Proseminar Online-Algorithmen

Zielpunktsuche in einer unbekannten Umgebung (mit Sichtsystem)

Vortrag von Joachim Voll.

24.01.2001

Proseminar : Online-Algorithmen
Prof. Dr. Rolf Klein

Inhalt

- Problemvorstellung, Zieldefinition … 3
- Das Brückenproblem (kurze Rekapitulation) … 4
- 1. Erweiterung – m > 2 … 5
  - 1. Strategie … 6
    - Beweis der Kompetitivität … 6
  - 2. Strategie … 7
    - Beweis der Kompetitivität … 8
- 2. Erweiterung – einfaches Polygon (Korridorpolygon) … 9
- 3. Erweiterung – Sichtsystem/ Sichtbarkeitspolygon vis(p) … 10
  - Beweis : Weglängenverhältnis $\leq \frac{n}{2} - 1$ … 11
  - Erkenntnisse … 12
- 4. Erweiterung – beliebige Polygone … 13
  - SPT (shortest path tree) … 14
  - Beweis der Kompetitivität … 15
  - Probleme und Lösungen … 16
- 5. Erweiterung – zyklische Liste … 17
- Praktische Verbesserung/ Ziel … 19

Problemvorstellung und Zieldefinition

Problem :

Wir suchen eine Strategie um in einem unbekannten Polygon mit einem Roboter von einem Startpunkt zu
einem Zielpunkt zu finden.

Ziel :

Bei dieser Strategie soll der gefundenen Weg kompetitiv zu dem optimalen Weg sein und die Strategie soll in jedem Fall den Zielpunkt finden.

Vorgehen :

Ich werde bei dem bekannten Brückenproblem beginnen und dieses schrittweise erweitern

Das Brückenproblem (kurze Rekapitulation)

Ausgangspunkt der Betrachtungen ist das bekannte **Brückenproblem:**

Gesucht ist ein nicht bekannter Zielpunkt auf zwei Halbgeraden.

Startpunkt	:	s	
Zielpunkt	:	t	
Halbgeraden	:	m_n	
Suchtiefe	:	f_j	
Kosten (optimal)	:	K_{OPT}	
Kosten (der Strategie)	:	K_S	Hierbei bezeichnen die Kosten die Weglänge

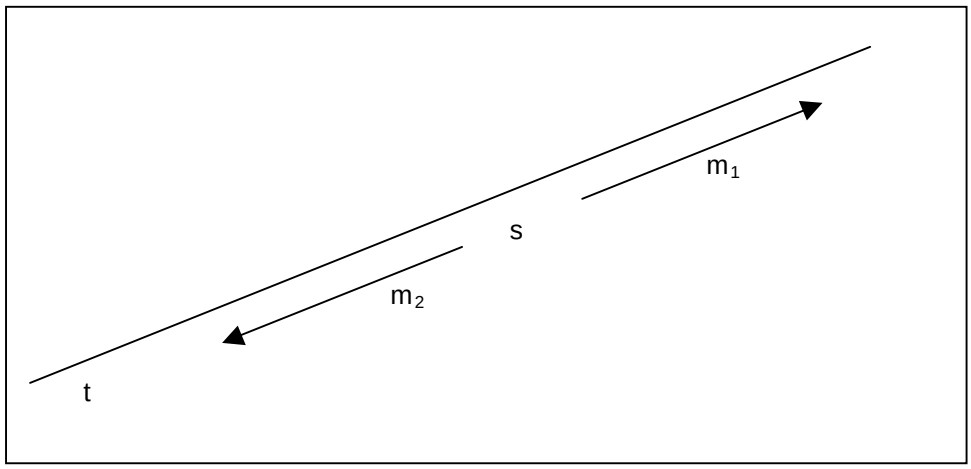

Abb. 1

Zur Lösung dieses Problems haben wir im wesentlichen zwei Ansätze gehabt:

1. inkrementelle Suchtiefenvergrößerung (unberücksichtigt)
2. exponentielle Suchtiefenvergrößerung

Mit der exponentiellen Suchtiefenvergrößerung von $\mathbf{f_j = 2^n}$ haben wir die (optimale) **9-Kompetitivität** für das allgemeine Brückenproblem bewiesen.

1. Erweiterung m > 2

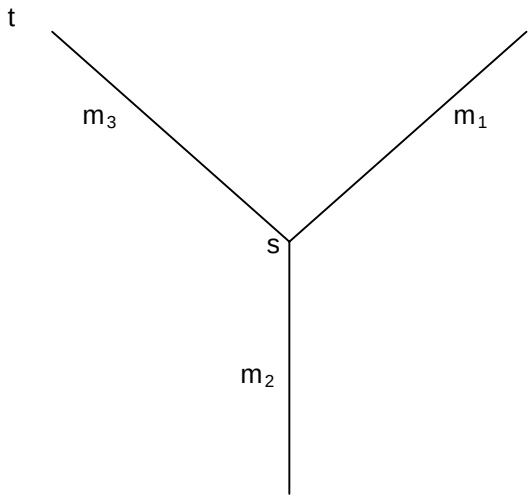

Abb. 2

Bezeichnungen

Startpunkt	:	s	
Zielpunkt	:	t	
Halbgeraden	:	m_n	
Suchtiefe	:	f_j	
Kosten (optimal)	:	K_{OPT}	
Kosten (der Strategie)	:	K_S	Hierbei bezeichnen die Kosten die Weglänge

1. Erweiterung m > 2

1. Strategie : Doubling auf $m > 2$
(d.h. Suchtiefenverdopplung nach jeder Runde)

worst case: in der Runde mit der Suchtiefe 2^j wird der Zielpunkt auf der Halbgerade m_j um ein ε verfehlt, darauf hin beginnt eine neue Runde mit der Suchtiefe 2^{j+1}.

Suchtiefe : $f_j = 2^j$

Weglänge bis zur Runde 2^j : $2m\sum_{i=0}^{j} 2^i$

Weglänge der neuen Runde : $2(m-1)2^{j+1}$

Weglänge von s zu t (K_{OPT}) : $2^j + \varepsilon$

Kompetitivitätsbeweis :

$$\rightarrow \frac{K_S}{K_{OPT}} = C = \frac{2m\sum_{i=0}^{j} 2^i + 2(m-1)2^{j+1} + 2^j + \varepsilon}{2^j + \varepsilon}$$

$$= \frac{2m\sum_{i=0}^{j} 2^i + 2(m-1)2^{j+1}}{2^j + \varepsilon} + \frac{2^j + \varepsilon}{2^j + \varepsilon}$$

$$\leq \frac{2m2^{j+1} + 2(m-1)2^{j+1}}{2^j} + 1$$

$$= 8m - 3$$

Der Kompetitivitätsfaktor ist also $\leq 8m-3$

1. Erweiterung m > 2

2. Strategie : exponentielle Suchtiefenvergrößerung

worst case: in der Runde mit der Suchtiefe f_j wird der Zielpunkt auf der Halbgerade m_j um ein ε verfehlt, darauf hin beginnt eine neue Runde mit der Suchtiefe f_{j+1}.

Suchtiefe : $f_j = \left(\frac{m}{m-1}\right)^j$

Weglänge bis zur Runde 2^j : $2\sum_{i=1}^{j} f_i = 2\sum_{i=1}^{j}\left(\frac{m}{m-1}\right)^i$

Weglänge der neuen Runde : $2\sum_{i=1}^{m-1} f_i$

Weglänge von s zu t (K_{OPT}) : $2^j + \varepsilon$

Betrachtung der Gesamtkosten der Strategie :

$$2\sum_{i=1}^{j} f_i + 2\sum_{i=1}^{m-1} f_i + f_j + \varepsilon$$

$$= 2\sum_{i=1}^{j+m-1}\left(\frac{m}{m-1}\right)^i + f_j + \varepsilon$$

$$< 2\frac{\left(\frac{m}{m-1}\right)^{j+m} - 1}{\frac{m}{m-1} - 1} + f_j + \varepsilon$$

$$\leq 2(m-1)\left(\frac{m}{m-1}\right)^{j+m} + f_j + \varepsilon$$

1. Erweiterung m > 2

2. Strategie : exponentielle Suchtiefenvergrößerung

Kompetitivitätsbeweis :

$$K_S(P) \le C \cdot K_{OPT}(P) + A$$

$$\rightarrow \quad 2(m-1)\left(\frac{m}{m-1}\right)^{j+m} + f_j + \varepsilon \le C + \left(f_j + \varepsilon\right) + A$$

$$\frac{K_S(P)}{K_{OPT}(P)} \le C$$

$$\rightarrow \quad \frac{2(m-1)\left(\frac{m}{m-1}\right)^{j+m} + f_j + \varepsilon}{f_j + \varepsilon}$$

$$= \frac{2(m-1)\left(\frac{m}{m-1}\right)^{j+m}}{\left(\frac{m}{m-1}\right)^{j} + \varepsilon} + \frac{f_j + \varepsilon}{f_j + \varepsilon}$$

$$\le 2(m-1)\left(\frac{m}{m-1}\right)^{m} + 1$$

$$= 2m\left(1 + \frac{1}{m-1}\right)^{m-1} + 1$$

$$\le 2me + 1$$

Bei der exponentielle Suchtiefenvergrößerung kann der kompetitive Faktor also nicht größer sein als ***2me+1***

2. Erweiterung einfaches Polygon (Korridorpolygon)

Wir erweitern nun die Umgebung auf ein einfaches Polygon in dem sich der Roboter bewegen soll.

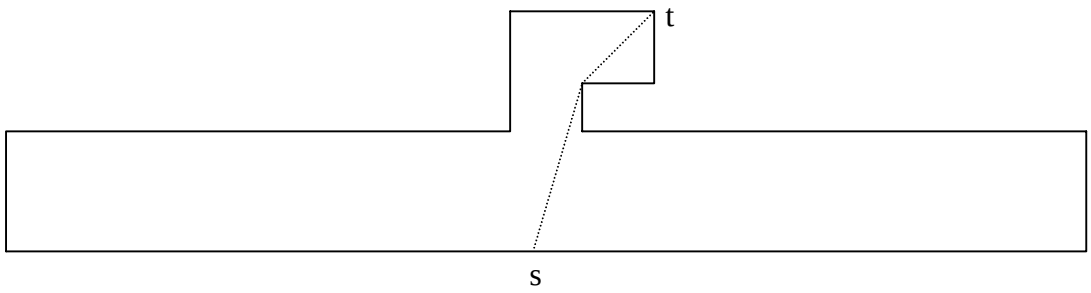

Abb. 3

Ferner gelten folgende Bezeichnungen:

P	:	Polygon
s	:	Startpunkt
t	:	Zielpunkt
p	:	aktuelle Position
n	:	Anzahl der Ecken im Polygon
π	:	zurückgelegter Weg von s nach p

Wir finden selbst bei exponentieller Suchtiefenvergrößerung **keinen kompetitiven Faktor** zum optimalen Weg. Daher erweitern wir unseren Roboter nun mit einem Sichtsystem.

Wir geben für spätere Betrachtungen ab nun stets die Anzahl der Ecken in dem Polygon an (*hier* $n=10$)

3. Erweiterung Sichtsystem / Sichtbarkeitspolygon vis(p)

Sichtsystem (optimal) :

- 360°-Sicht
- 3D-Sicht (bsw. durch 2 Kameras)
- Zielpunkt optisch markiert
- Startposition bekannt

Verarbeitung der Sichtinformation in p in einem **Sichtbarkeitspolygon *vis(p)***.

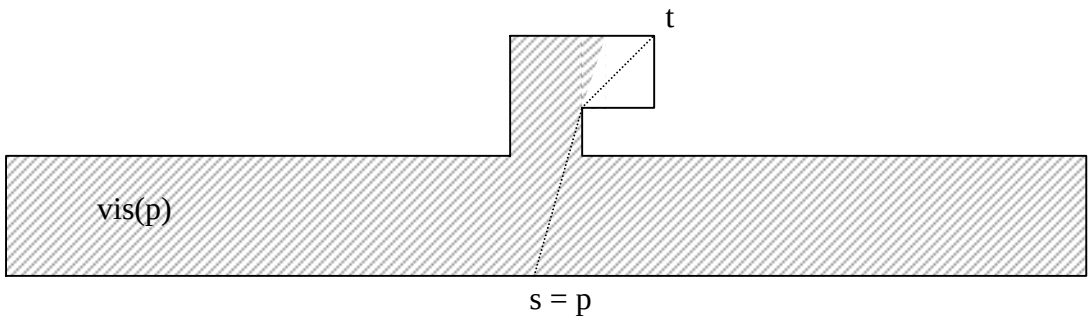

Abb. 4

Speicherung aller bisher bekannten Teile von P in einer **partielle Karte**

formal:

$vis(p) \subset P$ — Das **Sichtbarkeitspolygon** enthält alle von p aus sichtbaren Punkte von P und ist somit eine Teilmenge von P

$\bigcup_{p \in \pi} vis(p)$ — d.h. die **partielle Karte** enthält genau die Punkte die während des Weges von s nach p aus sichtbar waren.

Man kann beweisen (Lemma) :

Keine Strategie für die Suche nach einem Zielpunkt kann in beliebigen Polygonen mit n Ecken stets einen kleineren Faktor als $\frac{n}{2}-1$ für das Weglängenverhältnis garantieren.

3. Erweiterung Sichtsystem / Sichtbarkeitspolygon vis(p)

Beweis (anschaulich) :

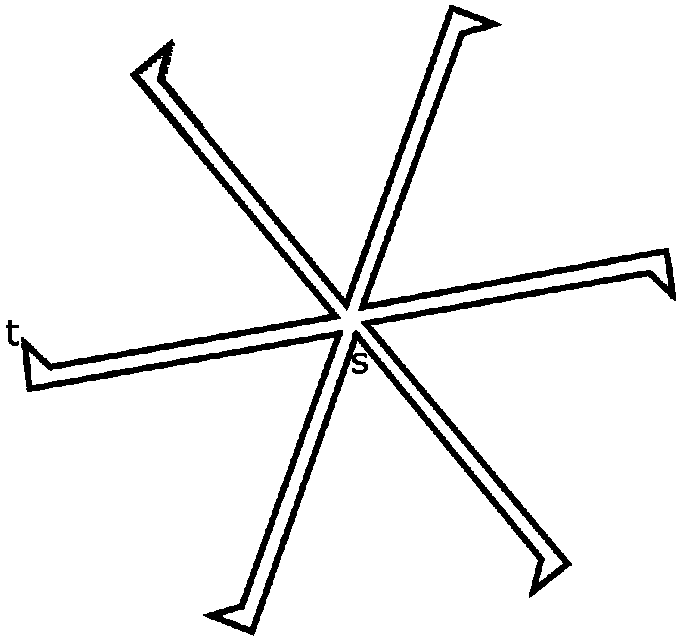

Abb. 5

s = Startpunkt
t = Zielpunkt

Ecken = n = 24
Korridore = 6

Länge eines Korridors = 1

Weiten und Längen der abgeknickten Ecken sind als vernachlässigbar klein anzuneh-men

- P hat $\frac{n}{4}$ Korridore
- Die Korridore laufen in s zusammen

Worst case-Betrachtung :

t befindet sich im zuletzt besuchten Korridor

	K_S	K_{OPT}
$\Rightarrow$	$\left(\frac{n}{4}-1\right)2+1=\frac{n}{2}-1$	1

Womit das Lemma bewiesen wäre.

3. Erweiterung Sichtsystem / Sichtbarkeitspolygon vis(p)

Erkenntnisse :

- Wenn wir eine Strategie angeben können bei der das Weglängenverhältnis stets beschränkt ist, so kann diese Schranke keine Konstante.
- Wir haben gesehen: im schlimmsten Fall kann der zurückgelegte Weg linear von der Anzahl der Polygonecken abhängen
- Wir haben somit bewiesen: wir können stets ein Weglängenverhältnis in $\Omega(n)$ erwarten, wenn wir eine Lösungsstrategie angeben können
- Im Fall von Korridorpolygonen können wir die exponentielle Suchtiefenvergrößerung anwenden.
  Damit können wir bei diesen Polygonen ein Weglängenverhältnis von $O(n)$ garantieren.

Wir wollen nun versuchen von Korridorpolygonen zu beliebigeren Polygonen zu verallgemeinern.

4. Erweiterung – beliebige Polygone

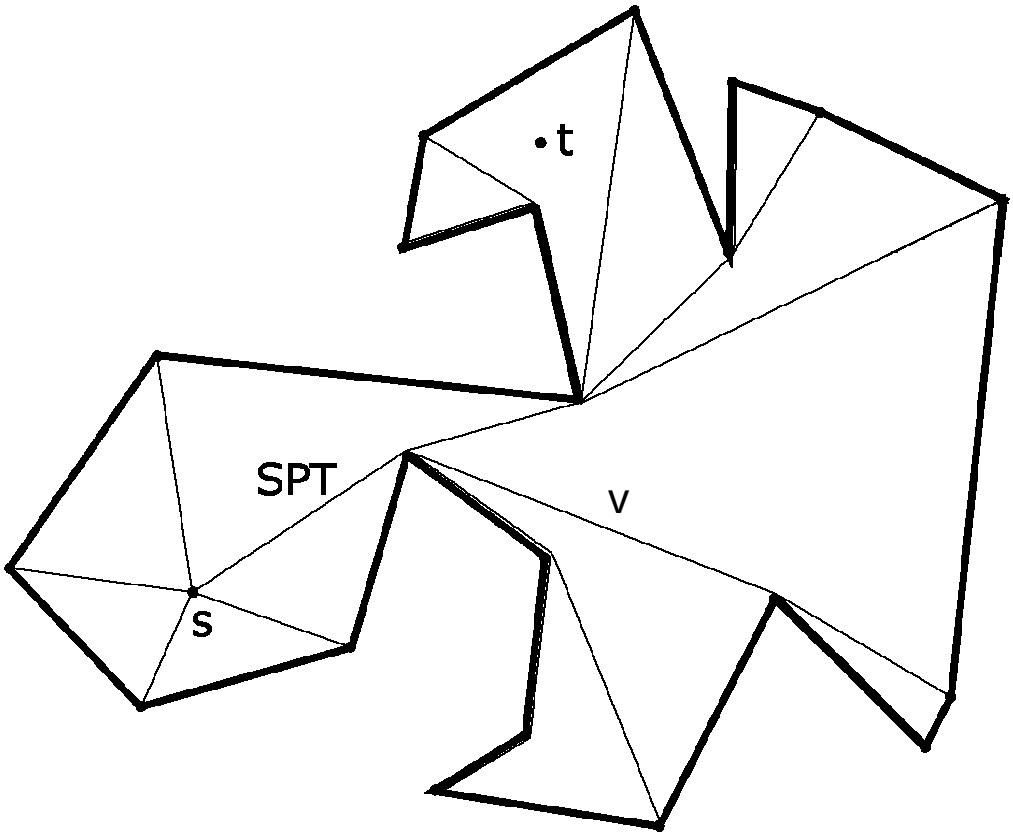

Abb. 6

Wir betrachten in dem Polygon nun die kürzesten Wege
von s zu allem Eckpunkten von P.

Wie sich leicht erkennen lässt (siehe Seite 14) bilden diese kürzesten Wege einen Baum. Diesen nennt man **SPT** (shortest path tree).

Der **SPT** besitzt in einem Polygon mit n Ecken
und einem Startpunkt s in P genau n+1 Knoten,
darunter $m \leq n$ viele Blätter.

<u>Vereinfachung:</u>
Roboter geht erst in v direkt zu t (vgl. Praxis)

4. Erweiterung – beliebige Polygone

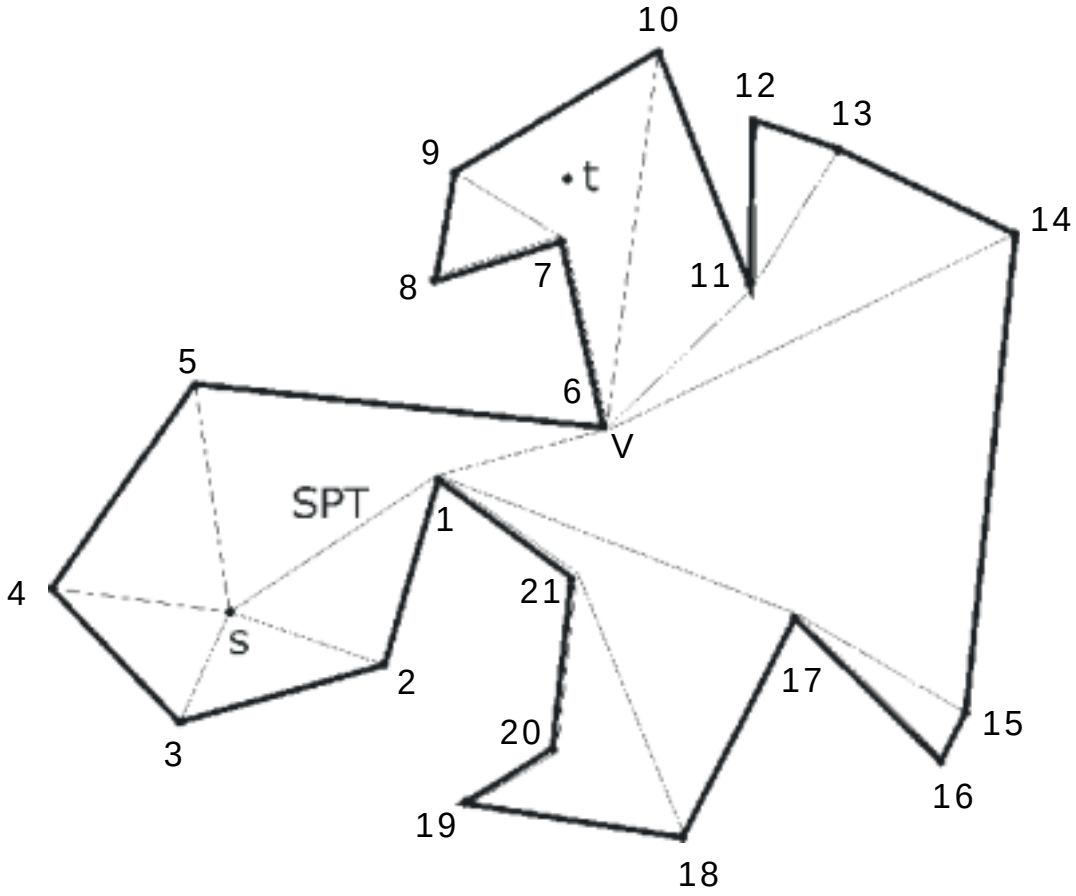

Abb. 7

Zum Polygon gehöriger SPT :

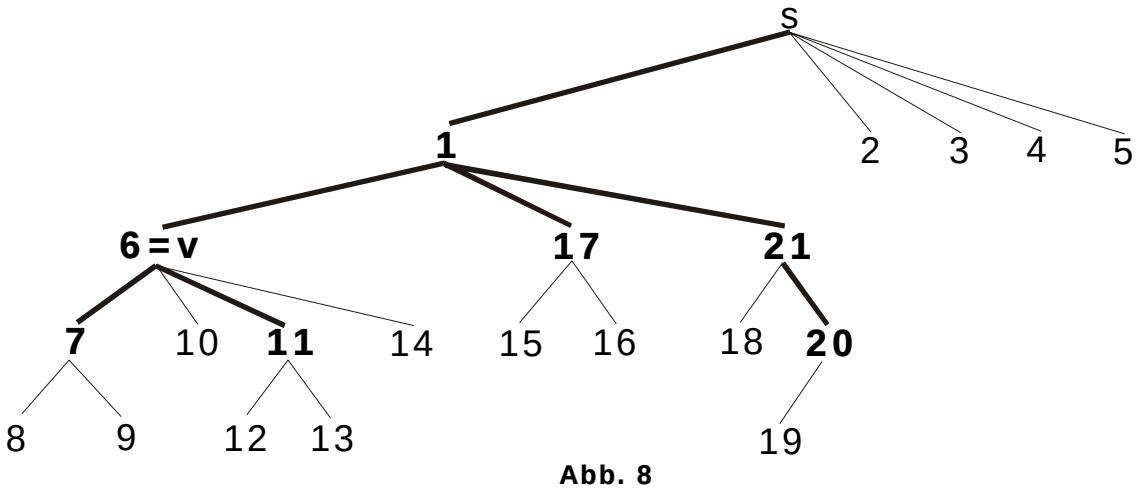

Abb. 8

4. Erweiterung – beliebige Polygone

Bezeichnungen/ Definitionen :

w(s,p) : Länge des tatsächlich zurückgelegten Wegs von s nach p
d(s,p) : Länge des kürzesten Wegs von s nach p

Beweis der Kompetitivität

Wir wissen :
v liegt auf dem Weg von s nach t, somit ergibt sich

$$d(s,t) = d(s,v) + |vt|$$

Wegen $d(s,v) \leq w(s,v)$ ergibt sich folgendes Weglängenverhältnis:

$$\frac{w(s,t)}{d(s,t)} = \frac{w(s,v)+|vt|}{d(s,v)+|vt|} \leq \frac{w(s,v)}{d(s,v)} < 2em+1$$

(Beweis analog zum Beweis ab Seite 6)

D.h. wir können einen kompetitiven Faktor erreichen, der linear von m (Anzahl der Blätter von SPT in s) abhängt.

Probleme

1. **wie kann der Roboter den Vergrößerungsfaktor $\frac{m}{m-1}$ für die Suchtiefe bilden ?**

Antwort:
nicht möglich, da der Algorithmus online arbeitet.

Lösung:
zurückgreifen auf Doubling-Strategie

Erklärung:
Dieses ist eine Alternative, da Doubling auch linear von m abhängt und sich unterhalb 8m bewegt (wie bewiesen)

2. **wie soll der Roboter den kompletten SPT von s aus kennen ?**

Antwort:
kein Problem

Erklärung:
da zu jedem Zeitpunkt der SPT soweit entwickelt ist wie Roboter sehen kann, ist auch genügend Information für die Planung des nächsten Schritts vorhanden. Dies gilt da der Roboter für jeden Punkt p in P der bereits gesichtet ist kürzester Weg s → p kennt.

„Beweis“:
Dieses gilt, da p, sobald gesichtet, zur aktuellen partiellen Karte P’ gehört (wie auch s). Also enthält P’ auch den kürzesten Weg von s→p in P; dieses lässt sich anschaulich leicht beweisen.

5. Erweiterung – zyklische Liste

Finden wir eine Strategie die es dem Roboter ermöglicht mit Doubling einen Zielpunkt zu finden haben wir bewiesen, dass es eine Strategie gibt, die $8n$-kompetitiv ist.

zyklischen Liste L :
verwaltet alle zum Zeitpunkt schon sichtbar gewesenen kürzesten Wege zu allen Eckpunkten von P

Vorgehensweise und Aktualisierung von L :

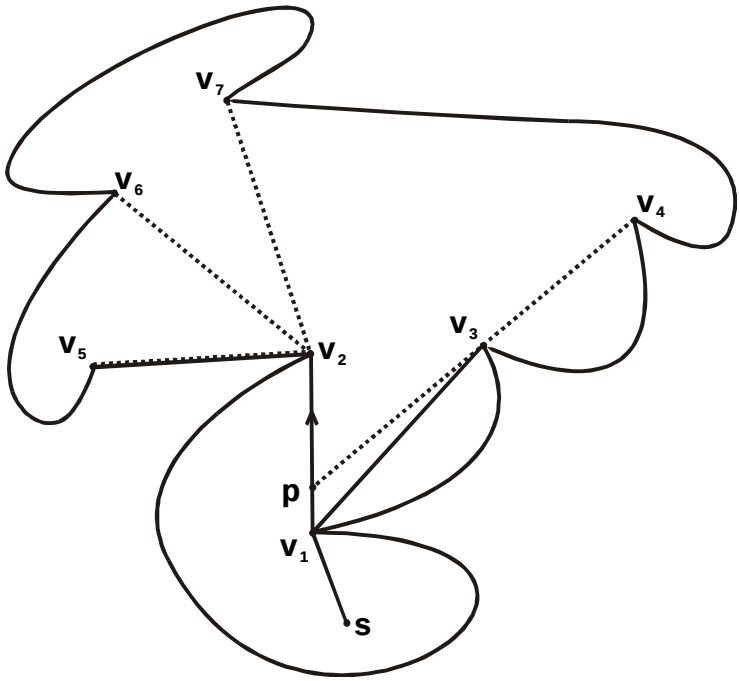

Abb. 9

Vorgehensweise und Aktualisierung von L

Schritt	**Weg**	**SPT**	**L**
0	-	s, v1	(s, v1)
1	(s, v1)	s, v1, v2, v3	(s, v1, v2) (s, v1, v3)
2	*(s, v1, p)*	*s, v1, v2, v3, v4*	*(s, v1, v2)* *(s, v1, v3, v4)*
3	(s, v1, v2)	s, v1, v2, v3, v4, v5, v6, v7	(s, v1, v2, v5) (s, v1, v2, v6) (s, v1, v2, v7) (s, v1, v3, v4)

Algorithmusabhängig ob Aktualisierung hinsichtlich v4 im Schritt 2 durchgeführt wird, oder erst wenn dieser Weg an der Reihe ist abgelaufen zu werden.

Im letzteren Fall würde v4 bei Ankunft in v3 „wiederentdeckt“

Wir haben somit gezeigt, dass es für Roboter mit Sichtsystem eine kompetitive Strategie für die Suche nach einem Zielpunkt in einem einfachen Polygon mit n Ecken gibt, die den Faktor $8n$ garantiert!

Praktische Verbesserungen / Ziel

In der Praxis lässt sich der Faktor von $8\mathrm{m}$ verbessern :

1. Rückkehr zur Wurzel s nicht nötig.
   entweder Rückkehr zu Knoten an denen sich Wege teilen bsw.
   (v_2,v_1) dann (v_1,v_3)
   oder „freies Bewegen“ im Raum
   bsw. (v_7,v_4)

2. Weg von Knoten zu Blättern nicht nötig
   ausnutzen der Tatsache, dass im Inneren einfacher Polygone
   jeder Punkt von einem inneren Knoten vom SPT sichtbar ist.
   → Beispiel siehe Abbildung 8. Hierbei sind die wichtigen Wege im SPT fett gedruckt.

Da sich der obige Ansatz auf beliebige Umgebungen verallgemeinern lässt, haben wir mit diesem Ansatz eine effiziente Alternative für Roboter mit Sichtsystemen zur Strategie Bug vorgestellt

... und unser Ziel damit erreicht.

Für Interessierte :

http://wwwpi6.fernuni-hagen.de/Geometrie-Labor/apps/stip/