

Praktikum zu Genetische Algorithmen

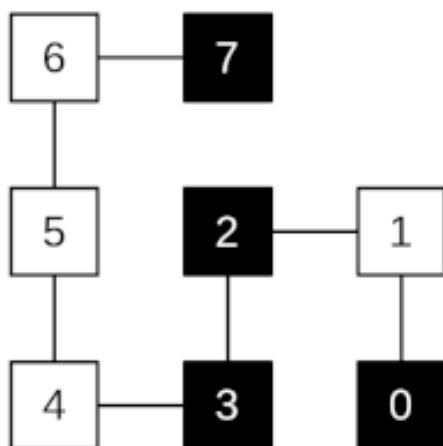
Ihre Aufgabe ist es, einen genetischen Algorithmus zu implementieren, welcher gute Faltungen einer Aminosäuren-Sequenz im 2d HP-Modell findet. Hierfür sollen Sie die Programmiersprache Java verwenden. Die Endabnahme ist am letzten Termin und entscheidet darüber, ob Sie das Praktikum bestanden haben. Das bestandene Praktikum ist Voraussetzung um an dem Genetische Algorithmen Leistungsnachweis teilnehmen zu können. Neben der reinen Implementation des Genetischen Algorithmus gibt es zwei Aufgabenblätter, welche Sie bearbeiten müssen.

Termin 1

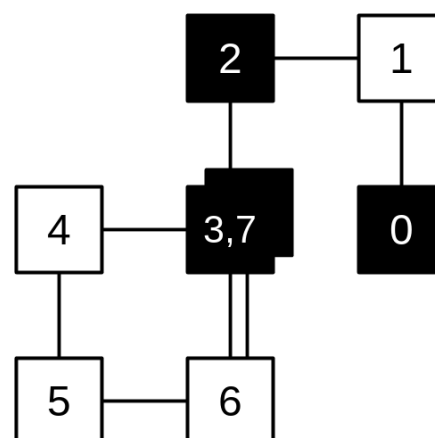
Sie haben eine Datenstruktur definiert, damit man Lösungskandidaten erzeugen kann. Weiterhin haben Sie eine Methode („Fitness-Funktion“) implementiert, die einen Lösungskandidaten als Eingabe nimmt und dessen Fitness berechnet. Diese Fitness ist eine Gleitkommazahl. Testen Sie die Richtigkeit Ihrer Implementation, indem Sie als Testdaten geeignete Faltungen von Hand definieren und prüfen, ob Ihre Fitnessfunktion daraus die richtigen Ergebnisse berechnet.

Abnahme zu Termin 1, fällig an Termin 1 oder Termin 2:

Kodieren sie die unten stehende Faltung 1 und lassen Sie deren Fitness berechnen. Geben Sie hierzu die Fitness, als auch die Anzahl der Hydrophob/Hydrophob-Kontakte sowie die Anzahl der Überlappungen aus. Machen Sie das Gleiche für Faltung 2. Dort ist die Aminosäure 3 an dem gleichen Platz auf dem Gitter wie die Aminosäure 7. Welche Werte berechnet Ihre Fitnessfunktion nun?



Faltung 1



Faltung 2

Termin 2

Sie haben mit der Implementation Ihres GAs begonnen und können den zu einem Lösungskandidaten zugehörigen Phänotyp visualisieren. Die Datei *GraphikBeispiel.java* zeigt Ihnen, wie Sie in Java einfache 2d Graphiken erstellen können. Dabei muss für jede Aminosäure klar erkennbar sein, ob sie hydrophil oder hydrophob ist und welchen Index sie in der zugrunde liegenden Sequenz besitzt. Tip: Verwenden Sie hierzu die Klasse *BufferedImage* und speichern Sie das erzeugte Diagramm bei Bedarf als eine PNG-Datei ab. Ebenso sollte die Fitness, die Anzahl der hydrophob/hydrophob-Kontakte sowie der Überlappungen angezeigt werden. Weiterhin erzeugt Ihre Software eine Log-Datei. Diese Datei ist eine einfache CSV-Datei oder TAB separated und in jeder Generation wird eine Zeile angehängt. Jede Zeile enthält die folgenden sechs Spalten: Generationsnummer, durchschnittliche Fitness in dieser Generation, Fitness des besten Lösungskandidaten in dieser Generation, Fitness des besten bisher gefundenen Lösungskandidaten, Anzahl der hydrophob/hydrophob-Kontakte des besten bisher gefundenen Lösungskandidaten, Anzahl der Überlappungen im besten bisher gefundenen Lösungskandidaten. Verwenden Sie zum Testen Ihres Codes die Benchmark-Sequenzen aus der Datei *Examples.java*.

Abnahme zu Termin 2, fällig an Termin 2 oder Termin 3:

Starten Sie den Genetischen Algorithmus mit einer Population von 100 Kandidaten und lassen Sie ihn 100 Generationen laufen. Verwenden Sie Fitness-proportionale Selektion, aber keine Mutation und auch kein Crossover. Visualisieren Sie die Ergebnisse aus der Log-Datei in einer Tabellenkalkulation und erklären Sie den Verlauf der Kurven.

Termin 3

Sie haben nun die fehlenden Elemente Ihres GA implementiert und dabei die Fitness-Funktion aus dem ersten Meilenstein und die Visualisierung aus dem zweiten Meilenstein eingebaut. Sie können nun den GA starten. Er verwendet Fitness-proportionale Selektion, 1-Punkt-Crossover und Punktmutation. Sie können die beste Faltung visualisieren, die Ergebnisse aus der Log-Datei in eine Tabellenkalkulation importieren und interpretieren. Testen Sie Ihren Algorithmus mit den Benchmark-Sequenzen und bewerten sie die dabei erzeugten Verläufe aus den jeweiligen Log-Dateien.

Abnahme zu Termin 3, fällig zu Beginn von Termin 3:

Sie haben die Aufgaben im Aufgabenblatt zum Termin 3 gelöst.

Termin 4

Sie haben die Turnier-Selektion als wahlweise Alternative zur Fitness-proportionalen Selektion implementiert, ebenso eine generationsabhängige Steuerung der Mutationsrate. Die jeweilige Mutationsrate haben Sie als weiteren Datenspalte in die CSV-Datei geschrieben.

Sie haben eine Parametrisierung Ihres Algorithmus gefunden.

Abnahme zu Termin 4, fällig zu Beginn von Termin 4:

Sie haben die Aufgaben im Aufgabenblatt zum Termin 4 gelöst.

Termin 5

Sie haben das GA-Praktikums-Testat erhalten. Hierzu haben Sie gezeigt, dass Ihr Algorithmus mit den Benchmark-Sequenzen gute Ergebnisse erzielt. Weiterhin konnten Sie Fragen im Kontext Ihres Algorithmus angemessen beantworten.