Assignment 14 Selection Mechanisms and Selection Pressure

Consider the following (single-objective) optimization problem, which we want to optimize using an Evolutonary Algorithm (EA):

$$max \quad f(x) = -x^4 + 8x^2 + 10x + 1000$$
$$x \in [-8, 8]$$

This function has a local maximum at $x \approx -1.5425$ and a global maximum around $x \approx 2.2597$. The population of an EA in a given timestep t consists of the following eight solutions:

$$x^{(1)} = -7.0 \quad x^{(2)} = -2.0 \quad x^{(3)} = -1.0 \quad x^{(4)} = -0.5$$
$$x^{(5)} = 1.3 \quad x^{(6)} = 2.0 \quad x^{(7)} = 2.5 \quad x^{(8)} = 3.0$$

- Calculate the fitness values and the relative fitness values of this population.

Fitness values,

$f(x^{(1)}) = -1079$	$f(x^{(2)}) = 996$	$f(x^{(3)}) = 997$	$f(x^{(4)}) = 996.93$
$f(x^{(5)}) = 1023.66$	$f(x^{(6)}) = 1036$	$f(x^{(7)}) = 1035.93$	$f(x^{(8)}) = 1021$

$$\sum_{i=1}^{8} f\, i = 6027.53$$

Relative Fitness values,

$p_1 = -0.18$	$p_2 = 0.1652$	$p_3 = 0.16540$	$p_4 = 0.1654$
$p_5 = 0.17$	$p_6 = 0.17187$	$p_7 = 0.17186$	$p_8 = 0.1693$

- Is the Roulette Wheel selection applicable to this problem? Which modifications might be necessary?

With the above values, RWS implementation is not applicable since we have a negative probability value which means for selecting $x^{(1)}$ it has a negative chance of selection which is undesirable.

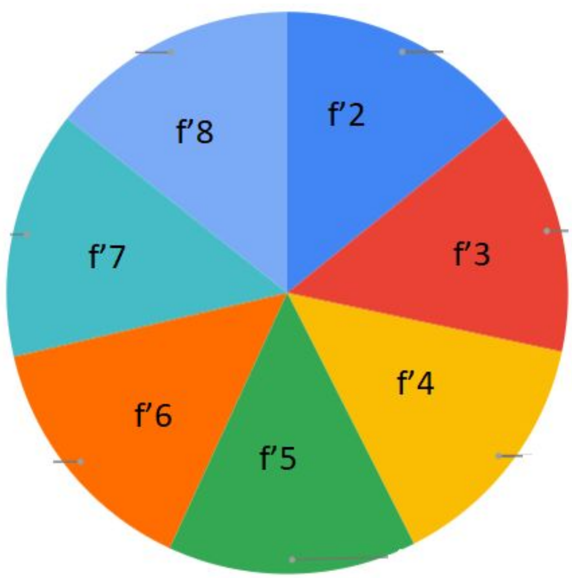
The roundabout to this is to normalize all the fitness values which eliminate one or more solutions are not fit for implementation.

$$f_i' = \frac{f_i - f_{worst}}{f_{best} - f_{worst}}$$

We take f_{best} as 1036 ≈ $x^{(6)}$ and f_{worst} as -1079 ≈ $x^{(1)}$

Normalized Relative Fitness values,

f'_1 = 0	f'_2= 0.98	f'_3= 0.98	f'_4= 0.98
f'_5= 0.99	f'_6= 1.0	f'_7= 0.99	f'_8= 0.99



So, after normalization, we observe a zero-probability for $x^{(1)}$, which means that event never happens and so we implement our RWS with 7 solutions in it with their new relative probability of selection. Also, the influence of the worst solution can be seen notably in the fitness values.

- Select from this population three individuals with Roulette Wheel selection and three individuals with Tournament selection. Explain how these two methods work.

Roulette-Wheel Selection:

- A fitness proportional selection method, where individual solutions are assigned a probability(p_i) based on their fitness value (f_i).
- The wheel is divided into N divisions (population size), where the size of each is marked in proportion to the fitness of each solution.
- Thereafter, the wheel is spun N times, each time choosing the solution indicated by the area covered by a pointer/dice/coin.
- If an individual has a higher fitness value then any other, then RWS will select/prefer that solution more often than any other solution.
- On the contrary, if all solutions have an equal fitness value then everyone will be selected equally with the least selection pressure.

By RWS,

Solutions	Fitness, f'	p_i
$x^{(2)}$	0.98	0.141
$x^{(3)}$	0.98	0.141
$x^{(4)}$	0.98	0.141
$x^{(5)}$	0.99	0.143
$x^{(6)}$	1	0.144
$x^{(7)}$	0.99	0.143
$x^{(8)}$	0.99	0.143

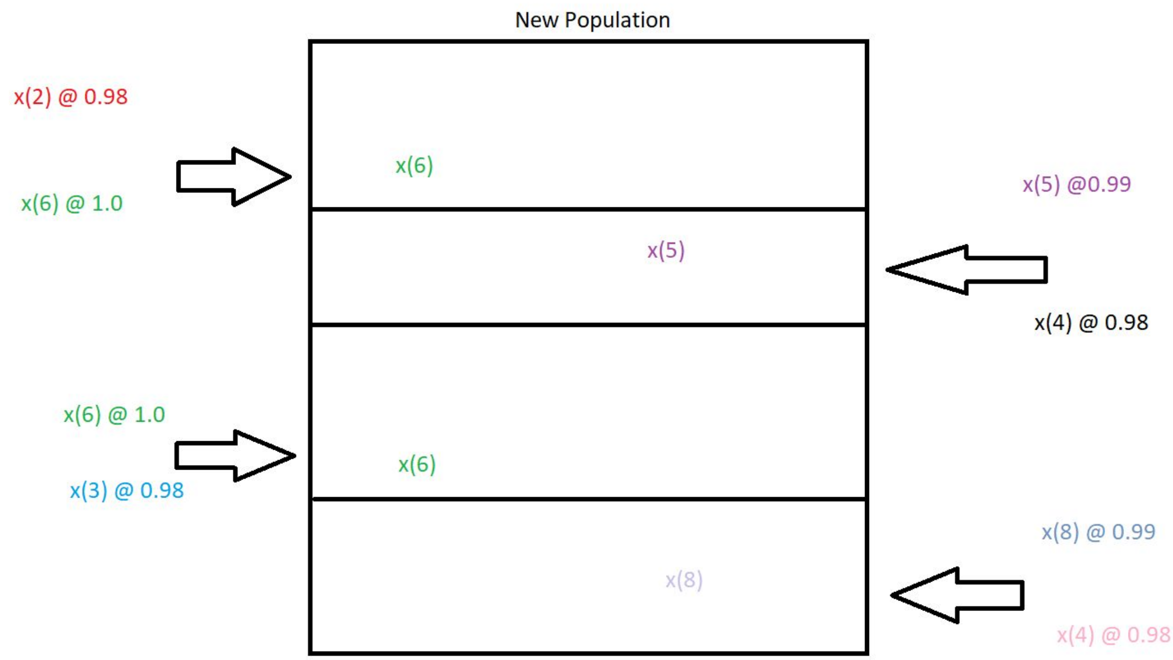
Solution $x^{(6)}$ is the one which has slightly highest probability which makes it as a candidate for definite selection(super individual). Alongside, out of

$x^{(2)}$,$x^{(3)}$,$x^{(4)}$ anyone can be selected as they all have an identical probability of selection and so with $x^{(5)}$,$x^{(7)}$,$x^{(8)}$ with least selection pressure. As a result, there is a case for selecting bad solutions since it is made random.

Tournament selection:

- In this selection method, tournaments are played between two solutions and the better one is chosen.
- The two other solutions are picked again in another slot and the pool is filled with the better among those.
- When carried out systematically, each solution can be made to participate exactly in two tournaments.
- The best one will win in both times, thereby making duplicate copies of it in the new population.
- Similarly, the worst solution will loose in both tournaments and will be eliminated from the population.
- Individuals (q) are randomly chosen, usually $q=2$
- Ease of implementation, better convergence leading to efficient computation.

By Tournament selection,



- Take a look at the concept of selection pressure from the lecture. Which differences do you expect for the convergence speed (= number of generations the EA might need to reach the global maximum) if Roulette Wheel selection or Tournament selection is used in this situation?

Since RWS uses the probability values along with the fitness values the solution having the largest probability gets more chance of selection very often leading to a less diverse solution (no selection pressure, no search focus) the convergence speed is fast.

Whereas in Tournament selection we play tournaments among solutions by their fitness values we, however, obtain diverse solutions at the cost of higher selection pressure leading to lesser convergence speed compared to RWS.