

11. Evolution

Jean Baptiste de Lamarck (1744-1829)

- prägte den Begriff "Biologie"
- biologische Arten sind über lange Zeiträume veränderlich: Theorie einer allmählichen "Evolution" (Entwicklung)
- die Veränderungen haben mit funktionaler Anpassung zu tun
- die Veränderungen hängen ab von der Intensität des Gebrauchs bestimmter Organe (Bsp. Giraffenhals) – Weitergabe erworbener Eigenschaften

die letztgenannte Annahme wurde von späteren Evolutionstheorien verworfen

(kann aber in künstlichen Systemen realisiert werden: Systeme mit "Lamarckscher Evolution")

Charles Darwin (1809-1882)

London 1859: "The Origin of Species"

(Alfred R. Wallace war unabhängig zu ganz ähnlichen Ergebnissen gekommen)

- biologische Arten durch generationenübergreifenden Entwicklungsprozess aus anderen biologischen Arten entstanden
- die Veränderung steht in Zusammenhang mit der Anpassung (*adaptation*) der Individuen einer Art an die Erfordernisse ihrer Umwelt
- Darwin: "This principle of preservation or the *survival of the fittest*, I have called Natural Selection" (natürliche Selektion / Auslese)
- Paradigmenwechsel der Biologie: von beschreibender und ordnender zu erklärender und nach Ursachen forschender Wissenschaft

beachte: Darwin wusste nichts über Genetik, Mutationen oder DNA!

Was stellt das "Prinzip der natürlichen Auslese" dar?

- eine Tautologie (es überleben die, die am besten überleben können)?
 - einen Erfahrungssatz über eine Eigengesetzlichkeit speziell der *lebenden* Systeme, wie wir sie kennen?
 - eine Gesetzmäßigkeit, die lediglich bei Lebewesen bevorzugt in Erscheinung tritt, aber nicht auf diese beschränkt ist?
- Darwin hat sich nicht klar geäußert
 - heute vertritt man die 3. Position: Darwinsches Selektions- und Evolutionsverhalten lässt sich im Labor und im Computer reproduzieren

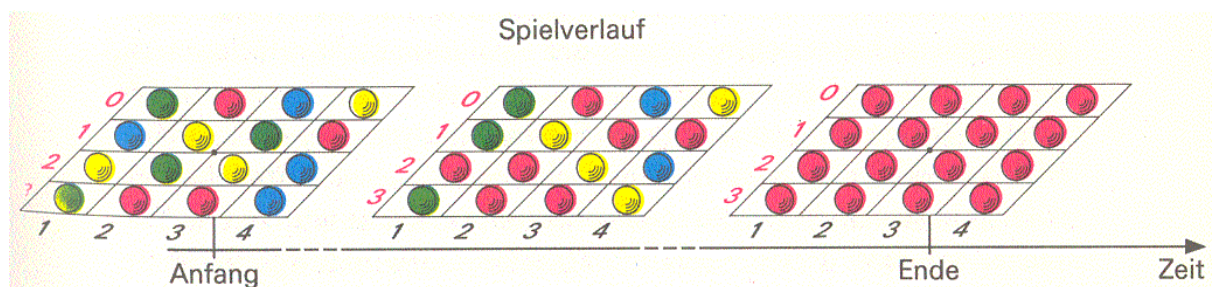
Aber: die Bedingungen

- für Auftreten von Selektion
- für den Zusammenhang von Anpassung und Selektion sind nicht trivial

Beispiel:

Kugelspiel "Selektion" (nach Eigen & Winkler 1979)

- gespielt wird auf einem endlichen quadratischen Gitter
- auf jedes Gitterfeld passt genau eine Kugel
- es gibt eine endliche Zahl von Kugelfarben (z.B. 4)
- zu Beginn sind alle Farben in gleicher Menge auf dem Gitter vertreten und regellos verteilt; alle Felder sind besetzt
- dann werden – strikt abwechselnd – die beiden folgenden Regeln angewandt:
- es wird zufällig eine Position im Gitter bestimmt; die dort befindliche Kugel wird entfernt
- es wird eine weitere Position bestimmt; die dort befindliche Kugel wird verdoppelt, ihr Duplikat kommt auf den leeren Platz vom Schritt vorher.



Ergebnis: es kommt immer zur Selektion *einer* Kugelfarbe, die schließlich das gesamte Gitter füllt.

⇒ in diesem Fall wäre der Begriff "fittest" für die selektierte Kugelfarbe tatsächlich tautologisch (es hängt nur vom Zufall ab, welche Farbe "gewinnt" – kein Einfluss von Umwelt und Anpassung!)

Variante des Kugelspiels: wenn man bei jedem "Reproduktionsschritt" mit gewisser Wahrscheinlichkeit eine "Mutation" (zufällige Änderung der Farbe) zulässt, findet *keine* eindeutige Selektion mehr statt!

Merke:

- nicht immer hängt Selektion mit Anpassung zusammen
- nicht immer begünstigt das Vorkommen von Mutationen die Selektion

zu Darwins Zeit Unklarheit über die Vererbung

- Annahme einer Vermischung der (quantitativen) Eigenschaften bei der Vererbung (*blending inheritance*)
- dann müssten sich mit der Zeit alle Unterschiede nivellieren

Neodarwinismus (Beginn 20. Jh.):

- Integration der Mendelschen Genetik (Vorliegen diskreter Allele, Dominanz etc.), der Zellbiologie und Embryologie in die Darwinsche Theorie
- *Weissmann-Prinzip*: der Genotyp wirkt auf den Phänotyp, nicht umgekehrt (Ausschluss Lamarck'scher Vererbung)

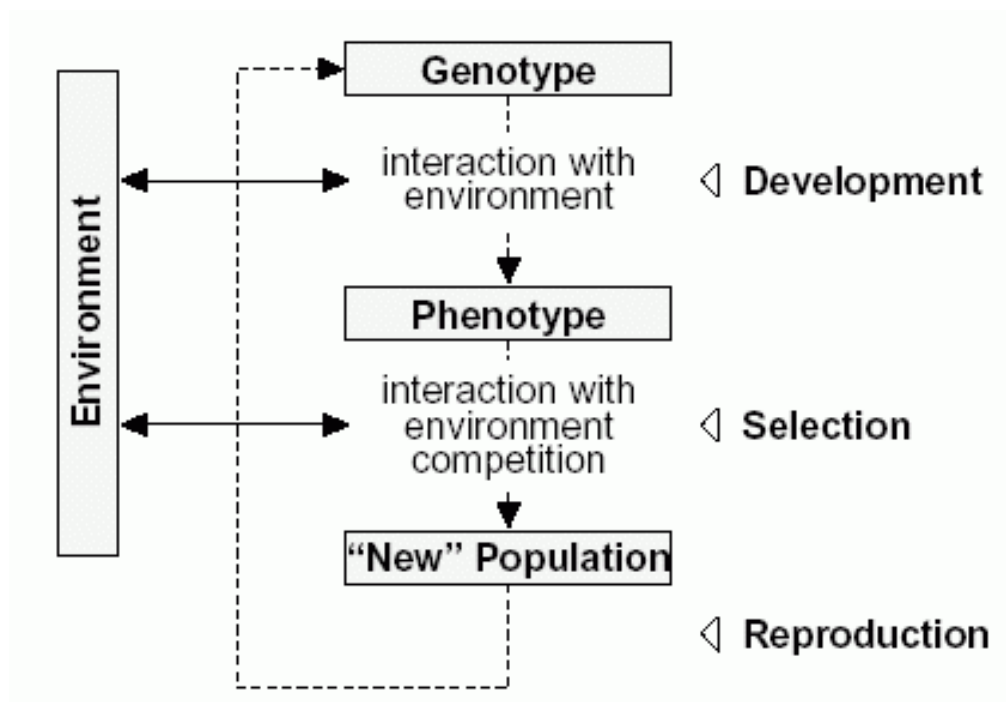
"Synthetische Theorie" (bis heute):

- Integration von Populations- und Molekulargenetik
- Erkennen der Rolle von Mutationen und crossing over
- Evolution wird definiert als die Änderung der Häufigkeit von Genen innerhalb des "Genpools" einer Population von Individuen
- John B. S. Haldane, Ronald A. Fisher, Sewall Wright: exakte Herleitung der Wirkung natürlicher Selektion mit math. und populationsgenetischen Mitteln

Angenommen, ein einzelnes Gen in einer Population hat zwei Allele A und a , und A verschafft seinem Träger einen geringfügigen Selektionsvorteil. Kann die auf die Population einwirkende natürliche Selektion dann die Häufigkeit des Allels A erhöhen? – Ja (Haldane, Fisher, Wright)

d.h.: Darwinsche Evolution "kann funktionieren".

Der Prozess der Evolution im Verständnis der "synthetischen Theorie" (nach Pfeifer et al. 2002):



Aber: vieles bleibt unklar!

Stuart Kauffman:

"Fast 140 Jahre nach Darwins zukunftsweisendem Buch kennen wir immer noch nicht die Leistungsfähigkeit und die Grenzen der natürlichen Selektion, wissen immer noch nicht, welche Arten komplexer Systeme durch einen Evolutionsprozess zusammengebaut werden können..."

(Kauffman 1995)

Nichttriviale Fragen:

- unter welchen Bedingungen führt Darwinsche Evolution zu "verbesserten" (optimalen?) Lösungen?
- was ist unter der selektierenden "Umwelt" zu verstehen? (für eine Zelle innerhalb eines Organismus etwas anderes als für den Organismus selbst...) – mehrere Skalenebenen
- welche Rolle spielen selektionsneutrale Änderungen (*genetische Drift*)? Nach der Theorie von Motoo Kimura eine sehr wichtige...
- "wieviel" kann in der Evolution auf einmal verändert werden? Ist die Evolution stetig oder schubweise ("Gradualismus" versus "Punktualismus")? Macht die Evolution "Sprünge" ("Saltationismus")?
- wie "zufällig" sind Mutationen? Es ist bekannt, dass sich bestimmte Teile des Genoms häufiger verändern als andere, und dass es Schutzmechanismen gegen Mutationen gibt – "Evolution der Mutationsraten"? Steuert die Evolution ihren eigenen "Motor"?
- was ist die "Zielscheibe" der Evolution (das Angriffsstück der Veränderung)?
 - das einzelne Gen
 - das Genom eines Individuums
 - der Genpool einer Population
 - eine biologische Art
 - ein ganzes Ökosystem
 - die ganze Erde als "Super-Organismus"

die beiden Extrempositionen sind unter den Bezeichnungen "das egoistische Gen" (*The selfish gene*; R. Dawkins) und "Gaia-Hypothese" (J. Lovelock) bekannt geworden.

Richard Dawkins: "Das egoistische Gen" (1976, dt. 1978):
die Theorie legt nahe, dass die Evolution primär auf Gene wirkt

– dies äußert sich u.a. in Verhaltensphänomenen wie dem Töten von Löwenjungen durch ein Männchen, das ein Löwenrudel als neues Alpha-Männchen übernommen hat (so wird die maximale Ausbreitung der eigenen Gene sichergestellt – entgegen den "Interessen" der Population oder der Art)

aber wie kommt dann "altruistisches Verhalten" zustande, von dem es auch Beispiele im Tierreich gibt?

- Evolution von kooperativen Strategien (Axelrod et al.)

Die *selfish gene* - Hypothese bietet hier Erklärungen an

→ Soziobiologie

→ der Organismus (auch der Mensch) ist im hohen Maße "Sklave seiner Gene", ja, lediglich deren "Verbreitungsmaschine"

aber: es gibt auch Kritik an dieser einseitigen Interpretation (vgl. Hemminger 1994):

- Gene steuern sich gegenseitig und bilden Netzwerke
- es gibt in der Natur auch kooperative Systeme, deren Untereinheiten nicht dasselbe genetische Material teilen (wie es das Erklärungsmodell der Soziobiologie fordert)
- der Selektionsvorgang ist ein dynamisches Gesamtgeschehen, an dem sowohl die Gene, als auch die Organismen, als auch deren Umwelt teilhaben
- ebenso einseitig könnte man vom "Egoismus der Organismen" sprechen, dem die Gene dienen müssen (denn Gene, die zu geringer Fitness der Organismen führen, werden in der Population eliminiert)
- Dawkins wertet den kurzlebigen Organismus als "weniger wichtig" als das langlebige Gen – metaphysische Wertung
- "Universalitätsbedürfnis" einiger Biologen? ("Wir können alles erklären – Psychologie, Soziologie etc. werden überflüssig")

Immerhin postuliert Dawkins in der menschlichen Kultur "Meme", die, analog zu Genen in der Biologie, eine eigene Ebene der Replikation konstituieren (und ebenfalls die Menschen für ihre Ausbreitung funktionalisieren: durch Imitationslernen).

- "Egoistisches Gen" vs. "Gaia-Hypothese": Variante des Konflikts zwischen Reduktionismus und Holismus

Wichtig ist:

- die grundlegenden Begriffe "genetische Variabilität" und "Selektion" machen nur Sinn in der Anwendung auf *Populationen*, d.h. wenn mehrere Individuen vorliegen:
- *Parallelismus* der Individuen einer Population als Voraussetzung Darwinscher Evolution
- Vererbung als Form zeitlicher Persistenz in diskreten Iterationsschritten (des Vorgangs der Replikation)

⇒ sowohl *Parallelismus* als auch *Iteration* sind essenziell für Darwinsche Evolution (aus algorithmischer Sicht)

- Begrenzung der Ressourcen, allgemeiner gesagt: *Endlichkeit* ist Voraussetzung für Möglichkeit der Selektion (u. damit für Evolutionsfähigkeit)

auch die *Sterblichkeit* der höheren Organismen ist aus dieser Perspektive zu sehen!

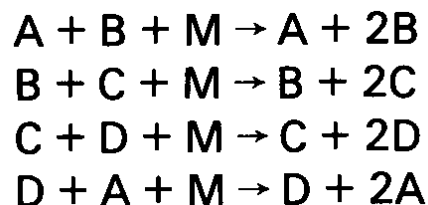
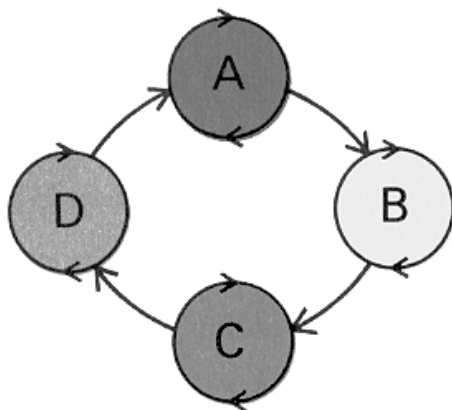
"genomische" vs. "nichtgenomische" Evolution:

- genomische Evolution: lässt sich anhand von Populationen von Zeichenketten (Nukleotidsequenzen) beschreiben
Trennung von Genotyp und Phänotyp, Weissmann'sches Prinzip
- nichtgenomische Evolution: keine Trennung von Genotyp und Phänotyp möglich

Beispiele für lebensnahe Systeme, bei denen Genotyp = Phänotyp ist: *autokatalytische Netzwerke* (siehe Kapitel 7)

Einfache Beispiele solcher Netzwerke:
"Hyperzyklen" (Eigen, Schuster)

jeder Reaktand katalysiert die eigene Entstehung und die des nächsten Reaktanden im Zyklus



(aus Eigen & Winkler 1979)

Solche Systeme können leicht mit "Kugelspielen" auf einem CA-artigen Gitter simuliert werden
(Kugel darf nur dann verdoppelt werden, wenn auf einem der Nachbarfelder eine Kugel der im Zyklus "vorangehenden" Farbe liegt)
– es kommt typischerweise zu Oszillationen (vgl. Belousov-Zhabotinsky-Reaktion)

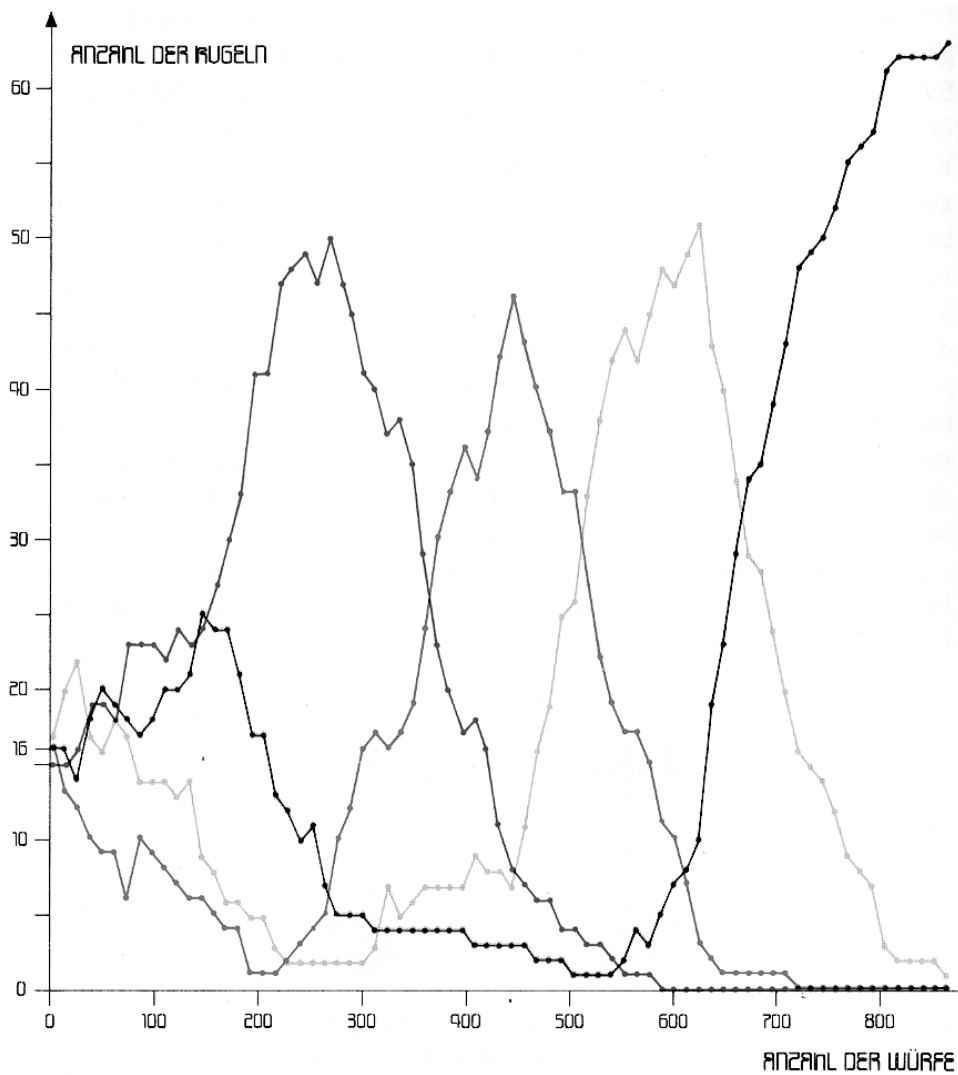
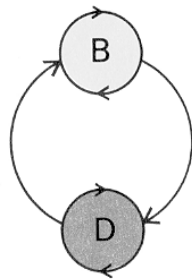
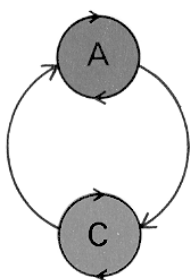


Abb. 52 Computeraufzeichnung eines »Hyperzyklus«-Spiels. Das Diagramm zeigt, wie sich die einzelnen Populationen von A (Rot), B (Gelb), C (Grün) und D (Blau) in Abhängigkeit von der Zahl der Würfe verändern. Der Zyklus stirbt aufgrund einer Fluktuationkatastrophe aus.

Eigenschaften von Hyperzyklen:

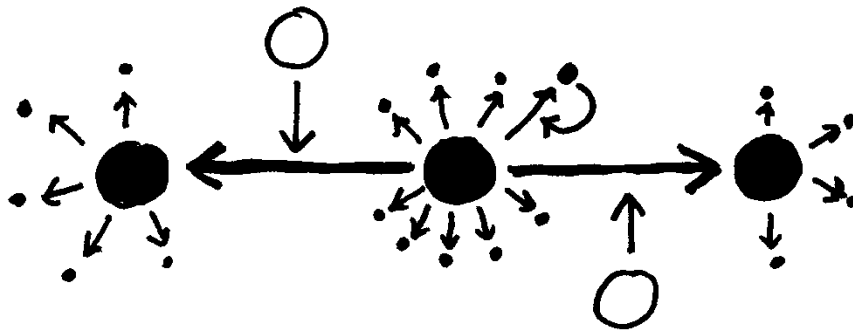
- die miteinander verkoppelten "Individuen" sind aufeinander angewiesen und damit koexistent. Im Existenzbereich des gesamten Zyklus ist jedes einzelne Mitglied stabil.
- Der Zyklus als ganzes ist nach außen hin "wettbewerbsfreudig" und fordert eine irreversible "Alles-oder-Nichts"-Entscheidung heraus



→ nur (A,C) oder (B,D) "überlebt"

Wie können autokatalytische Netzwerke evolvieren?
("Verdrängungswettbewerb" allein würde nichts Neues aufbauen können)

"chemischer Schatten" eines Substrats:
durch "zufällige", nichtkatalysierte (langsame) Reaktionen treten zu jedem Reaktanden des Netzwerks weitere Moleküle (Reaktionsprodukte) auf

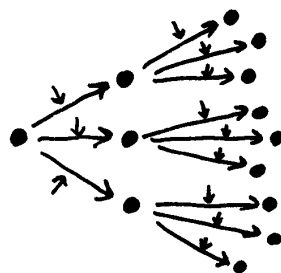


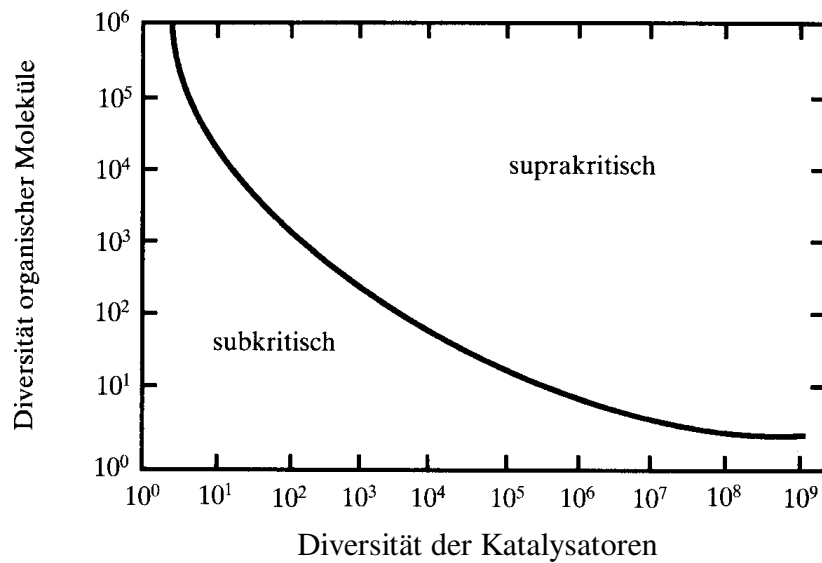
wenn eines davon die eigene Entstehung (direkt oder indirekt) katalysiert, kann es zu neuem Mitglied des Netzwerks werden – "Mutation"

Zahl der möglichen Reaktionen steigt stärker (quadratisch?) als die Zahl der vorhandenen Reaktionspartner (vgl. Kap. 7)
⇒ "kritische Diversität" der Molekülarten ist nötig, damit autokatalytische Abgeschlossenheit spontan entsteht

Regimes des dynamischen Verhaltens:

- "subkritisches Verhalten" einer "Molekül-Suppe": Diversität zu gering, es wird zu wenig katalysiert, kaum neue Moleküle entstehen
- "suprakritisches Verhalten": immer neue Moleküle werden durch katalysierte Reaktionen erzeugt, diese wirken ihrerseits katalytisch... lawinenartig entstehen neue Moleküle!





- Die Zellen der lebenden Organismen müssen subkritisch sein (sonst wäre kein geregelter Zellstoffwechsel möglich)
- Vermutung (St. Kauffman): die Biosphäre der Erde als Ganzes ist suprakritisch! (damit: Evolution "offen")

Stuart Kauffman:

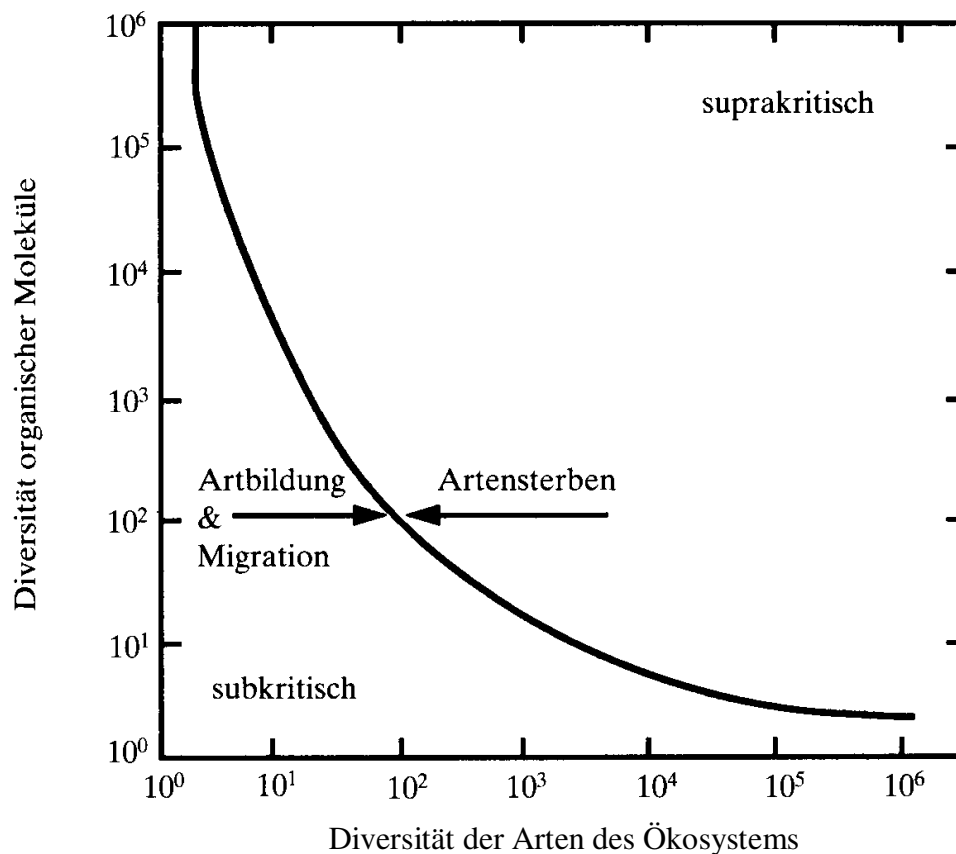
"Die Tatsache, dass wir unsere Speisen essen und nicht mit ihnen verschmelzen, weist... auf einen grundlegenden Sachverhalt hin: Die Biosphäre selbst ist suprakritisch. Unsere Zellen dagegen sind subkritisch. Würden wir mit dem Salat verschmelzen, dann würde die molekulare Diversität, die eine solche Verschmelzung in unseren Zellen erzeugt, eine umwälzende suprakritische Explosion auslösen. Die explosionsartige Zunahme neuer Molekülarten wäre für die unglückseligen Zellen, in denen dieser Prozess stattfände, schon nach kurzer Zeit tödlich. Die Tatsache, dass wir essen, ist kein Zufall, nicht einfach eine der zahlreichen möglichen Methoden, auf die die Evolution gekommen ist, um neue Moleküle in unsere metabolischen Netzwerke einzuspeisen. In den Vorgängen Essen und Verdauen spiegelt sich... die Notwendigkeit wider, uns vor der suprakritischen molekularen Diversität der Biosphäre zu schützen." (Kauffman 1995, S. 189)

- Bedeutung der Zellmembranen (vergleichbar den Graphitstäben in einem Kernreaktor, die ungebremste Kettenreaktion verhindern!)
- Bedeutung des Immunsystems

- subkritisch-suprakritische Grenze gibt obere Schranke für die molekulare Diversität einer Zelle \Rightarrow die Komplexität der Zelle ist begrenzt

Hypothesen (Kauffman, S. 198):

- Ökosysteme evolvieren zur subkritisch-suprakritischen Grenze und verharren dann dort, weil sie durch evolutionäre "Gelegenheiten" aus dem subkritischen Regime und durch Extinktionsdruck aus dem suprakritischen Regime herausgedrängt werden.



- Während jedes einzelne Ökosystem an der Grenze zwischen Sub- und Suprakritizität angesiedelt ist, erzeugen die Ökosysteme in ihrer Gesamtheit aufgrund ihrer stofflichen Wechselwirkungen eine suprakritische Biosphäre, die zwangsläufig immer komplexer wird.

genomische Evolution:

- hat sich als Spezialfall der nichtgenomischen entwickelt
- RNA, DNA und Proteine als spezielle Molekülklassen, die Spezialfunktionen in einem autokatalytischen Netzwerk übernehmen
- auch diese Separierung der Funktionen schützt die Zelle vor Abgleiten in die Suprakritizität
- Herausbildung dieser Aufgabenteilung vermutlich sehr früh in der Geschichte des Lebens
- Selektion in dieser Phase nach "Alles-oder-Nichts"-Prinzip: Universalität des genetischen Codes!

Beschreibung der Erbinformation hier in Form von Strings möglich

→ vereinfacht Beschreibung und theoretische Behandlung solcher Systeme

- thermodynamische Beschreibung
- einfache Manipulierbarkeit der "Genome" in ALife-Experimenten
- Beschreibung der genet. Struktur konkreter Populationen mittels Distanzmaßen zwischen Zeichenketten
- Fitness als direkte "Funktion" des Genom-Strings
- Genetische Algorithmen

wir müssen uns in den folgenden Abschnitten klar sein, dass wir nur eine Teilmenge möglicher evolvierbarer Systeme betrachten!

Beschreibung von Genom-Räumen und Populationen in thermodynamischen Begriffen

Thermodynamik: Versuch, Ensembles vieler, einfacher Teile durch aggregierte (statistische) Größen zu beschreiben (Temperatur, Druck, Entropie...)

Dynamisches System:

- *Phasenraum* = Menge aller möglichen Zustände des Systems
- *Regel* = Rechenvorschrift, die angibt, wie sich der Zustand des Systems zeitabhängig verändert (Dynamik des Systems)

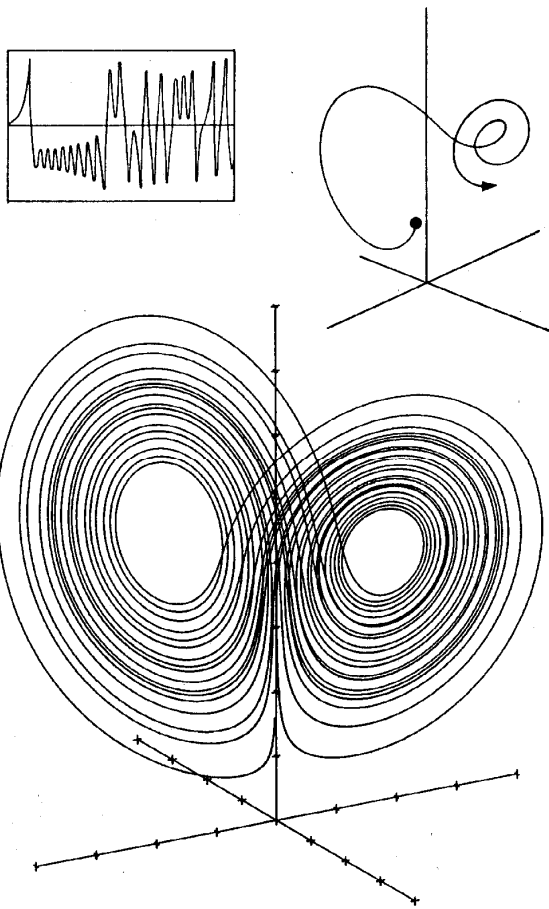
als Regeln häufig verwendet: Differentialgleichungen (kontinuierliches Modell); Funktion, die iteriert angewandt wird (diskretes Modell: Zustände werden nur an abzählbar vielen Zeitpunkten betrachtet).

Beispiel:

Pendel. Zustand: beschrieben durch Position und Geschwindigkeit (2D-Phasenraum (x, v)). Regel: Pendelgesetz aus der Mechanik.

implizite Annahme in der Theorie dynamischer Systeme:
wenn ein beliebiger Anfangszustand *exakt* bekannt ist, lassen sich *prinzipiell* alle zukünftigen Zustände des Systems berechnen.

Orbit eines Punktes im Phasenraum: Menge aller Folgezustände, die sich durch Anwendung der Regel ergeben. (bei iterierter Funktion f , also diskretes dyn. System: Orbit = Iteriertenfolge $x, f(x), f(f(x)), \dots$; bei kontinuierlichem Modell: stetige *Trajektorie* des Punktes.)

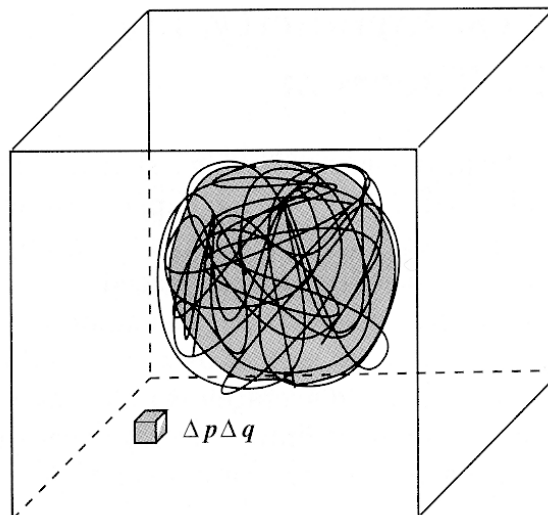


oben links:
Diagramm des zeitlichen
Verlaufs einer System-
eigenschaft, z.B. der Ge-
schwindigkeit

oben rechts:
Orbit des entsprechenden
Punktes im Phasenraum

unten: Attraktor, der den Punkt
schließlich "einfängt"

Die "Dichte" der Orbits im Phasenraum wird quantifiziert durch die Wahrscheinlichkeitsdichtefunktion (statist. Verteilungsfunktion) ρ des Systems:
die Wahrscheinlichkeit, das System in einem Zustand innerhalb einer gegebenen, messbaren Teilmenge des Phasenraums vorzufinden, ist das Integral von ρ über diese Teilmenge.



Mittelung einer Quantität f über die Zustände des Systems:

$$\langle f \rangle = \int f(p, q) \rho(p, q) dp dq$$

Ein System heißt *ergodisch*, wenn jeder Orbit den gesamten Teil des Phasenraums, wo ρ nicht verschwindet, erfüllt.

In diesem Fall kann man den Mittelwert einer Größe f auch durch Mittelwertbildung über die Zeit gewinnen:

$$\langle f \rangle = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T f(p(t), q(t)) dt$$

Raum der Genomzustände:

wird identifiziert mit dem Raum aller Nukleotidsequenzen (einer gegebenen maximalen Länge)

beispielsweise für Sequenzen der Länge 10^9 : 4^{10^9} mögliche Zustände \Rightarrow weit mehr als das Alter des Universums (und in diesem Zustandsraum können *nicht* a priori große Äquivalenzklassen gebildet werden)

\Rightarrow die Evolution kann nur einen verschwindend kleinen Teil des Genomraumes durchlaufen:

nichtergodischer Charakter der Evolution

\Rightarrow der Zustand des Systems hängt in essenzieller Weise von seiner *Geschichte* ab (anders als bei ergodischen Systemen)

Darwinsche Evolution hat "historischen" Charakter!

"Kontingenzprinzip": Eigenschaften des real vorkommenden Lebens können vom historischen "Ur-Genotyp" (Anfangszustand) abhängen

*Fortführung der Übertragung thermodynamischer
Begriffsbildungen:*

Genotyp (Sequenz) – entspr. den Ortskoordinaten und Momenten in der klassischen Thermodynamik

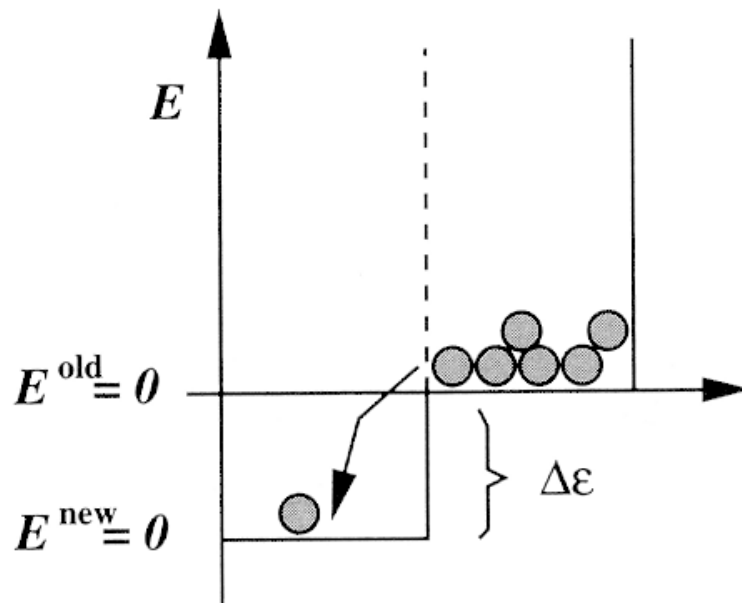
Inferiorität $I(s)$ eines Genotyps s : $I(s) = \log \varepsilon_{best} - \log \varepsilon(s)$,
wobei $\varepsilon(s)$ die Replikationsrate von s und ε_{best} die beste in der
Population vorkommende Replikationsrate ist

- bei Abwesenheit von Mutationen tendiert die Population durch den Selektionseffekt zu einem Zustand mit durchschnittlicher Inferiorität 0

Inferiorität entspr. der *Energie* eines Systems

Eine Mutation, die einen String mit höherer Replikationsrate liefert als die bisher beste, macht den alten Nullzustand "metastabil" und führt zu einer "Renormalisierung" des Inferioritätsmaßes (da fast alle Elemente der Population plötzlich eine höhere Inferiorität als zuvor haben):

$$\varepsilon_{best}^{neu} = \varepsilon_{best}^{alt} + \Delta\varepsilon$$



dieser Übergang kann als *Phasenübergang* mit latenter Wärme $\Delta\varepsilon$ angesehen werden

Die *Entropie* eines Systems ist ein Maß für die Unsicherheit über den Zustand des Systems

– nimmt zu mit der Anzahl der verschiedenen Zustände, die das System annehmen kann

im Genomraum: Entropie $S = \log N_g$,

wobei N_g die Anzahl der verschiedenen Sequenzen in der Population ist (Maß für die Diversität der Population)

Wenn die relativen Häufigkeiten ρ_i der Genotypen bekannt sind:

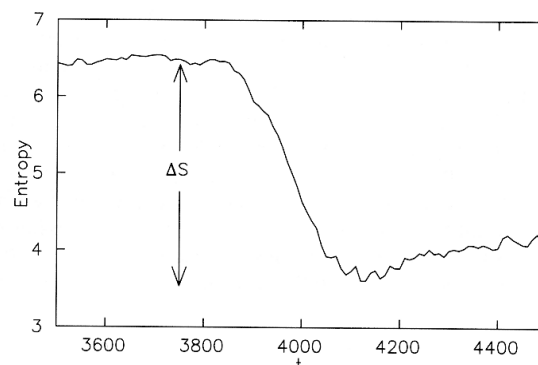
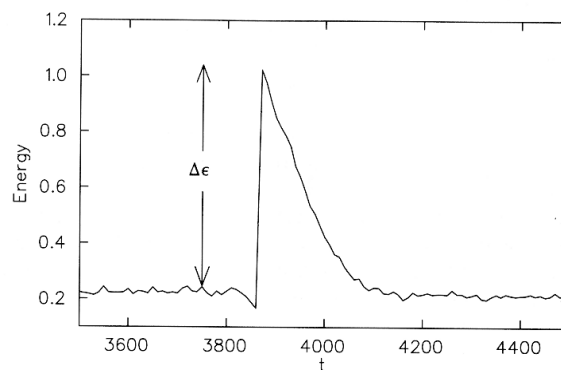
bedingte Entropie
$$S = -\sum_{i=1}^{N_g} \rho_i \log \rho_i$$

Die *Temperatur* eines thermodynamischen Systems ist def. als die inverse Änderungsrate der Entropie mit der Energie:

$$\frac{1}{T} = \frac{dS}{dE}$$

im Genomraum ist diese Größe proportional zur *Mutationsrate*.

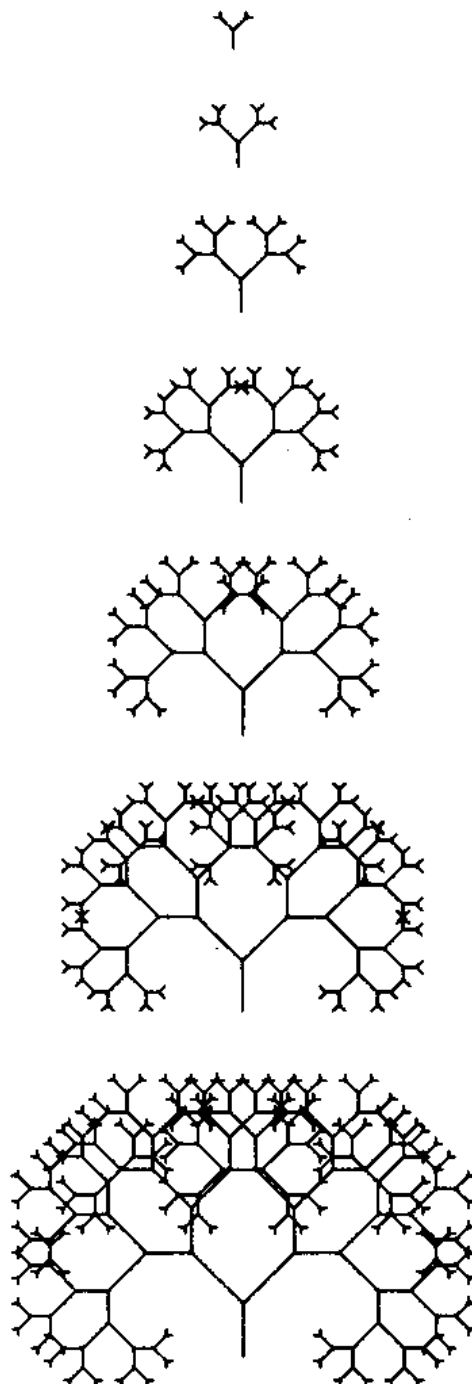
Phasenübergang beim Auftreten einer neuen "bestangepassten" Mutante in einem ALife-Experiment (aus Adami 1998):



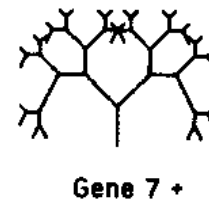
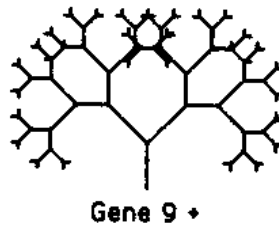
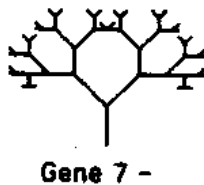
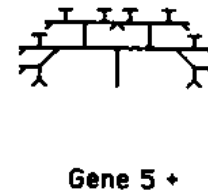
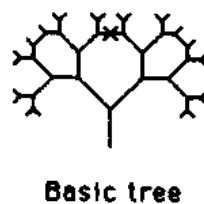
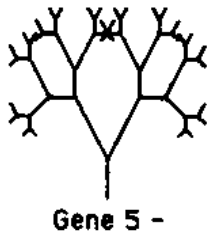
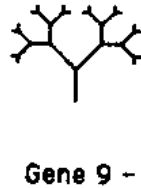
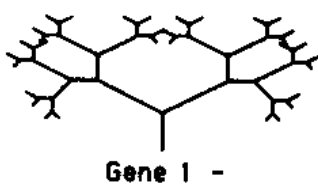
Beispiel eines einfachen Genomraumes, versehen mit morphologischen Phänotypen:

Richard Dawkins' "*Biomorphe*" (Begriff von Desmond Morris)

Morphologie: Baum mit dichotomer Verzweigung
9 Gene, Allele jeweils durch ganze Zahlen codiert
eines der Gene steuert die Rekursionstiefe:



die anderen Gene beeinflussen unterschiedliche metrische Größen (Längenverhältnisse, Winkel):



- *Mutationen* addieren +1 oder -1 zum Wert eines der Gene hinzu
- keine geschlechtliche Vermehrung (Rekombination)

zu einem "Eltern-Biomorph" wird eine Menge mutierter "Nachkommen" generiert

die *Selektion* erfolgt interaktiv durch den *menschlichen Benutzer* (*animal breeder selection*, vgl. Tier- und Pflanzenzüchtung)

– Wirkung: (ästhetische) Entscheidungen des Benutzers werden kumuliert (*cumulative selection*)

interaktiv:

<http://islwww.epfl.ch/~moshes/biomorphs/evolution.html>
(dort mit Rekombinationsmöglichkeit erweitert)

Dawkins schreibt:

Als ich das Programm schrieb, kam mir niemals der Gedanke, daß es etwas anderes entwickeln könnte als eine Varietät von baumähnlichen Gestalten. Ich hatte Trauerweiden erhofft, Libanonzedern, Pyramidenpappeln, Meeresalgen, vielleicht Hirschgeweihe. Nichts in meiner Intuition als Biologe, nichts in meiner 20jährigen Erfahrung im Programmieren von Computern, und nichts in meinen verrücktesten Träumen hatte mich auf das vorbereitet, was tatsächlich auf dem Bildschirm erschien. Ich kann mich nicht daran erinnern, an welchem Punkt der Sequenz es mir zu dämmern begann, daß eine durch Evolution entstandene Ähnlichkeit mit einem Insekt möglich war. Voller Argwohn begann ich zu züchten, Generation auf Generation – und von jedwedem Kind, das am meisten wie ein Insekt aussah. Mein ungläubiges Erstaunen wuchs in gleichem Maße wie die sich entwickelnde Ähnlichkeit ... Ich spüre immer noch das Triumphgefühl, das mich erfüllte, als ich diese sonderbaren Kreaturen zum ersten Mal vor meinen Augen entstehen sah. Ganz deutlich hörte ich die triumphierenden ersten Akkorde von *Also sprach Zarathustra* (dem Thema zum Film *2001*) in meinem Geist. Ich konnte nicht essen, und in jener Nacht schwärmten »meine« Insekten hinter meinen Augenlidern herum, als ich zu schlafen versuchte.⁹

(Dawkins 1990, S. 77)

Später: Schwierigkeit, die "Insekten" im Genomraum "wiederzufinden" (erste Version des Computerprogramms speicherte nur die Phänotypen ab!)

– nichtergodischer Charakter wird schon in diesem einfachen Beispiel deutlich

Bei meinen Wanderungen durch die entfernten Gefilde des Landes der Biomorphe erhielt ich Feenkabben, Aztekentempel, gotische Kirchenfenster, Eingeborenenzeichnungen von Känguruhs und, bei einer bemerkenswerten, aber nicht wiederholbaren Gelegenheit, eine passable Karikatur des Wykeham-Professors für Logik.¹⁰

Beispiele:



Schwabenschwanz



Mann mit Hut



Mondlandefähre



Präzisionswaage



Köcherfliege



Skorpion



Fadenspiel



Laubfrosch



Spitfire



Gekreuzte Säbel



Bienenorchidee



Kopffüßer



Insekt



Fuchs



Lampe

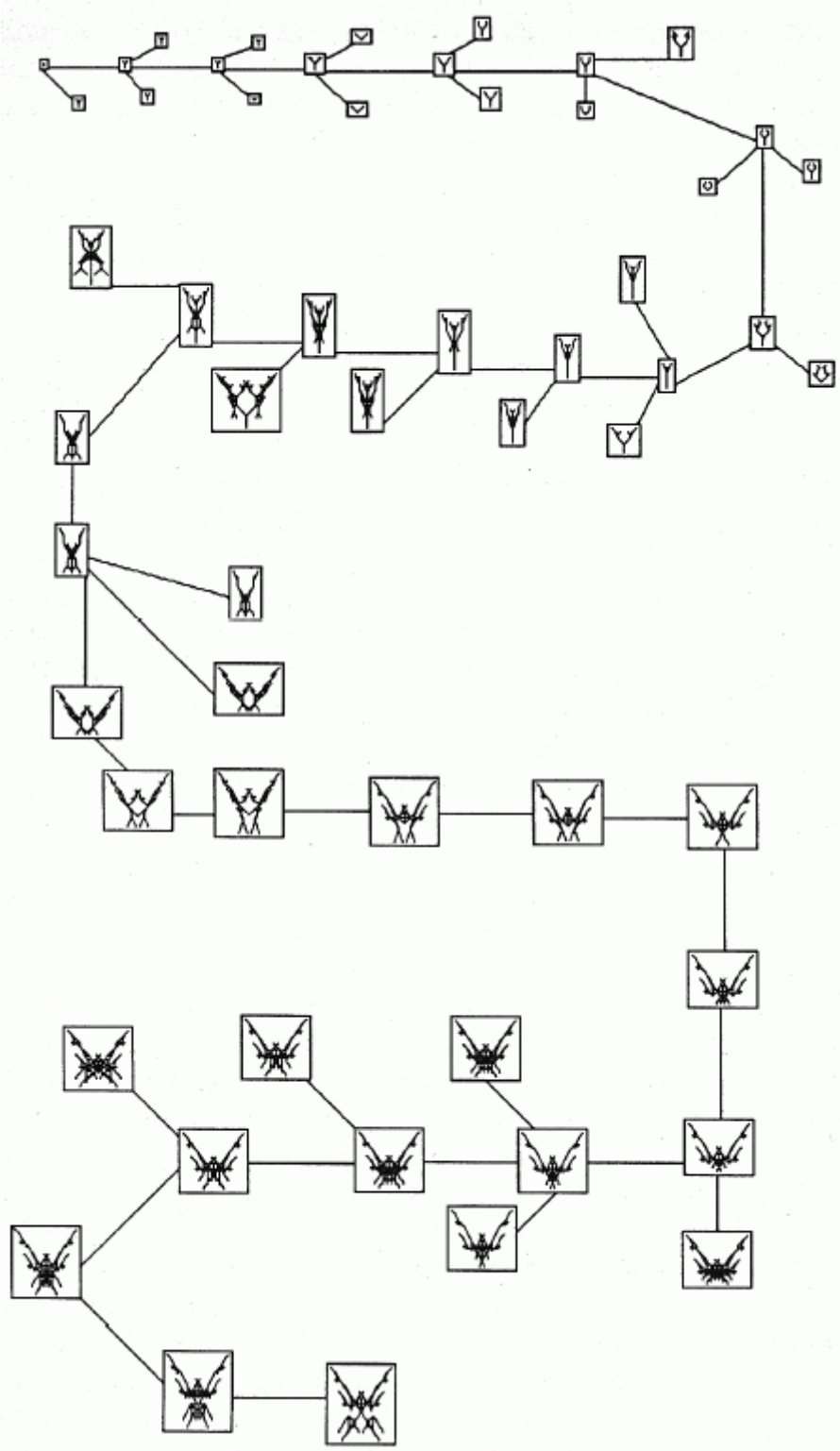


Springende Spinne



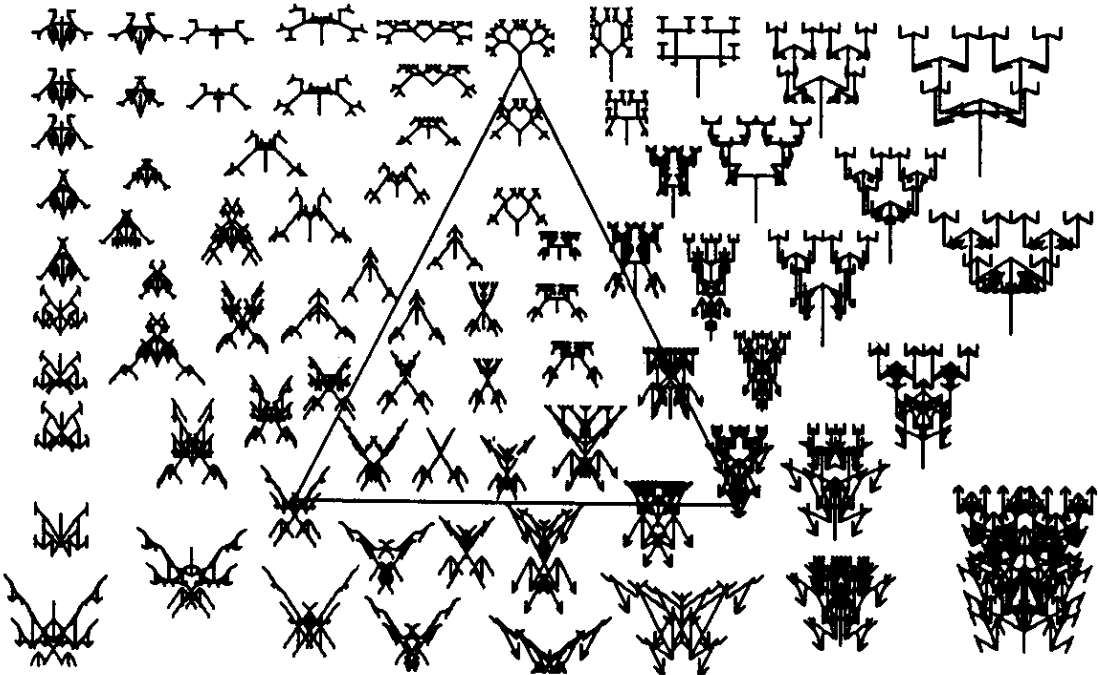
Fledermaus

Beispiel der Entwicklung eines "Insekts" in 29 Generationen:



(Dawkins 1990)

Visualisierung einer Ebene im Genomraum (aufgespannt durch die 3 Ecken des Dreiecks) anhand der Phänotypen:



Beschreibung der Struktur von Populationen

Diversität (bzw. Entropie) als Anzahl verschiedener Genome: sehr grobes Maß, berücksichtigt nicht die Abstände im Genomraum

Abstandsmaße für Zeichenketten:

- *Hamming-Distanz* (für gleichlange Strings; Anzahl der Positionen mit unterschiedlicher Belegung)
- *Levenshtein-Distanz* (Editierabstand): minimale Zahl von Editieroperationen (bzw. Mutationen), die erforderlich ist, um den einen String in den anderen zu transformieren (Editier-Operationen: Überschreiben, Einfügen und Löschen je eines einzelnen Zeichens)
- *relativer Editierabstand*: Editierabstand, geteilt durch die Länge der längeren von beiden Zeichenketten

Taxonomische Struktur:

Einteilung einer Population in *taxonomische Kategorien*

Kategorien der biologischen Systematik:

Art = Spezies, Gattung, Familie, Ordnung, Klasse, Abteilung, Stamm, Reich

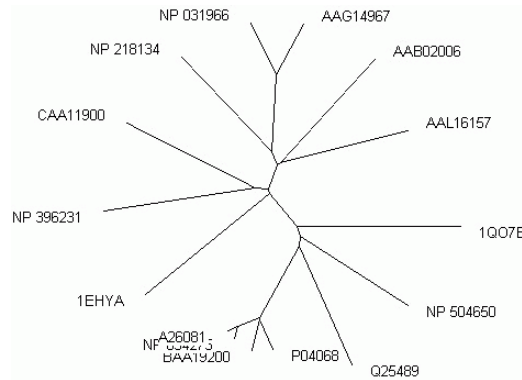
(Carl von Linné)

Mitglieder derselben Kategorie haben untereinander geringere Distanz als zu Mitgliedern anderer Kategorien der gleichen Hierarchie-Ebene

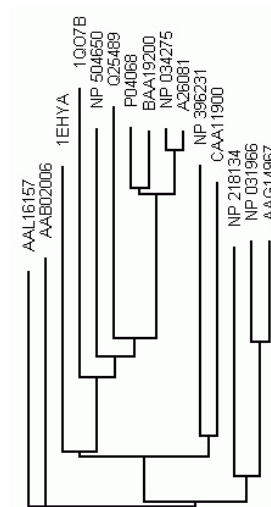
Visualisierung in *phylogenetischen Bäumen* (Verwandtschaftsbäumen):

Kantenlängen sollen zu Genom-Distanzen proportional sein

- ohne Auszeichnung eines Wurzelknotens:



- mit Wurzelknoten; Längen werden nur auf den Vertikalen gewertet:

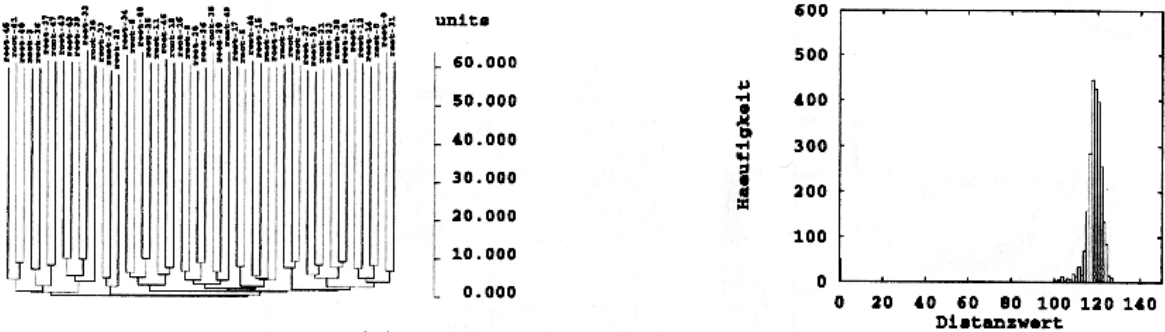


Distanzverteilung

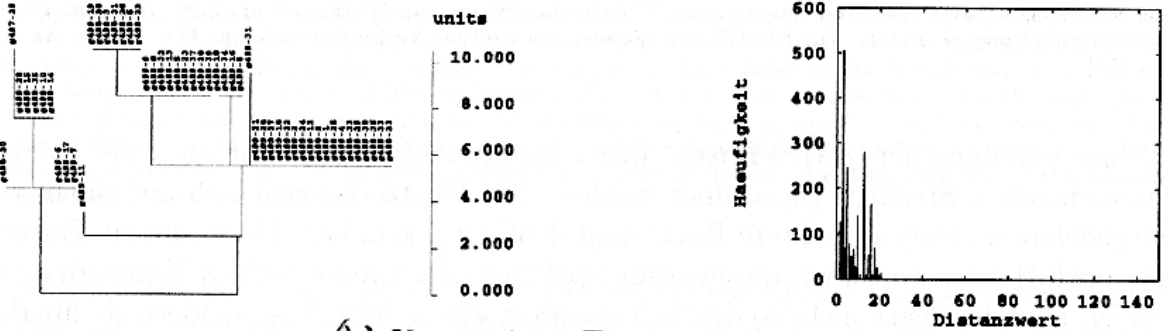
Sei P eine Population aus n Individuen und d ein diskretes Distanzmaß auf P . Dann ist die Distanzverteilung F def. durch

$$F(k) = |\{ (x, y) \mid x, y \in P, d(x, y) = k \}| / 2.$$

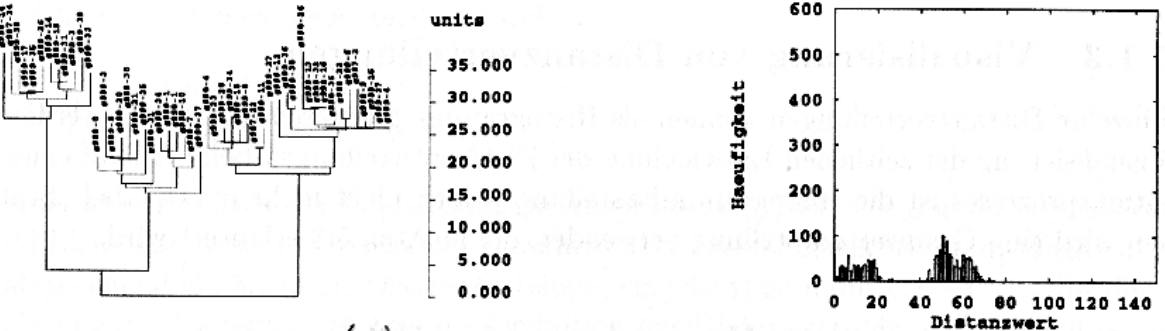
Beispiele für Populationen mit unterschiedlicher taxonomischer Struktur:



(a) Randomisierte Taxonomie



(b) Konvergierte Taxonomie



(c) Komplexe Taxonomie

(aus Kim 1996)

Distanzverteilungskomplexität (DVK):

Shannon-Entropie der normalisierten Distanzverteilung.

Sei f die Verteilung der relativen Häufigkeiten der Distanzwerte:

$$f(k) = \frac{F(k)}{\sum_i F(i)} = \frac{2 \cdot F(k)}{N(N-1)}$$

dabei sei N die Größe der Population.

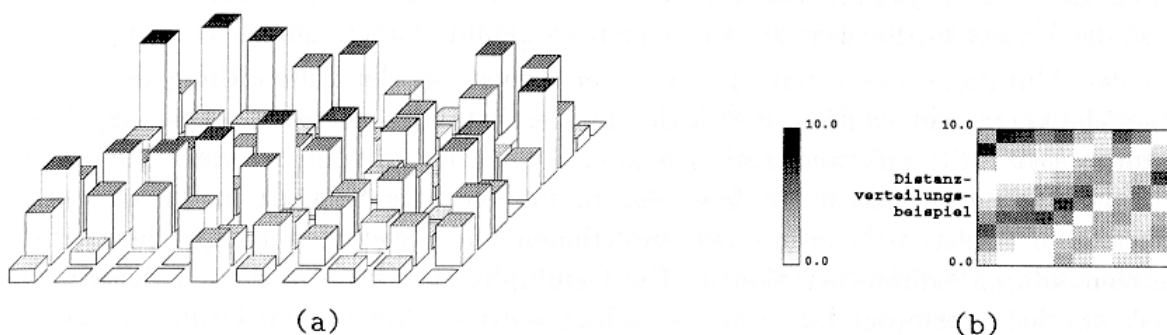
Dann ist die Distanzverteilungskomplexität

$$C = -\sum_k f(k) \log(f(k))$$

(Kim 1996).

Hohe DVK: gleichmäßige Verteilung der Distanzwerte,
komplexe taxonomische Struktur!

Visualisierung der dynamischen Entwicklung von
Distanzverteilungen (z.B. im Verlauf einer künstlichen
Evolution):



Serie von 10 Verteilungen (Zeitachse = horizontal),

(a) dreidimensional, (b) mit Grauwertcodierung

globale Strukturen treten in der Grauwertcodierung deutlicher hervor

(Kim 1996)

Explizitmachen der Fitness

im "Biomorph"-Beispiel fand die Selektion benutzergesteuert (interaktiv) statt

Annahme jetzt: jedem möglichen Genom kann eine Fitness (= Wahrscheinlichkeit, genetisch zu überdauern) *fest zugeordnet* werden

Achtung: das ist Idealisierung!

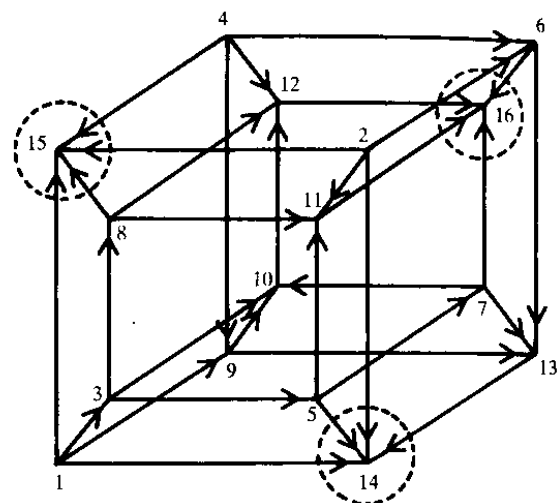
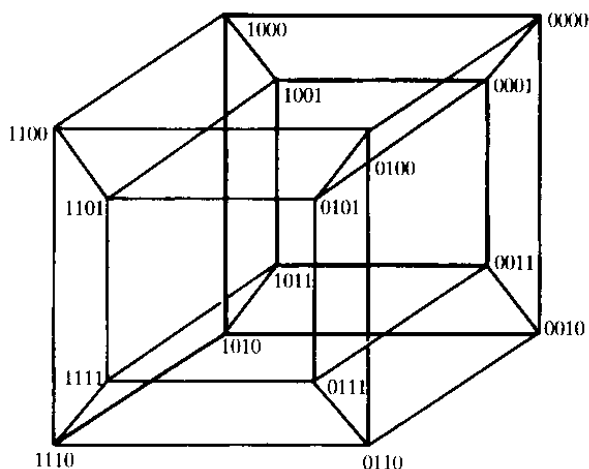
- Überleben und Fortpflanzung kann auch in der Natur von zufälligen Ereignissen abhängen
- Fitness kann von Konkurrenten (und damit von der Zusammensetzung der Population) abhängen
- die gleiche Genom-Zeichenkette kann unter verschiedenen Bedingungen verschiedene Fitness haben

Unter der obigen Annahme kann Evolution als Lösungsversuch einer *Optimierungsaufgabe* (Extremalproblem) angesehen werden:

Finde den Ort (im Genom-Raum) mit maximaler Fitness!

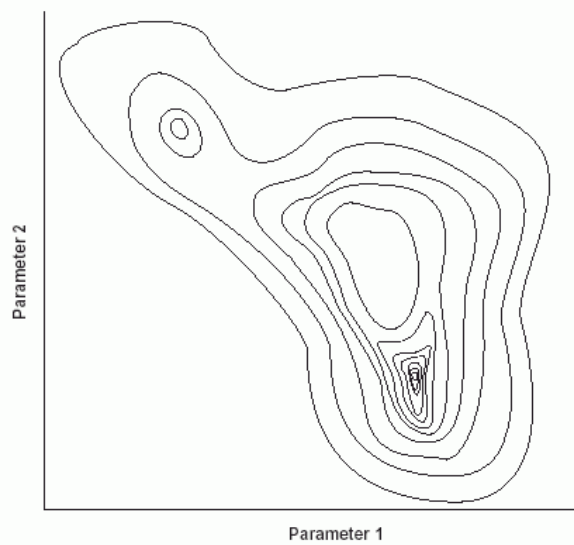
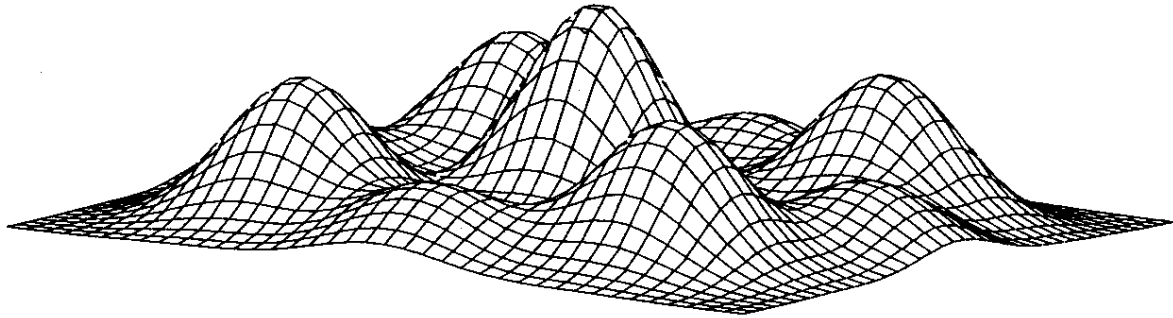
"Fitnesslandschaft" (Begriff nach Sewall Wright 1932):

jedem Punkt des (hochdimensionalen) Sequenzraumes wird seine Fitness als Zahlenwert zugeordnet (*Fitnessfunktion*)



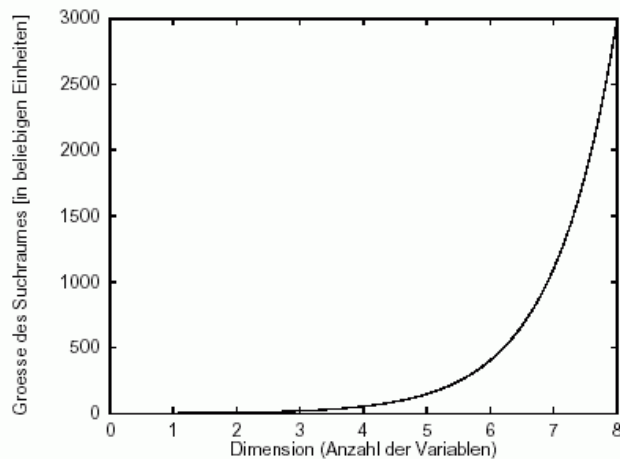
alle Genome der Länge 4 (links); rechts: Belegung mit Fitness-Werten; eingekreist: lokale Maxima ("Gipfel") der Fitnesslandschaft (Kauffman 1995)

2D-Analogen:



beachte: wegen der Hochdimensionalität des Genom-Raumes kann die Landschafts-Analogie irreführend sein!

insbesondere wächst die Größe des Genom-Raumes exponentiell mit der Dimension



Evolutionäre Algorithmen

Grundidee:

- Setze eine Population von Individuen in den Genom-Raum (der mit einer *festen* Fitness-Funktion ausgestattet ist)
- lasse diese evolvieren:
durch Anwendung von Evolutions-Operatoren
 - Mutation
 - Rekombination
 - (evtl. weitere)
 - und Selektion (Überleben von Individuen, bevorzugt solcher mit hoher Fitness)
- Ziel: die Population soll sich zum globalen Fitness-Maximum hinbewegen

Verschiedene Varianten:

- Genetische Algorithmen (John Holland; Mitte 60er Jahre)
- Evolutionsstrategien (Rechenberg, Schwefel, Bientert; 60er Jahre)
- Evolutionary Programming (Fogel 1960)
- genetisches Programmieren (Koza 1987)

John Holland:

- erster Inhaber eines Informatik-Dokortitels in den USA
- Studenten seines Kurses "Theory of Adaptive Systems" waren maßgeblich an der Entwicklung der ersten GA beteiligt
- Holland machte keine Tests und Experimente mit dem Computer: "Ich bin immer ein Papier- und Bleistift-Mensch gewesen..." (Levy 1993, S. 199)
- GA-Experimente seines Doktoranden David Goldberg auf Apple-II-PC, verwendete GA zur Optimierung des Auffindens von Erdgasvorkommen

Genetische Algorithmen

- arbeiten auf Strings (oft Binärstrings)
- verwenden Rekombination ("mating" von 2 Individuen und crossing-over der Genome) und meistens auch Mutation
- Selektion der zu paarenden Individuen nach einer Selektionsregel, die die Fitness der Individuen einbezieht

Basic algorithm:

- Initialize population P
- Repeat for some length of time
 - Create an empty population, P'
 - Repeat until P' is full:
 - Select two individuals from P based on some fitness criterion
 - Optionally "mate", and replace with offspring
 - Optionally mutate the individuals
 - Add the two individuals to P'
 - Let P now be equal to P'

Verschiedene Selektionsregeln:

- Auswahl nur aus den besten 50% (oder 20%, etc.)
 - Nachteil: Verringerung der genetischen Diversität
- Zufallsauswahl, wobei die Wahrscheinlichkeit zur Fitness proportional ist ("*roulette wheel selection*") – gibt auch "den Schlechten" eine Chance
- "Wettkampf-Selektion": 2 Individuen werden völlig zufällig ausgewählt, das fittere von beiden wird verwendet
- Rang-Selektion: Wahrscheinlichkeit der Auswahl ist proportional zur Rangposition (bei Sortierung nach Fitness)
- bestes Individuum überlebt "sicher" (um Verminderung der Fitness der Population zu verhindern), die Auswahl der übrigen erfolgt nach einer der obigen Regeln
- ...

Selektion und Reproduktion

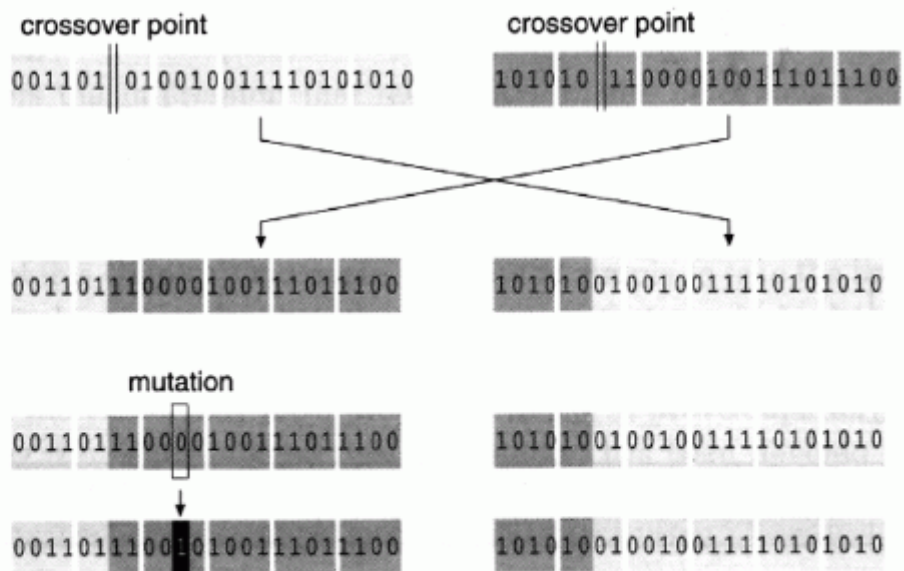
(hier mit einer weiteren Selektionsstrategie):

a. selection

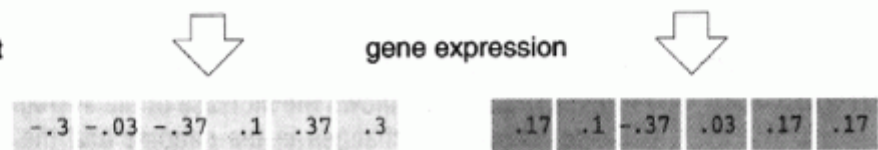
1. take the individual with the highest fitness
2. choose another individual from the population at random, irrespective of fitness, for sexual reproduction
3. add the fittest individual to the new population

	fittest individual (highest rank)						other individual					
	1	2	3	4	5	6	1	2	3	4	5	6
initial genome	0	0	1	1	0	1	0	1	0	1	0	0
encoded weights	-0.3	-0.17	-0.37	0.03	0.17	0.17	0.17	0.23	-0.5	0.1	0.37	0.3

b. reproduction

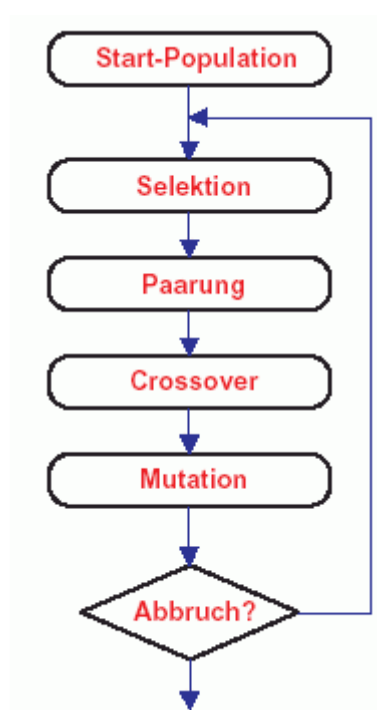


c. development



(Schritt c, "development", entfällt bei den meisten GA)

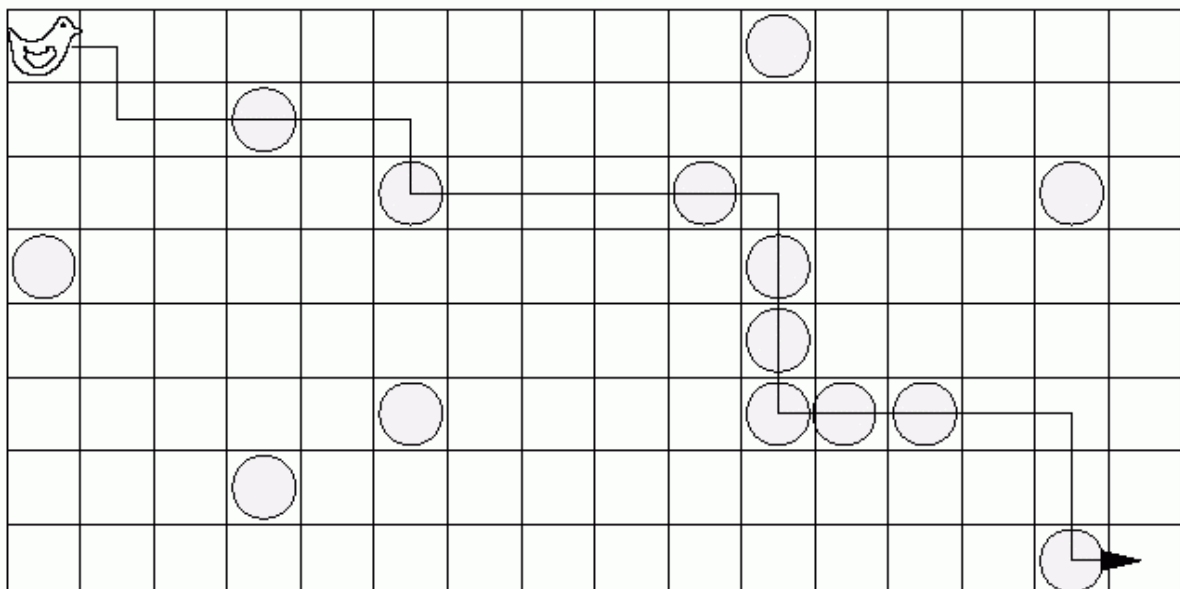
Zusammenfassung: Flussdiagramm des GA



Codierung realer Probleme für die Lösung mittels GA:

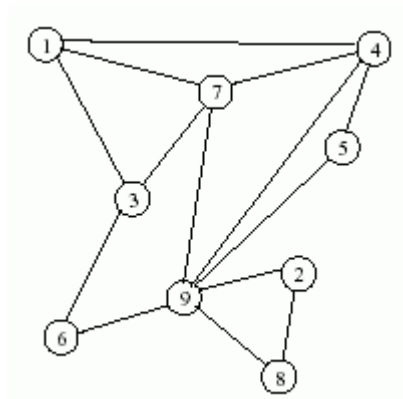
- Beispiel "Hungrige Ente"

Eine Ente, die nur nach rechts und unten schwimmen kann, soll einen Weg durch den Teich finden, bei dem sie möglichst viele Futterbrocken erreicht



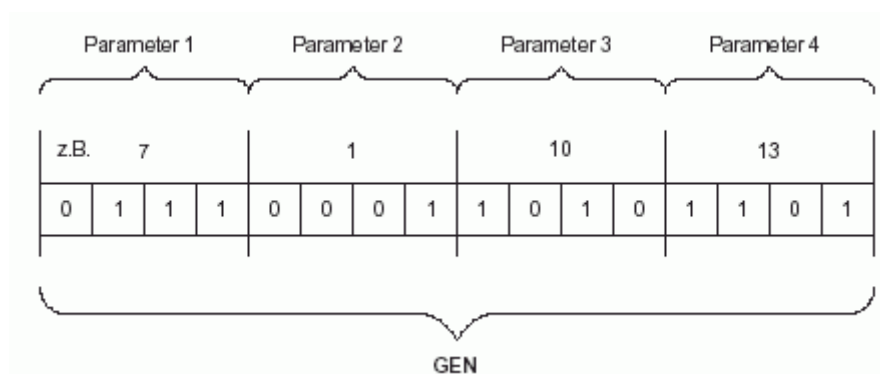
Codierung als Binärstring (0 = nach rechts, 1 = nach unten)

- Problem des Handlungsreisenden (*travelling salesman problem*): Auffinden der kürzesten Rundreise durch alle Ecken eines gegebenen Graphen



Codierung: Permutation der Ecken-Labels

- Maximum einer Funktion mit 4 Variablen:
Binärcodierung der 4 Parameter in Blöcken fester Länge, diese werden aneinandergehängt



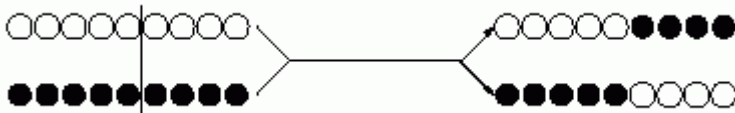
- Bei Zahlencodierung ist der Gray-Code angemessener als die gewöhnliche Binärcodierung (bessere Entsprechung der Differenz zwischen den Zahlen zur Hamming-Distanz der Genome)

Binärer Code	Gray Code	Dezimalzahl
000	000	0
001	001	1
010	011	2
011	010	3
100	110	4
101	111	5
110	101	6
111	100	7

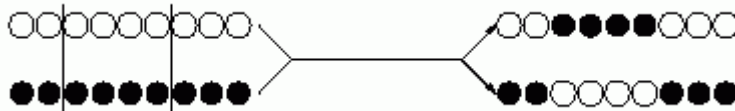
Evolutions-Operatoren:

Crossing over (Kreuzung, Rekombination)

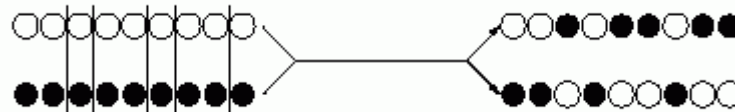
einfaches crossing over: die Zeichenketten werden an einer (zufällig bestimmten) Stelle getrennt und es werden 2 Nachkommen erzeugt, indem der hintere Teilstring vertauscht wird



doppeltes crossing over:



mehrfaches crossing over:



Variante für Permutationsstrings:

ersetze in den Nachkommen doppelt auftretende Zahlen so lange durch andere Zahlen, bis keine Zahlen mehr doppelt auftreten



9 ↔ 6 3 ↔ 2 2 ↔ 4

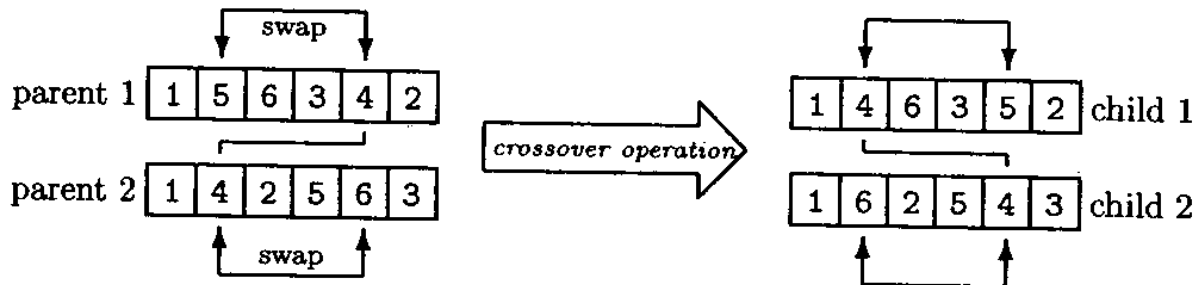
3
2 6
1 4 8 9 2 4 9 7 5

1 3 8 6 2 4 9 7 5

4
2 9
7 5 3 6 3 2 8 1 6

7 5 4 9 3 2 8 1 6

andere crossover-Variante für Permutationsstrings:
 Position zufällig auswählen, 2. Pos. ergibt sich durch
 entsprechenden Eintrag im 2. String, zwischen beiden Pos. wird
 in beiden Strings (einzeln) getauscht



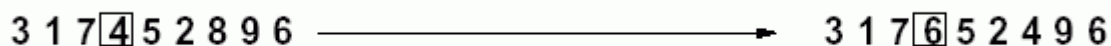
Mutation

Zufallsauswahl einer Position im Genom
 bei Binärstring: dort das Bit "flippen"



bei Zahlenstring:

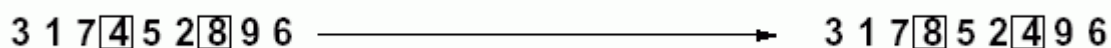
Wert x durch $(Max+1)-x$ ersetzen



oder durch zufällig gezogene Zahl

bei Permutationsstring:

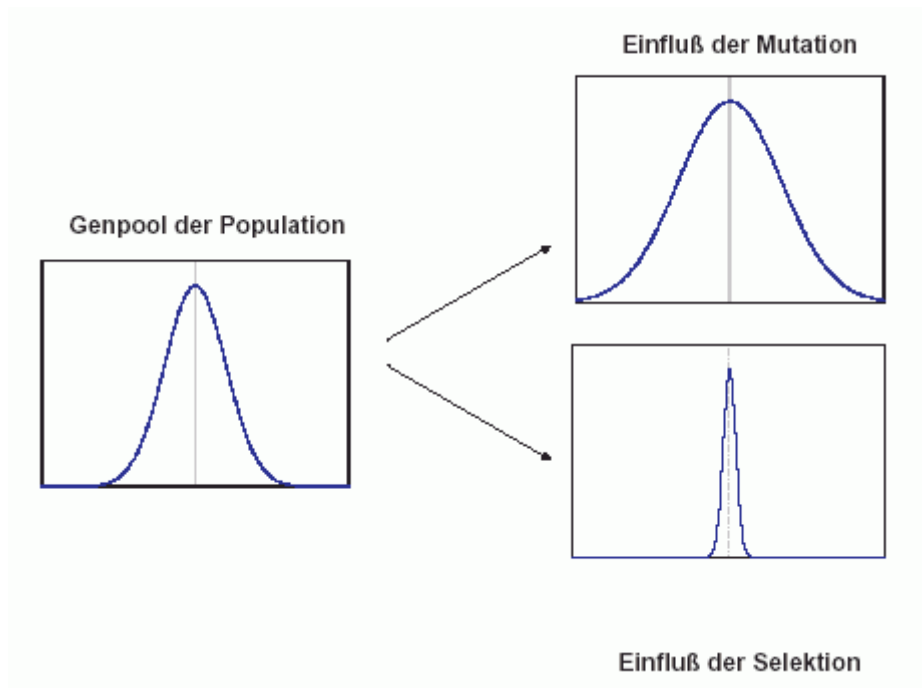
zwei Stellen auswählen und die dortigen Einträge vertauschen



zu beachten:

Mutation und Selektion müssen vernünftig "ausbalanciert" sein

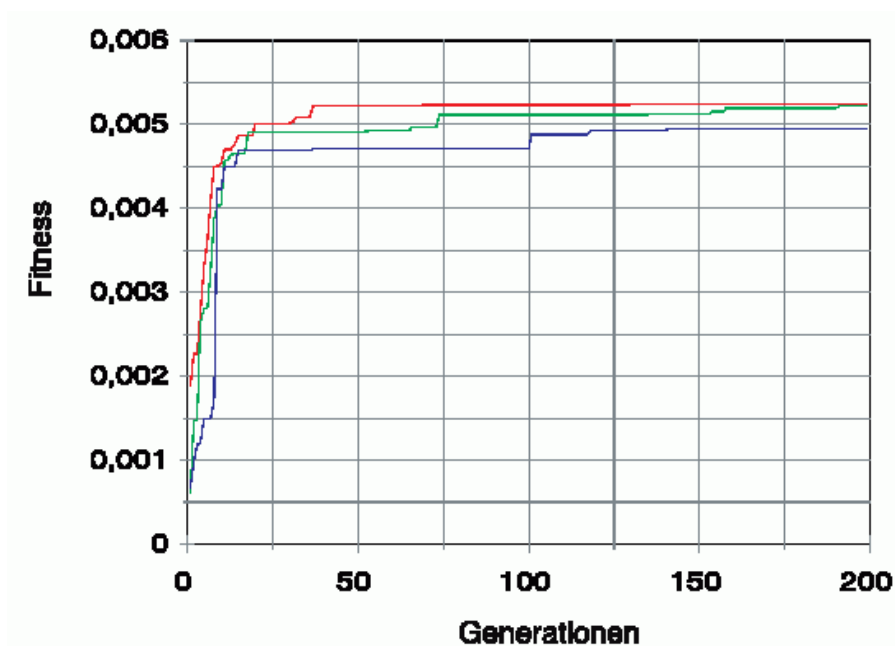
- Mutation erhöht die Diversität des Genpools (eventuell zu stark: Algorithmus wird ungerichtet, Population diffundiert von Maximalstellen wieder fort)
- Selektion vermindert sie (wenn zu stark: "Festsitzen" auf lokalen, nicht-globalen Maxima)



Iteration des GA:

Oft werden mehrere GA-Läufe mit unterschiedlichen Startpopulationen hintereinander durchgeführt, um die Wahrscheinlichkeit zu vergrößern, tatsächlich das globale Maximum der Fitness zu finden
(Beachte: keine Garantie, dieses zu finden! – GA sind *heuristische* Algorithmen)

Typische Fitness-Kurven des jeweils besten Individuums einer Population in 3 GA-Läufen:



Faustregeln für GA (nach Güde 2001):

- Populationsgrößen sollten ≥ 100 sein
- schwierigere Probleme erfordern größere Populationsgrößen
- Anwendungshäufigkeit der Operatoren:
crossing over ca. 80%, unveränderte Übernahme in die nächste Generation ca. 20%, Mutationsrate ca. 0,1%

Beispiel für Anwendung eines GA:

- Individuen sind Strings aus Buchstaben
- Fitness misst die Nähe zum Zielstring "furious green ideas sweat profusely"
- Populationsgröße 500
- crossover-Rate 75%, Mutationsrate 1%
- → Zielstring wird in 46 Generationen erreicht

Time	Average Fitness	Best Fitness	Best String
0	0.035314	0.200000	"pjrmrbynrksxiidwctxfodkodjjzfunpk "
1	0.070000	0.257143	"pjrmrbynrksxiidnybvsqcq piisyexdt"
⋮	⋮	⋮	⋮
25	0.708686	0.771429	"qurmous gresn idnasvsweqt prifuseky"
26	0.724286	0.800000	"qurmous green idnasvsweqt prifuseky"
⋮	⋮	⋮	⋮
36	0.806514	0.914286	"uurious green idnas sweqt profusely"
37	0.820857	0.914286	"qurmous green ideas sweqt profusely"
⋮	⋮	⋮	⋮
41	0.895943	0.942857	"uurious green idnas sweat profusely"
42	0.908457	0.971429	"qurrious green ideas sweat profusely"
⋮	⋮	⋮	⋮
45	0.927714	0.971429	"qurrious green ideas sweat profusely"
46	0.936800	1.000000	"furious green ideas sweat profusely"

(aus Flake 1998)

weitere Varianten evolutionärer Algorithmen:

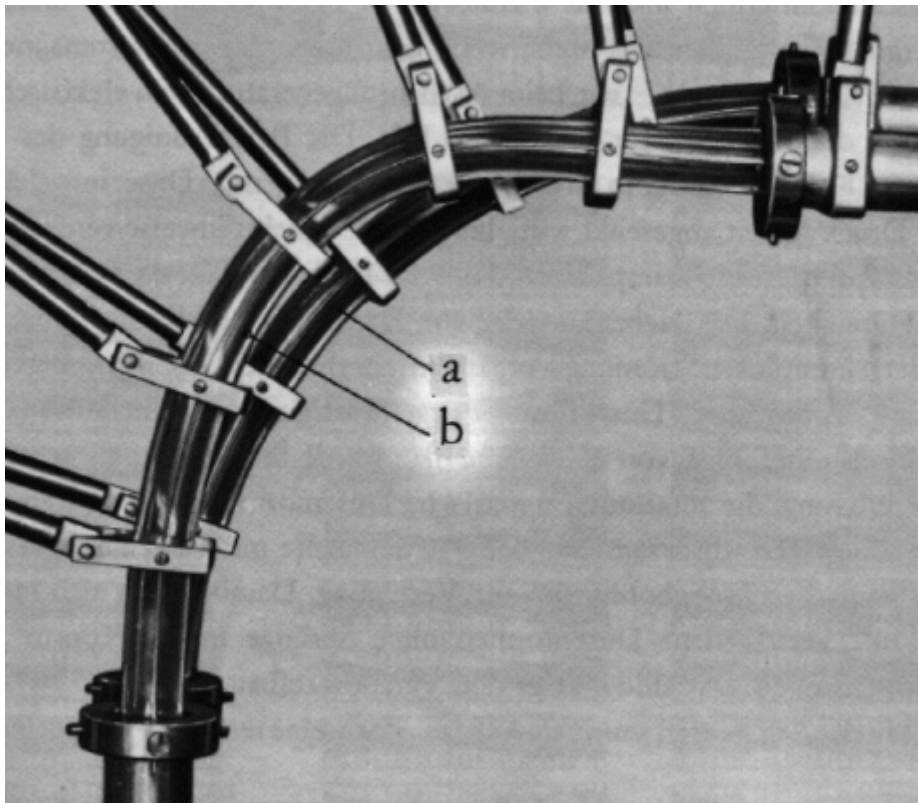
Evolutionstrategien (ES)

- anfänglich eine Methode zum Finden optimaler Parameterkombinationen in Laborexperimenten (damals noch nicht rechnergestützt), Populationsgröße dann = 1
- Genom besteht aus Vektoren reeller Zahlen
- Mutation, Rekombination
- alle Komponenten eines Vektors werden simultan mutiert
- "Gen-Expression": Vektoren beschreiben reale Objekte
- 2 Vektoren je Individuum: Objekt-Parameter und Strategie-Parameter
- die Strategie-Parameter kontrollieren die Mutationen der Objekt-Parameter (und werden selbst optimiert – "Selbst-Adaptation")

Anwendungsbeispiel: Minimierung des Flusswiderstandes in einer Rohrverbindung (Rechenberg)

a: naive Lösung (viertelkreisförmige Verbindung)

b: optimale Lösung, durch ES gefunden



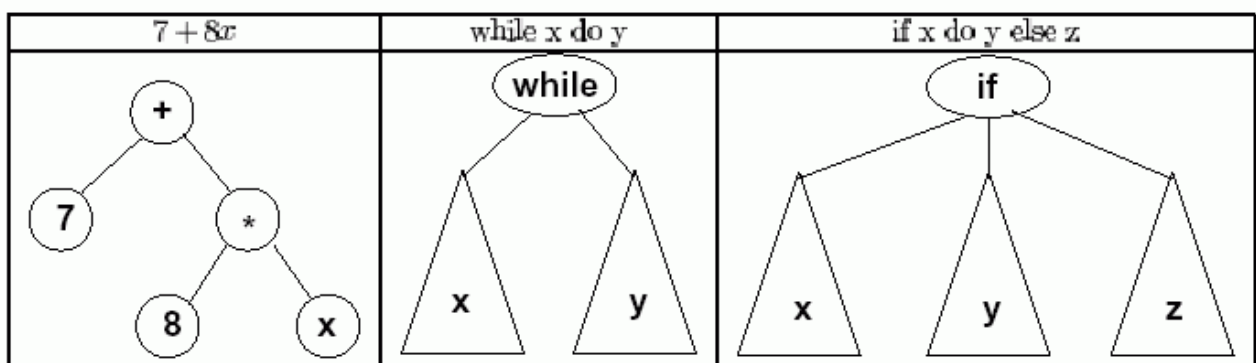
Evolutionäres Programmieren (EP)

- wie bei ES werden Vektoren anstelle von Strings evolviert
- kein crossing over
- Mutationen mittels normalverteilter Störterme
- Mutationsrate wird im Laufe eines Optimierungslaufes vermindert (entspr. Senkung der Temperatur)
- Verwandtschaft zum simulated annealing

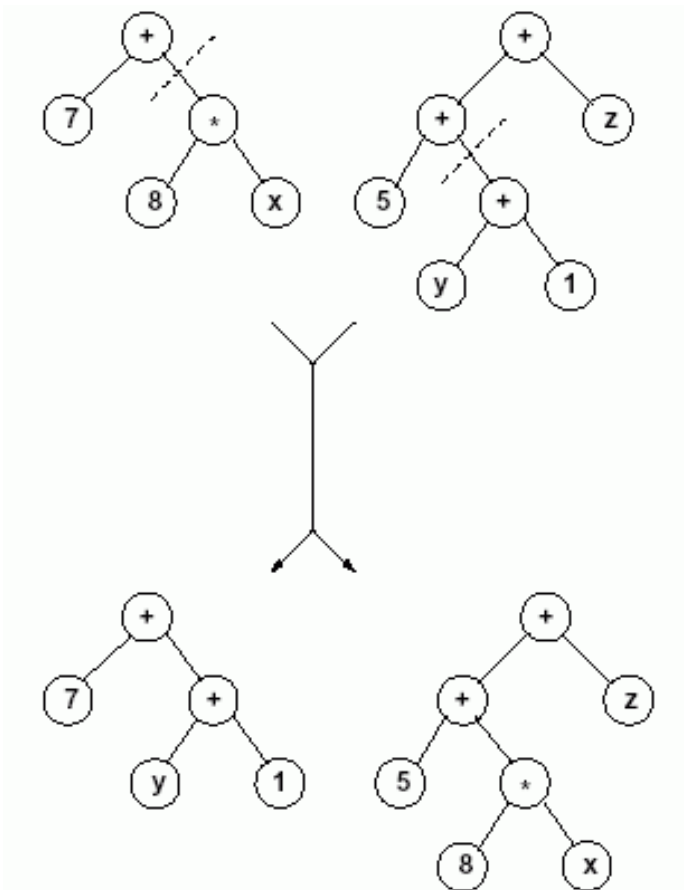
Genetisches Programmieren (GP; Koza 1987)

- komplette Computerprogramme werden evolviert
- Individuen = Terme (oft Lisp-Ausdrücke)
- Blattknoten = Konstanten, Variablen, Register oder externe Inputs
- Genome (Terme) haben variable Länge
- gegen Gefahr des unbegrenzten Wachstums: oft "Straf-Term" für Länge bei der Fitnessberechnung
- Koza verwendete keine Mutationen, nur crossing over!

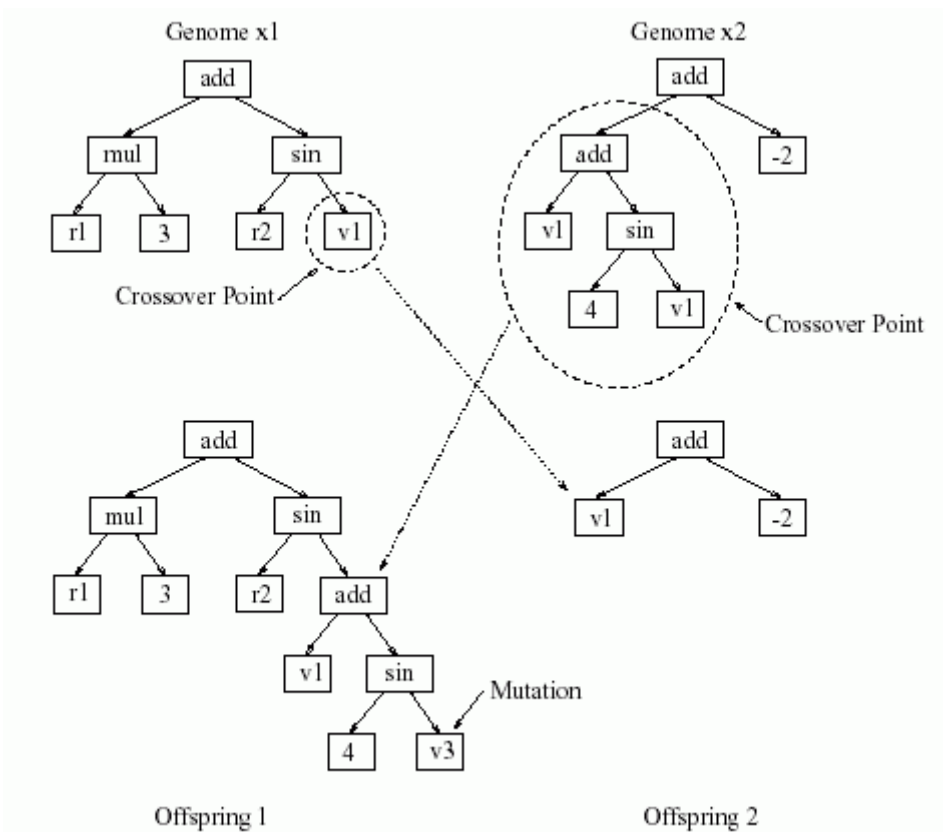
Beispiele für Terme:



Beim *crossing over* werden Subterme ausgetauscht:



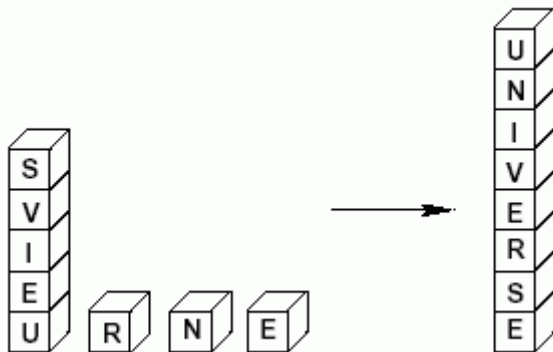
anderes Beispiel:



Anwendungsbeispiel für GP (aus Güde 2001):

wieder sollen Buchstaben so sortiert werden, dass ein vorgegebenes Wort entsteht

– aber diesmal evolviert nicht das Wort, sondern ein Programm, das Buchstaben von einem Vorrat ("Tisch") auf einen Stack legen kann (oder umgekehrt):



Befehlsvorrat:

Terminalzeichen:

CS	current stack
NN	next necessary

Operatorzeichen:

MS x	move to stack
MT x	move to table
EQ x y	equal
NOT x	not
DU x y	do y until x

Mit dieser Sprache wurde eine Zufallspopulation von Programmen erstellt und auf 166 verschiedene Buchstabenkonfigurationen angesetzt. Nach nur 10 Generationen gab es ein Programm, welches in der Lage war, alle 166 Aufgaben zu meistern:

(EQ (DU (MT CS) (NOT CS)) (DU (MS NN) (NOT NN)))

weitere Beispiele für Anwendungen evolutionärer Algorithmen:

Evolution von Formelausdrücken zur Approximation von Zahlenfolgen, deren Anfangsstück vorgegeben ist
(Neuse 2002)

- typisches KI-Problem
- die 200 ersten Werte der Folge werden vorgegeben
- Operatoren:
 - 0-stellig: ganzzahlige Konstanten 0 bis 255; e und π
 - 1-stellig: $-x$, $1/x$, \exp , \log , sqrt , Kopierbefehl
 - 2-stellig: $+$, $-$, $*$, $/$, Potenz
- unmittelbare und adressbezogene Interpretation von Speicherinhalten möglich (vgl. Core Wars)
- 16 Zellen für Zwischenergebnisse
- move- und copy-Bits zum "Umschreiben" des Programms
- fester Speicherbereich von 64 Drei-Byte-Befehlen (diese 64 Felder ergeben zusammen 1 Individuum) – in einfacheren Versionen auch 16 oder 32 Befehle
- binäre Codierung
- in jeder Generation Überprüfung der Übereinstimmung von Formelergbnis und vorgegebener Folge nur anhand von 5 zufällig gewählten Beispiel-Zahlen (aus den 200 vorgegebenen; Rechenzeitgründe!) – für alle Individuen einer Generation dieselben
- GA mit Populationsgröße 120
- Fitness: Minimum der Quotienten berechnete Zahl / vorgegebene Zahl und der inversen Quotienten
- Selektion: Kombination von Truncation (nur die bessere Hälfte der Population überlebt) und roulette-wheel-Selektion
- Mutation (bitweise) und Crossing over; Mutationsrate zwischen 10% und 30%
- Implementation auf PC

Beispiel: Fibonacci-Zahlen F_n :

1; 1; 2; 3; 5; 8; 13; 21; ... ; $2,80571173 \times 10^{41}$

- mögliches (ideales) Ergebnis wäre die Binet'sche Formel gewesen:

$$F_n = \frac{\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n}{\sqrt{5}}$$

- der GA war (bisher) nicht erfolgreich im Finden einer korrekten, exakten Formel (die extrapolationsfähig ist)
- bisher bestes Ergebnis nach ca. 40 000 Generationen:

$$f(x) = \frac{A(x)}{B(x)C(x)} \quad \text{mit}$$

$$A(x) = \sqrt{e^x} \cdot \pi^{\sqrt{x/71}/(x+\sqrt{x})}$$

$$B(x) = \frac{x}{71} + e^{\frac{x}{71}} + \frac{\left(\sqrt{\log x}\right)^{x/71}}{\pi}$$

$$C(x) = \sqrt{e^{x/71}} + \pi^{D(x)}$$

$$D(x) = \frac{\sqrt{\frac{x}{71}} \left(\frac{x}{71} \cdot e^{\frac{x}{71}-1} - x - \sqrt{x} \right)}{x + \sqrt{x}}$$

Abweichungen $f(n)/F_n$:

liegen zwischen 0,99 und 1,015 im gesamten Bereich von $n=8$ bis $n=200$ (Fitness bei 98,6%)

aber für $n=250$: 0,943, $n=400$: 0,632, $n=500$: ca. 10^{-5} !

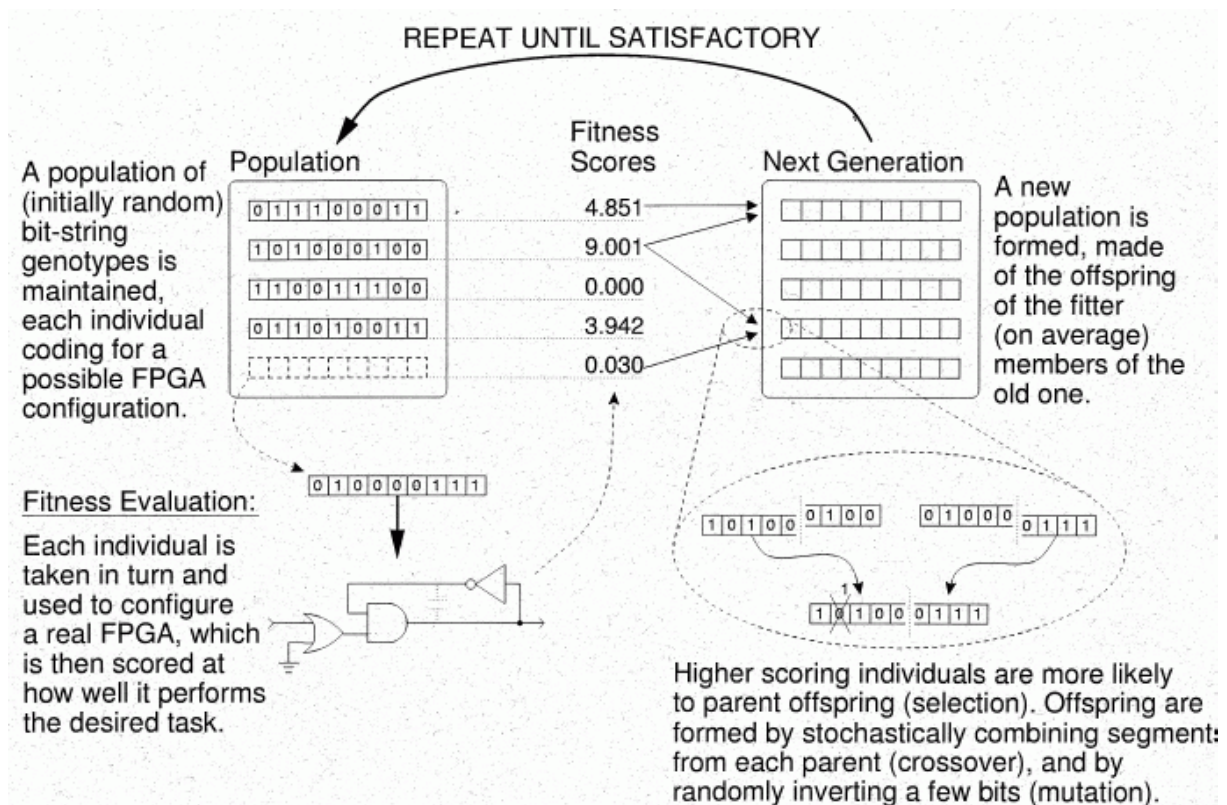
- d.h. die gefundene Approximationslösung gilt nicht mehr in einem Bereich, der über den während des GA getesteten Bereich (1–200) hinausgeht
- wenn man ϵ und π als Konstanten ausschließt, ist das Ergebnis (bisher) schlechter!

Evolution von Hardware (Adrian Thompson, zit. nach Pfeifer et al. 2002)

Konfiguration von FPGAs (*Field Programmable Gate Arrays*):

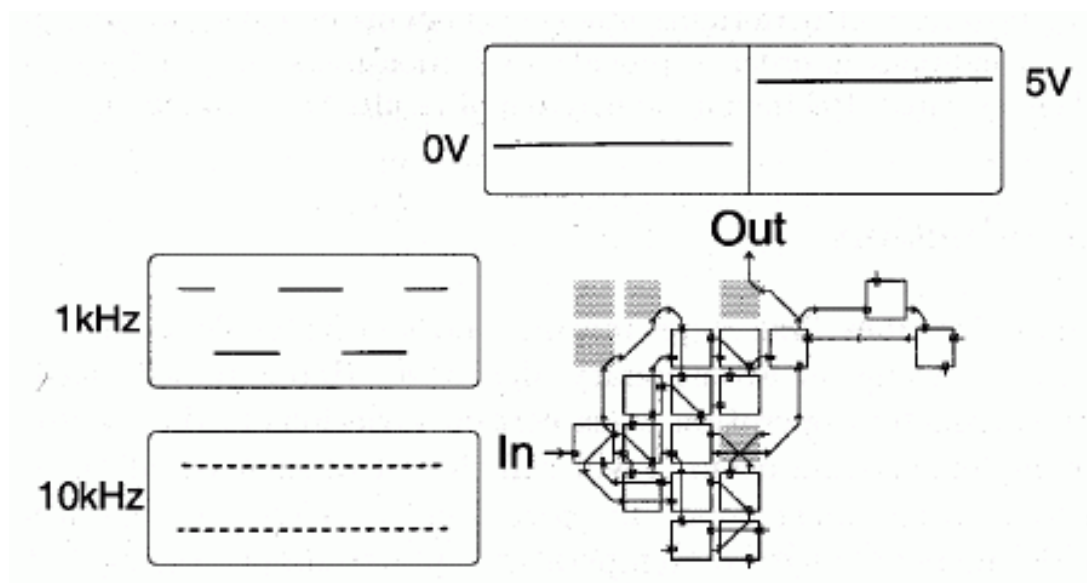
- VLSI-Chip mit vorgefertigten Komponenten, die durch konfigurierbare Schalter verbunden sind
- Schalterstellungen sind in "configuration memory" gespeichert
- dieser Speicher kann über angeschlossenen Rechner manipuliert werden
- jede Schalterstellung erzeugt einen neuen realen physikalischen Schaltkreis, dessen Eigenschaften mit Test-Software geprüft werden können (Fitness-Auswertung)

GA-Anwendung:



in Thompson's Beispiel war die Aufgabe des Schaltkreises, 1kHz- und 10kHz-Audiosignale (als Input) zu unterscheiden (Mustererkennung).

- Fitnessfunktion: Differenz der Ausgabespannungen bei diesen beiden Input-Signalen.
- keine synchronisierende Uhr eingesetzt!
- beliebige Eigenschaften der Silizium-Schaltkreise können ausgenutzt werden



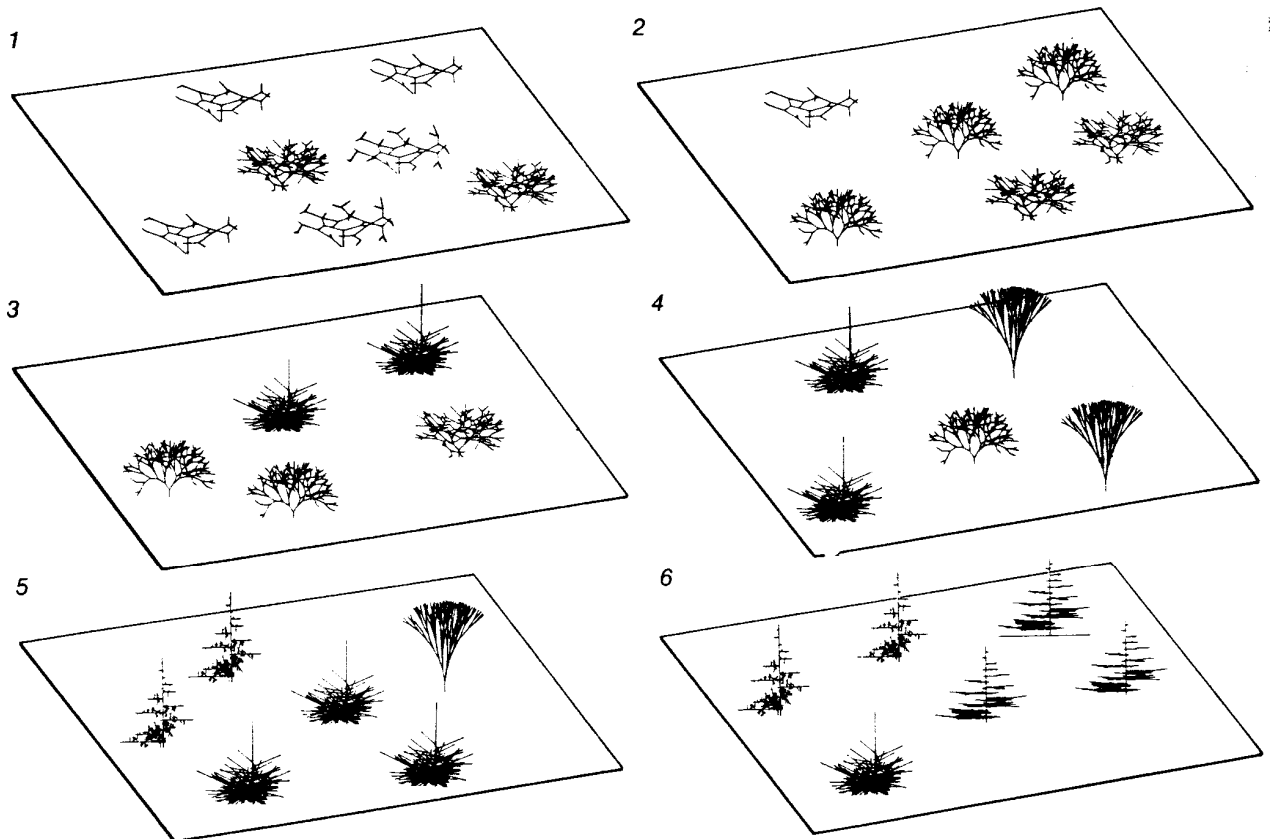
Ergebnisse:

- der GA ist erfolgreich
- perfektes Verhalten wird schon mit einem 10×10 -Array aus dem insgesamt zur Verfügung stehenden 64×64 -Array erreicht; konventioneller Schaltungsentwurf würde einen größeren Bereich beanspruchen
- innerhalb des 10×10 -Bereichs werden nicht alle Zellen benutzt; einige werden nur "indirekt" benutzt (ohne direkte Verbindung: graue Zellen in der Abb.) – müssen auf andere Weise mit dem Rest der Schaltung interagieren (elektromagnet. Kopplung)
- somit "unerwartete" Lösung; deutliche Unterschiede zu konventionellem Schaltungsentwurf

Anwendung von GA auf Morphogenese (vgl. Kap. 4)

Evolution der Landpflanzen (Niklas; vgl. oben)

selbes multikriterielles Optimierungsproblem, jetzt mittels GA gelöst (hier für Optimierung der Lichtinterzeption und Sporenausbreitung):



im Prinzip erhält man dieselben "adaptive walks" wie bei der direkten "hill climbing"-Methode

– bei niedrigdimensionalen Genomräumen und (relativ) "einfachen" Fitnesslandschaften bietet der GA keine Vorteile gegenüber konventionellen Optimierungsmethoden

Evolution von L-Systemen

bietet sich an, da schon symbolische Repräsentation der Morphologie vorhanden

Regelsystem = Genotyp,

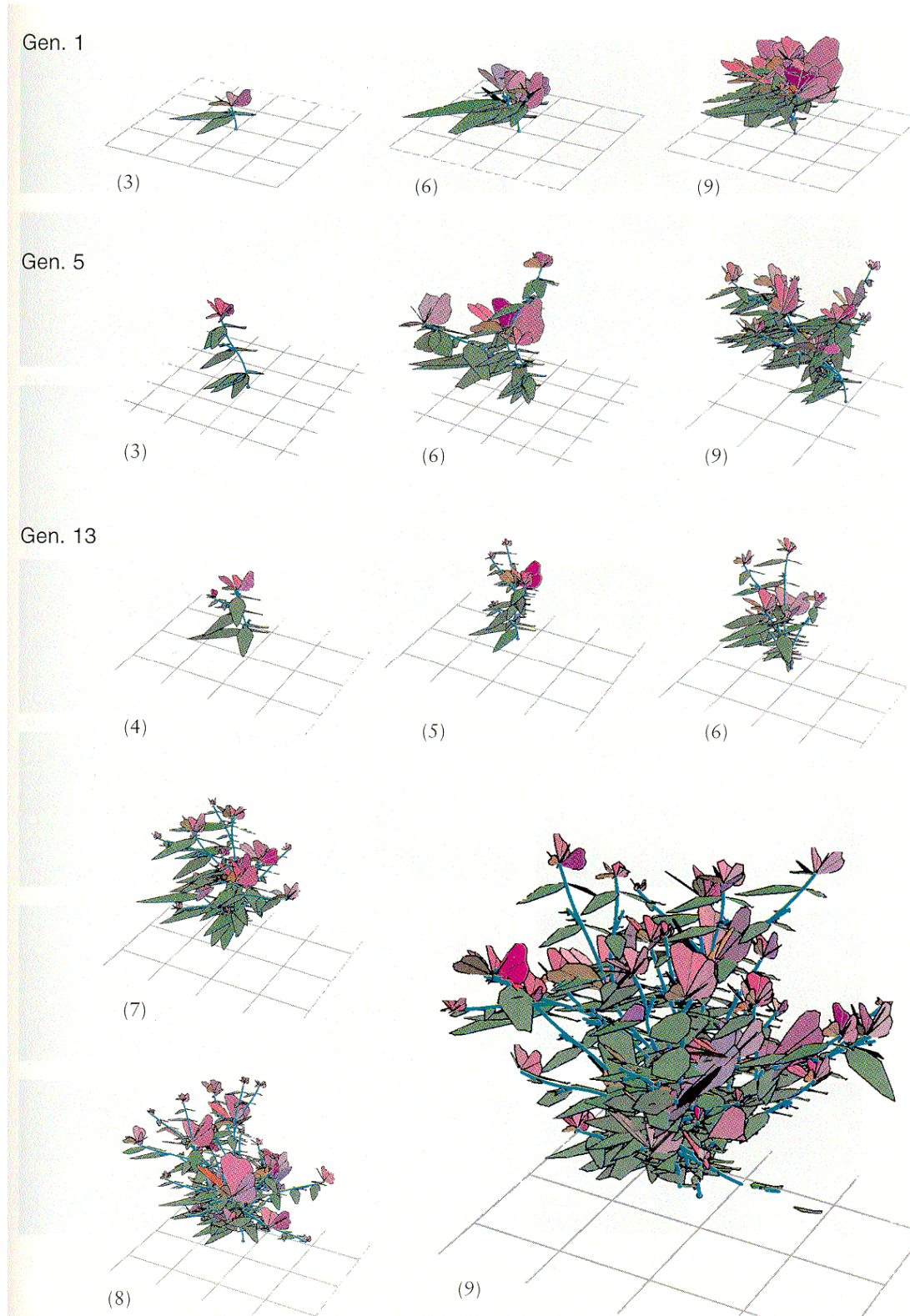
3D-Struktur (nach Turtle-Interpretation) = Phänotyp

Jacob 1997: ArtFlowers-System (mit Mathematica realisiert)

- Term-Repräsentation der L-Systeme
- Die L-Systeme sind die Individuen, die zugehörigen Phänotypen entstehen durch Anwendung von N Schritten und anschließender Turtle-Interpretation (nach jedem GA-Schritt für alle Individuen der Population!)
- Evolutionsoperatoren können sowohl komplette Regeln, als auch Sequenzen oder geklammerte Module innerhalb der rechten Regelseiten verändern
- mit Hilfe von "Schablonen" wird sichergestellt, dass nur Terme selektiert werden, für die die betreffenden Operatoren auch sinnvoll sind
- die Schablonen sind mit Gewichten versehen, so dass z.B. die Veränderung einer kompletten Regel seltener ist als die von Teilausdrücken der rechten Regelseite
- Evolutionsoperatoren: Mutation, crossing over, Deletion, Duplikation, Permutation
- Fitness (beispielsweise): Produkt aus Anzahl der Blüten und Volumen der bounding box der Pflanze nach vorgegebener Schrittzahl

Beispiel: Wachstumsschritte einiger Pflanzenstrukturen aus evolvierten L-Systemen

("Gen." bezeichnet die Generationszählung des GA, die eingeklammerte Ziffer die Zahl der abgearbeiteten Schritte des L-Systems)



(aus Jacob 1997)

Evolution von zellulären Automaten (Mitchell 1996)

Einfache Codierung als String: Output der Transitionsfunktion des CA wird für alle möglichen Inputkombinationen (z.B. in lexikografischer Ordnung) hintereinandergeschrieben

Nichttriviale Aufgabenstellung:

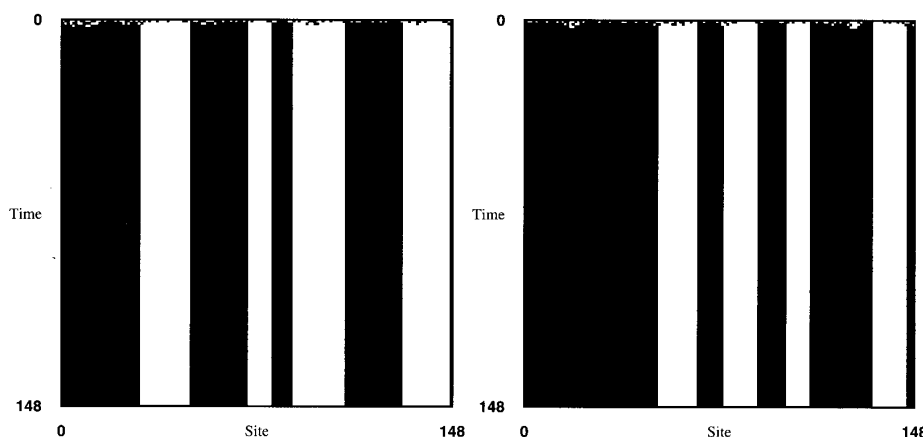
es sollen 1-dim. binäre CA (mit festem Radius r , auf ringförmigem Band fester, ungerader Länge) gefunden werden, die zu einer gegebenen Zahl M folgendes leisten:

- wenn die durchschnittliche Dichte ρ der Einsen in der Anfangskonfiguration kleiner als $\frac{1}{2}$ ist, liefert der CA nach M Schritten ein Band, was mit lauter Nullen besetzt ist
- wenn $\rho > \frac{1}{2}$ ist, ein Band mit lauter Einsen

(" $\rho_c = \frac{1}{2}$ - Aufgabe"; $\rho_c =$ "kritische Dichte")

Beachte: die Aufgabe verlangt Informationsfluss über "große" Distanzen (im Vergleich zum Radius r der Nachbarschaft), da die Dichte der Einsen in der Anfangskonfiguration großräumig unausgewogen sein kann.

Eine einfache "Mehrheitsregel" (Ergebnis der Transitionsfunktion = 1, wenn die Mehrheit der Zellen in der Nachbarschaft Wert 1 hat, sonst 0) erfüllt die Aufgabe *nicht*.



Verhalten eines 1-dim. CA mit Mehrheitsregel, $r = 3$, Zeitverlauf von oben nach unten. Linke Startkonfiguration mit $\rho < \frac{1}{2}$, rechte mit $\rho > \frac{1}{2}$.

GA-Anwendung:

- Bandlänge $N = 149$
- Populationsgröße 100
- Initialisierung: 100 zufällig gewählte CA-Regeln
- jedes Individuum (CA-Regel) wird an 100 Startkonfigurationen getestet (Test *aller* möglichen 2^{149} Startkonfigurationen wäre unmöglich!)
- die Startkonfigurationen werden zufällig bestimmt, aber mit einem Algorithmus, der uniforme Verteilung der Einsendichte im Intervall $[0; 1]$ sicherstellt (völlig zufällige Auswahl der Startkonfigurationen würde "zu schwere" Testfälle liefern, da die meisten dann $\rho \approx 1/2$ hätten).
- $M = 2N$
- Fitness einer Regel: Anteil der 100 Startkonfigurationen, für die sie das korrekte Endmuster liefert ("teilweise richtige" Lösungen werden nicht mitgezählt)
- Selektion: "Elite-Auswahl" (beste 20%) + crossover aus der Elite (80%) mit anschließender doppelter Mutation
- GA mit 100 Generationen
- Bestimmung der Fitness ist rechenintensivster Teil
- 300 GA-Läufe mit unterschiedlichen Zufallszahlen

Ergebnisse:

meistens evolvieren nichttriviale, aber relativ unkomplizierte Regeln

z.B. ϕ_a : liefere Band mit Nullen, wenn nicht ein genügend langer Block aufeinanderfolgender Einsen in der Startkonfiguration vorliegt. Ein solcher wird expandiert und liefert ein Band mit nur Einsen. (Fitness $\approx 0,9$)

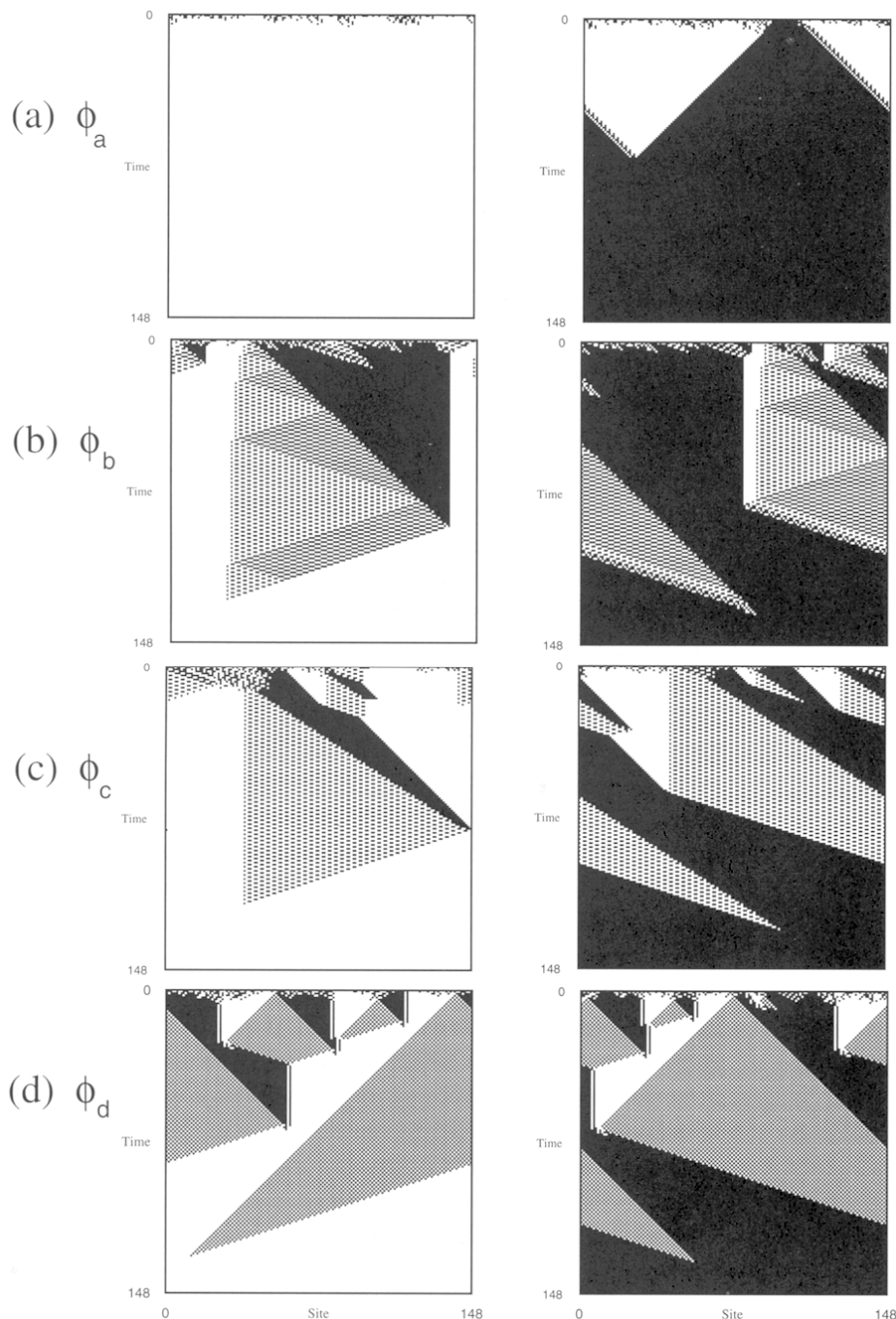
– funktioniert "meistens", weil bei hoher Einsen-Dichte meistens auch ein genügend langer Block auftritt (aber dies ist nicht garantiert).

- diese "block-expanding"-Strategien waren für die Autoren des GA überraschend
- führen aber nur lokale Berechnungen durch, kein Fern-Transfer von Information, der erhofft war
- Fitness wird schlecht bei Einsen-Dichte nah bei 0,5

in einigen Läufen evolvierten bessere Regeln mit "komplexeren" Strategien

- "Partikel-basierte" Regeln mit Signalweiterleitung über größere Distanzen
- keine Regel schafft eine Fitness von 100%

4 Beispiele für die Performance evolvierter CA-Regeln (linkes Diagramm startet jeweils mit $\rho < 1/2$, rechtes mit $\rho > 1/2$) (das graue Gebiet in (d) ist ein Schachbrettmuster aus Nullen und Einsen) – die Fitness wächst von (a) bis (d):

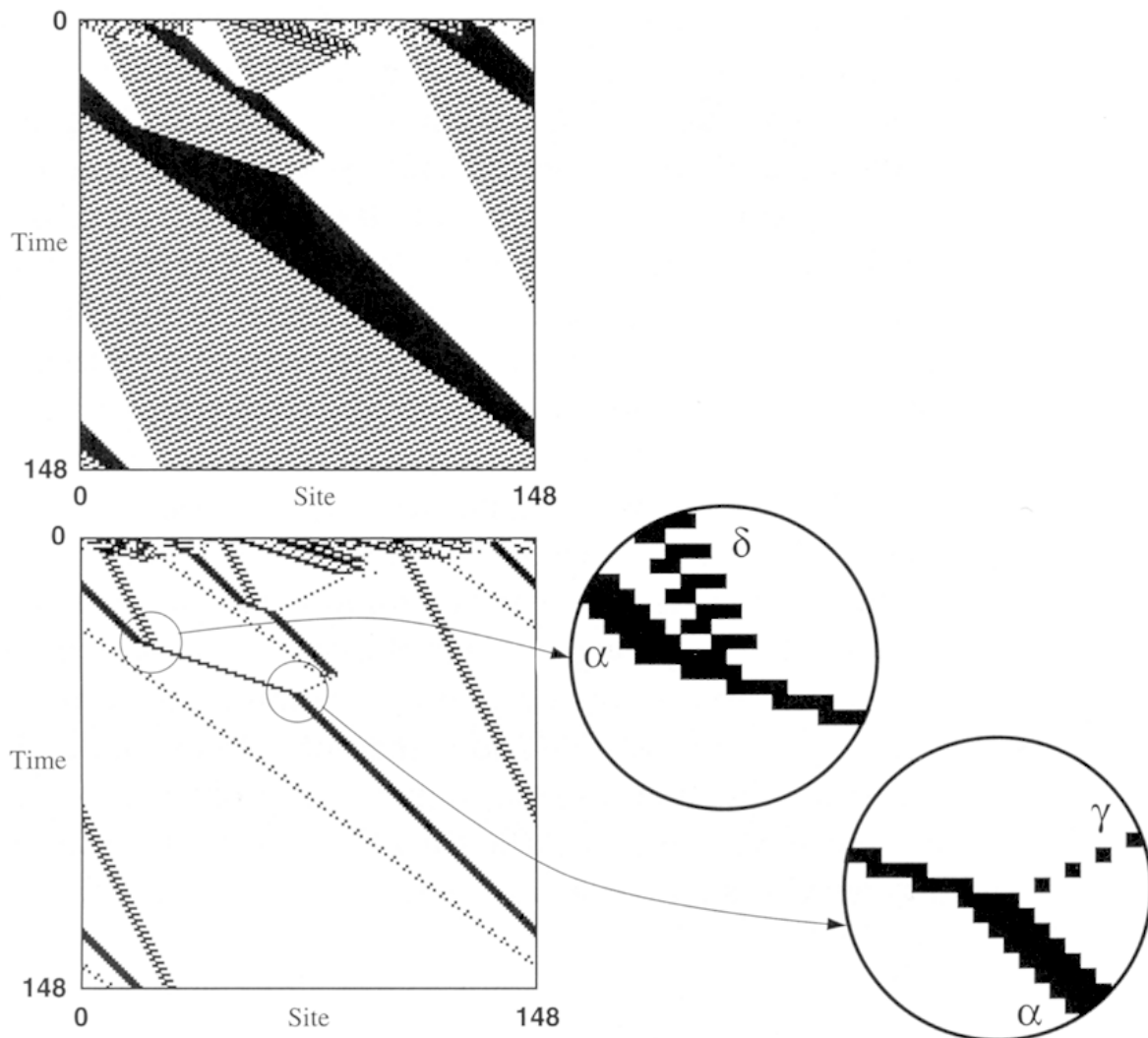


Analyse solcher CA-Regeln:

"Computational mechanics" (Hanson & Crutchfield), betrachtet "reguläre Bereiche" (mit Mustern aus Wörtern einer regulären Sprache), deren Grenzen = "Partikel" = Informationsträger, Kollisionen zwischen Partikeln bewirken Informationsverarbeitung

- Partikel und ihre Interaktionen als "high-level"-Sprache zur Beschreibung von Berechnungen in räumlich ausgedehnten, homogenen Systemen wie CAs.

Analyse eines evolvierten CA durch Herausfiltern aller regulären Bereiche und Betrachtung der Partikel:



Karl Sims' Virtual Creatures

(SIGGRAPH 1994 und Artificial Life IV - Konferenz 1994)

Ziel:

Es sollen 3D-Individuen entwickelt werden, die in einer virtuellen Welt mit simulierter Physik (Gravitation, Reibung, Kollisionserkennung, Flüssigkeitsdynamik...) spezielle Aufgaben besonders gut bewältigen können

- Struktur (Morphogenese) und Verhalten (Regeln, Kontrollstrukturen: neuronales Netz) sind durch den genetischen Algorithmus variierbar

Kodierung der Kreaturen: Gestalt

Genotyp: Strukturbeschreibung

Phänotyp: Ausprägung

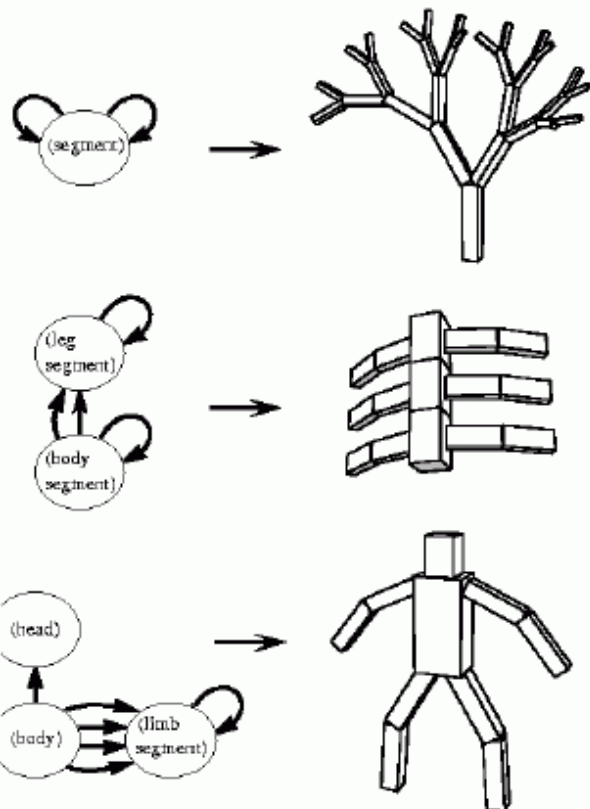
Die Erbinformationen über die Gestalt der Lebewesen wird als *gerichteter Graph* gespeichert.

Die **Knoten** entsprechen je einem **Körperteil** und enthalten folgende Informationen:

- Größe
- Gelenktyp
- Rekursionslimit
- Neuronen
- Verbindungen

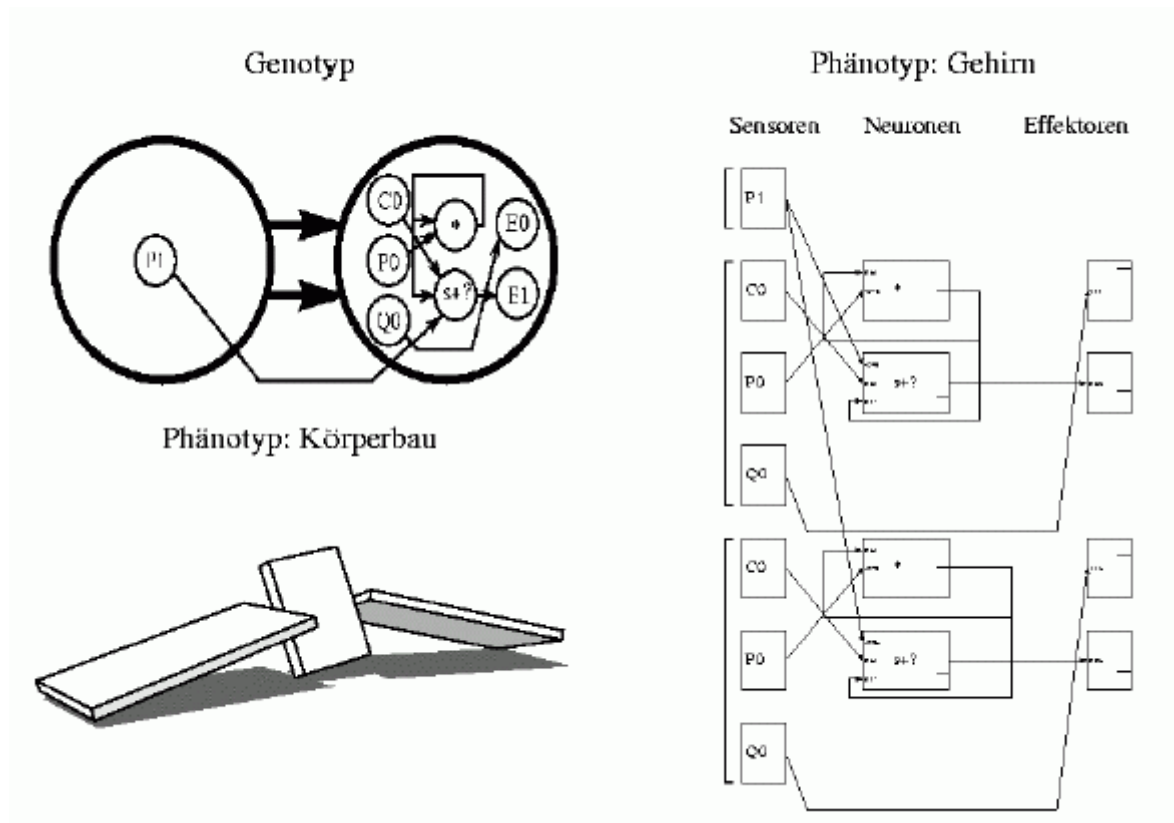
Die **Kanten** entsprechen den **Gelenken** und enthalten folgende Informationen:

- Position
- Orientierung
- Skalierungsfaktor
- Nur-Ende-Flag



Aufbau einer Kreatur:

- segmentiertes System mit Gelenken
- Winkelsensoren und Effektoren für jedes Gelenk
- neuronale Schaltkreise in jedem Segment + "Gehirn" (neuronales Netz)



Codierung durch Graphen:

- "äußerer Graph": Anordnung der Segmente (Glieder) und Gelenke
- "innerer Graph": Verschaltung der Sensoren, Neuronen und Effektoren

Genotyp enthält auch Informationen über Sensoren, Effektoren und Neuronen

Evolution

Ein Genotyp wird in folgenden Schritten mutiert:

1. Interne Parameter der Knoten werden z. T. verändert.
2. Ein neuer Knoten wird hinzugefügt.
3. Parameter der Kanten werden z. T. verändert; evtl. zeigt die Kante anschließend auf einen anderen Knoten.
4. Neue Kanten werden evtl. hinzugefügt.
5. Nicht (mehr) befestigte Knoten werden entfernt

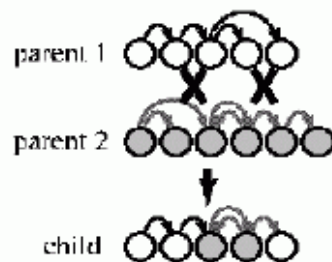
Das Verfahren besteht aus zwei Phasen:

Zuerst wird der äußere Graph bearbeitet, dann der innere.

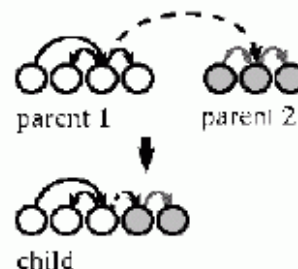
Methoden:

1. [40%] Kopieren des Genotyps; dabei treten **Mutationen** auf
2. [30%] **Crossover** (evtl. mit Mutationen)
3. [30%] **Grafting** (evtl. mit Mutationen)

a. Crossovers:



b. Grafting:



Verwendung von gerichteten Graphen erfordert spezielle Verfahren für "Crossover" (Variante: "Grafting" = Pfropfung)

- Selektion: "truncation", d.h. nur die besten 20 % einer Generation überleben für die Reproduktion
- zusätzlich ist die Anzahl der Nachkommen proportional zur Fitness für die spezifische, vorgegebene Aufgabe

Der beschriebene genetische Algorithmus wurde angewendet, um Lebewesen zu generieren, die eine der folgende Aufgaben besonders gut beherrschen:

1. **Schwimmen**
2. **Fortbewegen an Land**
3. **Springen**
4. **Folgen einer Lichtquelle**
5. Sich im **Wettkampf** gegen ein anderes Individuum durchsetzen

Zur Implementierung:

- Populationsgröße: 300
- jeweils 50 - 100 Generationen pro Evolution
- jeweils max. 10 sec. pro Individuum und Generation zur Qualitätsbewertung
- **Laufzeit:** 3 - 4 Std. pro Evolution mit 100 Generationen auf einer CM-5 mit 32 Prozessoren

weitere Aufgabe: Kontrolle über einen Würfel gewinnen (und um diesen mit anderen Kreaturen konkurrieren) –
Fitnessfunktion: Nähe zum Würfel am Ende einer festgelegten Zeitspanne

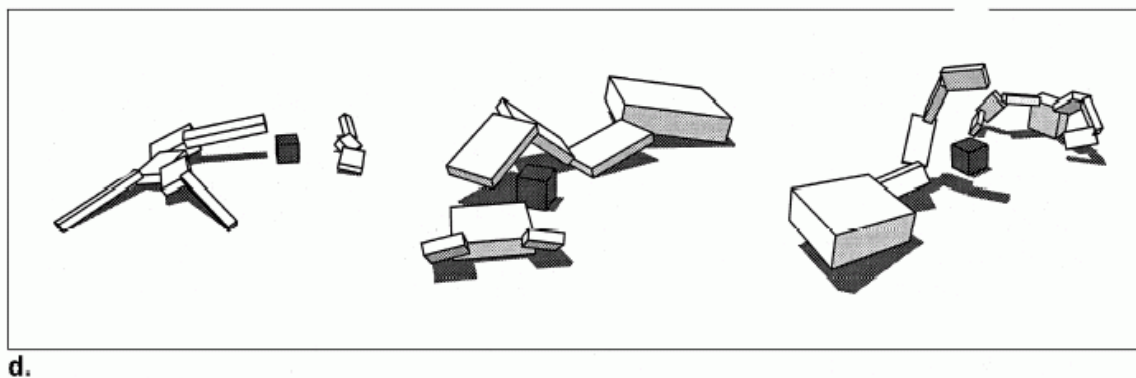
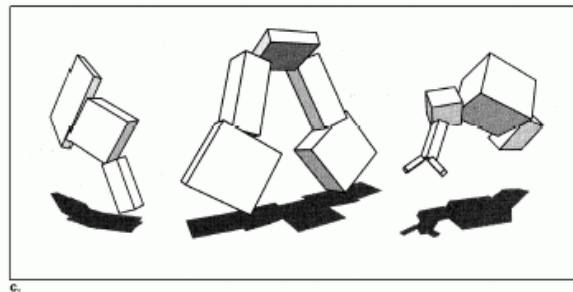
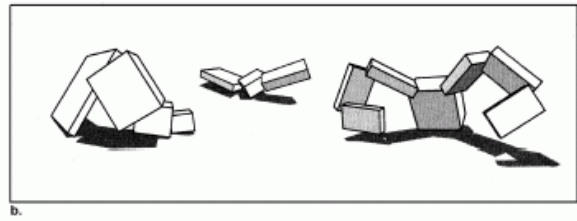
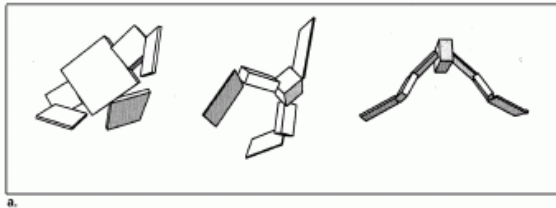
Ergebnisse:

oben links: Aufgabe "Schwimmen"

oben rechts: Aufgabe "Laufen"

Mitte: Aufgabe "Springen"

unten: Aufgabe "Um Kontrolle über Würfel konkurrieren"



Beobachtungen:

- auch hier "Überraschungseffekte", innovative Lösungen (z.B. Fortbewegung durch Sich-überschlagen)
- darunter auch Lösungen, die in der Natur nicht vorkommen
- hohe Rechenzeit-Anforderungen aufgrund von sehr großem Genom-Raum
- je mehr Einschränkungen man auferlegt, desto weniger Rechenzeit ist nötig, aber auch desto weniger Innovation

das Grundprinzip von Sims' "Creatures" wurde später in einem gleichnamigen Computerspiel weiterverarbeitet.

Stephen (Steve) Grand 1996, "Creatures" (Cyberlife Research Ltd.)

http://www.gamewaredevelopment.co.uk/creatures_index.php



Genetischer Algorithmus mit *intrinsicischer Adaptation*

(der Fitnesswert eines Genoms ist nicht schon durch das Genom eindeutig festgelegt, sondern hängt auch von der übrigen Population und von räumlichen und zeitlichen Interaktionen zwischen den Individuen ab):

LindEvol (Jan Kim 1996)

ALife-Experimente zur Simulation der Evolution von Pflanzen
– verschiedene Varianten

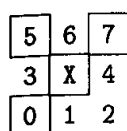
Grundideen:

- physikalische Komponenten des Modells: Raum (= 2-dim. oder 3-dim. Gitter), Zeit (diskrete Schritte), Energie ("Photonen", kommen von oben; Aktionen einer Pflanzenzelle nur nach Absorption eines Photons möglich)
- genetische Komponenten: Gene, Genome, Genregulation, Mutationen (kein crossing over)
- ökologische Komponenten: Interaktion zwischen Individuen, Konkurrenz um Ressourcen "Energie" und "Raum"

2-dim. Struktur: endlich ausgedehntes, zylindrisch geschlossenes Gitter ("Welt" mit Boden und Decke) mit quadratischen Zellen, Moore-Nachbarschaft

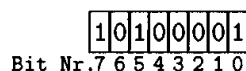
- eine Pflanzenzelle besetzt genau eine Gitterposition, und eine Gitterposition kann von höchstens einer Pflanzenzelle besetzt sein
- einem Pflanzen-Individuum sind endlich viele Pflanzenzellen zugeordnet
- interne Parameter einer Pflanzenzelle: Energieparameter 0 oder 1, bei einigen Varianten zusätzlich 16- oder 32-bit interner Zustand
- lokale Struktur einer Zelle: Belegung der Nachbarzellen (Codierung von 2-dim. lok. Strukturen in 8-bit-Zahlen)

lokale Struktur:



→

Achtbitzahl:



Pflanzenzellen können *Aktionen* ausführen, von denen die meisten Energie verbrauchen (d.h. nur nach Absorption eines Photons möglich sind):

- Generierung einer energielosen Nachbarzelle (Index der zu besetzenden Nachbarzelle kann durch Parameter spezifiziert werden)
- "flyingseed": neue Pflanze (aus einzelner, energieloser Zelle bestehend) wird an einer zufällig ausgewählten freien Gitterposition am Boden der "Welt" generiert
- "localseed": neue Pflanze wird senkrecht unter der Mutterpflanzenzelle generiert
- Exponent zur Bestimmung der Mutationsrate wird um 1 erhöht oder vermindert
- einzelnes Bit des internen Zellzustands für den nächsten Zeitschritt wird gesetzt (verbraucht keine Energie)

Photonen wandern vertikal abwärts und werden von Pflanzenzellen mit 50% Wahrscheinlichkeit absorbiert

⇒ Lichteinstrahlung ist die Energiequelle, die alle "Lebensvorgänge" antreibt

Eine *Pflanze* enthält neben ihren Zellen

- ein *Genom*: eine Menge von Regeln "Zustand → Aktion", die als Binärstrings codiert werden (am Anfang des Simulationslaufs wird Menge von Genomen zufallsgeneriert)
- einen Mutationsexponenten a (Mutationsrate = Basisrate des Mutationstyps $\times m^a$, $m = \text{Konstante}$)
- Information über Gesamtzahl ihrer Zellen und Gesamtzahl ihrer energiereichen Zellen ("Gesamtenergie" der Pflanze)

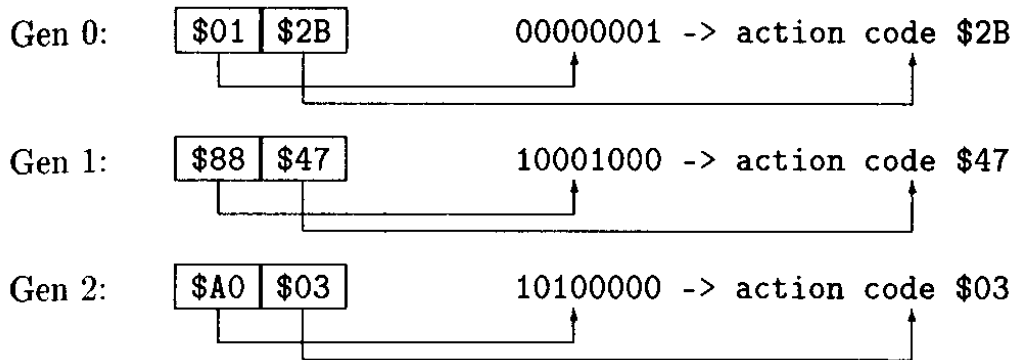
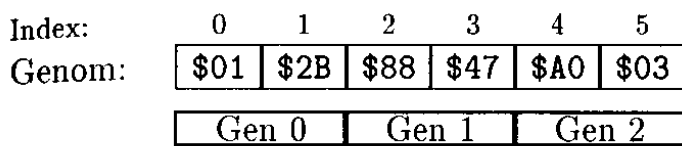
Eine im Genom enthaltene Regel (ein "Gen") wird *aktiviert*, wenn der Zustand einer Zelle (einschließlich der Nachbarschaftssituation) zur linken Regelseite "passt"

- für energieverbrauchende Aktionen muss die Zelle zusätzlich gerade ein Photon absorbiert haben
- von mehreren aktivierten Genen für energieverbrauchende Aktionen wird nur das erste im Genom ausgeführt

⇒ durch Generierung einer Nachbarzelle können bei anderen Zellen und bei der Mutterzelle selbst Gene "ein-" und "ausgeschaltet" werden, da sich die lokale Umgebung ändert (es entsteht ein regulatorisches Netzwerk)

Blockorientierte Genominterpretation:

jedes Gen besteht aus 2 aufeinanderfolgenden Bytes im Genom, die linke und rechte Regelseite codieren

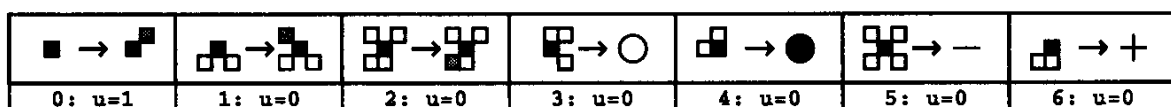


Darstellung eines Genoms (Beispiel):

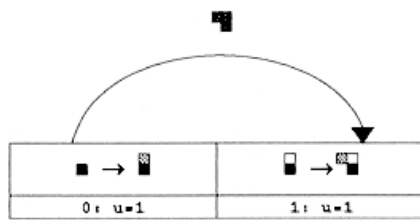
Listing des Genoms:

0	[1]	: 00000000 -> divide 7	00 07
1	[0]	: 00000101 -> divide 5	05 05
2	[0]	: 10100011 -> divide 0	a3 b8
3	[0]	: 11000100 -> flying seed	c4 c0
4	[0]	: 01001000 -> local seed	48 d7
5	[0]	: 10100101 -> mut-	a5 eb
6	[0]	: 00000011 -> mut+	03 f4

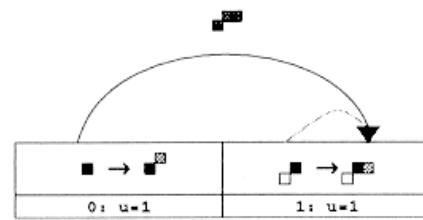
Graphische Darstellung des Genoms:



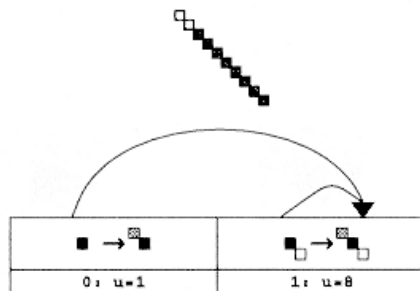
Beispiele für Pflanzen mit blockorientierter Genominterpretation, mit ihren regulatorischen Netzwerken:



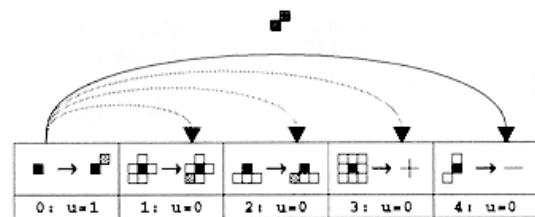
(a)



(b)



(c)



(d)

(energiereiche Zellen sind schwarz gezeichnet)

Mutationsoperatoren:

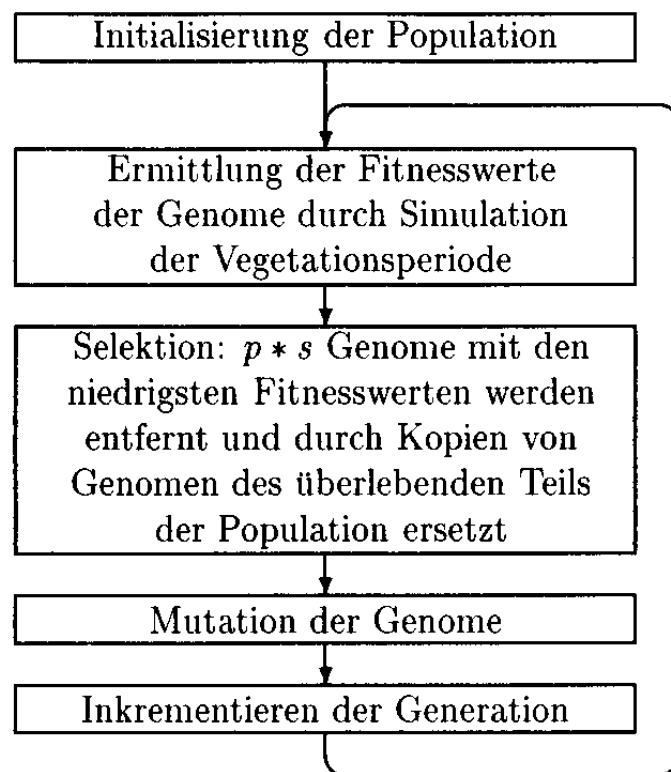
- Änderung (bytweweise oder bitweise)
- Insertion
- Deletion
- Genduplikation

1. Variante: "*LindEvol-GA*":

- 2-dim. Gitter
- keine zusätzlichen Zustandsparameter der Zellen
- keine samenproduzierenden Zellaktionen, Vermehrung von Pflanzen erfolgt allein durch den Selektionsmechanismus eines GA
- blockorientierte Genominterpretation
- keine bitweisen Mutationen, keine Duplikationen

eine Generation von Pflanzen wächst jeweils eine "Vegetationsperiode" lang gemeinsam in der Welt, diese besteht aus fester Zahl (`num_days`) von Wachstumsschritten ("Tagen").

- Startzustand jeder Vegetationsperiode: zufällig auf dem Boden verteilte Pflanzen, die nur aus 1 energielosen Zelle bestehen
- Fitnesswert einer Pflanze: Anzahl der energiereichen Zellen nach Ablauf der Vegetationsperiode



(p = Populationsgröße, s = Selektionsrate)

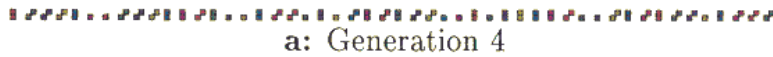
Typische Parameter:

Populationsgröße: 50
Mutationsfaktor $m = 2$
Breite der Welt: 150
Höhe der Welt: 30
Genomlänge der Startpopulation: 20 Gene (40 Bytes)
Länge einer Vegetationsperiode: 30 Tage

Anzahl Generationen: 5000
Anzahl Aktionscodes für Wachstums-Aktionen: 224
Anzahl Aktionscodes für Mutationsratenveränderung: je 16
Mutationsrate: 0,01 (= "moderat")
Selektionsrate: 0,5 (= "moderat")

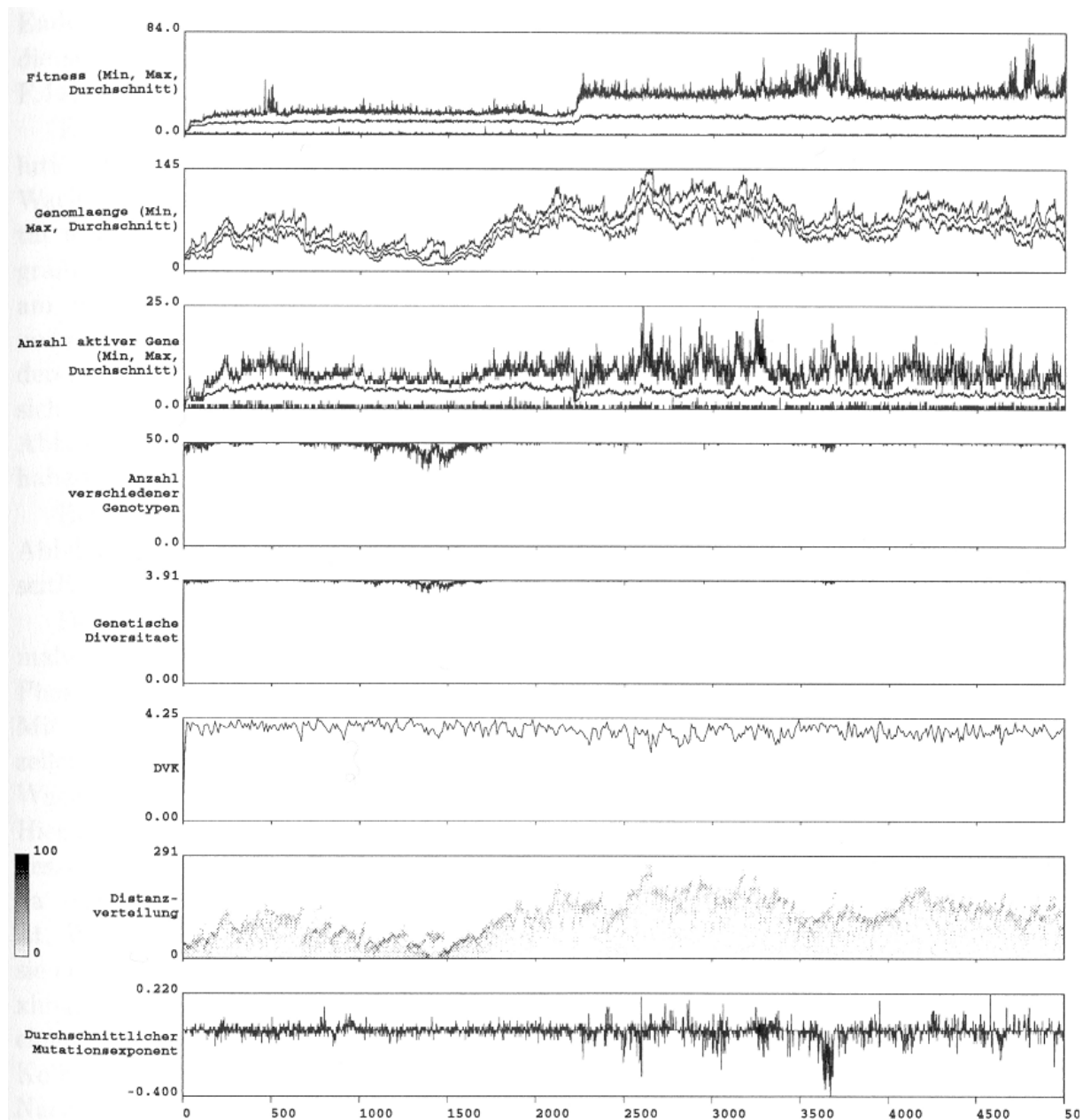
Ergebnis eines typischen LindEvol-GA-Laufs:

(jeweils Darstellung der gesamten "Welt" am Ende der Vegetationsperiode, verschiedene Pflanzen durch verschiedene Farben markiert)



verschiedene "Phasen" der Evolution an Morphologien und Fitnesswerten erkennbar

statistische Übersicht des Simulationslaufs:



Anfangsphase (Generation < 30): kleine Pflanzen niedriger Fitness, genetisch festgelegte Größe

bei Gen. 30: Übergang zu unbeschränktem Wachstum ("evolutionärer Sprung")

Gen. 100: "Erfindung" von kleinen Seitenzweigen

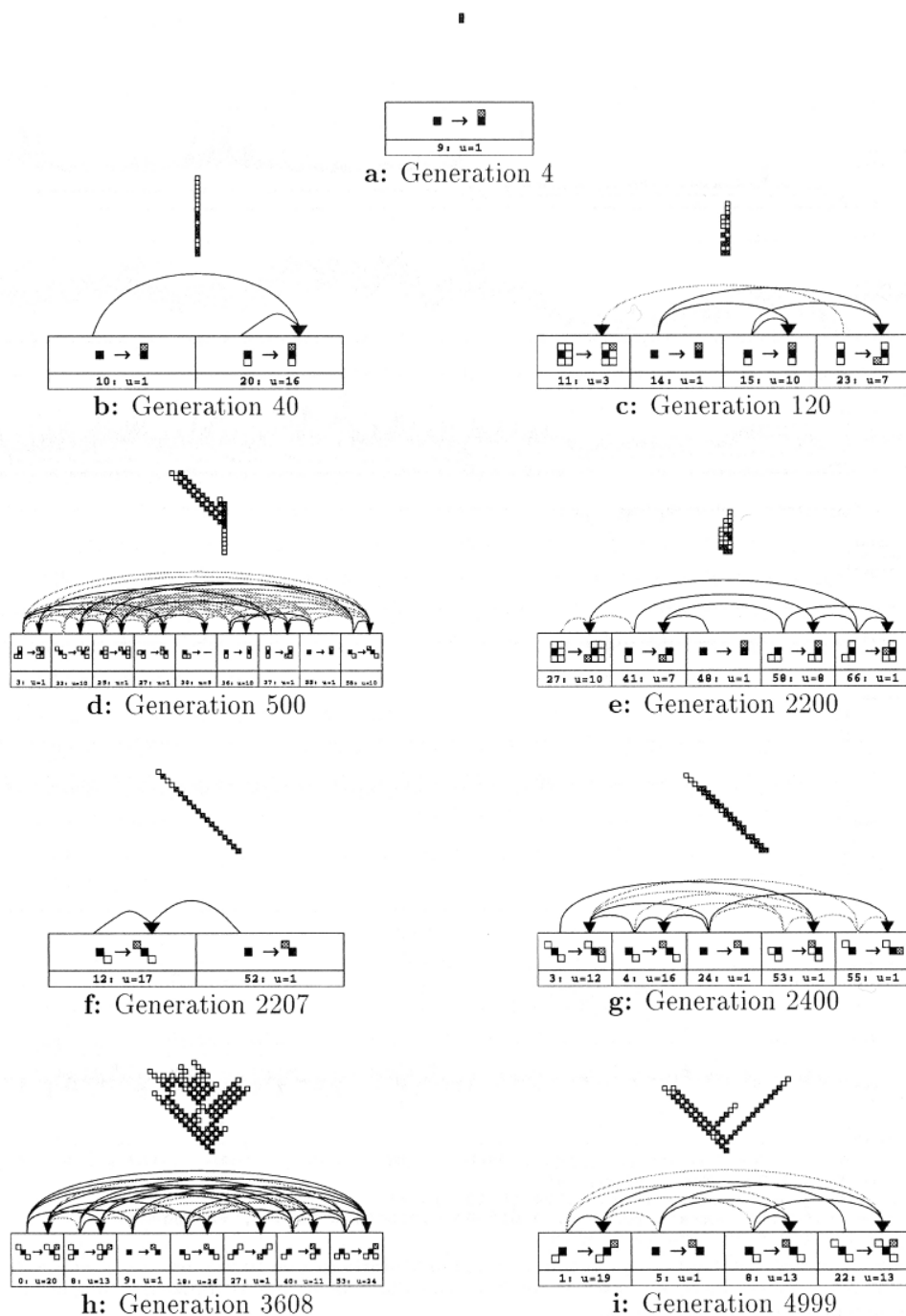
Gen. 500: unbeschränkt wachsende diagonale Seitenzweige treten auf (aber: → räuml. Konkurrenz), verschwinden wieder

Gen. 2200: grundlegende Bauplanänderung: diagonaler Hauptspross, setzt sich durch (Anzahl der Zellen, die an einem Tag ein Photon absorbieren können, ist größer als bei vertikal wachsenden Pflanzen)

Gen. 2400: Verdickung des Hauptsprosses

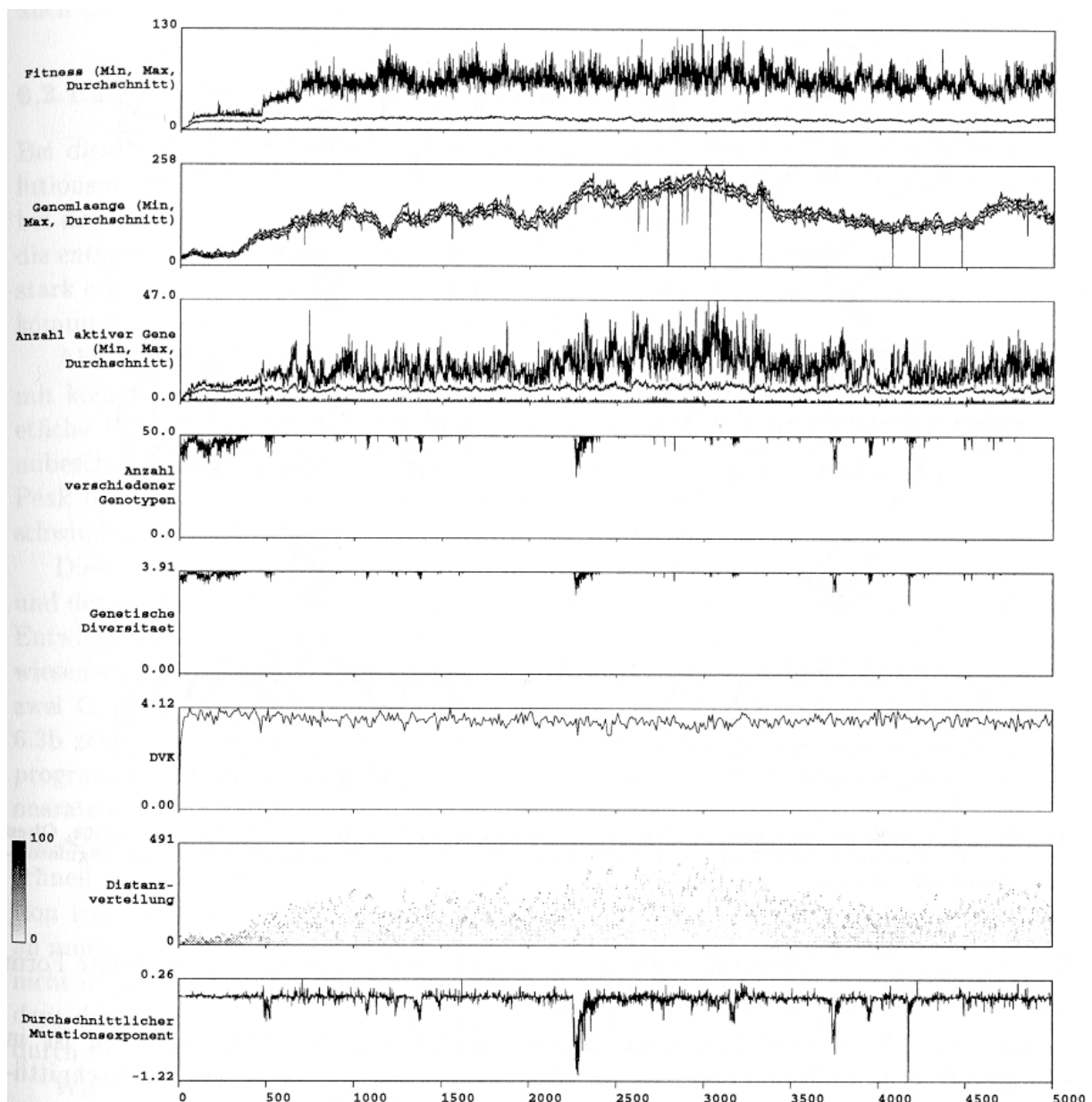
Gen. 3500-3800: buschförmige Pflanzen mit vielen diagonalen Seitentrieben, verschwinden wieder

einige für die Phasen charakteristische Regelungsnetzwerke:



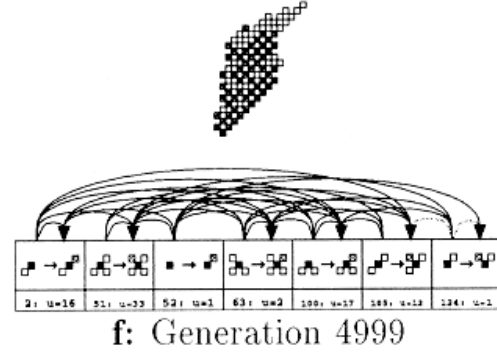
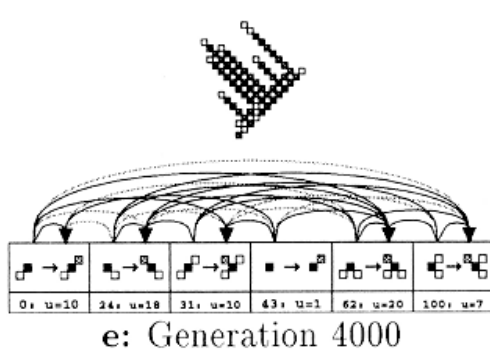
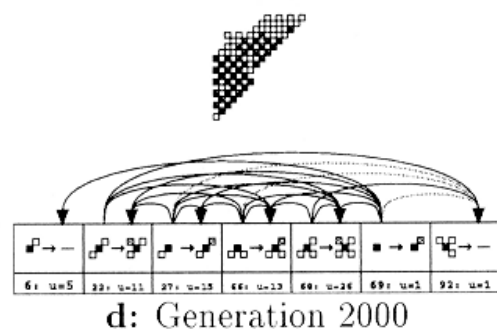
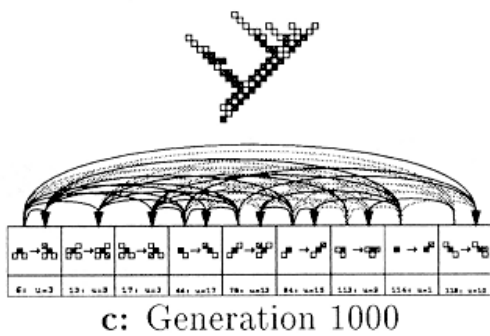
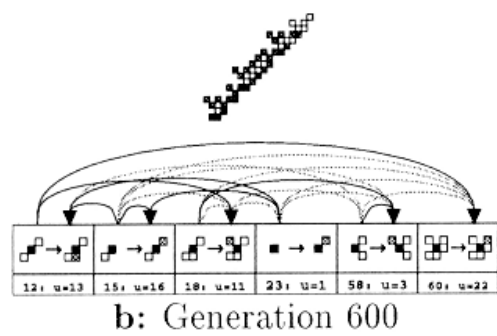
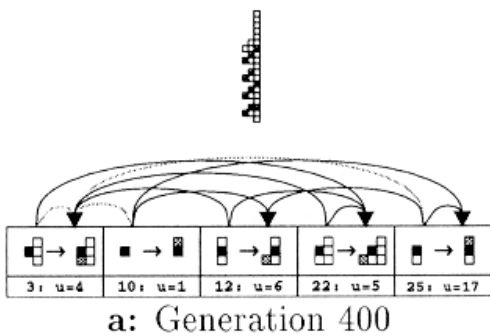
- es kommt zu einer Zunahme der Fitness und der morphologischen Komplexität
- breite Distanzverteilungen der Genome
- aber keine größere Distanzverteilungskomplexität als bei einem Kontrollversuch mit "neutraler Evolution" (= keine Heranziehung der Fitnesswerte bei der Selektion)
- Mutationsraten werden lediglich in einer kurzen Episode aktiv verändert

Vergleichslauf mit "scharfer Selektion" (Selektionsrate 0,8):



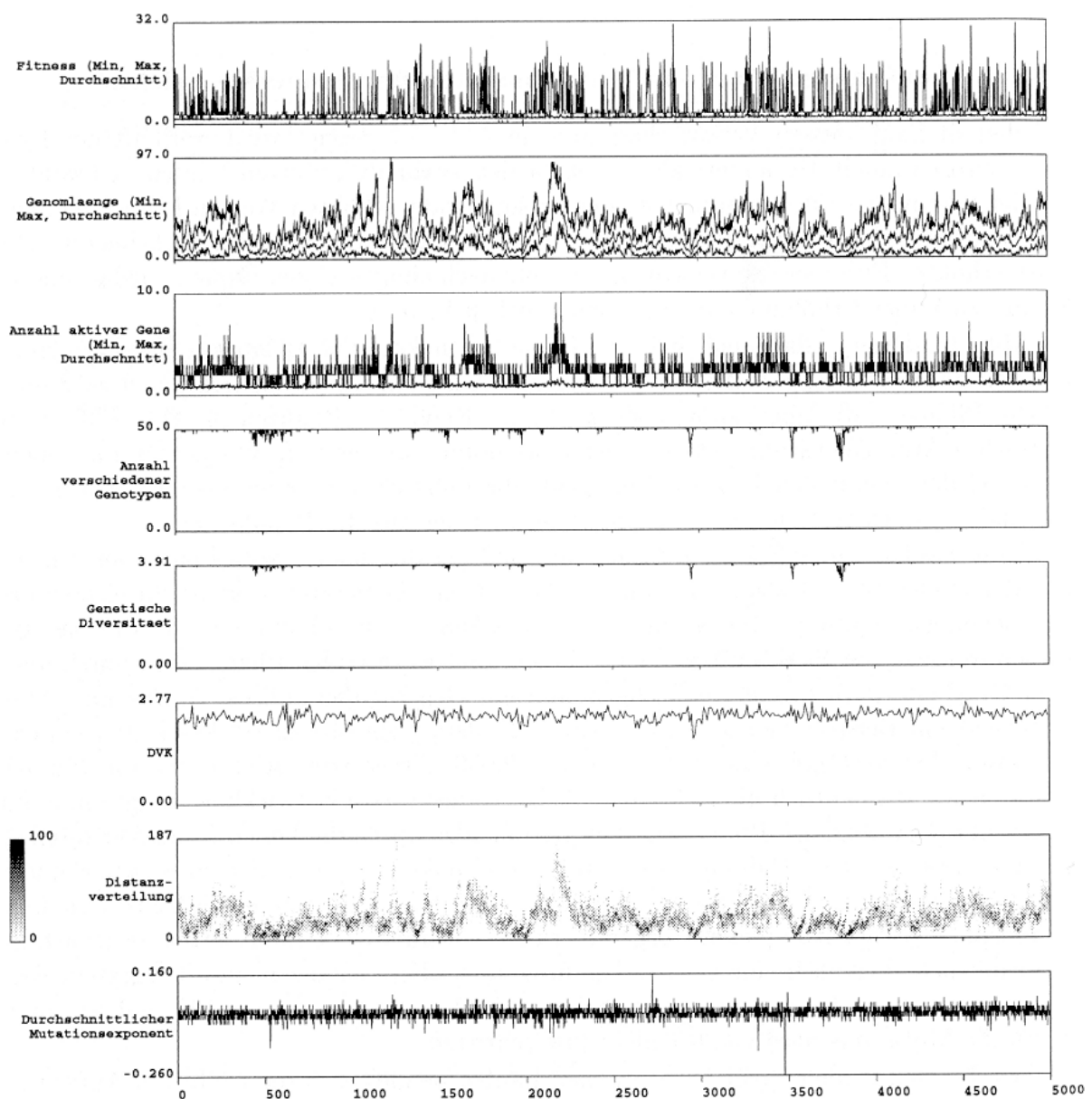
- ähnliche "Evolutionssprünge" wie im Lauf mit moderater Selektion
- Fitnesswerte erreichen höhere Maxima, aber keine höheren Durchschnittswerte
- buschartige Pflanzen erringen die Vorherrschaft
- bei scharfer Selektion notwendiger hoher Fitnesswert kann nur durch Überwuchern von Nachbarpflanzen erreicht werden
- höhere Genomlängen (erforderlich für die Busch-Architektur)

Beispiele für evolvierte Genome und Morphologien:



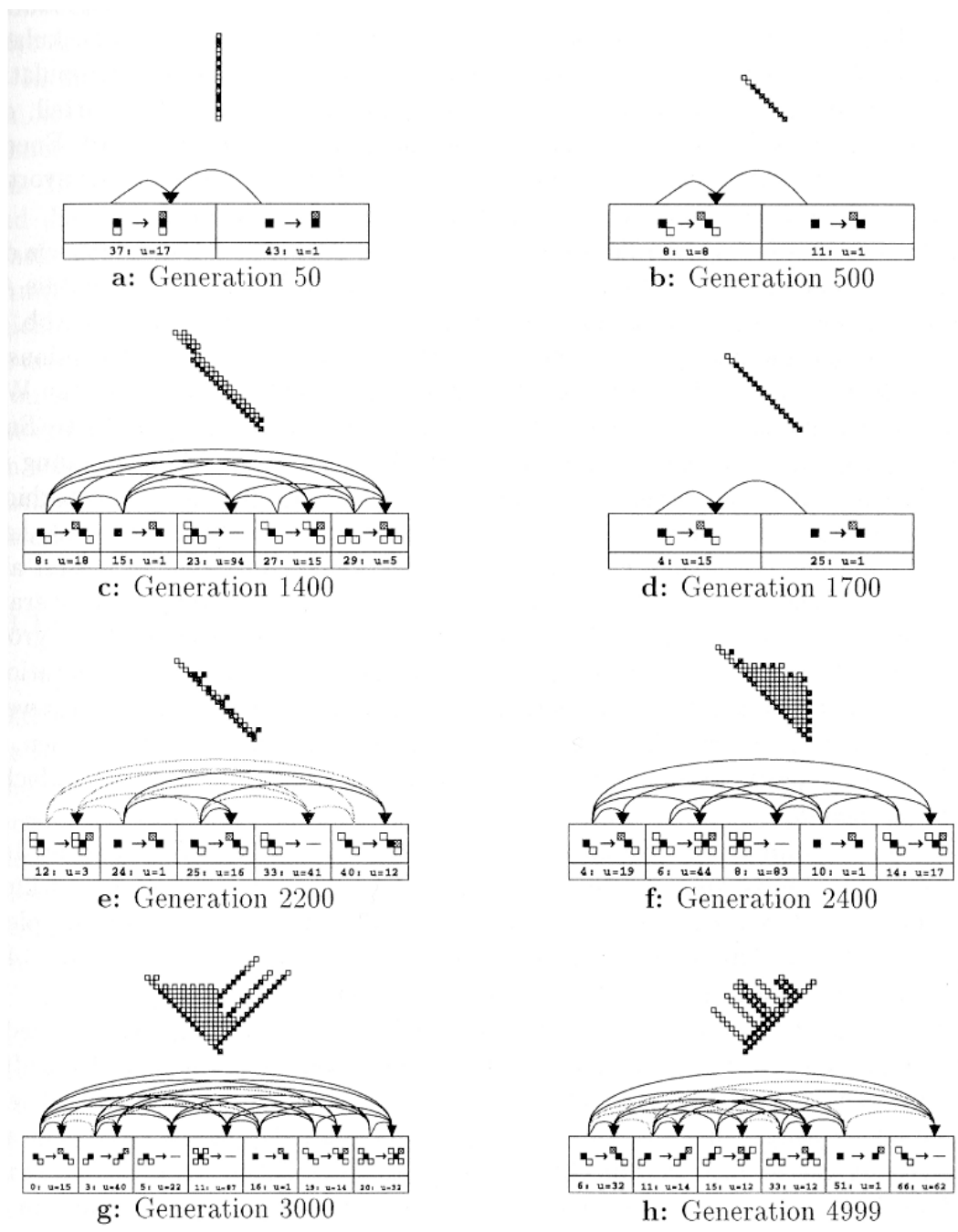
Vergleichslauf mit hoher Mutationsrate (0,1) (und moderater Selektion):

- keinerlei Entwicklung von Formen mit komplexerer Morphologie
- auch nach 5000 Generationen existieren etliche Pflanzen, die überhaupt nicht wachsen
- unbeschränkt wachsende Formen treten gelegentlich auf, aber verschwinden wieder trotz Selektionsvorteil
- "kritische Fehlergrenze" der Mutation überschritten
- große Distanzen zwischen den Genomen, kleine DVK

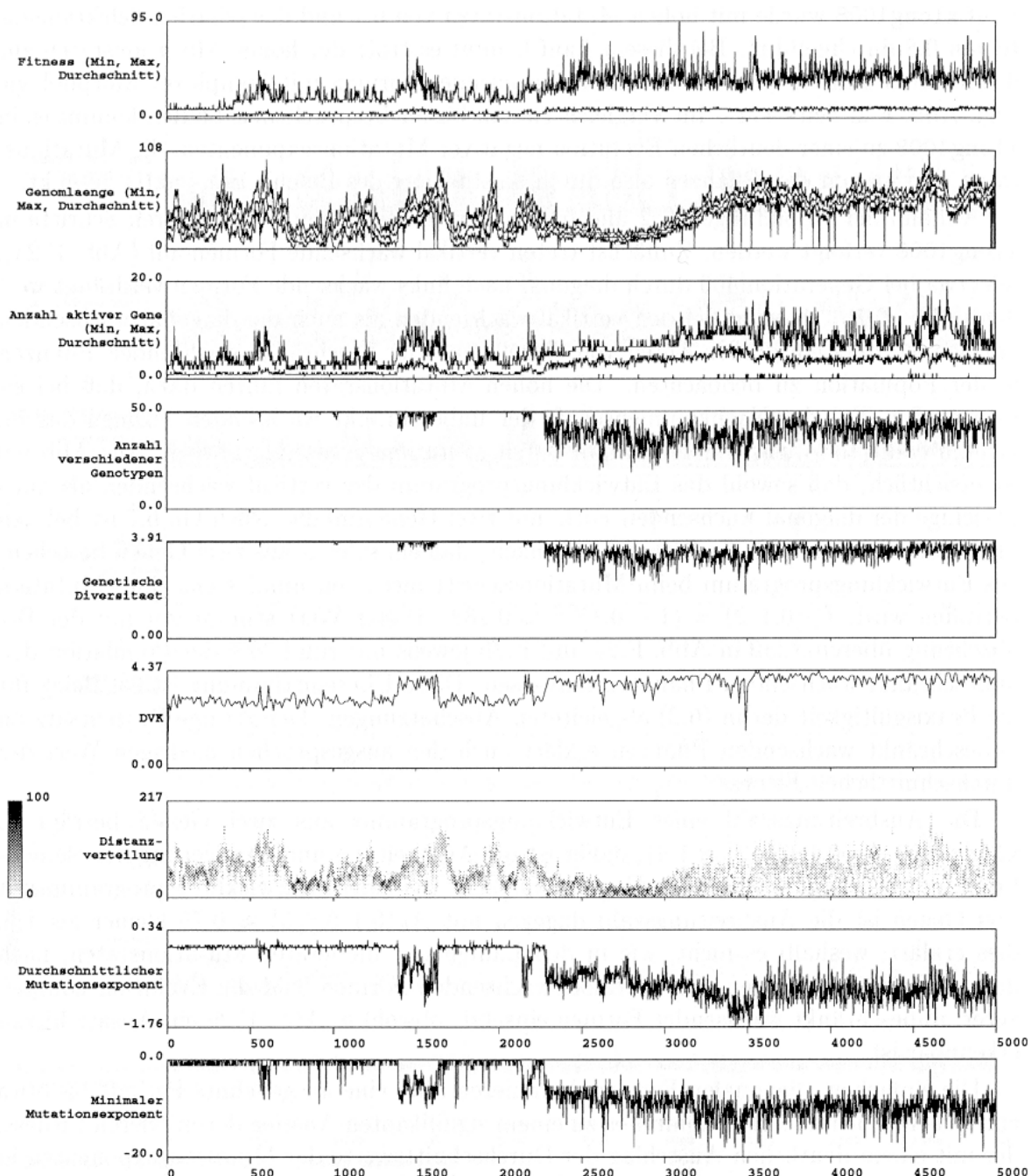


Kombination von hoher Mutationsrate (0,1) und scharfer Selektion (0,8):

- dauerhafte Etablierung unbeschränkt wachsender Formen mit komplexer Morphologie
- deutliche Evolution negativer Mutationsexponenten (aktive Senkung der Mutationsraten)!
- ca. bei Generation 1400 entstehen Pflanzen mit zweiter Zellschicht, "Knicken" in der Achse und Genen mit "Mehrfachfunktion" (s. Abb. unten, c); Zellen der aufgelagerten Schicht führen Aktion "mut-" durch



- mutationsratensenkende Form hat keinen unmittelbaren Selektionsvorteil durch höhere Fitness, sondern aufgrund größerer Ausbreitungsrate ihrer Genome durch die niedrigeren Mutationsraten
- Dominanz mutationsratensenkender Formen geht bei Generation 1650 aus ungeklärten Gründen zuende
- bei Generation 2200 treten wieder solche Formen auf, entwickeln sich weiter zu buschigen Formen
- deutlicher Anstieg der DVK, wenn Mutationsraten aktiv gesenkt werden

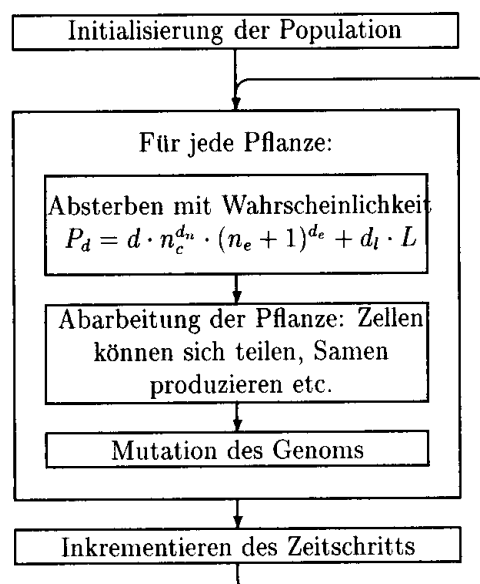




- Evolution von Genomen mit aktiv gesenkten Mutationsraten heißt: Evolution zur "edge of chaos" (kritische Fehlergrenze), von der "Chaos"-Seite her!

2. Variante: "LindEvol-B"

- 2-dim. Gitter; nur Energieparameter, kein interner Zustand (wie in LindEvol-GA)
- blockorientierte Genominterpretation; bitweise Änderungs-mutationen, Insertionen, Deletionen
- zusätzliche Zellaktionen: Samenausbreitung (lokal und fernwirkend)
- kein starrer Selektionsmechanismus in Abhängigkeit von Energiegehalt; Vermehrung erfolgt durch Samenausbreitung
- Pflanzen sterben durch zufallsabhängigen Tod oder durch Angriff; keine Synchronisierung der Lebensphasen!
- Wahrscheinlichkeit für zufallsabhängigen Tod hängt ab von Größe, "Überhangskoeffizient" (räuml. Asymmetrie) und Energiegehalt der Pflanze
- "Angriff" erfolgt, wenn bei Besetzung einer Nachbarzelle diese bereits besetzt ist
- Wahrscheinlichkeit für Erfolg eines Angriffs ist umgekehrt proportional zur Gesamtenergie der angegriffenen Pflanze – im Erfolgsfall stirbt die angegriffene Pflanze
- keine doppelte Zeit-Schleife mehr:



Konsequenzen:

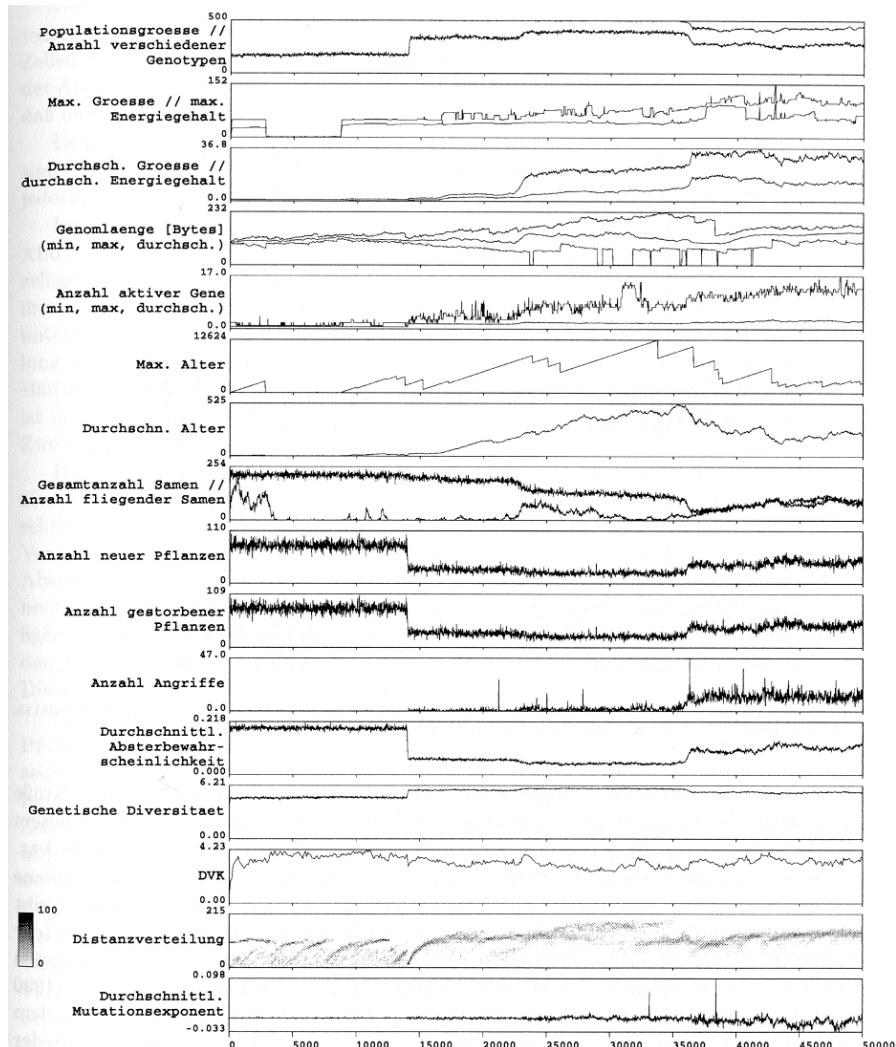
- die Populationsgröße ist veränderlich
- die Pflanzen können auch alle aussterben
- eine einzelne Pflanze kann die ganze Welt überwuchern – solche "superstabilen" Pflanzen entstehen tatsächlich:



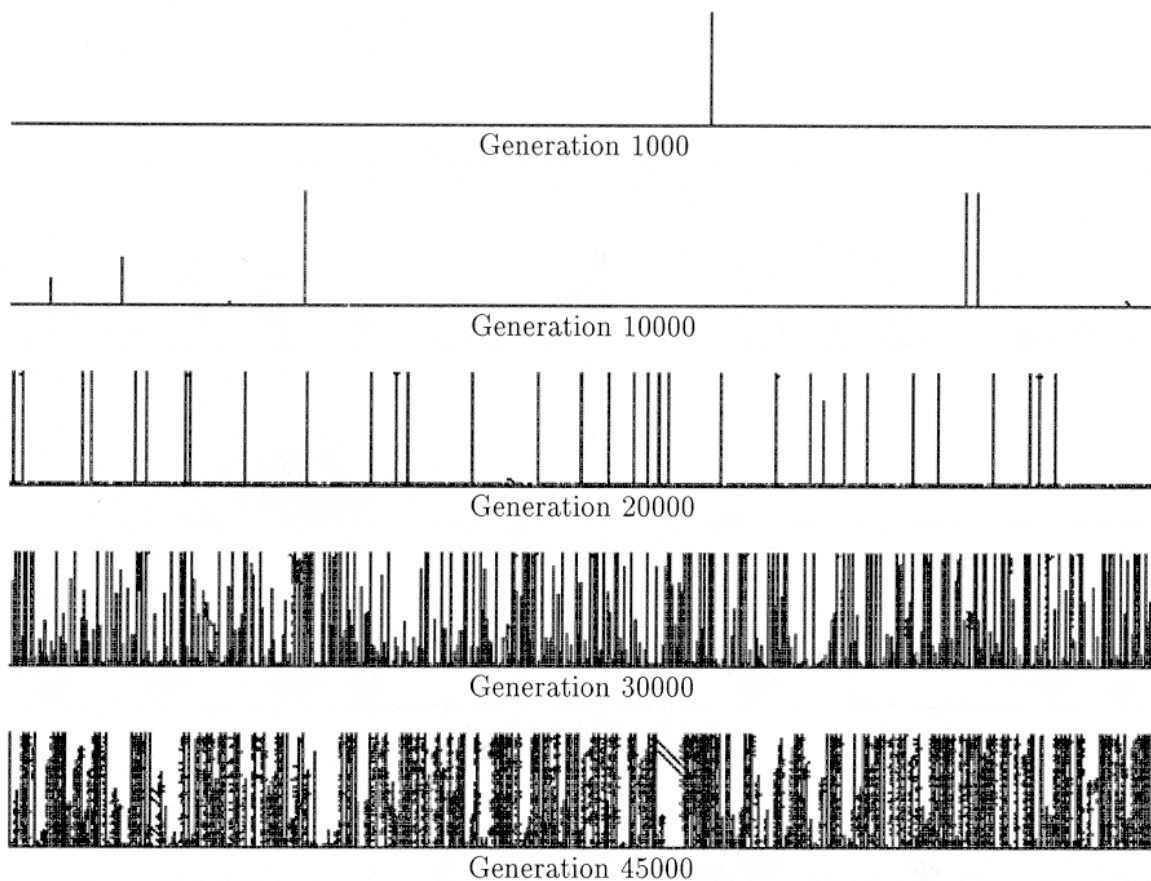
$$a: d_l = 0.02$$

- bei kleinen Werten der Absterbewahrscheinlichkeit sind primitive, einzellige Formen sehr stabil; bei zu hohen Werten stirbt dagegen die gesamte Population in der Anfangsphase
- Startpopulation darf nicht zu klein sein (gewählt wurde $N = 500$), da ein großer Anteil der initialen Pflanzen nicht überlebensfähig ist

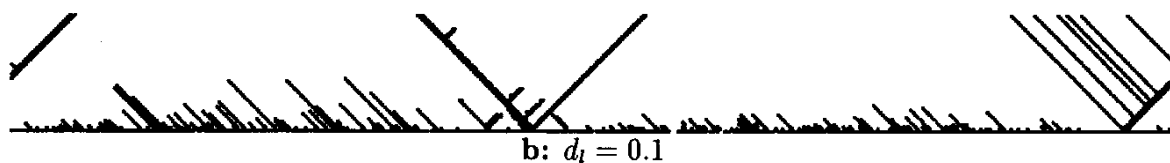
Beispiel-Lauf:



- verschiedene Evolutionssprünge
- am Anfang sind die vertikalen Pflanzen "steril", erreichen aber sehr hohes Alter
- erst nach 25000 Zeitschritten treten unbeschränkt wachsende, vermehrungsfähige Formen auf
- Raumkonflikte und Angriffe treten erst nach Zeitschritt 35000 im nennenswerten Ausmaß auf



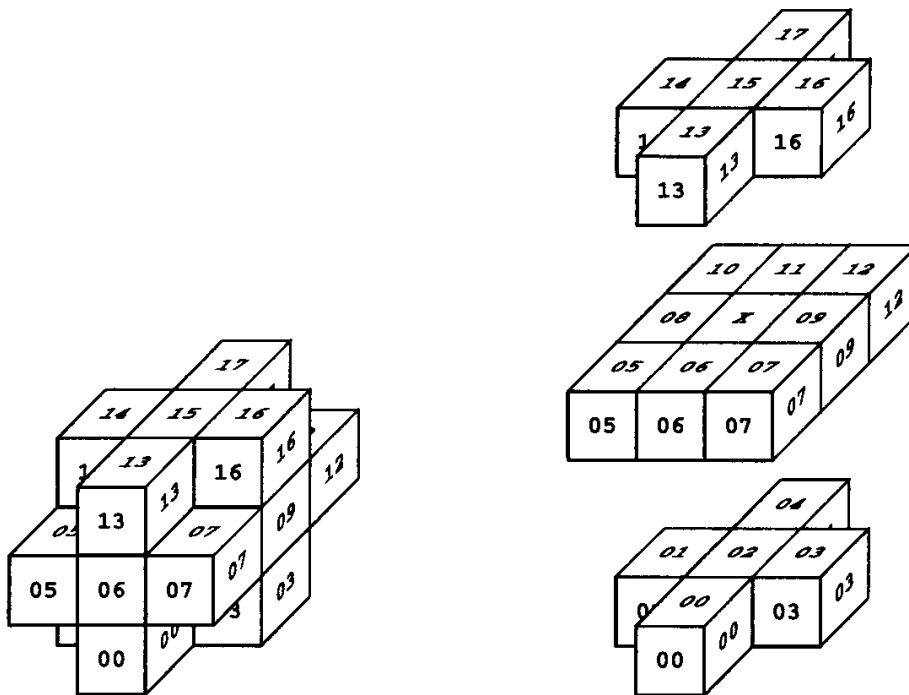
bei anderer Parametrisierung können auch diagonal wachsende Formen die Vorherrschaft erringen:



(nach ca. 50000 Schritten)

- aktive Änderung der Mutationsraten trat in geringerem Umfang auf als in LindEvol-GA

Versuch der Verallgemeinerung auf 3 Dimensionen: 18-Zellen-Nachbarschaft



Übertragung der blockorientierten Genominterpretation (nun mit 18-Bit-Codierung der lokalen Situation einer Zelle):
scheitert!

keine Evolutionsprozesse; die Anfangspopulation stirbt stets aus, ohne Nachkommen zu produzieren

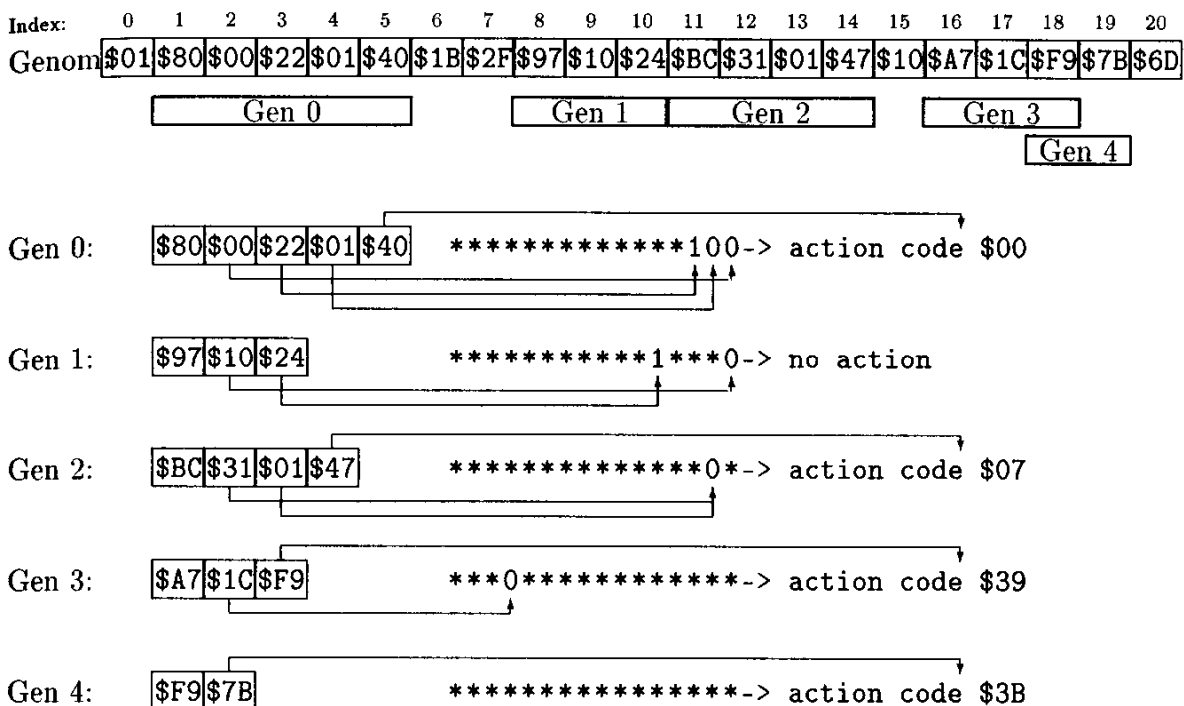
- Wahrscheinlichkeit für Aktivierung eines Gens ist bei 18 Bit-Zellzuständen zu gering
- Generierung eines "Keimzellgens" zur Erzeugung einer vermehrungsfähigen Pflanze ist zu unwahrscheinlich
- 1000-fach größere Startpopulation wäre nötig (zu rechenintensiv)

Ausweg: "promotor-orientierte Genominterpretation"

- analog zur Natur: Markierung von Anfang und Ende eines Gens durch Promotor- und Terminator-Regionen (mit speziellen Codes)
- Gene haben variable Länge
- Gene können sich gegenseitig überlappen

Realisierung der promotor-orientierten Genominterpretation in LindEvol:

- Promotor = Byte, in dem Bit 7 gesetzt ist
- Terminator = Byte, in dem Bit 6 gesetzt ist
- terminiertes Gen = Abschnitt von einem Promotor bis zu einem Terminator, in dem sich keine Promotoren oder Terminatoren befinden
- nichtterminiertes Gen = Abschnitt zwischen 2 Promotoren (oder von einem Promotor bis zum Ende des Genoms), in dem keine Promotoren oder Terminatoren liegen
- bei terminierten Genen steht in den Bits 0-5 des Terminators der Aktionscode
- nichtterminierte Gene lösen keine Aktion aus
- linke Regelseiten können jetzt auch Variablen (wildcard-Symbol *) enthalten
- jedes Byte zwischen einem Promotor und einem Operator (das nicht selbst Promotor oder Terminator ist) spezifiziert ein Bit der linken Regelseite



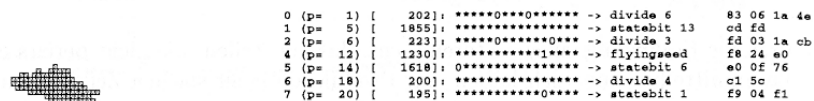
3. Variante: "LindEvol-P"

- 2-dim. Gitter
- Zellen mit internem 16-Bit-Zustandsparameter
- Zellaktionen: wie in LindEvol-B, zusätzlich "Zustand setzen"
- promotororientierte Genominterpretation
- Mutationen: bitweise Änderungen, byteweise Insertionen und Deletionen, Genduplikationen
- Vermehrung, Absterben, Angriff wie in LindEvol-B

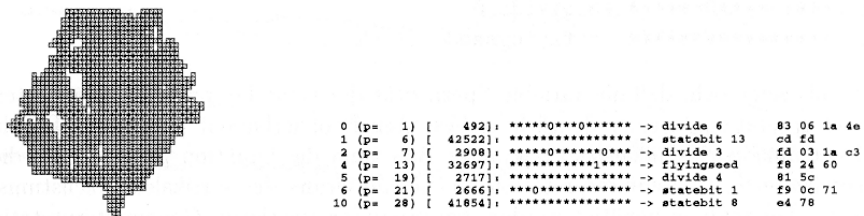
Ergebnisse:

- in vielen Fällen entwickelt sich wesentlich komplexere Dynamik als in den bisherigen Varianten
- flexible Längen und Genspezifikation mit wildcards führt zu deutlich besserer Evolvierbarkeit
- aber größerer Rechenaufwand

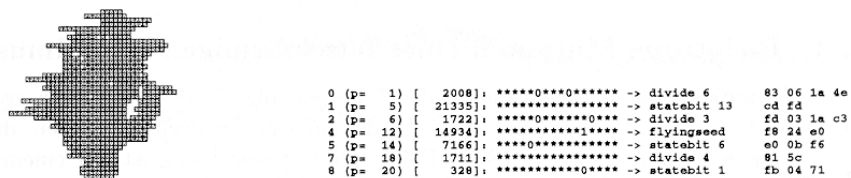
Beispiel-Pflanzen aus einem Simulationslauf:



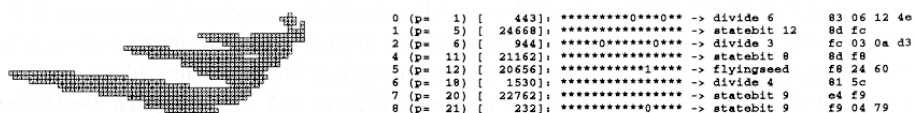
a: Pflanze aus Zeitschritt 9995 mit Entwicklungsprogramm



b: Pflanze aus Zeitschritt 10110 mit Entwicklungsprogramm

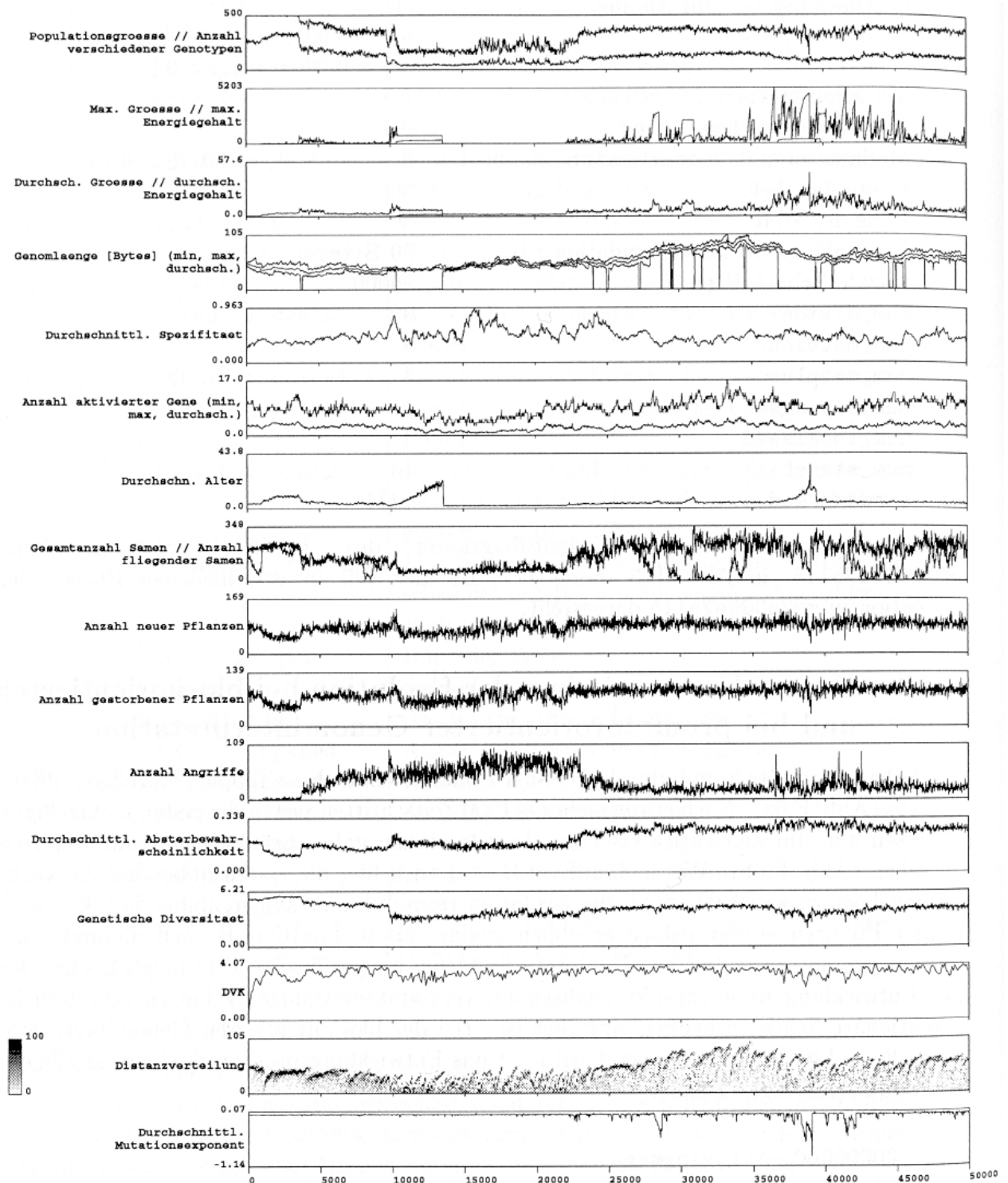


c: Pflanze aus Zeitschritt 10110 mit Entwicklungsprogramm



d: Pflanze aus Zeitschritt 10265 mit Entwicklungsprogramm

Verlaufsdigramme:



- auf die (relativ schnelle) Evolution unbeschränkt wachsender Pflanzen folgt als weiterer Schritt das Auftreten von Pflanzen mit Breitenausdehnung (zunächst nur nach links)
- beidseitig sich ausbreitende Strukturen kurz vor Zeitschritt 10000

- busch- und baumartige Pflanzen sterben vor Schritt 11000 aus, werden von "angriffslustigen" Formen verdrängt, die nur horizontal am Boden entlangwachsen
- bei Schritt 22000 diagonal nach rechts wachsende Formen
- ab Schritt 36000 Riesenformen, die z.T. Energiekrisen auslösen
- ab Schritt 45000 wieder weniger Riesenpflanzen und weniger Angriffe (→ Evolution kooperativeren Wachstumsverhaltens?)

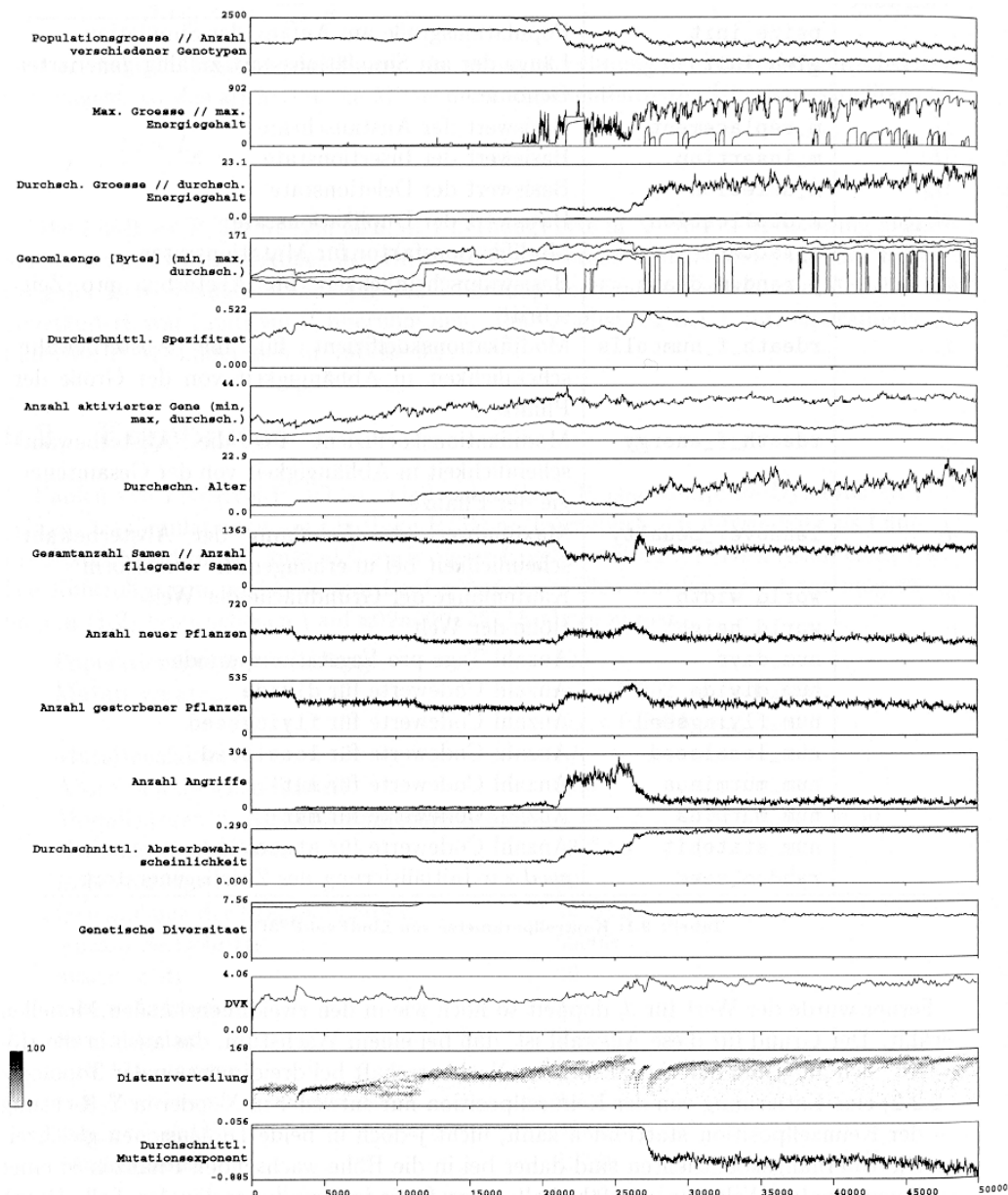


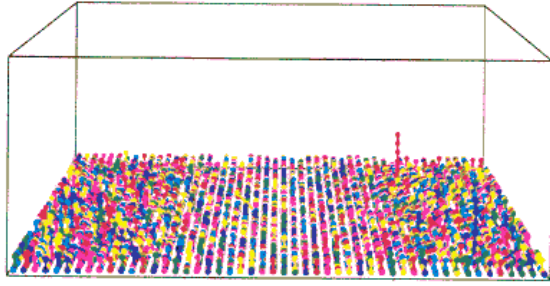
4.Variante: "LindEvol-P/3D" (3-dim. Version von LindEvol-P)

- 3-dim. Gitter
- interner Zellzustandsparameter mit 32 Bit
- Überhangskoeffizient als Mittelwert aus komponentenweisen Überhangskoeffizienten (für Absterbewahrscheinlichkeit)
- sonst wie bei 2D-Version

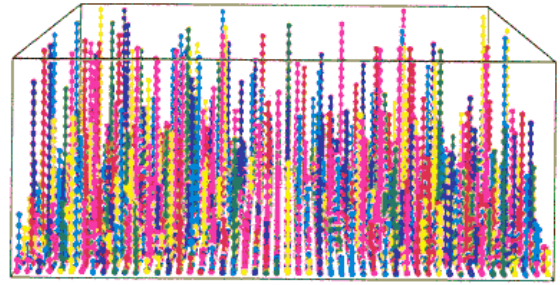
Ergebnisse:

- komplexe Dynamik; Auftreten der "fortgeschrittenen" Formen erfolgt aber langsamer als in 2D
- aktives Absenken des Mutationsexponenten ab ca. Schritt 26000

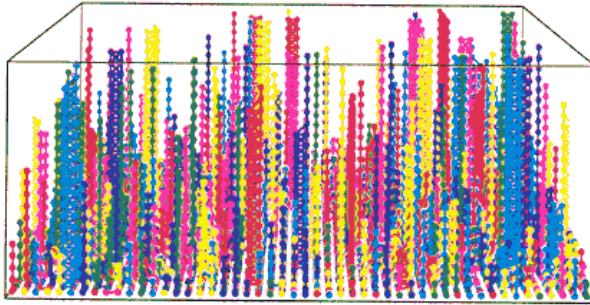




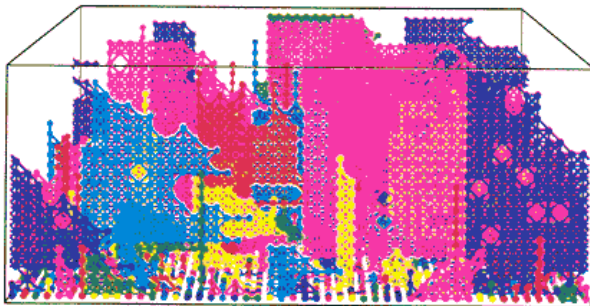
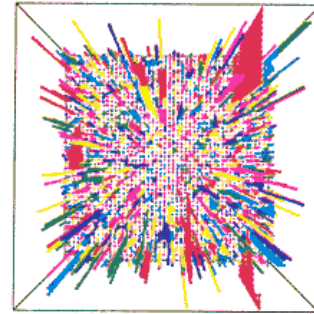
a: Zeitschritt 8000



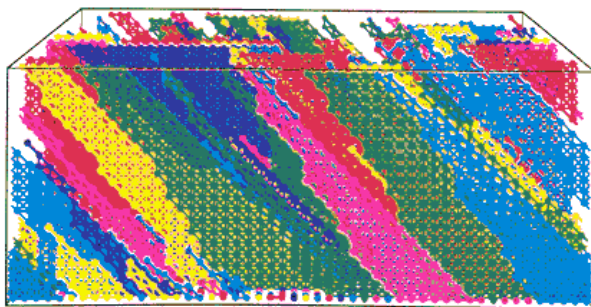
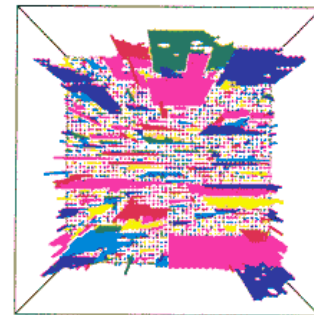
b: Zeitschritt 16000



c: Zeitschritt 20000, Vorderansicht und Vogelperspektive



d: Zeitschritt 24000, Vorderansicht und Vogelperspektive



e: Zeitschritt 40000, Vorderansicht und Vogelperspektive

