

# Bioinspired Optimization and Design

---

## Project 2: Knapsack Problem – Part II

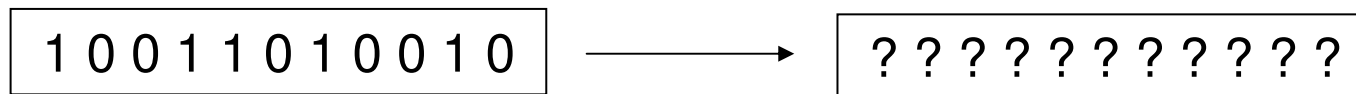
Tamara Ulrich

[tamara.ulrich@tik.ee.ethz.ch](mailto:tamara.ulrich@tik.ee.ethz.ch)

# Task 1 a)

a) Define a mutation operator for the knapsack problem. Note: you can make use of the neighborhood function.

- The mutation operator can be seen as the counterpart to the neighborhood function in local search



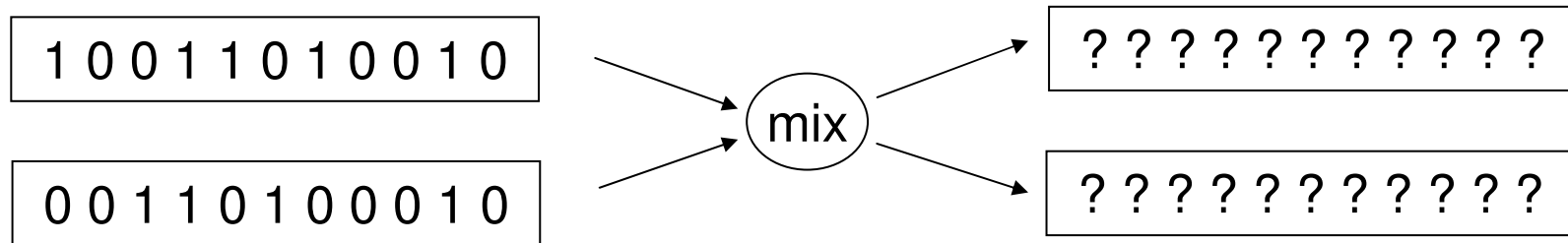
- Properties of the mutation operator:
  1. Every solution can be generated from every other solution by means of mutation with a probability greater than 0.
  2.  $d(x, x') < d(x, x'') \Rightarrow \text{Prob}[\text{mut}(x) = x'] > \text{Prob}[\text{mut}(x) = x'']$



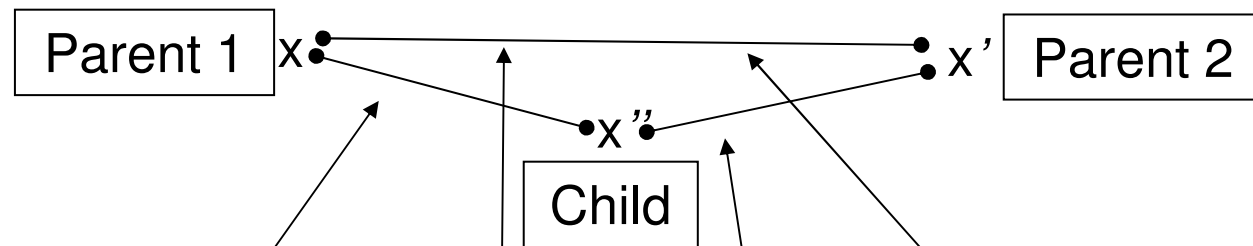
- Define a mutation operator which *should* have the above properties.

# Task 1 b)

b) Define a recombination operator, which produces two offspring solutions from a given pair of parent solutions.



- The recombination operator distinguishes evolutionary algorithms from other randomized search algorithms; similarly to mutation operators, a desirable property of a recombination operator is:



$$x'' = \text{recomb}(x, x') \Rightarrow d(x, x'') \leq d(x, x') \quad \square \quad d(x', x'') \leq d(x, x')$$

- Your recombination operator again *should* have that property.

## Task 1 c), d)

*c) Define a mating selection operator.*

*d) Define an environmental selection operator.*

- Two types of selection schemes can be distinguished:
  - ① **stochastic selection** consisting of
    - **sampling rate assignment**  
 $Q_i$  = probability that individual  $i$  is chosen
    - **sampling**  
choose  $N$  individuals according to their sampling rates
  - ② **deterministic selection**

# Task 1 c), d)

c) Define a mating selection operator.

d) Define an environmental selection operator.

- Two types of selection schemes can be distinguished:
  - stochastic selection** consisting of → **Mating Selection**
    - sampling rate assignment**  
 $Q_i$  = probability that individual  $i$  is chosen
    - sampling**  
choose  $N$  individuals according to their sampling rates
  - deterministic selection** → **Environmental Selection**

## Task 1 e)

e) *Implement an evolutionary algorithm with population size 100 and the operators you defined in a)– d). Use the objective function from project 1 as the fitness function. Run the algorithm on the problem instance given on the lecture website for 1000 generations. Create a plot to show how the best objective function value of the population develops over time (plot generations vs. best profit reached so far)*



# Task 1 e) : Evolutionary Algorithm

- 1: Randomly choose  $x_1, x_2, \dots, x_N$  from  $X$
- 2: Calculate  $f(x_1), f(x_2), \dots, f(x_N)$  ← Project 1
- 3: Initialize memory, i.e.,  $M = \{(x_1, f(x_1)), (x_2, f(x_2)), \dots, (x_N, f(x_N))\}$
- 4: Set iteration counter  $t = 0$
- 5: Assign each  $x_i$  in  $M$  a fitness value  $F_i = f(x_i)$
- 6: **loop**
- 7:    $t = t + 1$
- 8:    $M' =$  mating selection from  $M$  ← Task 1 c)
- 9:    $M'' =$  mutation and recombination on  $M'$  ← Task 1 d)
- 10:   Calculate  $f(x_i)$  for  $x_i$  in  $M''$
- 11:   Assign each  $x_i$  in  $M''$  a fitness value  $F_i = f(x_i)$
- 12:    $M =$  environmental selection on  $M$  and  $M''$  → plot point ( $t, \max(F_i)$ )
- 13:   **if**  $t > t_{MAX}$  **then**
- 14:     Output best solution  $x^*$  stored in  $M$
- 15:     Stop
- 16:   **end if**
- 17: **end loop**

## Task 1 f) g)

*f) How would you classify the constraint handling technique used so far? Suggest an alternative constraint handling technique and describe how you would have to change the algorithm accordingly.*

- So far you set the objective function to 0 if the items exceed the maximum weight.
- Advantages/disadvantages?
- Suggest new constraint handling.

*g) Run the modified algorithm on the same instance of e) and produce a similar plot. What differences do you see?.*

- Please plot the old (1e) and new (1g) results to the same figure.



# Task 1 h)

*h) Competition: Optimize the knapsack problem given in the problem instance on the website. You can either use the EA developed in this task, or some other optimization algorithm of your choice. Report the best solution found (both the chosen items and the objective value). Note: you can run the algorithm as long as you like, there is no limit to the number of iterations.*

- Be sure to pick the correct problem instance.
- The best solution wins.
- You can use any algorithm you like, as long as you implemented it yourself.
- If you decide to use the EA developed in this task, you can of course modify it.

## Project 2: Knapsack Problem - Part 2

### Attestation Conditions

In order to fulfill the attestation conditions, at least 40 points need to be reached for each project and 200 points in total. Task 1 has to be submitted by March 26, 24:00 and task 2 by April 2 24:00.

### Assistance

Teaching Assistant [Tamara Ulrich](#)

### Materials

### Tasks

[Exercise sheet \(pdf\)](#)

### Instances

#### Task 1

[Subtasks e\) and g\)](#)  
[Subtask h\)](#)

Competition



#### Task 2

[Subtask b\)](#) (identical to the instance from subtasks e) and g))

### Data Format

All problem parameters are stored in ASCII text files. Each line contains one integer number followed by a newline character. The numbers are sorted in the following order:

```
<number of items n>
<profit of item 1>
<profit of item 2>
...
<profit of item n>
<weight of item 1>
<weight of item 2>
...
<weight of item n>
<capacity>
```

For example :

```
3
125
321
105
```

# Task 2

In this task, the knapsack problem is viewed as a multiobjective optimization problem, where the objectives are to simultaneously maximize the profit and minimize the weight. Since the weight is now an objective and not a constraint anymore, all solutions are feasible. The aim is to find a good approximation of the Pareto front.

	Singleobjective (Task 1)	Multiobjective (Task 2)
Profit	Maximize	Maximize
Weight	Do not exceed capacity	Minimize

## Task 2 a)

*a) We want to re-use the algorithm from Task 1. Describe, which parts of the algorithm have to be modified and how.*

- representation?
- objective value calculation / fitness assignment?
- mutation?
- recombination?
- mating selection?
- environmental selection?
- ...

## Task 2 b)

- b) Run the algorithm on the problem instance given on the website.  
Report the Pareto front approximation found after 500 and after 1000 generations as a two dimensional plot, where the x-axis denotes the profit and the y-axis the weight.

