

Intelligenz von Schwärmen:
Grundlagen, Simulations-Modelle und Anwendungen

Nicolai Marquardt
nicolai.marquardt@medien.uni-weimar.de
VR-Seminar Prof. B. Fröhlich
Wintersemester 2003/2004

Bauhaus-Universität Weimar
2. April 2004

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 1 | Einleitung | 2 |
| 2 | Grundlagen | 3 |
| 2.1 | Ameisen-Kolonien | 3 |
| 2.2 | Orientierung und Kommunikation von Ameisen | 3 |
| 2.3 | Osmotropotaxis | 4 |
| 3 | Simulations-Modell | 5 |
| 3.1 | Modell der Proto-Schwärme | 5 |
| 3.1.1 | Grundlagen des Modells | 5 |
| 3.1.2 | Drehungen und Richtungswechsel | 7 |
| 3.1.3 | Lernprozess und Erinnerung | 7 |
| 3.1.4 | Zusammenfassung | 8 |
| 3.2 | Simulation | 8 |
| 3.3 | Auswertungen und Ergebnisse | 9 |
| 3.3.1 | Memory-Effect | 10 |
| 3.3.2 | Bootstrapping | 11 |
| 4 | Andere Modelle und Simulationen | 13 |
| 4.1 | Picking and Dropping | 13 |
| 4.2 | Distanz-Bestimmungen im Parameter-Raum | 15 |
| 4.3 | Erweiterungen | 16 |
| 4.4 | Ansatz von Vitorino Ramos | 16 |
| 4.5 | ACLUSTER-Algorithmus | 17 |
| 5 | Anwendungen | 20 |
| 5.1 | Clustering von Web-Usage-Data | 20 |
| 5.2 | Clustering von Mustern (Continuous Data) | 21 |
| 5.3 | Clustering von Texten und Inhalten | 22 |
| 6 | Zusammenfassung und Ausblick | 24 |
| A | Acluster-Algorithmus | 25 |

Kapitel 1

Einleitung

Die Ideen und Theorien zur Intelligenz von Schwärmen stellen einen der spannenden Forschungsbereiche dar, in dem sich Untersuchungen von Biologen, Mathematikern und Informatikern treffen und erst durch deren Vernetzung neue Ansätze und Algorithmen entwickelt werden können.

In der folgenden Ausarbeitung werde ich einige der Theorien zur Schwarm-Intelligenz erläutern und die entwickelten Modelle und Simulationen ausführlich betrachten. Zunächst werden die Eigenschaften real existierender Schwärme (Biologie) erläutert, um schließlich im folgenden Kapitel das Modell der Proto-Schwärme nach Chialvo und Millonas zu entwickeln. Darauf folgen weitere Ansätze der Schwarm-Simulation (vor allem zum Aufheben und Ablegen von Objekten), und ich gehe näher auf den von Vitorino Ramos entwickelten ACLUSTER-Algorithmus ein. Die Anwendung dieses Algorithmus wird im letzten Kapitel anhand von drei Simulationen betrachtet.

Die zentrale Frage zur Schwarm-Intelligenz lautet sicher: Was unterscheidet diese Algorithmen von bestehenden Lösungen und sind sie diesen über- oder unterlegen? Aber sicher gehört auch die Frage dazu, ob sich durch diese Art der Lösungsfindung ganz neue Möglichkeiten ergeben.

Prinzipiell geht es bei den Simulationen ausgehend von dem Verhalten der Schwärme vor allem darum, eine Problemstellung dadurch zu lösen, daß einfache, beschränkte Agenten in einem begrenzten Umfeld agieren und wahrnehmen können. Die einzelnen Agenten überschauen nur ihr lokales Umfeld und somit ist die Betrachtung der sich ergebenden Lösung erst auf einer übergeordneten Ebene möglich. Das bedeutet, daß die Agenten zwar mit ihrem Verhalten und den Aktionen zu einer Lösung des Problems beitragen, die Lösung selbst jedoch nicht überschauen können. Es ergibt sich ein globales Problemlösemuster, das weit über die Erkenntnisse des einzelnen Individuums hinausgeht. Die Umgebung dient also als das Medium der Kommunikation. Weitere daraus resultierende Eigenschaften und eine ausführlichere Erklärung folgt in den nächsten Kapiteln.

Kapitel 2

Grundlagen

2.1 Ameisen-Kolonien

Die Kolonien bestehen aus vielen, einfachen Individuen. Jedes dieser Individuen hat dabei eine überschaubare Anzahl von definierten Verhaltensregeln (die jedoch durchaus auch in Erweiterungen später während der Simulation modifiziert werden können).

Ein einzelnes Individuum hat nur sehr geringe kognitive Fähigkeiten, d.h. daß fast nie eine Erinnerung/Speicherung von Informationen über einen längeren Zeitraum stattfindet. Es wird nur das Umfeld der lokalen Umgebung betrachtet (z.B. Pheromonerfassung) und auch nur in diesem Umfeld sind Aktionen möglich (wie z.B. das später betrachtete Aufheben und Ablegen von Objekten). Wichtig ist, dass es keine direkte Kommunikation zwischen den Agenten des Systems gibt.

Aus den genannten Eigenschaften und zahlreichen Studien zum Verhalten der Ameisen-Kolonien ergibt sich eine ganz wesentliche Charakteristik:

1. **Flexibel**

Sie können sich sehr schnell an eine geänderte Umgebung anpassen. Dies trifft sowohl auf Umwelteinflüsse zu als auch auf unvorhergesehene Ereignisse.

2. **Robust**

Auch wenn einzelne Ameisen oder ganze Gruppen ausfallen, kann das System noch bestehen und zu einer Problemlösung gelangen. Dies liegt vor allem an dem geringen (bis nicht vorhandenen) Grad der Spezialisierung einzelner Agenten. Um so höher der Grad der Spezialisierung ansteigt, desto drastischer würde sich das Ausscheiden einzelner Agenten im Gesamtsystem bemerkbar machen.

3. **Selbstorganisierend**

Es ist keine übergeordnete Koordinierungsebene notwendig, denn alle Aktivitäten der Ameisen in ihrer Umgebung werden von dem Individuum direkt kontrolliert.

2.2 Orientierung und Kommunikation von Ameisen

Ameisen, bzw. auch Insekten im allgemeinen, haben verschiedene Möglichkeiten sich in ihrem Terrain zu orientieren. So findet zum Beispiel die optische Orientierung anhand von Geländemarken oder auffälligen Gegebenheiten statt. Auch nach Lichtquellen können sich die Ameisen orientieren: bei der Bewegung auf Lichtquellen zu spricht man von der *Phototaxis*, bei der Bewegung in einem bestimmten Winkel dazu von der *Menotaxis*. Zu letzterem wäre auch die Orientierung nach der Sonne zu nennen. Auch eine Orientierung anhand der Schwerkraft ist möglich (*Geomenotaxis*); es wird dann ein bestimmter Winkel relativ zur Erdanziehung beibehalten. [WIL90]

Eine weitere wichtige Orientierungsmöglichkeit der Ameisen ist die chemische Absonderung von Pheromonen und deren Erfassung (Messung der Konzentration). Zwei unterschiedliche Methoden werden unterschieden: das Erfassen von Konzentrationen an zwei aufeinander folgenden unterschiedlichen Stellen und anschließender Konzentrationsvergleich (*Klimotaxis*) und das gleichzeitige Erfassen von Konzentrationen mit zwei Fühlern und somit der unmittelbare Vergleich der gefühlten Konzentrationen (*Osmotropotaxis*). Letztere ist die entscheidende Basis für die späteren Simulations-Modelle.

Schließlich haben Ameisen auch die Möglichkeit direkt zu Kommunizieren. Dies kann durch ihre Körpersprache (Bewegungen, Abfolgen, Tänze), die Fühlersprache oder auch Akustisch geschehen. Eingesetzt wird diese Kommunikation vor allem bei der Rekrutierung (eine Ameise möchte dass ihr weitere folgen) und der Alarmierung (im Falle einer drohenden Gefahr) [WIL90]. Die Koordinierung aller üblichen Tätigkeiten und Aktionen findet aber (wie bereits erwähnt) ausschließlich indirekt statt: durch Pheromon-Konzentrationen, das umgebende Terrain und die Aktionen der anderen Ameisen.

2.3 Osmotropotaxis

Bei der bereits erwähnten Osmotropotaxis werden von den Ameisen durch Drüsen produzierte Pheromon-Markierungen hinterlassen. Diese Pheromon-Konzentrationen können nun von den zwei Fühlern der Ameise erfasst und verglichen werden. Vorallem auf ihrer Suche nach neuen Futterquellen erweist sich diese Methode als sehr hilfreich.¹

Es findet bei der Absonderung der Pheromone auch eine Diffusion um die Spurlinie herum, d.h. dass sich die Pheromone auch (in geringerer Konzentration) um den Ort der Aussonderung feststellen lassen. Eine Polarisierung (also erkennbare Unterscheidung zwischen Richtungen) gibt es nicht. Wichtig ist auch, daß die Pheromon-Konzentrationen mit der Zeit schwächer werden, denn die Pheromone verdampfen. Nach Untersuchungen von Wilson sinken bei der Feuerameise die Konzentration einer Pheromonmarkierung nach etwa 100 Sekunden unter die verhaltensauslösende Schwelle [WIL90]. Auch wenn viele Aspekte zur Osmotropotaxis und den Pheromon-Markierungen bereits untersucht wurden, sind jedoch auch noch viele der Eigenschaften unbekannt und erfordern noch eingehendere Forschungen der Biologen.

¹Das lässt sich leicht veranschaulichen: Gehen vom Ameisenbau aus zwei Ameisen zu einer Futterquelle so ist diejenige Ameise am schnellsten zurück am Ameisenbau, die den kürzeren Weg zurücklegt. Wenn ihr nun weitere Ameisen folgen, findet eine stetige Verstärkung der Pheromon-Pfade entlang dieses kürzesten Weges statt.

Kapitel 3

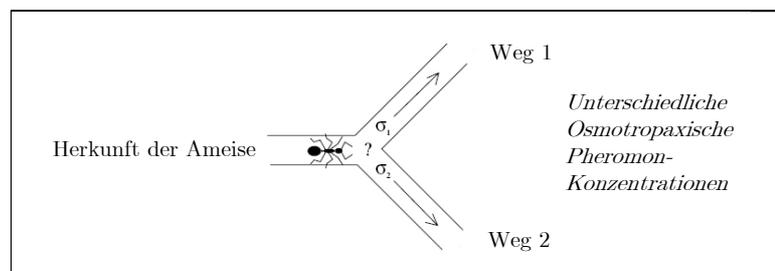
Simulations-Modell

3.1 Modell der Proto-Schwärme

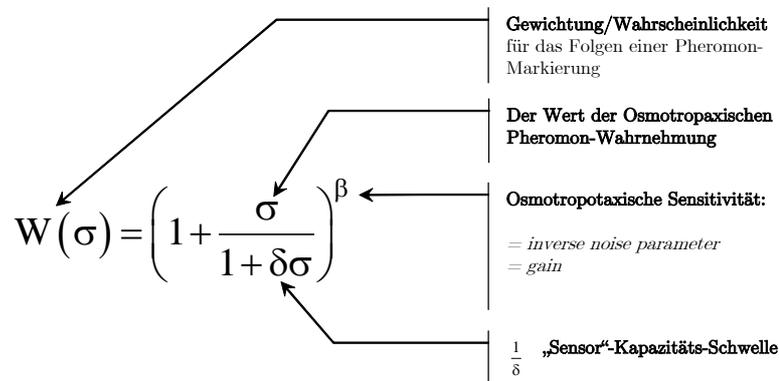
3.1.1 Grundlagen des Modells

Um diese Eigenschaften der Schwärme nun in einer Simulation abbilden zu können, entwickelten Chialvo und Millonas das Modell der Proto-Schwärme [CHI95]. Anforderungen an dieses Modell waren, dass nur eine minimale Kombination an Signalen (bzw. verändernden Variablen) benötigt wird und daß die Simulation in einer isolierten, begrenzten Umgebung stattfindet. Indem nämlich auf möglichst einfache Weise die Strukturen und Vorgänge der Natur abgebildet werden sollte es möglich werden, die zugrunde liegenden Mechanismen zu untersuchen. Deshalb wurden durchaus einige der Komponenten der Realität ausgelassen, mit dem positiven Effekt, eine analysierbare Situation zu erhalten [CHI95].

Wir betrachten nun die Entscheidung der Ameise bei ihrer Wegfindung. Während sich die Ameise auf ihrem Pfad bewegt hat sie immer wieder die Wahl zwischen verschiedenen Richtungen: sie kann die Pheromon-Konzentrationen der Richtungen erfassen und vergleichen.

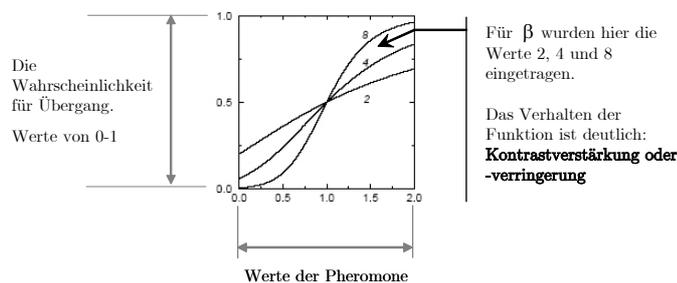


Die aktuelle Position jeder Ameise wird durch eine Phasenvariable dargestellt: von der aktuellen Position r mit dem Winkel θ geht die Ameise über zu Position r' mit dem Winkel θ' . Der Winkel θ gibt dabei die Orientierung der Ameise an. Für diesen Übergang wurde nun eine Übergangs-/Reaktionsfunktion gebildet. Diese basiert zum einen auf Experimenten mit Ameisen-Kolonien, zum anderen aber auch auf Erkenntnissen der Physik zum Noise Potential Gradient"[MIL94]. Es wurde nun durch Millonas und Chialvo eine Transformation dieser komplexen Formeln in eine Übergangsregel mit nur noch zwei Parametern durchgeführt:

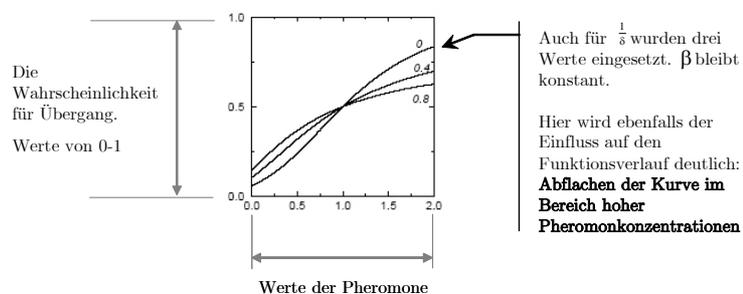


$W(\sigma)$ gibt nun für alle gemessenen Pheromon-Konzentrationen die Wahrscheinlichkeit für das Wählen des entsprechenden Pfades.

Die Osmotropotaxische Sensitivität ist eine Regulierung der Gewichtung von unterschiedlichen gemessenen Pheromon-Werten für die Entscheidung des Weges: hat β einen hohen Wert, so hat die Pheromon-Konzentration hohen Einfluß auf die Entscheidung, hat β einen kleinen Wert, so ist die Konzentration weit weniger entscheidend. In der Abbildung sind für $\beta = 2, \beta = 4$ und $\beta = 8$ die Wahrscheinlichkeiten abhängig von den erfassten Pheromon-Werten dargestellt. Die Auswirkungen der unterschiedlichen Werte für die Osmotropotaxische Sensitivität ist in der folgenden Grafik dargestellt:



Die Kapazitätsschwelle des Sensors wird durch $\frac{1}{\delta}$ angegeben. Bei hohen Pheromon-Konzentrationen tritt eine Sättigung auf und es kann nur noch eingeschränkt zwischen verschiedenen Konzentrationen unterschieden werden. [CHI95]



3.1.2 Drehungen und Richtungswechsel

Drehungen in einem kleinen Winkel sind wesentlich wahrscheinlicher als Drehungen um große Winkel. Am unwahrscheinlichsten ist die Drehung in die entgegengesetzte Richtung um 180° und allgemein gilt, daß es bei den Ameisen eine Tendenz zum geradeaus laufen gibt. Somit wird ein Gewichtungsfaktor für die Richtungsänderung eingeführt. $w(\Delta\theta)$ ist eine symmetrische Funktion um den Ausgangspunkt, und gibt zu jedem relativen Winkel den Gewichtungsfaktor an. Bei einer diskreten Gitterstruktur mit 8 Nachbarfeldern sind dies für $\Delta\theta$ fünf unterschiedliche Werte. Somit kommen für $w(\Delta\theta)$ die nachfolgenden Gewichtungen zustande:

In Feldern: $w(\Delta\theta)$

| | | |
|------|---|-----|
| 1/20 | 1/12 | 1/4 |
| 1/12 |  | 1/2 |
| 1/4 | 1/2 | 1 |

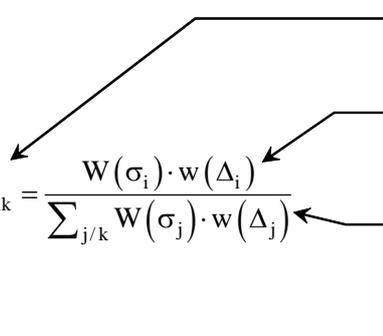
kommt von Nord-West

| | | |
|------|---|------|
| 1/2 | 1 | 1/2 |
| 1/4 |  | 1/4 |
| 1/12 | 1/20 | 1/12 |

kommt von Süden

Aus den bisher zusammengestellten Funktionen lässt sich nun eine Übergangswahrscheinlichkeits-Funktion für den Wechsel von einem Feld k zu i definieren. Es gehen hier die Formel für die Pheromon-gewichtung und für die relative Richtungsänderung mit ein, und es wird über alle benachbarten Felder normiert.

$$P_{ik} = \frac{W(\sigma_i) \cdot w(\Delta_i)}{\sum_{j/k} W(\sigma_j) \cdot w(\Delta_j)}$$



Gewichtung/Wahrscheinlichkeit
für den Übergang von Feld k zu i

Die Gewichtung des Feldes i :
Pheromon-Gewichtung
multipliziert mit dem Faktor für
rel. Drehung

Normierung:
 $\sum_{j/k}$ summiert alle Gewichtungen
der Felder j die an k angrenzen (also
insgesamt 8 Stück)

3.1.3 Lernprozess und Erinnerung

Um nun den Lernprozess der Kolonie simulieren zu können müssen noch zwei wesentliche Faktoren angegeben werden. Zum einen die Pheromon-Abgabe-Rate η die bestimmt, welche Konzentration zu jedem Zeitschritt abgegeben wird. Es wird vorgegeben, dass keine Diffusion um die Spurlinie herum stattfindet, und auch dass es keine sequentielle Abgabe von Pheromonen gibt, sondern nur die konstante Abgabe der Pheromone bei jedem Bewegungsschritt (dies ändert sich in einem später betrachteten Modell).

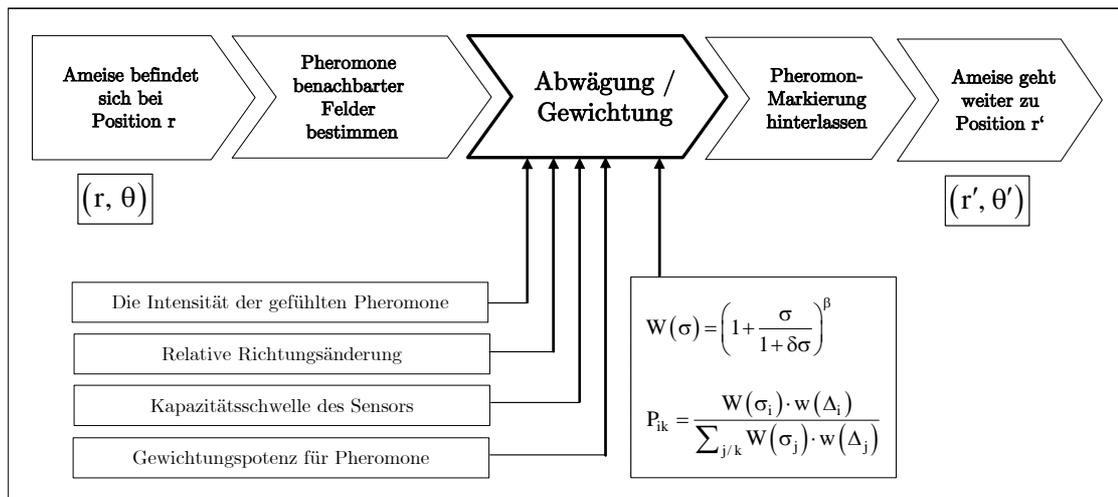
Der zweite Faktor ist die Verdampfungs-Rate κ , mit der das 'Vergessen' des Systems simuliert werden kann. Somit müssen Pheromon-Markierungen regelmäßig erneuert werden damit die Konzentrationen nicht innerhalb kurzer Zeit unter die Wahrnehmungsschwelle sinken.

Die Ameisen der Simulation haben keine individuellen Erinnerungen und es findet auch keine direkte

Kommunikation zwischen ihnen statt. Das Wissen und die Wahrnehmung der Ameisen beschränkt sich auf benachbarte Felder und deren Attribute. Somit wird in dem gesamten System jeweils zu jeder Ameise eine gewisse Anzahl an Schritten in die Vergangenheit gespeichert und das Wissen der Wege und Pfade befindet sich in dem Kollektiv der Ameisen. Das System vergisst (wegen Verdampfungs-Rate κ) etwa nach den Zeitschritten $\tau = \frac{1}{\kappa}$ alle Informationen über Wege wenn sie nicht in der Zeit erneuert werden.

3.1.4 Zusammenfassung

In dieser Grafik sind die in den vorherigen Abschnitten erläuterten Formeln nocheinmal zusammenfassend dargestellt. Diese bilden das System der Protoschwärme nach Chialvo und Millonas [CHI95].



Schritte zur Bestimmung der nächsten Position jedes Agenten.

3.2 Simulation

Für die Simulation werden noch einige Festlegungen für die Umgebung und die zu wählenden Parameter getroffen:

- Raum**
 Eine diskrete Gitterstruktur mit einer Gesamtgröße von 32x32 Feldern. Auch andere Strukturen wären hier denkbar (z.B. Wabenstruktur), sind aber komplexer in den Berechnungen.
- Zeit**
 Eine Ameise kann einen Aktions-Zyklus zu jedem Zeittakt ausführen. Dieser Zyklus besteht aus: Pheromone aller benachbarten felder Messen, Entscheidungen, Pheromonabgabe und Bewegung.
- Ränder des Gitters**
 Die Ränder stellen Schleifen dar und somit bildet sich ein toroider Körper. Wenn eine Ameise an einen Rand des Gitters kommt so gelang sie zum gegenüberliegenden Rand. Somit werden Sammlungen der Ameisen an den Rändern des Gitters vermieden.
- Plazierung und Anzahl der Ameisen**
 Die Plazierung der Ameisen erfolgt nach einem stochastischen Muster. Eine sinnvolle Besetzungsanzahl kann mathematisch berechnet werden oder durch Experimente bestimmt werden [CHI95].

- **Pheromone**

Das Pheromon-Feld des Gitters wird mit dem Wert 0 initialisiert. Hier wären eventuell auch vordefinierte Strukturen denkbar, um beispielsweise Reaktionen auf vorgegebene Situationen zu studieren. Dies findet in einigen späteren Experimenten von Chialvo und Millonas Verwendung.

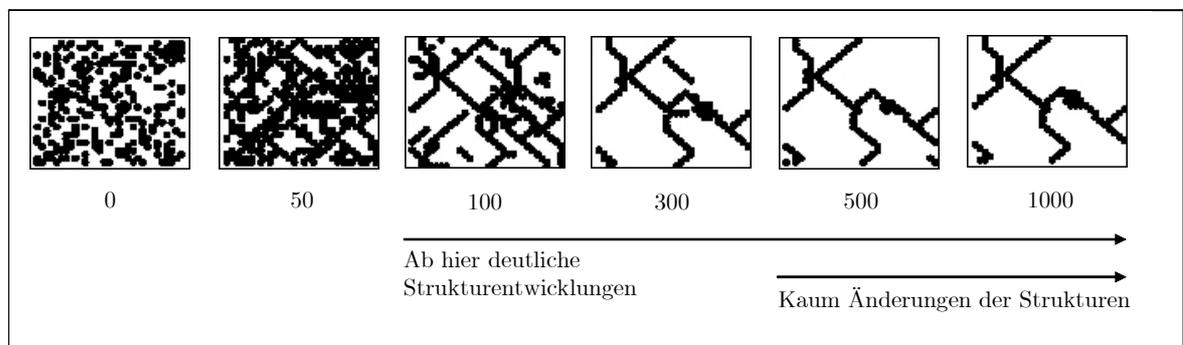
- **Dauer**

Die Simulation wird insgesamt 10^6 Zeitschritte umfassen (ideale Dauer der Simulation wurde durch Experimente bestimmt).

Weiterhin wurden die folgenden Parameter (mathematisch und experimentell) bestimmt:

- Anzahl der Ameisen: 307
- Pheromon-Abgabe η : 0,07
- Verdampfungsrate κ : 0,015
- Osmotropotaxische Sensitivität β : 3,5
- Kapazitätsschwelle des Sensors δ : 0,2

Die nun durchgeführte Simulation mit allen angegebenen Parametern führte zu der folgenden Entwicklung der Pheromon-Pfade:



Simulationsverlauf der Experimente nach Chialvo und Millonas. [CHI95].

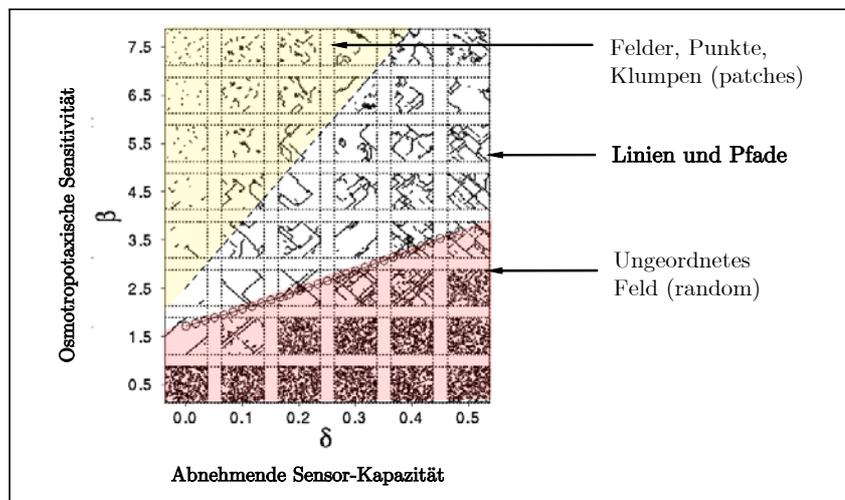
3.3 Auswertungen und Ergebnisse

Nach der Studie vieler verschiedener Parameter-Konfigurationen zeichneten sich drei Klassen von möglichen End-Zuständen ab:

- Ungeordnetes Feld
- Linien und Pfade (diese Entwicklung ist unter den im vorherigen Kapitel angegebenen Parametern zu beobachten)
- Felder, Punkte, Klumpen (patches)

Dabei bildete sich immer eine der angegebenen Konfiguration und blieb nach einem Zeitpunkt t_s bestehen bis zum Ende der Simulation. Die Entwicklung der Linien und Pfade folgte dabei immer dem Muster, dass zuerst bis etwa zum Zeitabschnitt $t=500$ eine Kondensationsphase stattfand (in dem Netzwerk bildeten sich die Pfade) und anschließend eine Vereinfachungsphase (viele der Wege lösen sich wieder auf) [RAU95].

Welche der drei Formen sich während der Simulation bildet hängt von der Wahl der beiden Parameter β und δ ab (und zu geringerem Einfluß auch von den weiteren Umgebungs-Variablen). Werden die Parameter β und δ innerhalb bestimmter Definitionsbereiche modifiziert, kann man anhand der Ergebnisse folgende Aufstellung anlegen (physiological Phase-Plot") [RAU95]:



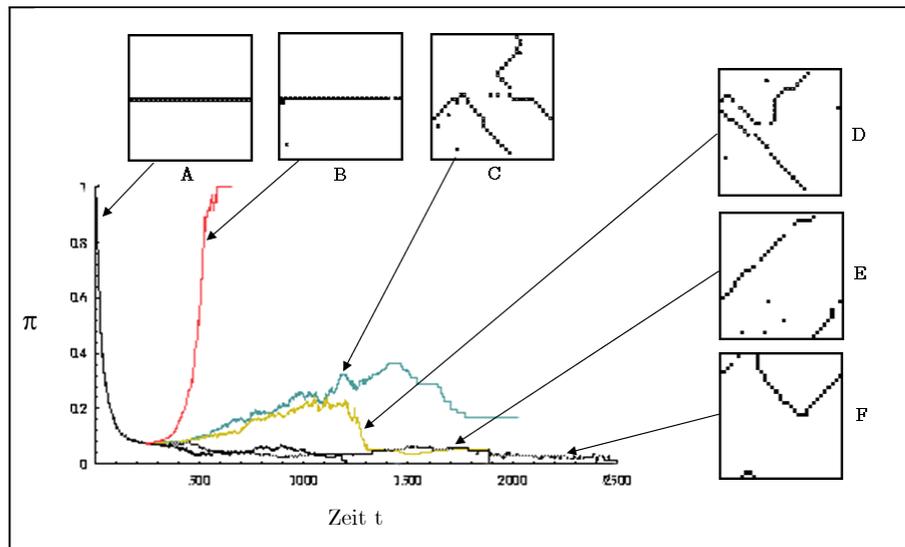
'Physiological Phase Plot'. Nach 1000 Zeitschritten bilden sich die drei Arten von End-Zuständen: Ungeordnetes Feld, Linien/Pfade und Punkte/Klumpen. [CHI95]

Die Linie aus Kreisen stellt dabei die 'Phase-Transition-Line' dar; diese ist der Übergang von dem ungeordneten Pheromon-Feld hin zur Bildung der Wege. Um nun die drei Bereiche innerhalb dieses Parameter-Raumes und auch die 'Phase-Transition-Line' weiter zu untersuchen führten Chialvo, Millonas und Rauch die beiden folgenden Experimente durch.

3.3.1 Memory-Effect

Zuerst werden die Werte von β und δ so gewählt, dass sich ein Netzwerk von Pfaden und Wegen bildet. Nach einiger Zeit der Durchführung der Simulation mit diesen Parametern wird der Wert der osmotropotaxischen Sensitivität verringert, und zwar so weit, dass die Ameisen die gebildeten Pfade verlassen und sich nun zufällig in dem Gitter bewegen (β ist nun unterhalb der Phase-Transition-Line", $\beta < \beta_c$). Die ursprünglichen Pheromon-Spuren verdampfen nun mit der Rate κ . Die gebildeten Pheromon-Strukturen lösen sich auf und es ergibt sich eine zunehmend stochastische Verteilung der Ameisen auf dem Feld (entsprechend auch der Pheromon-Pfade).

Nach verschiedenen Zeitpunkten wurde nun die osmotropotaxische Sensitivität β wieder auf den ursprünglichen Wert zurückgesetzt. Interessant ist dabei vor allem die Frage, wie viele der ursprünglichen Wege des Netzwerkes (vor der Absenkung von β) sich nun wieder bildeten.



Memory-Effect. [CHI95] [RAU95]

In der Grafik zu erkennen sind die verschiedenen Zeitpunkte der Zurücksetzung von β . Auf der Y-Achse wird der Wert π im Intervall $[0,1]$ angegeben der angibt, wie viele Felder des aktuellen Pheromon-Feldes mit dem ursprünglichen Muster übereinstimmen ($\pi = 1$ ist dabei die absolute Übereinstimmung). Bei dem Graphenverlauf des Versuches B kann man erkennen, dass sich das ursprüngliche Netzwerk wieder vollständig gebildet hat. Beim Zeitpunkt der Rücksetzung von β waren bei B bereits 90% des Netzwerkes zerstört.

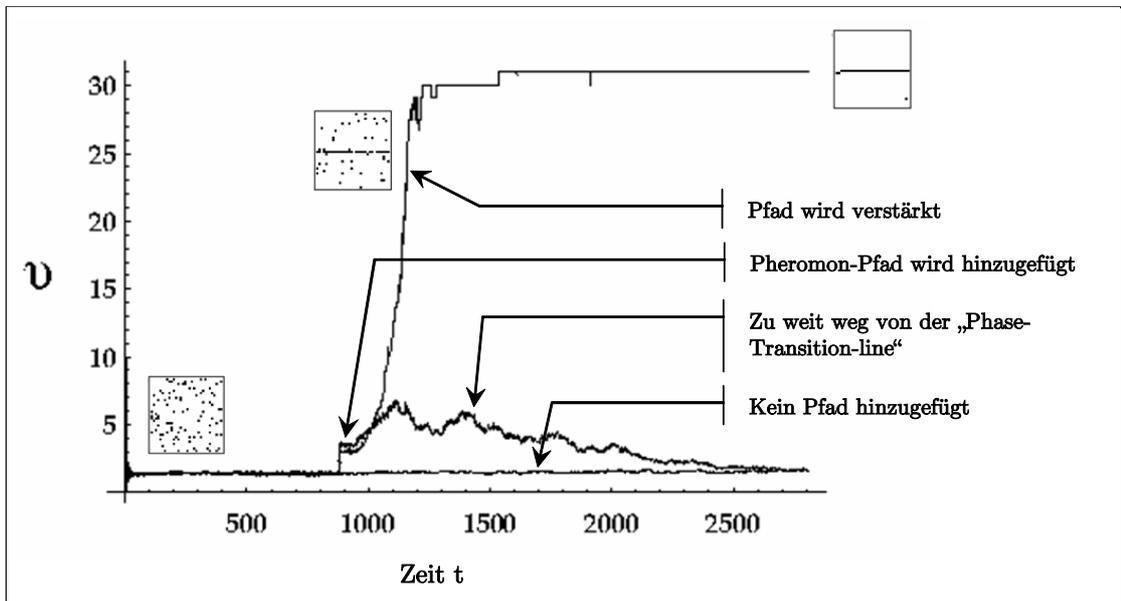
Bei allen späteren Zurücksetzungs-Zeitpunkten findet nur noch eine teilweise (bis zu fast gar keiner) Rekonstruktion des Netzwerkes statt. Die Pheromon-Spuren waren zu diesen Zeitpunkten schon zu sehr verschwunden aufgrund der Verdampfung.

3.3.2 Bootstrapping

Bei der folgenden zweiten Experiment-Serie wurde das sogenannte Bootstrapping-Verhalten¹ untersucht. Dabei wird eine Simulation angelegt, in der die Parameter-Werte von β und δ kurz unterhalb der Phase-Transition-Line liegen. In der gestarteten Simulation bildet sich nun erwartungsgemäß kein Muster sondern es bleibt ein ungeordnetes Feld.

Während der Simulation wird für einen kurzen Zeitraum ein horizontaler Pheromon-Pfad ergänzt. Für diesen Pfad werden die Pheromon-Konzentrationen entlang dieses Pfades nur gering erhöht. Interessanterweise kommt es nun bei einigen Fällen der Experimente zu einer Verstärkung dieses Pfades bis hin zur Bildung des einzigen Weges der Ameisen entlang dieses Pfades. Wenn sich allerdings die Parameterwerte zu weit entfernt von der Phase-Transition-Line befinden ändert sich an der Netzwerk-Bildung der Simulation nichts.

¹im übertragenen Sinn: biolog., Entfaltung der Vielfalt des Genoms in den einzelnen Zellen.



Bootstrapping-Effect [CHI95] [RAU95]

Kapitel 4

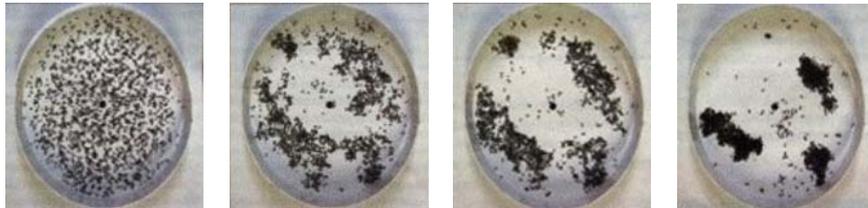
Andere Modelle und Simulationen

In diesem Kapitel betrachten wir einige weitere Ansätze und Algorithmen zu dem Verhalten von Schwärmen und dessen Simulation. Dies sind vor allem Theorien von Deneubourg, Lumer, Faieta, Bonabeau, Dorigo und Ramos.

Miteinander gekoppelt werden diese Verfahren schließlich in dem ACLUSTER-Algorithmus von V. Ramos, der in Abschnitt 4.5 und 4.6 näher erläutert wird.

4.1 Picking and Dropping

Inspiziert wurden diese Ansätze des Aufhebens und Ablegens (Picking and Dropping) von Objekten durch die Untersuchungen von L. Chrétien¹, der in Experimenten das Anordnen toter Ameisen durch ihre Kolonie untersuchte. Bei diesen Anordnungen bilden sich Cluster in denen die Körper gesammelt werden.



Eine Sequenz der Anordnungen von toten Ameisen durch eine Ameisen-Kolonie. (Durchmesser 25 cm, Zeitpunkte der Aufnahmen: 2, 6, 26, 40 Stunden) Quelle: Chrétien, L., 1996

Die Umsetzung in einem Simulations-Modell entwickelte Deneubourg 1991 [DEN91]²; er bezeichnet die Bildung der Cluster als einen Spezialfall des Sortierens. Die Agenten seines Systems sollten nun die Möglichkeit erhalten, Objekte bewegen zu können und es sollte ein Regelwerk vorgeben werden, mit dessen Hilfe sich ebenfalls Cluster bilden. Vorerst beschränkt sich Deneubourg auf nur eine Sorte von Objekten.

Objekte sind ebenfalls auf dem begrenzten Feld (Gitter) plaziert, auf denen sich auch die Ameisen bewegen. Kommt eine Ameise zu einem Feld auf dem sich ein Obejekt befindet, muss sie die Entschei-

¹Chrétien, L.: Organisation Spatiale du Matériel Provenant de l'excavation du nid chez Messor Barbarus et des Cadavres d'ouvrières chez Lasius niger (Hymenopterae: Formicidae), Ph.D. dissertation, Université Libre de Bruxelles, 1996

²Die entwickelten Ansätze von Deneubourg wurden in realen Roboter-Simulationen verwendet (Hardware).

Roboter- und Computer-Simulation arbeiten häufig an den identischen Problemstellungen, da sich ja die Agenten der Simulationen auch ähnlich wie Roboter verhalten sollen (und umgekehrt).

dung über das Auheben des Objektes treffen (also auch feststellen, ob sich das Objekt nicht schon bereits in einem Cluster befindet). Analog gilt dies für das Ablegen. Für die Wahrscheinlichkeiten des Aufhebens und Ablegens wurden die folgenden beiden Formeln entwickelt:

Aufheben von Objekten:

$$P_p = \left(\frac{k_1}{k_1 + f} \right)^2$$

Wahrscheinlichkeit dafür, dass ein unbeladener Agent ein Objekt aufnimmt
Reaktions-Schwelle (Konstante)
f = Objekte, die in direkter Nachbarschaft vorkommen

Wenn dabei $f \ll k_1 \implies P_p \approx 1$, d.h. die Wahrscheinlichkeit ist sehr hoch dass ein Objekt aufgehoben wird, wenn nicht viele Objekte in der Nachbarschaft sind. Wenn $f \gg k_1 \implies P_p \approx 0$, d.h. dass Objekte nicht von bereits bestehenden Clustern entfernt werden.

Ablegen von Objekten

$$P_d = \left(\frac{f}{k_2 + f} \right)^2$$

Wahrscheinlichkeit dafür, dass ein beladener Agent sein Objekt abwirft
Reaktions-Schwelle k_2
Objekte, die in direkter Nachbarschaft vorkommen

Wenn dabei $f \ll k_2 \implies P_d \approx 0$, d.h. es soll nichts abgelegt werden, wenn sich nur wenige Objekte in der Nachbarschaft befinden. Wenn $f \gg k_2 \implies P_d \approx 1$, d.h. dass bei vielen Objekten in der Umgebung das getragene Objekt abgeworfen werden kann.

Mit diesen beiden Formeln lassen sich bereits Simulationen durchführen, jedoch bezeichnet Ramos das Resultat als nicht zufriedenstellend [ABR03]. Es sind über 5 Millionen Iterationen notwendig um annehmbare Ergebnisse zu erzielen (bei 100x100 Gitter mit 400 Objekten und 10 Agenten). Weiterer Kiritpunkt ist, daß es nicht möglich sei Objekte zu unterscheiden³.

³Zum Beispiel sortiert die Leptothorax Ameise ihre Larven in konzentrischen Kreisen, in dem die jüngsten der Brut ganz innen liegen und die älteren in den äußeren Kreisen. Solch eine Sortierung kann man mit den aktuellen Formeln nicht vornehmen.

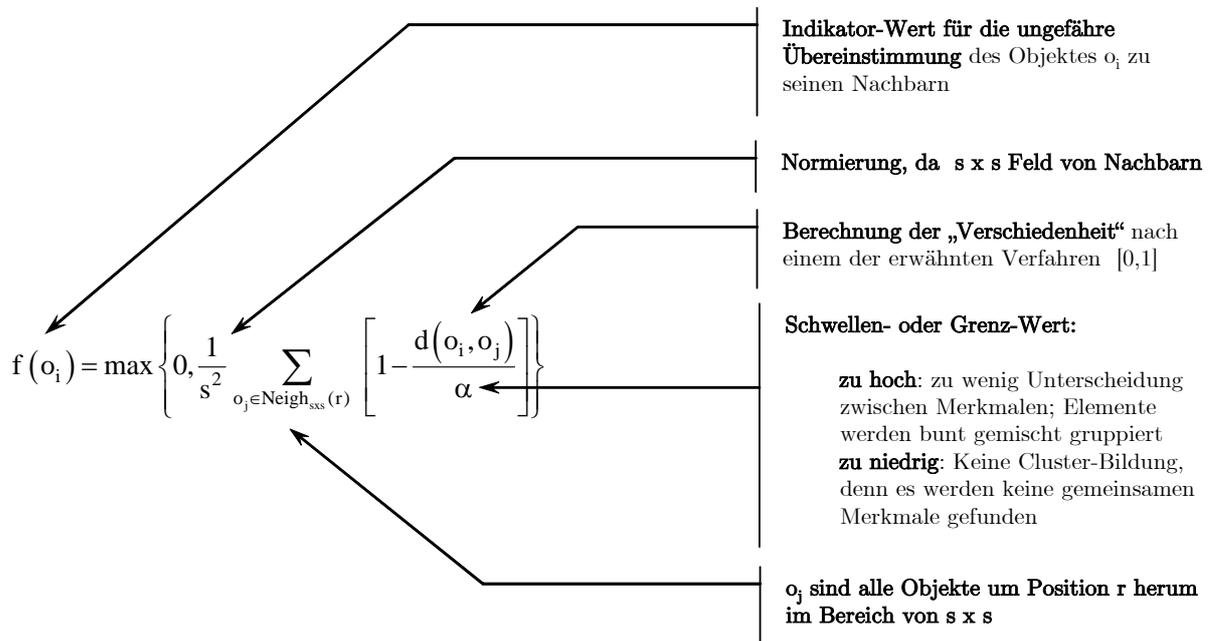
4.2 Distanz-Bestimmungen im Parameter-Raum

Um die Nutzung einer solchen Simulation für die Datenanalyse zu ermöglichen führen Lumer und Faieta [LUM94] eine Funktion für die Distanzbestimmung bzw. der Unähnlichkeit zwischen zwei Objekten ein. Der Wertebereich dieser Funktion $d()$ liegt im Intervall $[0, 1]$ und bedeutet bei der Unterscheidung von zwei Objekten o_i und o_j :

$$\begin{aligned} d(o_i, o_j) = 1 &\implies \text{Objekte sind gleich.} \\ d(o_i, o_j) = 0 &\implies \text{Objekte sind verschieden.} \end{aligned}$$

Um nun mehr als zwei Fälle unterscheiden zu können bekommen die Objekte zur Unterscheidung mehrere Attribute bzw. Eigenschaften. Diese befinden sich plziert in dem Raum der Eigenschaften (Attribute-Space) und werden als Vektoren der Größe n angegeben, die Werte vom Typ Real enthalten. Somit befinden sich alle Objekte exakt plziert im \mathbb{R}^n .

Um die Distanz zwischen zwei Objekten zu bestimmen müssen nun Berechnungen auf den Attributen der Objekte mit Hilfe von Verfahren wie der Euklidischen Norm, Infinite Norm oder der 'Mahalanobis metric' durchgeführt werden [RAM02]. Schließlich soll es möglich sein, daß die Attribute des \mathbb{R}^n auf einen niedriger dimensionierten Raum abgebildet werden (Mapping, Abbildung): $\mathbb{R}^n \rightarrow \mathbb{R}^m, m < n$. Dabei sollen möglichst viele der Nachbarschaftsbeziehungen des höher dimensionalen Raumes auch in dem niedriger dimensionierten Raum übertragen werden; allerdings kann es natürlich auch möglich sein, daß entfernte Punkte des \mathbb{R}^n auf identische Punkte im \mathbb{R}^m abgebildet werden. Eine solche Übereinstimmungs-Abbildung kann man durch die folgende Formel berechnen [LUM94]:



Mit Hilfe dieser Übereinstimmungs-Formel lassen sich auch die Formeln für das Aufheben und Ablegen von Objekten neu angeben:

$$\begin{aligned} P_p(o_i) &= \left(\frac{k_1}{k_1 + f(o_i)} \right)^2 \\ P_d(o_i) &= 2 * f(o_i) \text{ wenn } f(o_i) < k_2 \\ P_d(o_i) &= 1 \text{ wenn } f(o_i) \geq k_2 \end{aligned}$$

4.3 Erweiterungen

Folgende Erweiterungen wurden in dem Modell von Lumer und Faieta [LUM94] noch angewendet.

1. **Variable Geschwindigkeiten**

Die Geschwindigkeit der Agenten variiert im Intervall $[1, v_{\max}]$. v_{\max} ist die maximale Anzahl von Feldern die ein Agent während einem Zeitschritt weiter gehen kann. Die variablen Geschwindigkeiten führen dazu, dass schnellere Agenten für die Cluster-Bildung in einer höheren Skalierung sorgen und die langsameren Agenten die Arbeit im Detail durchführen.

2. **Kurzzeitgedächtnis**

Die Agenten können sich die letzten m Objekte die abgelegt wurden mit ihrer Position merken. Bei neuen gefundenen Objekten kann der Agent überprüfen, ob er von einem ähnlichen Objekt bereits eine Position kennt. Diese Erweiterung um ein Gedächtnis reduziert die Bildung von Mini-Clustern.

3. **Verhaltens-Wechsel der Agenten**

Die Ameisen bekommen die Fähigkeit, Cluster wieder aufzulösen. Dies ist nur unter bestimmten Bedingungen möglich und hilft als Ausweg aus einer Sub-Optimalen Lösung in welcher der Algorithmus fest steckt.

Diese drei Erweiterungen optimieren zwar die Clustering des Algorithmus von Lumer und Faieta, sie steigern aber auch die Komplexität des Verfahrens und somit die notwendigen Berechnungen je Zeitabschnitt. Deshalb entscheidet sich V. Ramos dafür, diese Erweiterungen nicht zu implementieren und stattdessen die Theorien von Chialvo und Millonas [CHI95] zu integrieren. Später wird er noch diese unterschiedlichen Verfahren im Vergleich auf ihre Effizienz untersuchen (*siehe Abschnitt 4.5*).

4.4 Ansatz von Vitorino Ramos

Ramos greift in seinem Modell nicht nur die Ideen von Lumer & Faieta und Deneubourg auf, sondern auch das Konzept von Chialvo und Millonas mit der Aussonderung von Pheromonen. Allerdings werden die Pheromone nun nicht mehr in einer konstanten Rate η abgegeben, sondern die Rate ist abhängig von den Objekten im Nachbarschaftsbereich:

$$T = \eta + p * \Delta h.$$

Δh sind die Elemente in einem vordefinierten Umfeldbereich (häufig 3x3-Feld) und p ist eine Konstante (empirisch auf 0,0025 festgelegt) [RAM02]. Die Pheromonspuren führen zu wesentlich stabileren Konfigurationen.

$$T_{\theta}(s) = \frac{s^n}{s^n + \theta^n}$$

Ramos verwendet $n = 2$ (aber auch Werte von $n > 2$ möglich und mit ähnlichen Ergebnissen)

Aus dieser allgemeinen Formel T_{θ} für die Reaktion auf einen Stimulus gelangt Ramos zu der Reaktionsformel χ (für die Reaktion auf umliegende Objekte):

$$\chi = \frac{n^2}{n^2 + 5^2}$$

für ein Nachbarschaftsfeld von 3x3 Feldern. (n ist die Anzahl der Objekte auf diesen Feldern).

Für die Berechnung der Verschiedenheit bzw. Distanz in dem Attribute-Space der Eigenschaften definiert Ramos die Distanz d :

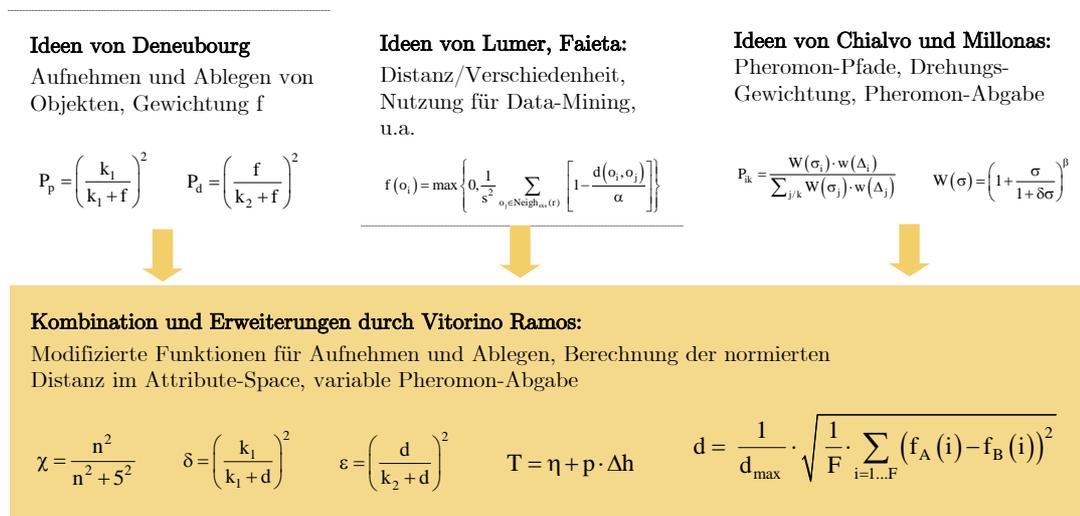
$$d = \frac{1}{d_{\max}} * \sqrt{\frac{1}{F} * \sum_{i=1..F} (f_A(i) - f_B(i))^2}$$

wobei gilt: F = Anzahl der Features (Attribute), $f_A(i)$ = das Feature i vom Objekt A, d_{\max} = Normalisierung.

Daraus ergeben sich die Formeln für das Aufheben und Ablegen von Objekten:

$$\begin{aligned} \text{Aufheben:} \quad \delta &= \left(\frac{k_1}{k_1 + d} \right)^2 \\ \text{Ablegen:} \quad \varepsilon &= \left(\frac{d}{k_2 + d} \right)^2 \end{aligned}$$

Mit Hilfe dieser Formeln evaluiert V. Ramos verschiedene Kombinationen um die Effizienz der gewählten Algorithmen zu vergleichen (*siehe Kapitel 4.5*). Hier eine Zusammenfassung des Modells von V. Ramos:

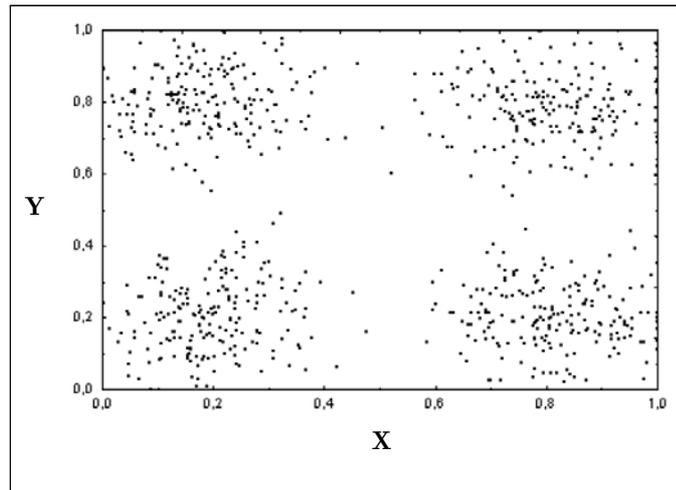


4.5 ACLUSTER-Algorithmus

Dieser von V. Ramos entwickelte Algorithmus setzt sich aus den folgenden Bestandteilen zusammen: Formeln für das Aufheben und Ablegen von Objekten, Reaktions- und Grenzwerten-Formeln, Formeln für Pheromengewichtung, Einfluß der Richtungsänderung als Gewichtungsfaktor, eine gewichtete Pheromonabgabe und ein stochastischer Einfluß auf einige der berechneten Größen.

Eine vollständige, kommentierte Fassung des Pseudocodes zum ACLUSTER-Algorithmus befindet sich im *Anhang A*.

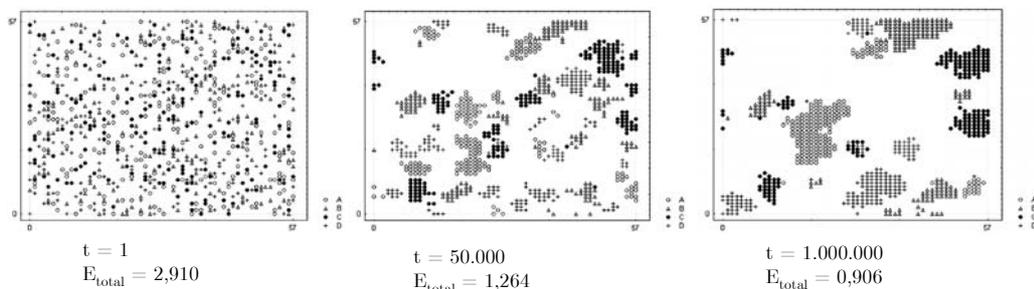
Es werden für die Simulation 800 Attribut-Paare generiert die folgendermaßen erzeugt werden: um vier Punkte im Koordinatensystem werden anhand der gauss'schen Verteilung die 800 Punkte bestimmt, die normalverteilt um dieser vier Initialisierungspunkte liegen. Die Parameter sind dann schließlich jeweils der X- und der Y-Wert. Durch diese zwei Achsen wird der Parameter-Raum aufgespannt.



Attribute-Space: Gauss'sche Verteilungen um vier Punkte [ABR03]

Die Objekte (Punkte) werden nun zufällig auf dem Gitter der Größe 57x57 verteilt. Verwendet werden für die Simulation 80 Agenten. Diese Werte ergeben sich aus empirisch bestimmten Formeln: $A = 4 * n_0$, $n_a = A/40$, $n_a/n_0 = 1/10$. (daraus ergibt sich: $A = 57 * 57 = 3249$ Felder, $n_a = 3249/40 \approx 80$ Agenten, $n_0 = 10 * n_a = 800$ Elemente).

In der folgenden Abbildung ist der Verlauf der durchgeführten Simulation zu erkennen; wobei zur leichteren Unterscheidung jeweils die zusammengehörigen Punkte der gauss'schen Verteilung ein gleiches Darstellungssymbol besitzen (für den Algorithmus nicht ersichtlich, außer natürlich indirekt durch Angaben des Parameter-Raumes).



Simulationsverlauf [ABR03]

Die von Ramos entwickelten Picking- und Dropping-Wahrscheinlichkeiten sollten nun in ihrer Effizienz mit den Formeln nach Lumer und Faieta [LUM94] verglichen werden. Die folgenden vier Kombinationen der Funktionen wurden für die Evaluierung verwendet.

| Funktions-Typ | Wahrscheinlichkeit für Aufheben | Wahrscheinlichkeit für Ablegen |
|--------------------|---|--|
| 1 Ramos | $P_p = (1-\chi) \cdot \varepsilon$ | $P_d = \chi \cdot \delta$ |
| 2 Ramos | a) $P_p = (1-\chi) \cdot \varepsilon$ b) $P_p = \varepsilon$ | a) $P_d = \chi \cdot \delta$ b) $P_d = \delta$ |
| 3 Ramos | a) $P_p = 1-\chi$ b) $P_p = \varepsilon$ | a) $P_d = \chi$ b) $P_d = \delta$ |
| 4 Lumer, Faieta | $P_p(o_i) = \left(\frac{k_1}{k_1 + f(o_i)} \right)^2$ | $P_d(o_i) = 2 \cdot f(o_i)$ wenn $f(o_i) < k_2$ $P_d(o_i) = 1$ wenn $f(o_i) \geq k_2$ |

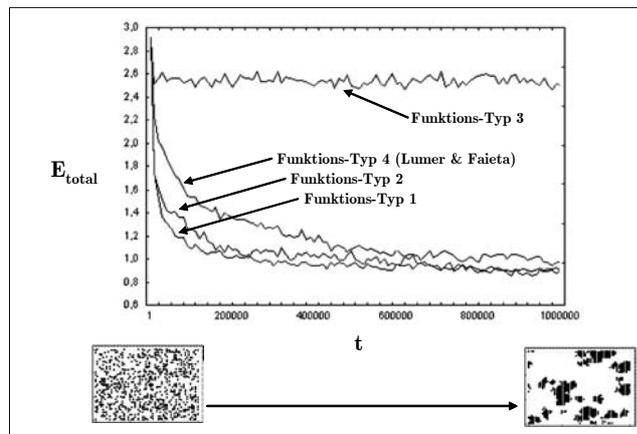
Kombinationen der Formeln nach Lumer&Faieta und V. Ramos [ABR03]

In der folgenden Grafik wird die Effizienz der Verfahren als Graph dargestellt; die X-Achse trägt die Zeit ab (von 1 bis 10^6), die Y-Achse hat die Dimension der totalen Entropie. Diese ist definiert durch $E_{total} = E_A + E_B + E_C + E_D$, wobei gilt:

$$E_\xi = \frac{\sum \sum e_i}{n * e_{max}}; \xi = A, B, C, D$$

dabei gilt: $e_i = 1$ wenn Objekt an Position e_i zum Typ ξ gehört oder das Feld unbesetzt ist, andernfalls ist $e_i = 0$. $e_{max} = 8$ (die umliegenden Felder als Region) und $n = 200$ (jeweils 200 Elemente für A,B,C,D). Das Maximum der Entropie für E_ξ ist 1, deshalb gilt: $\max(E_A + E_B + E_C + E_D) = 4$.

Der Funktions-Typ 1 nach Ramos erwies sich als das effizienteste Verfahren, die Formeln nach Lumer & Faieta und Ramos' Funktions-Typ 2 liegen etwas dahinter. Die Kombination der Formeln nach Typ 3 führte zu keiner Ordnung der Elemente.



Entropie-Verlauf der vier Funktions-Arten von V. Ramos und Lumer & Faieta [LUM94] [ABR03]

Kapitel 5

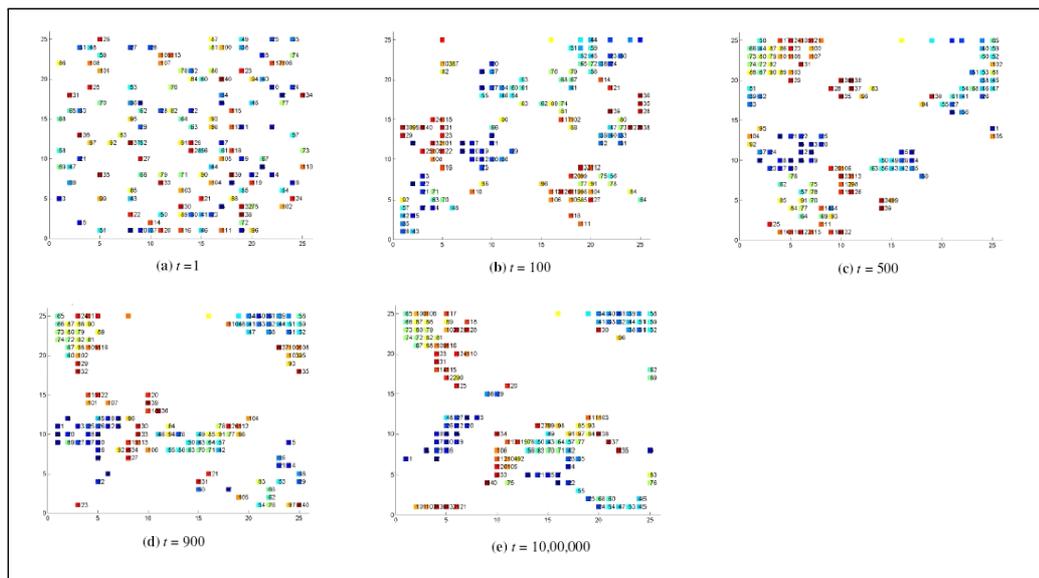
Anwendungen

5.1 Clustering von Web-Usage-Data

Der Entwickelte Algorithmus soll nun zum Sortieren und Anordnen von Daten genutzt werden. Dabei entscheidet sich Ramos für die gespeicherten Web-Nutzungs-Daten zur Website der Universität Lissabon [RAM04].

Es soll ein Clustering der Traffic-Daten anhand bestimmter gemeinsamer Eigenschaften stattfinden, um später nach der Optimierung der Datensätze sog. „Web-Usage-Patterns“ zu erhalten. Anschließend möchte Ramos einen linearen genetischen Algorithmus für die Analyse der Nutzungs-Trends auf den vorsortierten Daten ausführen. Eine detailliertere Beschreibung und Literaturangaben zu den linear genetischen Algorithmus-Ansätze ist unter [RAM04] zu finden.

Die Attributwerte die für die Distanz-Bestimmungen verwendet werden sollen sind die Daten der Log-Files. In diesen stehen angesammelte Datenmengen zur Nutzungsdauer einzelner Nutzer, deren Seitenaufrufe, Frequenzen der Nutzung u.a. Weitere Parameter der festgelegten Umgebung durch Ramos sind: 25x25 Gitter (toroid), Typ-1-Übergangsfunktion, $k_1 = 0, 1$, $k_2 = 0, 3$, Ameisen: 14.



Entwicklung der Sortierung von Web-Nutzungs-Daten. (Typ 1 Übergangsfunktion) [RAM04]

Bei Zeiteinheit 1 sind die Datensätze zufällig auf der Gitterstruktur verteilt. Bis zur letzten Zeiteinheit (wie bei den meisten Simulationen bisher: 10^6) kann man die Entwicklung der Cluster-Bildung verfolgen.

Ramos vergleicht nun die Effizienz seines ACLUSTER-Algorithmus mit drei anderen Verfahren: das erste Verfahren basiert auf sog. β elf-organizing maps mit Hilfe künstlicher neuronaler Netze, der zweite Algorithmus basiert ebenfalls auf β elf-organizing maps", nutzt aber wie ACLUSTER einen linear-genetischen Algorithmus und der dritte Algorithmus ist i-Miner (dieser verwendet Evolutions- und Fuzzy-Techniken). In dem Effizienz-Vergleich von Ramos stellt sich heraus, dass der ACLUSTER die Effizienz des linear-genetischen Verfahrens steigern kann, jedoch nicht mit der Geschwindigkeit und Präzision des i-Miner-Algorithmus mithalten kann (Vgl. [RAM04]).

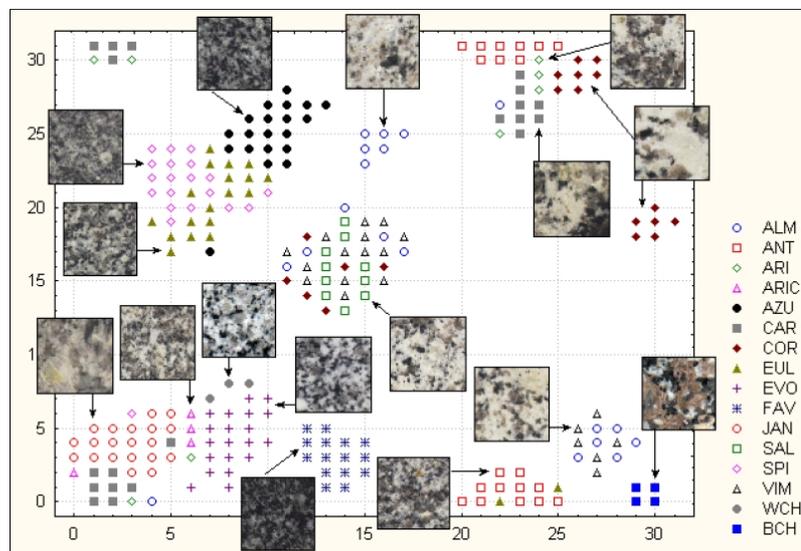
5.2 Clustering von Mustern (Continuous Data)

Der ACLUSTER-Algorithmus kann auch auf Datenbeständen ausgeführt werden, bei denen nicht alle Datensätze seit Beginn der Simulation vorhanden sind. In der folgenden Anwendung wurden die Datensätze in sechs Gruppen in das System zur Gruppierung gegeben. [ABR03]

Es sollen verschiedene Arten von Granit-Steinen zu Clustern zusammengefasst werden. Dazu wurden von 16 Granit-Typen insgesamt 244 Bilder zusammen mit ihren jeweiligen Eigenschaften beschrieben: geologisch, physikalisch, mineralogisch, mechanisch, usw. Diese Eigenschaften wurden in 117 'Features' abgebildet.

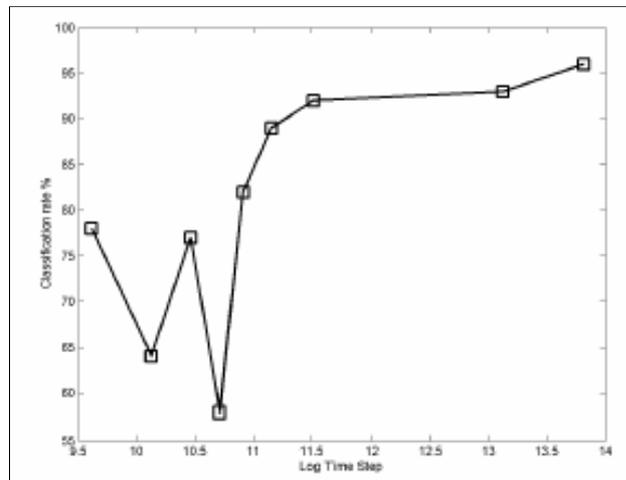
Zusätzlich wurden durch einen Algorithmus zur mathematischen Morphologie [SER82] weitere 60 'Features' berechnet. Dieser Algorithmus gibt diese 60 Unterscheidungsmerkmale anhand der Untersuchung der Bilddaten aus, indem eine Analyse der Muster, Farben und Strukturen stattfindet.

Die Simulation wurde mit den 244 Bildern und dem Parameter-Raum von 177 'Features' ausgeführt, Feldgröße: 32×32 , Ameisen: 24, Zeitschritte: 10^6 .



In der Grafik zu sehen sind die Anordnungen der einzelnen Graphit-Arten zu Gruppen

Um nun die kontinuierlichen Eingaben der Datensätze (6 insgesamt) zu betrachten kann man in dem folgenden Diagramm den Einsatz der letzten beiden Daten-Einheiten im Diagramm sehen: Einheit der X-Achse sind die Zeitschritte (logarithmisch), die der Y-Achse ist die Klassifikationsrate (in Prozent).



Zeitverlauf der Klassifikationsrate

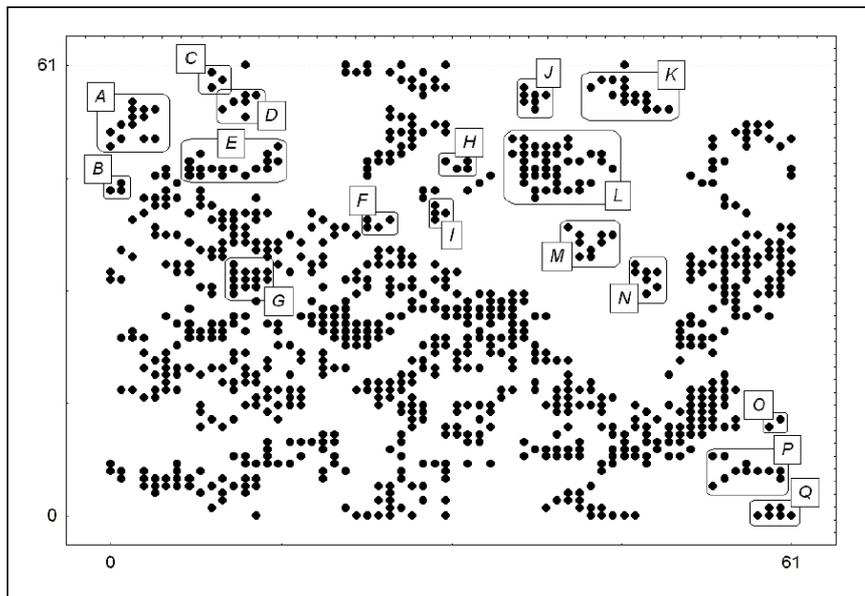
Die Klassifikationsrate wird dabei nach dem k-NNR-Prinzip (nearest neighbour rule") berechnet: In diesem Verfahren werden prozentual alle Elemente als gruppiert angesehen, bei denen sich k Nachbarn mit zugehörigen Parametern auf den angrenzenden acht Feldern befinden (für k wird in dieser Grafik der Wert 3 verwendet).

Der Einsatz des Algorithmus auf kontinuierlichen Datensätzen konnte bei dieser und weiterer Simulationen sehr erfolgreich eingesetzt werden. Diese Anwendungen sind sehr gute Beispiele für die Flexibilität des Einsatzes von Schwarm-Intelligenz-Systemen.

5.3 Clustering von Texten und Inhalten

Ein weiteres Anwendungs-Experiment von Ramos ist der Versuch, durch den Acluster-Algorithmus eine sinnvolle Gruppierung relevanter Wörter eines spanischen Zeitungsartikels zu erreichen. Hierbei muss wiederum ein externer Algorithmus für die Berechnung der Distanz-Werte im Parameter-Raum verwendet werden. Es kommt ein LSA-Algorithmus (latent semantic analysis) zum Einsatz, der Berechnungen anhand der Wörter durchführen kann, die in einem multidimensionalen Raum abgebildet sind. Diese Abbildungen sind sehr komplex und erfolgen auf der Basis von Kriterien zur sinnvollen und kontext-abhängigen Zuweisung der Wörter zu bestimmten Parameter-Werten. Schliesslich werden noch Nutzer-Anforderungen (seine Präferenzen oder Ziele), die Dokumente und die Anfragen als Vektoren implementiert, so dass der LSA-Algorithmus nun auf diesen Vektoren Operationen, Verknüpfungen und Beziehungen ausführen und bestimmen kann. [QUE01]

Nach den durchgeführten Simulationen lassen sich zwar einige Wortgruppierungen erkennen die in einem Kontext zueinander stehen, aber Ramos selbst bezeichnet die Simulation als "weit entfernt von einem optimalen Resultat"[RAM02]



Anordnung der 931 Wörter nach der Anwendung des Acluster-Algorithmus. Gekennzeichnet sind einige der zusammengehörigen Wortgruppen.

Kapitel 6

Zusammenfassung und Ausblick

Die im Rahmen des VR-Seminars erarbeiteten Theorien zur Schwarm-Intelligenz gewährten mir einen faszinierenden und herausfordernden Einblick in neue Algorithmen-Techniken zur Lösung verschiedenster Probleme. Es war sehr interessant zu verfolgen, wie nach den Anfängen des Modells der Proto-Schwärme schrittweise eine Erweiterung und Optimierung dieses Modells durch zahlreiche Wissenschaftler erfolgte. Und nach den im letzten Kapitel betrachteten Anwendungen des ACLUSTER-Algorithmus wird besonders deutlich, dass sich hier zwar ein flexibles und leistungsfähiges Werkzeug entwickelt hat, seine Weiterentwicklung und Optimierung aber bei weitem noch nicht beendet ist.

Denn aus den in dieser Ausarbeitung betrachteten Ansätzen, Simulationen und Anwendungen ergeben sich aus meiner Sicht einige interessante zukünftige Forschungsfelder. So wird sicher an einer weiteren Optimierung des ‚Clusterings‘ gearbeitet werden müssen um auch ein Data-Mining mit großen Daten-Mengen, und vor allem in großen Attribut-Räumen zu ermöglichen.

Auch sollten viele der bestehenden Algorithmen und Verfahren weitgehender evaluiert werden und in Benchmark-ähnlichen Tests zu alternativen Algorithmen verglichen werden. Hierfür gilt es, eine homogene Test-Umgebung zu erstellen, und möglichst viele der Algorithmen modular in einem Framework oder Toolkit zur Verfügung zu stellen (ähnlich dem Swarm.Org-Toolkit).

Schließlich bedeutet auch die Verknüpfungen zwischen den Swarm-Algorithmen und weiteren mathematischen Verfahren, wie beispielsweise bei der Distanzbestimmung im Parameter-Raum o.ä., noch eine große Herausforderung.

Anhang A

Acluster-Algorithmus

| | |
|--|---|
| <pre>For every item o_i do Place o_i randomly on grid End For For all agents do Place agent at randomly selected site End For</pre> | <p>Initialisierung: Alle Elemente o_j und die Ameisen = Agenten auf dem Gitter zufällig positionieren</p> |
|--|---|

| | |
|--|---|
| <pre>For $t = 1$ to t_{max} do For all agents do $sum = 0$ Count the number of items n around site r</pre> | <p>Wiederholung / Schleife: t_{max} Zeit-Schritte (10^6), Und alle Agenten durchgehen</p> |
| <pre> If ((agent unladen) and (site r occupied by item o_i)) then For all sites around r with items present do Compute d, τ, e and P_p Draw random real R between 0 and 1 If ($R = P_p$) then $sum = sum + 1$ End If</pre> | <p>Agent unbeladen und ein Objekt auf Platz r: Für jedes Element Berechnungen durchführen, für stochastischen Entscheidungs-Anteil) In ‚sum‘ die getroffenen Entscheidungen zählen</p> |

| | |
|---|--|
| <pre> End For If (($sum = n/2$) or ($n = 0$)) then Pick up item o_i End If </pre> | <p>Wenn sum größer als die Hälfte der Anzahl der umgebenden Objekte ist oder gar kein Objekt mehr in der Nachbarschaft ist: Objekt aufheben</p> |
| <pre> Else If ((agent carrying item o_i) and (site r empty)) then For all sites around r with items present do Compute d_i, d and P_d Draw random real number R between 0 and 1 If ($R = P_d$) then $sum = sum + 1$ End If End For If ($sum = n/2$) then Drop item o_i End If End If </pre> | <p>Agent beladen und KEIN Objekt auf Platz r: Für jedes Element Berechnungen durchführen, Weiterhin analoges Vorgehen wie beim Aufheben des Objektes</p> |
| <pre> Compute $W(s)$ and P_{ik} Move to a selected neighboring site that is not occupied by other agent </pre> | <p>Berechnungen zur Bewegung der Agenten (nach Chialvo und Milonas)</p> |
| <pre> Count the number of items n around that new site r Increase pheromone at site r according to n, that is: $P_r = P_r + [? + (n/a)]$ End For Evaporate pheromone by K, at all grid sites </pre> | <p>Berechnung der Pheromon-Abgabe Pheromone verdampfen</p> |
| <pre> End For Print location of items </pre> | |

$k1 = 0.1, k2 = 0.3, K = 0.015, ? = 0.07, a = 400,$
 $\beta = 3.5, \gamma = 0.2, t_{max} = 10^6$ steps.

← **Verwendete Parameter**

Literaturverzeichnis

- [ABR03] Ramos, Vitorino; Abraham, Ajith: Swarms On Continous Data
Technical University of Lisbon, Portugal, 2003, <http://alfa.ist.utl.pt/~cvrm/staff/vramos/>
- [CHI95] Chialvo, Dante; Millonas, Mark: How Swarms build Cognitive Maps
The Santa Fe Institute, Neural Systems, Memory and Aging, University of Arizona In Luc Steels (Ed.): The Biology and Technology of Intelligent Autonomous Agents, pp. 439-450, NATO ASI Series, 1995
- [DEE90] Deerwester, S.; et. Al.: Indexing by Latent Semantic Analysis,
Journal of the American Soceity for Information Science, 41, pp. 391-407, 1990
- [DEN91] Deneubourg, J-L, Gross, S., et. Al.: The Dynamic of Collective Sorting Robot-like Ants and Ant-Like Robots,
SAB ,90, 1st Conference On Simulation of Adaptive Behaviour: From Animals to Animats. Cambridge, MA: MIT Press, 1991
- [DOR04] Dorigo, Marco: „Ant Colony Optimization“-Project,
Université Libre des Bruxelles <http://iridia.ulb.ac.be/~mdorigo/ACO/index.html>, Stand Januar 2004
- [FRE01] Freeman, Walter J.: How Brains Make Up Their Minds
Columbia University Press, 2001
- [LUM94] Lumer, E.D.; Faieta, B.: Diversity and Adaption in Populations of Clustering Ants
in Cliff, D., et. Al: From Animals to Animats 3, Proc. Of the 3rd Int. Conference on the Simulations of Adaptive Behaviour Cambridge, MIT Press / Bradford Books, 1994
- [MIL94] Millonas, M. M.: Pattern Formation in Physical and Bilogical Systems (ed. P. Claudis).
Santa Fe Institute in the Sciences of Complexity, Reading, Massachussetts: Addison-Wesley, 1994
- [QUE01] Quesada, J.F., et.Al.: Personalization as Disambiguation: LSAmercury, a information filtering Enginge based on latent semantic analysis, Preprint in Elsevier Science, 11 May 2001
- [RAM00] Ramos, Vitorino; Almedia, Fillipe: Artificial Ant Colonies in Digital Image Habitats A Mass Behaviour Effect Study on Pattern Recognition (2000)
<http://alfa.ist.utl.pt/~cvrm/staff/vramos/> 2nd Int. Workshop on Ant Algorithms, pp. 113-116, Brussels, Belgium, 2000
- [RAM02] Ramos, Vitorino; Merelo, Juan: Self-Organized Stigmertic Document Maps: Environment as a Mechanism for Context Learning, CVRM-IST, Paris, 2002

- [RAM04] Abraham, Ajith; Ramos, Vittorio: Web Usage Mining Using Artificial Ant Colony Clustering and Linear Genetic Programming
Department of Computer Science, Oklahoma State University; 2004, CVRM, Technical University of Lisbon
- [RAU95] Rauch, Eric: How Swarms build Cognitive Maps;
Datenmaterial zum veröffentlichten Paper und weitere Anmerkungen,
Weblink: <http://cnls.lanl.gov/~rauch/swarmweb>, Stand der Website vom 15.01.2004
- [RMP99] Ramos, Vitorino; Pina., P.; Muge., F.: From Feature Extraction to Classification: A Multidisciplinary Approach applied to Portuguese Granites, 11th Scan. Conf. On Image Analysis, pp. 817-824, Greenland, 1999
- [RMP02] Ramos, Vitorino; Pina., P.; Muge., F.: Self-Organized Data and Image Retrieval as a Consequence of Inter-Dynamic Relationships in Artificial Ant Colonies,
CVRM-Geo Systems Centre, Tech. Univ. of Lisbon (IST), 2002
- [SER82] Serra, J.: Image Analysis and Mathematical Morphology, Academic Press, London, 1982
- [WIL90] Wilson, Edward O; Hölldobler, Bert: The Ants,
Springer Verlag, 1998 (Cambridge, Belknap 1990)