1 Motivation

Die im folgenden vorgestellten Verfahren dienen der Lösung eines Optimierungsproblems:

Definition 1 *Optimierungsproblem*
Gegeben einen **Suchraum** $S \in \Re^n, n \in \aleph$ *und eine* **Fitnessfunktion** $f : S \rightarrow R$*, berechne den* **Zustand** $s \in S$ *für den* f *maximal wird.*

Ist kein Algorithmus bekannt, der eine Lösung effizient konstruiert, können klassische Optimierungverfahren in Betracht gezogen werden. Diese machen jedoch Annahmen über die zu optimierende Funktion, und können so für viele Optimierungsprobleme nicht verwendet werden:

- Extremwertberechung über Ableitungen : f ist 2-fach stetig differenzierbar.
- Gradientenmethode : f ist differenzierbar
- Simplex : f ist linear und hat lineare Nebenbedingungen
- Ennumeration : Der Suchraum ist endlich ("klein genung").

In diesem Fall bleiben nur noch probabilistische Verfahren. Die wichtigsten sind:

- Selektionsverfahren : Threshold Accepting, Sinflut Methode, Simulated Annealing
- Genetische Algorithmen und Evolutionsstrategien
- Genetische Programmierung

2 Genetische Algorithmen

2.1 Algorithmus

Der Begriff "Genetischer Algorithmus" ist nicht ganz eindeutig. Es existieren verschiedene Varianten. Die Unterschiede sind allerdings klein, deswegen wird im Folgenden nur die von J.H. Holland 1975 eingeführte Variante genauer betrachtet.

Der Suchraum ist gegeben durch $S = 0, 1^n, n \in \aleph$ (es können aber auch andere Alphabete benutzt werden). Die Elemente des Suchraums werden als *Induvidien* bezeichnet, und haben eine feste Länge. Induvidien können durch *genetische Operationen* verändert werden. In der Grundform sind dies

Mutation Wähle zufällig (d.h. i.d.R gleichverteilt) ein Induviduum aus der Population aus und flippe ein Bit an einer ebenfalls zufällig gewähltes Position.

Rekombination Zwei zufällig gewählte Induviduen x, y werden durch eine Funktion $\delta : S \times S \rightarrow S \times S$ *gekreuzt*. Bei der Auswahl werden jedoch Induviduen mit einer höheren Fitness bei der Auswahl bevorzugt. Eine einfache Variante von δ ist die Vertauschung von Teilen des "Erbgutes".

Reproduktion Ein Induviduum wird zufällig für das Überleben in der nächsten Generation ausgeählt. Fitte Induviduen werden bevorzugt.

Dies führt auf den Algorithmus:

```
(1)   Erzeuge eine Anfangspopulation P mit N Induviduen
      und definiere P' als eine leere Menge.
(2)   Berechne fuer alle Induviduen von P die Fitness.
(3)   Fuehre eine der folgenden Operationen aus:
         Rekombination, mit Wahrscheinlichkeit p(C)
         Mutation, mit Wahrscheinlichkeit p(M)
         Reproduktion, mit Wahrscheinlichkeit p(R)
      wobei p(c) + p(R) + p(M) = 1
(4)   Fuege die neuen bzw. ausgewaehlten Individuen zur
      neuen Population P' hinzu
(5)   Ist |P'| < |P| fahre bei (3) fort.
(6)   Pruefe das Abbruchkriterium. Falls es nicht
      erfuellt ist, setze P = P' und fahre bei (2) fort.
(7)   Gib das Individuum mit der hoechsten Fitness aus.
```

Im folgenden werden noch einige Standartvarianten der genetischen Operatioen vorgestellt.

2.2 Auswahlverfahren

Genetische Algorithmen verändern Induviduen durch genetische Operationen. Welche Induviduen zu veränderen sind soll dabei über ein an der Fitness orientiertes Auswahlverfahren ermittelt werden, dass fitte Induviduen bevorzugt. Mögliche Verfahren sind z.B.

Roulette Sei $f(j)$ die Fitness des j-ten Induviduums der Popolation. Dann ist die *totale Fitness* durch $F(k) = \sum_{j=1}^{k} f(j), k \in |P|$ definiert. Ziehe gleichverteilt eine Zahl zwischen 1 und $F(|P|)$ und wähle das Induviduum p mit $F(p-1) \leq z < F(p)$.

Zufällige Paare Ziehe gleichverteilt zwei Induviduen und wähle das Induviduum mit der höheren Fitness.

(N,μ)-Selektion Sortieren die Induviduen nach ihren Fitnesswerten. Wähle aus den μ fittesten Induviduen gleichverteilt ein Induviduum aus.

2.3 Rekombination

Zunächst werden über ein Auswahlverfahren zwei Induviduen x, y ausgewählt. Dann werden die Rekombinationsfunktion δ zwei Kopien einer Kombination von x und y erzeugt. Nachfolgend werden einige Varianten beschrieben. Dabei bezeichnet $x[j]$ den Wert des Genoms von x and der Stelle j.

Ein-Punkt Die Genome der Eltern x, y werden bis zu einem zufällig ermittelten Bit/Gen vertauscht.

Zwei-Punkt Die Genome der Eltern x, y werden zwischen zwei zufällig ermittelten Bits/Genen vertauscht.

Uniform Crossover Es wird ein zufälliger Bitvektor der Länge der Eltergenome erzeugt. Dann werden die Bits/Gene der Eltern an den Positionen vertauscht, an denen der Bitvektor eine 1-Eintrag besitzt.

Mittelwert Nur für reellwertige Genomkodierungen verwendbar. Das Genom der Nachkommen z wird durch $z[j] = (x[j] + y[j])/2 \ \forall j$ definiert.

Partially matched Crossover Es existieren Anwendungen, bei denen die Gene eine Induviduums nicht mehrfach vorkommen dürfen. Zunächst wird eine Zwei-Punkt Mutation zwischen zwei Punkten i, j durchgeführt. Nun werden bei beiden Induviduen die Gene, die vertauscht wurden wieder zurückgetauscht. Allerdings nur, *wenn sie ausserhalb des Intervalls $[i..j]$ liegen, und insgesamt mehrfach vorkommen*. Diese Vertauschungen können mehrfach notwendig sein.

2.4 Mutation

In der Regel ist die Mutationsrate klein anzusetzen, da Mutationen sinvolle Informationen zerstören können. Andererseits kann man auf Mutationen nicht gänzlich verzichten, denn sie helfen lokale Extrema zu verlassen.

2.5 Links

Im Internet finden sich Webseiten und Tutorials zum Thema zuhauf. Als Einführung ist auch das Buch *Optimierung mit genetischen und selektiven Algorithmen*, W.Kinnebrock, Oldenburg Verlag hilfreich.