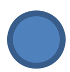
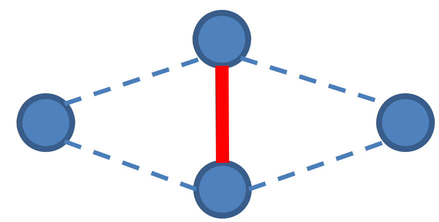
CS3233
Competitive Programming

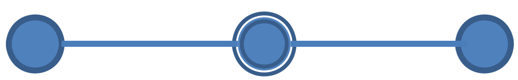
Dr. Steven Halim

Outline

- Mini Contest #6 + Break + Discussion + Admins
- Graph Matching
  - Overview
  - Unweighted MCBM: Max Flow, Augmenting Path, Hopcroft Karp's
    - Relevant Applications: Bipartite Matching (with Capacity), Max Independent Set, Min Vertex Cover, Min Path Cover on DAG
  - Weighted MCBM: Min Cost Max Flow (overview only)
  - Unweighted MCM: Edmonds's Matching
  - Weighted MCM: DP with Bitmask (only for small graph)

Graph Matching

- A matching (marriage) in a graph G (real life) is a subset of edges in G (special relationships) such that no two of which meet at a common vertex (that is, no affair!)
- Thus a.  b.  c. 

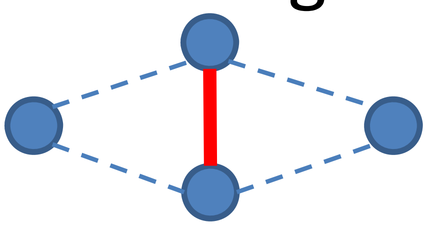
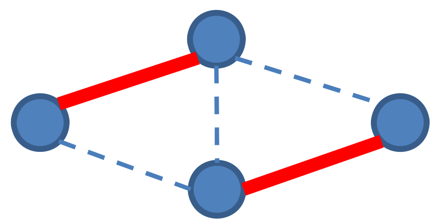
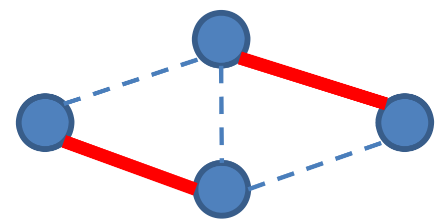
  are matchings (red thick edge),
- But d. 

  is not since there is an overlapping vertex

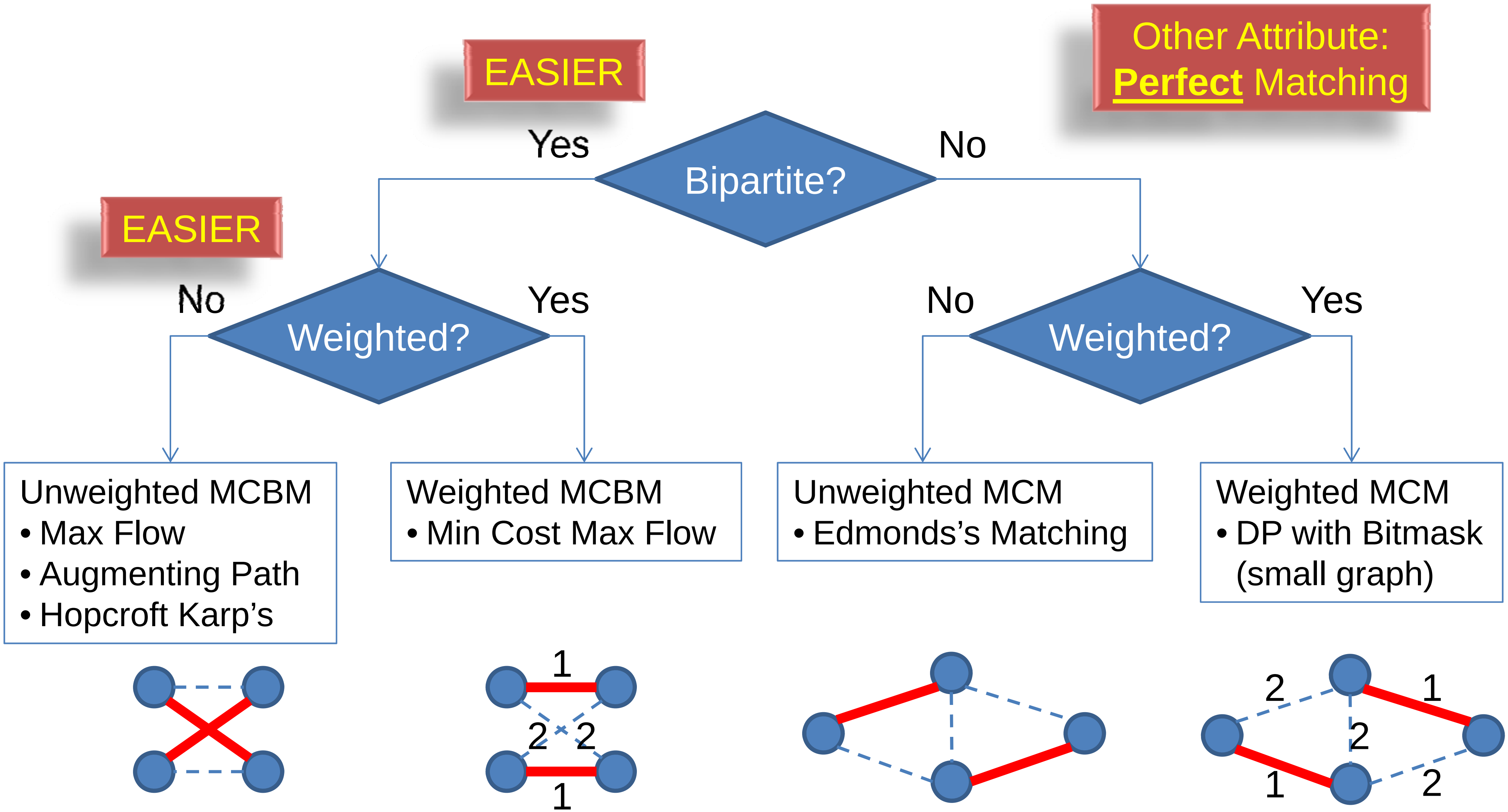
Max Cardinality Matching (MCM)

- Usually, the problem asked in graph matching is the size (cardinality) of a maximum matching
- A maximum matching is a matching that contains the largest possible number of edges

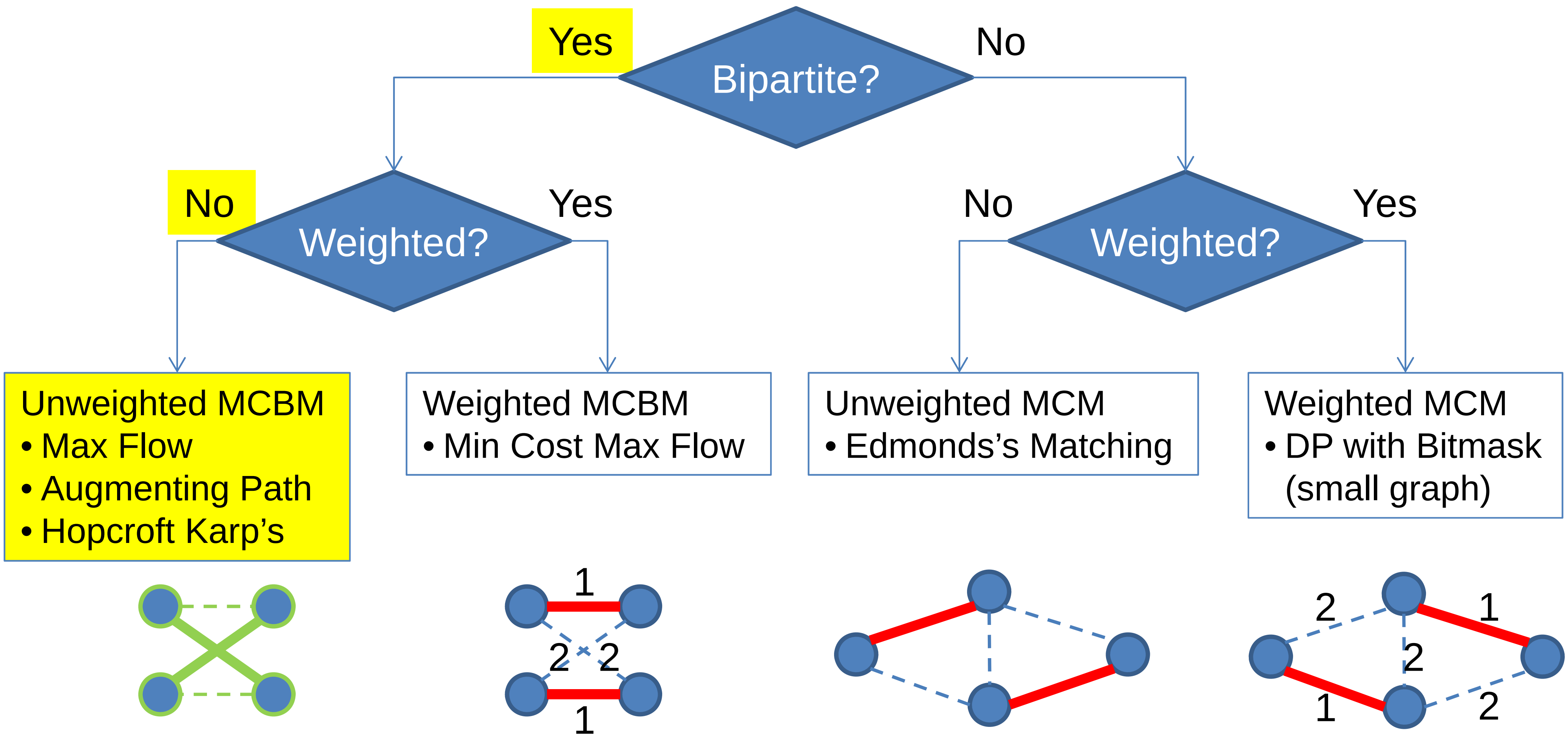
Examples

-  is a maximum matching (0 matching) (no edge to be matched)
-  is also a maximum matching (1 matching) (no other edges to be matched)
- But  is not a maximum matching as we can change it to  or  (2 matchings)

Types of Graph Matching



Types of Graph Matching



Solutions:

Max Flow

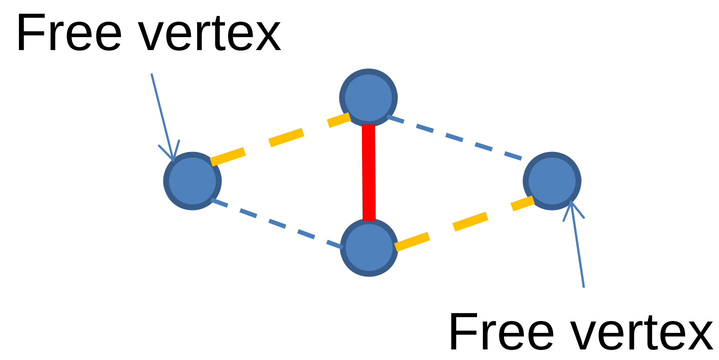
Augmenting Path Algorithm

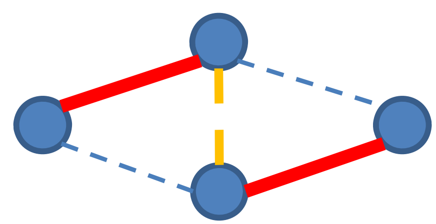
Hopcroft Karp's Algorithm

UNWEIGHTED MCBM

Augmenting Path

- In this graph, the path colored **orange(unmatched)-red(matched)-orange** is an augmenting path
- We can flip the edge status to **red-orange-red** and the number of edges in the matching set increases by 1





MC Bipartite Matching (MCBM)

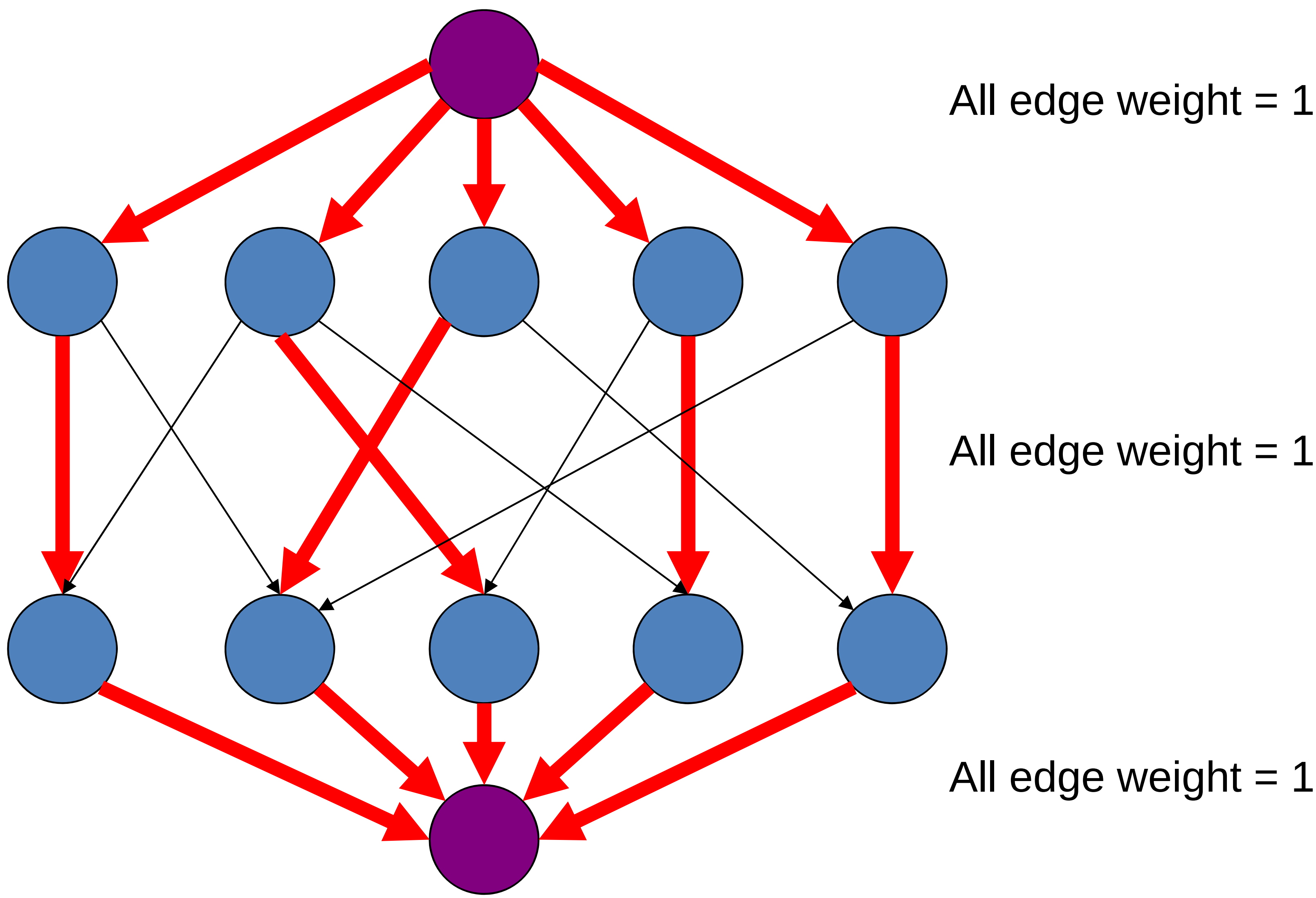
- A Bipartite graph is a graph whose vertices can be divided into two disjoint sets X and Y such that every edge can only connect a vertex in X to one in Y
- Matching in this kind of graph is **a lot easier** than matching in general graph

Finding MCBM by reducing this problem into

MAX FLOW

http://www.comp.nus.edu.sg/~stevenha/visualization/matching.html

Max Flow Solution for MCBM



Time Complexity: Depends on the chosen Max Flow algorithm

Finding MCBM via

AUGMENTING PATH ALGORITHM

Augmenting Path Algorithm

- Lemma (Claude Berge 1957):

A matching M in G is maximum
iff there is no more augmenting path in G

- Augmenting Path Algorithm is a simple $O(V*(V+E)) = O(V^2 + VE) \sim= O(VE)$ implementation of that lemma

The Code (1) ☺

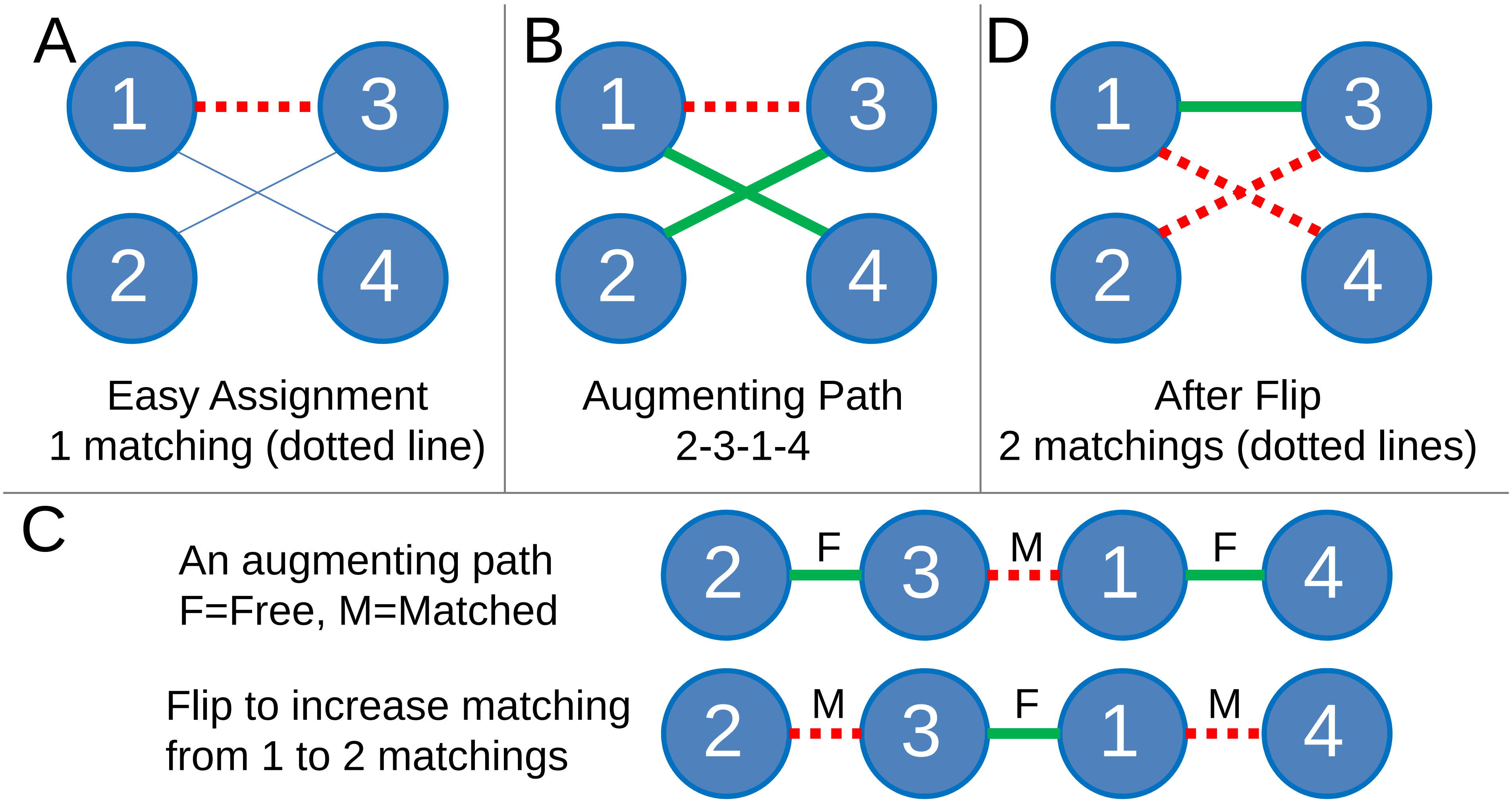
```
vi match, vis; // global variables

int Aug(int l) { // return 1 if ∃ an augmenting path
  if (vis[l]) return 0; // return 0 otherwise
  vis[l] = 1;
  for (int j = 0; j < (int)AdjList[l].size(); j++) {
    int r = AdjList[l][j].first;
    if (match[r] == -1 || Aug(match[r])) {
      match[r] = l;
      return 1; // found 1 matching
  } }
  return 0; // no matching
}
```

The Code (2) ☺

```
// in int main(), build the bipartite graph
// only directed edge from left set to right set is needed

  int MCBM = 0;
  match.assign(V, -1);

  for (int l = 0; l < Vleft; l++) {
    vis.assign(Vleft, 0);
    MCBM += Aug(l);
  }

  printf("Found %d matchings\n", MCBM);
```

http://www.comp.nus.edu.sg/~stevenha/visualization/matching.html

Augmenting Path Algorithm



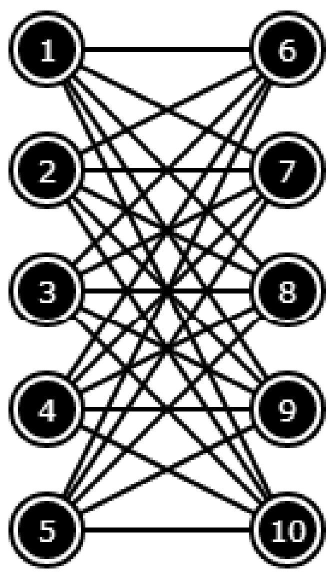
Finding MCBM via

HOPCROFT KARP'S ALGORITHM

http://www.comp.nus.edu.sg/~stevenha/visualization/matching.html

An Extreme Test Case…

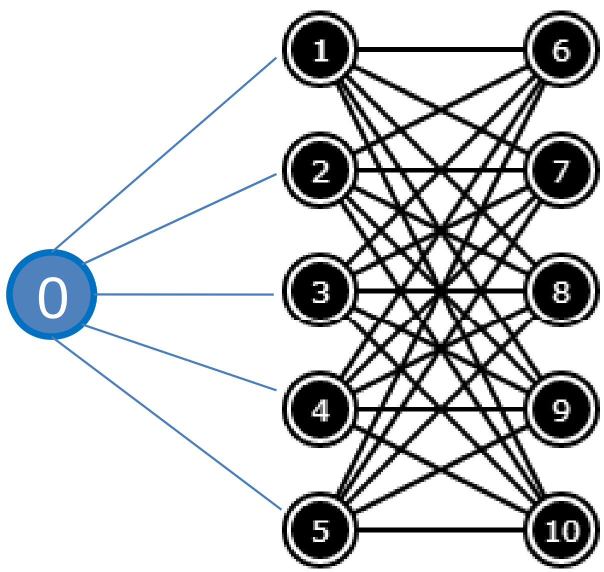
- A Complete Bipartite Graph $K_{n,m}$, **V=n+m** & **E = n*m**
- Augmenting Path algorithm → O((n+m)*n*m)
  - If m = n, we have an $O(n^3)$ solution, OK for n ≤ 200
- Example with n = m = 5



http://www.comp.nus.edu.sg/~stevenha/visualization/matching.html

Hopcroft Karp's Algorithm (1973)

- Key Idea:
  - Find the shortest augmenting paths first from all free vertices (with BFS)
  - Run similar algorithm as the Augmenting Path Algorithm earlier (DFS), but now using this BFS information



http://www.comp.nus.edu.sg/~stevenha/visualization/matching.html

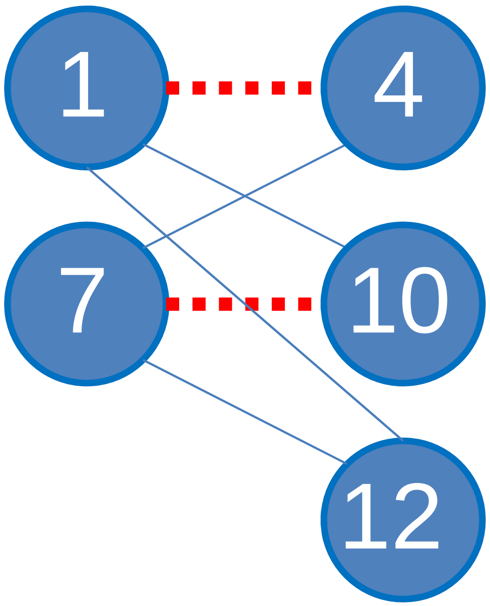
Hopcroft Karp's Algorithm (1973)

- Hopcroft Karp's runs in $O(E\sqrt{V})$, proof omitted
  - For the extreme test case in previous slide, this is O(n*m*sqrt(n+m))
  - With m = n, this is about $O(n^{5/2})$, OK for $n \leq 600$
- Question: Is this algorithm **must be learned** in order to do well in programming contest?

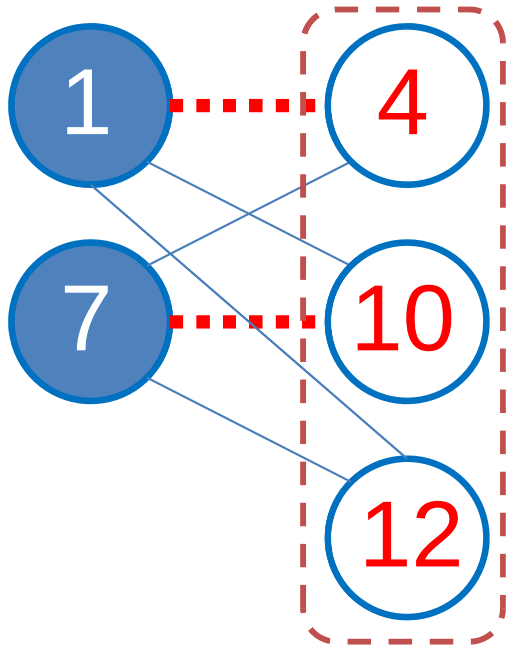
EXAMPLES OF MCBM IN PROGRAMMING CONTESTS

Popular Variants

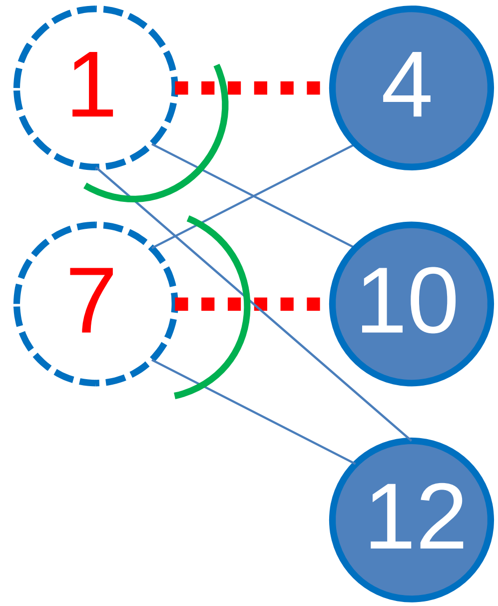
Max Independent Set / Min Vertex Cover



A. MCBM



B. Max Independent Set
MIS: V – MCBM

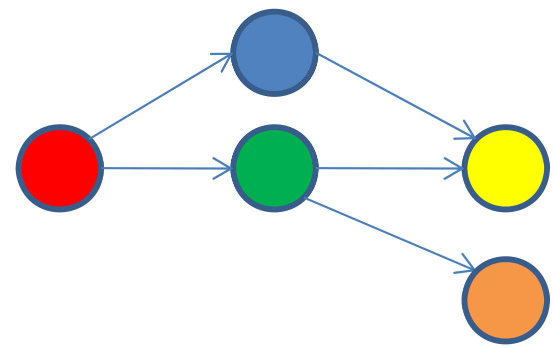


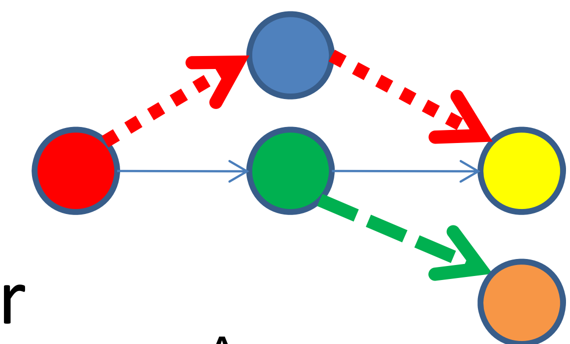
C. Min Vertex Cover
MVC: MCBM

(König's theorem)

Min Path Cover
in DAG

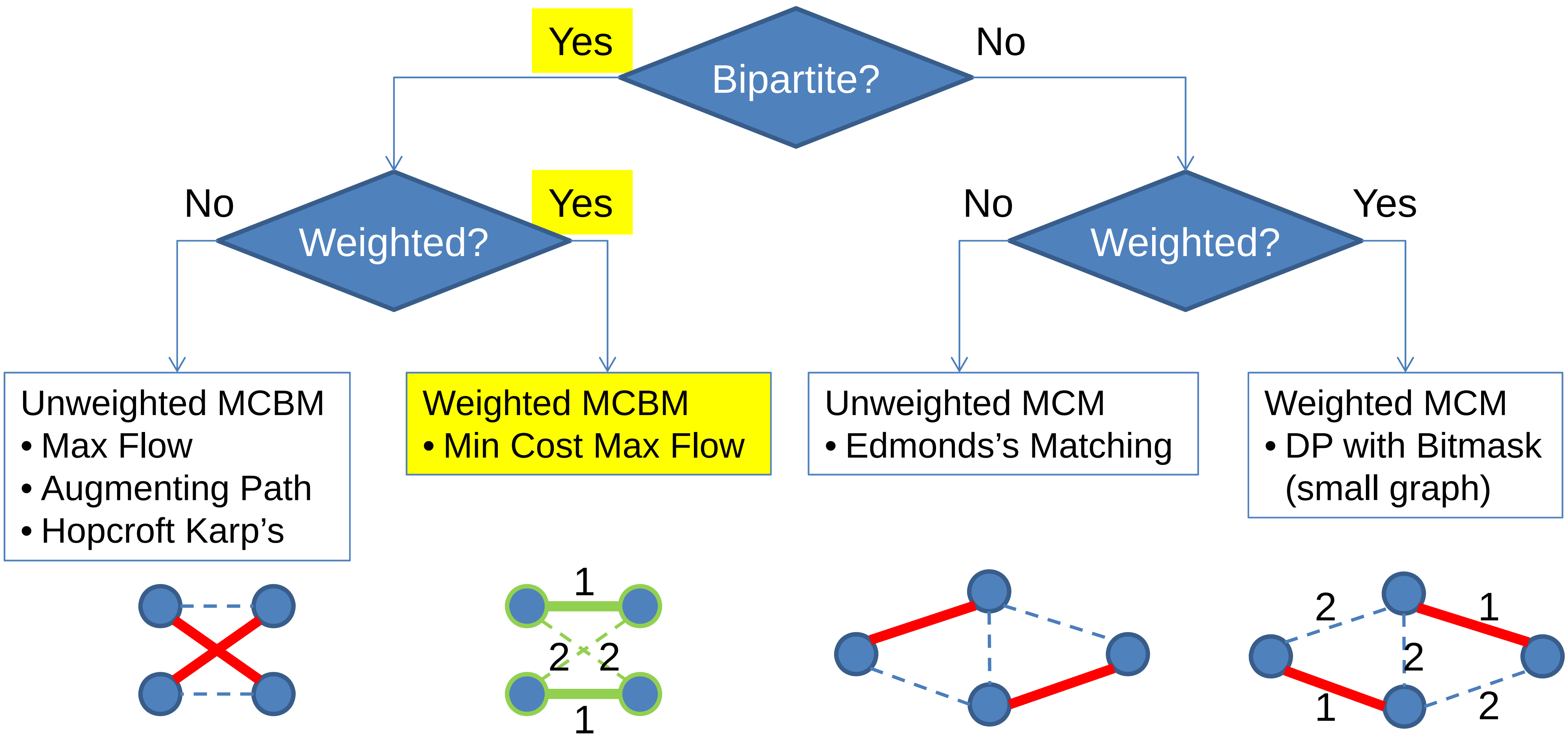
- Illustration:
  - Imagine that vertices are passengers, and draw edge between two vertices if a single taxi can satisfy the demand of both passengers on time…
  - What is the minimum number of taxis that must be deployed to serve all passengers?
- This problem is called: Min Path Cover
  - Set of directed paths s.t. every vertex in the graph belong to at least one path (including path of length 0, i.e. a single vertex)





Answer:
2 Taxis!

Types of Graph Matching



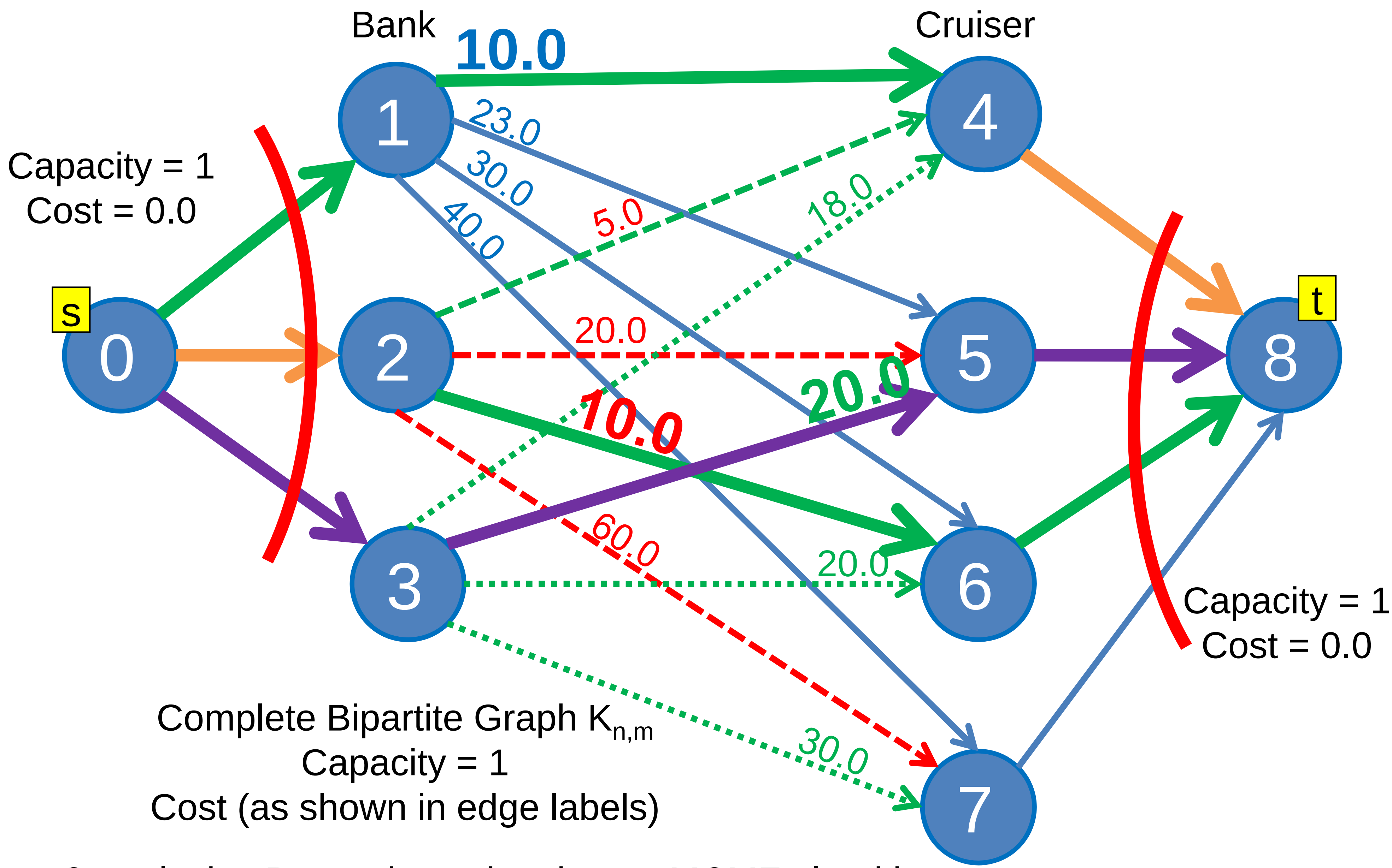
Solution:

Min Cost Max Flow (Overview Only)

WEIGHTED MCBM

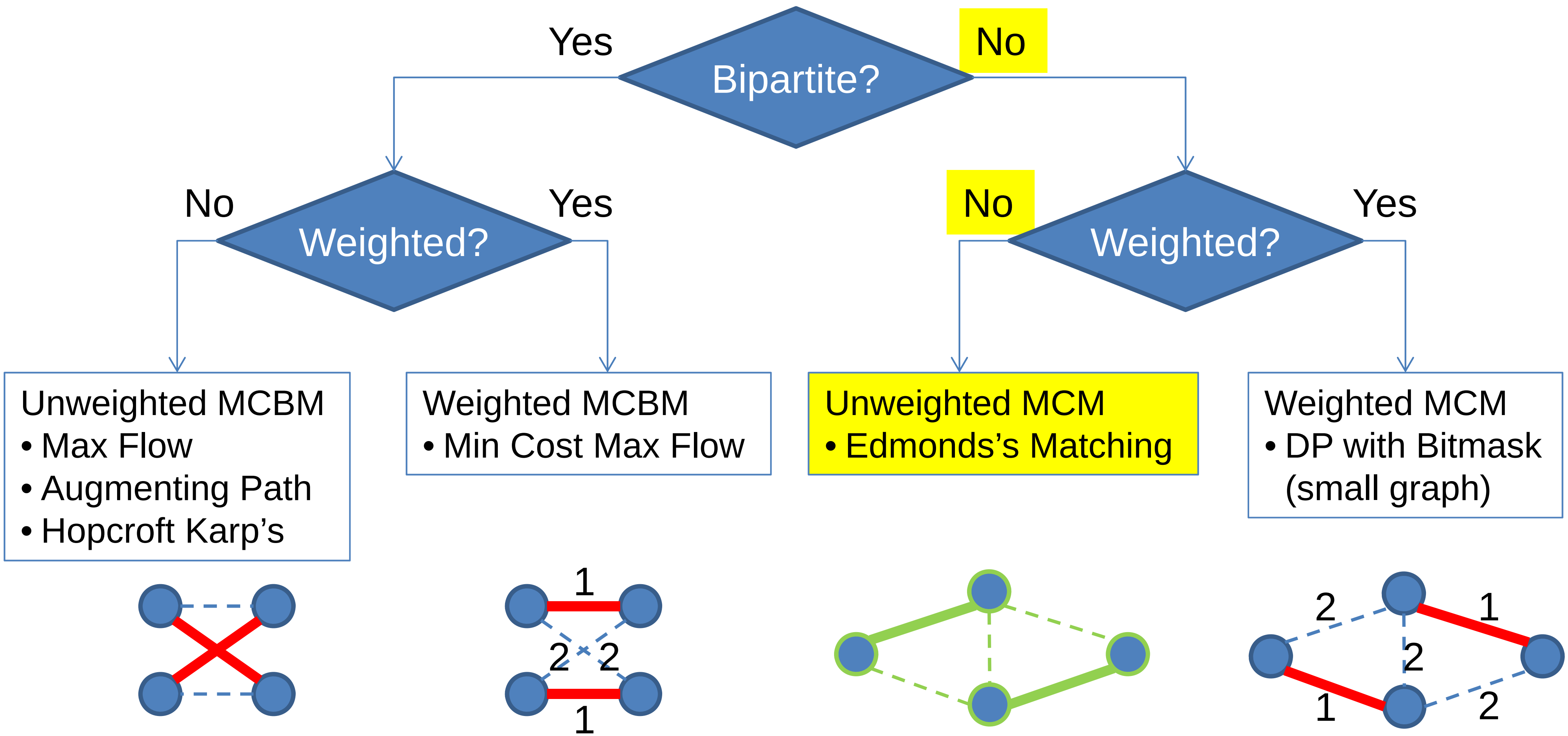
UVa 10746 (Solution)

Min Cost so far =
0 + **5.0** + 0 +
0 + 10.0 **- 5.0** + 10.0 + 0 +
0 + 20.0 + 0 = 40.0



Time Complexity: Depends on the chosen MCMF algorithm

Types of Graph Matching



Solution:

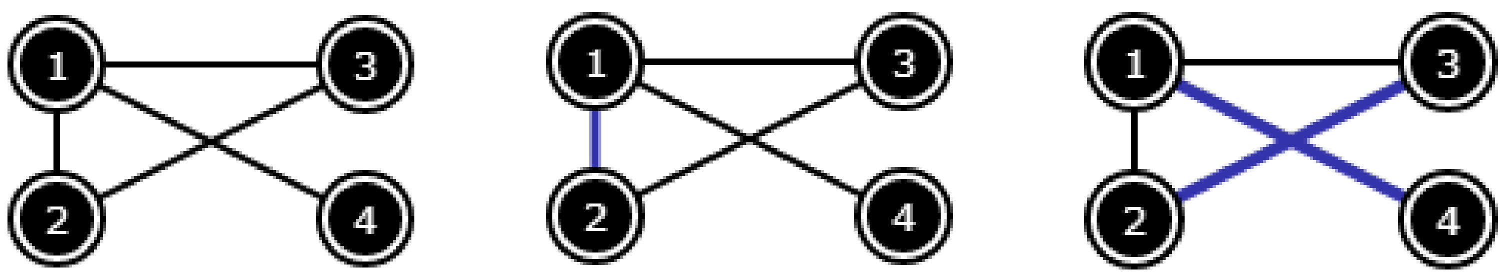
Edmonds's Matching Algorithm

UNWEIGHTED MCM

http://www.comp.nus.edu.sg/~stevenha/visualization/matching.html

Blossom

- A graph is not bipartite if it has at least one odd-length cycle (blossom)
- What is the MCM of this non-bipartite graph?



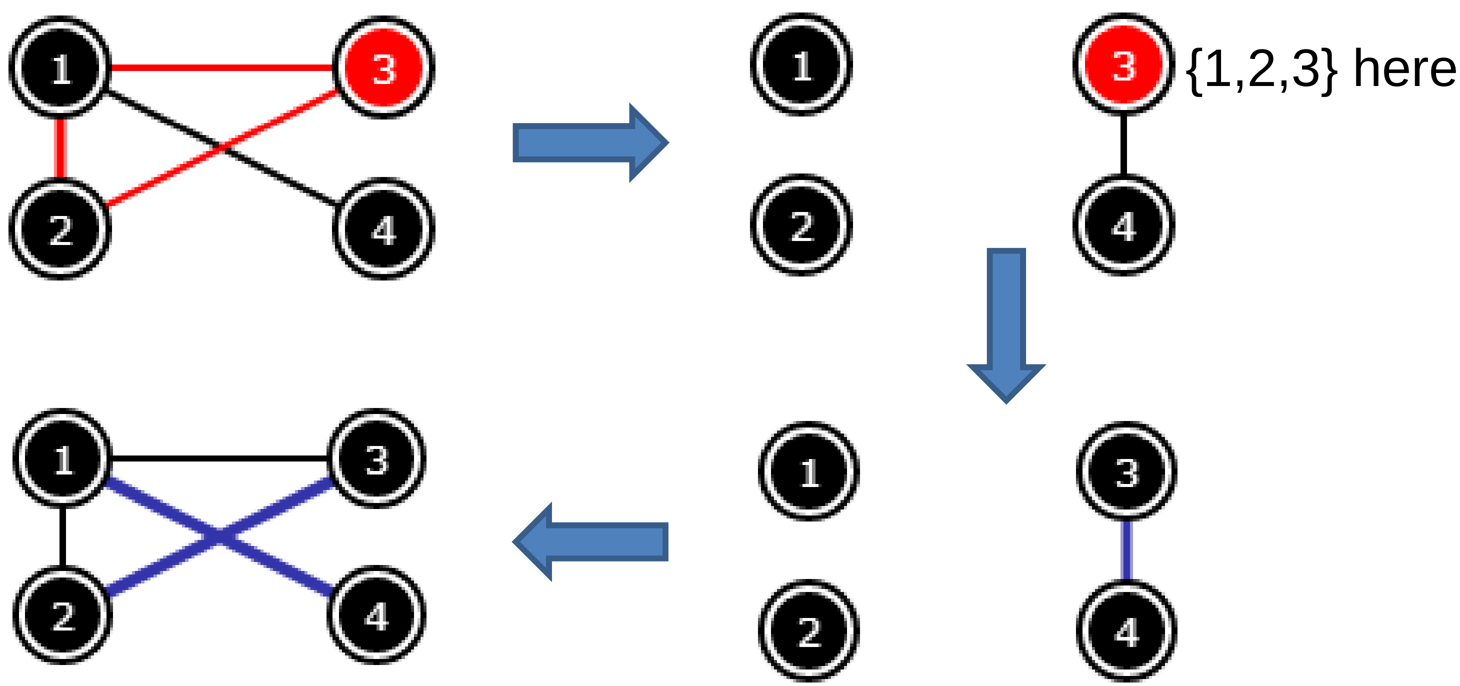
MCM = 2

- Harder to find augmenting path in such graph

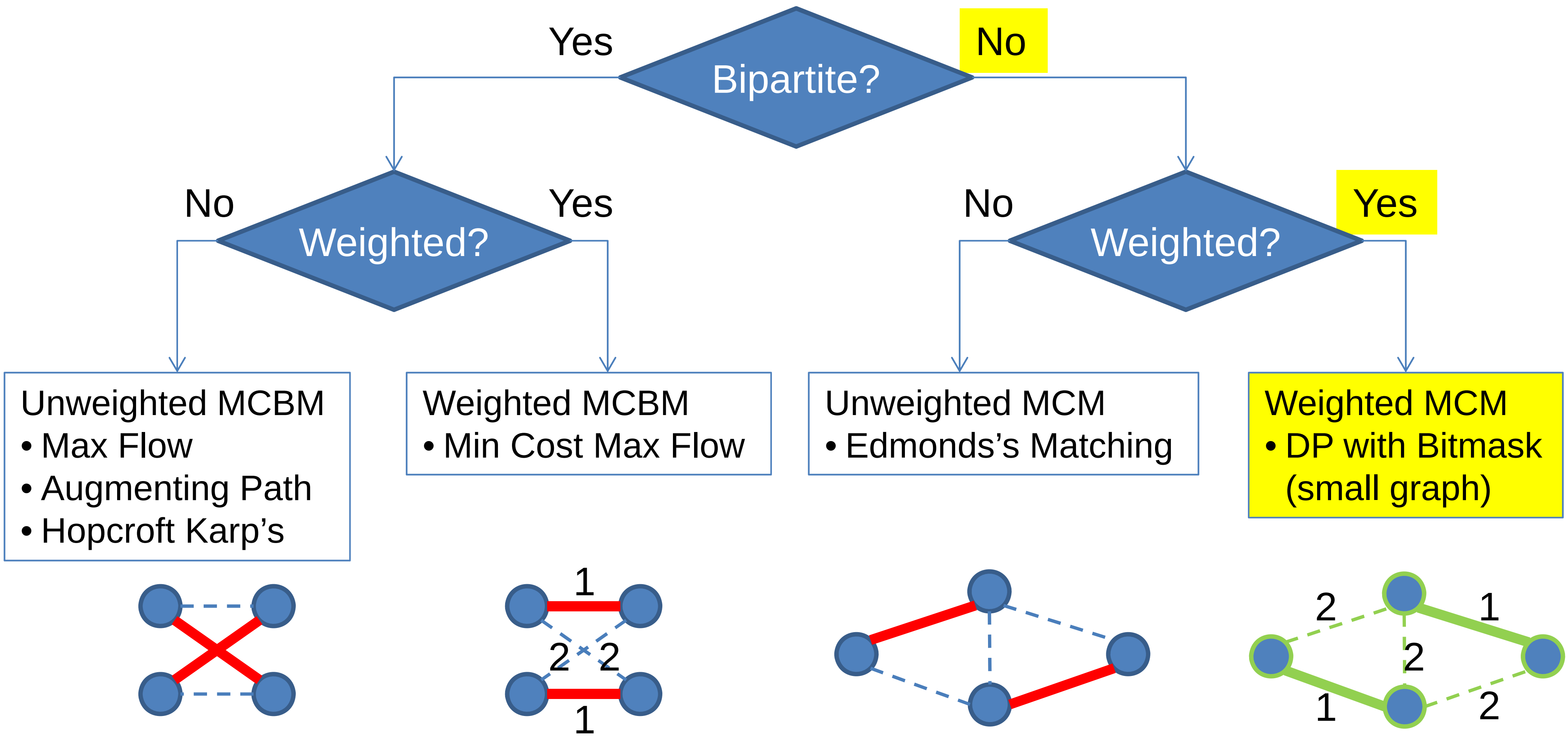
http://www.comp.nus.edu.sg/~stevenha/visualization/matching.html

Blossom Shrinking/Expansion

- Shrinking these blossoms (recursively) will make this problem "easy" again



Types of Graph Matching



Solution:

DP with Bitmask (only for small graph)

WEIGHTED MCM

Graph Matching in ICPC

- Graph matching problem is quite popular in ICPC
  - Sometimes 0 problem but likely 1 problem in the set
  - Perhaps disguised as other problems, e.g. Vertex Cover, Independent Set, Path Cover, etc → reducible to matching
- If such problem appear and your team can solve it, very good ☺
  - Your team will have +1 point advantage over significant # of other teams who are not trained with this topic yet…
- For IOI trainees… all these Graph Matching stuffs…
  - **THEY ARE NOT IN THE SYLLABUS TOO :O:O:O…**

References

- **CP2.9, Section 4.7.4, 9.15 ☺**
- **New write up about Graph Matching**