Planning & Scheduling
Assignment 1

Lavinia Pulcinella
Student ID: i6233926

April 2020

1 Graded Exercise 1.3

Consider the following greedy algorithm for interval scheduling:
Let I be the set of the input intervals $I_1, ..., I_n$. Let S be the solution set, initialized to an empty set i.e.

1. $S = \emptyset$
2. **While there are intervals in I,**
   - **take an interval $I_i \in I$ with the smallest number of overlaps with intervals in I into the solution set S**
   - **remove all intervals that overlap with I_i from I.**

Analyze the approximation ratio of this greedy algorithm on inputs, where the induced interval graph by the intervals has a maximum clique of size 2.

First of all, an interval graph is the resulting graph from a set of intervals where each interval corresponds to a vertex and each edge represents an overlap between two intervals. An induced graph must be chordalized i.e. it cannot contain cycles without chords. Thus, a cycle of 4 or more vertices cannot be considered an interval graph.
To analyse the greedy algorithm above we need to consider an interval graph that has maximum clique of size 2. This implies that, graphically, the induced interval graph would either be a ”line” graph (where the first and last vertex are NOT connected) or a star graph, which both have maximum clique size of 2. That is:

- A ”line” graph is equivalent to the idea that every interval doesn’t overlap with more than 2 other intervals at each point in time.
- A star graph implies that n-1 intervals don’t overlap with more than one other interval while the remaining one may overlap with all of them.

This means that we are not considering cases in which an interval overlaps with say 3 other intervals because the induced interval graph would result in a graph containing at least one clique of size 3.

It follows that in such a setup the greedy algorithm described above is actually always optimum, hence the approximation ratio is 1.

1. In the line graph example, consider the first and last interval i.e. I_1 and I_n, overlapping with I_2 and I_{n-1} respectively i.e. with just one interval each.
   Intervals I_2 and I_{n-1} overlap with two intervals in total each, namely I_2 with I_1 and I_2 while I_{n-1} with I_{n-2} and I_n.
   We know that the greedy algorithm selects intervals with a minimum number of overlaps first. It follows that intervals I_1 and I_n are the first ones selected into the solution set S. Going further in this "shrinking" fashion (i.e. moving towards the centre) the last intervals to be selected would be I_{i-1} and I_i which overlap with each other thus, the greedy algorithm selects either one of the two and considers the choice into the solution set S.
   Imagining the intervals on two levels, the second "shifted" from the first. It is clear how the optimal solution would be containing $\lceil n/2 \rceil$ intervals. Which is exactly what the Greedy algorithm does.
   Hence, $\rho = \frac{|GREEDY|}{|OPT|} = \frac{\lceil n/2 \rceil}{\lceil n/2 \rceil} = 1$

2. In the star graph example, one interval, say I_1 underlines all remaining $n-1$ intervals. This means that I_1 overlaps with all the $n-1$ of them.
   On the other hand, each interval from I_2 through I_n overlaps with just one interval i.e. I_1. It follows that the algorithm selects one of the intervals from I_2 to I_n, say I_2 and removes the overlapping interval I_1 leaving intervals $(I_3, ..., I_n)$ with no overlappings. The greedy algorithm selects $n-1$ intervals as the optimal solution clearly would.
   Hence, $\rho = \frac{|GREEDY|}{|OPT|} = \frac{n-1}{n-1} = 1$