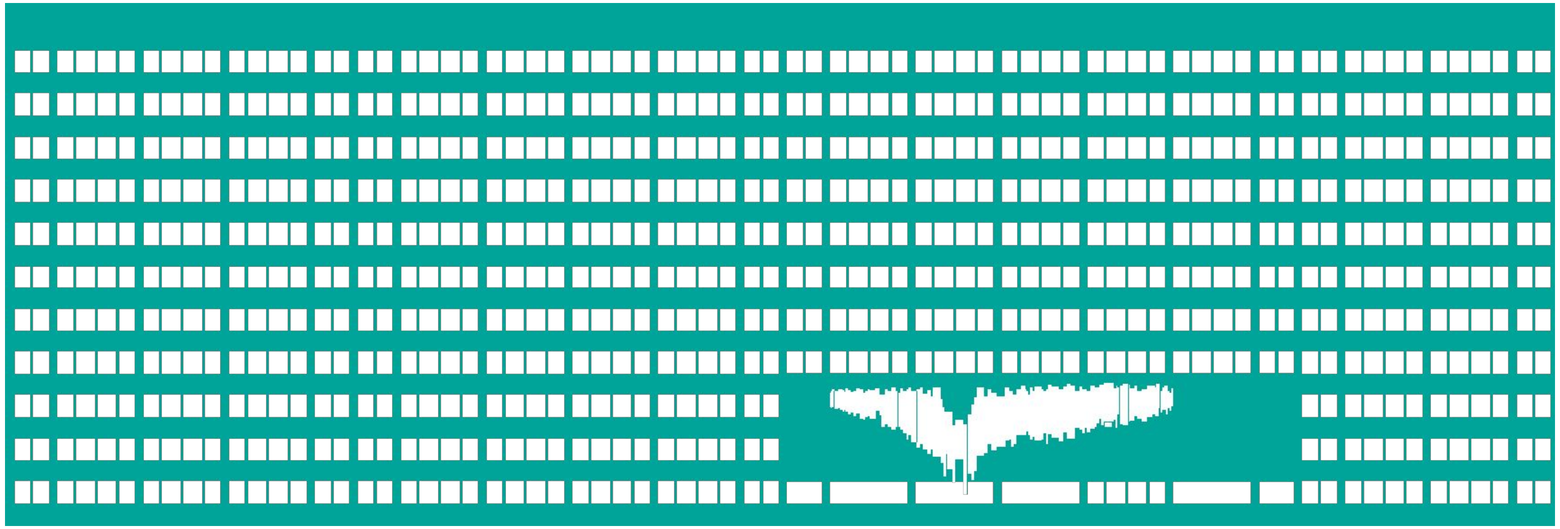
VŠB TECHNICKÁ UNIVERZITA OSTRAVA

VSB TECHNICAL UNIVERSITY OF OSTRAVA



www.vsb.cz

Biologically inspired algorithms
Exercise 11

Ing. Lenka Skanderová, Ph. D.

Content

- NSGA II – Non-Dominated Sorting Genetic Algorithm

History

- Published in April 2002
- Pareto-ranking approach
- Fast non-dominated sorting approach
- Diversity preservation using fitness crowding

Example

- Find an optimal solution for the objective functions:
  - $f_1(x) = -x^2$
  - $f_2(x) = -(x-2)^2$

- The solution must be optimal for both objective functions

- $-55 < x < 55$

Example – Maximization/Maximization

- Find an optimal solution for the objective functions:
  - $f_1(x) = -x^2$
  - $f_2(x) = -(x-2)^2$

- Generate the first population:

$$P_x = [-2 \;\; -1 \; 0 \; 2 \; 4 \; 1]$$

- Evaluate objective functions f_1 and f_2

$$f_1(x) = [-4 \;\; -1 \; 0 \;\; -4 \;\; -16 \;\; -1]$$

$$f_2(x) = [-16, -9, -4, 0, -4, -1]$$

Example – Maximization/Maximization

- We need to get Pareto ranks (denoted as Q)

- Create three sets:
  - $Q = \{\}$ … Pareto rank
  - $S = \{\}$ … For each solution, the dominated (worse) solutions are recorded
  - $n = \{\}$ … For each solution, the number of dominating (better) solutions are recorded

Example – Maximization/Maximization

- Start with the first solution $x = -2$, $f_1(x) = -4$, $f_2(x) = -16$
- Compare this solution with each other one:
  - -4 < -1 and -16 < -9:
    - $S_0 = \{\}, n_0 = 1$
  - -4 < 0 and -16 < -4:
    - $S_0 = \{\}, n_0 = 2$
  - -4 = -4 and -16 < 0:
    - $S_0 = \{\}, n_0 = 3$
  - **-4 > -16** and -16 < -4:
    - $S_0 = \{\}, n_0 = 3$
  - -4 < -1 and -16 < -1:
    - $S_0 = \{\}, n_0 = 4$

$f_1(x)$	-4	-1	0	-4	-16	-1
$f_2(x)$	-16	-9	-4	0	-4	-1

For the first solution we will get:

$$S_0 = \{\}, n_0 = 4$$

Example – Maximization/Maximization

- Continue with the second solution $x = -1$, $f_1(x) = -1$, $f_2(x) = -9$
- Compare this solution with each other one:
  - -1 > -4 and -9 > -16:
    - $S_1 = \{0\}, n_1 = 0$
  - -1 < 0 and -9 < -4:
    - $S_1 = \{0\}, n_1 = 1$
  - -1 > -4 and -9 < 0:
    - $S_1 = \{0\}, n_1 = 1$
  - -1 > -16 and -9 < -4:
    - $S_1 = \{0\}, n_1 = 1$
  - -4 < -1 and -9 < -1:
    - $S_1 = \{0\}, n_1 = 2$

$f_1(x)$	-4	-1	0	-4	-16	-1
$f_2(x)$	-16	-9	-4	0	-4	-1

For the first solution we will get:

$$S_1 = \{0\}, n_1 = 2$$

Example – Maximization/Maximization

- Continue with the third solution $x = 0, f_1(x) = -1, f_2(x) = -9$
- Compare this solution with each other one:
  - 0 > -4 and -4 > -16:
    - $S_2 = \{0\}, n_2 = 0$
  - 0 > -1 and -4 > -9:
    - $S_2 = \{0, 1\}, n_2 = 0$
  - 0 > -4 and -4 < 0:
    - $S_2 = \{0,1\}, n_2 = 0$
  - 0 > -16 and -4 = -4:
    - $S_2 = \{0, 1, 4\}, n_2 = 0$
  - 0 > -1 and -4 < -1:
    - $S_2 = \{0,1,4\}, n_2 = 0$

$f_1(x)$	-4	-1	0	-4	-16	-1
$f_2(x)$	-16	-9	-4	0	-4	-1

For the first solution we will get:

$$S_2 = \{1\}, n_1 = 2$$

Example – Maximization/Maximization

- Continue with the fourth solution $x = 2$, $f_1(x) = -4$, $f_2(x) = 0$
- Compare this solution with each other one:
  - -4 = -4 and 0 > -16:
    - $S_3 = \{0\}, n_3 = 0$
  - -4 < -1 and 0 > -9:
    - $S_3 = \{0\}, n_3 = 0$
  - -4 < 0 and 0 > -4:
    - $S_3 = \{0\}, n_3 = 0$
  - -4 > -16 and 0 > -4:
    - $S_3 = \{0, 4\}, n_3 = 0$
  - -4 < -1 and 0 > -1:
    - $S_2 = \{0,4\}, n_3 = 0$

$f_1(x)$	-4	-1	0	-4	-16	-1
$f_2(x)$	-16	-9	-4	0	-4	-1

For the first solution we will get:

$$S_2 = \{1\}, n_1 = 2$$

Example – Maximization/Maximization

- Continue with the fifth solution $x = 4$, $f_1(x) = -16$, $f_2(x) = -4$ .Compare this solution with each other one:
  - -16 < -4 and -4 > -16:
    - $S_4 = \{\}, n_4 = 0$
  - -16 < -1 and -4 > -9:
    - $S_4 = \{\}, n_4 = 0$
  - -16 < 0 and -4 = -4:
    - $S_4 = \{\}, n_4 = 1$
  - -16 < -4 and -4 < 0:
    - $S_4 = \{\}, n_4 = 2$
  - -16 < -1 and -4 < -1:
    - $S_4 = \{\}, n_4 = 3$

$f_1(x)$	-4	-1	0	-4	-16	-1
$f_2(x)$	-16	-9	-4	0	-4	-1

For the first solution we will get:

$$S_2 = \{1\}, n_1 = 2$$

Example – Maximization/Maximization

- Continue with the sixth solution $x = 1$, $f_1(x) = -1$, $f_2(x) = -1$ .Compare this solution with each other one:
  - -1 > -4 and -1 > -16:
    - $S_5 = \{0\}, n_5 = 0$
  - -1 = -1 and -1 > -9:
    - $S_5 = \{0,1\}, n_5 = 0$
  - -1 < 0 and -1 > -4:
    - $S_4 = \{0, 1\}, n_4 = 0$
  - -1 >-4 and -1 < 0:
    - $S_4 = \{0,1\}, n_4 = 0$
  - -1 > -16 and -1 > -4:
    - $S_4 = \{0,1,4\}, n_4 = 0$

$f_1(x)$	-4	-1	0	-4	-16	-1
$f_2(x)$	-16	-9	-4	0	-4	-1

For the first solution we will get:

$$S_2 = \{1\}, n_1 = 2$$

Example – Maximization/Maximization

- We have the following sets:
  - $S = \{\{\ \},\{0\},\{0,1,4\},\{0,4\},\{\ \},\{0,1,4\}\}$
  - $n = \{4, 2, 0, 0, 3, 0\}$
- Select all solutions where $n = 0$
  - x_2, x_3, x_5 and put them to the first rank $\rightarrow Q_1 = \{x_2, x_3, x_5\}$
- Solution x_2 dominates to solutions x_0, x_1, x_4:
  - Go to the set n and for x_0, x_1, x_4 add -1 in $n \rightarrow n = \{3, 1, 0, 0, 2, 0\}$
- Solution x_3 dominates to solutions x_0 and x_4:
  - Go to the set n and for x_0, x_4 add -1 in n $\rightarrow n = \{2, 1, 0, 0, 1, 0\}$
- Solution $x_5$dominates to solutions x_0, x_1, x_4:
  - Go to the set n and for x_0, x_1, x_4 add -1 in $n \rightarrow n = \{1, \mathbf{0}, 0, 0, \mathbf{0}, 0\}$

$\Rightarrow Q_2 = \{x_1, x_4\}$

Example – Maximization/Maximization

- Repeat the previous process to get the next Pareto rank Q_3:
  - $Q_3 = \{x_0\}$

- Now we have divided the solutions into Pareto ranks:
  - $Q_1 = \{x_2, x_3, x_5\}$ best
  - $Q_2 = \{x_1, x_4\}$ worse
  - $Q_3 = \{x_0\}$ worst

- We can use GA:
  - For each solution, select a random individual
  - Crossover
  - Mutation
  - Append a new offspring to the population
  - At the end of the generation, we will have $2NP$ individuals

Example – Maximization/Maximization

- We can use GA:
  - For each solution, select a random individual
  - Crossover
  - Mutation
  - Append a new offspring to the population
  - At the end of the generation, we will have $2NP$ individuals
  - We will use the non-dominaned sorting to select NP best individuals

- Crossover for a 1-dimensional problem :

```
if np.random.uniform() < 0.5:
        return (solution1.params + solution2.params)/2
else:
        return (solution1.params - solution2.params)/2
```

Example – Maximization/Maximization

- Mutation for a 1-dimensional problem :

```
    if np.random.uniform() < 0.5:
        return cross + np.random.uniform(0,1,dimension) #cross - list of parameters
    else:
        return cross
```

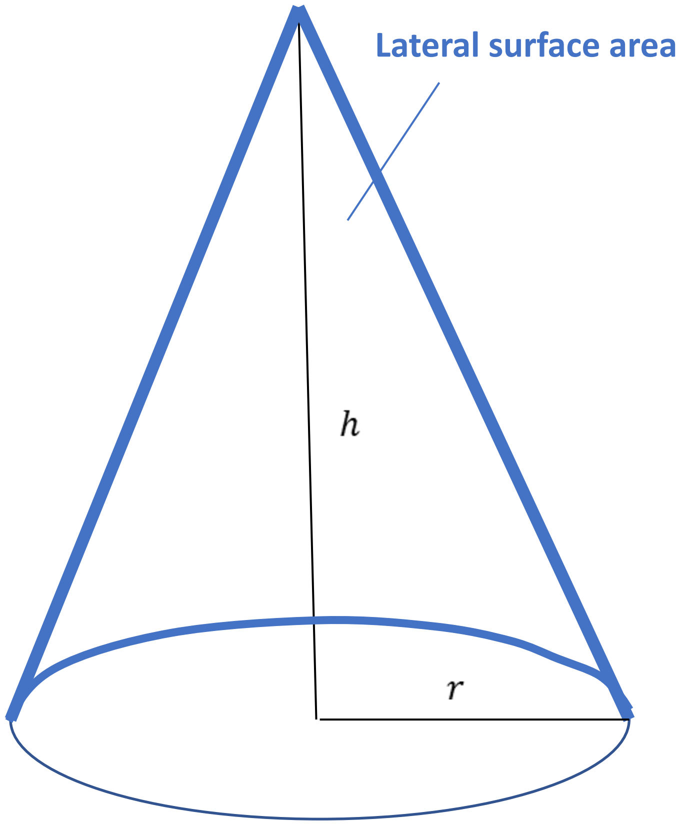
Task

Implement the NSGA II and use it to solve the following cone problem:

- Let r be a radius of cone
- Let h be a height of cone

- Minimize the cone's lateral surface area S and total area T
- $V = \frac{\pi}{3} r^2 h$
- $s = \sqrt{r^2 + h^2}$
- $S = \pi r s$
- $T = \pi r (r + s)$
- $0 < r < 10, 0 < h < 20$

- Visualize the set of solutions, **highlight the Pareto set**



Literature:

[1] Rao, R. Venkata, Vimal J. Savsani, and D. P. Vakharia. "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems." *Computer-Aided Design* 43.3 (2011): 303-315.

Thank you for your attention

Ing. Lenka Skanderová, Ph.D.

EA 407
+420 597 325 967

lenka.skanderova@vsb.cz

homel.vsb.cz/~ska206