Integer and Combinatorial Optimization: Matching Problems

John E. Mitchell

Department of Mathematical Sciences
RPI, Troy, NY 12180 USA

January 2019
Matchings

We have a graph $G = (V, E)$. A matching in $G$ is a subset $M \subseteq E$ so that no two edges in $M$ share an endpoint.

In a perfect matching $M$, every vertex $v \in V$ is the endpoint of exactly one edge in $M$. The matching in the picture is a perfect matching.
Matchings

We have a graph $G = (V, E)$. A matching in $G$ is a subset $M \subseteq E$ so that no two edges in $M$ share an endpoint.

In a perfect matching $M$, every vertex $v \in V$ is the endpoint of exactly one edge in $M$. The matching in the picture is a perfect matching.
Weighted matching problems

The edges $e \in E$ can have weights $w_e$. We may be interested in finding a maximum weight matching, or a minimum weight perfect matching. These problems cannot be solved using a greedy algorithm.

For example, a greedy approach fails to find a maximum weight matching for the following graph:
Weighted matching problems

The edges $e \in E$ can have weights $w_e$. We may be interested in finding a **maximum weight matching**, or a **minimum weight perfect matching**. These problems **cannot be solved using a greedy algorithm**.

For example, a greedy approach fails to find a maximum weight matching for the following graph:

Greedy solution:
Weighted matching problems

The edges $e \in E$ can have weights $w_e$. We may be interested in finding a **maximum weight matching**, or a **minimum weight perfect matching**. These problems **cannot be solved using a greedy algorithm**.

For example, a greedy approach fails to find a maximum weight matching for the following graph:

**Optimal solution:**
Solving (weighted) matching problems

Nonetheless, the problems can be solved in polynomial time by using an alternating path method:

*Find a path that alternates between matched and unmatched edges, and swap them.*

When the graph is bipartite, can solve the problem by solving an assignment problem. May need to add edges with value 0, or dummy nodes.