

## Sachbericht zum Verwendungsnachweis

### Teil II: Eingehende Darstellung

Vorhaben	<b>VeryHuman</b> <b>Lernen und Verifikation Komplexer Verhalten für Humanoide Roboter</b>
Förderkennzeichen	<b>01IW20004</b>
Zuwendungsempfänger	Deutsches Forschungszentrum für Künstliche Intelligenz GmbH Trippstadter Straße 122, D-67663 Kaiserslautern
Ausführende Stelle	Robotics Innovation Center & FB Cyber-Physical Systems Robert-Hooke-Straße 1, 28359 Bremen
Projektleiter	Prof. Dr. Rolf Drechsler Prof. Dr. Frank Kirchner
Bewilligungszeitraum	01.06.2020 – 31.05.2024
Autoren	Melya Boukheddimi, Christoph Lüth
Dokumentenkenung	DFKI-HB-Rep-VeryHuman-02
Dokumentenversion	1.0 (Final)
Erstellungsdatum	11. November 2024

Das diesem Bericht zugrunde liegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IW20004 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.



## II.1 Einleitung

Das vorliegende Dokument bildet den Abschlussbericht des Vorhabens VeryHuman (Lernen und Verifikation Komplexer Verhalten für Humanoide Roboter). VeryHuman wurde durch das Bundesministerium für Bildung und Forschung (BMBF) unter dem Förderkennzeichen 01IW20004 vom 01.06.2020 bis 31.05.2024 gefördert.

Die dargestellten Arbeiten umfassen alle vom DFKI geleisteten Arbeiten im Vorhaben VeryHuman. Abbildung II.1 auf Seite 4 gibt den Überblick über die Aufteilung der Arbeitspakete.

In dieser eingehenden Darstellung werden die im Rahmen von VeryHuman durchgeführten Arbeiten ausführlicher dargestellt.

## II.2 Verwendung der Zuwendung

Wir geben eine kurze Zusammenfassung der wesentlichen Ergebnisse des Projektes, gefolgt von einer ausführlichen Präsentation der Arbeiten in den vier Arbeitsbereichen.

### II.2.1 Zusammenfassung der wesentlichen Ergebnisse

Die wichtigsten Ergebnisse des Projektes lassen sich wie folgt zusammenfassen:

- Wir haben einen Lehrplan mit vereinfachten Systemen (einfaches Pendel, lineares umgekehrtes Pendel usw.) entworfen, um Unterschwingungen und die formale Verifikation in dynamischen Systemen zu untersuchen.
- Wir haben die Hardware- und Softwareplattform für ein Drehmomentbegrenzt einfaches Pendel als erstes kanonisches Beispiel für die Untersuchung der Unterschwingung im Rahmen des Projekts entwickelt.
- Wir haben die analytische innere Schätzung des Anziehungsbereichs des LQR-Reglers für ein Drehmomentbegrenzt einfaches Pendel abgeleitet.
- Wir haben das Laufen in der Simulation mit optimaler Steuerung erfolgreich erreicht.
- Wir haben erfolgreich grundlegende Balancier- und Hockerverhaltensweisen auf dem realen System getestet.
- Wir haben unsere Arbeit zur analytischen Schätzung des Anziehungsbereichs für die Verifizierung von LQR-Reglern in der IEEE Conference on Decision and Control (CDC) 2022 veröffentlicht.
- Wir haben eine experimentelle Überprüfung der Hocke und des einbeinigen Balancierens mit einem Motion-Capture-System durchgeführt und die Möglichkeit der Verwendung von Mocap für die Zustandsrückmeldung bei der Steuerung von Humanoiden demonstriert. Dieser Artikel wurde in der Zeitschrift MDPI Sensors veröffentlicht.
- Wir haben einen neuartigen hybriden numerischen und analytischen Ansatz zur Lösung der Kinematik und Dynamik von serienparallelen hybriden humanoiden Robotern entwickelt, der erhebliche Verbesserungen in der Simulation zeigt. Dieser Ansatz wurde auf der IROS 2022 präsentiert.
- Wir haben ein neues einbeiniges System entwickelt, um das Sprungverhalten zu untersuchen, und eine variable Höhensteuerung implementiert, die auf durchgängigem verstärkendem Lernen basiert.

Tabelle II.1: Übersicht über die Arbeitspakete in VeryHuman.

Arbeitsbereich A	<b>Allgemeine Methodik</b>	
AP A.1	Integriertes Lernen	3 PM
AP A.2	Terminologie	3 PM
AP A.3	Symbolische Belohnungsfunktion	3 PM
Arbeitsbereich B	<b>Symbolisches Modell des Gehverhaltens</b>	
AP B.1	Konzeptionelle Sprache und Formalismus	6 PM
AP B.2	Modellbildung durch Beobachtung des Gehverhaltens	7 PM
AP B.3	Symbolische Spezifikation des korrekten Verhaltens	9 PM
AP B.4	Formales Modell der Starrkörperdynamik	6 PM
AP B.5	Verifikation und Validierung	8 PM
AP B.6	Kontext	6 PM
Arbeitsbereich C	<b>Symbolisch erweitertes Verhaltenslernen</b>	
AP C.1	Teststand des RH5 Demonstratorsystems	7 PM
AP C.2	Modellbasierte optimale Regelung für humanoides Laufen	12 PM
AP C.3	Analyse des Demonstratorverhaltens	7 PM
AP C.4	Lernbasierte Regelung für humanoides Laufen	9 PM
AP C.5	Simulation der Starrkörperdynamik	10 PM

- Die Ergebnisse der Ableitung von Belohnungsfunktionen für die Politik des Verstärkungslernens, die auf der symbolischen Beschreibung eines einfachen zweibeinigen Modells für ein Gehverhalten basieren, wurden 2023 auf der 62. IEEE Conference on Decision and Control (CDC) veröffentlicht.
- Die Arbeit zum dynamischen und schnellen Gehen auf dem komplexen humanoiden Ganzkörperroboter RH5 wurde 2023 erfolgreich auf der IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids) veröffentlicht.
- Die Ergebnisse der Untersuchung zur vollen Ausschöpfung der Fähigkeiten eines serienparallelen hybriden Humanoiden durch Ganzkörper-Trajektorienoptimierung. Diese Studie über ein schnelles, dynamisches Heben eines 15 kg schweren Gewichts mit dem humanoiden Roboter RH5 wurde 2023 auf der IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) veröffentlicht.

Es folgt eine detailliertere Darstellung der Resultate der einzelnen Arbeitspakete.

## II.2.2 Arbeitsbereich A: Allgemeine Methodik

Das Ziel der Pakete in diesem Arbeitsbereiches war es, die allgemeine Methodik und das Fachwissen der beiden Forschungsgruppen RIC und CPS optimal zu kombinieren.

**Terminologie (AP A.2):** Um die Wissenslücken zwischen den beiden Projektpartnern zu schließen, wurde eine Reihe von neun internen Workshops organisiert, in denen ein gemeinsames Verständnis der Problemstellung sowie ein Überblick über bestehende Ansätze und gemeinsame Terminologie entwickelt worden ist. Darüber hinaus wurde ein Überblick über verschiedene Ansätze der Fortbewegung mit Beinen abgeleitet. Vereinfachte Bewegungsmodelle, welche die wesentliche Dynamik des menschlichen Gehens erfassen, sind in der Humanoidenforschung recht verbreitet. Diese Modelle beruhen auf der Lage

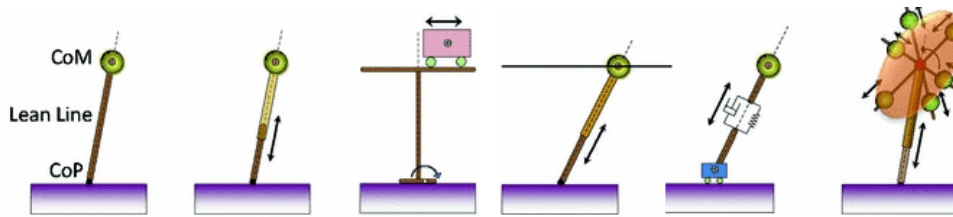


Abbildung II.1: Übersicht Bewegungsmodelle

des Massenschwerpunkts (CoM), des Druckzentrums (CoP) und der sie verbindenden Linie. Abb. II.1 zeigt sechs dieser häufig verwendeten Modelle. Sie sind Varianten der einfachen Pendeldynamik, die ein häufig untersuchte System ist.

**Begriffe der symbolischen Belohnungsfunktion (AP A.3):** Im Zusammenhang mit der zweibeinigen Robotik gibt es frühere Forschungen [18, 25] auf dem Gebiet der Biomechanik, die das Wesen des menschlichen Gehens erfassen. Darauf aufbauend haben wir eine symbolische Beschreibung des Gehens in Form einer orthantischen Sequenz im Zustandsraum verwendet. Diese wird weiter in den APs B.2 und C.4.3 beschrieben.

**Integriertes Lernen des symbolischen und subsymbolischen Ansatzes (AP A.1):** Sowohl die optimale Steuerung als auch das Verstärkungslernen beruhen auf einer symbolischen Kosten- oder Belohnungsfunktion, die bestimmte Bewegungen verbietet und andere ermutigt. Daher ist es notwendig, dass man in der Lage ist, Kandidaten für solche Funktionen zu generieren, vorzugsweise durch die Analyse aufgezeichneter menschlicher Bewegungen.

Wir haben jedoch festgestellt, dass die Analyse von Daten aus der Literatur ebenso fruchtbar sein kann; die oben erwähnte symbolische Beschreibung des Gehens auf der Basis von Orthanten wurde auf diese Weise abgeleitet.

## II.2.3 Arbeitsbereich B: Symbolisches Modell des Gehverhaltens

**Konzeptuelle Sprache und Formalismus (WP B.1):** Der Zustand eines dynamischen Systems ist als hochdimensionaler Vektor  $q$  gegeben, der aus allen Gelenkwinkeln und -dehnungen sowie ihren jeweiligen zeitlichen Ableitungen  $\dot{q}$  besteht. Der durch diese Vektoren aufgespannte Raum heißt Konfigurationsraum. Bei der genauen Konfiguration des Roboters, einschließlich Masse und Trägheitsparameter, die beschreibt die Differentialgleichung (ODE)

$$\dot{q} = f(q) \quad (\text{II.1})$$

Beschreiben der zeitlichen Entwicklung des Systems mit den Methoden der aus der klassischen Mechanik. Da sowohl RL als auch OC auf dieser mathematischen Beschreibung beruhen, muss jedes formale Modell, das in diesem Projekt entwickelt wird, auf diese Ebene übersetzt werden. Damit wird ein doppeltes Ziel verfolgt:

- Ersten können wir zunächst die Menge aller möglichen Konfigurationen auf eine Menge geeigneter Konfigurationen beschränken, was die explorative Techniken wie das Verstärkungslernen leichter anwendbar sind und zu besseren Belohnungsfunktionen führt;
- Zweitens können wir, indem wir auf einer abstrakteren Ebene argumentieren, Eigenschaften wie die Stabilität zeigen.

Aussage	Wirkung
$x := \theta$	diskrete Zuweisung (Zuweisung des aktuellen Wertes des Terms $\theta$ an Variable $x$ )
$x := *$	einer Variablen einen beliebigen reellen Wert zuweisen $x$
$\alpha; \beta$	Sequentielle Zusammensetzung
$\alpha \cup \beta$	nicht-deterministische Wahl
$\alpha^*$	nicht-deterministische Wiederholung (beliebig oft)
$?F$	Überprüfe Bedingung $F$ hält im aktuellen Zustand, bricht den Lauf ab wenn dies nicht der Fall ist
$(x'_i = \theta_i)_{i=1, \dots, n} \& Q$	Evolviert $x_i$ entlang der Differentialgleichungen für eine beliebige Zeitspanne, beschränkt auf den Bereich der maximalen Evolvierung $Q$

Tabelle II.2: Hybrid programs in dL.

Symbolische Beschreibungen wie “die Knie sind über den Füßen”, beziehen sich auf Beziehungen von diskreten Entitäten (Knie, Füße) in Zeit oder Raum, die formal durch räumliche oder zeitliche Berechnungen beschrieben werden [20]. Es gibt eine ganze Reihe von Ansätzen, wie die einfache Kreuz [26] oder Doppelkreuz [43] Kalküle, die versuchen, die räumliche Beziehung von Entitäten (nicht notwendigerweise Punkte) auf einer Ebene mit binären oder ternären Beziehungen zu axiomatisieren, aber diese Kalküle befassen sich eher mit Invarianz, Komplexität oder Entscheidbarkeit und nicht um die Beschreibung dynamischer Systeme; es bleibt eine ziemlich große Lücke zwischen der analytischen Beschreibung eines kontinuierlichen dynamischen Systems durch die Gleichung (II.1).

Wir können damit beginnen, die Lücke zu schließen, indem wir Eigenschaften über ihren Zustandsraum definieren. Formal gesehen sei  $Prop$  eine Menge von Aussagen, dann können wir jedem Zustand  $\mathbf{q}$  eine Menge von Eigenschaften zuordnen, die in diesem Zustand gelten Zustand gelten:

$$L : Q \rightarrow 2^{Prop}$$

wobei  $Q$  der Konfigurationsraum (Menge aller Systemzustände) ist. Für Wenn der Roboter beispielsweise zwei Füße und zwei Knie hat, die als Komponenten  $i, j$  und  $k, l$  des Konfigurationsraumes, dann ist “die Knie sind über den Füßen” ist definiert durch das Prädikat  $\mathbf{q}_i \geq \mathbf{q}_k \wedge \mathbf{q}_j \geq \mathbf{q}_l$ . Die Orthanten unten sind ein Beispiel für eine solche Beschreibung.

Diese einfache Formulierung erlaubt es uns jedoch nicht, das zeitliches Verhalten zu beschreiben, und darüber hinaus erlaubt sie keine diskreten Zustandsänderungen, die durch Ereignisse wie das Aufsetzen eines Fußes auf den Boden gegeben sind, und der Wechsel vom freien Schwingen zum Tragen des Gewichts. Hierfür benötigen wir die Theorie der hybriden Automaten [29, 15], die eine endliche Kontrolle mit kontinuierlichen Variablen kombinieren. Wir verwenden die moderne Formulierung von hybriden Automaten im Sinne von *dynamic logic* [42] (dL), da sie über eine Beweisunterstützung verfügt, nämlich den *Keymaera-X* Beweiser [27].

In der dynamischen Logik werden hybride Automaten als *hybride Programme* angegeben, wie in Tabelle II.2. Diese nehmen hier Rolle von Kripke-Strukturen ein; mit ihnen können wir Modelle unseres Systems definieren. Die logischen Eigenschaften werden in der *hybriden Logik*, definiert, die aus Ausdrücken besteht, die über reellwertigen Ausdrücken  $\theta$  wie folgt aufgebaut sind:

$$\phi ::= \theta_1 \sim \theta_2 \mid \neg \phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \phi_1 \rightarrow \phi_2 \mid \forall x. \phi \mid \exists x. \phi \mid [\alpha]\phi \mid \langle \alpha \rangle \phi \quad (\text{II.2})$$

wobei  $\alpha$  ein hybrides Programm ist,  $\theta_i$  reellwertige arithmetische Ausdrücke sind,  $\sim \in \{=, \neq, \geq, >, <, \leq\}$ , und die Variablen sich über  $\mathbb{R}$ . Jedes hybride Programm  $\alpha$  ist mit modalen Operatoren  $\langle \alpha \rangle$  und  $[\alpha]$ , verbunden, wobei  $\langle \alpha \rangle \phi$  gilt, wenn  $\phi$  in einigen (mindestens einem) Zustand nach der Ausführung von  $\phi$

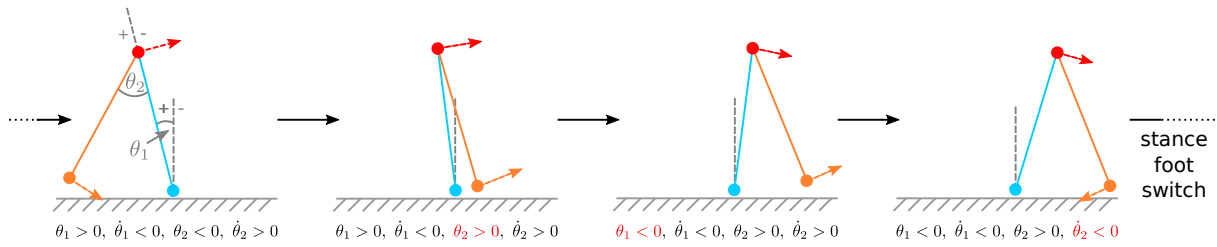


Abbildung II.2: Untermenge der Orthanten und ihrer Abfolge für den Gangzyklus des Compass-Walkers.

gilt, und  $[\alpha]\phi$  gilt, wenn  $\phi$  in allen Zuständen nach der Ausführung von  $\phi$  gilt (man beachte, dass hybride Programme nichtdeterministisch sind und daher eine beliebige Anzahl von Zuständen erreichen können). Für die vollständige Definition der Relation  $\nu \models \phi$ , siehe [40, 41].

Der Keymaera-X Beweiser nimmt als Eingabe eine Formel der hybriden Logik, die das zu verifizierende(n) Programm(e) über modale Operatoren wie oben beschrieben enthält, und versucht zu bestimmen, ob die Formel gilt oder nicht, indem er den Zustandsraum erkundet und einfache Beweisverpflichtungen gegenüber anderen Beweisen (wie Z3 oder Mathematica) ausführt. Dies ist ein halbautomatischer Prozess, der insbesondere bei Schleifen ( $\alpha^*$ ) die Unterstützung des Benutzers durch annotierte Invarianten erfordert. Bei einem Modell des Systems als hybrides Programm kann der Keymaera-X genutzt werden, um Eigenschaften zu zeigen, die in hybrider Logik ausgedrückt werden können. Nehmen wir zum Beispiel ein System  $\Theta$  mit dem Zustandsraum  $Q$  und dem Prädikat  $safe \subset Q$ , welches ausdrückt, dass sich das System in einem sicheren Zustand befindet, dann wird die Behauptung, dass das System letztendlich immer in einem sicheren Zustand endet; dieses wird als  $[\Theta]safe$  formuliert. Wir werden weitere Beispiele dafür in WP B.5 zeigen.

**Ableitbares, beobachtbares Modell aus simuliertem Verhalten (WP B.2):** Um symbolische Definitionen des Gehverhaltens mit der kontinuierlichen Beschreibung solcher Systeme zu verbinden, haben wir Modelle des Gehens aus der medizinischen Literatur betrachtet. Perry [39] stellt fest, dass der Gangzyklus in verschiedene Phasen unterteilt werden kann, wobei z. B. wechseln die Beine zwischen Stand (Bein in Kontakt mit dem Boden) und Schwung (Bein in freier Bewegung), und gibt Charakterisierungen des Gehverhaltens in solchen Begriffen mit zunehmender Detaillierung. Indem wir die verschiedenen Phasen des Gangzyklus symbolisch beschreiben, können wir eine abstrakte Charakterisierung des Gehens vornehmen. Als Testumgebung für unseren Ansatz haben wir den Compass-Walker verwendet. Dieses abstrakte System ermöglicht es, Strategien zu entwickeln und zu testen, die später auf komplexere Systeme übertragen werden können. Bei der Beobachtung der Gangart eines passiven Compass-Walkers, der durch die Schwerkraft einen Abhang hinuntergetrieben wird, stellt man fest, dass der Phasenraum  $(\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2)$  in Orthanten (oder Hyperquadranten) unterteilt werden kann, die in einer festen Reihenfolge besucht werden müssen. Bei den Orthanten ist die Folge

$$\begin{aligned}
 (\theta_1 > 0, \dot{\theta}_1 < 0, \theta_2 < 0, \dot{\theta}_2 > 0) &\rightarrow (\theta_1 > 0, \dot{\theta}_1 < 0, \theta_2 > 0, \dot{\theta}_2 > 0) \\
 &\rightarrow (\theta_1 < 0, \dot{\theta}_1 < 0, \theta_2 > 0, \dot{\theta}_2 > 0) \\
 &\rightarrow (\theta_1 < 0, \dot{\theta}_1 < 0, \theta_2 > 0, \dot{\theta}_2 < 0) \\
 &\rightarrow (\theta_1 > 0, \dot{\theta}_1 < 0, \theta_2 < 0, \dot{\theta}_2 > 0)
 \end{aligned}$$

die strikt eingehalten wird. Diese Abfolge ist in Abb. II.2 für das Gehen auf einer ebenen Fläche und Verwendung eines virtuellen Schwerkraftreglers, der das Gefälle des Bodens simuliert, dargestellt.

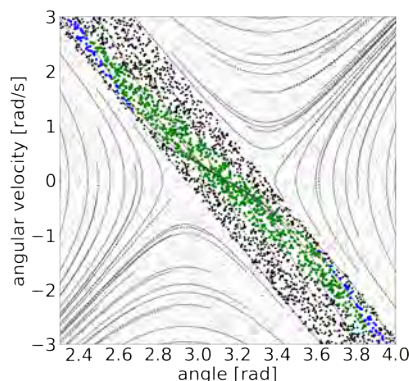


Abbildung II.3: Anziehungsbereich mit Phasenbild des Systems. Verwendete Parameter:  $g = 9.81m/s^2$ ,  $m = 0.5kg$ ,  $l = 0.5m$ ,  $B = 0$ ,  $u_{\max} = 0.5Nm$ ,  $Q = id$ ,  $R = 1$ ,  $N = 0$

**Symbolische Spezifikation des korrekten Verhaltens (WP B.3):** Die Ergebnisse der optimalen Steuerung und des inversen Verstärkungslernens können von vorberechneten Trajektorien, denen das System unter Verwendung zusätzlicher stabilisierender Regler folgt, bis hin zu vollwertigen Reglern reichen. In jedem Fall wollen wir sicherstellen, dass diese Ergebnisse zum richtigen Verhalten des jeweiligen Systems führen. Dazu gehört der Nachweis, dass ein bestimmter Regler eine bestimmte Trajektorie tatsächlich stabilisiert, sowie die Angabe von Konfigurationen, bei denen ein bestimmter Regler versagen könnte. Dies ist besonders wichtig bei untermotorisierten Systemen. Per Definition kann ein voll aktiver Roboter jeden Punkt in seinem Konfigurationsraum einnehmen und dort zum Stillstand kommen, während ein System wie ein zweibeiniger Roboter sehr wohl das Gleichgewicht verlieren und umkippen kann. Ein sehr nützliches Instrument, um zu spezifizieren, was Stabilität genau bedeutet, ist die Lyapunov-Analyse. Die Grundidee dieser Analyse ist die folgende: Wenn wir für ein dynamisches System  $\dot{q} = f(q)$  mit  $f$  kontinuierlich eine kontinuierlich differenzierbare Skalarfunktion  $V$  finden können, die am Ursprung verschwindet und für alle  $q$  in einer Region um den Ursprung herum folgende Bedingungen erfüllt

$$V(q) > 0 \text{ and } \dot{V}(q) \leq 0, \quad (\text{II.3})$$

dann ist der Ursprung ein stabiler Punkt des Systems. Wenn die Ungleichung in der zweiten Bedingung streng ist, konvergiert ein System, das innerhalb der angegebenen Region startet, zum Ursprung. Dieser Bereich wird als Anziehungsbereich bezeichnet. Dieser Begriff der Stabilität lässt sich auf natürliche Weise auf globale Aussagen über die Stabilität ausdehnen und kann auch zur Überprüfung der Stabilität ganzer Trajektorien anstelle einzelner Punkte verwendet werden. Als ersten Versuch, Probleme dieser Art anzugehen, haben wir uns ein drehmomentbegrenztes einfaches Pendel angesehen. Ein solches Pendel kann in die aufrechte Position gebracht und dort mit einer Kombination aus einem Energieformregler und einem LQR-Regler stabilisiert werden. Ersterer bringt das Pendel in die Nähe der aufrechten Position, während letzterer in der Lage ist, es um diese Position herum zu stabilisieren, wenn es nahe genug ist. Die genaue Festlegung, wann das Pendel „nahe genug“ ist, ist jedoch nicht ganz trivial.

Durch die Anwendung von Methoden der Lyapunov-Analyse auf die numerisch abgeleiteten Lösungen des zugrundeliegenden LQR-Problems können wir zeigen, dass der Anziehungsbereich der LQR-Regler im Prinzip recht groß ist, wenn man die Drehmomentbegrenzungen außer Acht lässt. Um speziell die Drehmomentbegrenzungen zu berücksichtigen, haben wir analytische Lösungen für die dem LQR-Problem entsprechende linearisierte Systemdynamik untersucht. Durch Lösen des ODE-Systems konnten wir eine analytisch ausgedrückte hinreichende Bedingung für die Zugehörigkeit einer bestimmten Konfiguration zum Anziehungsbereich finden.

Um unsere Ergebnisse zu validieren, haben wir die obige Bedingung für 10000 zufällig ausgewählte

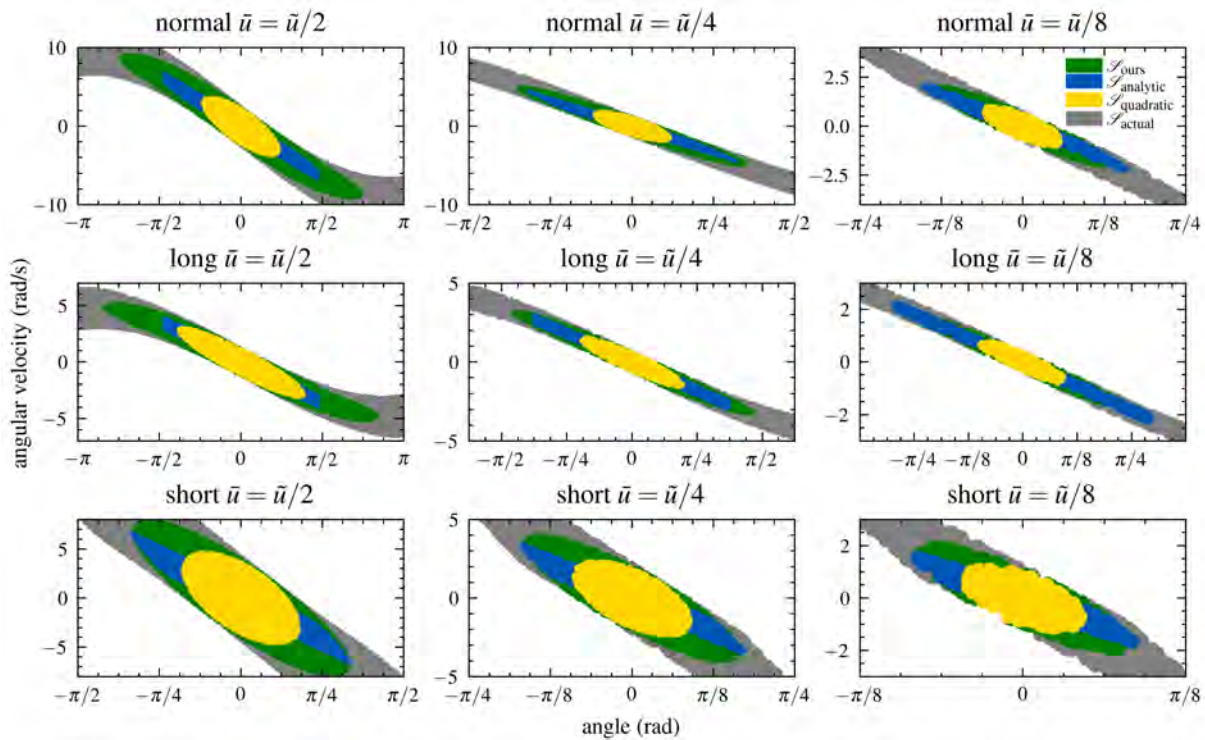


Abbildung II.4: Vergleich der verbesserten ROA-Schätzung mit anderen Methoden.

Zustände bewertet, während wir gleichzeitig die numerische Simulation durchführten und die Drehmomente überwachten. Das Ergebnis ist in Abb. II.3 dargestellt. Der grüne Bereich erfüllt die abgeleiteten Kriterien. Kleine schwarze Punkte stellen Anfangszustände dar, die der Regler nicht stabilisieren konnte (numerisch überprüft). Große schwarze Punkte können durch den LQR-Regler stabilisiert werden, verletzen aber die Drehmomentgrenzen (numerisch geprüft). Blaue Punkte sind stabilisiert und verletzen numerisch die Grenze nicht, konnten aber durch unsere Bedingung nicht gefunden werden. Wie erwartet, erfüllte keine der untersuchten Anfangsbedingungen die abgeleitete Bedingung, verletzte aber die Drehmomentgrenze oder konnte vom LQR-Regler nicht stabilisiert werden. Die gestrichelten Linien zeigen das Phasenbild des Systems. Die geringe Anzahl blauer Punkte zeigt an, dass unsere Bedingung eine gute innere Annäherung an die tatsächliche Region of Attraction beschreibt. Neben der numerischen Validierung führen wir derzeit auch Experimente an einem physikalischen System durch. Die Experimente sind inzwischen abgeschlossen, und die Ergebnisse wurden veröffentlicht [6].

In Folgearbeiten haben wir den obigen Ansatz zu einer Lyapunov-basierten ROA-Schätzmethode verfeinert, die speziell für linear-quadratische Reglersysteme (LQR) entwickelt wurde. Die vorgeschlagene Methode integriert verschiedene Energiefunktionen und reglerspezifische quadratische Funktionen, die dann mit einem auf gezielten Stichproben basierendem ROA-Berechnungsprozess kombiniert werden, um den ROA für umgekehrte Pendel mit verschiedenen Parametern zu schätzen. Wir demonstrieren die Wirksamkeit unserer Methode anhand des LQR-gesteuerten, drehmomentbegrenzten invertierten Pendels. Die experimentellen Ergebnisse zeigen die Praxistauglichkeit und Effektivität der Methode in realen Szenarien, die eine schnelle und genaue ROA-Schätzung ermöglicht und damit die Sicherheit und Stabilität des Systems gewährleistet. Wir haben unsere Methode mit anderen Ansätzen verglichen, und die experimentellen Ergebnisse zeigen, dass unsere Methode in Bezug auf den ROA-Bereich im Allgemeinen überlegen ist und die genauesten Schätzungen liefert. **Abbildung II.4** vergleicht das Ergebnis der neuen ROA-Schätzung mit früheren Methoden, einschließlich des oben beschriebenen analytischen An-

satzes. Die Abbildung zeigt die ROA-Schätzung für das umgekehrte Pendel, wobei der gelbe Bereich den traditionellen quadratischen Ansatz, der blaue den analytischen Ansatz von oben, der grüne den verbesserten Ansatz und der graue die tatsächliche ROA darstellt. (Die neun Szenarien sind die Kombination von drei verschiedenen Pendelkonfigurationen — normal, lang, kurz — der Masse, Die Diagramme sind Phasenraumdiagramme, wobei der Winkel  $\theta$  auf der X-Achse und die Winkelgeschwindigkeit  $\theta'$  auf der Y-Achse liegen. Jedes Diagramm zeigt das Ergebnis von 100.000 Simulationen mit zufällig gewählten Anfangsparametern). Wie man sieht, liefert der grüne Bereich die genaueste Schätzung, außer für die lange Konfiguration mit hoher Drehmomentgrenze. Weitere Einzelheiten sind in [3] zu finden.

**Formalisiertes Modell der Rigid Body Dynamics (WP B.4):** Ein wesentlicher Bestandteil für ein effizientes Training von RL oder den Entwurf von OC-Controllern ist eine genaue Beschreibung der Systemdynamik in Form einer ODE (Modellidentifikation) sowie analytische Gradienten einiger kinematischer und dynamischer Größen. Einige grundlegende Arbeiten zu den analytischen Zeitableitungen der Equations of Motions (EOM) wurden hier unter Ausnutzung der Schraubentheorie und Lie-Gruppen-Methoden in der Mehrkörperdynamik durchgeführt [9, 10]. Wie oben angedeutet, sind die Masse- und Trägheitsparameter ein integraler Bestandteil dieser Arbeit. Wir arbeiten derzeit daran, Konzepte der geometrischen dynamischen Parameteridentifikation [34] auf unsere einfachen Systeme anzuwenden.

**Verifizierung & Validierung (WP B.5):** In der bisherigen Forschung gibt es eine Reihe von Ansätzen, die mit unserer Forschung in Verbindung stehen. Sie reichen von Verifikationsrahmen, die Lyapunov-Kandidaten generieren und validieren [19], bis hin zur Validierung von Lösungen für Differentialgleichungen [30]. Wir haben uns für die Verwendung des Werkzeuges Keymaera-X [27] entschieden, das ein spezielles Beweisverfahren für hybride Programme bietet. Wir geben hier ein einfaches Beispiel, um zu veranschaulichen, wie die Modellierung der Stabilität eines einfachen Pendels bewiesen werden kann. Betrachten wir die folgende Aussage über das einfache Pendel: *“Für alle möglichen Anfangswinkel wird ein einfaches, nicht angetriebenes, lineares Pendel innerhalb einer endlichen Zeitspanne immer in die stabile Position nach unten.”*. Dies kann in dynamischer Logik wie folgt ausgedrückt werden:

$$\forall \theta_{start}. \exists T. (g > 0 \wedge m > 0 \wedge l > 0 \wedge 0 < b < 1) \rightarrow \left[ \begin{array}{l} \theta := \theta_{start}; \theta' := 0; (t' = 1, \theta'' = \frac{-m \cdot g \cdot l \cdot \sin(\theta) - b \cdot \theta'}{ml^2}) \\ (\forall t. t > T \implies \theta = 0) \end{array} \right] \quad (\text{II.4})$$

Diese Formel hat die Struktur  $pre \rightarrow [\alpha] \phi$ , die ausdrückt, dass bei gegebener Vorbedingung  $pre$  und nach allen möglichen Durchläufen des hybriden Programms  $\alpha$  die Aussage  $\phi$  gilt. Hier ist  $\phi \equiv (\forall t, t > T \implies \theta = 0)$  und besagt, dass das Pendel ab dem Zeitpunkt  $T$  senkrecht ist ( $\theta = 0$ ). Die Voraussetzungen  $g > 0 \wedge m > 0 \wedge l > 0 \wedge 0 < b < 1$  besagen, dass die Schwerkraft  $g$ , die Pendelmasse,  $m$  und die Länge  $l$  positive reelle Werte sind und dass der Reibungsfaktor  $b$  ein positiver reeller Wert kleiner als 1 ist. Das hybride Programm  $\alpha$  besteht aus drei Anweisungen: die ersten beiden Zuweisungen  $\theta := \theta_{start}; \theta' := 0$  von Anfangswerten für die Variable  $\theta$  und ihre Ableitung; und der dritte Aussage  $t' = 1, \theta'' = \frac{-m \cdot g \cdot l \cdot \sin(\theta) - b \cdot \theta'}{ml^2}$  die das Verhalten im zeitlichen Verlauf ausdrückt. Hier drückt,  $t' = 1$  drückt den linearen Ablauf der Zeit aus und die zweite Gleichung ist die Bewegungsgleichung (EOM) des freien einfachen linearen Pendels.

Wir haben die Korrektheit der in [6] analytisch ermittelten ROA mit Hilfe von Keymaera-X bewiesen. Der ROA wird als maximaler Drehmomentgrenzwert ausgedrückt, den ein LQR-Regler auf die Achse des Pendels anwenden kann. Im Vergleich zum obigen Beweis umfassen die Vorbedingungen (neben den oben genannten) Bedingungen für die Anfangswerte  $\theta$  und  $\theta'$ . Die Nachbedingung ist, dass  $\theta$  im Bereich der Anziehung bleibt (wo der LQR-Regler das Pendel in der Nähe der Aufwärts-Rechts-Position

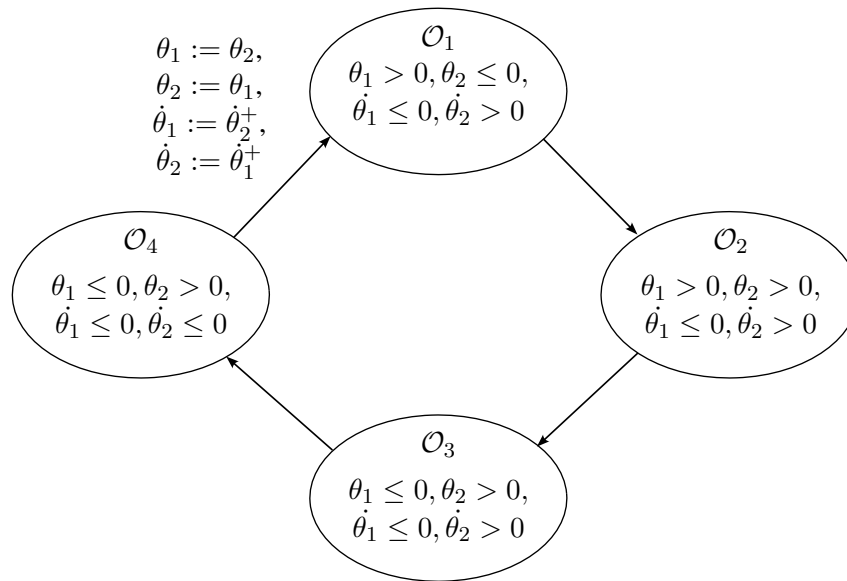


Abbildung II.5: Hybrider Automat zur Modellierung der Orthant-Sequenzen.

stabil halten kann). Der EOM enthält einen zusätzlichen Term, der das vom LQR-Regler aufgebrachte Drehmoment sowie weitere Faktoren wie Reibung enthält.

Wir haben den Orthanten-Ansatz aus WP B.2 als hybriden Automaten formalisiert, wobei die Zustände des Automaten den verschiedenen Orthanten entsprechen, wie sie durch die Parameter des Walkers definiert und in **Abbildung II.5** visualisiert sind. Der daraus resultierende hybride Automat reduziert den Systemzustandsraum und ermöglicht ein effizienteres Reinforcement Learning (siehe WP C.4.3 on page 27). Dieser Ansatz zur Kombination von Lernen und symbolischem Denken kann als Blaupause für die Generierung von Belohnungsfunktionen dienen; für weitere Details siehe die daraus resultierende Veröffentlichung [7].

## II.2.4 Arbeitsbereich C: Symbolisch verstärkte Verhaltenskontrolle

Dieses Arbeitspaket befasst sich mit dem symbolisch unterstützten Lernen und der Steuerung komplexer Verhaltensweisen einer humanoiden Plattform, indem die Parallelen zwischen verstärkendem Lernen und optimaler Steuerung genutzt werden. Darüber wurde ein Motorprüfstand und eine Starrkörperdynamiksimulation des komplexen humanoiden Roboters entwickelt.

**Prüfstand des RH5-Demonstratorsystems (AP C.1):** Dieses Arbeitspaket befasst sich mit der Funktionserhaltung des RH5-Demonstratorsystems, der Entwicklung eines neuen Motorprüfstandes und der Integration des humanoiden Roboters auf einem Portalkransystem für Gehversuche.

*Funktionserhalt/Reparatur des RH5 (AP C.1.1):* Im Rahmen der Roboterwartung wurden verschiedene kleinere Arbeiten am Roboter durchgeführt. Dazu gehörten die Wartung von Spindelgetrieben, der Austausch und die Rekalibrierung von Sensoren und die Kalibrierung von Gelenkwinkeln. Darüber hinaus wurden aufgrund von Problemen mit den Anschlussflanschen der Hüftantriebe überarbeitete Antriebseinheiten entwickelt. Vier neue Antriebseinheiten wurden konstruiert und im System in den Hüftgelenken verbaut. Die Verbesserungen betreffen vor allem die Verwendung eines ICMU-Sensors zur Kommutierung des Antriebs als Ersatz für die bisher verwendeten Hall-Sensoren. Außerdem wurde der Abtriebsflansch des Antriebs so verändert, dass die Abtriebswelle nun aus Stahl besteht. Zuvor gab es

mechanische Probleme mit der Verschraubung im Aluminiumflansch, so dass das erforderliche Spitzendrehmoment nicht sicher übertragen werden konnte. Die neuen Antriebe wurden erfolgreich in der Anlage in Betrieb genommen.

Um die Fähigkeiten des humanoiden Roboters RH5 zu verbessern, wurden mehrere Änderungen vorgenommen:

- Es wurden neue Spindeln für die Beinlinearantriebe bestellt, um die Geschwindigkeit des Hüft- und Kniegelenks zu erhöhen. Die Spindeln haben jetzt eine größere Teilung von 10 mm/Umdrehung im Vergleich zu den bisher verwendeten Teilungen von 4 mm und 5 mm.
- Alternative Antriebseinheiten für den Beintrieb wurden als zusätzliche Lösung für die Erprobung höherer dynamischer Laufmuster bestellt.
- Es wurden Adapterflansche und neue Konstruktionsteile entworfen, um die alternativen Antriebseinheiten zu integrieren.
- Die Rechenleistung wurde durch den Austausch von GPU-Einheiten erhöht.
- Sensoren, z. B. IMU/GPS, Stereo-Vision-Sensoren und Motorpositionssensoren, wurden auf den neuesten Stand der Technik gebracht.
- Zusätzliche Batterien wurden bestellt und in austauschbare Rucksäcke integriert, um die erneute Platzierung der Batterien und Langzeittests zu erleichtern.
- Die Motorsteuerungen wurden als Ersatzteile bestellt, um die weitere Betriebsfähigkeit des Roboters zu gewährleisten.

#### *Motorprüfstand(AP C.1.2):*

Um die Roboter mit geeigneten Aktoren auszustatten, wurde ein Motorprüfstand geplant. Ein erstes Angebot überstieg die verfügbaren Mittel, so dass eine Eigenentwicklung vorgenommen wurde. Unser Konzept sieht ein Prüfgestell für Antriebe mit kleiner Übersetzung mit einer maximalen Geschwindigkeit von 40 rad/s und einem maximalen Drehmoment von 50 Nm und eine zweite Prüfstrecke für Antriebe mit großer Übersetzung mit einer Geschwindigkeit von 10 rad/s und 300 Nm Drehmoment vor.

Im aktuellen Berichtszeitraum wurde der Motorprüfstand fertiggestellt. Die Sensoren und Messtechnik wurden in Betrieb genommen und in die Software TwinCat 3.1 integriert. Es wurde eine einfache grafische Benutzeroberfläche zur Inbetriebnahme der Belastungsmaschine und zur Anzeige der Sensorwerte implementiert. Bild II.6a zeigt den Prüfstand mit den beiden Prüfstrecken, wobei in der 50-Nm-Strecke ein Probekörper montiert ist. Der Prüfling wurde mit einem Wellenausrichtgerät vermessen, wobei ein horizontaler Versatz von 0,2 mm und ein vertikaler Versatz von 0,02 mm gemessen wurden. Daraus ergibt sich eine Versatzbelastung von ca. 100 N in horizontaler Richtung während der Messung. Abbildung II.6b zeigt die Benutzeroberfläche. Hier sind die möglichen Regelungsarten (Drehzahlregelung und Drehmomentregelung) sowie das gemessene Drehmoment dargestellt.

Erste Ergebnisse des Motorprüfstandes sind in Abbildung II.7. dokumentiert. Hier ist das gemessene Drehmoment zwischen Prüfling und mechanischer Last aufgetragen. Als Prüfling dient der am DFKI RIC weit verbreitete *Robodrive* 50x08 BLDC-Antrieb in Kombination mit einem *HarmonicDrive*-Getriebe CPL-14 mit einem Übersetzungsverhältnis von  $i = 1 : 50$ . Als Motortreiber wird das im Haus hergestellte Elektronikpaket LIMES-BLDC eingesetzt. Ein Sensor misst den Eingangsstrom des Treibers. Der Prüfling wurde über eine Geschwindigkeits-Strom-Regelkaskade mit einer Geschwindigkeit von 100



(a) Prüfstand mit Prüfling in der 50-Nm-Prüfstrecke.

(b) GUI für die Inbetriebnahme des Motorprüfstandes.

Abbildung II.6: Motorprüfstand.

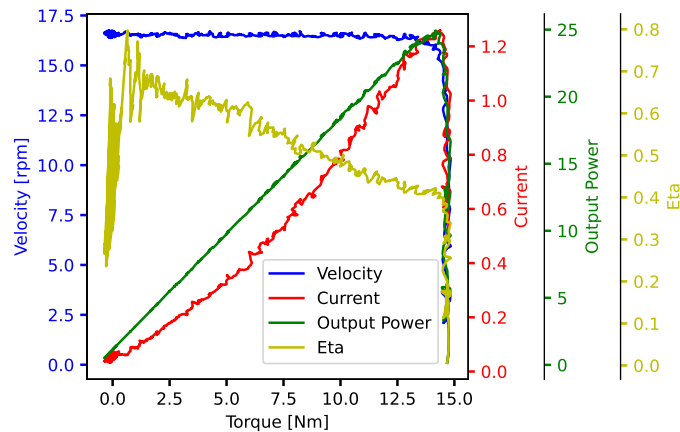


Abbildung II.7: Erste Ergebnisse vom Motorprüfstand.

deg/s gesteuert. Die mechanische Belastung wurde über eine Drehmomentrampe mit einem maximalen Drehmoment von 16 Nm in 5 s und einer Haltedrehmomentzeit von 1 s. gesteuert. Im Diagramm ist zu erkennen, dass der maximale Wirkungsgrad aufgrund der internen Getriebeeigenschaften bereits bei 2 Nm erreicht wird. Die Ausgangsleistung steigt analog zum Drehmoment an, während bei 15 Nm der Prüfling stehen bleibt.

*Integration des Systems auf einem Portalkran (AP C.1.3):* Um flexiblere Gehversuche durchführen zu können, wurde ein Portalkran aufgebaut und das RH5-System an dem Kran montiert. Es kann mit einem elektrisch betriebenen Kettzug angehoben werden. Für Gehversuche wird der Roboter auf den Boden gestellt und die Kette leicht entlastet, so dass das System auf seinen eigenen Beinen stehen und gehen kann, aber Stürze und Beschädigungen vermieden werden. Abb. II.8 zeigt den Roboter und den Kran.

**Modellbasierte optimale Steuerung des zweibeinigen Gehens (AP C.2):** Ein System mit zeitdiskreter Dynamik kann modelliert werden als

$$\mathbf{x}_{i+1} = \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i), \quad (\text{II.5})$$

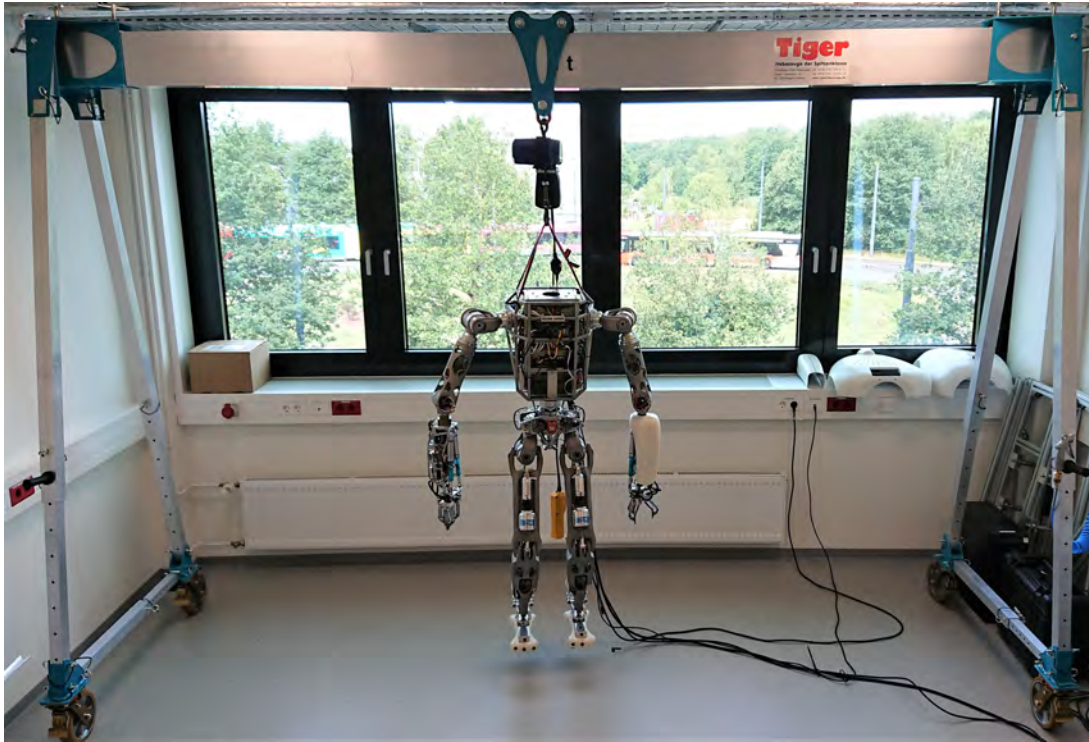


Abbildung II.8: RH5 montiert auf einem Portalkran: neue Hardware-Infrastruktur für den Test des RH5-Humanoiden

wobei  $\mathbf{f}$  die Entwicklung des Zustands  $\mathbf{x} \in \mathbf{R}^n$  von der Zeit  $i$  bis  $i + 1$  bei gegebener Steuerung  $\mathbf{u} \in \mathbf{R}^m$  ist. Die *Gesamtkosten*  $J$  einer Trajektorie können als Summe der laufenden Kosten  $\ell$  und der Endkosten  $\ell_f$  ausgehend vom Anfangszustand  $\mathbf{x}_0$  aund der Anwendung der Steuerungssequenz  $\mathbf{u}$  entlang des endlichen Zeithorizonts geschrieben werden:

$$J(\mathbf{x}_0, \mathbf{u}) = \ell_f(\mathbf{x}_N) + \sum_{i=0}^{N-1} \ell(\mathbf{x}_i, \mathbf{u}_i). \quad (\text{II.6})$$

Die Kosten  $\ell$  zu einem diskreten Zeitpunkt (d. h. einem Knoten) der Optimierung hängen von der zugewiesenen Gewichtung  $\alpha_c$  und der entsprechend dem Kostenterm  $\Phi_c$  as  $\ell = \sum_{c=1}^C \alpha_c \Phi_c(\mathbf{x}, \mathbf{u})$ . Daher schreiben wir das generische Problem der optimalen Steuerung als

$$\begin{aligned} \mathbf{X}^*, \mathbf{U}^* = \arg \min_{\mathbf{X}, \mathbf{U}} \ell_N(x_N) + \sum_{k=0}^{N-1} \int_{t_k}^{t_k + \Delta t} \ell_k(\mathbf{x}, \mathbf{u}) dt, \\ \text{s.t.} \quad \underline{\mathbf{u}} \leq \mathbf{u} \leq \bar{\mathbf{u}}, \quad \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \end{aligned} \quad (\text{II.7})$$

wobei eine vollständige Trajektorie  $\mathbf{X}$ ,  $\mathbf{U}$  eine Folge von Zuständen  $\mathbf{X} = \{\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_N\}$  und Steuereingängen  $\mathbf{U} = \{\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_{N-1}\}$  ist, welche Gleichung ((II.5)) und die Systemdynamik erfüllen, und  $\underline{\mathbf{u}}$  und  $\bar{\mathbf{u}}$  sind die untere bzw. obere Drehmomentgrenze des Systems. Zur Lösung des Trajektorienoptimierungsproblems von Gleichung (II.7), verwenden wir den Box-FDDP-Algorithmus [37], der in der Open-Source-Bibliothek CROCODDYL [36] öffentlich verfügbar ist. Der Box-FDDP-Algorithmus kann dank seines direkt-indirekten Hybridisierungsansatzes hochdynamische Bewegungen berechnen.

*Optimierungsbeschränkungen:* Wir berücksichtigen Nebenbedingungen des Trajektorienoptimierungsproblems über eine Kostenbestrafung in Gleichung (II.7). Kostenterme können entweder Gleichheits-

oder Ungleichheitsrestriktionen enthalten, die im Folgenden beschrieben werden. Im Falle von Gleichheitsbeschränkungen kann eine beliebige Aufgabe als quadratischer Regulatorterm formuliert werden als

$$\Phi_c = \| \mathbf{f}(t) - \mathbf{f}^{\text{ref}}(t) \|_2^2, \quad (\text{II.8})$$

wobei  $\mathbf{f}(t)$  und  $\mathbf{f}^{\text{ref}}$  aktuelle bzw. Referenzmerkmale sind. Der DDP-Algorithmus nutzt die Ableitungen dieser Reglerfunktionen, d. h. er berechnet die Jacobianer und Hessianer der Kostenfunktionen. Wir verwenden Gleichheitsbeschränkungen für das CoM-Tracking ( $\Phi_{\text{CoM}}$ ) und das Tracking der linken und rechten Fußposition ( $\Phi_{\text{foot}}$ ).

Ebenso wichtig für eine physikalisch konsistente Trajektorienoptimierung ist die Berücksichtigung von Grenzen, wie z. B. Robotergerenzen und Stabilitätsbeschränkungen. Diese Ungleichheitsbedingungen können ebenfalls als Bestrafungsterm einbezogen werden. Zu diesem Zweck verwenden wir einen begrenzten quadratischen Term als

$$\Phi_c = \begin{cases} \frac{1}{2} \mathbf{r}^T \mathbf{r} & | \underline{\mathbf{r}} > \mathbf{r} > \bar{\mathbf{r}} \\ 0 & | \underline{\mathbf{r}} \leq \mathbf{r} \leq \bar{\mathbf{r}}, \end{cases} \quad (\text{II.9})$$

wobei  $\mathbf{r}$  ein Restvektor ähnlich wie in (II.8) and  $\underline{\mathbf{r}}$  und  $\bar{\mathbf{r}}$  die untere bzw. obere Grenze darstellen. Im Rahmen unserer Arbeit definieren wir Ungleichheitsrestriktionen für die Positions- und Geschwindigkeitsgrenzen der Gelenke ( $\Phi_{\text{joints}}$ ), Reibungskegelrestriktionen ( $\Phi_{\text{friction}}$ ) und den Druckmittelpunkt ( $\Phi_{\text{CoP}}$ ).

Zusätzlich zu den beschriebenen Einschränkungen für die Aufgaben und die physikalische Konsistenz, optimieren wir für die Minimierung der Drehmomentes ( $\Phi_{\text{torques}}$ ) und regulieren die Roboterhaltung ( $\Phi_{\text{posture}}$ ).

*Stand der Technik bei den Methoden der optimalen Steuerung (AP C.2.1):* Die Trajektorienoptimierung (TO) ist ein numerisches Optimierungsverfahren, das darauf abzielt, eine Zustandssteuerungssequenz zu finden, die eine Kostenfunktion lokal minimiert und eine Reihe von Einschränkungen erfüllt. Die TO, die auf einer reduzierten Schwerpunktsdynamik [14] basiert, ist zu einem beliebigen Ansatz geworden. Die Verfolgung von Schwerpunktbewegungen erfordert jedoch eine sofortige Rückkopplungssteuerung, bei der typischerweise quadratische Programme mit Aufgabenraumdynamik gelöst werden [21]. Während TO auf der Grundlage reduzierter Dynamikmodelle großartige experimentelle Ergebnisse gezeigt hat [23], hat sich gezeigt, dass Ganzkörper-TO effizientere Bewegungen mit geringeren Kräften und Stößen erzeugt [16]. Zu diesem Zweck konzentrieren wir uns auf eine machbarkeitsgesteuerte DDP-Variante [38], die Box-FDDP [37] genannt wird, um dynamische Ganzkörperbewegungen effizient zu berechnen. Sie ist in der Open-Source-Bibliothek CROCODDYL [36] öffentlich verfügbar. Die mit diesen Solvern erzeugten Trajektorien erfordern jedoch oft einen zusätzlichen stabilisierenden Regler, um das Verhalten in einem anderen Simulator oder dem realen Roboter zu reproduzieren [28].

*Integration der ausgewählten optimalen Kontrollmethode in HyRoDyn (AP C.2.2):* Ein Hauptziel bei der Trajektorienoptimierung für beingebundene Systeme ist es, eine ausgeglichene Bewegung zu gewährleisten, die verhindert, dass der Roboter rutscht und herunterfällt. Um diese Belastung durch einen zusätzlichen stabilisierenden Regler zu reduzieren, stellen wir die Stabilität des Roboters sicher durch Anwendung des Konzepts des Kontaktschlüsselkegels [17] anstelle des weithin akzeptierten Nullmomentenkriteriums [44] in der Formulierung unserer Ganzkörper-Trajektorienoptimierung. Man beachte, dass die letztgenannte Methode aufgrund der Annahmen ausreichend hoher Reibung und der Existenz einer planaren Kontaktfläche eingeschränkt ist; stattdessen eignet sich die erstgenannte Methode auch für die OC mit mehreren Kontakten. Zu diesem Zweck modellieren wir 6D-Oberflächenkontakte in der OC-Formulierung von Gleichung (II.7) mit speziellen Ungleichheitsbedingungen für die Einseitigkeit

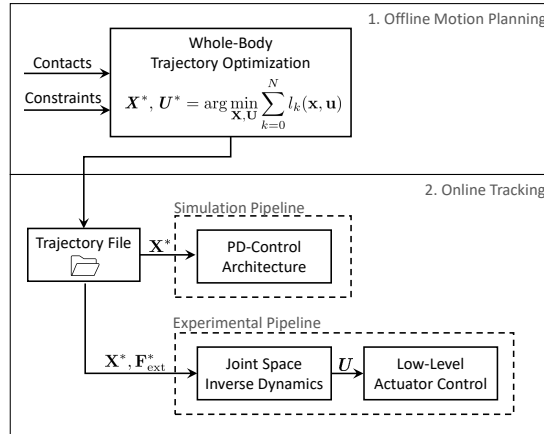


Abbildung II.9: Überblick über den Simulations- und Versuchsablauf.

der Kontaktkräfte, die Coulomb-Reibung der resultierenden Kraft und den Druckmittelpunkt (CoP) innerhalb des Stützbereichs jedes Fußes in Kontakt:

$$\lambda^z > 0, \quad |\lambda^x| \leq \mu\lambda^z, \quad |\lambda^y| \leq \mu\lambda^z, \quad |X| \geq c_x, \quad |Y| \geq c_y. \quad (\text{II.10})$$

In (II.10) bezeichnet  $\mu$  den statischen Reibungskoeffizienten und modelliert einen räumlichen Reibungskegel, und  $c_x$  und  $c_y$  bezeichnen die Position des CoP in Bezug auf die Dimensionen X und Y der rechteckigen Roboterfüße. Wir bezeichnen diesen Bewegungsplanungsansatz als *contact stability soft-constrained DDP* [22].

Wir verfolgen die mit dem gewählten Trajektorienoptimierungsansatz geplante Bewegung in Echtzeit mit einem PD-Regler im PyBullet-Simulator und mit einer Gelenkraum-Online-Stabilisierung auf dem realen System, wie in Abb. II.9 dargestellt. Der Soft-Constrained-DDP-Ansatz zur Kontaktstabilität berechnet inhärent ausgeglichene Bewegungen, die in einer Trajektorien-datei zusammengefasst werden. Sie enthält die optimalen Zustandstrajektorien  $\mathbf{X}^*$ , die OC-Eingaben  $\mathbf{U}^*$ , und die daraus resultierenden Kontaktschlüssel  $\mathbf{F}_{\text{ext}}^*$ , die auf die Füße wirken. Die Trajektorien werden mit Hilfe von kubischen Splines auf 1 kHz interpoliert, um eine gleichmäßige Bewegung zu gewährleisten. Die geplanten Bewegungen werden auf der Grundlage eines baumartigen Robotermodells errechnet. Für die dynamische Echtzeitsteuerung erweist sich dieses vereinfachte Modell als ausreichend, wenngleich die Genauigkeit reduziert ist [32]. Dennoch bleibt das Problem, die Ergebnisse aus dem unabhängigen Gelenkraum in den Aktuatorikraum zu transformieren. Wir verwenden das modulare Software-Framework HyRoDyn (Hybrid Robot Dynamics) [33] um die für das serialisierte Robotermodell erzeugten Trajektorien abzubilden und die Kräfte der jeweiligen Linearaktoren zu berechnen. Low-Level-Aktorsteuerungen kompensieren Abweichungen von den Referenztrajektorien. Analog zur Simulationspipeline verwendet dieser Echtzeitregelungsansatz eine kaskadierte Rückführung von Position, Geschwindigkeit und einem zusätzlichen Stromregelkreis. HyRoDyn sowie die Low-Level-Steuerung des RH5-Roboters werden mit der Middleware Robot Construction Kit (Rock) [31], implementiert, die auf dem Orocos Real Time Toolkit basiert und das Software-Rückgrat des humanoiden Roboters RH5 bildet.

*Experimente zum Gleichgewicht und Gehen von Humanoiden (AP C.2.3):* Es wurde ein Rahmenwerk für die automatische Generierung optimaler Trajektorien für verschiedene Roboterkonfigurationen, z.B. ohne Kopf oder ohne Arme, entwickelt. Der Benutzer muss im Wesentlichen die gewünschte Konfiguration des humanoiden Roboters, die Art der Bewegung und die Bewegungsparameter definieren, wie in Tabelle II.3. zusammengefasst. Die Plots, die Bewegungsanzeige und die Protokolldateien sind dann bereit,

Tabelle II.3: Bewegungsmerkmale und Gewichtung der Optimierungsbedingungen.

Antrag	Merkmale			Einschränkungen bei der Optimierung					
	Länge (m)	Höhe (m)	Zeit (s)	Tasks $\Phi_{\text{foot}}$	Stabilität $\Phi_{\text{CoM}}$	Grenzw $\Phi_{\text{friction}}$	Regularisierung $\Phi_{\text{joint}}$	$\Phi_{\text{posture}}$	$\Phi_{\tau}$
Dynamische Kniebeugen	–	0.14	10	$1e^6$	$1e^8$	$1e^2$	$1e^0$	$1e^1$	$1e^{-1}$
Statisches Gehen	0.1	0.05	1.8	$1e^8$	$1e^6$	$1e^2$	$1e^6$	$1e^1$	$1e^{-1}$
Dynamisches Gehen	0.8	0.05	1.8	$1e^8$	$1e^6$	$1e^2$	$1e^0$	$1e^1$	$1e^{-1}$
Einbeinige Balance	–	0.15	17.43	$1e^8$	$1e^6$	$1e^2$	$1e^0$	$1e^1$	$1e^{-1}$

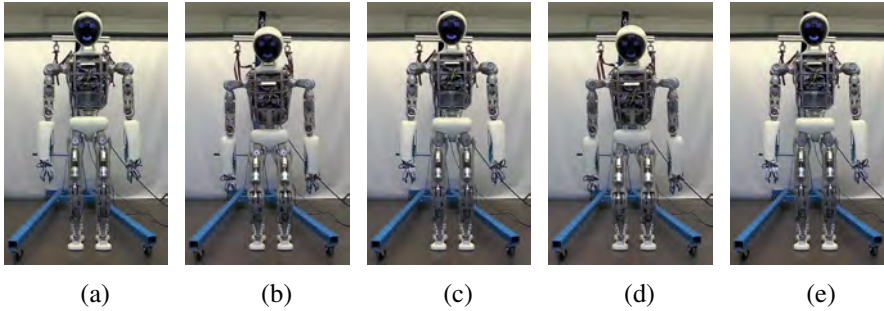


Abbildung II.10: Kniebeugen-Experiment: Abfolge von schnellen Kniebeugen von (a) einer Ausgangsposition über (b,d,f) Absenken des CoM um 15 cm und (c,e,g) Zurückkehren zur Ausgangsposition.

am realen Roboter gemäß der in Fig. II.9. beschriebenen Pipeline getestet zu werden. Es wurden mehrere menschenähnliche Verhaltensweisen implementiert: statisches Gehen, dynamisches Gehen, einbeiniges Balancieren und Hocken, wie im Folgenden beschrieben:

1. Dynamische Kniebeugen: Das schnelle Hockexperiment beinhaltet eine schnelle vertikale Basisbewegung im Bereich von 15 cm innerhalb von zwei Sekunden, während der Roboter im Gleichgewicht bleibt. Die Bewegungen wurden mit der gewählten Methode der Trajektorienoptimierung berechnet und mit einer Gelenkraum-PD-Kontrolle sowohl in der Simulation als auch im realen System stabilisiert (siehe Fig. II.10).
2. Statisches Gehen: Wir gehen von einer quasi-statischen Bewegung aus und verwenden daher ein statisches Stabilitätskriterium, indem wir sicherstellen, dass der projizierte Schwerpunkt innerhalb der konvexen Hülle des Stützpolygons liegt. Zu diesem Zweck besteht das Optimierungsproblem aus insgesamt fünf Fortbewegungsphasen, die in Fig. II.11 (Meilenstein 1) dargestellt sind. Ausgehend von (a) einer Ausgangsposition wird (b) der Masseschwerpunkt (CoM) über den linken Fuß verschoben, (c) ein Schritt nach rechts gemacht, (d) der Masseschwerpunkt über den rechten Fuß verschoben, (e) ein Schritt nach links gemacht und (f) der Masseschwerpunkt auf die halbe Länge der Linie, die beide Fußmittelpunkte schneidet, verschoben, während zur Ausgangsposition zurückgekehrt wird. Zusätzlich wird der Masseschwerpunkt für eine konstante Höhe über den gesamten Gang optimiert. Tabelle II.3 gibt einen kompakten Überblick über die gewünschten Gangcharakteristika und die angewandten Randbedingungen der Optimierung.

Durch die Einbeziehung der Kontaktstabilität in unseren vorgeschlagenen Ansatz war es möglich, die Lauftrajektorie in einem anderen dynamischen Simulator, nämlich PyBullet, mit einer einfachen PID-Regelung auf Aktuatorebene zu stabilisieren (Meilenstein 1). Diese Bewegung wurde auch auf dem realen System ausgeführt. Der Roboter war in der Lage, einen statischen Gang in alle Richtungen ohne weitere Anpassungen auf dem realen System auszuführen, bevor er die Übertra-

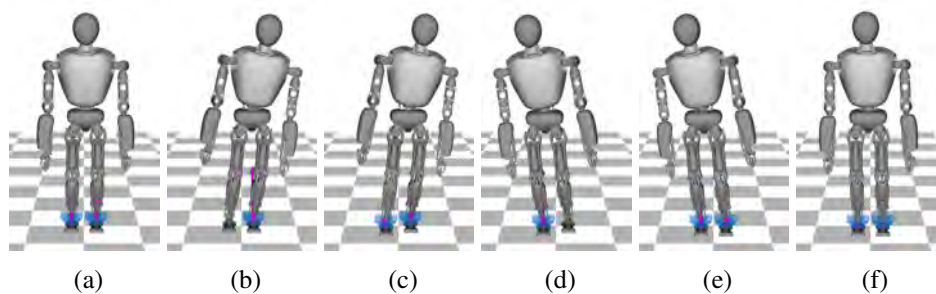


Abbildung II.11: Statisches Gehen auf der Grundlage von CoM-Bewegungen [Video]

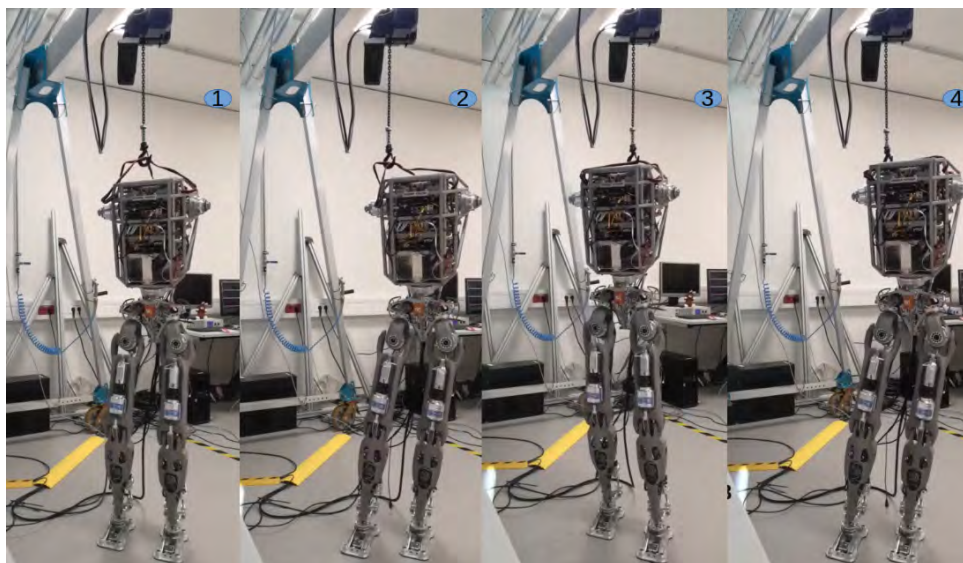


Abbildung II.12: RH5 führt ein statisches Gehen aus

gung von der Simulation in die reale Welt durchführte; die resultierende Bewegung ist in Fig. II.12 zu sehen.

3. **Dynamisches Gehen:** Ein wesentliches Merkmal des dynamischen Gehens ist, dass der projizierte Schwerpunkt des Roboters im Gegensatz zum statischen Gehen nicht immer im Stützpolygon liegen muss. Daher muss diese Beschränkung zu diesem Zweck gelockert werden. Wir konnten das dynamische Gehen mit verschiedenen Gehgeschwindigkeiten im Bereich von 0,1 bis 1 m/s realisieren. Ein Beispiel für dynamisches Gehen ist in Fig. II.13 und Fig. II.14 zu sehen. Wir haben festgestellt, dass bei Gehgeschwindigkeiten von mehr als 0,35 m/s der Solver oft mit einer vordefinierten CoM-Trajektorie warmgestartet werden muss, um eine praktikable Lösung zu finden, wie es in [16] geschieht. Dank der von uns vorgeschlagenen soft-constrained DDP [22] können viele dieser dynamischen Gänge mit niedrigeren Geschwindigkeiten im PyBullet-Simulator mit einem einfachen PID-Regler stabilisiert werden. Schnellere dynamische Gänge erfordern jedoch einen zusätzlichen bewegungsstabilisierenden Regler, der im Rahmen des M-RoCK-Projekts entwickelt wird.
4. **Einbeiniges Balancieren:** Bei der einbeinigen Balancierbewegung beginnt der Roboter die Bewegung mit beiden Füßen in zwangsweisem Kontakt mit dem Boden. Von dieser Phase mit zwei Füßen wechselt der Roboter in die Phase mit einem Fuß, indem er den CoM nach rechts bewegt und

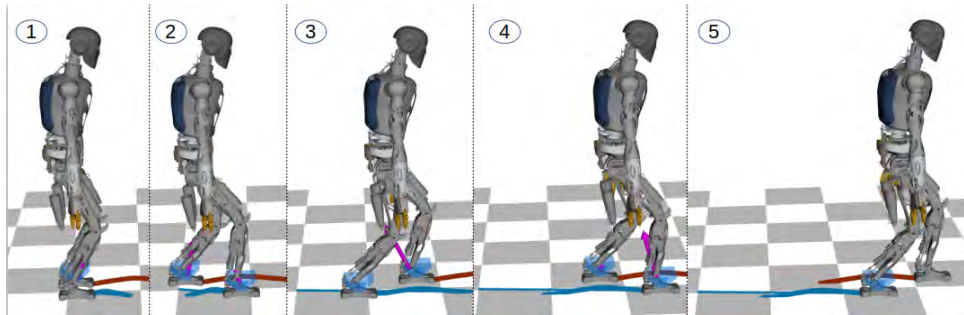


Abbildung II.13: Simulation des dynamischen Gehens mit einer Geschwindigkeit von ca. 0,15 m/s.



Abbildung II.14: Von rechts nach links: RH5 bei der Ausführung eines dynamischen Laufs in einer simulierten Umgebung mit WBC.

den linken Fuß auf eine Höhe von 15 cm anhebt, wie in Fig. II.15a. gezeigt. Das OC-Problem wurde unter Verwendung der in Tabelle II.3 aufgeführten Optimierungskosten entsprechend den Bewegungsmerkmalen und Optimierungseinschränkungen formuliert. Anschließend wurde die Bewegung auf dem realen Roboter mit der WBC-Methode durchgeführt, wobei die Lücke zwischen der Simulation und der Realität überbrückt wurde (siehe Fig. II.15b).

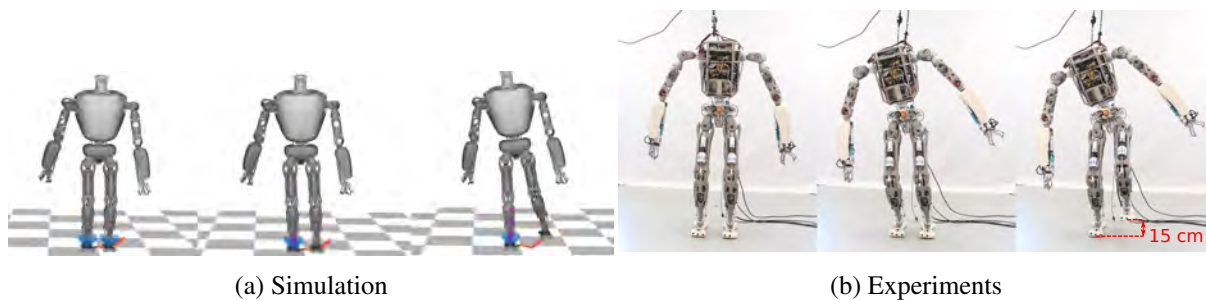


Abbildung II.15: Einbeiniges Balancieren in der Simulation und im realen Experiment

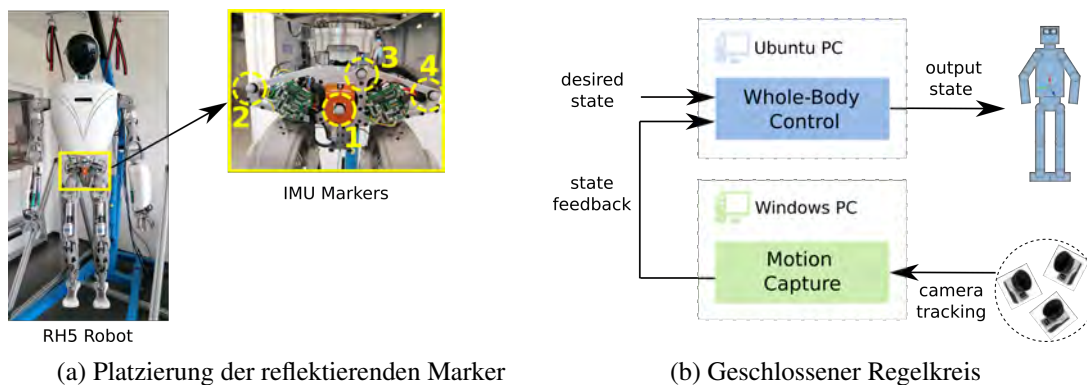


Abbildung II.16: Zustandsrückmeldung der Bewegungserfassung im geschlossenen Regelkreis.

*Kombination von Lern- und Kontrollmethoden (AP C.2.4):* Es gibt zahlreiche Parallelen zwischen verstärkendem Lernen (*reinforcement learning*, RL) und optimalen Kontrollmethoden, die beide in der Theorie der dynamischen Programmierung (einer Erweiterung der Theorie von Hamilton und Jacobi) begründet sind. Oftmals werden verschiedene Terminologien für denselben Zweck verwendet: Aktion — Steuereingabe, Controller — Agent, Cost to go — Wertfunktion, usw.. Bislang haben beide Methoden für niedrigdimensionale Systeme (drehmomentbegrenzt einfaches Pendel, Hüpfbein) in Simulationen und realen Experimenten gleich gut abgeschnitten (siehe AP C.4). RL scheint eine größere Allgemeinheit als die in diesem Projekt gewählte Methode der optimalen Steuerung zu bieten, so dass die resultierenden Regler robuster gegenüber schwer zu modellierenden Effekten wie Kontakt und Gelenkreibung sein könnten. Bei hochdimensionalen Systemen wie dem humanoiden Roboter RH5 scheint RL jedoch stärker unter dem Fluch der Dimensionalität zu leiden als DDP-basierte Methoden der optimalen Steuerung, da DDP die Problemstruktur im Vergleich zu RL besser ausnutzt. Ein Beispiel dafür ist, dass RL-Laufsteuerungen für reale Roboter für niedrigdimensionale Zweibeiner gezeigt wurden [35], aber nicht für vollständige Humanoide. Daher kann RL für eine robuste Bewegungsplanung an einem vereinfachten Modell des menschlichen Gehens verwendet werden, das dann zum Warmstart optimaler Kontrollmethoden oder zur Modellierung der Restdynamik des realen Roboters verwendet werden kann, um die Simulations-Realitäts-Lücke zu schließen.

**AP C.3: Analyse des Verhaltens des Demonstrators** Dieses Arbeitspaket beinhaltet die Übertragung von Lern- und Steuerungsstrategien auf das reale System und den Vergleich mit der Grundwahrheit aus dem Bewegungserfassungssystem.

*Bestimmung eines geeigneten Verhaltens und einer Disparitätsbeschreibung in der Simulation (AP C.3.1):* Die optimalen Steuerungstrajektorien oder Verstärkungslernstrategien sind möglicherweise nicht auf ei-

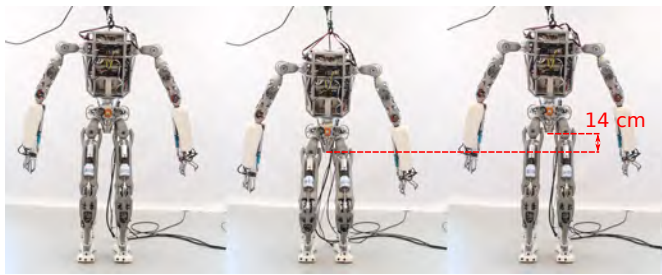
ne andere Simulationsumgebung und den realen Roboter übertragbar (wie in AP C.2.3 diskutiert). Während der erste Fall als Sim-2-Sim-Lücke bekannt ist, wird der zweite Fall als Sim-2-Real-Lücke bezeichnet. Um die Übertragung der Bewegungen auf das reale System zu ermöglichen, müssen die Bewegungstrajektorien mit einem zusätzlichen Stabilisierungsregler stabilisiert werden. Für einfache Bewegungen kann eine PID-Regelung auf Aktuatorebene ausreichen, für komplexe dynamische Bewegungen ist jedoch eine QP-basierte Ganzkörperregelung (WBC) erforderlich. Eine QP-basierte WBC für den RH5 wird im Rahmen des M-RoCK-Projekts entwickelt. Abb. II.14 zeigt die Ausführung des dynamischen Gangs in der Simulationsumgebung RaiSim unter Verwendung des WBC-Ansatzes. Der Roboter ist in der Lage, den OC-Trajektorien zu folgen und die Bewegungen erfolgreich auszuführen. Während der Bewegungsausführung wurden Instabilitäten auf der Ebene der Knöchel beobachtet, die zu größeren Fehlern entlang der  $y$ -Achse des CoM-Profiles bei der Ausführung des dynamischen Gangs führten. Die Füße werden mit der Knöchel-Admittanz-Methode erfolgreich stabilisiert. Da diese Methode jedoch durch direkte Aktualisierung der befohlenen Referenzen funktioniert, kann WBC deren Einfluss nicht vorhersehen, sondern muss ihn in der folgenden Kontrolliteration kompensieren. Das Verhalten des Fußgelenks kann eine Folge der Unterschiede zwischen der Simulation der optimalen Steuerung und der simulierten Umgebung sein, die für die Online-Stabilisierung verwendet wird, insbesondere in Bezug auf die Berechnung der Kontakte. Der Unterschied zwischen Simulation und Simulation ist nur ein Bruchteil dessen, was ein Unterschied zwischen Simulation und Realität sein kann, aber er beweist bereits die Notwendigkeit der Online-Stabilisierung und bewertet die Vorteile der Ganzkörperregelung.

*Implementierung des Bewegungserfassungssystems (AP C.3.2):* Wir haben ein kamerabasiertes Motion-Capture-System eingerichtet, um die Position und Orientierung der schwebenden Basis des Roboters zu verfolgen. Das System besteht aus mindestens 3 Kameras, die über Ethernet mit einem Windows-PC verbunden sind. Eine hohe Verfolgungsfrequenz von 800 Hz wird erreicht, indem die Kameraauflösung als Kompromiss zwischen Markergröße und Verfolgungsgeschwindigkeit reduziert wird. Die Messlatenz liegt in einem niedrigen Bereich von 2-4 ms, was es uns ermöglicht, die Kameraverfolgung für eine Echtzeit-Feedback-Steuerung zu nutzen. Um die schwimmende Basis des Roboters zu verfolgen, platzieren wir eine Reihe von 4 reflektierenden Markern auf dem IMU-Starrkörperrahmen (siehe Abb. II.16a). Die verfolgte Position und Orientierung des IMU-Rahmens des Roboters wird in Bezug auf das Weltkoordinatensystem der Kamera definiert, das während der Kalibrierungsroutine für die Bewegungserfassung willkürlich gewählt wird. Um den verfolgten Starrkörper im Roboterweltkoordinatensystem auszudrücken, definieren wir das Roboterweltkoordinatensystem als die Bodenprojektion des Wurzelrahmens an der Ausgangsposition. Nach der Transformation des von der IMU verfolgten Starrkörpers vom Kamera- in das Roboterwelt-Koordinatensystem können wir die Position und Ausrichtung der schwebenden Basis des Roboters in Roboterwelt-Koordinaten abrufen. Die schwebende Basis wird dann, wie in Abb. II.16b dargestellt, an den Feedback-Regler übergeben.

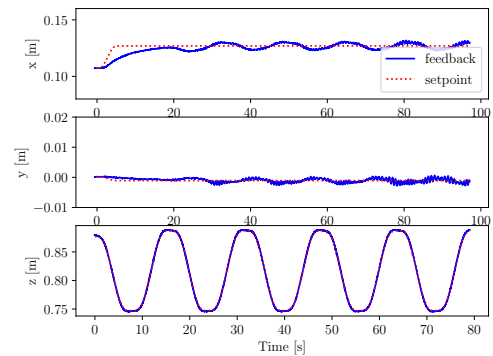
*Experimente mit dem realen System (AP C.3.3):*

*RH5 in der Hocke:* Wir haben Hockexperimente mit dem humanoiden Roboter RH5 durchgeführt und dabei externe Bewegungserfassung verwendet, um Zustandsrückmeldungen für den Ganzkörperregler zu erhalten (siehe Abb. II.16b). Wir erreichten ein robustes und stabiles Hockverhalten bei verschiedenen Geschwindigkeiten und Höhen, mit Hocken bis zu 14 cm Höhenunterschied, wie in Abb. II.17a. Die entsprechenden Diagramme sind in Abb. II.17b. zu sehen. In ähnlicher Weise wurde das einbeinige Balancierverhalten auch am realen Roboter für verschiedene Höhen des angehobenen Beins einschließlich Höhen bis zu 15 cm realisiert (siehe Fig. II.15). Die Ergebnisse sind in [11] veröffentlicht worden.

*RH5 bei der Durchführung dynamischer Bewegungen:* Dynamische Gehbewegungen wurden erfolgreich auf dem humanoiden Roboter RH5 durch den Einsatz externer Bewegungserfassung implementiert, die den anfänglichen Haltungszustand für die Ganzkörpersteuerung lieferte. Diese Methode ermöglichte ein



(a) Zeitraffer



(b) Verfolgung von Koordinaten mit gleitender Basis

Abbildung II.17: Hockende Bewegung unter Verwendung von externem Motion Capture State Feedback mit RMSE von  $C_x = 0.004$ ,  $C_y = 0.001$  und  $C_z = 0.001$ .

stabiles und robustes dynamisches Gehen bei unterschiedlichen Geschwindigkeiten und Schrittlängen, wodurch die Grenzen des Roboters sowohl in Bezug auf die Geschwindigkeit als auch auf den Bewegungsumfang effektiv ausgereizt wurden. Unseres Wissens nach ist dies das erste Mal, dass ein dynamisches Gehen bis zu 0,43 m/s erreicht wurde, wie in Fig. II.18 dargestellt. Sieht man von Robotern mit aktiven Zehengelenken ab, gehört der RH5 zu den schnellsten Humanoiden mit ähnlicher Größe und Antriebsmodalität. Darüber hinaus werden die dynamischen Fähigkeiten des RH5 in weiteren Experimenten mit schnellem Steppen an Ort und Stelle (Fig. II.19) und langem Schritt (Fig. II.20) demonstriert. Diese Ergebnisse wurden durch die Kombination von Zero-Moment-Point-Bewegungsplanung und Ganzkörpersteuerungsansätzen in einer maßgeschneiderten Weise erzielt, die geeignet ist, eine hohe Leistung von positionsgesteuerten Robotern ohne vollständige Zustandsrückmeldung zu erreichen. Die Ergebnisse wurden in [1] veröffentlicht.

*Erzeugung von hochdynamischen, anthropomorphen Bewegungen mit einem humanoiden Roboter:* Um eine hochdynamische und anthropomorphe Bewegung mit Hilfe eines humanoiden Roboters zu erzeugen, untersuchen wir, wie wir die Fähigkeiten unseres serienparallelen Hybridroboters RH5 voll ausschöpfen können. Heutzutage wird die seriell-parallele Hybridtopologie häufig bei der Entwicklung von humanoiden Robotern verwendet, aber bei der Bewegungsoptimierung werden sie oft vernachlässigt, und die Bewegungen werden für eine serielle Abstraktion des Systems berechnet, und die Lösung wird dann auf die Aktuatorkoordinaten abgebildet. Dies hat zur Folge, dass die volle Leistungsfähigkeit des Roboters nicht ausgenutzt werden kann. Wir haben eine dynamische anthropomorphe Gewichtshebebewegung unter Verwendung der Konstruktionspezifikationen des RH5-Robotermodells entwickelt. Seine seriell-parallele Hybridkonstruktion bietet eine hohe Steifigkeit und eine große Traglastkapazität. Um diese Fähigkeiten zu nutzen, ist es jedoch wichtig, das gesamte Robotermodell zu berücksichtigen. Zu diesem Zweck haben wir die Einschränkungen der Schleifenverschlüsse in den Prozess der Bahnoptimierung einbezogen. Auf diese Weise konnten wir die Einschränkungen des baumartigen Abstraktionsmodells überwinden, das derzeit zur Modellierung und Steuerung von Systemen mit Schleifenschlussmechanismen verwendet wird. Auf diese Weise konnten wir sowohl in der Simulation als auch im realen System eine dynamische Hebebewegung mit einem Gewicht von 15 kg erreichen. Durch die Verwendung des vollständigen Modells werden nicht nur die Nutzlastfähigkeiten verbessert, sondern auch der Arbeitsbereich des Roboters besser ausgenutzt. Diese Eigenschaft wurde bei der Gewichtshebebewegung deutlich, bei der keines der drei baumartigen Abstraktionsmodelle erfolgreich eine optimale Trajektorie zum Er-

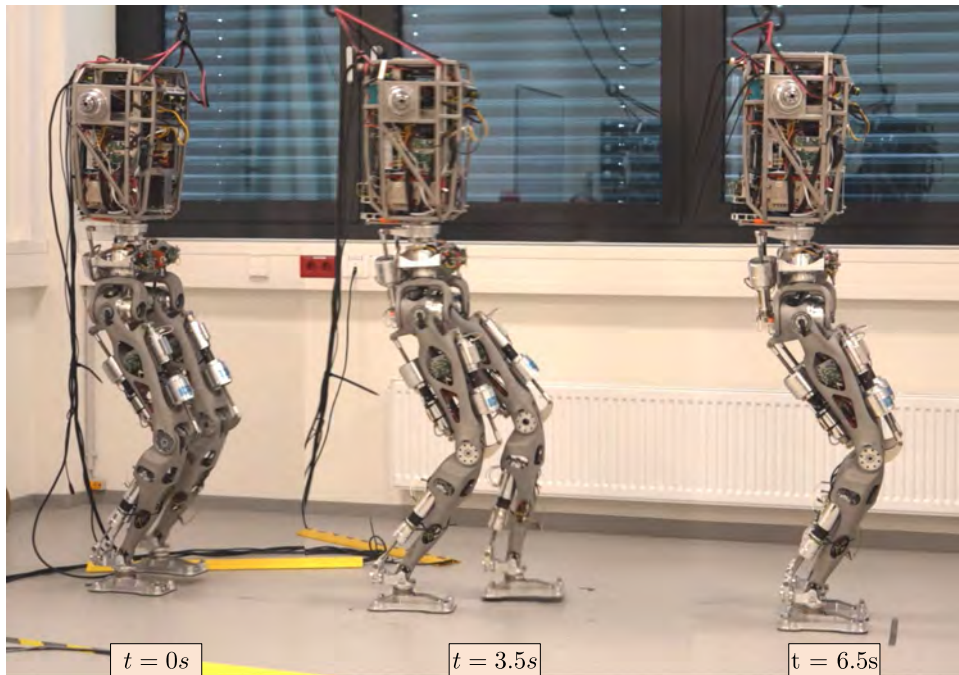


Abbildung II.18: RH5-Roboter beim schnellen dynamischen Gehen.

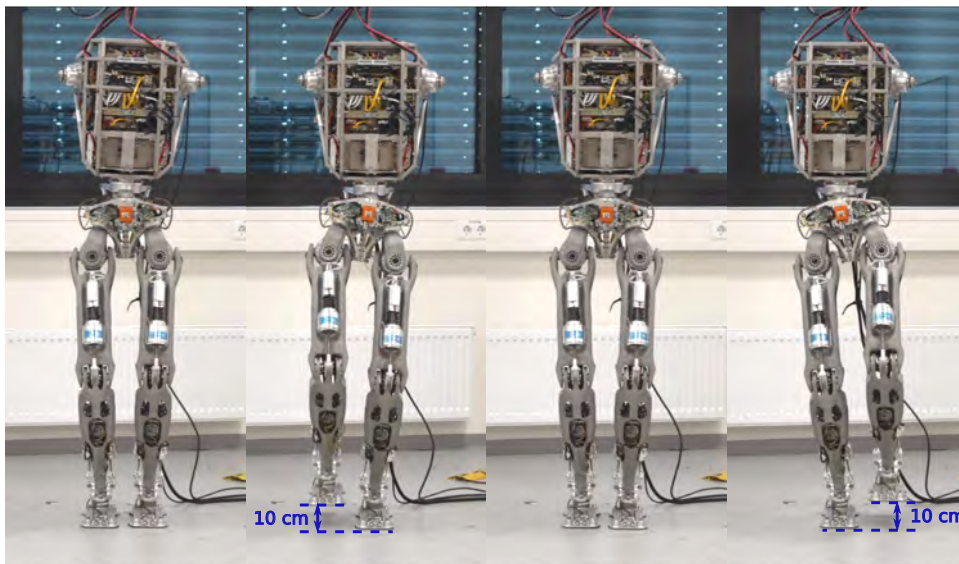


Abbildung II.19: RH5-Roboter beim schnellen dynamischen Gehen auf der Stelle.

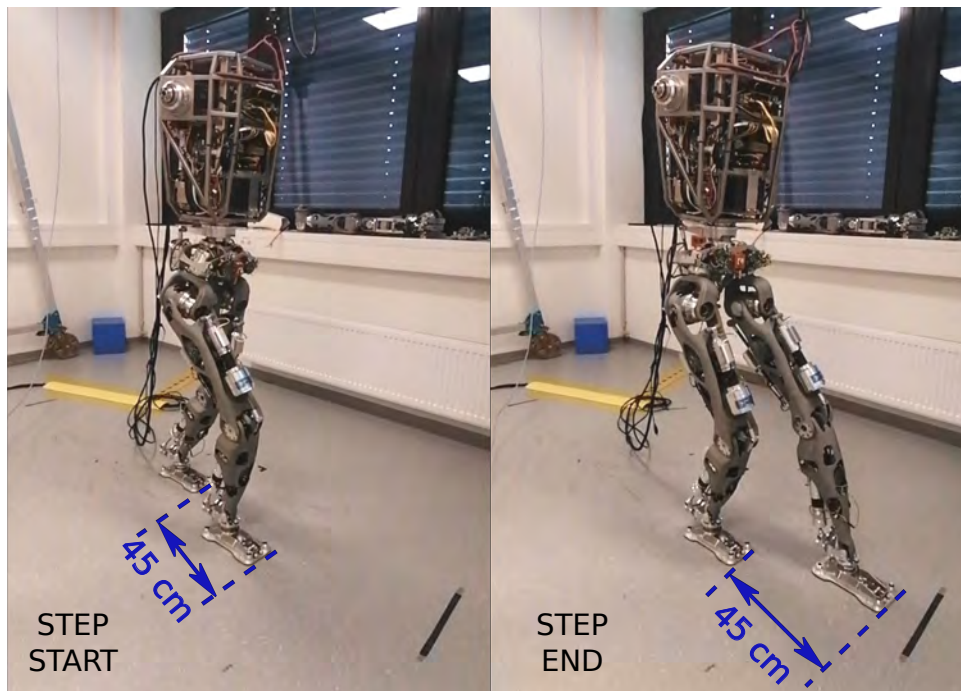


Abbildung II.20: RH5-Roboter beim schnellen Steppen an Ort und Stelle.

reichen der Zielkonfiguration beim Heben der Lasten erreichen konnte (Fig. II.21) während das vollständige Modell diese Aufgabe erfolgreich bewältigte (Fig. II.22). Die Ergebnisse sind in [2] veröffentlicht worden.

**Lernbasierte Steuerung für humanoides Gehen (AP C.4):** *Aufbau des Rahmens und erste Tests (AP C.4.1):* Als Voraussetzung für das endgültige Ziel der Entwicklung eines hybriden, auf Lernen und optimaler Steuerung basierenden Steuerungsansatzes für unsere Zielanwendung des humanoiden Gehens konzentrieren wir uns zunächst auf vereinfachte Modellsysteme, die mit Modellen von Gehsystemen verbunden sind. Für ein einfaches, drehmomentbegrenztes Pendel, das untersteuert ist, wurde ein RL-Regler nach dem SAC-Algorithmus (Soft Actor Critic RL) trainiert. Die Pendelparameter für das Training wurden an das physikalische System angepasst, so dass die Leistung des Reglers unter realistischen Bedingungen überprüft werden kann. Der Regler wurde im reinen Drehmomentregelungsmodus eingesetzt, d. h. während der Ausführung des RL-Drehmomentreglers wurden keine zusätzlichen Zielpositions- und Geschwindigkeitsinformationen an den Regler gegeben. Es konnte nachgewiesen werden, dass der RL-Regler das Pendel für Drehmomentgrenzen zwischen 0,7 und 2,5 Nm (die ungefähre Grenze, bei der das Pendel vollständig betätigt wird) aufschwingen und stabilisieren kann. Abb. II.23 zeigt Zeitserienschnappschüsse aus einem Experiment mit dem realen System. Wir haben gezeigt, dass der Regler sich von Störungen im aufrechten Zielzustand und während des Aufschwingvorgangs erholen kann.

Auch Gewichtsveränderungen des Pendels in einem kleinen Bereich hatten keinen Einfluss auf die Leistung des Reglers. Dennoch ist es erwähnenswert, dass die Regelungsleistung nicht spürbar durch das Restrauschen der Sensormesswerte beeinträchtigt wurde.

Um Modelle für die kontaktreiche Fortbewegung zu entwickeln und weiteres Fachwissen für den Einsatz von lernbasierten Reglern auf realer Hardware aufzubauen, haben wir die Generierung von Reglern für variable Sprunghöhen für ein Monoped untersucht. In diesem Fall beruht die klassische Kontrolltheorie typischerweise auf der genauen Erkennung verschiedener Sprunghöhen, wie Abheben oder Aufsetzen,

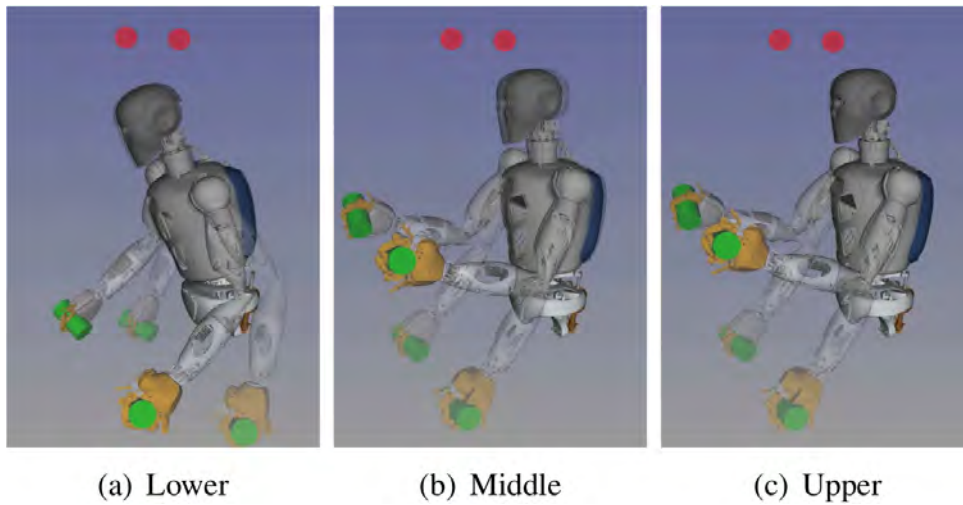


Abbildung II.21: Screenshots der Simulationen des Roboters RH5 Manus, der 15 kg hebt, mit dem seriellen Abstraktionsmodell für untere, mittlere und obere Grenzen. Die unscharfen Bilder entsprechen dem Anfangszustand des Roboters und die roten Blasen bezeichnen die Ziele des Endeffektors.

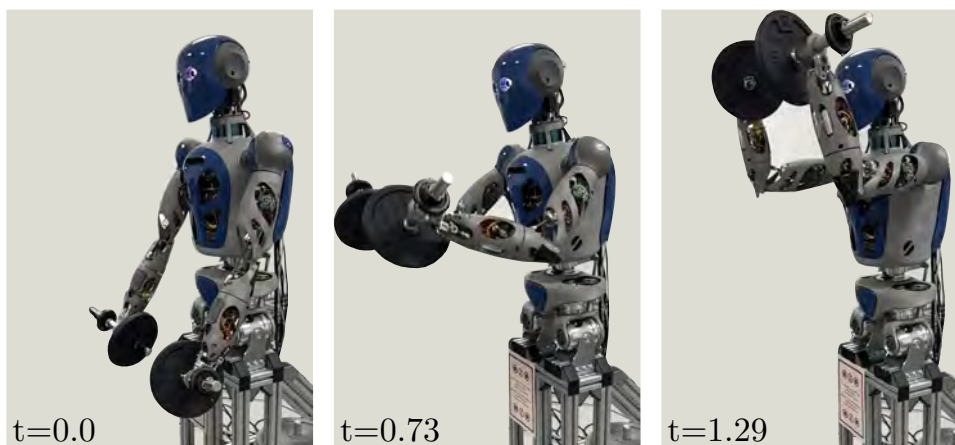


Abbildung II.22: Der RH5-Roboter erreicht eine dynamische Hubkapazität von 15 kg, indem er die Möglichkeiten der hybriden Serien-Parallel-Topologie des RH5 voll ausschöpft.



Abbildung II.23: Unterschwingung des Pendels auf einem realen Testsystem mit einer Drehmomentgrenze von 0,9 Nm. Am Ende wird eine Störung an der aufrechten Zielposition kompensiert.

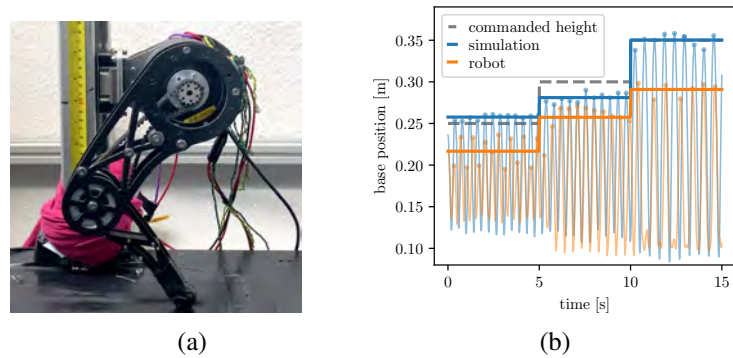


Abbildung II.24: Variable Sprunghöhensteuerung für ein selbstgebautes Monoped.

und der Verwendung unterschiedlicher Regler für jede Phase. Daher ist es ein geeignetes Verhalten für die Einbeziehung von lernbasierten Ansätzen. Wir haben einen Ende-zu-Ende-RL-Drehmomentregler entwickelt, der relevante Sprungphasen implizit erkennt und die Notwendigkeit manueller Heuristiken für die Zustandserkennung beseitigt. Wir erweitern auch eine Methode für den Sim-2-Real-Transfer auf kontaktreiche dynamische Aufgaben, was zu einem erfolgreichen Einsatz auf dem Roboter ohne weitere Parameterabstimmung führt. Abb. II.24 zeigt den selbstgebauten Roboter und einen Vergleich der Sprunghöhen bei wechselnden Höhenbefehlen, sowohl in der Simulation als auch am realen Roboter.

*Integration eines differenzierbaren Dynamikmodells (AP C.4.2):* Optimale Kontrollmethoden, wie DDP, verwenden bereits ein differenzierbares Dynamikmodell. Das ursprüngliche Ziel war ein erlerntes Dynamikmodell für schnelle modellbasierte End-to-End-RL, was im aktuellen Stand nicht mehr von hoher Priorität war.

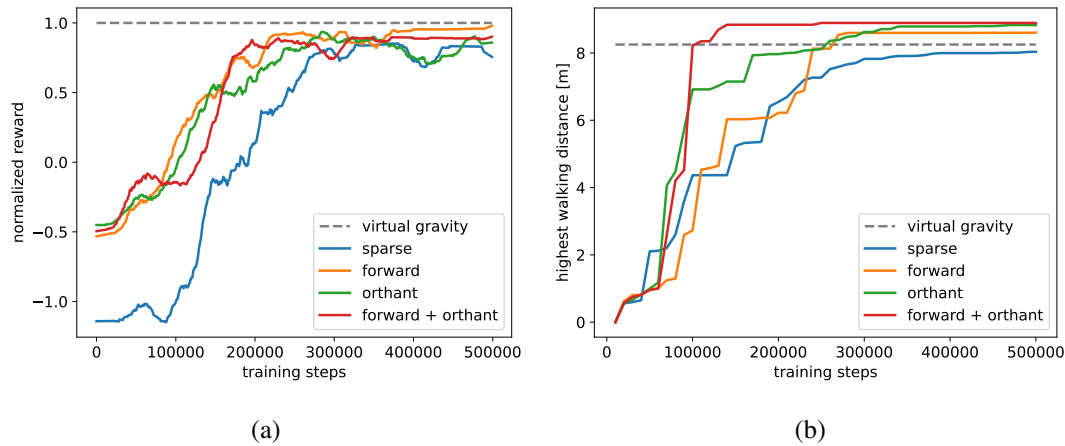


Abbildung II.25: Leistung der Kompass-Ganggenerierung bei verschiedenen Belohnungskonfigurationen.

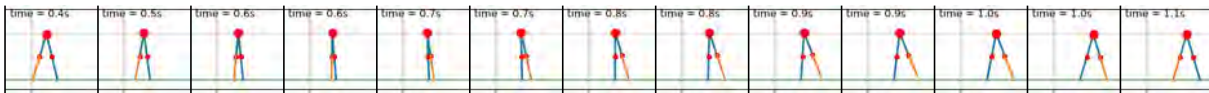


Abbildung II.26: Schnappschüsse des stabilen Kompass-Gangs. Orange markiert das Schwungbein.

*Laufen lernen und kontinuierliche Anpassung (AP C.4.3):* Um eine Verbindung zwischen einer symbolischen Definition des sicheren Gehverhaltens und numerischen Methoden zur Erzeugung dieses Verhaltens herzustellen, haben wir den Gangzyklus des Kompass-Geher (*compass walker*) mit Hilfe von Orthanten beschrieben. Aus AP II.2.3 auf Seite 7 ist bekannt, dass der Phasenraum  $(\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2)$  in 16 Orthanten unterteilt ist, die in einer festen Reihenfolge besucht werden müssen. Diese Orthantenfolge kann in der numerischen Belohnungsdefinition verwendet werden, indem das Verbleiben in der gewünschten Orthantenfolge gefördert und Zustände in nicht besuchte Orthanten oder andere falsche Orthantenübergänge bestraft werden. Dazu werden Prädikate verwendet, die definieren, ob die aktuelle Orthante und der mögliche aktuelle Orthantenübergang in die gewünschte Laufsequenz passen. Wir gehen davon aus, dass eine solche Belohnung das Erlernen eines stabilen Gehverhaltens auf einem ebenen Boden mit dem Kompass-Geher beschleunigt. Um dies zu beurteilen, vergleichen wir mit einer spärlichen Belohnung, die nur die in einer bestimmten Zeit zurückgelegte Strecke berücksichtigt. Darüber hinaus analysieren wir auch eine Belohnung, die auf einem anderen Prädikat basiert, nämlich dass die Projektion der Hüftgeschwindigkeit in Vorwärtsrichtung immer positiv ist. Während letzteres ein häufig verwendeter Belohnungsbegriff beim verstärkenden Lernen oder bei der optimalen Steuerung ist, ist die Verwendung von Orthant-Sequenzen unseres Wissens nach neu. Wir evaluieren auch eine Kombination aus Vorwärtsgeschwindigkeitsbelohnung und Orthant-Sequenzbelohnung. Abb. II.25 zeigt die Ergebnisse dieser Pilotstudie und Abb. II.26 zeigt einen resultierenden Gang. Hinsichtlich der erreichten Laufdistanz über die Zeit stellen wir fest, dass die spärliche Belohnung zu einer Konvergenz bei der geringsten zurückgelegten Distanz führt. Die auf dem Orthant-Prädikat basierende Belohnung ist der geschwindigkeitsbasierten Belohnung ebenbürtig oder potentiell überlegen, da beide zu höheren zurückgelegten Distanzen führen. Betrachtet man die Konvergenzdistanz in der spärlichen Rewardkonfiguration als Referenz, so zeigen beide Methoden eine ungefähr zweifache Trainingsbeschleunigung. Die Kombination beider Rewards schneidet am besten ab und führt zu einer etwa fünffachen Beschleunigung. Alle prädikatsbasierten Reward-Setups erreichen höhere Laufdistanzen als der virtuelle Schwerkraft-Controller, der als analytisch abgeleitete Basislinie dient. Die Ergebnisse wurden in [7] veröffentlicht.

	definition	weight
Linear velocity tracking	$\phi(\mathbf{v}_{b,xy}^* - \mathbf{v}_{b,xy})$	$1dt$
Angular velocity tracking	$\phi(\boldsymbol{\omega}_{b,z}^* - \boldsymbol{\omega}_{b,z})$	$0.5dt$
Linear velocity penalty	$-\mathbf{v}_{b,z}^2$	$4dt$
Angular velocity penalty	$-  \boldsymbol{\omega}_{b,xy}  ^2$	$0.05dt$
Joint motion	$-  \ddot{\mathbf{q}}_j  ^2 -   \dot{\mathbf{q}}_j  ^2$	$0.001dt$
Joint torques	$-  \boldsymbol{\tau}_j  ^2$	$0.00002dt$
Action rate	$-  \mathbf{q}_j^*  ^2$	$0.25dt$
Collisions	$-n_{collision}$	$0.001dt$
Feet air time	$\sum_{f=0}^4 (\mathbf{t}_{air,f} - 0.5)$	$2dt$

Tabelle II.4: Definition der Belohnungsfunktionen, mit  $\phi(x) := \exp(-\frac{||x||^2}{0.25})$ . Die z-Achse ist auf die Schwerkraft ausgerichtet.

Joint positions	$\mathbf{q}_j$
Joint velocities	$\dot{\mathbf{q}}_j$
Joint accelerations	$\ddot{\mathbf{q}}_j$
Target joint positions	$\mathbf{q}_j^*$
Joint torques	$\boldsymbol{\tau}_j$
Base linear velocity	$\mathbf{v}_b$
Base angular velocity	$\boldsymbol{\omega}_b$
Commanded base linear velocity	$\mathbf{v}_b^*$
Commanded base angular velocity	$\boldsymbol{\omega}_b^*$
Number of collisions	$n_c$
Feet air time	$\mathbf{t}_{air}$
Environment time step	$dt$

Tabelle II.5: Definition der Symbole.

*Übertragung auf den RH5-Ganzkörper-Humanoiden unter Verwendung von massivem parallelem Reinforcement Learning (AP C.4.4):* Um den Kreislauf zwischen AP C.3 und AP C.4 zu schließen, konzentrierten wir uns auf das Erlernen des dynamischen Gehens auf einem komplexen untermotorisierten Mehrkörpersystem. Der neue humanoide Roboter RH5 wurde für diese Studie ausgewählt, um das Laufen zu lernen. Die Software Issac Gym Deep Reinforcement Learning (DRL) wurde zum Trainieren der Strategie in einer massiv parallelen DRL-Konfiguration verwendet. Es wurde der Algorithmus Proximal Policy Optimization (PPO) verwendet. PPO ist ein On-Policy- Algorithmus, der einen bestimmten Satz von Daten sammelt, bevor er mit der nächsten Aktualisierung der Richtlinie fortfährt. Issac Gym ermöglicht es uns, alle Operationen durchzuführen und alle Daten auf der GPU zu speichern, was zu einer effizienteren Berechnung führt. Die erhaltene Strategie wurde mit 4096 Robotern und einer Stapelgröße von 98304 trainiert, die wir für 1500 Strategieaktualisierungen in ca. 20 Minuten trainieren. Jeder Roboter wurde in einer Umgebung mit unterschiedlichen Parametern (wie Reibung) trainiert. Das Training wurde auf einer einzigen GPU- Workstation für alle Roboter gleichzeitig durchgeführt. Um die Anpassungsfähigkeit der trainierten Strategie an die simulierte reale Übertragung zu gewährleisten, haben wir den Bodenreibungskoeffizienten randomisiert, den Beobachtungen Rauschen hinzugefügt und die Roboter während der Episode zufällig geschoben, damit sie eine stabilere Haltung einnehmen. Jeder Roboter hat einen Reibungskoeffizienten, der gleichmäßig im Bereich [0,5, 1,25] verteilt ist. Die Schubzyklen finden alle 10 s statt. Die Basis der Roboter wird dann auf  $\pm 1$  m/s in x- und y-Richtung beschleunigt. Die

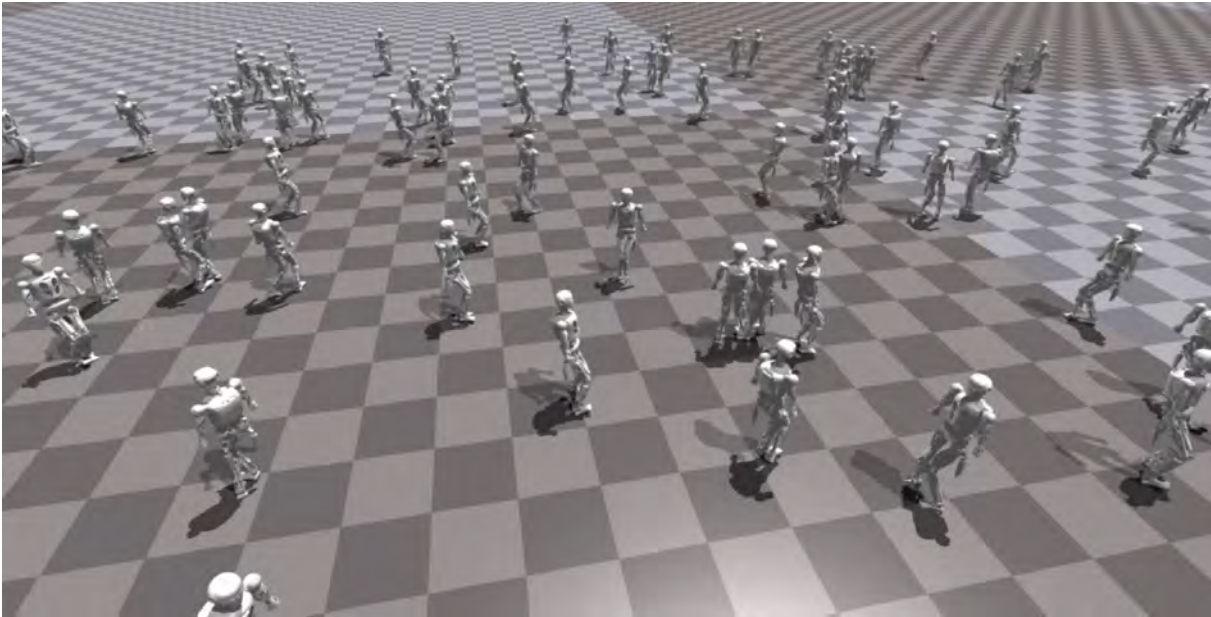


Abbildung II.27: Die resultierende Strategie wurde an einer Population von 100 RH5-Robotern in ihren jeweiligen Umgebungen getestet. In diesem Bild sind RH5-Roboter zu sehen, die im dynamischen Simulator von Issac Gym in alle Richtungen laufen.

für diese Politik vorgesehenen Belohnungsfunktionen sind in den Tabellen II.4 and II.5 aufgeführt.

Die resultierende Simulation der Laufstrategie ist in Fig. II.27 zu sehen. Die Strategie wurde an einer Stichprobe von 100 Robotern getestet. Jeder Roboter befindet sich in seiner randomisierten Umgebung, wie oben für die Trainingsphase beschrieben. Die humanoiden Roboter waren in der Lage, in 20 Minuten Training auf einem einzigen GPU von einem Ende zum anderen zu laufen. Das Gehen erfolgt derzeit in alle Richtungen auf einer ebenen Fläche. In Zukunft wollen wir dieses Ergebnis erweitern, um das Gehen für den komplexen humanoiden Roboter RH5 auf unebenem Gelände zu erlernen und die Lücke zwischen Simulation und Realität zu schließen, indem wir die resultierende Bewegung auf dem realen System validieren.

**Simulation der Starrkörperdynamik (AP C.5):** Dieses Arbeitspaket besteht aus der Simulation der Starrkörperdynamik des serienparallelen Hybrid-Humanoiden RH5. Da Hybrid Robot Dynamics (HyRoDyn) [33] in minimalen Koordinaten arbeitet, sind die resultierenden Bewegungsgleichungen gewöhnliche Differentialgleichungen (ODE) im Gegensatz zu algebraischen Differentialgleichungen (DAE) in maximalen Koordinaten (z. B. Open Dynamics Engine or ADAMS MSC). Die Fähigkeiten der HyRoDyn-Softwarebibliothek wurden durch die Integration von Runge-Kutta- und Newton-Euler-Integrations schemata erweitert, so dass sie die für die Simulation von Mehrkörpersystemen erforderlichen gewöhnlichen Differentialgleichungen lösen kann. Abb. II.28 zeigt die Simulation des freien Falls des RH5-Beins unter Schwerkraft mit den Anfangsbedingungen  $\mathbf{q}_i = \dot{\mathbf{q}}_i = \mathbf{0.0}$ . Alle Schleifenbeschränkungen werden bei dieser Simulation analytisch berücksichtigt, und es ist kein zusätzliches numerisches Stabilisierungsverfahren erforderlich.

Um die Allgemeingültigkeit von HyRoDyn zu verbessern, wurde ein neuartiger hybrider numerisch-analytischer Ansatz zur Lösung von Schleifenschließungsbeschränkungen entwickelt. Die zentrale Idee besteht darin, die Modularität des Roboterdesigns zu nutzen, um den analytischen Schleifenschluss für

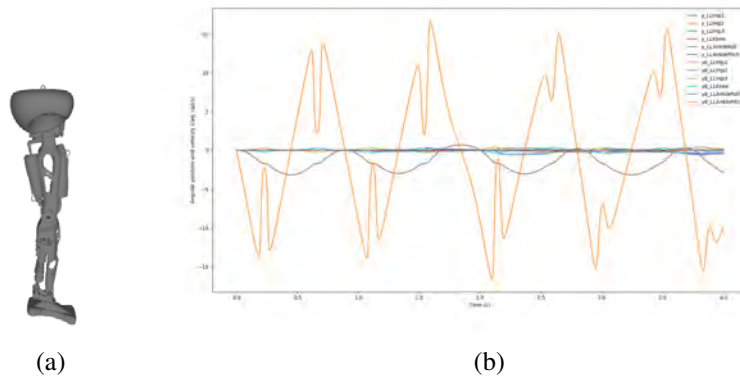
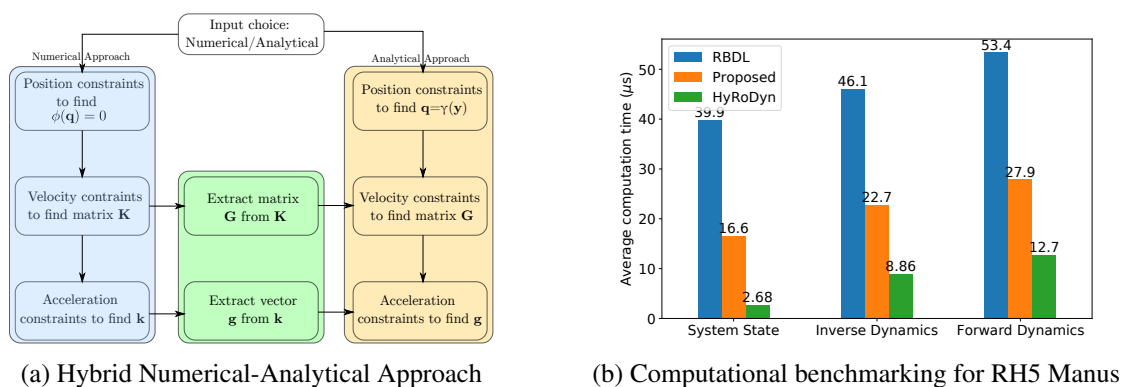
Abbildung II.28: Free-fall simulation of RH5 leg under gravity with initial conditions  $\mathbf{q}_i = \dot{\mathbf{q}}_i = \mathbf{0.0}$ 

Abbildung II.29: Hybrider numerisch-analytischer Ansatz und rechnerische Benchmarks

die bekannten Teilmechanismen und den numerischen Schleifenschluss für Teilmechanismen, für die keine analytischen Lösungen verfügbar sind, zu kombinieren. Letzteres wird durch die Umwandlung der impliziten Nebenbedingungen in explizite Beschränkungen (siehe Abb. II.29a). Außerdem wurde der Algorithmus für die Vorwärtsdynamik erheblich verbessert und vermeidet die explizite Inversion der Massenmatrix. Stattdessen wird der seriell-parallele Hybridroboter mit Hilfe von Constraint Embedding in ein serielles System umgewandelt und der rekursive Gelenkkörper-Algorithmus (ABA), der ursprünglich von Featherstone [24] für serielle Systeme vorgeschlagen wurde, wird dann auf das umgewandelte System angewendet. Diese beiden neuen Entwicklungen bieten einen zusätzlichen Vorteil in Bezug auf die Berechnungseffizienz, wie in Abb. 29b gezeigt. Einzelheiten finden sich in unserer Veröffentlichung [8].

### II.3 Wichtigste Positionen des zahlenmäßigen Nachweises

Siehe Anhang: Zahlenmäßiger Verwendungsnachweis

### II.4 Notwendigkeit und Angemessenheit der geleisteten Arbeit

Der Verlauf der Arbeit im Projekt VeryHuman wurde vom internationalen wissenschaftlichen Beirat (*scientific advisory board*, SAB) des DFKI regelmäßig kontrolliert. Grundlage dieser Kontrolle waren die turnusmäßig eingereichten schriftlichen Projektfortschrittsberichte und der Endbericht, sowie die Präsentationen.

tionen des Projektes auf den SAB-Sitzungen im Projektverlauf. Anlässlich der Abschlußpräsentation des Projektes am 2. Oktober auf der 65. Sitzung des SAB in Osnabrück urteilte der SAB:

*This project, from the two DFKI groups CPS and RIC, aims to bridge the gap between symbolic methods and robotics for the formal verification of dynamic systems. The report for this third and last year responds to previous SAB comments and indicates the new results for this period (Jan. 2023 – May 2024). Notably among these are the following developments:*

- *Symbolic specification of dynamic movements using Lyapunov functions and sampling-based calculations of region-of-attraction for linear quadratic regulators; machine-checked proof of the stability of a quadratic controller for the inverted pendulum using Lyapunov.*
- *Significant progress on the RH5 robot with a stable and robust dynamic walking at varying speeds and step lengths using external motion capture to provide posture state to the whole-body controller.*
- *Use of a reinforcement learning software to successfully synthesize a walking policy tested in simulation on a flat surface with symbolically expressed guidance for RL to improve the learning rate and robustness of the learned policy.*
- *Overall, the project has made good progress on the problems of robotics design, analysis, and verification of stability and safety properties using formal specifications. It has successfully demonstrated the value of hybrid models and the integration of the project's developed methods in a real physical robot. The results have been well received through several publications in very good venues (4 journal and 8 conference papers). SAB commend the team for the good achievements of this project and look forward to further dissemination of V&V methods in robotics and dynamic systems and their further development in followup projects such as ActGPT.*

Der SAB beurteilte also die geleisteten Arbeiten als sehr erfolgreich (*commend the team for good achievements*).

Ferner wurden jenseits der beantragten Mittel keine zusätzlichen Ressourcen für die Erreichung der Projektziele benötigt.

## II.5 Voraussichtlicher Nutzen und Verwertbarkeit

Sowohl die humanoide Robotik wie auch auf verstärkendem Lernen aufbauende Verfahren des maschinellen Lernens haben in den letzten Jahren einen ungeahnten Aufschwung erfahren. In vielen Bereichen, wie der Bilderkennung, Spracherkennung, Sprachverarbeitung, automatischer Übersetzung finden sich innovative und erfolgreiche Anwendungen. Der inhärente, prinzipielle Nachteil dieser Verfahren ist dabei die Problematik, dass ihre Korrektheit nicht einfach nachgewiesen werden kann.

Ein prominentes Anwendungsbeispiel ist die Mensch-Roboter-Interaktion, wie der Robot-Co-Worker, wo Roboter im industriellen Umfeld zusammen mit Menschen arbeiten, oder Serviceroboter in Haushalten, Hotel oder sogar Krankenhäusern. Auch hier möchte man die mächtigen Fähigkeit von verstärkendem Lernen nutzen, gleichzeitig aber harte Sicherheitsaussagen über das Verhalten des Systems machen können.

In diesen Fällen muss eine belastbare, nachvollziehbare Sicherheitsaussage über das System getroffen werden, ohne dass ein formales Modell des Systems (und damit dessen Sicherheit) überhaupt vorliegt. Hier können die in VeryHuman entwickelten Techniken helfen, systematisch eine Sicherheitsaussage

herzuleiten, zu formalisieren, und zu verifizieren. Dies trifft insbesondere auf Robotersysteme zu, die durch verstärkendes Lernen oder ähnliche Techniken gesteuert werden (bspw. industrielle Roboter, die aus Bewegungsmustern lernen), und für die eine Sicherheitsaussage getroffen werden muss (bspw. der Manipulator trifft nie einen Werker). Auf diese sind die hier entwickelten Techniken sehr direkt anwendbar.

Insgesamt hat das Projekt also einen hochrelevanten Aspekt in einem aktuellen Forschungsfeld erfolgreich adressiert. Damit ist die wissenschaftliche und technische Anschlussfähigkeit durch zwei Aspekte gegeben: die Verifikation und der Roboter RH5.

- Das Ende des Projektes erlernte Verhalten von RH5 ist noch vergleichsweise instabil und nicht für alle Kontexte geeignet. Zu akquirierende Folgeprojekte werden diesen Gang weiter stabilisieren und verallgemeinern. Insbesondere soll untersucht werden, wie durch den Einsatz von großen Sprachmodellen (*large language models*, LLMs) das Training des Geh- und sonstigen Verhaltens von Robotern weiter verbessert werden kann.
- Auf der anderen Seite können an die Verifikation dieses konkreten erlernten Verhaltens mit der dabei entwickelten Methodologie Projekte für die Verifikation anderer Verhalten und Kontrollalgorithmen anschließen. Auf diesem Gebiet besteht noch enormer Forschungsbedarf, da die Kombination von formaler Verifikation und Kontrolltheorie noch nicht gut erforscht ist.

## II.6 Bekannt gewordener Fortschritt

Während der Projektlaufzeit wurden mehrere humanoide, zweibeinige Roboter kommerziell verfügbar. Bei diesen handelt es sich um geschlossene Systeme, die für gegebene Abläufe und Situationen programmiert werden können; offenes Lernen, wie es in diesem Projekt für RH5 praktiziert wurde, ist dabei nicht vorgesehen. Die Programmierung dieser Systeme ist nicht einsehbar, und kann daher weder mit unserem Ansatz verglichen noch wie in VeryHuman formalisiert und verifiziert werden. Allerdings unterstreichen diese Systeme eine zunehmende Kommerzialisierung und insbesondere Vergrößerung des Marktes für humanoide Roboter.

## II.7 Veröffentlichungen

### II.7.1 Publikationsübersicht

Im Folgenden sind die im Verlauf des Projektes unter Beteiligung von Projektmitarbeitern entstandenen Publikationen aufgeführt. Insgesamt sind 13 Publikationen entstanden.

Zeitschriftenartikel	4
Begutachtete Konferenzen	8
Nichtbegutachtete Workshops und Pre-Prints	1

## II.8 Publikationen des Projektes

- [1] I. Bergonzani, M. Popescu, S. Kumar, and F. Kirchner. Fast dynamic walking with RH5 humanoid robot. In *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*, pages 1–8. IEEE, 2023.

- 
- [2] M. Boukheddimi, R. Kumar, S. Kumar, J. Carpentier, and F. Kirchner. Investigations into exploiting the full capabilities of a series-parallel hybrid humanoid using whole body trajectory optimization. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10433–10439. IEEE, 2023.
  - [3] E. N. D. Coskun, K. c. Coskun, C. Lüth, and D. Hutter. Fast and accurate estimation of regions-of-attraction for LQR-controlled systems. Submitted to ICCAD 2024.
  - [4] J. Eßer, S. Kumar, H. Peters, V. Bargsten, J. d. G. Fernandez, C. Mastalli, O. Stasse, and F. Kirchner. Design, analysis and control of the series-parallel hybrid rh5 humanoid robot. In *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, pages 400–407, 2021.
  - [5] M. Gleirscher, R. Massoud, D. Hutter, and C. Lüth. How deduction systems can help you to verify stability properties. In *63rd IEEE Conference on Decision and Control (CDC-2024), December 16-19, Milano, Italy, 2024*. Preview published as <https://arxiv.org/abs/2404.10747>.
  - [6] L. Gross, L. Maywald, S. Kumar, F. Kirchner, and C. Lüth. Analytic estimation of region of attraction of an LQR controller for torque limited simple pendulum. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 2695–2701, 2022.
  - [7] D. Harnack, C. Lüth, L. Groß, S. Kumar, and F. Kirchner. Deriving rewards for reinforcement learning from symbolic behaviour descriptions of bipedal walking. In *62nd IEEE Conference on Decision and Control (CDC). IEEE Conference on Decision and Control (CDC-2023), December 13-15, Marina Bay Sands, Singapore, 2023*. Preview published as <https://arxiv.org/abs/2312.10328>.
  - [8] R. Kumar, S. Kumar, A. Müller, and F. Kirchner. Modular and hybrid numerical-analytical approach — a case study on improving computational efficiency for series-parallel hybrid robots. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3476–3483, 2022.
  - [9] S. Kumar and A. Müller. Nth order analytical time derivatives of inverse dynamics in recursive and closed forms. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1918–1924, 2021.
  - [10] A. Mueller and S. Kumar. Closed-form time derivatives of the equations of motion of rigid body systems. *Multibody System Dynamics*, 53(3):257–273, 2021.
  - [11] M. Popescu, D. Mronga, I. Bergonzani, S. Kumar, and F. Kirchner. Experimental investigations into using motion capture state feedback for real-time control of a humanoid robot. *Sensors*, 22(24), 2022.
  - [12] C. Stoeffler, A. del Rio Fernandez, H. Peters, M. Schilling, and S. Kumar. Kinematic analysis of a novel humanoid wrist parallel mechanism. In O. Altuzarra and A. Kecskeméthy, editors, *Advances in Robot Kinematics 2022*, pages 348–355, Cham, 2022. Springer International Publishing.
  - [13] F. Wiebe, J. Babel, S. Kumar, S. Vyas, D. Harnack, M. Boukheddimi, M. Popescu, and F. Kirchner. Torque-limited simple pendulum: A toolkit for getting familiar with control algorithms in underactuated robotics. *Journal of Open Source Software*, 7(74):3884, 2022.

# Literaturverzeichnis

- [14] Bernardo Aceituno-Cabezas, Carlos Mastalli, Hongkai Dai, Michele Focchi, Andreea Radulescu, Darwin G Caldwell, José Cappelletto, Juan C Grieco, Gerardo Fernández-López, and Claudio Semini. Simultaneous contact, gait, and motion planning for robust multilegged locomotion via mixed-integer convex optimization. *RA-L*, 3(3):2531–2538, 2017.
- [15] R. Alur, T.A. Henzinger, and P.-H. Ho. Automatic symbolic verification of embedded systems. In *1993 Proceedings Real-Time Systems Symposium*, pages 2–11, December 1993.
- [16] Rohan Budhiraja, Justin Carpentier, Carlos Mastalli, and Nicolas Mansard. Differential dynamic programming for multi-phase rigid contact dynamics. In *Humanoids*. IEEE, 2018.
- [17] Stéphane Caron, Quang-Cuong Pham, and Yoshihiko Nakamura. Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench cone for rectangular support areas. In *ICRA*. IEEE, 2015.
- [18] Debora Clever, Malin Kopitzsch, Martin Felis, and Katja Mombaur. Inverse optimal control based identification of optimality criteria in whole-body human walking on level ground. In *Geometric and Numerical Foundations of Movements*, volume 117 of *Springer Tracts in Advanced Robotics*, pages 1192–1199, 06 2016.
- [19] Hongkai Dai, Benoit Landry, Lujie Yang, Marco Pavone, and Russ Tedrake. Lyapunov-stable neural-network control. In *Proceedings of Robotics: Science and Systems*, Virtual, July 2021.
- [20] Frank Dylla, Jae Hee Lee, Till Mossakowski, Thomas Schneider, André Van Delden, Jasper Van De Ven, and Diedrich Wolter. A Survey of Qualitative Spatial and Temporal Calculi: Algebraic and Computational Properties. *ACM Computing Surveys*, 50(1):7:1–7:39, April 2017.
- [21] J. Engelsberger, A. Werner, C. Ott, B. Henze, M. A. Roa, G. Garofalo, R. Burger, A. Beyer, O. Eiberger, K. Schmid, and A. Albu-Schäffer. Overview of the torque-controlled humanoid robot toro. In *Humanoids*, 2014.
- [22] Julian Esser. Highly-dynamic movements of a humanoid robot using whole-body trajectory optimization. Master’s thesis, University of Duisburg-Essen, Nov 2020.
- [23] Shamel Fahmi, Carlos Mastalli, Michele Focchi, and Claudio Semini. Passive whole-body control for quadruped robots: Experimental validation over challenging terrain. *RA-L*, 4(3):2553–2560, 2019.
- [24] Roy Featherstone. *Rigid body dynamics algorithms*. Springer, 2014.
- [25] Martin L. Felis and Katja Mombaur. Synthesis of full-body 3-D human gait using optimal control methods. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1560–1566, May 2016.
- [26] C. Freksa and K. Zimmermann. On the utilization of spatial structures for cognitively plausible and efficient reasoning. In *[Proceedings] 1992 IEEE International Conference on Systems, Man, and Cybernetics*, pages 261–266 vol.1, October 1992.
- [27] Nathan Fulton, Stefan Mitsch, Jan-David Quesel, Marcus Völp, and André Platzer. KeYmaera X: An Axiomatic Tactical Theorem Prover for Hybrid Systems. In Amy P. Felty and Aart Middeldorp, editors, *Automated Deduction - CADE-25*, Lecture Notes in Computer Science, pages 527–538, Cham, 2015. Springer International Publishing.
- [28] Kevin Giraud-Esclasse, Pierre Fernbach, Gabriele Buondonno, Carlos Mastalli, and Olivier Stasse. Motion planning with multi-contact and visual servoing on humanoid robots. In *SI*. IEEE, 2020.

- 
- [29] T.A. Henzinger. The theory of hybrid automata. In *Proceedings 11th Annual IEEE Symposium on Logic in Computer Science*, pages 278–292, July 1996.
- [30] Fabian Immler. A verified ODE solver and the Lorenz attractor. *Journal of Automated Reasoning*, 61(1):73–111, Jun 2018.
- [31] Sylvain Joyeux. Rock: the robot construction kit, 2013.
- [32] Shivesh Kumar, Julius Martensen, Andreas Mueller, and Frank Kirchner. Model simplification for dynamic control of series-parallel hybrid robots - a representative study on the effects of neglected dynamics shivesh. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5701–5708, 2019.
- [33] Shivesh Kumar, Kai Alexander von Szadkowski, Andreas Müller, and Frank Kirchner. An analytical and modular software workbench for solving kinematics and dynamics of series-parallel hybrid robots. *Journal of Mechanisms and Robotics*, 12(2):1–13, 2020.
- [34] Taeyoon Lee, Patrick Wensing, and Frank Park. Geometric robot dynamic identification: A convex programming approach. *IEEE Transactions on Robotics*, PP:1–18, 12 2019.
- [35] Zhongyu Li, Xuxin Cheng, Xue Bin Peng, Pieter Abbeel, Sergey Levine, Glen Berseth, and Koushil Sreenath. Reinforcement learning for robust parameterized locomotion control of bipedal robots. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2811–2817, 2021.
- [36] Carlos Mastalli, Rohan Budhiraja, Wolfgang Merkt, Guilhem Saurel, Bilal Hammoud, Maximilien Naveau, Justin Carpentier, Sethu Vijayakumar, and Nicolas Mansard. Crocodyl: An Efficient and Versatile Framework for Multi-Contact Optimal Control. In *ICRA*, 2020.
- [37] Carlos Mastalli, Wolfgang Merkt, Josep Marti-Saumell, H. Ferrolho, Joan Sola, Nicolas Mansard, and Sethu Vijayakumar. A direct-indirect hybridization approach to control-limited ddp. *arXiv:2010.00411*, 2021.
- [38] David Mayne. A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems. *International Journal of Control*, 3(1):85–95, jan 1966.
- [39] Perry, Jaqueline. *Gait Analysis — Normal and Pathological Function*. SLACK Incorporated, Thorofare, NJ, 1992.
- [40] André Platzer. Differential Dynamic Logic for Hybrid Systems. *Journal of Automated Reasoning*, 41(2):143–189, August 2008.
- [41] André Platzer. Logics of Dynamical Systems. In *2012 27th Annual IEEE Symposium on Logic in Computer Science*, pages 13–24, June 2012.
- [42] André Platzer. *Logical Foundations of Cyber-Physical Systems*. Springer International Publishing, Cham, 2018.
- [43] Alexander Scivos and Bernhard Nebel. Double-Crossing: Decidability and Computational Complexity of a Qualitative Calculus for Navigation. In Daniel R. Montello, editor, *Spatial Information Theory*, Lecture Notes in Computer Science, pages 431–446, Berlin, Heidelberg, 2001. Springer.
- [44] Miomir Vukobratović and J Stepanenko. On the stability of anthropomorphic systems. *Mathematical biosciences*, 15(1-2), 1972.