

Motivation

- Erste Einblicke in die Robotik durch Steuerung und Navigation eines Roboters.
- erlerntes Wissen aus vorherigen absolvierten Studienfächern anwenden
- Roboter Pioneer 3DX, Simulation mit MobileSim und Matlab



Robotersteuerung

Manuelle Steuerung

- W → Vorwärts
- A → Links
- D → Rechts
- S → Rückwärts



Abb. 1: Manuelle Steuerung

Occupancy Grid

Erstellen der Karte:

- Durch Koppelnavigation erstellen der näheren Umgebung
- Darstellung in einem OccupancyGrid, welches die Karte in ein Raster einteilt

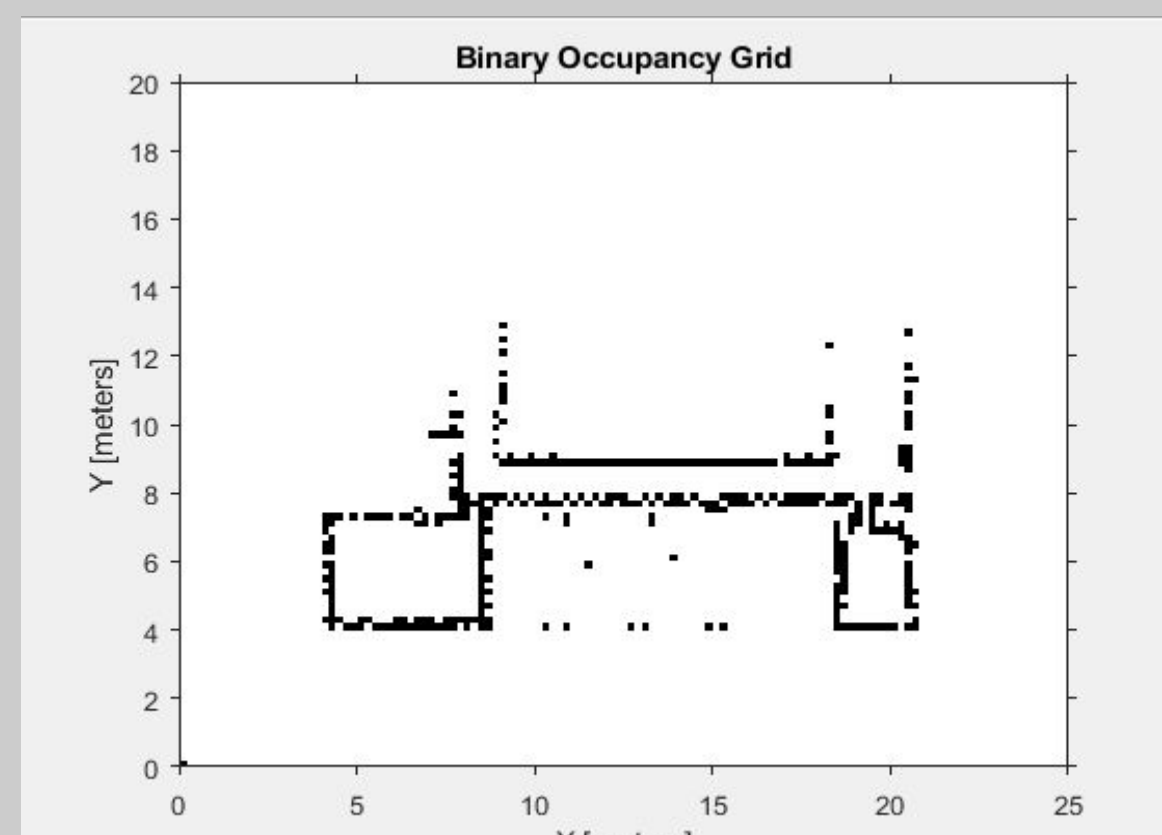


Abb. 4: Darstellung der Karte

Erstellen Gesamtkarte:

- GPS-Daten einlesen und in Occupancy Grid darstellen

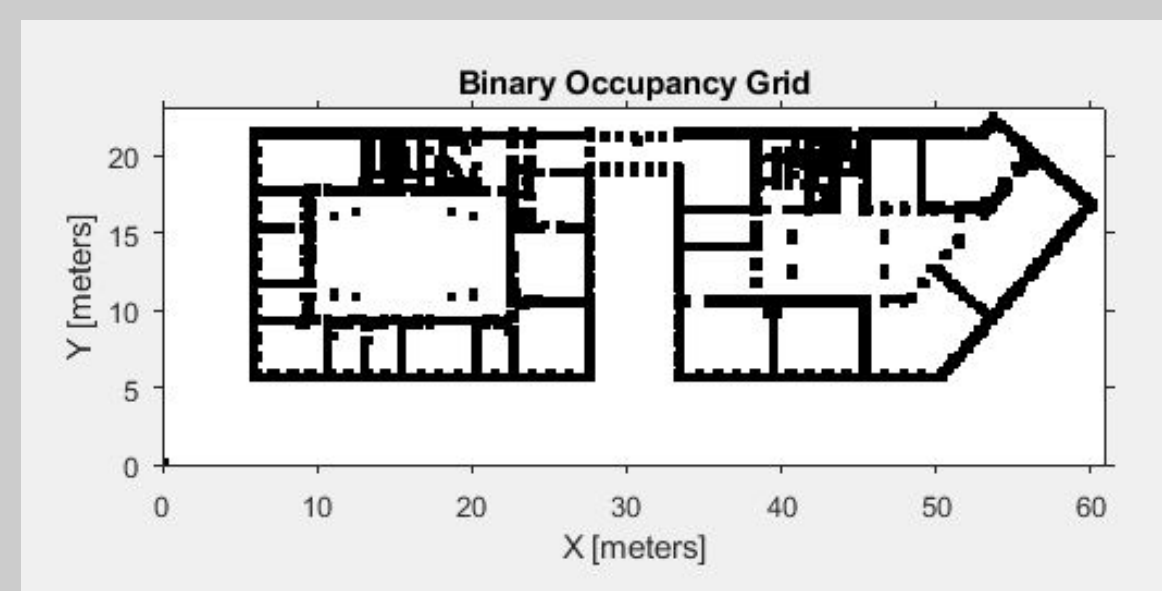


Abb. 5: Komplette Karte

Homeing

Rückkehr zur Startposition

- Messen der Distanz zu den Stützpunkten
- Berechnung der Differenz

$$d_{1-4,S} = \sqrt{(x_{1-4,S} - x_R)^2 + (y_{1-4,S} - y_R)^2}$$

Abb. 7: Formel für Homeing

- minimal berechnete Distanz wird als kürzeste Strecke interpretiert
- Rückkehr zur Startposition

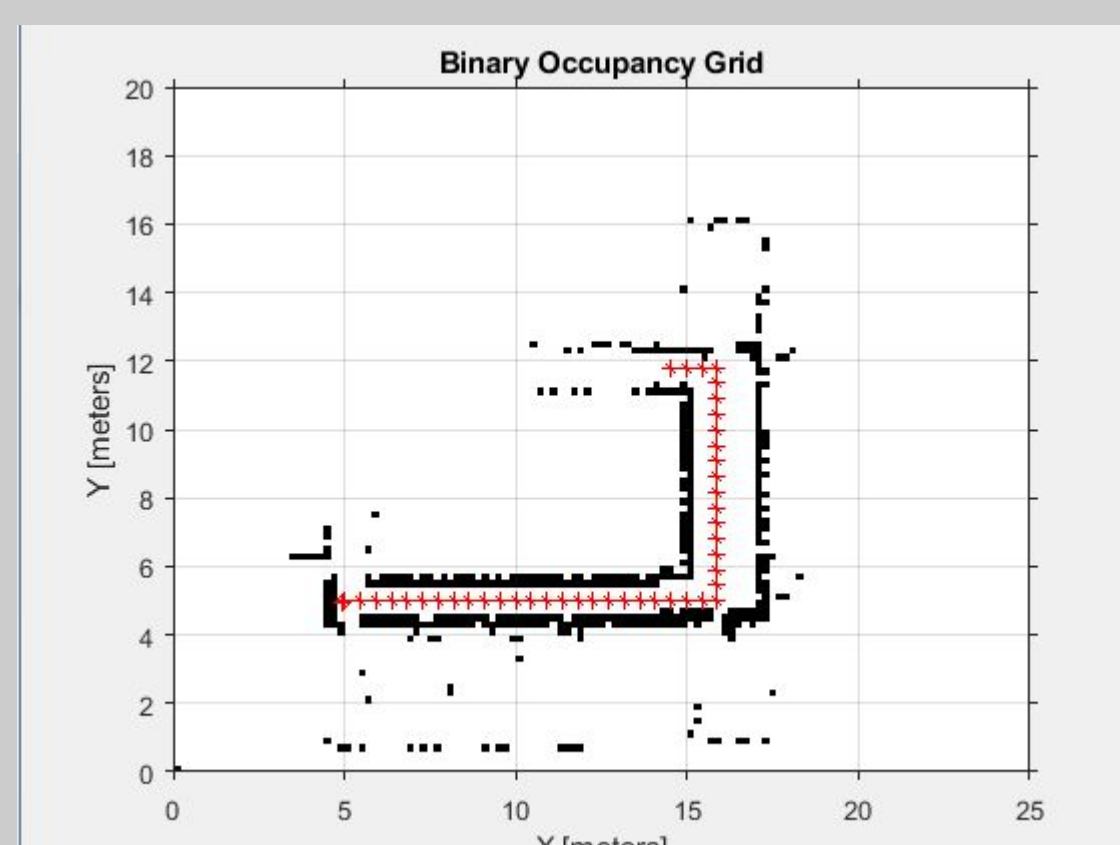


Abb. 8: Darstellung Trajektorie

Ausblick

- höhere Positionsgenauigkeit durch die Implementation eines Kalmanfilters
- Ergänzung eines optischen Sensors für realitätsnahe 3-D Abbildungen
- Verbindung mit der Matlab Mobile App um den Roboter fernzusteuern
- Aufsuchen einer Ladestation bei niedrigem Batteriewert
- Datenaustausch über TCP oder UDP

Map Building

Koppelnavigation:

- Verwendung der 16 Sonarsensoren
- Sonar misst Distanz zur Wand
- Berechnung der Positions- und Winkeländerung durch polares Anhängen

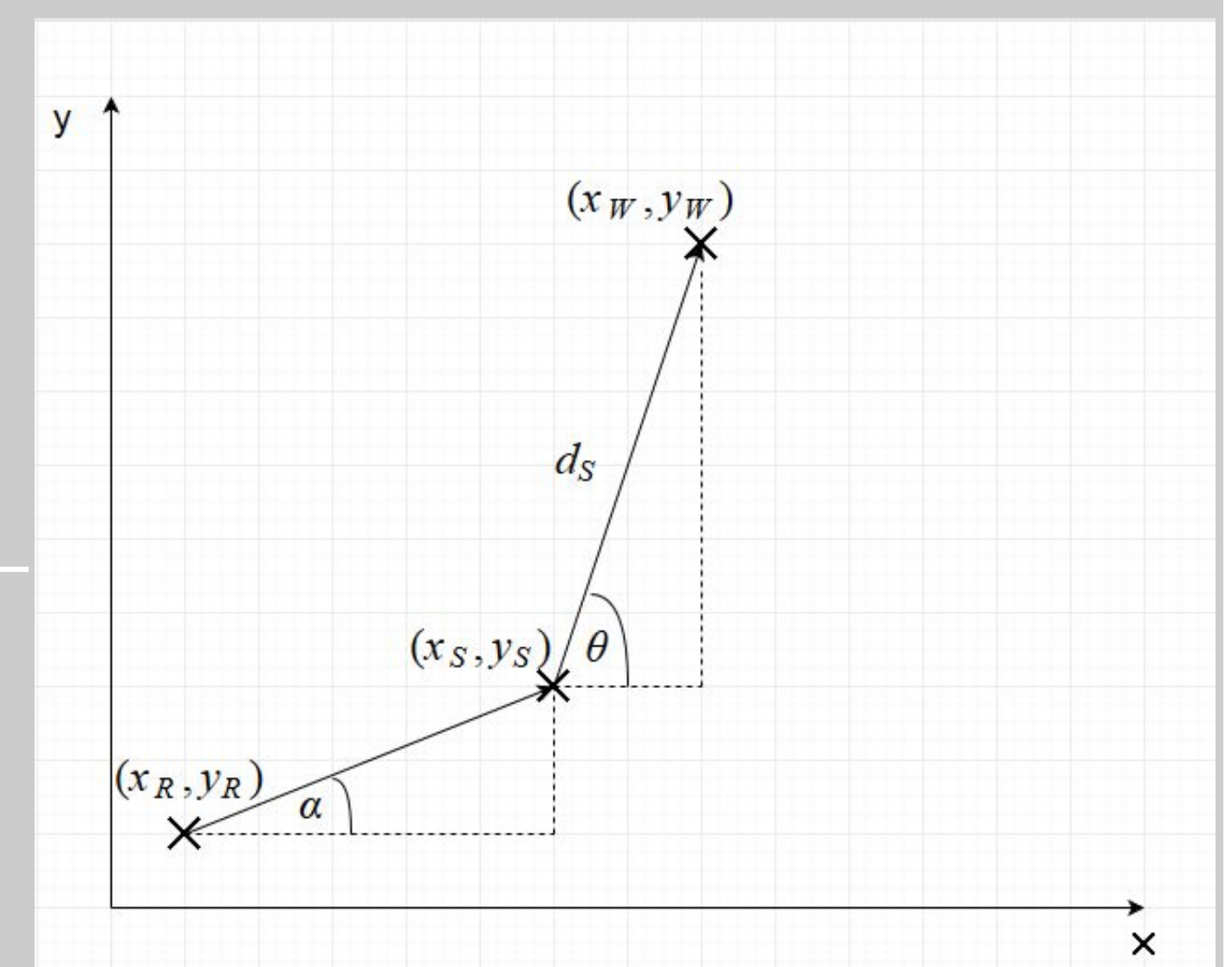


Abb. 2: Darstellung Koppelnavigation

$$x_W = ((d_S + \sqrt{x_S^2 + y_S^2}) * \cos(\alpha + \theta) + x_R)$$

$$y_W = ((d_S + \sqrt{x_S^2 + y_S^2}) * \sin(\alpha + \theta) + y_R)$$

Abb. 3: Formel Koppelnavigation

Package Drop

Briefkasten ansteuern

- Briefkasten misst Distanz zum Roboter
- Ab einem Schwellwert von 1.5 Metern teilt der Briefkasten dem Roboter dies mit
- Roboter richtet sich zum Briefkasten aus und steuert selbstständig auf diesen zu.
- Nach Ankunft kann ein neuer Briefkasten angesteuert werden

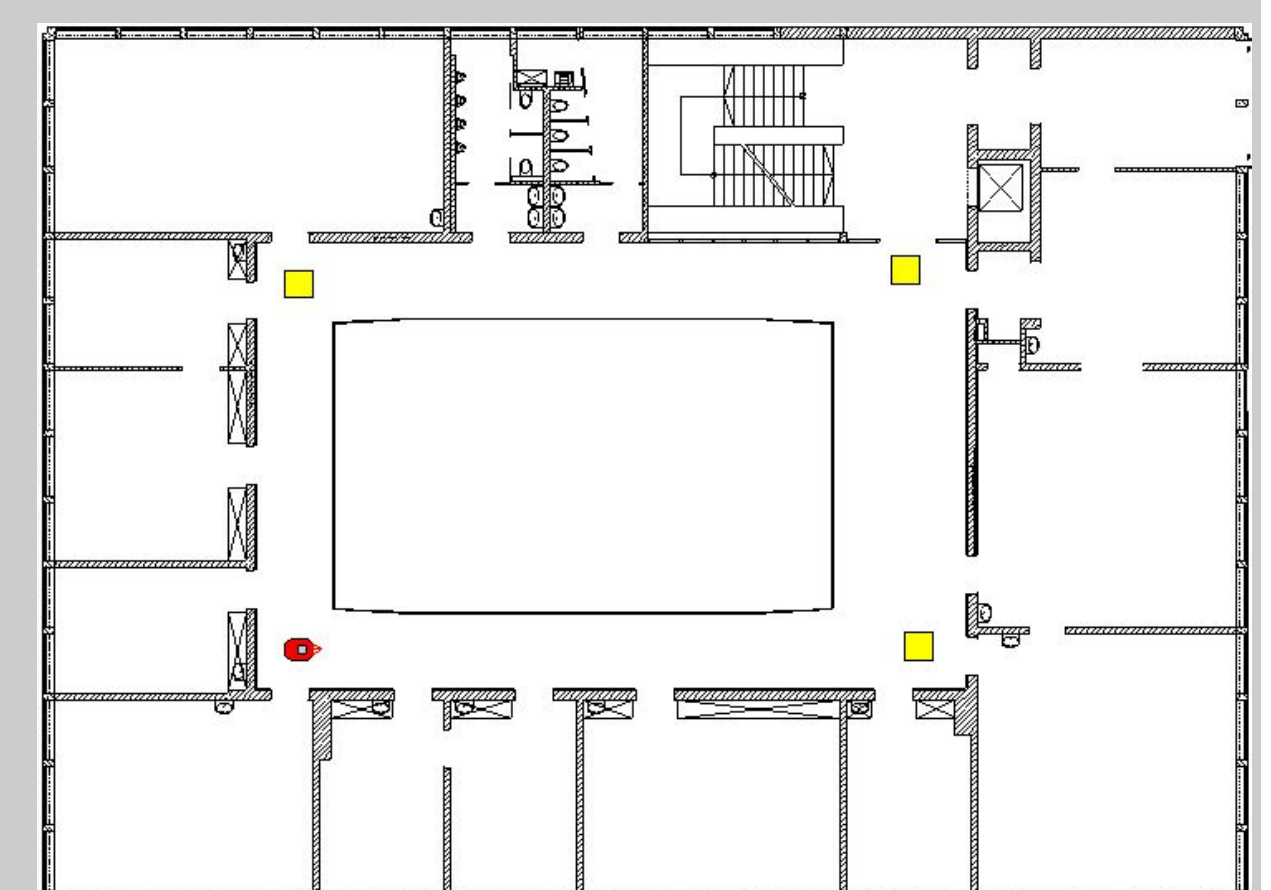


Abb. 6: Briefkasten und Roboter

Collision Avoidance

Kollision vermeiden

- Empfindlichkeit der vorderen, seitlichen und hinteren Sensoren definieren
- Distanz zum Hindernis mittels der Sonarsensoren messen
- Roboter stoppt automatisch um den Zusammenstoß zu vermeiden und dreht sich um 180°

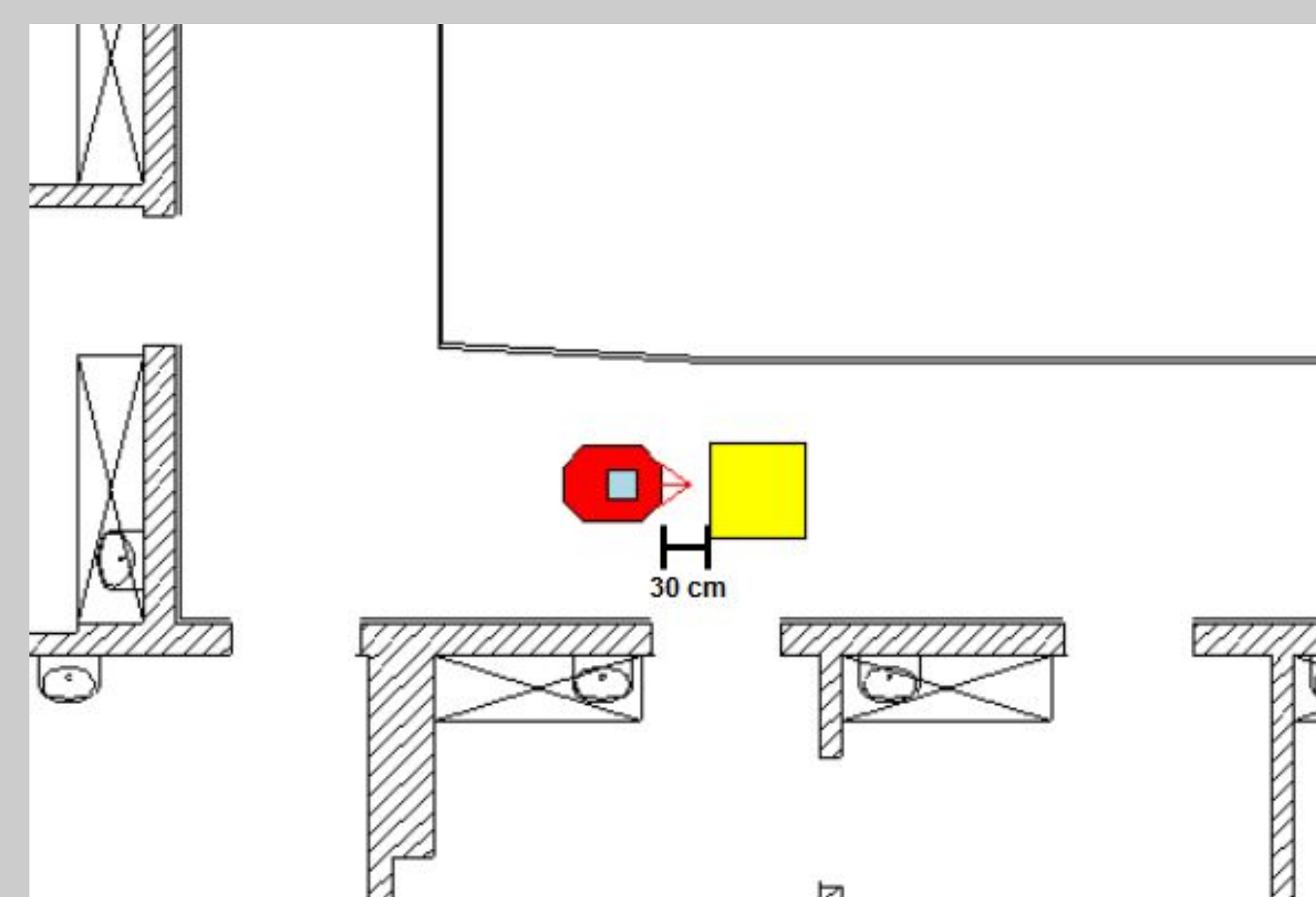


Abb. 9: Vermeiden einer Kollision

Danksagung

Wir danken dem Labor für Navigation für die Bereitstellung der Hardware und Software.

Kontakt

Prof. Dr. Thomas Abmayr – thomas.abmayr@hm-muenchen.de

Florian Folger – florian.folger@hm-muenchen.de Henry Tran – henry.tran@hm-muenchen.de