|  |  |  |
| --- | --- | --- |
| Příjmení: **Holcman** | Jméno: **Jiří** | Obor: **NKYB** |
| Předmět: **AK9EV** | | Datum: **27.12.2021** |
| Název protokolu: **Povinný zápočtový úkol** | | |

# Zadání

Vaším úkolem bude implementace libovolného evolučního algoritmu a vytvoření jednoduchých statistických a grafických výstupů.

Algoritmy (vyberte libovolný jeden z totoho seznamu):

* DE/jDE
* SOMA (AllToOne strategie)
* **PSO**

Testovací funkce (Dimenze D=10 a D=30):

* 1st DeJong function
* 2nd DeJong function
* Schweffel function
* Rastrigin

Funkce - definice v níže přiloženém souboru

FES bude nastaven na 5 000 x D. Algoritmus se musí spustit opakovaně 30x pro každou zkušební funkci - pro získání nějakého statistického základu - vypočítáte (z 30 nejlepších výsledků) Min, Max, Mean, Median a Std. Dev.

Musíte také potvrdit vaše výsledky vykreslením nejlepších řešení z každé iterace - tj. Konvergenční graf. Vaším úkolem bude vykreslit:

 Konvergenční graf všech 30 běhů v jednom grafu (30 čar v 1 grafu) - celkem 8 grafů (1 algoritmus x 4 funkce x 2 nastavení D)

Konvergenční graf průměrného nejlepšího výsledků - tj. průměrné nejlepší řešení v každé iteraci (z 30 běhů) - celkem 8 grafů ( 1 algoritmus x 4 funkce x 2 nastavení D).

Berte v úvahu následující a nastavení parametrů:

Nelze opustit vyhledávací prostor - při vytváření nových řešení - zkontrolujte hranice typické pro každou testovací funkci. Pokud opustíte vyhledávací prostor - aplikujte libovolnou zvolenou funkci pro kontrolu hranic (random, periodic, reflection...).

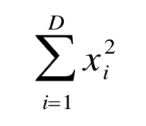
Nastavení – pro DE/jDE (NP = 50, F = 0.5, CR = 0.9 (pro jDE je to pouze init nastavení, zbytek dle článku J. Brest et. al), pro SOMA (NP = 50, prt = 0.3, PathLength = 3, step = 0.33), pro PSO (NP = 50, c1 a c2 = 2.0, inertia weight: wstart = 0.9, wend = 0.4).

# Vypracování

Tato programovací úloha sestávala z mnou vybraného algoritmu **PSO** a něj byla použita každá ze čtyř testovacích funkcí, každá pro dimenzi 10 a 30. Pro ověření výsledků a otestování běhu programu jsou přiloženy jak zdrojové kódy, tak i výsledné PDF s grafy výsledků. Celý program je realizován v **MATLABu R2017b** a spouští se skriptem **start.m**.

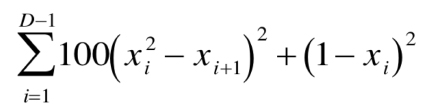
V programu MATLAB byl naprogramován algoritmus **PSO** (PSO.m). Jako základní parametry byly určeny v zadání rozsahy testovacích funkcí, ve kterých se musíme pohybovat a počet dimenzí funkce. Celkový počet dotazů na testovací funkci byl omezen na 5000 x dimenze. Cílem bylo hledat minimum funkce. Celý tento životní cyklus byl opakován 30x aby bylo možno vytvořit nějaký statistický výstup.

**1st DeJong function**

****

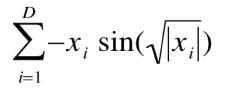
Rozsah: -5 až 5

**2nd DeJong function**

****

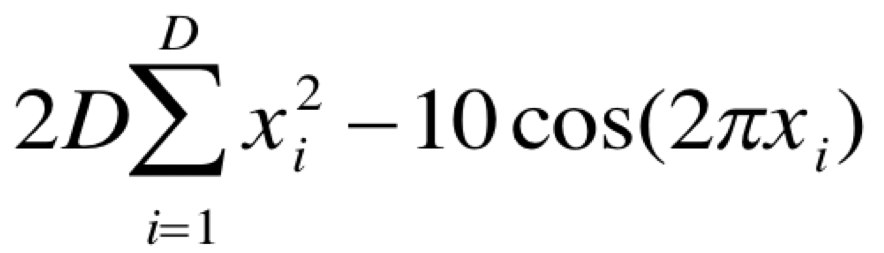
Rozsah: -2,2 až 2,2

**Schweffel function**

****

Rozsah: 0 až 520

**Rastrigin function**



Rozsah: -2,2 až 2,2

**PSO algoritmus**

Tento algoritmus je jedním z nejpopulárnějších metaheuristických optimalizačních algoritmů, založeném na „inteligentním“ chování hejna. Počítá se zde s rychlostním vektorem, který je ovlivněn celkovým a lokálním nejlepším výsledkem. Nejedná se tedy o typické evoluční algoritmy, které obsahují křížení a mutaci.

Celý algoritmus je založen na určité velikosti populace, kde pro každého jedince je vypočítán rychlostní vektor pro další pohyb v závislosti na poloze lokálního a globálního minima dle vzorce:



Celý cyklus se opakoval celkem: 5000 x dimenze/populace, v našem případě 1000x pro dimenzi 10 a 3000x pro dimenzi 30. Velikost populace byla 50 jedinců. Maximální velikost kroku byla omezena na 0,001\*šíře prostoru. Pokud jsme se ocitly mimo zadaný rozsah, bya velikost opuštění prostoru odečtena (nebo přičtena) od hranice. Vše se opakovalo celkem 30x a byl vytvořen výstup v podobě statistických hodnot (Min, Max, Mean, Median a Std. Dev) a grafů řešení.

Jako první byl vytvořeny konvergenční grafy 30 běhů (konkrétních a průměrných hodnot) pro testovací funkci:

**DeJong 1**, dimenze **10** a **30** (statistické hodnoty jsou uvedeny v grafu hned pod řádkem názvu grafu):







**DeJong 2**, dimenze **10** a **30** (statistické hodnoty jsou uvedeny v grafu hned pod řádkem názvu grafu):









**Schweffel**, dimenze **10** a **30** (statistické hodnoty jsou uvedeny v grafu hned pod řádkem názvu grafu):









**Rastrigin**, dimenze **10** a **30** (statistické hodnoty jsou uvedeny v grafu hned pod řádkem názvu grafu):









# Závěr

Cílem projektu bylo aplikovat zvolený metaheuristický algoritmus a na něm demonstrovat možnosti, jaké nám poskytuje při hledání řešení. Procvičit si praktické naprogramování a aplikaci algoritmu a seznámit se s výsledky ve formě grafů a statistických výstupů.