

Seminarbericht 2012

Ameisen-Intelligenz

Von der Natur in die Informatik



Fachbereich
Modul
Studierende

Professor

Informatik
Informatik Seminar
Marc Rufer
Martin Moser
Dr. Jürgen Eckerle



Abstract

Beim genaueren Betrachten des Verhaltens bei der Futtersuche von Ameisen in der Natur fällt auf, dass sie bereits nach kurzer Zeit optimale Pfade zu den Futterstellen finden. Computerwissenschaftler haben dieses Verhalten in die Informatik übertragen, um damit Optimierungsprobleme näherungsweise lösen zu können. Dazu wurde das Verhaltensmodell in Form von Algorithmen, sogenannte Ameisenalgorithmen, umgesetzt. Meist sind solche Algorithmen auf die Suche in Graphen ausgelegt. Aus diesem Grund müssen die Probleme natürlich in Form von Graphen dargestellt werden können. Das Anwendungsgebiet für Ameisenalgorithmen in der Informatik ist riesig. Vor allem im Bereich der Optimierung sind Ameisenalgorithmen kaum mehr wegzudenken.



Inhaltsverzeichnis

Abstract	II
1 Einleitung	1
2 Von realen Ameisen zu künstlichen Ameisen	2
2.1 Reale Ameisen.....	2
2.2 Adaption in die Informatik.....	3
3 Ameisenalgorithmen	5
3.1 Die ACO-Familie	5
3.2 Ant System (AS).....	6
3.3 Ant Colony System (ACS)	8
3.4 Elitist Ant System	9
4 Travelling Salesman Problem mit AS	10
4.1 Problemstellung	10
4.2 Lösungsverfahren mit AS	10
5 Real-Life Anwendungen	13
6 Fazit	14
Glossar	15
Literaturverzeichnis	16



1 Einleitung

Im Rahmen des Moduls „Informatik Seminar“ ging es darum, einen Bericht zum Thema Ameisen-Intelligenz zu verfassen. Das Thema Ameisen-Intelligenz ist dem Begriff Schwarmintelligenz, auch bekannt als kollektive Intelligenz, untergeordnet. Das Stoffgebiet der Schwarmintelligenz befasst sich mit intelligenten Verhaltensweisen von Organismen, welche durch Kommunikation und spezifische Handlungen von Individuen hervorgerufen werden. In Bezug auf die Ameisen-Intelligenz, werden also die Verhaltensweisen einer ganzen Kolonie betrachtet.

Dieses Dokument hat zum Ziel, bestimmte Verhaltensweisen von Ameisen sowie deren Nutzen in der Informatik aufzuzeigen. Der Fokus liegt dabei auf dem Verhaltensmodell bei der Futtersuche. Doch was macht die Ameisen aus Sicht der Informatik überhaupt interessant? Wo kann Ameisen-Intelligenz in der Technik eingesetzt werden? Auf solche Fragen wird in den nachfolgenden Kapiteln eingegangen. Dafür wird das Verhaltensmodell bei der Futtersuche in der Natur betrachtet. Mit Hilfe von Duftstoffen markieren Ameisen zurückgelegte Pfade zwischen Futterquellen und ihrem Nest. Kurze Pfade werden stärker mit Duftstoffen gekennzeichnet und dadurch häufiger begangen. In der Informatik wird dieses Modell in Form von Algorithmen umgesetzt. Diese sogenannten Ameisenalgorithmen ermöglichen die Suche von optimalen Pfaden in Graphen. In diesem Bericht wird ein Hauptaugenmerk auf solche Ameisenalgorithmen gelegt. Ausserdem wird einer der Algorithmen am Beispiel des Travelling Salesman Problems demonstriert.



2 Von realen Ameisen zu künstlichen Ameisen

In diesem Kapitel wird das Verhaltensmodell bei der Futtersuche von Ameisen in der Natur erklärt und anschliessend dessen Adaption in die Informatik aufgezeigt. Das Modell steht in direktem Zusammenhang mit der Kommunikation der Ameisen. Aus diesem Grund wird hier auch auf die Kommunikation zwischen Ameisen eingegangen. Bei den Kommunikationsarten beschränkt sich der Bericht jedoch auf die Kommunikation mit Hilfe von Duftstoffen bei der Futtersuche. Doch zu Beginn erst einmal ein kurzer Überblick über die Spezies der Ameisen.

2.1 Reale Ameisen

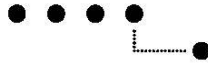
Ameisen haben Menschen schon immer fasziniert. Bei der Betrachtung von einzelnen Ameisen gewinnt man den Eindruck, sie seien rastlos, gehen in jede Richtung und es scheint, als hätten sie keine Ahnung, was sie tun. Einzelne Ameisen sind zwar nicht dumm, aber sie handeln weitgehend aufgrund von Instinkten. Erst die Zusammenarbeit von mehreren Hundert, Tausend oder sogar Millionen Ameisen ermöglicht das arttypische Verhalten.

Ameisen bewohnen unseren Planeten seit mehreren Millionen Jahren und sind in fast jedem erdenklichen Ökosystem anzutreffen. Der Grund für diesen Erfolg liegt an ihrem hochentwickelten Sozialsystem. Nicht alle Ameisen gehen derselben Beschäftigung nach. Nebst der Königin, welche für die Fortpflanzung verantwortlich ist, gibt es verschiedene Typen von Arbeiterinnen. Die einen sind beispielsweise für das Beschützen der Königin und der Brut verantwortlich, andere für die Futtersuche, etc. Es gibt also klare Strukturen innerhalb der Kolonien, welche die Arbeitsaufteilung regeln.

Ameisen können natürlich auch miteinander kommunizieren. Gerade für die Futtersuche ist die Kommunikation unumgänglich. Die Kommunikation erfolgt grösstenteils über verschiedene Duftstoffe, sogenannte Pheromone. Pheromone sind chemische Substanzen, welche im Körper von speziellen Zellen erzeugt und über Drüsen ausgeschüttet werden. Erkannt werden sie von anderen Ameisen mit Hilfe der Fühler. Jede Kolonie charakterisiert sich durch viele unterschiedliche Pheromone. Diese werden zu folgenden Zwecken eingesetzt:

- Markierung des Territoriums
- Markierung des Nesteinganges
- Erkennung von Leichen
- Erkennung der Königin
- Alarmierung
- Pfadmarkierung bei der Futtersuche

Bei der Futtersuche spielen Pheromone eine zentrale Rolle. Ameisen finden nämlich dank der Pheromone ideale Wege zu den Futterstellen. Zuerst laufen die Ameisen ziemlich zufällig vom Nest weg. Sobald sie eine Futterstelle gefunden haben, kehren sie mit etwas Futter zum Nest zurück. Auf dem Rückweg von der Futterstelle zum Nest markieren sie den zurückgelegten Weg mit Pheromonen. Nachfolgende Ameisen wählen mit höherer Wahrscheinlichkeit einen solchen Pfad. Natürlich verdunsten die Pheromone nach und nach. Dies hat zur Folge, dass längere Pfade nach einer gewissen Zeit kaum noch Duftspuren aufweisen, da sie in gleicher Zeit weniger oft begangen werden können. Dadurch werden solche Pfade von anderen Ameisen weniger begangen. Optimale, kurze Pfade weisen zur gleichen Zeit eine höhere Pheromkonzentration auf und werden von anderen Ameisen bevorzugt.



Mit der Zunahme an Pheromonen in einem gewissen Gebiet nimmt folglich auch die Anzahl Ameisen in diesem Gebiet zu, wodurch wieder die Pheromone zunehmen.

Diese Vorgehensweise bei der Futtersuche ermöglicht es den Ameisen sehr schnell optimale Pfade zu guten Futterstellen finden.

2.2 Adaption in die Informatik

Die Idee, Ameisen als Modell für die Entwicklung von Methoden zur Lösung von kombinatorischen Problemen anzusehen, war nicht das Resultat eines spontanen Prozesses. Vielmehr entstand diese Idee aufgrund von Beobachtungen in der Natur und den daraus gewonnenen Erkenntnissen.

Wieso wurde gerade das Modell der Ameisen als Basis genommen? Bei Beobachtungen von Ameisen und den Effekten ihrer Handlungen wurden bemerkenswerte Eigenschaften erkannt, welche für die Informatik von Bedeutung sind. Ameisen gehören zu der ältesten Spezies unseres Planeten. In den vielen Millionen Jahren, in denen sie unseren Planeten besiedeln, haben sie sich anatomisch kaum verändert. Trotzdem ist es ihnen gelungen, sich an teilweise drastische Änderungen in ihrer Umgebung anzupassen. Ausserdem haben Ameisen die Fähigkeit, sich selbst zu organisieren, Arbeitsaufgaben untereinander zu verteilen und neue Lebensräume zu besiedeln. All diese Eigenschaften sind in Bezug auf Optimierungsprobleme von Bedeutung. Schliesslich sollen aus dem Modell gewonnene Lösungsansätze auf verschiedene Optimierungsprobleme adaptiert werden können.

Das Verhaltensmodell der Ameisen bei der Futtersuche wird in der Informatik in Form von Algorithmen abgebildet. Ameisenalgorithmen sind ein approximatives Verfahren zur näherungsweise Lösung von Optimierungsproblemen.

Graphen eignen sich besonders gut zur Modellierung von Optimierungsproblemen. Aus diesem Grund sind Ameisenalgorithmen auf das Suchen in Graphen ausgelegt. Gesucht werden beispielsweise Routen, ein oder mehrere ideale Pfade oder eine Gruppe von Knoten. Im weiteren Verlauf beschränken wir uns auf das Suchen von Pfaden.

Ein System, welches auf sogenannten intelligenten Ameisen basiert, beinhaltet folgende Eigenschaften:

- Umgebung (in unserem Fall ein Graph)
- Einige Ameisen
- Speichermechanismus
- Mechanismus für die Vergesslichkeit

Die Funktionalität eines solchen Systems basiert als Ganzes auf der Bewegung der Ameisen in ihrer Umgebung und der Abgabe von Informationen in dieser Umgebung. Die Ameisen werden dabei mit Hilfe von Ameisenalgorithmen simuliert. Die abgelegten Informationen werden auch Pheromone genannt. Damit diese Informationen in der Umgebung persistent bleiben, braucht es einen Speichermechanismus. Pheromone dienen der indirekten Kommunikation unter den Ameisen, indem die in der Umgebung abgelegten Informationen von den Ameisenalgorithmen für die Suche berücksichtigt werden. In Bezug auf Graphen werden die Pheromone auf den Kanten und Knoten abgelegt. Ameisen entscheiden aufgrund der Pheromone, welchen Weg sie einschlagen. Eine Ameise wählt mit höherer Wahrscheinlichkeit eine Kante mit einer hohen Pheromonzkonzentration, als eine Kante mit



geringer Konzentration. Folge dessen werden die künstlichen Ameisen von Gebieten mit hoher Pheromonkonzentration angezogen. Statistisch gesehen steigt die Anzahl von Ameisen in einer Region mit der Konzentration der Pheromone, wodurch sich wieder die Anzahl der Ameisen erhöht. Um diesen Effekt etwas abzuschwächen braucht es einen Mechanismus für die Vergesslichkeit. Dieser Mechanismus wird in Form der Pheromon-Evaporation, also der Verdunstung von Pheromonen, implementiert.

Die Ameisenalgorithmen, welche im nächsten Kapitel vorgestellt werden, wurden auf Basis eines bestimmten Optimierungsproblems entwickelt. Die Ausgangslage dabei ist ein Graph, worin alle Knoten miteinander verbunden sind. Die gesuchte Lösungsstruktur ist ein Pfad in Form eines Graphen, der jeden Knoten genau einmal enthält. Die Aufgabe der Algorithmen besteht darin, ausgehend von einem Startknoten, einen optimalen Pfad zu finden bei dem jeder Knoten im gegebenen Graphen genau einmal besucht wird. Um das Problem lösen zu können, muss noch die Distanz zwischen den Knoten berücksichtigt werden. Dazu werden auf den Kanten Distanzattribute eingesetzt. Die Algorithmen müssen also bei der Entscheidung, welchen Weg sie nehmen, einerseits die Menge der Pheromone auf den verschiedenen Kanten, aber auch die Distanz berücksichtigen.

Die Lösung eines Optimierungsproblems mit Hilfe eines Ameisenalgorithmus ist ein Ablauf, der in folgende Teilabläufe unterteilt werden kann:

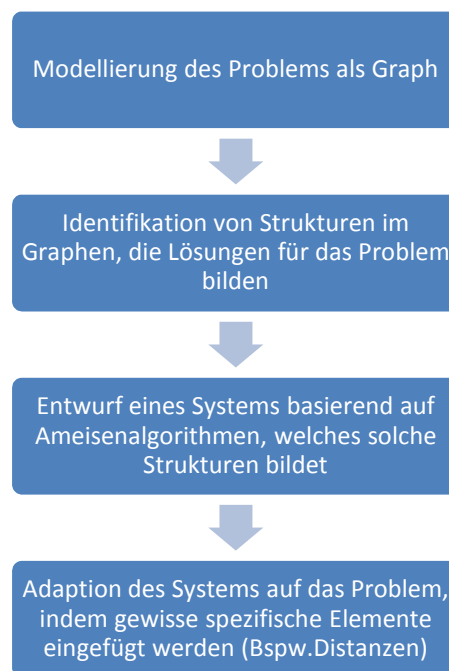


Abbildung 1: Schematischer Ablauf bei der Lösung eines Optimierungsproblems mit Ameisenalgorithmen

Wie in Abbildung 1 dargestellt, können solche Systeme mit Ameisenalgorithmen auch auf andere Probleme adaptiert werden. Beispielsweise auf die Suche von optimalen Pfaden zwischen 2 gegebenen Punkten (z.B. Nest und Futterquelle). Dazu müsste das System und die Algorithmen aber spezifisch auf dieses Problem angepasst werden.



3 Ameisenalgorithmen

Ameisenalgorithmen gehören zu den Metaheuristiken, die auf dem modellhaften Verhalten von realen Ameisen bei der Futtersuche basieren.

Um die nachfolgend vorgestellten Algorithmen verstehen zu können sind noch einige wichtige Informationen bezüglich der allgemeinen Funktionsweise solcher Algorithmen nötig.

Damit Systeme, basierend auf Ameisenalgorithmen, Lösungen für Optimierungsprobleme finden können, müssen mehrere künstliche Ameisen, eine sogenannte Generation, simuliert werden und der ganze Prozess mehrmals wiederholt werden. Die Ameisen einer Generation werden zufällig auf verschiedene Knoten im Graph abgesetzt. Von diesem Startknoten aus suchen sie individuell einen Pfad. Sobald alle Ameisen jeden Knoten genau einmal erreicht haben, werden die Pheromone, bezüglich der Formel des jeweiligen Algorithmus, auf allen Kanten aktualisiert. Der gesamte Vorgang wird anschliessend mit mehreren Generationen wiederholt. Die Grösse einer Generation, sowie die Anzahl Wiederholungen der Durchläufe ist Abhängig vom jeweiligen Optimierungsproblem.

Es gibt auch Ansätze, bei denen die Pheromone jedes Mal, wenn eine Ameise einen neuen Knoten erreicht, aktualisiert werden. Dieser Bericht befasst sich aber ausschliesslich mit dem erstgenannten Ansatz. Nach dem letzten Durchlauf, wird der Weg, auf den die Ameisen der letzten Generation konvergieren, als Lösung gewählt.

Vor dem ersten Durchlauf des Algorithmus, wenn die Kanten noch frei von Pheromonen sind, entscheiden die Ameisen zufällig, welchen Pfad sie wählen. Es gibt aber auch die Möglichkeit, die Kanten mit einer geringen Pheromonkonzentration zu initialisieren.

3.1 Die ACO-Familie

Die Methode mit dem Namen *Ant System* (AS), erschien erstmals 1991 in einer Arbeit von Marco Dorigo. Dabei handelt es sich um eine Metaheuristik für die Optimierung von kombinatorischen Problemen. Der Algorithmus basiert auf dem Verhalten und der asynchronen Kommunikation von Ameisenpopulationen. Das AS war der erste Algorithmus seiner Art. Alle Methoden, respektive Algorithmen, welche den Ansatz des AS verfolgen gehören zur Familie der Ant Colony Optimization (ACO).

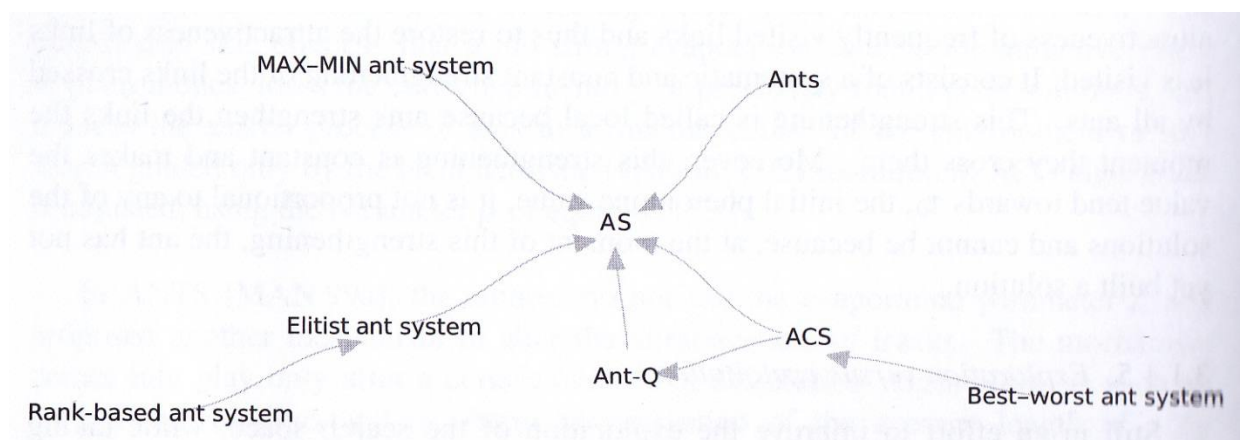


Abbildung 2: Vererbungsgraph der ACO-Familie. Ein Pfeil von A nach B bedeutet, dass der Algorithmus A von B erbt. (Quelle: Buch „Artificial Ants“)



Die nachfolgend vorgestellten Algorithmen gehören, wie in Abbildung 2 ersichtlich, zur ACO-Familie, wobei nur einige der oben abgebildeten Algorithmen vorgestellt werden.

3.2 Ant System (AS)

Der Algorithmus Namens Ant System wurde ursprünglich für das Travelling Salesman Problem (TSP) entwickelt, welches in Kapitel 4.1 vorgestellt wird. Das AS kann jedoch ziemlich einfach auch auf andere Optimierungsprobleme adaptiert werden.

Der AS-Algorithmus ist sehr übersichtlich und dient, wie in Abbildung 2 ersichtlich, als Grundlage für viele weiterentwickelte Ameisenalgorithmen. Die Ausgangslage für das AS ist ein Graph bestehend aus einem Set N von Knoten und einem Set E von Kanten, welche die Knoten miteinander verbinden. Ausserdem ist jede Kante mit einem Wert d_{ij} versehen, welcher die Distanz zwischen den Knoten i und j beschreibt.

Auf den Knoten des Graphen werden vom AS nun die künstlichen Ameisen einer Generation platziert, deren Aufgabe darin besteht, kürzeste Pfade zu suchen. Die Funktion dieser Generation kann folgendermassen schematisiert werden:

- In einem iterativen Verfahren entscheidet sich nun jede Ameise, ausgehend von ihrem aktuellen Standort, welchen Knoten sie als nächstes besucht. Die Entscheidung erfolgt aufgrund der Attraktivität der verfügbaren Knoten.
- Knoten, die bereits besucht wurden, werden von den Ameisen nicht mehr berücksichtigt. So wird der mehrfache Besuch eines Knotens ausgeschlossen
- Sobald die Ameisen alle Knoten einmal besucht haben, kehren sie zum Startpunkt zurück.
- Anschliessend wird die Attraktivität der Kanten verändert, indem die Ameisen auf alle von ihnen besuchten Kanten Pheromone hinterlegen

Eine Iteration beinhaltet die Entscheidung, welcher Knoten als nächstes besucht wird, und die anschliessende Bewegung zum ausgewählten Knoten. Eine Ameise k entscheidet also zur Zeit t , auf welchem Knoten sie sich zur Zeit $t + 1$ befinden wird. Nach n Iterationen - n ist die Anzahl Knoten im Graphen - werden alle Ameisen wieder an ihrem jeweiligen Startpunkt sein. Damit jede Ameise weiss, welche Knoten sie bereits besucht hat, unterhalten alle Ameisen eine sogenannte Tabuliste. Darin werden bereits besuchte Knoten in der entsprechenden Reihenfolge abgespeichert. Bei der Wahl des nächsten zu besuchenden Knotens werden von einer Ameise nur noch Knoten berücksichtigt, welche noch nicht in ihrer Tabuliste vorhanden sind.

Die Attraktivität von Kanten in einem Graphen basiert auf deren Pheromonmenge. Sei $\tau_{ij}(t)$ die Menge der Pheromone auf der Kante zwischen den Knoten i und j zur Zeit t . Die Aktualisierung der Pheromonmenge erfolgt aufgrund folgender Formel:

$$\tau_{ij}(t + n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}$$

ρ dient als Parameter zur Justierung der Erhaltung der Pheromone. Somit entspricht $(1 - \rho)$ dem Verdunstungs- respektive Evaporationsfaktor.



$\Delta\tau_{ij}$ lässt sich folgendermassen berechnen:

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k$$

$\Delta\tau_{ij}^k$ ist die Menge der Pheromone, die von der Ameise k auf der Kante (i, j) während eines Durchlaufs deponiert wird. m ist die Anzahl Ameisen einer Generation.

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{wenn Ameise } k \text{ die Kante } (i, j) \text{ im aktuellen Durchlauf überquert hat} \\ 0 & \text{sonst} \end{cases}$$

Dabei ist Q eine Konstante und L_k die Länge des Pfades, welcher von der Ameise k gefunden wurde. Die Division durch L_k hat zur Folge, dass bei langen Pfaden weniger Duftstoff auf den entsprechenden Kanten hinterlassen wird. Dadurch haben schlechte Pfade weniger Einfluss auf die Lösung.

Die Wahrscheinlichkeit, dass sich eine Ameise, aus Sicht ihrer Position, für eine bestimmte Kante entscheidet, hängt einerseits von deren Attraktivität ab. Aber auch die Distanz der Kante hat einen Einfluss. Dazu wird die Menge der Pheromone auf einer Kante mit dem reziproken Wert der Distanz multipliziert:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in \text{Nachbarn}_k} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta} & \text{wenn } j \in \text{Nachbarn}_k \\ 0 & \text{sonst} \end{cases}$$

$$\eta_{ij} = \frac{1}{d_{ij}}$$

Nachbarn_k ist ein Set von Knoten, welche von der Ameise k noch nicht besucht worden sind. α und β sind zwei Parameter, womit die Gewichtung der Pheromone im Verhältnis zur Distanz justiert werden kann.



Abschliessend kann der AS-Algorithmus, in Form von Pseudocode, wie folgt beschrieben werden:

```

1. Initialisierung
   -  $t \leftarrow 0$ 
   -  $\tau_{ij}(t) \leftarrow c$ 
2. for  $k = 1$  to  $m$  do
3.     Ameise  $k$  zufällig auf einen Knoten  $v$  setzen
4.     Knoten  $v$  in Tabuliste  $tabu_k$  einfügen
5. end for
6. for  $t = 1$  to  $n - 1$  do
7.     for  $k = 1$  to  $m$  do
8.         Wahl des nächsten Knoten  $v'$  anhand der Wahrscheinlichkeiten  $p_{ij}^k(t)$ 
9.         Ameise  $k$  auf den Knoten  $v'$  bewegen
10.        Knoten  $v'$  in  $tabu_k$  einfügen
11.    end for
12. end for
13. Aktualisieren der Pheromonwerte anhand der Gleichung
    
$$\tau_{ij}(t + n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}$$


```

Abbildung 3: Pseudocode des AS-Algorithmus

3.3 Ant Colony System (ACS)

Der Ant Colony System Algorithmus, ist eine Weiterentwicklung auf Basis des Ant Systems und stammt aus dem Jahre 1997. Entwickelt wurde er von Luca Maria Gamberdella. Der Unterschied zum AS-Algorithmus besteht darin, dass nur der beste Pfad, welcher während eines Durchlaufs von einer Ameise gefunden wurde, mit Pheromonen gekennzeichnet wird. Sei S_{ib} die beste gefundene Lösung, also der kürzeste gefundene Pfad, so werden die Pheromonwerte der Kanten, welche in S_{ib} enthalten sind, folgendermassen aktualisiert:

$$\Delta\tau_{ij} = \frac{1}{L_{ib}}$$

Dabei ist L_{ib} die Länge des kürzesten Pfades.

Die Pheromonwerte aller Kanten, die in den anderen gefundenen Pfaden enthalten sind, werden somit durch den Evaporationsfaktor verringert. Folglich werden diese Kanten für die Ameisen weniger attraktiv, da sich deren Pheromonwert dem Initialwert τ_0 annähert.

Der Sinn dieses Algorithmus besteht also darin, weniger oft oder gar nie begangene Kanten attraktiver zu machen. Dadurch erhöht sich die Wahrscheinlichkeit, dass alle Kanten mindestens einmal überquert werden. Es wird also die Erkundung des gesamten Graphen gefördert.



3.4 Elitist Ant System

Der Elitist Ant System Algorithmus ist ebenfalls eine direkte Weiterentwicklung des Ant Systems. Die Idee hinter diesem Algorithmus ist, die global beste Lösung S_{gb} im Aktualisierungsprozess der Pheromonwerte zu berücksichtigen. Mit der global besten Lösung ist der kürzeste Pfad seit Beginn der Suche gemeint. Dazu wird der Aktualisierungsprozess des Pheromonwerts der Kanten, die der global besten Lösung angehören, zusätzlich erhöht.

Dafür wurde die Grundformel $\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}$ zur Aktualisierung der Pheromonwerte aus dem AS angepasst:

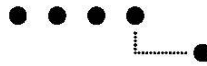
$$\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij} + \Delta\tau_{ij}^*$$

$\Delta\tau_{ij}^*$ wird folgendermassen berechnet:

$$\Delta\tau_{ij}^* = \begin{cases} \sigma \cdot \frac{Q}{L_{gb}} & \text{wenn die Kante } (i,j) \text{ zur global besten Lösung } S_{gb} \text{ gehört} \\ 0 & \text{sonst} \end{cases}$$

L_{gb} beschreibt die Länge von S_{gb} . Der Parameter σ wird > 1 gewählt und dient dazu, die Gewichtung der global besten Lösung im Aktualisierungsprozess einzustellen.

Mit den oben gezeigten Anpassungen ist es möglich, die Attraktivität der Kanten, welche zu S_{gb} gehören, für die nächsten Ameisen zu steigern.



4 Travelling Salesman Problem mit AS

Im ersten Teil dieses Kapitels wird das Travelling Salesman Problem, abgekürzt mit TSP, vorgestellt. Im zweiten Teil werden anhand einer visuellen Demonstration die Effekte des Ant System Algorithmus bei der Anwendung auf das TSP aufgezeigt.

4.1 Problemstellung

Worum geht es beim Travelling Salesman Problem überhaupt?

Beim TSP handelt es sich um ein in der Graphentheorie sehr bekanntes Problem. Gegeben sind eine bestimmte Anzahl Städte. Die Aufgabe besteht nun darin, für einen Handlungsreisenden eine möglichst kurze Reiseroute zu suchen, wobei jede der n Städte genau einmal besucht werden darf. Von einer bestimmten Stadt aus kann jede andere Stadt erreicht werden.

4.2 Lösungsverfahren mit AS

Es gibt eine Vielzahl von heuristischen Methoden, um dieses Problem optimal lösen zu können. Der Ant System Algorithmus stellt eine Möglichkeit dar, dieses Problem optimal zu lösen.

Dazu muss das Problem in Form eines Graphen modelliert werden:

- Knoten = Städte
- Kanten = Wege zwischen den Städten
- Jede Kante wird mit einer Distanzangabe versehen
- Jede Kante wird mit einem Initialwert an Pheromonen versehen

Pro Durchlauf des Algorithmus werden dann m Ameisen auf die Reise geschickt.

Die Effekte des Ant System Algorithmus werden nachfolgend anhand von Bildern aus einer Simulation aufgezeigt. Die Ausgangslage sieht wie folgt aus:

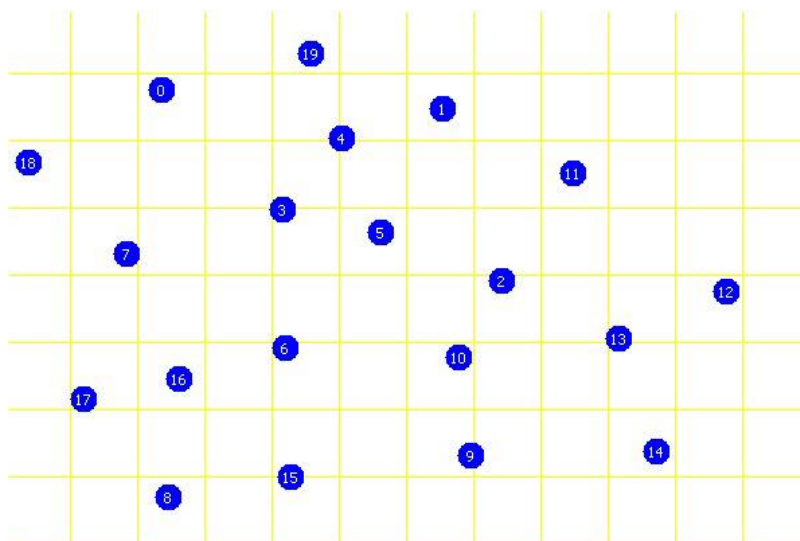
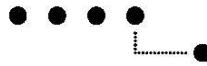


Abbildung 4: Ausgangslage (Quelle: <http://yzgrafik.ege.edu.tr/projects/TSPAntSim/TSPAntSim.html>)



Es sind also 20 Städte (blaue Punkte) gegeben. Mit Hilfe des gelb hinterlegten Gitternetzes wird die Distanz zwischen den Städten berechnet. Nun werden 10 künstliche Ameisen nach den Regeln des AS-Algorithmus auf die Reise geschickt.

Nach 5 Durchläufen ergibt sich folgendes Bild

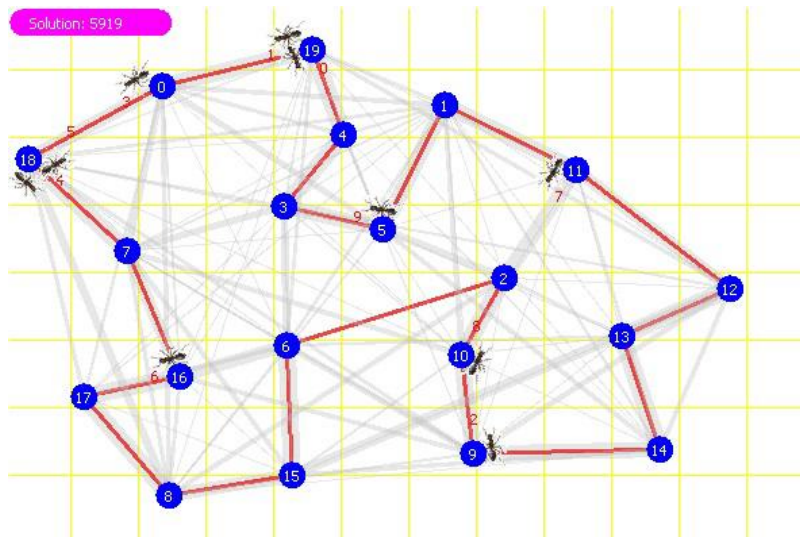


Abbildung 5: Lösung nach 5 Durchläufen (Quelle: <http://yzgrafik.ege.edu.tr/projects/TSPAntSim/TSPAntSim.html>)

Die bis dahin gefundene Lösung ist rot markiert. Die Menge der Pheromone wird anhand der Stärke der grauen Linien dargestellt.

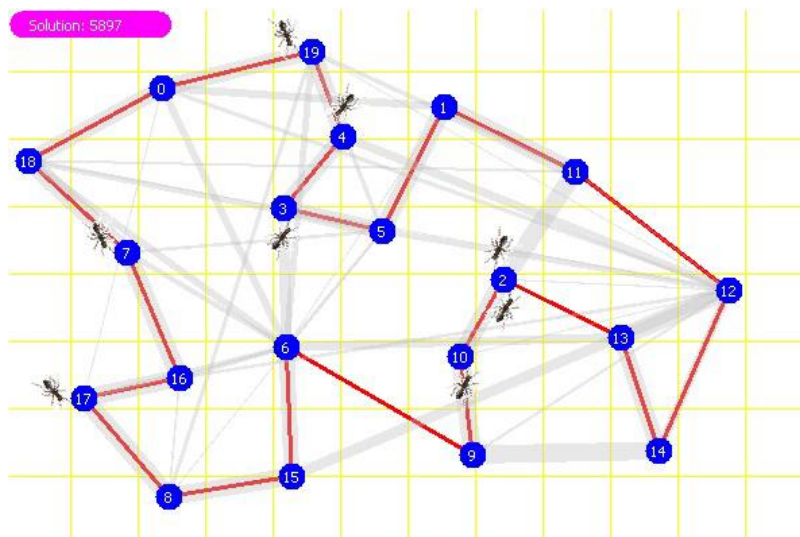


Abbildung 6: Lösung nach 20 Durchläufen (Quelle: <http://yzgrafik.ege.edu.tr/projects/TSPAntSim/TSPAntSim.html>)

Nach 20 Durchläufen haben die Ameisen die bestehende Lösung optimiert. Ausserdem sind die Pheromonwerte einiger Kanten aufgrund der Evaporation bereits auf den Wert τ_0 gesunken und sind dadurch im Bild nicht mehr sichtbar.

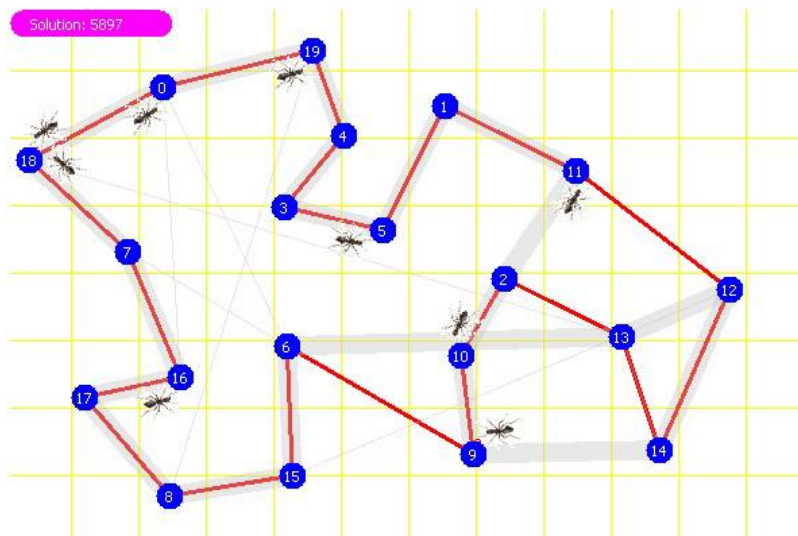
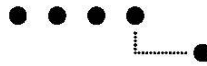
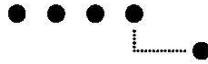


Abbildung 7: Lösung nach 50 Durchläufen (Quelle: <http://yzgrafik.ege.edu.tr/projects/TSPAntSim/TSPAntSim.html>)

Nach 50 Durchläufen ist die Simulation beendet. Das System hat mit Hilfe des AS-Algorithmus eine optimale Lösung gefunden. In den letzten 30 Durchläufen hat sich der Lösungspfad nicht mehr verändert, sondern nur noch verstärkt. Die Verstärkung des Lösungspfadess erfolgte auf Kosten einiger Kanten, welche aufgrund der Verdunstung im Bild gar nicht mehr erscheinen. Der Lösungspfad würde sich auch nach weiteren Durchläufen wohl kaum verändern.

Vergleicht man Lösungspfade dieses Problems, welche mit dem AS-Algorithmus gefunden wurden, mit gefundenen Pfaden aufgrund anderer Heuristiken stellt man fest, dass die vom AS-Algorithmus produzierten Lösungspfade für maximal 50 Städte von sehr guter Qualität sind. Enthält der Graph mehr als 50 Städte werden andere Ameisenalgorithmen verwendet, da diese bessere Lösungen finden können.



5 Real-Life Anwendungen

Auch in der realen Welt existieren viele Optimierungsprobleme, bei denen die verschiedenen Ameisenalgorithmen eingesetzt werden können, um optimale Lösungen zu finden. Nachfolgend werden einige Anwendungsbereiche genannt und kurz erklärt, wozu solche Algorithmen im entsprechenden Bereich angewendet werden können.

- **Logistik**

- Zur Berechnung von Ausliefer Routen (Beispielsweise für Gütertransport-, Post- oder Busrouten).
- Zur Steuerung von Hochregallagern mit mehreren Liften und Schächten.

- **Produktion**

Zur Losbildung, um Rüstzeiten minimieren zu können unter Berücksichtigung von Endterminen.

- **Telefonie**

- Zur schnellen Suche von freien Strecken während des Betriebs (z.B. Antnet).
- Zur optimalen Planung von Telefonnetzen.

- **Personaleinsatzplanung**

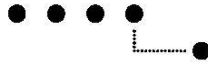
Bei Fluggesellschaften werden Flugbegleiter und Piloten unter Berücksichtigung von Ruhephasen monatlich eingeteilt.



6 Fazit

Wie aus dem Bericht hervorgeht, sind Ameisenalgorithmen ein wichtiges Hilfsmittel, um gute Lösungen für Optimierungsprobleme finden zu können. Vor der Entwicklung solcher Algorithmen war es nicht möglich für einige Optimierungsprobleme gute Lösungen zu finden. In der Informatik ist die Ameisen-Intelligenz kaum mehr wegzudenken. Die Anwendungsmöglichkeiten sind noch lange nicht ausgeschöpft. Basierend auf dem Grundmodell werden immer wieder neue Ansätze entwickelt, um Lösungen für neue Probleme zu finden.

Auch in anderen technischen Gebieten orientiert man sich oft an der Natur um gewisse Probleme lösen zu können. Die Adaption erfolgt aber überall auf dieselbe Art und Weise. Das Modell aus der Natur muss in die Technik übernommen und auf das gegebene Problem angepasst und gegebenenfalls erweitert werden.



Glossar

Graph	Stellt eine abstrakte Datenstruktur, bestehend aus Objekten und deren Verbindungen untereinander, dar. Die Objekte werden dabei als „Knoten“ und die Verbindungen als „Kanten“ bezeichnet
Algorithmus	Ein Handlungsablauf zur Lösung eines Problems, welcher aus einer endlichen Anzahl von eindeutigen Schritten besteht
Pheromon	Botenstoffe, respektive organische Moleküle, die zur biochemischen Kommunikation zwischen Lebewesen dienen
Evaporation	Verdunstung von Pheromonen
Route	Beschreibt den genauen Weg zwischen mehreren Punkten. In Bezug auf Graphen kann in einer Route eine bestimmte Kante oder ein bestimmter Knoten mehrmals auftauchen
Pfad	Beschreibt den genauen Weg zwischen mehreren Punkten. In Bezug auf Graphen können in einem Pfad Kanten und Knoten nicht mehrfach auftauchen
Metaheuristik	Bezeichnet in der Informatik einen Algorithmus zur näherungsweisen Lösung von Optimierungsproblemen
Reziproker Wert	Kehrwert einer Zahl. (Bsp.: Zahl = 4, reziproker Wert = $\frac{1}{4}$)
Heuristik	Bezeichnet das analytische Vorgehen, bei dem mit begrenztem Wissen und wenig Zeit gute Lösungen gefunden werden können



Literaturverzeichnis

Literatureintrag

Nicolas Monmarché, Frédéric Guinand, Patrick Siarry,
From Collective Intelligence to Real-life Optimization and Beyond, Artificial Ants,
ISTE Ltd, Great Britain (2010)

Literatureintrag

Silke Brand, Ant Colony Optimization - Ameisenkolonie-Optimierung, Grin Verlag,
Norderstedt Deutschland, 1. Auflage (2008)

Literatureintrag

Mathias Plüss, Die einzelne Ameise ist keine Ameise, Das Magazin, 38: 13-19 (2012)