

Organic Computing 2

Lösungsvorschlag Blatt05

Lukas Huhn Qiang Chang Victor Gerling

18. Juni 2019

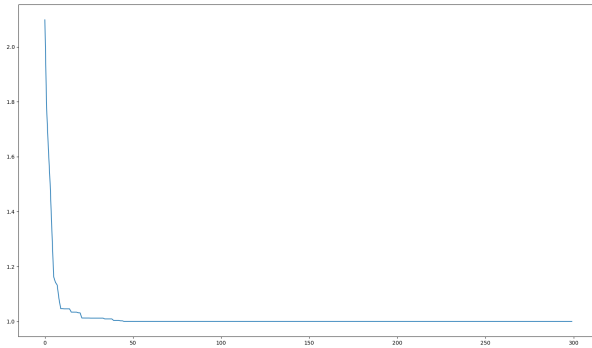
Universität Augsburg

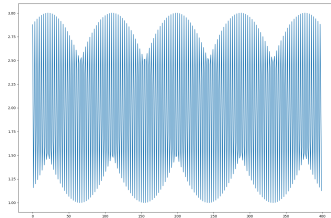
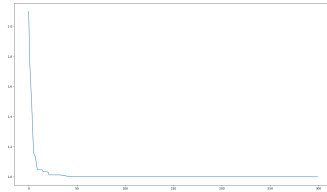
Institut für Informatik

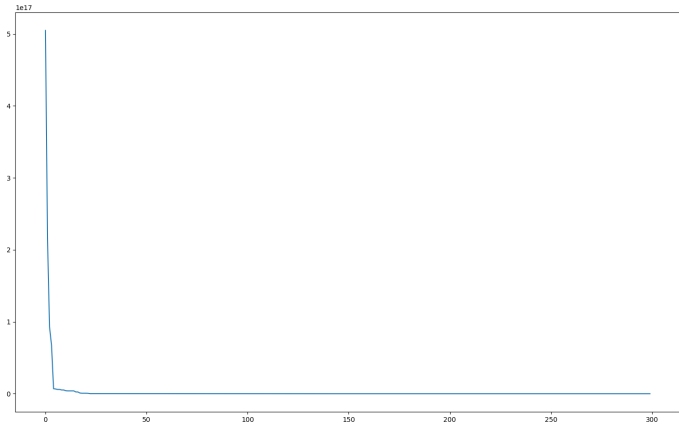
Lehrstuhl für Organic Computing

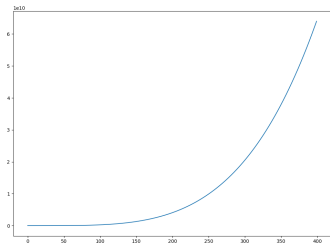
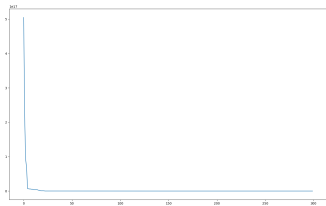
1. Aufgabe 02

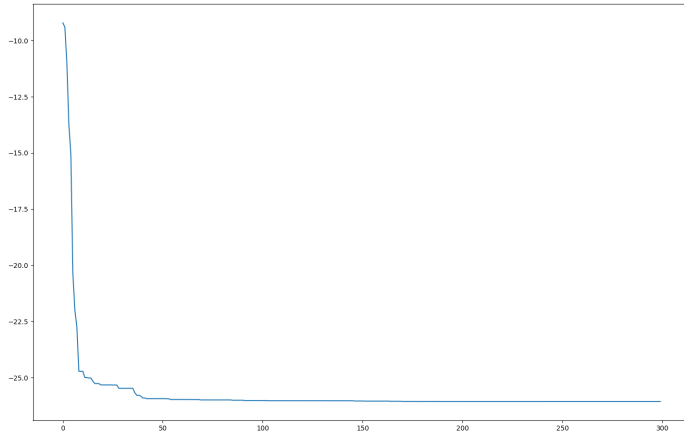
Aufgabe 02

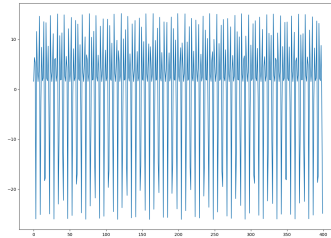
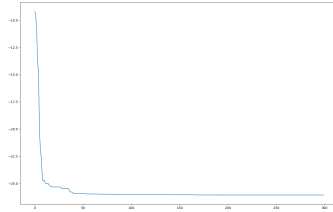


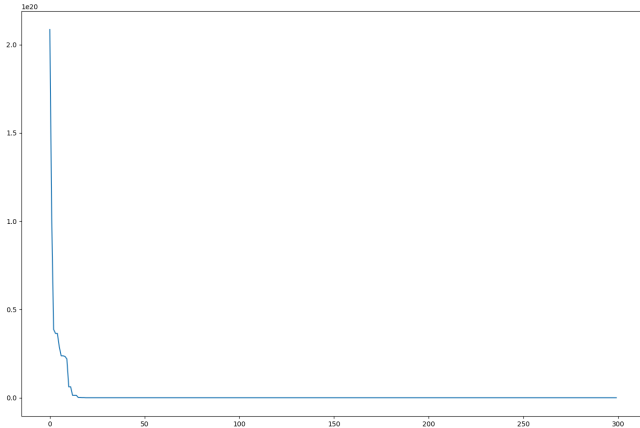


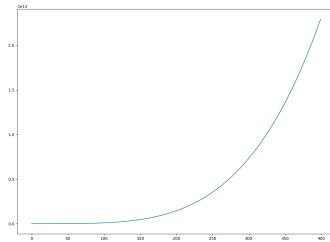
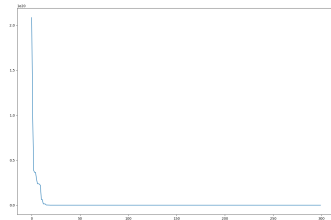


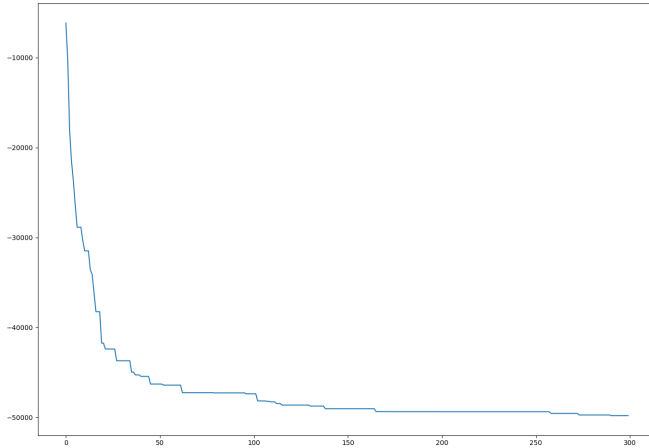


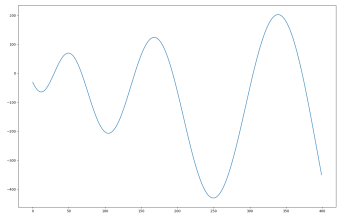
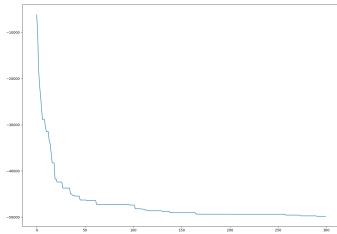












Lässt sich eine Aussage treffen, bei welcher Blackbox dieses Verfahren am besten funktioniert? Können Sie sich denken, weshalb?

- BB1: Optimum wird recht schnell gefunden
- BB2: Optimum wird nahezu sofort gefunden
- BB3: findet sein Optimum bei Iteration 40
- BB4: findet sein Optimum bei Iteration 15
- BB5: wird sehr lange brauchen um sein Optimum zu finden
- \Rightarrow BB2 funktioniert am besten.

Ist dieses Verfahren effizient? Wie könnte man das Verfahren leicht(!) abwandeln, um ein (noch) effizienteres Verfahren zu erhalten?

- Verfahren ist nicht effizient, vorherige Erkenntnisse werden nie mit zur Rate gezogen
- Sobald man merkt, dass die BB auf größer/kleiner werdende Werte anspringt Bereich der Zufallszahl anpassen. Man wählt dann also zwischen [current-best, MAXVALUE] oder [MINVALUE, current-best].