Complex Systems Engineering

Selbstorganisierende Systeme

Christin Seifert

15. November 2017

University of Passau, WS 2017/2018

Überblick

1. Einführung
2. Biologische Vorbilder
3. Abgeleitete Optimierungsalgorithmen
4. Quantifizierung
5. Zusammenfassung

Einführung

Selbstorganisation

- Komplexe Strutkuren können aus einfachen Regeln entstehen (Bsp. Fraktale)
- Aus anfänglich wenig Ordnung kann mehr Ordnung durch Zusammenspiel von einfachen autonomen Agenten entstehen (z.B. in Zellulären Automaten)
- In solchen Systemen ist Selbstorganisation ist eine *emergente* Eigenschaft
- Die Eigenschaft der Selbstorganisation kann nicht direkt aus den Eigenschaften der Teile abgeleitet werden (Können Sie für Wolfram Regel 145 vorhersagen, wie sich das System über die Zeit entwickeln wird?)

Definition Selbstorganisation

Selbstorganisierendes System ist ein System, in dem sich eine höhere Ordnung durch lokales Zusammenspiel von Agenten ergibt, die einfachen Regeln folgen. Der Prozess der Systementwicklung ist spontan und geschieht ohne äußere Steuerung.
Elemente der Selbstorganisation

- Mehrere Akteure
- Interaktionen zwischen Akteuren (entweder direkt oder indirekt über die Umwelt)

Beispiele Selbstorganisation

Abbildung 1: Galaxy mit Stern NGC 4656 (NASA multimedia data base `https://www.nasa.gov/multimedia/`)

Abbildung 2: Stare im Schwarm (CC-BY-SA 2.0, John Holmes, via Wikimedia Commons)

Beispiele Selbstorganisation

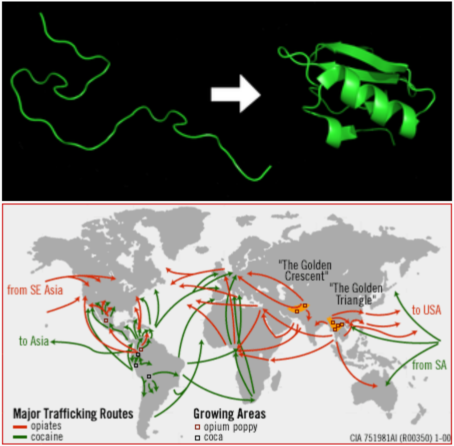


Abbildung 3: Spontane Proteinfaltung und Internationale Drogenrouten (Public Domain, via Wikimedia Commons)

- Leben
- Schneekristalle
- Scale-Free-Netzwerke
- Bienenstöcke
- Ameisenkollonien
- Fischschwärme
- Phasenübergänge (Plasma, Gas, Flüssigkeit, Festkörper)
- Aktienanleger*innen (Herdenverhalten)

Definition Emergenz

Emergenz ist die Herausbildung von neuen Eigenschaften oder Strukturen eines Systems infolge des Zusammenspiels seiner Elemente. Solche Eigenschaften bezeichnet man als emergente Eigenschaften eines Systems.

- Positive Emergenz: Ausbildung erwünschter Eigenschaften
- Negative Emergenz: Ausbildung unerwünschter Eigenschaften

Beispiele emergenter Eigenschaften

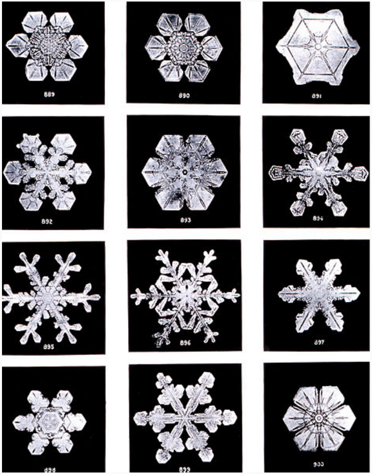


Abbildung 4:
Schneeflockenmuster nach Wilson Bentley (Public Domain, via Wikimedia Commons)

- Bewegungsmuster in Fisch- und Vogelschwärmen
- Wellenmuster am Strand
- Wellenstruktur der Laola-Welle im Stadium (lokale Regeln)
- Form der Bienenwaben und Termitenhügel
- Bildung eines Staus „aus dem Nichts“
- Temperatur eines Körpers (kein einzelnes Teilchen besitzt die Eigenschaft „Temperatur“)

Biologische Vorbilder

Honigbienen

- Jede Honigbiene ist ein Individuum
- Es ist nicht anzunehmen, dass eine Biene „weiß“, wie die Struktur, des gesamten Bienenstockes auszusehen hat
- Konstruktion eines Bienenstockes dauert länger als mehrere durchschnittliche Bienenleben
- Der Bienenstock ist hochkomplex, symmetrisch und viel größer als eine einzelne Biene

Abbildung 5: Honigbiene in Waben(CC-SA 3.0, Waugsberg, via Wikimedia Commons)

Honigbienen

- Bienenwaben haben perfekte Hexaederstruktur
- Es wird angenommen, dass Bienen runde Waben bauen wollen
- Hexagonale Waben entstehen dadurch, dass in den Nachbarzellen ebenfalls Bienen runde Waben erstellen wollen und die entstehenden Druckkräfte zu hexagonalen Mustern führen

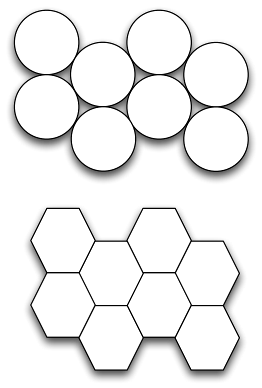


Abbildung 6: Muster der Bienenwaben. Oben: Idee, unten: Ergebnis

- Die Wabe muss sowohl den Königinnenfuttersaft als auch das Ei beherbergen, ein flacher Boden wäre suboptimal
- Der Boden einer Wabe hat eine Einbuchtung, die einer dreiseitigen Pyramide entspricht
- Die Bestimmung des optimalen Winkels der Pyramidenseiten zur Minimierung des Arbeits- und Materialaufwandes ist kein triviales mathematisches Problem
- Die Bienenwabe zeigt genau die optimalen Winkel von $109^\circ 26'$ und $70^\circ 34'$ [4]

$\Rightarrow$ Bienenwaben sind ein Beispiel dafür, dass optimale Systemeigenschaften entstehen, obwohl deren Regeln nicht explizit in der Systemkonstruktion hinterlegt sind

Ameisen

- Ameisen haben verschiedene Sensoren, um ihre Umwelt wahrzunehmen
- Gerüche nehmen sie über Antennen wahr, die ihnen die Richtung, die Art und die Intensität des Geruches anzeigen
- Ameisen können unterscheiden, ob Gerüche von Ameisen aus dem eigenen Ameisenstaat kommen oder von fremden Ameisen

Abbildung 7: Ameisenstraße
(CC-SA 3.0, Fir0002, via Wikimedia Commons)

Futtersuche [3]

- Futtersuchende Ameisen hinterlassen eine Pheremonspur auf dem Rückweg von der Futterquelle
- Andere Ameisen folgen diesen ausgewiesenen Wegen und verstärken den Geruch, wenn sie Futter gefunden haben
- Wenn die Futterquelle leer ist, wird der Geruch nicht mehr verstärkt und verblasst mit der Zeit
- Wenn der Weg zur Futterquelle versperrt ist, verlässt die Ameise den Pfad, sucht eine neue Futterquelle und markiert den kürzesten Rückweg von der neuen Futterquelle

$\Rightarrow$ Ameisen finden kollaborativ den kürzesten Weg zu den besten Futterquellen

Ameisen

Das Doppelbrückenexperiment [3, 1]

- Experimente zur Pfadsuche bei Ameisen
- Welchen Weg vom Nest zu Futter wählen Ameisen?
- In einem Experiment beide Pfade gleich lang, im anderen Experiment ein Pfad viel länger
- Symmetrisches Setup: Ameisen verwenden mit der Zeit nur eine der beiden Brücken. Die Auswahl ist zufällig, beide Brücken werden ungefähr in gleich vielen Versuchen gewählt.
- Asymmetrisches Setup: Ameisen verwenden den kürzeren Pfad.

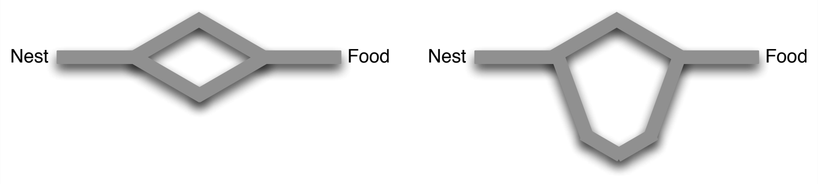


Abbildung 8: Symmetrisches (links) und asymmetrisches Setup

Abgeleitete Optimierungsalgorithmen

Das Problem des Handlungsreisenden

Es sollen die 15 größten Städte Deutschlands[1] besucht werden.
Welche Reiseroute minimiert die Gesamtwegstrecke?

Stadt	Einwohnerzahl
Berlin	3.520.031
Hamburg	1.787.408
München	1.450.381
Köln	1.060.582
Frankfurt am Main	732.688
Stuttgart	623.738
Düsseldorf	612.178
Dortmund	586.181
Essen	581.624
Leipzig	560.472
Bremen	557.464
Dresden	543.825
Hannover	532.163
Nürnberg	509.975
Duisburg	491.231

[1] Quelle der Daten: Statistisches Bundesamt, 2015

Wie viele mögliche Rundreisen gibt es in einem vollständigen Graphen mit

- 3 Knoten,
- 4 Knoten,
- 5 Knoten

Lösung

- 1
- 3
- 12

- Für die Länge der Rundreise spielt es keine Rolle, mit welcher Stadt begonnen wird, d.h. es kann eine Stadt zufällig fest gewählt werden
- Auswahl zweite Stadt: 14 Möglichkeiten
- Auswahl dritte Stadt: 13 Möglichkeiten
- usw.
- Es ist egal, ob die Rundreise in der einen oder anderen Richtung abgefahren wird, d.h. die Anzahl der Möglichkeiten halbiert sich
- Anzahl möglicher Rundreisen

$$14 \cdot 13 \cdot \ldots \cdot 1 = \frac{14!}{2} = 43.589.145.600$$

- Angenommen, die Berechnung der Länge einer Route dauert 1 msec, dann dauert die Berechnung aller Routen $14! : 2 : 1000 : 60 : 60 : 24 : 360 \approx 1,4$ Jahre

Das Problem des Handlungsreisenden

Abbildung 9: Kürzeste Strecke durch die 15 größten Städte Deutschlands
(Public Domain, via Wikimedia Commons)

Das Problem des Handlungsreisenden

- Das Problem des Handlungsreisenden (auch Travelling Salesperson Problem, kurz TSP) ist NP-vollständig[2]
- Es lässt sich somit nicht in polynomieller Zeit optimal lösen und gehört zu den schwierigsten Problemen der Informatik
- Anwendungsprobleme, denen TSP zugrunde liegt, sind z.B.
  - Chip-Design (Optimierung der Verbindungen zwischen Schaltkreisen)
  - Auslieferung von Pakten durch Paketdienste
  - Besuche von Kunden eines Serviceunternehmens
  - Genomsequenzierung (DNA-Teilstränge sind „Städte “, Ähnlichkeit zwischen Teilsequenzen entspricht der Entfernung)

[2] Unter der unbewiesenen Annahme, dass $P \neq NP$

Das Problem des Handlungsreisenden

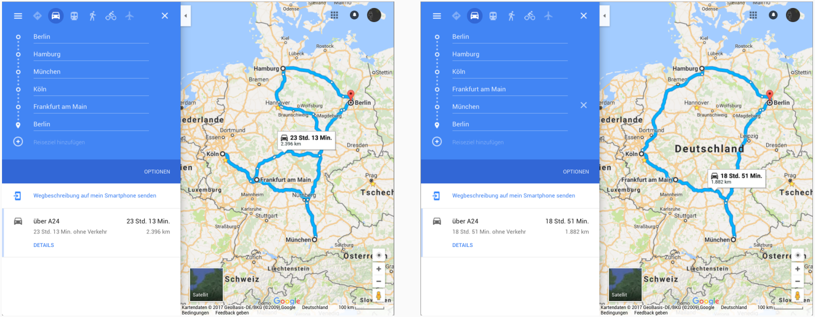


Abbildung 10: Zwei mögliche Lösungen des TSP für die 5 größten deutschen Städte, 2396 km und 1882 km (Screenshots mit maps.google.de)

Definition Kombinatorisches Optimierungsproblem

Ein kombinatorisches Optimierungsproblem ist definiert durch das Tripel (S, f, Ω), wobei

- S: Menge aller möglichen Lösungen
- f: Zielfunktion
- Ω: Menge von Randbedingungen

$\tilde{S} \subseteq S$, die Ω erfüllen, heißen *zulässige Lösungen*.

Ziel der kombinatorischen Optimierung, ist das Finden derjenigen zulässigen Lösungen, für die die Zielfunktion global optimal (d.h. minimal oder maximal) ist.

Definition Kombinatorisches Optimierungsproblem

Ein kombinatorisches Optimierungsproblem ist definiert durch das Tripel (S, f, Ω), wobei

- S: Menge aller möglichen Lösungen
- f: Zielfunktion
- Ω: Menge von Randbedingungen

$\tilde{S} \subseteq S$, die Ω erfüllen, heißen *zulässige Lösungen*.

Benennen Sie mögliche (S, f, Ω) für das TSP.

TSP ist ein kombinatorisches Optimierungsproblem.

- Menge der möglichen Lösungen Ω: geordnete Liste von Städten, also eine Reise
- Zielfunktion: Länge des Reiseweges (z.B. in km, als Zeit- oder Spritverbrauch)
- Randbedingungen: Anfangspunkt gleich Endpunkt, alle Städte kommen genau einmal vor
- Ziel ist das Finden der Liste von Städten, die die Randbedingung erfüllen (d.h. zulässige Lösungen sind) und die Zielfunktion (z.B. die Reisezeit) minimieren

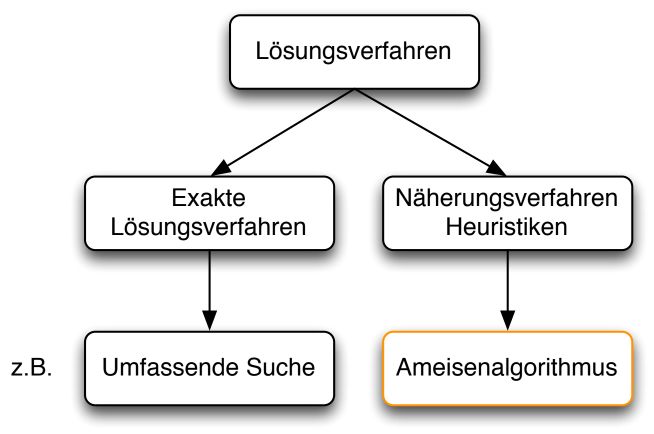


Abbildung 11: Lösungsansätze für kombinatorische Optimierungsprobleme

Darstellung des TSP als graphentheoretisches Problem

TSP als Graph

TSP kann als vollständiger, gewichteter Graph $G = (N, A)$ modelliert werden, mit

- N: Menge der Knoten (Städte)
- A: Menge der Kanten (Wege)
- $d_{i,j}$: Kantengewichte für Kanten zwischen Knoten i und j (Weglänge)

Optimierungsziel ist das Finden einer Sequenz von Knotenindizes, sodass die Weglänge minimiert wird

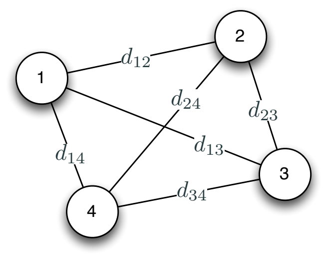


Abbildung 12: Vollständiger, gewichteter Graph mit 4 Knoten

Ameisenalgorithmus für TSP

- m Ameisen werden zufällig auf Städten platziert
- Die Kanten bekommen zwei Gewichte, die Pheremonstärke τ_{ij} („Wie gut riecht es nach optimaler Lösung“) und den heuristischen Wert η_{ij} („Wie gut ist die Kante “)[3]

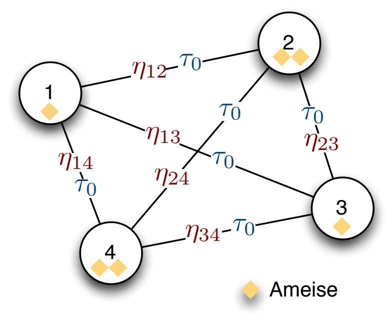


Abbildung 13: Initialisierung

$$\tau_{ij} = \tau_0 = \frac{m}{C}$$

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

m :Anzahl Ameisen

d_{ij} :Distanz zwischen Stadt i und j

[3] C ist ein kritischer Parameter, für eine Diskussion siehe [2]

1. Generierung der Rundreisen

- Jede der m Ameisen konstruiert ihre eigene Rundreise
- In jeder Stadt i muss die Ameise entscheiden, welche Stadt sie als nächstes besuchen soll
- Entscheidung hängt von der Pheremonstärke τ_{ij} und der Länge der Kante η_{ij} ab
- Wenn Ameise k sich gerade in Stadt i aufhält, ist die Wahrscheinlichkeit, dass sie in Stadt j geht, gegeben durch

Probabilistische Aktionsregel

$$p_{ij}^k = \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{l \in N_i^k} \tau_{il}^\alpha \eta_{il}^\beta}, \quad \text{if } j \in N_i^k$$

Ameisenalgorithmus für TSP

$$p_{ij}^k = \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{l \in N_i^k} \tau_{il}^\alpha \eta_{il}^\beta}, \quad \text{if } j \in N_i^k$$

- N_i^k: zulässige Nachbarschaft von Ameise k in Stadt i, d.h. alle Städte, die die Ameise bisher noch nicht besucht hat
- Je höher τ_{ij}^α oder/und η_{ij}^α desto höher die Wahrscheinlichkeit, Stadt j zu besuchen
- Die Summe normalisiert den Wert über alle alternativen Pfade, die von i ausgehen und zulässig sind
- α und β sind Parameter, die bestimmen, welchen Einfluss die Pheremonspur bzw. die heuristische Funktion hat
    - $\alpha = 0$ - Pheremone spielen keine Rolle, Ameise geht zur Stadt, die auf dem kürzesten Weg zu erreiche ist; klassische Greedy-Suche
    - $\beta = 0$ - Distanzen der Städte spielen keine Rolle
    - Empirisch gute Parameter: $\alpha = 1, \beta \in \{2, 3, 4, 5\}$

2. Aktualisierung der Pheremonspur

- Ameise k hat eine zulässige Rundreise T^k erzeugt
- C^k sei die Länge der Tour T^k
- Der Pheremonwert einer Kante wird
  1. erniedrigt, um die Verflüchtigung von Pheremonen zu simulieren

$$\tau_{ij} \leftarrow (1-p)\tau_{ij}, \quad \text{mit } 0 < p \leq 1 \text{ Verflüchtigungsrate}$$

  2. erhöht, um einen Wert, der dem Grad der Optimalität der gefunden Spur entspricht

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^{n} \Delta\tau_{ij}^k$$

$$\Delta\tau_{ij}^k = \begin{cases} \frac{1}{C^k} & \text{falls Kante}(i,j) \in T^k \\ 0 & \textit{sonst} \end{cases}$$

ACO Algorithmus

Data: Graph G(N,A), *m*, *C*, α, β
Result: Rundreise für G
Initialize $\tau_{ij} = \frac{m}{C}$, $\eta_{ij} = \frac{1}{d_{ij}}$;
while *not converged* **do**
  for *each ant k* **do**
    `/* Ant moves forward */`
    follow-pheromon-trail();
    $T^k \leftarrow$ von Ameise *k* erzeugte Tour;
  end
  for *each ant k* **do**
    `/* Ant circles back its trail */`
    update-pheremon-trail()
  end
end

Algorithm 1: Sketch of ACO algorithm

Ameisenalgorithmus für TSP

Mögliche Abbruchkriterien:

- Anzahl Updates ohne Verbesserung der Lösung
- Verbrauchte CPU-Zeit
- Anzahl Iterationen

Es gibt eine Vielzahl an Erweiterungen des ACO-Algorithmus [2], hier wurde der Basisalgorithmus vorgestellt.

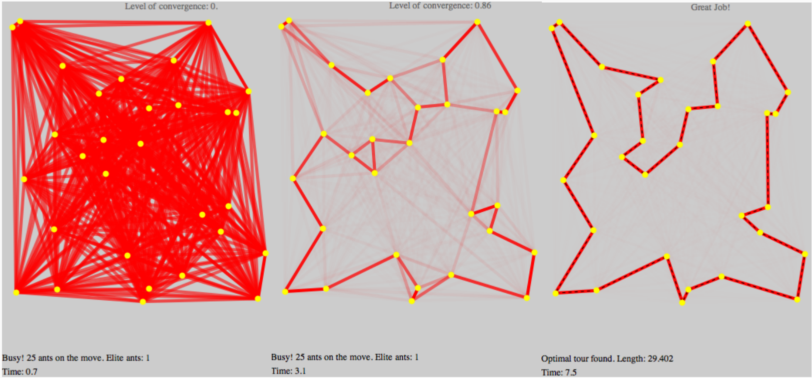


Abbildung 14: Beispiel mit 30 Städten erzeugt mit Wolfram Demo
`http://demonstrations.wolfram.com/AntColonyOptimizationACO/`

Ameisenalgorithmus für TSP

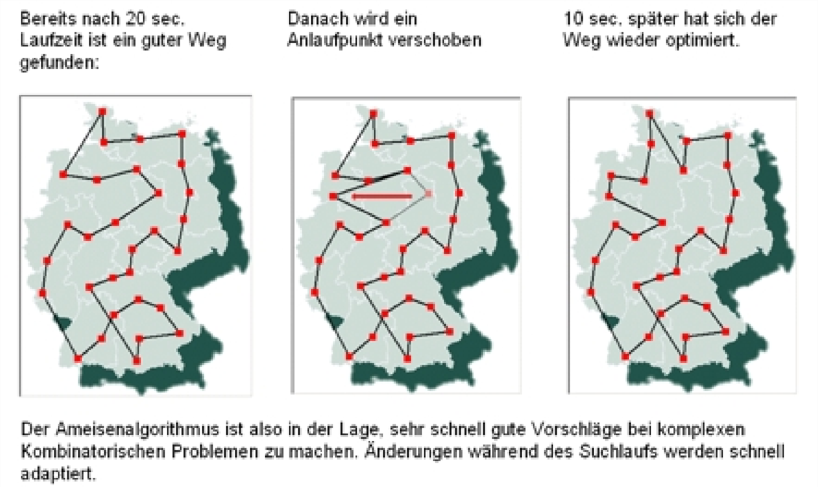


Abbildung 15: Adaptivität des ACO Algorithmus (CC-SA 3.0, Carpe Retem, via Wikimedia Commons)

Ameisenalgorithmus

Vorteile des ACO-Algorithmus

- Inhärente Parallelität (Kolonie von Ameisen)
- Positives Feedback führt zur Verstärkung guter Lösungen
- Kann adaptiv implementiert werden (z.B. Hinzufügen neuer Städte)
- Konvergiert (gegen lokal optimale Lösung)

Nachteile des ACO-Algorithmus

- Theoretische Analyse des Algorithmus ist schwierig
- Global optimale Lösung nicht garantiert (Heuristik)
- Anzahl Iterationen bis Konvergenz nicht abschätzbar

Quantifizierung

Entropie

- Selbstorganisation kann über Entropie quantifiziert werden
- Entropie ist ein Maß für den Informationsgehalt basierend auf Wahrscheinlichkeiten von Ereignissen
- 1949 von Claude Shannon definiert
- Generelle Idee:
  - Ist ein Ereignis unwahrscheinlich, dann ist die Information, die man bekommt, wenn es auftritt, sehr hoch
  - Ist ein Ereignis wahrscheinlich, dann ist die Information über sein Auftreten, wenn es auftritt, niedrig

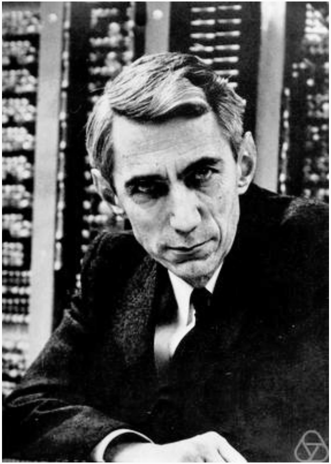


Abbildung 16: Claude Shannon (CC-SA 2.0, MFO, via Wikimedia Commons)

Informationsgehalt eines Ereignisses

Der Informationsgehalt I eines Ereignisses, das mit Wahrscheinlichkeit p auftritt, ist definiert als

$$I = \log_2 \frac{1}{p} = -\log_2 p$$

Die Einheit für den Informationsgehalt (IG) ist bit.[4]

Beispiele:

- IG des Ereignisses *Kopf* bei Münzwurf: $\log_2 \frac{2}{1} = 1$
- IG des Ereignisses *5* bei Würfelwurf: $\log_2 \frac{6}{1} \approx 2,59$
- IG des Ereignisses *gerade Zahl* bei Würfelwurf: $\log_2 \frac{6}{3} = 1$

[4] Wenn der Logarithmus zur Basis 2 verwendet wird.

Aufgabe

Wie hoch ist jeweils der Informationsgehalt folgender Ereignisse:

1. Ziehen des Herz As' in einem Skat-Deck (32 Karten)
2. Ziehen eines As' in einem Skat-Deck
3. Eine Uni hat 4 Fakultäten. An Fakultät A sind 300 Studierende, an Fakultät B 200 Studierende und an Fakultät C sind 500 Studierende. Wie hoch ist der IG des Ereignisses "Treffen eines Studierenden der Fakultät A".

Lösung

1. 5
2. 3
3. 1,7

Entropie

- Der Informationsgehalt kann für Ereignisfolgen mit ungleichförmigen Verteilungen generalisiert werden.
- Berechnet wird die gewichtete Summe der Informationsgehalte der Einzelereignisse, Gewicht ist die Auftrittswahrscheinlichkeit des Ereignisses. Das Ergebnis ist die Entropie.

Entropie

Die Entropie H für eine Ereignisfolge ist definiert als

$$H = \sum_{i=1}^{n} p_i \log_2 \frac{1}{p_i}$$

wobei n die Anzahl möglicher Ereignisse ist.

- Es gilt $0 \leq H \leq \log_2(n)$.
- Gleichverteilungen haben Entropie maximale, Verteilungen mit $p_i = 1, p_k = 0 \quad \forall k \neq i$ Entropie Null.

Beispiel. Informationsgehalt von Zeichenfolgen.

Folge 1: AABAACAAA
Folge 2: ABACCBACA
Folge 3: ABABACCCB

- Folge 1:
  - Buchstabe A, Wahrscheinlichkeit des Auftretens $\frac{7}{9}$
  - Buchstaben B und C, Wahrscheinlichkeit des Auftretens jeweils $\frac{1}{9}$
  - $H_1 = \frac{7}{9} \log_2 \frac{9}{7} + 2 \cdot \frac{1}{9} \log_2 \frac{9}{1} \approx 0,985$
- Folge 2:
  - Buchstabe A, Wahrscheinlichkeit des Auftretens $\frac{4}{9}$
  - Buchstabe B, Wahrscheinlichkeit des Auftretens $\frac{2}{9}$
  - Buchstabe C, Wahrscheinlichkeit des Auftretens $\frac{3}{9}$
  - $H_2 = \frac{4}{9} \log_2 \frac{9}{4} + \frac{2}{9} \log_2 \frac{9}{2} + \frac{3}{9} \log_2 \frac{9}{3} \approx 1,530$
- Folge 3:
  - Wahrscheinlichkeit des Auftretens aller Buchstaben $\frac{3}{9}$
  - $H_3 = 3 \cdot \frac{3}{9} \log_2 \frac{9}{3} \approx 1,585$

AUFGABE

Berechnen Sie die Entropie der Zeichenfolgen:

Folge A: 0000010
Folge B: 01001101

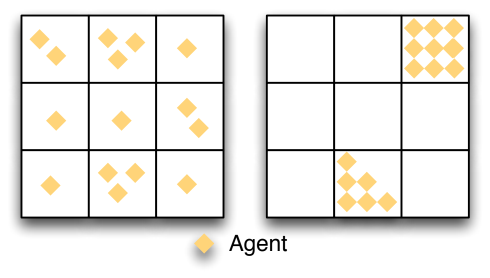
Lösung Folge A

- Zeichen 0, Wahrscheinlichkeit des Auftretens $\frac{6}{7}$
- Zeichen 1, Wahrscheinlichkeit des Auftretens $\frac{1}{7}$
- H = 6/7 log (7/6) + 1/7 log (7/1) $\approx$ 0.59

Lösung Folge B

- Zeichen 0, 1 Wahrscheinlichkeit des Auftretens $\frac{4}{8}$
- H = 2 * 1/2 log (2) = 1

Beispiel. Verteilung von Agenten auf einem zweidimensionalen Grid.



Linkes Grid:

$$H_l = 5 \cdot \frac{1}{15} \log_2 \frac{15}{1} + 2 \cdot \frac{2}{15} \log_2 \frac{15}{2} + 2 \cdot \frac{3}{15} \log_2 \frac{15}{3} \approx 3,006$$

Rechtes Grid:

$$H_r = \frac{9}{15} \log_2 \frac{15}{9} + \frac{6}{15} \log_2 \frac{15}{6} \approx 0,971$$

Entropiedifferenz

- Das Maß an Selbstorganisation kann als Entropiedifferenz quantifiziert werden.
- Die absolute Entropiedifferenz vergleicht die Entropie zum Zeitpunkt t mit der maximal möglichen Entropie (theoretischer Wert).
- Die relative Entropiedifferenz vergleicht die Entropie zwischen zwei Systemzuständen.

Absolute Entropiedifferenz

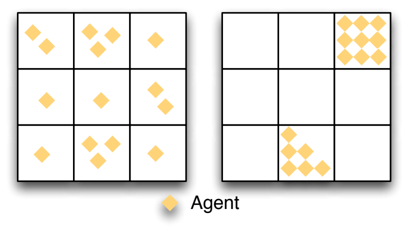
$$\Delta H^{abs}(t) = H_{max} - H(t), \quad \text{t - Zeitpunkt}$$

Relative Entropiedifferenz

$$\Delta H^{rel}(t_1, t_2) = H(t_1) - H(t_2), \qquad t_1, t_2 \text{ - Zeitpunkte}$$

Entropiedifferenz Ameisenalgorithmus

- Alle Pfade entsprechen möglichen Ereignissen.
- Die Wahrscheinlichkeit eines Pfades ist der Anteil der Ameisen, die diesen Pfad zu einem bestimmten Zeitpunkt geht.
- Beispiel: Gridzellen entsprechen Pfaden, Agenten sind Ameisen, links Zeitpunkt t_1, rechts t_2



- $\Delta H^{abs}(t_2) = H_{max} - H(t) = \log_2(15) - 0,971 \approx 2,94$
- $\Delta H^{rel}(t_2, t_1) = H(t_2) - H(t_1) \approx 1,775$
- Der Ameisenalgorithmus vermindert somit die Entropie.

Zusammenfassung

- In selbstorganisierenden System entsteht eine höhere Ordnung durch lokales Zusammenspiel von einfachen Agenten.
- In selbstorganisierenden Systemen lassen sich emergente Eigenschaften beobachten, das sind Eigenschaften, die erst durch das Zusammenspiel der Agenten im System entstehen.
- Es gibt viele natürliche selbstorganiserende Systeme.
- Selbstorganisierende Systeme dienen als Grundlagen für Optimierungsalgorithmen.
- Die Selbstorganisation kann mit Hilfe der Entropiedifferenz quantifiziert werden.

Wichtige Konzepte

- Selbstorganisation
- Emergenz
- Problem des Handlungsreisenden/Traveling Salesperson Problem (TSP)
- Ameisenalgorithmus/Ant Colony Optimization (ACO)
- Heuristik
- Kombinatorische Optimierung
- Entropie
- Entropiedifferenz
  - absolute
  - relative

Weiterführende Literatur

- Ant Colony Optimization
  (PDF Version online verfügbar)
  Marco Dorigo und Thomas Stüzle. *Ant Colony Optimization*. The MIT Press, 2004. URL: https://pdfs.semanticscholar.org/7c72/393febe25ef5ce2f5614a75a69e1ed0d9857.pdf

Literatur

[1] J. -L. Deneubourg u. a. "The self-organizing exploratory pattern of the argentine ant". In: *Journal of Insect Behavior* 3.2 (März 1990), S. 159–168. ISSN: 1572-8889. DOI: 10.1007/BF01417909. URL: https://doi.org/10.1007/BF01417909.

[2] Marco Dorigo und Thomas Stüzle. *Ant Colony Optimization*. The MIT Press, 2004. URL: https://pdfs.semanticscholar.org/7c72/393febe25ef5ce2f5614a75a69e1ed0d9857.pdf.

[3] S. Goss u. a. "Self-organized shortcuts in the Argentine ant". In: *Naturwissenschaften* 76.12 (Dez. 1989), S. 579–581. ISSN: 1432-1904. DOI: 10.1007/BF00462870. URL: https://dipot.ulb.ac.be/dspace/bitstream/2013/19271/1/042GossNaturwissenschaften89.pdf.

[4] Maurice Maeterlinck. *The Life of the Bee*. George Allen & Unwin Ltd, 1901. URL: https://sunsetridgemsbiology.wikispaces.com/file/view/The-Life-of-the-Bee.pdf.