



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

# The Evolution of Movement

Torben Wallbaum  
Anwendungen 2 - Ausarbeitung

# The Evolution of Movement

## Torben Wallbaum

Ausarbeitung eingereicht im Rahmen von Anwendungen 2  
im Studiengang Master Informatik  
Department Informatik  
in der Fakultät Technik und Informatik  
der Hochschule für Angewandte Wissenschaften Hamburg

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

Abgegeben am 29. August 2010

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>4</b>
<b>2</b>	<b>Systemübersicht</b>	<b>5</b>
2.1	Central Pattern Generator . . . . .	5
2.2	Evolutionäre Algorithmen . . . . .	6
2.3	Model . . . . .	7
2.4	Systemaufbau . . . . .	7
<b>3</b>	<b>Natural Motion - Reil and Husbands</b>	<b>8</b>
3.1	Evolution eines zentralen Mustergenerators . . . . .	8
3.1.1	Central Pattern Generator . . . . .	8
3.1.2	Evolutionäre Algorithmen . . . . .	9
3.2	Euphoria . . . . .	10
3.3	Endorphin . . . . .	11
<b>4</b>	<b>Multi-legged Robots - Valsalam und Miikkulainen</b>	<b>12</b>
4.1	Multi-legged Locomotion . . . . .	12
<b>5</b>	<b>From Swimming to Walking - Ijspeert et al</b>	<b>15</b>
5.1	Aufbau . . . . .	15
<b>6</b>	<b>Zusammenfassung</b>	<b>17</b>
<b>7</b>	<b>Ausblick</b>	<b>18</b>
	<b>Literaturverzeichnis</b>	<b>19</b>
	<b>Abbildungsverzeichnis</b>	<b>20</b>

# 1 Einleitung

Charakter- und Objektanimation stellen Animatoren zuweilen vor vor komplexe und zeit-intensive Herausforderungen. In Zeiten immer komplexerer Animationsformen sind die Anforderungen an Detailtreue und Perfektion kaum noch zu erfüllen. Selbst mit ausreichenden Ressourcen können Animationen nicht in aller Detailtiefe erzeugt werden. Im Bereich der Spieleindustrie wird dieses Problem oft deutlich- zum Beispiel bei modernen Next-Gen Spielen. So wirken Animationen abgehackt, stark repetitiv oder gar nur für einige ausgewählte Aktionen vorhanden. Abhilfe kann hier das, Motion Capturing schaffen. Hiermit werden die Bewegungen realer Personen erfasst und dafür genutzt, eine virtuelle Figur ausdrucksstark und detailgetreu zu animieren.

Diese Technik stößt jedoch zuweilen an ihre Grenzen. So ist zum einen die benötigte Ausrüstung, sofern von guter Qualität, in einem Preisklasse von mindestens 5.000 - 10.000 Euro anzusiedeln, zum anderen ist die Immobilität der gesamten Anlage oft ein Problem. Einige Anlagen kann man so zum Beispiel nicht im Freien nutzen. Ein weiteres Problem ist die eingeschränkte Durchführbarkeit bestimmter Aktionen. So sind gefährliche oder umständliche Bewegungsabläufe teils nicht von den agierenden Personen ausführbar. Auch können - besonders in Sportspielen - nicht alle Bewegungen berücksichtigt werden. Bei Fußball- oder Footballsimulationen wird zum Beispiel lediglich eine bestimmte Anzahl von Bewegungen eingefangen und dann mehrfach im fertigen Spiel wiederverwendet.

Diese Ausarbeitung beschäftigt sich mit Techniken, die ein Erlernen von Bewegungsabläufen mittels evolutionärer Ansätze ermöglicht. Dieses System steuert einen Rigid-Body mittels einer Kombination aus künstlichen Muskeln und Nerven und ermöglicht so eine Animation bis in kleinste Detailstufen. Die Basis bildet ein sogenannter Central Pattern Generator (CPG), welcher in natürlicher Form innerhalb der Wirbelsäule von Tieren zu finden ist. Durch diesen wird eine Oszillation erzeugt, die die Bewegung für vorhandene Gelenke vorgibt. Aufgrund der umfangreichen Parameter dieses Systems, bedient man sich der evolutionären Algorithmen, um dieses Optimierungsproblem in endlicher Zeit anzunähern.

Der Inhalt dieser Arbeit gliedert sich wie folgt: Kapitel 2 bietet einen Einblick in den grundsätzlichen Aufbau und die verwendeten Technologien zur Umsetzung eines Systems zur evolutionären Bewegungsteuerung. In Kapitel 3 bis 5 werden Beispiele für Arbeiten auf diesem Gebiet dargestellt. Diese unterscheiden sich zuweilen sowohl im Aufbau, als auch in den gesetzten Zielen bzw. den unterschiedlichen Anforderungen. In Kapitel 6 wird eine Zusammenfassung gegeben. Zum Schluss gibt Kapitel 7 einen Ausblick auf kommende Arbeiten zum Thema. Für detaillierte Systembeschreibungen und Erklärungen wird auf weitere Publikationen verwiesen.

## 2 Systemübersicht

Ein System zur evolutionären Bewegungssteuerung lässt sich in drei Hauptbestandteile einteilen. Diese sind zum einen der Central Pattern Generator, der durch Erzeugung einer Oszillation die Bewegung steuert. Der evolutionäre Algorithmus, durch den die Parameter des CPG eingestellt werden sowie das zu bewegendes Model selbst. Dies kann sowohl einen 3D Rigid-Body darstellen, oder zum Beispiel ein Roboter sein. Im nachfolgenden wird jedes Modul beschrieben und eine Einführung in den Gesamtaufbau eines solchen Systems gegeben.

### 2.1 Central Pattern Generator

Unter dem Begriff Central Pattern Generator (CPG, dt. 'Zentraler Mustergenerator'), verbirgt sich ein Neuronales Netz welches durch biologische Vorbilder bei Tieren inspiriert ist. Die Gemeinsamkeiten mit klassischen neuronalen Netzen liegt vor allem in der Struktur und des Aufbaus des Netzes. Ein eigenständiges Lernverhalten ist bei CPGs nicht vorhanden. Stattdessen wird das Netz mittels Optimierungsalgorithmen verbessert und an seine Aufgabe angepasst. Hauptbestandteil eines CPG ist ein neuronaler Oszillator. Dieser erzeugt rhythmische Signale, ohne dabei auf externe Sensoren oder andere Eingaben angewiesen zu sein. Ein weiterer Vorteil neben der Unabhängigkeit von äußeren Reizen, ist die Fähigkeit der Selbstadaption. Diese bietet die Möglichkeit, sich an unterschiedliche Einflüsse oder Gegebenheiten anzupassen und einzustellen. So ist das biologischen Vorbild in der Lage, einen Übergang zwischen verschiedenen Gangarten oder gar einen Wechsel von einer Schwimm- zu einer Laufbewegung zu vollziehen.

Um eine Bewegung aus den erzeugten Signalen des CPGs zu erhalten, werden diese an ein künstliches System von Muskeln weitergegeben. Diese erzeugen dann Kräfte an Gelenken und können somit eine Bewegung herbeiführen. Oft wird zur Umsetzung und Steuerung der Signale in Kräfte ein Regler eingesetzt, der zum Beispiel Kräfte in Winkelstellungen für die künstlichen Gelenke umrechnet und eine entsprechende Regelung durchführt. Somit ergibt sich für jeden neuronalen Oszillator ein Freiheitsgrad (DOF).

Aufgrund seiner Einfachheit und Effizienz wird hier der CPG nach Matuoka betrachtet [2]. Hierbei kommen jeweils zwei Neuron pro neuronalem Oszillator zur Anwendung. Diese sind gegenläufig und unterdrücken sich somit gegenseitig in ihrer Erregung. Diese gegenseitige Unterdrückung ist dem Vorbild von Muskelbewegungen nachempfunden. Ein Neuron ist hierbei für die Streckung (Extensor) und das andere für die Beugung (Flexor) des künstlichen Muskels zuständig.

Die Anwendungsfälle eines solchen Oszillators sind vielfältig und wurden in der Vergangenheit in unterschiedlichsten Kontexten genutzt. So erzeugten Thorsten Reil und Phil

Husband im Jahr 2002 [3] mittel CPG und genetischen Algorithmen eine Laufbewegung für zweibeinige Rigid-Bodys innerhalb einer physikbasierten 3D-Simulation. Im Projekt von Vinod K. Valsalam and Risto Miikkulainen [4] wird die Bewegung von mehrbeinigen-Robotermodellen durch diese Verfahren generiert. Hier können vier verschiedene Gangarten erzeugt werden, welche sich in Passgang, ähnlich der Laufbewegung bei Pferden, dem Trab sowie zwei Arten des Springens unterscheiden. Ein weiteres Beispiel für die Anwendung eines CPG ist bei Auke Jan Ijspeert et al [1] zu finden. Mittels eines Salamander-ähnlichen Roboters mit einem künstlichen pendant zur natürlichen Wirbelsäule, wird hier der Übergang von Schwimm- in Laufbewegungen erzeugt.

Diese drei Beispiele aus einer Vielzahl von möglichen Anwendungen sollen in den Kapiteln 3 bis 5 näher erläutert werden.

## 2.2 Evolutionäre Algorithmen

Aufgrund der Komplexität eines CPG und eines sehr großen Suchraums, der bei zehn Neuronen etwa 120 Dimensionen beträgt, ist eine analytische Lösung nicht praktikabel. Um die große Menge an Parametern richtig zu bestimmen, wird sich oft der evolutionären Algorithmen bedient. Zumeist verwendet man Methoden der genetischen Algorithmen, welche an die natürliche Evolution angelehnt sind. Die unterschiedlichen Individuen einer Generation werden hierbei miteinander verglichen und anhand einer Fitness-Funktion bewertet. Im Anschluss wird eine Selektion durchgeführt, wobei die stärksten Individuen ausgewählt werden, um in eine neue Generation überführt zu werden. Mittels einer Rekombination (Crossover) werden die Eigenschaften der neuen Population miteinander vermischt. Hieraus entstehen Nachfolger, die gute Eigenschaften ihrer Vorgänger kombinieren. Schlechte Eigenschaften werden so von Generation zu Generation herausgefiltert, in dem sie nicht in neue Generationen übertragen werden. Zur Vermeidung von lokalen Optima werden im Anschluss Mutationen durchgeführt. Hierbei werden einzelne Ausprägungen des neuen Individuums zufällig neu besetzt. Zwar entsteht hierdurch die Möglichkeit einer massiven Verschlechterung, allerdings können so auch neue bessere Individuen entstehen, die sich in Folgegenerationen positiv auswirken. Die neu generierte Generation von Parametern wird anschließend zur Einstellung der neuronalen Oszillatoren verwendet. Durch eine genügend große Anzahl an Individuen und einer ausreichenden Anzahl an Generationen werden die Parameter soweit optimiert, dass die gewünschte Bewegung erzeugt wird.

Die Optimierung der Parameter kann durchaus auch durch andere Algorithmen erfolgen. So gehört das simulierte Abkühlen (engl. simulated annealing), sowie der Simulated Annealing Algorithmus ebenfalls zur Klasse der Algorithmen um globale, nichtlineare Optimierungen durchzuführen. Auf diese soll in diesem Kontext allerdings nicht weiter eingegangen werden.

## 2.3 Model

Die Modelle, welche für die Umsetzung der Bewegung genutzt werden, können von unterschiedlicher Art sein. So ist eine Simulation mittels 3D-Model innerhalb einer physikalischen Welt ebenso denkbar, wie die Umsetzung von Bewegungsabläufen bei Robotern oder anderen dynamischen Systemen. Zur Anschauung der Funktionalität wird häufig auch eine rein zweidimensionale Präsentation gewählt. Das zugrundeliegende Model stellt - unabhängig von Darstellung oder Form - das künstliche Muskelsystem mit Knochen, Sehnen und Gelenken dar. Diese setzen die rythmischen Signale in reale Bewegungsabläufe um. Die unterschiedlichen Systeme sind jeweils anderen Umweltbedingungen ausgesetzt und somit auch auf andere Art zu behandeln. So ist ein Roboter allen natürlichen Umwelteinflüssen ausgesetzt, gerade wenn dieser in freier Natur eingesetzt wird, wohingegen eine 3D Simulation nur innerhalb einer künstlichen physikalischen Welt existiert. Diese und andere Kriterien sind bei der Konzeption einer solchen Anwendung zu bedenken.

## 2.4 Systemaufbau

Der Aufbau des Gesamtsystems hängt vom jeweiligen Anwendungszweck und dem gewünschten Ergebnis ab. Man findet bei der Untersuchung der Systeme jedoch häufig einen ähnlichen Grundaufbau vor. Die drei Hauptbestandteile des Systems sind wie oben beschrieben zum einen der CPG, der evolutionäre Algorithmus, sowie das zu steuernde Model. Die Anzahl der verwendeten neuronalen Oszillatoren hängt von der Anzahl der zu bewegenden Gelenke ab. So können wie bei Reil und Husband [3], abhängig von der Anzahl der Freiheitsgrade des Gelenks, auch mehrere Oszillatoren pro Gelenk verwendet werden. So werden für Beugung und Streckung der Gelenke meist unterschiedliche Oszillatoren verwendet, um diese unabhängig voneinander zu regeln. Um eine stabile Bewegung zu erreichen, ist mittels e. A. eine große Anzahl an Generationen zu durchlaufen. Wie hoch diese Anzahl ist, kann im Vorfeld nur schlecht bestimmt werden und sollte von Fall zu Fall unterschiedlich sein. Besondere Aufmerksamkeit gilt der Wahl der richtigen Fitness-Funktion. Diese ist maßgeblich für die Bewertung der einzelnen Individuen und bestimmt somit die Güte der Populationen. Ist eine ausreichend gute Bewertung der Individuen nicht gewährleistet, verringert sich die Chance auf eine globale Optimierung des Problems.

## 3 Natural Motion - Reil and Husbands

Die Natural Motion Ltd, mit Hauptsitz in Oxford, ist ein Unternehmen, welches Produkte auf dem Gebiet der automatisierten Bewegungsgenerierung anbietet. Hierzu wird das hauseigene Verfahren der Dynamic Motion Synthesis (DMS) verwendet. Es werden sowohl Authoring-Tools als auch Engines angeboten, die es ermöglichen, Charakteren in Spielen oder Filmproduktionen aufwendige und realistische Animationen zu verleihen.

Basierend auf einer Arbeit durch Reil und Husbands aus dem Jahr 2002 [3], werden Animationen durch Kombination von künstlicher Intelligenz, Biomechanik und dynamischer Simulation erzeugt. Nachfolgend werden sowohl die Arbeit als auch zwei der Produkte im Detail vorgestellt.

### 3.1 Evolution eines zentralen Mustergenerators

Die Arbeit „Evolution of Central Pattern Generator for Bipedal Walking in a Real-Time Physics Environment“ stellt wohl die Grundlage für die heutigen Produkte der Natural Motion Ltd dar. Die Forschungsarbeit, erstellt an der University of Oxford, verwendet die im Kapitel „Systemübersicht“ vorgestellten Elemente CPG und evolutionäre Algorithmen zur Erzeugung von Laufbewegungen eines zweibeinigen 3D-Modells. Desweiteren wurden Versuche unternommen, eine gerichtete Bewegung zu erzeugen. Hierzu wurden Soundquellen eingebunden, die die Laufrichtung für das Modell vorgeben sollten.

#### 3.1.1 Central Pattern Generator

Der in dieser Arbeit verwendete Mustergenerator besteht aus zehn neuronalen Oszillatoren, welche alle miteinander verbunden sind. Sechs dieser Oszillatoren stellen sogenannte Motorneuronen dar. Mit ihnen werden Gelenke gesteuert, welche die Kräfte auf das 3D-Modell umsetzen und somit eine Laufbewegung herbeiführen. Die vier weiteren Neuronen haben direkte Verbindungen zu den Motorneuronen und stabilisieren somit das neuronale Netz weiter.

Das Verhalten eines einzelnen Neurons wird durch verschiedene Parameter beeinflusst. Neben den Gewichtungen, die zwischen den unterschiedlichen Verbindungen innerhalb des Netzwerkes vorhanden sind, steuern vor allem zwei weitere Parameter das Verhalten. Zum einen eine Zeitkonstante und zum anderen einen sogenannten Bias. Bei jedem Simulationsschritt wird die Aktivität eines Neurons berechnet und entsprechend in eine Bewegung umgesetzt.

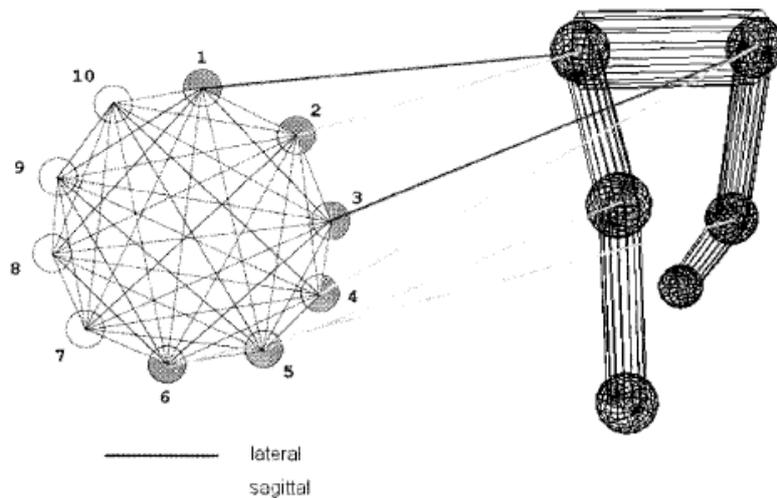


Abbildung 3.1: Verbindungen zwischen Oszillatoren und Model [3]

### 3.1.2 Evolutionäre Algorithmen

Die Parameter des Mustergenerators werden mittels genetischem Algorithmus eingestellt und optimiert. Zu den Parametern gehören sowohl die Gewichtungen der Verbindungen als auch eine Zeitkonstante und ein Bias für alle Neuronen. Um eine gleichbleibende Darstellung zu erreichen, wurde zur Repräsentation der Daten ein Kodierungsschema festgelegt.

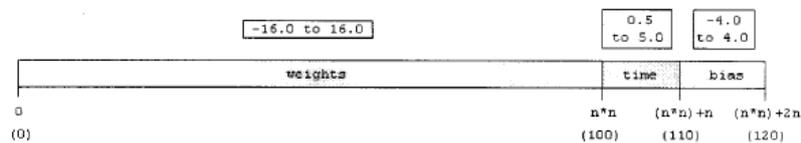


Abbildung 3.2: Kodierungsschema für Parameter [3]

Die Parameter sind hier als Zahlen in bestimmten Bereichen definiert. Insgesamt entsteht so ein Schema mit einer Länge von 120 Stellen bei einer Anzahl von zehn Neuronen. Auf diesem so entstandenen Schema können die genetischen Operatoren wie Rekombination und Mutation einfach ausgeführt werden. Eine Population besteht hier aus einer Anzahl von 50 Individuen. Die Parameter jedes einzelnen Individuums werden anhand einer Vorgabe zufällig eingestellt. Nach der Simulation aller einzelnen Individuen einer Population wird anhand einer Fitness-Funktion eine Platzierung ermittelt. Zur Bestimmung der Fitness wird hier die erreichte Distanz ermittelt, die das Individuum zurückgelegt hat. Um sehr schlechte Ergebnisse von vornherein auszuschließen, muss eine Vorbedingung erfüllt sein. Diese ist, dass über die gesamte Simulationszeit die Gelenke für die Hüfte immer oberhalb der Gelenke für die Knie liegen muss. Wäre diese nicht erfüllt, würde es bedeuten, dass das Model sich durch eine Art Kriechen o. ä. fortbewegt. Die Selektion nach der Bewertung der Individuen erfolgt rangbasiert. Hierzu wird die untere Hälfte der Liste entfernt und mit einer Kopie der oberen Hälfte ersetzt. Eine Rekombination der einzelnen Individuen erfolgt hier nicht, da aufgrund von Experimenten die Effizienz innerhalb die-

ser Problemstellung nicht bewiesen werden konnte. Die Mutationsrate ist hier so angelegt, dass in etwa eine Änderung pro Chromosom entsteht.

Innerhalb der Arbeit wurden 100 Durchläufe mit jeweils 120 Generationen durchgeführt. In untenstehender Abbildung ist die Optimierung über die Laufzeit gut nachzuvollziehen. So sind die Parameter der Oszillatoren circa nach der 67. Generation stabil und die zurückgelegte Distanz nahezu gleichbleibend - mit einer durchschnittlichen Fitness von ca 15 Metern und einer Top-Fitness von ca 20 Metern innerhalb einer Simulationszeit von 50 Sekunden.

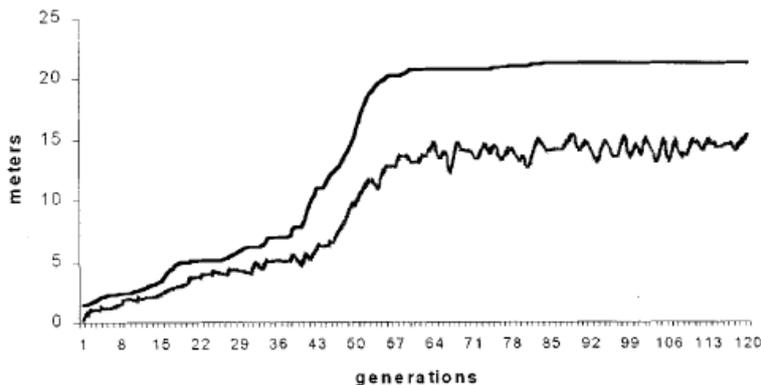


Abbildung 3.3: Fitness-Graph für stabile Laufbewegungen [3]

## 3.2 Euphoria

Euphoria ist eines der beiden Hauptprodukte der Natural Motion Ltd. Es ermöglicht eine Echtzeitanimation von Spielcharakteren und anderen physikalischen Objekten. Statt innerhalb eines Spiels vordefinierte Animationen einzusetzen, die von Hand oder durch kostenintensive Motion-Capturing Verfahren erstellt werden, kann Euphoria mit seiner Dynamic Motion Synthesis zur Animationsgenerierung genutzt werden. Es lässt sich vollständig in eine vorhandene Game Engine integrieren und ist sowohl für die Next-Gen Konsolen Playstation 3 und Xbox 360 als auch für den PC erhältlich. Der Euphoria:Core interagiert sowohl mit der Game-Engine als auch mit der verwendeten Physik-Engine um Verhalten und Bewegung der Objekte zu generieren. Basierend auf einer Simulation von sowohl Körper als auch künstlichen Muskeln und Nerven, entsteht so eine Echtzeitsimulation, die realitätsnahe Animationen erzeugt.

Drei Beispiele für eine Integration der Euphoria-Engine finden sich in den Spielen Grand Theft auto IV, Star Wars - The Force unleashed und Backbreaker. Letzteres ist ein American Football-Spiel, in welchem der Spieler in die Rolle eines Footballspielers schlüpft. Hier ist durch in Echtzeit erstellte Animationen bei Zusammenstößen zwischen Spielern eindrucksvoll nachzuvollziehen, welche Ergebnisse durch diese Art der Animationsgenerierung erzielt werden können.

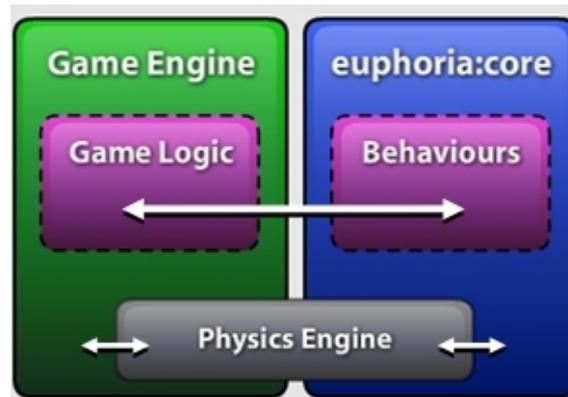


Abbildung 3.4: Euphoria Integration - <http://www.naturalmotion.com/euphoria.htm>

### 3.3 Endorphin

Endorphin, eine Art Authering-Tool für Animationsabläufe ist das zweite große Produkt der Firma Natural Motion. Der Nutzer kann komplexe Szene zusammensetzen, in der die virtuellen Akteure agieren. Die Bewegung der Charaktere wird durch die Anwendung generiert und muss nicht von Hand erstellt werden. Ebenso können verschiedene Arten von Verhalten angelegt werden. Die so erzeugten Animationen können auch exportiert und an anderer Stelle noch verfeinert oder aufbereitet werden. Gerade durch die Möglichkeit komplexe Szene zusammensetzen, ist diese Software auch für Filmproduktionen von Interesse. So können auch gefährliche oder zu kostspielige Szenen erstellt werden, ohne auf die Arbeit von Stuntmans angewiesen zu sein.

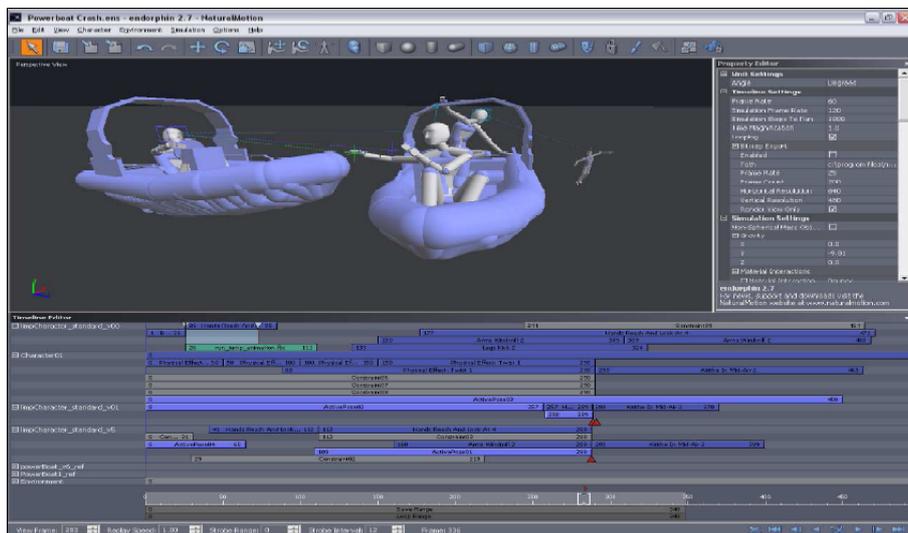


Abbildung 3.5: Arbeitsumgebung von Endorphin

## 4 Multi-legged Robots - Valsalam und Miikkulainen

Die Arbeit von Valsalam und Miikkulainen entstand 2008 an der Universität Texas in Austin. Sie beschäftigt sich mit der Neuroevolution für mehrbeinige Bewegung. Innerhalb der Arbeit wird hierfür ein 3D-Model eines vierbeinigen Roboters verwendet. Dieser Ansatz soll allerdings laut den Autoren ebenso für eine höhere Anzahl an Beinen sowie einen erhöhten Freiheitsgrad für die Gelenke Gültigkeit haben [4].

### 4.1 Multi-legged Locomotion

Roboter mit Beinen haben im Vergleich zu Robotern, die sich mit Rädern oder anderen Mittel vorwärts bewegen, eine erhöhte Mobilität. Gerade bei unwegsamem Gelände mit Hindernissen oder lockerem Untergrund sind Beine bei der Fortbewegung vorteilhaft. Im Vergleich zu Robotern mit Rädern o. ä. ist die Steuerung von Beinen allerdings auch umständlicher und erfordert viel Feinabstimmung um einen stabilen Gang zu erzeugen. Aus diesem Grund wird die Konfiguration meist per Hand durchgeführt.

Valsalam und Miikkulainen hingen erzeugen durch Neuroevolution die Möglichkeit, die Abstimmung der Beine automatisiert durchzuführen. So kann das Model ebenso verschiedene Gangarten wählen, zwischen diesen wechseln und sich an neue Geländearten anpassen. Zu den unterschiedlichen Gangarten gehören der Passgang, ähnlich der Laufbewegung bei Pferden, dem Trab, sowie zwei Arten des Springens.

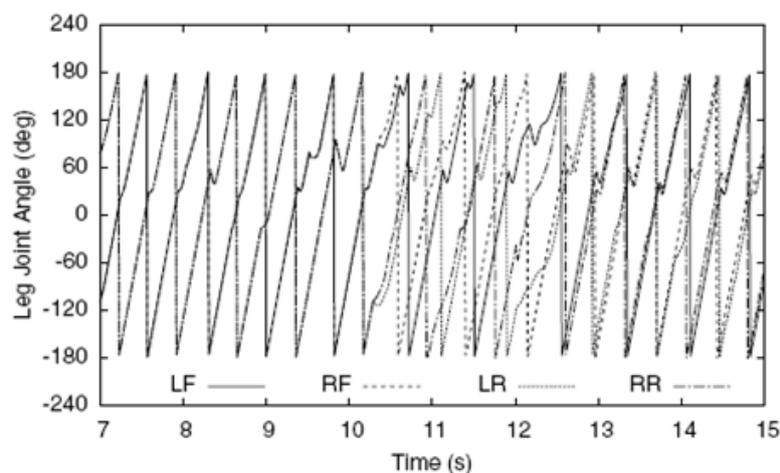


Abbildung 4.1: Veränderung der Signale bei einem Wechsel der Gangart [4]

Der Wechsel von einer zur anderen Gangart kann auch während einer Vorwärtsbewegung durchgeführt werden. So ist es dem Roboter möglich, ein auftauchendes Hindernis zu überwinden, in dem er von einer Lauf- in eine Springbewegung wechselt. Eine Reaktion auf Hindernisse oder Umwelteinflüsse ist im Projekt von Reil und Husbands nicht möglich. Würde hier ein Hindernis auftauchen, würde das Model stolpern und hinfallen.

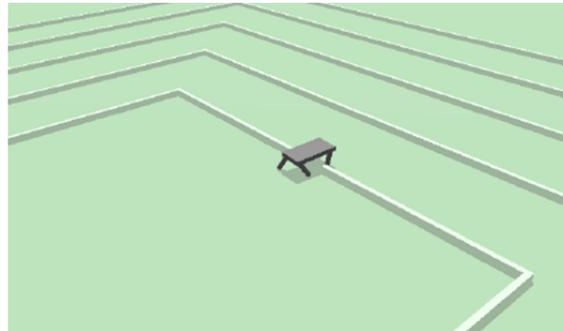


Abbildung 4.2: Wechseln der Gangart bei Hindernissen [4]

Um eine Reduktion des Suchraums bei der Findung der richtigen Parameter herbeizuführen, machen sich die Autoren die im Model vorhandenen Symmetrien zunutze. Hierbei werden Parameter jeweils nur für ein Bein von zum Beispiel zwei aktiven Beinen bestimmt. Da bei allen Gangarten immer mehrere Beine die gleiche Bewegung durchführen, ist dies eine gute Möglichkeit, Berechnungen und Optimierungsvorgänge einzusparen. Eine weitere Möglichkeit die genutzt wird ist die Aufteilung in vier Module. Hierbei wird die Struktur des neuronalen Netzes aufgebrochen um den benötigten Suchraum zu verkleinern. Die Module werden unabhängig voneinander optimiert und steuern je einen Ausgang.

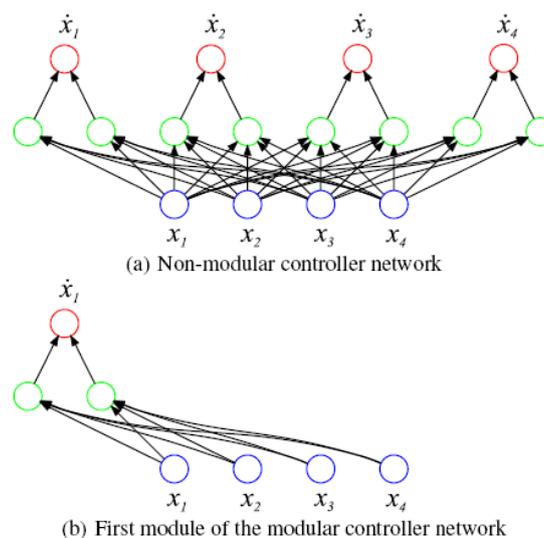


Abbildung 4.3: Reduzierung des Suchraums durch Modularisierung [4]

Die entstanden Module stellen ein „Coupled cell systems“ (Gekoppeltes-Zell-System) dar. Hierunter versteht man eine Reihe von dynamischen Systemen, dessen Zustände sich

gegenseitig beeinflussen. Die Art der Kopplung und die Art der Zustände, definieren das Gesamtsystem. Mittels dieser Zellsysteme ist es möglich, eine Art des Zentralen Muster-generators zu erzeugen. Die hierdurch entstehenden Muster werden auch hier genutzt, um die Gelenke des Modells anzutreiben.

# 5 From Swimming to Walking - Ijspeert et al

Die Arbeit „From swimming to walking with a salamander robot driven by a spinal cord model“ von Ijspeert et al. entstand als Gemeinschaftsprojekt der School of Computer and Communication Sciences (Schweiz), Institut national de la santé et de la recherche médicale (Bordeaux) sowie der Universität Bordeaux im Jahr 2007 [1]. Ziel war es hier, einen Roboter, der einem Salamander nachempfunden ist, durch den Einsatz von CPG zu steuern. Evolutionäre Algorithmen kommen hier nicht zum Einsatz, trotzdem soll es im folgenden vorgestellt werden, da es ein eindrucksvolles Anwendungsbeispiel eines CPG aufzeigt.

## 5.1 Aufbau

Der Aufbau des Systems ähnelt vom Grunde her dem der anderen Systeme. Entlang der künstlichen Wirbelsäule des Roboters sind neuronale Oszillatoren angebracht. Diese bilden den sogenannten Body-CPG und steuern die Bewegung des Körpers. Der Body-CPG besteht aus 16 Oszillatoren, die lediglich mit ihren nächsten Nachbarn verbunden sind. Die von ihnen erzeugten Signale werden zur Steuerung von sechs Gleichstrom-Motoren verwendet, die die einzelnen Glieder miteinander verbinden.

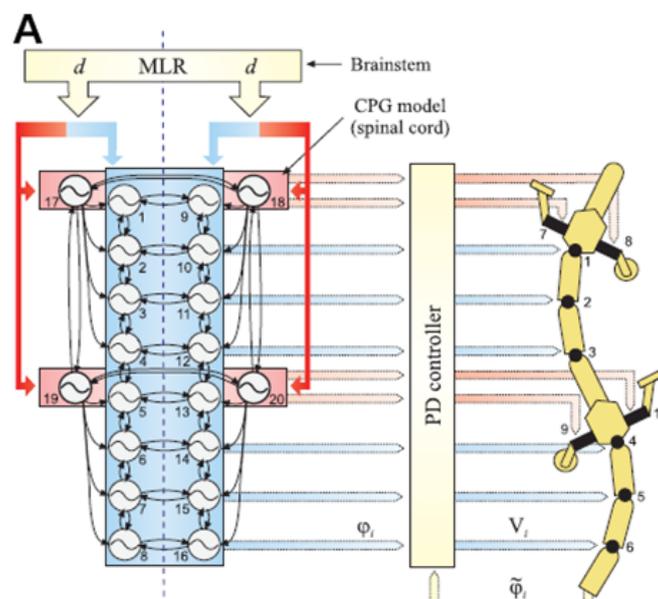


Abbildung 5.1: Aufbau des Salamander-Roboters [1]

Vier weitere Motoren bewegen die FüÙe des Roboters, diese werden durch vier Oszillatoren gesteuert, welche sich innerhalb der FüÙe befinden. Diese werden als Limb-CPG bezeichnet. Durch die Oszillatoren werden Winkelstellungen erzeugt, welche durch die Verwendung eines PD-Reglers in Steuersignale umgesetzt werden.

Dem Roboter ist es möglich, durch Änderung der Frequenz der Oszillatoren eine Bewegungsänderung durchzuführen. Hierbei wird ein Wechsel von einer Schwimm- zu einer Laufbewegung vollzogen - oder umgekehrt. Die Steuerung der Bewegung erfolgt über eine Schnittstelle, die dem Mittelhirn nachempfunden ist. Durch Stimulierung mit elektrischen Impulsen kann sowohl die Art der Bewegung als auch die Richtung und Geschwindigkeit induziert werden. Ist die Limb-CPG aktiv überschreibt diese die Signale der Body-CPG und erzwingt somit einen Wechsel von einer schwimmenden Körperbewegung zu einer laufenden. Ermöglicht wird dies durch eine stärkere Kopplung zwischen Limb-CPG zu Body-CPG als von Body-CPG zu Limb-CPG.

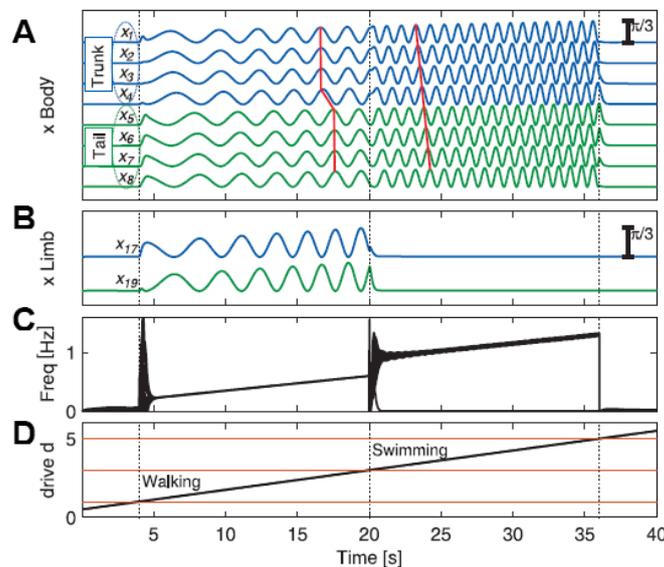


Abbildung 5.2: Frequenzänderung bei Bewegungswechsel [1]

Im Vergleich zur Bewegung des natürlichen Salamanders ist bei dem Model eine große Ähnlichkeit zu erkennen. Die begrenzte Anzahl an Gelenken am Robotermodel führen dabei allerdings zu einer leicht veränderten Bewegung, besonders bei der S-förmigen Körperbewegung.

## 6 Zusammenfassung

Die hier vorgestellten Systeme zeigen auf unterschiedliche Art und Weise die Anwendung und Umsetzung von evolutionärer Bewegungssteuerung. Sowohl Projekte aus dem Bereich Robotik als auch reine Simulationen wurden vorgestellt und beschrieben. Die Anzahl an weiteren Beispielen ist umfangreich und kann hier nicht in aller Tiefe wiedergegeben werden. Jedoch sind viele Projekt im Grundaufbau ähnlich und verfolgen den gleichen Ansatz. Unterschiede sind zumeist im Aufbau des CPG zu erkennen. Da hier ein, zum Anwendungszweck passender, Aufbau gewählt werden muss, sind Unterschiede im Detail nicht verwunderlich. Auch bei der jeweiligen Konfiguration des verwendeten evolutionären Algorithmus sind Unterschiede festzustellen. So ist die verwendete Bewertungsfunktion bei unterschiedlichen Projekten anderes gewählt. Auch hier ist der jeweilige Anwendungszweck maßgeblich für den Aufbau und muss individuell angepasst werden. Wie im Projekt „From swimming to walking“ kann auch gänzlich auf die Verwendung eines evolutionären Algorithmus verzichtet werden. Dann sind allerdings alle dem CPG zugehörigen Parameter per Hand einzustellen.

Die Steuerung von zwei- oder mehrbeinigen Systemen stellt auch heute noch eine große Herausforderung dar. Ebenso ist die Erstellung von komplexen Animationen und Bewegungsabläufen teilweise sehr mühsam und zeitweise auch unmöglich oder zu kostenintensiv. Die Verwendung von Mustergeneratoren zusammen mit der evolutionären Optimierung ist eine gute Alternative. Durch die Möglichkeit der automatischen Steuerung und Regelung in Echtzeit, kann sie bei vielen Problemstellungen behilflich sein. In Kombination mit einem Optimierungs-Algorithmus - in diesem Fall eines evolutionären Algorithmus - kann der verwendete CPG auf das jeweilige Anwendungsgebiet angepasst und abgestimmt werden. Eine händische Anpassung der Parameter ist bei vielen Konfigurationen nicht mehr zu leisten, und so stellt der evolutionäre Algorithmus eine gute Ergänzung dar.

## 7 Ausblick

Die Vorstellung von artverwandten Artikeln und Projekten aus dem Bereich der evolutionären Bewegungssteuerung, welche in diesem Bericht erfolgte, stellt eine Vorbereitung für das Masterseminar im dritten Semester des Master-Studiengangs Informatik an der Hochschule für Angewandte Wissenschaften in Hamburg dar. Gleichzeitig erfolgt parallel hierzu ein Projekt in dem die gewonnenen Erkenntnisse praktisch umgesetzt werden. Innerhalb dieses Projektes sollen mittels CPG und evolutionären Algorithmen erste Versuche mit einem eigenen System durchgeführt werden. Die Erfahrungen sollen im kommenden Semester innerhalb des zweiten Projektes weiter vertieft werden und auf den praktischen Teil der Masterarbeit hinführen. Die theoretischen Betrachtungen und die Beschäftigung mit anderen Arbeiten zum Thema, helfen bei einem Einstieg in die vorliegende Materie und unterstützen bei der Themenfindung für die Masterthesis, welche im vierten Semesters des Studienganges erarbeitet wird. Innerhalb des Masterseminars im folgenden Semester, wird eine Thesis-Outline entstehen, die einen Überblick über die zu erarbeitende Masterthesis gibt.

# Literaturverzeichnis

- [1] IJSPEERT, Auke J. ; CRESPI, Alessandro ; RYCZKO, Dimitri ; CABELGUEN, Jean-Marie: From Swimming to Walking with a Salamander Robot Driven by a Spinal Cord Model. In: *Science* 9 (2007), Nr. Vol. 315 No. 5817, pp. 1416-1420
- [2] MATSUOKA, Kiyotoshi: Mechanisms of Frequency and Pattern Control in the Neural Rhythm Generators. In: *Biological Cybernetics* (1987), Nr. 56, 345-353
- [3] REIL, Torsten ; HUSBANDS, Phil: Evolution of Central Pattern Generators for Bipedal Walking in a Real-Time Physics Environment. In: *IEEE Transactions on Evolutionary Computation* (2002), Nr. Vol. 6 No. 2
- [4] VALSALAM, Vinod K. ; MIKKULAINEN, Risto: Modular Neuroevolution for Multi-legged Locomotion. In: *GECCO* (2008)

# Abbildungsverzeichnis

3.1	Verbindungen zwischen Oszillatoren und Model [3]	9
3.2	Kodierungsschema für Paramter [3]	9
3.3	Fitness-Graph für stabile Laufbewegungen [3]	10
3.4	Euphoria Integration - <a href="http://www.naturalmotion.com/euphoria.htm">http://www.naturalmotion.com/euphoria.htm</a>	11
3.5	Arbeitsumgebung von Endorphin	11
4.1	Veränderung der Signale bei einem Wechsel der Gangart [4]	12
4.2	Wechseln der Gangart bei Hindernissen [4]	13
4.3	Reduzierung des Suchraums durch Modularisierung [4]	13
5.1	Aufbau des Salamander-Roboters [1]	15
5.2	Frequenzänderung bei Bewegungswechsel [1]	16