CS202: Design and Analysis of Algorithms

Assignment 3

Instruction: Submit a Word or PDF document and keep your answer within 1 page for each of the following questions.

[Question 1] Graph Algorithms

This question is an application of the shortest path problem.

Tom Bruce is an FBI agent who has just stolen the State Jewel from the Kremlin Palace in Moscow, and needs to travel safely to a location where a helicopter awaits to get him out of the city. Unfortunately, each street in Moscow may have police on patrol who may arrest him. The city street network is defined by a graph (V,E), where V is the set of street intersections and E is the street joining two intersections. Let p(u,v) denote the probability that Tom may be caught by the patrol on street (u,v). For simplicity, assume each street can be traversed in both directions with same level of risk. Furthermore, assume that there is at most one street between two intersections, and that the risks are independent.

Suppose that Agent Tom has complete information on the network topology of the city, as well as intelligence information on the street patrols.

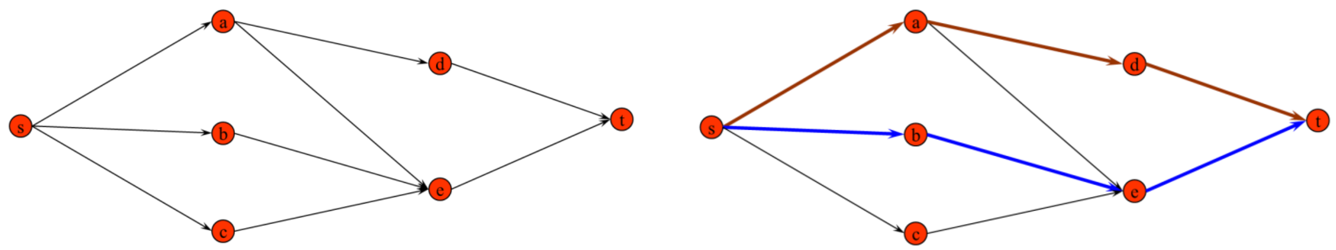
a) (3 marks) Explain how Agent Tom can modify the Dijkstra's algorithm to find the safest path to the helicopter (i.e. the path with the highest probability that he will not be caught). For this purpose, the most straightforward solution is to reproduce the algorithm given in the class slides, and highlight the modifications.

b) (3 marks) Now suppose the helicopter must leave by a certain timing *Tmax*, and the safest route may miss this timing from the time Agent Tom starts his journey. The goal is to find the safest route that must satisfy the timing constraint. For simplicity, assume that the start time is 0, and each edge has a weight representing the travel time. Discuss how the algorithm can be extended to accommodate this constraint.

[Question 2] Network flows and Linear Programming

Let G=(V,E) be an unweighted directed acyclic graph with a designated source s and destination t. We know how to find a path between s and t. Now consider an optimization problem where our task is to find a maximum number of disjoint paths from *s* to *t*. Two paths are said to be disjoint if they do not share a common vertex nor edge (except at *s* and *t*).

For example, given the graph on the left, a solution is given on the right:



(a) (3 marks) Show how the above problem can be converted to a maximum flow problem. (You need to discuss how to construct the flow network and how the resulting flow can be used to derive the set of disjoint paths.)

(b) (2 marks) Show how the above problem can be formulated as a linear program.

(c) (1 mark) From the asymptotic complexity point of view, is it more efficient to apply the Ford-Fulkerson algorithm to solve the resulting max-flow problem given in (a), or to solve the linear program in (b) using the Simplex algorithm? Explain.