



FAKULTÄT FÜR
INFORMATIK

Kartenerstellung und Lokalisierung

Softwareprojekt Teamrobotik SS2010

Thanh Hong Nguyen

Tran Tuan Nguyen

Duc Tam Nguyen

02.06.2010



Gliederung

- Projektbeschreibung / Aufgabenstellung
- Projektziel
- Idee
- Implementierung
- Erreichte Ziele
- Probleme
- Projektplan
- Meilensteine
- Quelle



Projektbeschreibung / Aufgabenstellung

- Kartenerstellung und Lokalisierung
 - Fusion und Auswertung der Daten von Laserscanner und Ultraschallsensoren
 - Erstellung einer Umgebungskarte
 - Bestimmung der eigenen Position anhand der Odometrie

[4]



Projektziele

- **Muss-Ziele**
 - Kartenerstellung der Umgebung
 - Bestimmung der eigenen Position
 - Ständige Aktualisierung der Karten während der Bewegung des Roboters

- **Soll-Ziele**
 - Strukturierte Benutzeroberfläche
 - Optimierung des Programms
 - Bessere Positionsbestimmung und Kartenerstellung (mit Gruppe 3)

- **Optional-Ziele**
 - Verbessern der Karte durch Ultraschallsensordaten
 - Autonome Umgebungsexploration (mit Gruppe 2)



Idee

- Initialisierung der Karte und Roboterposition
- lokale Hindernispositionen aus dem Laserscanner (Abstand und Winkel von Punkten zum Roboter)
- Umrechnung lokaler Hindernispunkten in globalen Koordinaten mithilfe mathematischer Modellen
- Aktualisierung der Roboterposition nach jeder Bewegung mithilfe der Odometriedaten



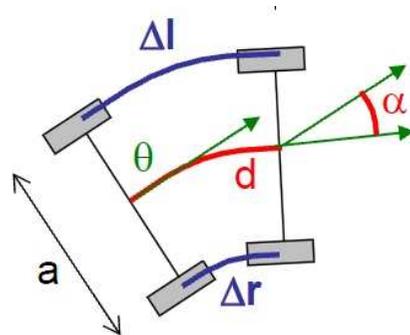
Implementierung (1)

- Gitterkarte statt Linienkarte
- 2-dimensionales Array vom Typ `int` repräsentiert die Karte der Umgebung
 - Jeder Eintrag des Arrays speichert Information einer Zelle der Karte
 - Keine Daten vorhanden : -1
 - Freie Punkte : 0
 - Hindernis : 1
 - Roboterposition : 2
 - Roboterachse : 3
 - Roboterblickrichtung : 4
 - Für jede Konstante wird dann mit einer Farbe in der Karte dargestellt



Implementierung (2)

- Roboterposition aktualisieren
 - Der Roboter misst zu diskreten Zeitpunkten t den mit dem linken bzw. rechten Rad zurückgelegten Weg Δl und Δr .
 - In diesem Zeitintervall ändert sich der Orientierungswinkel um α und der Roboter legt einen Weg d zurück.



$$d = \frac{\Delta l + \Delta r}{2}$$

$$\alpha = \frac{\Delta r - \Delta l}{a}$$

[3]



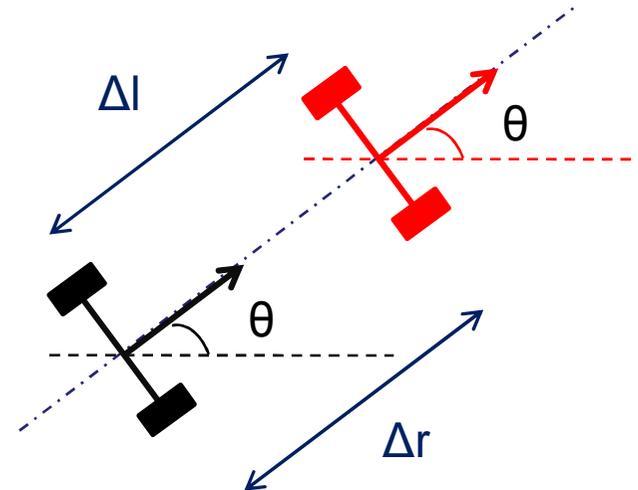
Implementierung (3)

- Roboter fährt geradeaus

$$x_{t+1} = \begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x + d \cos(\theta + \frac{\alpha}{2}) \\ y + d \sin(\theta + \frac{\alpha}{2}) \\ \theta + \alpha \end{pmatrix} \quad \text{wobei } x_t = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$$

$$\Delta r = \Delta l = d$$

$$\alpha = \frac{\Delta r - \Delta l}{a} = 0$$



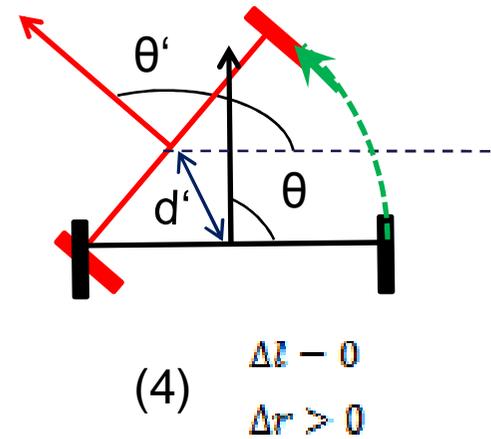
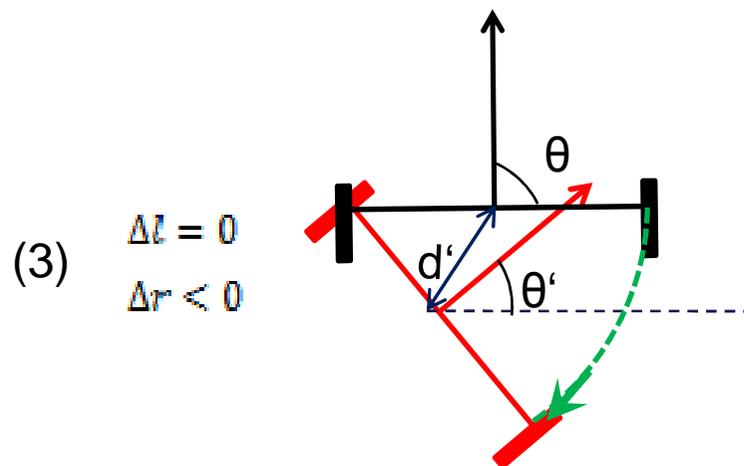
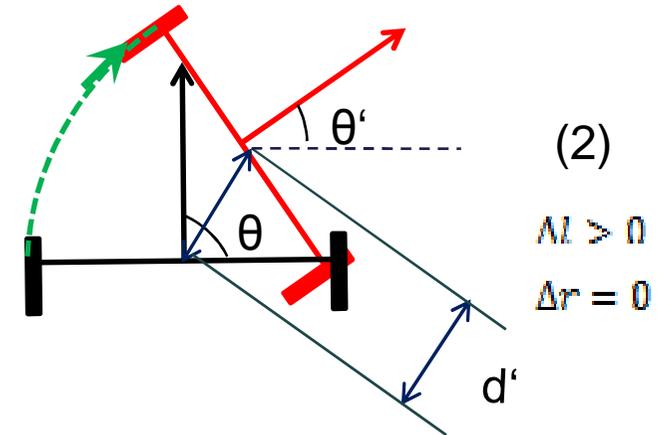
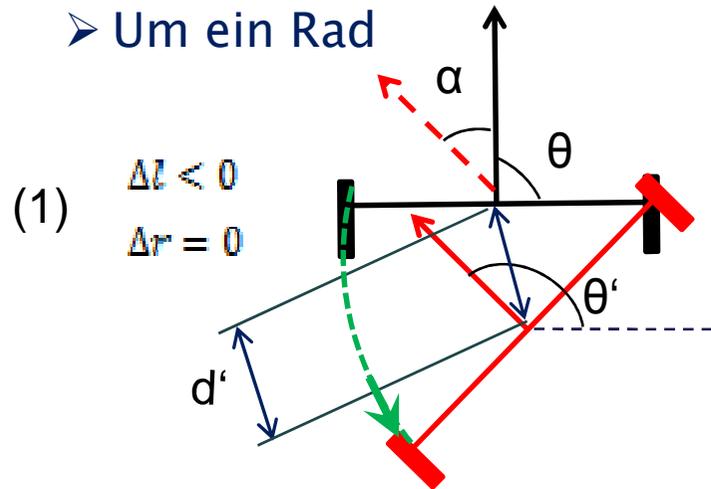
[3]



Implementierung (4)

➤ Roboter dreht sich um

➤ Um ein Rad





Implementierung (5)

➤ Für diese Fälle :

$$d' = \sqrt{\frac{a^2}{2} - \frac{a^2}{2} \cos \alpha}$$

$$x_{t+1} = \begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x + d' \cdot \cos \left(\theta + \frac{\alpha}{2} + k \cdot \pi \right) \\ y + d' \cdot \sin \left(\theta + \frac{\alpha}{2} + k \cdot \pi \right) \\ \theta + \alpha \end{pmatrix} \text{ wobei } x_t = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$$

$$k = \begin{cases} 0, & \text{wenn } \Delta r > 0 \text{ oder } \Delta l > 0 \text{ (Fall (2) und (4))} \\ 1, & \text{sonst (Fall (1) und (3))} \end{cases}$$



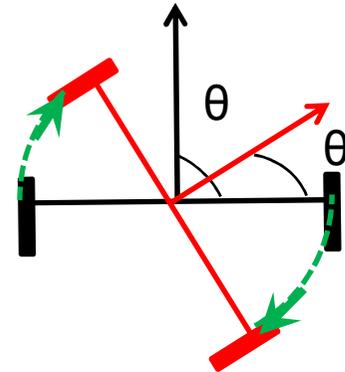
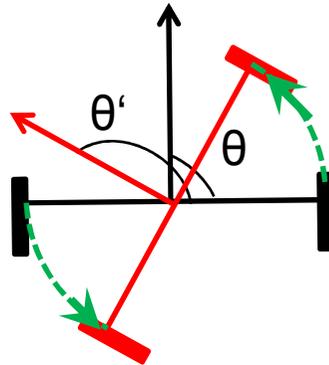
Implementierung (6)

- Roboter dreht sich um
 - Um den Mittelpunkt der Vorderachse

$$\Delta r < 0$$

$$\Delta l > 0$$

$$|\Delta r| = |\Delta l|$$



$$\Delta r > 0$$

$$\Delta l < 0$$

$$|\Delta r| = |\Delta l|$$

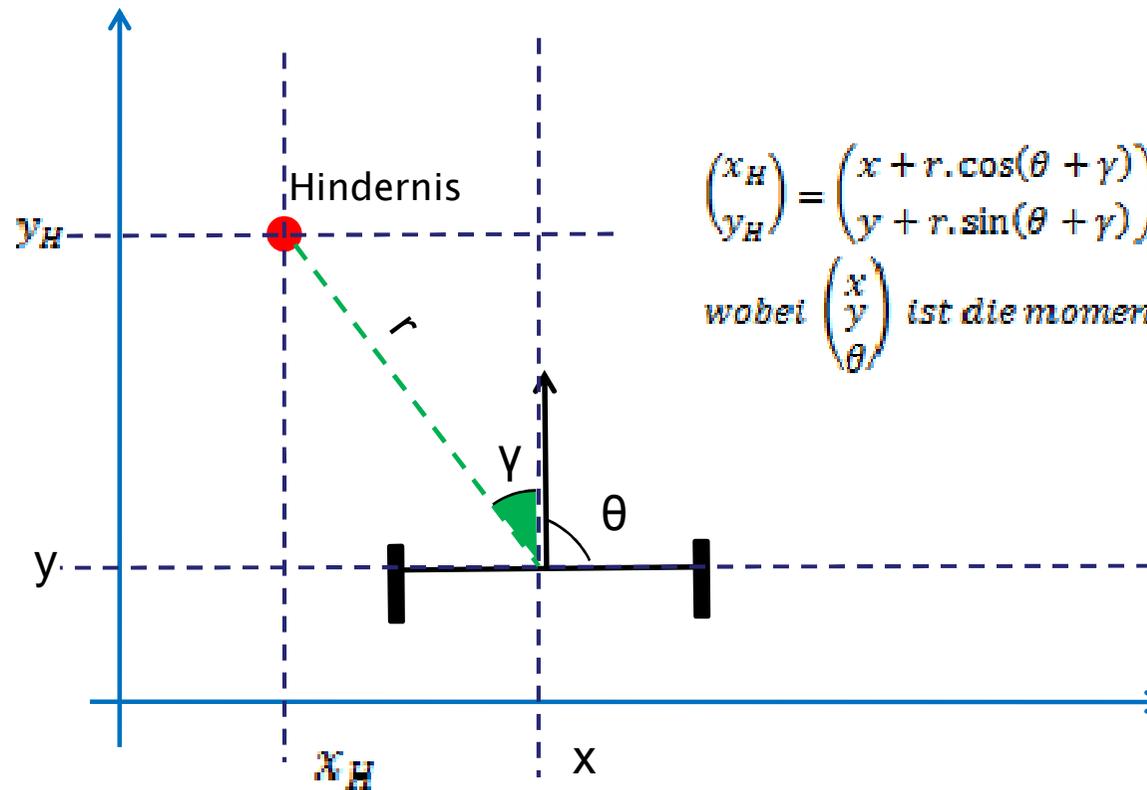
- Für diesen Fälle :

$$x_{t+1} = \begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta + \alpha' \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta + 2 \cdot \frac{\Delta r}{a} \end{pmatrix} \text{ wobei } x_t = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$$



Implementierung (7)

- Umrechnung lokaler Hindernispositionen in globalen Koordinaten



$$\begin{pmatrix} x_H \\ y_H \end{pmatrix} = \begin{pmatrix} x + r \cdot \cos(\theta + \gamma) \\ y + r \cdot \sin(\theta + \gamma) \end{pmatrix}$$

wobei $\begin{pmatrix} x \\ y \\ \theta \end{pmatrix}$ ist die momentane Koordianten des Roboters



Erreichte Ziele

➤ Muss-Ziele

- Kartenerstellung der Umgebung 
- Bestimmung der eigenen Position 
- Ständige Aktualisierung der Karten während der Bewegung des Roboters 

➤ Soll-Ziele

- Strukturierte Benutzeroberfläche
- Optimierung des Programms
- Bessere Positionsbestimmung und Kartenerstellung (mit Gruppe 3)

➤ Optional-Ziele

- Verbessern der Karte durch Ultraschallsensordaten
- Autonome Umgebungsexploration (mit Gruppe 2)



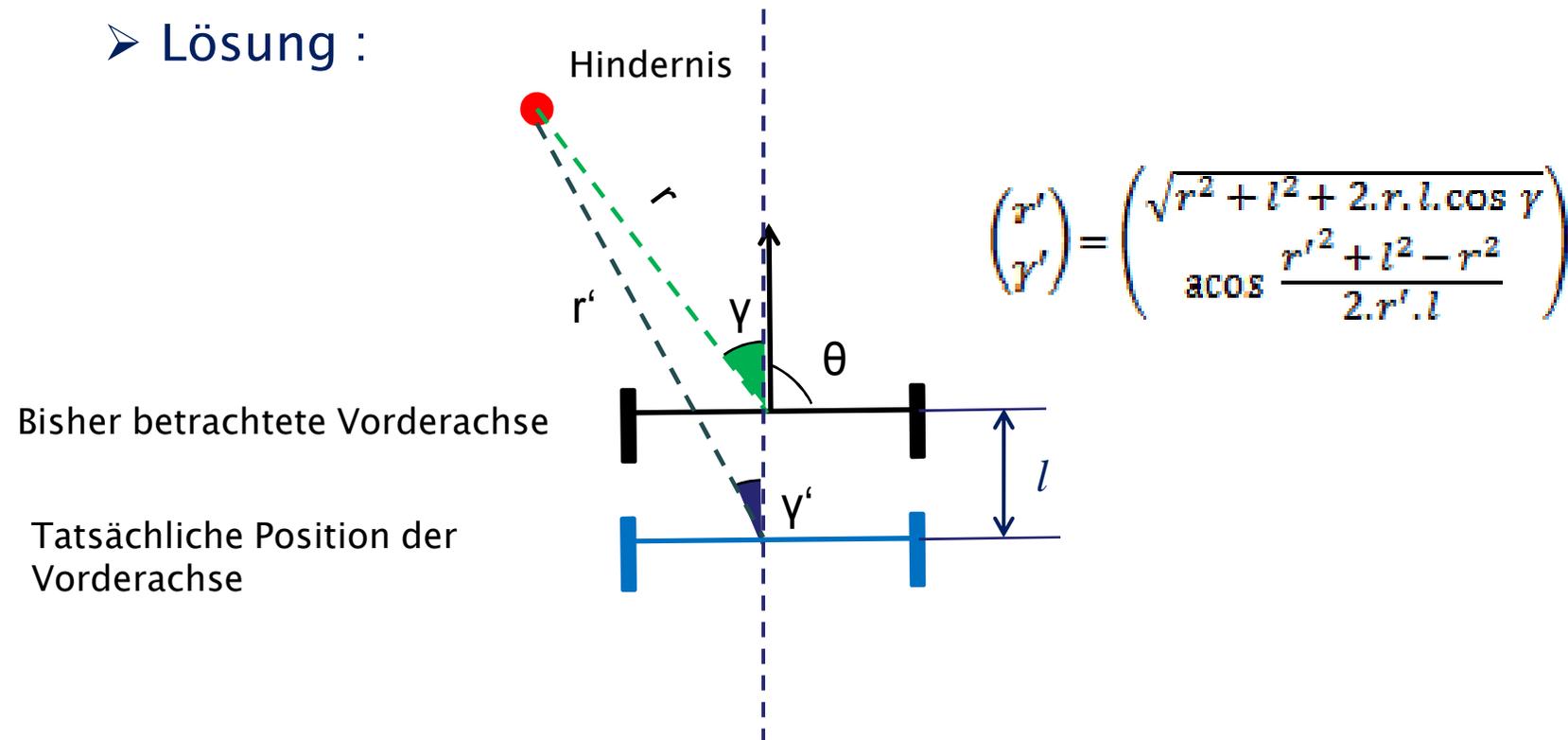
Probleme (1)

- Laserscanner ist nicht auf der Vorderachse, sondern hat einen kleinen Abstand dazu
 - Lösung :
 - Betrachten, als ob der Laserscanner tatsächlich auf der Vorderachse läge
 - Die vom Laserscanner aufgenommenen Daten (Abstand und Winkel zum Hindernis) mit mathematischen Modellen anpassen (umwandeln)

Probleme (2)

- Laserscanner ist nicht auf der Vorderachse, sondern hat einen kleinen Abstand dazu

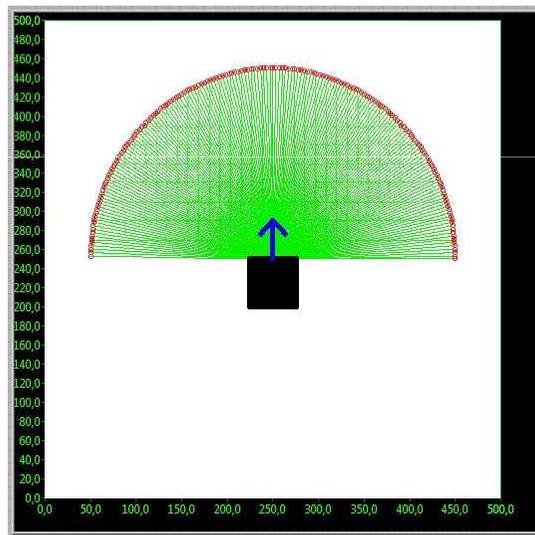
- Lösung :



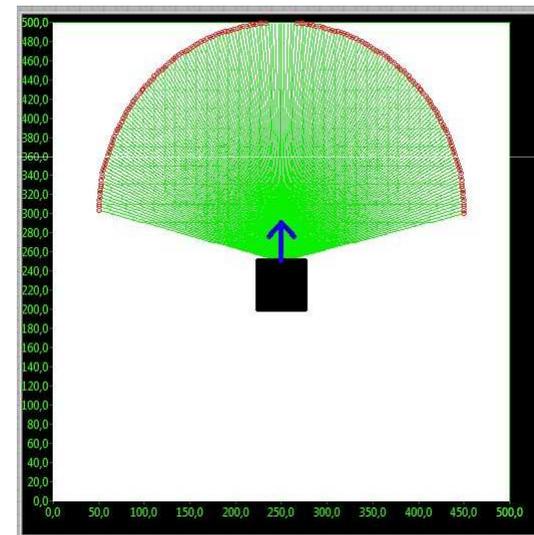


Probleme (3)

- Laserscanner ist nicht auf der Vorderachse, sondern hat einen kleinen Abstand dazu
 - Lösung :



Vorher



Abstand zw. Laserscanner und
Vorderachse hinzufügen



Offene Probleme

- Fehler der Odometrie
 - Roboter fährt nicht geradeaus
 - Unebenheiten
- Begrenzte Speicher vom Roboter
- Große Zeitaufwand bei der Darstellung großer Karte/Array
- Darstellung dynamischer Objekten nicht möglich



Meilensteine

Datum	Aufgabe	
05.05.2010	Konkrete Ideen für die Implementierung	✓
14.05.2010	Darstellung einer dreifarbigen Karte	✓
28.05.2010	Positionsbestimmung anhand der Odometrie	✓
16.06.2010	Karte anhand der Daten aufbauen	✓
23.06.2010	Bereit für die Generalprobe	
30.06.2010	Erfolgreiche Abschlusspräsentation	



Vielen Dank für die Aufmerksamkeit!



Quelle

1. Wolfram Burgard: Martial Hebert Springer Handbook of Robotics Chapter 36. World Modeling. (2007)
2. Markus Köppen, Andreas Hasselberg, René Zimmermann: Kick-off Präsentation Gruppe 1.(2009)
3. O. Bittel: Mobile Roboter-Selbstlokalisierung. (2009)
4. http://www-ivs.cs.uni-magdeburg.de/EuK/lehre/sopras/teamRobotik_ss10/Einf%C3%BChrung.pdf (Stand 03.05.2010)