Künstliche Evolution — eine intelligente Problemlösungsstrategie?

Hans-Paul Schwefel und Thomas Bäck
Universität Dortmund, Fachbereich Informatik

Hans-Paul Schwefel, Univ. Prof. Dr.-Ing., ist seit 1985 Inhaber des Lehrstuhls für Systemanalyse im Fachbereich Informatik der Universität Dortmund. Mit Evolutionsstrategien beschäftigte er sich schon 1965 im Rahmen seiner Diplomarbeit am Institut für Strömungstechnik der Technischen Universität Berlin.

Thomas Bäck, Dipl.-Inform., ist seit 1990 wissenschaftlicher Mitarbeiter am Lehrstuhl für Systemanalyse und arbeitet z.Zt. im Rahmen eines von der DFG geförderten Forschungsprojektes zum Thema „Untersuchung der Einsatzmöglichkeiten von Genetischen Algorithmen und Classifier-Systemen zur technischen Optimierung“.

(Abstract)

Auf der Suche nach natürlichen Vorbildern für intelligentes Verhalten bot das Instrument, mit dem wir auf diese Suche gingen, unser Gehirn, sich in nicht ganz uneigennütziger Weise erst einmal selbst an. Den Prozeß, der zu seiner Entwicklung geführt hat, die natürliche Evolution, war es geneigt, in diskriminierender Weise für primitiv und ineffizient zu halten. Aufgrund eines falschen Modells der Evolution „bewiesen“ einige Proponenten sogar die Unmöglichkeit der evolutiven Entstehung höherer Lebewesen einschließlich derer Gehirne. So ist es dann auch kein Wunder, daß erste ernsthafte algorithmische Nachahmungen von Evolutionsprozessen in den 60er Jahren lange Zeit unbeachtet blieben oder gar rigoros abgelehnt wurden. Erst in den letzten Jahren wurden Evolutionäre Algorithmen halbwegs hoffähig, unter anderem dadurch, daß sie eine nicht unbedeutende Rolle bei den Bemühungen derjenigen spielen, welche sich unter der Flagge „Artificial Life“ von der traditionellen KI abgesetzt haben.

Monte-Carlo-Evolution

Rein zufällige Mutationen und eine falsch verstandene Selektion, die nur das beste aller Individuen aus einer großen Schar blinder Versuche als überlebensfähig auswählt, bilden den Kern des frühesten Modells der Evolution (siehe z.B. Ashbys Homöostat [Ash60]). Angewandt auf die Suche nach einem ausgezeichneten Teilraum des $\mathbb{R}^n$ mit Kantenlänge d aus einem Suchraum mit Kantenlänge D im Sinne der Lösung eines CSP (Constraint Satisfaction Problem) oder eines Optimierungsproblems mit Genauigkeitsforderungen $\epsilon = d/D$ für alle n beteiligten Variablen, ist die Wahrscheinlichkeit für einen blinden Zufallstreffer bei einem Experiment

$$p_1 = \left(\frac{d}{D}\right)^n \quad . \tag{1}$$

Führt man — nacheinander oder gleichzeitig — N solcher *Monte-Carlo-Versuche* durch, so ist die Gesamtwahrscheinlichkeit, mindestens einen Treffer zu erzielen

$$p_N = 1 - [1 - p_1]^N = 1 - \left[1 - \left(\frac{d}{D}\right)\right]^N \quad , \tag{2}$$

bzw. die Anzahl benötigter Experimente bei vorgegebener Trefferwahrscheinlichkeit p_N

$$N = \frac{\log(1 - p_N)}{\log\left[1 - (d/D)^n\right]} \quad , \tag{3}$$

was sich für $d \ll D$ und $n \gg 1$ vereinfacht zu

$$N \approx -\ln(1 - p_N) \cdot \left(\frac{D}{d}\right)^n \quad . \tag{4}$$

Aus diesem Sachverhalt läßt sich zweierlei schlußfolgern, nämlich:

- daß der mit der Zahl der Entscheidungsvariablen bzw. Freiheitsgrade exponentiell steigende Aufwand prohibitiv für die Anwendung auf große und genau zu lösende Probleme ist,

- daß eine deterministische *Rastermethode* in jedem Falle vorzuziehen ist, wenn die Wahrscheinlichkeit des Gesamterfolges größer als etwa 63% sein soll.

Für $n = 50.000.000$ Nukleotidbasen in einem menschlichen DNS-Strang mit je 4 Ausprägungsmöglichkeiten reichen die nur etwa 10^{17} Sekunden der Erdgeschichte bei weitem nicht aus, um die kombinatorisch erforderlichen $4^{50.000.000} \approx 10^{30.103.000}$ Experimente durchzuführen. Selbst dann, wenn wir als Generationsdauer nur eine Stunde (typisch für Bakterien und Hefen) sowie eine erdumspannende Schichtdicke der Lebewelt von zwei Metern annehmen, in der sich dann — dicht gepackt — jeweils etwa 10^{34} Bakterien am Experiment hätten beteiligen können, müßten wir noch lange auf die Lösung warten. Parallelität allein hilft bei diesem „Fluch der Dimension“ (nach Bellman) nicht viel weiter. Abgesehen von Koordinierungs- und Speicherproblemen für Zwischenergebnisse, hätten so in 3,5 Milliarden Jahren nur knapp 80 Nukleotidbasen in einer optimalen Weise angeordnet werden können.

Dieses negativ ausgelaufene Gedankenexperiment muß aber nicht zulasten der Evolution ausgelegt werden, sondern wohl eher zulasten des unangemessenen Modells.

Genetische Algorithmen

Um das von Holland [Hol75] eingeführte Evolutionsmodell zu erläutern, muß erst einmal eine Problemklasse benannt werden, auf die der Algorithmus angewendet werden soll. Als einen typischen Vertreter wählen wir hier ein Parameter-Optimierungsproblem mit n Entscheidungsvariablen x_i ($i \in \{1, ..., n\}$).

Gegeben sind eine *Zielfunktion*

$$f : M \subseteq \mathbb{R}^n \to \mathbb{R}\ ,\ M \neq \emptyset \tag{5}$$

und q *Restriktionen* in Form von Ungleichungen

$$M = \{x \in \mathbb{R}^n \mid g_j(x) \geq 0\ \forall j \in \{1, \ldots, q\}\} \quad . \tag{6}$$

Gesucht ist das *globale Minimum* $x^* \in M$:

$$\forall x \in M\ :\ f(x) \geq f(x^*) = f^* \quad , \tag{7}$$

das minimum minimorum unter den *lokalen Optima* $\hat{x} \in M$:

$$\exists \epsilon > 0\ \forall x \in M\ :\ \|x - \hat{x}\| < \epsilon \Rightarrow f(x) \geq f(\hat{x}) = \hat{f} \quad . \tag{8}$$

Genetische Algorithmen (GA) operieren mit einer *Population* P von λ *Individuen*, die jeweils einen Vektor der Entscheidungsvariablen in binär kodierter Form repräsentieren:

$$\begin{array}{rcl} a_k^t & \in & I = \{0,1\}^l \quad \forall k \in \{1, \ldots, \lambda\} \\ P^t & = & (a_1^t, \ldots, a_\lambda^t) \in I^\lambda \quad . \end{array} \tag{9}$$

Das hochgestellte t ist der Generationenzähler, der beim Start auf Null gesetzt wird.

Bei Vorliegen eines binären Optimierungsproblems ($n = l$) kann diese Repräsentationsform direkt verwendet werden. Im Falle reellwertiger Variablen $u_i \leq x_i \leq v_i$ mit oberen und unteren Schranken als einer Spezialform

$$g : \prod_{i=1}^{n} [u_i, v_i] \to \mathbb{R} \tag{10}$$

für die Restriktionen benötigt man eine Dekodierungs-Funktion

$$\Gamma_{l_x}^i : \{0,1\}^{l_x} \to [u_i, v_i] \quad , \tag{11}$$

z.B.

$$\Gamma_{l_x}^i(\alpha_1 \ldots \alpha_{l_x}) = u_i + \frac{(v_i - u_i)}{2^{l_x} - 1} \cdot \left(\sum_{j=1}^{l_x} \alpha_j 2^{j-1} \right) \quad , \tag{12}$$

woraus sich als Länge des Bitstrings, welcher ein Individuum charakterisiert, $l = n \cdot l_x$ ergibt. Der Generationswechsel vollzieht sich in folgenden Teilschritten:

$t := 0;$
initialisiere $P^0 := (a_1^0, \ldots, a_\lambda^0) \in I^\lambda;$
bewerte $P^0;$
while *Abbruchkriterium nicht erfüllt* **do**
  $t := t + 1;$
  selektiere $P'^t := s(P^{t-1});$
  rekombiniere $P''^t := r(P'^t);$
  mutiere $P^t := m(P''^t);$
  bewerte $P^t;$
od

Die Zahl der Individuen bleibt also konstant λ während dieser künstlichen Evolution. Der Selektionsoperator s

$$s : I^\lambda \to I^\lambda \ , \ s(P^{t-1}) = P'^t \tag{13}$$

arbeitet auf der Basis einer Reproduktionswahrscheinlichkeit $p_s(a_k^{t-1})$, die umso höher ist, je günstiger der Zielfunktionswert $f(a_k^{t-1})$ des Individuums a_k^{t-1} $(1 \leq k \leq \lambda)$ im Vergleich zu den übrigen Mitgliedern der Population liegt. Auf Feinheiten wie den Unterschied zwischen der klassischen *proportionalen* [Hol75] und der neueren *rangbasierten* Selektion [Bak85] soll hier nicht eingegangen werden. Wegen des späteren Vergleichs mit den Evolutionsstrategien sei aber betont, daß die für GA benutzte Selektion erstens stochastischer Natur ist und zweitens prinzipiell jedem Individuum einer Elternpopulation die Chance einräumt, seine genetische Information in die nächste Generation einzubringen, wobei drittens hervorragende Individuen besonders viele Nachkommen haben.

Bei der Nachahmung der Rekombination werden mit exogen vorgegebener Wahrscheinlichkeit p_r, z.B. $p_r = 0.6$, Teilstücke der Bitstrings zweier Individuen miteinander vertauscht.

Unabhängig von den durch l_x gegebenen Grenzen zwischen den unterschiedliche Entscheidungsvariablen kodierenden Teilstücken des Bitstrings werden einer (traditionelles Ein-Punkt Crossover) oder mehrere Crossover-Punkte zufällig ausgewählt. Beim Ein-Punkt Crossover wird die Information hinter dem Crossover-Punkt zwischen den beiden beteiligten Individuen ausgetauscht, während beim verallgemeinerten Mehr-Punkt Crossover alternierend ein zwischen zwei Crossover-Punkten liegendes Teilstück ausgetauscht oder beibehalten wird. Es ist offensichtlich, daß mit zunehmender Zahl der Crossover-Punkte die Stärke der Durchmischung genetischer Information steigt, aber die Frage nach der günstigsten Anzahl von Crossover-Punkten ist heute noch gänzlich unbeantwortet. Häufig werden zwei oder acht solcher Positionen rein zufällig ausgewählt.

Dem *Crossover-Operator*

$$r : I^\lambda \times [0,1] \to I^\lambda \ , \ r(P'^t, p_r) = P''^t \tag{14}$$

(hier als Populationen ineinander überführender Makro-Operator beschrieben) nachgeschaltet ist die Anwendung des *Mutations-Operators*

$$m : I^\lambda \times [0,1] \to I^\lambda \ , \ m(P''^t, p_m) = P^t \tag{15}$$

(ebenfalls als Makro-Operator formuliert). Mit ebenfalls fest vorgegebener Mutationswahrscheinlichkeit p_m, z.B. $p_m = 0.001$, werden einzelne Bits in den individuellen Strings — völlig unabhängig von deren Position — verändert von Null nach Eins bzw. umgekehrt. Diese Art Mutationen, die im Gegensatz zur Rekombination ein seltenes Ereignis darstellt, bringt je nach der Lage des Bits in dem l_x langen Teilstück einer kodierten Entscheidungsvariablen nur kleine oder auch riesengroße Änderungen mit sich. Im Gegensatz zu Evolutionsstrategien darf Mutation daher nur als Hintergrund-Operator eine Rolle spielen. Benötigt wird dieser aber für den Fall, daß im Laufe der Optimumsuche in der Population das für die richtige Lösung benötigte Bit völlig verlorengegangen ist.

Die ersten Arbeiten von Holland, z.B. [Hol62], blieben lange Zeit weitgehend unbekannt, erst mit seinem Buch [Hol75] sowie der Dissertation von De Jong [Jon75] begann sich die Idee auszubreiten. Hauptsächliche Anwendungsgebiete betrafen die Lösung von Problemen, die sich als diskrete Optimierungsaufgaben formulieren ließen und mit Hilfe von Digitalrechnern gelöst wurden. Ein öffentlich verfügbares GA-Programm stammt von Grefenstette [Gre87], eine zusammenfassende Darstellung vieler Einzelaspekte und einiger Anwendungen von Goldberg [Gol89]. Seit 1985 findet alle zwei Jahre eine internationale Konferenz, die sog. ICGA (International Conference on Genetic Algorithms and their Applications) statt, bisher stets in den USA. Zu theoretischen Aussagen über Konvergenzsicherheit und Konvergenzgeschwindigkeit sowie Implementierungen für Parallelrechner sei auf das Kapitel „Theoretische Aussagen zu Evolutionären Algorithmen“ verwiesen.

Evolutionsstrategien

Im Jahre 1973 erschien als erstes einer breiteren Öffentlichkeit zugängliches Werk über Evolutionsstrategien die Dissertation „Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution“ von Rechenberg [Rec71, Rec73]. Auch hier reichen die ersten, weitgehend unbekannt gebliebenen Arbeiten der „Arbeitsgruppe Evolutionstechnik“ an der Technischen Universität Berlin zurück bis in die Mitte der 60er Jahre. Ausgangspunkt war der Versuch, eine effektive und effiziente Strategie für die experimentelle Optimierung zu kreieren. Als sich herausstellte, daß die traditionellen Verfahren der numerischen Optimierung für diese Aufgabe nicht recht taugten, entstand als erster Grundalgorithmus die später (1+1)–ES genannte *zweigliedrige Evolutionsstrategie*.

(1+1)–ES

Für ein restringiertes Parameter-Optimierungsproblem mit reellwertigen Entscheidungsvariablen lautet der Grundalgorithmus:

$t := 0;$
initialisiere $P^0 := x_P^0$; *so, daß* $\forall j : g_j(x_P^0) \geq 0$;
while *Abbruchkriterium nicht erfüllt* **do**
  mutiere $P^t \; : \; x_N^t := x_P^t + \sigma \cdot \Delta x^t$
    mit der Wahrscheinlichkeitsdichte
    $p(\Delta x_i^t) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(\Delta x_i^t)^2\right)$;
  bewerte $P^t \; : \; f(x_P^t), f(x_N^t)$;
  selektiere P^{t+1} *aus* P^t:
    wenn $f(x_N^t) \leq f(x_P^t)$
    und $\forall j : g_j(x_N^t) \geq 0$
    dann $x_P^{t+1} := x_N^t$
    sonst $x_P^{t+1} := x_P^t$;
  $t := t + 1$;
od

Dabei gehorchen die Mutationen einer n-dimensionalen Standard–Normalverteilung mit Erwartungswert Null und einer Kovarianzmatrix $C = I$, deren Diagonalelemente Eins und alle übrigen Elemente Null sind. Die Standardabweichung σ, im folgenden auch (mittlere) *Mutationsschrittweite* genannt, muß anfangs exogen vorgegeben werden. Im Prinzip können auch individuelle σ_i passend zu den x_i verschieden voneinander vorgegeben werden, ihr Verhältnis untereinander bleibt aber im Laufe einer künstlichen Evolution erhalten.

Anschaulich ausgedrückt, erzeugt ein Vorfahr also einen Nachkommen, dessen Position im $\mathbb{R}^n$ der Entscheidungsvariablen im Mittel dem des Elters entspricht, durch ungerichtete Umwelteinflüsse aber etwas abweicht. Große Abweichungen sind seltener als kleine — entsprechend der Wahrscheinlichkeitsdichteverteilung der Gaußschen Glockenkurve. Der Nachkomme ersetzt den Vorfahren, wenn er ihn im Sinne der Zielfunktion übertrifft, anderenfalls überlebt der Vorfahr die Generation. Bei geringer Varianz σ^2 verhält sich das Verfahren wie eine stochastische Gradientenmethode, erfordert aber — und das war für die experimentelle Optimierung wichtig, ist aber für viele Computer-Anwendungen ebenfalls erwünscht — keine Information über partielle Ableitungen der Zielfunktion.

Zur Lösung des Problems, eine geeignete Varianz σ^2 zu finden, betrachtete Rechenberg zwei n-dimensinale Situationen

$$\begin{aligned} f_1(x) &= \mathrm{F}(x_1) = c_0 + c_1 x_1 && \text{sog. } \textit{Korridormodell} \\ & \quad \forall i \in \{2, \ldots, n\} \; : \; -b/2 \leq x_i \leq b/2 && \\ f_2(x) &= c_0' + c_1' \cdot \sum_{i=1}^{n} (x_i - x_i^*)^2 && \text{sog. } \textit{Kugelmodell} \end{aligned} \tag{16}$$

und fand eine geeignete Regel, um näherungsweise für beide Fälle die Größe σ jeweils optimal anzupassen, ohne etwas über die Topologie zu wissen. Allein aus der beobachteten relativen Erfolgshäufigkeit w_e der auftretenden Mutationen kann die 1/5–*Erfolgsregel* folgende Empfehlung abgeben:

$$\text{wenn} \quad w_e \begin{cases} > \frac{1}{5} & , \text{ dann } \sigma^t = \sigma^{t-1}/c \\ < \frac{1}{5} & , \text{ dann } \sigma^t = \sigma^{t-1} \cdot c \\ = \frac{1}{5} & , \text{ dann } \sigma^t = \sigma^{t-1} \end{cases} \tag{17}$$

Für die Konstante $c < 1$ gab Schwefel [Sch75] den Wert $\sqrt[n]{0.85}$ an.

(μ+1)–ES

Rechenberg entwarf auch eine erste Variante mit mehr als nur zwei Individuen pro Generation, bei der Rekombination als weiterer Variationsmechnanismus ermöglicht wird:

$t := 0$;
initialisiere $P^0 := (a_1^0, \ldots, a_\mu^0) \in I^\mu$;
bewerte P^0;
while *Abbruchkriterium nicht erfüllt* **do**
  rekombiniere $a_{\mu+1}^t = r(P^t)$;
  mutiere P^t : $a_{\mu+1}^t := a_{\mu+1}^t + \sigma \cdot \Delta x^t$;
  bewerte $a_{\mu+1}^t$;
  selektiere P^{t+1} *aus* P^t:
    wenn $f(a_{\mu+1}^t) < f(a_k^t) := \max\{f(a_i^t) \mid i \in \{1, ..., \mu\}\}$
    und $\forall j \in \{1, ..., q\} : g_j(a_{\mu+1}^t) \geq 0$
    dann $P^{t+1} := (a_1^t, \ldots, a_{k-1}^t, a_{\mu+1}^t, a_{k+1}^t, \ldots, a_\mu^t)$
    sonst $P^{t+1} := P^t$;
  $t := t + 1$;
od

Diese noch rein sequentiellem Denken verhaftete ES-Variante ähnelt im Selektionsschritt den beiden sog. Polytop-Verfahren — Simplex-Strategie von Nelder und Mead [NM65] bzw. Complex-Strategie von M. J. Box [Box65] — bei denen ebenfalls die bisher schlechteste Position durch die neue ersetzt wird, solange diese nicht die schlechteste aller Positionen ist. Rechenberg konnte an einem Beispiel aufzeigen, daß sich die Rekombination bei einem binären Optimierungsproblem sehr positiv auf die Konvergenzgeschwindigkeit auswirkt. Eine Selbstadaptation der Schrittweiten war auf dieser Basis aber nicht möglich, vielmehr wurde weiterhin die 1/5–Erfolgsregel als exogene Steuerung benutzt.

(μ+λ)–ES **und** (μ,λ)–ES

Die Verallgemeinerung der Idee einer — im Prinzip parallele Operationen einbeziehenden — *mehrgliedrigen Evolutionsstrategie*, bei der nun zum ersten Mal auch die wesentlichen Strategieparameter (z.B. alle σ_i) den Prinzipien Mutation, Rekombination und Selektion unterliegen, stellte Schwefel [Sch75, Sch77, Sch81] zuerst vor. Mehrgliedrige ES operieren mit $\mu > 1$ Eltern und λ Nachkommen in jeder Generation, wobei $\lambda > \mu$ für die Komma-Variante notwendig ist, weil die Eltern der nächsten Generation sich nur aus den μ besten der λ Nachkommen rekrutieren, während in der Plus-Version Eltern „überleben“ können, falls sie nicht von ihren Nachkommen übertroffen werden im Hinblick auf ihre Zielfunktionswerte. Das Grobschema des Algorithmus lautet wie folgt:

$t := 0;$
initialisiere $P^0 := (a_1^0, \ldots, a_\mu^0) \in I^\mu$
wobei $I = \mathbb{R}^w$ *bzw.* $a_k = (x_i, c_{ij} = c_{ji}\ \forall i, j \in \{1, \ldots, n\});$
bewerte $P^0;$
while *Abbruchkriterium nicht erfüllt* **do**
  rekombiniere $a_k'^t := r(P^t)\ \forall k \in \{1, \ldots, \lambda\};$
  mutiere $a_k''^t := m(a_k'^t);$
  bewerte $P''^t := (a_1''^t, \ldots, a_\lambda''^t);$
  selektiere $P^{t+1} := s(P''^t);$ *bzw.* $P^{t+1} := s(P''^t, P^t);$
  $t := t + 1;$
od

Jedes Individuum ist nun aber nicht nur durch einen Vektor x der *Objektvariablen*, sondern auch noch durch einen zusätzlichen Vektor von *Strategievariablen* charakterisiert. Dabei können bis zu n unterschiedliche Varianzen $c_{ii} = \sigma_i^2$ $(i \in \{1, \ldots, n\})$ sowie bis zu $n \cdot (n-1)/2$ Kovarianzen c_{ij} $(i \in \{1, \ldots, n\},\ j \in \{i+1, \ldots, n\})$ der verallgemeinerten n-dimensionalen Normalverteilung mit der Wahrscheinlichkeitsdichtefunktion

$$p(\Delta x) \quad = \quad \sqrt{\frac{\det A}{(2\pi)^n}} \exp\left(-\frac{1}{2}(\Delta x)^T A(\Delta x)\right) \tag{18}$$

einbezogen werden, so daß $w = n \cdot (n+1)/2$ Strategieparameter während der Optimumsuche selbsttätig angepaßt werden. Um sicherzustellen, daß die Kovarianzmatrix A^{-1} positiv semi-definit bleibt, wurden statt der Koeffizienten c_{ij} äquivalente Drehwinkel α_j der Hauptachsen der konzentrischen Hyperellipsoide verwendet, welche Höhenlinien gleicher Mutationswahrscheinlichkeitsdichte darstellen. Diese Winkel dürfen dann im Bereich $0 \leq \alpha_j \leq 2\pi$ frei variieren. Der *Mutations-Operator* muß entsprechend erweitert werden:

$$m(a_k^t) \quad = \quad a_k''^t = (x', \sigma', \alpha') \in I \quad , \tag{19}$$

wobei komponentenweise folgende Operationen durchzuführen sind:

$$\begin{aligned} \sigma_i' &= \sigma_i^t \cdot \exp(\tau_0 \cdot \Delta\sigma_0) \cdot \exp(\tau \cdot \Delta\sigma_i) \\ \alpha_j' &= \alpha_j^t + \beta \cdot \Delta\alpha_j \\ x_i' &= x_i^t + \Delta x_i(\sigma', \alpha') \quad . \end{aligned} \tag{20}$$

Somit sind die Mutationen der Objektvariablen gemäß den Werten des Vektors α miteinander korreliert, wobei der Vektor σ als eine Art Skalierung der (linearen) Metrik aufgefaßt werden kann. Die Änderungen $\Delta\sigma$ und $\Delta\alpha$ gehorchen jeweils wieder unkorrelierten Wahrscheinlichkeitsdichteverteilungen mit Erwartungswerten Null und Varianzen Eins, wobei die Konstanten τ_0, τ und β recht unempfindliche exogene Parameter auf einer höheren Stufe darstellen. Bei der Schrittweitenvariation ist der angegebene zweistufige Prozeß unbedingt notwendig. Der durch τ_0 geprägte Zufallsfaktor wirkt auf alle Komponenten σ_i in gleicher Weise, während der durch τ skalierte Zufallsfaktor für jede Komponente verschieden ist. Nur so bleiben alle benötigten Freiheitsgrade erhalten. Im Falle $\tau_0 = 0$ und $n \gg 1$ würde sich sonst nämlich die Gesamtschrittweite der Mutationen

$$\sigma_{gesamt} \quad = \quad \sqrt{\left(\sum_{i=1}^{n} \sigma_i^2\right)} \tag{21}$$

kaum noch verändern können. Daß diese künstliche Evolutions-Variante nicht ganz unnatürlich ist, belegen die von der Biologie bestätigten Befunde zur Polygenie und Pleiotropie. Diese besagen, daß eine phänotypische Eigenschaft von mehreren Genen gesteuert wird bzw. daß ein Gen mehrere phänotypische Eigenschaften beeinflußt. Ein gut bekanntes Beispiel hierfür ist das allometrische Wachstum. Geht man davon aus, daß sich solche Korrelationen durch die Selektion gemäß den herrschenden Umweltbedingungen ausprägen, dann kann man sie — ähnlich den gelernten Gewichtsfaktoren in künstlichen neuronalen Netzen oder auch den Koeffizienten der inversen Hesseschen Matrix in Quasi-Newton-Methoden — als ein speziell kodiertes *internes Modell* der Außenwelt interpretieren.

Die interessante Frage lautet jetzt, ob es zum Erlernen eines passenden internen Modells eines überwachten Anpassungsalgorithmus' bedarf oder ob sich diese strategischen Parameter im Rahmen des üblichen Selektionsschemas „von selbst“ anpassen. Die noch interessantere Antwort auf diese Frage lautet [Sch87], daß unter geeigneten Bedingungen dieser Selbstadaptationsprozeß tatsächlich funktioniert, obwohl die Güte der strategischen Parameter nur indirekt über die damit erzielten Erfolge auf der Ebene der Objektvariablen, gemessen an den zugehörigen Zielfunktionswerten, erfolgt. Hierauf wird später noch näher eingegangen.

Der *Rekombinations-Operator* bei Evolutionsstrategien mischt die genetische Information zweier (oder auch mehrerer) Eltern, wobei die einzelne Objektvariablen betreffenden Informationsbruchstücke allerdings nicht auseinandergerissen werden:

$$r(P^t) \quad = \quad a' = (x', \sigma', \alpha') \in I \tag{22}$$

wobei, stellvertretend für alle Komponenten der drei Vektoren, die Regeln hier nur für eine Objektvariable x'_i aufgeführt sind:

$$x'_i \quad = \quad \begin{cases} x_{S,i} & \text{ohne Rekombination} \\ x_{S,i} \text{ oder } x_{T,i} & \text{diskrete Rekombination} \\ x_{S,i} + \chi \cdot (x_{T,i} - x_{S,i}) & \text{intermediäre Rekombination} \\ x_{S_i,i} \text{ oder } x_{T_i,i} & \text{global, diskret} \\ x_{S_i,i} + \chi_i \cdot (x_{T_i,i} - x_{S_i,i}) & \text{global, intermediär} \end{cases} \tag{23}$$

Die Indizes S und T bezeichnen zwei beliebig ausgewählte Eltern und der Faktor χ eine im Intervall $[0, 1]$ gleichverteilte Zufallszahl. Die hier global genannten Varianten, die zwar algorithmisch einfach zu realisieren sind, in der Natur jedoch nicht auftreten, verwenden für jede einzelne Genkomponente einen je eigenen, ausgewürfelten Vorfahr (global diskret) oder einen Mix aus je zwei beliebigen Eltern, wobei der Mischfaktor χ_i für jede Komponente verschieden sein kann. Wählt man $\chi = 0.5$ im Falle der einfachen intermediären Rekombination, dann wirkt solch ein Mittelungsprozeß natürlich diversitätsmindernd. In Bezug auf die Objektvariablen scheint dies nicht günstig zu sein, in Bezug auf die Strategievariablen allerdings eher günstig, weil es einer Überanpassung entgegenwirkt.

Theoretische Aussagen zu Evolutionären Algorithmen

Für die (1+1)–ES hat Born [Bor78] schon sehr früh einen Nachweis erbracht, daß — wie für stochastische Algorithmen üblich — eine ϵ-Nachbarschaft des (globalen!) Optimums im Grenzfall unendlich vieler Iterationen mit Wahrscheinlichkeit Eins gefunden wird. Eine stringentere Form dieses *Konvergenzbeweises* stammt von Rudolph [Rud92]. Sie beruht auf folgenden recht moderaten Forderungen an die Problemstruktur bzw. die verwendete Wahrscheinlichkeitsdichteverteilung für die Mutationen:

$$\begin{array}{ll} \text{(I)} & f^* > -\infty \\ \text{(II)} & \exists \epsilon > 0 \ : \ \mu(L_{f^*+\epsilon}) > 0 \\ \text{(III)} & p(\Delta x) \text{ stückweise stetig bzw. } > 0 \ \forall \Delta x \\ \text{(IV)} & \sum\limits_{t=0}^{\infty} p^t(\Delta x) = \infty \end{array} \tag{24}$$

Die zweite Forderung nach einem positiven Lebesgue-Maß der durch die Genauigkeitsforderung ϵ abgesteckten Niveau-Menge dürfte direkt einleuchten, da anderenfalls die Suche nach dem Optimum der sprichwörtlichen Suche nach einer Nadel im Heuhaufen gleich käme.

Wichtig bei der Anwendung der 1/5–Erfolgsregel nach Rechenberg ist laut (III) und (IV), daß man eine untere Schranke für die Mutations– „Schrittweite“ $\sigma \geq \sigma_{\min} > 0$ setzt. Aus praktischen Gründen wird man sie so wählen, daß sich Mutationen wenigstens auf der letzten vom Rechner gespeicherten Stelle der Objektvariablen noch bemerkbar machen.

Für die Genetischen Algorithmen gibt es einen ähnlichen *Konvergenzbeweis* erst durch Hartl [Har90], wobei neben der Forderung nach positiver Mutationswahrscheinlichkeit eine normalerweise nicht beachtete Zusatzbedingung eingeführt werden mußte: daß nämlich die jeweils bisherige Bestlösung im Genpool erhalten bleiben muß, wenn alle Nachkommen Verschlechterungen darstellen (Monotonieforderung).

Analog dazu kann solch ein Konvergenzbeweis für die mehrgliedrigen ES nur für die Plus-Variante, nicht jedoch für die Komma-Variante geführt werden, bei der die Lebensdauer der Eltern generell auf nur eine Generation beschränkt ist. Andererseits erwies sich das Prinzip des Vergessens für das Erlernen der passenden Strategievariablen als äußerst nützlich [Sch87]. Noch nicht systematisch untersucht wurde der zwischen Komma- und Plus-Varianten anzusiedelnde Fall einer künstlichen mehrgliedrigen Evolution mit zwar endlicher, aber über eine Generation hinausgehender maximaler Lebensdauer $1 < \kappa < \infty$ der Individuen.

Wichtiger noch als Konvergenzbeweise — jedenfalls für den praktischen Einsatz der Verfahren — sind Betrachtungen zur *Konvergenzgeschwindigkeit* bzw. Zeitkomplexität.

Für Genetische Algorithmen wird dazu üblicherweise auf der Basis der sog. *Schemata* (Unterräume des l-dimensionalen Sequenzraumes von Binärfolgen) argumentiert, wobei die Wirkung der Selektion im Vordergrund steht. Schemata werden als Elemente $H \in \{0,1,*\}^l$ notiert, wobei ein Bitstring $a \in \{0,1\}^l$ *Instanz* von H heißt, wenn a an allen Positionen mit H identisch ist, an denen in H eine Null oder Eins steht. Der „$*$" hat die Bedeutung eines Wildcard–Symbols. Formal ist die Menge $\mathcal{I}(H)$ aller Instanzen eines Schemas H beschreibbar als Abbildung

$$\begin{aligned}&\mathcal{I} : \{0,1,*\}^l \to \mathcal{P}(I)\\ &\mathcal{I}(H = (\gamma_1 \dots \gamma_l)) = \{(\alpha_1 \dots \alpha_l) \mid \forall i \in \{1,\dots,l\} :\ \gamma_i \in \{0,1\} \Rightarrow \alpha_i = \gamma_i\}\,.\end{aligned} \tag{25}$$

Im folgenden wird anstelle $a \in \mathcal{I}(H)$ einfach $a \in H$ geschrieben. Die wesentliche Idee bei der Betrachtung von Schemata liegt darin, daß es gute Schemata sind, die sich aufgrund der Selektion nach und nach in der Population durchsetzen — sofern sie nicht durch Rekombination oder Mutation zerstört werden.

Um die Wirkung von Rekombination und Mutation auf Schemata quantifizieren zu können, benötigt man Kenngrößen für Länge und Spezifität der Schemata. Die *Ordnung* $o(H)$ eines Schemas H wird durch die Anzahl fixierter Positionen in H gegeben, d.h.

$$\begin{aligned}&o : \{0,1,*\}^l \to \{0,\dots,l\}\\ &o(H = (\gamma_1 \dots \gamma_l)) = |\{i \mid \gamma_i \in \{0,1\}\}| \quad ,\end{aligned} \tag{26}$$

während man unter der *definierenden Länge* $\delta(H)$ den Abstand zwischen den äußersten fixierten Positionen in H versteht:

$$\begin{aligned}&\delta : \{0,1,*\}^l \to \{0,\dots,l-1\}\\ &\delta(H = (\gamma_1 \dots \gamma_l)) = \max\{i \mid \gamma_i \in \{0,1\}\} - \min\{i \mid \gamma_i \in \{0,1\}\} \quad .\end{aligned} \tag{27}$$

Mit Hilfe dieser Maße ergibt sich die Wahrscheinlichkeit, daß ein Schema H durch Ein-Punkt-Crossover nicht zerstört wird, zu

$$1 - p_r \cdot \frac{\delta(H)}{l-1} \quad , \tag{28}$$

und die Wahrscheinlichkeit, daß es den Mutationsoperator unbeschadet übersteht, zu

$$(1-p_m)^{o(H)} \approx 1 - o(H) \cdot p_m \quad \text{für } p_m \ll 1 \quad . \tag{29}$$

Um die Selektion mit einzubeziehen, betrachtet man die Selektionswahrscheinlichkeiten der Individuen a_i^t einer Population $P^t = (a_1^t, \dots, a_\lambda^t)$ gemäß *proportionaler* Selektion:

$$p_s(a_k^t) \;=\; \frac{f(a_k^t)}{\sum_{j=1}^{\lambda} f(a_j^t)} \quad . \tag{30}$$

$m(H^t)$ bezeichne die Anzahl von Individuen aus der Population P^t, welche Instanzen des Schemas H^t sind. Für die Veränderung dieser Anzahl unter Wirkung der Selektion erhält man:

$$\begin{aligned} m(H^{t+1}) &= m(H^t)\, f(H^t)/\bar{f}^t \\ \bar{f}^t &= \frac{1}{\lambda}\sum_{j=1}^{\lambda} f(a_j^t) \\ f(H^t) &= \frac{1}{m(H^t)} \sum_{a_j^t \in H^t} f(a_j^t) \quad . \end{aligned} \tag{31}$$

Hierbei ist $\bar{f}^t$ die durchschnittliche Fitneß der Gesamtpopulation, und $f(H^t)$ bezeichnet die durchschnittliche Fitneß derjenigen Individuen aus der Population, welche zugleich Instanzen von H^t sind. Falls nun ein Schema H^t überdurchschnittlich hohe Fitneß aufweist, d.h. $f(H^t) = \bar{f}^t + c \cdot \bar{f}^t$ mit konstant angenommenem $c > 0$, so gilt

$$m(H^{t+1}) \;=\; m(H^t) \cdot (1 + c) \tag{32}$$

bzw., beginnend in Generation Null, nach t Zeitschritten:

$$m(H^t) \;=\; m(H^0) \cdot (1 + c)^t \quad . \tag{33}$$

Selektion allein läßt also die Zahl der Instanzen überdurchschnittlich guter Schemata in der Population exponentiell anwachsen auf Kosten der Schemata mit Fitneßwerten unterhalb des Durchschnitts. In Kombination mit den Überlebenswahrscheinlichkeiten für Rekombination und Mutation ergibt sich (unter Vernachlässigung von Termen höherer Ordnung) die als *Schema-Theorem* bekannte zentrale Aussage der GA-Theorie [Hol75]:

$$m(H^{t+1}) \;\geq\; m(H^t) \cdot \frac{f(H^t)}{\bar{f}^t} \cdot \left(1 - p_r \frac{\delta(H^t)}{l - 1} - o(H^t) p_m\right) \quad . \tag{34}$$

Das Schema-Theorem besagt, daß überdurchschnittlich gute, kurze Schemata geringer Ordnung (sog. *building blocks*) innerhalb der nachfolgenden Generationen exponentiell vermehrt werden. Die Anzahl der schlechteren Schemata in der Population nimmt entsprechend exponentiell ab. Durch Verwendung eines spieltheoretischen Argumentes auf der Basis einer Analogie zum zwei- und mehrarmigen Banditen charakterisiert Holland die proportionale Selektion als nahezu optimale Strategie [Hol75]. Es sei an dieser Stelle darauf hingewiesen, daß die praktische Bedeutung dieser Theorie mehr und mehr umstritten ist, weil insbesondere die endliche Populationsgröße der Implementierungen den Voraussetzungen der Theorie nicht gerecht wird.

Für $(1{+}\lambda)$-Strategien, insbesondere für $\lambda = 1$, liefe eine ähnliche Betrachtung darauf hinaus, bei konstantem σ die Zielannäherung beim Kugelmodell f_1 bzw. den zurückgelegten Weg pro Generation beim Korridormodell f_2 zu betrachten. Dies hat Rappl getan [Rap84]. Aber schon Rechenberg ging weit darüber hinaus, soweit es die zweigliedrige ES betrifft. In Abhängigkeit von σ berechnete er den Erwartungswert für die

Zielannäherung bzw. den zurückgelegten Weg, deren Werte pro Generation er *Fortschrittsgeschwindigkeit* φ nannte. Folgende Gesetzmäßigkeiten fand er heraus:

$$\begin{aligned} \varphi_1 &= \frac{\sigma}{\sqrt{2\pi}}\left(1-\sqrt{\frac{2}{\pi}}\frac{\sigma}{b}\right)^{n-1} && \text{für } n \gg 1 \\ \varphi_2 &= \frac{\sigma}{\sqrt{2\pi}}\left(\exp\left(-\left(\frac{n\sigma}{\sqrt{8}r}\right)^2\right)-\sqrt{\pi}\frac{n\sigma}{\sqrt{8}r}\left(1-\operatorname{erf}\left(\frac{n\sigma}{\sqrt{8}r}\right)\right)\right) && \text{für } n \gg 1 \quad , \end{aligned} \tag{35}$$

wobei $\operatorname{erf}(x)$ die Gaußsche Fehlerfunktion, r den Abstand zum Ziel bzw. örtlichen Krümmungsradius beim Kugelmodell und b die Korridorbreite bzw. b^{n-1} den Querschnitt beim Korridormodell darstellt. Ebenso betrachtete er die Erfolgswahrscheinlichkeit, das ist der Erwartungswert des Verhältnisses von erfolgreichen Mutationen zu allen Mutationen:

$$\begin{aligned} w_1 &= \frac{1}{2}\left(1-\sqrt{\frac{2}{\pi}}\frac{\sigma}{b}\right)^{n-1} && \text{für } n \gg 1 \\ w_2 &= \frac{1}{2}\left(1-\operatorname{erf}\left(\frac{n\sigma}{\sqrt{8}r}\right)\right) && \text{für } n \gg 1 \quad . \end{aligned} \tag{36}$$

Von besonderem Interesse sind natürlich Lage und Höhe der Maxima, also σ_{opt} und $\varphi_{max} = \varphi(\sigma_{opt})$. Diese lassen sich durch Nullsetzen der Ableitungen analytisch ermitteln zu:

$$\begin{aligned} \sigma_1^{\text{opt}} &= \sqrt{\frac{\pi}{2}}\cdot\frac{b}{n} \quad &&; \quad \varphi_1^{\max} = \frac{1}{2e}\cdot\frac{b}{n} \\ \sigma_2^{\text{opt}} &\approx 1.224\cdot\frac{r}{n} \quad &&; \quad \varphi_2^{\max} \approx 0.2025\cdot\frac{r}{n} \end{aligned} \tag{37}$$

mit den dazugehörigen Erfolgswahrscheinlichkeiten

$$\begin{aligned} w_1^{\text{opt}} &= \frac{1}{2e} \approx 0.184 \approx \frac{1}{5.4} \\ w_2^{\text{opt}} &\approx 0.270 \approx \frac{1}{3.7} \quad . \end{aligned} \tag{38}$$

Aus diesem Ergebnis erkennt man, wieso Rechenberg auf die Idee der 1/5–Erfolgsregel kam. Leider gibt es aber auch Situationen, in denen deren Anwendung zu einer vorzeitigen Konvergenz führt, z.B. wenn die Lösung am Rand des zulässigen Bereichs liegt, oder die lokale Topologie der Höhenlinien konstanter Zielfunktionswerte Ecken aufweist, so daß selbst im Grenzfall gegen Null gehender Mutationsschrittweiten die erzielbare maximale Erfolgswahrscheinlichkeit unterhalb 0.2 liegt. Man kann übrigens „dimensionslose" Größen $\frac{\varphi\cdot n}{r}$ bzw. $\frac{\varphi\cdot n}{b}$ und $\frac{\sigma\cdot n}{r}$ bzw. $\frac{\sigma\cdot n}{b}$ einführen, aus deren Auftragung gegeneinander man ablesen kann, daß Mutationsschrittweiten nicht exakt auf ihren Optimalwert eingestellt werden müssen. Innerhalb der Bandbreite einer ganzen Zehnerpotenz für σ — von Rechenberg als *Evolutionsfenster* bezeichnet — erzielt man mindestens 50% der maximal möglichen Fortschrittsgeschwindigkeit.

Strukturell gleiche Betrachtungen für die $(1+\lambda)$–ES und $(1,\lambda)$–ES liegen ebenfalls seit langem vor [Sch75, Sch81]. Da die mehrgliedrigen ES in der Lage sein sollten, die

Mutabilitäten selbst zu adaptieren, ist eine exogene Steuerung nicht nötig, also auch keine 1/5–Erfolgsregel. Auffällig ist das Verhalten der Komma–Varianten bei zu großen Werten für σ. Dann kann nämlich die Konvergenzgeschwindigkeit sogar *negativ* werden. Selbst der beste unter λ Nachkommen, der ja zum Elter der nächsten Generation avanciert, kann dann schlechter als sein Vorfahr sein. Bei der Plus-Variante würde der Vorfahr überleben und weitere Nachkommen erzeugen, bis einer davon ihn übertrifft. Das ist zwar nicht naturgemäß, aber im Sinne der effizienten Optimumsuche scheinbar günstiger. Ebenso unsinnig scheint dann auch die Maßnahme zu sein, von λ Nachkommen nicht nur den besten, sondern auch schlechtere, bis hin zum nur noch μ.–besten zu Eltern der nächsten Generation werden zu lassen. Da beide Argumente, pro Komma–Variante und pro $\mu = 1$ aus der theoretischen Analyse so evident sind, stießen lange Zeit (μ,λ)–ES auf Skepsis bis Ablehnung. Das einzige, aber doch sehr starke Argument dafür ist die theoretisch bisher leider noch nicht untermauerte, aufgrund von Computer-Experimenten aber ganz überzeugende Tatsache, daß die *Selbstadaptation* der strategischen Parameter sonst nicht funktioniert. Drei Bedingungen konnten für das erfolgreiche Lernen auf zwei Ebenen herausgefiltert werden:

(1) Das *Vergessen* selbst guter Positionen im Raum der Objektvariablen muß zugelassen werden, um Überanpassungen beim Aufbau des internen Modells ausmerzen zu können. Die Komma-Strategie ist also entgegen den aus der Theorie ableitbaren Vermutungen günstiger, besonders, wenn λ klein ist.

(2) Die Selektion darf nicht zu hart sein. Der jeweils beste Nachkomme (Experte) hat nicht das für den weiteren Fortschritt „richtige“ interne Modell. Das „Wissen“ ist in der Population *verteilt*. Diese muß eine Balance zwischen Exploration und Ausnutzung des Wissens aufrechterhalten, was nur durch hinreichende Diversität, auch auf der Ebene der internen Modelle der Individuen, gewährleistet werden kann.

(3) *Rekombination* ist ein essentieller Faktor, um — besonders in kleinen Populationen — die in den Genen gespeicherten Informationen effektiv nutzen zu können. In Bezug auf die Objektvariablen (x) scheint die diskrete Rekombination, in Bezug auf die strategischen Parameter die intermediäre Rekombination günstiger zu sein.

Interessant ist in diesem Zusammenhang auch die Beobachtung, daß hohe Konvergenzgeschwindigkeiten der Population nicht etwa aus einer sehr exakten Anpassung der internen Modelle der einzelnen Individuen resultieren. Vielmehr ist das Kollektiv durch sein verteiltes Wissen erfolgreich, wobei insbesondere die Rekombination synergistische Effekte freisetzt im Sinne von „Das Ganze ist mehr als die Summe seiner Teile“ [Sch87].

Parallelrechner–Implementierungen

Ein Motiv für den Entwurf mehrgliedriger ES war die Ausnutzung künftiger Parallelrechnerarchitekturen. Die natürliche Evolution ist ein Paradebeispiel für einen parallelen Lernalgorithmus. Es bietet sich geradezu an, die Auswertung der Zielfunktionswerte für die Nachkommen einer Generation gleichzeitig vorzunehmen. Dies ist für praktische Anwendungen häufig der aufwendigste Teil des Algorithmus. Hierbei kann sogar auf Synchronität verzichtet werden. Während ein Individuum schon bei Auswertung der ersten Restriktion als letal erkannt wird, muß bei einem anderen vielleicht nach allen Restriktionsfunktionen auch noch die Zielfunktion berechnet werden (oder umgekehrt — je nach Anwendung). Dann entstünden unnötige Wartezeiten, um die Generationen zu takten. Statt eines global kontrollierten *Master-Slave* Prinzips sollte daher eine dezentralisierte, auf lokalen Interaktionen basierende *Message-Passing* Implementierung sinnvoller sein.

Man kann aber statt einer solchen feinkörnigen Parallelität auch eine grobkörnige konzipieren [Hof91, Rud91]. Hierbei beherbergt jeder Prozessor, z.B. einer Transputer-Farm, jeweils eine ganze Teilpopulation (*Dem*), die je für sich evolviert. Von Zeit zu Zeit oder auch regelmäßig können per *Migration* die ansonsten isolierten Deme Information austauschen. Für Probleme der globalen, z.B. auch diskreten Optimierung, scheint unvollständiger Informationsaustausch eher von Vorteil zu sein. Nur für den unimodalen Fall läßt sich aus der Theorie für das Kugelmodell das optimale Verhältnis von μ zu λ ableiten, eigentlich ja nur für $\mu = 1$.

Auf einem SISD-Rechner müssen alle Nachkommen einer Generation seriell erzeugt und bewertet werden. Dann ist also das Maximum von φ/λ gesucht. Für die Plus-Variante liegt das zugehörige λ^{opt} bei Eins. Dann ist die (1+1)–ES vorzuziehen. Bei der Komma-Variante liegen für die beiden Modellfunktionen diese Werte bei

$$\begin{aligned} \lambda_1^{\text{opt}} &= 6 \\ \lambda_2^{\text{opt}} &= 5 \quad . \end{aligned} \tag{39}$$

Hat man einen MIMD-Rechner zur Verfügung, so ist λ^{opt} gleich der Anzahl der einsetzbaren Prozessoren. Fällt davon einmal einer aus, so wird die Suche höchstens etwas langsamer. Die mehrgliedrige ES ist also extrem fehlertolerant. Das optimale Verhältnis zwischen Eltern und Nachkommen liegt nach einschlägigen Tests mit einer Zielfunktion

$$f_3(x) = \sum_{i=1}^{n} i \cdot x_i^2 \quad , \tag{40}$$

bei der die Mutationsraten sich laufend und passend zueinander einstellen müssen, um maximale Konvergenzgeschwindigkeiten zu erreichen, bei etwa 12/100 oder 1/8, wie aus der Abb. 1 zu entnehmen ist. Für multimodale Topologien sollte der Selektionsdruck eher verringert werden, also bei konstantem λ die Zahl μ der Eltern erhöht werden.

Bezüglich paralleler GA Implementierungen sei auf [GS91, SM91b, SWM91] verwiesen. Einen allgemeinen Überblick über die Parallelisierungsmöglichkeiten Evolutionärer Algorithmen gibt Hoffmeister [Hof91].

Anwendungen und Ausblick

Ein vor kurzem unternommener Versuch, aus der einschlägigen Literatur Anwendungen von Evolutionären Algorithmen zusammenzustellen, war äußerst ergiebig [BHS92]. Es

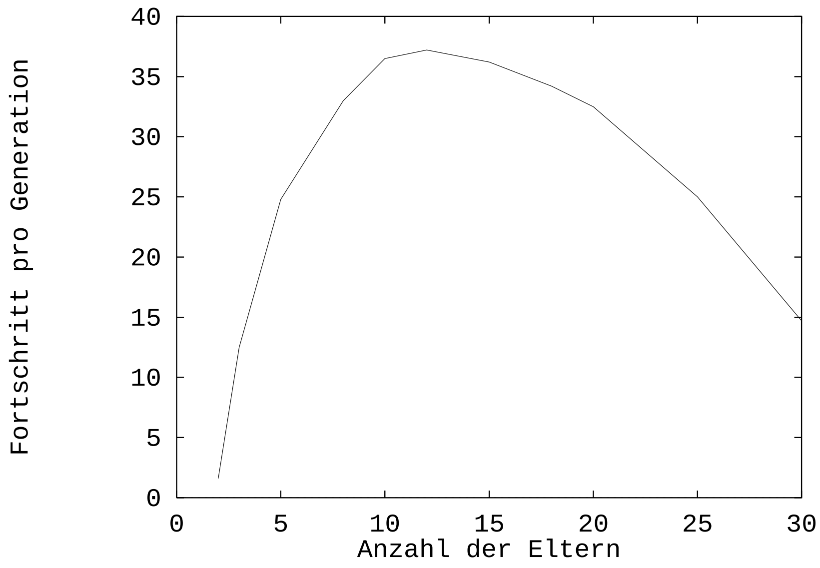

Abbildung 1: Fortschrittsgeschwindigkeit einer $(\mu,100)$–ES für verschiedene Werte von μ

gibt kaum eine Anwendungsdisziplin, die hierbei fehlt. Bezüglich eines umfangreichen vergleichenden Tests aller Varianten der Evolutionsstrategien mit einer Reihe traditioneller numerischer Optimierverfahren muß hier auf die Literatur [Sch77, Sch81] verwiesen werden. Angemerkt sei lediglich, daß die mehrgliedrigen Varianten die höchste *Zuverlässigkeit* bei 50 Testproblemen mit oder ohne Restriktionen aufwiesen, ja sogar oftmals das globale unter mehreren lokalen Minima fanden. Mit der Zahl der Variablen stieg die Rechenzeit bei einem quadratischen Problem kubisch an, was den Erwartungen für ein pfadorientiertes Verfahren entspricht. Die oben erwähnten Polytop–Strategien schnitten erheblich schlechter ab. Dividiert durch die Zahl der im Prinzip parallel ausführbaren Operationen für die $\lambda = 100$ Nachkommen pro Generation, lieferte eine (10,100)–ES sogar die zweithöchsten Konvergenzraten — nach der auf quadratische Probleme zugeschnittenen Methode der variablen Metrik von Davidon.

Was dringend erforderlich wäre, ist eine Erweiterung der theoretischen Basis. Es fehlt z.B. eine wohldefinierte Abgrenzung des Einsatzbereichs Evolutionärer Algorithmen. Niemand wird sie vernünftigerweise dort anwenden, wo traditionelle Optimiermethoden ihre Domäne haben, beispielsweise in der konvexen Optimierung. Andererseits sind EA keine Wunderwaffe bei der Suche nach der sprichwörtlichen Stecknadel im Heuhaufen. Ihre Vorzüge haben sie auf einem — eben noch nicht definierten — Zwischenterrain. ES können multimodale Probleme recht gut lösen, wenn es einen halbwegs monoton ansteigenden Weg zum Maximum bzw. Abstiegsweg zum Minimum gibt, z.B. bei Ackleys Funktion (Abb. 2, links). Selbst bei fraktaler Antwortfläche (Abb. 2, rechts) lassen sie einen nicht ganz im Stich. Stochastischen Störungen gegenüber sind sie unempfindlich, und die Komma–Varianten können ohne weiteres einem davondriftenden Optimum folgen [KS92].

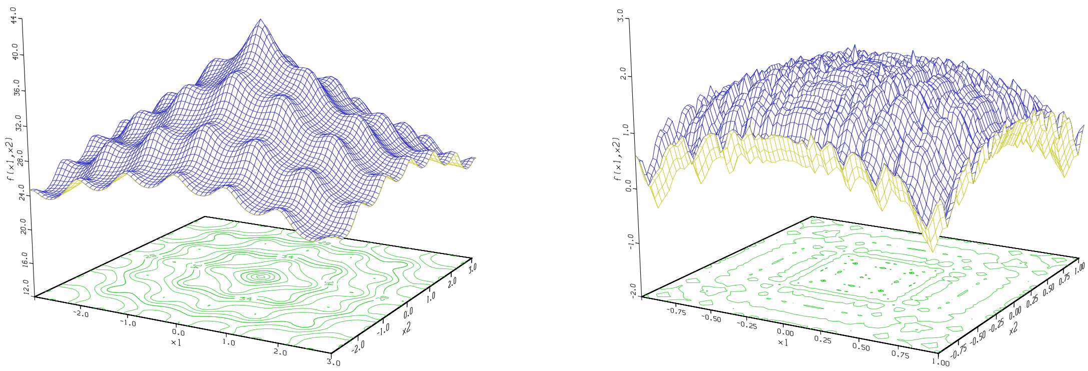

Abbildung 2: Ackleys Funktion (links) und fraktale Weierstrass-Mandelbrot Funktion (rechts)

Leider scheint es aber außerordentlich schwierig zu sein, entsprechende theoretisch fundierte Nachweise solcher experimentellen Befunde zu liefern, besonders für $\mu > 1$, verallgemeinerte maximale Lebensdauer, Selbstadaptation der kompletten Kovarianzmatrix der Mutationswahrscheinlichkeitsdichte, asynchrone Parallelität etc..

Ähnliches gilt für die GA, die auch nicht nur zur Lösung von Optimierungsaufgaben, wie hier dargestellt, taugen. Ihr größter Unterschied zu den ES besteht wohl darin, daß die Suche durch die Bitstring–Operationen einen mehr volumenorientierten Charakter erhält. Sie eignen sich besonders für Aufgaben, bei denen sich die zu variierenden Größen in natürlicher Weise durch ein Alphabet niedriger Kardinalität darstellen lassen, z.B. im Bereich des *induktiven Lernens* mit *Classifier Systemen* [BGH89] oder (in eingeschränkter Form) der *Genetischen Programmierung* [Koz89].

Gelungen ist inzwischen auch eine Erweiterung auf den Fall der *mehrfachen Zielsetzung* bei den ES [Kur91]. Hierbei erwies sich die Nachahmung der Polyploidie (mehr als ein Gensatz pro Individuum) mit Dominanz und Rezessivität als besonders erfolgreich. Auf diese Weise kann die Population die Menge der paretooptimalen Lösungen in einem Zuge aufspüren. Der Entscheidungsträger braucht also nicht im vorhinein oder im Dialog subjektive Wertungen einzubringen, um zu einer Kompromißlösung zu gelangen, wie es bei herkömmlichen MCDM-Ansätzen (multiple criteria decision making) erforderlich ist.

Besonders vielversprechend scheinen auch *hybride Formen* aus GA und ES zu werden. So gelang erstmals eine Selbstadaptation der Mutationsrate bei einem Genetischen Algorithmus, der die (μ,λ)-Selektion übernahm [Bäc91]. Für einen speziellen Fall konnte Bäck neben der Formulierung einer Abschätzung der Konvergenzgeschwindigkeit sogar zeigen, daß die Selbstadaptation der Mutationsrate den Algorithmus zu einer Erzeugung des nahezu optimalen Verlaufes der Mutationsrate befähigt, womit der Wirkungsmechanismus der Selbstadaptation einer Erklärung näher gebracht wurde. Weitere Rekombinationstypen von Evolutionären Algorithmen sind von den nun auch diesseits des Atlantiks alle zwei Jahre stattfindenden PPSN-Konferenzen (Parallel Problem Solving from Nature, siehe [SM91b]) zu erwarten.

Ganz am Rande sei noch darauf hingewiesen, daß Evolutionäre Algorithmen auch als dynamische Simulationsmodelle für andere Multi-Agenten-Systeme aufgefaßt werden können. Deren Verhaltensweisen sind beispielswesie für Ökologen [AL91] und Ökonomen [Wit92] von großem Interesse für das Verständnis von Phänomenen in der realen Welt.

Sind nun — um die Frage aus der Überschrift wieder aufzugreifen — die hier betrachteten Algorithmen intelligente Problemlösungsstrategien? Einerseits sind heutige Evolutionäre Algorithmen erst noch recht primitive Nachahmungen von einigen Mechanismen der biologischen Evolution. Deren erstaunliche Leistungen begründen andererseits aber immer noch die Hoffnung, daß sich durch besseres Verständnis realer Phänomeme, gepaart mit adäquater Nachahmung — hier mit „Künstliche Evolution“ bezeichnet — in Zukunft noch mehr Gewinn aus solchen Modellen kollektiver Verhaltensweisen ziehen läßt. Auch bei der Problemlösung durch Menschen setzen Informatiker schließlich auf Teamarbeit, um effektiv und effizient zu sein.

Literatur

[AL91] P. M. Allen und M. J. Lesser. Evolution: Travelling in an imaginary landscape. In J. D. Becker, I. Eisele, and F. W. Mündemann, (Hrsg.), *Parallelism, Learning, Evolution*, *Lecture Notes in Artificial Intelligence* Vol. 565, S. 419–441. Springer, Berlin, 1991.

[Ash60] W. R. Ashby. *Design for a Brain.* Wiley & Sons, New York, 2. Auflage, 1960.

[Bäc91] Thomas Bäck. Self-adaptation in genetic algorithms. In *Proceedings of the First European Conference on Artificial Life, 11.-13. Dezember*, Paris, 1991. The MIT Press.

[Bak85] James Edward Baker. Adaptive selection methods for genetic algorithms. In J. J. Grefenstette, (Hrsg.), *Proceedings of the First International Conference on Genetic Algorithms and Their Applications*, S. 101–111, Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1985.

[BGH89] L. B. Booker, D. E. Goldberg, und J. H. Holland. Classifier systems and genetic algorithms. *Artificial Intelligence*, S. 235–282, 1989.

[BHS92] Thomas Bäck, Frank Hoffmeister, und Hans-Paul Schwefel. Applications of evolutionary algorithms. Report des Lehrstuhls für Systemanalyse (LS XI) SYS–2/92, Universität Dortmund, Fachbereich Informatik, 1992.

[Bor78] Joachim Born. *Evolutionsstrategien zur numerischen Lösung von Adaptationsaufgaben.* Dissertation A, Humboldt-Universität, Berlin, 1978.

[Box65] M.J. Box. A new method of constrained optimization and a comparison with other methods. *The Computer Journal*, 8:42–52, 1965.

[Gol89] David E. Goldberg. *Genetic algorithms in search, optimization and machine learning.* Addison Wesley, Reading, Massachusetts, 1989.

[Gre87] John J. Grefenstette. *A User's Guide to GENESIS.* Navy Center for Applied Research in Artificial Intelligence, Washington, D. C., 1987.

[GS91] Martina Gorges-Schleuter. Explicit parallelism of genetic algorithms through population structures. In [SM91a], S. 150–159.

[Har90] Richard F. Hartl. A global convergence proof for a class of genetic algorithms. Technische Universität Wien, 1990.

[Hof91] Frank Hoffmeister. Scalable parallelism by evolutionary algorithms. In M. Grauer und D. B. Pressmar (Hrsg.), *Parallel Computing and Mathematical Optimization*, *Lecture Notes in Economics and Mathematical Systems*, Vol. 367, S. 177–198, Springer, Berlin, 1991.

[Hol62] John H. Holland. Outline for a logical theory of adaptive systems. *Journal of the Association for Computing Machinery*, 3:297–314, 1962.

[Hol75] John H. Holland. *Adaptation in natural and artificial systems.* The University of Michigan Press, Ann Arbor, 1975.

[Jon75] Kenneth De Jong. *An analysis of the behaviour of a class of genetic adaptive systems.* PhD thesis, University of Michigan, 1975. Diss. Abstr. Int. 36(10), 5140B, University Microfilms No. 76–9381.

[Koz89] John R. Koza. Hierarchical genetic algorithms operating on populations of computer programs. In N. S. Sridharan, (Hrsg.), *Eleventh international joint conference on artificial intelligence*, S. 768–774. Morgan Kaufmann Publishers, Los Altos, California, 1989.

[KS92] Künstliche Evolution als Modell fï natürliche Intelligenz. Bionik–Kongreß, Wiesbaden, Juni 1992 (im Druck).

[Kur91] Frank Kursawe. A variant of Evolution Strategies for vector optimization. In [SM91a], S. 193–197.

[NM65] J.A. Nelder und R. Mead. A simplex method for function minimization. *The Computer Journal*, 7:308–313, 1965.

[Rap84] Georg Rappl. *Konvergenzraten von Random-Search-Verfahren zur globalen Optimierung.* Dissertation, Hochschule der Bundeswehr München, November 1984.

[Rec71] Ingo Rechenberg. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.* Dissertation, Technische Universität Berlin, 1971.

[Rec73] Ingo Rechenberg. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.* Frommann–Holzboog Verlag, Stuttgart, 1973.

[Rud91] Günter Rudolph. Global optimization by means of distributed evolution strategies. In [SM91a], S. 209–213.

[Rud92] Günter Rudolph. Parallel approaches to stochastic global optimization. In *Proceedings of the European Workshop on Parallel Computing*, Barcelona (Sitges), Spanien, 23.-24. März 1992. IOS Press, Amsterdam 1992.

[Sch75] Hans-Paul Schwefel. *Evolutionsstrategie und numerische Optimierung.* Dissertation, Technische Universität Berlin, Mai 1975.

[Sch77] Hans-Paul Schwefel. *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie*, *Interdisciplinary systems research*, Vol. 26. Birkhäuser, Basel, 1977.

[Sch81] Hans-Paul Schwefel. *Numerical Optimization of Computer Models.* Wiley, Chichester, 1981.

[Sch87] Hans-Paul Schwefel. Collective phenomena in evolutionary systems. In *Preprints of the 31st Annual Meeting of the International Society for General System Research, Budapest*, Vol. 2, S. 1025–1033, Juni 1987.

[SM91a] Hans-Paul Schwefel und Reinhard Männer, (Hrsg.). *Parallel Problem Solving from Nature*, *Lecture Notes in Computer Science* Vol. 496. Springer, Berlin, 1991.

[SM91b] Piet Spiessens und Bernard Manderick. A massively parallel genetic algorithm: Implementation and first analysis. In Richard K. Belew und Lashon B. Booker, (Hrsg.), *Proceedings of the Fourth International Conference on Genetic Algorithms and their Applications*, S. 279–286, Morgan Kaufmann Publishers, San Mateo, California, 1991.

[SWM91] T. Starkweather, D. Whitley, und K. Mathias. Optimization using distributed genetic algorithms. In Schwefel und Männer [SM91a], S. 176–186.

[Wit92] U. Witt, (Hrsg.). *Explaining Process and Change — Approaches to Evolutionary Economics.* The University of Michigan Press, Ann Arbor, 1992.