

HUMBOLDT-UNIVERSITÄT ZU BERLIN
MATHEMATISCH-NATURWISSENSCHAFTLICHE FAKULTÄT
INSTITUT FÜR INFORMATIK

Zweibeiniges Laufen für Humanoide Roboter im RoboCup

Studienprojekt

eingereicht von: Steffen Kaden

Gutachterin: Prof. Verena V. Hafner

Inhaltsverzeichnis

1	Einleitung	5
1.1	Beitrag der Arbeit	5
1.2	Struktur der Arbeit	5
1.3	Kontext der Arbeit: RoboCup	6
1.4	Grundlegende Modelle	6
1.5	Verwandte Arbeiten	8
2	Laufalgorithmus des RoboCup-Teams NaoTH	13
2.1	Überblick	13
2.2	Planungsphase	13
2.3	Ausführungsphase	15
2.4	Stabilisierung	15
3	Empirische Analyse der Laufbewegung	17
3.1	Methoden und Werkzeuge	17
3.2	Experimente	20
4	Zusammenfassung und Ausblick	31

1 Einleitung

Eine der größeren Herausforderungen der humanoiden Robotik ist das zweibeinige Laufen. Es ermöglicht im Vergleich zu beräderten Robotern vielfältige Formen der Fortbewegung innerhalb einer breiten Palette von Terrains. Im Vergleich zu der Fortbewegung mit mehr Beinen zeichnet sich das zweibeinige Laufen prinzipiell durch eine hohe Wendigkeit aus, ist jedoch auch von geringerer Stabilität. Insbesondere die Stabilität stellt eine große Herausforderung dar. Meist ermöglichen zusätzlichen Stabilisierungsroutinen erst eine erfolgreiche zweibeinige Fortbewegung, welche jedoch noch weit von der Stabilität des menschlichen Laufens entfernt ist.

Ein Beispiel für erfolgreiche, zweibeinige Fortbewegung, neben anderen wie die Präsentationen von ASIMO und den Videos von PETMAN und Atlas, ist in den Fußballligen des RoboCups zu sehen. Der RoboCup bietet mit seinen humanoiden Fußballligen ein einfaches und doch herausforderndes Terrain zur kompetitiven Evaluierung von Laufalgorithmen zweibeiniger Roboter.

1.1 Beitrag der Arbeit

Im Rahmen dieser Arbeit wird der aktuelle Laufalgorithmus des RoboCup Teams NAO Team Humboldt restrukturiert, mit dem Ziel die Schnittstellen der einzelnen Teilkomponenten des Algorithmus klar zu definieren und die Teilkomponenten stärker zu separieren. Durch die stärkere Separierung der Teilkomponenten wird die empirische Untersuchung des Einflusses einzelner Parameter, Optionen oder Teilkomponenten erleichtert. Zudem wird sich von der Restrukturierung erhofft, dass der Laufalgorithmus leichter wart- und erweiterbar wird.

Außerdem werden Werkzeuge entwickelt, die die empirische Untersuchung der Laufbewegung unterstützen. Sie sollen somit einen ersten Schritt zur Objektivierung der Wertung der Effektivität einzelner Parametrisierungen, Optionen und Teilkomponenten bilden.

Anschließend werden die entwickelten Werkzeuge genutzt, um den Einfluss von Parametern auf das Laufen, des Teams NaoTH empirisch zu untersuchen. So wird der Einfluss der ZMP-Trajektorie, als auch der Einfluss eines Gelenkpositionsskalierungsfaktors untersucht.

1.2 Struktur der Arbeit

Der Rest dieses Kapitels gibt einen kurzen Einblick in den RoboCup, der als Kontext der Arbeit zusehen ist, sowie einen kurzen Überblick zu verwandten Arbeiten in der

RoboCup Standard Platform League. Im darauf folgenden Kapitel wird der Laufalgorithmus des NAO Team Humboldt detailliert erläutert. Das 3. Kapitel widmet sich der empirischen Analyse der Laufbewegung des Laufalgorithmus des NAO Team Humboldt anhand zweier, konkreter Fragestellungen, die sich im Rahmen des RoboCups stellen. Zum Schluss folgt eine Zusammenfassung und ein kurzer Ausblick.

1.3 Kontext der Arbeit: RoboCup

Der RoboCup ist eine internationale Plattform mit Wettbewerbscharakter zur wissenschaftlichen Förderung im Bereich der Robotik, autonomer Systeme und Künstlicher Intelligenz. In den seit 1997 jährlich stattfindenden RoboCup-Wettkämpfen können neue Algorithmen und Ansätze in öffentlichkeitswirksamen Disziplinen wie Fußball (RoboCup Soccer), Katastrophenschutz (Rescue League seit 2001), industriellen Produktionszenarien (RoboCup Logistics League seit 2012) und Servicerobotik im häuslichen (RoboCup@Home seit 2006) oder arbeitstechnischen Kontext (RoboCup@Work seit 2016) kompetitiv verglichen werden.

Ziel ist es in der Mitte des 21. Jahrhunderts mit einem Team von vollautonomen, humanoiden Robotern den amtierenden FIFA-Weltmeister regelkonform in einem Fußballspiel zu besiegen. Um dieses Ziel zu erreichen, gibt es mehrere Ligen im Bereich RoboCup Soccer, die sich auf unterschiedliche Punkte fokussieren. Eine dieser Ligen ist die Standard Platform League (SPL), welchen einen starken Fokus auf die Algorithmen legt, indem die zu verwendende Roboterplattform in den Wettkämpfen festgelegt ist. 2008 löste der zweibeinige Roboter NAO (von Aldebaran Robotics) den vierbeinigen Roboter AIBO (von Sony) in der Standard Platform League als Roboterplattform ab. Seitdem gehört das zweibeinige Laufen zu den Schlüsselkompetenzen in den Wettbewerben dieser Liga, wie schon zuvor in der Humanoid League, welche im Unterschied zur SPL keine Standardplattform vorschreibt.

Wie die meisten Sportarten setzt das erfolgreiche Spielen von Fußball eine gewisse Kontrolle und Wissen über die eigenen motorischen Fähigkeiten voraus. Das Laufen nimmt im Fußball als laforientierte Sportart damit eine zentrale und grundlegende Position unter den motorischen Fähigkeiten ein. Alle weiteren spielspezifischen Fähigkeiten können erst durch ein schnelles und stabiles Laufen zur Geltung kommen.

Bis heute hat sich das Lauftempo und damit das Spieltempo in der SPL signifikant beschleunigt. Insbesondere für neue Teams mit Eigenentwicklungen existiert eine hohe Einstiegshürde und kleinere Teams haben Probleme in allen Bereichen konkurrenzfähig zu bleiben. So bauen mittlerweile neue Teams auf von anderen Teams bereitgestellten Lösungen auf.

1.4 Grundlegende Modelle

An dieser Stelle sollen die zwei meist genutzten grundlegenden Modelle kurz erläutert werden.

3D Linear Inverted Pendulum Mode

Das 3D Linear Inverted Pendulum Mode (3D-LIPM) [15] modelliert die Dynamik eines invertierten Pendels im dreidimensionalen Raum, wobei angenommen wird, dass der Massenschwerpunkt (center of mass, CoM) stets auf einer konstanten Höhe bleibt. Des Weiteren wird angenommen, dass das Modell seiner Eigendynamik folgt. Durch diese Annahmen werden die nicht linearen Differenzialgleichungen, die die Dynamik des Inversen Pendels beschreiben, linearisiert, lösbar und bezüglich der x und y-Richtung entkoppelt. Die Position und die Geschwindigkeit kann dann mit folgenden Gleichungen bestimmt werden:

$$x(t) = x_0 \cdot \cosh(k \cdot t) + \dot{x}_0 \cdot \frac{1}{k} \cdot \sinh(k \cdot t) \quad (1.1)$$

$$\dot{x}(t) = x_0 \cdot k \cdot \sinh(k \cdot t) + \dot{x}_0 \cdot \cosh(k \cdot t) \quad (1.2)$$

mit $k = \sqrt{\frac{g}{h}}$, wobei g der Gravitationskonstante entspricht, h der Höhe und $x_0, \dot{x}_0 \in \mathbb{R}^2$ der Position und Geschwindigkeit des Massenschwerpunktes zum Zeitpunkt 0. Der Ankerpunkt des Pendels liegt dabei im Fuß des Roboters.

Das Timing des Supportwechsels und die Lage des Fußpunktes beeinflussen die Endgeschwindigkeit und ermöglichen so das Steuern des Massenschwerpunktes. Dieser Ansatz zeichnet sich durch eine hohe Reaktivität aus.

Cart-Table-Model

Das Cart-Table-Model [14] basiert auf den eben vorgestellten 3D-LIPM. Während für einen gegebenen Startzustand im 3D-LIPM die Trajektorie einfach verfolgt wird und der Zero Moment Point (ZMP) stets implizit im Fußpunkt des Pendels liegt, wird hier eine gewünschte ZMP-Trajektorie definiert und daraus die Trajektorie des Massenschwerpunktes bestimmt.

Die ZMP-Gleichung ergibt sich aus der Dynamik des 3D-LIPM und lautet:

$$p_y = y - \frac{z_c}{g} \cdot \ddot{y} \quad (1.3)$$

$$p_x = x - \frac{z_c}{g} \cdot \ddot{x} \quad (1.4)$$

mit g als Gravitationskonstante, z_c als konstante Höhe und p_i als ZMP-Koordinate als Summe aus Ort und Beschleunigung in der jeweiligen Richtung. Die Dynamik des CoM wird als einfaches, diskretes dynamisches System modelliert mit dem Ruck (Änderung der Beschleunigung) als Stellgröße u . Somit kann unter Verwendung der Optimal-Control-Theory ein Feedback-Controller entworfen werden, der folgenden Performance Index minimiert und damit eine CoM Trajektorie erzeugt, die die gewünschten ZMP-Trajektorie (weitestgehend) erzeugt.

$$J = \sum_{i=k}^{\infty} \{Q_e e(i)^2 + \Delta x^T(i) Q_x \Delta x(i) + R \Delta u(i)^2\} \quad (1.5)$$

1 Einleitung

Mit $e(i)$ als dem Fehler zwischen Ist- und Soll-ZMP zum diskreten Zeitpunkt i , $\Delta x(i)$ als die inkrementelle Zustandsänderung und $\Delta u(i)$ als inkrementelle Änderung im Steuersignal (dem Ruck).

Der Vorteil ist, dass man die Positionen der Fußpunkte theoretisch beliebig festlegen kann, der ZMP muss nur innerhalb des Fußes liegen. Jedoch wird in diesem Ansatz ein Preview-Zeitfenster benötigt, was die Reaktivität reduziert.

1.5 Verwandte Arbeiten

Im Folgenden wird auf existierende Lösungen in der Standard Platform League eingegangen.

UNSW Australia

Abwandlungen des Laufalgorithmus von UNSW Australia [9] werden von MiPal[7], Luxembourg United [5], MRL-SPL [2], RoboEireann [26], WrightOcean [27] und B-Human[18], als auch von allen Teams, die auf dem aktuellen Framework von B-Human aufbauen genutzt.

Der Lauf basiert auf einem Inversen Pendel Modell, welches die Dynamiken in der Sagittal- und Frontalebene als unabhängig betrachtet. Bei jedem Fußwechsel wird ein Energieverlust durch das Auftreten angenommen und durch die Reduzierung der Geschwindigkeit modelliert. Wenn der Fuß den Roboter nach oben drückt, wird implizit dem System wieder Energie hinzugefügt. Äußere Störeinflüsse werden durch kleine einer Gaußverteilung mit Erwartungswert 0 folgenden Änderungen der Geschwindigkeit beim Fußwechsel modelliert.

Die ganze Zeit über wird das System mithilfe eines geschlossenen Regelkreises stabilisiert, der in der Frontalebene auf Basis der Fußsensoren und in der Sagittalebene auf Basis des Gyrometers wirkt. Der Fußwechsel wird durch den Vorzeichenwechsel der Projektion des Centre of Pressure (kurz CoP) auf die Querachse registriert. An diesem Übergang werden die Schwungphasen synchronisiert, was eine Stabilisierung in der Frontalebene zur Folge hat. Um den Torso bezüglich der Sagittalebene in vertikaler Position zu halten wurde ein Reinforcement Learning Ansatz in einer Simulation entwickelt. Dessen Policy kann durch einen Feedforward-P-Controller approximiert werden, welcher als Regelgröße das Gyrometer verwendet und den Winkel des Fußgelenks so modifizieren, dass jegliches Schwingen des Torsos verhindert wird.

Änderungen in den Laufparametern (z. B. Richtung) werden unmittelbar zu Beginn der nächsten Laufphase angewendet. Während des Schrittes wird versucht die Füße stets parallel zum Boden zu halten.

B-Human

Das Framework von B-Human ist eine beliebte Ausgangsbasis von neuen oder kleineren Teams. Mittlerweile basieren die Frameworks von Camellia Dragons[10], JoiTech-SPL[6],

SPQR [21], UChile [22], TJArk [16] und UT Austin Villa [25] auf dem Framework von B-Human. Außerdem plant das Team Bembelbots seinen eigenen Laufalgorithmus durch den von B-Human zu ersetzen [4].

Seit 2017 basiert der Lauf von B-Human auf dem Lauf von UNSW Australia.[18] Der Lauf wurde ins B-Human-Framework integriert und angepasst. So wurde die Inverse Kinematik durch die B-Human-Eigene ersetzt, der Zustandsübergang aufgrund der Gewichtsverlagerung für seitliche Schritte erweitert, unterschiedliche P-Gains für den Gyro-Kontroller für Vorwärts- und Rückwärtslaufen implementiert, die Positionen der Arme berücksichtigt und die Beschleunigung durch ein lineares Modell ersetzt.

HULKs

Die Schrittplanung wird als geometrisches Problem mit eingeschränktem Lösungsraum formuliert [1]. Dabei werden die translatorische und rotatorische Komponente eines Schrittes getrennt. Der rotatorische Anteil darf dabei nicht über einem Maximalwert steigen und nur beschränkt wachsen, relative zur Rotation des vorherigen Schrittes. Die Translation wird so eingeschränkt, dass sie in der Schnittmenge zwischen dem Kreis mit der aktuellen Roboterposition und maximaler Schrittweite als Radius und dem Kreis an der Position, die erreicht wird, wenn der letzte Schritt wiederholt würde, mit der maximalen Schrittweite als Radius. Dabei wird jedoch der Radius des ersten Kreises reduziert in Abhängigkeit der Zielfußrotation und eines Bremskriteriums.

Der Lauf basiert auf dem Modell des Linear Inverted Pendulum Mode Modells[17]. In y-Richtung wird die Bewegung des Roboters durch ein Standard-LIPM approximiert. In der x-Richtung wird die Trajektorie des LIPM durch ein Spline approximiertes. Der Spline wird genutzt, um die Grenzbedingungen der Schrittphasen zu beschreiben. Der Spline wird durch die Messungen des CoM aktualisiert und zur Ermittlung des ZMPs verwendet. So wird es ermöglicht den Lauf gegebenenfalls anzupassen.

Des Weiteren wird die Schritthöhe in Abhängigkeit der Oberkörperneigung angepasst. Wenn eine Instabilität mithilfe des Gyrometers detektiert wurde, so wird der Bereich für einen gültigen ZMP virtuell eingeschränkt. Es wird ein P-Kontroller genutzt, der am Fußgelenk wirkt und Gyrofeedback nutzt, um den Oberkörper zu stabilisieren und so ungewünschte Oszillationen zu verhindern.[1]

Nao Devils Dortmund

Der Laufalgorithmus des Teams Nao Devils Dortmund[12] wird von ihnen erfolgreich im Wettbewerb eingesetzt und vom Dutch Nao Team gerade in ihr Framework integriert[24].

Nachdem die Zielgeschwindigkeit übergeben wurde, werden die benötigte Schrittweite berechnet. Die resultierenden Fußpositionen werden genutzt, um ein ZMP Trajektorie zu bestimmen. Die ZMP Trajektorie wird dabei in x-Richtung mit der gewollten Geschwindigkeit linear fortgesetzt und in y-Richtung durch eine eindimensionale Bezierkurve mit vier Kontrollpunkten beschrieben. Analog zum Cart-Table-Modell wird mithilfe des Flexible Inverted Pendulum (FLIP) eine Massenschwerpunkttrajektorie bestimmt, die die geforderte ZMP-Trajektorie realisiert. Das FLIP Modell erweitert

das Cart-Table-Modell um einen weiteren Massenschwerpunkt, der über eine Feder und Dämpfer mit dem Hauptmassepunkt verbunden ist. Dieser kleinere Massenschwerpunkt wird beschleunigt und die Beschleunigung über die Feder und den Dämpfer auf die Hauptmasse übertragen. So werden verschiedene flexible Elemente in der Kinematik des Roboters modelliert. Um den Lauf weiter zu stabilisieren wird der tatsächliche Zustand des CoMs mithilfe eines Beobachters geschätzt und bei der Berechnung des nächsten Zielzustandes berücksichtigt. Die Orientierung des Torsos um die y-Achse wird mit einem PD-Controller auf Basis des Gyrometers stabilisiert[11].

NomadZ

Der Laufalgorithmus des Teams NomadZ basiert auf dem des Teams B-Human, wurde jedoch durch Silvan Plüss in seiner Studienarbeit[20] 2014 erweitert und ist stets noch die Basis des aktuellen Laufens[8]. Anstatt mithilfe eines iterativen Algorithmus eine Menge von nicht linearen Bewegungsgleichungen zu lösen und somit Motorpositionen zu bestimmen, wird ein Linear Quadratic Regulator (LQR) mit einem hybriden, zeitdiskreten Zustandsraum genutzt. Das zeitdiskrete System ist für eine feste Schrittdauer konstant. Die Schrittweite kann durch die definierte Schrittdauer und der gewollten Geschwindigkeit ermittelt werden. Durch die physikalischen Beschränkungen der Kinematik des Roboters ist für eine feste Schrittdauer die maximale Geschwindigkeit beschränkt. Es werden verschiedene Schrittdauern mit charakteristischen Geschwindigkeiten definiert, die sich im Geschwindigkeitsraum überschneiden, d. h. eine Geschwindigkeit v kann sowohl mit der Schrittdauer T_1 als auch mit der Schrittdauer T_2 erreicht werden. Diese Schnittmenge ermöglicht es zwischen den verschiedenen Schrittdauern zu wechseln. Für jede Schrittdauer gibt es einen LQR-Controller, der die charakteristische Geschwindigkeit versucht zu halten.

NTU RoboPAL

Der Laufalgorithmus basiert auf dem 3D LIPM von Kajita et. al. [15] Es wird ein Kalmanfilter genutzt, um den aktuellen Zustand zu schätzen. Dabei wird die zeitliche Verzögerung der Messungen berücksichtigt. Als Fußtrajektorie des Schwingfußes wird eine sinusförmige Funktion verwendet. [19]

Nao-Team HTWK

Das Laufen des Nao-Teams HTWK[13] basiert nicht auf dem Inverted-Pendel-Model ähnelt dem Lauf vorgestellt in [3] erweitert um einen selbst entwickelten Balancing Algorithmus. Deshalb wird im Folgenden das in [3] präsentierte Laufen vorgestellt. Der Lauf basiert auf eine Central Pattern Generator. Anstatt alle Gelenke separat zu steuern, wird ein abstrakter Aktuatorraum genutzt der im Prinzip aus dem Kugelkoordinaten des Fußes bezüglich der Hüfte und den Pitch- und Rollwinkel des Fußgelenks besteht. Eine zentrale Uhr durchläuft Werte im Intervall $[-\pi, \pi]$. Ihre Frequenz definiert die Schrittfrequenz. Die Phasen für das linke und recht Bein sind bezüglich der zentralen Uhr

verschoben. In Abhängigkeit von der angeforderten Schrittrichtung, -geschwindigkeit und -rotation werden zusammen mit der Beinlänge und der Schrittfrequenz Amplituden weitere Schwingungen bestimmt. Diese Schwingungen beschreiben die Verschiebung des Massenschwerpunktes, das Anziehen des Schwungbeins, den Wechsel des Standbeins, die Schwenkbewegung des Schwungbeins und Ausgleichsbewegungen des Oberkörpers und der Arme um den Lauf weiter zu stabilisieren.

2 Laufalgorithmus des RoboCup-Teams NaoTH

Die Implementierung des Laufalgorithmus des RoboCup Teams NaoTH basiert in weiten Teilen auf der Arbeit von Yuan Xu [28] und orientiert sich im Kern am Cart-Table-Model nach Kajita et al. [14]. Die Codebasis ist leider recht unübersichtlich und trotz eines Aufräumversuchs in 2015 nicht leicht zugänglich für damit unerfahrene Studenten. Insbesondere sind die Teilkomponenten und ihre Schnittstellen nicht ersichtlich. Desweiteren gibt es eine große Anzahl von Parametern, bei denen nicht genau klar ist, wo sie überall genutzt werden bzw. wie sie wechselwirken. Diese Probleme erschweren eine empirische Untersuchung einzelner Parameter, Optionen und Teilkomponenten des Laufens. Um diese Probleme zu beheben wurde die Implementierung des Laufalgorithmus überarbeitet und modularisiert. Es ordnete sich nun in die Modulararchitektur ein mit dem Walk2018 als Motion-Modul und seinen Komponenten als Submodule. Dabei werden Repräsentationen auf dem Blackboard als Kommunikationsschnittstelle zwischen den einzelnen Submodulen genutzt. Im Folgenden wird die aufgeräumte Struktur des Laufalgorithmus des Teams NaoTH erläutert.

2.1 Überblick

In Abbildung 2.1 ist die aktuelle Struktur des Laufalgorithmus innerhalb des Frameworks zu sehen. Der Lauf unterteilt sich dabei in grob in eine Planungsphase und eine Ausführungsphase. Eben so lässt sich jeder Schritt in diese Phasen unterteilen. Die Dauer der Phasen entsprechen der Schrittdauer, welche abhängig von Parametern konstant ist oder zwischen 3 Werten variiert. Ein Schritt verfügt über Zählervariablen, die den Submodulen signalisieren, wo und in welcher Phase sich der Schritt befindet.

2.2 Planungsphase

In der Planungsphase übernimmt der FootStepPlanner von Cognition den im MotionRequest enthaltenen WalkRequest. Der WalkRequest enthält unter anderem Informationen über die gewünschte Zielposition (x, y) und Rotation θ . Jedes Mal, wenn ein Schritt fertig geplant ist, fügt der FootStepPlanner anhand des geforderten WalkRequest eine neu Zielposition für den nächsten freien Fuß zum StepBuffer hinzu. Dabei wird die Zielposition beschränkt um die Ausführbarkeit und Stabilität des Laufs zu gewährleisten. Jedes Mal, wenn ein Schritt ausgeführt wurde, dann entfernt der FootStepPlanner diesen Schritt aus dem StepBuffer.

2 Laufalgorithmus des RoboCup-Teams NaoTH

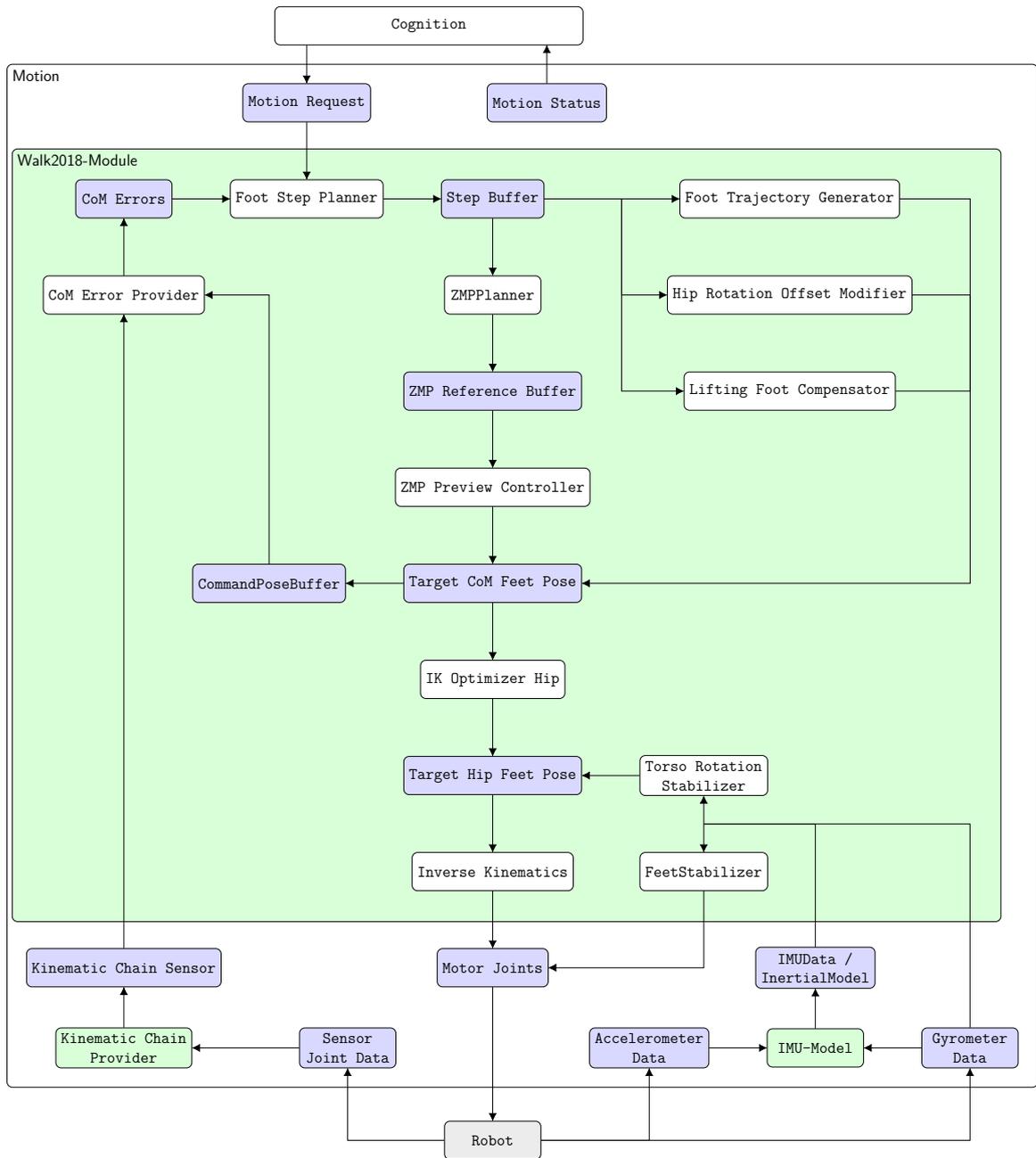


Abbildung 2.1: Der Laufalgorithmus des Teams NaoTH in der Übersicht. Weiße Blöcke entsprechen Submodulen und blaue entsprechen Repräsentationen auf dem Blackboard im Framework.

Die Zielpositionen, die im StepBuffer hinterlegt sind, nutzt der ZMPPlanner als Stützpunkte für die Beschreibung des ZMP-Verlaufs im Raum. Es werden dabei zwei grundlegende Trajektorien unterstützt. Die erste Trajektorie entspricht dabei, bis auf Offsets, der angeforderten Fußposition. Die zweite Trajektorie nutzt die geforderten Fußpositionen um in x-Richtung eine Gerade und in y-Richtung eine Bezierkurve zu generieren. Diese Idee stammt aus [12]. Es wird für jeden Zeitschritt der Planungsphase ein Punkt dem ZMPReferenceBuffer hinzugefügt.

2.3 Ausführungsphase

In jedem Zyklus bestimmt der ZMPPreviewController aus dem ZMPReferenceBuffer die aktuelle Sollposition, -geschwindigkeit und -beschleunigung. Parallel dazu werden anhand des auszuführenden Schrittes die Zielposition des Schwingfußes, die Zielneigung des Oberkörpers und ein Offset für die Hüftposition bestimmt. Zusammen bilden sie die TargetCoMFeetPose. Da der Massenschwerpunkt nur indirekt kontrolliert werden kann, wird eine TargetHipFeetPose durch eine Optimierung in der inversen Kinematik bestimmt, die die geforderte Relation zwischen Fußposition und Massenschwerpunkt durch eine Relation zwischen Fuß und Hüftposition approximiert. Im Anschluss wird im TorsoRotationStabilizer ein PPD-Controller genutzt, um den Oberkörper zu stabilisieren. In der Stellgröße geht die aktuelle Orientierung, die Abweichung von der Sollgeschwindigkeit und die Ableitung dieses Fehlers ein. Anschließend werden mithilfe der inversen Kinematik die benötigten Gelenkstellungen berechnet. Zum Schluss, bevor die Kommandos an den Roboter geschickt werden, modifiziert ein P-Controller auf Basis, der Rotationsgeschwindigkeit und Orientierung des Torsos die Roll- und Pitchgelenke der Füße.

2.4 Stabilisierung

In Abbildung 2.1 ist zu sehen, dass an drei Stellen Sensorfeedback genutzt wird, um den Lauf zu stabilisieren. Das erste Feedback ist der Fehler zwischen geforderten und gemessenen Massenschwerpunkt, welcher vom FootStepPlanner genutzt. Je nach Richtung des Fehlers wird die Fußposition angepasst, d. h. ist der Massenschwerpunkt zu weit rechts, so wird die Fußposition von der geforderten Zielposition aus weiter nach rechts verschoben. Somit soll der Fehler im folgenden Schritt kompensiert werden. Die beiden anderen Stellen sind der FeetStabilizer und der TorsoRotationStabilizer. Beide zielen darauf ab den Torso aufzurichten und Schwingungen des Oberkörpers zu reduzieren. Dabei beeinflussen sie indirekt den CoM und damit auch den ZMP. Wie sich diese Stabilisatoren konkret auf das Einhalten der Referenztrajektorien des CoMs und des ZMPs auswirken, ist noch zu untersuchen.

Sollte der CoMError zu groß werden, so fügt der FootStepPlanner Nullschritte ein, um ein Aufschaukeln zu verhindern. Nullschritte sind Schritte, in denen sich die Füße nicht bewegen und der ZMP zwischen den Füßen unterhalb des CoMs liegen soll. Dies

passiert so lange, bis der CoMError wieder in einem validen Bereich liegt.

Neben diesem Feedback wurden einige Feedforward-artige Mechanismen implementiert. Dabei ist zu beachten, dass viele Komponenten während der Wettbewerbe entstanden und noch nicht verifiziert wurde, ob diese Komponenten wirklich den gewünschten Effekt haben. In HipRotationOffsetModifier gibt es die Möglichkeit in Abhängigkeit der Änderung der Schrittweite den Oberkörper nach vorne zu lehnen. Dies sollte das Drehmoment durch die zusätzliche Beschleunigung des Fußes kompensieren. Der ZMP-Planner ermöglicht den CoM bezüglich des Fußes in Abhängigkeit der Schrittweite zu verschieben. Wenn der Roboter größere Schritte durchführt, dann wird der Ziel-ZMP in Richtung der Fußspitze verschoben. Dadurch wird auch der Massenschwerpunkt nach vorne bewegt. Dieser Ansatz ist inspiriert durch die Beobachtung, dass Menschen bei schnellerem Laufen ihren Schwerpunkt scheinbar weiter nach vorne verlagern.

3 Empirische Analyse der Laufbewegung

Wie im vorherigen Abschnitt stellenweise angemerkt wurde, verfügt der Laufalgorithmus über Parameter, Optionen und Teilkomponente, die die Stabilität des Laufs verbessern sollen. Um die Effektivität der einzelnen Maßnahmen zu untersuchen, wurde ein Python-Tool entwickelt, welches ein Log analysiert und Statistiken berechnet. Anhand der extrahierten Kennwerte kann anschließend eine Entscheidung getroffen werden, welche Komponenten im Wettbewerb verwendet werden.

Insbesondere während eines Wettbewerbs ist die manuelle Kalibrierung zeitlich sehr aufwendig und unter den gegebenen Umständen, wie Umgebungslautstärke, Zeitdruck und Übermüdung, ist eine reine visuelle Bewertung der Parameter problematisch. Das entwickelte Python-Tool ist somit ein erster Schritt diesen Vorgang zu vereinfachen und zu objektivieren.

3.1 Methoden und Werkzeuge

Um die Effektivität einzelner Maßnahmen zu beurteilen, werden verschiedenen Kennzahlen über das Laufen erhoben. Die kleinste Auflösung, über die als sinnvoll erachtete Kennzahlen beim Laufen bestimmt werden können, ist die Schrittebene. Dabei wird das Laufen als eine Folge sich wiederholender Trajektorien betrachtet. Jede Trajektorie wird als Trial interpretiert und es ist möglich statistische Kennwerte sowohl innerhalb eines Trials (intratrial) als auch zwischen den Trials (intertrial) zu bestimmen. In den zwei folgenden Fragestellungen wird die Intertrial-Verteilung des kumulierten, absoluten Fehlers,

$$e_j = \sum_{i=0}^M |x_i^{soll} - x_i^{ist}| \tag{3.1}$$

wobei M der Anzahl der Frames des Schrittes j entspricht, eine zentrale Rolle einnehmen. Im Weiteren wird diese Verteilung als InterTCAED (**inter-Trial Cumulated Absolute Error Distribution**) bezeichnet. Die Verteilung wird dabei durch ein Tupel aus Minimum, unteres, mittleres und oberes Quartil sowie Maximum dargestellt. Durch die Verwendung des absoluten Fehlers wird verhindert, dass sich die Fehler durch eine Oszillation um den Sollzustand über den Schritt auslöschen und die Verwendung der Quantil auf eben der Schritte ermöglicht eine größere Robustheit gegenüber Schritten mit Ausreißercharakter.

Um eine Statistik über die Schritte zu ermitteln ist eine Vorverarbeitung der Logdaten

3 Empirische Analyse der Laufbewegung

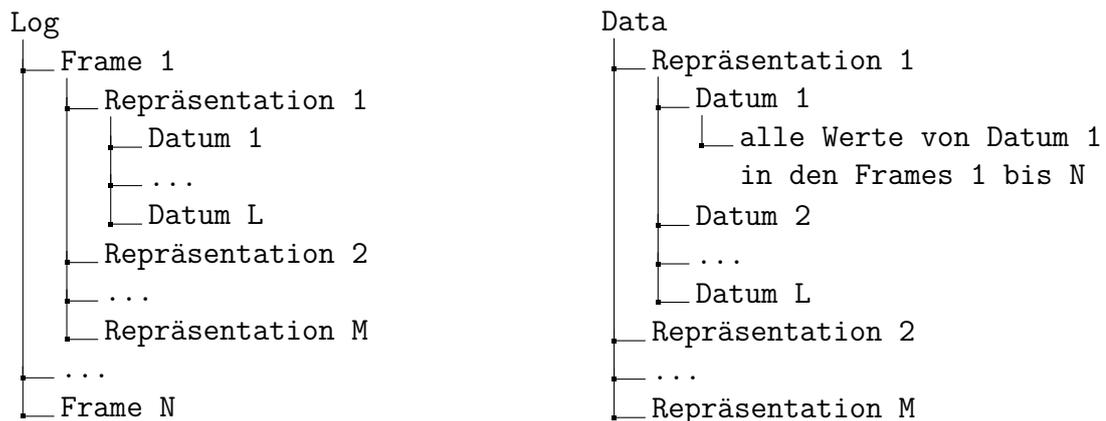


Abbildung 3.1: Links ist die Struktur der Logdatei zu sehen und rechts die Struktur der Daten nach der "Entfaltung und Plättung"

nötig. Als Erstes werden die Daten umstrukturiert. Während bei der Aufnahme es einfacher ist die Daten Frame für Frame anzuordnen und in jedem Frame die Werte der Repräsentation entsprechend der Struktur zu speichern, ist es für die Analyse besser ein "Zeitreihenformat" vorliegen zu haben. Im Prinzip wird die verschachtelte Struktur des Logs entfaltet und geplättet. Eine schematische Darstellung ist in Abbildung 3.1 zu sehen.

Anschließend wird die Verzögerung zwischen den Motorkommandos und den Sensorwerten entfernt. Die Verzögerung beträgt circa 40 ms und wird symmetrisch verteilt, d. h. es wird angenommen, dass es eine Verzögerung von 20 ms gibt zwischen dem Abschicken des Kommandos und dessen tatsächlichen Ausführung sowie eine Verzögerung von 20 ms zwischen der tatsächlichen Messung und der Bereitstellung des Messwertes im Programm.

Nun werden alle Zeitpunkte aus den Zeitreihen entfernt, die nicht während der Ausführung der Laufbewegung aufgenommen wurden. Im Anschluss kann das Zerteilen der Zeitreihen in einzelne Schritte erfolgen und abgebrochene Schritte, welche durch einen Sturz des Roboters entstehen können, entfernt werden. Zuletzt werden die Schritte noch danach sortiert, ob der linke, rechte oder kein Fuß gehoben wurde. In der Abbildung 3.2 ist die finale Datenstruktur schematisch dargestellt, welche die anschließende Analyse erleichtert.

In Abbildung 3.3 sind beispielhafte Ansichten der GUI zur Exploration der vorverarbeiteten Daten zu sehen. Die im vorherigen Abschnitt beschriebene finale Datenstruktur wird zur Bestimmung von Kennzahlen genutzt. Jeder Schritt wird als eigener Trial interpretiert und es werden sowohl Inter- als auch Intratrialstatistiken berechnet. Als Kennzahlen werden zurzeit der Mittelwert, die Standardabweichung, das Minimum, das Maximum, der Median und das untere und obere Quartil bestimmt. Gibt es Ist-Soll-Korrespondenzen, so werden die Statistiken auch über die relativen und absoluten Fehler bestimmt. Es werden 3 Darstellungsarten generisch unterstützt, so könne die Daten der Schritte einzeln und zusammengefasst als Mittelwert-Standardabweichung

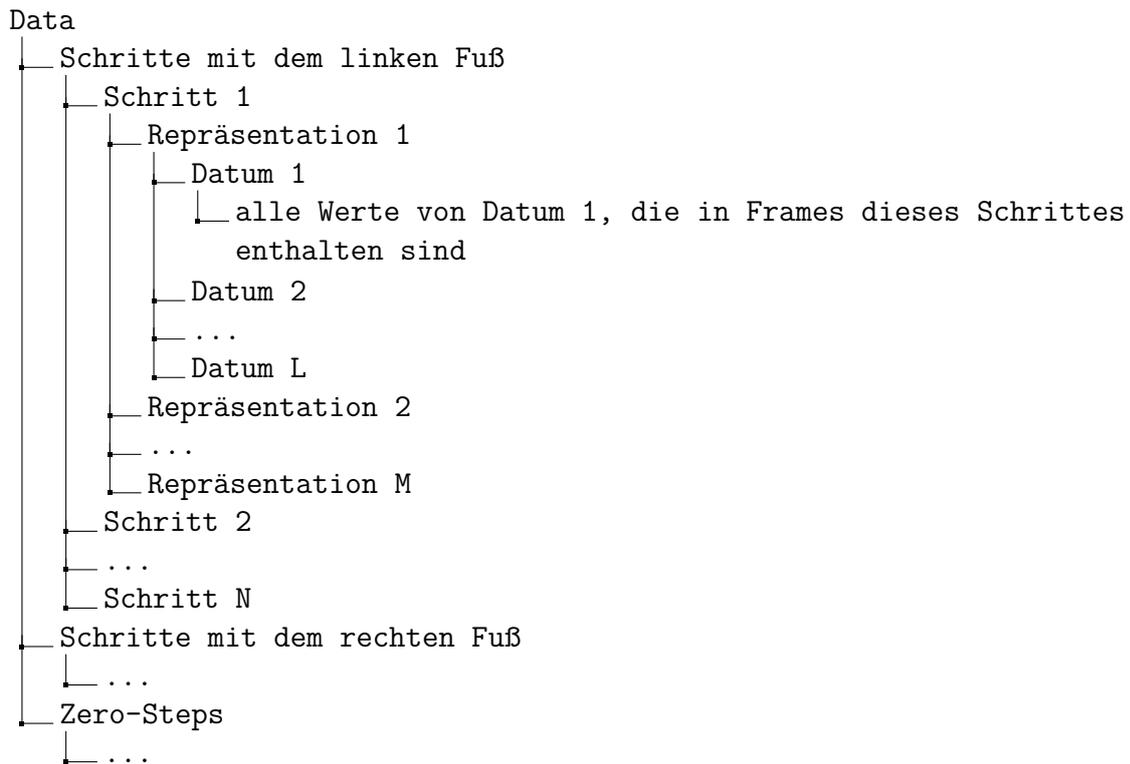


Abbildung 3.2: Die finale Struktur der vorverarbeiteten Daten.

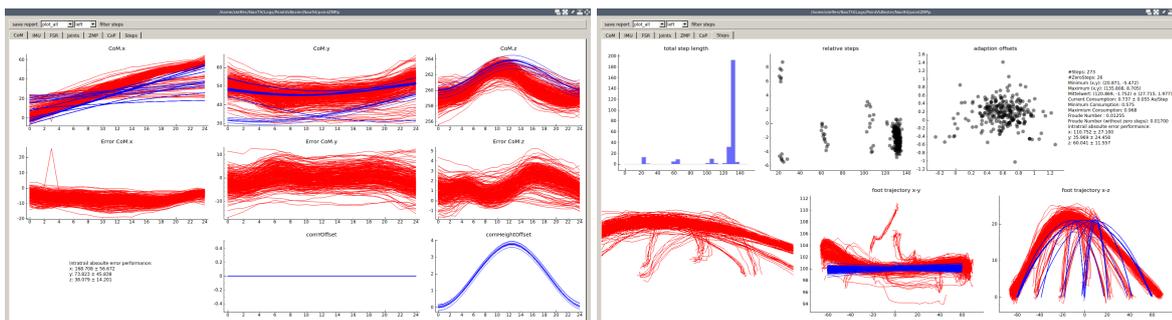


Abbildung 3.3: Beispielhafte Ansichten der GUI zur Exploration der Daten. Links sind die Trajektorien des Center of Mass und rechts Trajektorien des Schwingfußes sowie einige Daten über einige Schritte zu sehen.

oder Boxplot dargestellt werden. Es ist möglich die Daten nur für das rechte, linke oder beide Beine zusammen darstellen zu lassen. Nach der Betrachtung können die ermittelten, statistischen Kennzahlen in CSV-Dateien zur weiteren Verwendung exportiert werden.

3.2 Experimente

3.2.1 ZMP-Trajektorien

Während des RoboCup 2017 in Japan wurde eine neue Zieltrajektorie für den ZMP basierend auf den Erkenntnissen aus [12] und [23] entwickelt. Zuvor bestand die Referenztrajektorie aus einem Punkt, der innerhalb der Fußkontaktfläche vorgegeben wird. Wechselt der Standfuß, so springt die Vorgabe von der Kontaktfläche des alten Standfußes zur Kontaktfläche des neuen Standfußes. Es gibt also keinen Zeitpunkt, zu dem die Zielvorgabe außerhalb der Standfußfläche liegt. Im weiteren Verlauf wird diese Trajektorie zur einfacheren Unterscheidung als Punkt-ZMP bezeichnet.

Bei der während des RoboCup 2017 erstellten Trajektorie wird der zeitliche Verlauf des ZMPs entlang der y-Achse durch eine Bezierkurven 3. Grades und entlang der x-Achse durch eine Gerade beschrieben. Zu Beginn des Schrittes liegt der Ziel-ZMP zwischen den beiden Füßen und wandert in Richtung des Standfußes. Nachdem der Schritt seinen zeitlichen Mittelpunkt überschritten hat, wandert die Zielvorgabe für den ZMP wieder in Richtung Robotermitte, um beim Ende des Schritts wieder zwischen den Füßen zu liegen. Die Hoffnung war, dass dadurch die Bewegung des Roboters "flüssiger" wird bzw. etwaige starke Beschleunigungen, die den Roboter destabilisieren, durch die nicht sprunghafte Änderungen der ZMP-Vorgabe reduziert werden. Im weiteren Verlauf wird diese Trajektorie zur einfacheren Unterscheidung als Bezier-ZMP bezeichnet.

Leider ist es schwer visuell zu überprüfen, welche der Trajektorien besser für den Roboter geeignet ist.

Experimenteller Aufbau

Es wurde der Roboter NAO der Firma SoftBank Robotics verwendet. Dabei wurden die Parameter vom RoboCup 2018 verwendet und bis auf die Parameter für die ZMP-Trajektorie und die Schrittsynchronisation (deaktiviert) nicht verändert. Der NAO-Roboter mit dem Namen Nao94 lief auf dem Kunstrasenfeld des NaoTH-Labors in den Tests jeweils für 10 Sekunden gerade aus. Das Verhalten forderte dabei normale Schritte an, d. h. der Path-Planner wurde nicht verwendet. Dies wurde 10 mal wiederholt, wobei insgesamt circa 540 Schritte (Punkt-ZMP 546, Bezier-ZMP 548) vom Roboter durchgeführt wurden. Dabei wurden für die Analyse notwendige Repräsentationen aufgenommen und mit der oben erläuterten Software verarbeitet.

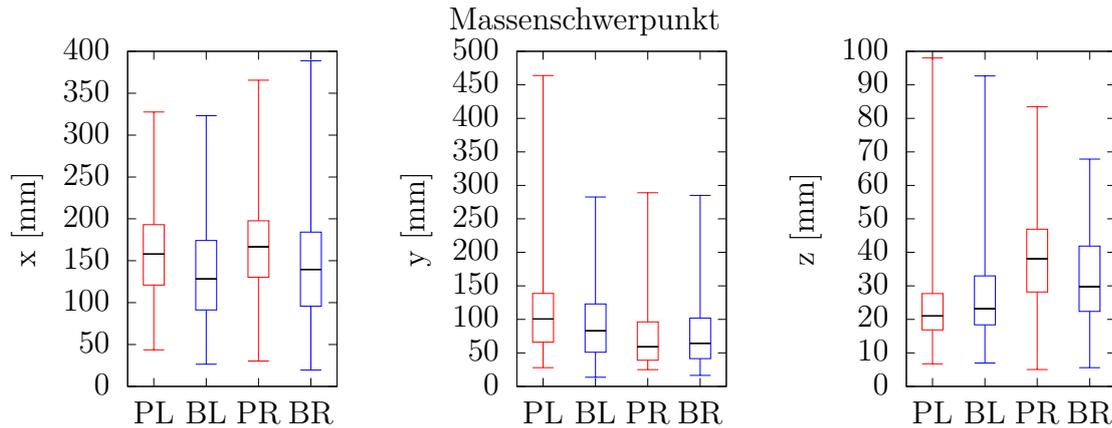


Abbildung 3.4: InterTCAED-Werte des CoM-Fehlers in x, y und z auf dem linken (L) und rechten (R) Standbein mit dem Punkt-ZMP (P) und dem Bezier-ZMP (B)

Auswertung

Betrachtet man die InterTCAED-Werte des Centre Of Mass in Abbildung 3.4 so fällt auf, dass der Fehler in der x Komponente mit dem Bezier-ZMP geringer ausfällt als unter Verwendung des Punkt-ZMPs. In der y-Richtung reduziert sich mit dem Bezier-ZMP der Fehler im Median geringfügig auf dem linken Bein und steigt minimal auf dem Rechten. Umgekehrtes gilt für die z-Richtung. Interessant ist, dass sich die InterTCAED-Werte für die x-Richtung ähnlich für beide Beine verhalten, in den anderen beiden Richtungen jedoch nicht. In der y-Richtung sind die InterTCAED-Werte der Schritte auf dem rechten Bein unabhängig von der Wahl des ZMPs im Median kleiner, währenddessen in der z-Richtung die InterTCAED-Werte der Schritte auf dem linken Bein kleiner sind.

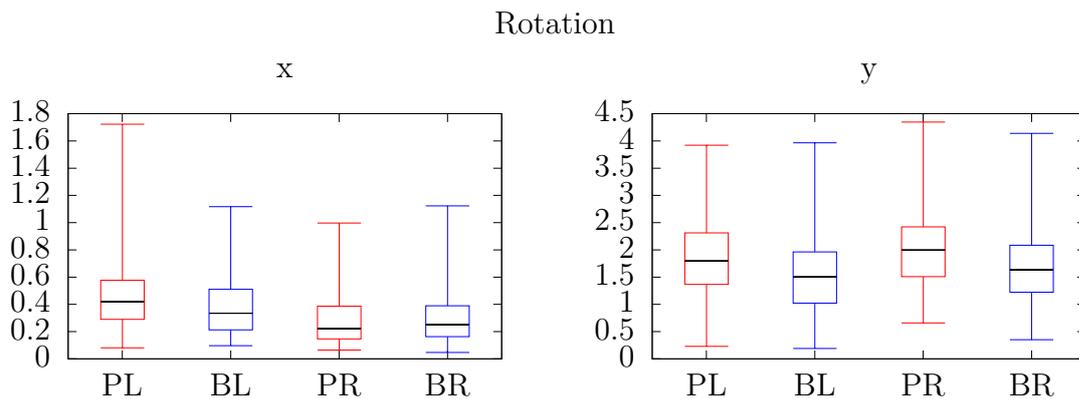


Abbildung 3.5: InterTCAED-Werte der Abweichung von der Soll-Orientierung in Rotationsvektordarstellung in x und y auf dem linken (L) und rechten (R) Standbein mit dem Punkt-ZMP (P) und dem Bezier-ZMP (B)

3 Empirische Analyse der Laufbewegung

Die Orientierung verbessert sich mithilfe des Bezier-ZMPs geringfügig bis auf die Rotation um x auf dem rechten Bein, wo der Median des InterTCAED-Wertes leicht gestiegen ist (Abbildung 3.5).

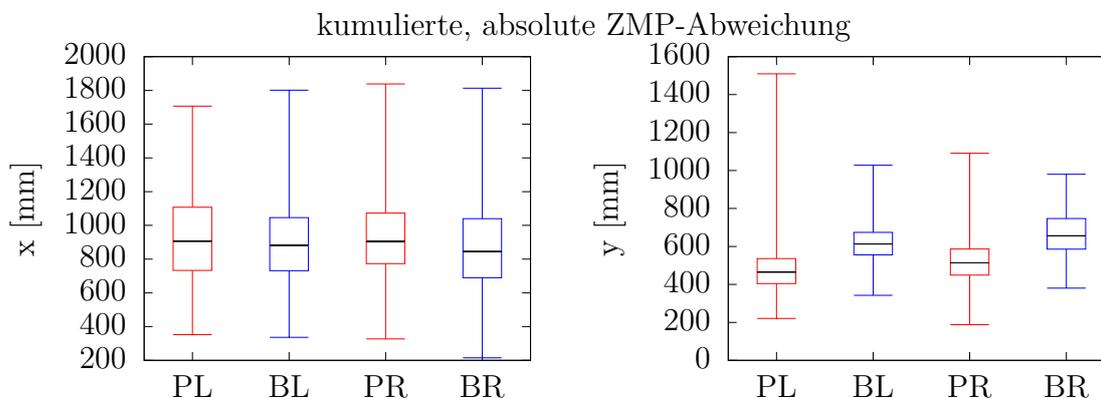


Abbildung 3.6: InterTCAED-Werte des ZMP in x und y auf dem linken (L) und rechten (R) Standbein unter Verwendung des Punkt-ZMP (P) und des Bezier-ZMP (B) als Referenz.

Die InterTCAED-Werte der mithilfe des Cart-Table-Modells geschätzten ZMP-Trajektorie bezüglich x verändert sich kaum und bleibt auf gleichem Niveau. Im Gegensatz dazu verschlechtern sich die InterTCAED-Werte im Median für die ZMP-Trajektorie bezüglich y bei der Verwendung des Bezier-ZMPs (Abb. 3.6.)

Betrachtet man zusätzlich die Inter-Trial-Verteilungen (InterTD) der Trajektorien des mit dem Cart-Table-Model ermittelten ZMPs zu den einzelnen Zeitpunkten der Schritte unter Verwendung des Point und Bezier-ZMPs als Referenz in der x- und y-Richtung (Abb. 3.7) so fällt auf, dass diese Trajektorien sich im Median kaum unterscheiden. Interessanterweise scheint die Wahl der ZMP-Trajektorie nahezu keinen Einfluss auf die resultierende ZMP-Trajektorie zu haben. So ist in x-Richtung kein lineares Ansteigen des ZMPs über den Schritt zu beobachten und in y-Richtung weder ein konstantes Verhalten für den Punkt-ZMP noch die Bezier-Kurve für den Bezier-ZMP zu sehen. Die Steigerung des InterTCED-Wertes für y könnte somit durch die geänderte Referenztrajektorie in y erklärt werden.

Für diese Beobachtung kann es mehrere Ursachen sowohl bezüglich der Ausführung als auch der Messung geben. Eine mögliche Ursache kann der Optimal-Controller sein, der für diese beiden ZMP-Zieltrajektorien ähnliche CoM-Trajektorien produziert und damit ähnliche gemessene ZMP-Trajektorien bei der Ausführung entstehen. Dies wäre dann ein grundsätzliches Problem und könnte nur durch eine Änderung der Gewichte des Performanceindex behoben werden. Zum Optimal-Controller sei noch angemerkt, dass bei der Ermittlung der Kontrollparameter keine Beschränkungen des physikalischen Systems einfließen, d. h. theoretisch könnten vom Roboter Beschleunigungen und Änderungen dieser gefordert werden, die er aus physikalischen Gründen nicht erzeugen kann. Eine weitere Ursache könnte sein, dass zurzeit nur die momentane Position

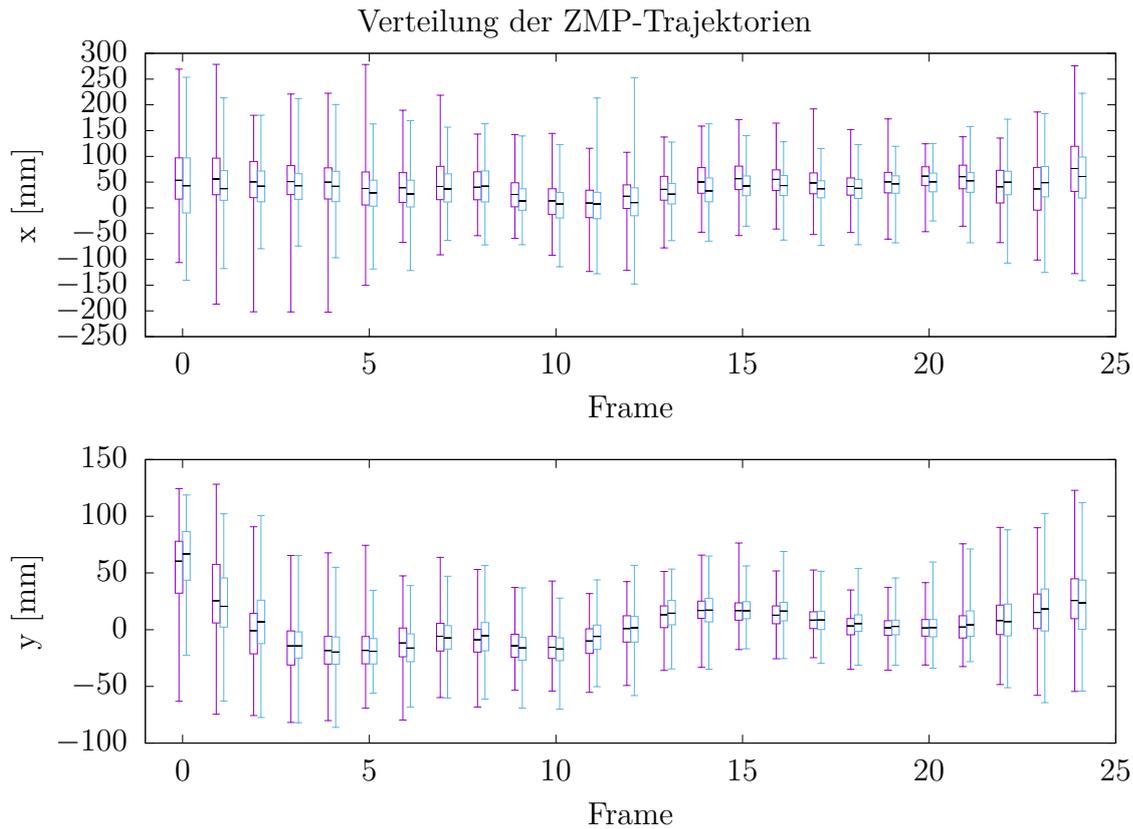


Abbildung 3.7: Die Inter-Trial-Verteilungen (InterTD) des ZMPs in x und y auf dem rechten Standbein unter Verwendung des Punkt-ZMP (Violett) und des Bezier-ZMP (Hellblau) als Referenz.

des Controllers versucht wird zu erreichen und nicht der komplette Zustand inklusive Geschwindigkeit und Beschleunigung. Da insbesondere die Beschleunigung nicht beachtet wird, jedoch explizit mit dem ZMP zusammen hängt, kann es sein, dass die gewünschte Trajektorie nicht erreicht wird. Dieses Problem bezüglich der Ausführung, könnte durch die Verwendung von State-Feedback im Optimal-Controller realisiert werden, da der Controller die Abweichung vom Sollzustand der Lage, Geschwindigkeit und Beschleunigung für die nächsten Kontrollsignale beachten kann. Bezüglich der Messung ist es problematisch, dass im Cart-Table-Model die vom Accelerometer gemessene Beschleunigung als Approximation der CoM-Beschleunigung verwendet wird. Es existiert zurzeit kein Modell oder Filter für den CoM und dessen Dynamik. Somit wird weder das Rauschen der Gelenke, das sich auf die ermittelte CoM-Position auswirkt, als auch das Sensorrauschen der Inertial Measurement Unit (IMU), sowie die Offsets des Accelerometers, die sich auf die geschätzte Beschleunigung auswirken, berücksichtigt. Ein Modell, das dies berücksichtigt könnte zu genaueren Schätzungen des ZMPs führen.

Eine weitere Möglichkeit den ZMP zu messen ist der Centre of Pressure (CoP), welcher mit dem ZMP übereinfällt, sofern er existiert. Existiert der ZMP nicht, so müsste der CoP am Rand der von den Kraftmesssensoren aufgespannten Polygons sein oder

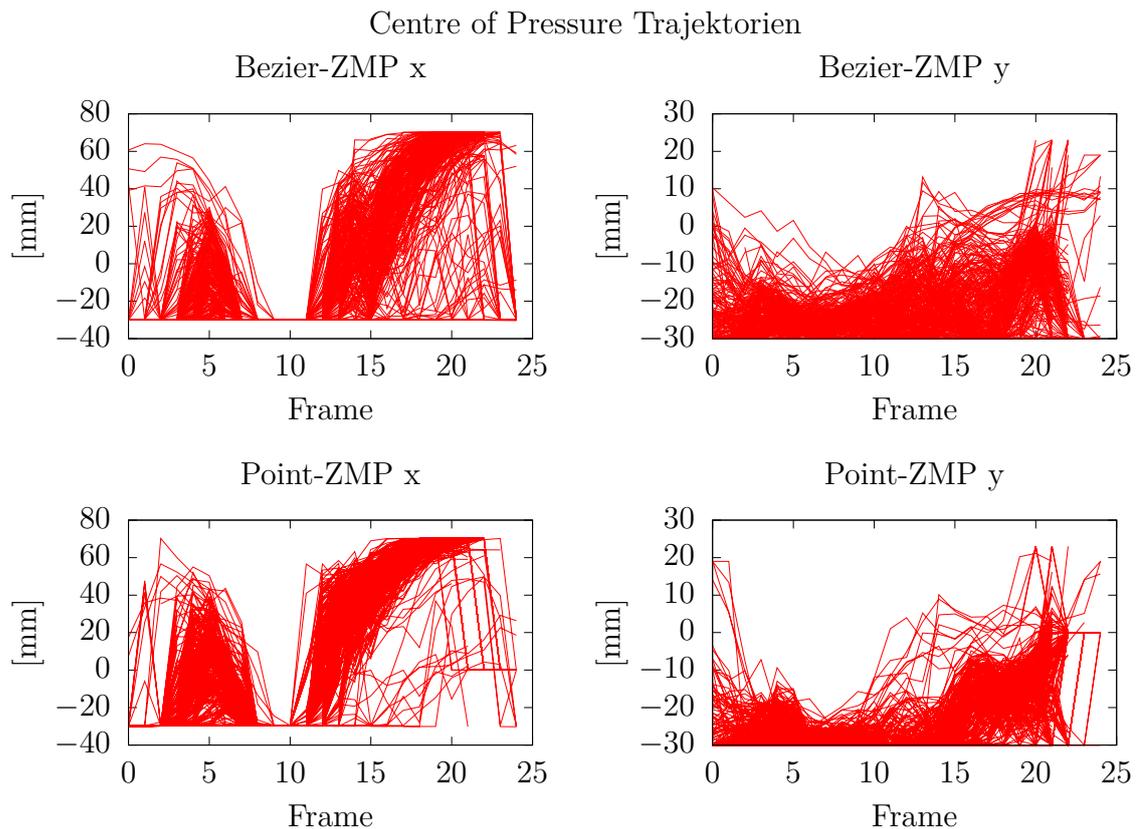


Abbildung 3.8: Der Verlauf des Centre of Pressure innerhalb eines Schrittes mit dem linken Bein in x (links) und y (rechts) Richtung. Oben sind die Trajektorien für den Bezier-ZMP und unten für den Point-ZMP zu sehen.

im Zentrum des Polygons mit Amplitude 0 liegen. In 3.8 sind die Trajektorien des Centre of Pressure in x und y zu sehen. Es fällt auf, dass der CoP meist am Fußende beginnt, sich kurzzeitig schnell nach vorn und wieder zurückbewegt, um dann ab der Mitte des Schrittes wieder bis ganz nach vorne zu wandern. Während dessen verbleibt der CoP die meiste Zeit auf der äußeren Hälfte des Fußes. Dabei verbleibt besonders in der ersten Hälfte des Schrittes der Centre of Pressure unter der Verwendung des Point-ZMPs näher an der Außenkante als unter der Verwendung des Bezier-ZMPs.

Betrachtet man nur die Beobachtungen des CoMs, der Torsoorientierung und des CoPs in Bezug zur anfänglichen Fragestellung, so scheint der Bezier-ZMPs während eines Schrittes wünschenswerte Auswirkungen zu haben und sollte diesen Ergebnissen nach bevorzugt in Wettbewerben Verwendung finden.

3.2.2 Parameter Hip-Roll-Scaling

Ein Parameter, der auf der Gelenkebene wirkt, ist der Hip-Roll-Scaling-Faktor. Der Parameter wurde hinzugefügt, als bei älteren Naos beobachtet wurde, dass die Roboter in der Hüfte einknicken. Die Zielposition des Hip-Roll-Gelenks wird mit diesem Faktor

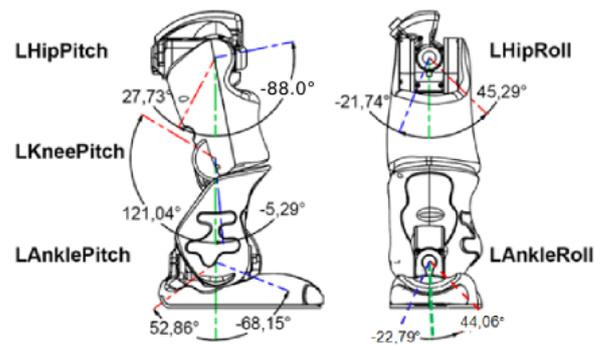


Abbildung 3.9: Schematische Darstellung des linken Beins und des Bewegungsraums seiner Gelenke. Der Parameter Hip-Roll-Scaling schränkt den Bewegungsspielraum zum Roboter hin für das Hip-Roll-Gelenk ein. Bildquelle: http://doc.aldebaran.com/2-1/_images/hardware_llegjoint.png

multipliziert, falls das Gelenk einen Winkel ansteuern soll, der einem Nachinnenknicken entspräche (siehe Abbildung 3.9). Je kleiner der Faktor, desto näher bleibt die Zielvorgaben für das Gelenk beim Winkel 0° , was der Gelenkposition beim aufrechten Stehen auf einer Ebene entspricht. Es wird somit im Kontrollraum eingeschränkt. Dies passiert transparent für alle übergeordneten Stabilisierungsmechanismen. Es soll nun überprüft werden, ob dieser Parameter noch benötigt wird.

Experimenteller Aufbau

Es wurde der Roboter NAO der Firma SoftBank Robotics verwendet. Dabei wurden die Parameter vom RoboCup 2018 verwendet und bis auf die Parameter für das Hip-Roll-Scaling und die Schrittsynchronisation (welche deaktiviert wurde) nicht verändert. Der NAO-Roboter mit dem Namen Nao94 lief auf dem Kunstrasenfeld des NaoTH-Labors auf der Stelle. Das Verhalten forderte dabei normale Schritte an, d. h. der Path-Planner wurde nicht verwendet. Um den Einfluss des Hip-Roll-Scaling Parameters zu untersuchen werden die resultierenden Trajektorien unter Verwendung des aktuellen Standardwertes von 0.6 verglichen mit den resultierenden Trajektorien unter Verwendung eines Parameterwertes von 1, was keiner Einschränkung des Gelenks entspricht. Es wurden jeweils 295 Schritte vom Roboter durchgeführt. Dabei wurden für die Analyse notwendige Repräsentationen aufgenommen und mit der oben erläuterten Software verarbeitet.

Auswertung

In Abbildung 3.10 sind die InterTCAED-Werte des Standbeins und des Schwungbeins zu sehen. Die kumulierten, absoluten Fehler im Hip-Roll-Gelenk sind im Median mit einem Hip-Roll-Scaling-Wert von 1.0 sehr leicht reduziert. Interessanterweise sind die Auswirkungen im Hip-Pitch-Gelenk am größten. Dabei ist zu beachten, dass sich das untere und obere Quartil kaum verändern.

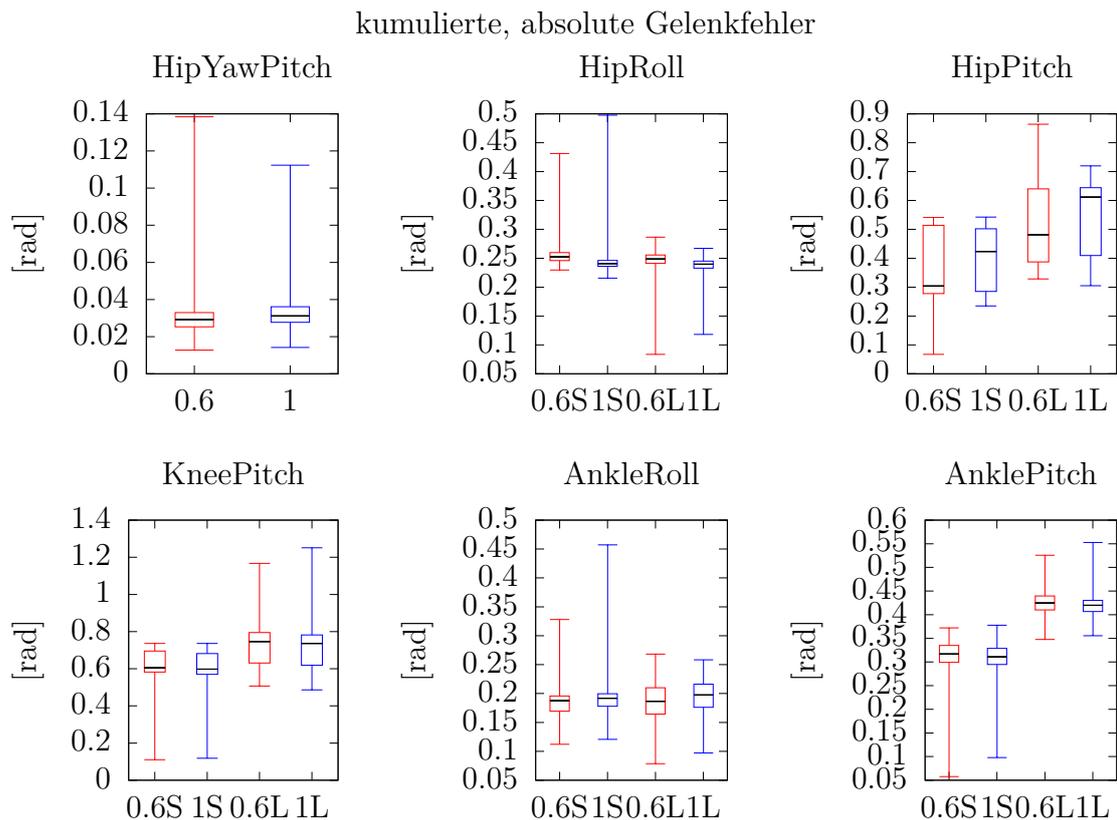


Abbildung 3.10: InterTCAED-Werte der Beingelenke für die Werte 0.6 und 1 für den Parameter Hip-Roll-Scaling des Standbeins (S) und des Schwungbeins (L).

Betrachtet man die Fehler in für den Center Of Mass so fällt auf, dass wenn der Bewegungsspielraum des Hip-Roll-Gelenks nicht eingeschränkt wird, die Streuung des Fehlers des Center of Mass steigt. Der Fehler steigen insbesondere im Median für die y und die z Komponente, sinken jedoch minimal für die x-Komponente. Unterteilt man die Fehler auf Schritte mit linkem und rechtem Standbein, so ist in Abbildung 3.11 zu sehen, dass der Fehler in x und y auf dem rechten Bein größer als auf dem Linken sind. Für die z Komponente verhält es sich umgekehrt.

Die Orientierung des Torsos (siehe Abbildung 3.12) lässt eine leichte Verbesserung um die x-Achse auf dem rechten Bein und eine leichte Reduzierung des Fehlers im Median um die y-Achse mit einer leichten Steigerung der Streuung des oberen Quantils erkennen. Es fällt auf, dass der Median um die x-Achse auf dem rechten Bein stets höher ist als auf dem linken Bein.

In Abbildung 3.13 sind die Trajektorien des jeweiligen Schwungbeins in den Koordinaten des Standfußes in der y-z-Ebene zu sehen. Es fällt auf, dass der Schwungfuß sich als erstes leicht vom Standfuß entfernt und anschließend beim Anheben des Fußes Richtung Standfuß driftet. Nachdem der Schritt zur Hälfte ausgeführt wurde und wieder Richtung Boden bewegt wird, bewegt sich der Schwungfuß relativ weit nach außen vom

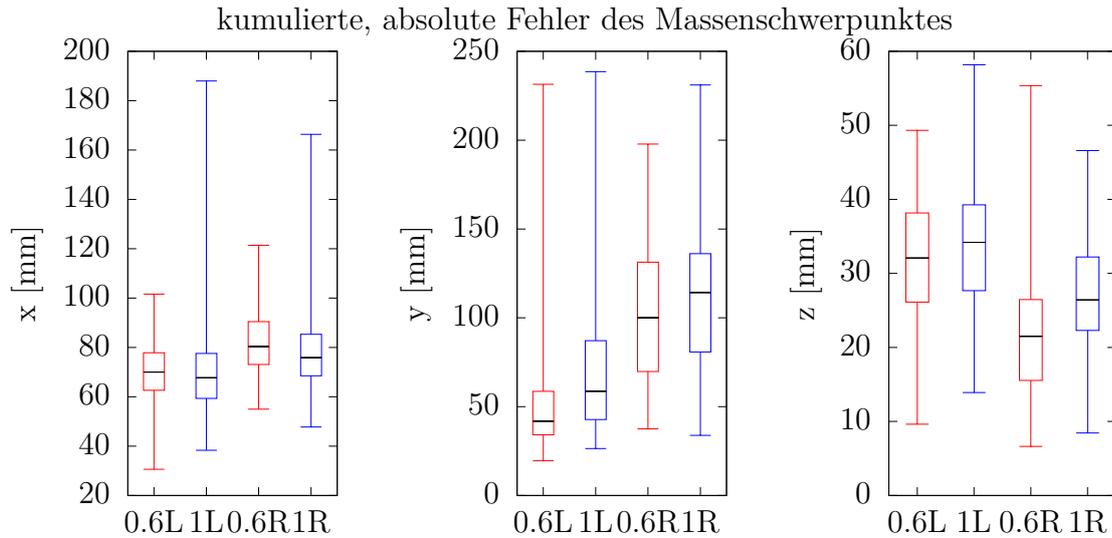


Abbildung 3.11: InterTCAED-Werte des Centre of Mass im Standfuß unter Berücksichtigung der IMU aufgeteilt in Schritte auf dem linken Bein (L) und dem rechten Bein (R)

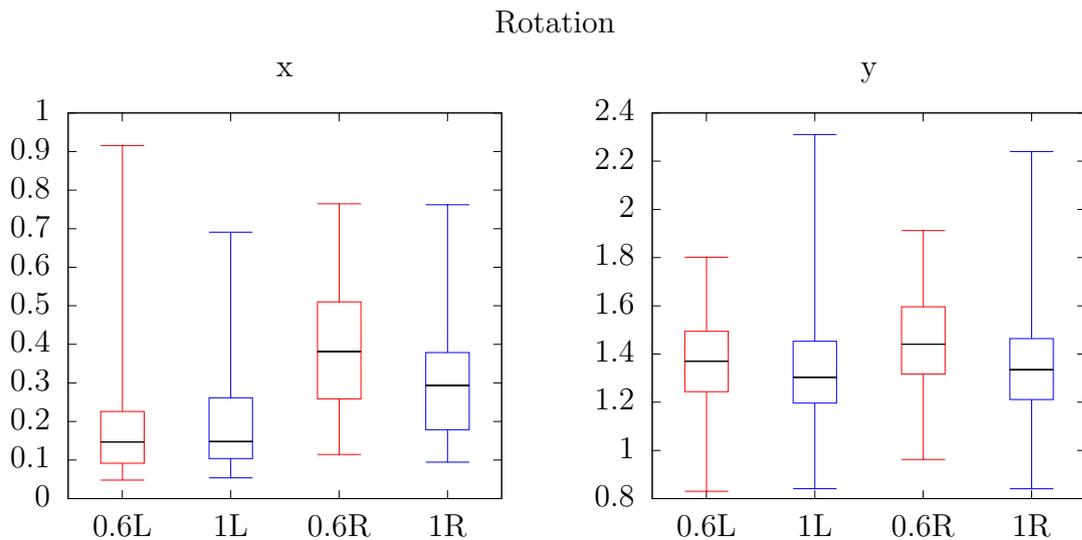


Abbildung 3.12: Kumulierte, intratrial, absolute Fehlerverteilung der Orientierung des Torsos aufgeteilt in Schritte auf dem linken Bein (L) und dem rechten Bein (R) und kombiniert (C)

Standfuß wieder weg. Die Spannweite in der y -Achse beträgt ungefähr $4 - 5\text{mm}$. Es fällt auf, dass die Füße mit dem Parameter 1.0 nicht so weit hochgehoben werden, wie beim Parameter 0.6.

In Abbildung 3.14 sind die InterTCAED-Werte aufgetrennt nach Parameter und Schwungfuß zu sehen. Die kumulierten Fehler in der y -Richtung werden im Median

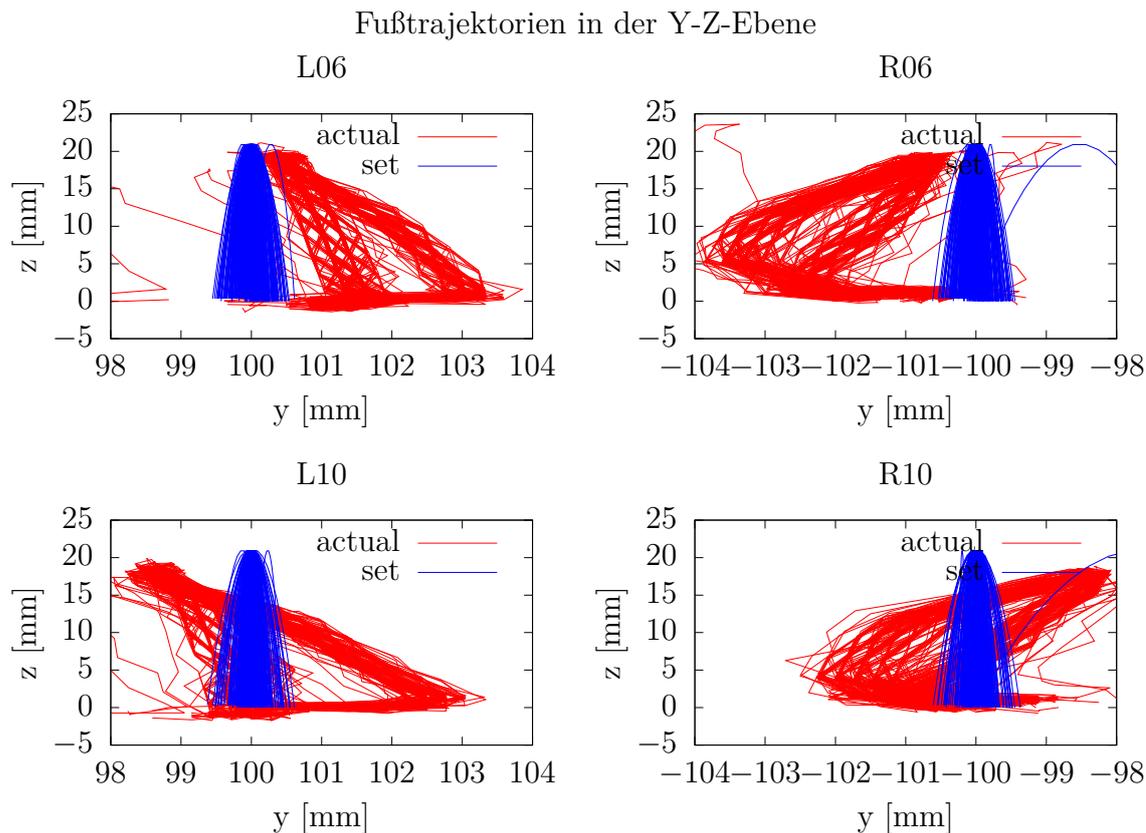


Abbildung 3.13: Die Trajektorien des linken (L) und des rechten (R) Schwungbeins für die Parameterwerte 0.6 (06) und 1.0 (10).

reduziert. Dabei nimmt die Streuung für den rechten Schwungfuß ab und für den linken Schwungfuß zu.

Die Verwendung eines Hip-Roll-Scaling-Faktors von 1.0 kann nach derzeitigem Kenntnisstand nicht empfohlen werden. Die minimalen Verbesserungen im Median der Torsoorientierung können die stärkere Streuung in der Orientierung, sowie den im Median gestiegenen Fehler der CoM-Position, nicht rechtfertigen. Es bleibt zu untersuchen, weshalb diese Änderung so einen kontraintuitiven Effekt im Hip-Pitch-Gelenk hat. Unter Umständen müssen die Parameter für die Torso- und Fußstabilisierung angepasst werden, da diese nun einen größeren, direkteren Einfluss auf das Hip-Roll-Gelenk haben.

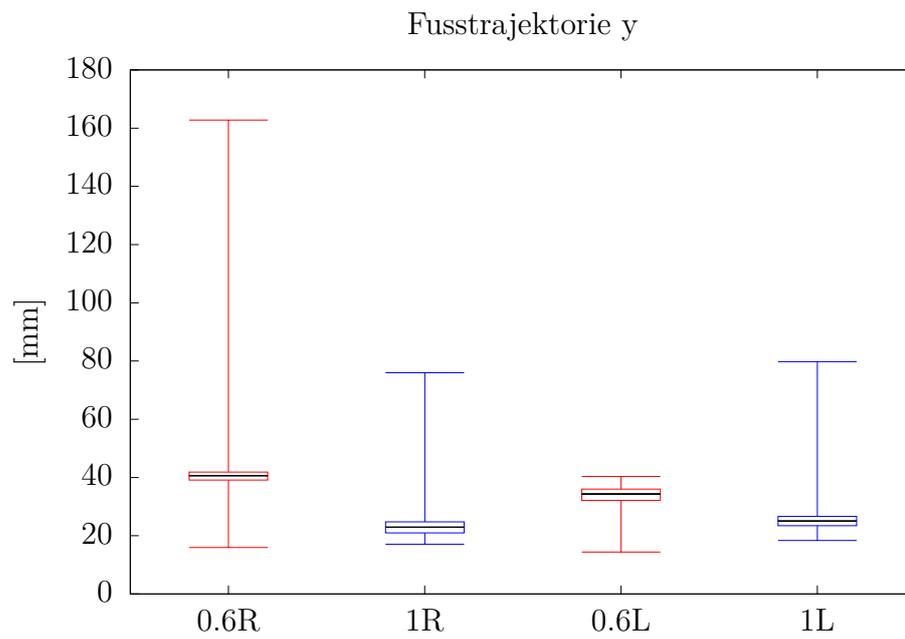


Abbildung 3.14: Die kumulierten absoluten intertrial Fehlerverteilung der y Trajektorie des Schwungfußes im Standfuß.

4 Zusammenfassung und Ausblick

In der vorliegenden Arbeit wurde zunächst ein grober Überblick über den aktuellen Stand in der Standard Platform League des RoboCups bezüglich der verwendeten Lauf-Algorithmen gegeben. Anschließend wurde der aktuelle Lauf-Algorithmus des NAO Team Humboldt näher erläutert, welcher im Rahmen dieser Arbeit überarbeitet und modularisiert wurde. Des Weiteren wurde ein Tool entwickelt, dass die manuelle Kalibrierung des Lauf-Algorithmus vereinfachen und objektivieren soll, entwickelt und bei der Untersuchung der ZMP-Trajektorie und des Hip-Roll-Scaling-Parameters verwendet.

Das Analyse-Tool ist zurzeit noch recht rudimentäre. So ist es z. B. auf der Schritzebene allein nicht möglich das Aufschwingverhalten des Torsos über mehrere Schritte zu untersuchen und zu bewerten. Eine Erweiterung auf Schrittfolgen scheint deshalb sinnvoll. Außerdem wäre eine Analyse mithilfe von Korrelationen zwischen den einzelnen Größen interessant, wie z. B. zwischen den CoM- und Gelenkfehlern. Eine Untersuchung von realen Spieldaten sollte als Langzeitziel in Betracht gezogen werden. Um dies zu ermöglichen müssen einige Anpassungen vorgenommen werden. So sollten Filterfunktionen in der GUI eine Exploration der Daten vereinfachen und es müssen Schritte variabler Länge unterstützt werden.

Es wird zurzeit nur die Eigenstabilität untersucht, d. h. die Performance des Laufs hängt nur von der Eigendynamik des Roboters und vom Feld ab. Im Wettbewerb interagieren die Roboter jedoch auch physisch. Somit sollte die Stabilität des Laufens auf äußere Einflüsse, z. B. durch Stöße und auf dem Boden liegenden Hindernissen, untersucht werden. Dabei sollte besonderes Augenmerk auf die Reproduzierbarkeit der Experimente gelegt werden.

Bezüglich des Lauf-Algorithmus scheint der nächstliegende Schritt den Zustand als Feedback im Optimal-Controller einfließen zu lassen. So wird die CoM-Trajektorie ohne Feedback verfolgt und die vorgegebenen Geschwindigkeits- und Beschleunigungsreferenzen nicht beachtet. Da das Laufen von der Bewegung her recht repetitiv ist, könnte die Anwendung von "Iterative Learning Control", "Reinforcement Learning" oder die Optimierung der Modellparameter helfen die Grundstabilität zu verbessern. Das Laufen könnte sich so über die Zeit den unterschiedlichen Feldeigenschaften anpassen ohne das ein manueller Eingriff nötig ist.

Literaturverzeichnis

- [1] Darshana Adikari, Georg Felbinger, Arne Hasselbring, Yuria Konda, René Kost, Pascal Loth, Lasse Peters, Nicolas Riebesel, Thomas Schattschneider, Felix Warmuth, and Felix Wege. HULKs Team Research Report 2017. Technical report, Hamburg University of Technology, 2017. https://hulks.de/_files/TRR_2017.pdf.
- [2] Masoud Khairi Atani, Sina Moqadam Mehr, Ali Sirouszia, Erfan Kouzehgaran, Mohammad Ali Sharpasand Farzad F. Bigelow, Aref Moqadam Mehr, and Novin Shahroudi. MRL-SPL Team Description for RoboCup 2017. Technical report, Mechatronics Research Lab., Dept. of Computer and Electrical Engineering, Qazvin Azad University, Qazvin, Iran, 2017. https://www.robocup2017.org/file/symposium/soccer_std_plf/MRL-SPL_TDP_Robocup_2017.pdf.
- [3] Sven Behnke. Online Trajectory Generation for Omnidirectional Biped Walking. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pages 1597–1603, 2006.
- [4] Tim Bohn, Sina Ditzel, Dipl.-Inf. Andreas Fürtig, Louis Glaser, Timm Hess, Hans-Joachim Hammer, Elena Knorr, Kyle Rinfreschi, Tim Schön, Jens-Michael Siegl, Stefanie Steiner, Felix Weiglhofer, and Philipp Wörner. Bembelbots Frankfurt. Technical report, Goethe University, 2018. <http://www.jrl.cs.uni-frankfurt.de/web/wp-content/uploads/2018/01/teamreport18-bembelbots.pdf>.
- [5] Claudio Cimorelli, Yan Medernach, Gary Cornelius, Ana P. Marquez Sanchez, Gabriele Sartor, Monia Chowdhri, Alexander Eyjofsson, and Patrice Caire. Luxembourg United Team Description Report 2017. Technical report, Interdisciplinary Centre for Security, Reliability and Trust (SnT) of the University of Luxembourg, 2017. https://robocup2017.org/file/symposium/soccer_std_plf/7a156a11b879601374df64401fb2e692.pdf.
- [6] Jorge Luis Copete, Masaka Kataoka, Yuki Yamashita, Koki Ichinose, Wu Binyi, Kazuki Tachikawa, Kyoichiro Kobayashi, Naoki Umeda, Kohei Fukuda, Niaty Rawal, Yuji Kawai, and Minoru Asada. JoiTech-SPL Team Description 2017. Technical report, Adaptive Machine Systems, Graduate School of Engineering, Osaka University, 2017. https://www.robocup2017.org/file/symposium/soccer_std_plf/JoiTech-SPL_TDP2017_May2017.pdf.
- [7] Donovan Crichton, Vladimir Estivill-Castro, Jonathan Ferrer-Mestres, Hector Geffner, Eugene Gilmore, Mick Hawkins, René Hexel, Misbah Javaid, Dimitri Joukoff,

- Carl Lusty, Brenainn Moushall, and Rafael Pous. Mi-Pal 2017 Team Description Paper. Technical report, Institute of Intelligent and Integrated Systems, Griffith University, Queensland 4111, Australia and Departament de Tecnologies de la Informació i les Comunicacions, Universitat Pompeu Fabra, Barcelona 08018, Spain, 2017. https://www.robocup2017.org/file/symposium/soccer_std_plf/TDPTemplate.pdf.
- [8] Georgios Darivianakis, Benjamin Flamm, Simon Maurer, Marc Naumann, Louis Lettry, and Alex Locher. NomadZ Team Description Paper for RoboCup 2017. Technical report, Automatic Control Lab Computer Vision Lab D-ITET and ETH Zurich, 2017. https://robocup2017.org/file/symposium/soccer_std_plf/NomadZ_TDP_2017.pdf.
- [9] Bernhard Hengst. rUNSWift Walk2014 Report RoboCup Standard Platform League. Technical report, School of Computer Science & Engineering, University of New South Wales, Sydney 2052, Australia, 2014. <http://cgi.cse.unsw.edu.au/~robocup/2014ChampionTeamPaperReports/20140930-Bernhard.Hengst-Walk2014Report.pdf>.
- [10] Kenta Hidaka, Tatsuya Tsubakimoto, Toshiyuki Tanaka, Kenya Kumagai, Hiroaki Matsubara, Yoh Aizawa, Yuto Iwai, Marina Nakagawa, Kana Futatsuishi, Nodoka Mori, Kosei Ohkusu, Kazuho Takahashi, Yoshiyuki Uemura, Kazuki Ito, Haruki Niwa, Takuo Suzuki, and Kunikazu Kobayashi. Camellia Dragons Team Description Paper for Robocup 2017. Technical report, School of Information Science and Technology, Aichi Prefectural University, 2017. https://robocup2017.org/file/symposium/soccer_std_plf/CamelliaDragonsTDP2017final.pdf.
- [11] Matthias Hofmann, Arne Moos, Fabian Rensen, Ingmar Schwarz, and Oliver Urbann. Playing Soccer Outdoors with Humanoid Robots. In *Proceedings of the 11th Workshop on Humanoid Soccer Robots in conjunction with the 2016 IEEE-RAS International Conference on Humanoid Robots*, 2016.
- [12] Matthias Hofmann, Ingmar Schwarz, Oliver Urbann, and Aaron Larisch. Nao Devils Dortmund Team Report 2017. Technical report, Robotics Research Institute, Technische Universität Dortmund, 2017. <https://github.com/NaoDevils/CodeRelease/blob/CodeRelease2017/TeamReport2017.pdf>.
- [13] Nao-Team HTWK. Team Research Report Nao-Team HTWK. Technical report, HTWK Leipzig, 2017. https://htwk-robots.de/documents/TRR_2017.pdf?lang=en.
- [14] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Kazuhito Yokoi, and Hirohisa Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, pages 1620–1626. IEEE, 2003.

- [15] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kazuhito Yokoi, and Hirohisa Hirukawa. The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems: Expanding the Societal Role of Robotics in the the Next Millennium*, pages 239–246. IEEE, 2001.
- [16] Shu Li, Zhiying Zeng, Ruiming Zhang, Zhongde Chen, and Dairong Li. TJArk Team Description Paper & Research Report 2016. Technical report, TJArk, Robots & Intelligent Systems Laboratory Tongji University, 2016. https://github.com/TJArk-Robotics/coderelease_2016/blob/master/TJArk%20Team%20Research%20Report%202016.pdf.
- [17] Nicolas Riebesel, Arne Hasselbring, Lasse Peters, and Finn Poppinga. Team Research Report 2016. Technical report, Hamburg University of Technology, 2016. https://hulks.de/_files/TRR_2016.pdf.
- [18] Thomas Röfer, Tim Laue, Yannick Bülter, Daniel Krause, Jonas Kuball, Andre Mühlenbrock, Bernd Poppinga, Markus Prinzler, Lukas Post, Enno Roehrig, René Schröder, and Felix Thielke. B-Human Team Report and Code Release 2017. Technical report, Deutsches Forschungszentrum für Künstliche Intelligenz und Universität Bremen, 2017. <https://www.b-human.de/downloads/publications/2017/coderelease2017.pdf>.
- [19] Chi-Sheng Shih, Kun-Li Lin, Sun-An Chiang, Tzu-Hao Hu, Wei-Kang Fu, Chun Hu, Hao-Yu Chen, Ya-Yuan Cheng, and Tzu-Wei Chao. NTU RoboPAL Team Report 2015. Technical report, Department of Computer Science and Information Engineering National Taiwan University, 2015. <https://drive.google.com/file/d/0B1d3UWuPZfdyVj1KRU1iTvpqUGc/view>.
- [20] Pluess Silvan. Dynamic Analysis and Control of a Robotic Bipedal Locomotion System. Technical report, Swiss Federal Institute of Technology (ETH), 2014. <http://www.robocup.ethz.ch/link/1JMk9w>.
- [21] V. Suriani, D. Albani, A. Youssef, F. Del Duchetto, A. Nania, D.D. Bloisi, L. Iocchi, and D. Nardi. SPQR RoboCup 2016 Standard Platform League Team Description Paper. Technical report, Department of Computer, Control, and Management Engineering, Sapienza University of Rome, 2016. http://www.robocup2016.org/media/symposium/Team-Description-Papers/StandardPlatform/RoboCup_2016_SPL_TDP_SPQR.pdf.
- [22] UChileRoboticsTeam. Team Research Report 2016, 2016. <http://uchilert.amtc.cl/wp-content/uploads/2016/11/TeamResearchReport.pdf>.
- [23] Oliver Urbann and Stefan Tasse. Observer based biped walking control, a sensor fusion approach. *Autonomous Robots*, 35(1):37–49, 2013.

- [24] Douwe van der Wal, Pieter Kronemeijer, and Caitlin Lagrand. Dutch Nao Team - Technical Report. Technical report, University of Amsterdam, 2017. http://www.dutchnaoteam.nl/wp-content/uploads/2018/01/dnt_techreport_2017.pdf.
- [25] UT Austin Villa. 2017 Code Release. Code release at <https://github.com/LARG/utaustinvilla3d>, February 2018.
- [26] Rudi Villing, John McDonald, Simon O’Keeffe, and Louis Gallagher. RoboEireann Team Description RoboCup 2017 Standard Platform League. Technical report, Department of Electronic Engineering Department of Computer Science Maynooth University, 2017. https://www.robocup2017.org/file/symposium/soccer_std_plf/TeamDescriptionPaper2017.pdf.
- [27] Yanbin Wang, Chunxue Li, Shuai Ma, Xinkai Wang, Yimin Wei, Gaohuan Lv, and Fei Liu. WrightOcean Team Description for Robocup 2017. Technical report, Ludong University, 2017. https://www.robocup2017.org/file/symposium/soccer_std_plf/1806ddae79fc9c6585cb70326cb8fc3a.pdf.
- [28] Yuan Xu. *From Simulation to Reality: Migration of Humanoid Robot Control*. PhD thesis, Mathematisch-Naturwissenschaftlichen Fakultät II der Humboldt-Universität zu Berlin, 2014.