Technische Universität München
Institut für Informatik
Prof. Dr. Angelika Steger

SS 1999
Merkblatt 7
15. Juni 1999

Praktikum Diskrete Optimierung

Genetische Algorithmen

1 Allgemeine Beschreibung

Eine allgemeine Einführung in genetische Algorithmen erhalten Sie in der Praktikumsbesprechung. Dabei handelt es sich um einen Auszug aus

> Colin R. Reeves, *Modern Heuristic Techniques*, in 'Modern Heuristic Search Methods', V. J Rayward-Smith, I. H. Osman, C. R. Reeves and G. D. Smith (editors), pp. 17-23, John Wiley & Sons, 1996.

2 Anwendung auf Netzwerkplanung

In diesem Abschnitt gehen wir darauf ein, wie genetische Algorithmen auf unser Netzwerkplanungsproblem angewandt werden können.

2.1 Schema des Algorithmus

```
pop := Startpopulation;
while not finished do
    // Calculate fitness of individuals
    eval(pop);
    // Select pairs according to fitness values
    mates := select(pop);
    // Construct new population
    newpop := ∅;
    for all pair in mates do
        Perform crossover with probability p;
        Perform mutation with probability q;
        newpop := newpop ∪ pair;
    od
    print statistics;
od
```

2.2 Wahl der Repräsentation

Das wahrscheinlich schwierigste (und für die Performance wichtigste) Teilproblem bei der Umsetzung eines genetischen Algorithmus liegt in der Codierung der Elemente des Suchraums. Hierbei ist es üblich, diese sogenannten *Chromosomen* als Strings von Bits, Zeichen oder Zahlen[1] zu realisieren. Es gibt allerdings auch die Möglichkeit, kompliziertere Datenstrukturen, wie Bäume oder mehrdimensionale Arrays zu betrachten.

Bei der Wahl der Repräsentation sollte man berücksichtigen, wie ein Crossover durchgeführt werden soll. Genetische Algorithmen basieren auf der Annahme, daß es in den Chromosomen „gute“ Teilstrukturen gibt, die im Laufe des Algorithmus zu einer insgesamt guten Lösung zusammengesetzt werden. Man sollte deshalb überprüfen, ob die Kombination Repräsentation/Crossover in der Lage ist, dies zu erreichen.

Für das Netzwerkoptimierungsproblem könnte man ein Chromosom beispielsweise als Vektor aller Pfade realisieren, d. h. jede Komponente gibt an, welcher Pfad für das zugehörige Knotenpaar verwendet wird. Auf diesem Vektor könnte dann ein klassisches Crossover durchgeführt werden. Diese Variante bringt allerdings das Problem mit sich, daß die Menge aller möglichen Routen in der Population bei jeder Generation erhalten bleibt. Man sollte das Crossover also dahingehend modifizieren, daß auch Teilpfade wieder zu einem neuen Pfad kombiniert werden können. Beispielsweise könnte man an ein zweistufiges Crossover denken, bei dem einerseits die Vektoren der Pfade „gekreuzt“ werden, aber auch andererseits zwischen den Pfaden ein Crossover erfolgt. Dies wird realisiert, indem die Pfade selbst wieder als Vektoren interpretiert werden.

2.3 Fitness Scaling

Bei der *Roulette Wheel Selection* tritt folgendes Problem auf: Wenn der Algorithmus bereits eine Weile gelaufen ist und die Population eine gewisse Konvergenz erreicht hat, dann unterscheiden sich die Fitness-Werte innerhalb der Population nur noch wenig. In diesem Fall nimmt der Selektionsdruck ab, da auch relativ schlechte Individuen noch eine gute Chance haben, zum Crossover ausgewählt zu werden. Dieses Phänomen kann man durch Skalierung der Fitnesswerte umgehen.

Hierzu verwendet man am einfachsten eine lineare Abbildung: Die Fitness des i-ten Individuums in Generation t sein $f(i)$. Wir berechnen die skalierte Fitness $f'(i)$ durch

$$f'(i) := a \cdot f(i) + b$$

[1] Es hängt von der Anwendung ab, welche Granularität hier sinnvoll ist. Wenn im Problem beispielsweise Gleitkommazahlen vorkommen, dürfte es nicht sinnvoll sein, diese als Bitstrings zu betrachten. Diese wäre erstens ineffizient und würde zweitens beim Crossover zu ungewünschten Effekten führen.

für geeignete Parameter a und b, die wie folgt gewählt werden: Die mittlere Fitness $\bar{f}$ der Population soll durch die Skalierung erhalten bleiben, d. h. $\bar{f} = \bar{f}' = a\bar{f} + b$. Dadurch wird garantiert, daß im Erwartungswert ein Individuum mit durchschnittlicher Fitness selektiert wird.

Ferner wünschen wir, daß ein Individuum mit maximaler Fitness $f_{\max}$ auf $f'_{\max} = C \cdot \bar{f} = a \cdot f_{\max} + b$ abgebildet wird. Durch diese zwei Bedingungen sind die Parameter a und b eindeutig festgelegt. Die Konstante C kann so interpretiert werden, daß im Erwartungswert C Individuen mit maximaler Fitness bei Selektion gewählt werden. Dabei haben sich Werte $1.2 \leq C \leq 2$ bewährt.

Bei diesem Verfahren gibt es allerdings noch ein Problem: Es ist nicht ausgeschlossen, daß kleine Fitnesswerte auf negative Zahlen abgebildet werden, was natürlich für die Roulette Wheel Selection nicht brauchbar ist. In diesem Fall könnte man a und b so wählen, daß weiterhin $\bar{f} = \bar{f}'$ gilt und ferner $af_{\min} + b = 0$. Das heißt, das schlechteste Individuum erhält skalierte Fitness Null und ist somit von der Selektion ausgeschlossen. Auch hier sind die Parameter durch die zwei Gleichungen eindeutig bestimmt.

Ein anderes Verfahren ist das sogenannte *σ-Scaling*. Hierbei wird die Abbildung durch die Standardabweichung σ der Population definiert. Es gilt

$$f'(i) := \begin{cases} 1 + \frac{f(i) - \bar{f}}{2\sigma} & \text{für } \sigma \neq 0 \\ 1.0 & \text{für } \sigma = 0 \end{cases}$$

Negative Fitnesswerte müssen hierbei wieder gesondert behandelt werden.

2.4 Performance-Analyse

Zum Einstellen der Parameter ist es nützlich, kleine Statistiken über den Verlauf der Fitnesswerte während der Generationen anzufertigen. Hierbei sind beispielsweise Minimum, Maximum, Mittelwert und Standardabweichung der Fitness von Interesse. Wünschenswert ist ein Verhalten, bei dem eine mäßig schnelle Konvergenz erfolgt.

Wenn die Population zu schnell homogen wird, könnte dies am Selektionsverfahren oder an einer zu kleinen Mutationswahrscheinlichkeit liegen. Wenn sich die Fitnesswerte andererseits völlig zufällig entwickeln, sollte man vielleicht weniger Mutationen durchführen oder die Crossover-Operation ist ungünstig definiert und zerstört die Bausteine optimaler Lösungen anstatt sie zu kombinieren.