



Universität Hamburg
Fakultät für Mathematik,
Informatik und Naturwissenschaften
Department Informatik

ENTWICKLUNG EINER JAVA SIMULATIONSUMGEBUNG FÜR LEGO MINDSTORMS NXT ROBOTER

Masterarbeit zur Erlangung des akademischen Grades Master of Education
im Studiengang Lehramt an Gymnasien M.Ed.

Pamina Maria Berg

Hamburg, den 3. Januar 2016

Erstgutachter

Dr. Guido Gryczan

Zweitgutachter

Jun.-Prof. Dr. Maria Knobelsdorf

Betreuer

Fredrik Winkler

Till Aust

ABBILDUNGSVERZEICHNIS

2.1	Der Startbildschirm des LEGO Mindstorms NXT-Programms	8
2.2	Fahren eines Roboters auf einer schwarzen Linie	8
2.3	Der Startbildschirm von Enchanting	9
2.4	Beispiel eines Enchanting Programms	10

INHALTSVERZEICHNIS

1	Einleitung	1
2	Ausgangssituation	3
2.1	Das LEGO Mindstorms NXT System	3
2.2	Die Arbeit mit LEGO-Robotern im Unterricht	4
2.2.1	Curriculare Einordnung	4
2.3	Bisher verfügbare Softwarelösungen	7
2.3.1	LEGO Mindstorms NXT	7
2.3.2	Enchanting	10
2.3.3	BlueJ	10
2.4	leJOS	10
3	Anforderungen an die Neuimplementierung	12
3.1	Schülerperspektive	13
3.2	Lehrkraft	14
3.3	Erweiterbarkeit	14
4	Entwickelte Software	15
4.1	Beschreibung der Software	15
4.1.1	Lichtsensoren	15
4.2	Softwarearchitektur	15
5	Zusammenfassung und Ausblick	16

KAPITEL 1

EINLEITUNG

Die Arbeit mit Robotern bereichert den Informatikunterricht und das Nachmittagsprogramm vieler Schulen seit Jahren. Auf spielerische Art und Weise sollen Schülerinnen und Schüler (im Folgenden SuS abgekürzt) mit einfachen Konstrukten der objektorientierten Programmierung umzugehen lernen. Dies geschieht zum einen mit Drag-and-Drop Softwareangeboten wie das standardmäßig mit ausgelieferte NXT Mindstorms Tool , oder auch Enchanting. Zum anderen bietet sich ab der Mittelstufe (Klasse 7 – 10) die Arbeit mit BlueJ zur Erstellung erster selbstgeschriebener Programme an. Hierzu kann eine BlueJ Extension genutzt werden, die mit der Java Virtual Machine leJOS NXJ für NXT Roboter arbeitet.

Referenz:NXT
Software
Referenz: En-
chanting

Doch immer wieder stoßen Lehrkräfte in den Schulen auf Hardwareprobleme jeglicher Art, wie zum Beispiel das Fehlen von einer ausreichenden Anzahl an Robotern im Unterricht oder auch fehlende Firmwareupdates oder defekte Sensoren.

Auch ist das Zusammen- und wieder Auseinanderbauen der Roboter ein Aufwand, der nicht für den alltäglichen Unterricht geeignet ist. Um kleine Aufgaben mit Java zu lösen, wie zum Beispiel das Fahren einer S-Kurve oder das Anhalten auf einem bestimmten Punkt, muss bisher immer erst der Roboter gestartet, der Code in BlueJ verfasst, auf den Roboter übertragen und dieser dann zu einem geeigneten Parcours gebracht werden.

Um diese Probleme zu umgehen und den Einstieg in die objektorientierte Programmierung über die Benutzung von LEGO Mindstorms Robotern zu ermöglichen, soll im Rahmen dieser Masterarbeit eine Simulationsumgebung für die Arbeit mit LEGO Mindstorms NXT Roboter entstehen.

Referenz

KAPITEL 2

AUSGANGSSITUATION

2.1. Das LEGO Mindstorms NXT System

Die ersten computergesteuerten LEGO Produkte wurden bereits 1986 veröffentlicht. In einer Zusammenarbeit von LEGO Education und dem Massachusetts Institute of Technology (MIT) wurde LEGO TC LOGO entwickelt. Dies war eine spezielle Abwandlung der Programmiersprache LOGO, mit der zusammengesetzte LEGO-Modelle gesteuert werden konnten [Rol14].

Die Entwicklung eines programmierbaren LEGO-Steins begann 1988 und erreichte ihren Höhepunkt mit der Vorstellung des ersten MINDSTORMS Systems im Januar 1998, bei der der LEGO MINDSTORMS RCX Intelligent Brick, ein Microcomputer und somit das Kernstück des RCX-Systems, und das Robotics Invention System im Museum of Modern Art in London vorgestellt wurden.

Bereits zwei Monate nach Verkaufsstart wurde die FIRST LEGO League (FLL) gegründet – eine Zusammenarbeit zwischen LEGO und FIRST (For Inspiration and Recognition of Science and Technology), die den Grundstein für die heute noch bestehende Wettbewerbsliga legte [Rol14].

Die Vorstellung und der Verkaufsstart der Nachfolge-Roboter des RCX-Systems, den LEGO MINDSTORMS NXT Robotern, fand im August 2006 statt. Diese damals neu entwickelten Roboter sind auch, dank eines Updates in 2009, fast zehn Jahre später noch in den Schulen und Universitäten zu finden. Im April 2005 fand die erste FLL Weltmeisterschaft in Atlanta, Georgia, statt und bis heute bieten die Weltmeisterschaften einen

Anlaufpunkt für Jugendliche auf der ganzen Welt, die ihr Können und ihre Roboter auf die Probe stellen wollen [Lego].

2.2. Die Arbeit mit LEGO-Robotern im Unterricht

Die Robotik als Teilgebiet der Informatik in der Schule gewinnt zunehmend an Bedeutung. Nicht nur die Programmierung steht im Vordergrund, sondern auch die Kompetenz, Probleme unter bestimmten Aufgabenstellungen zu lösen. BERNIS und SCHMIDT, die Verfasser eines der Kernwerke für den Unterricht mit dem LEGO MINDSTORMS System, schreiben, dass bei der Arbeit mit Robotern im Unterricht „durch eine konkrete Aufgabenstellung, das problemspezifische Konstruieren und Programmieren eines Roboters sowie durch das letztendliche Testen der gesamte Ablauf eines Problemlösungsprozesses kennen gelernt“ [Ber10, S.2] wird.

Es erfolgt nun zunächst die curriculare Einordnung des Themas Roboter im Unterricht in die Bildungspläne aus Hamburg. Danach werden verschiedene Arbeitsmethoden im Zusammenhang mit dieser Art von Unterricht vorgestellt.

2.2.1. Curriculare Einordnung

Die fachlichen Inhalte, die mithilfe des Einsatzes von Robotern im Unterricht vermittelt werden können, sind sowohl im Bildungsplan der Sekundarstufe I für das Gymnasium (Klasse 7 – 10) und die Stadtteilschule (Klasse 7 – 11), als auch im Bildungs- bzw. Rahmenplan für die Gymnasiale Oberstufe, der Sekundarstufe II, verankert.

Gymnasium Sekundarstufe I (7 – 10) Informatik Wahlpflicht

Der Einsatz von Robotern kann auf die Module 2 und 3 des Bildungsplanes für die Sekundarstufe I des Gymnasiums bezogen werden. Hier heißt es

Modul 2

- *Abläufe analysieren und umgangssprachlich beschreiben, zu Algorithmen formalisieren und mit einer formalen Sprache implementieren*
- *Umgang mit einer einfachen Entwicklungsumgebung*
- *Testen, Ergebnisse interpretieren und bewerten*
- *Grundlagen der prozeduralen Programmierung: Sequenz, Alternative, Wiederholung, Prozedur bzw. Funktion*

Modul 3 – Kontext Daten und Prozesse

- *Grundlagen der prozeduralen Programmierung*
- *Abläufe formalisieren*
- *Algorithmen mit einer formalen Sprache implementieren*
- *Testen, Ergebnisse interpretieren und bewerten*

Zitat Bildungsplan

STS Sekundarstufe I (7 –11) Informatik

Ähnlich ist es im Bildungsplan für die Stadtteilschulen. Hierbei werden allerdings innerhalb der Module verbindliche Inhalte formuliert.

So sind im Modul 2 die Inhalte *Algorithmen* und *prozedurale Programmierung: Sequenz, Alternative, Wiederholung, Funktion* zu finden, die, wie der Bildungsplan vorschlägt, in das Unterrichtsvorhaben "*Wir lassen Roboter arbeiten*" integriert werden können.

Im Modul 3 sollen im zweiten Halbjahr die verbindlichen Inhalte

- *Grundlagen der prozeduralen Programmierung*
- *Abläufe formalisieren*
- *Algorithmen mit einer formalen Sprache implementieren*
- *Testen, Ergebnisse interpretieren und bewerten*

Quelle

unterrichtet werden.

Gymnasiale Oberstufe – Vorstufe

In der Vorstufe sollen sich die SuS mit den Inhalten des Moduls *Daten und Prozesse* beschäftigen. Diese beinhalten das Analysieren und umgangssprachliche Beschreiben von Abläufen, das Strukturieren von Daten sowie die Verwendung von Variablen und Parametern, das Formalisieren von Abläufen, die Grundlagen der prozeduralen Programmierung, die Implementation von Algorithmen mit einer formalen Sprache, das Testen

Quelle

und das Interpretieren und Bewerten von Ergebnissen.

Gymnasiale Oberstufe – Studienstufe

In der Studienstufe gibt es den verbindlichen Inhalt *Objektorientierte Modellierung*, der die folgenden Punkte umfasst:

- *Idee des OO-Konzepts mit Objekten und ihrer Kommunikation, Vererbung und Nutzerbeziehung*
- *Erarbeitung der Sprachelemente der verwendeten objektorientierten Programmiersprache, Berücksichtigung von Programmierkonventionen, Nutzen von Bausteinen/Libraries*
- *Nutzung einer IDE mit UML-Diagrammen und Quellcode zur schrittweisen Implementierung eines Informatiksystems.*

Zitat

Genereller Einsatz von Robotern als Hilfsmittel im Unterricht

Da laut Bildungsplan die projektorientierte Arbeit in Informatik in der Studienstufe einen bedeutenden Stellenwert hat, bietet es sich natürlich an, eine Projektarbeit zum Thema *Einsatz von Robotern* mit Bezug auf verschiedene Ligen (schulisch und universitär) zu gestalten, bei der als Produkt eine bestimmte Aufgabe mit den Robotern erledigt werden soll.

2.3. Bisher verfügbare Softwarelösungen

In diesem Kapitel wird nun eine Auswahl von verfügbaren und im Unterricht getesteten Softwarelösungen zur Programmierung der NXT Roboter vorgestellt.

2.3.1. LEGO Mindstorms NXT

Das LEGO Mindstorms NXT-Programm ist eine von der Firma LEGO zur Verfügung gestellte Programmieroberfläche für NXT-Roboter.

Die Abbildung 2.1 zeigt den Startbildschirm der Entwicklungsumgebung. Dieser stellt neben zwei Videoanleitungen als Hilfe für Anfänger auch das *Robo Center* zur Verfügung, in dem verschiedene von LEGO vorgeschlagene Roboter-Modelle, die mithilfe des Bausatzes des NXT zusammengesetzt werden können, vorgestellt werden .

Sobald eine neue oder bestehende Programmdatei geöffnet wird, können die SuS mithilfe von Blöcken, die unter anderem der Bewegung und Sensorik dienen und dabei einfache Steuerungsmechanismen sowie Elemente der Regelungstechnik zur Verfügung stellen, ihren Roboter programmieren.

Was passiert hier noch?

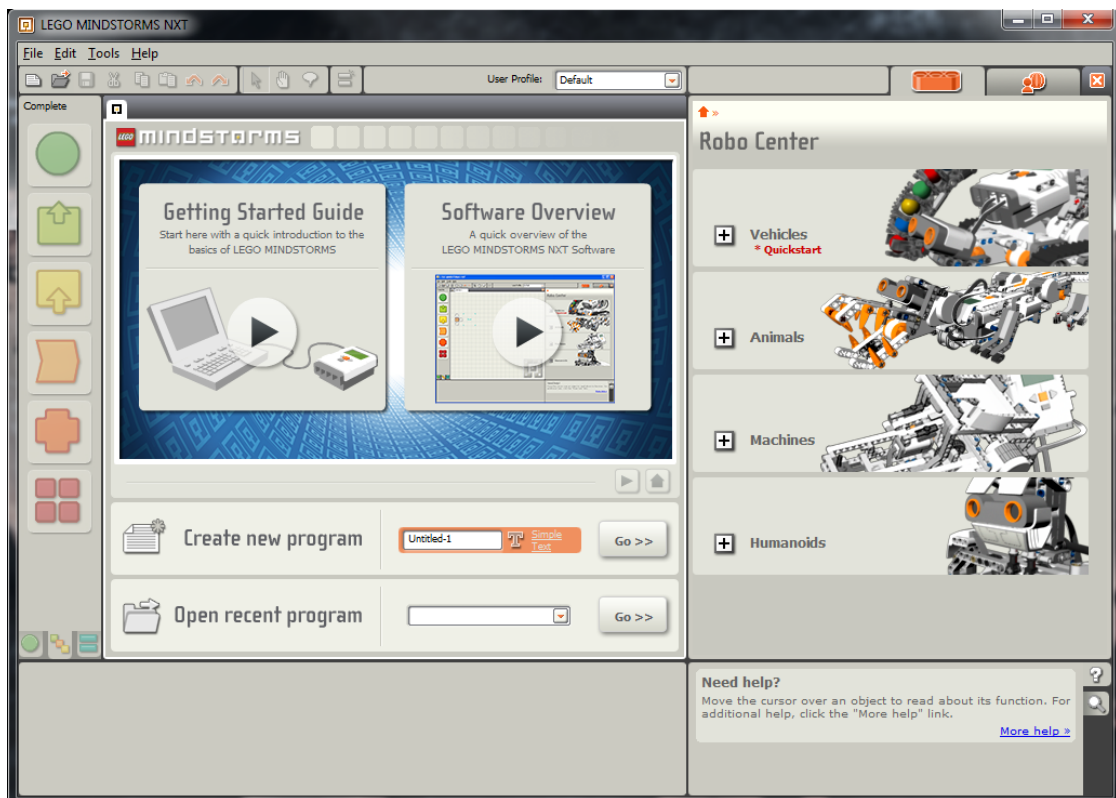


Abbildung 2.1: Der Startbildschirm des LEGO Mindstorms NXT-Programms

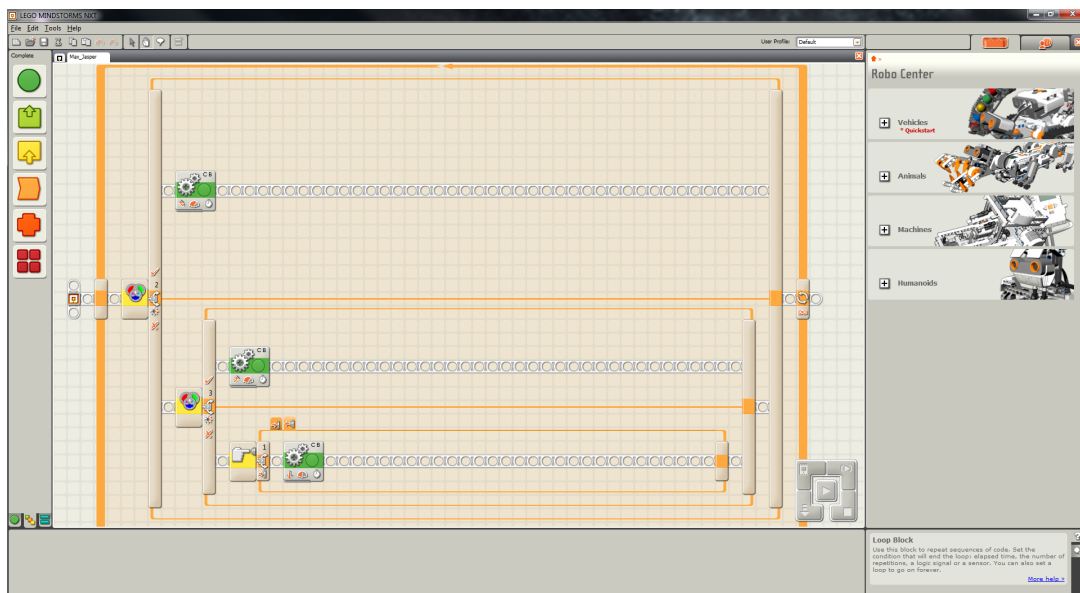


Abbildung 2.2: Fahren eines Roboters auf einer schwarzen Linie

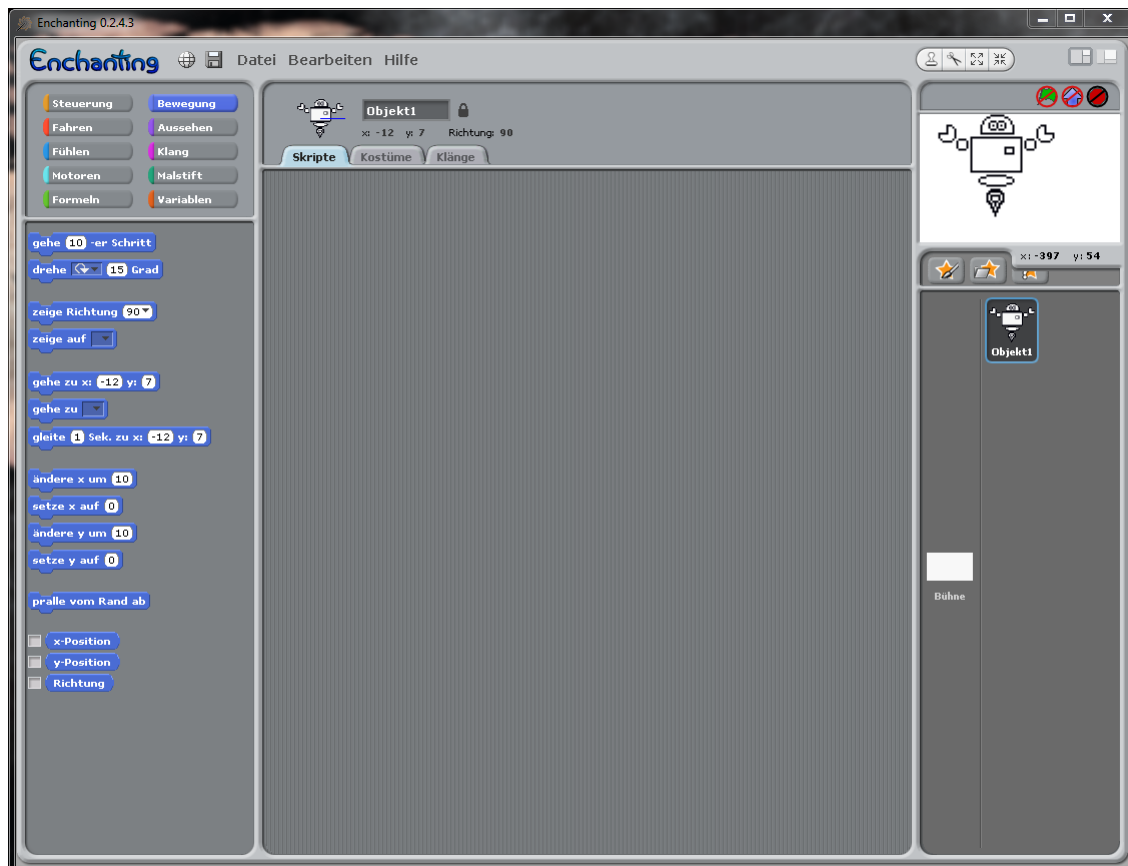


Abbildung 2.3: Der Startbildschirm von Enchanting

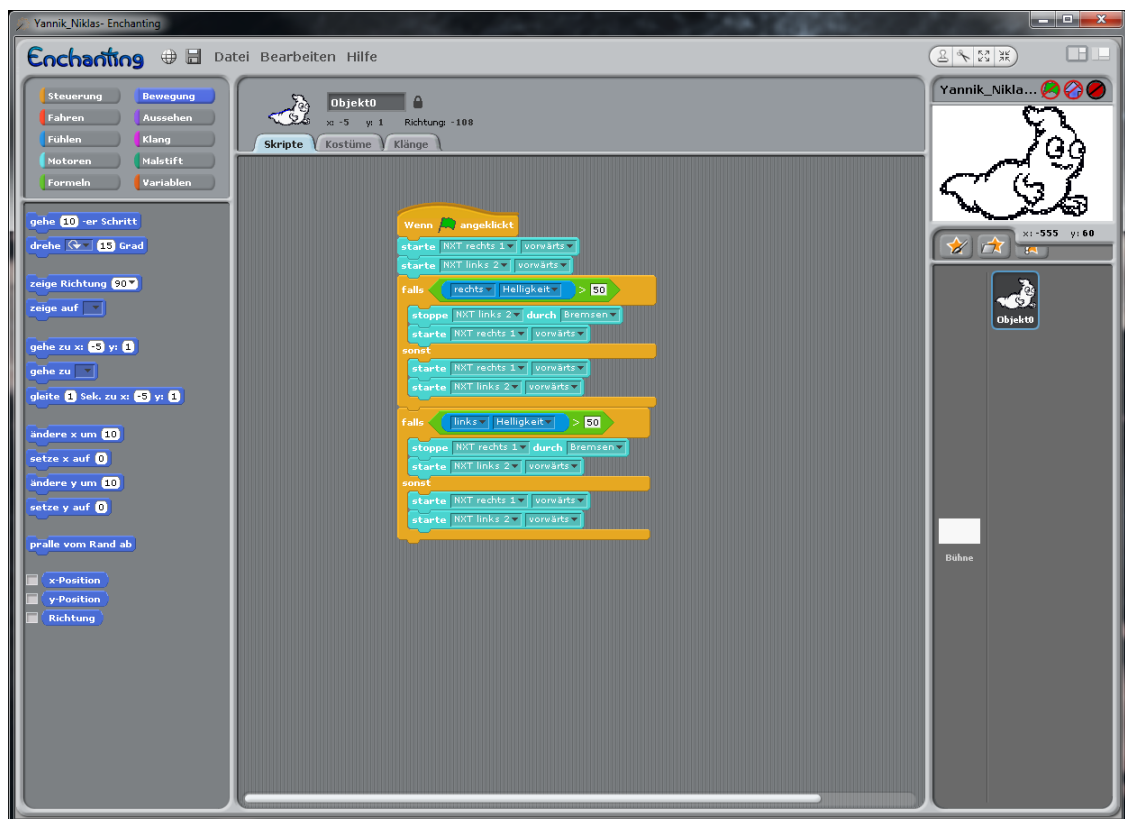


Abbildung 2.4: Beispiel eines Enchanting Programms

2.3.2. Enchanting

Enchanting ist eine an *Scratch* anknüpfende Entwicklungsumgebung. Wie das in 2.3.1 beschriebene Programm wird auch hier mithilfe von Drag-and-Drop gearbeitet.

2.3.3. BlueJ

2.4. leJOS

leJOS ist eine kleine Java Virtual Machine und stellt alle Klassen der NXJ API zu Verfügung, mithilfe derer Lego Mindstorms NXT Roboter in Java programmiert werden

können [leJOS].

KAPITEL 3

ANFORDERUNGEN AN DIE NEUIMPLE- MENTIERUNG

Nicht nur in der Schule, sondern insbesondere in der Wissenschaft spielt Simulation in der Robotik eine wichtige Rolle [Her12, S.13]. Die Anforderungen, die an die wissenschaftlichen Simulationsumgebungen der Robotik gestellt werden, fassen HERTZBERG, LINGEMANN und NÜCHTER wie folgt zusammen:

- *„Die Umgebung muss hinreichend gut simuliert sein. Die Simulation einer Flughafenterminalhalle muss zum Beispiel „zufällig“ umherlaufende Fluggäste mit Gepäck und Transportkarren umfassen.*
- *Der Roboter in seiner Funktionalität muss hinreichend gut simuliert sein. Das betrifft seine Aktionen wie auch seine Sensorik.*
- *Relevante Ungenauigkeit technischer Sensoren und Effektoren muss abgebildet werden. Kann zum Beispiel im realen Bild einer Kamera auf dem Roboter ein Orientierungspunkt im Gegenlicht der Fensterfront unsichtbar werden, muss die Simulation diesen Effekt reproduzieren.*
- *Die „Wahrheit“ im Simulator ist tabu! Natürlich ist im Simulator der Zustand jedes simulierten Objekts präzise bekannt, einschließlich der Position, Richtung und Geschwindigkeit des Roboters. Die Roboterkontrollsoftware darf hierauf nicht zugreifen, um Information über die Umgebung zu erhalten – das geht nur über die simulierten Sensoren. (Für die externe Bewertung des Roboterhaltens ist der*

Vergleich zwischen der Wahrheit im Simulator und der Information in der Roboterkontrollsoftware aber erlaubt.)

- *Der Simulator sollte für den simulierten Roboter die identische Schnittstelle wie der reale Roboter zwischen Roboterkontrollsoftware einerseits und Robotersensorik und -aktuatorik andererseits verwenden; die Roboterkontrollsoftware soll also code-identisch für den realen oder den simulierten Roboter verwendet werden.“*
[Her12, S.14]

In den folgenden beiden Unterkapiteln werden nun die Anforderungen an die Neuimplementierung des Simulator-Prototyps für LEGO Mindstorms NXT Roboter aus Sicht der Schüler und der Lehrer dargestellt, sowie Parallelen und Unterschiede zu den oben dargestellten herausgearbeitet.

3.1. Schülerperspektive

Da es sich bei der Simulationsumgebung um einen Prototypen handelt, der möglicherweise schon ab der fünften, regelmäßig aber ab der siebten Klasse eingesetzt werden soll, steht ein Augenmerk ganz besonders im Fokus: die Einfachheit. Sowohl in der Bedienung des Simulators als auch im Aussehen sollten klare und leicht erkennbare Strukturen vorherrschen.

Des weiteren sollte darauf geachtet werden, dass die SuS für die Programmierung des Roboters und die Benutzung des Simulators eine einheitliche Syntax verwenden können. Hierzu muss die Simulationsumgebung genau die Methoden anbieten, die auch bei dem realen Roboter die Steuerung der Motoren und den Zugriff auf Sensordaten ermöglichen. Dies entspricht dem zweiten Aspekt nach HERTZBERG ET AL., da der simulierte Roboter alle für die Arbeit mit SuS zentralen Funktionen anbieten und sich hinreichend ähnlich wie ein Roboter in der Realwelt verhalten soll. Des weiteren wird hiermit auch der Aspekt der code-identischen Roboterkontrollsoftware erfüllt, da die SuS das selbe Programm für den Simulator, wie auch für den Roboter selbst benutzen können sollten. Ein weiterer Aspekt, den HERTZBERG ET AL. beschreiben sind die Parcours selbst. Diese sollten möglichst realitätsgetreu umgesetzt werden. Das heißt, dass Ausschnitte eines bestehenden realen Wettbewerbsparcours genutzt werden sollten, um für die SuS

eine möglichst realitätsnahe Testumgebung anzubieten.

3.2. Lehrkraft

Für Lehrkräfte ist es erfahrungsgemäß wichtig, dass die Simulationsumgebung alle essentiellen Bausteine der objektorientierten Programmierung im Kontext der Lego-Roboter zur Verfügung stellt. Hierzu gehört das Austesten des Fahrens mithilfe einer Schleife, sowie der verschiedenen Sensoren. Hierzu sollte es eine Auswahl an verschiedenen Parcours geben, damit der Fokus bei jedem Parcours auf einer einzigen Sache liegt. Sollen etwa die angebauten Lichtsensoren getestet werden, so bietet es sich an, den Parcours einfach aus einem weißen Hintergrund und einer schwarzen Linie, die entweder senkrecht in der Nähe des Roboters platziert ist, oder als S-Kurve eine Teilstrecke des realen Labyrinths darstellt, bestehen zu lassen.

Des weiteren ist es wichtig, dass sich Lehrerinnen und Lehrer schnell in die Simulationsumgebung einarbeiten können, um bei Fragen der SuS sofort Hilfe leisten zu können.

3.3. Erweiterbarkeit

KAPITEL 4

ENTWICKELTE SOFTWARE

4.1. Beschreibung der Software

4.1.1. Lichtsensoren

Die Lichtsensoren sind zwei von der leJOS-API vorgegebene Exemplare der Klasse `LightSensor`. Diese sind im Prototyp der Simulation jeweils im Abstand von fünf Pixeln links und rechts von der Spitze des Roboter-Dreiecks angebracht. Da es sich bei der Implementation um einen Prototyp handelt, gibt es noch keine optische Repräsentation der beiden Sensoren am Roboter.

Die Klasse `Roboter` enthält die Methoden zur Abfrage der Positionen der Sensoren im Bezug auf die x- und y-Achse: `gibXLichtRechts()`, `gibYLichtRechts()`, `gibXLichtLinks()` und `gibYLichtLinks()`. Diese werden benötigt, um das Auslesen des Farbwerts des Pixels unter dem jeweiligen Sensor zu ermöglichen.

4.2. Softwarearchitektur

KAPITEL 5

ZUSAMMENFASSUNG UND AUSBLICK

LITERATURVERZEICHNIS

- [Abe01] Michael Abend. “Robotik und Sensorik. Darstellungsschwerpunkt: Selbständige Entwicklung „unscharfer“ Algorithmen zur räumlichen Orientierung (unter Verwendung des LEGO-Mindstorms-Systems)”, *Schriftliche Prüfungsarbeit zur zweiten Staatsprüfung für das Amt des Studienrats*, Berlin, 2001
- [Ber10] Karsten Berns, Daniel Schmidt. *Programmierung mit LEGO MINDSTORMS NXT. Robotersysteme, Entwurfsmethodik, Algorithmen*, Springer Heidelberg Dordrecht London New York, 2010
- [Her12] Joachim Hertzberg, Kai Lingemann, Andreas Nüchter. *Mobile Roboter. Eine Einführung aus Sicht der Informatik*, Springer-Verlag Berlin Heidelberg, 2012
- [Hub07] Peter Hubwieser. *Didaktik der Informatik*, 3. Auflage, Springer-Verlag Berlin Heidelberg, 2007
- [Lego] o.V. URL: <http://www.lego.com/en-us/mindstorms/history>, Abgerufen am 06.11.2015, LEGO, 2015
- [leJOS] o.V. URL: <http://www.lejos.org/nxj.php>, Abgerufen am 14.12.2015, leJOS Java for Lego Mindstorms, 2015
- [Nie99] Jürg Nievergelt. *Roboter programmieren ein Kinderspiel. Bewegt sich auch etwas in der Allgemeinbildung?*, *Informatik-Spektrum* 22:5, S. 364–375, 1999
- [Rol14] Mark Rollins. *Beginning LEGO MINDSTORMS EV3*, Apress, Berkeley, CA, 2014

- [Sch04] Rafael Schreiber. "Der Einsatz von LEGO-Mindstorms im Informatikunterricht der 11. Klasse der Leonard-Bernstein-Oberschule. Sicherung und Transfer grundlegender algorithmischer Strukturen in NQC.", *Schriftliche Prüfungsarbeit im Rahmen der zweiten Staatsprüfung für das Amt des Studienrats*, Berlin, 2004
- [Sto01] Matthias Stolt. "Roboter im Informatikunterricht", 2001
- [Wag05] Oliver Wagner. LEGO Roboter im Informatikunterricht. Eine Untersuchung zum Einsatz des LEGO-Mindstorms-Systems zur Steigerung des Kooperationsvermögens im Informatikunterricht eines Grundkurses (12. Jahrgang, 2. Lernjahr) der Otto-Nagel-Oberschule (Gymnasium)", *Schriftliche Prüfungsarbeit im Rahmen der zweiten Staatsprüfung für das Amt des Studienrats*, Berlin, 2005

"Hiermit versichere ich, dass ich die Arbeit selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel – insbesondere keine im Quellenverzeichnis nicht benannten Internet-Quellen – benutzt habe, die Arbeit vorher nicht in einem anderen Prüfungsverfahren eingereicht habe und die eingereichte schriftliche Fassung der auf dem elektronischen Speichermedium entspricht."

Hamburg, 3. Januar 2016

.....
Pamina Maria Berg