

**Benutzerhandbuch zur
graphischen Oberfläche
der Monte-Carlo-Lokalisation**

AI-Projekt Sommersemester 2016

Arno v. Borries,
Jan Phillip Kretzschmar,
Andreas Walscheid

Betreuer:
Prof. Dr. Heiner Klocke,
Dipl. Inform. Alex Maier

26. September 2016

Inhaltsverzeichnis

1	Initialisierung des Programms	2
2	Lokalisierung durchführen	3
3	Zusatzfunktionalität	4
3.1	Graphische Darstellung	4
3.2	Einstellungen	5
4	Roboter	9

1 Initialisierung des Programms

Um mit der im AIMA-Framework bereits enthaltenen graphischen Oberfläche eine Monte-Carlo-Lokalisation durchzuführen, muss zunächst der Roboter initialisiert werden und eine Karte der Problemumgebung geladen werden.

Um den virtuellen Roboter der Beispielumgebung des AIMA-Frameworks zu nutzen wird lediglich der Button "Initialize Robot" (Abbildung 1) betätigt. Zusätzlich muss und bei den Lego Mindstorms NXT Robotern in einem neuen Dialog der Robotername und der Name der auf dem Roboter auszuführenden Programmdatei angegeben werden. Beispiel: Robotername: C_GRUEN, Program: MclDaemon.nxj (eine MclDaemon.nxj Datei liegt dem Projekt bei) (Abbildung 2).

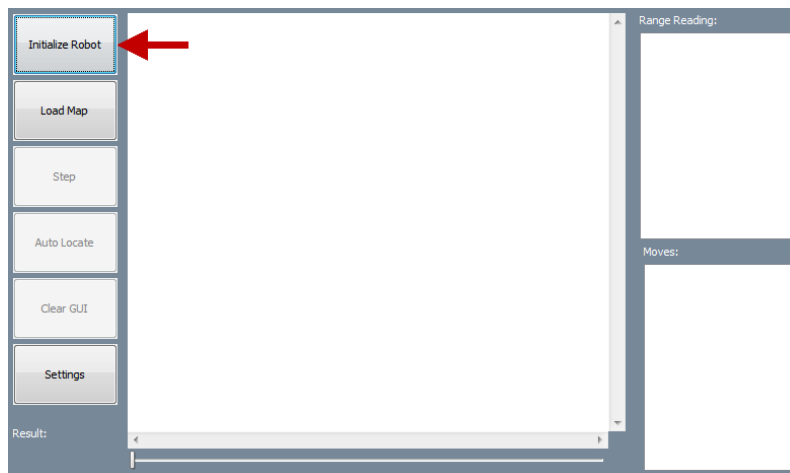


Abbildung 1: Anfangszustand der graphischen Oberfläche. Der gekennzeichnete Button öffnet die Ansicht zum Verbindungsaufbau mit dem physischen Roboter.

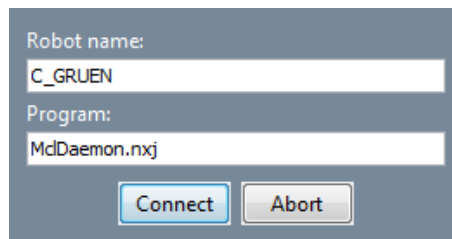


Abbildung 2: Eingabedialog zur Verbindung mit einem NXT Roboter.

Eine Karte der Problemumgebung wird über den Button "Load Map" (Abbildung 3) geladen. Im darauf folgenden Dateibrowser kann dann eine Kartendatei ausgewählt und geöffnet werden. Die Datei muss dabei im SVG-Format vorliegen. Wenn die Datei ausgewählt und geöffnet wird, wird die Karte geladen (Abbildung 4). Sobald die Karte erfolgreich geladen wurde, werden alle restlichen Buttons aktiviert und die eigentliche Lokalisation kann durchgeführt werden.

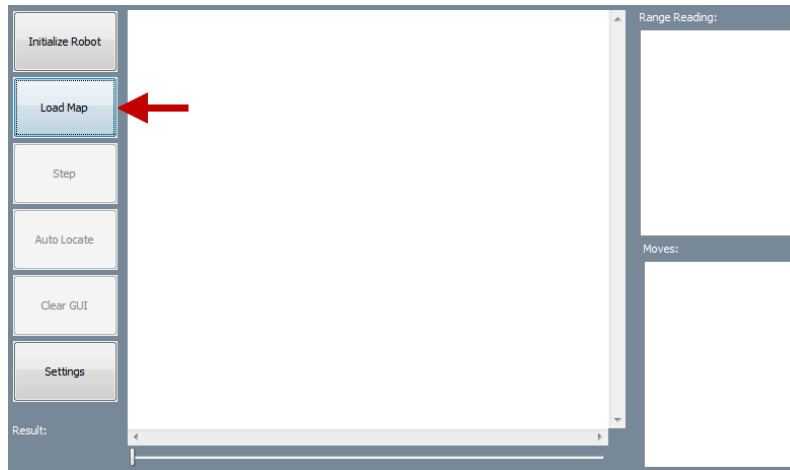


Abbildung 3: Der gekennzeichnete Button öffnet ein Dateiauswahlfenster, durch das die zu ladende Karte bestimmt wird.

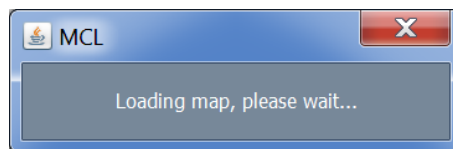


Abbildung 4: Hinweisfenster während des Ladevorgangs.

2 Lokalisierung durchführen

Bei der Lokalisierung mit dieser Software stehen zwei Möglichkeiten zur Verfügung, einmal die manuelle und einmal die voll automatisierte Variante. Bei beiden Varianten werden rechts in den Tabellen die “Entfernungsmessungen” (*Range Reading*) und die “Bewegungen” (*Moves*) ausgegeben.

Bei der manuellen Lokalisierung wird durch den Button “Step” ein Durchlauf des Algorithmus ausgeführt. Dieser besteht aus einer Bewegung, einer Messung der Umgebung mit Hilfe der Sensoren und einer anschließenden Neubewertung und -anordnung der Partikel (Abbildung 5).

Bei der voll automatisierten Lokalisierung (Abbildung 6) werden diese Durchläufe solange nacheinander durchgeführt, bis der Benutzer dies über den gleichen Button wieder beendet (Abbildung 7).

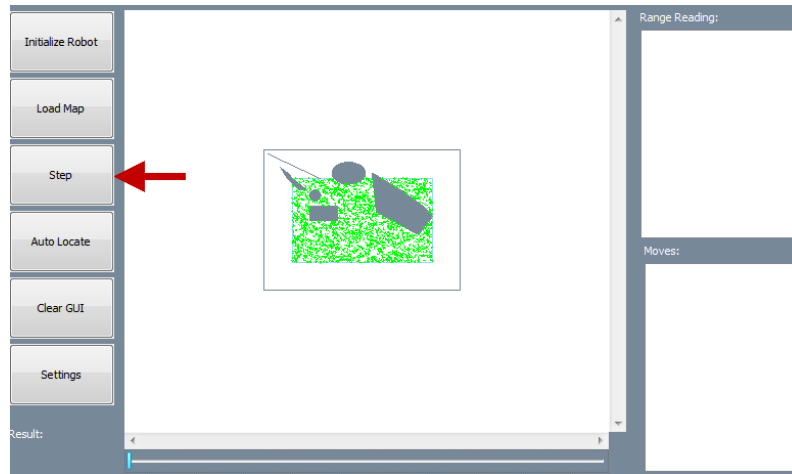


Abbildung 5: Der gekennzeichnete Button führt einen einzelnen Durchlauf des MCL-Algorithmus durch.

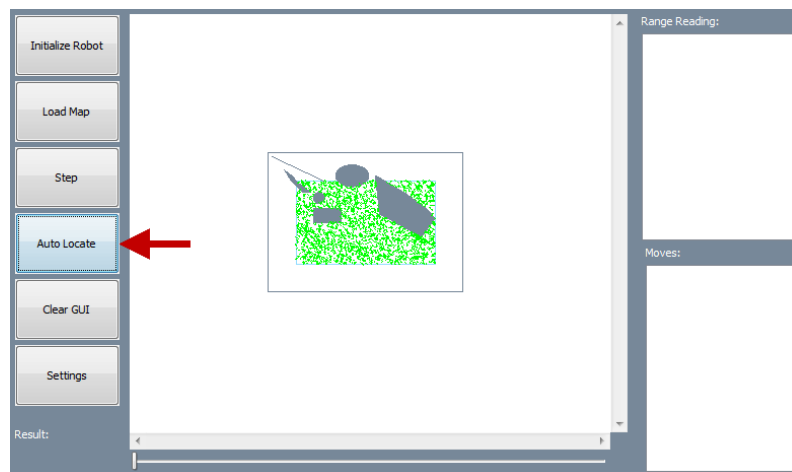


Abbildung 6: Der gekennzeichnete Button führt den MCL-Algorithmus in einer Schleife aus, bis Dieser durch den Benutzer wieder gestoppt wird.

3 Zusatzfunktionalität

3.1 Graphische Darstellung

Mit Hilfe der Schaltfläche "Clear GUI" (Abbildung 8) werden sämtliche bisher ausgegebenen Daten aus der graphischen Benutzeroberfläche gelöscht und die Karte wird wieder in ihren Anfangszustand zurückversetzt. Dieser Button ist insbesondere sinnvoll, um die Lokalisation manuell mit einer zufälligen Partikelwolke neu zu starten.

Die Software bietet darüber hinaus eine Zoomfunktion für die Karte. Wenn die Karte zu groß für den sichtbaren Bereich wird, aktivieren sich automatisch eine horizontale und eine vertikale Scrollbar, die es ermöglichen, den Bildausschnitt zu jedem beliebigen Punkt der Karte zu verschieben (Abbildung 9).

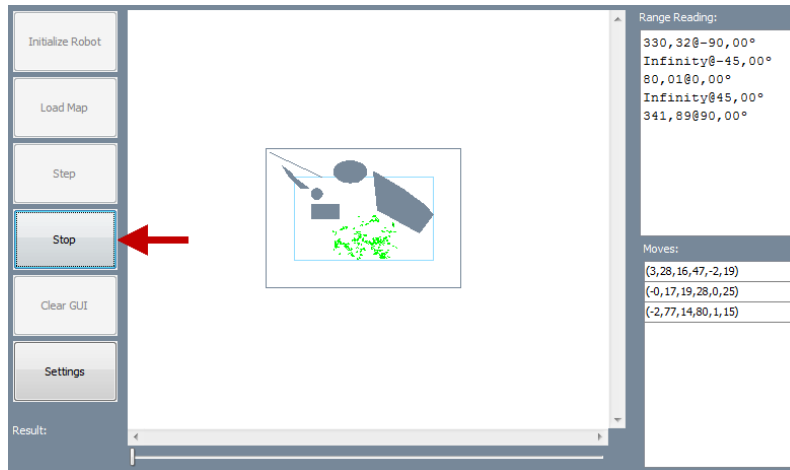


Abbildung 7: Der gekennzeichnete Button stoppt den in einer Schleife laufenden MCL-Algorithmus wieder.

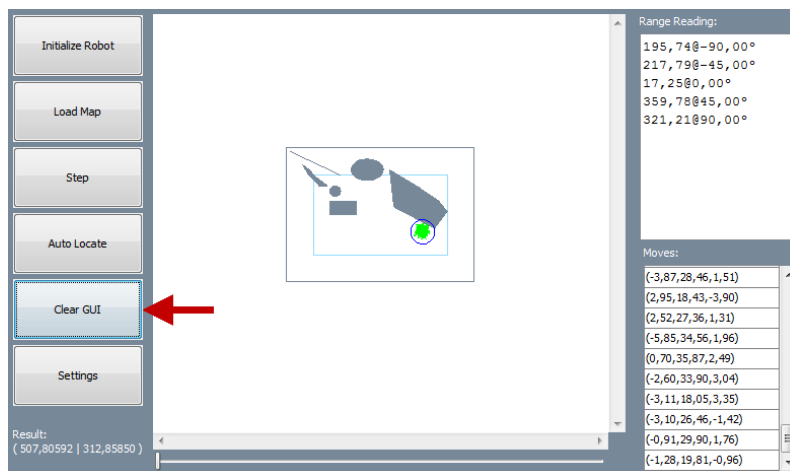


Abbildung 8: Der gekennzeichnete Button löscht alle Datenausgaben aus der Karte und den Ausgabefeldern.

3.2 Einstellungen

In den Einstellungen, die über den Button “Settings” aufgerufen werden, besteht die Möglichkeit, umgebungsspezifische Einstellungen vorzunehmen (Abbildung 10), wie zum Beispiel:

- Anzahl Partikel (Particle count)
- Minimales Partikelgewicht (Min. particle weight)
- Maximale Partikeldistanz (Max. particle distance)
- Maximale Sensormessung (Max. sensor range)
- Rauschen auf die Drehbewegung (im Bogenmaß) (Move rotation noise (rad))

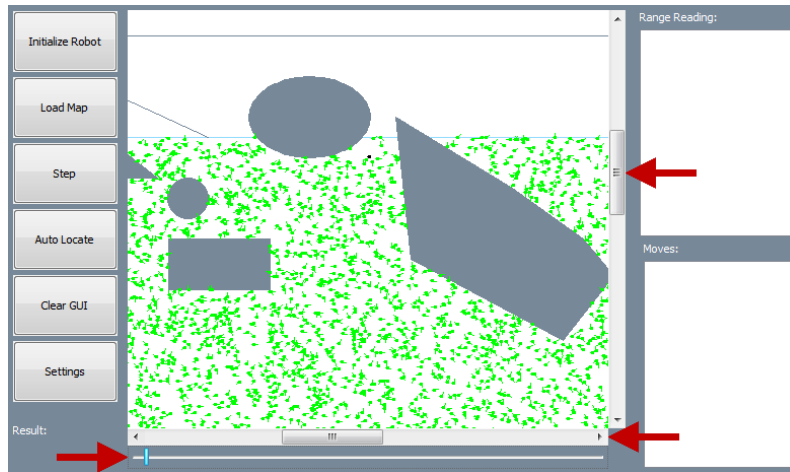


Abbildung 9: Der gekennzeichnete Schieberegler ermöglicht es, in die Karte hinein zu zoomen. Sollte die Karte durch den Zoom größer dargestellt werden, als Platz zur Verfügung steht, kann über die beiden Scrollleisten der sichtbare Kartenausschnitt verschoben werden.

- Rauschen auf die Vorwärtsbewegung (Move distance noise)
- Rauschen auf die Entfernungsmessung (Range reading noise)
- Minimale Bewegungsdistanz (Min. move distance)
- Maximale Bewegungsdistanz (Max. move distance)
- Delta zwischen Entfernungsmessungen für null Gewichtung (Bad range delta)
- Sensor-Messwinkel (Range reading angles)

Zusätzliche Einstellungsmöglichkeiten bei NXT Robotern: (Abbildung 11)

- Geschwindigkeit der Rotationen (Rotation speed)
- Geschwindigkeit der Bewegungen (Präzision vs. Geschwindigkeit) (Travel speed)
- Mindestabstand zu frontalen Hindernissen (Object clearance)
- Farbsensor Hell/Dunkel Mittelwert (Color cutoff)
- Lichtsensor Hell/Dunkel Mittelwert (Light cutoff)
- Automatische Einstellung der Mittelwerte (Autotune cutoffs: Button)
- Bewegungsprofil Linie (GET_LINE_MOVE) oder Zufall (GET_RANDOM_MOVE) (Move command)
- Winkel mit dem Suchbewegungen begonnen werden (Search start angle)
- Mehr Ausgaben auf dem NXT Display (Verbose NXT)

Das Einstellen der Mittelwerte zwischen hell und dunkel für den Farb- und Lichtsensor kann automatisch über den Button “Autotune cutoffs” durchgeführt werden. Dazu wird der Roboter möglichst gerade auf der hellen Linie, der gefolgt werden soll platziert. Nach Betätigung des Buttons führt der Roboter eine Bewegungssequenz aus. Sobald der NXT dies beendet hat, werden die gefundenen Werte in die beiden Sensorfelder eingetragen.

Mit dem Button “Save” werden Änderungen an den Einstellungen gespeichert und mit “Revert” können die aktuell gespeicherten Einstellungen wiederhergestellt werden (Abbildung 10).

Abbildung 10: Zu sehen sind die Einstellungen für die virtuelle Beispielumgebung. Alle Einstellungen besitzen eine kurze Beschreibung und ein Textfeld zur Eingabe der gewünschten Werte.

Eine besondere Einstellungsmöglichkeit bieten die “Range Reading Angles”. Hier werden die Winkel eingestellt, um die der Roboter bzw. der Kopf des Roboter sich dreht und Messungen durchführt (Abbildung 12).

Um einen neuen Winkel hinzuzufügen, wird der Button “Add Angle” genutzt. Ein neuer Winkel mit dem Wert 0.0 wird dann hinzugefügt. Um diesem dann den gewünschten Winkel zu geben muss nun in dem Feld “Selected Angle” der Wert geändert werden. Das Grad-Zeichen muss nicht zwingend hinzugefügt werden, dies geschieht automatisch beim Betätigen der Entertaste, wodurch auch der eingegebene Wert übernommen wird.

Mehrere Winkel können jedoch schnell hinzugefügt werden, indem zunächst ein Startwinkel hinzugefügt oder durch die Tabelle oder den Kreis ausgewählt wird. Dieser Winkel sollte der kleinste gewünschte Winkel sein. Im nächsten Schritt wird im Feld “Shift Angle by” ein Abstand eingegeben, welcher definiert, wie weit die Winkel auseinander liegen sollen. Damit wird nach jedem Klick auf den unteren Button “Add Angle” ein Winkel hinzugefügt, der um den eingegebenen Abstand verschoben wurde.

Max. sensor range	255.0
Bad range delta	10.0
Move rotation noise (deg)	1.0
Move distance noise	1.7
Range reading noise	1.0
Min. move distance	10.0
Max. move distance	20.0
Rotation speed	100.0
Travel speed	50.0
Object clearance	25.0
Color cutoff	5
Light cutoff	40
<input type="button" value="Autotune cutoffs"/>	
Move command	GET_LINE_MOVE
Search start angle	5
Verbose NXT	false
Range reading angles	

Abbildung 11: Zu sehen sind die Einstellungen für die NXT Roboter. Neben den Einstellungen, die bereits aus der virtuellen Umgebung stammen, kommen noch Einstellungen hinzu, die speziell den Roboter betreffen. So müssen zum Beispiel die Sensoren eingestellt werden. Hierzu existiert aber auch ein automatisches Programm (“Autotune cutoffs” Button).

Um einen Winkel zu ändern oder zu löschen, wird der entsprechende Winkel in der Tabelle oder im Kreis per Doppelklick ausgewählt. Der Winkel wird im Feld “Selected Angle” angepasst und die Eingabe wird mit der Entertaste bestätigt. Um den ausgewählten Winkel zu löschen, wird der Button “Delete Angle” betätigt.

4 Roboter

Um mit einem NXT Lego Mindstorm Roboter eine Lokalisierung durchzuführen, sollte der Abschnitt “Monte-Carlo-Lokalisation auf Lego Mindstorms NXT” der Dokumentation “Monte-Carlo-Lokalisation” gefolgt werden. Eine Anleitung, um einen geeigneten Roboter zu konstruieren, liegt dem Projekt bei. Wichtig ist insbesondere, dass in der Datei “McIdaemon.java” die Motoren und die Sensoren korrekt zugewiesen sein müssen.

Der Variablen “HEAD_MOTOR” muss der Motor zugewiesen werden, der den Kopf des Roboters dreht. Bei den Lego Mindstorm Robotern ist dies der Motoanschluss “A”, “B” oder “C”. Die anderen beiden Motoren, die für die Räder bzw. Raupenkettens zuständig sind, müssen ebenfalls auf diese Weise zugewiesen werden (zu den Variablen “LEFT_MOTOR” und “RIGHT_MOTOR”). Eine mögliche Konfiguration ist auf der nachfolgenden Abbildung gezeigt (Abbildung 12).

Auf die gleiche Weise müssen der Ultraschallsensor, der Farbsensor und der Lichtsensor zugewiesen werden. Dafür stehen die Variablen “ULTRASONIC_PORT”, “COLOR_PORT” und “LIGHT_PORT” bereit.

Als weitere Einstellungen sind noch die Variablen “WHEEL_DIAMETER” für den Durchmesser der Reifen und “TRACK_WIDTH” für die Spurbreite einzustellen. Weiterhin muss der Parameter “HEAD_GEAR_RATIO”, welcher beschreibt, welches Übersetzungsverhältnis der Kopfmotor zum tatsächlichen Kopf besitzt, eingestellt werden. Ebenso muss das Übersetzungsverhältnis beim Durchmesser der Reifen mit einbezogen werden, sollten Zahnräder verwendet werden.

```
private static final RegulatedMotor HEAD_MOTOR = Motor.B;  
private static final RegulatedMotor LEFT_MOTOR = Motor.A;  
private static final RegulatedMotor RIGHT_MOTOR = Motor.C;  
private static final SensorPort ULTRASONIC_PORT = SensorPort.S1;  
private static final SensorPort COLOR_PORT = SensorPort.S3;  
private static final SensorPort LIGHT_PORT = SensorPort.S4;  
private static final int HEAD_GEAR_RATIO = -1;  
private static final double WHEEL_DIAMETER= 3.4d;  
private static final double TRACK_WIDTH= 18.1d;  
private static final int CUTOFF_MEASUREMENTS = 11;
```

Abbildung 12: Die Einstellungen, die sich im Daemon befinden, müssen nur ein einziges Mal angepasst werden. Diese beziehen sich lediglich auf den Aufbau des Roboters bezüglich seiner Maße und der Anschlüsse.