



Doctoral Thesis

Motor control with graphical models

Author(s):

Göhlsdorf, Dennis

Publication Date:

2012

Permanent Link:

<https://doi.org/10.3929/ethz-a-007366037> →

Rights / License:

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

Diss. ETH No. 20271

Motor Control with Graphical Models

A dissertation submitted to the
ETH ZURICH

for the degree of
Doctor of Sciences

presented by
DENNIS GÖHLSDORF

Dipl.-Inf. Eberhard Karls Universität Tübingen
Dipl.-Biochem. Eberhard Karls Universität Tübingen

born April 18th, 1978

citizen of Germany

accepted on the recommendation of
Prof. Dr. Rodney J. Douglas, examiner
Dr. Matthew M. Cook, co-examiner
Prof. Dr. Jörg Conradt, co-examiner

2012

Abstract

Biological nervous systems can deal with many real-world natural stimuli and outperform any computer algorithm available today in most tasks involving interaction with real objects as required for example in interactive motor control problems. Moreover, they are able to learn an internal model of a controllable system solely from experience, unlike today's most successful control algorithms. The computational architecture expressed by brains is fundamentally different from modern computer hardware, in that it is composed of billions of small computational units with information stored only in the effectiveness of the communication channels between these units. Despite this restricted form of information storage, brains are able to devise complex sequences of actions which span many time scales.

We investigate neurally inspired algorithms that learn how to control systems of unknown mechanical structure solely from data collected from experience. These algorithms are designed to run on distributed computational architectures composed of many small and locally operating units. Further, we explore how motions spanning multiple time scales can be planned on such architectures.

Graphical models with discrete-state variables are used to model the relation of both input and output variables of a controllable system. After maximum likelihood learning within these graphs, the sum-product algorithm can be used to infer output values consistent with the current goal during live control. A hierarchical architecture is used to model multiple temporal and spatial scales in order to carry out long sequences of actions without the need to plan the entire action at once. All information is stored exclusively at the nodes of the employed graphical models and both during learning and live control, all computation takes place locally.

We demonstrate how factor graphs modelling instantaneous dynamics can be used to control a simple compliant robotic system and how complex robot kinematics can be factorized into products of simple functions, which can form the core of a basic control algorithm. Furthermore, we describe the concept of hierarchical navigation networks (HNN), which form a new class of graphical model topologies that are based on a hierarchical decomposition of state space. In these models, instantaneous dynamics at different

spatial and temporal scales are modelled at different hierarchy levels. Given a distant goal, higher levels of this hierarchy impose subgoals that are closer to the current position onto lower levels, which eventually allows the determination of an immediate action at the lowest level. Detailed inference algorithms for these models are found, as well as local learning rules for the model parameters.

All models that we employ are self-contained - they learn solely from experience and without the necessity for any prior information or reward signal. By relying on a distributed computation scheme in which the computational units have access only to local messages, our approach to motor control and planning problems shares important features with biological nervous systems. The HNNs we develop introduce a new concept in the modelling of temporal processes and hierarchical plans using graphical models. We believe that the three approaches to motor control and planning presented in this thesis form an important step towards self-contained distributed computational architectures that can learn to control robotic systems on their own.

Zusammenfassung

Nervensysteme von biologischen Organismen und im speziellen das menschliche Gehirn sind in der Lage, eine Vielzahl von natürlichen Stimuli zu verarbeiten. Dabei übertreffen sie alle derzeit verfügbaren Computeralgorithmen bei der Lösung von Aufgaben, die eine Interaktion mit der realen Welt voraussetzen, wie zum Beispiel der Kontrolle von motorischen Systemen. Erstaunlicherweise sind sie in der Lage, ein internes Verständnis von beliebigen steuerbaren Systemen einzig durch die Interaktion mit diesen aufzubauen, was sie von den performantesten Steueralgorithmen unterscheidet. Gleichzeitig ist offensichtlich, dass Berechnungen im Gehirn auf eine Weise durchgeführt werden, die sich fundamental von der Rechenart auf moderner Computerhardware unterscheidet. Anstelle einer oder weniger zentraler Steuereinheiten ist das Gehirn aus vielen Milliarden winziger Recheneinheiten aufgebaut. Informationen werden in diesen Recheneinheiten gespeichert indem die Effektivität der Kommunikationskanäle moduliert wird. Trotz dieses scheinbaren Nachteiles, nur lokale Berechnungen durchführen zu können und keine globale Steuereinheit zu besitzen, sind Nervensysteme in der Lage, eine grosse Vielfalt an Bewegungen von zu planen, welche sich über verschiedene Zeiträume erstrecken können.

In dieser Arbeit untersuchen wir Algorithmen, welche von biologischen Nervensystemen inspiriert sind und allein durch Analyse von beobachteten Zuständen eines zuvor un-spezifizierten steuerbaren Systems lernen, dieses zu kontrollieren. Diese Algorithmen sind darauf ausgelegt, auf verteilten Rechenarchitekturen zu laufen, welche aus vielen kleinen und lokal operierenden Einheiten aufgebaut sind. Weiterhin betrachten wir Möglichkeiten, auf die sich lange Sequenzen von Bewegungen auf solchen Rechenarchitekturen planen lassen.

Wir verwenden Graph-basierte Wahrscheinlichkeitsmodelle, die Variablen mit diskreten Zuständen enthalten, um den Zusammenhang zwischen Eingabe- und Ausgabevariablen eines steuerbaren Systems abzubilden. Diese Modelle werden zunächst nach dem Verfahren der grössten Wahrscheinlichkeit (engl. ‘maximum likelihood’) trainiert. Ein einfacher Algorithmus basierend auf dem Summenprodukt zweier Vektoren kann dann verwendet werden, um Ausgabewerte zu berechnen, wenn nur die Zustände der Eingabevariablen bekannt sind, wie zum Beispiel bei der Steuerung des Systems. Eine hierarchis-

che Modellarchitektur wird eingesetzt, um verschieden grosse zeitliche und räumliche Grössenordnungen zu repräsentieren und um Bewegungen von unterschiedlicher Länge zu planen. Alle eingesetzten Rechenmodellen bestehen aus vielen kleinen Recheneinheiten, welche jeweils nur auf lokal gespeicherte Informationen zugreifen können. Sowohl während der Lernphase als auch der Anwendung dieser Modelle findet die Informationsverarbeitung ausschliesslich in diesen Recheneinheiten statt.

Wir führen vor, wie Faktor Graphen, welche die unmittelbare Bewegungsdynamik eines Systems abbilden, zur Steuerung eines einfachen nachgiebigen robotischen Systems eingesetzt werden können und wie die komplexe Kinematik eines Roboterarms als Produkt einfacher Funktionen dargestellt werden kann, um dann den Kern eines simplen Steueralgorithmus' zu bilden. Darüber hinaus beschreiben wir eine neue Klasse von Graphbasierten Rechenmodellen, welche auf einer hierarchischen Aufteilung des Raumes aufbauen. In diesen hierarchischen Navigations-Netzwerken (HNN) werden die unmittelbare Veränderungen eines Systems und ihre Ursachen in unterschiedlichen räumlichen und zeitlichen Massstäben betrachtet. Soll eine Veränderung herbeigeführt werden, welche viele Einzelaktionen benötigt, so ermitteln die höheren Hierarchieebenen in diesen Netzwerken Zwischenziele, welche den unteren Hierarchieebenen vermitteln werden. Auf der untersten Hierarchieebene kann auf diese Weise eine unmittelbare Einzelaktion bestimmt werden, welche im Sinne der erwünschten Veränderung ist. Wir erklären die genauen Regeln, nach denen in diesen Netzwerken Schlussfolgerungen getroffen werden und beschreiben lokal funktionierende Lernregeln, die zum Aufbau dieser hierarchischen Netzwerke führen.

Alle Kontrollalgorithmen, welche wir in dieser Arbeit verwenden, sind in sich geschlossen und benötigen keine Vorabinformationen über die Struktur des zu kontrollierenden Systems. Die Parameter dieser Algorithmen werden ausschliesslich durch Analyse von Beobachtungsdaten trainiert. Ferner können alle Berechnungen in diesen Algorithmen verteilt auf viele kleine Recheneinheiten durchgeführt werden. In diesen Punkten ähneln die von uns vorgestellten Methoden den Rechenabläufen in biologischen Nervensystemen. Die HNNs, welche wir entwickelt haben, stellen ein neues Konzept dar, wie zeitliche und hierarchische Zusammenhänge in verteilten Rechenstrukturen dargestellt werden können. Zusammenfassend sind wir überzeugt davon, dass die in dieser Arbeit vorgestellten Ansätze zur Steuerung von motorischen Systemen und zur Bewegungsplanung einen wichtigen Schritt darstellen auf dem Weg zu vollständig in sich geschlossenen, auf parallelen Architekturen durchführbaren Rechenalgorithmen, welche von sich aus lernen können, robotische Systeme zu steuern.