| **CS4234: Optimisation Algorithms** | Tutorial 7 |
|---|---|
| Graph-Matching Part 2 | |
| *V1.7: A/Prof Steven Halim* | *October 16, 2023* |

## Discussion Points

**Q1:** Midterm Test quick debrief (about $\approx$ [10..15]m).

**Q2:** Try Section B of `https://www.comp.nus.edu.sg/~stevenha/cs4234/final-S1-2021.pdf`.
Try to complete your understanding of this specific algorithm by solving one of the task in PS4 that can be solved with this algorithm.

Kuhn-Munkres algorithm is one of the possible algorithm that can be used to find the solution of (max) weighted `Max-Cardinality-Bipartite-Matching` (MCBM). In this tracing question, we will check your understanding about this algorithm. You are given a $7 \times 7$ matrix $M$ that describes a (transformed into) complete Bipartite Graph $K_{7,7}$. The left set contains vertex [0/1/../6] (the rows in $M$) and the right set contains vertex [7/8/../13] (the columns in $M$). A non-negative value $M[i][j]$ signifies that if we match vertex $i$ with vertex $j$, we will obtain $M[i][j]$ profit. Value $M[i][j] = -\infty$ signifies that there is actually no (directed) edge between vertex $i$ to vertex $j$ in the initial Bipartite Graph (see part 1). Note that although for this specific question, using weight $-1$ for non-existent edge $i \to j$ will not actually change the final answer (see part 4) for this test case. Nowadays, it is safer to set $-\infty$ for such non-existent edges.

| | **7** | **8** | **9** | **10** | **11** | **12** | **13** |
|---|---|---|---|---|---|---|---|
| **0** | -∞ | 8 | 6 | -∞ | -∞ | -∞ | -∞ |
| **1** | 1 | 4 | -∞ | -∞ | -∞ | -∞ | -∞ |
| **2** | 4 | -∞ | 1 | -∞ | -∞ | -∞ | -∞ |
| **3** | -∞ | -∞ | -∞ | 9 | 7 | 8 | 6 |
| **4** | -∞ | -∞ | -∞ | 7 | -∞ | 3 | -∞ |
| **5** | -∞ | -∞ | -∞ | 6 | -∞ | -∞ | -∞ |
| **6** | -∞ | -∞ | -∞ | -∞ | -∞ | -∞ | 0 |

1. Draw the initial Bipartite Graph where edge $i \to j$ with $M[i][j] = -\infty$ does not exist.
2. How many different perfect MCBMs are there in a complete Bipartite Graph $K_{7,7}$?
3. What is the maximum weight of the perfect MCBM? (if you run Kuhn-Munkres algorithm (see below) or via Complete Search (see above), this answer should be correct, i.e., $OPT$; otherwise if you has a bug (or you guess this answer) and your answer is $X$, then your score is max(0, score-for-this-question - abs($OPT$-$X$)).
4. Show the workings of your Kuhn-Munkres algorithm.
   Show the *initial* equality graph and the initial labels of each vertex.
   Show the progress of Kuhn-Munkres algorithm
   Show the *final* equality graph with the selected perfect matching with max total profit.

**Q3:** Solve this graph matching problem: `https://onlinejudge.org/external/108/10888.pdf`.
What type is it? MCBM? weighted MCBM? MCM? weighted MCM?

**Q4:** Mr. Kwan is a postman who wants to deliver a bag of mails to houses in an unnamed Chinese city. The houses in that city are located along the streets and no house is located at a junction. Streets in that city are one-way and Kwan is a very law-abiding citizen and will never walk that one-way street in the opposite direction. Mr. Kwan wants to design a route whereby he starts from a starting junction 0 (his favorite landmark in the city), go through *every* one-way street in that city *at least once*, and returns to that starting junction 0 again to deliver the mails. Kwan knows that his government has designed the city well enough so that there will always be at least one such route. Kwan just want to know what is *the minimum number of streets* that he has to traverse to accomplish his objective. There are $V$ junctions and $E$ streets in that Chinese city (graph $G$).

### Complete Search

To kick start your understanding of this Combinatorial Optimization Problem (COP), please do a complete search on the small instance of this problem shown in Figure 1.

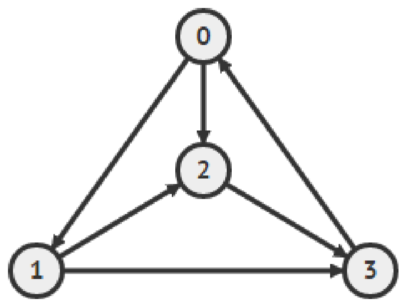


Figure 1: Small instance of this problem.

The minimum number of streets that Mr. Kwan has to traverse to accomplish his objective is optimal route taken by Mr. Kwan is ________ streets/edges.
Just to ensure that you do not put random answer (although the original question only asks for the minimum number of streets), please write down the optimal route here:
$0 \rightarrow$ ________________________________________________ $\rightarrow 0$.

### Complexity Class

Is this COP NP-hard? (circle one)

| YES | NO |
|---|---|

### Special Case

Suppose that the city has one more additional property, i.e., the number of streets that goes into a junction is *always equal* to the number of streets that goes out from that junction, i.e., there is a "flow conservation" property. For example, imagine that there is no junction 2 (and all edges associated with it) in Figure 1 above and all the other 3 junctions $\{0, 1, 2\}$ will each have 1 street that goes in/out of those junctions so Mr Kwan's route will be something like $0 \rightarrow 1 \rightarrow 3 \rightarrow 0$.

Does this additional property simplify the COP?
If yes (Obviously the answer is a yes :) otherwise the questions in this section are meaningless), design a polynomial time algorithm to solve this problem!

What is the time complexity of your polynomial time algorithm?

### General Case

Now solve the general case of this COP using *any algorithm that you have learned in class* and analyze the time complexity of your solution!

# Postscript

In the last three tutorials (skipping Monday, 13 Nov 2023 due to Public Holiday (Deepavali) in-lieu), we will go back to NP-hard optimization problems again, but this time we will use a different search paradigm: Stochastic Local Search.

Also, you can query the TA for any last minute hints for PS4 that will due soon.