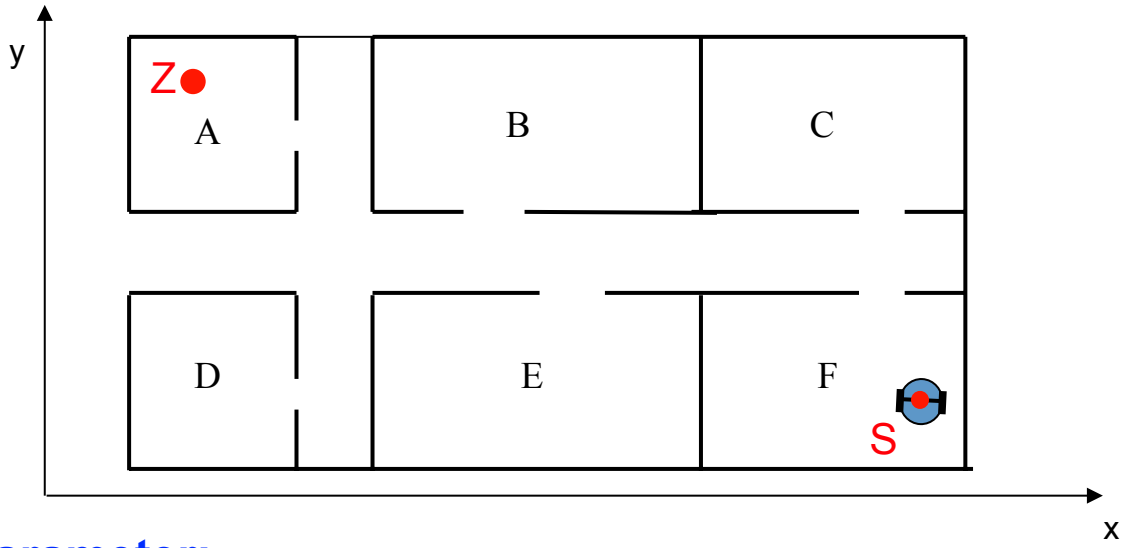
Navigation: Einführung und reaktive Navigation

- Einführung
  - Ziel, Architektur, Komponenten
- Reaktive Navigation
  - Braitenberg-Vehikel
  - Zusammenschalten verschiedener Navigationsverfahren
  - Histogramm-Verfahren
  - Dynamik-basiertes Verfahren

Problemstellung

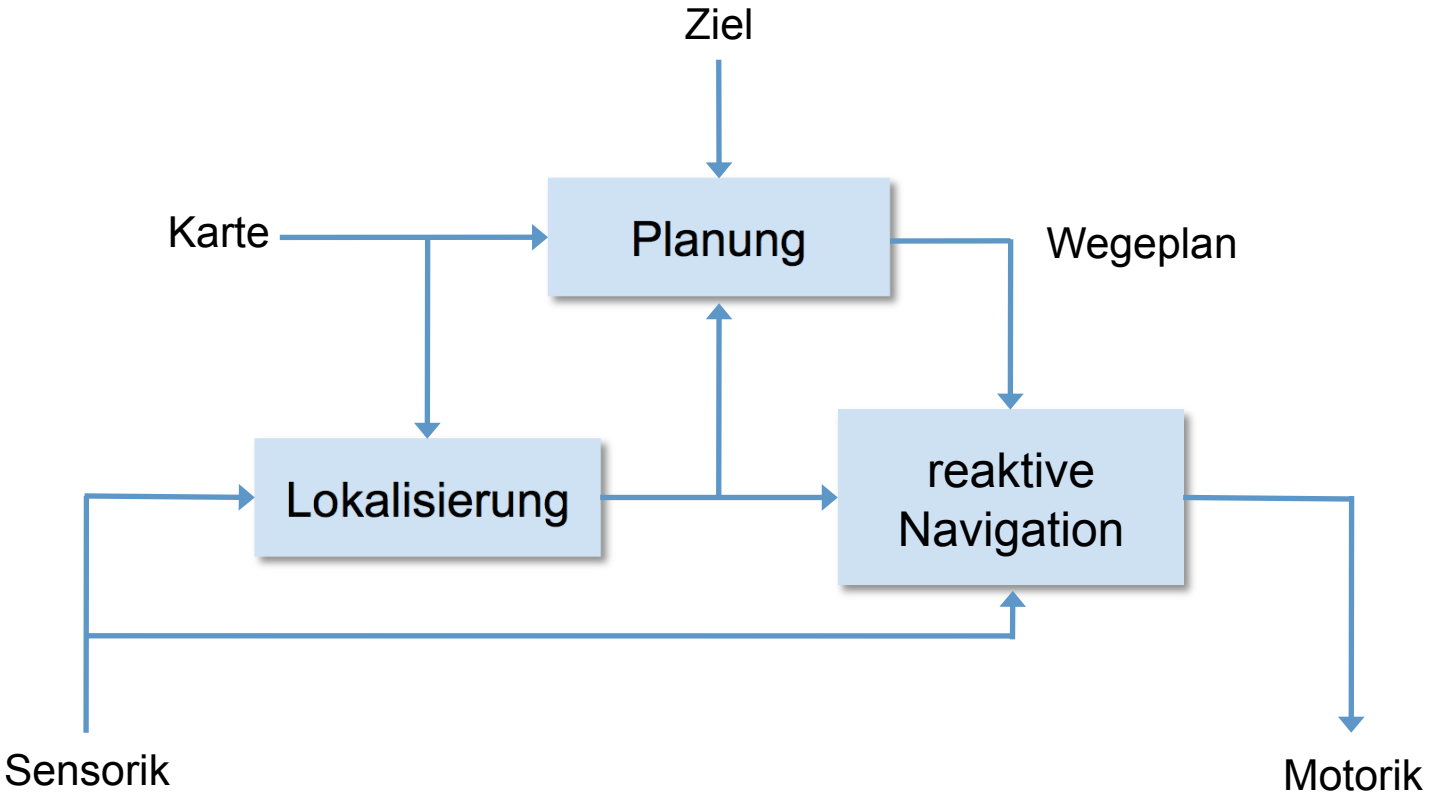
- Bewege Roboter kollisionsfrei von einem Startpunkt S zu einem Zielpunkt Z.



Problemparameter:

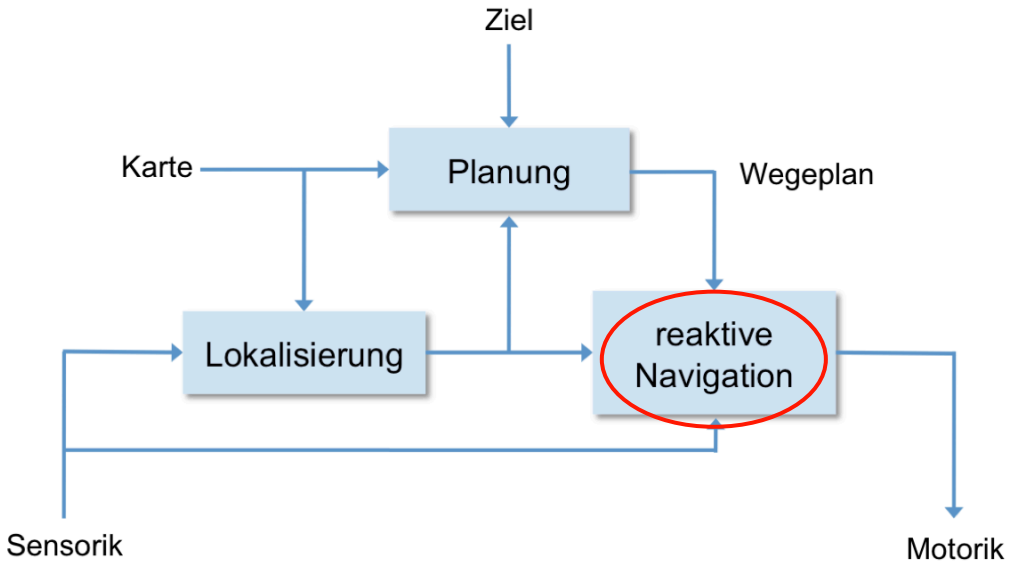
- Roboter:
  - Ausmaße und dynamische Eigenschaften
- Umgebung:
  - Beschaffenheit:
    In/Outdoor, Outdoor: künstlich/natürlich, Land/Wasser, ...
  - dynamische Hindernisse
  - Karte

Typische Architektur und Komponenten



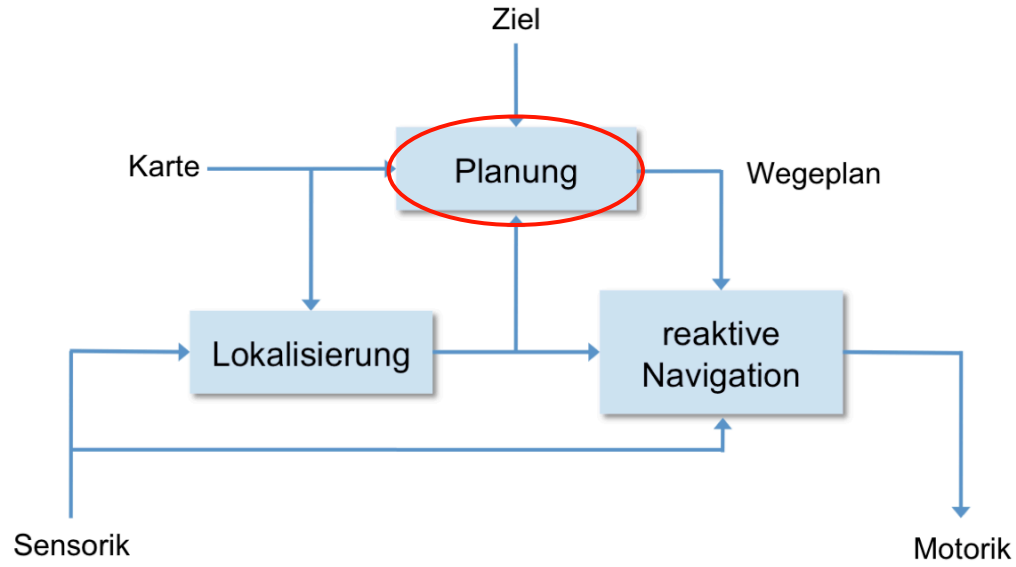
- Zweistufige Architektur:
  - Kartenbasierte Planung eines kollisionsfreien Wegs
  - Ausführen des geplanten Wegs durch reaktive Navigation

Reaktive Navigation



- Führt direkte Steuerung des Roboters durch:
  z.B. durch Setzen der Translations- und Rotationsgeschwindigkeit.
- Sensordaten können gefiltert sein und/oder über ein zeitlich begrenztes Intervall integriert werden: lokale Umgebungskarte.
- Aktuelle Positionsschätzung kann mit einbezogen werden.
- Üblicherweise wird aus einem Wegeplan
  die nächste Zielkoordinate vorgegeben.
- 10 – 100 Hz

Planung



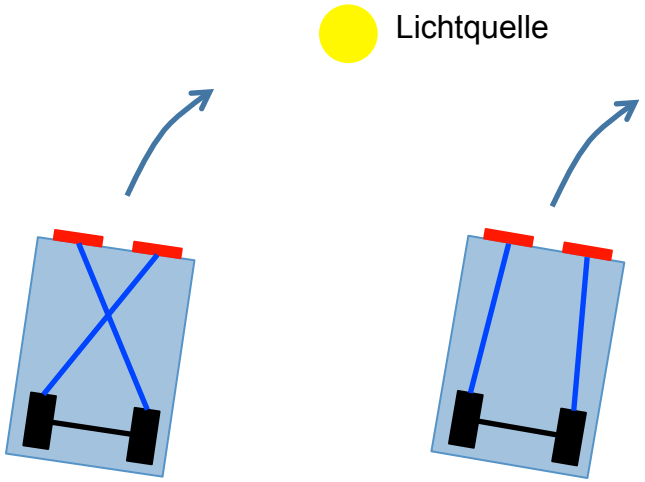
- Erstellung eines kollisionsfreien Wegeplans von der aktuellen Position zur Zielposition.
- Üblicherweise wird ein Plan mit Hilfe einer Karte erstellt.
- Dabei werden nur statische Hindernisse berücksichtigt.
- Meist wird Roboter als punktförmig angenommen
  (Hindernisse lassen sich um Roboterradius aufblasen)
- Meist keine Berücksichtigung der dynamischen Fähigkeiten des Roboters.
- Evtl. Unterstützung einer Planänderung (Sackgassenproblematik)
- ≤ 1 Hz

Navigation: Einführung und reaktive Navigation

- Einführung
  - Ziel, Architektur, Komponenten
- Reaktive Navigation
  - Braitenberg-Vehikel
  - Zusammenschalten verschiedener Navigationsverfahren
  - Histogramm-Verfahren
  - Dynamik-basiertes Verfahren

Braitenberg Vehikel

- Valentino Braitenberg: *Vehicles - Experiments in Synthetic Psychology,* 1984.
- Einfache Roboter die komplexes Verhalten realisieren sollen.
- Sensoren sind direkt mit Motoren gekoppelt
  (ganz allgemein gewichtete Summe)
- Einfache Sensorik:
  Sensoren liefern umgebungsabhängig einen skalaren Wert;
  z.B. Helligkeitssensor, Entfernungssensor, etc.



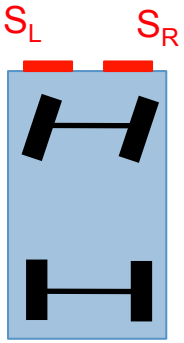
Aggressives Verhalten:
Roboter fährt auf
Lichtquelle zu

Ängstliches Verhalten:
Roboter fährt von
Lichtquelle weg

Simulation eines aggressiven Roboters (1)

- Roboter mit Ackermannsteuerung (Automobil);
  Steuerung der Geschwindigkeit v und des Lenkwinkels γ
- Helligkeitssensor misst Helligkeitswert, der umgekehrt proportional zum Abstand zur Lichtquelle (x_Q,y_Q) angenommen wird.
  Wert wird auf [0,1] normiert.

Lichtquelle Q

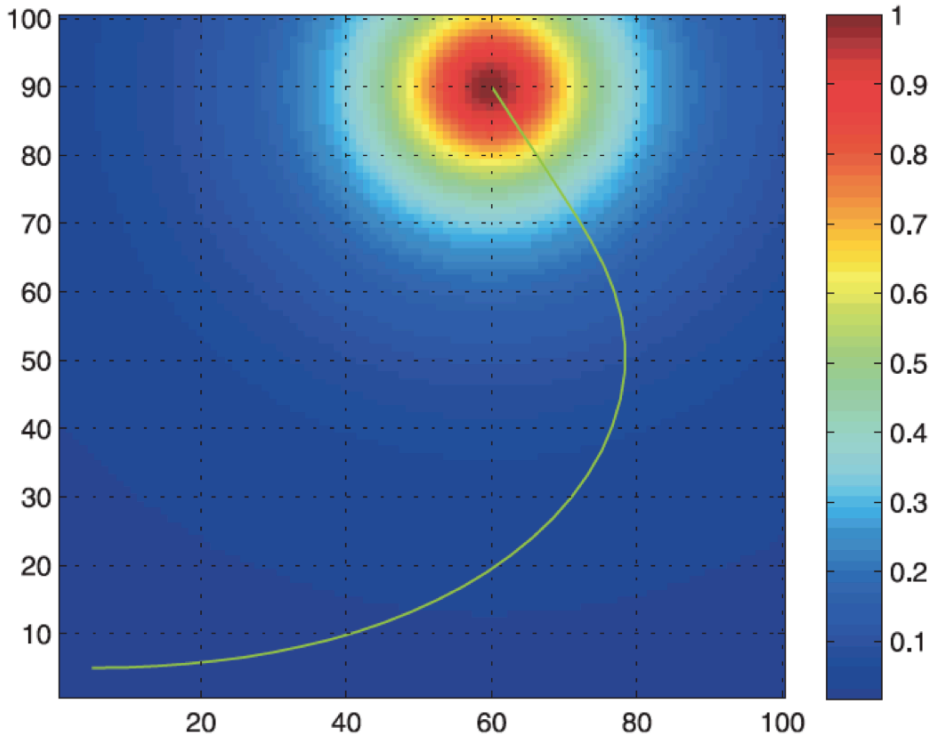


$$s = \frac{k_1}{(x - x_Q)^2 + (y - y_Q)^2 + k_1}$$

$$v = 2 - s_R - s_L$$

$$\gamma = k_2(s_R - s_L)$$

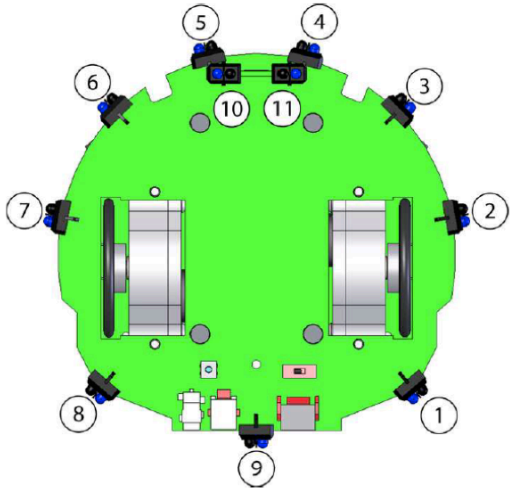
Simulation eines aggressiven Roboters (2)



- Lichtquelle bei Q = (60,90)
- Farbe stellt Entfernung zur Lichtquelle dar.

- Pfad des aggressiven Braitenberg-Vehikels, der zum Helligkeitsmaximum (= Position der Lichtquelle) führt.
- aus [Corke 2011]

Beispiel: Hindernisvermeidung mit Khepera III



IR-Sensoren s_1 bis s_9.

Linker Motor: $LM = \sum_i w_{l,i}\, s_i$

Rechter Motor: $RM = \sum_i w_{r,i}\, s_i$

IR-Sensor	1	2	3	4	5	6	7	8	9
$w_{l,i}$	2	-2	-4	-12	5	2	2	2	4
$w_{r,i}$	2	2	2	5	-12	-4	-2	2	4

rechts vorne

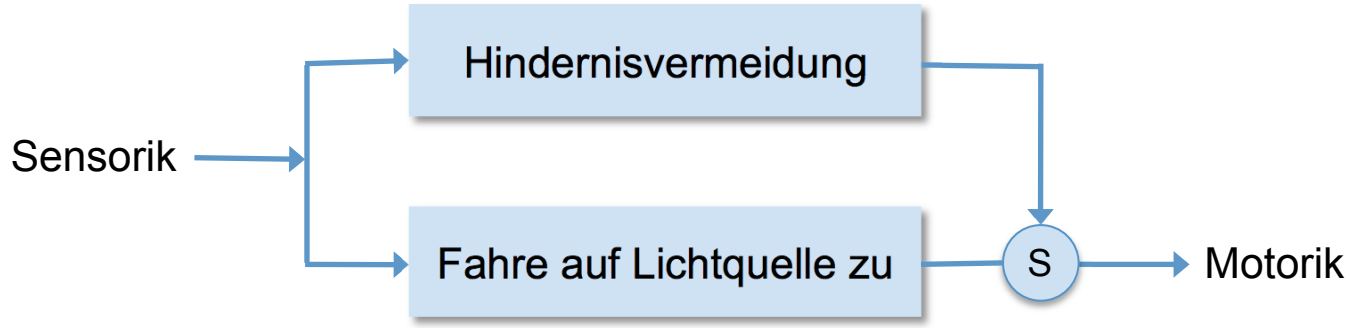
links vorne

Navigation: Einführung und reaktive Navigation

- Einführung
  - Ziel, Architektur, Komponenten
- Reaktive Navigation
  - Braitenberg-Vehikel
  - Zusammenschalten verschiedener Navigationsbausteine
  - Histogramm-Verfahren
  - Dynamik-basiertes Verfahren

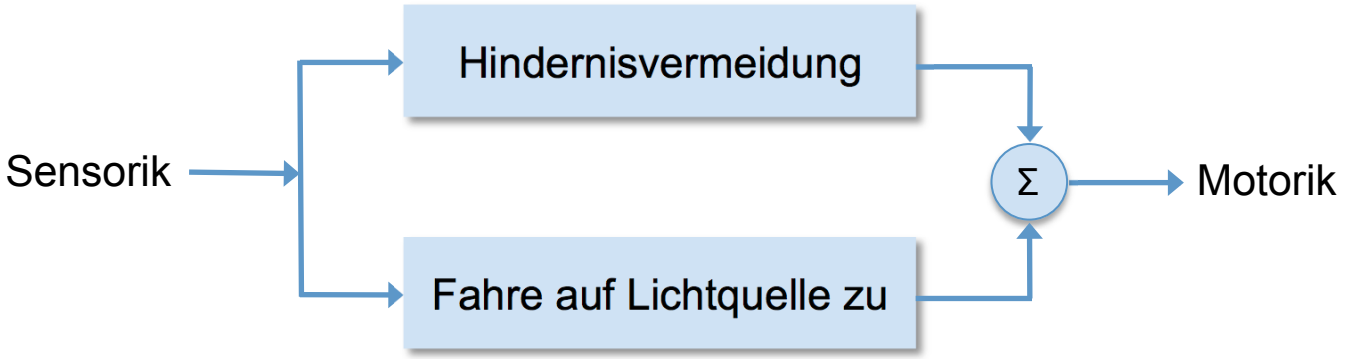
Subsumptionsarchitektur

- Mit einer Subsumptionsarchitektur lassen sich aus mehreren einfachen Verhalten (behaviour) ein komplexeres Verhalten erzielen.
- Dazu werden die Verhalten verschiedenen Ebenen zugeordnet. Die Verhalten der höheren Schichten können die Verhalten der unteren Schichten blockieren (suppress).
- Beispiel für zweischichtiges Verhalten:
  Der Roboter fährt auf die Lichtquelle zu, führt aber eine Hindernisvermeidung durch, sobald ein Hindernis auftaucht.



Fusionierung

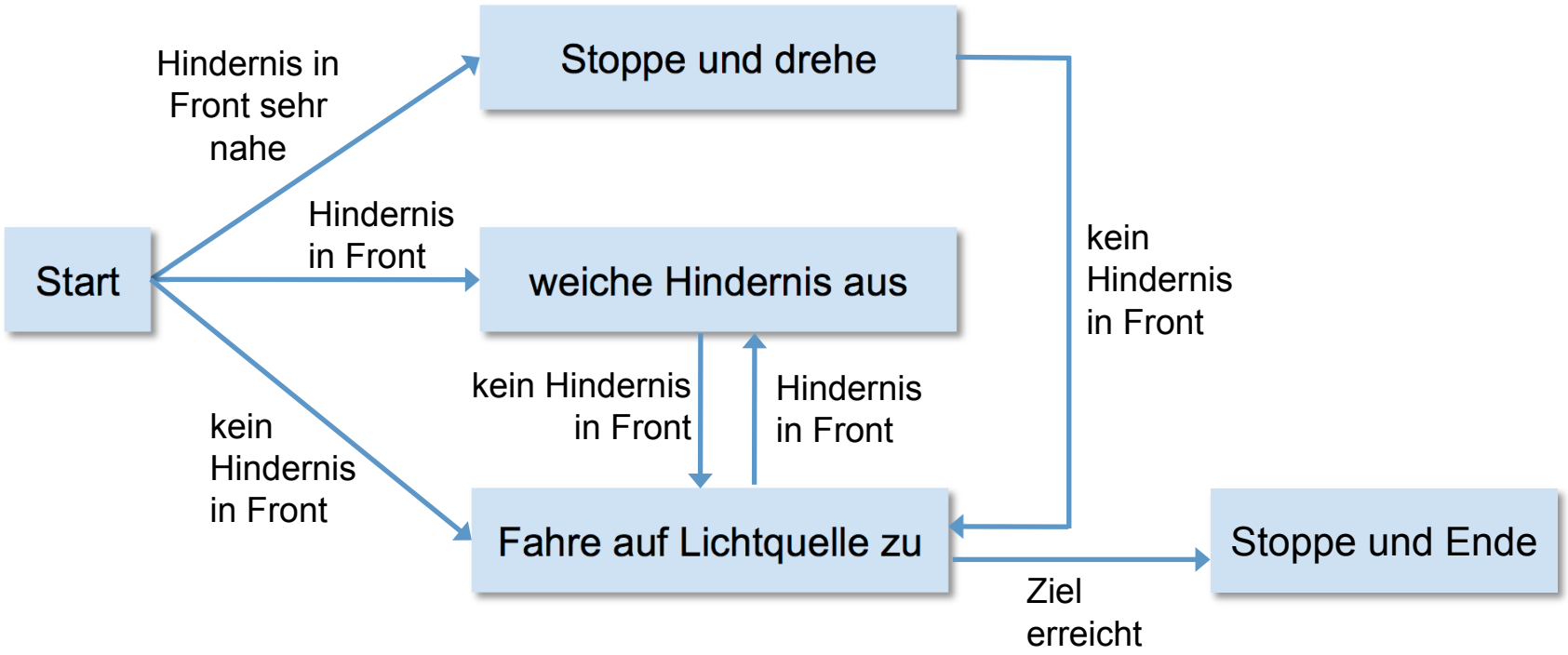
- Die Ausgaben mehrerer parallel laufender Verhalten werden durch eine gewichtete Summe zu einer Ausgabe fusioniert.



- Die Gewichte der Fusionierung kann von den Sensorwerten abhängen.

Endlicher Zustandsautomat

- Jedem Zustand wird ein Verhalten zugeordnet.
- Zustandswechsel wird über Sensorwerte gesteuert.

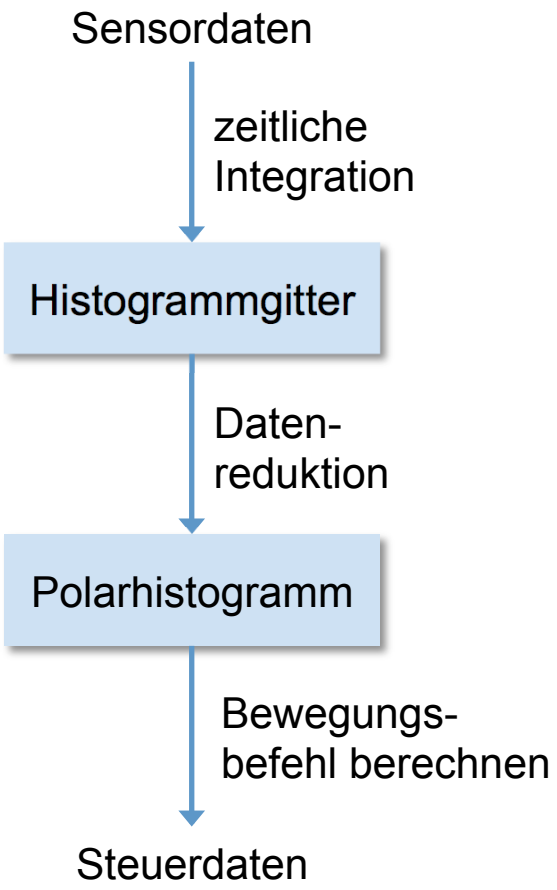


Navigation: Einführung und reaktive Navigation

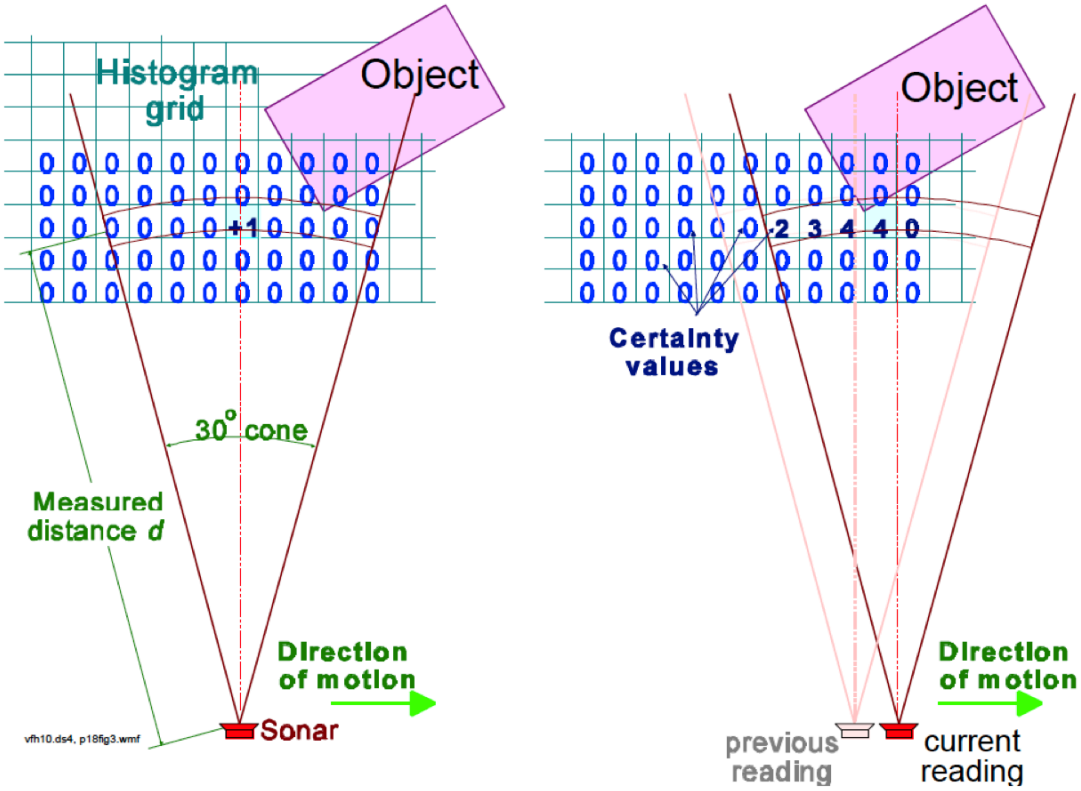
- Einführung
  - Ziel, Architektur, Komponenten
- Reaktive Navigation
  - Braitenberg-Vehikel
  - Zusammenschalten verschiedener Navigationsbausteine
  - Histogramm-Verfahren
  - Dynamik-basiertes Verfahren

Vector Field Histogram (VFH)

- Borenstein und Koren: VFH 1991.
- Weitere Verbesserungen durch
  Ulrich und Borenstein:
  VFH+ 1998 (Dynamik u. Größe des Roboters)
  VFH* 2000 (= VHF + A*)
- Aufbau einer lokalen Umgebungskarte (Histogrammgitter, histogram grid) durch zeitliche Integration von Sensor-Abstandsinformationen
- Aus der Umgebungskarte wird für die einzelnen Richtungen
  (bzgl. Roboter-Kooordinatensystem) die Hindernisdichte berechnet: Polarhistogramm
- Auf Basis des Polarhistogramms wird Bewegungsbefehl (v, ω) berechnet



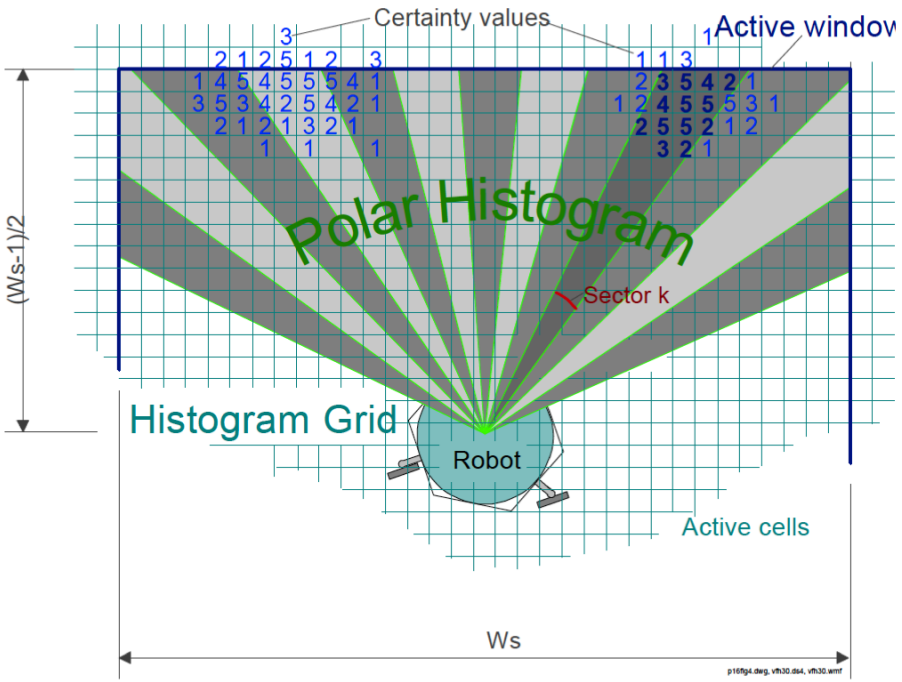
Histogrammgitter



Abbildungen (auch auf den folgenden Seiten) aus [Borenstein und Koren, 1991].

- Für jeden Sensor-Abstandswert wird die entsprechende Zelle des zweidimensionalen Histogrammgitters inkrementiert.
- Sensorwerte werden über einen längeren Zeitraum gesammelt.
- Histogrammgitter wird auf einen lokalen Bereich um den Roboter (z.B. 5m*5m) begrenzt mit einer Zellgröße von 0.1m.

Polar Histogram



- Jede Zelle i des Histogrammgitters hat einen Belegheitswert c_i
- Für jede belegte Zelle i werden Polarkoordinaten d_i, α_i (bzgl. Roboter-KS) berechnet.
- Daraus wird Einflussfaktor m_i bestimmt (a,b Konstante):

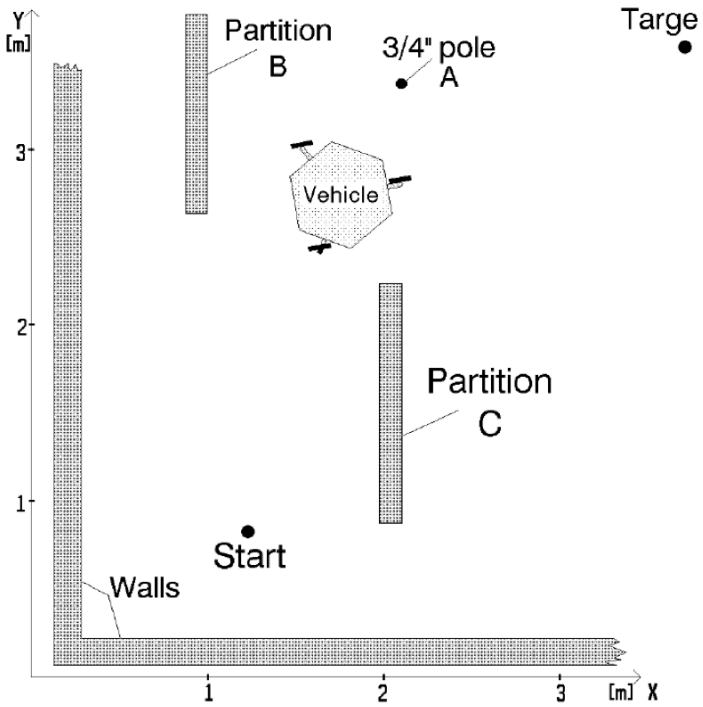
$$m_i = c_i^2(a - bd_i)$$

- Winkel werden in Sektoren s aufgeteilt (z.B. 72 Sektoren mit 5 Grad)
- Für jeden Sektor s wird Belegheitsdichte $h(s)$ ermittelt (Polar-Histogramm)

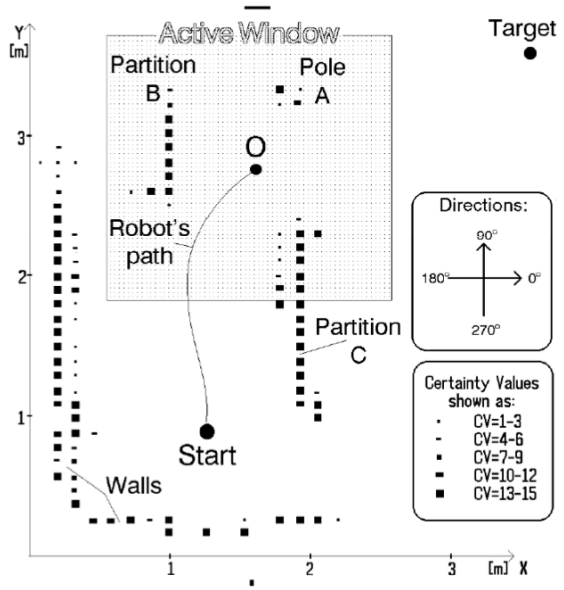
$$h(s) = \sum_{Zelle\ i\ in\ s} m_i$$

- Polarhistogramm wird noch geglättet

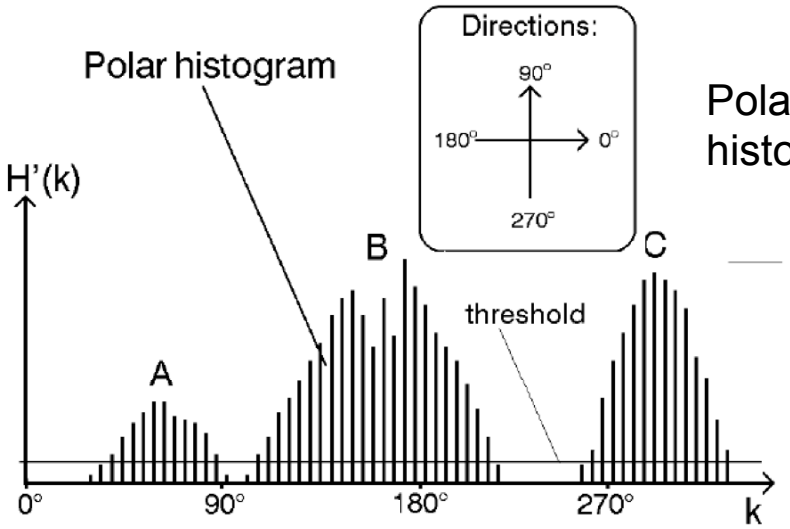
Typisches Szenario



Roboter und Umgebung



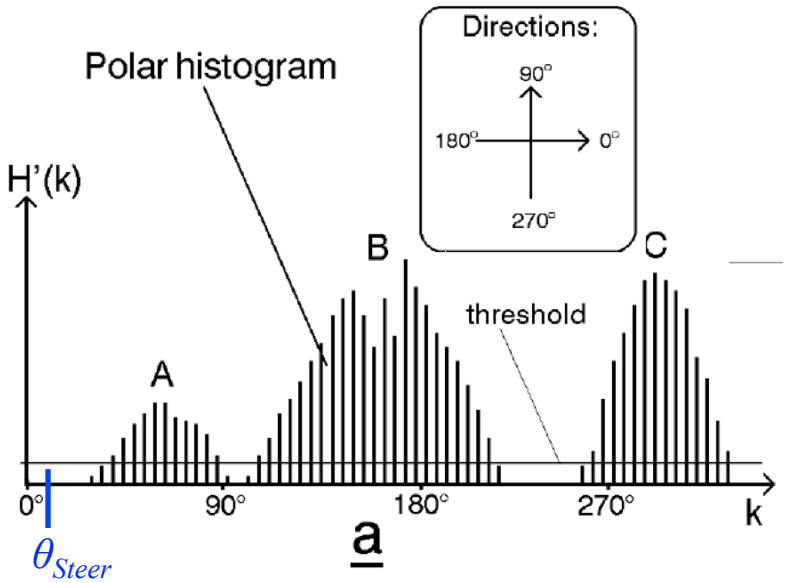
Histogramm-gitter

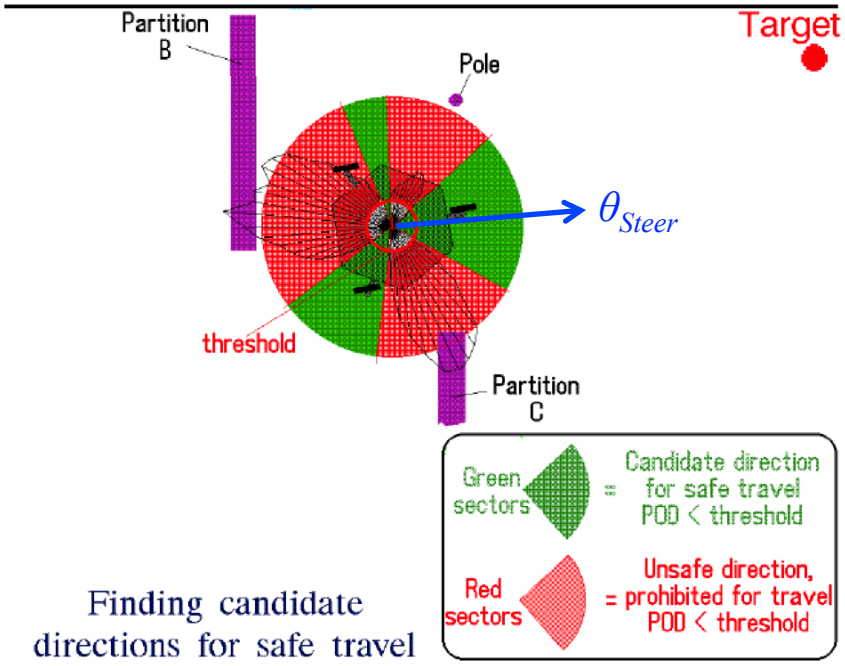


Polar-histogramm

Berechnung der Bewegungsrichtung

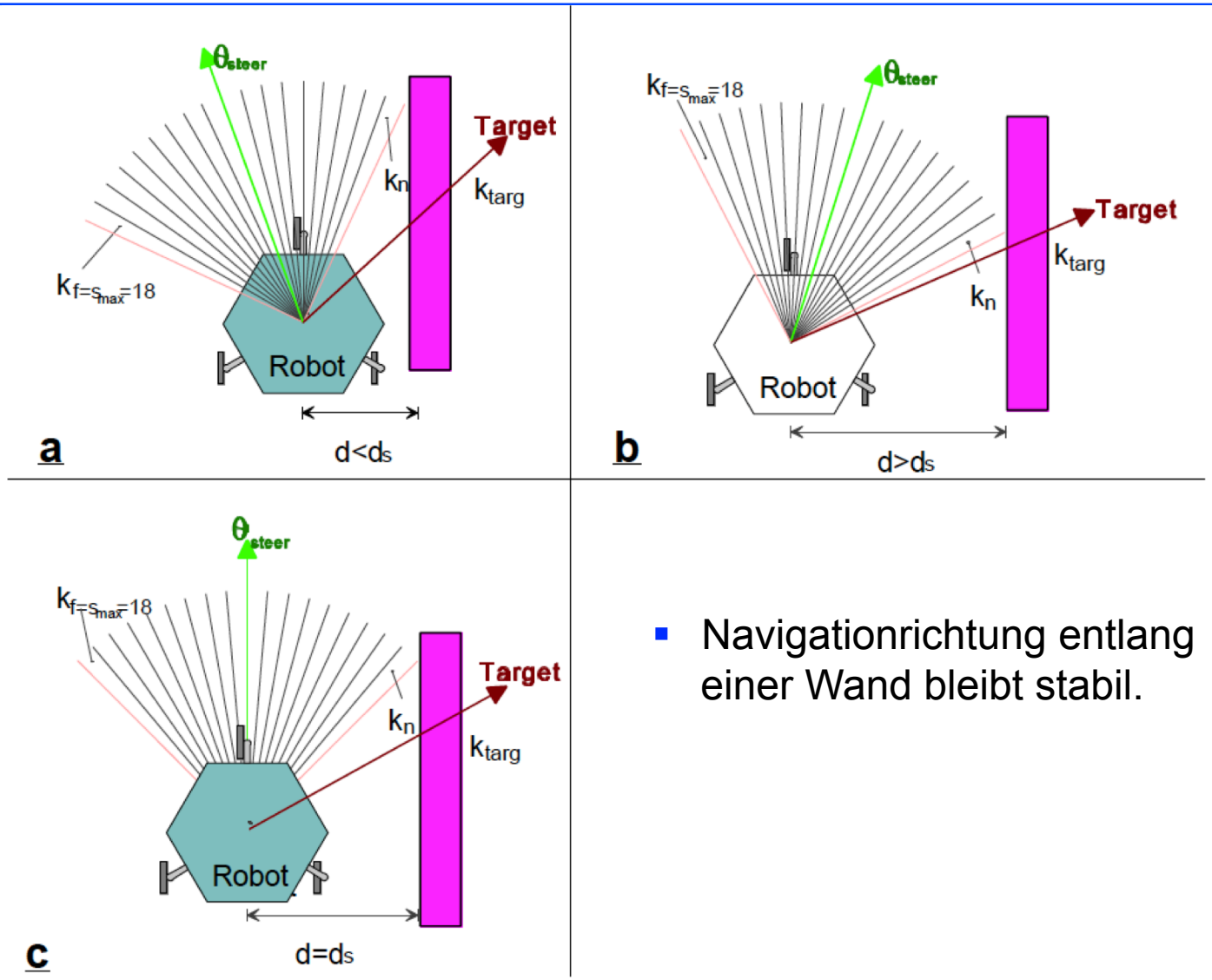
- Im Polarhistogramm werden alle Täler (lokale Minima) betrachtet, deren Histogrammwerte unterhalt einer bestimmten Schwelle (threshold) liegen.
- Es wird dajenige Tal gewählt, das am besten zur Ziel-Richtung passt.
- Als Bewegunsgrichtung θ_{Steer} wird die Talmitte gewählt.





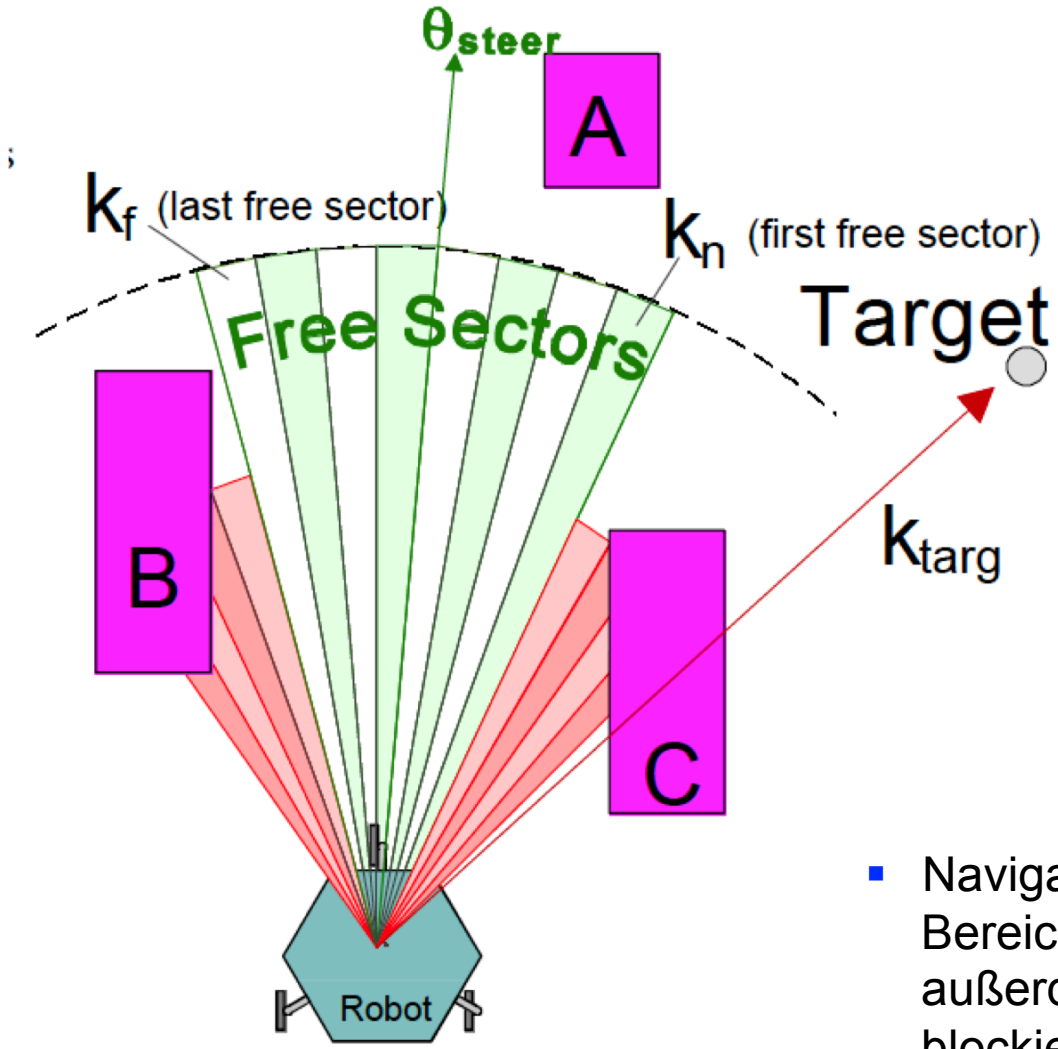
Finding candidate directions for safe travel

Berechnung der Bewegungsrichtung - Beispiel



- Navigationrichtung entlang einer Wand bleibt stabil.

Berechnung der Bewegungsrichtung - Beispiel



- Navigation in einem engen Bereich. Die Zielrichtung ist außerdem durch ein Hindernis blockiert.

Berechnung des Bewegungsbefehls

- Rotationsgeschwindigkeit ω wird proportional zur Abweichung zwischen tatsächlicher Ausrichtung θ und gewünschter Ausrichtung θ_{Steer} (begrenzt auf ω_{max}) gesetzt:

$$\omega = \min(\omega_{\max}, k * (\theta_{steer} - \theta))$$

- Translationsgeschwindigkeit v wird umgekehrt proportional zur Hindernisdichte $h(\theta_{Steer})$ in gewünschter Richtung und umgekehrt proportional zur Rotationsgeschwindigkeit ω gesetzt. Bei freier Geradeaus-Fahrt wird Maximalgeschwindigkeit gewählt.

$$v = v_{\max}\left(1 - \frac{h(\theta_{steer})}{h_0}\right)\left(1 - \frac{\omega}{\omega_{\max}}\right)$$

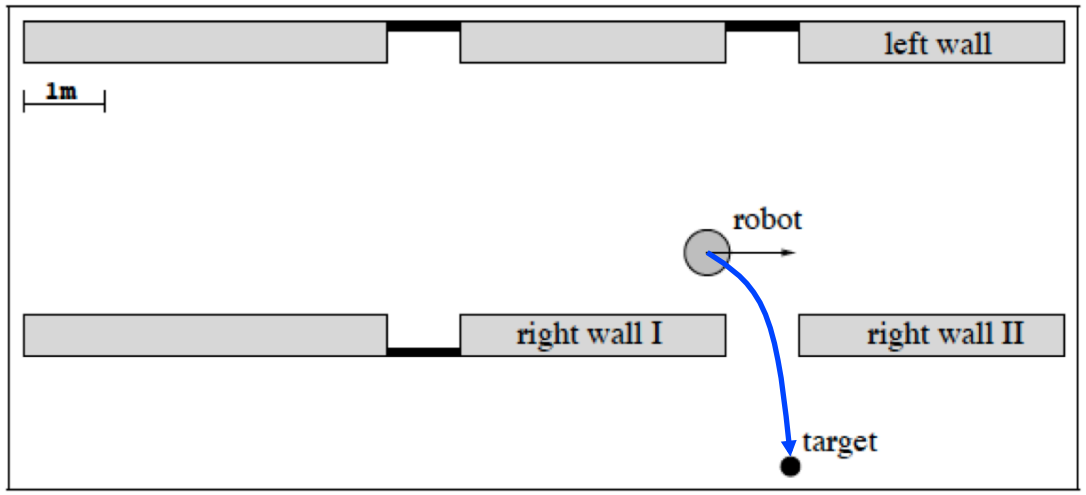
- Verfahren wurde im Praxiseinsatz mit ω_{max} = 120 Grad/sec und v_{max} = 0.8 m/sec erfolgreich getestet.

Navigation: Einführung und reaktive Navigation

- Einführung
  - Ziel, Architektur, Komponenten
- Reaktive Navigation
  - Braitenberg-Vehikel
  - Zusammenschalten verschiedener Navigationsbausteine
  - Histogramm-Verfahren
  - Dynamik-basiertes Verfahren

Dynamic Window Approach (DWA)

- Fox, Burgard und Thrun,
  The Dynamic Window Approach to Collision Avoidance, 1997.
- Auswahl des Steuerbefehls (v, ω) unter Berücksichtigung der dynamischen Fähigkeiten des Roboters (Beschleunigung und Winkelbeschleunigung)



alle Abbildungen auch auf den folgenden Seiten aus
[Fox, Burgard u. Thrun, 1997]

Nur bei bestimmten Geschwindigkeiten und dynamischen Fähigkeiten ist es sinnvoll, den Roboter direkt zum Ziel steuern

DWA-Verfahren

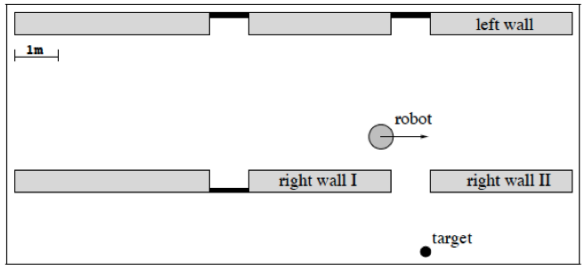
1. Festlegung des Suchraums für Zielfunktion:

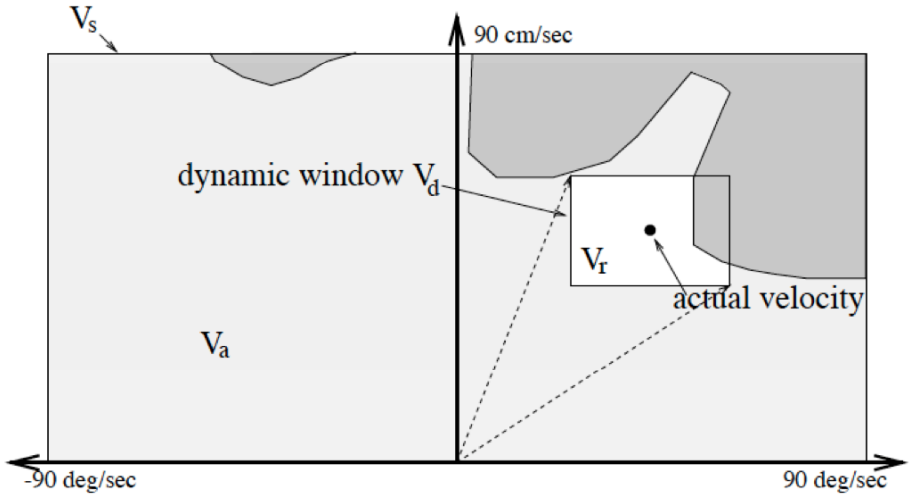
   Es werden alle Steuerbefehle (v, ω) betrachtet, die aufgrund der kinematischen Fähigkeiten des Roboters (Beschleunigung und Winkelbeschleunigung) möglich und aufgrund der Hindernisse erlaubt (kollisionsfrei) sind.

2. Auswahl des bestmöglichen Steuerbefehls (v, ω) mit einer dreiteiligen Zielfunktion:
   - gute Zielausrichtung
   - weite Entfernung von Hindernissen
   - hohe Geschwindigkeit

Suchraum

- Festlegung der kinematisch möglichen und der zulässigen (d.h. kollisionsfreien) Steuerbefehle *(v, ω)*





V_s = Bereich mit allen möglichen Steuerbefehlen.

V_d = kinematisch mögliche Steuerbefehle.

V_a = zulässige Steuerbefehle (d.h. kollisionsfrei).

- Suchraum für Zielfunktion:

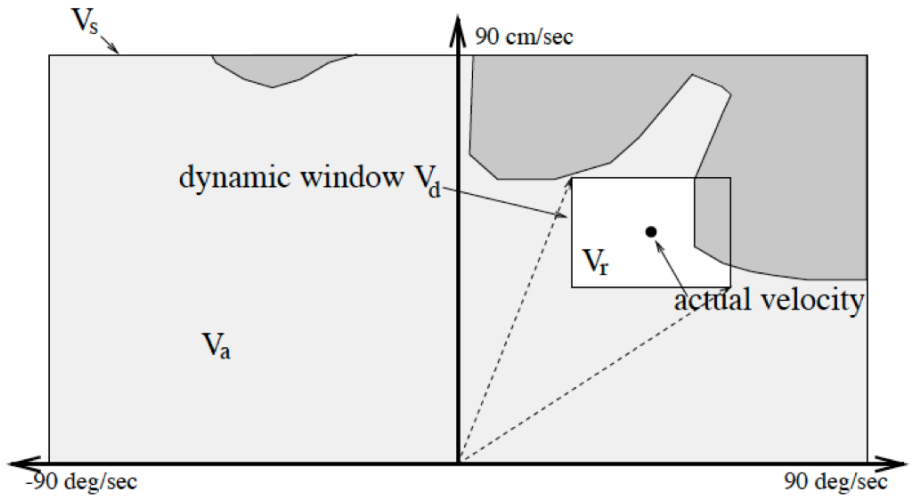
$$V_r = V_s \cap V_d \cap V_a$$

Kinematisch mögliche Steuerbefehle

- V_d umfasst alle Geschwindigkeiten, die im nächsten Zeitschritt T erreichbar sind.
- Dabei wird von einer maximal möglichen Beschleunigung a und Winkelbeschleunigung α des Roboters ausgegangen.

$$V_d = \{(v,\omega) / v_a - aT \le v \le v_a + aT \text{ und}$$
$$\omega_a - \alpha T \le \omega \le \omega_a + \alpha T\}$$

- Dabei ist v_a , ω_a die aktuelle Geschwindigkeit bzw. Winkelgeschwindigkeit.

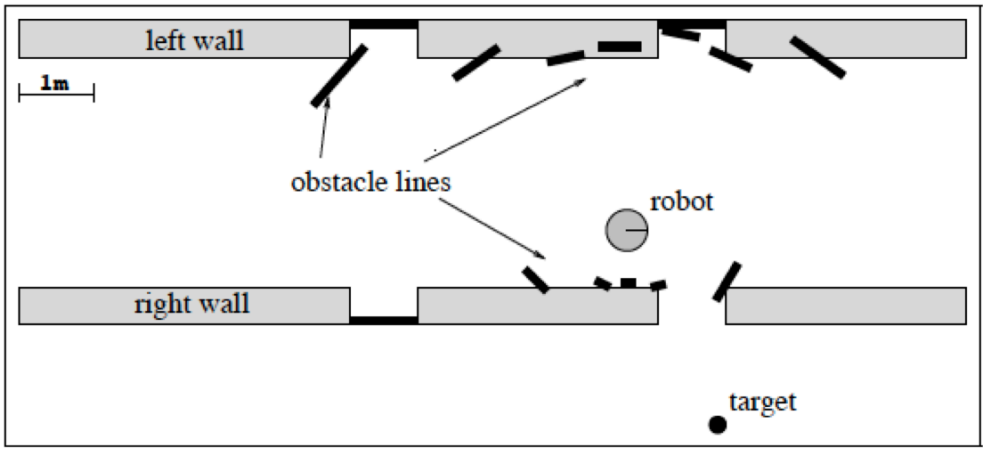


Zulässige Steuerbefehle

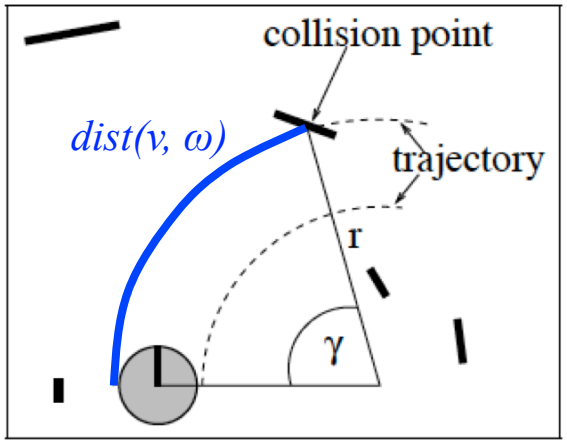
- Ein Steuerbefehl (v, ω) wird als zulässig definiert, wenn der Roboter beim Fahren des Kreisbogens, der durch (v, ω) definiert ist, rechtzeitig vor dem nächsten Hindernis bremsen kann.
- Dazu wird definiert:

  $dist(v, \omega)$ = Abstand zum nächsten Hindernis,
  beim Befahren des durch (v, ω) definierten Kreisbogens.

- Minimialer Bremsweg muss dann ≤ $dist(v, \omega)$ sein.

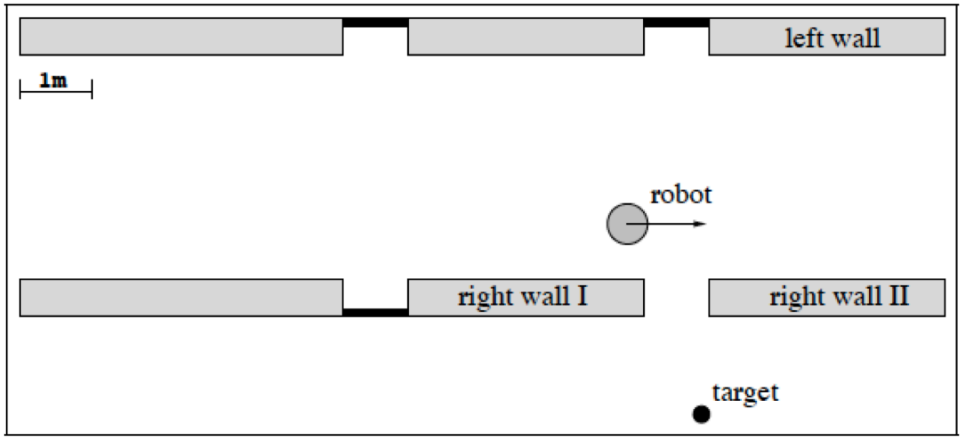


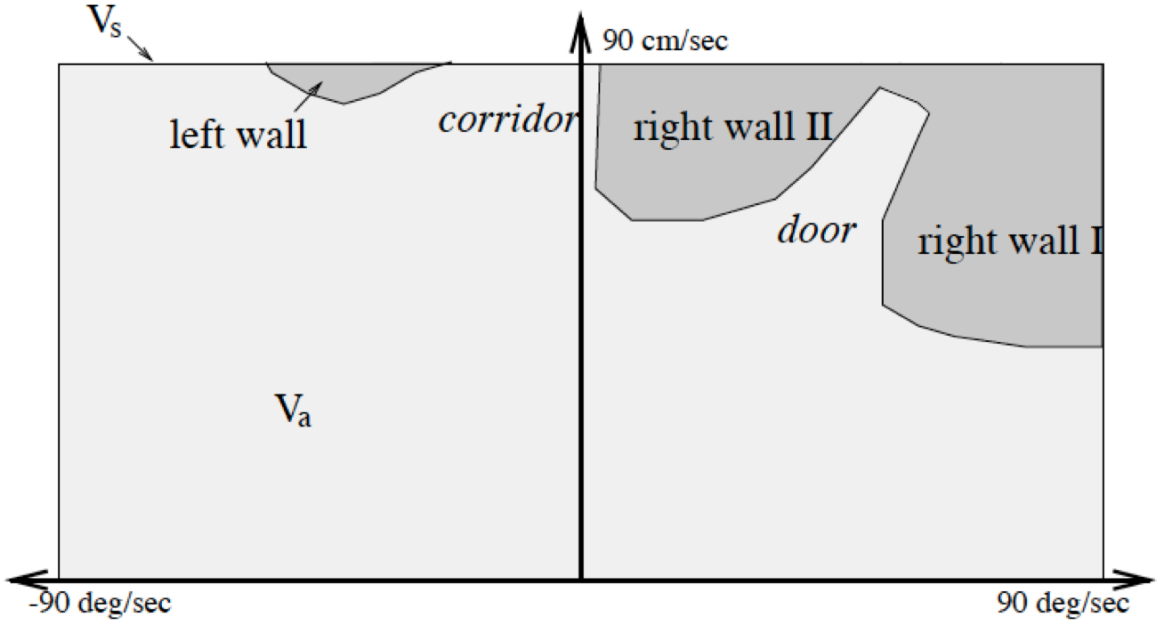
Umgebung mit Hindernispunkten



Durch Steuerbefehl (v, ω)
ist Kreisbogen mit
Radius $r = v/\omega$ definiert.

Beispiel für zulässige Steuerbefehle





- Zulässige Steuerbefehle sind hellgrau dargestellt.
- Nicht zulässige dunkelgrau.
- Maximale Beschleunigung ist dabei 0.5m/sec^2 und 60°/sec^2

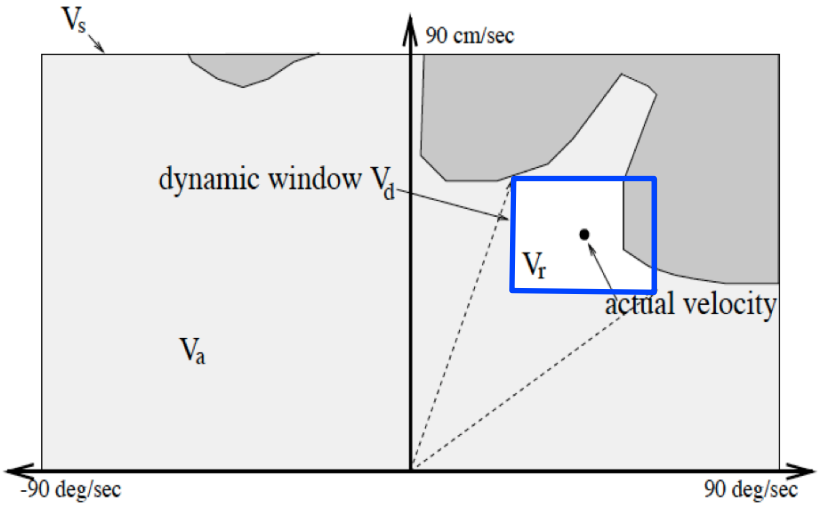
Zielfunktion

- Auf jeden Steuerbefehl im diskretisiertem Suchraum wird folgende Zielfunktion angewandt.

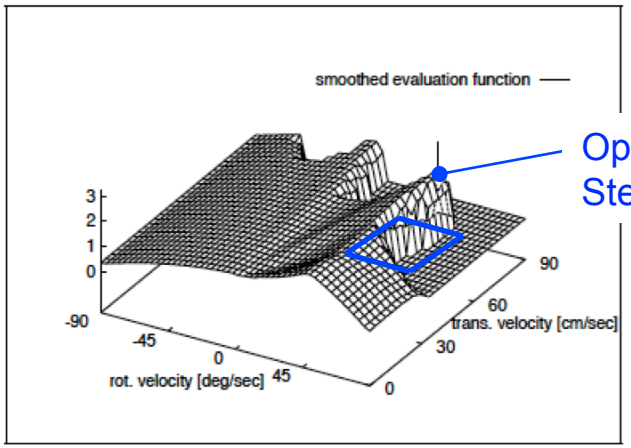
$$G(v, \omega) = a*heading(v, \omega) + b*dist(v, \omega) + c*velocityt(v, \omega)$$

$heading(v, \omega)$	Zielausrichtung
$dist(v, \omega)$	Hindernisentfernung
$velocity(v, \omega) = v$	Geschwindigkeit

- Die Zielfunktion G wird geglättet und Steuerbefehl mit maximalem Zielwert gewählt.



Suchraum

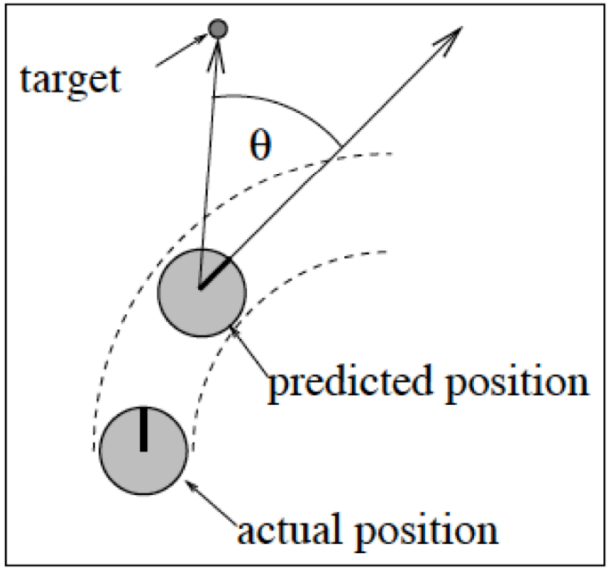


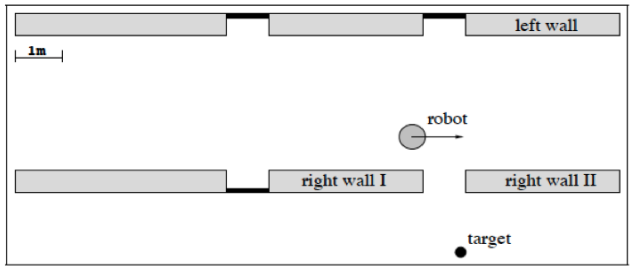
Zielfunktion G für kompletten Bereich V_S

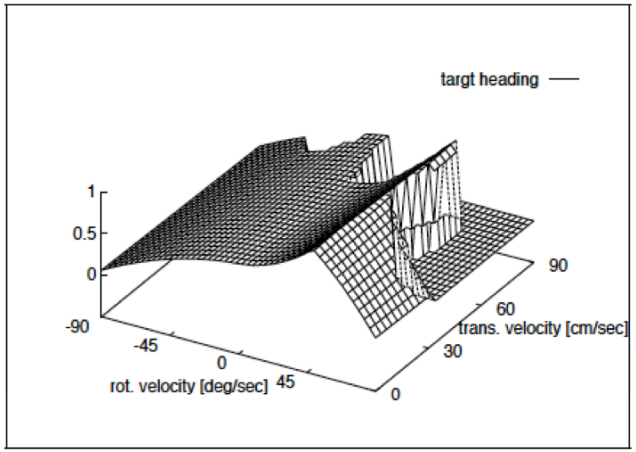
Zielfunktion - heading

heading(v, ω) =

Abweichung der Ausrichtung des Roboters von der Zielausrichtung, nachdem der Roboter den Steuerbefehl *(v, ω)* ein Zeitschritt T lang durchgeführt hat.



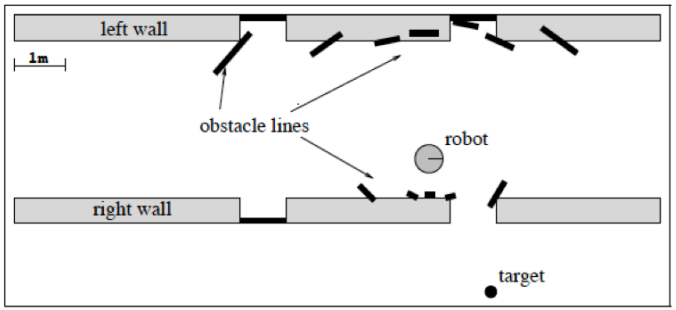




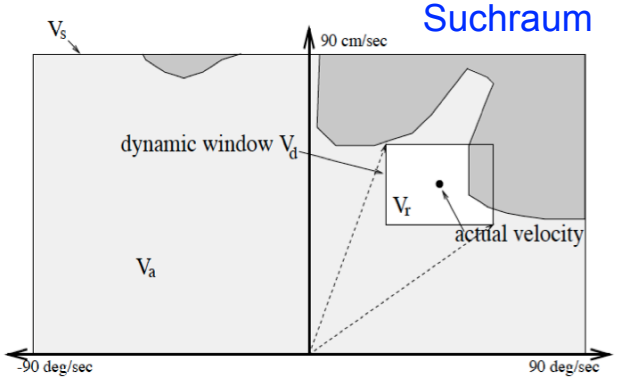
heading(v, ω)

DWA Beispiel

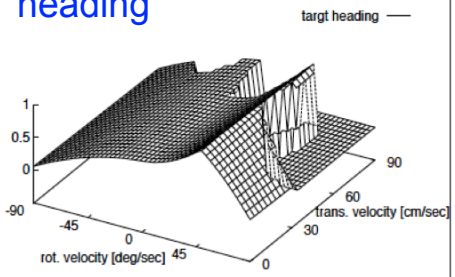
Umgebung mit Hindernissen



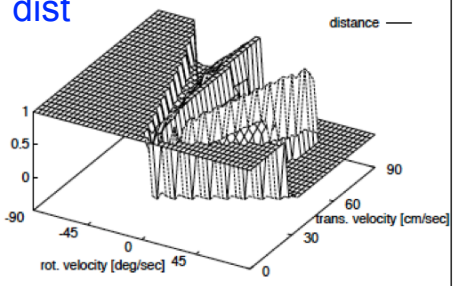
Suchraum



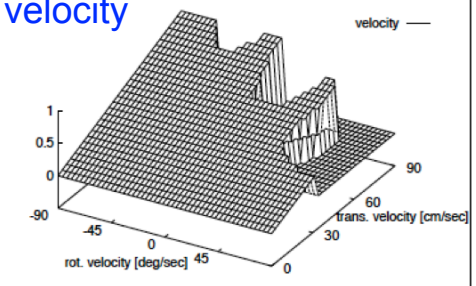
heading



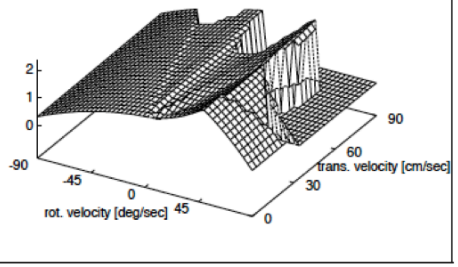
dist



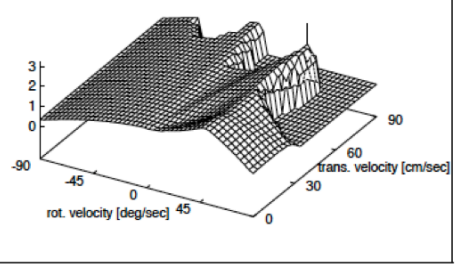
velocity



Zielfunktion G

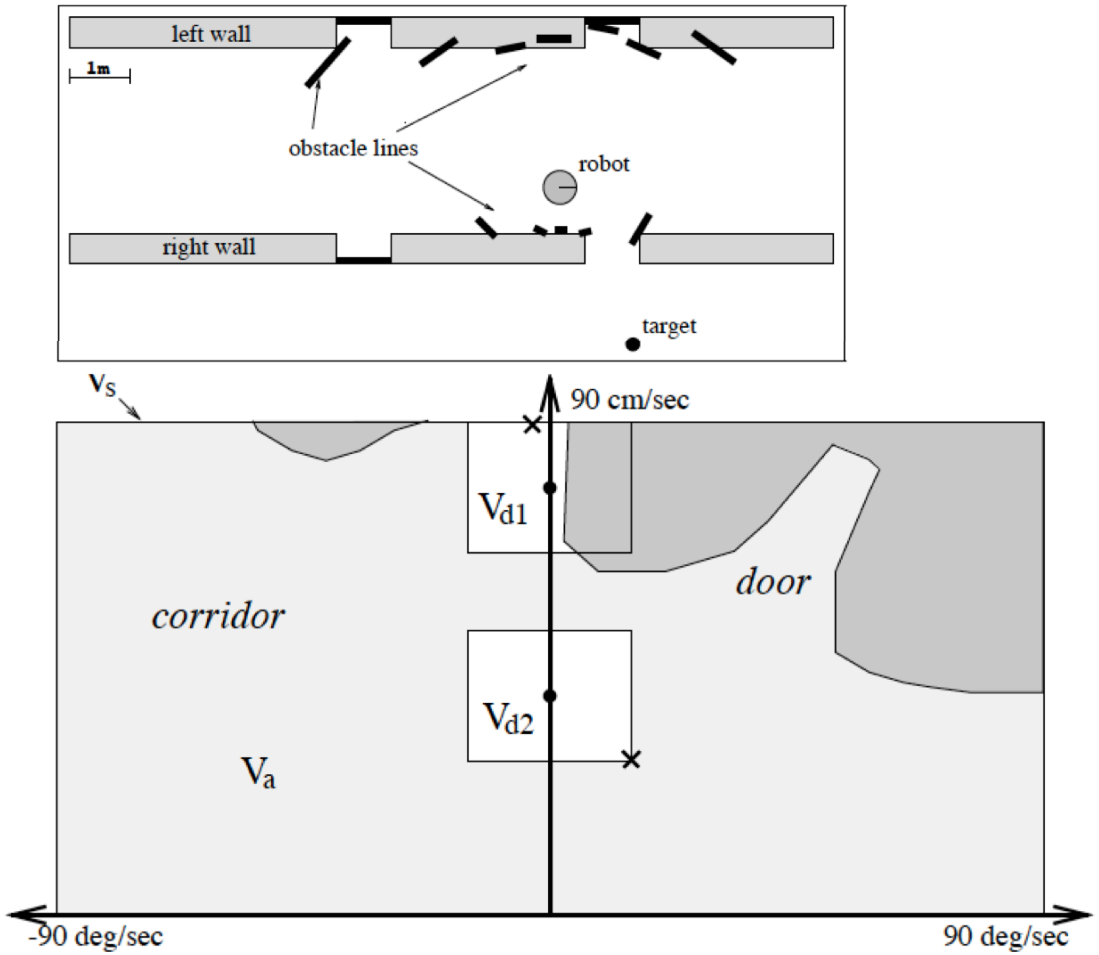


Geglättete Zielfunktion



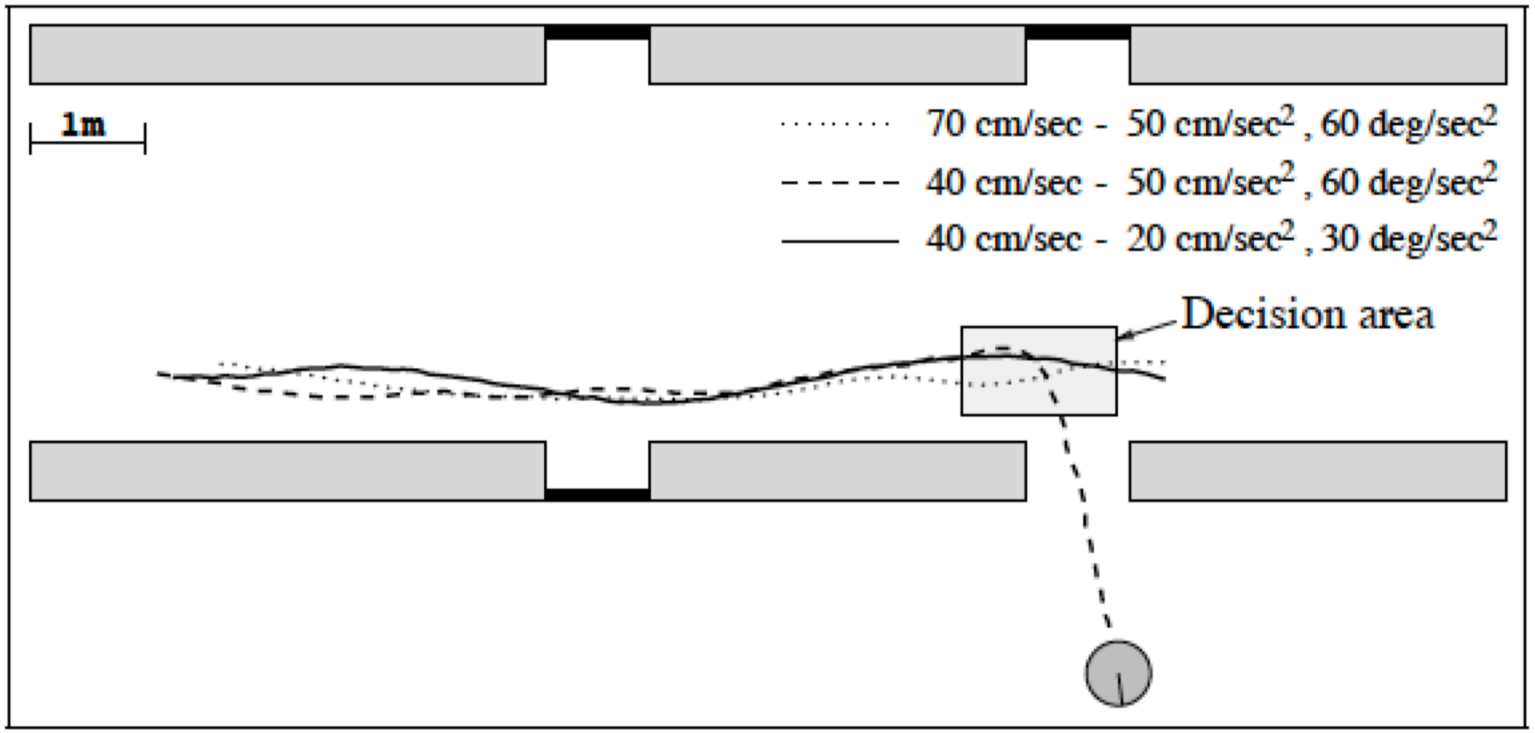
G = 2.0*heading
+ 0.2*dist
+ 0.2*velocity

DWA bei unterschiedlichen Geschwindigkeiten

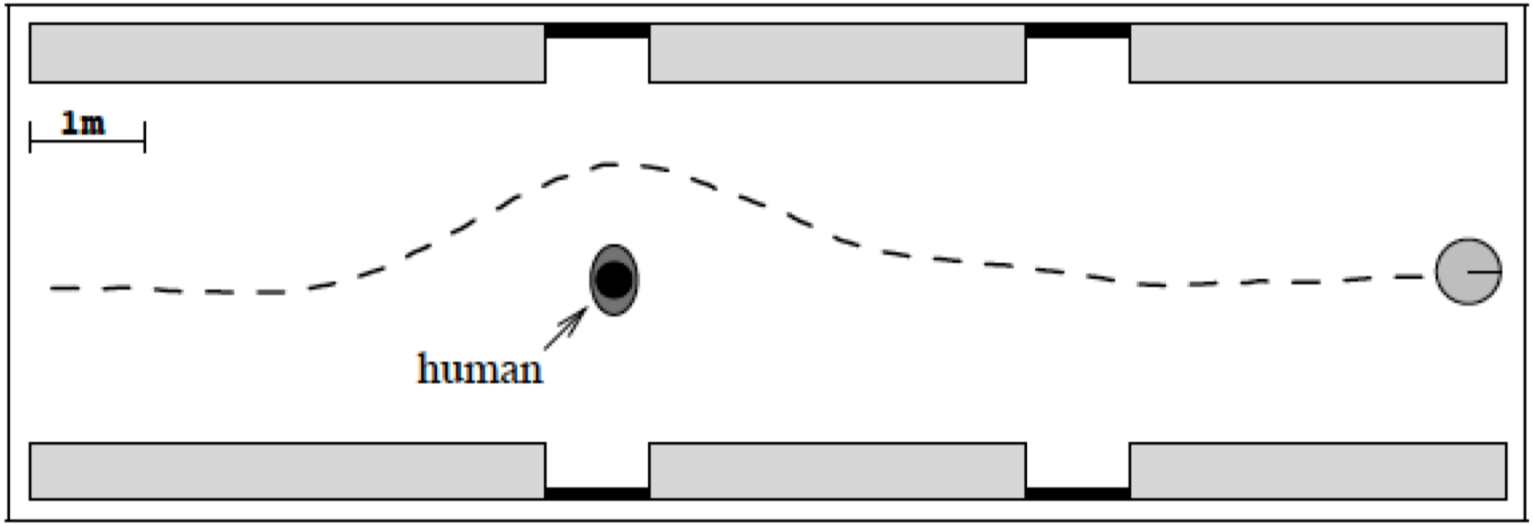


- Dynamik-Fenster bei 0.75 m/sec und 0.4 m/sec
- Berechneter optimaler Steuerbefehl durch x dargestellt.

Navigationspade für verschiedene dyn. Bedingungen

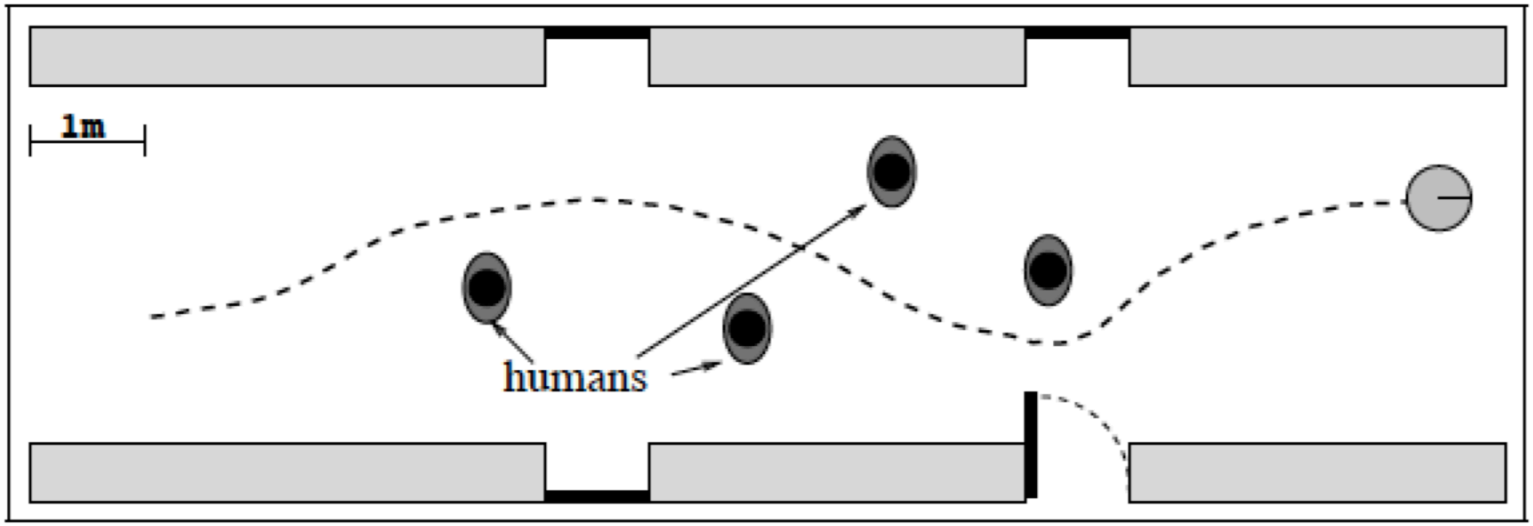


Navigationspfad mit dynamischem Hindernis



- Geschwindigkeit bei Umfahrung der Person: 0.55 m/sec
- Durchschnittgeschwindigkeit: 0.72 m/sec

Navigationspfad mit mehreren Hindernissen



- Geschwindigkeit bei Umfahrung der Personen maximal 0.95 m/sec
- Geschwindigkeit bei Engstelle an der Türe: 0.2 m/sec
- Durchschnittgeschwindigkeit: 0.65 m/sec