



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

BEES ALGORITHM

TECHNICKÁ ZPRÁVA

SOFT COMPUTING

AUTOR PRÁCE

JIŘÍ VÁCLAVIČ

BRNO 2023

Kapitola 1

Princip algoritmu

Algorithm 1 Pseudokód algoritmu BA

```
1: Inicializace:
2: Náhodně vygenerujte počáteční populaci  $sbi = sbi_1, sbi_2, \dots, sbi_n$ 
3: Ohodnotte kvalitu každého jedince:  $f(sbi_i)$  pro  $i = 1 \dots ns$ 
4: Uložte nejlépe ohodnoceného jedince do  $sbbest$ .
5: while není splněno ukončovací kritérium do
6:   Selekce:
7:   Vyberte  $m$  nejlépe ohodnocených jedinců a označte je jako  $selectedSbi$ 
8:   Vyberte z  $selectedSbi$  ještě  $e$  elitních jednotek.
9:   for každý elitní jedinec  $elitniSbi$  do
10:     Rekrutujte  $fb_j$  pro  $j = 1 \dots nfe$  a proveďte lokální prohledávání.
11:     Umístěte  $fb_j$  náhodně do okolí  $r$  příslušného elitního jedince.
12:   end for
13:   for každý zbývajících jedinec v  $selectedSbi$  do
14:     Rekrutujte  $fb_j$  pro  $j = 1 \dots nfo$  a proveďte lokální prohledávání.
15:     Umístěte  $fb_j$  náhodně do okolí  $r$  příslušného jedince.
16:   end for
17:   Výběr do další generace:
18:   Z každé skupinky ( $sbi + k$  ní přiřazené  $fb_j$ ) vyberte do další generace skautů  $sb$ 
    nejlépe hodnoceného jedince.
19:   Aplikace GA:
20:   Na zbývajících  $(ns - m)$  nevybraných jedincích aplikujte klasický GA algoritmus
    (globální prohledávání).
21:   Aktualizace nejlepší jedince:
22:   if některý jedinec má lepší ohodnocení než  $sbbest$  then
23:     Aktualizujte  $sbbest$  na tohoto jedince.
24:   end if
25: end while
26: return Nejlepší hodnotu  $sbbest$  jako výsledek algoritmu.
```

Algoritmus Bees Algorithm (BA) je optimalizační metoda inspirovaná chováním včel při hledání potravy. Jeho cílem je nalézt optimální řešení úlohy, která je formulována jako minimalizace nebo maximalizace cílové funkce. V průběhu algoritmu probíhá postupné generování a ohodnocení populace včel. Nejlépe ohodnocená včela je udržována v paměti jako aktuální nejlepší pozice. Elitní včely procházejí lokálním prohledáváním s pomocí dalších včel, a stejný postup se aplikuje i na zbývající členy populace. Nová generace včel je vytvořena výběrem nejlépe ohodnocených jedinců z každé skupiny. Na zbytku populace je aplikován klasický genetický algoritmus. Nejlepší pozice je aktualizována, pokud některá včela dosáhne lepšího ohodnocení. Algoritmus opakuje své kroky, dokud nejsou splněna ukončovací kritéria, a výsledné řešení je dáno pozicí nejlepší včely. Pseudokód 1 výše ukazuje princip fungování algoritmu.

Kapitola 2

Způsob implementace

Zde jsou popisy funkcí a tříd:

Třída `BeeAlgorithmGUI`:

- Metoda `__init__(self, root)`: Inicializuje GUI pro vizualizaci algoritmu s různými ovládacími prvky, jako jsou posuvníky, tlačítka a kanvas pro vykreslování.
- Metoda `create_slider_box(self, label, variable, from_value, to_value, command, resolution=1)`: Vytváří skupinu ovládacích prvků pro nastavování hodnot pomocí posuvníku.
- Metody `update_space_range`, `update_max_iterations`, `update_scout_bees_ratio`, `update_colony_size`: Aktualizují hodnoty příslušných parametrů při pohybu posuvníky.
- Metoda `start_algorithm(self)`: Spouští algoritmus s vizualizací na základě aktuálních hodnot parametrů.

Funkce `fitness_function(x, func=0)`:

- Implementuje jednoduchou fitness funkci, kterou bude algoritmus maximalizovat (nebo minimalizovat).

Funkce `initialize_colony(colony_size, dimension, space_range=(-200, 200))`:

- Inicializuje populaci včel s náhodnými pozicemi v daném prostoru.

Funkce `visualize_colony(canvas, colony, color, radius=10)`:

- Vykresluje aktuální stav populace včel na kanvas.

Funkce `visualize_progress(canvas, iteration, best_solution, best_fitness)`:

- Vykresluje informace o průběhu algoritmu na canvas.

Funkce `bee_algorithm_with_visualization(...)`:

- Implementuje algoritmus včelí kolonie s vizualizací.
- Postupně provádí skauty (globální hledání) a ostatní včely (lokální hledání směrem k informacím poskytnutým skautům).
- Pravidelně aktualizuje vizualizaci a informace o nejlepším řešení.

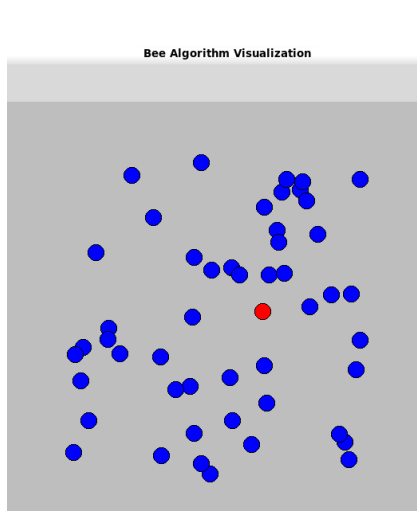
Funkce `probabilistic_selection(scout_bees, total_fitness, func)`:

- Realizuje pravděpodobnostní výběr včel v závislosti na jejich fitness hodnotách.

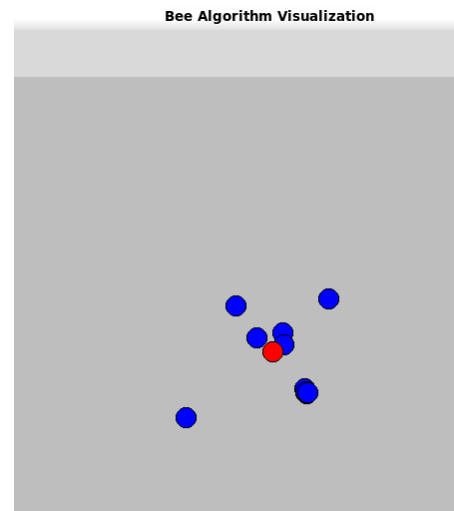
Funkce `main()`:

- Vytváří instanci třídy `BeeAlgorithmGUI` a spouští hlavní smyčku Tkinter.

Kód vytváří uživatelské rozhraní pro vizualizaci algoritmu BA s možností nastavování různých parametrů. Po stisknutí tlačítka "Start Bee Algorithm" spouští algoritmus a vizualizuje jeho průběh. Na obrázku 2.1 lze vidět počáteční stav včel a jejich ohodnocení pro problém minimalizace kvadratické funkce. Na obrázku 2.2 je již stav vizualizace po provedení algoritmu BA, kde bylo nalezeno optimální řešení (červená barva) a ostatní včely se k němu velice zcela přiblížily. Populace je na obou obrázcích stejně veliká, ale obrázku 2.2 sdílí některé včely stejné ohodnocení, takže se vzájemně překrývají.



Obrázek 2.1: Poloha řešení (včel) před provedením algoritmu, kde červenou se značí nejlepší řešení (ohodnocení).²



Obrázek 2.2: Poloha řešení (včel) po provedení algoritmu, kde červenou se značí nejlepší řešení (ohodnocení).⁴

Kapitola 3

Ovládání programu

3.1 Spuštění programu

Spuštění na systému Linux:

```
pip install tk - instalace grafického rozraní pythonu  
python bees__algorithm.py
```

Spuštění na serveru merlin přes X11 tunelování:

```
ssh -X merlin.fit.vutbr.cz  
python bees__algorithm.py
```

3.2 Popis GUI

Hlavní okno (root):

Okno, které zobrazuje hlavní GUI.

Canvas (šedé podokno):

Slouží pro vykreslování grafických prvků, zejména pro vizualizaci hledání nejlepšího ohodnocení včely a průběhu algoritmu.

Posuvníky (Scale):

Posuvníky umožňují uživateli nastavovat parametry algoritmu. Jsou zde čtyři posuvníky:

- **Space distance:** Nastavuje rozsah prostoru, ve kterém se včely pohybují.
- **Max Iterations:** Určuje maximální počet iterací algoritmu.
- **Scout Bees ratio:** Nastavuje poměr včel, které budou provádět globální hledání (skauti).
- **Colony Size:** Určuje velikost včelí kolonie.

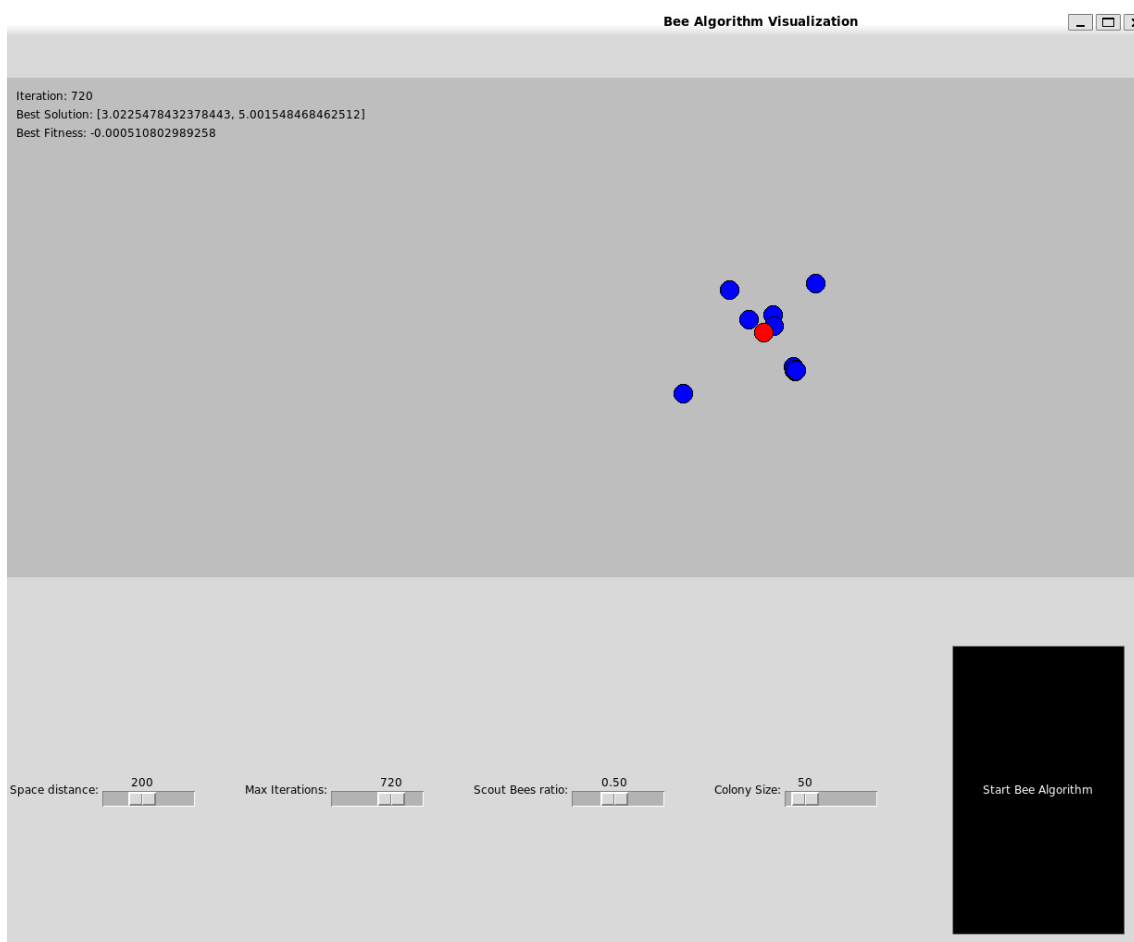
Tlačítko "Start Bee Algorithm":

Po stisknutí tohoto tlačítka se spouští algoritmus pro vizualizaci včelí kolonie s aktuálními nastaveními parametrů.

Metody třídy `BeeAlgorithmGUI`:

- `create_slider_box`: Vytváří skupinu ovládacích prvků pro nastavování hodnot pomocí posuvníku.
- `update_space_range`, `update_max_iterations`, `update_scout_bees_ratio`, `update_colony_size`: Aktualizují hodnoty příslušných parametrů při pohybu posuvníky.
- `start_algorithm`: Spouští algoritmus s vizualizací na základě aktuálních hodnot parametrů.

Celkově lze GUI popsat jako interaktivní prostředí, které umožňuje uživateli jednoduše nastavovat parametry algoritmu a vizualizovat jeho průběh. Vykreslené včely na kanvasu reprezentují členy kolonie, a uživatel může sledovat, jak se jejich polohy mění během průběhu algoritmu.



Obrázek 3.1: GUI¹.