Matching Algorithm

- **Input: An X,Y-bigraph G, a matching M in G, and the set U of M-unsaturated vertices in X.**
- **Output: an M-augmenting path or a vertex cover of size |M|.**

(0) begin by labeling with (*) all vertices in U call all vertices in U unscanned. Go to (1)

(1) if in the previous step, no new label has been given to a vertex of X, then stop. Otherwise go to (2).

(2) while there exists a labeled but unscanned vertex of X, select a labeled but unscanned vertex of X, say x_i, and label with (x_i) all vertices in Y which are joined to x_i by an edge not belonging to M and which have not been previously labeled. The vertex x_i is now scanned. If there are no labeled but unscanned vertices go to (3)

(3) If in step (2), no new label has been given to a vertex of Y, then stop. Otherwise go to (4)

(4) While there exists a labeled, but unscanned vertex of Y, select a labeled but unscanned vertex of Y, say y_i, and label with (y_i) any vertex of X which is joined to y_i by an edge belonging to M and which has not been previously labeled. The vertex y_i is now scanned. If there are no labeled but unscanned vertices, go to (1).

Breakthrough and non-breakthrough

- Breakthrough: there is a labeled vertex of Y which does not meet an edge of M

In the case of breakthrough occurring in the algorithm, an M-augmenting path can be find and a matching with one more edge than M can be constructed.

- Non-breakthrough: each vertex of Y which is labeled also meets some edge of M.

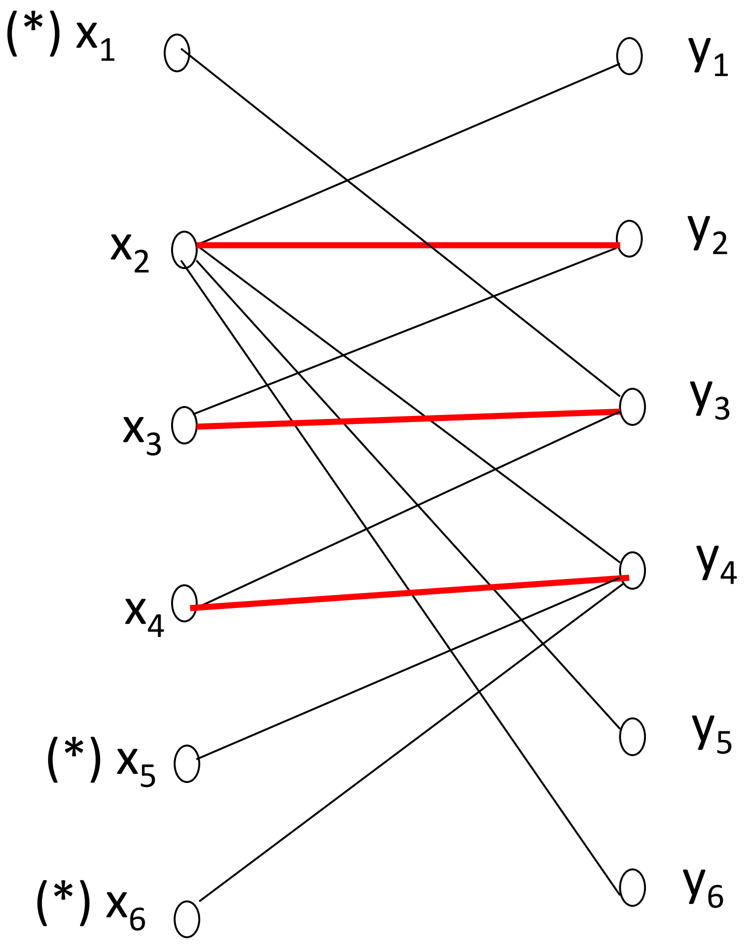
In the case of non-breakthrough, we conclude that M is the max-matching.

Matching Theorem

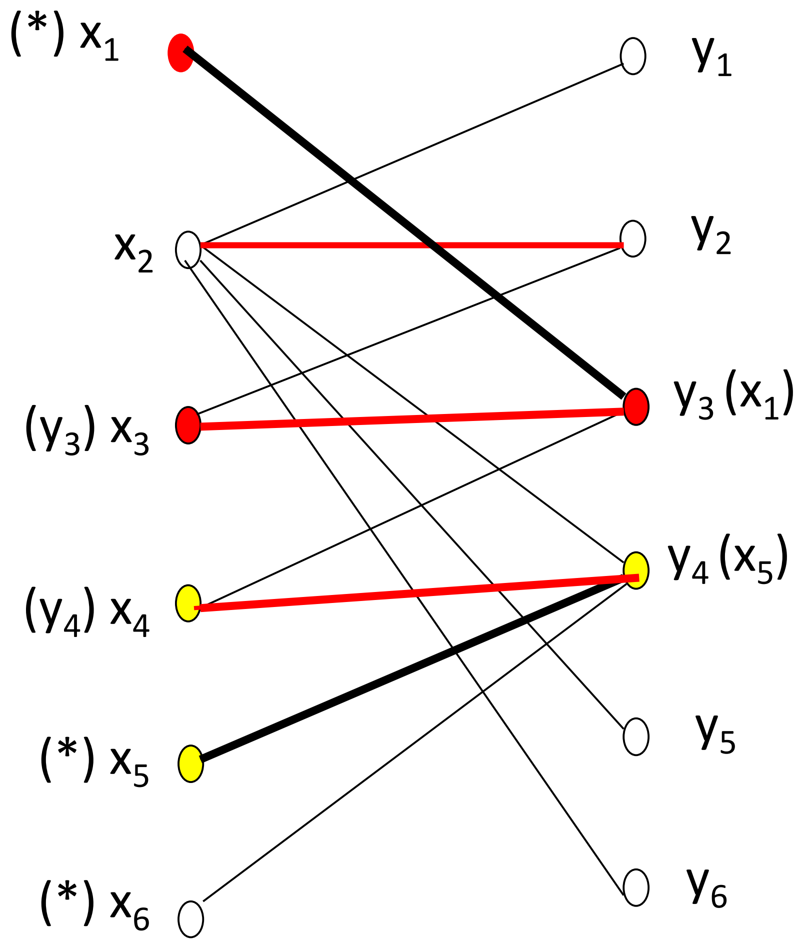
- **Repeatedly applying matching Algorithm to a bipartite graph produces a matching and a vertex cover of equal size**

Example

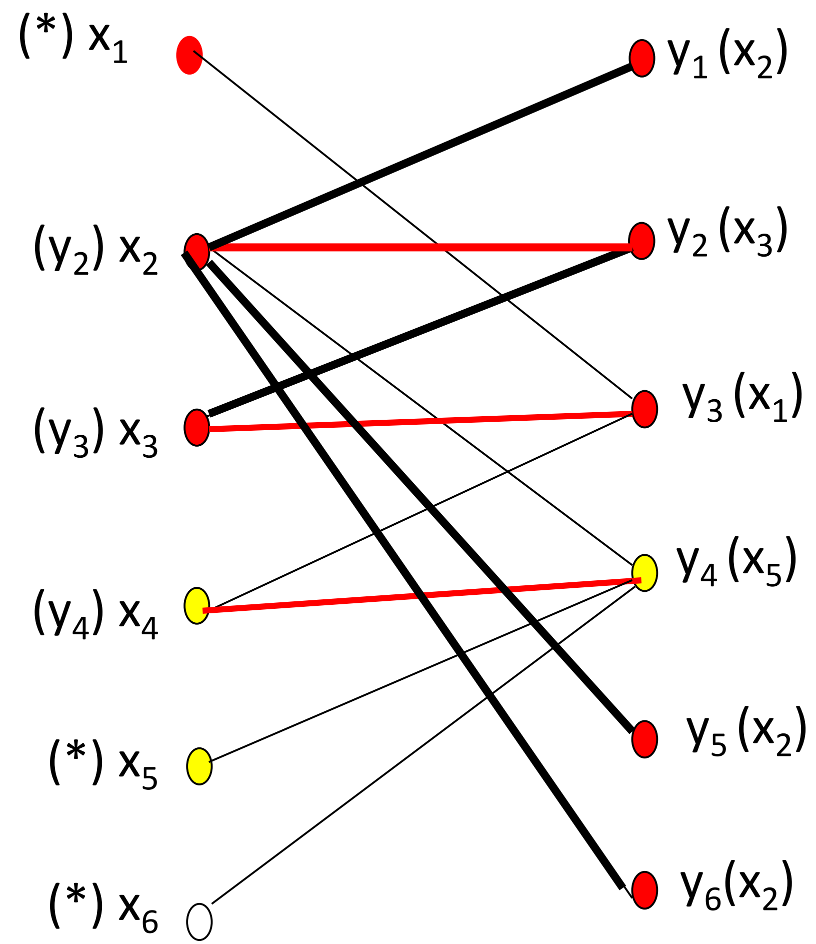
- **Determine a max-matching in the bigraph G by applying the matching algorithm. We input the matching $M^1 = \{(x_2, y_2), (x_3, y_3), (x_4, y_4)\}$ of size 3 and $U = \{x_1, x_5, x_6\}$.**



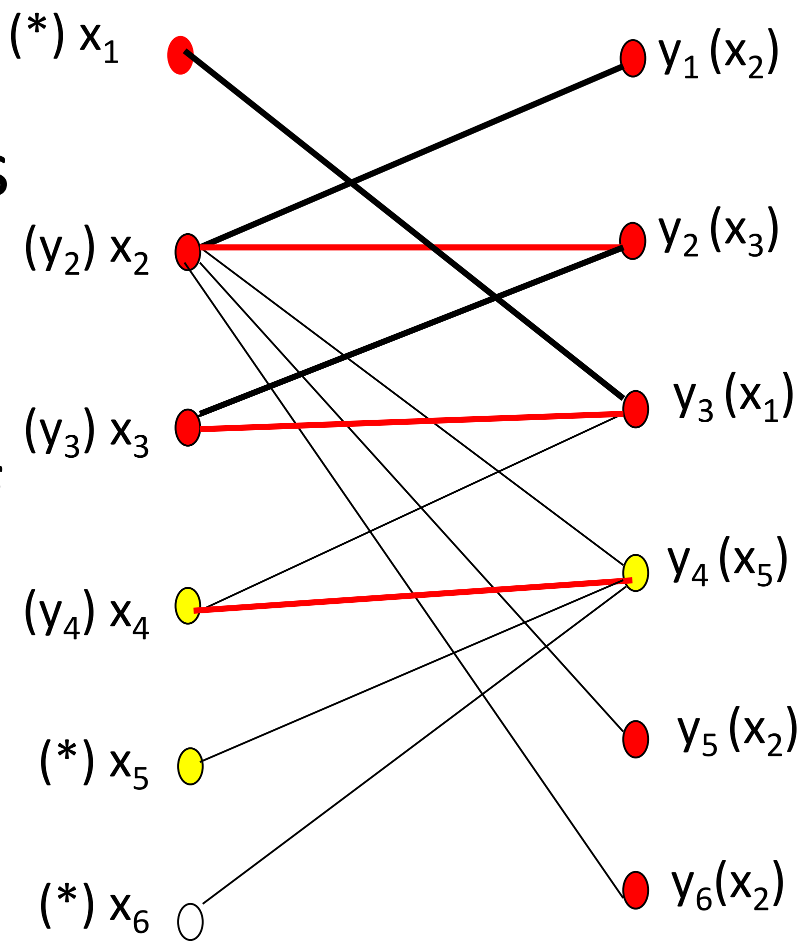
- (i) (0) The vertices x_1, x_5 and x_6 are labeled (*).
- (ii) (2) scan the vertices in U in turn, and label y_3 with (x_1) and y_4 with (x_5). Since all vertices incident to x_6 already have a label, no vertex of Y get label (x_6).
- (iii) (4) scan the vertices y_3 and y_4 labeled in (ii), and label x_3 with (y_3) and x_4 with (y_4).



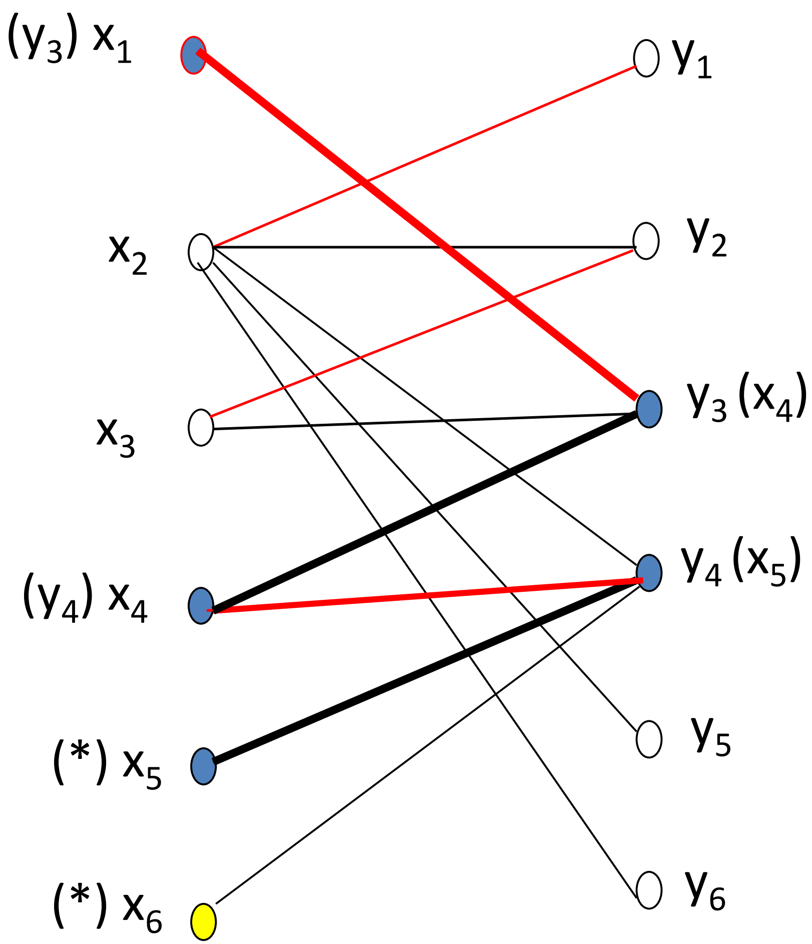
- (iv) (2) we scan the vertices x_3 and x_4 labeled in (iii), and label y_2 with (x_3)
- (v) (4) scan the vertex y_2 labeled in (iv), and label x_2 with (y_2)
- (vi)(2) scan the vertex x_2 labeled in (v), and label y_1, y_5 and y_6 with (x_2)
- (4) scan the vertices y_1, y_5 y_6 labeled in (vi), and find that no new labels are possible.



- We have achieved breakthrough. We find the M^1-augmenting path $r = y_1x_2y_2x_3y_3x_1$ using the labels as a guide. Then
- $M^2 = \{(x_4, y_4), (y_1, x_2), (y_2, x_3), (y_3, x_1)\}$ is a matching of four edges.

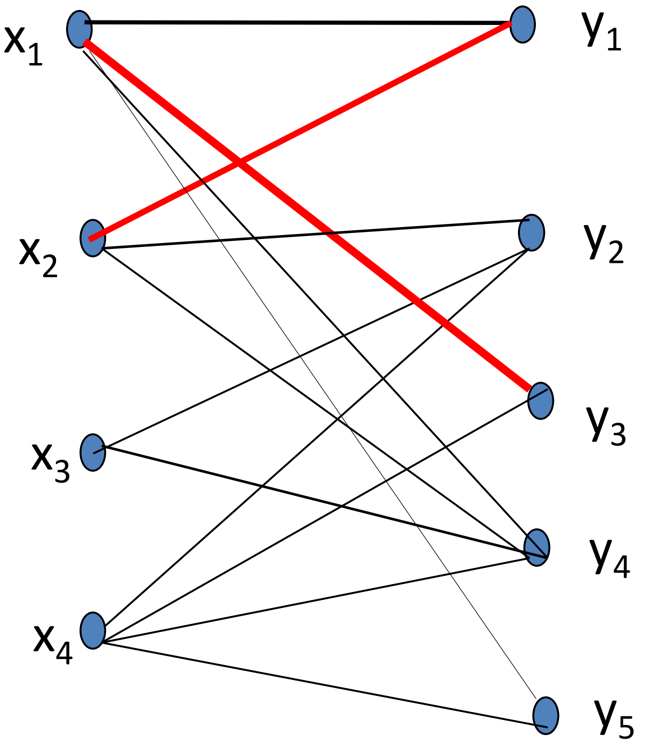


- Continue to apply the algorithm to M^2. The resulting labeling of the vertices is as shown. In this case breakthrough has not occurred. Hence, M^2 is a max-matching of size 4, and the set
- $S = \{x_2, x_3, y_3, y_4\}$ of size 4, consisting of unlabeled vertices of X and labeled vertices of Y is a min-cover.



Exercise

- Determine the max-matching of the right graph by applying the matching algorithm. We choose the red edges and obtain a matching M^1.



Time Complexity of the matching Algorithm

- Let G be an X,Y-bigraph with n vertices and m edges.

1. The matching algorithm executes at most n/2 times since c(G)<=n/2.
2. Each execution of the algorithm considers each edge at most once.

$\Rightarrow$ The execution time of the algorithm is O(mn).