

Ingenieursprojekt

für

Produktionstechnische Schwerpunktsfächer

Abstandsregelung am Beispiel eines LEGO MINDSTORMS
EV3-Roboters

Aufgabensteller:

Professor Dr.-Ing. Peter Zentgraf

Bearbeitet von:

Hofmann Tobias, Matrikelnummer 817598

Constantin Maier, Matrikelnummer 820923

Schiefner Dominik, Matrikelnummer 819752

Erklärung zu Ingenieursprojekt

Name, Vorname der Studierenden: Hofmann Tobias

Maier Constantin

Schiefner Dominik

Hochschule Rosenheim

Studiengang: Bachelor Produktionstechnik

Hiermit erklären wir, dass wir das Ingenieursprojekt mit dem Thema „Abstandsregelung am Beispiel eines LEGO MINDSTORMS EV3-Roboters“ selbständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen oder Hilfsmittel benützt, sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

.....
(Ort, Datum)

.....
(Hofmann Tobias)

.....
(Ort, Datum)

.....
(Maier Constantin)

.....
(Ort, Datum)

.....
(Schiefner Dominik)

Zielsetzung

Ziel des Ingenieursprojekts ist vorrangig die Auslegung eine Abstandsregelung am Beispiel eines LEGO MINDSTORMS EV3-Roboters, wie es in der Automobilindustrie üblich ist. Das System soll stabil und robust gegen Störungen sein.

Inhaltsverzeichnis

Inhaltsverzeichnis	4
1. Einleitung	5
2. Wichtigste Bestands- und Zusatzteile	6
2.1 Wichtige Bestandteile des EV3-Roboters.....	6
2.2 Wichtige Zusatzteile	10
3. Verbindung zwischen Roboter und PC aufbauen.....	12
3.1 IP-Adressen ändern	12
3.2 Wichtige Voraussetzungen	13
3.3 Vorgehen beim Anschalten	13
4. Versuch	14
4.1 Hintergrund	14
4.2 Regelungstechnische Entwurfsaufgaben	14
4.3 Versuchsaufbau	15
4.3.1 Sensorüberprüfung, Abstandsvergleich realer- und gemessener Abstand.....	17
4.4 Versuchsdurchführung	18
4.4.1 Streckenidentifikation.....	18
4.4.2 Aufstellen des Regelkreises.....	20
4.4.3 Regler Entwurf	21
4.4.4 Überprüfung der Identifikation mit einer Sollwertregelung.....	23
4.4.5 Störungssimulation	24
4.4.6 Vergleich Realität und Simulation	25
Zusammenfassung.....	26
Abbildungsverzeichnis	27

1. Einleitung

Das Thema dieser Projektarbeit lautet „Abstandsregelung“ am Beispiel eines LEGO MIDSTORMS EV3-Roboters.



Abbildung 1: LEGO MINDSTORMS EV3

Zur Lösung der Aufgabe wurde zuerst ein geeigneter Roboter (Abbildung 2) aus den Teilen des LEGO MINDSTORMS EV3 (Abbildung 1) gebaut.

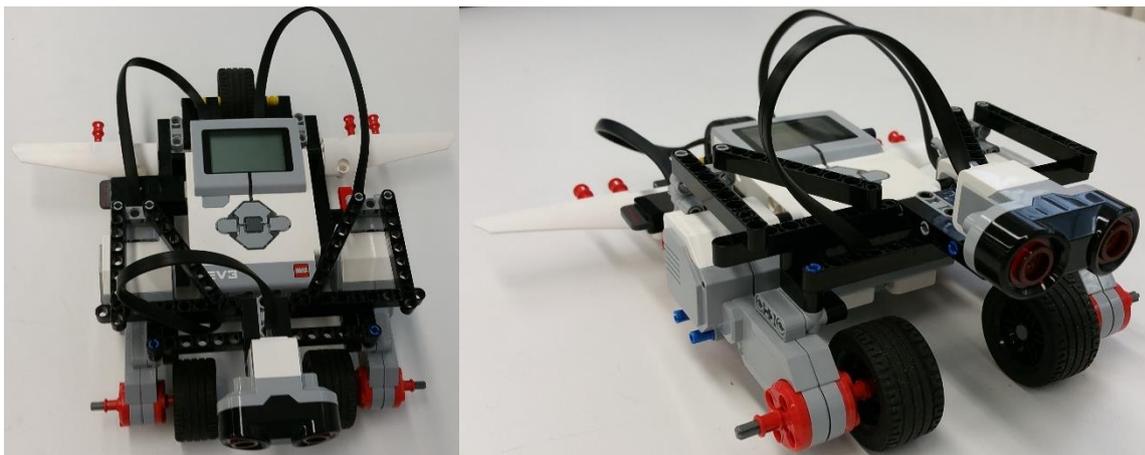


Abbildung 2: Selbstzusammengestellter Roboter

Dieser besteht im allgemeinen nur aus zwei Frontantrieben/Servomotoren mit jeweils einem Rad, einem Hinterrad, dem Gestell, der Steuerung/EV3-Baustein, der Verkabelung und dem Ultraschallsensor.

2. Wichtigste Bestands- und Zusatzteile

2.1 Wichtige Bestandteile des EV3-Roboters

Ultraschallsensor

Ein wichtiger Baustein für die Regelung ist der EV3 Ultraschallsensor. Der Sensor erzeugt Schallwellen und „liest“ deren Echos, um Objekte zu erkennen und deren Entfernung in cm (oder Zoll) zu messen. Darüber hinaus kann er auch Schallwellen aussenden oder auf den Empfang einer Schallwelle warten, die dann ein Programm auslöst.

Daten:

- Misst Entfernungen zwischen 3 und 250 cm (1 bis 100 Zoll)
- Messgenauigkeit: +/- 1 cm (0,394 Zoll)
- Leuchtet dauerhaft beim Senden und blinkt beim Empfangen
- Gibt den Wert „wahr“ zurück bei Erkennen eines anderen Ultraschall-Signals.
- Die EV3-Software erkennt den angeschlossenen Sensor automatisch



Abbildung 3: EV3 Ultraschallsensor

Servomotoren

Ein weiterer wichtiger Baustein sind die großen EV3 Servomotoren. Der große EV3 Servomotor ist ein leistungsstarker Motor mit eingebautem Rotationssensor, der eine Genauigkeit von einem Grad aufweist. Ein spezieller Regelkreis erlaubt es, den Motor mit einer bestimmten Drehgeschwindigkeit zu betreiben.

Daten:

- Messgenauigkeit des Rotationssensors von einem Grad
- Drehgeschwindigkeit maximal 160 bis 170 U/min
- Nennmoment (durchschnittliches Antriebsmoment): ca. 0,2 Nm
- Anfahrmoment (maximales, kurzfristig erreichbares Drehmoment): ca. 0,4 Nm
- Die EV3-Software erkennt den angeschlossenen Motor mit Rotationssensor automatisch

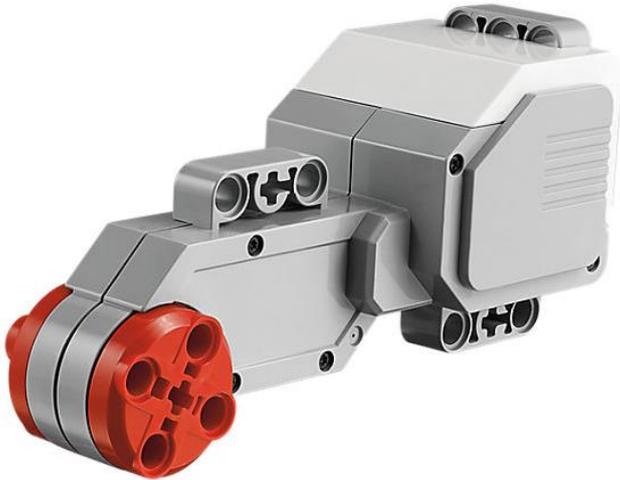


Abbildung 4: EV3 großer Servomotor

EV3-Baustein

Der programmierbare intelligente EV3-Baustein ist das zentrale Element der LEGO® MINDSTORMS® Education EV3 Roboter. Das beleuchtete Bedienfeld mit sechs Tasten zeigt mit verschiedenen Farben den jeweiligen Status des Bausteins an. Weitere Ausstattungsmerkmale: Hochauflösendes Schwarz-Weiß-Display, integrierter Lautsprecher, USB-Port, Mini-SD-Kartenleser sowie jeweils vier Ein- und Ausgänge. Der EV3-Stein unterstützt die Kommunikation über USB, Bluetooth und Wi-Fi (WLAN) zur Datenübertragung zu einem Computer. Zudem verfügt der EV3-Baustein über eine Programmieroberfläche, mit der Programmierung und Messwerterfassung ohne PC möglich sind. Der Baustein ist kompatibel zu Mobilfunkgeräten (iPhone, Android). Er kann entweder mit AA-Batterien oder mit dem EV3-Akku betrieben werden.



Abbildung 5: EV3 Baustein

Details:

- ARM 9-Prozessor mit Linux-basiertem Betriebssystem
- Vier Eingänge zur Messwerterfassung mit einer Abtastrate von maximal 1000 Messungen pro Sekunde
- Vier Ausgänge zur Ausführung von Befehlen
- Integrierter Programmspeicher mit 16 MB Flash-Speicher und 64 MB RAM
- Mini-SDHC-Kartenleser zur Erweiterung des Speichers um 32 GB
- Beleuchtetes, dreifarbiges Bedienfeld mit sechs Tasten; die Farbe zeigt den jeweiligen Status des Bausteins an.
- Hochauflösendes Display (178 x 128 Pixel) zur detaillierten Anzeige von Graphen und zur Beobachtung der Sensordaten
- Hochwertiger Lautsprecher
- Die auf dem Baustein vorgenommene Programmierung und Messwerterfassung kann in die EV3-Software hochgeladen werden.
- Die Kommunikation zwischen Computer und Stein kann über den integrierten USB-Port oder drahtlos mit dem WLAN Dongle bzw. über Bluetooth erfolgen.
- Stromversorgung durch sechs AA-Batterien oder den EV3-Lithium-Ionen-Akku (Kapazität: 2050 mAh)

2.2 Wichtige Zusatzteile

Netgear N150 Wireless Adapter

Der WLAN-Adapter von Netgear wird zur Verbindung zwischen PC und Roboter benötigt, da der EV3-Roboter kein integriertes WLAN besitzt.



Abbildung 6: Netgear N150 Wireless Adapter

Buffalo N600

Der WLAN-Router von Buffalo wird zur Verbindung zwischen PC und Roboter benötigt. Der Router wird nicht bei einem PC oder Laptop mit integrierten WLAN benötigt.



Abbildung 7: Buffalo N600

Diese beiden Komponenten empfehlen wir zum Aufbau einer Verbindung zwischen dem Roboter und eines PCs.

Handy-App

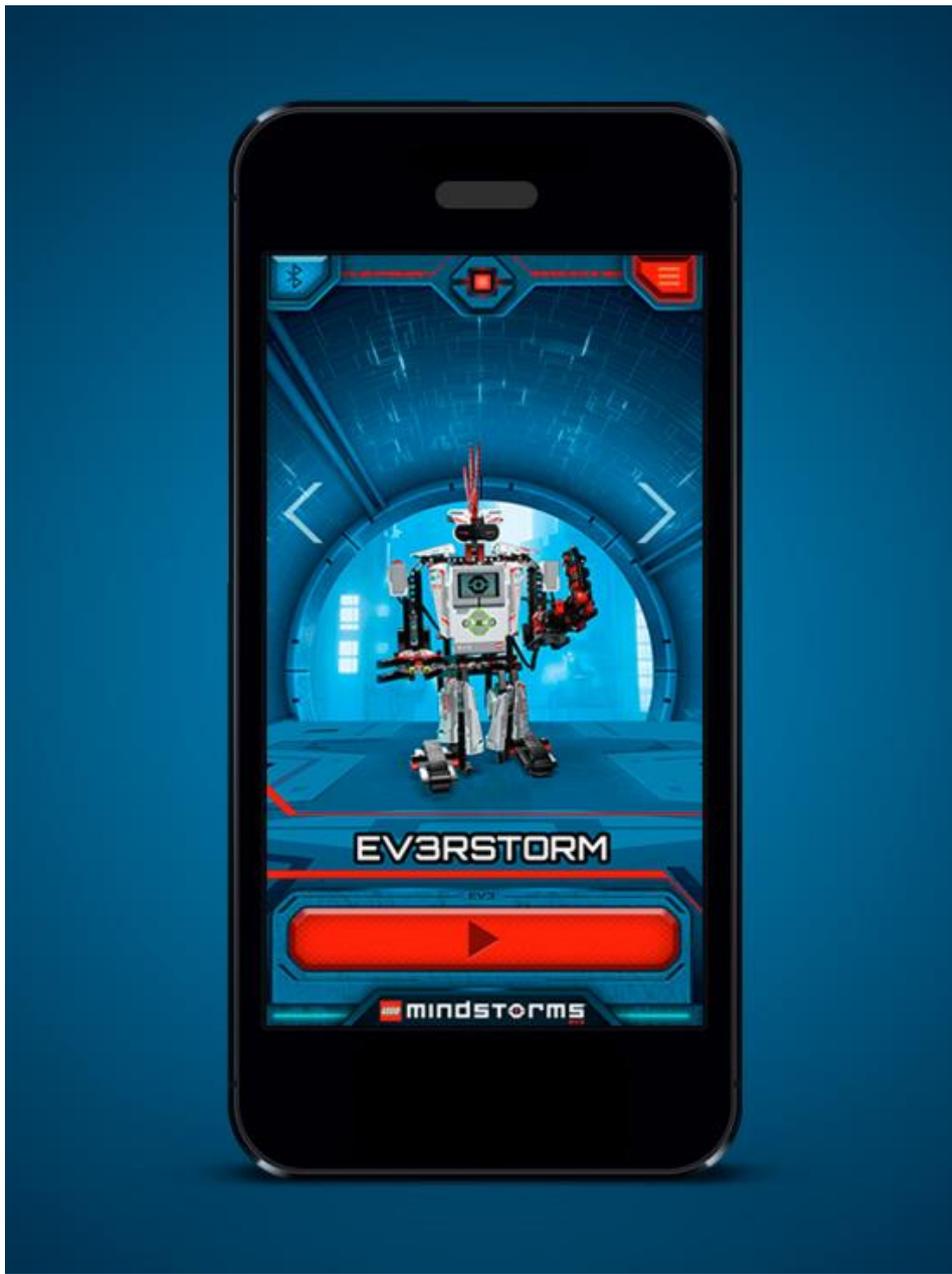


Abbildung 8: LEGO Mindstorms Commander App

LEGO® MINDSTORMS® Commander App auf das Smartphone laden und installieren. Zur Verbindung mit dem Roboter muss die Bluetooth Funktion am Handy und am Roboter (Einstellungen) aktiviert werden. Zur Erstellung einer eigenen Steuerung muss CREATE AND COMMAND YOUR OWN ROBOT ausgewählt werden. Joystick auswählen und Ports einstellen, anschließend wird mit dem Button rechts unten gestartet. EV3 beim Verbinden auswählen und Anweisungen folgen.

3. Verbindung zwischen Roboter und PC aufbauen

3.1 IP-Adressen ändern

1. IP-Adresse von Router herausfinden (auf der Fahne in der Router Rückseite) und an PC vergeben mit anderer Endung (.xx)

Systemsteuerung -> Netzwerkstatus und Aufgaben anzeigen -> Adapter Einstellungen ändern -> LAN-Verbindungen -> Eigenschaften -> Internetprotokoll

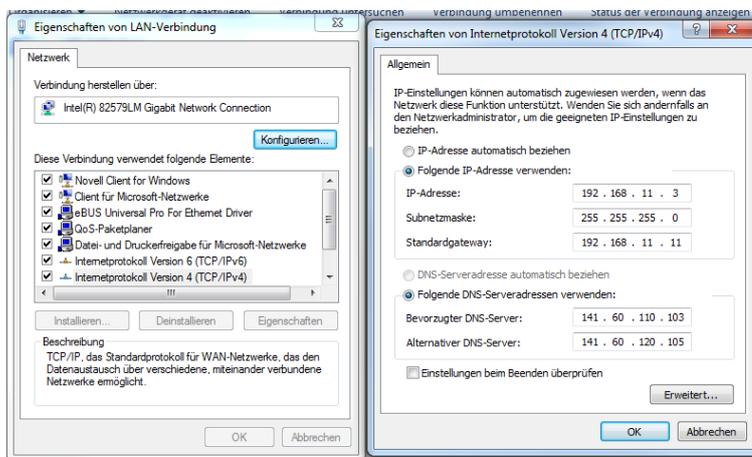


Abbildung 9: IP-Adresse Router PC

2. IP-Adresse von Router in Simulink eingeben mit anderer Endung (.xx).

Simulink -> Tools -> Run on Target Hardware -> Optionen -> Run on Target Hardware

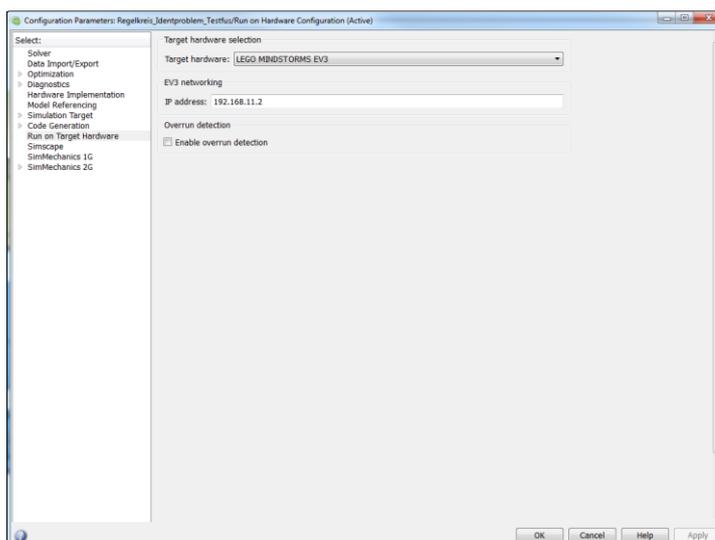


Abbildung 10: IP-Adresse Router Simulink

3. IP-Adresse von Router an EV3 Roboter vergeben (erfolgt automatisch).

3.2 Wichtige Voraussetzungen

- Lego-Software EV3 Firmware V1.06H auf EV3-Roboter installieren.
- Router unverschlüsselt machen: IP Adresse des Routers (192.168.11.1) in Explorer eingeben, Einloggen, Daten auf Router Rückseite (Fahne), nur einmalig

3.3 Vorgehen beim Anschalten

1. Router anschalten
2. PC anschalten
3. Router überprüfen
(192.168.11.1 in Explorer eingeben, Router-Login muss erscheinen)
4. Roboter anschalten
5. Roboter WLAN anschalten
6. Roboter mit Router verbinden
(Hauptmenü auf Schraubenschlüssel-Symbol=>WiFi=>WiFi=>Connections=>Buffalo=>Connect)
7. „Regelkreis_Param“ öffnen und 1x ausführen
8. „Regelkreis“ öffnen, Betriebsart über Schalter auswählen , Starten
9. „Messergebnisse_Plot“ öffnen
(nur bei stationärem Ziel mit definiertem Eingang möglich)

4. Versuch

4.1 Hintergrund

Abstandsregelungen sind schon länger in der Technik verbreitet. Neben Anwendungen in industriellen Vorrichtungen gewinnt es vor allem bei modernen Fahrzeugen an Bedeutung. Durch die stetige Erhöhung der Sicherheit in Fahrzeugen existieren bereits Abstandsregelungen, die Unfälle verhindern bzw. den Fahrkomfort erhöhen sollen. In den nächsten Jahren wird vor allem auch durch das autonome Fahren diese Art von Regelung deutlich an Bedeutung gewinnen.

4.2 Regelungstechnische Entwurfsaufgaben

Der Abstand zum vorausfahrenden Fahrzeug $y(t)$ ist die Regelgröße. Diese wird durch Ansteuerung der Antriebsmotoren mit der Stellgröße $u(t)$ so geregelt, dass der Verfolger dem vorausfahrenden Fahrzeug in einem vorgegebenen Abstand, den Sollwerte $w(t)$, folgt.

1. Da das vorausfahrende Fahrzeug nicht voraussehbar gesteuert wird ist die Regelstrecke nicht durch mathematisches Herleiten von Differentialgleichungen zu ermitteln.
2. Die Beschränkungen der maximal möglichen Stellgröße $u(t)$ sowie des größten messbaren Abstandes aufgrund des Sensors sind zu beachten.
3. Das Einschwingen soll so schnell wie möglich und aus Gründen der Sicherheit ohne Überschwingen erfolgen.

Parallelen zur Vorlesung

Kapitel 1

Aufgaben Regelung:

- Wert Konstant halten (Abstand)
- Störungen beseitigen (Bremsen und Beschleunigen des vorderen Fahrzeugs)

4.3 Versuchsaufbau



Abbildung 11: Versuchsaufbau

Das vorausfahrende Fahrzeug wird per Handy-App gesteuert. Der verfolgende Roboter hält den Abstand mit Hilfe der Regelung.

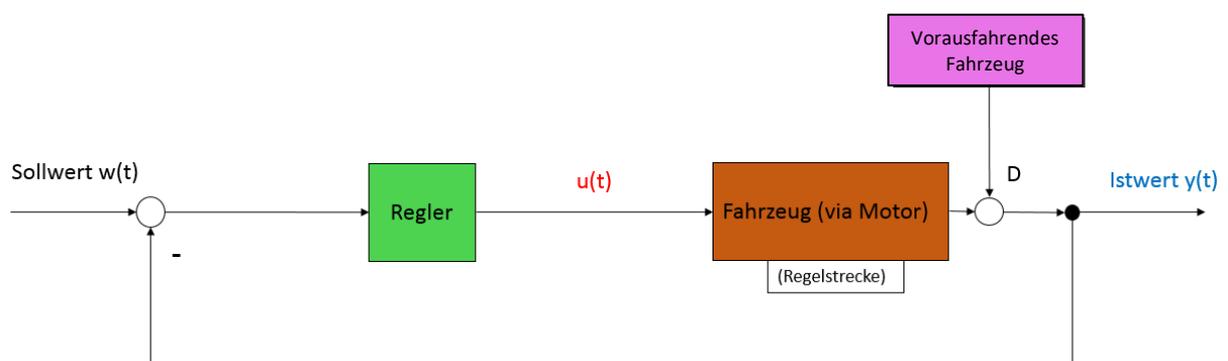


Abbildung 12: Regelkreis

Die Regelstrecke besteht aus zwei Gleichstrommotoren, welche den LEGO Mindstorms Roboter geradlinig bewegen. Die Antriebsmotoren werden mit einer Gleichspannung durch Batterien versorgt. Gleichzeitig wird mit Hilfe des Ultraschallsensors der Abstand zum vorderen Fahrzeug gemessen. Eine Drehbewegung der Motoren bzw. eine Bewegung eines der beiden Roboter bewirkt eine Änderung dieses Abstandswertes

Parallelen zur Vorlesung

Kapitel 1

- Erkennungsmerkmale einer Regelung:
 1. Messen (Abstand)
 2. Vergleichen (Soll-Ist-Abstand vergleichen)
 3. Verstellen (Drehbewegung Motor)
- Regelkreis ist eine geschlossene Wirkungskette
- Ursache (Bewegung Roboter) -> Funktion (Roboter) -> Wirkung (Veränderung Abstandswert)
- System ist Kausal, da Ausgang keinen Einfluss auf Eingang hat
- Regelung ist eine Festwertregelung da Sollwert konstant ist

4.3.1 Sensorüberprüfung, Abstandsvergleich realer- und gemessener Abstand

Messung	Real [cm]	Gemessen [cm]	Differenz [cm]
1	10	10,1	0,1
2	20	20,5	0,5
3	30	32,2	2,2
4	40	41,1	1,1
5	50	51,5	1,5
6	60	61,4	1,4
7	70	72	2
8	80	80,4	0,4
9	90	90,5	0,5
10	100	100,7	0,7

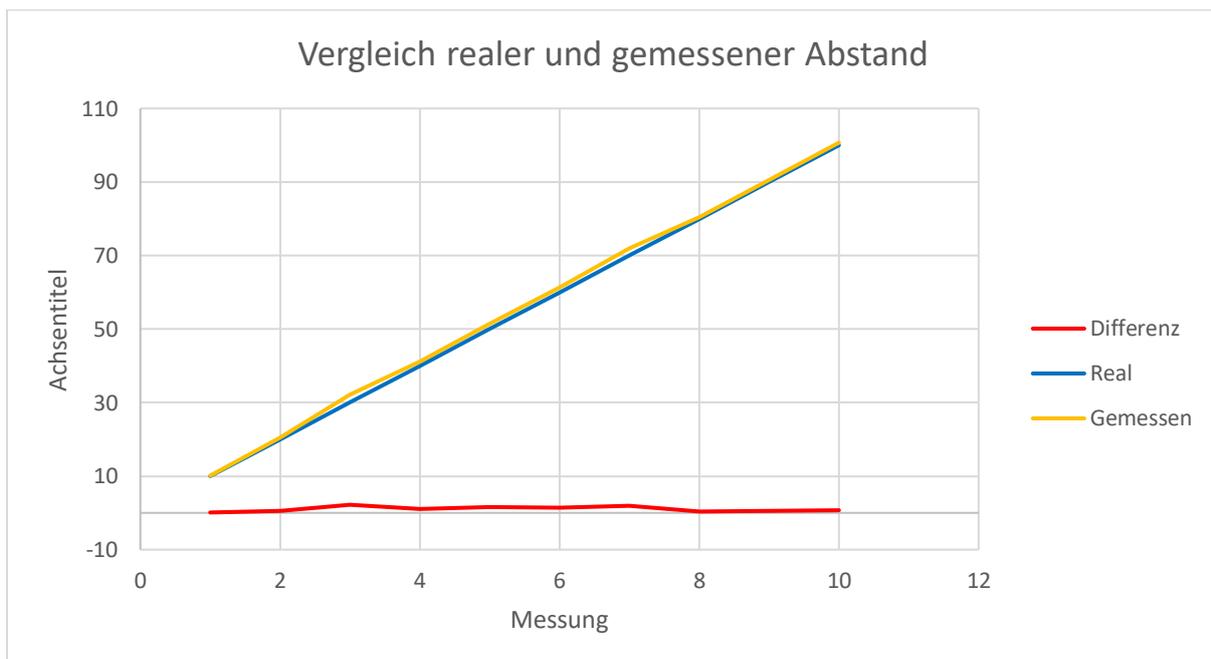


Abbildung 13: Vergleich realer und gemessener Abstand

Bei der Überprüfung des Sensors wurde festgestellt, dass die Abweichung zwischen realen und gemessenen Abstand bis zu 2,2 cm beträgt. Diese Doch sehr hohe Abweichung spielt jedoch nur eine untergeordnete Rolle, da sich der Roboter immer in einem gleichbleibenden Abstand zum vorrausfahrenden Fahrzeug befindet, in dem sich die Kurve annähernd linear verhält. Einen wesentlich größeren Einfluss nimmt die relativ schlechte Auflösung, von 1 cm, des Abstandssensors.

4.4 Versuchsdurchführung

4.4.1 Streckenidentifikation

- Die unbekannt Parameter der Übertragungsfunktion $G_s(s)$ (Regelstrecke) werden durch Analyse der gemessenen Ausgangssignale des Infrarotsensors, mit Hilfe eines möglichst anspruchsvollen vorgegebenen Eingangssignals auf die Motoren, bestimmt.
- Dies wird mit Hilfe des Lern-Tools pzMove durchgeführt.
- Der Streckentyp soll möglichst einfach ausgewählt werden: wenige Pole und Nullstellen.

Streckenübertragungsfunktion:

$$G(s) = e^{-0,07s} * \frac{-3,9531}{s^2 + 16,255 * s + 0,020678}$$

Parallelen zur Vorlesung

Kapitel 4

- Unterschiedliche Darstellungsformen der Übertragungsfunktion $G_s(s)$ (Multiplikativ und Additiv)
- Die Übertragungsfunktion $G_s(s)$ besteht bei uns aus zwei PT1 Gliedern
- Der P-Anteil bewirkt eine proportionale Verstärkung oder der Schwächung. Bei einem PT-Glied geschieht die Änderung nicht verzögerungsfrei und mit Übergang

Kapitel 6

- Ermittlung der Ein-Ausgangsstabilität der Regelstrecke
- Überprüfen von $G_s(s)$ auf Stabilität aufgrund der Pole

Simulinkprogramm zur Streckenidentifikation

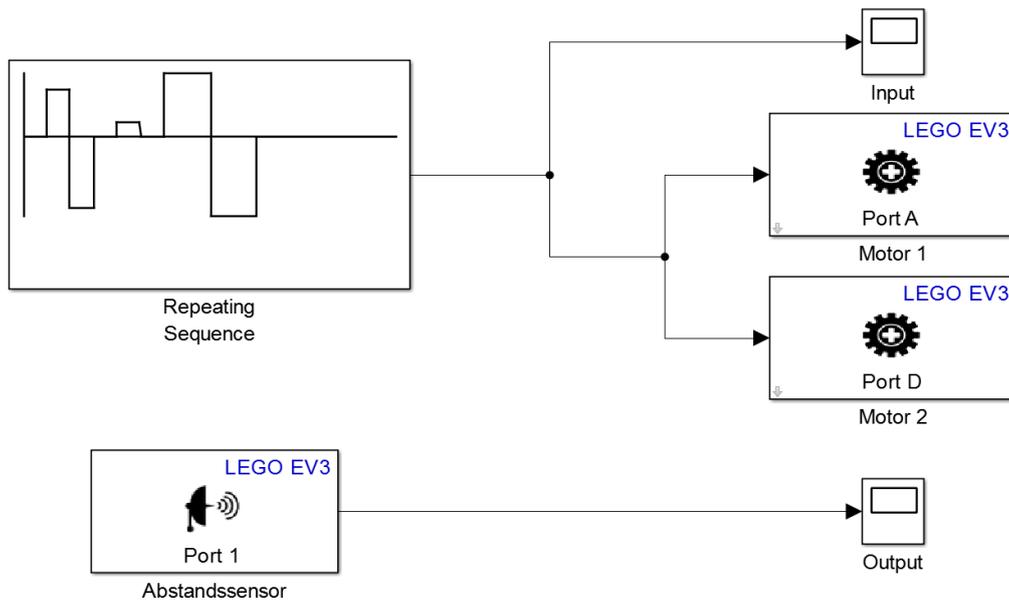


Abbildung 14: Simulink Programm zur Streckenidentifikation

pzMove

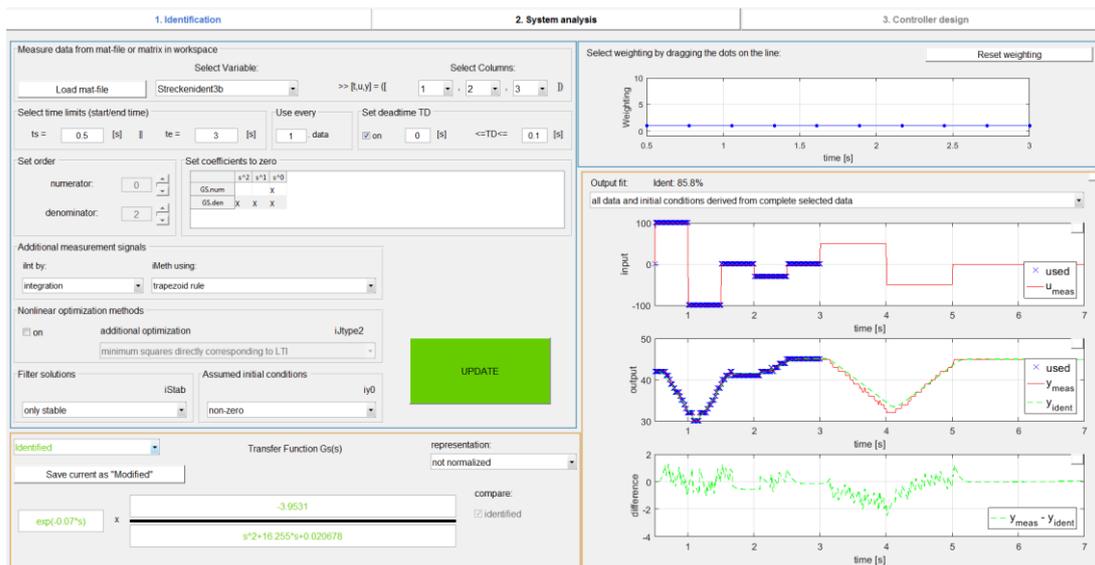


Abbildung 15: pzMove mit aufgenommenen Daten und Streckenübertragungsfunktion

Es wurde ein anspruchsvolles Eingangssignal gewählt um die Strecke eindeutig bestimmen zu können. Die Ein- und Ausgangsdaten wurden aufgezeichnet und anschließend zur Streckenidentifikation verwendet.

4.4.2 Aufstellen des Regelkreises

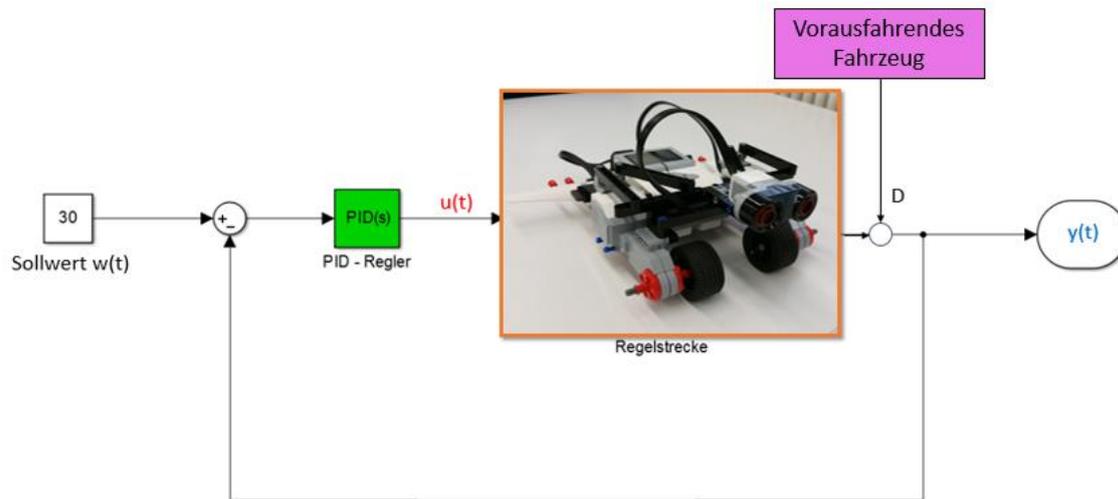


Abbildung 16: Regelkreis

Mit Hilfe von Simulink wurde ein Regelkreis erstellt. In diesem Regelkreis wurde bei der Regelstrecke die über pz-Move berechnete Regelstrecke hinterlegt. Der Entwurf des Reglers wurde ebenfalls mit pz-Move erstellt. Dies erfolgt im nächsten Schritt.

4.4.3 Regler Entwurf

- Vorauslegung des PID-Reglers mit Chien, Hrones und Reswick-Verfahren (no overshoot) und dem Tool pzMove
- Anschließendes Optimieren der Regelparameter durch Durchführung verschiedener Tests

Reglerübertragungsfunktion:

$$G_R(s) = e^{(-0*s)} * 1 * (-10,808 + \frac{-0,013739 * (-0,70945 * s)}{s * (0,1 * s + 1)})$$

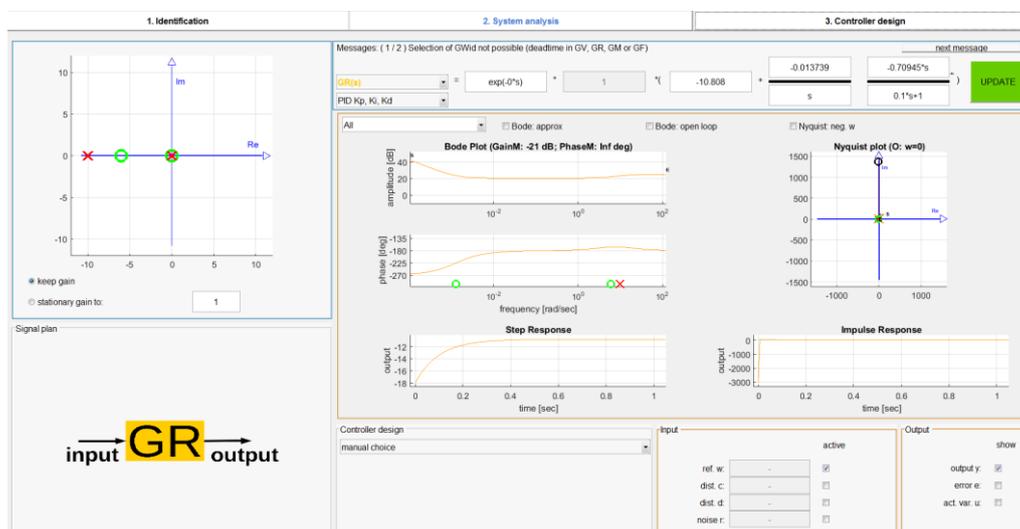


Abbildung 17: Regler Entwurf pzMove

In pz-Move konnten die Pol- und Nullstellen per Mausklick verschoben werden. Gleichzeitig wurden die Auswirkungen auf den Amplituden- und Frequenzgang beobachtet. So konnte das Reglerverhalten im Bezug auf Überschwingen, Regeldifferenz und Reaktionsgeschwindigkeit optimiert werden.

Parallelen zur Vorlesung

Kapitel 7

- Regelstrecke durch System gegeben durch pzMove, möglichst niedriger Zähler- und Nennergrad
 1. Anforderung an Regler: kein Überschwingen, schnell, möglichst ohne Schlupf, usw.
 2. Reglerstruktur (PID) und Regelparameter (pzMove) durch das Verfahren von Nichols
 3. Analyse des Regelkreisverhalten, überprüfen der Simulation mit Realität, Funktion durch Versuche überprüfen
 4. Bewertung des Ergebnis, bei nicht ausreichenden Ergebnis wieder bei 1. beginnen
- Reglerentwurf durch empirische Einstellregeln, da kein mathematisches Modell vorhanden, Regelstrecke aus pzMove

4.4.4 Überprüfung der Identifikation mit einer Sollwertregelung

Klassischerweise wurde die Güte des Reglers als erstes mit einer Sprungantwort überprüft.

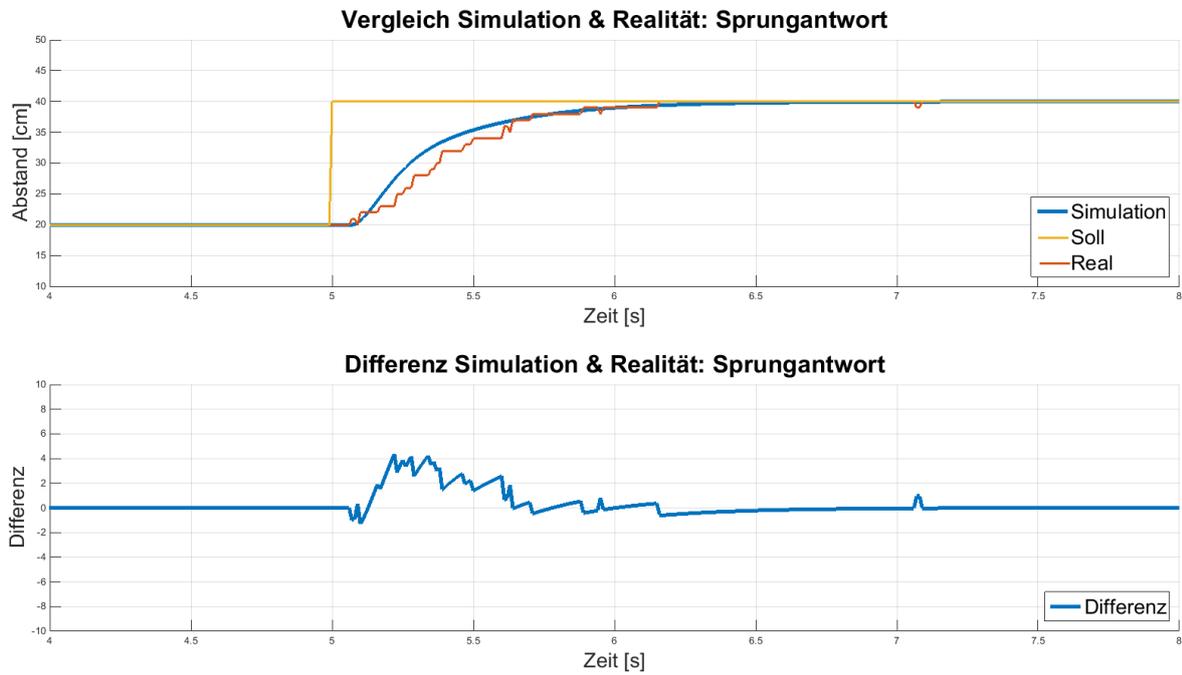


Abbildung 18: Vergleich Realität und Simulation bei einem Einheitssprung

Die Differenz von Simulation und Realität gibt Aufschluss über die Güte des Modells. Hierbei ist auch erkennbar, dass die bleibende Regelabweichung gegen Null geht.

4.4.5 Störungssimulation

In Simulink wurde zudem das Verhalten des Systems bei einer Störung simuliert.

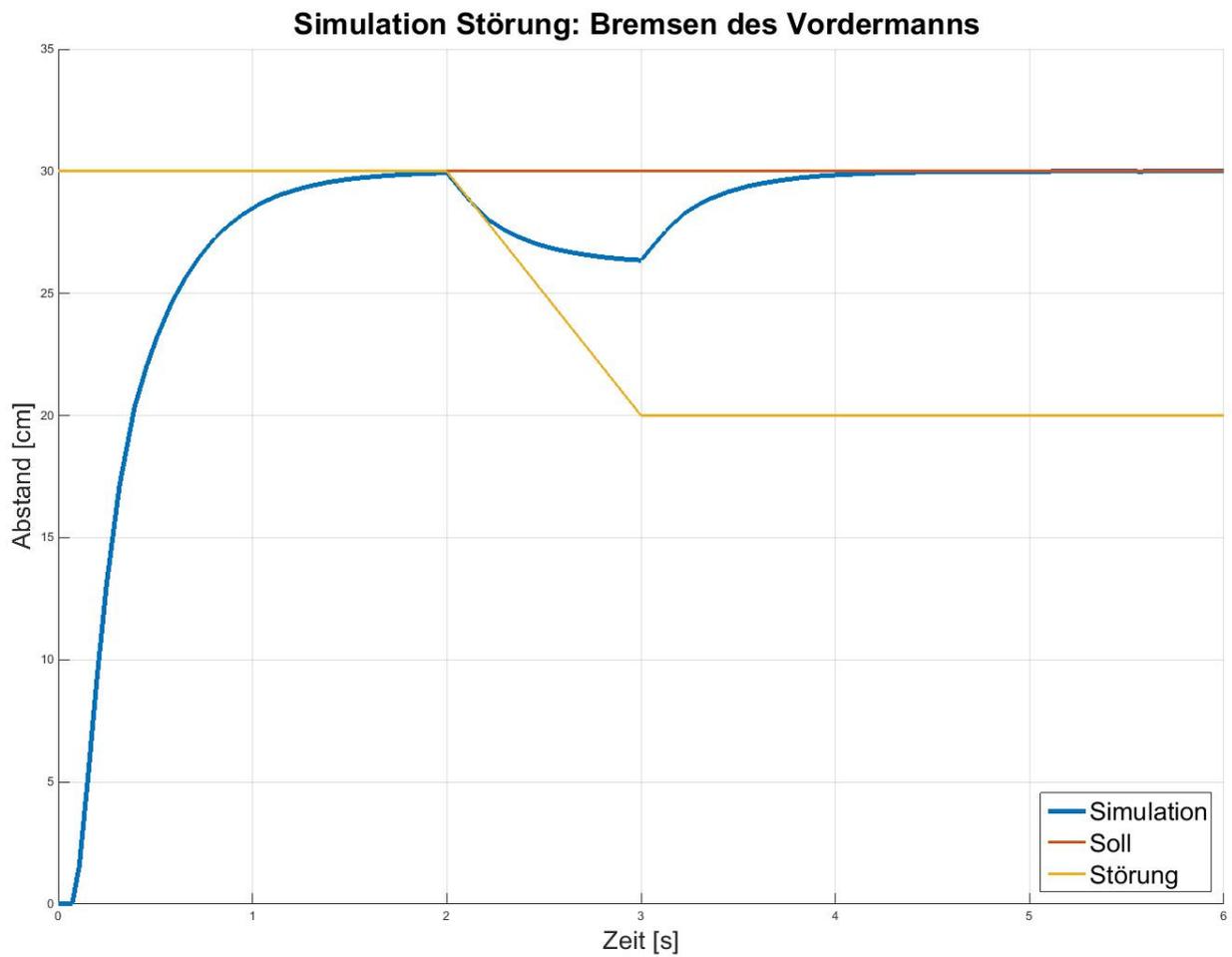


Abbildung 19: Simulation Störung durch Bremsen des Vordermanns

4.4.6 Vergleich Realität und Simulation

Statt eines festen Wertes wurde eine Funktion des Sollabstands auf das reale System geben, während der Roboter vor einer festen Wand stand. Dadurch wird eine Bewegung des vorausfahrenden Fahrzeugs nachgeahmt.

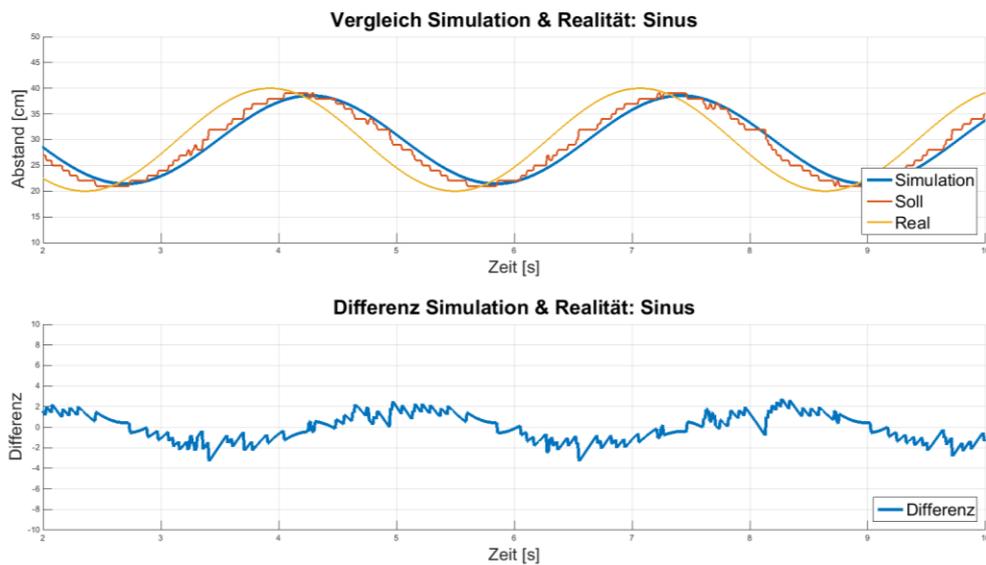


Abbildung 20: Vergleich Realität und Simulation anhand eines Sinus

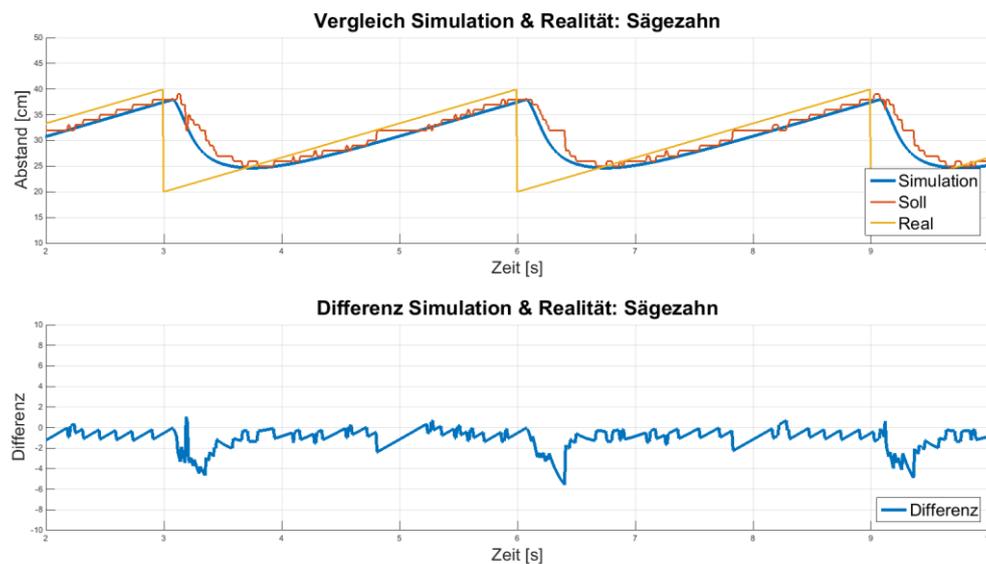


Abbildung 21: Vergleich Realität und Simulation anhand eines Sägezahn

Zusammenfassung

Abschließend lässt sich sagen, dass dieses Projekt ausgesprochen interessant, lehrreich und eine gute Ergänzung zur Theorie der Vorlesung war. Eine der größten Hürden war anfänglich die Vernetzung des EV3 Roboter mit der dazugehörigen Simulinkumgebung, da dies nur mit der passenden Lego-Firmware, sowie den zu Anfangs beschriebenen Wlan-Komponenten, möglich war. Die Streckenidentifikation, als auch die Reglerauslegung, konnten mit dem Lerntool pz-Move und den Erfahrungen aus den Praktikas rasch umgesetzt werden. Trotz der ungenauen Auflösung des Abstandssensors gelang es durch praktische Versuche den Regler weiter zu optimieren.

Anhand dieses Projekts wurde uns ein guter Einblick in die Schwierigkeiten und Herausforderungen der Regelungstechnik in realen, technischen Anwendungen vermittelt.

Es wäre denkbar, dass dieses Projekt anschließend im Rahmen eines Praktikumversuchs aufgegriffen wird oder als praktisches Vorzeigebeispiel in einer Vorlesung gezeigt wird.

Abbildungsverzeichnis

Abbildung 1: LEGO MINDSTORMS EV3.....	5
Abbildung 2: Selbstzusammengestellter Roboter.....	5
Abbildung 3: EV3 Ultraschallsensor	6
Abbildung 4: EV3 großer Servomotor	7
Abbildung 5: EV3 Baustein.....	8
Abbildung 6: Netgear N150 Wireless Adapter.....	10
Abbildung 7: Buffalo N600.....	10
Abbildung 8: LEGO Midstorms Commander App.....	11
Abbildung 9: IP-Adresse Router PC	12
Abbildung 10: IP-Adresse Router Simulink.....	12
Abbildung 11: Versuchsaufbau	15
Abbildung 12: Regelkreis	15
Abbildung 13: Vergleich realer und gemessener Abstand	17
Abbildung 14: Simulink Programm zur Streckenidentifikation.....	19
Abbildung 15: pzMove mit aufgenommenen Daten und Streckenübertragungsfunktion	19
Abbildung 16: Regelkreis	20
Abbildung 17: Regler Entwurf pzMove.....	21
Abbildung 18: Vergleich Realität und Simulation bei einem Einheitssprung	23
Abbildung 19: Simulation Störung durch Bremsen des Vordermanns	24
Abbildung 20: Vegleich Realität und Simulation anhand eines Sinus	25
Abbildung 21: Vegleich Realität und Simulation anhand eines Sägezahn	25