Towards a Theory of Representations for Genetic and Evolutionary Algorithms: Development of Basic Concepts and their Application to Binary and Tree Representations

Franz Rothlauf

Universität Bayreuth
franz@rothlauf.com

Die in dem vorliegenden Beitrag zusammengefasste Dissertationsschrift beschäftigt sich mit der Theorie von Repräsentationen für Genetische und Evolutionäre Algorithmen (GEA). GEAs sind leistungsfähige, naturanaloge heuristische Lösungsraumsuchverfahren, welche insbesondere bei semiformalen Problemen, für welche keine analytische Problembeschreibung vorliegen, effektiv eingesetzt werden können. GEAs imitieren die Prinzipien der natürlichen Evolution und wenden genetische Operatoren auf eine Repräsentation des zu lösenden Problems an. Da bisher nur wenig theoretische Erkennntnisse über Repräsentationen vorlagen, war die Entwicklung von geeigneten Repräsentationen bisher überwiegend ein Ergebnis zufälligem Probierens. Im Folgenden werden grundlegende theoretische Konzepte für Repräsentationen entwickelt. Im speziellen wird untersucht, wie redundante Repräsentationen, Repräsentationen mit unterschiedlich skalierten Allelen und Repräsentationen mit niedriger Lokalität die Leistungsfähigkeit von GEAs beeinflussen. Mithilfe dieser Konzepte wird der Einfluß der Repräsentation auf die Leistungsfähigkeit von GEAs für ganzzahlige und baumförmige Optimierungsprobleme untersucht. Abschließend werden theoriegeleitet neue Repräsentationen für Bäume entwickelt. Es zeigt sich, dass durch die vorgestellten Modelle sowohl das Verhalten von GEAs vorhergesagt, als auch deren Leistungsfähigkeit vorteilhaft beeinflusst werden kann.

1 Zielsetzung und Struktur der Arbeit

Für einen effektiven Einsatz von genetischen und evolutionären Algorithmen (GEA) ist zusätzlich zu leistungsfähigen genetischen Operatoren die geschickte Wahl einer geeigneten Repräsentation notwendig. Bei der praktischen Anwendung von GEAs stellt sich allerdings das Problem, eine passende Repräsentation für das Problem aus der Vielzahl an vorhandenen, unterschiedlichen Repräsentationen auszuwählen. Obwohl gezeigt wurde, dass die Wahl der Repräsentation einen großen Einfluss auf die Leistungsfähigkeit von GEAs hat [LV90], existieren immer noch keine fundierten theoretischen Modelle für Repräsentationen. Die Auswahl von passenden Repräsentationen beruht entweder auf der Intuition des Entwicklers oder auf umfangreichen empirischen Untersuchungen.

Das Ziel dieser Arbeit ist es, zu untersuchen, wie Repräsentationen die Leistungsfähigkeit von GEAs beeinflussen. Die Arbeit soll theoretische Erklärungsmodelle und Konzepte entwickeln und sie auf das Design, die Auswahl, die Beurteilung und den Vergleich von Repräsentationen anwenden. Hierbei soll besonders darauf geachtet werden, dass die zu entwickelnde Theorie auch praktisch anwendbar ist und nicht nur ein abstraktes theoretisches Gebilde darstellt. Die Entwicklung anwendbarer theoretischer Grundlagen soll es ermöglichen, schnell und einfach zu beurteilen, wie die Leistungsfähigkeit von GEAs durch den Einsatz von unterschiedlichen Repräsentationen verändert wird.

Im Abschnitt 2 wird gezeigt, dass die Leistungsfähigkeit von GEAs von drei verschiedenen Eigenschaften von Repräsentationen abhängt: Erstens von der Redundanz, zweitens von der unterschiedliche Gewichtung der Allele, und drittens von der Lokalität. Anschliessend wird in Abschnitt 3 untersucht, wie unterschiedliche Bitketten-Repräsentationen für ganzzahlige Probleme und unterschiedliche Repräsentationen für Bäume das Verhalten von GEAs beeinflussen. Abschliessend zeigt Abschnitt 4 auf, wie aufbauend auf die in Abschnitt 2 entwickelten Konzepte, neue Repräsentationen systematisch entworfen werden können. Die Arbeit schliesst mit einer kurzen Zusammenfassung.

2 Entwicklung theoretischer Konzepte für Repräsentationen

Im Folgenden werden Modelle vorgestellt, welche beschreiben, wie Redundanz, Gewichtung von Allelen und Lokalität die Leistungsfähigkeit von GEAs verändern.

2.1 Redundanz

Eine Repräsentation wird als redundant bezeichnet, wenn im Durchschnitt einem Phänotyp mehr als ein Genotyp zugeordnet wird. Im Folgenden werden, ohne Beschränkung der Allgemeingültigkeit, Bitketten-Repräsentationen verwendet. Zur Beschreibung von redundanten Repräsentationen wird die Ordnung k_r der Redundanz und die Anzahl r der genotypischen Building Blocks (BBs) [Gol89] der Länge kk_r, welche den besten phänotypischen BB der Länge k repräsentieren, eingeführt. k bezeichnet hierbei die Ordnung eines BBs. k_r legt fest, durch wieviele Bits eines Genotyps ein Bit eines Phänotyps definiert wird.

Eine genauere Untersuchung der Eigenschaften von redundanten Repräsentationen zeigt, dass redundante Repräsentationen zu einer unterschiedlichen Anzahl von BBs in der Startgeneration eines GEAs führen. Basierend auf früheren Arbeiten [HCPGM97] kann damit beschrieben werden, wie die notwendige Populationsgröße n von der Redundanz der Repräsentation (beschrieben durch k_r und r) abhängt:

$$n \approx -\frac{2^{k_r k-1}}{r} \ln(\alpha) \frac{\sigma_{BB}\sqrt{\pi m'}}{d}.$$

Hierbei ist α die Wahrscheinlichkeit, dass ein GEA den besten BB nicht findet, σ_{BB} die

Varianz eines BBs, $m' = m - 1$, m die Anzahl der BBs, d der Unterschied zwischen der Fitness des besten und zweitbesten BBs und k die Ordnung der BBs.

Es zeigt sich, dass n mit $O\left(\frac{2^{k_r}}{r}\right)$ wächst. Mit zunehmendem r nimmt n also ab. Beim Einsatz von redundanten Repräsentationen, welche allen Phänotypen die gleiche Anzahl von Genotypen zuordnen, gilt $r = 2^{k(k_r-1)}$ und das Verhalten von GEAs ändert sich nicht.

Basierend auf früheren Arbeiten [TG93, MG96] kann zusätzlich bestimmt werden, wie die Anzahl der Generationen bis zur Konvergenz durch Redundanz beeinflusst wird:

$$t_{conv} = \frac{\sqrt{l}}{I}\left(1 + \frac{\pi}{2} - \frac{r}{2^{k_r k-1}}\right).$$

Hierbei bezeichnet I den Selektionsdruck und l die Länge der binären Zeichenkette. Die Anzahl der Generationen bis zur Konvergenz wird also kleiner mit zunehmenden $r/2^{k_r}$.

Ein Vergleich von empirischen Ergebnissen mit den theoretischen Modellen zeigt, dass die theoretischen Modelle den Einfluss von redundanten Repräsentationen akkurat beschreiben. Da mit zunehmendem $r/2^{k_r}$ sich die Leistungsfähigkeit von GEAs erhöht, haben redundante Repräsentationen einen hohen Einfluß auf n und t_{conv}. Bei Repräsentationen, welche BBs hoher Fitness unterrepräsentieren, sinkt die Leistungsfähigkeit von GEAs, wohingegen eine Überrepräsentation zu einer Leistungssteigerung führt.

Es muss allerdings berücksichtigt werden, dass r von der Komplexität und Struktur des zu optimierenden Problems abhängt. Daher sind keine Aussagen über r möglich, solange keine Kenntnisse über die Struktur der optimalen Lösung existieren. Anwendern, welche redundante Repräsentationen einsetzen möchten und kein Wissen über ihr Problem besitzen, ist zu raten, Repräsentationen, welche alle Phänotypen gleichmäßig repräsentieren, zu benutzen. Andernfalls könnte die zufällige, ungünstige Wahl einer redundanten Repräsentation mit $r < 2^{k(k_r-1)}$ im Vergleich zu redundanzfreien Repräsentationen zu einer Verschlechterung der Leistungsfähigkeit von GEAs führen.

2.2 Gewichtung von Allelen

Es ist bekannt [Gol89], dass durch den Einsatz von gleichmäßig skalierten Repräsentationen BBs während des Ablaufs eines GEAs implizit parallel gelöst werden. Bei nichtgleichmäßig skalierten Repräsentationen, das heißt die Allele haben unterschiedlichen Einfluss auf die Gestaltung des Phänotypen, werden die BBs hingegen sequentiell gelöst (Domino-Konvergenz). Dies bewirkt eine Vergrößerung der Anzahl der Generationen und eine Verringerung der Leistungsfähigkeit von GEAs, da durch genetische Drift niederwertige Allele zufällig festgelegt werden und nicht mehr durch den GEA gelöst werden können. Drift entsteht dadurch, dass niederwertige Bits nicht von Anfang an dem Selektionsdruck I unterliegen. Dadurch schwankt die Anzahl der Allele, welche eine 0 oder 1 repräsentieren, zufällig um $0.5n$. Falls die Populationsgröße n entsprechend klein ist, kann es vorkommen, dass ein Allel zufällig entweder nur Nullen oder nur Einsen repräsentiert. Dadurch wird der Wert dieses Allels zufällig festgelegt und kann nicht mehr durch den GEA bestimmt werden.

Beim Einsatz von nicht-gleichmäßig skalierten Repräsentationen kann die notwendige Populationsgröße n berechnet werden als:

$$n \approx -\frac{1}{2}\ln\left(\frac{\alpha}{1-\alpha}\right)\sqrt{\pi\left(\frac{4}{3}m-1\right)}. \tag{1}$$

Die Populationsgröße n hängt nur von der Anzahl der BBs m und der Wahrscheinlichkeit α ab. Erwartungsgemäß nimmt die Populationsgröße n mit zunehmender Anzahl an exponentiell skalierten BBs m zu. Für die Anzahl der Generationen t_{conv} ergibt sich

$$t_{conv} = l_s\frac{\pi}{2}\frac{\sqrt{m}}{I} = \frac{l}{\sqrt{m}}\frac{\pi}{2I},$$

wobei $l = l_s m$ die Länge der Bitkette, l_s die Länge eines exponentiell skalierten BBs und I den Selektionsdruck bezeichnet. Im Gegensatz zu einer gleichmäßig skalierten Repräsentation ($t_{conv} = O(\sqrt{l_s m})$) steigt die Anzahl der Generationen für exponentiell skalierte Repräsentation mit $O(l_s\sqrt{m})$.

Bei Berücksichtigung von Drift werden die theoretischen Modelle komplexer. Zwei unterschiedliche Modelle können hierfür entwickelt werden. Das "approximated-drift"-Modell basiert auf den Driftmodellen von [Kim64] und überschätzt die Populationsgröße. Das "stair-case"-Modell überschätzt n für $t < 1.4n$ und unterschätzt n für $t > 1.4n$. t bezeichnet hierbei die Zeit ($0 \leq t \leq t_{conv}$). Im Folgenden soll kurz das "approximated-drift"-Modell vorgestellt werden:

$$\alpha_{drift}(\lambda) = \left(1 - s'(\lambda\frac{\pi}{2}\sqrt{\pi m})\right)\alpha + \frac{1}{2}s'(\lambda\frac{\pi}{2}\sqrt{\pi m}),$$

wobei

$$s'(t) = \begin{cases} 0 & \text{for } t < -n\ln(2/3), \\ 1 - \frac{3}{2}\exp(-t/n) & \text{for } t > -n\ln(2/3). \end{cases}$$

$\lambda \in [1, l]$ ist die Trennungslinie zwischen den schon konvergierten und noch nicht konvergierten Allelen und α wird aus Gleichung 1 berechnet.

Die Gültigkeit der theoretischen Modelle kann empirisch verifiziert werden. Es zeigt sich hierbei, dass die theoretischen Modelle, welche Drift vernachlässigen, nur bei genügend großer Anzahl m und geringer Länge l_s der exponentiellen BBs das Verhalten von GEAs korrekt beschreiben. Die theoretischen Modelle hingegen, welche Drift mitberücksichtigen, liefern recht genaue Abschätzungen für die Populationsgröße, beziehungsweise Erfolgswahrscheinlichkeit eines GEAs.

Die Untersuchung von nicht-gleichmäßig skalierten Repräsentationen zeigt, dass sich die Dynamik von GEAs verändert. Im Gegensatz zu gleichmäßig skalierten Repräsentationen werden die Allele nicht implizit parallel sondern sequentiell gelöst. Dadurch vergrößert sich die Anzahl der notwendigen Generationen und Drift reduziert zusätzlich die Qualität der Lösungen. Der Vorteil von nicht-gleichmäßig skalierten Repräsentationen liegt allerdings darin, dass schon nach kurzer Zeit die Allele gelöst werden, welche den größten Einfluss auf die Gestaltung des Phänotyps haben.

2.3 Lokalität

Die Aufgabe von Repräsentationen ist es, jedem Phänotyp einen entsprechenden Genotypen zuzuordnen. Bei der Verwendung von Bitketten der Länge l können hierbei den insgesamt 2^l verschiedenen Phänotypen die 2^l Genotypen auf $2^l!$ unterschiedliche Weise redundanzfrei zugeordnet werden. Jede dieser $2^l!$ unterschiedlichen Zuordnungen stellt eine eigene Repräsentation dar. Bei der Verwendung von unterschiedlichen Repräsentationen werden allerdings nicht nur die Phänotypen unterschiedlichen Genotypen zugeordnet, sondern auch die BBs in den Phänotypen entsprechenden BBs in den Genotypen. Das bedeutet also, dass die Struktur, das heißt Ordnung und Länge von BBs, durch Repräsentationen geändert werden kann. Es ist also möglich, die Schwierigkeit eines Problems durch entsprechende Repräsentationen sowohl positiv als auch negativ zu beeinflussen.

Bei der Untersuchung des Einflusses von Repräsentationen auf die Struktur der BBs und damit die Leistungsfähigkeit von GEAs zeigt sich, dass die Lokalität einer Repräsentation dabei von entscheidender Bedeutung ist. Die Lokalität d_m einer Repräsentation kann definiert werden als

$$d_m = \sum_{d^p_{\boldsymbol{x}_i,\boldsymbol{x}_j}=d^p_{min}} |d^g_{\boldsymbol{x}_i,\boldsymbol{x}_j} - d^g_{min}|,$$

wobei $d^p_{\boldsymbol{x}_i,\boldsymbol{x}_j}$ die phänotypische Distanz zwischen den Individuen $\boldsymbol{x}_i$ und $\boldsymbol{x}_j$, $d^g_{\boldsymbol{x}_i,\boldsymbol{x}_j}$ die genotypische Distanz, und d^p_{min}, beziehungsweise d^g_{min} die minimale Distanz zwischen zwei benachbarten Phänotypen beziehungsweise Genotypen darstellt. Die Metriken auf dem phänotypischen und genotypischen Lösungsraum sind so skaliert, dass $d^g_{min} = d^p_{min}$.

Die Lokalität einer Repräsentation beschreibt, wie ähnlich sich benachbarte Phänotypen im genotypischen Lösungsraum sind. Für den Fall, dass alle benachbarte Phänotypen ($d^p = 1$) auch Nachbarn im genotypischen Lösungsraum sind ($d^g = 1$), gilt $d_m = 0$. Eine genauere Untersuchung der Auswirkungen von Lokalität zeigt, dass Lokalität eine Voraussetzung für den effizienten Einsatz von mutationsbasierten Suchverfahren ist. Je höher die Lokalität einer Repräsentation ist, desto einfacher können Suchverfahren bei einfachen Problemen die optimale Lösung finden.

Deswegen sollten, um sicherzustellen, dass einfache und Probleme beschränkter Komplexität sicher gelöst werden, beim praktischen Einsatz von GEAs auf die Verwendung von Repräsentationen mit geringer Lokalität verzichtet werden. Nur unter der Voraussetzung, dass das zu lösende Problem die Lösungsfähigkeit des GEAs übersteigt, kann der Einsatz von Repräsentationen mit geringer Lokalität unter Umständen von Vorteil sein.

3 Analyse vorhandener Repräsentationen

Die Aussagekraft der in der Arbeit entwickelten Konzepte von Repräsentationen kann durch die Analyse von vorhandenen Repräsentationen illustriert werden.

3.1 Repräsentationen für ganzzahlige Optimierungsprobleme

Die am meisten verbreiteten Repräsentationen für ganzzahlige Optimierungsprobleme sind unäre, Gray- und binäre Kodierung. Bei der unären Kodierung wird ein Phänotyp x_p kodiert durch die Anzahl u der 1en im Genotyp $\boldsymbol{x}_g$. Damit ergibt sich $x_p = \sum_{i=0}^{l-1} x_{g,i}$, wobei l die Länge der Bitkette und $x_{g,i}$ den Wert des $i$ten Allels im Genotyp bezeichnet. Zur Kodierung von s unterschiedlichen Phänotypen ist eine Bitkette der Länge $l = s - 1$ notwendig. Dies führt zu Redundanz. Bei einer Bitkettenlänge von l können insgesamt nur $l+1$ verschiedene Phänotypen kodiert werden. Demzufolge wird der Phänotyp $x_p = i$ mit $0 \leq i \leq l$ durch $\binom{l}{x_p}$ unterschiedliche Genotypen repräsentiert.

Bei der Verwendung der binären Kodierung genügt für die Kodierung von s verschiedenen Phänotypen eine Länge der Bitkette von $l = log_2(x_{p,max})$. Der Phänotyp eines binär kodierten Individuums lässt sich berechnen durch $x_p = \sum_{i=0}^{l-1} 2^i x_{g,i}$. $x_{g,i}$ bezeichnet hierbei wie zuvor das $i$te Bit im Genotyp $\boldsymbol{x_g}$. Die Verwendung der binären Kodierung führt zu Domino-Konvergenz und dem verstärkten Auftreten von genetischer Drift. Darüber hinaus tritt das Problem der sogenannten Hamming-Klippe auf [SCED89]. Es zeigt sich hierbei, dass benachbarte Phänotypen nicht immer auch benachbarte Genotypen sind (vergleiche $x_p = 7/x_g = 111$ und $x_p = 8/x_g = 100$). Demzufolge ist die Lokalität der Repräsentation niedrig ($d_m \neq 0$).

Aufgrund der Probleme mit der Hamming-Klippe wurde für GEAs oft die Gray-Kodierung verwendet. Die Gray-Kodierung hat gegenüber der binären Kodierung den Vorteil, dass die Allele gleichmäßig skaliert sind und deswegen keine Probleme mit Domino-Konvergenz und genetischer Drift auftreten. Eine Gray-kodierte Bitkette kann gemäß folgender Vorschrift aus einer binär kodierten Bitkette berechnet werden:

$$x_i^{gray} = \begin{cases} x_i^{binary} & \text{wenn } i = 0 \\ x_{i-1}^{binary} \oplus x_i^{binary} & \text{sonst.} \end{cases}$$

$\oplus$ bezeichnet hierbei die modulo 2 Operation und x_i das $i$te Bit einer Bitkette.

Beim Einsatz der unären Kodierung tritt das Problem auf, dass die einzelnen Phänotypen durch eine unterschiedliche Anzahl von verschiedenen Genotypen repräsentiert werden. Damit hängt die Leistungsfähigkeit von GEAs sehr stark von der Lage der optimalen Lösung im Lösungsraum ab. Wenn die optimale Lösung überrepräsentiert wird, wird die optimale Lösung zuverlässig in kurzer Zeit gefunden. Falls hingegen die optimale Lösung unterrepräsentiert ist, nimmt die Leistungsfähigkeit von GEAs stark ab. In diesem Fall ist die unäre Kodierung deutlich schlechter als die binäre oder Gray-Kodierung.

Die Untersuchung der Gray- und binären Kodierung zeigt, dass beide Repräsentationen die Schwierigkeit des zu lösenden Problems verändern. Eine Analyse der Struktur der BBs in den Genotypen zeigt, dass Gray-Kodierung die Struktur der BBs wesentlich stärker ändert, als die binäre Kodierung. Dies ist umso überraschender, als die Lokalität der binären Kodierung niedriger ist als die der Gray-Kodierung.

3.2 Repräsentationen für Bäume

Die entwickelten theoretischen Konzepte können auch für die Analyse von Repräsentationen für Bäume verwendet werden.

3.2.1 Prüfernummern

Prüfernummern [Prü18] sind eine Repräsentation für Bäume, welche einer Zeichenkette der Länge $n-2$ eineindeutig und redundanzfrei einen Baum mit n Knoten zuordnet. Jedes der $n-2$ Elemente der Zeichenkette entstammt einem Alphabet mit n Zeichen. Prüfernummern wurden für die Kodierung von Bäumen in den letzten Jahren oft verwendet [GlJRR01]. Eine Analyse der bisher in der Literatur präsentierten Ergebnisse zeigt allerdings, dass es einen großen Unterschied in der Beurteilung der Leistungsfähigkeit von Prüfernummern gibt. Ein großer Teil der Veröffentlichungen berichtet von hoher Effizienz von GEAs. Es mehren sich aber auch Stimmen, welche GEAs mit Prüfernummern eine katastrophale Leistungsfähigkeit bescheinigen.

Es zeigt sich, dass die vorgestellten theoretischen Konzepte von Repräsentationen die unterschiedlichen Beobachtungen in der Literatur erklären können. Eine Untersuchung der Lokalität von Prüfernummern zeigt, dass Prüfernummern nur eine hohe Lokalität besitzen, wenn sie Sterne repräsentieren. Sobald eine Prüfernummer keine Sternstruktur repräsentiert, ist die Lokalität der Repräsentation sehr niedrig und benachbarte Phänotypen sind genotypisch vollkommen verschieden. Dies führt dazu, dass die Schwierigkeit von einfachen Probleme, und Problemen mit beschränkter Komplexität zunimmt. GEAs können einfache Probleme nur dann lösen, wenn die optimale Lösung einem Stern sehr ähnlich ist. Sobald die optimale Lösung einem Stern nicht ähnlich ist, versagen GEAs.

Durch die Anwendung der vorgestellten theoretischen Konzepte für Repräsentationen ist es also möglich, die unterschiedlichen Ergebnisse in der Literatur zu erklären. Wenn sehr kleine Probleme ($n < 10$) betrachtet werden oder die beste Lösung zufällig sternähnlich ist, kann die Verwendung von Prüfernummern zu guten Ergebnissen führen. Im Allgemeinen jedoch erlauben Prüfernummern keine sinnvolle Suche und GEAs versagen.

3.2.2 Link und Node biased Repräsentation

Die Link und Node biased (LNB) Kodierung [Pal94] verwendet für die Repräsentation von Bäumen einen gewichteten Vektor der Länge $l = n(n-1)/2 + n$. Entsprechend den Werten im Vektor wird die Distanzmatrix des Optimierungsproblems verändert. Zusätzliche Parameter (Knotengewicht P_2 und Kantengewicht P_1) sind notwendig, um den Einfluss der Werte im Vektor auf die Distanzmatrix zu steuern. Der Phänotyp ist letztendlich der auf der Basis der modifizierten Distanzmatrix ermittelte minimal spannende Baum.

Aus der Konstruktion der LNB-Kodierung ergibt sich, dass die Repräsentation redundant ist. Demzufolge hängt das Verhalten von GEAs stark davon ab, ob ein Teil der Phänotypen über- oder unterrepräsentiert sind. Eine genauere Untersuchung der Redundanz zeigt, dass mit zunehmendem Knotengewicht die LNB-Kodierung sternförmige Bäume stark überre-

präsentiert. Für sehr große Knotengewichte kann die LNB-Kodierung nur Sterne darstellen. Deswegen ist ein Einsatz der LNB-Kodierung mit großem Knotengewicht nur dann sinnvoll, wenn bekannt ist, dass die optimale Lösung einem Stern sehr ähnlich ist.

Ein weiteres Problem tritt allerdings auf, wenn Kanten- und Knotengewicht sehr klein sind. In diesem Falle haben die Werte des LNB-Vektors keinen Einfluss auf die Struktur der kodierten Lösung. Deswegen werden mit abnehmendem Knoten- und Kantengewicht die kodierten Lösungen dem minimal spannenden Baum der ursprünglichen Distanzmatrix immer ähnlicher. Im Extremfall, wenn beide Gewichte sehr klein sind, kann nur der minimal spannende Baum dargestellt werden.

Die Anwendung der LNB-Kodierung auf reale baumförmige Kommunikationsprobleme zeigt, dass GEAs mit dieser Kodierung bei geeigneter Wahl der Knoten- und Kantengewichte oft gute bis sehr gute Ergebnisse liefern. Der Grund ist darin zu finden, dass optimale Lösungen für Kommunikationsnetzwerkprobleme dem minimal spannenden Baum oft sehr ähnlich sind. Im Allgemeinen hängt allerdings die Repräsentation der Individuen sehr stark von den Knoten- und Kantengewichten ab. Falls keinerlei Wissen über die Struktur der optimalen Lösung existiert, ist der Einsatz eines großen Kantengewichts zu empfehlen. Dies garantiert, dass die Leistungsfähigkeit von GEAs unabhängig von der Struktur der optimalen Lösung ist.

3.2.3 Charakteristischer Vektor

Netzwerkstrukturen können sehr einfach durch eine Kodierung mit Hilfe eines charakteristischen Vektors (CVs) dargestellt werden. Hierbei wird durch ein binäres Chromosom der Länge $n(n-1)/2$ festgelegt, welche der $n(n-1)/2$ Kanten im Graph enthalten ist. Eine 1 an der $i$ten Stelle gibt an, dass dieser Link im Phänotyp benutzt wird.

Die CV-Kodierung stellt ein weiteres Beispiel für eine redundante Repräsentation dar. Beim Einsatz der CV-Kodierung zur Repräsentation von Bäumen tritt nämlich das Problem auf, dass auch ungültige Lösungen, welche keinen Baum darstellen, auftreten können. Derartige Lösungen müssen durch das zufällige Hinzufügen sowie Löschen von Kanten repariert werden. Die CV-Kodierung ist also redundant, da ein Phänotyp durch einen gültigen und viele ungültige Genotypen repräsentiert wird. Da allerdings alle Individuen gleichmäßig repräsentiert werden, ist die Leistungsfähigkeit von GEAs unabhängig von der Struktur der optimalen Lösung.

Durch das Reparieren von ungültigen Lösungen tritt das Problem von verdeckter Mutation ("stealth mutation") auf. Bei der Reparatur von ungültigen Lösungen kommen durch das zufällige Löschen und Hinzufügen von Kanten schon aus der Population ausgeschiedene Kanten wieder zurück. Daher führt das Reparieren von ungültigen Lösungen zu ähnlichen Effekten wie die Einführung von zusätzlicher Mutation. Dies führt bei kleinen und einfachen Problemen zu einer effizienteren Suche. Bei größeren Problemen ($n > 10$) wird allerdings durch die verdeckte Mutation die gezielte Suche nach guten Lösungen mehr und mehr durch Zufallssuche ersetzt und die Anzahl der Generationen t_{conv} nimmt deswegen drastisch zu. Als Ergebnis können GEAs größere Probleme beim Einsatz der CV-Kodierung nicht in akzeptabler Zeit zufriedenstellend lösen.

4 Theoriegeleitete Entwicklung neuer Repräsentationen

Mit Hilfe der theoretischen Konzepte können neue Repräsentationen theoriegeleitet entwickelt werden.

4.1 Network Random Keys (NetKeys)

NetKeys repräsentieren Bäume mit n Knoten durch einen gewichteten Vektor der Länge $n(n-1)/2$. Ähnlich wie bei random keys [Bea92] wird zuerst aus dem gewichteten Vektor ein Permutationsvektor der gleichen Länge gewonnen. Hierbei werden die Positionen des NetKey-Vektors entsprechend ihrer Wertigkeit angeordnet. Schließlich wird aus dem Permutationsvektor ein Baum konstruiert. Falls bei der Konstruktion des Baums das Einfügen von Kanten zu ungültigen Lösungen führt, werden diese Kanten übersprungen.

Eine genauere Untersuchung der NetKey-Kodierung zeigt, dass alle Lösungen gleichmäßig repräsentiert werden, Standardoperatoren die Struktur der BBs nicht zerstören, alle Allele den gleichen Beitrag zur Konstruktion des Phänotypen leisten, die Lokalität der Repräsentation hoch ist sowie die Schwierigkeit des Problems nicht durch die Repräsentation verändert wird. Darüber hinaus sind für die NetKey-Repräsentation keine Reparaturmechanismen notwendig und GEAs können Kanten bezüglich ihrer Wichtigkeit unterscheiden.

Die Abhängigkeit der notwendigen Populationsgröße N_{min} von der Größe des Optimierungsproblems kann für das einfache one-max tree-Problem berechnet werden als:

$$N_{min} \approx -\frac{\sqrt{\pi}}{4} \ln(\alpha) n^{1.5}.$$

n bezeichnet hierbei die Anzahl der Knoten. Die Populationsgröße wächst subquadratisch mit der Anzahl der Knoten. Für die Anzahl Generationen t_{conv} ergibt sich $t_{conv} \approx const * n$. Die Anzahl der Generationen nimmt linear mit der Anzahl der Knoten n zu.

Ein empirischer Vergleich von unterschiedlichen Repräsentationen bestätigt die hohe Leistungsfähigkeit der NetKey-Kodierung. Unabhängig von der Lage der optimalen Lösung finden GEAs zuverlässig und in kurzer Zeit das Optimum. Falls keinerlei Wissen über die Struktur der optimalen Lösung besteht, beziehungsweise nicht a priori bekannt ist, ist der Einsatz von NetKeys zu empfehlen.

4.2 Direkte Repräsentation (NetDir)

Direkte Repräsentationen verwenden für die Repräsentation von Phänotypen keine expliziten Genotypen, sondern Phänotypen werden direkt repräsentiert. Die NetDir-Kodierung ist ein Beispiel für eine direkte Repräsentation für Bäume. Hierbei wird ein Baum auch genotypisch als Baum repräsentiert. Da bei der Verwendung von direkten Repräsentationen

keine Standardoperatoren verwendet werden können, müssen problemspezifische Operatoren entwickelt werden.

Bei der Entwicklung von direkten Repräsentationen ist das Finden von geeigneten Repräsentationen einfach. Der Genotyp ist gleich dem Phänotyp. Die eigentliche Schwierigkeit bei der Entwicklung von direkten Repräsentationen liegt in der Entwicklung von geeigneten Operatoren.

Passende Mutationsoperatoren zu finden, ist im Allgemeinen trivial, sobald auf dem Lösungsraum eine geeignete Metrik definiert ist. Die Aufgabe von Mutation ist es, ein benachbartes Individuum zu erzeugen. Die Entwicklung von effizienten Crossoveroperatoren hingegen ist wesentlich aufwendiger, da die Struktur von BBs nicht durch deren Einsatz zerstört werden darf. Die Entwicklung derartiger Operatoren und Verfahren ist selbst für Bitketten-Repräsentationen sehr aufwendig [PGCP99]. Beim Einsatz von direkten Repräsentationen kann hingegen (fast) nie auf Standardoperatoren oder -verfahren zurückgegriffen werden, sondern es müssen für jedes Problem speziell angepasste Operatoren und Verfahren entwickelt werden. Darüber hinaus kann man bei der Entwicklung von direkten Repräsentationen auch nicht auf die Hilfe geeigneter Elemente einer Theorie von Repräsentationen zurückgreifen, da keine expliziten Repräsentationen existieren.

Aus diesen Gründen heraus ist die Entwicklung von effizienten direkten Repräsentationen außerordentlich aufwendig, da die Schwierigkeit nicht in der eigentlichen Repräsentation, sondern in der Entwicklung von passenden Crossoveroperatoren liegt. Im Gegensatz zu indirekten Repräsentationen kann die Überprüfung der Leistungsfähigkeit einer Repräsentation nur durch empirische Studien und nicht theoriegeleitet durchgeführt werden. Damit befindet man sich beim Einsatz von direkten Repräsentationen im gleichen Dilemma wie zu Beginn der vorliegenden Studie. Es existieren für direkte Repräsentationen keine theoretischen Erklärungsmodelle und die Analyse von Repräsentationen kann demzufolge auch nicht mit Hilfe fundierter Theorie durchgeführt werden. Die Suche nach effizienten, direkten Repräsentationen gleicht daher mehr einem blinden Stochern im Nebel als einer zielgerichteten Suche.

5 Zusammenfassung

Der vorliegende Beitrag beschreibt, wie durch die Entwicklung theoretischer Konzepte für Repräsentationen eine systematische Analyse und Entwurf von Repräsentationen möglich wird. Es wird aufgezeigt, wie redundante Repräsentationen, Repräsentationen mit unterschiedlich gewichteten Allelen und Repräsentationen mit niedriger Lokalität die Leistungsfähigkeit von GEAs beeinflussen. Basierend auf diesen Konzepten kann der Einfluß von Repräsentationen auf die Leistungsfähigkeit von GEAs für ganzzahlige und baumförmige Optimierungsprobleme bestimmt werden, sowie theoriegeleitet neue Repräsentationen für Bäume entwickelt werden.

Die Arbeit ermöglicht ein besseres Verständnis von Repräsentationen und versetzt Anwender in die Lage GEAs wesentlich effektiver und schneller für das Lösen von Problemen zu verwenden. Die bisher sehr zeitraubende Suche nach guten Repräsentationen kann ersetzt werden durch zielgerichtetes und theoriegetriebenes Entwerfen.

Literaturverzeichnis

[Bea92] J. C. Bean. Genetics and random keys for sequencing and optimization. Technical Report 92-43, Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, June 1992.

[GlJRR01] Jens Gottlieb, Bryant A. Julstrom, Günther R. Raidl, and Franz Rothlauf. Prüfer Numbers: A Poor Representation of Spanning Trees for Evolutionary Search. IlliGAL Report No. 2001001, University of Illinois at Urbana-Champaign, Urbana, 2001.

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

[HCPGM97] G. R. Harik, E. Cantú-Paz, D. E. Goldberg, and B. L. Miller. The gambler's ruin problem, genetic algorithms, and the sizing of populations. In T. Bäck, editor, *Proceedings of the Forth International Conference on Evolutionary Computation*, pages 7–12, New York, 1997. IEEE Press.

[Kim64] M. Kimura. Diffusion models in population genetics. *J. Appl. Prob.*, 1:177–232, 1964.

[LV90] G. E. Liepins and M. D. Vose. Representational issues in genetic optimization. *Journal of Experimental and Theoretical Artificial Intelligence*, 2:101–115, 1990.

[MG96] B. L. Miller and D. E. Goldberg. Optimal sampling for genetic algorithms. IlliGAL Report No. 96005, University of Illinois at Urbana-Champaign, Urbana, IL, 1996.

[Pal94] C. C. Palmer. *An approach to a problem in network design using genetic algorithms*. unpublished PhD thesis, Polytechnic University, Troy, NY, 1994.

[PGCP99] M. Pelikan, D. E. Goldberg, and E. Cantú-Paz. BOA: The Bayesian optimization algorithm. IlliGAL Report No. 99003, University of Illinois at Urbana-Champaign, Urbana, IL, 1999.

[Prü18] H. Prüfer. Neuer Beweis eines Satzes über Permutationen. *Archiv für Mathematik und Physik*, 27:742–744, 1918.

[SCED89] J. D. Schaffer, R. A. Caruana, L. J. Eshelman, and R. Das. A study of control parameters affecting online performance of genetic algorithms for function optimization. In J. D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms*, pages 51–60, San Mateo, CA, 1989. Morgan Kaufmann.

[TG93] D. Thierens and D. E. Goldberg. Mixing in genetic algorithms. In S. Forrest, editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 38–45, San Mateo, CA, 1993. Morgan Kaufmann.

Franz Rothlauf promovierte sich im November 2001 an der Universität Bayreuth mit der in diesem Beitrag vorgestellten Arbeit. Seine Arbeit wurde von Prof. Dr. Armin Heinzl (Universität Bayreuth) und Prof. Dr. David E. Goldberg (University of Illinois at Urbana-Champaign) betreut. Regelmäßige Forschungsaufenthalte am Illinois Genetic Algorithms Laboratory (IlliGAL) ermöglichten Ihm ein gründliches Studium von Genetischen und Evolutionären Verfahren.

Seine gegenwärtigen Forschungsgebiete liegen im Bereich Repräsentationen, benutzerfreundliche Problemlöser und Black-Box Optimierung. Darüber hinaus beschäftigt er sich mit Problemen von Multiagentensystemen und deren Kombination mit Genetischen Algorithmen.