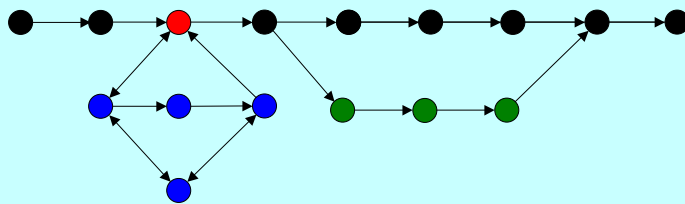
Sei  $W$  die Menge aller Wege von  $s$  nach  $z$  (dies können bis zu  $|V - 1|!$  zyklenfreie Wegesein).

Ein Weg  $w_1$  heißt *kürzester Weg* von  $s$  nach  $z$ :  $\Leftrightarrow \delta(w) \geq \delta(w_1)$  für alle  $w \in W$

(dieser ist i.a. nicht eindeutig). Sei  $w_1 \in W$  fest gewählt,

dann heißt  $w_2$  *zweitkürzester Weg* von  $s$  nach  $z$ :  $\Leftrightarrow \delta(w) \geq \delta(w_2)$  für alle  $w \in W \setminus \{w_1\}$ .

Ein Weg heißt *k - kürzester Weg* von  $s$  nach  $z$ :  $\Leftrightarrow \delta(w) \geq \delta(w_k)$  für alle  $w \in W \setminus \{w_1, \dots, w_{k-1}\}$ .

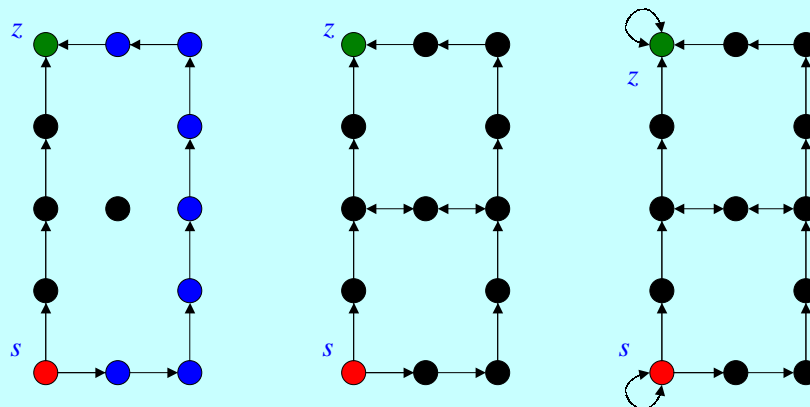


Usr/local/www/ifi/menschen/schmid/kkw.ppt

7

ifi

## Zweitkürzeste Wege in einem Graphen - (einführende Beispiele)



- Gesucht ist der zweitkürzeste Weg von  $s$  nach  $z$
- Dijkstra ist dem Problem zweit kürzester Wege nicht gewachsen
- Algorithmus oder Graphveränderung ist notwendig

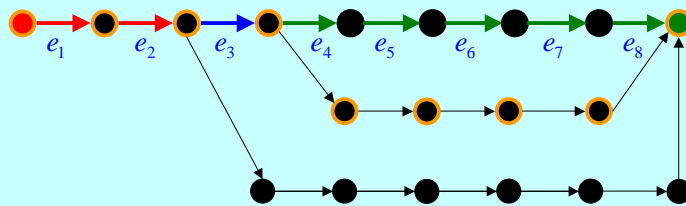
Usr/local/www/ifi/menschen/schmid/kkw.ppt

8

ifi

## Algorithmus von Pollack

- Dynamisches algorithmusveränderndes Verfahren
- Berechne den kürzesten Weg zwischen Start und Zielknoten (m Kanten)
- Entferne jeweils eine Kante ( $e_1, \dots, e_m$ ) des kürzesten Weges und berechne in dem neu entstandenen Graph wieder den kürzesten Weg  $w_2^i$  zwischen Start und Zielknoten (m Berechnungen)
- Der kürzeste Weg dieser m Berechnungen ist der zweitkürzeste Weg zwischen Start und Zielknoten



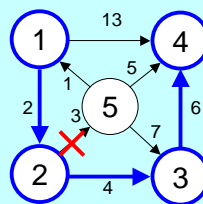
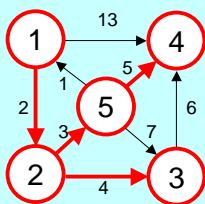
- Kante 1 oder 2 wird weggelassen – kein kürzester Weg wird berechnet
- Kante 3 wird weggelassen – drittkürzester Weg wird berechnet
- Kante 4 – 8 wird weggelassen – zweitkürzester Weg wird berechnet

Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

9

isi

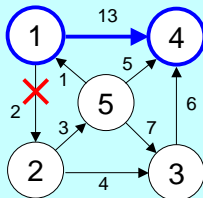
## Der zweitkürzeste Weg von 1 nach 4



$$e_2 = (2,5)$$

$$w_2^2 = (1,2), (2,3), (3,4)$$

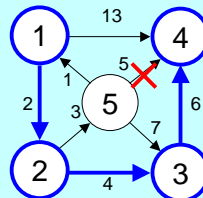
$$\delta(w_2^2) = 12$$



$$e_1 = (1,2)$$

$$w_2^1 = (1,4)$$

$$\delta(w_2^1) = 13$$



$$e_3 = (5,4)$$

$$w_2^3 = (1,2), (2,3), (3,4)$$

$$\delta(w_2^3) = 12$$

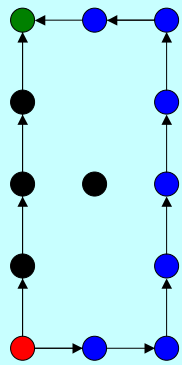
Damit ist der zweitkürzeste Weg  $w_2 = w_2^2 = w_2^3$   
mit  $\delta(w_2) = 12 > 10 = \text{Länge des kürzesten Weges}$

Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

10

isi

## Der Algorithmus von Pollack findet keine Schleifen

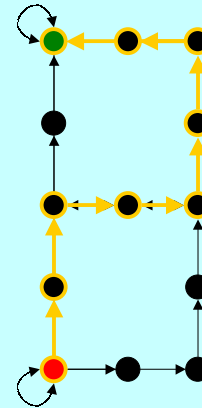


$$w_2^1 = w_2^2 = w_2^3 = w_2^4$$

Sei  $m_1 (< |V|-1)$  die Anzahl der Kanten des kürzesten Weges, so liegt der Aufwand für den zweitkürzesten Weg bei  $m_1 \cdot \text{Dijkstra} = O(m_1 \cdot (|V|+|E|) \log |V|) \leq O(|V| \cdot (|V|+|E|) \cdot \log |V|)$ .

Der Algorithmus berechnet den orangenen Weg als zweitkürzesten Weg (als  $w_2^3$  und  $w_2^4$ ). Dies ist aber nur der zweitkürzeste schleifenfreie Weg.

Weder zweitkürzeste Weg mit Start oder Endschleife noch der Weg mit innerer Schleife werden gefunden.



## Der Algorithmus von Hoffman & Pavley

Der zweitkürzeste Weg wird als Umweg des kürzesten Weges gesehen.

Berechne alle kürzesten Wege zum Zielknoten hin. Der kürzeste Weg  $w := ((s, v_1), (v_1, v_2), \dots, (v_n, z))$  von  $v_0 := s$  nach  $v_{n+1} := z$  wird damit ebenfalls berechnet.

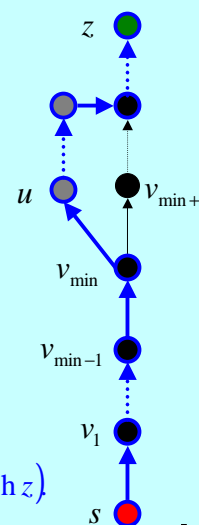
Nimm von jedem Knoten  $v_i$  von  $w$  jede Kante  $(v_i, u)$  mit

$$(*) \quad u \neq v_{i+1} \text{ und berechne den zweitkürzesten Weg als } w_2 := \min_{i=0}^{n+1} \min_u \{d(s, v_i) + \beta((v_i, u)) + d(u, z) \mid u \neq v_{i+1}\}$$

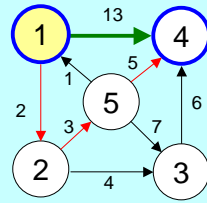
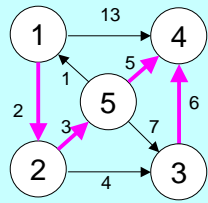
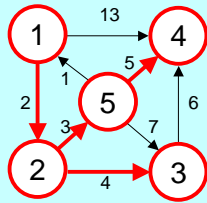
Der zweitkürzeste Weg ist also

$$w_2 := ((s, v_1), \dots, (v_{\min-1}, v_{\min}), \dots, (v_{\min}, u), \text{ kürzester Weg von } u \text{ nach } z)$$

Dieser Algorithmus findet auch Weg mit Zyklen

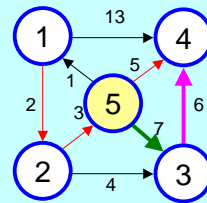
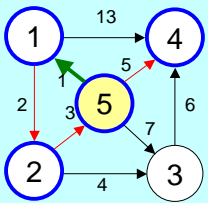
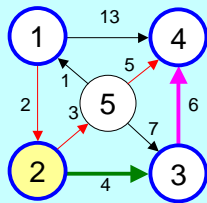


## Der zweitkürzeste Weg von 1 nach 4



$$w_2^1 = 1,4$$

$$\delta(w_2^1) = 13$$



$$w_2^4 = 1,2,5,3,4$$

$$\delta(w_2^4) = 18$$

$$w_2^2 = 1,2,3,4$$

$$\delta(w_2^2) = 12$$

$$w_2^3 = 1,2,5,1,2,5,4$$

$$\delta(w_2^3) = 16$$

$$w_2 = w_2^2 < w_2^1 < w_2^3 < w_2^4$$

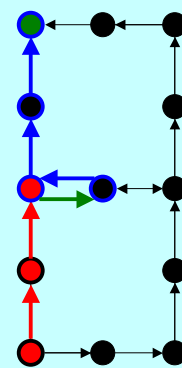
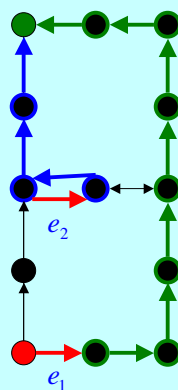
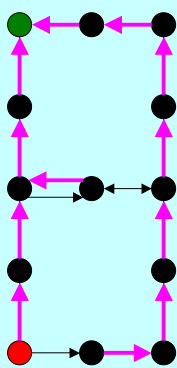
$$\text{mit } \delta(w_2) = 12 > 10$$

Usr/local/www/ifi/menschen/schmid/kkw.ppt

13

isi

## Schleifenbeispiel für den Algorithmus von Hoffman & Pavley



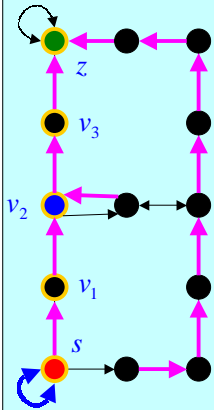
1. Die Kanten des kürzeste Wege Baumes zum Zielknoten hin sind violett
2. Die kürzesten Wege von den Endknoten der Kanten 1 und 2 werden betrachtet
3. Algorithmus erschafft zweitkürzesten Weg mit einem Zyklus

Usr/local/www/ifi/menschen/schmid/kkw.ppt

14

isi

## Schlingen am Anfang oder Ende und Aufwand



Gesucht ist wieder der zweitkürzeste Weg zwischen  $s$  und  $z$ .  
 Betrachte  $s = v_0$ . Die Kante  $(v_0, v_0)$  erfüllt die  
 Eigenschaft (\*)  $u \neq v_{i+1}$  nicht, denn  $v_0 \neq v_1$ . Damit wird der Weg  
 $s, s, v_1, v_2, v_3, z$   
 bei der Minimumbildung berücksichtigt und wird automatisch  
 zweitkürzester Weg.  $\delta(w_2) = 5$ . Wenn man davon ausgeht,  
 daß  $v_{n+2}$  (undefiniert)  $\neq v_{n+1}$  ist, so wird auch der Weg  
 $s, v_1, v_2, v_3, z, z$   
 gefunden, der ebenfalls zweitkürzester Weg ist.

Jeder Knoten kann im kürzesten Weg vorkommen kann und alle Knoten können  
 untereinander verbunden sein. Damit liegt der Aufwand des Verfahrens bei

$$O(|V| \cdot |V|) + O(\text{Dijkstra}) = O(|V|^2), \text{ da } |E| \leq |V|^2 \quad O(|V|^2) < O(|V| \cdot (|V| + |E|) \cdot \log |V|) \text{ (Pollack)}$$

Usr/local/www/ifi/menschen/schmid/kkw.ppt

15

isi

## Verfahrensvergleich

### Pollack

- Dynamisches Verfahren
- Berechne den kürzesten Weg  $w$  zwischen  $s$  und  $z$
- Lasse eine Kante  $e$  von  $w$  weg.
- Berechne den kürzesten Weg im Graphen ohne die Kante  $e$
- Minimiere über die so entstandenen kürzesten Wege
- Kürzester Weg ist gesperrt
- Nur schleifenfreie Wege werden gefunden



### Hoffman & Pavley

- Dynamisches Verfahren
- Berechne den kürzesten Weg  $w$  zwischen  $s$  und  $z$
- Nimm jede Kante  $e$  deren Anfangsknoten ein Knoten von  $w$  ist und selbst keine Kante von  $w$  ist
- Berechne den kürzesten Weg vom Endknoten von  $e$  nach  $z$  und addiere den Weg nach  $s$ .
- Minimiere über die so entstandenen kürzesten Wege
- Zweitkürzester Weg wird als Variante des kürzesten Weges gesehen
- Auch Wege mit Schleifen werden gefunden

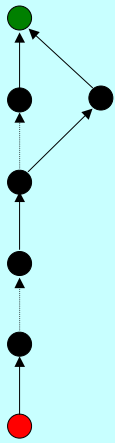
Usr/local/www/ifi/menschen/schmid/kkw.ppt

16

isi

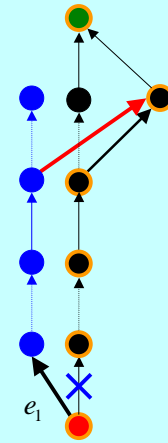


## Algorithmus von Azevedo modifiziert von Schmid



- Statisches Verfahren (verändert nur den Graph – nicht aber den kürzeste Wege- Algorithmus)
- Azevedo verwendet die Methode Pathdeletion (Wegeverbote).

1. Erschaffe einen neuen Startknoten und einen neuen Zielknoten (später).
2. Verdopple den kürzesten Weg ohne erste und letzte Kante bzw. Knoten. (Wegeverdopplung)
3. Weise der ersten Kante einen neuen Endknoten zu. (Umleitung)
4. Ziehe alle Kanten über die der kürzeste Weg verlassen werden kann. (Verbindung mit dem Ausgangsgraphen)

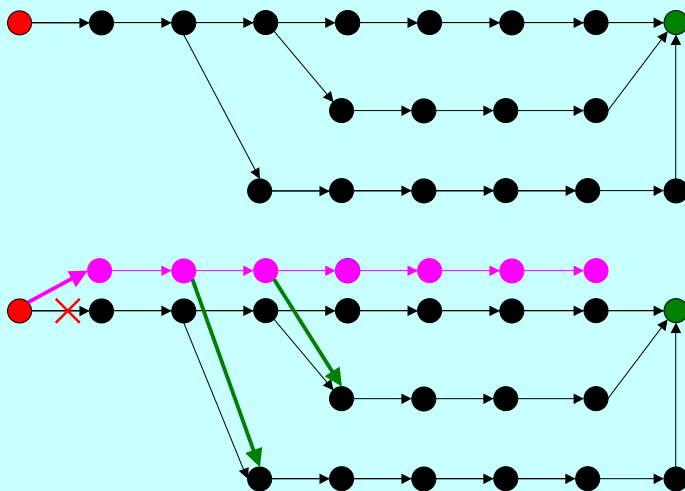


Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

17

isi

## Beispiel für den modifizierten Algorithmus von Azevedo



- Der kürzeste Weg ist nicht mehr begehbar. Der zweit und drittkürzeste Weg sind noch möglich.

Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

18

isi



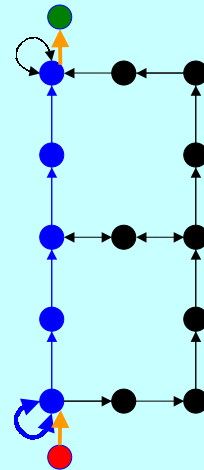
## Der erste Schritt

Problem:

Der Algorithmus von Azevedo kann in dieser Form einen zweitkürzesten Weg der Form Schlinge - (alter) kürzester Weg oder (alter) kürzester Weg – Schlinge nicht erkennen.

Lösung:

- Erschaffe einen neuen Startknoten und einen neuen Zielknoten
- Verbinde den neuen Startknoten mit dem alten Startknoten durch eine Kante der Länge 0.
- Verbinde den alten Zielknoten mit dem neuen Zielknoten durch eine Kante der Länge 0.

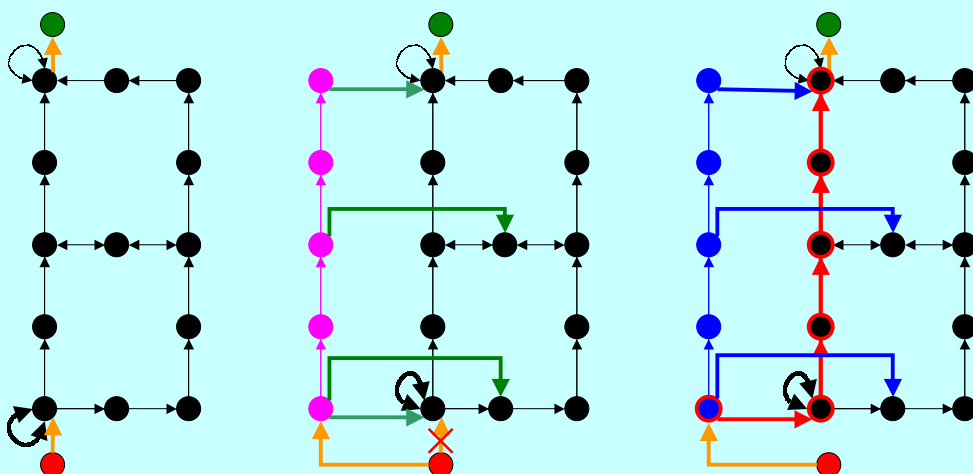


Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

21

ifi

## Beispiel für einen erweiterten Graphen incl. 1 Schritt



Der zweitkürzeste Weg (rot) beinhaltet die Schlinge beim Startknoten

Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

22

ifi

## Aufwand des Verfahrens von Azevedo

Der kürzeste Weg zwischen 2 Knoten kann maximal  $|V|$  Knoten und  $|V|-1$  Kanten beinhalten. Die Anzahl der vom kürzesten Weg wegweisenden Kanten ist maximal  $|E|-2$ . Damit ist die Anzahl der neu erschaffenen Elemente maximal  $|V|$  Knoten und  $|E|-2$  Kanten. Der Platzbedarf kann sich also fast verdoppeln. Der Aufwand für die Grapherweiterung ist  $O(|V|+|E|)$ .

Der Aufwand, den zweit-kürzesten Weg zu berechnen, kann also bei Anwendung des Dijkstra-Algorithmus *zwei* mal so lange dauern wie die Berechnung des kürzesten

Weges. Dies ist also  $O((|V|+|E|) \cdot \log |V|)$  sofern  $|E| \leq \frac{|V^2|}{\log |V|}$ .

Azevedo =  $O((|V|+|E|) \cdot \log |V|) \leq$  Hoffman =  $O(|V|^2)$

< Pollack =  $O(|V|(|V|+|E|) \cdot \log |V|)$



Usr/local/www/ff/k/menschen/schmid/kkw.ppt

23

isi

## Drittkürzeste Wege nach Pollack

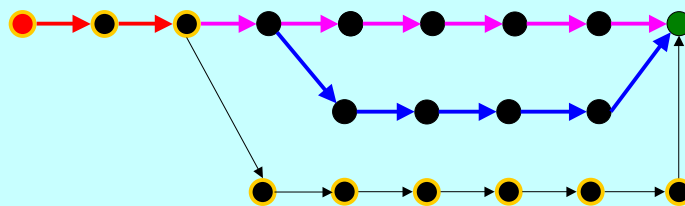
Berechnung des drittkürzesten Weges: Berechnung des drittkürzesten Weges:

Berechnen den kürzesten Weg  $w_1 = (e_1^1, \dots, e_{m_1}^1)$  und den zweitkürzesten Weg

$w_2 = (e_1^2, \dots, e_{m_2}^2)$  zwischen  $s$  und  $z$ . Sperre je ein Paar  $(e_i^1, e_j^2)$  aus  $(w_1 \times w_2)$

berechnen den kürzesten Weg  $w_3^{i,j}$  und bilde dann das Minimum

$$w_3 = \min_{i,j} \{ \delta(w_3^{i,j}) \}$$



Usr/local/www/ff/k/menschen/schmid/kkw.ppt

24

isi

## k kürzeste Wege nach Pollack

Berechnung des  $k+1$  kürzesten Weges:

Berechne die  $k$  kürzesten Wege  $w_1 = (e_1^1, \dots, e_{m_1}^1), \dots, w_k = (e_1^k, \dots, e_{m_k}^k)$

zwischen  $s$  und  $z$ . Sperre je ein  $k$  Tupel  $(e_{i_1}^1, \dots, e_{i_k}^k)$  aus  $(w_1 \times \dots \times w_k)$ .

berechne den kürzesten Weg  $w_k^{i_1, \dots, i_k}$  und bilde dann das Minimum

$$w_k = \min_{i_1, \dots, i_k} \{ \delta(w_k^{i_1, \dots, i_k}) \}$$

Der Aufwand liegt bei  $O(\prod_{i=1}^k m_i) \cdot O(\text{Dijkstra}) \leq O(|V|^{k+1})$ .

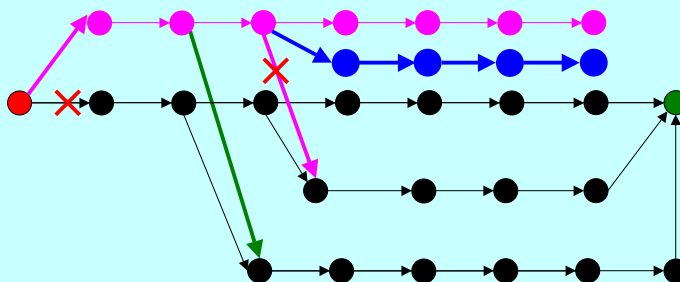


## Drittkürzeste Wege nach Azevedo

Erschaffe einen neuen Startknoten und einen neuen Zielknoten. Berechne den kürzesten Weg  $w_1$  und den zweitkürzesten Weg  $w_2 = (e_1^2, \dots, e_{m_2}^2)$ . Der gemeinsame maximale Anfangsweg  $(e_1^1 = e_1^2, e_2^1 = e_2^2, \dots, e_j^1 = e_j^2)$  muß nicht verdoppelt werden.

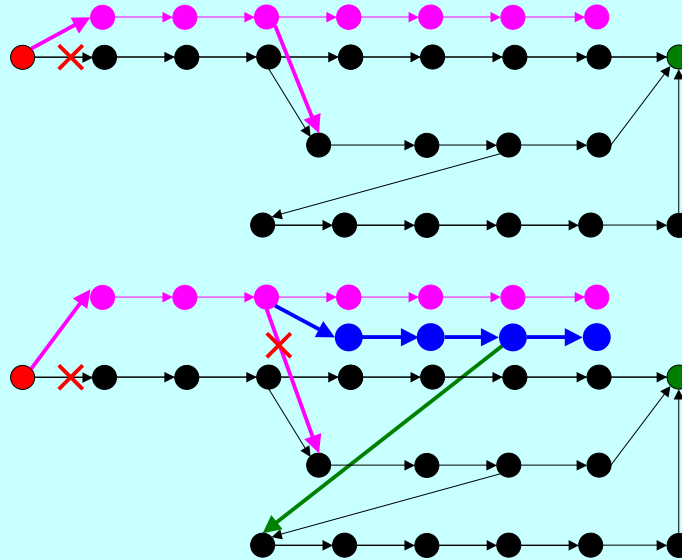
Es werden nur die Kanten  $(e_{j+2}^2, \dots, e_{m_2-1}^2)$  und die dazugehörigen Knoten

verdoppelt. Die Kante  $e_{j+1}^2$  über die der kürzeste Weg verlassen wird, bekommt



einen neuen Endknoten. Ziehe alle Kanten über die  $w_2$  verlassen werden kann.

## Drittkürzeste Wege nach Azevedo



Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

27

isi

## k - kürzeste Wege nach Azevedo

Berechnung des  $k+1$  kürzesten Weges:

Erschaffe einen neuen Startknoten und einen neuen Zielknoten. Berechne die  $k$ -kürzesten Wege  $w_1, \dots, w_k$  (mit  $w_i = (e_1^i, \dots, e_{m_i}^i)$ ). Der gemeinsame maximale Anfangsweg  $(e_1^l = e_1^k, e_2^l = e_2^k, \dots, e_j^l = e_j^k)$  mit  $l < k$ ,  $j$  maximal, muß nicht verdoppelt werden. Es werden nur die Kanten  $(e_{j+2}^k, \dots, e_{m_k-1}^k)$  und die dazugehörigen Knoten verdoppelt. Die Kante  $e_{j+1}^k$  über die  $w_l$  verlassen wird, bekommt einen neuen Endknoten. Ziehe alle Kanten über die  $w_k$  verlassen werden kann.

Der Aufwand liegt (vermutlich) bei  $O((|E| + |V|) \cdot (k-1) \cdot \log |V|)$ .



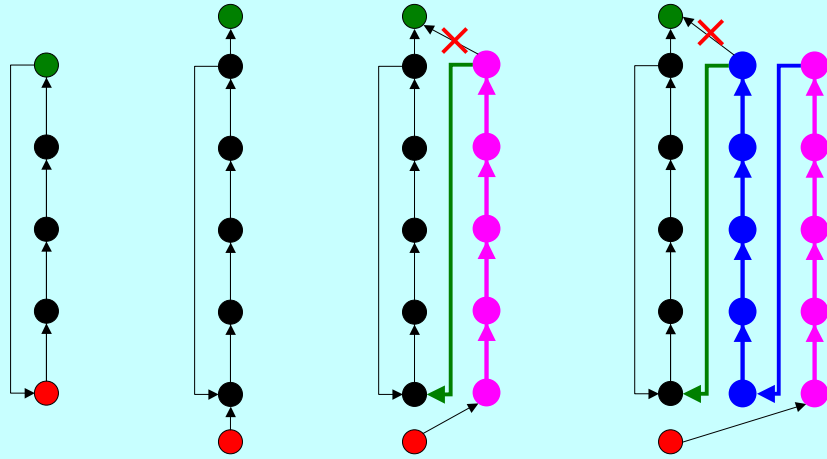
Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

28

isi

## Der schlimmste Fall

Gegeben sei ein Graph, der nur aus einem Zyklus besteht. Der  $k$ -kürzeste Weg von  $s$  nach  $z$  ergibt sich aus dem kürzesten Weg und dem anschließenden  $k-1$  maligen Durchlaufen des kompletten Graphen.



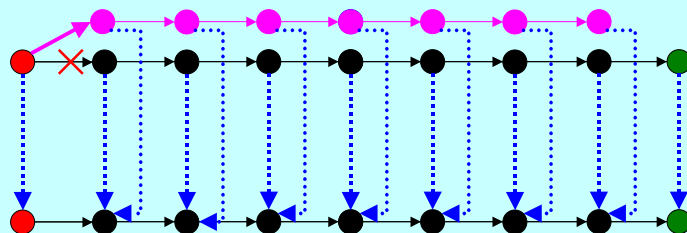
Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

29

ifi

## Beweis des Algorithmus von Azevedo

- Jeder Weg im erweiterten Graphen ist auch ein Weg im Ausgangsgraphen (die zeigt man über kanonische Projektionen = inverse Abbildung der Wegeverdopplung).
- Alle Wege (außer Wegen, die die  $k$ -kürzesten Wege enthalten) im Ausgangsgraphen, sind auch Wege im erweiterten Graphen (man konstruiert den Verlauf eines gegebenen Weges im erweiterten Graph).  
Genau die  $k$ -kürzesten Wege sind verboten.



Usr/local/www/ifi/k/menschen/schmid/kkw.ppt

30

ifi

## Literatur

- J.A.de Azevedo, J.J. Madeira, E. Q. Martins, F. M. Pires: A shortest path ranking algorithm. AIRO 90 – Models and Methods for Decision Support, Sorrent (Italien), Proceedings of the Annual Conference, Associazione Italiana di Ricerca Operativa (S. 1001-1011) 10.1990
- E. W. Dijkstra: A note on two problems in connection with graphs. Numerical Mathematics 1, (S. 269-271), 1959
- W. Knödel: Graphentheoretische Methoden und ihre Anwendungen. Springer Verlag Berlin, Heidelberg 1969
- M. Mack: Untersuchung von effizienten Algorithmen zur Bestimmung der k-kürzesten Wege innerhalb von ÖPNV-Verkehrsnetzen. Diplomarbeit Nr. 1374, Fakultät Informatik, Universität Stuttgart, 1996
- W. Schmid: Kürzeste Wege in Straßennetzen mit Wegeverboten. Verlag der Bayerischen Akademie der Wissenschaften, München, 2001



isi