Semesterarbeit

Evaluation von verschiedenen Ansätzen für Evolutionäre Algorithmen

Adrian Schmid

Zürich, 01.01.2014

Inhaltsverzeichnis

1 Einleitung

2 Projekt

2.1 Aufgabenstellung

2.2 Ausgangslage

2.3 Projektplanung

3 Grundlagen

3.1 Endliche Automaten und reguläre Sprachen

Reguläre Sprachen und *endliche Automaten* gehören zur Automatentheorie, einem Teilbereich der theoretischen Informatik. Zum besseren Verständnis dieser Arbeit habe ich folgend einige gängige und wichtige Konzepte der Automatentheorie frei nach dem Standartwerk von Hopcroft, Motwani und Ullman zusammengefasst. [2]

Alphabet Ein Alphabet ist eine endliche, nicht leere Menge von Symbolen. Üblicherweise wird ein Alphabet durch das Symbol Σ dargestellt.

Zeichenreihen Eine Zeichenreihe (auch Wort oder String) ist eine endliche Folge von Symbolen eines bestimmten Alphabetes.

Sprachen Eine Menge von Zeichenreihen aus Σ^*, wobei Σ ein bestimmtes Alphabet darstellt, wird als Sprache bezeichnet. Das heisst, eine Sprache ist eine Menge von Zeichenreihen die mit den Zeichen aus einem Alphabet Σ gebildet werden können. Bei der Bildung von Wörtern müssen nicht alle Zeichen des gegebenen Alphabets verwendet werden.

Bei der Klasse der *regulären Sprachen* handelt es sich um die Menger aller Sprachen, welche durch einen endlichen Automaten beschrieben werden können. Alle Sprachen dieser Klasse haben auch die Eigenschaft, dass Sie durch einen regulären Ausdruck dargestellt werden können. Das heisst, dass es für jede reguläre Sprache sowohl einen regulären Ausdruck als auch einen endlichen Automaten gibt, der alle Wörter der Sprache akzeptiert.

Ein deterministischer endlicher Automat besteht gemäss Hopcroft, Motwani und Ullman [2] aus:

1. einer endlichen menge von Zuständen die meist durch Q bezeichnet wird.
2. einer endlichen Menge von Eingabesymbolen (auch Alphabet Σ).
3. einer Übergangsfunktion δ der ein Zustand und ein Eingabesymbol übergeben werden und die einen Zustand zurückgibt.
4. einem Startzustand welcher Teil der Menge Q sein muss.
5. einer Menge F akzeptierender Zustände. Die Menge F ist eine Teilmenge von Q.

Oft werden endliche Automaten als Quintupel in der Form $A = (Q, \Sigma, \delta, q0, F)$ dargestellt.

Zum besseren Verständnis folgend ein Beispiel eines Automaten welcher alle Binärzahlen die durch drei Teilbar sind akzeptiert.

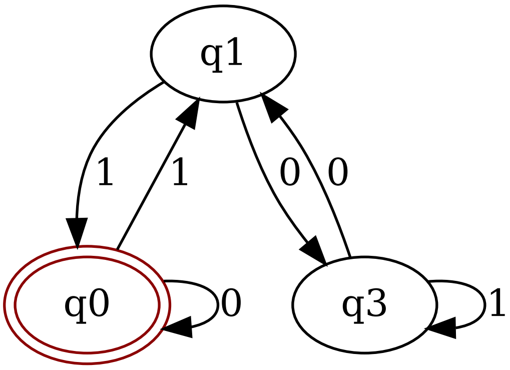

Abbildung 1: Beispiel: Endlicher Automat

Die Attribute des Quintupels für diesen Automaten sind:

1. $Q = \{q0, q1, q3\}$
2. $\Sigma = \{0, 1\}$
3. δ
4. $q0$
5. $F = \{q0\}$

Wobei man die Übergangsfunktion δ am einfachsten als Tabelle wie folgt darstellt:

	0	1
→q0*	q0	q1
q1	q3	q0
q3	q1	q3

Tabelle 1: Übergangstabelle

Konventionen Ich habe mich bei der Darstellung von Automaten im Rahmen dieser Arbeit auf folgende Konvention festgelegt:

- Zustände werden mit einem kleinen q gefolgt von einer Zahl bezeichnet (zB. $q1$)
- Zustände müssen nicht durchgängig nummeriert sein
- Zustände werden als Ellipsen dargestellt
- Zustandsübergänge sind beschriftete Pfeile
- der Startzustand erhält einen roten Rand
- alle akzeptierenden Zustände sind doppelt umrandet

3.2 Evolutionäre Algorithmen

Unter evolutionären Algorithmen (EA) verstehen wir randomisierte Heuristiken, die Suchprobleme näherungsweise durch vereinfachende algorithmische Umsetzung von Prinzipien der natürlichen Evolution zu lösen versuchen. Somit geben evolutionäre Algorithmen in der Regel weder eine Garantie bzgl. der benätigten Rechenzeit noch der Güte der ausgegebenen Lösung. Ein Suchproblem besteht darin, zu einer Zielfunktion ein Element aus deren Definitionsbereich zu finden, dessen Funktionswert möglichst gut ist. Darunter verstehen wir im Folgenden, wenn nicht ausdrücklich anders vermerkt, einen möglichst grossen Funktionswert, weshalb der Zielfunktions wert eines Elements auch als seine Fitness bezeichnet wird.

Der Aufbau eines evolutionären Algorithmus lässt sich dann grob wie folgt beschreiben: in jedem Schritt verwaltet er eine Menge fester Grösse von Suchpunkten, die so genannte Population, wobei jeder einzelne Suchpunkt auch als Individuum bezeichnet wird. Aus den Punkten der Population neue Punkte zu erzeugen, ist Aufgabe von Mutation und Rekombination. Dabei steht hinter der Mutation die Idee, jeweils nur ein einzelnes Individuum zufällig zu verändern, ohne dass andere Individuen dabei berücksichtigt werden. Durch Rekombination wird hingegen aus mehreren, meist zwei Individuen zufällig ein neues gebildet, das von diesen möglichst gute Eigenschaften übernehmen soll. Durch Mutation und Rekombination werden also neue Individuen (Kinder genannt) aus bestehenden Individuen (Eltern genannt) erzeugt. Beide Operatoren hängen oftmals stark von Zufallsentscheidungen ab. Jedoch fliesst in der Regel weder in Mutation noch Rekombination der Zielfunktionswert der Individuen ein.

Die Zielfunktion beeinflusst nur die Selektion. Dieser Operator wählt Individuen der Population aus, sei es zur Auswahl der Eltern für eine Rekombination oder Mutation oder, um aus der Menge von Eltern und Kindern die nächste Population zu wählen, was den Übergang zur nächsten Generation darstellt. Dadurch, dass die Selektion Punkte mit höherem Zielfunktionswert mit grösserer Wahrscheinlichkeit auswählt, soll erreicht werden, dass nach und nach immer bessere Punkte gefunden werden. [1]

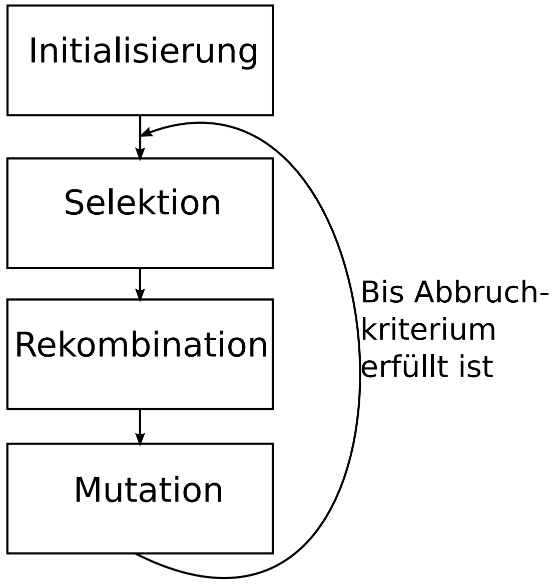

Abbildung 2: Evolutionärer Algorithmus

3.2.1 Anwendung auf unser Problem

TBD

4 Umsetzung

4.1 Automaten

4.2 Evolutionäre Algorithmen

Abbildungsverzeichnis

Tabellenverzeichnis

Literatur

[1] Stefan Droste. *Zu Analyse und Entwurf evolutionarer Algorithmen.* Universität Dortmund, Fachbereich Informatik, 2000.

[2] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie.* Pearson Education, 2002.