ADS: Algorithmen und Datenstrukturen 2

Teil 12

Dr. Thomas Efer

Institut für Informatik
Abteilung Automatische Sprachverarbeitung
Universität Leipzig

27. Juni 2018

[Letzte Aktualisierung: 29/06/2018, 10:47]

Nature-Inspired Computation

Themenüberblick:

- Einführung
- Zelluläre Automaten und agentenbasierte Simulationen
  - Elementare Zelluläre Automaten
  - Game of Life
  - Wireworld
  - Langton's Ant
- Metaheuristiken und Optimierung
  - Ameisenalgorithmen
  - Particle Swarm Optimization
  - Genetische Algorithmen *(revisited)*
- Künstliche Neuronale Netze
  - Biologische Grundlagen und notwendige Abstraktionen
  - Perzeptron: Strukturen, Aktivierung und Backpropagation
  - Komplexere Netzwerkarchitekturen

Wireworld - Regeln

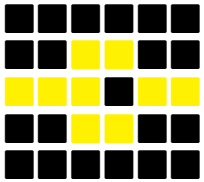
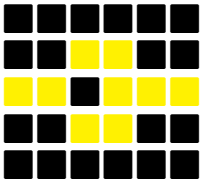
Zweidimensional, Moore-Nachbarschaft, vier Zustände:

- ■ - Nichts
- ■ - elektrischer Leiter
- ■ - „Elektronenkopf“
- ■ - „Elektronenende“

Die Regeln sind so gestaltet, dass sich das „Elektron“ pro Zeitschritt in seine Kopfrichtung bewegt (und da bei ggf. aufgesplittet oder ausgelöscht wird):

- ■ → ■
- ■ → ■
- ■ → ■
- ■ → {■, falls 1 oder 2 mal ■ in Nachbarschaft vorkommt, sonst ■}

Wireworld - Diode

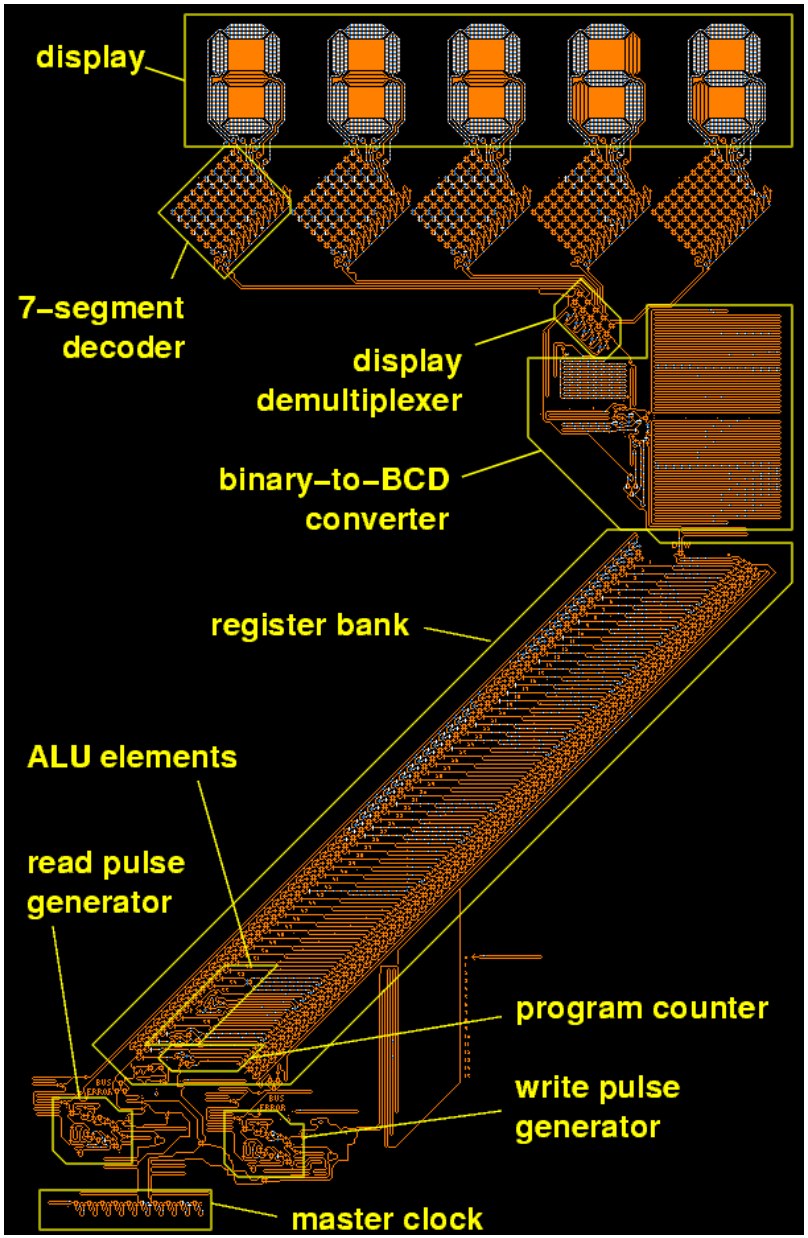
Durchlassrichtung  und Sperrichtung: 

vs.

Ausprobieren: `https://xithiox.github.io/wireworld/`

(Setzen mit der Maus, Tasten ändert Werkzeug: W=■, H=■, T=■, M=■, Leertaste simuliert, X animiert)

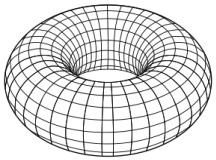
Wireworld - Komplexe Schaltkreise



Erklärung:
https://www.logre.eu/wiki/Projet_Wireworld/en

Zelluläre Automaten - Ausblick

Die n-dimensionalen Positionsräume lassen sich (wahlweise komplett oder nur teilweise) in endliche Welten umwandeln.



Dafür kann ein „Warping“ an den Seiten stattfinden (über Modulo-Operation) oder z. B. über die Definition fixer Werte am „Rand“.

Zelluläre Automaten können darüber hinaus auf Graphen ausgeführt werden, wobei die Positionsmenge der Kantenmenge entspricht und die Nachbarschaft durch Kantentraversierung gebildet wird. Zwei Fälle:

- Regelmäßige Gitter oder algebraische Verbände: Gleicher Knotengad für alle Knoten
- Beliebige, auch nicht zusammenhängende Graphen: Regeln müssen für Nachbarschaften variabler Mächtigkeit ausgelegt sein

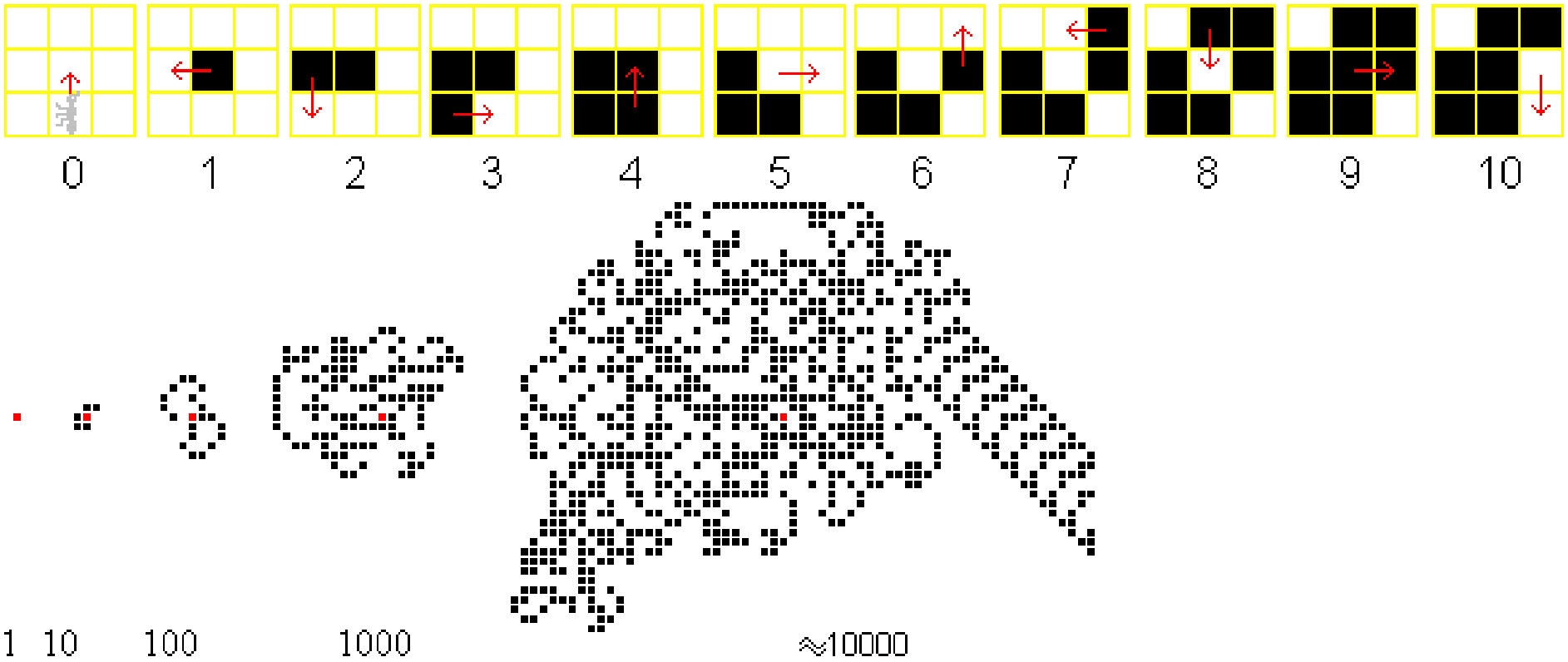
Langton's Ant

- erstmals 1986 von Christopher Langton beschrieben
- 2D-Spielfeld mit Zellen und zwei möglichen Zuständen ■ und □
- Zustandsänderungen nimmt allein die „Ameise“ vor:
  - merkt sich aktuelle Position
  - merkt sich aktuelle Richtung ($\rightarrow \uparrow \leftarrow \downarrow$)
  - „betritt“ eine Zelle, dreht sich anschließend bei
    - □ um 90° nach links
    - ■ um 90° nach rechts

    und invertiert den Zustand der Zelle.
- Ausprobieren:

  `http://www.dwmkerr.com/experiments/langtonsant/`
- mehrere Ameisen denkbar, ähnlich wie „agentenbasierte“ Simulationen

Langton's Ant

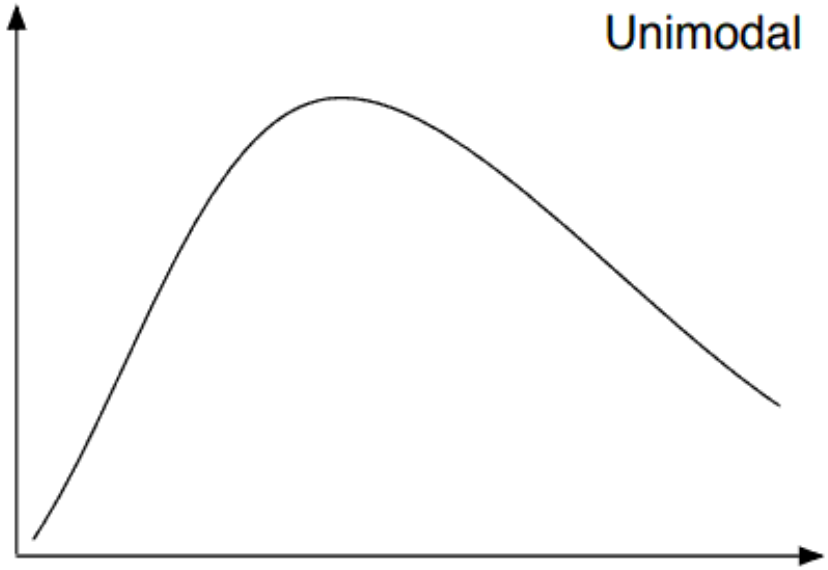


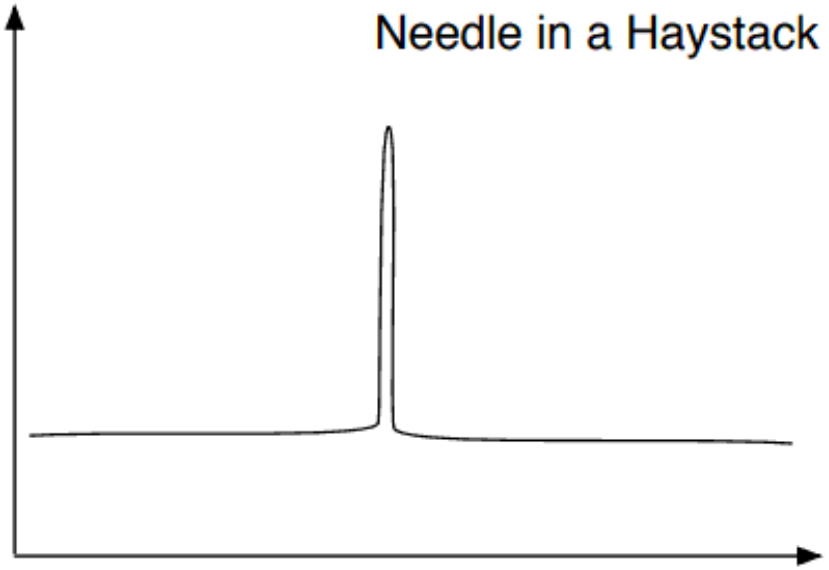
Wie kann dieses Verhalten im bisher kennengelernten Paradigma Zellulärer Automaten abgebildet werden?

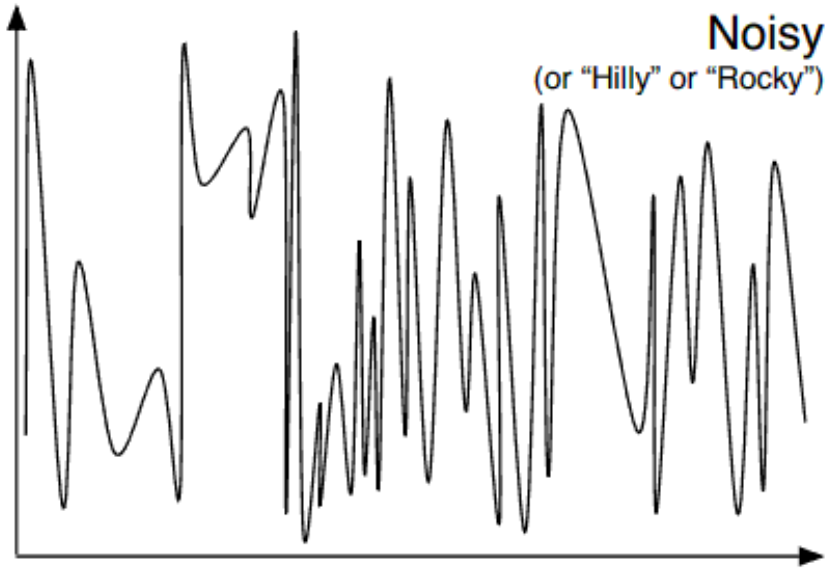
Gleich: mehr Ameisen

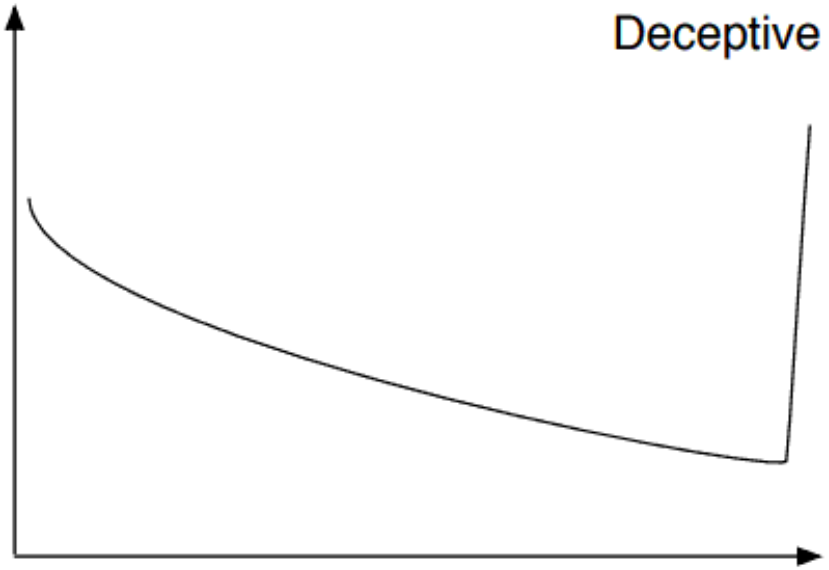
Metaheuristiken und Optimierung

- Metaheuristiken sind Verfahren der Stochastischen Optimierung (auch *Black-Box Optimization* genannt)
- Ansatz generischer Optimierungsverfahren im Bereich zwischen problemspezifisch entwickelten Annäherungsalgorithmen und Brute-Force-Ansätzen
- Daher am besten als ‘Plan B“, geeignet falls andere Optimierungstechniken nicht zielführend anwendbar sind
- Umfasst die Randomisierten Algorithmen (Walks und GA) aus VL 3
- Literatur z. B. „The Essentials of Metaheuristics“ (`https://cs.gmu.edu/~sean/book/metaheuristics/`)


Unimodal


Needle in a Haystack


Noisy
(or “Hilly” or “Rocky”)

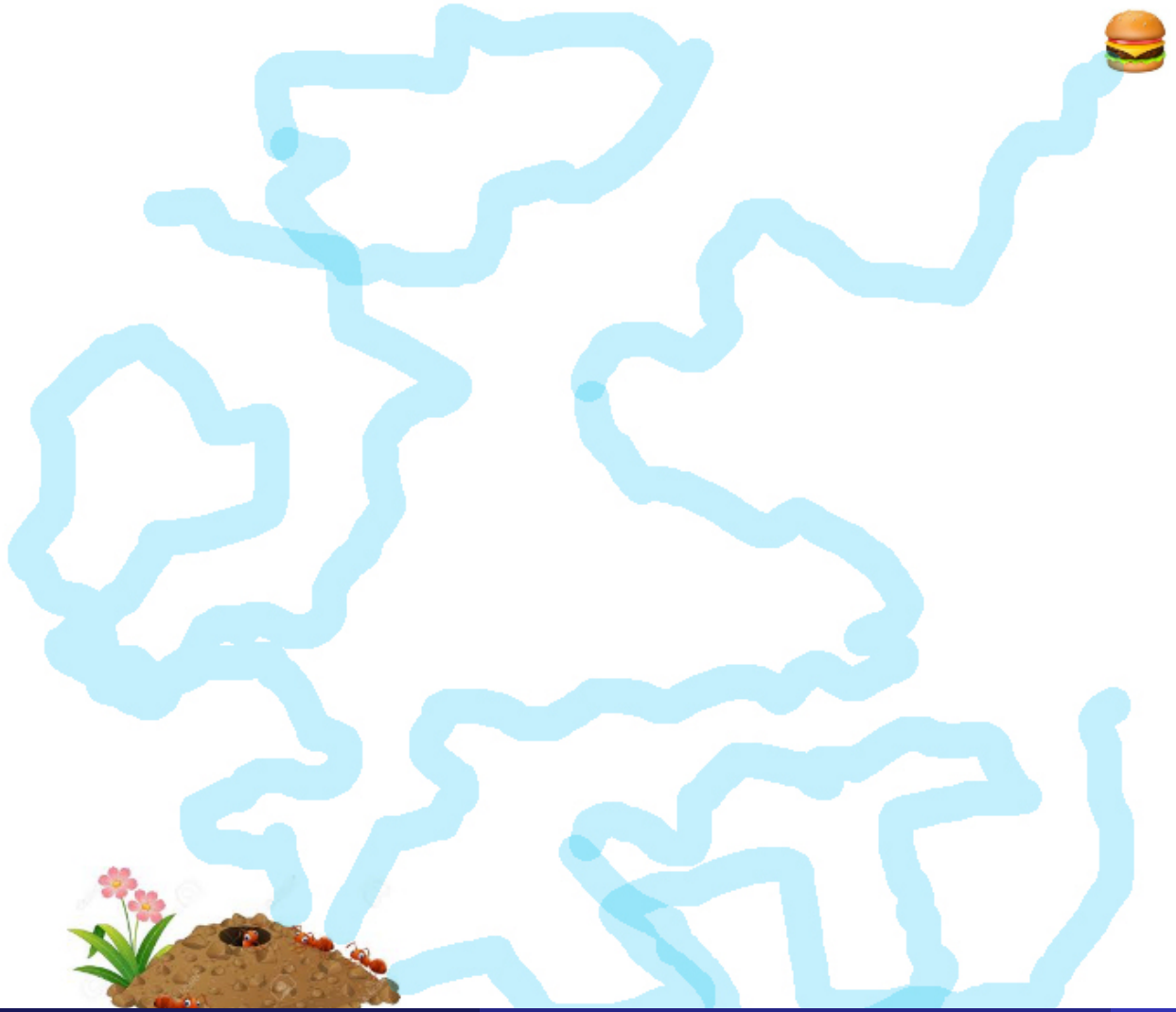

Deceptive

Ameisenalgorithmen

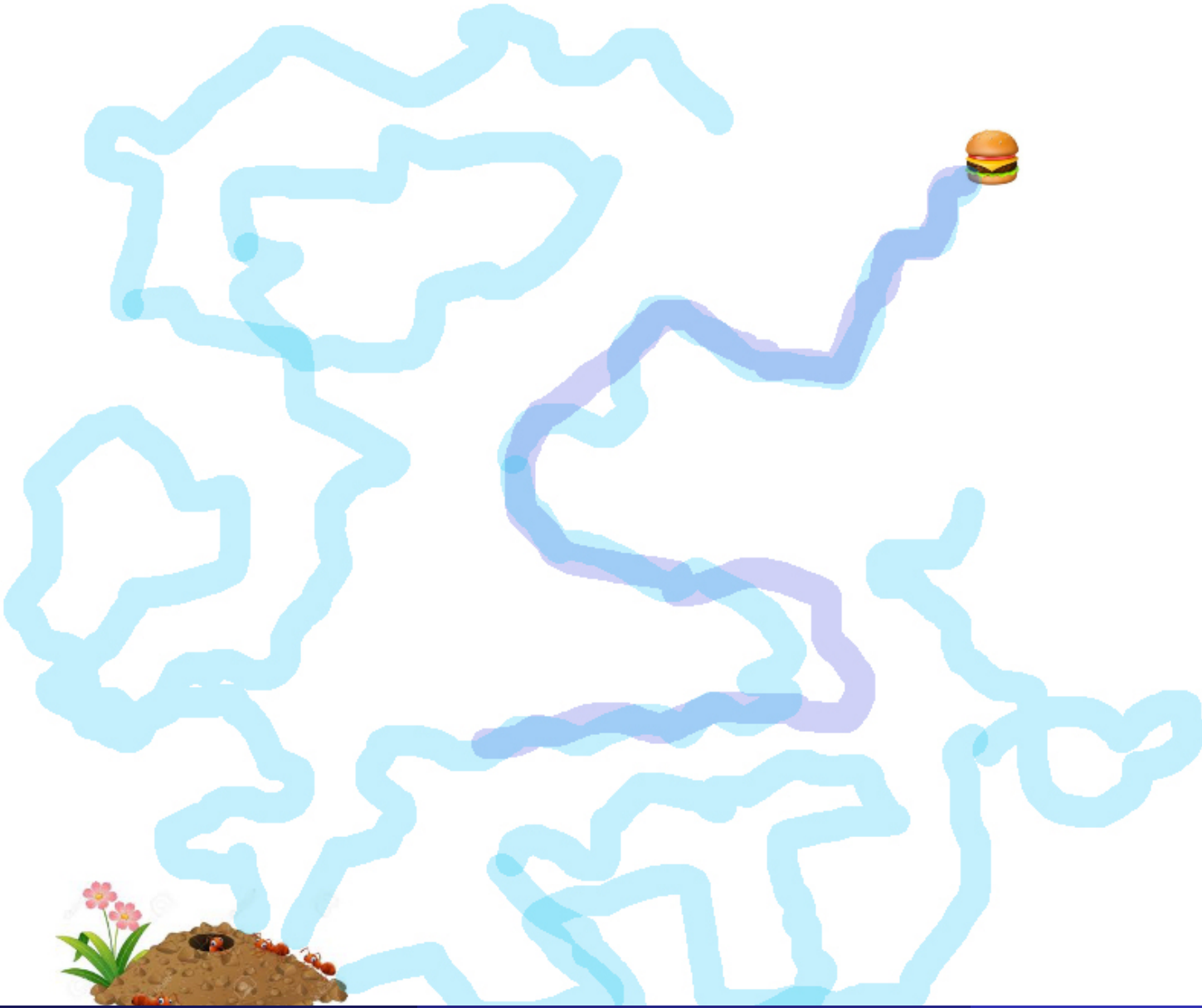
Ameisenalgorithmen

Ameisenalgorithmen

Ameisenalgorithmen

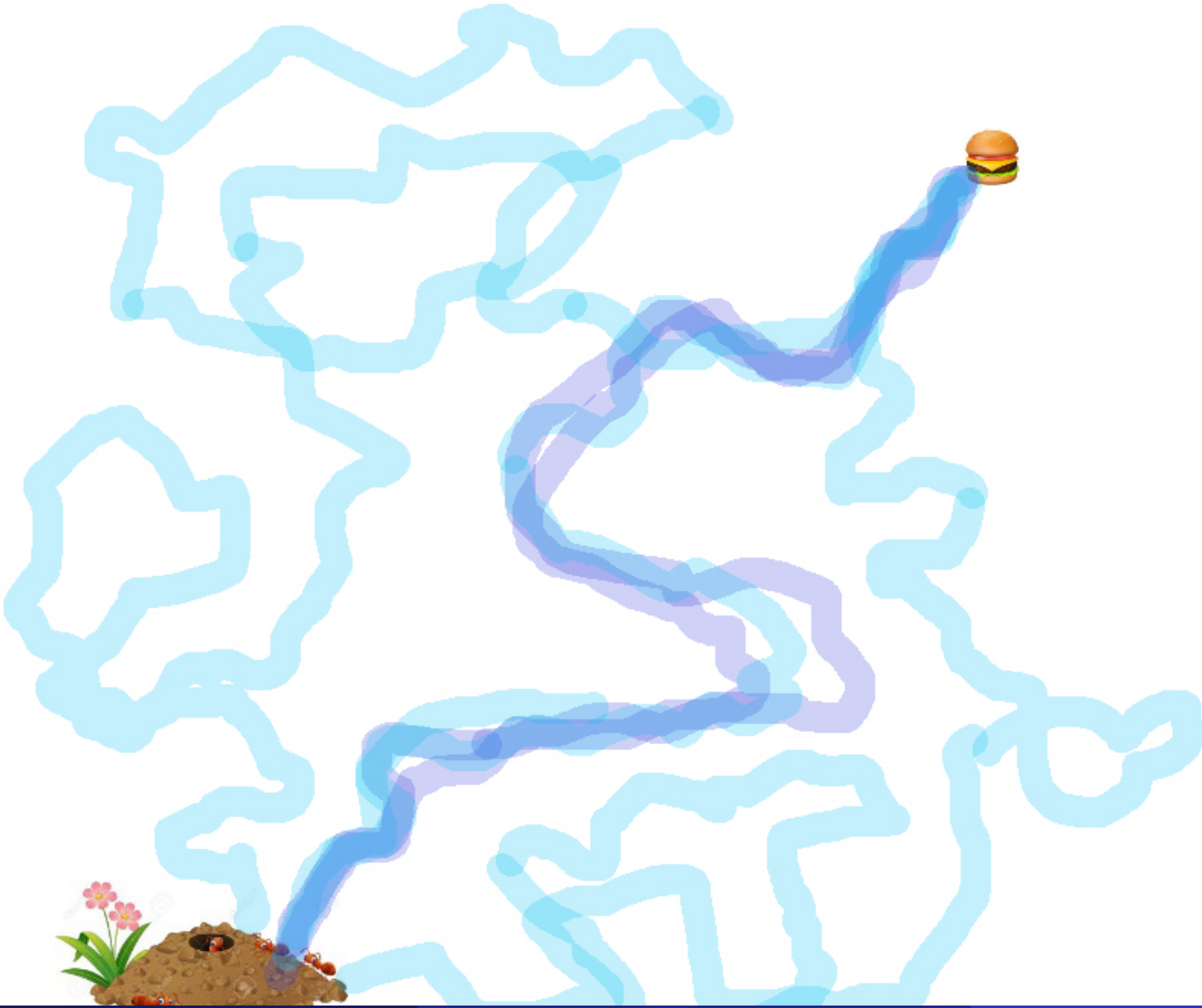


Ameisenalgorithmen



Ameisenalgorithmen

Ameisenalgorithmen



Ameisenalgorithmen

Ameisenalgorithmen

Die Hauptanwendung von Ameisenalgorithmen (*Ant Colony Optimization*) liegt in der Bearbeitung von Graphproblemen, insbesondere dem Finden von Pfaden mit bestimmten Eigenschaften (Shortest Paths, TSP, ...).

Exkurs: Schleimpilze

Weder Tier noch Pflanze, noch Pilz noch Bakterium...

(http://alifexi.alife.org/conference/alife-11)

Video: https://www.youtube.com/watch?v=M93J6y56deM

Particle Swarm Optimization

Particle Swarm Optimization

Particle Swarm Optimization

Idee: Gleichzeitige Evaluierung vieler Lösungskandidaten, diese agieren nicht isoliert, sondern als Gruppe.

Anders als in realen Schwärmen, wo sich Individuen hauptächlich von ihrer direkten Nachbarschaft leiten lassen, kann algorithmisch auf das gemeinschaftliche „Wissen“ aller Individuen zurückggegriffen werden.

Trägheit lässt die Partikel lokale Optima überwinden, längerfristig führt die Information über individuelle und kollektive Bestwerte und -positionen den Schwarm jedoch konvergieren.

Particle Swarm Optimization

Für das Update des Geschwindigkeitsvektors zum aktuellen Zeitpunkt t wird festgelegt auf

$$v_i(t+1) = v_i(t) + (c_1 \times rand() \times (p_i^{best} - p_i(t)) + (c_2 \times rand() \times (p^{gbest} - p_i(t))$$

für jedes $i$te Partikel mit den Gewichtungsfaktoren c_1 (kognitiv) und c_2 (sozial) in Abhängigkeit von den Positionen der Partikel und Optima und einer uniformen Randomisierung aus dem Interval $[0,1]$

Genetische Algorithmen

Idee: Gleichzeitige Evaluierung vieler Lösungskandidaten. Die Besten werden in einem „Pool“ vorgehalten und können sich über Generationen verändern und mit anderen vermischen.

Dabei wird strikt zwischen Phänotyp und Genotyp unterschieden. Für Ersteren kann eine Fitness ermittelt werden, nur letzterer kann direkt manipuliert werden.

Die Art und Weise der Veränderungen (Mutation) und Vermischung (Cross-Over), sowie Evolutionsstrategienn, Poolgröße, Zufallsparameter bestimmen, wie groß der Evolutionsdruck ist.

Genetische Algorithmen

Genetische Algorithmen sind immer dann besonders geeignet, wenn

- der Parameterraum für Lösungen
  - sehr groß ist,
  - nicht analytisch beschreibbar ist,
  - ungültige Konfigurationen besitzen kann oder
  - eine unstetige oder teuer zu evaluierende Kostenfunktion aufweist
- Lösungen mit knappen, im Idealfall seriellen Beschreibungen repräsentiert werden können (DNA)
- „Fähigkeiten“ der Lösungs-Individuen durch wenige einzelne Stellen in der Beschreibung maßgeblich beeinflusst werden (Lokalität und Disjunktion)
- die Kombination oder der Austausch der Fähigkeiten von Lösungsindividuen Aussicht auf Verbesserung der Fitness besitzen

Genetische Algorithmen

„Evolved Virtual Creatures“ – Ein anschauliches Beispiel zur Nachbildung natürlicher Evolutionsprozesse, Karl Sims (1994)

Simulation der Entwicklung von Form und Verhalten virtueller Kreaturen

Fitness-Ziel: Erfüllung von Aufgaben:

- Schwimmen (schnell)
- Laufen (schnell)
- Springen (hoch)
- Folgen (schnell)
- Gegenstand aneignen (nah)

Video: `https://www.youtube.com/watch?v=JBgG_VSP7f8`