Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für
Technische Informatik und
Kommunikationsnetze

Prof. L. Thiele

Hardware/Software Codesign - HS 15

Solution for Exercise Sheet 5

Issue Date: 21 October 2015
Discussion Date: 28 October 2015

5.1 Properties of Sets

Consider the following constrained multiobjective problem.

$$\begin{aligned} \min f_1(x) &= x^2 \\ \min f_2(x) &= (x-5)^2 \\ x &\in [-10, 10] \end{aligned} \tag{1}$$

Consider the following two *sets* of solutions.

$$\begin{aligned} S_1 &= \{9, 3, 8\} \\ S_2 &= \{1, 2, 7\} \end{aligned}$$

(a) For each set of solutions, construct the preordering imposed by the two objectives. Which set has a smaller number of pareto-dominated solutions?

Solution: *For S_1, the mapping of points to objectives:* $9 \rightarrow (81, 196)$, $3 \rightarrow (9, 4)$, $8 \rightarrow (64, 9)$.
The preordering is $3 \prec 8 \prec 9$.
The number of pareto-dominated solutions is 2, i.e. $\{8, 9\}$.

For S_2, the mapping of points to objectives: $1 \rightarrow (1, 16)$, $2 \rightarrow (4, 9)$, $7 \rightarrow (49, 4)$.
The preordering does not impose any relations.
The number of pareto-dominated solutions is 0.

S_2 has fewer pareto-dominated solutions.

(b) The hypervolume indicator is computed with respect to point or points in the objective space. For this problem propose a single point you would choose as reference and explain why?

Solution: *We are trying to minimize two objective functions. One possibility for the reference point in the objective space is to choose the highest possible values for the two objective functions. As we are constrained to use $x \in [-10, 10]$, such a reference point could be* $(100, 225)$.

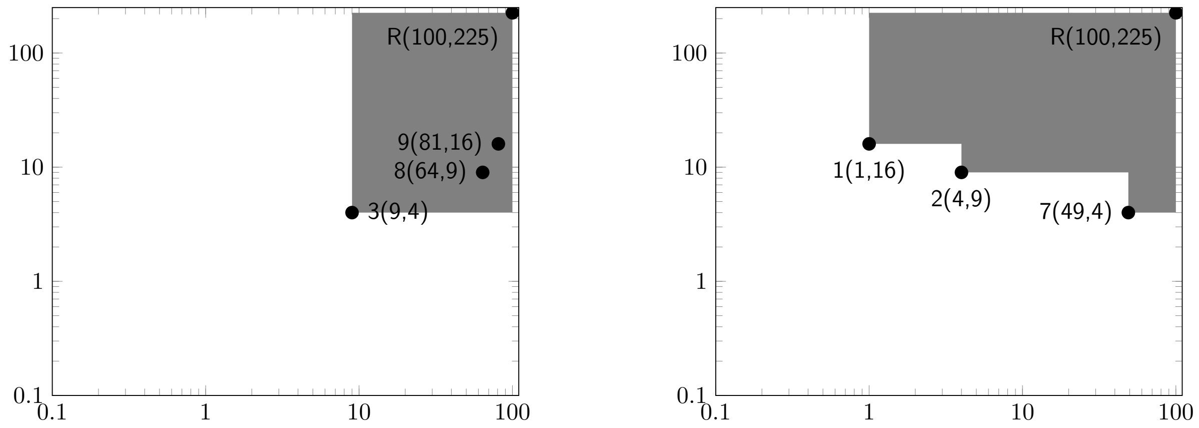
(c) For each set of solutions, compute the hypervolume indicator with respect to the point chosen in the previous question. Which set has a higher hypervolume indicator? You may use the following log plots to assist you in computing hypervolume.

Solution:

Hypervolume indicators are computed as the shaded areas in Figure 1.

For S_1: area $= (100 - 9) \times (225 - 4) = 20,111$.
For S_2: area $= (100 - 49) \times (225 - 4) + (49 - 4) \times (225 - 9) + (4 - 1) \times (225 - 16) = 21,618$.

The hypervolume indicator is greater for the set S_2.



(a) Solution set S_1 in the objective space

(b) Solution set S_1 in the objective space

Figure 1: Computing hypervolume indicator for S_1 and S_2

(d) Which of the two sets is a better approximation of the pareto-set? Why?

Solution: *The solution set S_2 is better approximation of the pareto-set, because it has fewer pareto-dominated points and a higher hypervolume indicator.*

5.2 Evolutionary Algorithm

Consider the following algorithm to solve the problem from (1). Here $F_H(\cdot)$ defines the hypervolume indicator of a set with respect to the point computed in the previous question.

```
Algorithm 1: My First Evolutionary Algorithm
Data: Solution set $S$
Result: Optimized solution set $S$
repeat
    Delete from $S$ all pareto-dominated solutions;
    $S' \leftarrow$ set of numbers by averaging each pair of values in $S$;
    $S'' \leftarrow S \cup S'$;
    Delete from $S''$ all pareto-dominated solutions;
    if $F_H(S'') \geq F_H(S)$ then
        $S \leftarrow S''$;
    end
until number of iterations $\leq 10$;
```

(a) Apply one iteration of the algorithm for the two given solution sets S_1 and S_2, and compute the resultant solution sets.

Solution:

For S_1 *We remove 9 and 8, as they are Pareto-dominated. Thus, $S_1' = \phi$ and $S_1'' = \{3\}$. As we have only one element in the solution set, the algorithm terminates with this solution.*

For S_2 *None of the initial points are Pareto-dominated. Then, $S_2' = \{1.5, 4.5, 4\}$. Them, $S_2'' = \{1, 1.5, 2, 4, 4.5, 7\}$. The solution 7 is dominated by both 4 and 4.5. Hence, we remove it from S_2''. Now we compute the hypervolume indicator of S_2''.*

(b) The proposed algorithm will not find the Pareto-optimal solution for (1) for either solution set S_1 and S_2. Explain why?

Solution: *For S_1, as seen in the very first step, the only solution is 3 and the algorithm terminates here.*

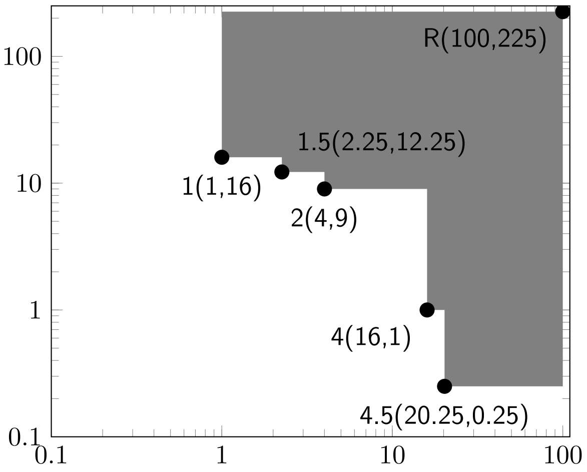


Figure 2: Computation of hypervolume indicator for S_2''

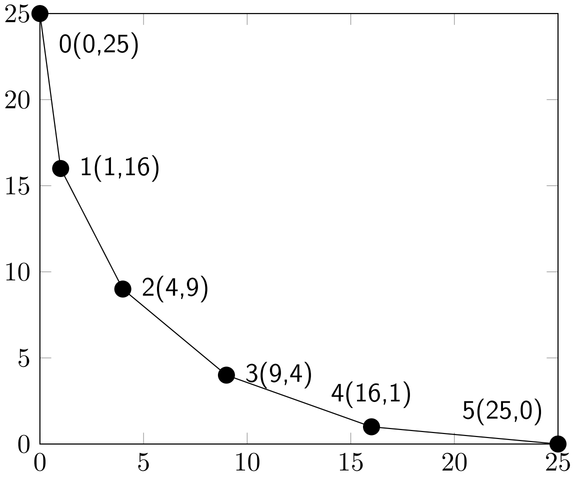


Figure 3: Integer Pareto-optimal solutions in the objective space

For S_2, because we use the average of two numbers (as the crossover operator) the computed solutions will be upper and lower bounded by the maximum and minimum points in the initial set, namely 1 and 7, respectively. This does not cover the whole decision space and may miss some Pareto-optimal solutions. For instance, 0 is a Pareto-optimal solution and will never be reached by the algorithm.

(c) How would you modify the algorithm to solve the problem in (1)? Can you identify the pareto-optimal set of solutions?

Solution: *The algorithm can be modified in different ways. One simple way is to add a mutation operation that takes any solution and add/subtracts a random number from it.*

We can show that each point in $[0, 5]$ is a pareto-optimal point. Then, the pareto-optimal front in the objective space is given by the set of points $(x^2, (x-5)^2)$ for $x \in [0, 5]$. The integer pareto-optimal solutions are plotted in Figure 3.

5.3 A Path-Finding Problem

Consider the following problem of planning the path of a robot on a terrain shown in the following figure. The entry and exit points and orientations of the robot are shown. Also shown are checkpoints which are places the robot must visit at least once.

The robot can either move ahead by a step of 1, or turn left or right by 90 degrees. With these atomic operations, we are to plan the path of the robot.

There are two quantities we want to minimize: the number of move operations, and the number of turn operations.

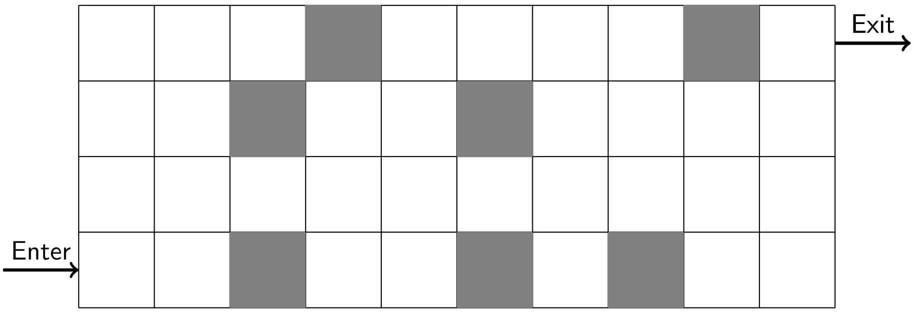


Figure 4: Terrain map for the robot with entry and exit points and orientations and checkpoints (gray boxes).

Propose an evolutionary algorithm to solve this problem. In particular, provide

(a) an appropriate encoding of the decision space,

Solution: *There are several correct solutions.*

One intuitive approach is to encode the entire path. However, with such an encoding there are two problems. Firstly, the decision space is very large - this may lead to longer optimization times or solutions of poor quality. Secondly, the crossover and mutation operations with such an encoding are not obvious. Consider for instance two paths of different lengths. What are reasonable crossover operations? These are important considerations in designing the encoding.

Another example is to encode the solution as a set of two $N-$tuples, where N is the number of checkpoints. The first N-tuple is a permutation of $\{1, 2, \ldots, N\}$. The second N-tuple has elements from the set $O = \{0, 90, 180, 270\}$ The interpretation of these two tuples is as follows. The first tuple gives the order in which the robot visits the checkpoints. The second tuple gives the orientation of the robot when visiting the corresponding checkpoint. An example solution for N =3 checkpoints is $((2, 1, 3), (90, 180, 90))$. The robot visits the checkpoints 2, 1, and 3 in order. The respective entry angles are 90, 180 and 90.

It is important that an encoded solution map to exactly one point in the objective space (has unique number of steps and number of turns). This is true with the specified encoding. Given the current position and angle, and a destination position and angle, there are finite number of paths that minimize both the number of moves and the number of turns. (Exercise: Verify this.) Thus, given an encoding as specified, we obtain a series of sub-paths that are concatenated to form he full path.

(b) explicit definition of any constraints,

Solution: *With the proposed encoding there are the following constraints on the tuples: the first tuple is a permutation and the second drawn from elements of the specific set O. Furthermore, all entry angles are not valid for all checkpoints. Forexample, checkpoints on the lowest row cannot be entered from south/90°. These are constraints on the decision space. There are no constraints in the objective space.*

(c) neighborhood operations of crossover and mutation,

Solution: *Neighborhood operations can be defined for the two tuples separately. The first tuple is a permutation and standard crossover operation using parts of two permutations (as discussed in the lecture) can be used. Swap operation can be used as a mutation. For the second tuple, an example of a crossover operation is adding up the tuples entrywise and taking a remainder with 360. Mutation operations may randomly change one or more entries to belong to the set O.*

(d) environmental selection operation.

Solution: *Any standard approach can be used - for instance based on the hypervolume indicator as discussed in the lecture.*

Manually try out a few steps of your proposed algorithm.