



Neuroevolution mit MPI

Analyse und Optimierung von NEAT für ein
verteiltes System

Masterthesis

zur Erlangung des akademischen Grades
Master of Science (M.Sc.)
im Studiengang Angewandte Informatik
an der Hochschule Flensburg

Simon Hauck

Matrikelnummer: 660158

Erstprüfer: Prof. Dr. rer. nat. Tim Aschmoneit
Zweitprüfer: Prof. Dr. rer. nat. Torben Wallbaum

18. Juni 2020

Neuroevolutionäre Algorithmen sind ein mögliches Optimierungsverfahren für neuronale Netze. Abhängig von dem verwendeten Algorithmus können die Gewichte der Verbindungen im Netz und die Struktur entwickelt und optimiert werden.

Der Optimierungsprozess ist, unabhängig vom Verfahren, sehr aufwändig und dementsprechend zeit- und rechenintensiv. Für eine schnellere Durchführung des Trainingsprozesses bieten sich Algorithmen an, die gut parallelisierbar sind. Die benötigte Ausführungszeit dieser kann durch Hinzufügen weiterer Rechenknoten mit geringem Aufwand maßgeblich reduziert werden.

Neuroevolutionäre Algorithmen bieten sich aufgrund der Verfahrensweise und der vielen unabhängigen neuronalen Netzen für eine parallele Ausführung an.

In dieser Arbeit wird, stellvertretend für neuroevolutionäre Algorithmen, der NeuroEvolution of Augmenting Topologies (NEAT) Algorithmus betrachtet. Dieser wurde im Jahr 2002 veröffentlicht und ist im Vergleich zu den damals bekannten Algorithmen besonders effizient. Zudem dient der Algorithmus als Grundlage für viele Erweiterungen. Die erhaltenen Ergebnisse dieser Arbeit lassen sich somit gut auf ebendiese Erweiterungen übertragen.

Im ersten Schritt dieser Arbeit wird die Laufzeit des NEAT Algorithmus mit verschiedenen Optimierungsaufgaben analysiert. Mit den erhaltenen Ergebnissen wird eine parallelisierte Implementierung erstellt. Diese führt mit unterschiedlich vielen Rechenknoten dieselben Optimierungsaufgaben durch. Am Ende dieser Arbeit werden die Ergebnisse von beiden Implementierungen verglichen.

Inhaltsverzeichnis

1	Motivation	1
1.1	Problemstellung	1
1.2	Ziel der Arbeit	1
1.3	Struktur der Arbeit	1
2	Grundlagen	2
2.1	Neuronale Netze	2
2.1.1	Das Vorbild - Biologische neuronale Netze	2
2.1.2	Das Neuron	2
2.1.3	Netzstrukturen	2
2.1.4	Optimierungsverfahren	2
2.2	Evolutionäre Algorithmen	3
2.3	NEAT	3
2.4	MPI	3
3	Analyse	4
3.1	Anforderungen	4
3.2	Softwarearchitektur und Implementierung	4
3.3	Testsetup	4
3.4	Evaluation	4
4	Software Architektur und Implementierung	5
5	Evaluation	6
5.1	Testsetup	6
5.2	Ergebnisse	6
6	Zusammenfassung und Ausblick	7
	Quellenverzeichnis	8
	Eidesstattliche Erklärung	9

Abbildungsverzeichnis

Akronymverzeichnis

T	Test
roAAAA	Very much A's
NEAT	NeuroEvolution of Augmenting Topologies
KNN	Künstliche neuronale Netze

1 Motivation

1.1 Problemstellung

1.2 Ziel der Arbeit

1.3 Struktur der Arbeit

2 Grundlagen

2.1 Neuronale Netze

Künstliche neuronale Netze (KNN) sind ein Teilgebiet des maschinellen Lernens, deren Geschichte 1943 mit McCulloch und Pitts beginnt, welche in ihrer Arbeit ein einfaches neuronales Netz mit Schwellwerten entwickeln. Dieses ist von biologischen neuronalen Netzen inspiriert und ermöglicht die Berechnung von logischen und arithmetischen Funktionen [1]. In den folgenden Jahrzehnten wird die Funktionsweise der neuronalen Netze weiterentwickelt und der Einsatz in verschiedensten Aufgabenfeldern ermöglicht. Hierzu zählen unter anderem die Klassifizierung von Bildern [2], Erkennen und Interpretation von Sprache [3], [4] sowie das selbständige Lösen von Computer- und Gesellschaftsspielen [5], [6].

In diesem Kapitel wird zuerst ...

2.1.1 Das Vorbild - Biologische neuronale Netze

2.1.2 Das Neuron

2.1.3 Netzstrukturen

2.1.4 Optimierungsverfahren

Das Gebiet der Künstlichen Neuronen Netze wird bereits seit 1943 erforscht

Test (T) and [7] with Stanley und Miikkulainen

asfkajsfö

2.2 Evolutionäre Algorithmen

2.3 NEAT

2.4 MPI

3 Analyse

3.1 Anforderungen

3.2 Softwarearchitektur und Implementierung

3.3 Testsetup

3.4 Evaluation

4 Software Architektur und Implementierung

5 Evaluation

5.1 Testsetup

5.2 Ergebnisse

6 Zusammenfassung und Ausblick

Quellenverzeichnis

- [1] Warren S McCulloch und Walter Pitts. 1943. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5, 4, 115–133.
- [2] Alex Krizhevsky, Ilya Sutskever und Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.
- [3] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath u. a. 2012. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *IEEE Signal processing magazine*, 29, 6, 82–97.
- [4] Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov und Michael Collins. 2016. Globally normalized transition-based neural networks. *arXiv preprint arXiv:1603.06042*.
- [5] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra und Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.
- [6] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot u. a. 2016. Mastering the game of go with deep neural networks and tree search. *nature*, 529, 7587, 484.
- [7] Kenneth O Stanley und Risto Miikkulainen. 2002. Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10, 2, 99–127.

Eidesstattliche Erklärung

This is the beginning