

# **Staatenbildende Insekten als Vorbilder für Software-Agenten**

**Seminarvortrag Bionik Sommersemester 2000**

**Axel Beckert, [abe@cs.uni-sb.de](mailto:abe@cs.uni-sb.de)**

**7. Juli 2000**

# Titelseite

- CAO → ACO (Ant Colony Optimization)
- Insekten, die soziale Gemeinschaften bilden dienen Informatikern als Vorbild zu sog. Agenten.

## Übersicht

<b>1</b>	<b>Motivation</b>	<b>3</b>
<b>2</b>	<b>Ant Colony Optimization (ACO)</b>	<b>4</b>
2.1	Vorbild Ameise . . . . .	4
2.1.1	Wie Ameisen den kürzesten Weg finden . . . . .	5
2.2	Anwendungsbeispiele für ACO . . . . .	6
2.2.1	Ameisen als Handlungsreisende . . . . .	6
2.2.2	Ameisen als „adaptive Router“ in Netzwerken . . . . .	9
2.2.3	Ameisen steuern Lieferwagen . . . . .	9
2.2.4	Arbeitsprozesse mit Ameisen verbessern . . . . .	9
<b>3</b>	<b>Weitere Beispiele für das Vorbild Insekt in der Informatik</b>	<b>10</b>
3.1	Teamwork ohne Kommunikation bei Ameisen und einfachen Robotern	10
3.1.1	CRIP: Multi-Robot Box-Pushing . . . . .	10
3.2	Lackierroboter verhalten sich wie Bienen . . . . .	12
3.3	Bankkunden werden wie tote Ameisen behandelt . . . . .	13
<b>4</b>	<b>Literatur</b>	<b>14</b>

# 1 Motivation

- In Insektenstaaten löst eine Vielzahl von selbstständigen Individuen gemeinsam komplexe Probleme durch einfache Interaktionen: „Schwarmintelligenz“
- Viele dieser Probleme sind sog.  $NP$ -harte Probleme.
- Das Verhalten und Zusammenspiel in Insektenstaaten kann als Vorbild für neue heuristische Algorithmen dienen.

# Motivation

- Jedes Individuum eines Insektenstaates handelt **scheinbar selbständig** und ist nur mit **einfachen Fähigkeiten** ausgestattet. Auch die **Kommunikation zwischen den einzelnen Individuen ist sehr beschränkt**, eine einzelne Ameise weiß nicht, was die meisten anderen Ameisen in der Kolonie machen. Trotzdem werden **hochkomplexe Probleme**, wie z. B. das **Finden des kürzesten Weges zu einer Futterquelle**, gelöst. Solche kollektiven Leistungen werden „**Schwarmintelligenz**“ genannt.

- Jedes **Individuum eines Insektenstaates** kann nur **kleine Leistungen** vollbringen. Im **Kollektiv** können dagegen **Leistungen vollbracht werden, die die Summe der Einzelleistungen übertreffen**. Simulationen zeigen, daß die Mechanismen<sup>1</sup>, die ein einzelnes Individuum der Art *Lasius niger* nutzt, effizienter sind als ungerichtete, zufällige Auskuntschaftung der Umgebung, aber weniger effizient als spezialisierte, gerichtete

---

<sup>1</sup>Mechanismen von *Lasius niger* bei der Wegsuche vom Futterplatz zum Nest:

- Bidirektionales Spurlegen
- Umkehren oder wählen anderer Alternativen bei zu langen Strecken.
- Berücksichtigen des Winkels des Weges gegenüber der Luftlinie zwischen Nest und Futterplatz. Einige Ameisen-Arten haben nämlich ein sehr gutes räumliches Gedächtnis. [Leerink, Schultz & Jabri, 1992]

Algorithmen. Trotzdem steigt die Leistung überproportional, wenn mehrere Individuen gleichzeitig die Umgebung erkunden. In diesem Zusammenhang wird auch häufig von **Synergie-Effekten** gesprochen.

- Ein Problem ist  $\mathcal{NP}$ -hart, wenn es mindestens so schwer ist, wie alle Probleme in  $\mathcal{NP}$  (nicht-polynomial). Sie lassen sich nur<sup>2</sup> mit exponentiellem Rechenaufwand (genau) berechnen. In vielen Anwendungen wird allerdings nicht das genaue Ergebnis benötigt, **eine Annäherung reicht oft aus**.

Entsprechend wird in der Informatik immer wieder nach besseren Approximationen bekannter  $\mathcal{NP}$ -harter Probleme gesucht.

- Gerade das kollektive Finden des kürzesten Weges zum Futterplatz bietet sich an, als **Vorbild für ein heuristischen Algorithmus** zur Lösung eines Traveling Salesman Problems (TSP) herangezogen zu werden.

---

<sup>2</sup>Man vermutet, daß  $\mathcal{P} \neq \mathcal{NP}$  gilt. Der Beweis (oder Gegenbeweis) ist **eine der wichtigsten offenen Fragen in der Theoretischen Informatik**.

## **2 Ant Colony Optimization (ACO)**

### **2.1 Vorbild Ameise**

Ameisen markieren Weg zur Futterquelle auf dem Rückweg zum Nest mit Pheromonen.

Anderer Ameisen folgen der Duftspur der Ameise zurück, die das Nest zuerst erreicht, also den kürzesten Weg eingeschlagen hatte. Dadurch wird diese Duftspur noch verstärkt und weitere Ameisen schlagen ebenfalls diesen Weg ein.

Mit der Zeit verdunsten die Markierungen wieder, so daß es auch die Möglichkeit für neue Alternativen gibt.

# Ant Colony Optimization (ACO)

Bei ACO handelt es sich um **heuristische Algorithmen**, die gefunden Ergebnisse sind **nicht notwendigerweise, aber häufig annähernd optimal**.

## Vorbild Ameise

Wege zu Futterplätzen („Ameisenstraßen“) werden mit **Pheromon** von den von der **Fut-terstelle zurückkehrenden Ameisen** markiert. Die Ameise, die den **kürzesten Weg** zurück „erwischt“ hat, **kehrt als erste zurück**. Ameisen aus dem Nest **folgen der für sie verlockend riechenden Duftspur** und **verstärken deren Intensität durch ihre eigenes Pheromon**. So folgen weitere Ameisen...

Auf der nächsten Folie wird dies an einem einfachen Beispiel erklärt.



## 2.1.1 **Wie Ameisen den kürzesten Weg finden**

Wie Ameisen den kürzesten Weg finden aus [Dorigo & Gambardella, 1996a, S. 2]

# Wie Ameisen den kürzesten Weg finden

- A Ameisen kommen am der Gabelung an.
- B Statistisch gesehen wählen im Durchschnitt jeweils gleichviele Ameisen den oberen und den unteren Weg.
- C Da sich Ameisen mit einer einigermäßen gleichmäßigen Geschwindigkeit bewegen, sind die Ameisen, die den kürzeren unteren Weg gewählt haben, schneller um das Hindernis herum als die Ameisen, die den längeren oberen Weg wählten.
- D Ameisen, die an den Gabelungen ankamen, nachdem die Ameisen, die den kürzeren Weg wählten, am Hindernis vorbei sind, nicht aber die, die den längeren Weg wählten, wählen den unteren Weg, da dort mehr Pheromone liegen und verstärken damit dessen Pheromonspur. Die Anzahl der Striche ist ungefähr proportional zur Menge an von Ameisen hinterlassenen Pheromonen auf der jeweiligen Strecke.

Auf diese Weise können Ameisen (und ihre virtuellen Pendant) auch sehr **schnell auf Veränderungen** in ihrer Umgebung **reagieren**, z. B. den kürzesten Weg um ein neues Hindernis.

Aber: Bei den an argentinischen Ameisen (*Linepithema humile*) gemachten Untersuchungen wurde gezeigt, daß sich das Pheromon nur sehr langsam wieder verflüchtigen und ein bei-

spielsweise später angebotener noch kürzer Weg nicht angenommen wird. Man kann dies bei „virtuellen Ameisen“ aber durch eine schnellere Verdunstung des „virtuellen Pheromons“ umgehen.

## 2.2 Anwendungsbeispiele für ACO

### 2.2.1 Ameisen als Handlungsreisende

#### **ACO-Algorithmus für das Traveling Salesman Problem:**

Virtuelle „Ameisen-Agenten“ ziehen unabhängig voneinander von Stadt zu Stadt, nahegelegene Städte bevorzugend und keine Stadt doppelt besuchend. Hat eine Ameise alle Städte besucht, geht sie ihre Wegstrecke zurück und markiert sie mit Pheromon. Je kürzer die Gesamtstrecke war, desto mehr Pheromon wird zur Markierung verwendet. Sind alle Ameisen wieder zurück, dann liegt auf den Strecken, die zu den kürzesten bisher gefundenen Rundreisen gehören, auch das meiste Pheromon.

Anschließend wird die Ameisen-Kolonie wieder auf die Reise geschickt, allerdings mit dem Unterschied, daß sie diesmal neben den nahegelegene Städten auch Verbindungen bevorzugen, auf denen viel Pheromon liegt.

Nun läßt man die Pheromone etwas verdunsten und schickt die Ameisen-Agenten wieder auf Tour. Dies wiederholt man nun viele Male. Mit der Zeit wird die Pheromondifferenz zwischen den längeren und kürzeren Rundreisestrecken immer größer.

Durch das Wechselspiel von Verdunstung und Verstärkung der Markierungen

**bleiben auf die Dauer wirklich nur bessere Verbindungen, Probleme mit lokalen Maxima wie bei den üblichen Gradienten-Verfahren gibt es nur in den ersten Iterationen.**

# Traveling Salesman Problem mit ACCO

## Traveling Salesman Problem

Das Traveling Salesman Problem ist **eines der bekanntesten  $\mathcal{NP}$ -harten Probleme** und wird in der Informatik aufgrund der **leichten Verständlichkeit** gerne als **Beispiel** und aufgrund der **hohen kombinatorischen Komplexität**<sup>1</sup> gerne als **Vergleichstest** [Reinelt, 1991] für Algorithmen dieser Art verwendet. Ein gutes Abschneiden von Algorithmen bei Tests mit dem Traveling Salesman Problem werden auch gerne als Beweis für die Nützlichkeit der Algorithmen gesehen.

---

<sup>1</sup>Bei  $n$  Städten gibt es  $(n - 1)!$  verschiedene Routen, das sind bei 15 Städten bereits Milliarden mögliche Routen.

## **Traveling Salesman Problem Beispiel: Problem „CCA0“**

Traveling Salesman Problem „CCA0“ aus [Dorigo, Maniezzo & Colomi, 1996]

## **Abbildung: Traveling Salesman Problem „CCA0“**

- a Pheromon-Verteilung zu Beginn der Suche
- b Pheromon-Verteilung nach 100 Zyklen

Bei Vergleichen mit anderen heuristischen Optimierungsalgorithmen (u. a. Genetische Algorithmen, Evolutionssalgorithmen) in [Dorigo et al., 1996, 1996] stach die ACO-Algorithmen immer am besten ab, sowohl was die Ergebnisse<sup>1</sup> als auch die Laufzeit betraf. Qualitativ am nächsten war in den meisten Fällen der Evolutionssalgorithmus, welcher aber 50 bis 100 mal so viele Iterationen für dasselbe Ergebnis brauchte. Allgemein etwas schlechter, zeitlich aber dazwischen lag der Genetische Algorithmus.

## **ACO**

Ein weiterer Vorteil des Systems ist seine Flexibilität und Sensitivität gegenüber Änderungen in der Domäne, falls beispielsweise Verbindungen temporär unpassierbar sind. Allgemein sind ACO-Methoden gut für dynamische Optimierungsabläufe geeignet, da durch die vorhandenen Pheromonspuren immer Alternativpläne vorhanden (und nicht erst noch neu berechnet

---

<sup>1</sup>Das Optimum wurde immer erreicht.



werden müssen) sind, so daß auf Veränderungen in der Domäne schnell auf Ausweichpläne umgeschaltet werden kann.

Man vermutet, daß diese Fähigkeit den „echten Ameisen“ einen Teil ihres ökologischen Erfolges beschert hat.

## **ACO**

Seit 1997 laufen auch vermehrt Forschungen mit ACO-Algorithmen auf Parallelrechnern. [Bullinger, Kotsis & Strauss, 1997]

## **2.2.2 Ameisen als „adaptive Router“ in Netzwerken**

Viel Pheromone an wenig belasteten Netzwerkknoten und -leitungen, seien sie für Telefonverbindungen oder Teile des Internets zuständig, können die Auslastung eines Netzwerkes besser verteilen und damit Belastungsspitzen vermeiden. Die „virtuelle Verdunstung“ sorgt auch hier wieder dafür, daß Informationen nicht zu veraltet sind.

Virtuelle Ameisen, die nach dem Routing-Algorithmus „AntNet“ arbeiten, verwenden unter anderen die France Télécom und MCI Worldcom in ihren Telefonnetzen, aber auch für andere Zwecke.

## **2.2.3 Ameisen steuern Lieferwagen**

Eine weitere Anwendung ist das ebenfalls gerne als Benchmark verwendete „Vehicle Routing Problem“. Es geht dabei darum, von einem Depot aus mit so wenig Lieferwagen und Gesamtfahrtstrecke wie möglich, alle Kunden anzufahren. Verschränkte Varianten haben noch Constraints bezüglich Zuladung und maximaler Wegstrecke eines einzelnen Lieferwagens.

## **2.2.4 Arbeitsprozesse mit Ameisen verbessern**

Bei Unilever werden mit einer ACO-Variante Betriebsabläufe optimiert. In den Prozess die sind verschiedensten Maschinentypen (Verpackungsanlagen, Mischmaschinen, Lagertanks, etc.) eingebunden.

## **3 Weitere Beispiele für das Vorbild Insekt in der Informatik**

### **3.1 Teamwork ohne Kommunikation bei Ameisen und einfachen Robotern**

Das kollektive Verhalten, wie es bei sozialen Insekten vorkommt, ist eine Form dezentraler Kontrolle, welche bei der Koordination mehrerer autonomer Roboter hilfreich sein kann. In [Kube & Zhang, 1993] wird ein Versuch beschrieben, in dem mehrere nicht kommunikationsfähige Roboter eine Aufgabe bewältigen, die jeder einzelne alleine nicht geschafft hätte.

## 3.1.1 CRIP: Multi-Robot Box-Pushing

Szenario aus [Kube & Zhang, 1993]

CRIP-Roboter von [Kube & Zhang, 1993]

### **3.2 Lackierroboter verhalten sich wie Bienen**

In Staaten von Honigbienen spezialisieren sich einzelne Individuen auf bestimmte Aufgaben. Ältere Bienen tendieren dazu, für die Arbeiterinnen im Stock Nahrung zu sammeln. Aber diese Aufgabenverteilung ist nicht fix: Gibt es zu wenig Nahrung, fangen auch Arbeiterinnen an, auf Nahrungssuche zu gehen.

Nach diesem Vorbild sollen sich die Lackierroboter eines LKW-Herstellers in den USA auf verschiedene Farben spezialisieren. Die Lackierroboter können nun die verwendete Farbe wechseln, aber dieser Vorgang kostet Zeit und Geld.

Allerdings hat man noch nicht ganz verstanden, wie die Bienen die Arbeitsteilung regeln, aus diesem Grund wurde für die Programmierung der Lackierroboter von Annahmen ausgegangen.

### 3.3 Bankkunden werden wie tote Ameisen behandelt

Einige Ameisenarten (z. B. *Messor sancta*) „entsorgen“ ihre Toten, in dem sie sie zu Haufen zusammen legen. Noch genauer sortieren Arbeiter der Art *Leptothorax unifasciatus*: Sie häufen ihre Larven und Puppen in Haufen an, außen die größeren, die kleineren dazwischen.

Nach dem Muster dieses Sortierverhaltens wird an der Universität London und der Firma Intervall Research in Palo Alto (Kalifornien) eine Methode entwickelt, mit der sich große Datenbanken analysieren lassen. So wollen Banken z. B. die Kreditwürdigkeit von Bankkunden allein darüber feststellen, ob die meisten anderen Kunden im selben „Häufchen“ kreditwürdig sind bzw. . waren.

Vorteil dieser Methode ist die einfache Visualisierung der Ergebnisse trotz des hochdimensionalen Suchraumes (Alter, Geschlecht, Einkommen, Familienstand, Immobilien, bereits genutzte Dienstleistungen der Bank, etc.) relativ einfach ist.

## 4 Literatur

- Bonabeau, E. & Théraulaz, G. [2000]. „Virtuelle Ameisen als Software-Agenten.“ *Spektrum der Wissenschaft, Mai 2000*, 72-78.
- Bullnheimer, B., Hartl, R. F. & Strauss, C. [1997]. *An improved ant system algorithm for the vehicle routing problem* (Tech. Rep.). Vienna, Austria: Institute of Management Science, University of Vienna.
- Bullnheimer, B., Kotsis, G. & Strauss, C. [1997]. *Parallelization Strategies for the Ant System* (Tech. Rep. Nr. Report No. 8). Vienna, Austria: Institute of Management Science and Institute of Applied Computer Science (SFB „Adaptive Information Systems and Modelling in Economics and Management Science“), University of Vienna.
- Colorni, A., Dorigo, M. & Maniezzo, V. [1991]. „Distributed Optimization by Ant Colonies.“ In *Proceedings of European Conference Artificial Life (ECAL91)* (Seite 134-142). Paris.
- Dorigo, M. [2000]. *ACO Webseite*. URL: <http://iridia.ulb.ac.be/~mdorigo/ACO/ACO.html>.
- Dorigo, M. & Di Caro, G. [1996]. „Heuristics from Nature for Hard Combinatorial Optimization Problems.“ In *International Transactions in Operational Research* (Vols. 3, 1, Seite 1-21).

- Dorigo, M. & Di Caro, G. [1998]. „Mobile Agents for Adaptive Routing.“ In *Proceedings of the 31<sup>st</sup> Hawaii International Conference on Systems Sciences (HICSS-31) - The Software Technology Track* (Vols. Vol. 26, No. 1). The Big Island of Hawaii.
- Dorigo, M. & Gambardella, L. M. [1996a]. *Ant Colony System: A Cooperative Learning Approach to the Traveling* (Technical Report Nr. TR/IRIDIA/1996-5). Bruxelles, Belgium: IRIDIA, Université Libre de Bruxelles.
- Dorigo, M. & Gambardella, L. M. [1996b]. *Ant colonies for the traveling salesman problem* (Technical Report Nr. TR/IRIDIA/1996-3). Bruxelles, Belgium: IRIDIA, Université Libre de Bruxelles.
- Dorigo, M., Maniezzo, V. & Colorni, A. [1991]. *Ant System: An Autocatalytic Optimizing Process* (Technical Report Nr. 91-016 (rev)). Milano, Italy: Laboratorio di Calcolatori, Dipartimento di Elettronica e Informazione, Politecnico di Milano.
- Dorigo, M., Maniezzo, V. & Colorni, A. [1996]. „The Ant System: Optimization by a colony of cooperating agents.“ In *IEEE Transactions on Systems, Man, and Cybernetics - Part B* (Vols. Vol. 26, No. 1, Seite 1-13). Bruxelles, Belgium.
- Dorigo, M. & Stützle, T. [1999]. „ACO Algorithms for the Traveling Salesman Problem.“ In K. Miettinen, M. Mäkelä, P. Neittaanmäki & J. Periaux (Hrsg.), *Evolutionary Algorithms in Engineering and Computer Science: Recent*



*Advances in Genetic Algorithms, Evolution Strategies, Evolutionary Programming, Genetic Programming and Industrial Applications.* John Wiley & Sons.

Kube, C. R. [2000]. *CRIP Webseite*. URL: <http://www.cs.ualberta.ca/~kube/crip.cgi>.

Kube, C. R. & Zhang, H. [1993]. *Collective Robots: From Social Insects to Robots* (Tech. Rep. Nr. Report No. 8). Edmonton, Alberta, Canada: Department of Computing Science, University of Alberta.

Leerink, L. R., Schultz, S. R. & Jabri, M. A. [1992]. *A Reinforcement Learning Exploration Strategy based on Ant Foraging Mechanisms* (Tech. Rep.). Sydney, Australia: Department of Electrical Engineering, The University of Sydney.

Reinelt, G. [1991]. *TSP LIB - A Traveling Salesman Problem (Benchmark) Library*. URL: <http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSP LIB95/TSP LIB.html>.

SWARM. [2000]. *The Swarm Simulation System: A Toolkit for Building Multi-Agent Simulations - Webseite*. URL: <http://www.swarm.org/>.