

Povinná programovací úloha

Vaším úkolem bude implementace 2 stochastických heuristických algoritmů a vytvoření jednoduchých statistických (a grafických) křížových srovnání.

Algoritmy:

- Random Search
- Stochastic Hill Climber/Simulované žíhání (pro získání bonusových bodů ke zkoušce)

Testovací funkce (Dimenze $D = 5$ $D=10$ a $D=20$):

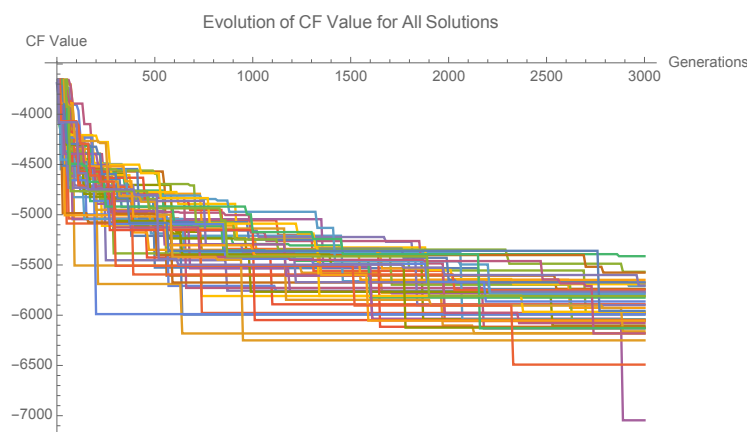
- 1st DeJong function
- 2nd DeJong function
- Schwefel function

Pro porovnání algoritmů musíte zachovat stejnou hodnotu FES (Cost Function Evaluations). FES lze nastavit např. na 10 000, tj. pokud implementujete hill climber s 10 lokálními řešeními (v rámci okolí), v každé JEDNÉ iteraci horolezeckého algoritmu provedete 10 FES, takže můžete mít celkem max. 1000 iterací. Náhodné vyhledávání - logicky existuje pouze jedno FES za iteraci.

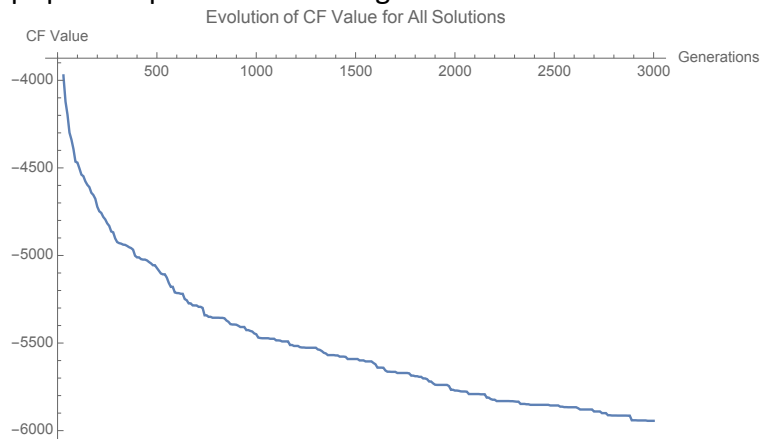
Oba dva algoritmy se musí spustit opakovaně 30x pro každou zkušební funkci a nastavení dimenze - pro získání nějakého statistického základu - vypočítáte (z 30 nejlepších výsledků) Min, Max, Mean, Median a Std. Dev. Hodnoty a porovnejte tyto hodnoty mezi 3 algoritmy pro každou zkušební funkci. Musíte také potvrdit vaše výsledky vykreslením nejlepších řešení z každé iterace - tj. Konvergenční graf. Vaším úkolem je vykreslit:

- Konvergenční graf všech 30 běhů v jednom grafu (30 čar v 1 grafu) - celkem 18 grafů (2 algoritmy x 3 funkce x 3 nastavení dimenze)
- Konvergenční graf průměrného nejlepšího výsledků - tj. Průměrné nejlepší řešení v každé iteraci (jeden graf – z těch 30 čar výše uděláte v každé iteraci průměr) - celkem 18 grafů (3 algoritmy x 3 funkce x 3 nastavení dimenze)
- Porovnání těchto průměrných konvergencí pro 2 algoritmy v jednom grafu pro každou testovací funkci - celkem 9 grafy (2 algoritmy v jednom grafu x 3 funkce x 3 nastavení dimenze)

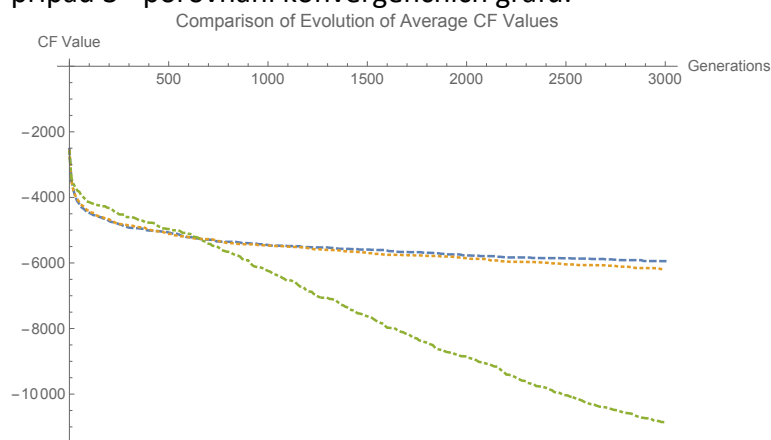
Příklady: případ 1: 30 konvergencí v jednom grafu



případ 2 – průměrná konvergence



případ 3 - porovnání konvergenčních grafů:



Berte v úvahu následující:

Nelze opustit vyhledávací prostor - při vytváření sousedních řešení - zkontrolujte hranice typické pro každou testovací funkci (-500 až 500 pro Schwefel, -5 až 5 pro obě funkce DeJong). Pokud opustíte vyhledávací prostor - vygenerujete nový bod okolí.

Nastavení – Max FES 10000, Dimension: 5, 10, 20, Neighborhood size 10 solutions, max 10% prohledávacího prostoru, pro Simulované žíhání max temp ??, min temp ??, cooling decr 0.98 – zde proveďte parameter tuning a vlastní výzkum vhodných max/min parametrů teploty – zda 1000/10, nebo 10000/100, 10000/1000 atd...