Wong Zhing Jieh Jack 201910656

Handin 2

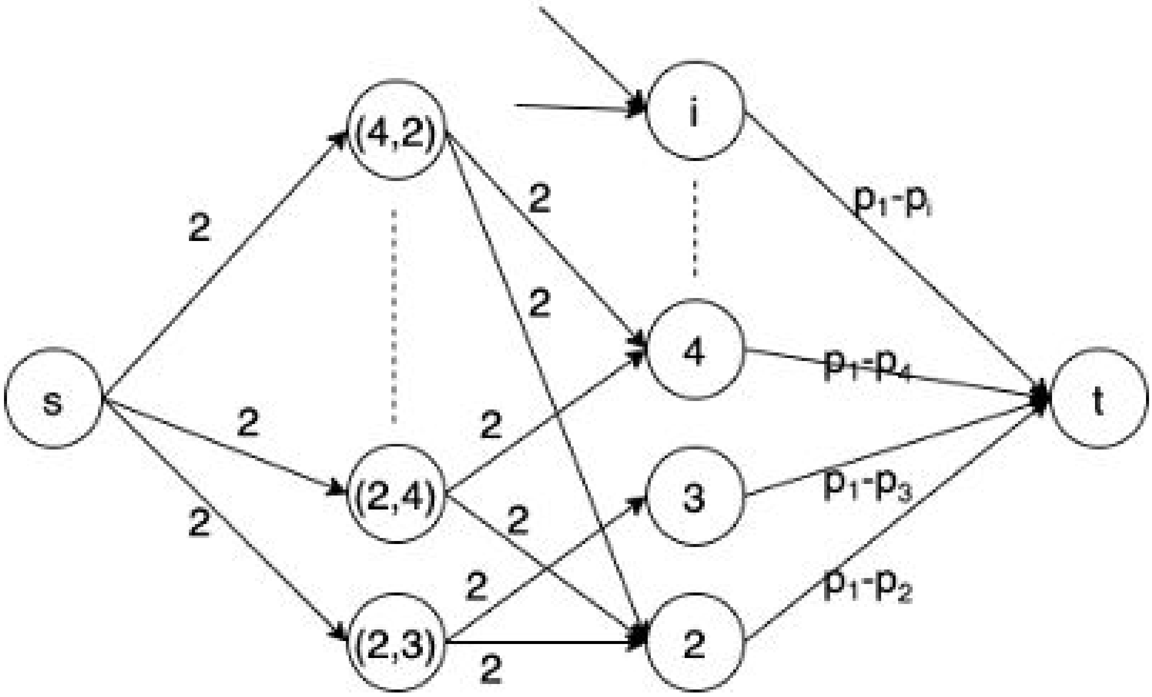

Figure 1

1. We can formulate the problem as a maximum (*s,t*)-flow problem as follows.

   We build a network *D = (N, A)* where *N* = {*s,t*} {(2,3),(2,4),…,(n,n-1)} {2,3,…,n}. A tuple represents a match between two teams. For example, (2,3) means team 2 plays against team 3. A number node represents the corresponding team. For example, node 2 represents team 2. Team 1 is excluded from the network *D*, as the goal of constructing *D* is to maximize the number of points attainable by other teams and see if any other team has higher points than team 1, assuming team 1 wins the remaining matches.

   The arcs of the network are as follows: From *s* we have an arc to every tuple (*a,b*), given an upper bound of 2 units. From every tuple we have an arc to the number node representing the team. For example, (2,3) will have an arc to node 2 and another arc to node 3. Each of these arcs is given an upper bound of 2 units. A flow to a number node means score (which is the flow) is awarded to the corresponding team. Since the conservation of flow on every node must be conserved, the maximum total outflows of a tuple must be 2 units, meaning either a team wins (2 points for the team) or both teams draw (1 point each). These are the only possible flow values when running max-flow algorithm such as Ford-Fulkerson algorithm since all arc capacities are integers (Integrality theorem).
   Finally, from every number node we have an arc to *t*, upper bounded by p_1-p_i where *i* is node *i.* This means team *i* is allowed to win at most p_1-p_i points.

   It may be worthwhile to point out that the resulting network is a 4-partite graph.

Handin 2

A feasible solution for this max-flow problem means that there exists some match(es) such that their scores are no greater than team 1's score. However, this does not guarantee that all matches are played. To ensure all listed matches are played, we need to ensure all out-going flows from *s* are 2 units. **Hence, we can say that if there exists a feasible solution such that all out-going flows from *s* are 2 units, then team 1 can win**. Else it cannot win.

2. Based on my max-flow network formulation, three-point rule cannot be easily formulated my formulated allows two out-going flows from a tuple to have flow values of 1 and 2 respectively, without violating conservation of flow. This means that according to the formulated network, it is possible for two competing teams to be awarded 1 point and 2 points respectively, which is not allowed in the soccer game. One possible flow is illustrated in the figure below (only shows partial graph with teams 2 and 3).

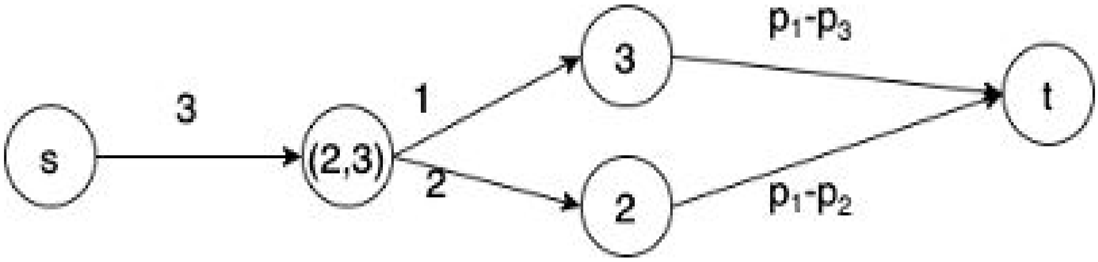

Figure 2

3. The three-point rule game can be modelled as an integer linear programming as such:
   Maximize $z = p_1 - p_i$
   s.t
   X_i = 1 if team a_i wins,
   Y_i = 1 if team b_i wins,
   Z_i = 1 if both team a_i and b_i draw,

   $X_i + Y_i + Z_i = 1 \ \forall i \in \mathbb{Z}^+$, since only one event must happen.

   $(3 \sum_{j=1,\, a_j=i} X_j + \sum_{j=1,\, a_j=i} Z_j) + (3 \sum_{j=1,\, b_j=i} Y_j + \sum_{j=1,\, b_j=i} Z_j) + p_i \leq p_1$

   If there is **no feasible** solution, then team 1 cannot win. Else, team 1 can win.