Advanced Algorithms - Homework 5

Question 1

**Players possible path**

Player 1:

Player 2:

Player 3:

**Profiles**

**Social Optimum** -

1. -
2. -
3. -
4. -
5. -

Question 2

1. In the profile , the cost of the strategy is , and the cost of the strategy is . The strategy is a NE if (players on top path are stable) and (players on lower path are stable) That is .

2. We need to find a function of social optimum then derive it, in order to find the global minimum. (we can make sure that is minimum by checking )

3. There is only one NE and is 🡪

Question 3

W.l.o.g let's say that the algorithm terminates when Assume towards contradiction that .

Mark and , .

If is moved from to . The condition of termination is that for every that scheduled in , moving form to will not lower.

If then

(because, )

(because what we assumed at the start)

must be the only job on . If not, , .

Hence is the only job on and . But

Contradiction! . Let be the approximation ratio, , .

In the worst case the algorithm terminates with 🡪 , then

W.l.o.g let , and let be the longest job that can be migrated that improves the gap.

Mark the gap as , moving improves the gap so the .

1. If then , then . The new gap , and in every iteration the gap decreases until it stops. will not be migrated again, therefore it is migrated at most once.
2. If then the gap decreased, and won't move again until but then migrating will only increase the gap so it won't move again.
3. If then the algorithm terminates.

Therefore, every job is moved at most once 🡪 algorithm terminates after at most moves.

Question 4

As we discussed in the class the worst case of the algorithm is when the cow walks towards the hole and just before the hole the cow turns around.

Let where

Each distance the cow walked 4 times except the last which the cow walks 2 times and the value.

The competitive ratio is 13

Question 5

Let = maximum match and = maximal match. Let be an edge in , then in there is at most edges touching the vertices of 🡪

We know that for any set the following statement is valid:

🡪 maximal matching is a 2-approximation for the maximum matching.

Online Algorithm

1. Let be empty set (the matching group of edges)
2. Foreach
   1. Foreach (foreach edge that touching and any )

If still exists, Then . Break to step 2.

Correctness

The algorithm finds matching because there are no 2 edges that sharing the same vertex (we are achieving it by "")

2-Competitive

Let be matching, the output of the algorithm. Assume towards contradiction that is not a maximal matching then there is matching that then there is an edge, s.t , this is impossible situation because when the vertex was revealed the algo chose not to add it because either is already deleted or another edge that touching is picked, hence is not matching and is maximal. Straight from 5.1 we can say that

Let and . Given a deterministic algorithm we need to prove that its competitive ratio is at least 2.

1. is revealed, edges are
2. have 2 options
   1. If selects reveal with edge 🡪 can't add into the match but ,
   2. If selects reveal with edge 🡪 can't add into the match but ,

Therefore, we can see that 🡪 is 2-competitive