

Der Xylophon-Spielende Roboter

Viviane Wolters, Digital Engineering
Otto-von-Guericke-Universität Magdeburg

Zusammenfassung—Im Zuge des Projektseminars „Elektrotechnik/Informationstechnik (LEGO Mindstorms)“ der Otto-von-Guericke-Universität Magdeburg wird es den Studenten ermöglicht, innerhalb von zwei Wochen, mit Hilfe von LEGO-Mindstorms-NXT-Bausätzen, eigene Projekte zu realisieren. Diese Arbeit dient als Bericht über eines der Projekte aus diesem Seminar - dem xylophon-spielenden Roboter. Dabei wird auf den mechatronischen Aufbau sowie die Steuerung des Roboters eingegangen. Während des Projektes aufgetretene und nicht zu lösende Probleme, wie die Spielgenauigkeit und die geringe Spielgeschwindigkeit des Roboters, werden außerdem beleuchtet.

Schlagerwörter—Xylophon, Roboter, mechanische Musikinstrumente, selbstspielende Musikinstrumente, LEGO Mindstorms

I. EINLEITUNG

ZEITGLEICH zur Entwicklung der Musikinstrumente wuchs der Wunsch des Menschen, diese selbstspielend zu gestalten. Antrieb für diesen Wunsch war das ureigene Bedürfnis des Menschen nach Musik. Wer zu Zeiten, in denen es weder Radios, CDs oder Internet gab, den Wunsch hatte Musik zu hören, musste entweder selbst ein Instrument spielen oder einen Konzertsaal besuchen. Somit galt das Hören von Musik lange als Privileg der adligen und gut bürgerlichen Bevölkerung. [1]

Die ältesten noch erhaltenen selbstspielenden Instrumente (auch mechanische Musikinstrumente genannt), sind Glockenspiele in Monumentaluhren, welche erstmals im 14. Jahrhundert in Kirchen zum Einsatz kamen. Durch die steigenden technischen und musikalischen Möglichkeiten entwickelten sich die mechanischen Musikinstrumente immer weiter, sodass Anfang des 19. Jahrhunderts sogenannte Musikmaschinen sogar selbstspielende Orchester konstruierten. Mit der Entwicklung des Grammophons und dem Rundfunk gerieten die mechanischen Musikinstrumente jedoch langsam in Vergessenheit. [2]

Durch die heutzutage immer mehr fortschreitende Digitalisierung und die stetig steigende Anzahl neuer Erkenntnisse im Forschungsbereich der Künstlichen Intelligenz besteht die Chance das selbstspielende Musikinstrumente in Zukunft wieder mehr Aufmerksamkeit erhalten. So wurde beispielsweise bereits ein marimba-spielender Roboter entwickelt, dem es mit Hilfe von Big-Data-Analysen und Deep Learning möglich ist, eigene harmonische Kompositionen zu kreieren und vorzutragen. [3]

In dieser Arbeit wird, angelehnt an die Idee der selbstspielenden Musikinstrumente, die Entwicklung eines xylophon-spielenden Roboters beschrieben. Auch wenn dieser Roboter keine künstliche Intelligenz besitzt, so wurde sich dennoch das

Ziel gesetzt einen Roboter zu erschaffen, welcher selbständig mehrere einfache Melodien auf einem Xylophon spielen kann. In den folgenden Abschnitten werden zunächst die, für dieses Projekt notwendigen, musikalischen Aspekte erläutert. Im Hauptteil dieses Berichtes wird auf den mechatronischen Aufbau und den verwendeten Algorithmus zur Ansteuerung des Roboters eingegangen. Abschließend werden die aufgetretenen Probleme sowie zukünftig möglichen Verbesserungen und Erweiterungen des Roboters beleuchtet.

II. VORBETRACHTUNGEN

Ein selbstspielendes Musikinstrument besteht im wesentlichen aus drei verschiedenen Komponenten: Dem Instrument (hier: das Xylophon), dem Antrieb (hier: der Roboterarm) und dem Toninformationsträger (hier: der Programmcode). Aufgrund dessen, dass die Informationen des zu spielenden Musikstückes in der Steuerung des Roboters hinterlegt und entsprechend verarbeitet werden müssen, ist für die Umsetzung des Projektes ein grundlegendes musikalisches Verständnis notwendig. Die wichtigsten musikalischen Parameter werden im Folgenden näher betrachtet.

A. Musikalische Parameter einer Komposition

Ein Musikstück (Komposition) besteht aus vielen verschiedenen Komponenten. Die wichtigsten zwei dieser Komponenten sind die Melodik und die Rhythmik.

Die Melodik beschreibt die Abfolge der Tonhöhen einer Komposition auch Melodie genannt. [4] Die wichtigsten musikalischen Parameter eines Melodieverlaufes sind die Richtung und die Tonabstände (Intervalle). Die wichtigsten Intervalle sind in Abbildung 1 dargestellt.

Die Rhythmik beschreibt die Abfolge von Noten- bzw. Pausenwerten. Diese Parameter geben Auskunft über die Tondauer einzelner Noten bzw. die Pausenzeiten an. [4] Diese beiden Zeiten stehen immer im Verhältnis zu der Anzahl der im Musikstück enthaltenen Noten bzw. Pausen pro Takt und der allgemeinen Spielgeschwindigkeit der Komposition. Abbildung 2 zeigt die in diesem Projekt verwendeten Noten- und Pausenwerte und deren Relation zueinander.

B. Das Xylophon

Das in diesem Projekt verwendete Xylophon besitzt zwölf Klangplatten mit einem Tonumfang von a bis e'' (vgl. Abbildung 4). Klangplatten für chromatische Zwischentöne wie beispielsweise das Fis oder Cis besitzt das Xylophon nicht, sodass nur Musikstücke welche in C-Dur geschrieben sind, gespielt werden können. Bei der Melodieauswahl für dieses Projekt ist somit zu beachten, dass diese mit den vorhandenen Tönen spielbar sind. Abbildung 3 zeigt eine C-Dur Tonleiter und den in diesem Projekt verwendeten Tonumfang.

Abstand	Beispiel	Name	deutsch
vom 1. zum 1. Ton		Prime	der erste
vom 1. zum 2. Ton		Sekunde	der zweite
vom 1. zum 3. Ton		Terz	der dritte
vom 1. zum 4. Ton		Quarte	der vierte
vom 1. zum 5. Ton		Quinte	der fünfte
vom 1. zum 6. Ton		Sexte	der sechste
vom 1. zum 7. Ton		Septime	der siebente
vom 1. zum 8. Ton		Oktave	der achte

Abbildung 1. die wichtigsten Intervalle eines Musikstückes [5]

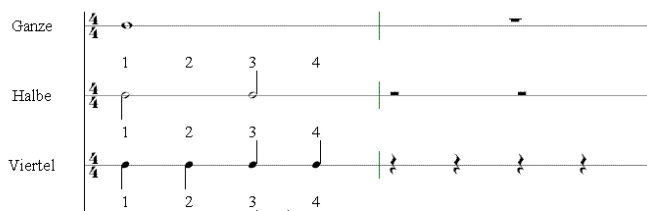


Abbildung 2. Für dieses Projekt relevante Noten- und Pausenwerte [6]

III. REALISIERUNG

Für den xylophon-spielenden Roboter werden Bauteile von mehreren LEGO-Mindstroms-NXT-Bausätzen, eine Grundplatte von LEGO Classic und ein kleines Xylophon mit Spielstab benötigt. Die Abbildung 4 zeigt den Aufbau des Xylophon-Roboters, welcher im Folgenden näher erläutert wird.

A. Mechanik

Für den Roboterarm werden zwei Motoren benötigt, welche für die vertikale und horizontale Bewegung des Spielstabs zuständig sind.

Der Motor für die Horizontalbewegung (hier: *Motor_H*) wird fest auf die Grundplatte montiert und bewegt mit Hilfe eines kleinen Zahnrades einen größeren Drehkranz auf dem der



Abbildung 3. C-Dur Tonleiter [7]

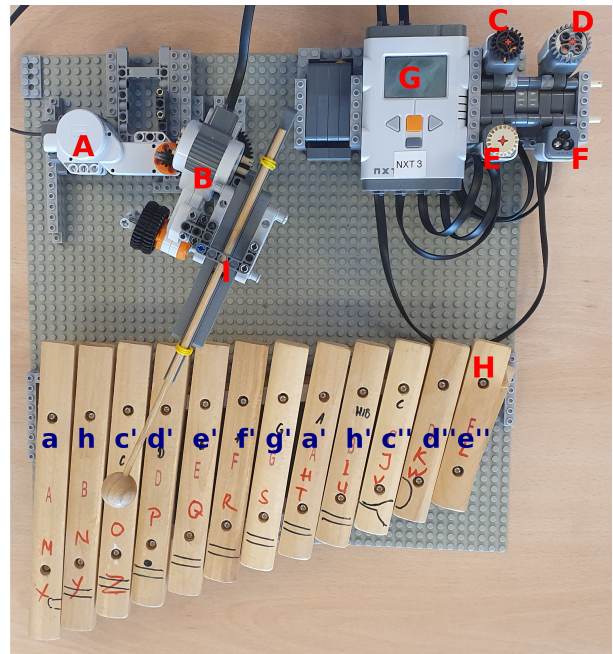


Abbildung 4. Aufbau Xylophon Roboter,

A: Motor_H (horizontale Bewegung), B: Motor_V (vertikale Bewegung), C: Tastensensor_{S1} (Spielstab nach rechts), D: Tastensensor_{S2} (Spielstab nach links), E: Tastensensor_{S3} (Spielstab nach oben/unten), F: Farbsensor (Melodieauswahl), G: NXT-Steuerungsgerät, H: Xylophon, I: Spielstabhalterung

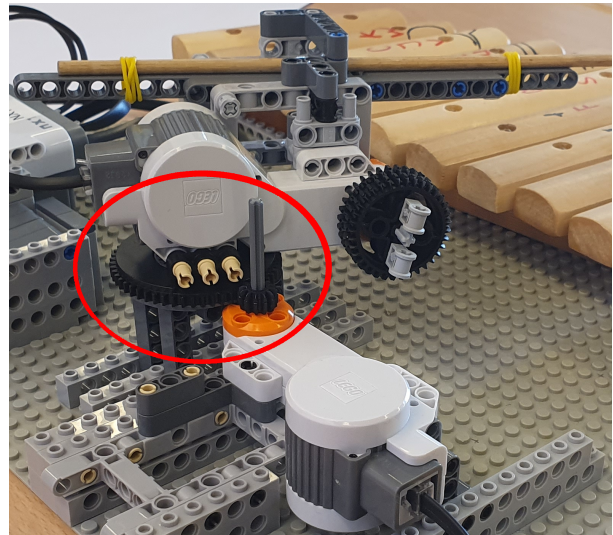


Abbildung 5. Drehkranz und Zahnrad für die horizontale Bewegung des Roboters

zweite Motor (hier: *Motor_V*) montiert ist (siehe Abbildung 5). Die Ansteuerung über die Zahnräder vergrößert das Übersetzungsverhältnis der Motorumdrehung zur Spielstabbewegung und verbessert die Genauigkeit der Ansteuerung der Klangplatten des Xylophons.

Der *Motor_V* bewegt eine Rotationsachse, an welcher ein kleiner Stift befestigt ist. Vertikal ausgerichtet, hält dieser Stift den, darüber befestigten, Spielstab in seiner höchsten Position. Wird der *Motor_V* angesteuert so rotiert dieser Stift und lässt den Spielstab nach unten schnellen. Pro Motorumdrehung könnten demnach zwei Töne auf dem Xylophon erzeugt werden.

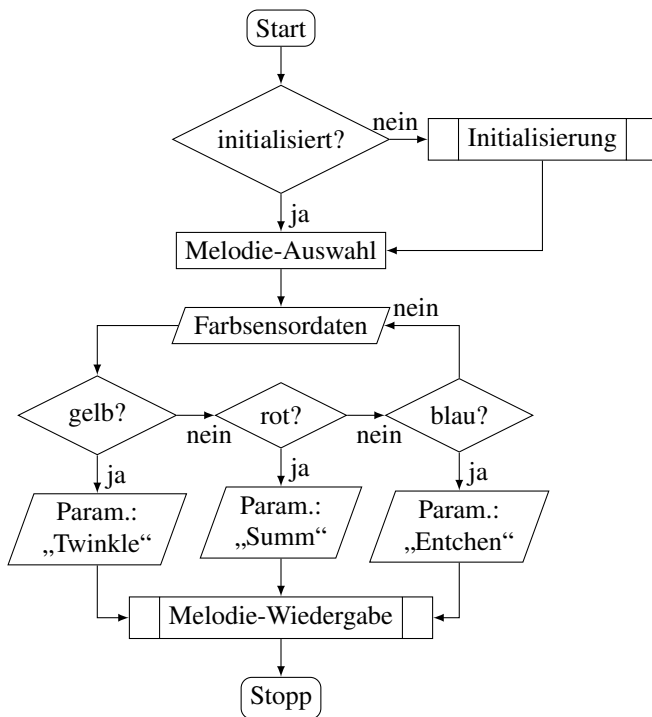


Abbildung 6. Programmablaufplan des Xylophon-Roboters

B. Elektronik

Drei Tastsensoren sowie ein Farbsensor bilden das Bedienfeld des Roboters. Die Tastsensoren dienen zur Handansteuerung des Roboterarms, um die Startposition des Spielstabes initialisieren zu können. Mit Hilfe des Farbsensors werden verschiedene Melodien ausgewählt.

C. Steuerung

1) *Programmablauf:* Das Roboterprogramm besteht aus drei grundlegenden Prozessen: der Initialisierung, der Melodie-Auswahl und der Melodie-Wiedergabe. Wie diese Prozesse zusammenarbeiten wird in der Abbildung 6 anhand eines vereinfachten Programmablaufplanes dargestellt.

Begonnen wird mit der Initialisierung der Startposition. Dabei wird der Spielstab mittels des *Tastsensor_{S1}* und *Tastsensor_{S2}* über der Klangplatte für die Note c' des Xylophons positioniert und mit Hilfe des *Tastsensor_{S3}* die höchste Position des Stabes angefahren. Durch gleichzeitiges Betätigen von *Tastsensor_{S1}* und *Tastsensor_{S2}* wird die Motorposition von *Motor_H* auf 0 gesetzt, die Initialisierung quittiert und abgeschlossen.

Nach der Initialisierung kann die abzuspielende Melodie gewählt werden, indem die entsprechende Farbe vor dem Farbsensor gehalten wird. Der Roboter startet nach dem Erkennen der Farbe automatisch mit dem Spielen der zugehörigen Melodie. Die Tabelle I zeigt die Zuordnung der Farben zu den im Zuge des Projektes programmierten Kompositionen.

2) *Melodie-Algorithmus:* Wie bereits in den Vorbetrachtungen erwähnt, sind die wichtigsten Parameter eines Musikstückes

Tabelle I
AUSWÄHLBARE MELODIEN UND DEREN FARBUORDNUNG

Farbe	Melodie
blau	Alle meine Entchen
rot	Summ summ summ
gelb	Twinkle twinkle little star

die verschiedenen Intervalle und die Richtung des Melodieverlaufes sowie die verschiedenen Noten- bzw. Pausenwerte. Zum Spielen des Xylophons müssen diese Parameter dementsprechend definiert werden. Anschließend können diese in einem Algorithmus verarbeitet und die Motoren des Roboters entsprechend angesteuert werden.

Die musikalischen Parameter werden in zwei separaten Arrays gespeichert. Das erste Array besteht aus den einzelnen Intervallen der Melodie, kombiniert mit einem entsprechenden Vorzeichen, um die Richtung des Melodieverlaufes beschreiben zu können. Das zweite Array besteht aus den Tonwerten des Musikstückes. Einfachheitshalber wurden existierende Pausenwerte auf die im Melodieverlauf vorherigen Notenwerte addiert, sodass zwischen diesen beiden Parametern nicht unterschieden werden muss.

Neben den beiden Arrays muss außerdem der Drehwinkel definiert werden welcher vom Motor gefahren werden muss, um das Sekunden-Intervall (siehe Abbildung 1) realisieren zu können. Dieses wurde im Zuge des Projektes experimentell ermittelt. Alle weiteren Intervalle können auf dessen Grundlage errechnet werden.

Der eigentliche Algorithmus (vgl. Algorithmus 1) besteht aus einer For-Schleife, die durch die beiden Parameter-Arrays iteriert:

In jedem Durchgang wird zunächst eine Fallunterscheidung durchgeführt. Ist das Intervall 0, bewegt sich der *Motor_H* nicht. Ist das Intervall kleiner als 0 (negatives Intervall) wird ein tieferer Ton als der vorherige angesteuert. Ist das Intervall größer als 0 wird dementsprechend ein höherer Ton angesteuert.

Die anzufahrenden Winkelpositionen des Motors ergeben sich hierbei aus dem Produkt des im Array hinterlegten Intervall und dem experimentell ermittelten Drehwinkel für das Sekunden-Intervall. Bei Intervallen größer als Vier muss noch eine kleine Korrektur an der anzufahrenden Winkelposition vorgenommen werden, da in diesen Fällen Abweichungen von der Soll-Position des Spielstabes zu beobachten waren. Die hier zu addierende Konstante muss, wie der Drehwinkel, experimentell ermittelt werden.

Nach der Fallunterscheidung und dem Ansteuern des *Motor_H* muss eine kurze Pause erfolgen. Würde dies nicht geschehen, käme es zu Störungen, da die Signale zu schnell an das Steuergerät gesendet würden. Die Länge dieser Pause muss ebenfalls experimentell bestimmt werden.

Nach der kleinen Pause wird der *Motor_V* angesteuert und die entsprechende Klangplatte angeschlagen. anschließend muss wieder eine Pause erfolgen, zum einen um erneuert zu garantieren, dass die Signale nicht zu schnell an das Steuergerät gesendet werden und zum anderen, um die verschiedenen Notenwerte zu realisieren. Auch hierbei muss experimentell

ermittelt werden, welche Pausenzeit mindestens einzuhalten ist. Dieser Wert definiert dann die Länge einer Viertelnote. Die anderen Notenwerte können danach entsprechend berechnet werden (siehe Abbildung 2).

Wenn die Parameter-Arrays vollständig von der For-Schleife durchlaufen sind, ist das Ende des Musikstückes erreicht und das Programm wird beendet.

Algorithm 1 Melodie-Wiedergabe (Pseudocode)

```

for  $i \leftarrow$  Laenge der Arrays do
  if  $\text{Intervall}(i) == 0$  then
    nichts passiert
  else if  $\text{abs}(\text{Intervall}(i)) >= 4$  then
     $M_H \text{Power} \leftarrow 100$ 
     $M_H \text{TachoLim.} \leftarrow \text{abs}(\text{Intervall}(i)) * \text{Drehw.} + c$ 
    SendToNXT
  else if  $\text{Intervall} < 0$  then
     $M_H \text{Power} \leftarrow 100 * -1$ 
     $M_H \text{TachoLim.} \leftarrow \text{abs}(\text{Intervall}(i)) * \text{Drehw.}$ 
    SendToNXT
  else
     $M_H \text{Power} \leftarrow 100$ 
     $M_H \text{TachoLim.} \leftarrow \text{abs}(\text{Intervall}(i)) * \text{Drehw.}$ 
    SendToNXT
  pause(minval)
   $M_V \text{Power} \leftarrow 100$ 
   $M_V \text{TachoLim.} \leftarrow 180$ 
  SendToNXT
  pause(Notenwert(i))

```

IV. ERGEBNISDISKUSSION

Es ist gelungen einen Roboter zu konstruieren, welcher einfache Melodien auf einem Xylophon wiedergeben kann. Dabei sind verschiedene Probleme aufgetreten, von denen nicht alle zufriedenstellend gelöst wurden konnten.

A. Genauigkeit der Ansteuerung

Immer mal wieder treten Abweichungen bezüglich der Soll-Position des Spielstabes auf, mit der Folge, dass falsche Töne angespielt werden.

Bei einem früheren Prototypen war es noch nicht einmal möglich, die einfache C-Dur Tonleiter fehlerfrei zu spielen. Eine Vergrößerung des Übersetzungsverhältnis von $Motor_H$ zur Spielstabbewegung durch Einsatz eines Drehkranzes, anstelle der zuvor erprobten direkten Montage des $Motor_V$ auf den $Motor_H$ (vgl. Abschnitt III.A Mechanik), konnte der Ungenauigkeit entgegen wirken.

Trotz alledem kommt es von Zeit zu Zeit zu ungewollten Abweichungen. Diese Abweichungen können unter anderem durch eine ungenaue Initialisierung (menschlicher Fehler) oder durch ein Verändern der Position des Xylophons hervorgerufen werden. Abhilfe würde hier eine automatisierte Initialisierung und eine verbesserte Fixierung des Xylophons verschaffen.

B. Geschwindigkeit des Roboters

Der Xylophon-Roboter kann Melodien nicht beliebig schnell abspielen. Gelangen die Signale zur Ansteuerung der Motoren zu schnell zum Steuergerät, führt dies zu einer Störung.

Das Spielen von Notenwerten kleiner als eine Viertelnote ist somit nicht möglich ohne eine Reduzierung der gesamt Geschwindigkeit, womit sich die Melodie verzerrt anhören würde. Um dieses Problem zu lösen, könnte ein weiterer Roboterarm installiert werden, sodass der Arbeitsraum zwischen den Robotern aufgeteilt werden kann. Dies könnte sich ebenfalls positiv auf die Spielgenauigkeit auswirken. Um dies umsetzen zu können, müsste jedoch ein zweites Steuergerät zum Einsatz kommen, da der eingesetzte NXT-Baustein nur drei Motoren ansteuern kann.

C. Komplexität der Melodien

Aufgrund des eingeschränkten Arbeitsbereiches des Roboterarmes, den Limits bezüglich der Spielgeschwindigkeit und des eingeschränkten Tonumfangs des Xylophons, können nur sehr einfache Melodien vom Roboter gespielt werden. Ein Zusammenschalten von mehreren Robotern mit unterschiedlichen Xylophonen wäre hierfür die Lösung.

V. ZUSAMMENFASSUNG UND FAZIT

Im Zuge dieser Arbeit wurde die Realisierung eines xylophon-spielenden Roboters von der Konstruktion bis hin zur Programmierung betrachtet und aufgetretene Probleme diskutiert. Mit dem entstandenen Roboter ist es möglich, einfache Melodien wie z.B. „Alle meine Entchen“ abzuspielen. Eine Garantie, dass dies fehlerfrei geschieht, gibt es allerdings nicht. Eine Verbesserung der Initialisierung könnte hierbei Abhilfe verschaffen.

An die technisch möglichen Grenzen gelangt der Roboter bezüglich komplexerer und vor allem schnellerer Musikstücke. Eine mögliche, auf den hier vorgestellten Roboter aufbauendes, Projekt könnte daraus bestehen, dieser Problematik mit Hilfe eines weiteren Roboterarmes entgegen zu wirken.

Trotz aller zukünftig möglichen Verbesserungen bleibt ein Roboter jedoch ein Roboter, welcher nur eine bestimmte zeitliche Abfolge von Tönen abspielen und nicht wie ein Mensch die eigenen Gefühle und Emotionen in die Interpretation des Musikstücks einfließen lassen kann.

LITERATURVERZEICHNIS

- [1] SURBER, HANSJÜRG: *Mechanische Musikinstrumente, Bedeutung, Funktionsweise und soziale Stellung*. <http://www.musikautomaten-ungarn.eu/de/geschichtemechanischemusik.html>. Version: 2017
- [2] TENTEN, WALTER AND GSMEV: *Mechanische Musikinstrumente*. <https://www.musica-mechanica.de/pages/de/geschichte.php>. Version: November 2015
- [3] GERSTEL, SEBASTIAN: *Xylophon-Roboter komponiert und spielt seine eigene Musik*. <https://www.elektronikpraxis.vogel.de/xylophon-roboter-komponiert-und-spielt-seine-eigene-musik-a-616930/>. Version: Juni 2017
- [4] KÖPPL, BIRGIT: *Musik-Analyse: Wichtige musikalische Parameter*. https://www.gym-raubling.de/medien/Unterricht/Musik/musik_analyse_klasse_10.pdf
- [5] KAISER, ULLRICH: *Intervalle bestimmen*. <https://musikanalyse.net/tutorials/intervalle-bestimmen/>. Version: 2018
- [6] KAISER-KAPLANER, JOHANNES: *Notenwerte (Klangdauer)*. https://www.musiklehre.at/1_002.htm. Version: 2019
- [7] KAISER-KAPLANER, JOHANNES: *Diatonik - Durtonleiter*. https://www.musiklehre.at/9_004.htm. Version: 2019