Semester I Examinations 2020/2021

Exam Code(s)	1MAI1, 1CSD1, 1SPE1, 2SPE1
Exam(s)	MSc in Computer Science (Artificial Intelligence)
Module Code(s)	CT5141
Module(s)	**Optimisation**
Discipline	School of Computer Science
Paper No.	1
External Examiner	Prof. Pier Luca Lanzi
Internal Examiner(s)	Prof. Michael Madden
	Dr. James McDermott *
Programme Coordinator(s)	Dr. Matthias Nickles

Instructions

Answer any 4 questions. All are worth equal marks.

You may answer either: in a Word document or similar, and then `CT5141_Optimisation_Answer_Sheet.docx` is suggested; or on paper, uploading a scan of the pages.

This is an **open-book** exam: you may **read** textbooks, notes, and **existing resources** on the internet.

You may **not communicate** with anyone, in person, via phone, internet, or otherwise. You may **not post questions** on internet sites or elsewhere during the exam.

Duration	2 Hours exam plus 30 minutes for upload
Number of pages	5 (including this page)
Discipline	Computer Science

Requirements

Release in Exam Venue	Yes ☒ No ☐
Release to Library	Yes ☒ No ☐

Question 1: Linear programming

A paint company manufactures paint by mixing three main ingredients. They aim to minimise costs by choosing the proportions of the three. They are required to produce exactly 1000L per day. The costs for the three ingredients are (0.25, 0.4, 0.6) EUR/L. Ingredient 1 must be filtered before use and the filtering machine has a capacity of 400L per day. Ingredient 2 fails to mix correctly unless there is an equal or greater amount of Ingredient 3. Ingredient 2 is manufactured in-house and only 400L is available per day.

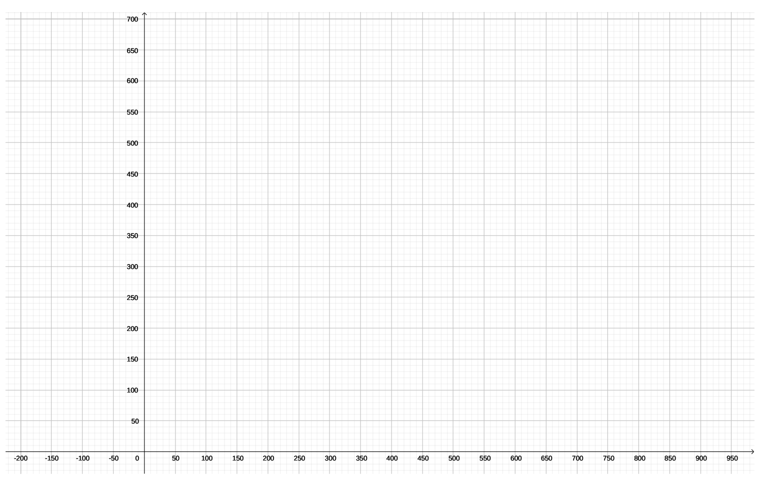
(a) Let us define three decision variables, x_1, x_2, x_3, representing the amounts of the three ingredients to be used. Formulate a linear programming problem in these three variables. **[5]**

(b) Using the fact that $x_1+x_2+x_3 = 1000$, eliminate the x_3 variable and re-formulate the problem. **[5]**

(c) Solve the problem graphically, clearly indicating the feasible area and at least two lines of equal cost. The image shown below is available as an optional template in Blackboard. You may also work on paper if you prefer. **[5]**

(d) State which constraint(s) are binding. **[5]**

(e) For any of the binding constraints, calculate the shadow price. **[5]**



Question 2: Integer programming

A travelling saleswoman is packing a suitcase for a sales trip. She has a large selection of jewellery items for sale. For every item i of jewellery, she can calculate the probability s_i of making a sale on the trip, and the profit p_i that would be achieved on a sale. Every item takes up a certain amount of space a_i in the suitcase, and the suitcase has a limit A on space, measured in cubic centimetres. She needs to decide which items to bring, to maximise expected profit.

(a) Formulate this problem as an integer programming problem. **[5]**

(b) Define the LP relaxation for this problem. What does the solution to the LP relaxation tell us about this problem? (You do not need to solve the LP relaxation). **[10]**

(c) Suppose we are running the *branch and bound* algorithm for this problem. At the first step we discover the solution: $x = (0, 1, 0.57, 0, 0, 1, 0, 1, 0, 1)$ with objective value EUR500. State the new problems that occur as a result (you do not need to solve them). **[5]**

(d) The set of jewellery items includes several necklaces. Every necklace has a matching bracelet. For each necklace, if she brings it, she must bring the matching bracelet also, but not necessarily vice-versa. There is a limit on the number of necklaces which can be brought. Formulate these extra conditions, defining any terms or notation used. **[5]**

Question 3: Choosing algorithms

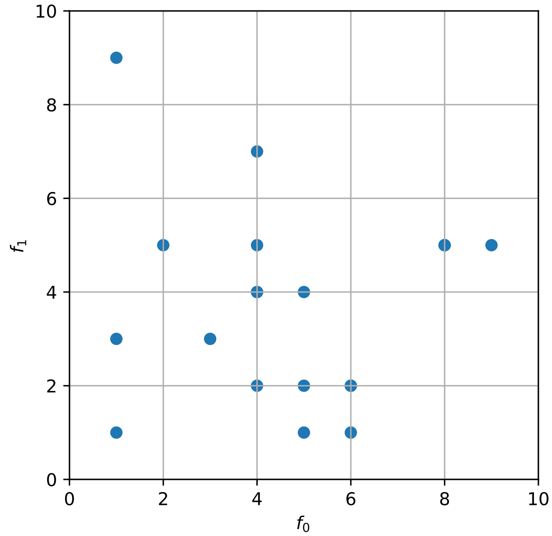
(a) Describe the properties that can cause local optima in the search landscape. **[5]**

(b) Suggest three distinct mutation operators suitable for a real vector representation with no constraints, and in each case describe how the exploration-exploitation balance can be controlled. **[5]**

(c) Both constructive heuristics and gradient descent assume a non-black box problem. In your own words, distinguish between their assumptions. **[5]**

(d) With the aid of a diagram, describe how "forces" act on graph nodes in the *graph layout* problem. Why does this help, relative to (e.g.) a metaheuristic? What properties of a problem allow such an approach? **[10]**

Question 4: Multi-objective optimisation



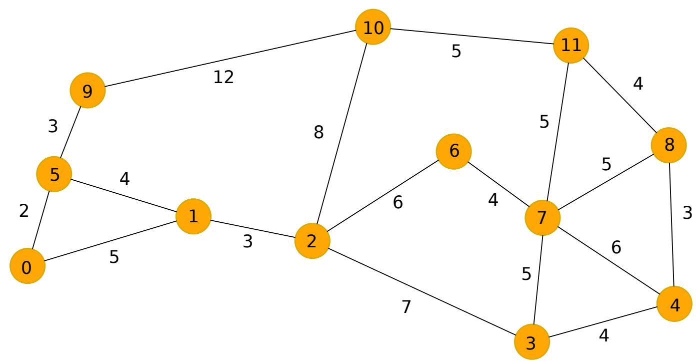
(a) Given the population shown above, where both objectives are to be **minimised**, carry out NSGA2 selection to select 8 offspring, and explain the process. **[10]**

(b) Can local optima arise in multi-objective optimisation problems? Explain in your own words, with the aid of a diagram if necessary. **[10]**

(c) Could we use a hill-climbing method for multi-objective optimisation? Explain in your own words why or why not. **[5]**

Question 5: Constructive heuristics and metaheuristics

On a small island there are 12 villages. There is a police station in Village 0. Every day the policeman must leave the police station and visit all other villages by bicycle, and return to the police station, using only the road network with travel times as shown below. To avoid any appearance of favouritism he must visit each village (other than Village 0) only once.



(a) Describe a deterministic greedy heuristic that starts at village 0, and demonstrate it on the given data. **[5]**

(b) Suppose instead the police station is at village 2. Demonstrate the same algorithm as in part (a), starting at village 2. What happens? **[5]**

(c) Describe a non-deterministic version of the algorithm. What are the potential advantages? **[5]**

(d) In your own words, describe a metaheuristic which could be used for this type of problem. **[5]**

(e) Some problems can be solved using either a Metaheuristic approach, or a Constructive Heuristic approach. What are the pros and cons? **[5]**