Günter Rudolph
Nicola Beume

Dortmund, 11.12.2006

Übungen zur Vorlesung

Fundamente der Computational Intelligence

Wintersemester 2006/07

Blatt 7

Aufgabe 7.1: Programmieraufgabe: Experimentelle Analyse (20 Punkte)
Entscheiden Sie durch eine experimentelle Studie, ob ein $(\mu+\lambda)$EA oder ein $(1,\lambda)$EA (Aufgabe 6.2) für die Rastrigin-Funktion besser geeignet ist, bei einer Anzahl von 200.000 Funktionsauswertungen.
Beschreiben Sie Ihr Vorgehen bei den einzelnen Aufgabenteilen außerhalb des Programmcodes. Dokumentieren Sie Ihren Programmcode und schicken Sie diesen per E-Mail mit einer Beschreibung zur Kompilierung und zur Ausführung. Schicken Sie außerdem die Daten Ihrer Experimente, deren Auswertung sowie zugehörige Abbildungen.

a) Implementieren Sie einen $(\mu+\lambda)$EA mit einem diskreten Rekombinationsoperator für eine reellwertige Repräsentation.

b) Implementieren Sie als Minimierungsproblem die Rastrigin-Funktion
$f(\mathbf{x}) = 10n + \sum_{i=1}^{n} x_i^2 \cos(2\pi x_i)$ mit $x_i \in [-5;5] \subset \mathbb{R}$, $i \in \{1,\ldots,n\}$ und $n = 20$.

c) Für das Optimierszenario sollen folgende Einstellungen für beide Algorithmen fest sein: Die Schrittweite der Mutation soll konstant 0,05 betragen, die Initialisierung soll zufällig gleichverteilt erfolgen. Variieren Sie die Parameter μ und λ des $(\mu+\lambda)$EA, indem Sie ein Gitter von Parameterkombinatonen erzeugen. Für beide Paramter werden in gleichmäßigem Abstand Werte gewählt und alle möglichen Kombination ausprobiert, sodass man eine Matrix von Parameterkombinationen erhält.
Variieren Sie außerdem das λ des $(1,\lambda)$EA.

d) Führen Sie eine Validierung Ihrer Experimentergebnisse durch. Starten Sie mit einigen vielversprechenden Parameterkombinationen 20 Läufe mit verschiedenen Zufallssaaten. Berechnen Sie über diese Läufe jeweils minimalen, maximalen und durchschnittlichen Funktionswert, sowie die Standardabweichungen. Vergleichen Sie diese Ergebnisse mit Ihrer Studie in Aufgabe c) und führen Sie eventuell mehr Wiederholungen Ihrer Studie in c) durch, falls die Abweichungen der Ergebnisse groß sind.

e) Geben Sie besonders gute Parameterkombinationen an. Welcher EA ist für das Problem besser geeignet? Beschreiben Sie Ihre Beobachtungen und Erkenntnisse jedes Arbeitsschrittes textuell und veranschaulichen Sie diese durch graphische Darstellung relevanter Läufe. Geben Sie statistische Maße (Minimum, Mittelwert, Median, Standardabweichung) der relevanten Daten an, um Ihre Aussagen zu unterstützen.

Aufgabe 7.2: Programmieraufgabe: Monte-Carlo-Simulation (10 Punkte)
Bestimmen Sie π experimentell mittels Monte-Carlo-Simulation. Monte-Carlo-Simulation bezeichnet dir Durchführung von Zufallsexperimenten, um punktweise Informationen einer Zufallsverteilung zu gewinnen.

Die Kreiszahl π ist definiert als die Fläche eines Einheitskreises (Radius=1). Erzeugen Sie zufällig gleichverteilt Punkte in einem Einheitsrechteck (Kantenlänge=1) und zählen Sie, wie viele Punkte innerhalb eines eingeschriebenen Viertel-Einheitskreises liegen. Die Anzahl der Punkte innerhalb des Viertel-Kreises geteilt durch die Gesamtzahl entspricht einer Approximation von $\pi/4$.

Es gibt bei vielen Problemen einen Trade-Off zwischen Genauigkeit der Approximation und Ressourceneinsatz. Man stellt sich also die Frage, ob ein höherer Einsatz von Ressourcen (hier: mehr erzeugte Punkte) noch lohnenswert ist. Finden Sie den Zusammenhang (linear, exponentiell,...) von Approximationsgüte und Anzahl der erzeugten Punkte heraus.

Aufgabe 7.3: Fragen?

Notieren Sie konkrete Fragen oder Bemerkungen über den Vorlesungsstoff, den Sie nicht verstanden haben oder über weitergehende Aspekte der CI.

Bearbeitungen bis 08.01.2007, 12.00 Uhr an Nicola Beume (nicola.beume@udo.edu; OH14, Raum 233). http://ls11-www.cs.uni-dortmund.de/people/beume/CI-ws0607/index.jsp