

9. Kooperation, verteilte Intelligenz und Netzwerke

- wie entsteht koordiniertes Verhalten mehrerer Organismen?
- Vorbilder: Schwarmtiere, Insektenstaaten, menschliche Gesellschaft
- "Sozionik": Erkundung hypothetischer Gesetzmäßigkeiten, nach denen menschliche Gesellschaften funktionieren, durch agentenbasierte Modelle ("kollektive" Erweiterung von ALife)
- Anwendungen in Ökologie, Soziologie, Ökonomie
- "distributed problem solving" – Anwendung auf konkrete AI-Probleme (z.B. kooperierende Roboter)

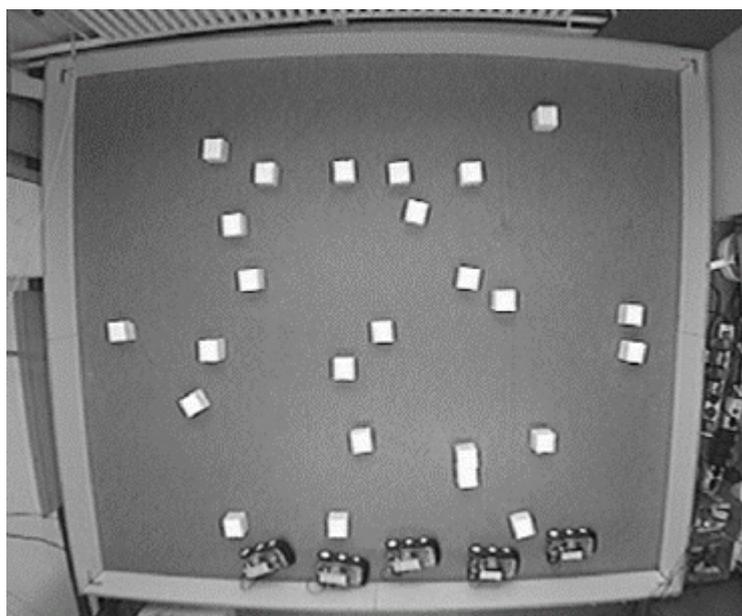
einfachste Form der Entstehung kollektiven Verhaltens:

- Anwendung fester, lokaler Regeln auf die einzelnen Agenten
- Gruppen-Verhalten als emergentes Ergebnis

Beispiel:

"Didabots" (Maris & te Boekhorst 1996, nach Pfeifer et al. 2002)

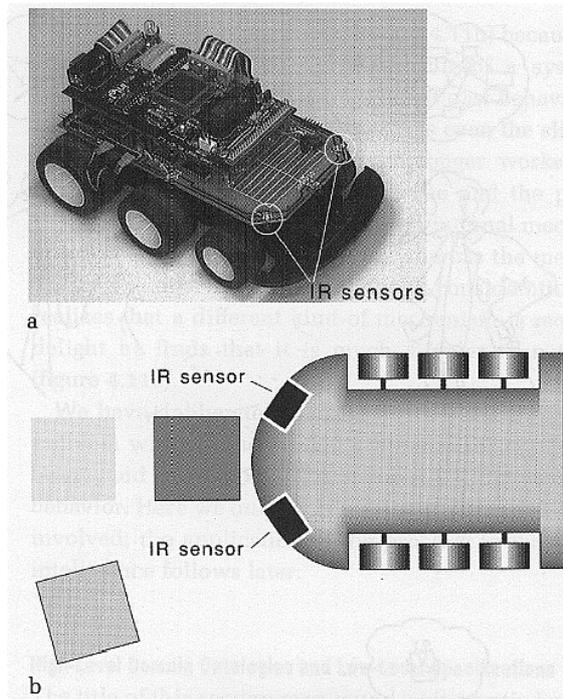
mehrere einfache Roboter in einer Arena mit Klötzchen



die Roboter sind mit 2 Infrarotsensoren ausgestattet ("Sehen" im Nahbereich).

Regeln:

- konstante Vorwärtsbewegung
- wenn Stimulus links, nach rechts drehen
- wenn Stimulus rechts, nach links drehen
- ein (kleines) Klötzchen genau in Bewegungsrichtung wird nicht wahrgenommen und weiter mitgeschoben



Ergebnis:



die Didabots "räumen auf"!

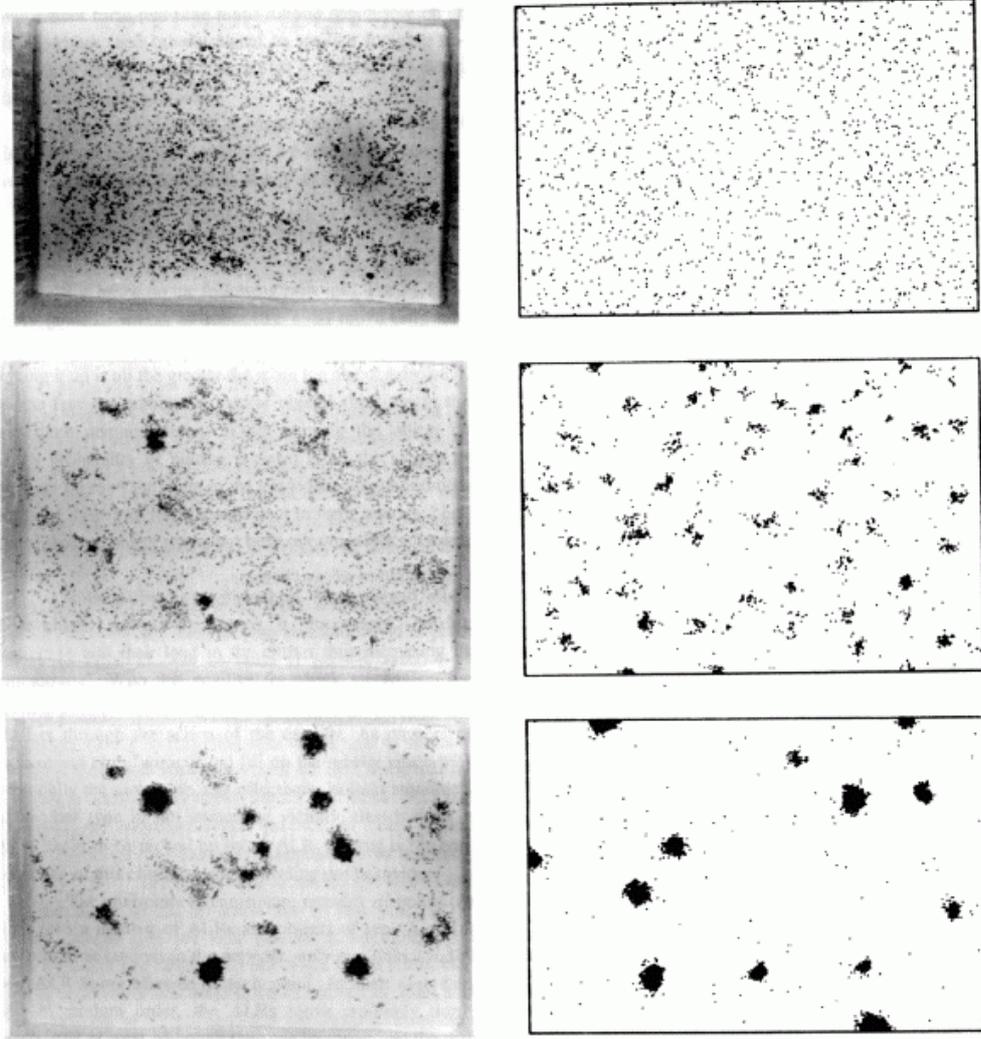
scheinbar komplexes Muster (Haufenbildung) aus einfachen lokalen Regeln

ähnlich bei Ameisen

Steuerung des Verhaltens über Wahrscheinlichkeiten des Aufnehmens und Ablegens von Objekten

- Ameise nimmt Objekt nur wahr, wenn unmittelbar davor
- Wahrscheinlichkeit des Aufnehmens am größten, wenn kein weiteres Objekt derselben Sorte in der Nähe
- für Ablegen umgekehrt

Ergebnisse:

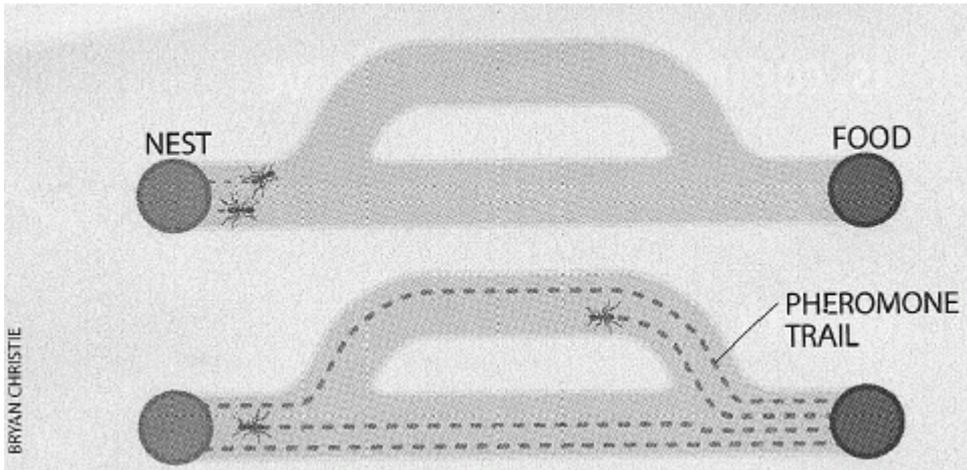


(links: echte Ameisen, rechts: Simulation)

Kommunikation von Ameisen durch "stigmergische Interaktion" (Informationsübertragung vermittelt durch die Umgebung):

Legen von Pheromonspuren

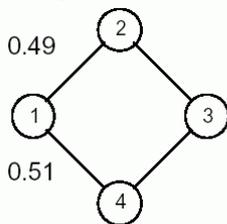
- je intensiver die Pheromonspur, desto größer die Wahrscheinlichkeit, dass weitere Ameisen diesem Pfad folgen
- dadurch Optimierung von Wegen möglich



Anwendung dieses Prinzips in Optimierungsproblemen: "Ant Algorithms", "Ant based control" (u.a. bei Hewlett-Packard und British Telecom für optimale Auslastung von Telefonnetzen)

Updaten einer "Pheromon-Tabelle" für jede Start-Ziel-Kombination in jedem Knoten durch virtuelle Ameisen, die das Netz durchwandern

Beispiel:



dest. node next node	2	4
2	0.95	0.05
3	0.49	0.51
4	0.05	0.95

Update-Formeln: $P_{new} = \frac{P_{old} + \Delta p}{1 + \Delta p}$ für den Knoten von dem die Ameise

gerade kommt, $P_{new} = \frac{P_{old}}{1 + \Delta p}$ für die übrigen

Anwendungen: Traveling Salesman Problem, Job scheduling, shortest common supersequence, Graphenfärbung...

anderes Beispiel: Schwarmverhalten (Vögel, Fische...)

"Boids" (Bird objects), Reynolds 1987

Regeln für Boids

3 Basisprinzipien für Boids:

- (1) *Kollisionsvermeidung*: Ein Boid versucht, Kollisionen zu vermeiden
- (2) *Geschwindigkeit annähern*: Ein Boid versucht, seine Geschwindigkeit der seiner Nachbarn anzupassen
- (3) *Schwarmzentrierung*: Ein Boid versucht, sich stets in der Nähe von anderen Boids aufzuhalten.

Kollisionsvermeidung:

mit anderen Schwarmobjekten und mit "fremden" Szenenobjekten – gewisser Mindestabstand wird während der Bewegung gewährleistet

Strategien zur Kollisionsvermeidung (Reynolds 1988):

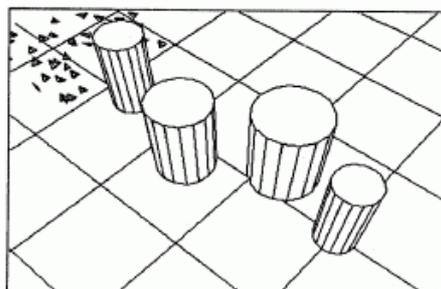
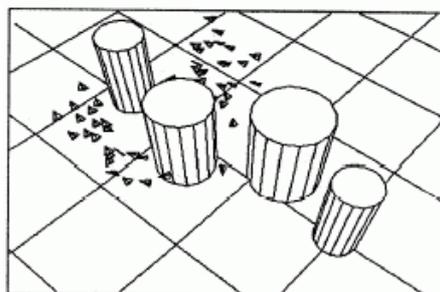
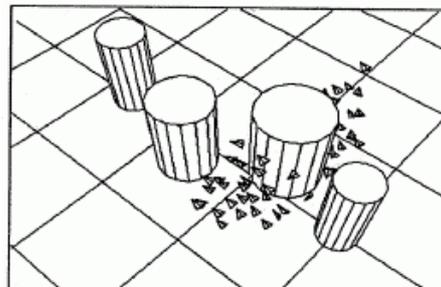
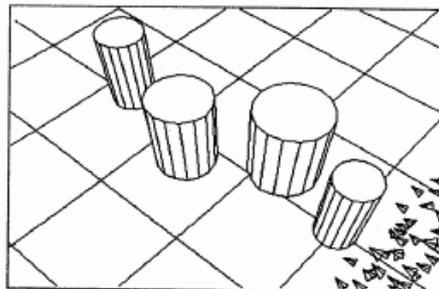
- Wegsteuern von Oberflächen (virtuelles Kraftfeld benutzen), Weg durch Öffnung suchen oder an Oberfläche entlang fliegen
- von Objekt-Mittelpunkten wegsteuern
- virtuelle Fühler ausstrecken, um mögliche Oberflächen direkt auf Kollisionskurs zu entdecken
- Ansteuern der in die eigene Sichtebene projizierten Umrisskanten (bzw. mit sicherem Abstand daran vorbei)

Schwarmverhalten (flock centering):

- jedes Boid will beim Schwarm bleiben
- mittlere Boids bleiben bei ihrer Richtung, äußere drängen nach innen
- wichtig: Wahrnehmungsbereich der Boids (endliche Distanz)
- keine Information über globalen Schwarm-Mittelpunkt,
- keine Gefolgschaft eines bestimmten "Führers"!

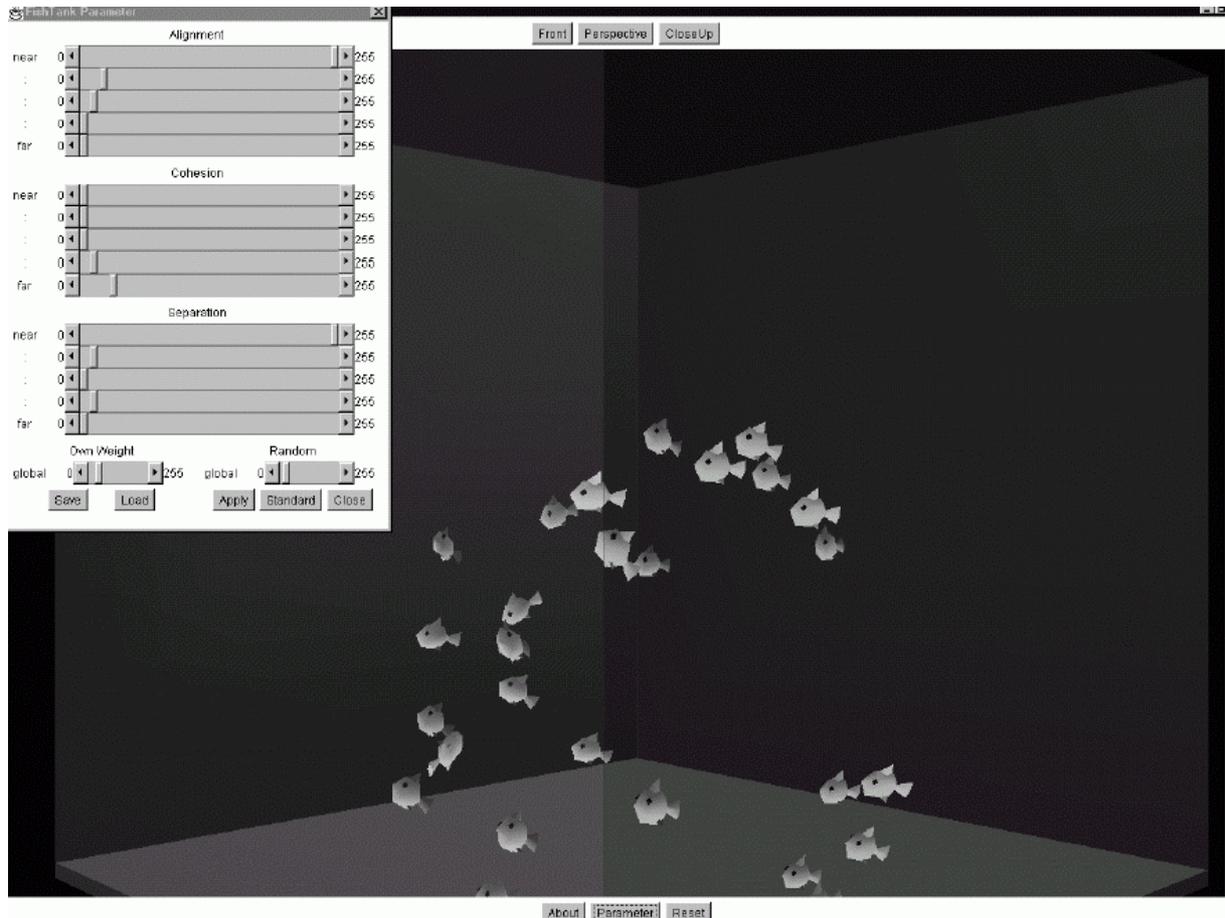
Navigation und Geschwindigkeit:

- Boid hat nur kleine Zahl von "wahrnehmbaren" Nachbarn, die zur Ermittlung der Durchschnittsgeschwindigkeit benutzt werden, an die das Boid sich anpasst
- Navigationsmodul kombiniert die 3 low-level Steuerprinzipien:
Kollisionsvermeidung > Geschwindigkeitsannäherung > Schwarmzentrierung
- Soll-Geschwindigkeit wird benutzt, um Ist-Geschwindigkeit allmählich anzupassen, ebenso für die Richtung: Vermeidung abrupter Änderungen
- je nach den Gewichtungsfaktoren im Navigationsmodul können unterschiedliche Verhaltensweisen modelliert werden



Komplikation: Einführung von Raubvögeln
– zusätzliche Verhaltensregeln (Fernhalten von den Raubvögeln, Vermeiden ihrer Flugrichtung)

interaktive Systeme mit Möglichkeit der Regelung der Verhaltenspräferenz-Parameter



Um komplexere "künstliche Gesellschaften" zu generieren:

- Verwendung komplexerer Agenten (mehr Regeln, interne Zustände)
- strukturierte Umwelt (die ebenfalls durch Regeln beschrieben wird)
- Lernfähigkeit (vgl. Kap. 4)
- experimentelle Simulationsumgebung, die das Testen diverser Regelsysteme ermöglicht

Beispiel: "**Sugarscape**" (Epstein & Axtell 1996)

Anspruch: Virtuelles Labor für die Sozialwissenschaften

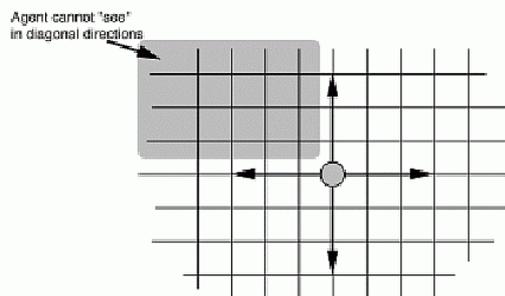
- Modell-Universum, dessen Regeln nach Bedarf verfeinert und ergänzt werden können

Die Umwelt:

- 2-dimensionales Gitter (50×50), Torus-Topologie
- dort wächst "Zucker" (in diskreten Mengen: 0 bis 4)
- in jeder Gitterzelle festgelegt: aktuelle Zucker-Menge und maximale Kapazität
- Zucker-Regenerationsregel: Zucker-Menge nimmt in jeder Gitterzelle linear zu mit Rate α pro Zeitschritt, bis max. Kapazität erreicht

grundlegende Regeln für die Agenten:

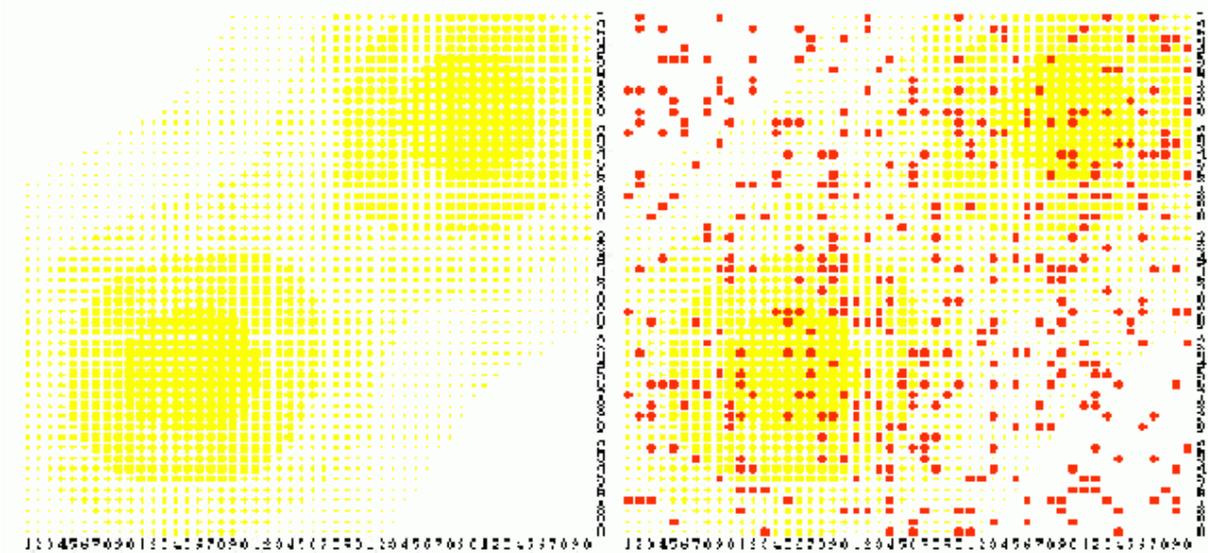
- Metabolismus: pro Zeiteinheit wird feste Menge Zucker verbraucht
- Agent stirbt, wenn kein Zucker mehr verfügbar
- Zuckervorrat (potenziell unbegrenzt) wird mitgeführt (Startvorrat > 0)
- endliches Blickfeld, Blick nur in die 4 Haupt-Himmelsrichtungen, Weite: für jeden Agenten zufällig bestimmt (zwischen 1 und 6 Felder)



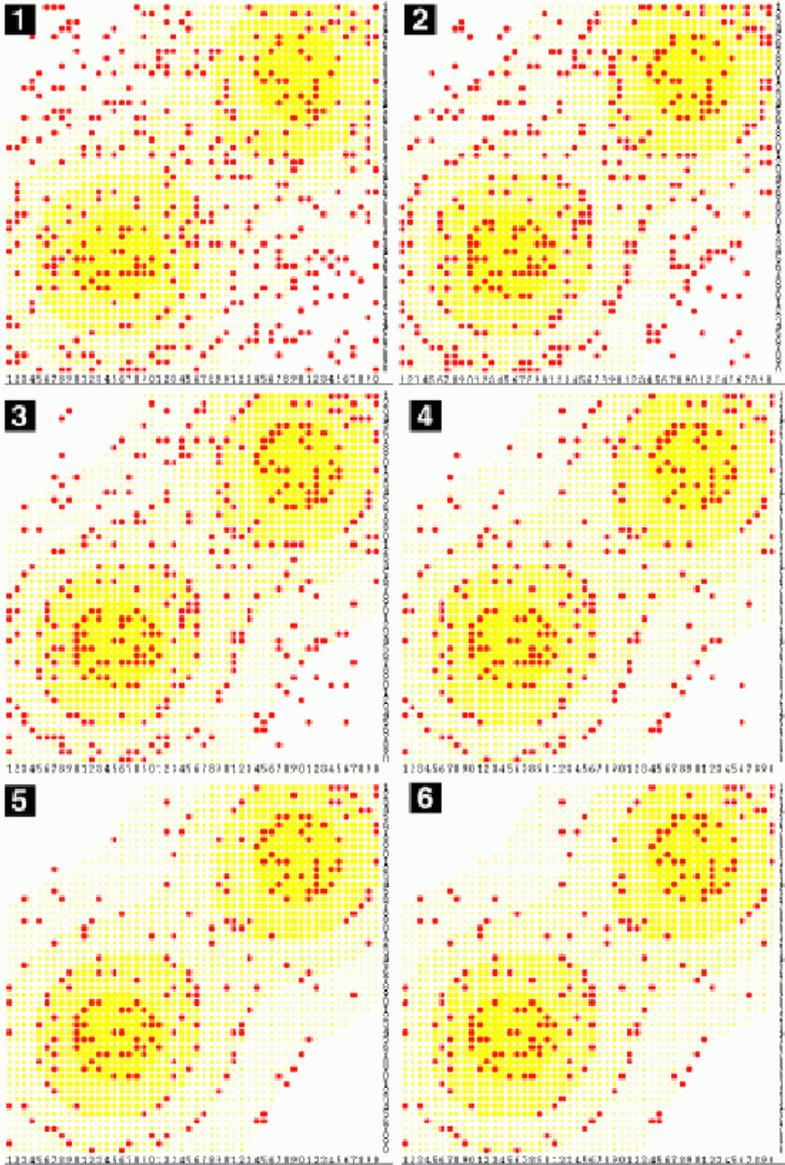
Bewegungsregel für Agenten:

- blicke soweit möglich in die 4 Blickrichtungen und suche die unbesetzten Gitterzellen mit dem meisten Zucker
- suche davon die nächste
- bewege dich dorthin
- "ernte" allen Zucker auf diesem neuen Gitterfeld

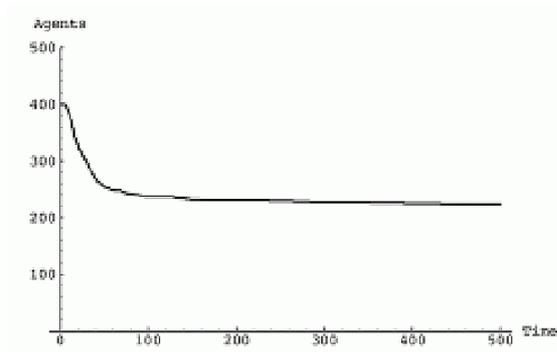
initiale Zuckerverteilung (links); initiale Verteilung der Agenten (rechts):



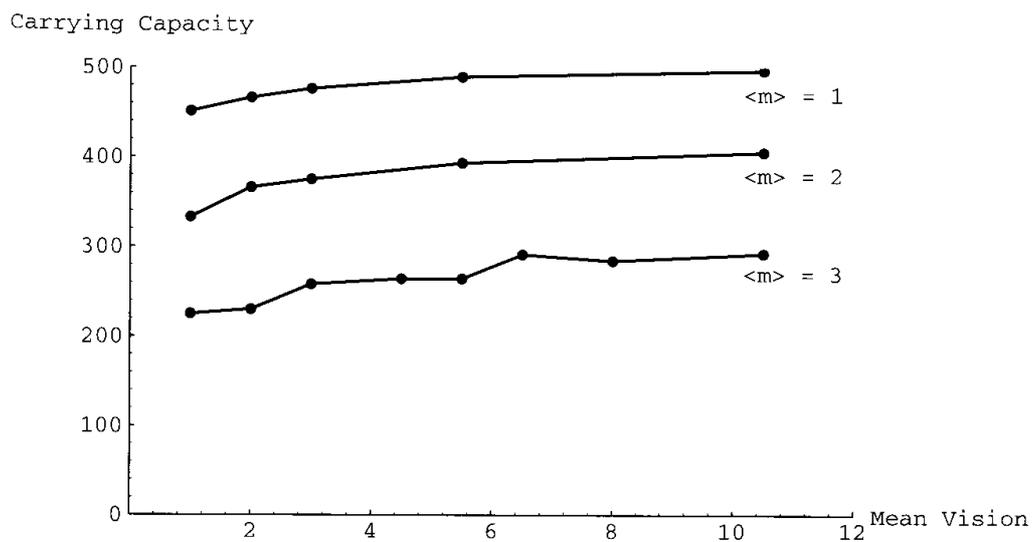
Entwicklung des Sugarscape aufgrund der obigen Regeln:



Entwicklung der Populationsgröße:



⇒ Konvergenz gegen eine "carrying capacity", die u.a. von der Wachsrate α des Zuckers und von der metabolischen Umsatzrate der Agenten abhängt – aber auch z.B. von der mittleren Sichtweite der Agenten:



Um Verteilung der Zuckervorräte ("Reichtum") zu studieren:
unendliches Akkumulieren von Zucker ist nicht realistisch

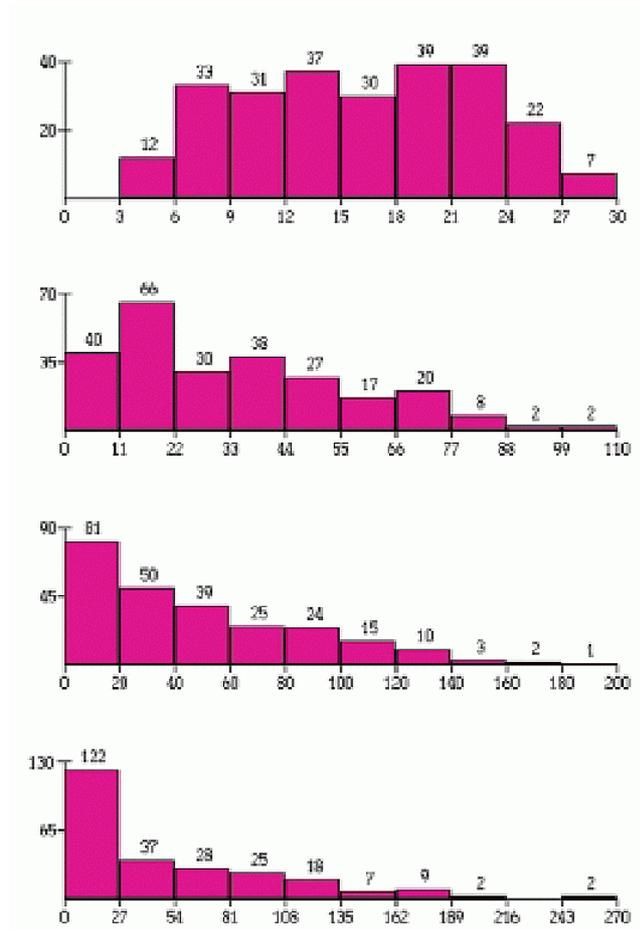
Modifikation der Regeln:

Einbau eines maximalen Lebensalters (zufällig gewählt aus Intervall $[a, b]$ für jeden Agenten); bei Tod eines Agenten wird ein neuer geboren mit zufälliger Position u. zuf. Zuckervorrat

es entwickelt sich "ökonomische Ungleichheit" als emergente Struktur:

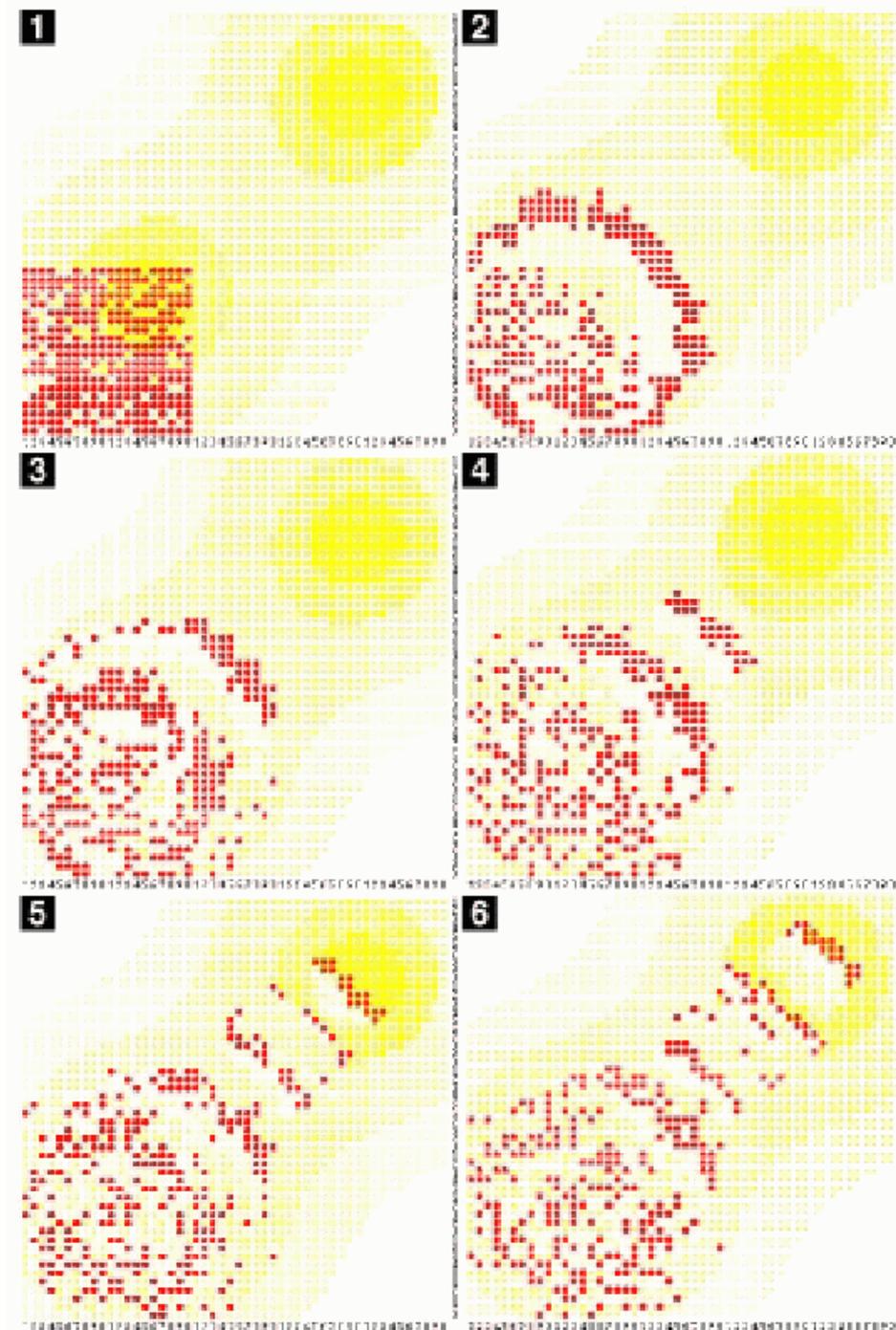
(im Beispiel ist $[a, b] = [60; 100]$; Anzahl Agenten = 250)

wealth histograms

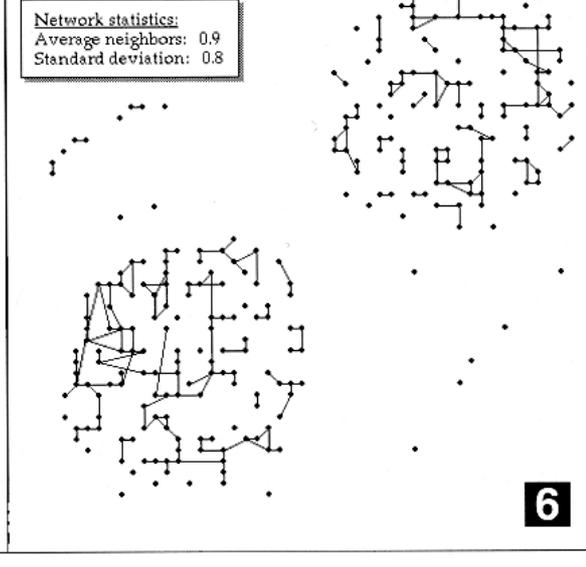
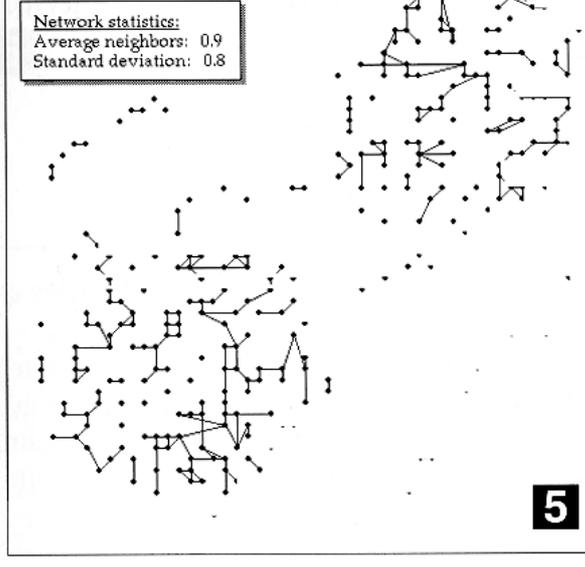
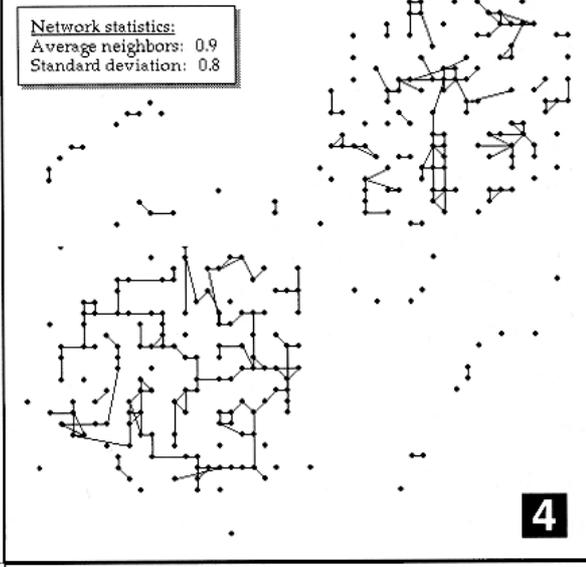
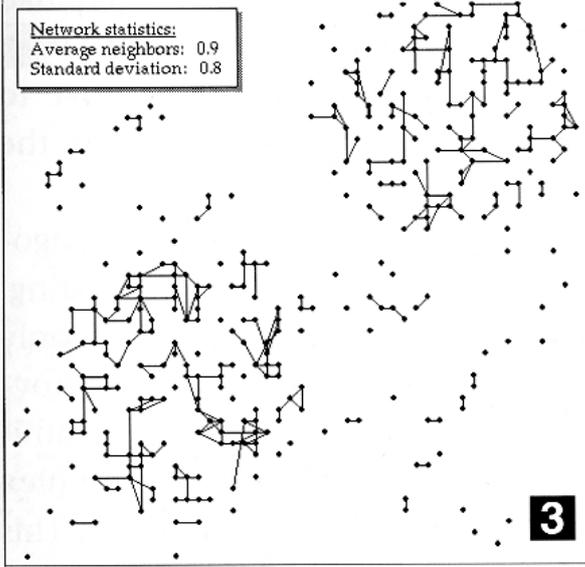
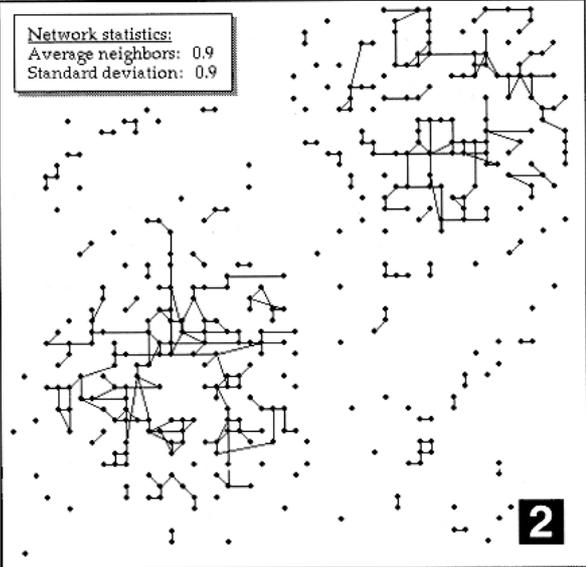
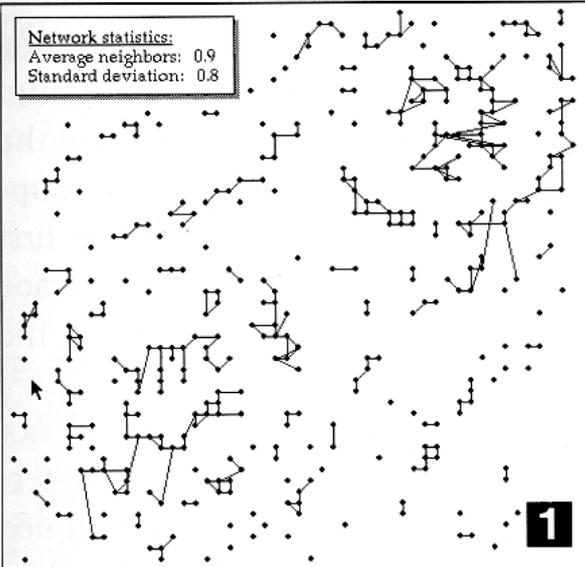


weitere emergente Strukturen:

- Migrationswellen
- diagonale Bewegungsrichtung der Population (einzelne Agenten können sich nicht diagonal bewegen!)



Aufbau von "Bekanntschafts-Strukturen" (Netzwerke früherer Nachbarn):



Einführung von sexueller Vermehrung und Vererbung:

- 2 Geschlechter von Agenten (zufällig verteilt)
- "Fertilität": Agent muss Altersgrenze überschritten und genug Zucker gesammelt haben
- Fortpflanzungsregel ("Agent sex rule"):
 - wenn fruchtbar, suche zufällig einen Nachbarn aus
 - wenn dieser auch fruchtbar *und* vom anderen Geschlecht *und* einer von beiden hat eine leere Nachbarzelle, dann mache ein Kind und positioniere dies in der leeren Nachbarzelle
 - dem Kind wird von beiden Eltern ein Zuckerbetrag mitgegeben
 - iteriere dies für alle Nachbarn.

Weitergabe genetischer Attribute durch Mendelsche Vererbung:

z.B. Sichtvermögen v und Metabolismus-Rate m

(m, v) kreuzt sich mit (M, V) :

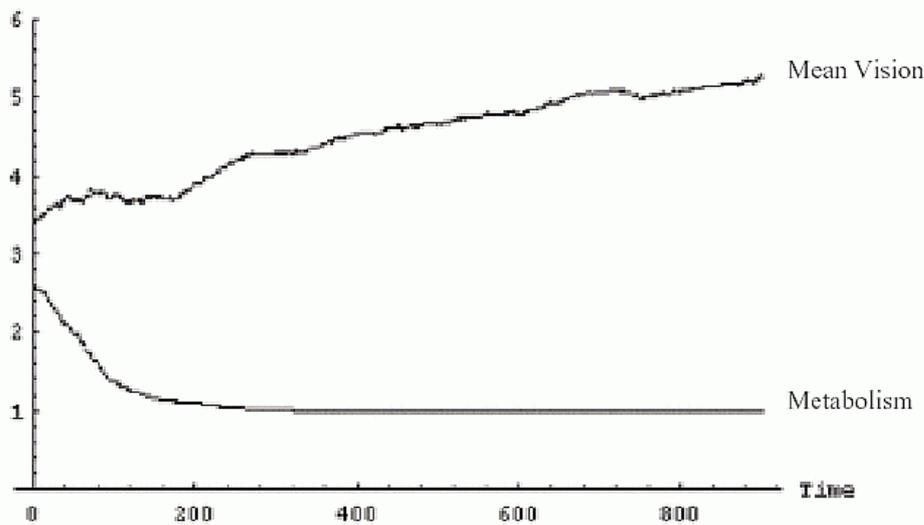
	<i>Metabolism</i>	
<i>Vision</i>	m	M
v	(m,v)	(M,v)
V	(m,V)	(M,V)

Zufallsauswahl aus den 4 Alternativen

⇒ genetischer Algorithmus

Selektion führt zu Veränderung der Häufigkeit der genetischen Merkmale in der Population

Entwicklung von mittl. Sichtweite und metabolischer Rate unter dieser Regel:



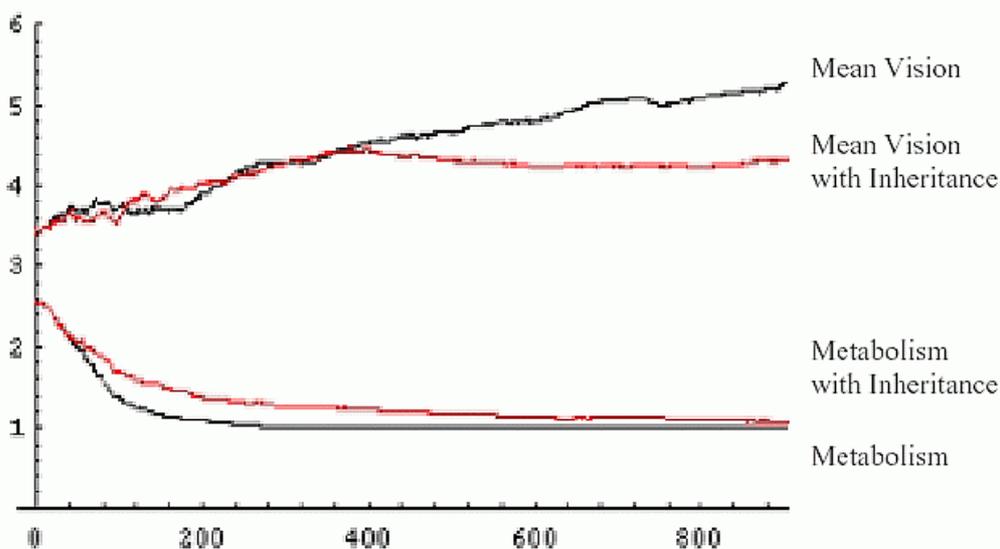
weitere Modifikation:

auch "Güter" (Zuckervorräte) können "vererbt" werden
(keine biologische, sondern kulturelle Regel!)

Vererbungsregel (*inheritance rule*):

Wenn ein Agent stirbt, wird sein Zuckervorrat zu gleichen Teilen unter all seinen lebenden Kindern aufgeteilt

Auswirkungen:



⇒ Selektionsdruck auf die Sichtweite wird offenbar durch die kulturelle Regel vermindert!

Einführung weiterer kultureller Merkmale:
codiert durch Bitstring s fester Länge
jedes Bit entspricht einem Merkmal

kulturelle Transmissionsregel ("tag-flipping"):

für jeden Nachbarn:

- eine Bitposition i wird zufällig ausgesucht
- wenn $\text{Nachbar.s}[i] \neq \text{selbst.s}[i]$, ändere nichts, sonst:
Nachbar.s[i] wird auf selbst.s[i] gesetzt

(sexuelle Vermehrung bei diesem Versuch ausgeschaltet)

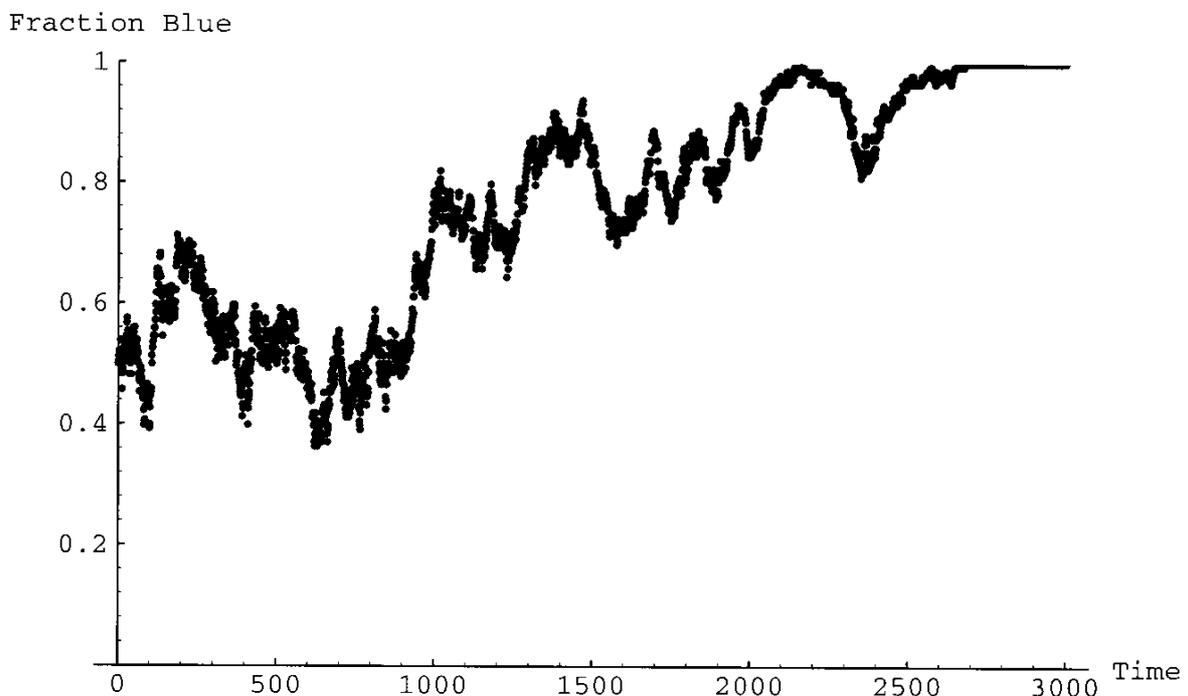
Folge:

mit der Zeit gleichen sich die Bitstrings immer mehr an

Gruppenmitgliedschafts-Definition: Agenten gehören "zu den Blauen", wenn sie mehr Nullen als Einsen in ihrem Bitstring haben, sonst "zu den Roten" (willkürliche Setzung)

Entwicklung des Anteils der "Blauen":

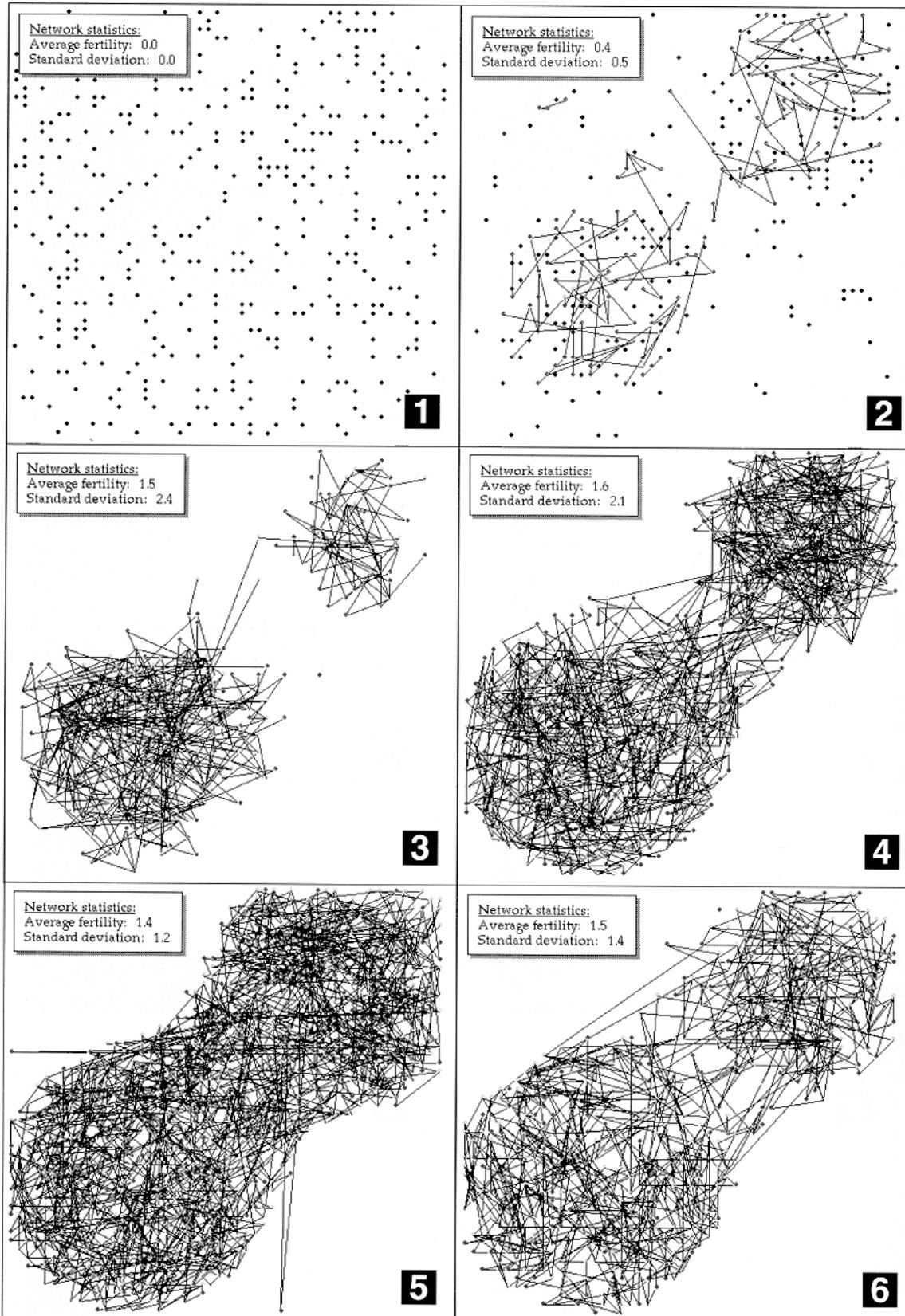
(zufälliger Ausgang – Symmetriebrechung)



Kombination mit sexueller Vermehrung erlaubt Unterscheidung von "horizontaler" und "vertikaler" (d.h. Eltern an Kinder) kultureller Transmission

weitere Studiengebiete im Sugarscape
(näheres bei Epstein & Axtell 1996):

- Formierung genealogischer Netzwerke



- Freundschaftsnetzwerke ("Freunde": Nachbarn mit ähnlichen Bitstrings kultureller Merkmale)
- Kampf (um Zucker) gegen Mitglieder anderer "Stämme"
- Handel (jetzt 2 essenzielle, nachwachsende Güter: "sugar and spice", die getauscht werden können)
- Marktentwicklung, Kreditverteilung
- Auswirkung wechselnder kultureller Präferenzen für Güter
- Handelsnetzwerke
- Auswirkung von Umweltverschmutzung ("pollution" entsteht durch Ernten und/oder durch Konsumption von Zucker)
- Seuchenausbreitung
- Immunsysteme

usw. usw. ...

Phänomen in (echten oder simulierten) sozialen Netzwerken (und in anderen großen Graphen, die aus Selbstorganisationsprozessen hervorgegangen sind – z.B. Internet; Elektrizitätsnetzwerke; Nahrungsnetze...):

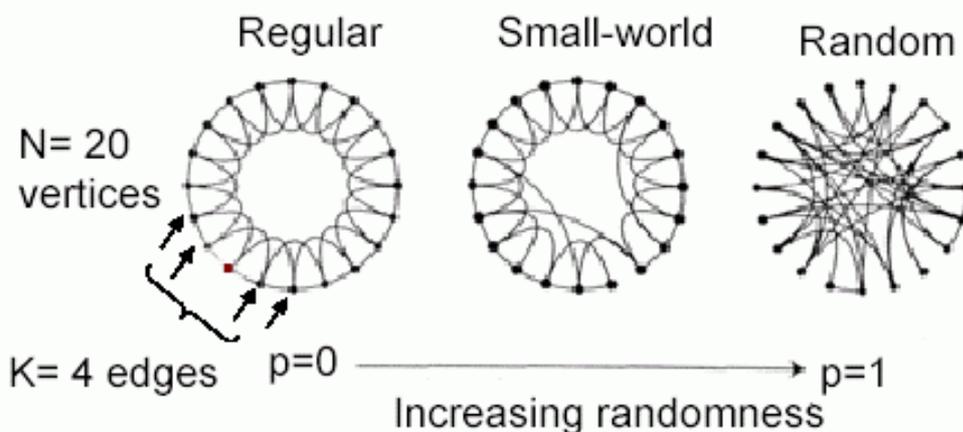
"Small World Effect"

bezieht sich (zunächst) auf kleine graphentheoretische Durchmesser:

- "six degrees of separation"
- Verknüpfung von 6 Bekanntschaften verbindet jeden mit jedem auf der Welt
- Experiment von Milgram 1967
- "die Welt ist doch klein!"
- dabei sind besonders die "schwachen Bindungen" wichtig ("The strength of weak ties"; Granovetter 1973)

graphentheor. Untersuchung von Watts & Strogatz (1998):

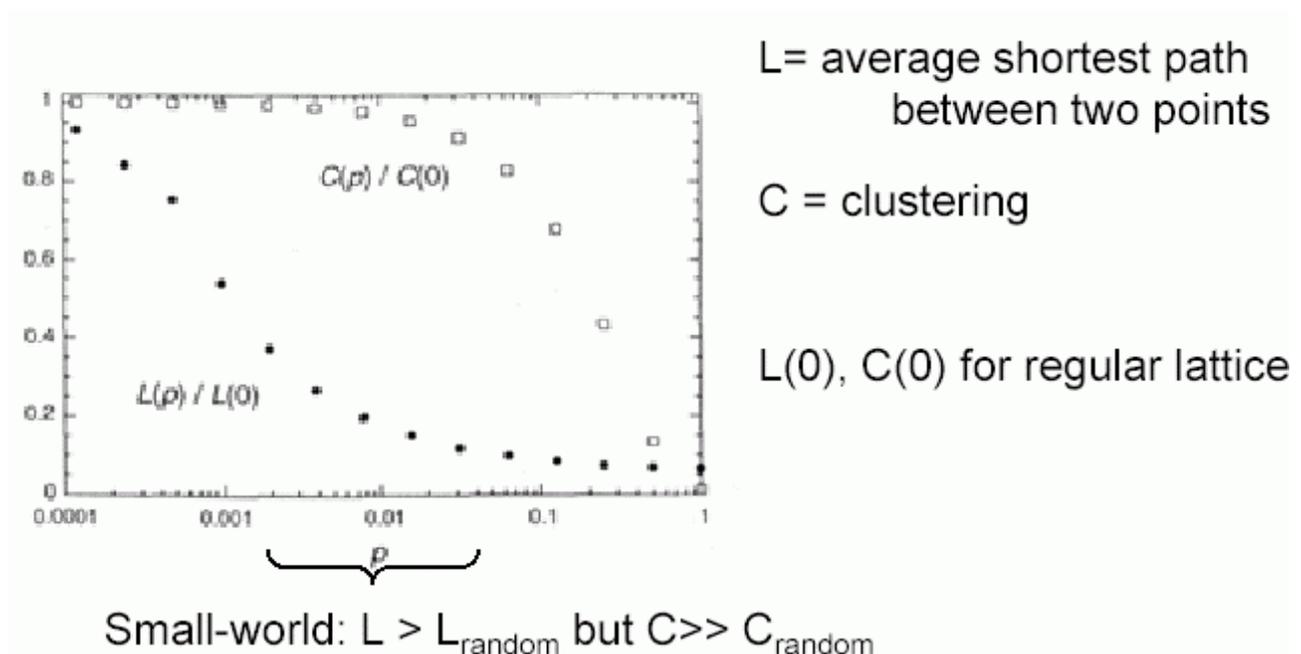
Zufallsgraphen, auf ringförmiger Knotenmenge definiert, einstellbarer Parameter p bestimmt, wieviele der zunächst regelmäßig zwischen eng benachbarten Knoten verlaufenden Kanten gelöst und zufällig neu verlegt werden



interessante Maße:

- *mittlere Pfadlänge* $L(p)$ (Länge des kürzesten Weges zwischen 2 Agenten im Netzwerk)
- *Clustering-Koeffizient* $C(p)$: misst, wieviele Nachbarn eines Agenten ihrerseits untereinander verbunden sind

Small World Effect:



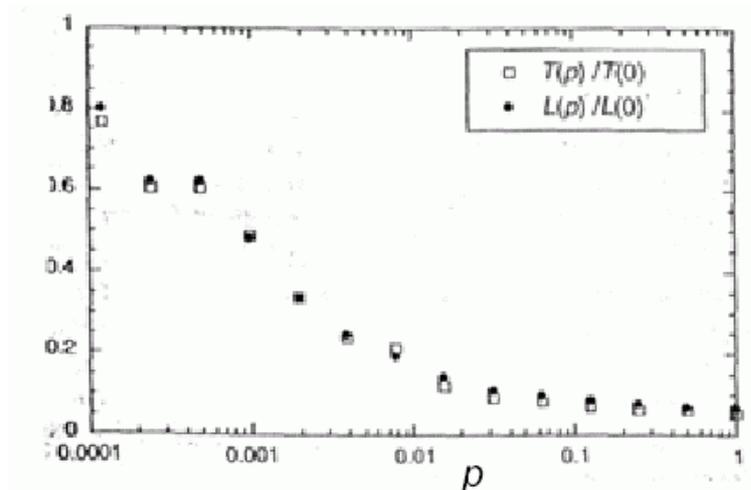
Messungen auch an realen Netzwerken

Beispiel: Netzwerk von Filmschauspielern – verbunden, wenn sie in einem Film zusammen gespielt haben

$$L = 3,6 > L_{\text{random}} = 3,0;$$

$$C = 0.79 \gg C_{\text{random}} = 0,00027 !$$

Auswirkungen z.B. auf die Seuchenausbreitung:



$T(p)$ = time required for global infection.

⇒ Gefahr der globalen Infektion deutlich größer in "Small World Netzwerken" als in reinen Zufallsnetzwerken