

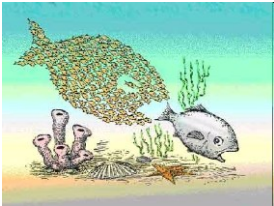
Schwarm- intelligenz AS1-8

Schwarmintelligenz

SciFi-Literatur:

1930: O. Stapledon - Marsinsekten

1964: S. Lem - "Der Unbesiegbare"



Inhalte

1. Einleitung

Was ist Schwarmintelligenz?

Warum ist Schwarmintelligenz interessant?

2. Ameisenalgorithmen

Natürliche Ameisen

Künstliche Ameisen

Anwendung: Das TSP-Problem

3. Bienenalgorithmen

Natürliche Bienen

Künstliche Bienen

Anwendung: Das Scheduling-Problem

Definition Schwarmintelligenz

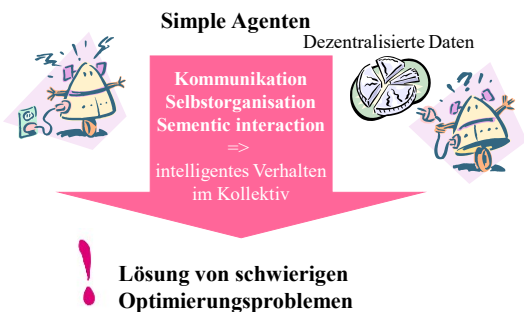
Schwarm: „verteilte KI“

- eine Menge von (mobilen) *Agenten*, die miteinander (direkt oder indirekt) *kommunizieren* können und
- im Kollektiv eine *verteilte Problemlösung* erzielen durch dezentrale und unüberwachte Koordination (**Selbstorganisation**)
- **Robustheit:** Das Gruppenziel wird erreicht, auch wenn einige Mitglieder dies nicht können,
- wobei dynamische Reaktionen auf eine veränderte Umwelt erfolgen (**Adaptivität**)

Beispiele:

- sozial lebende Insekten,
- Vogelschwärme,
- Fischschulen,
- der menschliche Körper

Schwarmintelligenz in der Informatik



Ameisenalgorithmen

Bienenalgorithmen

Ameisenalgorithmen

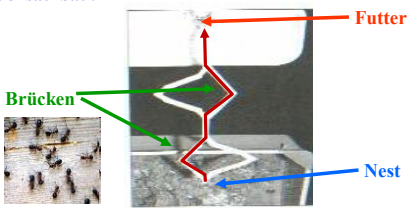


aus: Wikipedia

Agenten = Ameisen

Natürliche Ameisen

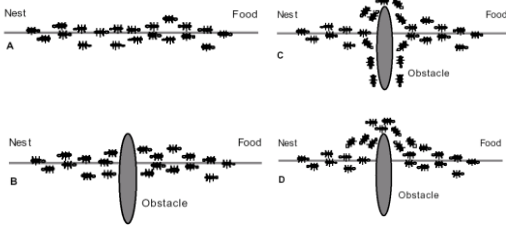
Versuchsaufbau:



Ameisenart *Linepithema humile* aus Argentinien
- zur Selbstorganisation fähig
- Semantic interactions: Pheromone

Natürliche Ameisen

Beobachtung

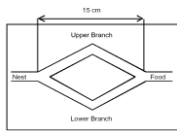


Natürliche Ameisen

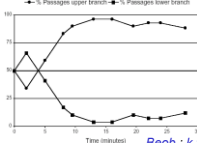
dynamische Fakten

- stärkere Pheromonbelegung \leftrightarrow höhere Begehungswahrscheinlichkeit
- Kürzerer Weg \leftrightarrow Ameise ist schneller wieder zurück, höhere Begehungswahrscheinlichkeit

Experiment: gleich lange Wege,



aber asymm. Wegsuche



Beob.: $k = 20, h = 2$

Modell

rel. Häufigkeit oben = $\frac{\text{Anzahl oben}}{\text{Gesamtzahl oben + unten}}$ $P_U(m) = \frac{(U_m + k)^h}{(U_m + k)^h + (L_m + k)^h}$

Künstliche Ameisen



Gemeinsame Analogien *Ant Colony Optimization Algorithms*

- ✓ **Agenten** = künstliche Ameisen, analog zu den echten
- ✓ optimaler Weg = **kürzester** Weg zwischen zwei Punkten
- ✓ Belegung mit Pheromonen (**Stigmergy**) als **zentrale** Idee
- ✓ Wahrscheinlichkeits-gestützte, **lokale** Entscheidungen

Künstliche Ameisen



Unterschiede künstlicher Ameisen zu natürlichen

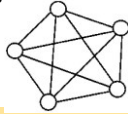
- ❖ Diskrete Welt mit Übergang zwischen diskreten Zuständen
- ❖ Interner Zustand (interner Speicher)
- ❖ Pheromonabsonderung ist problemspezifisch und unnatürlich.
- ❖ Zusatzfähigkeiten wie Vorhersehen, lokale Optimierung, back-tracking usw.

Künstliche Ameisen



Beispiel *Das Traveling Salesman Problem*
Ant System AS, Dorigo 1996

- zufällige Verteilung der Ameisen auf die Städte auf zu Beginn
- Kantenwahl: Kante mit höchstem Pheromonspiegel wird bevorzugt
- Kantenwahl: privates Gedächtnis M_k der Ameise: Was bin ich schon gelaufen?
- Modifikation des Pheromongehalts: local trail updating vs. global trail updating



Künstliche Ameisen



Algorithmus-Grundstruktur:

```

FOR t=1 TO tmax      In der Simulation
  FOR ant=1 TO m     für alle Ameisen
    build_tour TSP   wähle eine Tour
  NEXT ant
  update_pheromone  aktualisiere die Markierungen
NEXT t
    
```

Künstliche Ameisen: TSP



build_tour TSP *Wähle ein Tour*

α, β Konstanten

- Starte bei zufälliger Stadt mit Ameise k
- Bilde einmal für alle Kanten von Stadt i nach Stadt j die relative Gewichtung $a_{ij}(t) = \frac{w_{ij}(t)^\alpha (1/d_{ij})^\beta}{\sum_{r \in N_i} w_{ir}(t)^\alpha (1/d_{ir})^\beta}$

$w_{ij}(t)$ Pheromone, $1/d_{ij}$ Abstand Stadt i zu j
- Für alle Knoten: Errechne Selektionswahrscheinlichkeit p_{ij} von Ameise k für Kanten **nicht** besuchter Städte und wähle zufällig ein Kante daraus mit $p_{ij}^k(t) = \frac{a_{ij}(t)}{\sum_{r \in N_i^k} a_{ir}(t)}$

Alle nicht besuchten Wege zu i
- Lagere Pheromone ab:
 - $\Delta w_{ij} = \text{const}$ *ant-density strategy*
 - $\Delta w_{ij} = 1/d_{ij}$ *ant-quantity strategy*

Künstliche Ameisen: TSP



Update_pheromone

Nach jeder Tourensuche setze für alle Kanten der Tour

$$w_{ij}(t+1) = (1-\rho) w_{ij}(t) + \rho \Delta w_{ij}(t) \quad \text{Bewertung aktualisieren}$$

$$\text{mit } \Delta w_{ij}(t) = \sum_{k=1}^m \Delta w_{ij}^k(t) \quad \text{Bewertung aller Ameisen}$$

$$\text{und } \Delta w_{ij}^k(t) = 1/L^k(t) \quad \text{Tourlänge pro Ameise}$$

ABER: Konvergenzprobleme bei größeren Dimensionenzahl !

Künstliche Ameisen: Modifikation



Update_pheromone

Nach jeder Tourensuche setze für alle Kanten der besten Tour

$$w_{ij}(t+1) = (1-\rho) w_{ij}(t) + \rho \Delta w_{ij}(t) \quad \text{Tour markieren}$$

$$\text{mit } \Delta w_{ij}(t) = 1/L^+ \quad \text{beste Tourlänge aller Ameisen}$$

Online_update_pheromone

Nach jedem Kantenlauf setze

$$w_{ij}(t+1) = (1-\gamma) w_{ij}(t) + \gamma w_0 \quad \text{Spur verwischen}$$

$$\text{mit } w_0 = 1/n L_{mn} \quad \text{heuristische nearest-neighbor Tourlänge}$$

Ergebnisse



Vergleich mit anderen Verfahren (50-Städte-Probleme)

Problem name	ACS	SA	EN	SOM	FI
City set 1	5.86	5.88	5.98	6.06	6.03
City set 2	6.05	6.01	6.03	6.25	6.28
City set 3	5.57	5.65	5.70	5.83	5.85
City set 4	5.70	5.81	5.86	5.87	5.96
City set 5	6.17	6.33	6.49	6.70	6.71

Ant Colony System

Simulated annealing

Elastic Net

Self Organizing Map

Farthest Insertion

Ergebnisse

Vergleich mit anderen Verfahren

Problem name	ACS	GA	EP	SA	AG	Optimum
Oliver30	420	421	420	424	420	420
(30-city problem)	(423.74)	(N/A)	(423.74)	(N/A)	(N/A)	(423.74)
Iterations für beste Integer-Tour	[830]	[3,200]	[40,000]	[24,617]	[12,620]	
Eil50	425	428	426	443	436	425
(50-city problem)	(427.96)	(N/A)	(427.86)	(N/A)	(N/A)	(N/A)
	[1,830]	[25,000]	[100,000]	[68,512]	[28,111]	
Eil75	535	545	542	580	561	535
(75-city problem)	(542.31)	(N/A)	(549.18)	(N/A)	(N/A)	(N/A)
	[3,480]	[80,000]	[325,000]	[173,250]	[95,506]	
KroA100	21,282	21,761	N/A	N/A	N/A	21,282
(100-city problem)	(21,285.44)	(N/A)	(N/A)	(N/A)	(N/A)	(N/A)
	[4,820]	[103,000]	[N/A]	[N/A]	[N/A]	

Ant Colony System

Genetic Algorithms

Evolutionary Programming

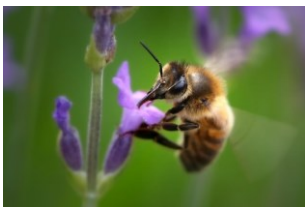
Simulated annealing

Annealing Genetic algorithm

Ameisenalgorithmen

Bienenalgorithmen

Bienenalgorithmen



aus: Wikipedia

Agent = Biene

Natürliche Bienen



m Futtersucher



Jede Biene, die zurückkommt, teilt ihr Ergebnis durch Schwänzeltanz mit.

Güte der Futterquelle = R (Distanz, Futtermenge, Futterqualität)

- Zeitl. Länge des Tanzes $\sim R$
- Richtung des Tanzes \sim Himmelsrichtung

Natürliche Bienen



- ❖ Weitere Futtersucher beobachten zufällig einen Tanz und fliegen los.
- ❖ Je **länger** der Tanz dauert, umso **mehr** Futtersucher fliegen los.
 - Gute Futterorte triggern positive Rückkopplung
 - Schlechte bedeuten eine relative Abnahme der Bienen

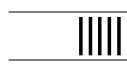
Künstliche Bienen



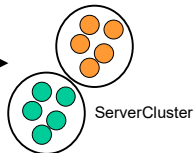
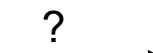
Anwendung: Scheduling von Internet-Servern



Blumenbeet



HTTP-Anfragen



ServerCluster

Scheduling



Zuordnungen

- **Blumenbeet** Warteschlange
- **Futtersuchbiene** Server
- **Gruppe der Futtersucher** virt. ServerCluster
- **Bienenstock** Menge aller Server
- **Futterqualität**, Menge, Distanz ServiceQualität, Menge, Dauer
- **Aufwand**, Futterort zu wechseln Migrationskosten
- **Schwänzeltanz** Nachricht
- **Tanzboden** Nachrichtenbrett

Künstliche Bienen



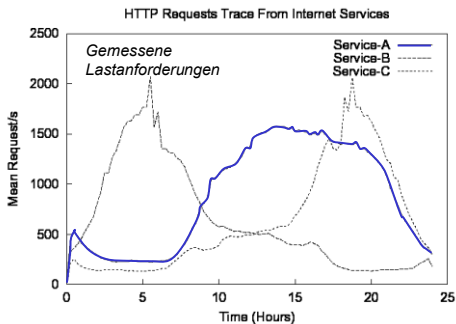
Jeder Server s_j des virt. ServerClusters VS_j

- Arbeitet eine Anfrage ab. Ertrag ist c_j [€/Auftrag]
- Heftet mit Wahrscheinlichkeit q eine Nachricht ans Brett
- Diese bleibt eine Zeit $T = c_j A$ erhalten.
- Vergleicht die eigene Profitrate P_i mit der von Gesamtgruppe P_{colony} und errechnet damit Wechsel $r_i(P_i)$



- **Futtersucher**: Liest mit Wahsch. r_j eine andere Nachricht vom Brett und befolgt sie.
- oder **Scout**: wählt direkt mit Wahsch. r_j ein anderes VS_j

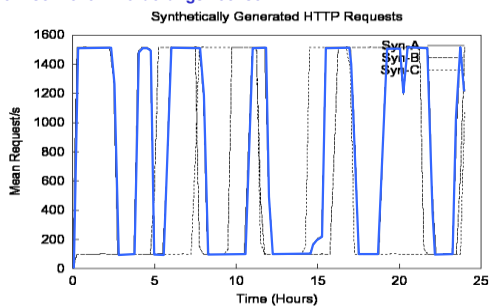
Anforderungen



Synthetische Daten



Sehr schnelle Anforderungswechsel



Ergebnisse



Mean requests per second for synthetically generated HTTP requests
Revenue(€)

	2-VirtServer-cluster		3-VirtServer-cluster	
	Actual	Projected	Actual	Projected
Revenue	901,203.00	901,203.00	1,119,200.00	1,159,039.58
Energy	976,213.00	976,213.00	1,003,050.00	1,003,050.00
Capacity	876,477.00	876,477.00	868,087.00	868,087.00
Service Revenue	876,477.00	876,477.00	868,087.00	876,210.90

Revenue for real (physical) service tracks
Revenue(€)

	2-Virt		3-Virt	
	Actual	Projected	Actual	Projected
Revenue	1,336,060.00	1,071,741.83	1,336,060.00	1,363,872.18
Energy	1,336,060.00	1,336,060.00	1,336,060.00	1,336,060.00
Capacity	1,108,564.58	1,108,564.58	1,108,564.58	1,108,564.58
Service Revenue	1,108,564.58	846,076.00	1,108,564.58	1,108,564.58

Ausblick

- Forschung in der Schwarmintelligenz hat gerade begonnen
- Umsetzung der vorgestellten Modelle basiert auf Annahmen und spiegeln nur teilweise die biologische Realität wider
- treibende Aspekte für die Entwicklung der informatischen Modelle sind die Effizienz, Robustheit und Flexibilität der biologischen Vorbildsysteme

Ausblick