

### Dritte Projektaufgabe: Operatoren eines EAs

Zeit: 35 Stunden

Es soll ein evolutionärer Algorithmus geschrieben werden, bei dem ein Individuum eine Strategie zur Lenkung eines fiktiven Staats darstellt.

Dabei erhält ein Individuum den „Zustand“ des Staats definiert über acht verschiedene Maßzahlen und muss entscheiden, in welcher Form es seinen Handlungsspielraum nutzen möchte. Hierzu kann eine gewisse Anzahl an Aktionspunkten auf fünf beeinflussbare Maßzahlen angewandt werden:

- Produktion – sowohl vergrößerbar als auch verkleinerbar
- Sanierung – nur vergrößerbar
- Aufklärung (Bildung) – nur vergrößerbar
- Lebensqualität – nur vergrößerbar
- Vermehrungsrate – direkt nur vergrößerbar

Anmerkung: In anderen Versionen kann die Vermehrungsrate nicht direkt beeinflusst werden – aber da es durch die Erlaubnis des direkten Einflusses nicht schwieriger wird, nehmen wir diese vorliegende Version des ursprünglichen Brettspiels.

Der evolutionäre Algorithmus soll nun ein Individuum entwickeln, das einen Staat über einen Zeitraum von 30 Jahren „regeln“ kann. Dabei kann naturgemäß aus unterschiedlich prekären Anfangssituationen heraus gestartet werden.

Die typische Standard-Ausgangssituation und die Punktbereiche, aus denen ich voraussichtlich die Startsituationen für den Test Ihrer Strategie benutzen werde, sind in der folgenden Tabelle dargestellt.

Aspekt	Standard	Bereich	Aspekt	Standard	Bereich
Aktionspunkte	8	5 ... 12	Lebensqualität	10	5 ... 20
Sanierung	1	1 ... 15	Vermehrungsrate	20	8 ... 22
Produktion	12	5 ... 25	Bevölkerung	21	10 ... 35
Umweltbelastung	13	5 ... 25	Politik	0	-5 ... 20
Aufklärung	4	1 ... 20			

Anmerkung: Ich bin mir nicht sicher, ob eine Ausgangssituation bestehend aus den angegebenen Extremwerten wirklich regulierbar ist.

In OPAL kann der Quelltext des Simulators, ein Beispielindividuum sowie ein kleines Beispielttestprogramm in der Datei `EvolAlgPr3.tgz` (gezipptes Tar-Archiv) heruntergeladen werden. Dort ist auch die JavaDoc-Dokumentation enthalten.

Bis zum **25. Januar 2013, 24:00 Uhr** sind in OPAL folgende Dateien abzugeben:

- Ihre Individuumsklasse als Java-Quelltext,
- Ihr serialisierter Zustand als `.ser`-Datei,
- ein Archiv mit Ihrem gesamten Quelltext der dritten Projektaufgabe und
- ein Bericht als PDF-Datei, dessen Aufbau im folgenden Abschnitt nochmals erläutert wird.

Der Bericht soll **genau** wie folgt aufgebaut sein:

1. Konzept: Eine kurze Darstellung des Gesamtkonzepts Ihres evolutionären Algorithmus
2. Repräsentation: Wie wird die Strategie in Ihrem Individuum dargestellt (als Genotyp)? Begründen Sie Ihre Entwurfsentscheidung.
3. Evolutionäre Operatoren: Wie arbeiten Mutation bzw. Rekombination falls vorhanden? Begründen Sie Ihre Entwurfsentscheidung.
4. Selektionsdruck: Wie wird Selektionsdruck erzeugt? Begründen Sie Ihre Entwurfsentscheidung.
5. Gütebewertung: Welche Kriterien der Simulationen werden wie zur Bewertung einer Strategie benutzt? Begründen Sie Ihre Entwurfsentscheidung.
6. Simulationsszenarien: Welche Ausgangssituation haben Sie bei der Evaluation des besten Individuums berücksichtigt?
7. Effizienz: Welche Laufzeit, Generationenzahl etc. haben Sie für die Erzeugung des besten Individuums in der einen relevanten evolutionären Optimierung benötigt?
8. Fazit: Wie würden Sie nachträglich die Qualität der gefundenen Strategie und des entwickelten evolutionären Algorithmus bewerten und einordnen?

Die folgenden Aspekte gehen dabei in die Bewertung der Abgaben mit gleicher Gewichtung ein:

- vergleichende Beurteilung der Lösungsstrategie auf mehreren Anfangsszenarien
- Beurteilung des im Bericht beschriebenen Gesamtkonzepts für die Optimierung

Bitte beachten Sie die folgenden Hinweise bei der Bearbeitung der Aufgabe:

- Bezeichnen Sie Ihr Individuum mit Ihrem Login-Namen (erster Buchstabe groß geschrieben) und legen Sie dieses im Paket `loesung` ab.
  - Bitte beachten Sie für die Serialisierung, dass lediglich die Individuumsklasse, Standardklassen und in Ihrer Klasse eingebettete Klassen zur Beschreibung des Zustands eines Individuums benötigt werden. Insbesondere können auch Verweise auf den Simulator oder auf Klassen Ihrer evolutionären Operatoren etc. nicht wieder deserialisiert werden.
  - Erzeugen Sie beim Serialisieren Ihrer „besten Strategie“ eine durch Ihren Login-Namen bezeichnete Datei `meinlogin.ser`.
  - Beim Aufruf der Methode `wendeDieStrategieAn()` werden nur noch Regeln o.ä. des Individuums ausgelesen – hier dürfen keine weiteren komplexen KI-Algorithmen, vorausschauende Simulationen der Spielwelt etc. durchgeführt werden.
  - Die Strategie des Individuum soll durch einen evolutionären Algorithmus erzeugt bzw. eingestellt werden.
-