Integer Linear Programming

Yulia Burkatovskaya

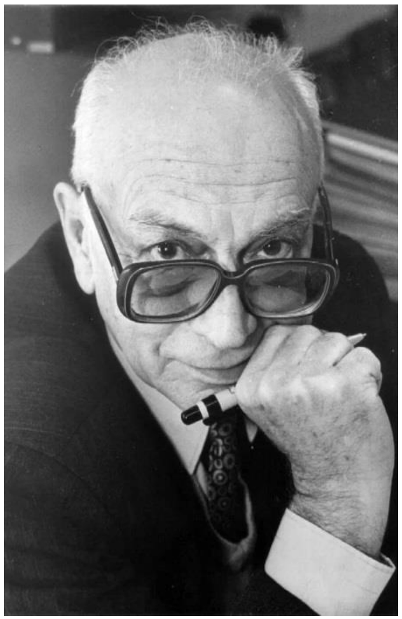
Outline

- Integer linear programming
- Augmenting path
- Maximum-cardinality matching
- Shortest cover
- Maximum flow

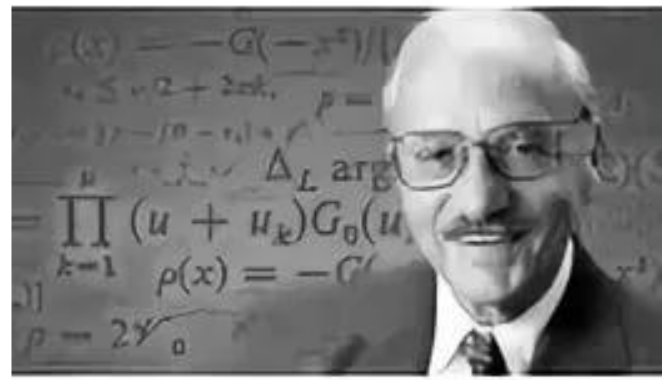
Linear programming

- **Linear programming** is a technique for the optimization of a linear objective function, subject to linear equality and linear inequality constraints.
- Its feasible region is **a convex polytope**, which is a set defined as the intersection of finitely many half spaces.
- A **linear programming algorithm** finds a point in the polyhedron where this function has the smallest (or largest) value if such a point exists.

Linear programming



- Leonid Cantorovich

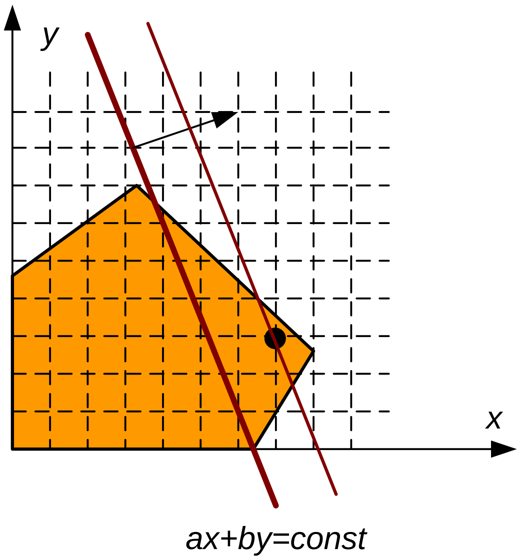


- George Bernard Dantzig

Integer linear programming

- $L(x) = CX \to max$
- $AX \leq B$
- $X \geq 0$

- $L(x) = \sum_{i=1}^{n} c_i x_i \to max$
- $\sum_{i=1}^{m} a_{j,i} x_i \leq b_j, j = 1, \dots, m;$
- $x_i \geq 0.$



Augmenting path

- **Augmenting path** – a pats in a graph, which improves a current solution of an integer linear programming problem

Maximum-cardinality matching

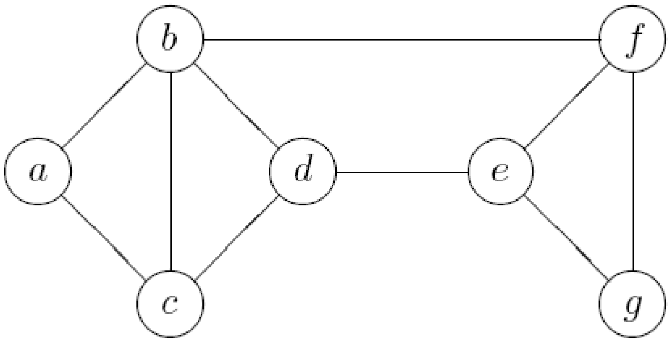
- A **vertex (edge) independent set** is a set of vertices (edges) of G so that no two vertices (edges) of the set are adjacent.
- The **edge independence number** β_1 of a graph G is the maximum number of independent edges.

Example.

$\{b, e\}$ – independent vertex set.

$\{ab, cd, fg\}$ – independent edge set.

$\beta_1 = 3$

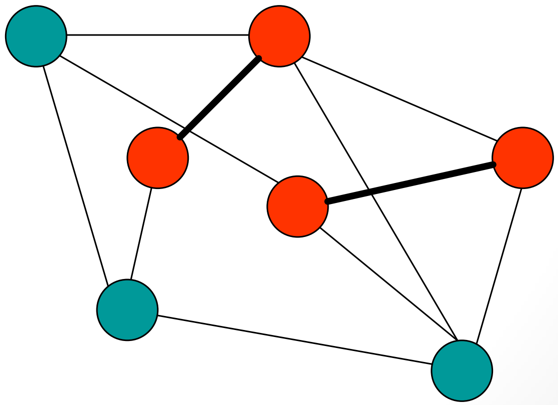


Maximum-cardinality matching

- **Matching** is an independent set of edges.

Let M be a matching in G(V,E).

- Two ends of an edge in M are **matched under** M.
- A matching M **saturates** a vertex v (and the vertex v is **M-saturated**) if some edge of M is incident with v; otherwise, the vertex v is **M-unsaturated**.

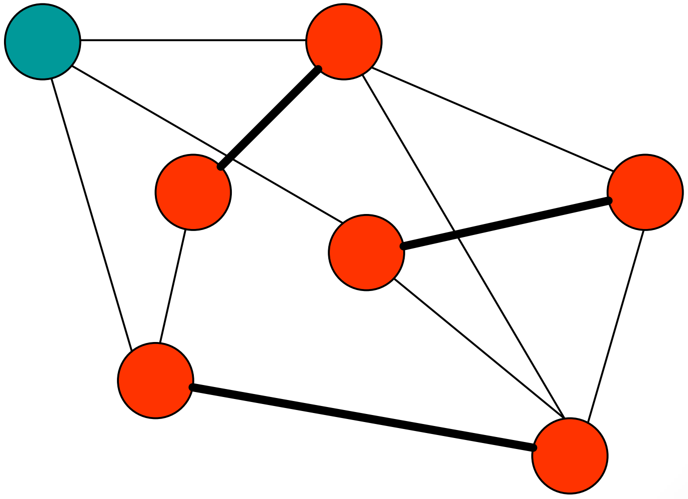


Maximum-cardinality matching

- If every vertex of G is M-saturated, the matching M is **perfect**.
- M is a **maximum-cardinality matching** in G, if $|M|=\beta_1$.
- Every perfect matching is a maximum one. A perfect matching does not always exist.

Example.

- A maximum-cardinality matching (not a perfect matching).



Maximum-cardinality matching

- $\xi_i = 1$ if and only if the edge *j* belongs to the matching;
- c_j is the weight of the edge *j*;
- *I* is the incidence matrix.

The problem can be stated as a discrete linear programming problem.

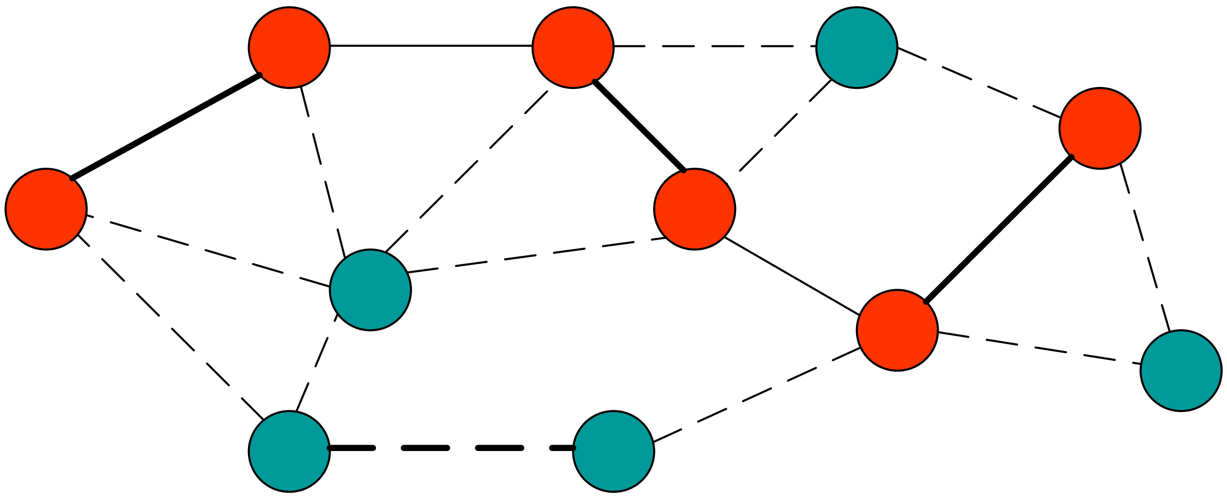
$$\sum_{j=1}^{q} c_j \xi_j \to \max;$$

$$\sum_{j=1}^{q} I_{kj} \xi_j \leq 1, \quad \forall k = 1, \ldots, p;$$

$$\xi_j \in \{0, 1\}.$$

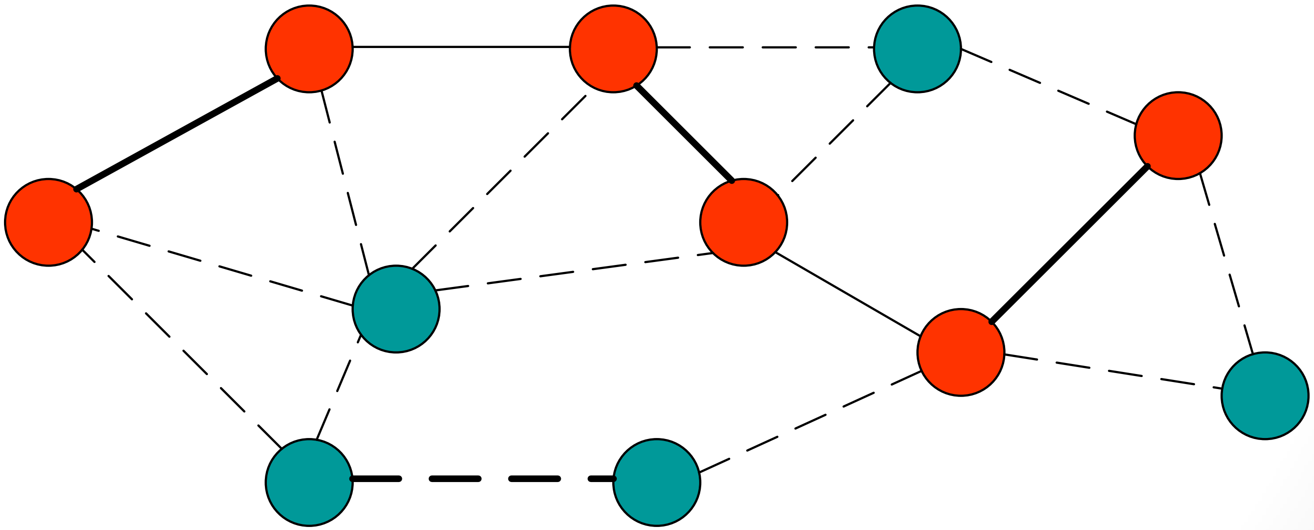
Maximum-cardinality matching

- Given a matching M in graph G, an *alternating path* is a path in which the edges are alternately in and out of matching M.



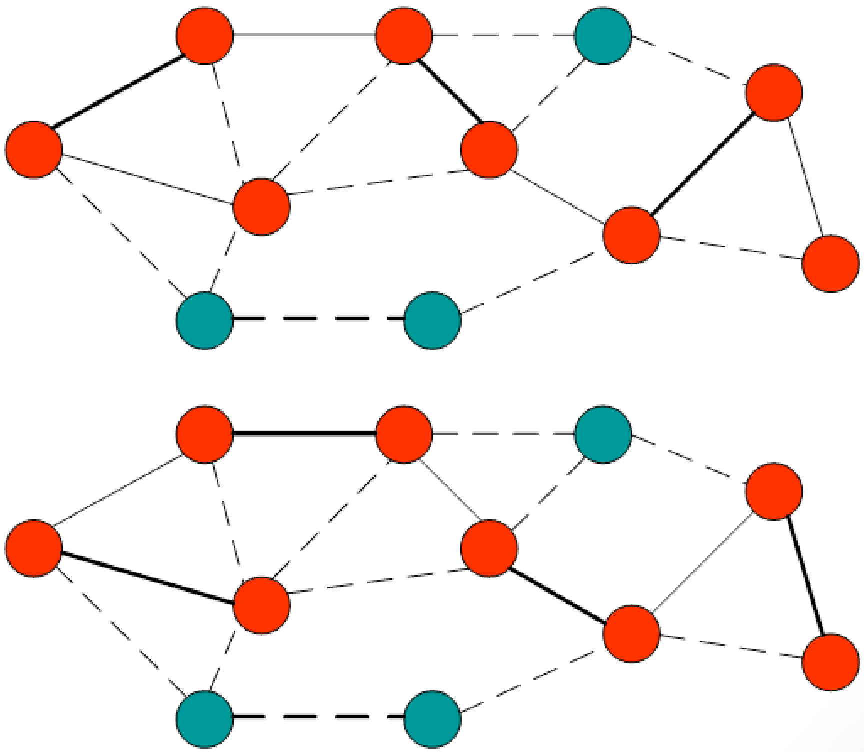
Maximum-cardinality matching

- An **augmenting path** is an alternating path whose first and last vertices are M-unsaturated.



Maximum-cardinality matching

- A matching is a maximum-cardinality matching if, and only if, it does not contain an augmenting path.
- If an augmenting path is found, the roles in the matching of the edges in this path are reversed. This creates a matching with greater cardinality.

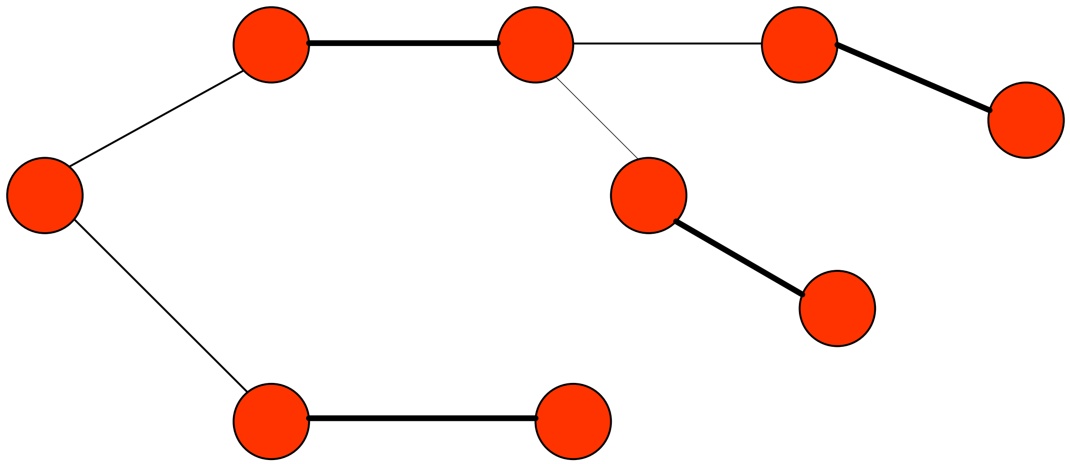


Maximum-cardinality matching

An **alternating tree** relative to a given matching M is a tree T for which:

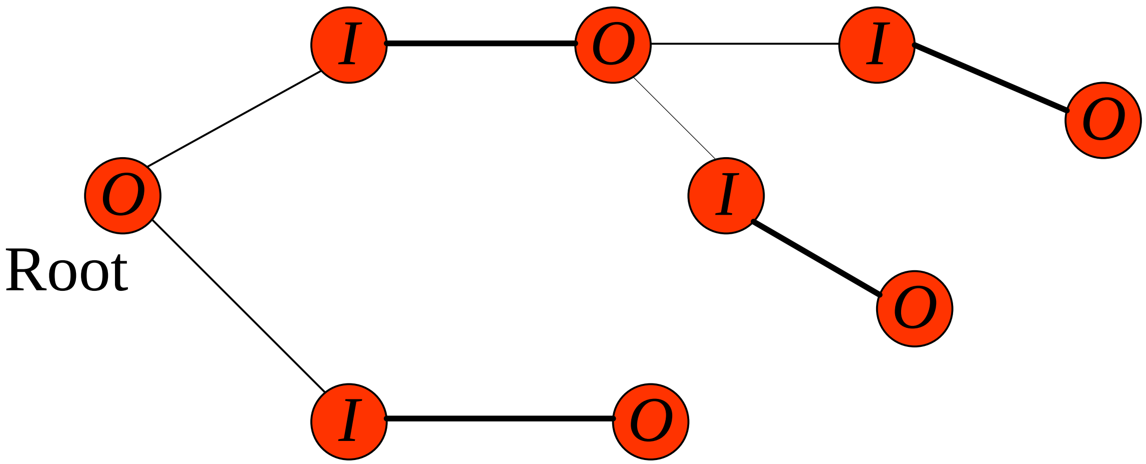
- One vertex of T is M-unsaturated and is called the *root* of T.
- All paths starting at the root are alternating paths.
- All maximal paths from the root of T are of even cardinality, i.e. contain an even number of edges.

Root



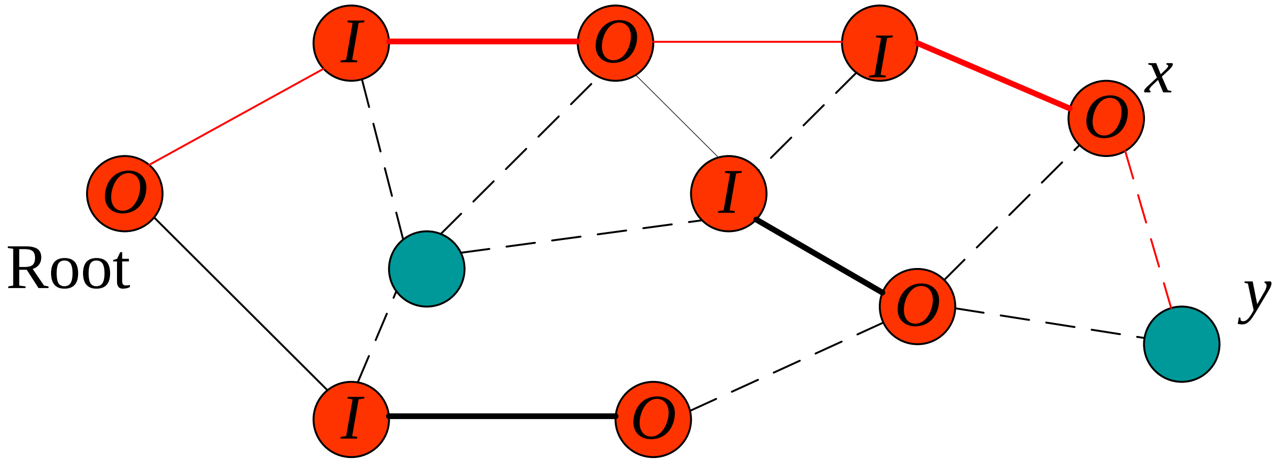
Maximum-cardinality matching

- Starting from the root of the tree and labeling it outer the vertices along any path starting from the root are labeled alternately *inner* and *outer*.
- The degree of all inner vertices is exactly 2 whereas the degree of an outer vertex can be any integer greater than or equal to 1.

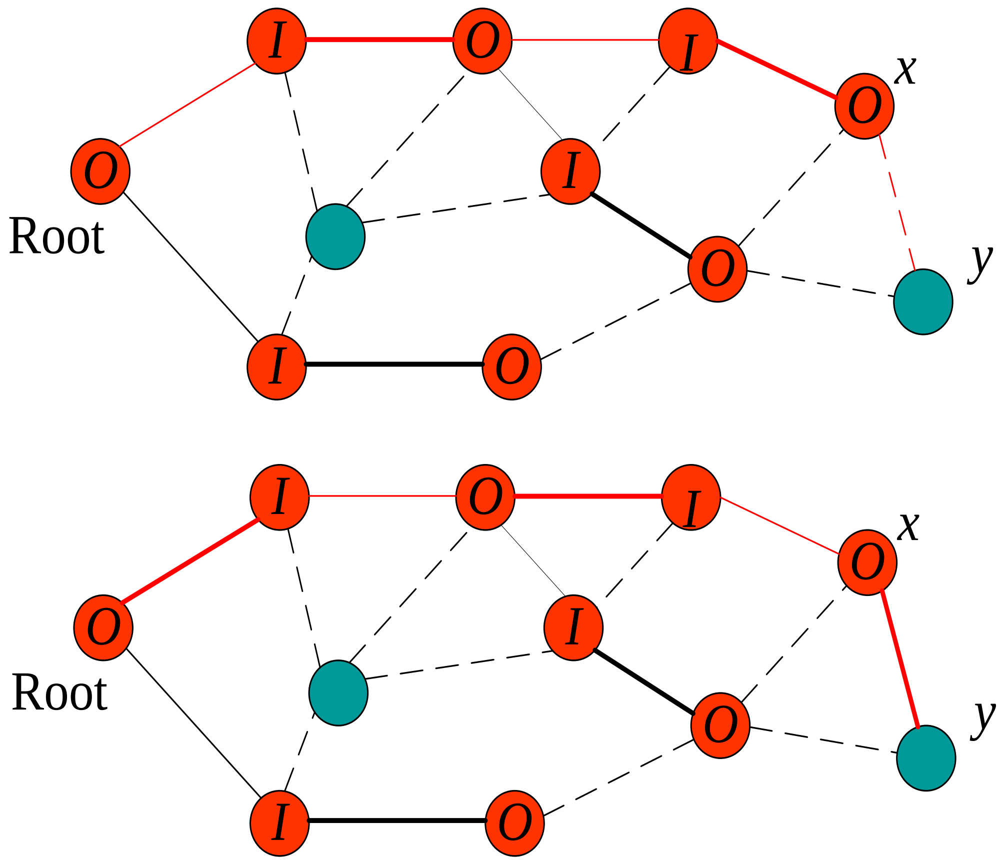


Maximum-cardinality matching

- An **augmenting tree** is an alternating tree relative to a given matching M whenever an edge exists from an outer vertex x of the tree to a M-saturated vertex y not in the tree.
- The unique path from the root of the tree to x plus link (x, y) is then an augmenting path.

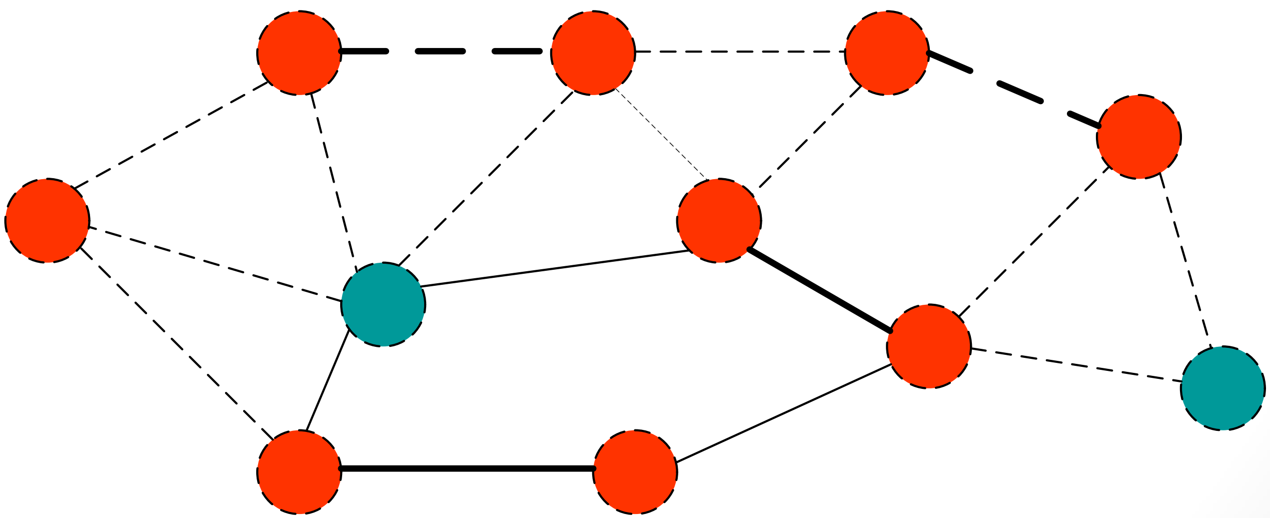


Maximum-cardinality matching



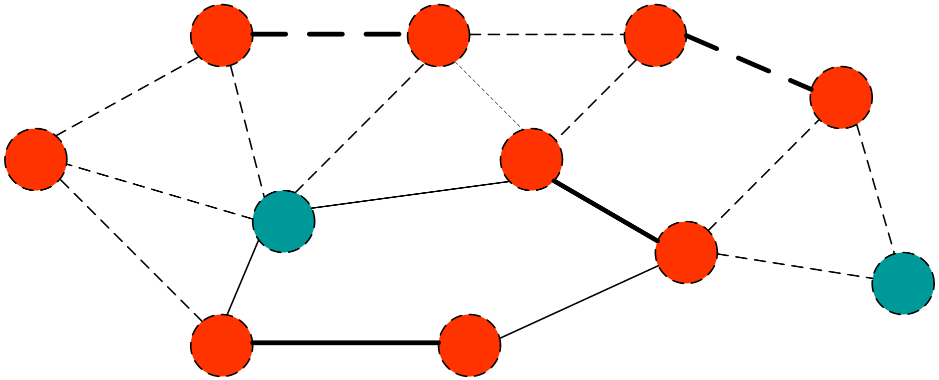
Maximum-cardinality matching

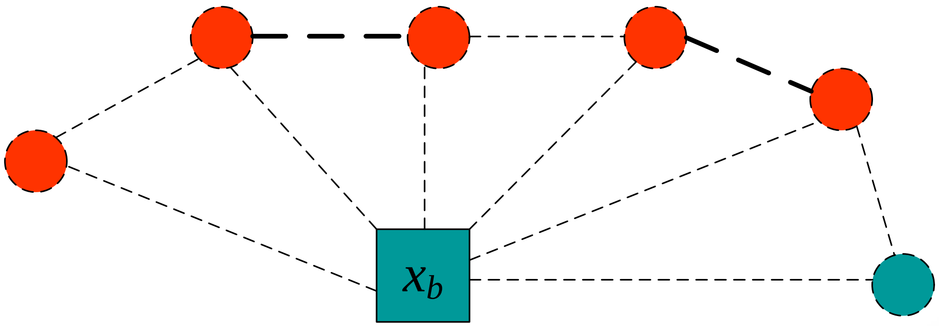
- A **blossom** with respect to a matching M is an augmenting path for which the initial and final exposed vertices are identical—i.e. the path forms a circuit—and the number of edges (or vertices) of the circuit is odd.



Maximum-cardinality matching

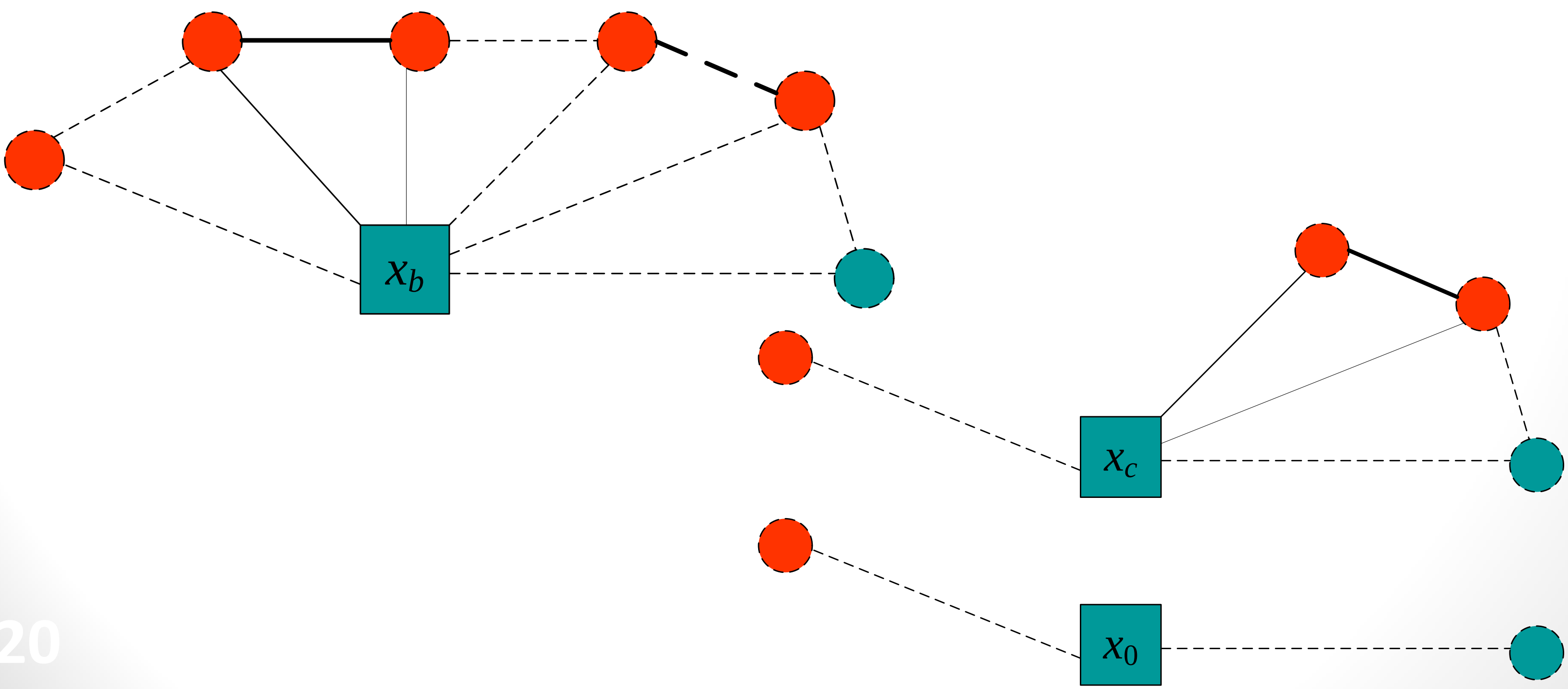
- Blossoms are *shrunk* to derive a new simpler graph. The *shrinking* of a blossom B implies the replacement of all vertices of B (say X_b) by a single new *pseudovertex* x_b.





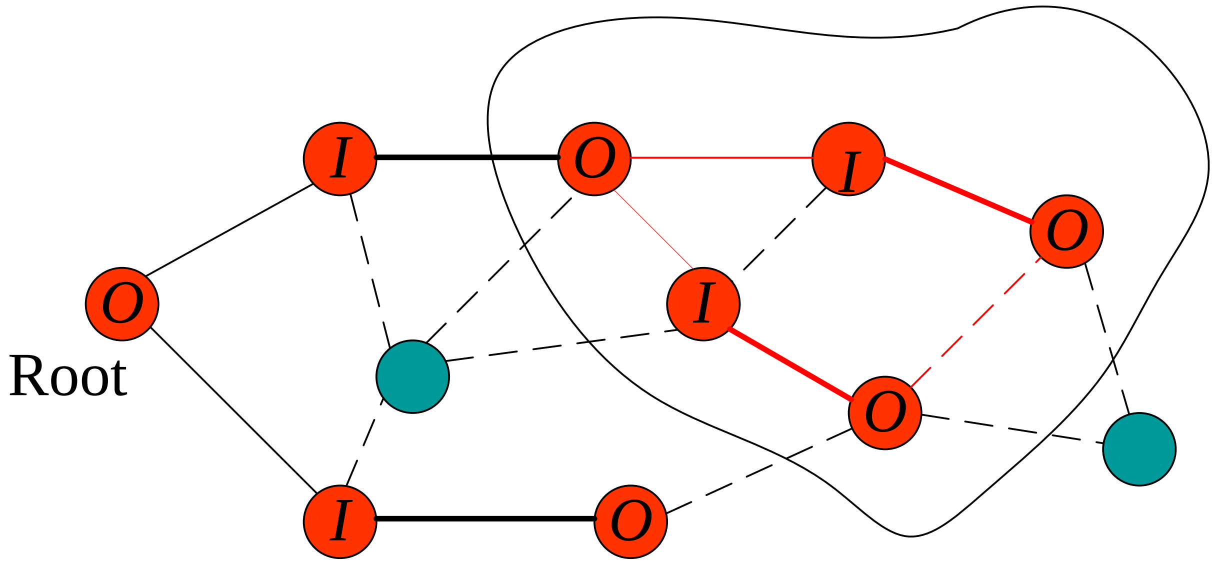
Maximum-cardinality matching

- In the simpler graph resulting from such a shrinking, vertex x_b may form a new blossom which is shrunk again and so on.
- The final blossom B0 which is not contained in any other blossom is called an *outermost blossom*.



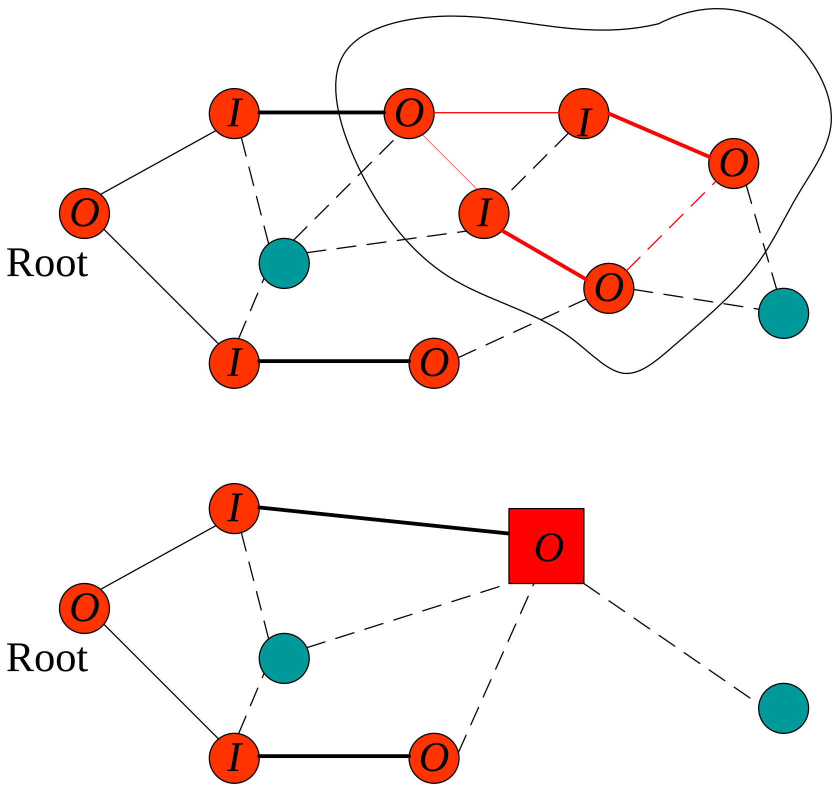
Maximum-cardinality matching

- A *blossomed tree* is an alternating tree relative to a given matching whenever a link exists between two outer vertices of the tree.



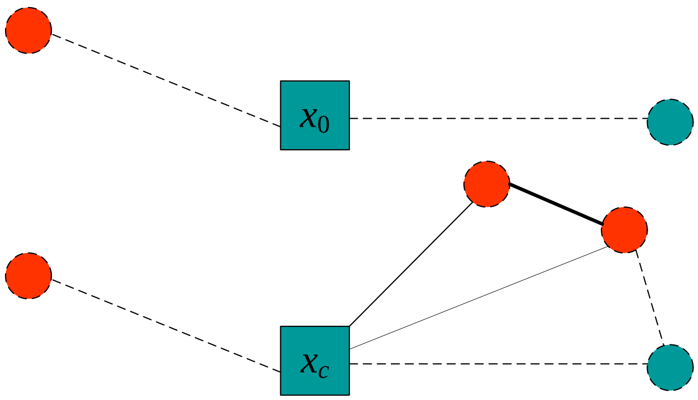
Maximum-cardinality matching

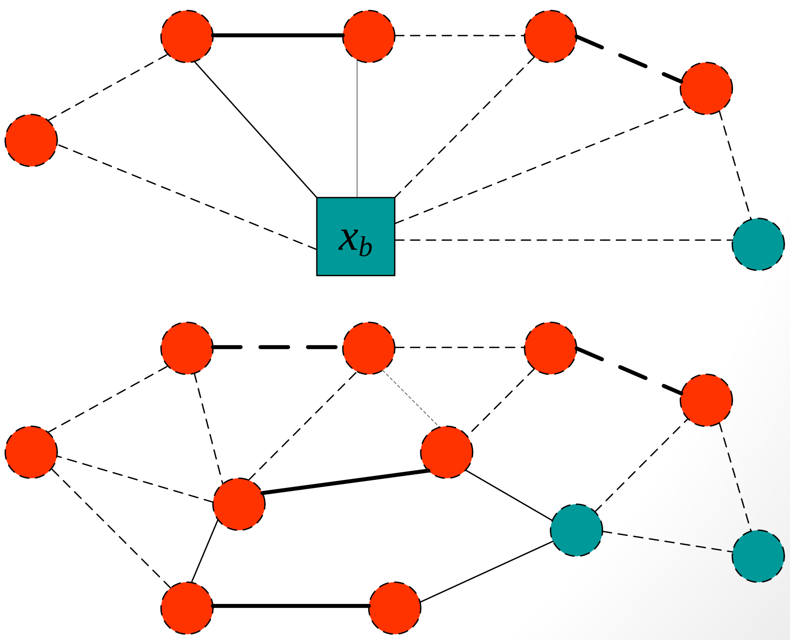
- Whenever a blossom B is shrunk, the resulting pseudovertex x_b is labelled an outer vertex.



Maximum-cardinality matching

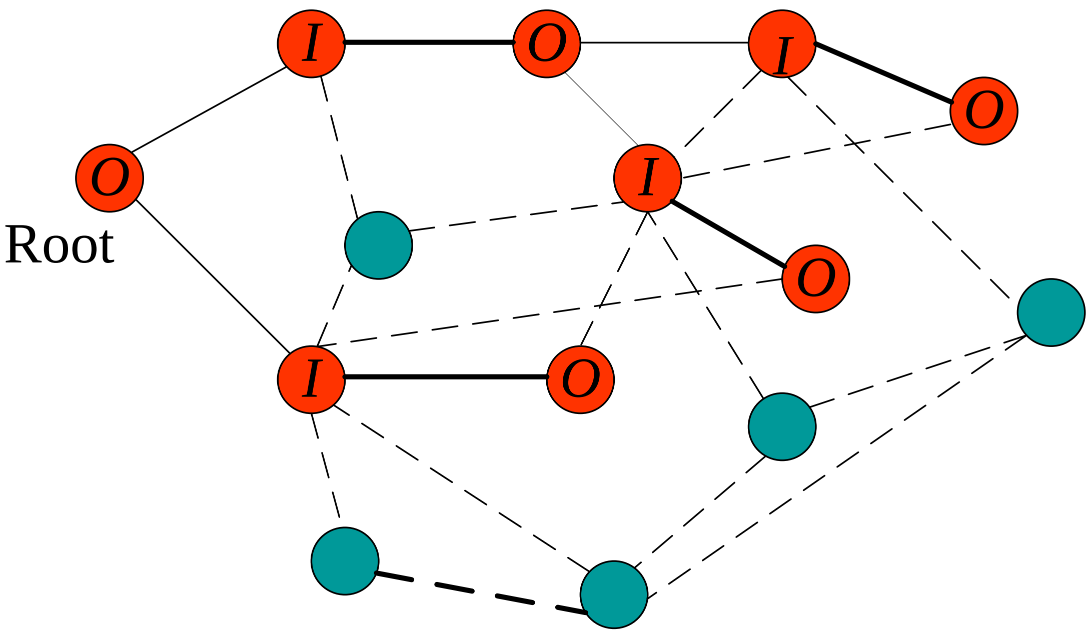
- If B is a blossom based on the odd vertex set X_b, and if x is any vertex in X_b, then there exists a maximum cardinality matching in the subgraph induced by X_b which leaves x unsaturated.





Maximum-cardinality matching

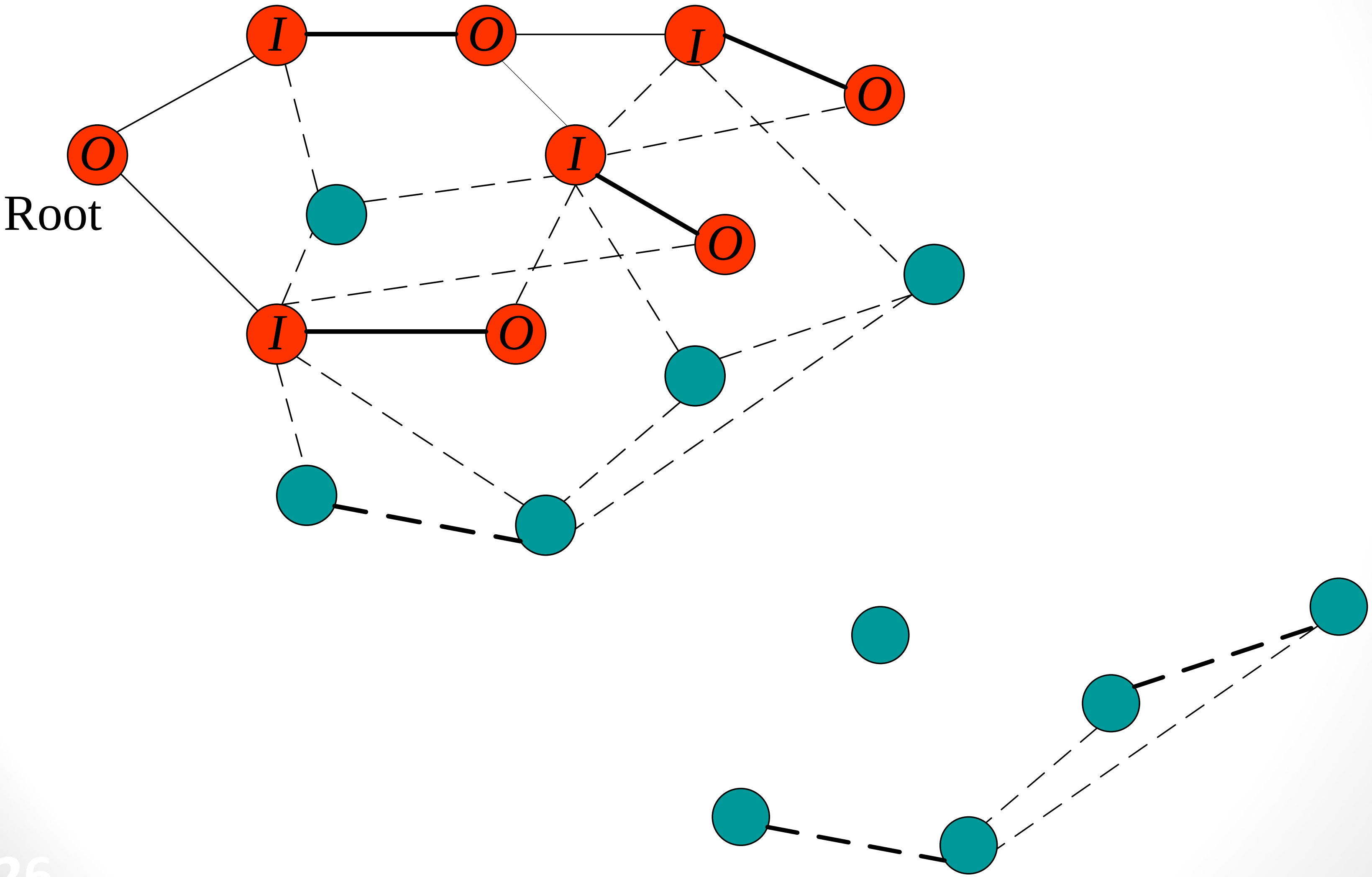
- A *hungarian* tree is an alternating tree in a graph in which all links having an outer vertex of the tree as one end, have an inner vertex also in the tree as the other end.



Maximum-cardinality matching

- Let H be a Hungarian tree in a graph $G = (V, E)$ and
- $G_0 = (V \setminus V_H, Y)$ be the subgraph of G excluding the set V_H of vertices of H.
- Then, if M_H is the matching in the tree H and M_0 is any maximum-cardinality matching in G_0, the set of edge $(M_H \cup M_0)$ is a maximum-cardinality matching in G.

Maximum-cardinality matching



Maximum-cardinality matching

An alternating tree is rooted at an exposed (unsaturated) vertex and grown by alternately adding links which are in and not in the matching until:

- either (i) The tree becomes **augmenting**,
- or (ii) The tree **blossoms**,
- or (iii) The tree becomes **Hungarian**.

Maximum-cardinality matching

- In case (i) the cardinality of the matching can be increased by one simply by tracing the augmenting path back to the root of the tree and then interchanging those edges of the path that belong to the matching with the ones that do not. After augmentation the tree is discarded, and a new tree is rooted at some remaining unsaturated vertex, if one exists.

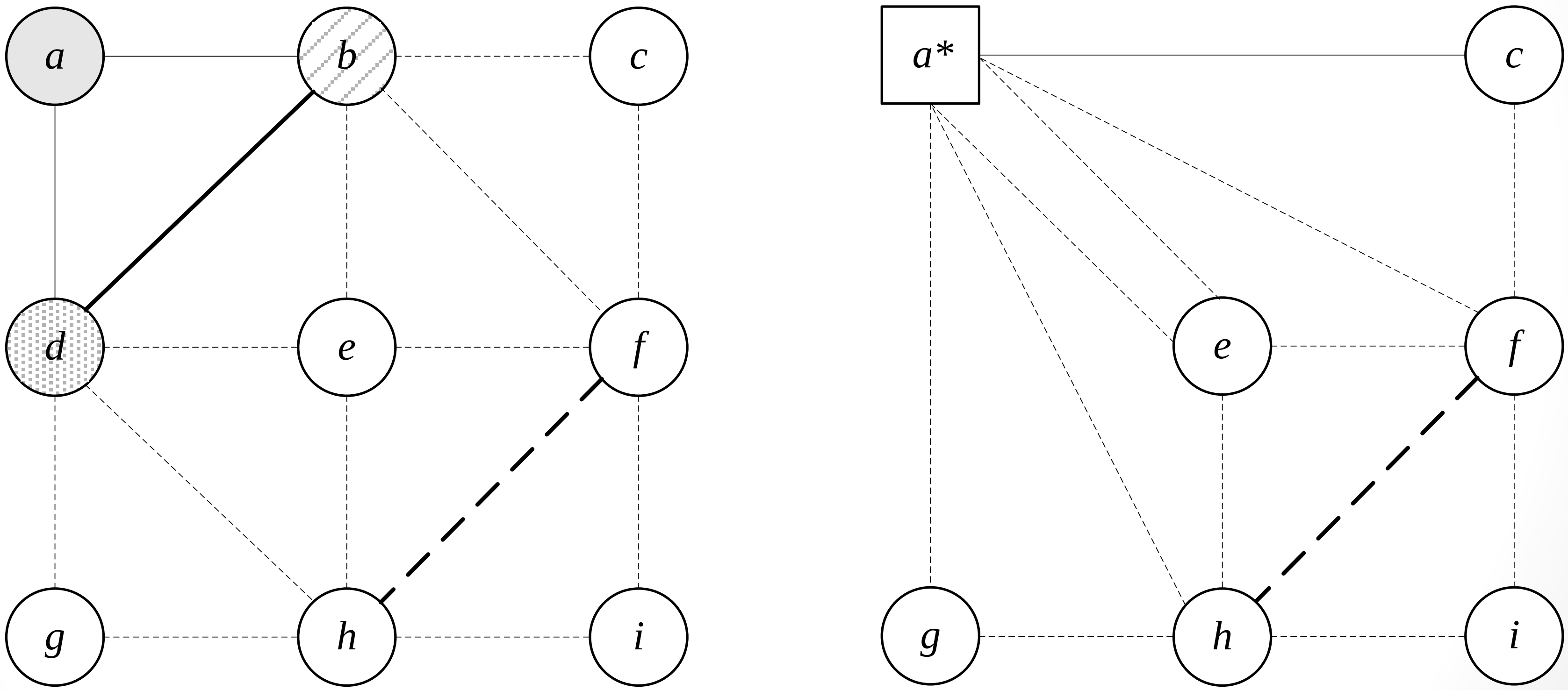
Maximum-cardinality matching

- In case (ii), the resulting blossom is identified, shrunk, and the growing of the tree continued in search for an augmenting path. As far as the computing is concerned, the shrinking of a vertex need not be done explicitly. All that is required is to mark all the vertices of the blossom as outer and set up labels on the vertices to indicate that they all belong to this blossom.
- The order in which these blossoms have been "shrunk" is important since at the end of the procedure the blossoms must be "expanded" in reverse order.

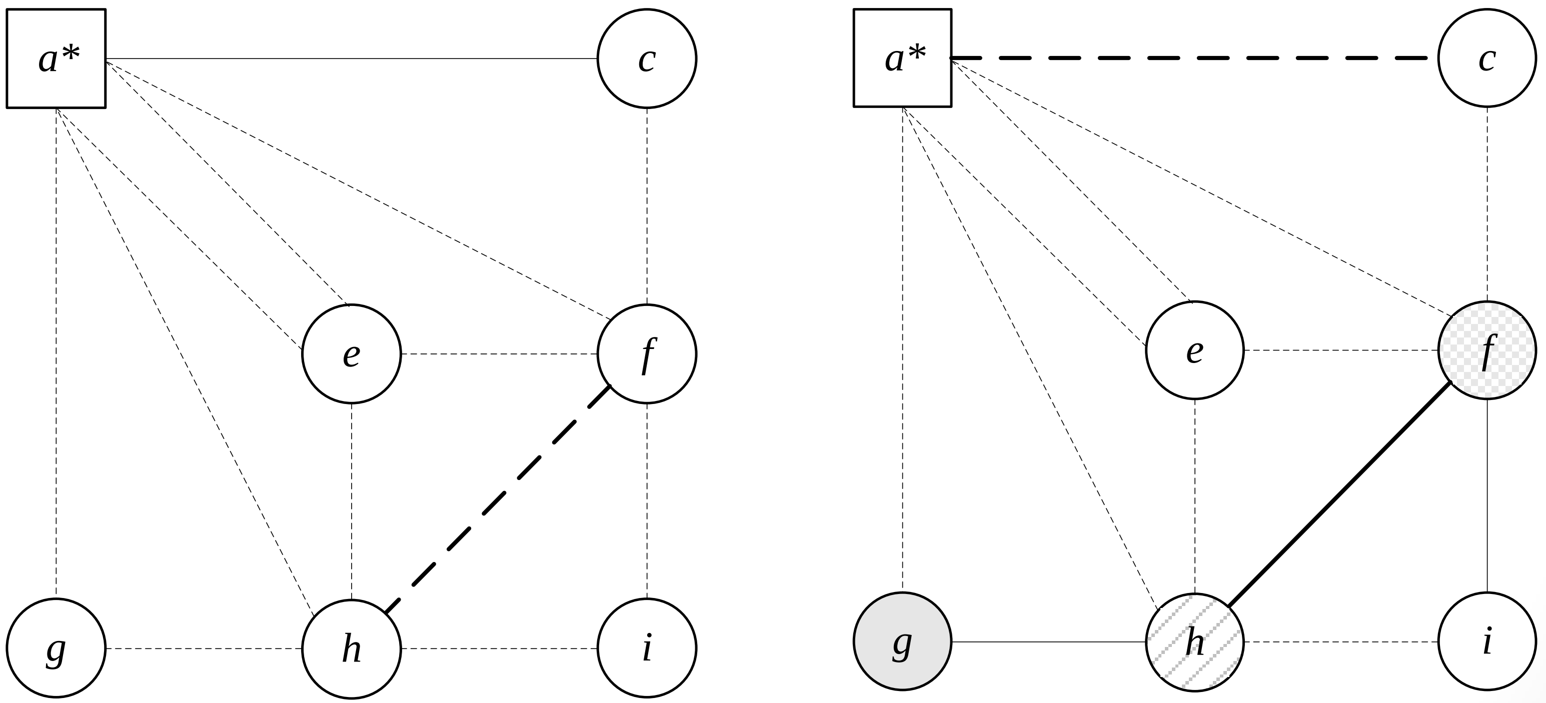
Maximum-cardinality matching

- In case (iii) the vertices of the Hungarian tree and their incident links are removed from the graph and the algorithm is reapplied to the remaining subgraph.

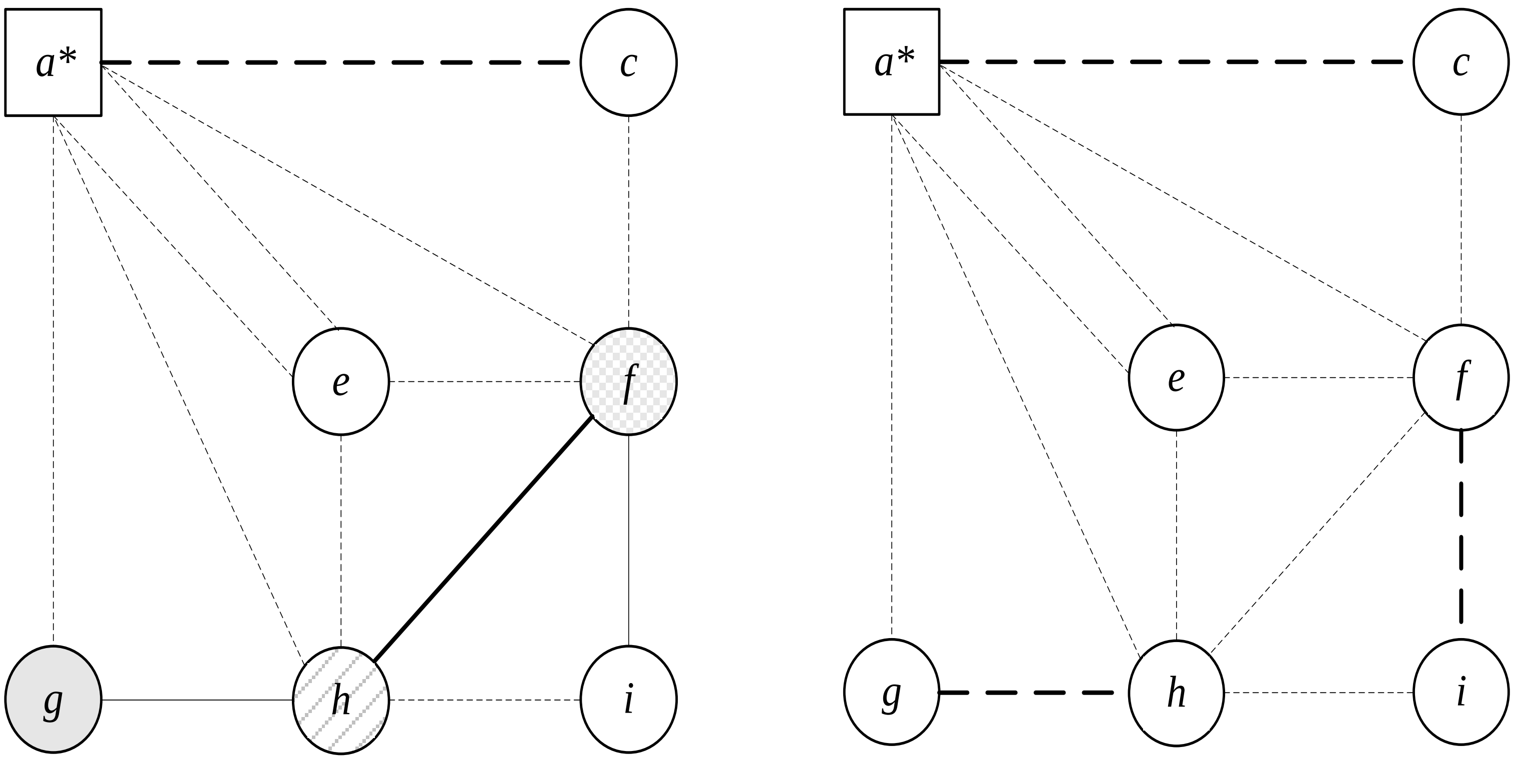
Maximum-cardinality matching



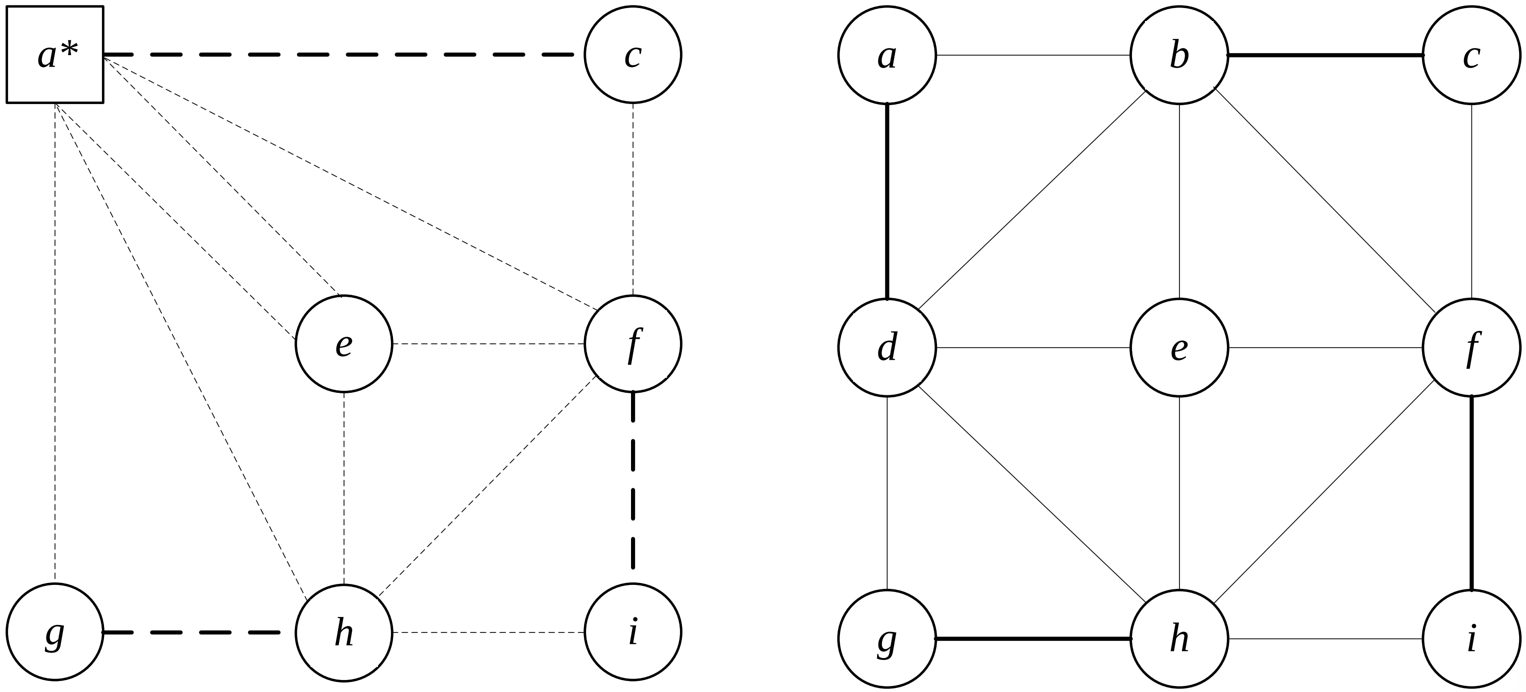
Maximum-cardinality matching



Maximum-cardinality matching



Maximum-cardinality matching



Shortest cover

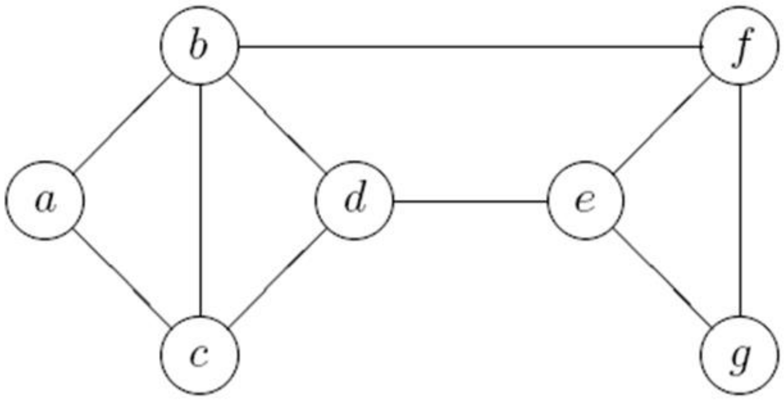
An edge **covers** a vertex if they are incident.

An **edge covering set (edge cover)** is a set of edges of G covering all vertices of G.

Example.

The edge ab covers the vertices a and b.

{ab,cd,de,fe,fg} – an edge cover.



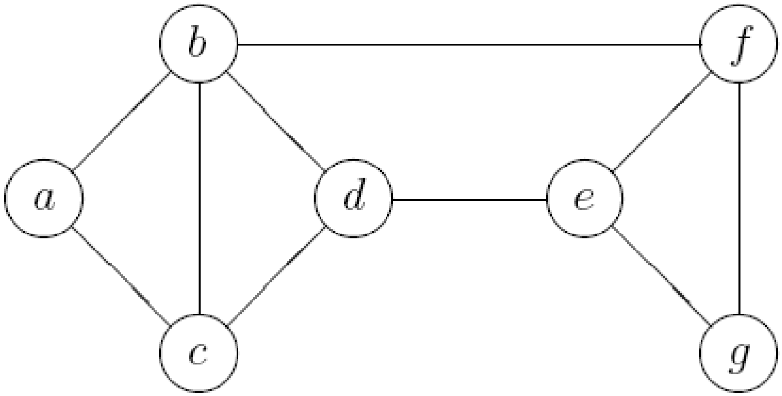
Shortest cover

A cover is called **shortest** when it contains the smallest possible number of edges.

Example.

{ab,cd,de,fe,fg} is not a shortest cover

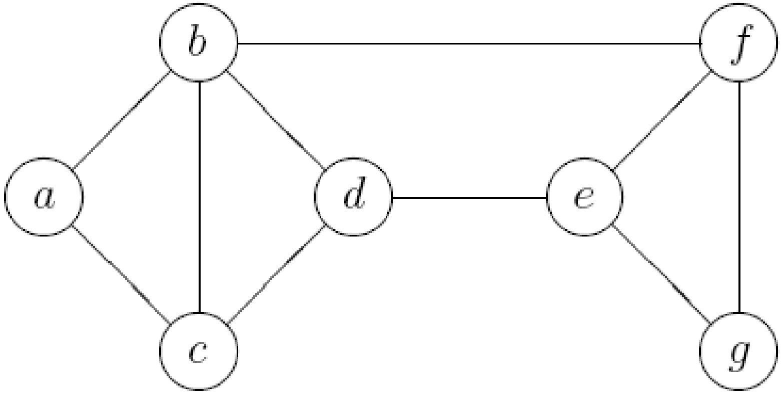
{ab,cd,ef,eg} is a shortest cover.



Shortest cover

The **edge cover number** α_1 of a graph G is the size of a shortest edge cover in a graph, i.e., the minimum number of edges covering all vertices.

Example. $\alpha_1 = 4$, {ab,cd,eg,ef} – shortest edge cover.



Shortest cover

- ξ_j = 1 if and only if the edge j belongs to the cover;
- c_j is the weight of the edge j;
- I is the incidence matrix.

The problem can be stated as a discrete linear programming problem (the shortest cover of the transposed incidence matrix).

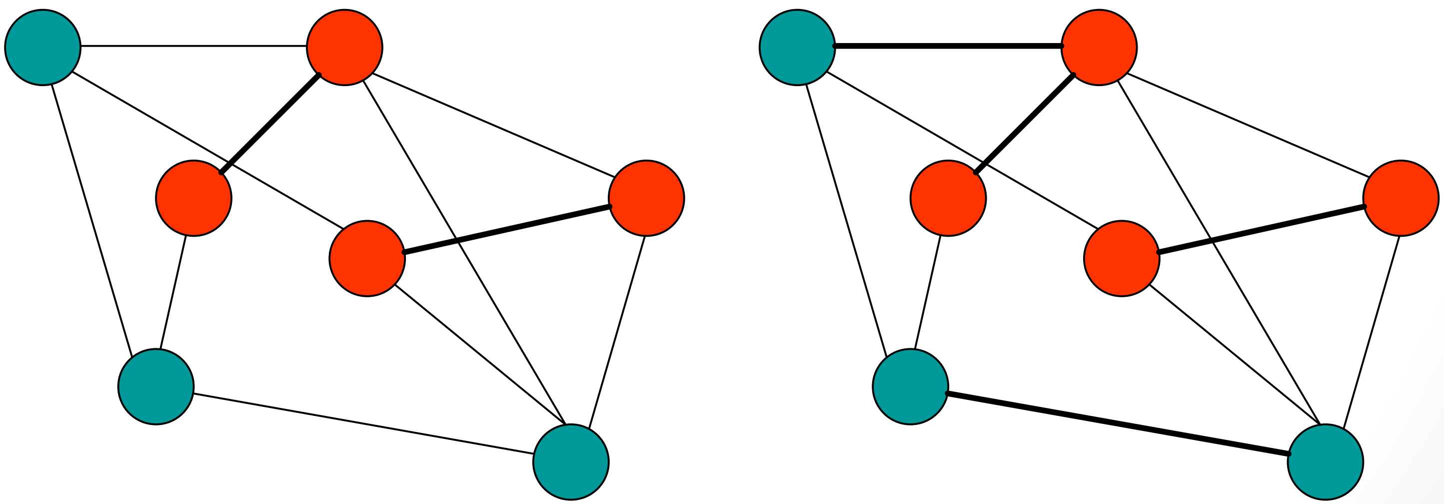
$$\sum_{j=1}^{q} c_j \xi_j \to \min;$$

$$\sum_{j=1}^{q} I_{kj} \xi_j \geq 1, \quad \forall k = 1, \dots, p;$$

$$\xi_j \in \{0, 1\}.$$

Shortest cover

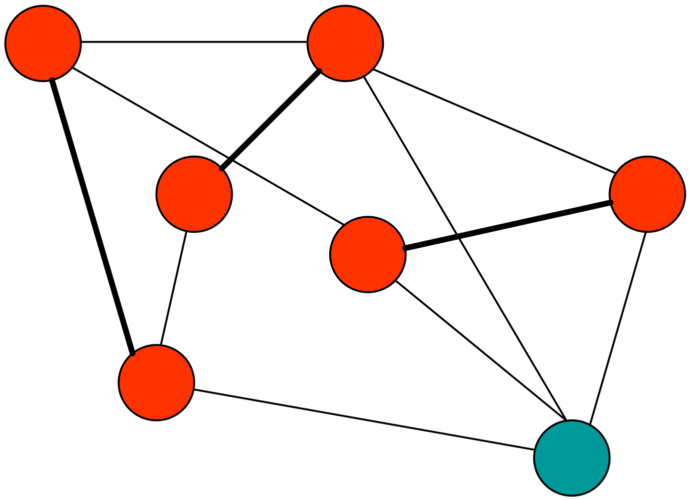
- A solution of the **maximum-cardinality matching problem** provides a solution of the **shortest cover problem**.
- From matching to cover: let M be a matching. Choose vertex v that is not covered by M. Add to M an edge incident to v. Repeat until there are no non-covered vertices, as a result get a cover C.

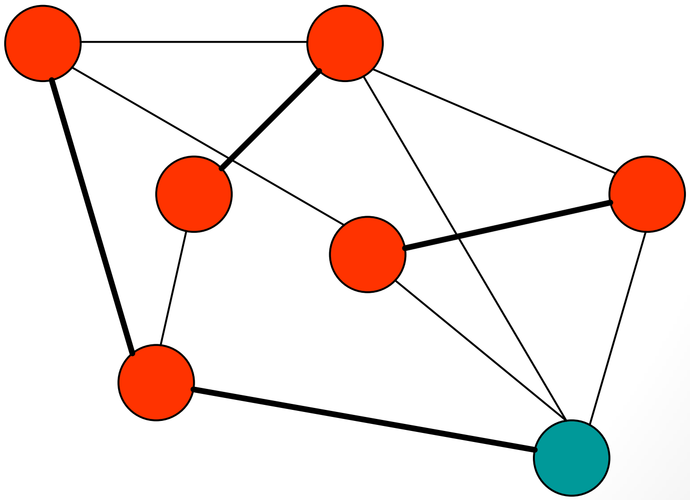


Shortest cover

If M is a maximum matching then C is a minimum cover.

- M covers 2|M| vertices.
- |C|=|M|+(p–2|M|), because if M is a maximum matching then there are no edges connecting vertices non-covered by M; hence, to cover the vertices we need V–2|M| edges.
- If $|M|=\beta_1$ then $|C|=\beta_1+(p-2\,\beta_1)= p-\beta_1 = \alpha_1$.

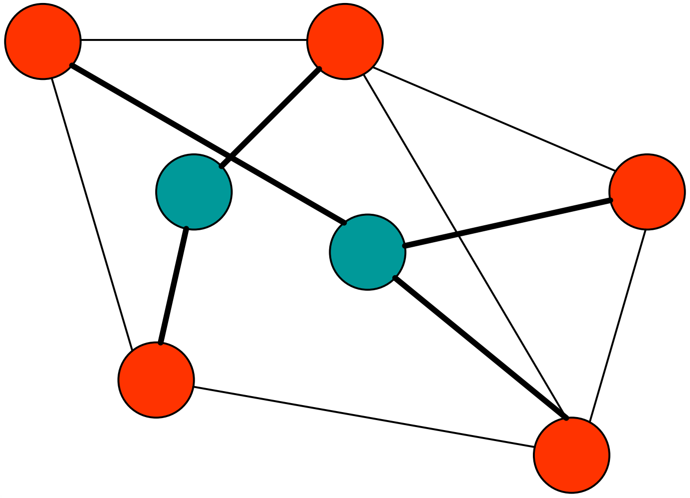




40

Shortest cover

- A solution of the **shortest cover problem** provides a solution of the **maximum-cardinality matching problem**.
- From cover to matching: let C be a cover. Choose vertex v that is incident to more then one edge of C. Remove from C any edge incident to v. Repeat until there are no vertices covered by several edges, as a result get a matching M.

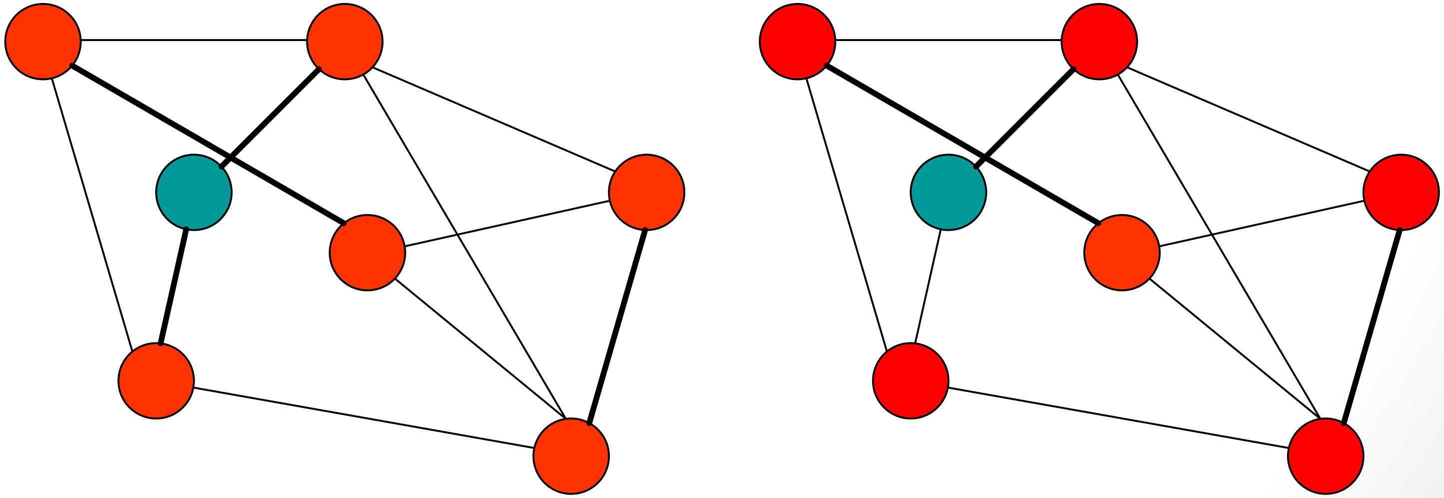


41

Shortest cover

If C is a minimum cover then M is a maximum matching.

- If C were a matching it would cover 2|C| vertices.
- We remove 2|C|–p edges
- If $|C| = \alpha_1$ then $|M| = \alpha_1 - (2\alpha_1 - p) = p - \alpha_1 = \beta_1$.



42