

# Funktionsprinzipien und Anwendung von Algorithmen zur Pfadplanung

Bernardo Barcia, Simon Deutscher, Felix Kalchschmid

Hochschule Trier

2. Juli 2020

## Grundlagen

- Was ist Pfadplanung?
- Darstellung des Raums
- Graphen

## Algorithmen

- Definition von Pfadplanungsalgorithmen
- Diskrete Pfadplanung:  
Beispiel Feasible Planning
- kontinuierliche Pfadplanung

## Anwendungen

- Rubik's Cube Rätsel
- Videospiele
- Digitales Planen von Fabrikrobotern
- Autonomes Fahren

# Was ist Pfadplanung?

## Eine Einführung

- Pfadplanung ist die Spezifizierung einer Aufgabe in einer Hochsprache, die ein Roboter verstehen und ausführen soll.
- Ein *Roboter*, *Agent* oder *Player* ist der Nutzer eines Plans, mit dem er Entscheidungen trifft.
  - Roboter sollen seiner Umgebung verstehen
  - Roboter haben Bewegungseinschränkungen



Abbildung 1: Ein Roboterarm

# Was ist Pfadplanung?

## Eine Einführung

- Pfadplanung ist die Spezifizierung einer Aufgabe in einer Hochsprache, die ein Roboter verstehen und ausführen soll.
- Ein *Roboter*, *Agent* oder *Player* ist der Nutzer eines Plans, mit dem er Entscheidungen trifft.
  - Roboter sollen seiner Umgebung verstehen
  - Roboter haben Bewegungseinschränkungen



Abbildung 1: Ein Roboterarm

- Der Arbeitsraum ist die Umgebung, in der sich der Roboter befindet
- Der Konfigurationsraum ist die Menge aller seiner Konfigurationen
  - Konfiguration ist die Position des Roboters in einer für ihn verständlichen Sprache
  - Hindernisse sind ungültige Konfigurationen
  - Freier Raum ist die Gruppe von Konfigurationen ohne Kollisionen

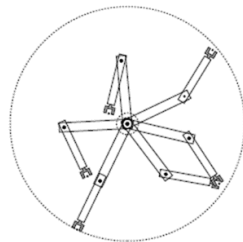


Abbildung 2: Konfigurationsraum eines Roboterarms

# Darstellung des Raums

## Arbeitsraum und Konfigurationsraum

- Der Arbeitsraum ist die Umgebung, in der sich der Roboter befindet
- Der Konfigurationsraum ist die Menge aller seiner Konfigurationen
  - Konfiguration ist die Position des Roboters in einer für ihn verständlichen Sprache
  - Hindernisse sind ungültige Konfigurationen
  - Freier Raum ist die Gruppe von Konfigurationen ohne Kollisionen

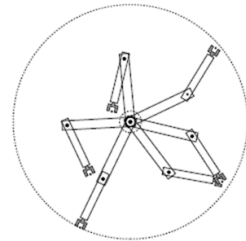


Abbildung 2: Konfigurationsraum eines Roboterarms

- Der Arbeitsraum ist die Umgebung, in der sich der Roboter befindet
- Der Konfigurationsraum ist die Menge aller seiner Konfigurationen
  - Konfiguration ist die Position des Roboters in einer für ihn verständlichen Sprache
  - Hindernisse sind ungültige Konfigurationen
  - Freier Raum ist die Gruppe von Konfigurationen ohne Kollisionen

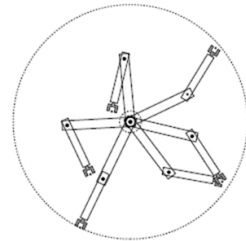
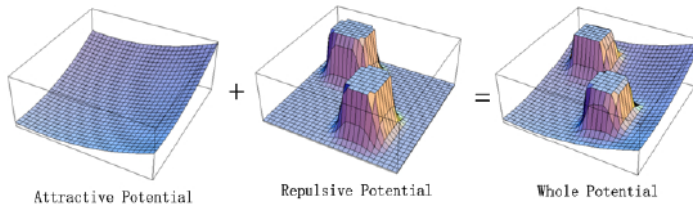


Abbildung 2: Konfigurationsraum eines Roboterarms

# Darstellung des Raums

## Potentialfeld

- Punkte in dem Konfigurationsraum bekommen eine Potentialvektor
  - Sie werden von Hindernisse abgestoßen
  - Sie werden vom Ziel angezogen





- Die Punkte des Konfigurationsraum verbinden, um eine Roadmap zu erzeugen
- Bewegung in Roadmap:
  - Roadmap betreten
  - Roadmap durchlaufen
  - Ziel erreichen
- Sichtbarkeitsgraphen sind Roadmaps

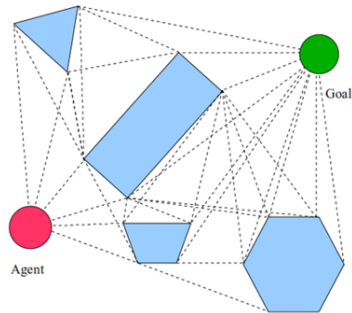


Abbildung 3: Sichtbarkeitsgraph

- Teilung des freien Raums in Zellen
  - Verbindungen zwischen anliegenden Zellen speichern
- Zellen enthalten Start- und Endkonfigurationen bei der Pfadfindung.

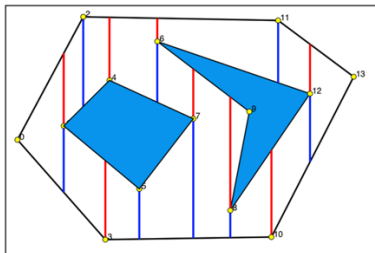
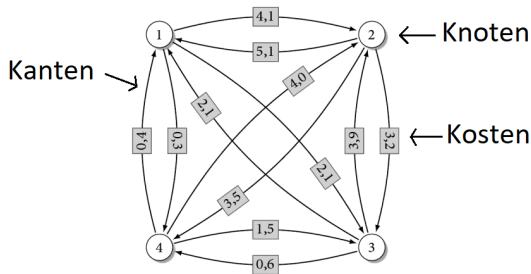
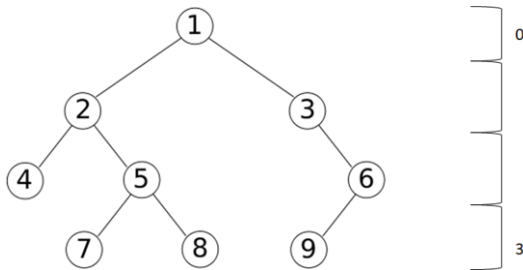


Abbildung 4: Zelldekomposition mit Polygonalen Hindernissen

- Graphen sind gut für Pfadfindung
- Ein Graph besteht aus Knoten, Kanten mit Kosten
- Ein gerichteter Graph unterscheidet ausgehende und eingehende Kanten



- Hierarchische Darstellung von Graphen
- Ein Baum hat ein Wurzel
- Die Entfernung von der Wurzel ist die Tiefe des Knotens



- Navigationsnetze definieren der freie Raum als polygonale Bereiche
- Verbindungen zwischen Bereichen sind im Navigationsgraph gespeichert
- Start und Endkonfigurationen befinden sich in den Bereichen

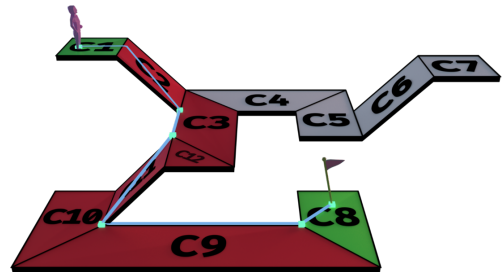


Abbildung 5: Navigationsnetz mit drei Ebenen

- Der Konfigurationsraum wird in Feldern definiert
  - Felder können verschiedene Formen haben
- Die Felder kennen ihre Nachbarn

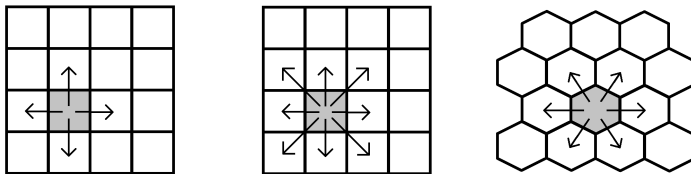


Abbildung 6: Drei verschiedene Rastergraphen

- Roadmap aus sichtbaren Punkten bilden
- Die Eckpunkte der Hindernisse verbinden
- Sensoren nehmen neue Information für den Roboter wahr, um neue Kanten zu bilden

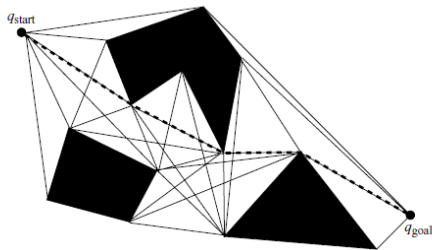


Abbildung 7: Sichtbarkeitsgraph mit polygonalen Hindernissen

## Grundlagen

- Was ist Pfadplanung?
- Darstellung des Raums
- Graphen

## Algorithmen

- Definition von Pfadplanungsalgorithmen
- Diskrete Pfadplanung:  
Beispiel Feasible Planning
- kontinuierliche Pfadplanung

## Anwendungen

- Rubik's Cube Rätsel
- Videospiele
- Digitales Planen von Fabrikrobotern
- Autonomes Fahren

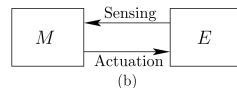
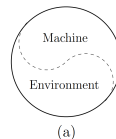


- Definition mit Hilfe der *Church-Turing-These*

### Church-Turing-These

Jede im intuitiven Sinne berechenbare Funktion ist **WHILE**-berechenbar.

- Diese Definitionen ist Unvollständig
- Ein Planer ist ein Algorithmus der einen Plan erstellt
- Ein Plan kann so die Interaktion abbilden



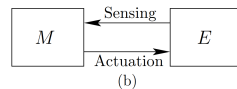
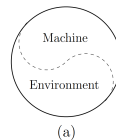
**Abbildung 8:** (a) Grenze zwischen Maschine und Umgebung ist fließend. (b) Maschine  $M$  interagiert mit der Umgebung  $E$  durch Sensorik und Antrieb.

- Definition mit Hilfe der *Church-Turing-These*

### Church-Turing-These

Jede im intuitiven Sinne berechenbare Funktion ist **WHILE**-berechenbar.

- Diese Definitionen ist Unvollständig
- Ein Planer ist ein Algorithmus der einen Plan erstellt
- Ein Plan kann so die Interaktion abbilden



**Abbildung 8:** (a) Grenze zwischen Maschine und Umgebung ist fließend. (b) Maschine  $M$  interagiert mit der Umgebung  $E$  durch Sensorik und Antrieb.

### Ausführen

- mit kodierter Eingabe den Roboter programmieren
- Spezialmaschine für Aufgabe erstellen

### Verbesserung

- Den Plan dem Planer übergeben.
- Neuberechnung unter Beachtung anderer Aspekte
- Verbesserungsprozess führt zum gewünschten Plan

### Hierarchische Inklusion

- Plan als Teilstück eines *Masterplan*
- Zerteilung des Problems zur leichteren Berechnung
- Eine Baumstruktur entsteht
- Mehrere Verschiebeoperationen für die Lösung erforderlich.

### Ausführen

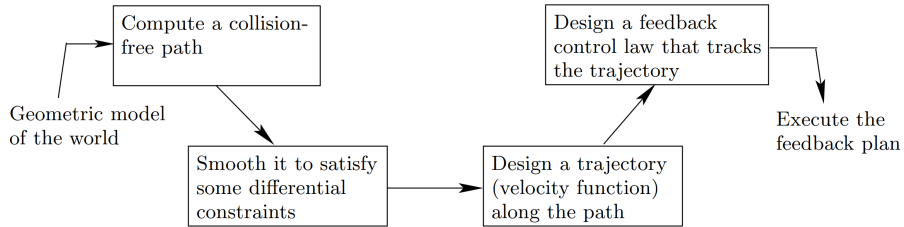
- mit kodierter Eingabe den Roboter programmieren
- Spezialmaschine für Aufgabe erstellen

### Verbesserung

- Den Plan dem Planer übergeben.
- Neuberechnung unter Beachtung anderer Aspekte
- Verbesserungsprozess führt zum gewünschten Plan

### Hierarchische Inklusion

- Plan als Teilstück eines *Masterplan*
- Zerteilung des Problems zur leichteren Berechnung
- Eine Baumstruktur entsteht
- Mehrere Verschiebeoperationen für die Lösung erforderlich.



**Abbildung 9:** Ein Verbesserungsprozess der sich in der Robotik bewährt hat.

### Ausführen

- mit kodierter Eingabe den Roboter programmieren
- Spezialmaschine für Aufgabe erstellen

### Verbesserung

- Den Plan dem Planer übergeben.
- Neuberechnung unter Beachtung anderer Aspekte
- Verbesserungsprozess führt zum gewünschten Plan

### Hierarchische Inklusion

- Plan als Teilstück eines *Masterplan*
- Zerteilung des Problems zur leichteren Berechnung
- Eine Baumstruktur entsteht
- Mehrere Verschiebeoperationen für die Lösung erforderlich.

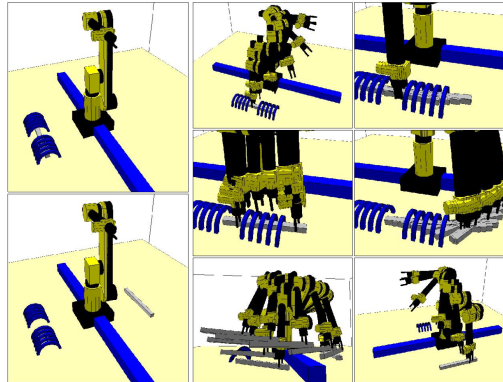


Abbildung 9: Roboter bewegt Stab von der Ausgangsposition zur Endposition

### diskret

- Zustandsraum ist abzählbar unendlich
- Geometrische Modelle finden keine Beachtung
- keine Bewegungseinschränkungen
- Systematisch: wenn ein Pfad existiert wird er gefunden

### kontinuierlich

- Alle Umgebungsinformationen (Industrieroboter, kontrollierte Umgebung)
- Unsicherheit (Roboter mit Kameras)
- Bewegungseinschränkung (Roboter mit Rädern, Auto)
- Vorherrschaft von diskreten Algorithmen auch für kontinuierliches Pfadplanen.



### diskret

- Zustandsraum ist abzählbar unendlich
- Geometrische Modelle finden keine Beachtung
- keine Bewegungseinschränkungen
- Systematisch: wenn ein Pfad existiert wird er gefunden

### kontinuierlich

- Alle Umgebungsinformationen (Industrieroboter, kontrollierte Umgebung)
- Unsicherheit (Roboter mit Kameras)
- Bewegungseinschränkung (Roboter mit Rädern, Auto)
- Vorherrschaft von diskreten Algorithmen auch für kontinuierliches Pfadplanen.

- Feasible Planning: ein Pfad zum Ziel wird gefunden
- Optimales Planen: Optimierung auf Zeit, Distanz, Anzahl Drehungen, usw.
- Die Liste  $Q$  enthält den aktuellen Knoten
- Knoten  $x_I$  ist als Startknoten gegeben
- Die Menge  $U(x)$  enthält alle unentdeckten Knoten
- Die Menge  $X_G$  enthält alle Zielzustände
- Vorgängerknoten müssen gespeichert werden

---

### FORWARD\_SEARCH

```
1   $Q.Insert(x_I)$  and mark  $x_I$  as visited
2  while  $Q$  not empty do
3       $x \leftarrow Q.GetFirst()$ 
4      if  $x \in X_G$ 
5          return SUCCESS
6      forall  $u \in U(x)$ 
7           $x' \leftarrow f(x, u)$ 
8          if  $x'$  not visited
9              Mark  $x'$  as visited
10              $Q.Insert(x')$ 
11          else
12              Resolve duplicate  $x'$ 
13  return FAILURE
```

---

- Feasible Planning: ein Pfad zum Ziel wird gefunden
- Optimales Planen: Optimierung auf Zeit, Distanz, Anzahl Drehungen, usw.
- Die Liste  $Q$  enthält den aktuellen Knoten
- Knoten  $X_I$  ist als Startknoten gegeben
- Die Menge  $U(x)$  enthält alle unentdeckten Knoten
- Die Menge  $X_G$  enthält alle Zielzustände
- Vorgängerknoten müssen gespeichert werden

---

### FORWARD\_SEARCH

```
1   $Q.Insert(x_I)$  and mark  $x_I$  as visited
2  while  $Q$  not empty do
3       $x \leftarrow Q.GetFirst()$ 
4      if  $x \in X_G$ 
5          return SUCCESS
6      forall  $u \in U(x)$ 
7           $x' \leftarrow f(x, u)$ 
8          if  $x'$  not visited
9              Mark  $x'$  as visited
10              $Q.Insert(x')$ 
11          else
12              Resolve duplicate  $x'$ 
13  return FAILURE
```

---

### Einfluss der Sortierung der Warteschlange

**Breadth-first Suchalgorithmus** verwendet FIFO Sortierung

- Systematisch, nicht zielgerichtet

**Depth-first Suchalgorithmus** verwendet LIFO Sortierung

- teilweise systematisch, aggressive expansion, schwer kontrollierbar

**Dijkstra-Algorithmus** bevorzugt kleine Kantengewichte

- optimal, unnötige schritte

**A\*-Algorithmus** schätzt Kosten zum Ziel

- optimal, zielgerichtet, schnell

- Unterschiede zur diskreten Pfadplanung
  - Der Zustandsraum ist überabzählbar unendlich
  - Der Zustandsraum ist zu groß, um explizit dargestellt zu werden
- Piano Mover's Problem
  - Ein Körper muss einen kontinuierlichen Bewegungspfad finden.
  - Geometrische Einschränkungen
  - kein Kontakt mit Hindernissen im Raum



Abbildung 10: mobile Roboter bewegen einen Flügel [?, Abb.1.5, S.9]

- Unterschiede zur diskreten Pfadplanung
  - Der Zustandsraum ist überabzählbar unendlich
  - Der Zustandsraum ist zu groß, um explizit dargestellt zu werden
- Piano Mover's Problem
  - Ein Körper muss einen kontinuierlichen Bewegungspfad finden.
  - Geometrische Einschränkungen
  - kein Kontakt mit Hindernissen im Raum



Abbildung 10: mobile Roboter bewegen einen Flügel [?, Abb.1.5, S.9]

- Unterschiede zur diskreten Pfadplanung
  - Der Zustandsraum ist überabzählbar unendlich
  - Der Zustandsraum ist zu groß, um explizit dargestellt zu werden
- Piano Mover's Problem
  - Ein Körper muss einen kontinuierlichen Bewegungspfad finden.
  - Geometrische Einschränkungen
  - kein Kontakt mit Hindernissen im Raum



Abbildung 10: mobile Roboter bewegen einen Flügel [?, Abb.1.5, S.9]

- Unterschiede zur diskreten Pfadplanung
  - Der Zustandsraum ist überabzählbar unendlich
  - Der Zustandsraum ist zu groß, um explizit dargestellt zu werden
- Piano Mover's Problem
  - Ein Körper muss einen kontinuierlichen Bewegungspfad finden.
  - Geometrische Einschränkungen
  - kein Kontakt mit Hindernissen im Raum



Abbildung 10: mobile Roboter bewegen einen Flügel [?, Abb.1.5, S.9]



### Zwei wichtige Motive in der Pfadplanung

#### Implizite Repräsentation des Zustandsraumes

- Explizite Darstellung des Zustandsraumes nicht möglich
- Daher Implizite Darstellung

#### Transformation von kontinuierlichen Modellen in diskrete

- combinatorial motion planning
- sampling based motion planning

### Zwei wichtige Motive in der Pfadplanung

#### Implizite Repräsentation des Zustandsraumes

- Explizite Darstellung des Zustandsraumes nicht möglich
- Daher Implizite Darstellung

#### Transformation von kontinuierlichen Modellen in diskrete

- combinatorial motion planning
- sampling based motion planning

### Zwei wichtige Motive in der Pfadplanung

#### Implizite Repräsentation des Zustandsraumes

- Explizite Darstellung des Zustandsraumes nicht möglich
- Daher Implizite Darstellung

#### Transformation von kontinuierlichen Modellen in diskrete

- combinatorial motion planning
- sampling based motion planning

## Grundlagen

- Was ist Pfadplanung?
- Darstellung des Raums
- Graphen

## Algorithmen

- Definition von Pfadplanungsalgorithmen
- Diskrete Pfadplanung: Beispiel Feasible Planning
- kontinuierliche Pfadplanung

## Anwendungen

- Rubik's Cube Rätsel
- Videospiele
- Digitales Planen von Fabrikrobotern
- Autonomes Fahren

# Anwendungen der Pfadplanung

## Das Rubic's Cube Rätsel

- Zustandsraum: Alle möglichen Farbverteilungen
- Aktionsrahmen: Die Menge aller möglichen Drehungen
- bestmöglicher Pfad: die geringste Anzahl an Drehungen, bis der Würfel einfarbige Seiten hat
- diskret

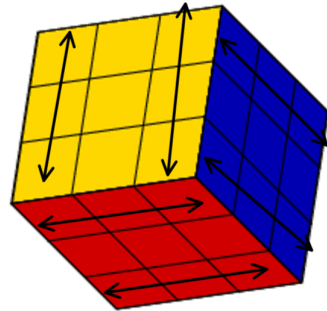


Abbildung 11: Der Rubik's Cube und sein Aktionsrahmen

# Anwendungen der Pfadplanung

## Das Rubic's Cube Rätsel

- Zustandsraum: Alle möglichen Farbverteilungen
- Aktionsrahmen: Die Menge aller möglichen Drehungen
- bestmöglicher Pfad: die geringste Anzahl an Drehungen, bis der Würfel einfarbige Seiten hat
- diskret

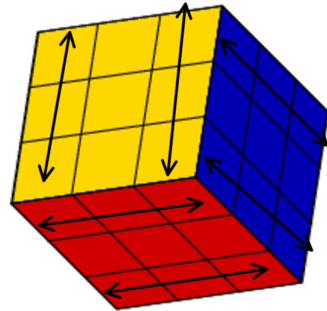


Abbildung 11: Der Rubik's Cube und sein Aktionsrahmen

# Anwendungen der Pfadplanung

## Das Rubic's Cube Rätsel

- Zustandsraum: Alle möglichen Farbverteilungen
- Aktionsrahmen: Die Menge aller möglichen Drehungen
- bestmöglicher Pfad: die geringste Anzahl an Drehungen, bis der Würfel einfarbige Seiten hat
- diskret

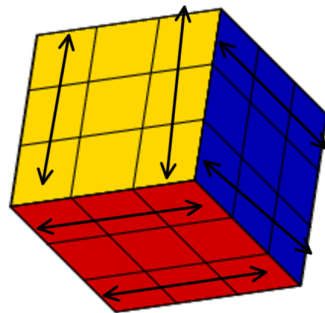


Abbildung 11: Der Rubik's Cube und sein Aktionsrahmen

# Anwendungen der Pfadplanung

## Das Rubic's Cube Rätsel

- Zustandsraum: Alle möglichen Farbverteilungen
- Aktionsrahmen: Die Menge aller möglichen Drehungen
- bestmöglicher Pfad: die geringste Anzahl an Drehungen, bis der Würfel einfarbige Seiten hat
- diskret

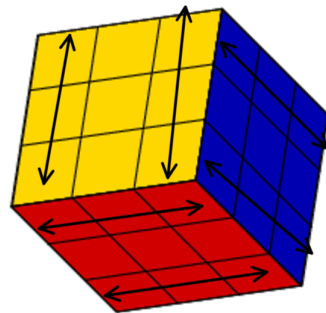


Abbildung 11: Der Rubik's Cube und sein Aktionsrahmen



# Anwendungen der Pfadplanung

## Videospiele

- Zustandsraum: Alle möglichen Positionen der zu bewegendenden Einheit und alle möglichen Map-Gegebenheiten
- Aktionsrahmen: alle möglichen Bewegungsrichtungen
- bestmöglicher Pfad: der schnellste Weg um Ziel
- diskret



Abbildung 12: Screenshot aus Civilisation V

- Zustandsraum: Alle möglichen Positionen der zu bewegendenden Einheit und alle möglichen Map-Gegebenheiten
- Aktionsrahmen: alle möglichen Bewegungsrichtungen
- bestmöglicher Pfad: der schnellste Weg um Ziel
- diskret



Abbildung 12: Screenshot aus Civilisation V

- Zustandsraum: Alle möglichen Positionen der zu bewegendenden Einheit und alle möglichen Map-Gegebenheiten
- Aktionsrahmen: alle möglichen Bewegungsrichtungen
- bestmöglicher Pfad: der schnellste Weg um Ziel
- diskret



Abbildung 12: Screenshot aus Civilisation V

- Zustandsraum: Alle möglichen Positionen der zu bewegendenden Einheit und alle möglichen Map-Gegebenheiten
- Aktionsrahmen: alle möglichen Bewegungsrichtungen
- bestmöglicher Pfad: der schnellste Weg um Ziel
- diskret



Abbildung 12: Screenshot aus Civilisation V

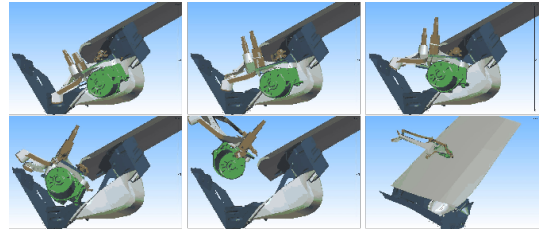
# Anwendungen der Pfadplanung

## Digitales Planen von Fabrikrobotern

Informatik  
Hauptcampus

H O C H  
S C H U L E  
T R I E R

- Zustandsraum: Alle möglichen Position der Fabrikroboter, des Autos und des zu montierenden Teils
- Aktionsrahmen: Die Menge aller möglichen Bewegungen des Roboters
- kontinuierlich
- Alle Umgebungsinformationen sind bekannt

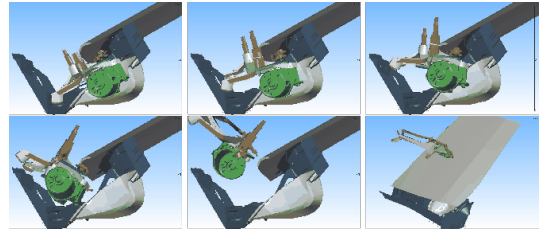


**Abbildung 13:** Scheibenwischer wird an Auto angebracht

# Anwendungen der Pfadplanung

## Digitales Planen von Fabrikrobotern

- Zustandsraum: Alle möglichen Position der Fabrikroboter, des Autos und des zu montierenden Teils
- Aktionsrahmen: Die Menge aller möglichen Bewegungen des Roboters
- kontinuierlich
- Alle Umgebungsinformationen sind bekannt



**Abbildung 13:** Scheibenwischer wird an Auto angebracht

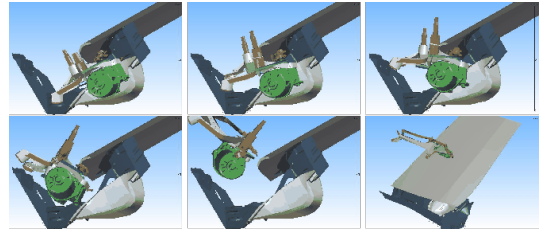
# Anwendungen der Pfadplanung

## Digitales Planen von Fabrikrobotern

Informatik  
Hauptcampus

H O C H  
S C H U L E  
T R I E R

- Zustandsraum: Alle möglichen Position der Fabrikroboter, des Autos und des zu montierenden Teils
- Aktionsrahmen: Die Menge aller möglichen Bewegungen des Roboters
- kontinuierlich
- Alle Umgebungsinformationen sind bekannt



**Abbildung 13:** Scheibenwischer wird an Auto angebracht

# Anwendungen der Pfadplanung

## Digitales Planen von Fabrikrobotern

Informatik  
Hauptcampus

H O C H  
S C H U L E  
T R I E R

- Zustandsraum: Alle möglichen Position der Fabrikroboter, des Autos und des zu montierenden Teils
- Aktionsrahmen: Die Menge aller möglichen Bewegungen des Roboters
- kontinuierlich
- Alle Umgebungsinformationen sind bekannt

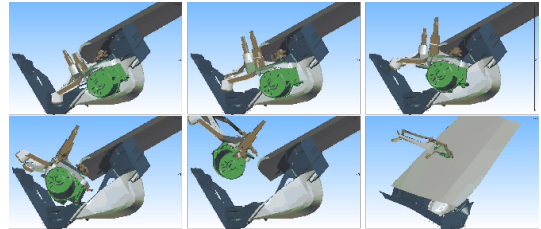


Abbildung 13: Scheibenwischer wird an Auto angebracht



- Zustandsraum: Jede mögliche Position auf der Welt und jede mögliche Umgebung
- Aktionsrahmen:
  - Beschleunigen
  - Bremsen
  - Lenken
  - blinken
  - Gang wechseln
- kontinuierlich
- Planung mit Ungewissheit
- Planung mit Bewegungseinschränkungen

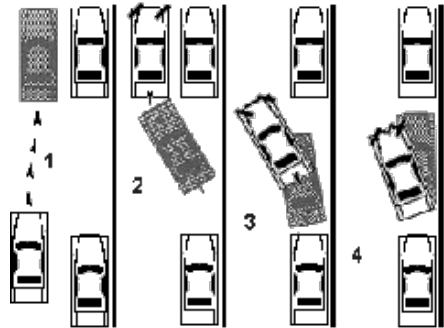


Abbildung 14: Autonomes Einparken

- Zustandsraum: Jede mögliche Position auf der Welt und jede mögliche Umgebung
- Aktionsrahmen:
  - Beschleunigen
  - Bremsen
  - Lenken
  - blinken
  - Gang wechseln
- kontinuierlich
- Planung mit Ungewissheit
- Planung mit Bewegungseinschränkungen

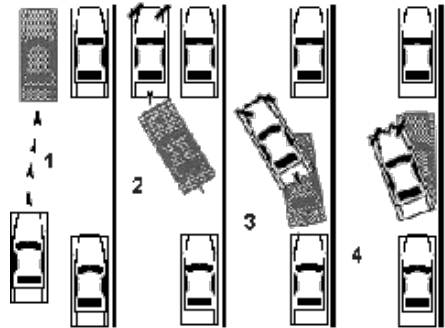


Abbildung 14: Autonomes Einparken

# Anwendungen der Pfadplanung

## Autonomes Fahren

- Zustandsraum: Jede mögliche Position auf der Welt und jede mögliche Umgebung
- Aktionsrahmen:
  - Beschleunigen
  - Bremsen
  - Lenken
  - blinken
  - Gang wechseln
- kontinuierlich
- Planung mit Ungewissheit
- Planung mit Bewegungseinschränkungen

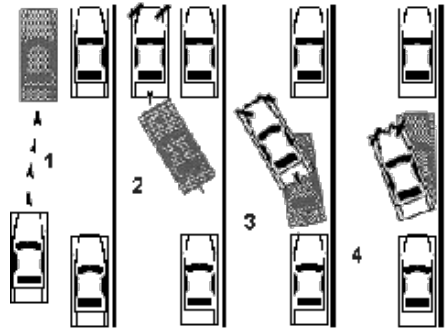


Abbildung 14: Autonomes Einparken

- Zustandsraum: Jede mögliche Position auf der Welt und jede mögliche Umgebung
- Aktionsrahmen:
  - Beschleunigen
  - Bremsen
  - Lenken
  - blinken
  - Gang wechseln
- kontinuierlich
- Planung mit Ungewissheit
- Planung mit Bewegungseinschränkungen

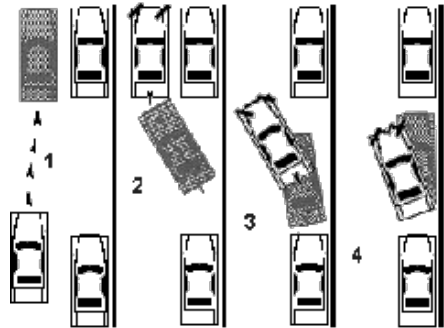


Abbildung 14: Autonomes Einparken

- Zustandsraum: Jede mögliche Position auf der Welt und jede mögliche Umgebung
- Aktionsrahmen:
  - Beschleunigen
  - Bremsen
  - Lenken
  - blinken
  - Gang wechseln
- kontinuierlich
- Planung mit Ungewissheit
- Planung mit Bewegungseinschränkungen

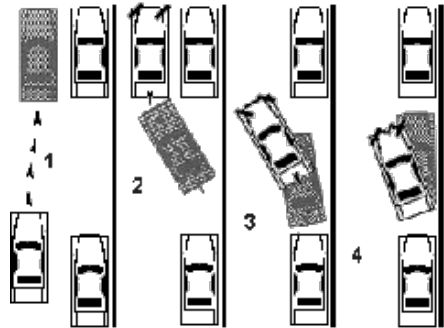


Abbildung 14: Autonomes Einparken

# Danke für eure Aufmerksamkeit!



Fragen gerne im Forum stellen!