COMP3121 Homework Q2

Arth Sanskar Patel
z5228942

August 6, 2020

1 Answer

Here we have a n x n chess board with bishops occupying k spots. We have to put rooks in such a position that the bishops cannot attack the rooks, i.e no rook can be placed in any diagonal that a bishop has occupied. Further more we have to place rooks so that no two rooks are in the same row or column.

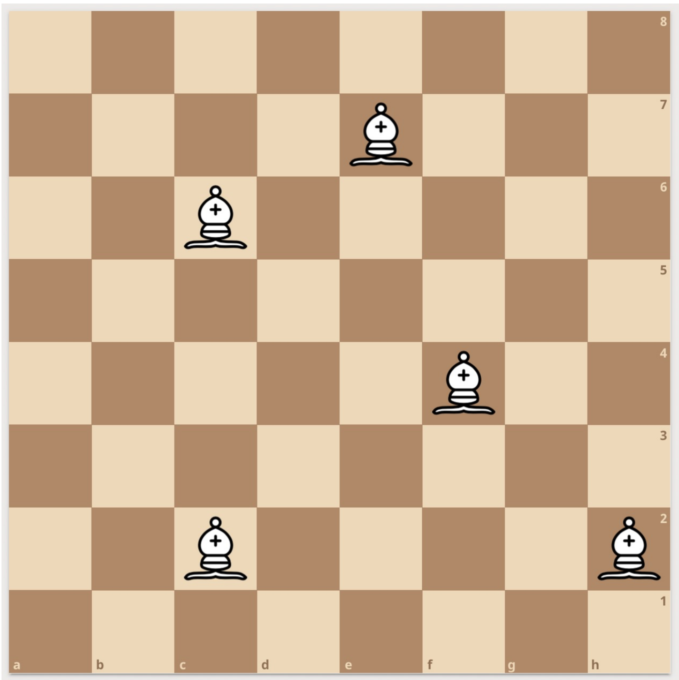

Figure 1: Example Board

This question can be solved using a bipartite graph. On the left side, all the vertices represent the columns and the vertices on the right represent all the rows. The edges in between connect the column number to the row number depending on the cell. For example if we have to get Cell(i,j) we will connect the vertex j on the left to the vertex i on the right. Here we will not connect the edges which represent the cells that are occupied by the bishops or the cells which the bishops can attack (all the cells in all 4 diagonal to a bishop). So now we are only left with edges that represent the cells where we can place our rooks without the bishops attacking. Now to make sure that we don't place more than 1 rook in a given column or a row, lets connect the columns (vertices on the left) to a super source with each edge having a capacity of 1. This will eliminate the question of placing more than 1 rook in 1 column. Similarly connect all the rows (vertices on the right) to a super sink with all the edges having capacity of 1. This will eliminate the question of placing more than 1 rook in 1 row. The edges that connect the columns and rows can be of infinite capacity or 1.

It won't matter since the bottle necking capacity are already there for each column and row connected to their super source and super sink respectively. Now finally run the Ford-Fulkerson algorithm on the bipartite graph to get the max flow. The total flow towards the super sink will be the number of rooks that you can place satisfying all the conditions.

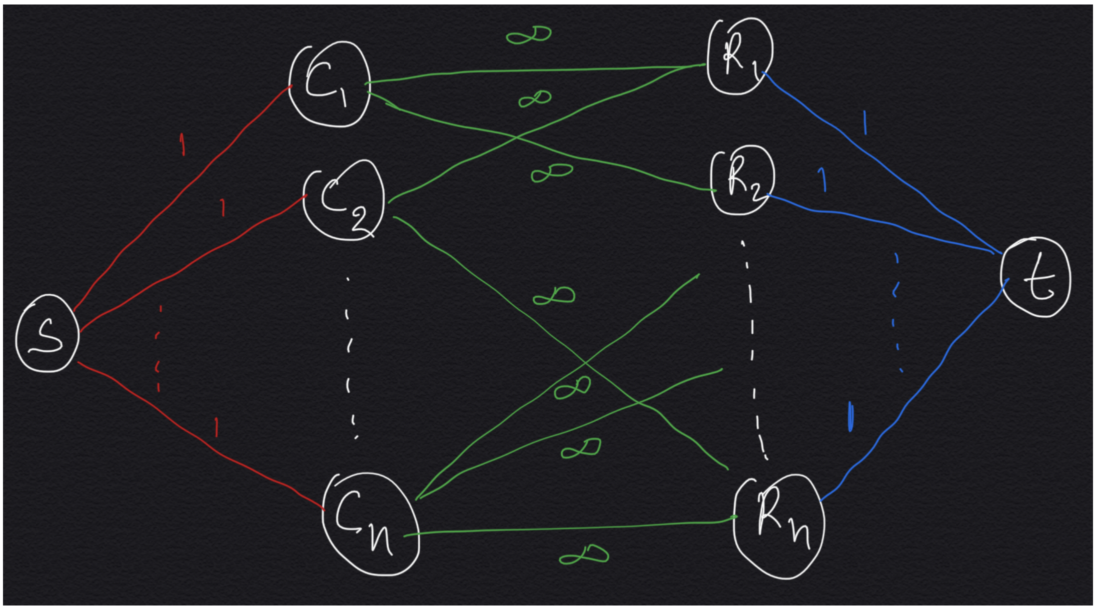

Figure 2: Bipartite Graph

Reference for this question was taken from https://cs.nyu.edu/courses/spring14/CSCI-UA.0480-004/Lecture16.pdf.