Bewegungsplanung bei unvollständiger Information

Sebastian Hempel

Aktuelle Forschungsthemen in der Algorithmik

Überblick

1. Einleitung
2. Auswege aus einem Labyrinth
3. Finden eines Ziels in unbekannter Umgebung
4. Kompetetive Strategien

1. Einleitung

Situation:

- Typisch: Alle Informationen bekannt
- Jetzt: Nur Teilinformationen vorhanden (z.B. bei autonomen Robotern)

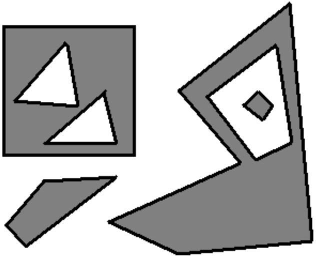
Neue Effizienzmaße:

- Planung ⇒ Rechenzeit
- Suche ⇒ Länge des Wegs

Ziel: Finden von Strategien.

Labyrinth/Umgebung

- Einfache, geschlossene polygonale Ketten
- keine Überschneidungen
- Ränder sind Wände
- „Innenhöfe“ möglich
- Außen freies Gebiet
- Bewegung auf freier Fläche



Roboter

Aufbau:

- Radius r
- Als punktförmig aufgefasst ⇒ Anpassung der Hindernisse
- Besitzt ein „Vorne“
- Idee: Lozano-Pérez

Ausstattung:

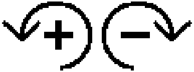
- Tastsensor
- Rundumsicht
- Position
- Drehwinkel

2. Auswege aus einem Labyrinth

Freie Universität Berlin

Roboter:

- Punktförmig
- Tastsensor
- Zurücksetzbarer Wineklzähler



Ziel: Finden einer Strategie, welche einen Weg aus unbekanntem Labyrinth findet, falls ein solcher existiert.

1.Strategie - Höhlenforscherstrategie

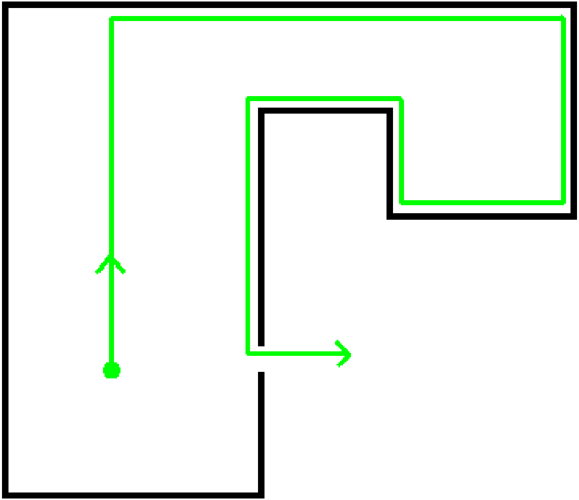
Freie Universität Berlin

- Linke Hand an de Wand
- ⇒ Bei Hindernis eine Rechtsdrehung

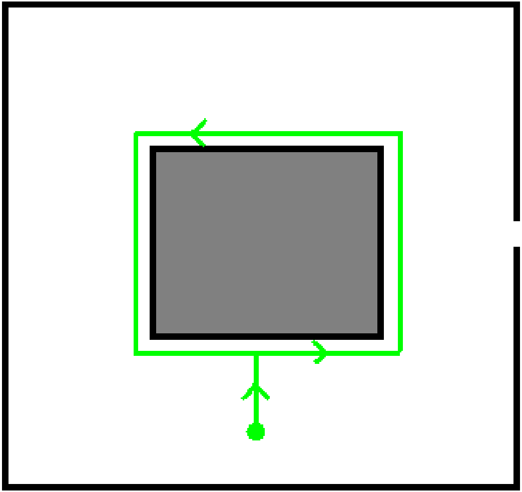
Algorithmus:

1. Wähle beliebige Richtung
2. Gehe bis Wand
3. Folge der Wand bis nach draußen

1.Strategie - Höhlenforscherstrategie



1.Strategie - Höhlenforscherstrategie



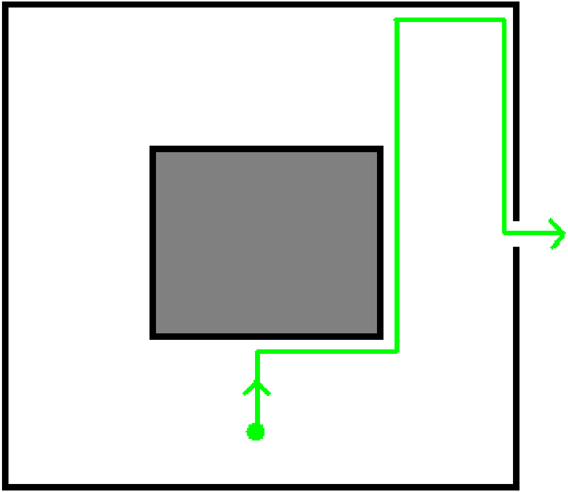
2. Strategie

- Linke Hand an de Wand
- ⇒ Bei Hindernis eine Rechtsdrehung

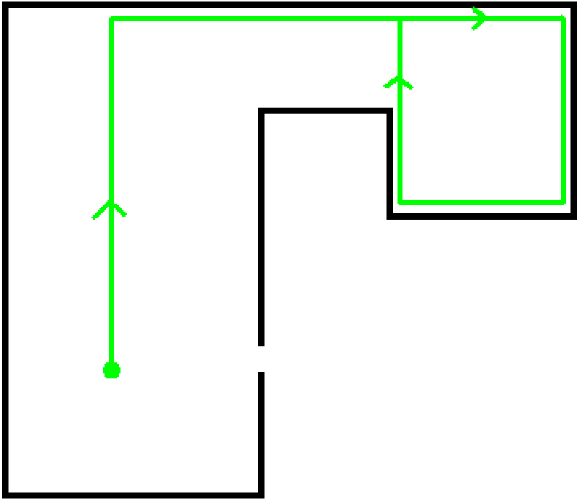
Algorithmus:

1. Initialisiere Winkelzähler w mit 0
2. Bis entkommen wiederhole:
    2.1 Gehe bis Wand
    2.2 Folge der Wand, ist w mod 360 = 0 ⇒ lösen

2. Strategie



2. Strategie



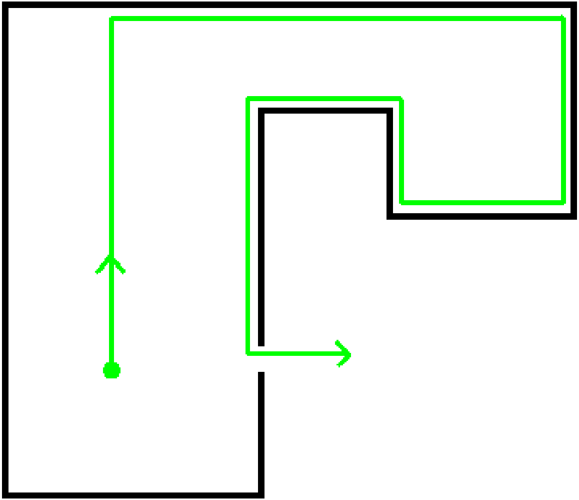
3. Strategie - Pledge-Algorithmus

- Linke Hand an der Wand
- ⇒ Bei Hindernis eine Rechtsdrehung

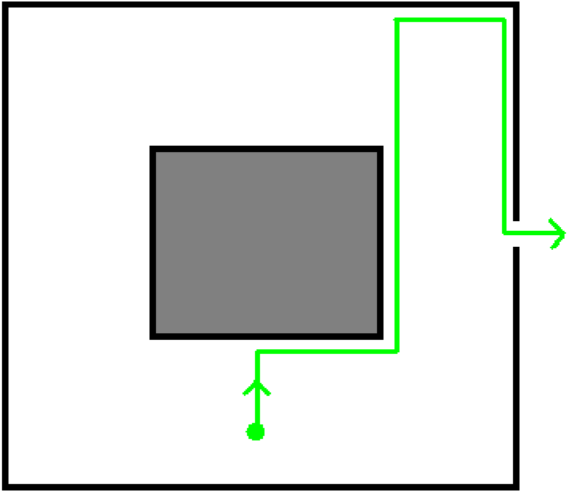
Algorithmus:

1. Initialisiere Winkelzähler w mit 0
2. Bis entkommen wiederhole:
    - 2.1 Gehe bis Wand
    - 2.2 Folge der Wand, ist w=0 ⇒ lösen

3. Strategie - Pledge-Algorithmus



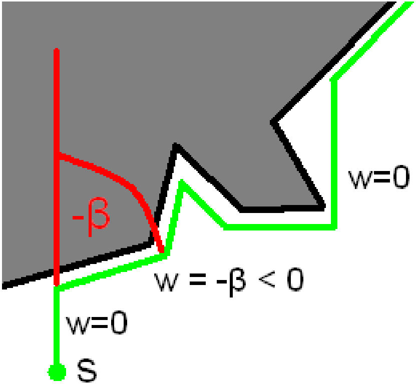
3. Strategie - Pledge-Algorithmus



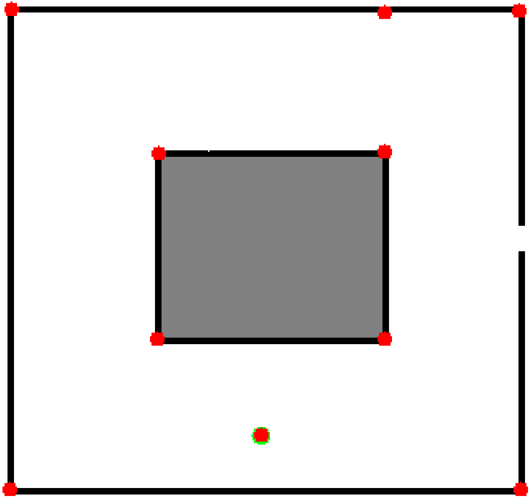
***Beweis*: Der Pledge-Algorithmus findet immer einen Weg aus einem (unbekannten) Labyrinth, falls ein solcher existiert.**

Freie Universität Berlin

Winkelzähler ≤ 0:

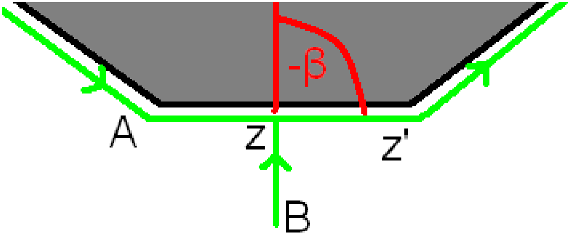


Endlich viele Punkte:

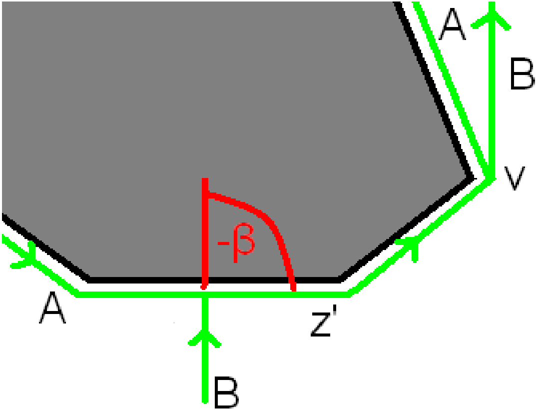


3. Strategie - Pledge-Algorithmus

Keine Kreuzung:



Keine Kreuzung:



3. Finden eines Ziels in unbekannter Umgebung

Roboter-Ausstattung:

- Tastsensor
- Aktuelle Position
- Koordinaten des Ziels
- Merken bekannter Punkte

Bug-Strategie

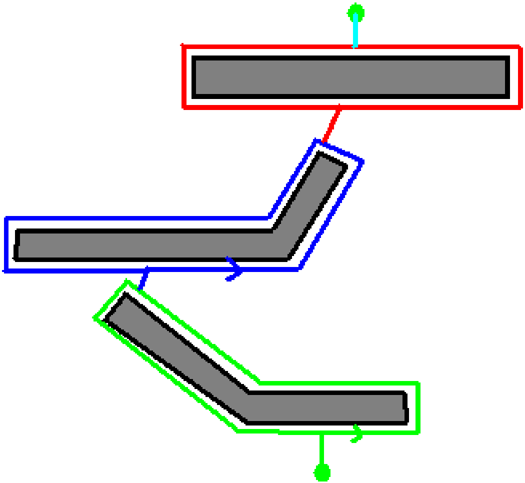
Freie Universität Berlin

Algorithmus:

1. Bis Ziel erreicht:
   1.1 Laufe auf Ziel zu bis eine Wand erreicht ist (Punkt A)
   1.2 Umrunde das Hindernis, merke Punkt D mit geringstem Abstand zum Ziel
   1.3 Gehe auf kürzestem Weg zu D und löse dich dort

Anmerkung: Wird das Ziel auf dem Weg erreicht, so verlasse die Schleife

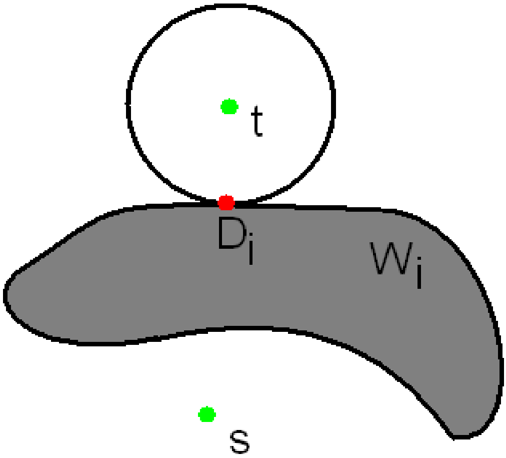
Bug-Strategie



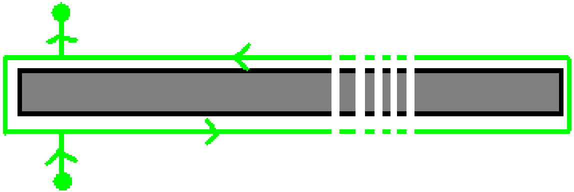
Bug-Strategie - Beweis

***Beweis*: Die Bug-Strategie finden immer einen Weg vom Startpunkt s bis zum Ziel t, falls ein solcher existiert.**

Bug-Strategie - Freier Weg



Bug-Strategie - Beliebig Schlecht



Schranke:

$$\text{Weglänge} \leq |st| + \frac{3}{2}\sum_{i=1}^{n} U_i$$

$U_i \ldots U_n$: Länge der Wände, auf der Punkte existieren, welche näher an t liegen als s.

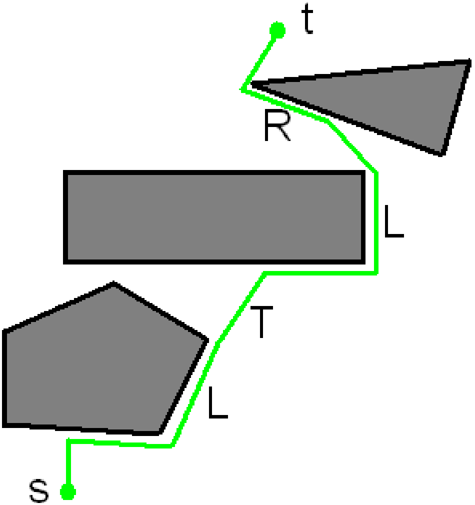
Um ein Ziel in unbekannter Umgebung zu finden genügen

- ein Tastsensor und
- ein Zielkompass.

Befehle zur Steuerung:

- T : Von Ecke lösen und auf Ziel zusteuern
- L : Mit der linken Seite an der Wand bleiben
- R : Mit der rechten Seite an der Wand bleiben

TLR



4. Kompetetive Strategien

- Biser: Aufgaben erfüllt, aber bel. Abweichung vom Optimum
- Gibt es Strategien, mit dem man dem Optimum nahe kommen kann?
- Ja. Zum Besipiel *bin packing*

Kompetetiv - Definition

Eine Strategie ist kompetetiv mit dem Faktor C, wenn:

$$K_s(P) \leq C * K_{opt}(P) + A \text{ mit: } C \geq 1 \text{ und } C, A \in \mathbb{R}$$

mit:

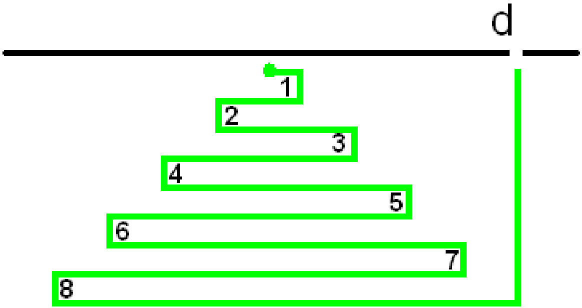
- einem Problem Π,
- einer Ausprägung $P \in \Pi$ des Problems,
- eine Strategie S, welche alle P löst,
- den Kosten $K_s(P)$ eines Problems P durch S und
- den optimalen Kosten $K_{opt}(P)$ von P.

Suche nach Tür in Wand

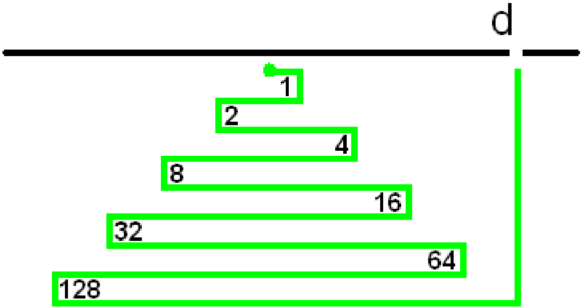
Problem:

- Tür in unendlich langer Wand
- Roboter sucht diese Tür
- Gesucht: kompetetive Strategie

Suche nach Tür in Wand - Inkrementell



Suche nach Tür in Wand - Verdoppelung



Exponentielle Vergrößerung der Suchtiefe

- Viele ähnliche Probleme
- Grund: $2^0 + 2^1 + 2^2 \ldots + 2^n < 2^{n+1}$

m Halbgraden:

- Zyklische Suche
- Strategie: $f_j = \left(\frac{m}{m-1}\right)^j$, kompetetiv mit $2em + 1$
- Hängt von m ab

Warum Verallgemeinerung?

Beispiel: Roboter in Polygon.

- Rand abfahren nicht kompetetiv $\Rightarrow$ Sichtbarkeitsgraph
- Weglänge hängt von Anzahl der Kanten ab

Verbesserungen

- Nicht jedes Mal zur Wurzel zurückkehren
- Abkürzungen bei Rückwegen nehmen
- Kanten zu Blättern auslassen

Außerdem:

- Beliebige Umgebungen
- N Dimensionen möglich

Applet

http://www.geometrylab.de/Pledge/

Danke für die Aufmerksamkeit.