Die Hausaufgaben müssen einzeln bearbeitet und abgegeben werden. Geben Sie die schriftlich zu bearbeitenden Aufgaben als pdf-Dateien ab.

Abgabefrist ist der 7.2.2024 – 12:15 Uhr

Aufbau Blatt

Bitte beachtet das abweichende Abgabedatum!

Außerdem ist die zweite Aufgabe eine freiwillige Bonusaufgabe für diejenigen, die gern auch die multikriterielle Optimierung einmal praktisch einsetzen wollen.

Abgabe

Wir benutzen für die Abgabe der Hausaufgaben Git. Jedes Repository ist nur für den Studierenden selbst sowie für die Betreuer und Korrektoren sichtbar.

Für die Hausaufgabe benötigen Sie **kein** neues Repository. Es wird das Repository benutzt, das für Übungsblatt 2 angelegt wurde.

Damit das Repository im Laufe der Vorlesung übersichtlich bleibt und wir sinnvoll korrigieren können, achtet bitte auf das Folgende:

- Der finale Commit für ein Übungsblatt muss mit einem Tag oder einer klaren Commit-Message versehen sein, der/die diesen Commit eindeutig als die finale Abgabe erkennen lässt.
- Erstellt **pro Übungsblatt** einen Ordner, in dem die Files liegen, die kein Code sind (Textfiles, Plots, ...).

Nicht oder zu spät gepushte (Teil-)Abgaben werden mit 0 Punkten bewertet!

Aufgabe 1 – Pareto-Optimalität und Survivor Selection in NSGA-II und SPEA2 (20P)

Gegeben sei das Paar reellwertiger Funktionen $f_i \colon \mathbb{R} \to \mathbb{R}$ per

$$\begin{cases} f_1(x) = x^2 \\ f_2(x) = (x-2)^2, \end{cases}$$

die beide *zu minimieren* sind (dieses Minimierungsproblem taucht als *Schaffer function No. 1* in Benchmarks zur multikriteriellen Optimierung auf).

1. Gegeben sei die Menge $P = \{-3, -1, 0, 0.5, 1, 1.75, 2, 4\} \subseteq \mathbb{R}$. Skizzieren oder plotten Sie das Bild von P im objective space bezüglich (f_1, f_2). In welchem Verhältnis stehen die Elemente aus P in Bezug auf die Pareto-Dominanz zueinander?

2. Skizzieren oder plotten Sie die Pareto-Front bezüglich (f_1, f_2) im $\mathbb{R}^2$. Welche Menge $M \subseteq \mathbb{R}$ ist die Pareto-Menge bezüglich (f_1, f_2), d. h., welche Elemente werden auf die Pareto-Front abgebildet?
3. Angenommen die Menge P aus 1. ist die Vereinigung einer Population mit berechneten Nachkommen im Laufe eines evolutionären Algorithmus, der (f_1, f_2) minimieren soll (mit $\mu = 4$). Erklären Sie sowohl für NSGA-II als auch SPEA2 wie die Survivor Selection auf dieser Menge abläuft und welche Elemente die nächste Generation bilden.

Zusatzaufgabe – Implementierung und Verwendung NSGA-II (30P)

In dieser Aufgabe geht es darum, den NSGA-II Algorithmus zu implementieren und zu verwenden, um das mehrdimensionale Rucksackproblem zu lösen.

Implementierung (20P)

Implementieren Sie, basierend auf Ihren bereits vorhandenen Implementierungen evolutionärer Algorithmen, den NSGA-II.

1. Implementieren Sie die schnelle Sortierung einer Menge bezüglich des non-domination Rangs (siehe Folie 300).
2. Implementieren Sie die Berechnung der crowding distance einer Menge von Elementen (siehe Folie 302).
3. Implementieren Sie einen Fitnessvergleich zweier Elemente, der die Möglichkeit von verletzten Nebenbedingungen berücksichtigt (siehe Folie 304).
4. Implementieren Sie davon ausgehend den NSGA-II, also deterministische $(\mu + \mu)$-Selektion für die Survivor Selection (mit Unterstützung für Nebenbedingungen) und Parent Selection über Turniere der Größe 2. Beachten Sie, dass Sie dafür für jedes Element die Crowding Distance innerhalb seiner non-domination front berechnen müssen (siehe insbesondere Folie 303).

Lösen des mehrdimensionalen Rucksackproblems (10P)

Lösen Sie mit Ihrer Implementierung des NSGA-II das mehrdimensionale Rucksackproblem. Dabei geht es darum, eine Menge von Gegenständen so auszuwählen, dass gleichzeitig der Wert mehrerer Rucksäcke maximiert wird, ohne die Kapazität eines der Rucksäcke zu überschreiten. Formal ist das Problem wie folgt definiert: Gegeben sind m Gegenstände, n Rucksäcke und die folgenden Größen für $1 \leq j \leq m$ und $1 \leq i \leq n$:

$$\begin{aligned} p_{i,j} &= \text{Profit des Gegenstands } j \text{ in Bezug auf Rucksack } i; \\ w_{i,j} &= \text{Gewicht des Gegenstands } j \text{ in Bezug auf Rucksack } i; \\ c_j &= \text{Kapazität des Rucksacks } i. \end{aligned}$$

Ziel ist es, einen Vektor $\mathbf{x} = (x_1, \ldots, x_m) \in \{0,1\}^m$ zu finden, sodass i) keine Kapazität überschritten wird, also für alle $1 \leq i \leq n$ gilt

$$\sum_{j=1}^{m} w_{i,j} \cdot x_j \leq c_i$$

und ii) $\mathbf{x}$ Pareto-optimal bezüglich der Fitnessfunktionen

$$f_i(\mathbf{x}) = \sum_{j=1}^{m} p_{i,j} \cdot x_j$$

ist $(1 \leq i \leq n)$, die *zu maximieren* sind.

1. Generieren Sie sich in den folgenden Schritten eine konkrete Instanz des mehrdimensionalen Rucksackproblems, das Sie dann im folgenden lösen:
   a) Wir setzen $m = 500$ und $n = 2$ (also 500 Gegenstände und 2 Rucksäcke).
   b) Generieren Sie (die insgesamt 2 000) Werte für $w_{i,j}$ und $p_{i,j}$ indem Sie jeden der Werte zufällig gleichverteilt aus $[10, 100] \subset \mathbb{N}$ sampeln.
   c) Setzen Sie die Kapazität c_i jeweils auf $c_i = \lfloor \frac{1}{2} \sum_{j=1}^{m} w_{i,j} \rfloor$.
2. Wenden Sie Ihren NSGA-II auf das generierte Problem an. Als Terminationskriterium können Sie eine feste Anzahl an Iterationen verwenden. Um sich einen Überblick über den Verlauf der Optimierung zu verschaffen, können Sie jeweils das Bild der Population im objective space plotten.