Cardiff School of Computer Science and Informatics

Coursework Assessment Pro-forma

Module Code : CM3109
Module Title : Combinatorial Optimisation
Lecturer: Richard Booth
Assessment Title : Problem Sheet
Assessment Number : 2
Date Set: 8/12/2022
Submission Date and Time : by 15/12/2022 at 9:30am
Feedback return date : 22/1/2023

If you have been granted an extension for Extenuating Circumstances, then the submission deadline and return date will be 1 week later than that stated above.

If you have been granted a deferral for Extenuating Circumstances, then you will be assessed in the summer resit period (assuming all other constraints are met).

This assignment is worth 70% of the total marks available for this module. If coursework is submitted late (and where there are no extenuating circumstances):

1. If the assessment is submitted no later than 24 hours after the deadline, the mark for the assessment will be capped at the minimum pass mark;
2. If the assessment is submitted more than 24 hours after the deadline, a mark of 0 will be given for the assessment.

Extensions to the coursework submission date can ***only*** be requested using the Extenuating Circumstances procedure. Only students with ***approved*** extenuating circumstances may use the extenuating circumstances submission deadline. Any coursework submitted after the initial submission deadline without *approved* extenuating circumstances will be treated as late.

More information on the extenuating circumstances procedure can be found on the Intranet: https://intranet.cardiff.ac.uk/students/study/exams-and-assessment/extenuating-circumstances

By submitting this assignment you are accepting the terms of the following declaration:

I hereby declare that my submission (or my contribution to it in the case of group submissions) is all my own work, that it has not previously been submitted for assessment and that I have not knowingly allowed it to be copied by another student. I understand that deceiving or attempting to deceive examiners by passing off the work of another writer, as one's own is plagiarism. I also understand that plagiarising another's work or knowingly allowing another student to plagiarise from my work is against the University regulations and that doing so will result in loss of marks and possible disciplinary proceedings[1].

[1] https://intranet.cardiff.ac.uk/students/study/exams-and-assessment/academic-integrity/cheating-and-academic-misconduct

Assignment

The 4 questions are described in detail in the attachment.

Learning Outcomes Assessed

- Explain the philosophy behind the term combinatorial optimisation.
- Understand the limitations of traditional methods, or when to apply appropriate nontraditional techniques.
- Understand current state of research regarding some classic optimisation problems, such as the Travelling Salesperson Problem.

Criteria for assessment

Credit will be awarded against the following criteria.

- [Correctness] Do the answers correctly address the requirements of each task?
- [Clarity] Are explanations and summaries easily understandable?
- [Understanding of concepts] Do the answers show an understanding of basic optimisation concepts?

Indication of level of attainment:

1st, 70-100%: rigorous, methodical, required justifications are fully convincing, content meets all requirements of the work, very few errors/omissions.

2.1, 60-69%: competent, reasoned, coherent, required justifications are clear with few gaps, few errors/omissions.

2.2, 50-59%: satisfactory, content meets many of the required elements, some errors/omissions.

3rd, 40-49%: Passable, justifications are understandable but with gaps, errors/omissions.

Fail, 0-39%: not passable, evident weaknesses, gaps in content, evident errors/omissions.

Feedback and suggestion for future learning

Feedback on your coursework will address the above criteria. Feedback and marks will be returned on 22/1/2023 via Learning Central.

Feedback from this assignment will be useful for CM3203.

Submission Instructions

You are required to answer 4 multi-part questions on different approaches to combinatorial optimisation, as described in detail in the attachment. The answers should be submitted as a single pdf file.

Description		Type	Name
Answers to all question parts	**Compulsory**	One PDF (.pdf) file	[student number].pdf

Any deviation from the submission instructions above (including the number and types of files submitted) may result in a mark of zero for the assessment or question part.

<u>Staff reserve the right to invite students to a meeting to discuss coursework submissions</u>

Support for assessment

Questions about the assessment can be asked on the Discussion Board on the module's Learning Central pages, or via email to the module team.

CM3109 Problem Sheet
Autumn semester 2022

ANSWER ALL PARTS OF ALL 4 QUESTIONS. Each question is worth 20 marks and the number of marks available for each question part is indicated.

Question 1

(a) Consider the following problem:

> *A wood products company makes two kinds of product: a chopping board and a knife holder. Each chopping board requires 1.4 minutes of cutting time and 5 minutes of gluing time, while each knife holder requires 0.8 minutes of cutting time and 13 minutes of gluing time. During a week, 56 minutes are available for cutting and 650 minutes are available for gluing. If the profit from chopping boards is £2 per unit and the profit from knife holders is £6 per unit, how many of each type of product should be produced in a week to maximise profit?*

Write down this problem in the form of a linear program in canonical form, indicating clearly the decision variables, objective function (and whether it is to maximised or minimised) and constraints. [6]

(b) Consider the following linear program, with decision variables x_1, x_2:

$$\text{maximise } 8x_1 + 30x_2$$

subject to the constraints:

$$\begin{aligned} -5x_1 + 10x_2 &\leq 50 \\ 14x_1 + 3x_2 &\leq 90 \\ x_1, x_2 &\geq 0 \end{aligned}$$

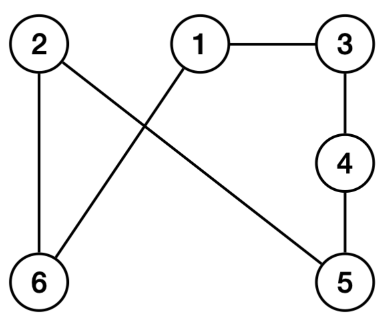
(i) Write down the initial simplex tableau for this problem. [4]

(ii) By starting from the tableau in part (i) above and completing the tableau method for the simplex algorithm, find an optimal solution for this program and determine the maximum attained value of the objective function. Show all intermediate tableaux in your working, and after each iteration of the main loop, sketch a graph indicating the position of the current basic feasible solution in the solution space. (Handwritten sketch, or screenshot using Desmos or another graphing software both acceptable) [10]

Question 2

(a) Explain why linear programming problems can be said to fall under the class of *combinatorial* optimisation problems. [4]

(b) Consider the following tour T for the set of nodes $\{1, 2, 3, 4, 5, 6\}$ in an instance of the Travelling Salesperson Problem:



Write down a tour S for $\{1, 2, 3, 4, 5, 6\}$ such that $S \neq T$ and S is a 2-change neighbour of T. [1]

(c) Consider the following statement:

Integer linear programming problems are, in general, computationally intractable.

Is this statement true or false? You must justify your answer. [3]

(d) Consider the following binary integer linear program:

$$\text{maximise } 7x_1 - 3x_2 - 15x_3$$

subject to the constraints:

$$\begin{array}{rcl} x_1 + 3x_2 - 4x_3 & \leq & 37 \\ -x_1 + x_2 & \leq & 0 \\ x_1, x_2, x_3 & \text{are} & \text{binary} \end{array}$$

Suppose we want to apply the branch and bound algorithm to solve this program. Choose a branching variable (stating it clearly) and write down the LP relaxations of the corresponding two sub-problems that we obtain. [2]

(e) Suppose we are given a weighted tournament T with n participants $\{1, 2, \ldots, n\}$ and with tournament matrix $[a_{ij}]$ (where $a_{ij} = 0$ if player i did not defeat j in the tournament, and a_{ij} is the margin of victory of i over j otherwise), and we wish to find a Kemeny ranking for T, i.e., a ranking $R = [i_1, i_2, \ldots, i_n]$ that minimises the Kemeny score:

$$c(R, T) = \sum \{a_{ij} \mid R \text{ disagrees with } T \text{ on } (i, j)\}.$$

Show that the problem of finding a Kemeny ranking for T can be posed as a binary integer linear programming problem by writing down appropriate choices for *(i)* the decision variables (and what they stand for), *(ii)* the objective function (and whether it is to be maximised or minimised), and *(iii)* the constraints. Explain your choices in each case.

[*Hint:* For the constraints, note that any ranking must satisfy at least the following properties: (1) for any three participants, if one participant is ranked above a second, and the second is ranked above a third, then the first is ranked above the third, (2) for any two distinct participants, one of them must be ranked above the other.] [10]

Question 3

(a) Suppose we are running a genetic algorithm with population size 4, and let the current population consist of the following four chromosomes, with associated fitness:

label	chromosome	fitness $f(x)$
A	101011	4
B	110111	5
C	100001	2
D	010110	3

Assuming we use fitness-proportionate selection to choose the parents for the next population:

(i) What is the probability that B will be chosen for selection first? Briefly justify your answer. [2]

(ii) What is the expected number of times that C will be chosen in the next population? Briefly justify your answer. [2]

(iii) Suppose we decided to implement fitness-proportionate selection via Stochastic Universal Sampling. How many times would A be chosen? Explain your answer. [3]

(b) In which circumstances, and for what reason, would you want to use Stochastic Universal Sampling to implement fitness-proportionate selection in a genetic algorithm? [3]

(c) Suppose the following two chromosomes have been selected to be parents during a run of a genetic algorithm:

$$A = 1000111, \qquad B = 0011010$$

Give exactly one example of a pair of offspring chromosomes that can be obtained from A and B via single-point crossover. Justify your answer. [1]

(d) Consider the following set of strings encoding solutions for a genetic algorithm:

$$\{11101, 11100, 10000, 10100, 11001, 10001, 11000, 10101\}$$

Write down a schema that represents this set of strings, and give its order and defining length. [3]

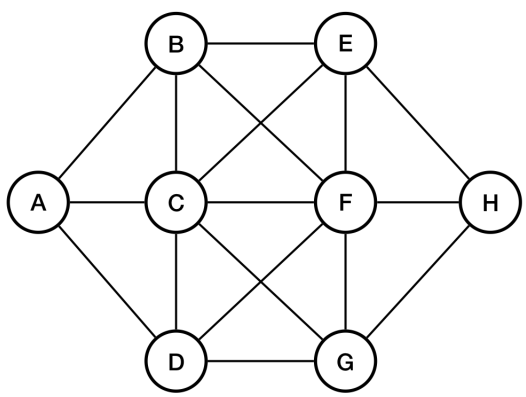
(e) This question is about comparing the different approaches that simulated annealing, genetic algorithms and tabu search take to solving optimisation problems.

(i) Clearly describe exactly *two* aspects that simulated annealing and genetic algorithms have in common, that are *not* shared by tabu search. [4]

(ii) Clearly describe exactly *one* aspect that simulated annealing and tabu search have in common, that is *not* shared by genetic algorithms. [2]

Question 4

Consider a constraint satisfaction problem with eight variables $\{A, B, C, D, E, F, G, H\}$ which are connected by the following graph:



Assume a set of binary constraints as follows:

C$_1$ The difference in value of any two variables connected by an edge is not equal to 1.

i.e., we have $|A - B| \neq 1$, $|A - C| \neq 1$, $|C - F| \neq 1$, etc.

(a) Assume the initial domains of the variables are as follows:

Variable	Domain
A	{1}
B	{2,3,4}
C	{3}
D	{4,5,6}
E	{4,5,6}
F	{3}
G	{4,5,6,7,8}
H	{7}

(i) Beginning with a constraint of your choice, reduce the domains as far as possible by running the AC-3 algorithm. You should complete a table similar to the template below: [7]

Constraint	Changes	Added

(ii) Given the initial domains above, determine whether {B,E} is path consistent with C. Justify your answer. [4]

(b) For the remaining part of the question we now assume that, **in addition to** the above set of binary constraints $\mathbf{C}_1$, we add the following further set of binary constraints.

$\mathbf{C}_2$ All variables must receive different values.

i.e., for each pair of distinct variables $V_1, V_2 \in \{A, B, C, D, E, F, G, H\}$ there is a constraint $V_1 \neq V_2$. We also assume that the initial domains of all the variables are $\{1, 2, 3, 4, 5, 6, 7, 8\}$.

Assume we are applying the backtracking algorithm to solve this CSP, and assume the algorithm has built the partial assignment $C = 1$, $F = 8$, $B = 3$ and $E = 5$ as in the following table:

	A	B	C	D	E	F	G	H
Initial domains	1,2,3,4, 5,6,7,8	1,2,3,4, 5,6,7,8	1,2,3,4, 5,6,7,8	1,2,3,4, 5,6,7,8	1,2,3,4, 5,6,7,8	1,2,3,4, 5,6,7,8	1,2,3,4, 5,6,7,8	1,2,3,4, 5,6,7,8
C = 1	3,4,5, 6,7,8	3,4,5, 6,7,8	1	3,4,5, 6,7,8	3,4,5, 6,7,8	3,4,5, 6,7,8	3,4,5, 6,7,8	2,3,4,5, 6,7,8
F = 8	3,4,5,6,7	3,4,5,6	1	3,4,5,6	3,4,5,6	8	3,4,5,6	2,3,4,5,6
B = 3	5,6,7	3	1	4,5,6	5,6	8	4,5,6	2,4,5,6
E = 5	6,7	3	1	4,6	5	8	4,6	2

(*i*) Find a solution to the problem by completing the table using the backtracking algorithm. You should use the Minimum Remaining Values (MRV) heuristic, or the degree heuristic if MRV leads to a tie, as well as the least-constraining value heuristic and forward checking. Explain your choices at each step. [9]