



Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Studienarbeit Anwendungen 1

Jan Ruhnke

A Walksystem for a Quadruped Rough Terrain Robot,
Controller Concepts

Jan Ruhnke

ruhnke@enapse.de

A Walksystem for a Quadruped Rough Terrain Robot, Controller Concepts

Studienarbeit eingereicht im Rahmen der Veranstaltung Anwendungen 1

im Studiengang Master of Science Informatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer : Prof. Dr. rer. nat. Gunter Klemke
Prüfer : Prof. Dr. rer. nat. Kai von Luck

Abgegeben am 28.02.2012

Jan Ruhnke

Thema der Ausarbeitung

A Walksystem for a Quadruped Rough Terrain Robot, Controller Concepts

Stichworte

USAR – Roboter, vierbeiniges Laufsystem, Rough Terrain Robot, Controller

Kurzzusammenfassung

Diese Arbeit befasst sich mit Controllerkonzepten für einen vierbeinigen Roboter, der speziell an unebenes Gelände angepasst werden soll. Es werden einige Ansätze diskutiert und ein möglicher Controller für ein verteiltes, eventbasiertes System skizziert.

Jan Ruhnke

Title of the paper

A Walksystem for a Quadruped Rough Terrain Robot, Controller Concepts

Keywords

Quadruped robot inspired by mammals, Search and Rescue Robotics, Rough Terrain Robot, Controller

Abstract

This thesis focuses on concepts of controller for a Quadruped Rough Terrain Robot. Some concepts are discussed, and outline a possible controller for a distributed event-based system.

Inhaltsverzeichnis

1	Problemstellung.....	5
1.1	Laufsystem Project „AMEE“	5
2	Forschungsrelevanz & Forschungsumfeld	6
3	Grundbehauptung & Forschungsgegenstand	7
4	Walking Gaits & Stabilität	8
5	Kontrollerkonzepte	9
5.1	Skizze eines Hauptkontrollers	11
6	Bein/Rad Hybridsysteme.....	14
7	Zusammenfassung	14
8	Risiken	15
9	Anhang	16
	Literaturverzeichnis & Quellen	17

1 Problemstellung

In vielen Ländern häufen sich Naturkatastrophen, Unfälle und Terroranschläge, bei denen eine für Menschen gefährliche Umgebung entsteht. Für die Hilfskräfte sind zerstörte Gebäude besonders schwierig zu beurteilen, da die Situation nicht immer aus der Ferne eingeschätzt werden kann. Zu diesem Zweck werden sogenannte USAR-Roboter in die Gebäude geschickt, die ein Abbild der Innenräume erstellen und dies den Helfern übermitteln. Für diese Aufgabe müssen USAR-Roboter extrem agil sein, um sich auf unebenem Gelände bewegen zu können. Es hat sich gezeigt das Roboter mit vier Beinen eine hohe Mobilität und Effektivität ermöglichen [Ruh11]. Im Anhang (9) wird die Problemstellung weiter konkretisiert.

1.1 Laufsystem Project „AMEE“



Abbildung 1-1 Simulation des Roboters AMEE

Diese Arbeit beschäftigt sich mit möglichen Lösungen für einen *Quadruped Rough Terrain Robot* des studentischen Projekts „AMEE“ [Ruh11]. Dabei handelt es sich um einen vierbeinigen, statischen / pseudo dynamischen Laufroboter. Für die Realisierung des kompletten Roboters wird die Arbeit der Bachelor Thesis von Björn Bettzüge [Bet10] und Jan Ruhnke [Ruh11] fortgesetzt. Es werden einige Ansätze für den Hauptkontrolller des Laufsystems vorgestellt. Diese Arbeit stellt aber

nicht die Hintergründe der mechanischen Lösung dar.

Ein Hauptproblem in der Entwicklung dieses Roboters ist das sichere Laufen auf stark unebenem Untergrund, wie beispielsweise eine teilweise verschüttete Treppe. Aufgrund von erfassten Bodenprofilen und Rückmeldungen der Beine, soll ein System entwickelt werden, das die Koordination der Beinbewegungen ermöglicht und immer den gesamten Roboter in einem sicheren Zustand [Ruh11] (SSM [McG79]) hält. Hierfür werden erste Konzepte entwickelt, die auch andere Arbeiten [Bet10] berücksichtigen.

Aufgrund der großen Anzahl an Veröffentlichungen seit dem Jahr 2007 und der Vorarbeit, wird hier in Vorbereitung auf AW2 eine Selektion der Verfahren getroffen. Auch werden vorbereitend schon Lösungsansätze skizziert, da ein Gesamtüberblick viele nicht anwendbare Konzepte beinhalten würde.

2 Forschungsrelevanz & Forschungsumfeld

Die Forschung an kommerziell nutzbaren Laufsystemen begann in den 80er Jahren. Eine der erfolgreichsten Forschungsgruppen im Bereich der Bipedalen-Roboter ist das HRI (Honda Research Institute). Hier wurden Grundkonzepte, wie beispielsweise der ZMP (Zero Moment Point) realisiert. Mitte der 90er Jahre stellte Honda den Roboter ASIMO [Hon07] vor. Zu dieser Zeit entstanden auch erste konkrete Konzepte und Ideen zu vier- und sechsbeinigen USAR-Robotern¹. Im Jahr 2005 wurden erste Prototypen zu vierbeinigen Robotern veröffentlicht, die speziell für die Mobilität auf unebenem Untergrund konzipiert wurden. Besonders interessant sind das deutsche Projekt „ARAMIES“ [Spe07] und das Projekt „Big Dog“ [Bos10] aus den USA.

Das Projekt „ARAMIES“ verwendet ein Laufsystem mit elektrischen Antrieben und versuchte einen „biologischen Kontroller“ [Spe07] umzusetzen. Es wurden aber feste Laufmuster (Walking Gaits) verwendet, die nach Erfahrungen aus dem „Learning Locomotion“ [Reb08] [Bru08] [Est05] Programm nicht immer vorteilhaft sind.

Das „Learning Locomotion“* Forschungsprogramm, unter der Leitung der DARPA, brachte ab 2005 einen Schub in die Forschung der sog. „Quadruped Rough Terrain Robots“. Das Programm wurde mit erheblichen finanziellen Mitteln ausgestattet. Die Firma „Boston Dynamics“ [Bos10] wurde beauftragt, alle interessierten US-Universitäten mit einem vierbeinigen Roboter Namens „Little Dog“ auszustatten, der eine vereinfachte und verkleinerte Version des Roboters „Big Dog“ [Bos10] ist. Im Jahr 2009 wurden die finanziellen Mittel für dieses Forschungsprojekt unter dem Namen „Advanced Locomotion“ stark erweitert.

Durch die vielen Veröffentlichungen dieses Forschungsprogramms, orientiert sich das Projekt „AMEE“ vorrangig am „Learning Locomotion“ Programm. Zurzeit gibt es kein kommerziell genutztes Laufsystem. Der Prototyp eines laufenden Harvesters, erscheint aber vielversprechend (Walking Harvester, Plustech Oy [John Deere] Anhang 9). Die Arbeit von John Rebula [Reb08] enthält einen vollständigen Überblick der aktuellen Forschung.

**Offizielle Kurzfassung des „Learning Locomotion“ Programms:*

*In November of 2005, the **Defense Advanced Research Projects Agency** initiated a new robotics program, **Learning Locomotion**, designed to solve some of the key outstanding issues. The expectation was that a combination of machine learning techniques and smart development would accelerate the pace of making autonomous legged systems robust and useful. [Def05]*

¹ Das Projekt „Dante“(1993) und „Dante 2“ waren mit ihrem sechsbeinigen Laufsystem, unter Beteiligung von John Bares und William (Red) L. Whittaker ihrer Zeit voraus. (NASA Ames Research Center, San Francisco)

3 Grundbehauptung & Forschungsgegenstand

Wie in der Bachelor Thesis [Ruh11] von Jan Ruhnke bereits erörtert, wird für den Verwendungszweck des Roboters AMEE eine low-cost Lösung für extremes Gelände angestrebt. Nach jetzigem Wissenstand (Februar 2011) kann dieser Roboter unter 20.000,- Euro realisiert werden. Als Folge dieses Kriteriums muss aber folgende Behauptung aufgestellt werden.

Behauptung: Ein statischer / pseudo dynamischer Läufer kann seine Nachteile gegenüber einem dynamischen Läufer im extremen Gelände ausgleichen.

Als Maßstab für einen Vergleich wird der Roboter „Big Dog“/„Little Dog“ [Bos10] herangezogen, da dieser die Summe der Erfahrungen aus dem „Learning Locomotion“ Forschungsprogramms darstellt. Konkret stellt sich die Frage, ob der technische Aufwand für einen dynamischen Läufer nötig ist. Besonders die Konstruktion des hydraulischen Laufsystems und das extrem kritische Timing der Controller verteuern den Roboter. In einem statischen Läufer, wie „AMEE“ können kostengünstige elektrische Antriebe verwendet werden. Zudem ist das Timing in den Controllern der Beine unkritisch, da der Roboter immer in einem sicheren Zustand([Ruh11]Abs.2.5) gehalten wird. Als Nachteil kann aber angesehen werden, dass der Roboter „AMEE“ keine so hohen Laufgeschwindigkeiten, wie der Roboter „Big Dog“ erreichen kann. Dies Resultiert aus der Konstruktion von „AMEE“, die nicht in der Lage sein wird zu springen. Im Gegensatz dazu ist es „Big Dog“ sogar möglich zu galoppieren. Damit kann „Big Dog“ in ebenem Gelände erheblich schneller laufen, was im extremen Gelände wiederum Nachteile zur Folge haben kann. Der Roboter „AMEE“ kann kontrolliert den Boden auf Tragfähigkeit testen ([Ruh11]Abs. 5.3.4.4.11), wogegen „Big Dog“ ein Zeitfenster von circa 50ms im ZMP hat, um einen sicheren Punkt für den Fuß zu finden [Bos10].

Als Folge ergibt sich die Frage, welches System im realen Gelände effizienter laufen kann?

Forschungsgegenstand: Realisierung eines funktionsfähigen Prototyps des Roboters „AMEE“.

Um die oben aufgestellte Behauptung und Fragestellung zu klären, muss der komplette Roboter realisiert werden. Die mechanische Konstruktion ist bereits abgeschlossen und kann realisiert werden. Softwareseitig müssen die Hauptcontroller inkl. Stabilitätskontrolle und

Beinsynchronisationssystem entwickelt werden. Wie die Veröffentlichung „A Controller for the LittleDog Quadruped Walking on Rough Terrain“ [Reb08] beschreibt, sind Simulationen nur bedingt zielführend, da die Problemstellung zu feingranular ist.

4 Walking Gaits & Stabilität

Die Laufmuster (Walking Gaits) stehen im direkten Zusammenhang mit der Stabilität eines statischen Läufers [McG79]. Es hat sich gezeigt, dass der sog. „creeping gait“ [Bru08] (Seite 379) bzw. „Save Walk“ ([Ruh11] Abs. 2.5.1), vorteilhaft in extremen Gelände ist. Dabei bleiben immer drei Beine auf dem Boden und es wird nur ein Bein in die vorgreifende Position bewegt. Sind alle vier Beine neu platziert worden, wird in einer Bewegung (bez. als „Pull“ [Ruh11]) der Robotertorso nach vorn gezogen. Der Vorteil liegt in einer hohen Stabilität des Roboters, da immer eine Dreipunktauflage existiert. Das zweite Walking Gait ist der sog. „Trot gait“ [Bru08] bzw. der „Fast Walk“ ([Ruh11] Abs. 2.5.2). Dabei befinden sich immer zwei Beine in Bewegung zur vorgreifenden Position. Das andere Beinpaar zieht währenddessen den Robotertorso nach vorn. Für diesen Laufmodus wird immer ein Vorderbein und das gegenüberliegende Hinterbein verwendet. Dieser Laufmodus eignet sich für schnelleres Laufen auf fast ebenem Untergrund. Diese beiden Laufmuster werden auch in den meisten neueren Veröffentlichungen [Reb08] [Spe07] [Bos10] [Kol09] [Adu07] verwendet und haben sich als besonders robust erwiesen. Im Gegensatz zu älteren Arbeiten zieht sich aber, dass feste Laufmuster im realen Gelände nachteilig wirken. Feste Laufmuster scheinen nur in Simulationen und Laborversuchen effizient zu sein [Reb08] [Kol09] [Zha06]. Der Wechsel zwischen den Laufmustern erfolgt entweder reaktiv über eine Fehlererkennung (Fehlritte und „pitch & roll detection“) oder deliberativ anhand der vorausschauend erfassten Geländetopologie (beispielsweise optische 3D Bodenerkennung). Für das Projekt „AMEE“ soll eine Kombination der Verfahren verwendet werden [Ruh11] [Bet12]. Um das angestrebte Laufmuster zu erzielen, wird im Hauptkontroller ein sog. „Leg Selection Controller“ verwendet (Kapitel 4). Dieser Controller berechnet für jedes Bein, die maximal machbare Schrittlänge und wählt ein Bein direkt aus. Durch einen vorgeschalteten „Gait Selection Controller“ wird, wie oben beschreiben, zwischen zwei simplen Regeln ausgewählt. Diese Regel besagt nur, wieviele Füße auf dem Boden stehen müssen. Dadruch ergeben sich zwangsläufig die angestrebten Laufmuster [Bru08] [Reb08] [Kol09]. Diese einfache Erweiterung der festen Laufmuster, macht das Laufsystem sehr robust und kann damit dynamisch auf extreme Geländeformationen bzw. Störungen reagieren. Hier zeigen sich auch erste Vorteile des statischen Läufers im extremen Gelände [Reb08].

5 Konrollerkonzepte

Für den Aufbau der Konroller gibt es zwei unterschiedliche Grundkonzepte, die sich von der Software bis in die Hardware und Mechanik auswirken.

Monolithisches Grundkonzept:

Häufig werden Roboter mit einem monolithischen Konroller (Abbildung 5-1) und ein darauf

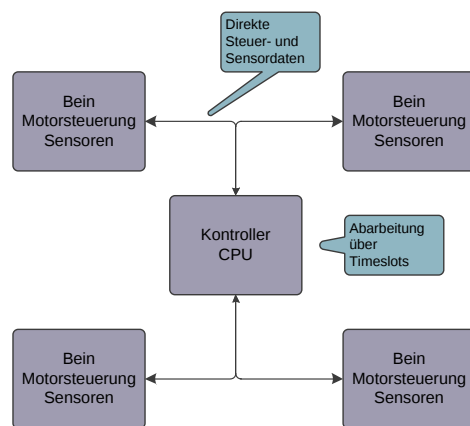


Abbildung 5-1 Monolithischer Konroller

abgestimmtes Hard- und Software Konzept entwickelt. Dabei werden alle Sensordaten und Steuerbefehle direkt von einem zentralen Konroller erfasst und bearbeitet. Die Datenverarbeitung wird mit einem Realtime-OS (RTOS) realisiert, bei dem feste Timeslots verwendet werden [Jew07]. Auch interne Berechnungen, wie beispielsweise die optische Bilderkennung, sind dem Konzept des RTOS unterworfen, was problematisch in der Abstimmung des Timings sein kann. Dieser Ansatz bildet einen Central Point of Failure und benötigt einen hohen Entwicklungsaufwand, da hier zeitkritische Aufgaben

und Prozesse mit unvorhersehbaren Laufzeiten vermischt werden [Jew07]. Um dieses Problem zu vereinfachen, können sog. Robotic-OS (ROS z.B. [Sta11]) verwendet werden die beispielweise spezielle Scheduler verwenden. Dies muss aber kritisch betrachtet werden, da ein „out of the Box“ ROS immer an die Hardware angepasst werden muss. Viele dieser ROS sind entweder auf Industrieroboter oder auf radgetriebene Roboter optimiert. Bei einem Laufsystem muss abgewogen werden, ob ein RTOS mit selbstentwickeltem Timing oder die Anpassung eines ROS schneller realisiert werden kann. Dieses Konzept minimiert den Hardwareaufwand, der aber mit einem hohen Softwareentwicklungsaufwand erkauft wird. Dies kann nur bei hohen Stückzahlen einen Kostenvorteil bringen [Jew07] und die Forschung wird stark von der Automobilindustrie vorangetrieben.

Verteiltes Grundkonzept:

Der verteilte Konroller (Abbildung 5-2) überträgt einige Teilaufgaben auf Coprozessoren. Eine bekannte Realisierung des Konzepts ist der Honda ASIMO [Hon07]. Hier werden direkte Steuerbefehle zu komplexeren Befehlen zusammengefasst. Beispielsweise regelt der Konroller nicht direkt die Antriebe, sondern übermittelt den Aktoren den anzunehmenden Winkel und die gewünschte Geschwindigkeit. Eine Rückmeldung der Sensoren erfolgt nur bei

abnormalem Verhalten oder bei erfolgreicher Abarbeitung des Befehls [Jan10]. Dieses Konzept wird auch für den „Little Dog“ verwendet [Reb08] [Kol09]. Leider sind hierzu keine

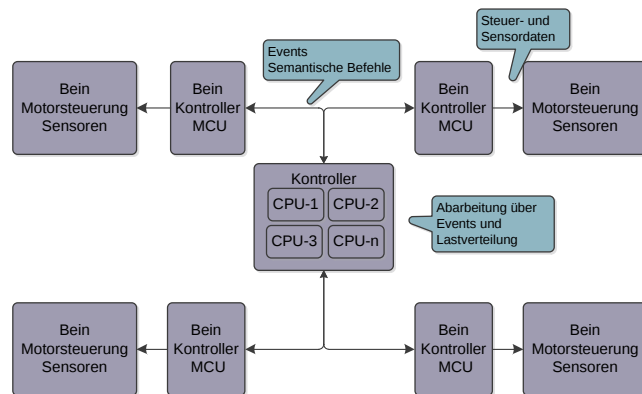


Abbildung 5-2 Verteilter Kontroller

genauen Details über die Realisierung veröffentlicht worden. Deshalb wird hier das erweiterte Konzept für das Projekt AMEE erläutert. Ein Bein des Roboters wird als eigenständiges Gerät angesehen und über semantische Befehle beauftragt. Jedes Gerät erhält dafür einen eigenen Coprozessor (MCU auf ARM Basis) mit eigenständiger Implementierung ([Ruh11]Abs.5ff). Der

Hauptkontroller hat keine Kenntnis darüber, wie z.B. das Bein die gestellte Aufgabe löst. Die komplette Steuerung und Regelung inkl. Logik (Kollisionskontrolle, inverse Kinematik usw.) ist im Bein-Coprozessor implementiert. Für gemeinsame Bewegungen der Beine, werden die Teilbewegungen in kleinere Bewegungen zerlegt und durch kurze Nachrichten synchronisiert. Ein Bein ist im Jahr 2010 komplett realisiert worden [Ruh11].

Durch dieses Konzept werden zeitkritische Tasks vom Hauptkontroller in die Coprozessoren verlagert, um kleinere und überschaubarere Problemstellungen zu erhalten. Dieses Konzept erlaubt die Nutzung von „Concurrency and Coordination Runtime“ (CCS) und „Decentralized Software Services“ (DSS) Frameworks, wie beispielsweise das „Robotics Development Studio®“ (RDS) von Microsoft. Das Framework ermöglicht es, dass z.B. die Beine mit dem Hauptkontroller über priorisierte Events, durch Queues, per Ethernet kommunizieren. Die Arbeit von Björn Bettzüge [Bet10] hat eine Verwendbarkeit für das Projekt AMEE bestätigt. Da RDS einen geringen Overhead hat, einfach zu implementieren ist und eine eigene Entwicklungsumgebung (RDS SDK) beinhaltet, erhoffen wir uns eine Entwicklungsvereinfachung und -beschleunigung des AMEE Projekts.

Weiterhin könnte ein Framework für automatische Lastverteilung verwendet werden (z.B. HPC (Microsoft High Performance Computing®)). Damit soll eine automatische Lastverteilung auf mehrere physikalische CPUs / Rechner möglich sein [Lan09]. Hier gilt es zu untersuchen, ob sich HPC für technische Systeme eignet. Der Einsatz von HPC wäre sinnvoll, da drei interne Intel Atom® Mainboards (12 logische Prozessoren) für den Roboter AMEE geplant sind. Vorteile wären die Ausfallsicherheit und die Parallelisierung von rechenintensiven Prozessen. Kritische Hard- und Softwarefehler könnten in HPC durch Failover Technologien (z.B. Hot CPU Plug n Play®, failover clustering services®) ausgeglichen werden. Theoretisch

soll der Totalausfall von zwei der drei Mainboards, nicht zu einem Versagen [Lan09] des Roboters führen.

Damit dieses System effizient arbeiten kann, müssen bei der Implementierung die einzelnen Probleme möglichst feingranular zerlegt [Reb08] werden und die Module eine schwache Abhängigkeit zueinander haben. Zudem müssen alle Schleifen und Iterationen mit dem parallelisierten Äquivalent aus .net ersetzt werden, damit die Prozesse optimal verteilt werden können.

5.1 Skizze eines Hauptkontrollers

Der folgende Hauptkontroller basiert auf dem Kontroller des „BigDogs“ (soweit veröffentlicht) [Bos10] und dem ihmc (Florida Institute For Human & Machine Cognition) [Reb08] /MIT [Kol09] Ansatz. Da ein Grundkonzept für das Projekt AMEE durch die Vorarbeit in den Bachelor-Theses [Ruh11] [Bet10] aufgestellt wurden, werden hier im Vorgriff auf AW2 bereits einige Abwandlungen mit vorgestellt. Diese Hauptkontrollerskizze beinhaltet nur die Komponenten für das Laufsystem mit Stabilitätskontrolle, ohne höhere Logik (Pfadplanung, Positionserfassung usw.).

Das Konzept [Reb08] [Kol09] sieht ein lose gekoppeltes Zwei-Layer-Konzept für den Hauptkontroller vor. Er ist in einen „*Deliberative Layer*“ (auch als Offline bezeichnet) und einen „*Reactive Layer*“ (Online) aufgeteilt. Nur der *Reactive Layer* kommuniziert direkt mit den Aktoren und Sensoren des Roboters. Die Module und Kontroller in den Layer werden *nicht* durch Events angestoßen, sondern berechnen getaktet mehrmals pro Sekunde das optimale Verhalten des Laufsystems. Durch einen Event wird nur die zurzeit optimale Lösung an die Beinkontroller übermittelt. Diese Events werden durch erfolgreiches Erreichen der Fußposition oder einen Fehler ausgelöst. Mögliche Fehler sind beispielsweise, dass der Fuß keinen Halt findet, der Roboter unerwartet kippt oder sonstige ungeplante Ereignisse. Durch die fast kontinuierliche Berechnung der optimalen Fußpositionen im *Deliberative Layer* kann der Roboter sehr schnell reagieren, da diese nur noch an die Beinkontroller übermittelt werden müssen. Es wird nur ein kompletter Bewegungsablauf aller vier Beine vorausberechnet [Kol09]. Dieses Konzept steht zwar teilweise im Gegensatz zum Subsumtion Konzept [Bro87] [Rod87] von Brooks², scheint aber in den obigen Realisierungen erfolgreich zu sein. Das Einfügen eines planenden Layers wird für dynamische [Kol09] und statische [Reb08] Läufer verwendet, da auch ein statischer Läufer im Gelände schnell auf Störungen des Bewegungsablaufs reagieren muss.

² Der Kontroller des „BigDog“ [Bos10] verwirft circa zweihundert optimale Lösungen pro Sekunde.

Deliberative Layer:

Dieser Layer (Abbildung 5-3) stellt eine planende Kontrollschicht dar, die nicht auf die

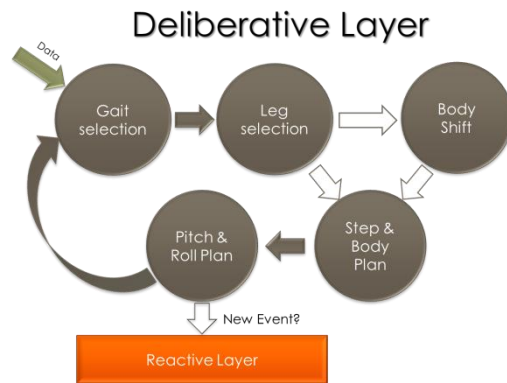


Abbildung 5-3 Deliberative Layer

Beinkontroller zugreifen kann. In der Literatur wird sie auch als „Walking Cost Plan“ [Kol09] oder einfach als „Motion Plan“ [Reb08] bezeichnet. Dieser Layer erhält seine Anweisungen von der übergeordneten Pfadplanung der höheren Logik. Er hat Zugriff auf alle Sensordaten und Planungsdaten aus einem nebenläufigen Modul, das sichere Schritte erlernt hat [Bet12]. Als erstes wird in der „Gait Selection“, wie in Abschnitt 4 dargestellt, das Laufmuster ausgewählt. Die „Leg Selection“ wählt das zu bewegende Bein mit der maximal erzielbaren Schrittweite aus (4). Kann

kein Bein mehr in die Bewegungsrichtung bewegt werden, erfolgt der sog. „Body Shift“ (in [Ruh11] als „Pull“ bezeichnet). Damit wird der Robotertorso in die Bewegungsrichtung gezogen [Reb08] [Kol09]. Der „Step & Body Plan“ führt virtuell diese Bewegung aus und prüft dabei auf mögliche Kollisionen mit anderen Beinen, dem Roboter Torso und erfassten Objekten. In diesem Modul wird ein Körpermodell bearbeitet, das auf einfachen Vektoren basiert. Es ist aus Laufzeitgründen relativ grob aufgelöst. Treten hier Fehler auf, wird der Ablauf mit verkürzter Schrittweite komplett neu angestoßen. Hierdurch wird sichergestellt, dass auch ein anderes Bein ausgewählt werden kann, wenn dieses nun eine größere Schrittweite hat. Im „Pitch & Roll Plan“ wird das zuvor erstellte Körpermodell mit einem Masseschwerpunkt-Modell (COG-Modell, Center of Gravity) erweitert ([Ruh11] Abs.2.5) und geprüft ob diese Gliederbewegung zum Kippen des Roboters führt [Reb08]. Weiterhin sollen im Projekt AMEE die auftretenden Drehmomente in den Antrieben simuliert³ [Ruh12] werden, um mögliche Totpunkte im Workspace der Beine zu erkennen. Sind alle simulierten Prüfungen erfolgreich verlaufen, werden die Fußpositionen zwischengespeichert, welche aber nur an den *Reactive Layer* übergeben werden, wenn ein neuer Event vorliegt. Dies ermöglicht dem Kontroller unverzüglich auf einen unerwarteten Event zu reagieren. Treten aber mehr als drei Fehler hintereinander auf, entsteht auch in diesem System eine Verzögerung und es werden alle Glieder während der Berechnung gestoppt. Dieser Layer simuliert zielführende Bewegungsabläufe und hält sie zwischengespeichert vor.

³ Die Betrachtung der Drehmomente ist im ihmcm/MIT Deliberative Layer nicht vorgesehen.

Reactive Layer:

Dieser Layer (Abbildung 5-4) kann direkte Befehle an die Beinkontroller senden [Reb08]

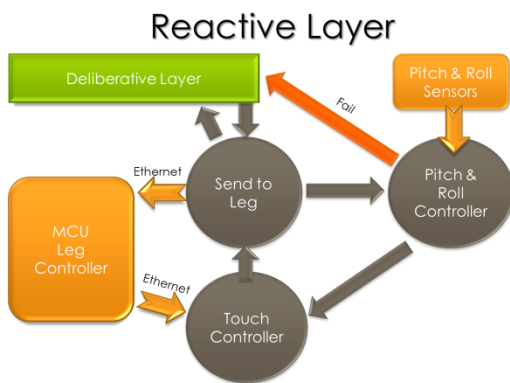


Abbildung 5-4 Reactive Layer inkl. AMEE MCU

[Kol09]. Er ist die ausführende Kontrollschicht im System und kann eigene Events auslösen. Auch diesem Layer liegen alle Sensordaten vor. Im „Send to Leg“ Kontroller werden die Befehle an die dementsprechenden externen Beinkontroller gesendet bzw. im ihmcs Ansatz [Reb08] intern übergeben. Im AMEE Projekt sollen hier auch die Beinkontroller synchronisiert werden, wenn eine Bewegung mit parallelen Bewegungen durchgeführt wird (z.B. die „Pull“ Bewegung) [Ruh11]. Im „Pitch & Roll“ Kontroller werden die Neigungswinkel des Robotertorsos überwacht. Wird hier ein maximaler

Wert überschritten, wird ein „Fail“ Event ausgelöst. Dieser Event wird im *Deliberative Layer* mit hoher Priorität bearbeitet. Es wird umgehend eine zwischengespeicherte Lösung vom *Deliberative Layer* zum *Reactive Layer* gesendet, um ein Kippen abzufangen [Reb08]. Im AMEE Projekt soll dieser Event mit minderer Priorität auch genutzt werden, um das lernende Fußpositionierungssystem [Bet12] zu „bestrafen“. Der „Touch-Controller“ (auch als Tip Detector bezeichnet [Reb08]) bearbeitet die Rückmeldungen der Beinkontroller. Da im Projekt AMEE die Beine selbständig die Tragfähigkeit des Untergrunds prüfen, braucht dieser Kontroller nur dementsprechende Events auszulösen. Im ihmcs Ansatz [Reb08] werden hier die entsprechenden Antriebsdrehmomente ausgewertet und intern in der Regelung verarbeitet. Zudem wird ein „Soft Force“ Kontroller beschrieben [Reb08] [Kol09], der den resultierenden Versatz des Roboters durch ungeplantes Aufsetzen des Fußes kompensiert. Im AMEE Ansatz wird dies durch die externen Beinkontroller [Ruh11] realisiert. Die einzelnen Kontroller arbeiten parallel und kommunizieren wiederum über Events.

Zusammenfassung der Layer:

Die oben beschriebenen Kontroller sollten sich gut für ein eventbasierendes System eignen [Kol09] [Reb08]. Durch die scharfe Problemabgrenzung kann eine lose Kopplung aller Kontroller und Module erreichen werden. Ob sich eine Verkürzung der Entwicklungszeit erreichen lässt, muss untersucht werden. Zudem ist mit diesem Ansatz zu prüfen, in wieweit sich Vorteile gegenüber anderen Kontrollerlösungen [Spe07] [Rod87] [Bru08] [Jew07] ergeben.

6 Bein/Rad Hybridsysteme

Um die Vorteile des statischen Läufers mit den Vorteilen des dynamischen Läufers zu

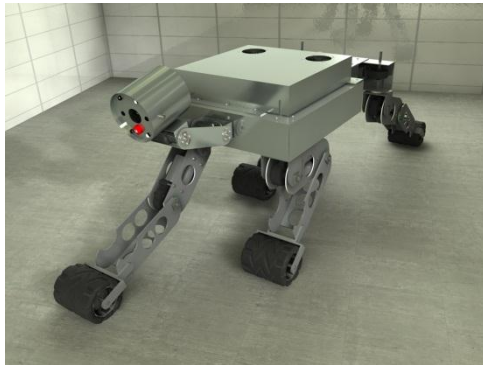


Abbildung 6-1 AMEE als Hybridsystem

kombinieren, soll als Nebenprojekt ein Bein/Rad Hybridsystem untersucht werden. Erste Erfahrungen wurden am *Chiba Institute of Technologie* in Japan mit einem vierrädrigen Fahrzeug gesammelt. Der Roboter ist in der Lage jedes Rad auf Hindernisse zu heben [Nak11]. Auch der Mars Rover „Curiosity“ [Nat11] verfügt über eine Mechanik, die einzelne Räder anheben kann. Beim Projekt „AMEE“ können die großen Füße [Ruh11] durch breite Räder ausgetauscht werden. Die Antriebe, die den Fuß

drehen, könnten die Räder treiben. Mit dieser Modifikation wären Untersuchungen ohne großen mechanischen Aufwand möglich. Der Nachteil des langsamen Laufens auf relativ ebenem Untergrund, würde mit einem Fahrmodus ausgeglichen werden. In ersten eigenen Simulationen konnte kaum eine Beeinträchtigung der Steigfähigkeit festgestellt werden.

7 Zusammenfassung

Mit einem verteilten und eventbasierenden Ansatz soll ein vierbeiniger Roboter für unebenes Gelände entwickelt und realisiert werden. Es werden auf den Ansatz optimierte Frameworks verwendet, die auch einen aktiven Support des Herstellers und eine zeitgemäße Entwicklungsumgebung bieten. Mittels Auslagerung der zeitkritischen Tasks in physikalische Sub- / Coprozessoren könnte sich der Entwicklungsaufwand verringern. Hierzu wurde schon ein Beinprototyp realisiert [Ruh11]. Durch ein Layerkonzept mit planendem und reaktivem Layer, sollen die Reaktionszeiten des Systems in kritischen Situationen verringert werden. Mit einer scharfen Abgrenzung der einzelnen Problemstellungen und die Zerlegung in einzelne Module und Kontroller, soll eine gute Verteilung auf einem Multicore- / Clustersystem ermöglicht werden. Mit diesen Konzepten wurden in den Referenzprojekten „Big Dog“ [Bos10] und „LittleDog“ [Kol09] [Reb08] sehr gute Ergebnisse erzielt. Das Projekt AMEE soll die Module und Kontroller noch stärker verteilen und schärfer abgrenzen.

Uns ist bewusst, dass durch den militärischen Hintergrund von „Big Dog“ nicht alle Implementierungsdetails veröffentlicht wurden.

8 Risiken

Für das Projekt „AMEE“ bestehen folgende Risiken:

1. Die finanziellen Mittel reichen nicht für eine Realisierung des Roboters aus.
Mögliche Lösung: Aktive Suche nach Sponsoren. Der Roboter könnte simuliert werden [Ruh12], was aber nach Ansicht einiger Wissenschaftler nicht zielführend ist [Reb08].
2. Dieses Projekt hat einen relativ großen Umfang. Es wäre möglich, dass die Studienzeit nicht ausreichend ist um den Roboter zu realisieren.
Mögliche Lösung: Es sind ca. 90% der mechanischen Konstruktion abgeschlossen. Die Fertigung könnte in der HAW-Zentralwerkstatt erfolgen. Die Softwarekonzepte sind auf eine schnelle Entwicklung abgestimmt. Die Vorarbeiten sind gut dokumentiert und könnten notfalls von anderen Studenten fortgesetzt werden.
3. Unsere Konzeption verletzt Patente. Dies ist unbedenklich, solange der Roboter nicht vermarktet wird.
4. Der Lastausgleich in einem Cluster erzeugt zu viel Overhead, wodurch andere Nachrichten im Bussystem/Netzwerk behindert oder verdrängt werden. Mögliche Lösung wäre ein Netzwerklastaussgleich oder die Verwendung von QoS.
5. Die Synchronisation der Beine, bei gemeinsamen Bewegungen, hat eine zu hohe Laufzeit. Mögliche Lösung wäre eine extra physikalische Synchronisationsleitung zwischen den Beinkontrollern. Dazu würde ein Beinkontroller als Koordinator des Laufsystems dienen.

9 Anhang

Die Problemstellung und der Zweck der Forschung wurden durch Marc Raibert, Kevin Blankespoor, Gabriel Nelson und Rob Playter sehr gut beschrieben.

Less than half the Earth's landmass is accessible to wheeled and tracked vehicles, yet people and animals can go almost anywhere on Earth. This situation motivates the development of robot vehicles that use legs for their locomotion, thereby embracing nature's mobility solution. The goal is to achieve animal-like mobility on rough and rugged terrain, terrain too difficult for any existing vehicle. [Bos10]

Whereas the Leg Laboratory robots did a good job of demonstrated the feasibility of dynamically balanced legged systems, they had two primary limitations that would need to be addressed to build practical legged vehicles. One is the need for on-board power so the robot could operate in the field without hoses and wires. Another is the need for control algorithms that provide locomotion and stability on rough terrain. We now turn to BigDog, a self-contained quadruped robot that uses many of the ideas and concepts of the Leg Laboratory robots, but also addresses the practical problems of onboard power and rough-terrain controls in order to move toward practical legged vehicles. [Bos10]

In Abbildung 9-1 werden die erwähnten Roboter dargestellt.

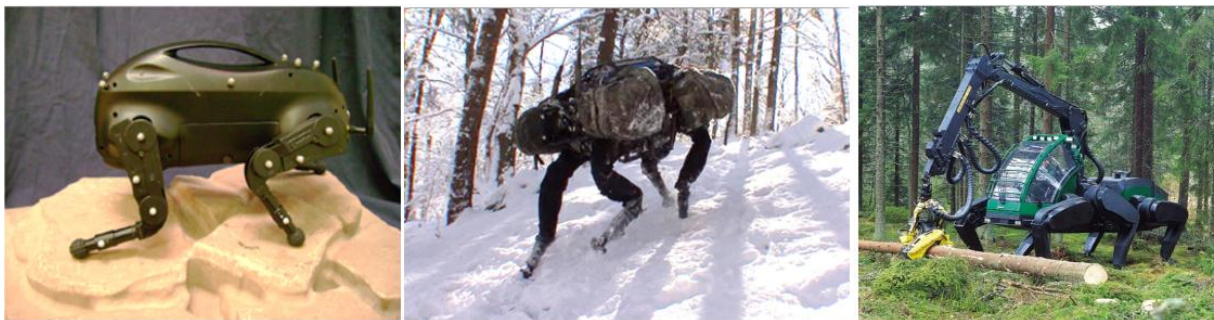


Abbildung 9-1 Little Dog(links), Big Dog(Mitte [Bos10]), Walking Harvester(rechts)[Plustech Oy, John Deere]

Literaturverzeichnis & Quellen

[Adu07] **Adukuzhiyil, Anish, Singh, Harshit and Vantimitta, Pavani. 2007.** *Robot Motion for Obstacle Negotiation*. [PDF] Stanford USA : Stanford University, 2007. IEEE Int. Conf. Robotics and Automation.

[Bet12] **Bettzüge, Björn. 2012.** *Foothold Selection for Quadruped Locomotion*. [PDF] Hamburg, Germany : HAW-Hamburg, 2012.

[Bet10] —. **2010.** *Machbarkeitsprüfung zur Entwicklung von SW-Anwendungen mit MS-Robotics Developer Studio für das Robocup Rescue Szenario*. [PDF] s.l., Hamburg : HAW Hamburg, Technische Informatik, Juli 2010.

[Bos10] **Boston Dynamics Inc. 2008, 2010.** *BigDog Overview*. [Internet] 78 Fourth Avenue, Waltham, MA, 02451-7507, US : Boston Dynamics Inc., 2008, 2010. <http://www.bostondynamics.com>.

[Bro86] **Brooks, Rodney A. 1986.** *A robust layered control system for a mobile robot*. [PDF] Massachusetts, USA : Journal of IEEE, 1986. ISSN: 0882-4967.

[Rod87] —. **1987.** *Asynchronous distributed control system for a mobile robot*. [PDF] Massachusetts, USA : MIT, 1987.

[Bro87] —. **1987.** *Planning is just a way of avoiding figuring out what to do next*. [PDF] Massachusetts, USA : Massachusetts Institute of Technology, 1987.

[Bru08] **Bruno Siciliano, Oussama Khatib (Eds.). 2008.** *Handbook of Robotics*. 80125 Napoli, Italy & CA 94305-9010, USA : Springer Science+Business Media, 2008. e-ISBN: 978-3-540-30301-5.

[Che09] **Cherouvim, Nicholas and Papadopoulos, Evangelos. 15 June 2009.** *Novel Energy Transfer Mechanism in a Running Quadruped Robot with One Actuator per Leg*. [PDF] 9 Heron Polytechniou Street, 15780 Athens, Greece : Department of Mechanical Engineering, National Technical University of Athens,, 15 June 2009. *Advanced Robotics* 24 (2010) 963–978.

[Def05] **Defense Advanced Research Projects Agency. 2005.** *Learning Locomotion*. USA : Government USA, 2005.

- [Est05] **Estremera, Joaquin and Gonzalez de Santos, Pablo. 2005.** *Generating continuous free crab gaits for quadruped robots on irregular terrain.* s.l. : IEEE Transactions on Robotics, 2005. 21(6):1067 – 1076.
- [Hon07] **Honda Motor Co., Ltd. September 2007.** *ASIMO Technical Information.* [PDF] s.l. : Honda Motor Co., Ltd., September 2007. kA.
- [Jan10] **Janssen, Herbert, Principal Scientist. 2010.** Personal Communication Honda ASIMO. [EMail/Phone]. Honda Research Institute Europe GmbH, Carl-Legien-Strasse 30, 63073 Offenbach/Main, Germany : s.n., 2010.
- [Jew07] **Jewgenij Botaschanjan, Alexander Gruler, Alexander Harhurin, Leonid Kof,. 2007.** *Towards Modularized Verification of Distributed Time-Triggered Systems.* [PDF] D-85748, Garching bei München, Germany : Institut für Informatik, TU München, BMW Group Research and Technology, 2007.
- [Kim] **Kimura, H., et al.** *Towards 3d adaptive dynamic walking of a quadruped robot on irregular terrain by using neural system model.* s.l. : 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems.
- [Kol09] **Kolter, J. Zico, Rodgers, Mike P. and Andrew, Y. 2009.** *A Control Architecture for Quadruped Locomotion Over Rough Terrain.* [PDF] CA 94305 USA : Computer Science Department, Stanford University, Stanford, 2009.
- [Lan09] **Lantz, Steven R. 2009.** *Cornell Uses High-Performance Computing to Speed Research, Extend Software Access, Case Study .* [PDF] NYC, USA : Cornell University, 2009.
- [McG79] **McGhee, R.B. and Iswandhi, G.I. 1979.** *Adaptive locomotion of a multi-legged robot over rough terrain.* [PDF] s.l. : IEEE Transactions on Systems, Man and Cybernetics, 1979. SMC-9(4):176 – 82.
- [Mur08] **Murphy, Robin R., et al. 2008.** Search and Rescue Robotics. [book auth.] Bruno Siciliano and Khatib Oussama. *Springer Handbook of Robotics.* Berlin : Springer, 2008, pp. 1151-1173.
- [Nac10] **Nachtigall, Werner. 2010.** *Bionik als Wissenschaft.* Heidelberg : Springer-Verlag, 2010. ISBN 978-3-642-10319-3 .
- [Nak11] **Nakajima, Shuro . 2011.** Walking Wheels. *Video.* Japan : Chiba Institute of Technology, 2011.

- [Nat11] **National Aeronautics and Space Administration (NASA). 2011.** *MSL Fact Sheet, Mars Science Laboratory*. [PDF] Pasadena, California : Jet Propulsion Laboratory, California Institute of Technology, 2011. JPL 400-1416 3/11.
- [Reb08] **Rebula, John R., et al. 2008.** *A Controller for the LittleDog Quadruped Walking on Rough Terrain*. [PDF] Florida 32502, USA : Florida Institute for Human and Machine Cognition, 2008. jrebula@alum.mit.edu.
- [Ruh11] **Ruhnke, Jan. 2011.** *Entwicklung und Realisierung eines vierbeinigen USAR-Roboter-Laufsystems*. [PDF] Hamburg, Germany : HAW-Hamburg Dep. Technische Informatik, Juni 02, 2011. Bachelor Arbeit.
- [Ruh12] —. **2012.** *Simulation der Drehmomente in einem Roboter Bein, Ausarbeitung Modellierung Technische Systeme*. HAW-Hamburg : s.n., 2012.
- [Sch10] **Schröder, Dierk. 2010.** *Intelligente Verfahren, Identifikation und Regelung nichtlinearer Systeme*. München : Springer, 2010. ISBN 978-3-642-11397-0.
- [Shk10] **Shkolnik, Alexander, et al. 12.2010.** *Bounding on Rough Terrain with the LittleDog Robot*. [PDF] MASSACHUSETTS USA : MASSACHUSETTS INST OF TECH , 12.2010. DOI: 10.1177/0278364910388315 The International Journal of Robotics Research.
- [Spe07] **Spenneberg, Dirk . 2007.** <http://robotik.dfki-bremen.de>. *ARAMIES*. [Online] April 1, 2007. [Cited: Juni 27, 2011.] <http://robotik.dfki-bremen.de/de/forschung/projekte/weltraumrobotik/aramies.html>.
- [Sta11] **Stanford Artificial Intelligence Laboratory. 2011.** *ROS (Robot Operating System)*. [Online] Stanford Artificial Intelligence Laboratory, 03 09, 2011. [Cited: 02 12, 2012.] <http://www.ros.org/wiki/ROS>.
- [Wet96] **Wettergreen, David. 1996.** *Gait Generation for Orthogonal Legged Robots*. [PDF] Riverside, CA 92505-3410, USA : National Aeronautics and Space Administration (NASA), 1996.
- [Zha06] **Zhang, Lei, Honda, Yoshinori and Inoue, Kousuke. 2006.** *Omnidirectional Static Walking of a Quadruped Robot*. [PDF] Japan : Ibaraki University, Ritsumeikan University, Computron Corporation, 2006. Journal of Robotics and Mechatronics Vol.18 No.1, 2006.

