the given matrix and imagine the difficulty with a matrix of size 50.

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Hall's theorem and bipartite matching algorithm

Hall's theorem gives a necessary and sufficient condition for the existence of a perfect matching in a bipartite graph. The proof we give here uses the idea of finding an 'augmenting path' (we will describe that in due time). This helps us to write an algorithm to find a maximum matching.

[22.17] **Notations**

a) For a set of vertex S, let $N(S)$ mean the set of neighbors of S (set of vertices that are adjacent to some vertex of S).

b) Let M be a matching and suppose that u and v are two nonmatching vertices, that is, neither u nor v is involved in the matching. A path of the form

$$[u = u_0, u_1, u_2, \ldots, u_{2k+1} = v],$$

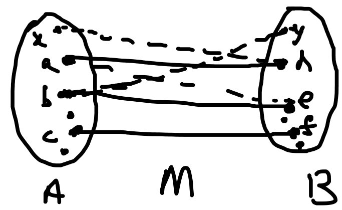
where the edges $[u_1, u_2], [u_3, u_4], \ldots, [u_{2k-1}, u_{2k}]$ are matching edges, is called an AUGMENTING PATH.

[22.18] **Hall's Theorem** Let G be a bipartite graph with parts A and B where $|A| = |B| = n$. Then G has a perfect matching iff G satisfies the condition '$|N(S)| \geq |S|$ for each $S \subseteq A$'.

Proof.

$\dfrac{\text{matched vertices}}{\text{unmatched vertices}}$

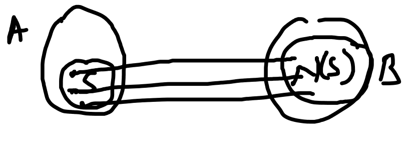
$x, y \longrightarrow$



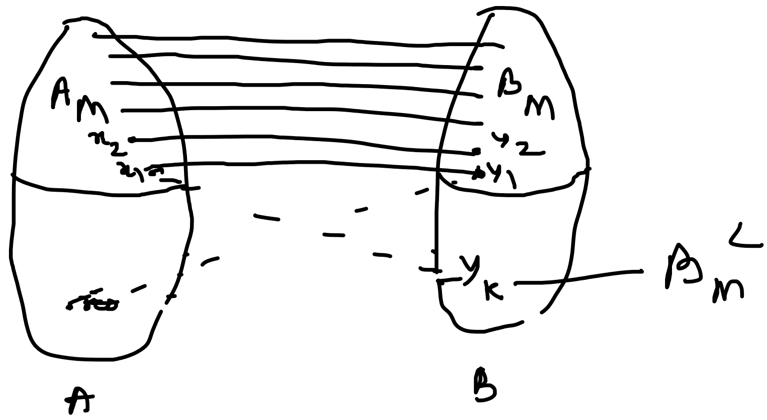
Assume such a path is there. $x - d \# a - e \# b - y \longrightarrow$ an M-augmenting path.

$$M = \{[a,d], [b,e], [c,f]\}$$

$$M' = M \setminus \{[a,d], [b,e]\} \cup \{[x,d], [a,e], [b,y]\}$$
$$= \{[c,f], [x,d], [a,e], [b,y]\}$$



Assume that $|N(S)| \geq |S|$, $\forall\, S \subseteq A$, $S \neq \emptyset$. Assume that G does not have a perf match. Let M be a maximum match. Let $x_0 \in A$ be an unmatched vertex



Take $S = \{x_0\}$. As $|N(S)| \geq 1$, x_0 is adj some vertex $y_1 \in B$.

If y_1 is unmatched, the $M \cup \{[x_0, y_1]\}$ is a larger matching. not possible.

so $y_1 \in B_M$. Let $x_1 \to$ match partner of y_1. Consider $S = \{x_0, x_1\}$.

As $|N(S)| \geq 2$, let $y_2 \neq y_1$ be another vertex in $N(S)$.

The necessity of the condition is easy it see. To show the sufficiency, suppose that the condition holds and still G does not have a perfect matching. Take a maximum matching M and an unmatched vertex $x_0 \in A$. Put the current group $CG = \{x_0\}$.

As $|N(CG)| \geq |CG|$, we see that CG must have at least one neighbor y_1 in B.

If y_1 is an unmatched vertex, then $M \cup \{[x_0, y_1]\}$ is a larger matching, a contradiction.

So let y_1 be a matched vertex. Assume it is matched to x_1. Put $CG = \{x_0, x_1\}$.

As $|N(CG)| \geq |CG|$, we see that CG must have at least two neighbors in B and hence CG has a new neighbor, say y_2.

If y_2 is an unmatched vertex, then we will show how to find a larger matching, and supply a contradiction.

So let y_2 be a matched vertex, which is matched to x_2. Put $CG = \{x_0, x_1, x_2\}$.

We cannot continue this indefinitely, as at each stage, the size of CG is increasing and it cannot exceed the order of G.

Hence, there will come a first stage $CG = \{x_0, x_1, \ldots, x_k\}$ such that CG has an unmatched neighbor y_{k+1}.

Now, from y_{k+1}, we trace back and find a path to x_0.

$$y_{k+1}, x_{i_1}, y_{i_1}, \ldots, x_{i_l}, y_{i_l}, x_0.$$

Notice that, this an augmenting path. Consider

$$M \setminus \{[x_{i_1}, y_{i_1}], \ldots, [x_{i_l}, y_{i_l}]\} \cup \{[y_{k+1}, x_{i_1}], [y_{i_1}, x_{i_2}], \ldots, [y_{i_l}, x_0]\}.$$

This is a larger matching. So we have a contradiction. ∎

23 Lecture 23

The star-prime algorithm or the bipartite matching algorithm

Based on the idea in the proof of Hall's theorem, we supply an algorithm to find a maximum matching in a bipartite graph. In graph theory, it is called 'bipartite matching algorithm'. This involves finding an augmenting path. We supply the algorithm in the form of finding a maximum independent set of zeros in a matrix.

An algorithm to find a maximum matching.

b1) Star a zero in the matrix if there is no ~~starred zero~~ zero-star in its column or row. Do it for all zeros. (We will read the entries of the matrix from the top-left, finish the first row, then the second row and so on, to mark these zeros. It makes it easier to discuss.)

b2) Cover each COLUMN containing a zero-star (by a line segment or dashed line segment).

b3) If all zeros of the matrix are covered then conclude that these zero-stars give us a maximum independent set.[a] STOP.

b4) There is an uncovered zero. Prime it. If there is no zero-star in its row, then go to b5). If it has a zero-star in its row, then cover the row of the zero-star and uncover the column.[b] In this process, the zero stars remain covered. Also the primes discovered till now are covered. Go to b3).

b5) We have got a zero-prime and in its row there is no 0^*. (This will lead us to a prime-star sequence or equivalently to an augmenting path.)

b51) If there is no 0^* in its column, then convert this $0'$ to a 0^*. (Thus, we have found a $0'$ in whose row or column, there is no 0^*. So we can safely add this one to our set of zero-stars. We have got a larger set of zero-stars.) Erase all primes and lines. Go to b2).

b52) There is a zero-star 0_1^* in its column. Initially this column was covered, but now it is not. So there is a zero-prime $0_2'$ in the row of 0_1^*. Suppose that there is a 0_3^* in the column of $0_2'$. As 0_2 was initially covered by the column line of 0_3^* and after that it was available as an uncovered zero, we see that the current line covering 0_3^* must be a row (as we uncover columns only). So there is a $0_4'$ in the row of 0_3^*. We continue to get a sequence $0', 0_1^*, 0_2', \cdots, 0_{2r}'$, where the column of $0_{2r}'$ does not have a 0^*. This an alternating prime-star-prime sequence. Interchange stars and primes of this sequence. Erase all primes and lines. (We have got a larger set of zero stars.) Go to b2).

[a] These zero-stars are independent by construction. Assume that we have used k lines to cover all the zeros. Consider any set of more than k zeros. Then by php, some two of them will lie on one of these lines. So they cannot be independent.

[b] Thus, this process can be taken at best n many times.

[23.1] **Theorem** The algorithm works correctly.

Proof. By construction, the set of zero-stars, at any stage is independent throughout the algorithm. Once the algorithm starts it can only stop at b3).

When the algorithm enters b3), we have only three possibilities.

Possibility 1) It has covered all the zeros by k lines, where k is the number of zero-stars at that stage. This means we cannot get more than k zero-stars. So it is maximum.

Possibility 2) We find an uncovered zero, with no zero-star in its row. In this case we go to b5) and get a larger set of independent zero-stars and start from b2) again. This step can also be used only finitely many times.

Possibility 3) We find an uncovered zero, with a zero-star in its row. In this case we change the column line segment to a row line segment. (This changes the line structures, but this can also be taken at most n times).

So essentially, when the algorithm stops, it has to stop at b3), with Possibility 1). ■

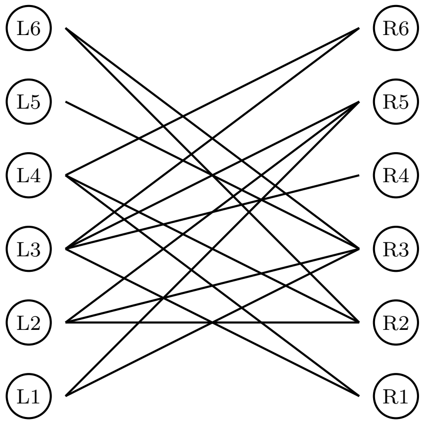
[23.2] <u>**Example**</u> `(Illustration of the star-prime algorithm.)` Consider $A = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$.

Use the star-prime algorithm to find a maximum independent set of zeros. Strictly follow the rule that the computer reads from the top-left position that is, $a_{11}, \ldots, a_{1n}, a_{21}, \ldots$. If you have used a set of lines for your purpose, you need not draw the table and lines. In stead write the lines, like, Stage k: R_1, R_3, C_1, C_4.

Answer.

$$A = \begin{bmatrix} 1 & 1 & 0 & 1 & 0^* & 1 \\ 1 & 0^* & 0 & 1 & 0 & 1 \\ 0^* & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0^* \\ 1 & 1 & 0^* & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

<u>Stage 1</u>	<u>Primed</u>
$C_1 C_2 C_3 C_6$	a_{15}
$C_1 C_2 R_1 C_6$	a_{23}
$C_1 R_2 R_1 C_6$	a_{34}
$R_3 R_2 R_1 C_6$	a_{41}
$R_3 R_2 R_1 R_4$	a_{53}



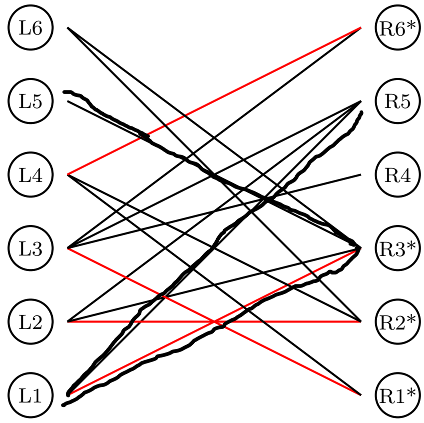
Prime-star-~~prime~~ sequence

$0'_{5,3}$, $0^*_{1,3}$, $0'_{1,5}$ $\longrightarrow$ $0^*_{5,3}$, $0'_{1,3}$, $0^*_{1,5}$

<u>New stars</u> $\cancel{0^*_{1,3}}$, $0^*_{2,2}$, $0^*_{3,1}$, $0^*_{4,6}$, $0^*_{5,3}$, $0^*_{1,5}$

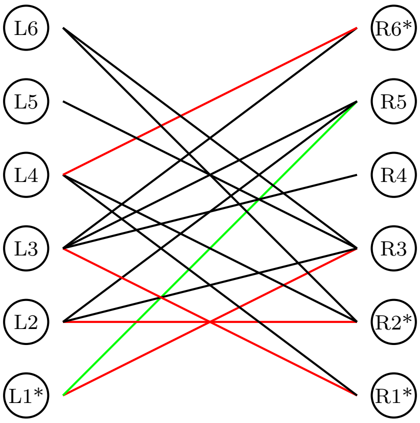
Initial $*$-zeros.

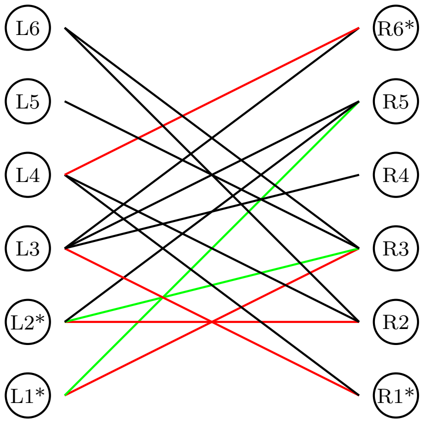
$$A = \begin{bmatrix} 1 & 1 & 0^* & 1 & 0 & 1 \\ 1 & 0^* & 0 & 1 & 0 & 1 \\ 0^* & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0^* \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

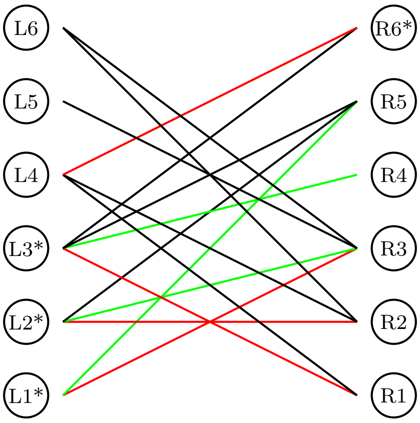


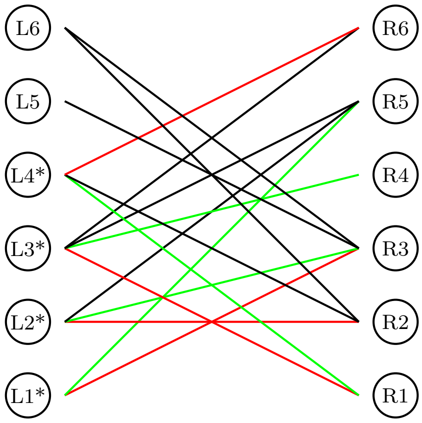
Stage	lines	′-zero
1.1	C_1, C_2, C_3, C_6	a_{15}
1.2	C_1, C_2, R_1, C_6	a_{23}
1.3	C_1, R_2, R_1, C_6	a_{34}
1.4	R_3, R_2, R_1, C_6	a_{41}
1.5	R_3, R_2, R_1, R_4	a_{53}

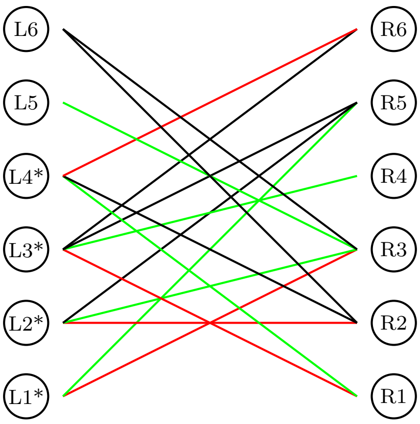
Prime an edge not covered by the stars.

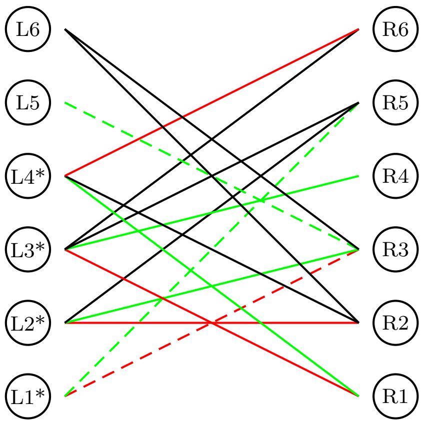








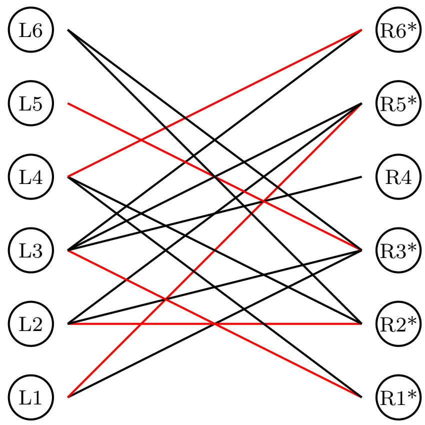




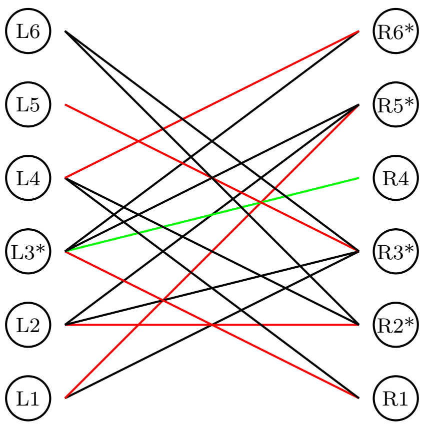
′-∗-sequence: a_{53}, a_{13}, a_{15}

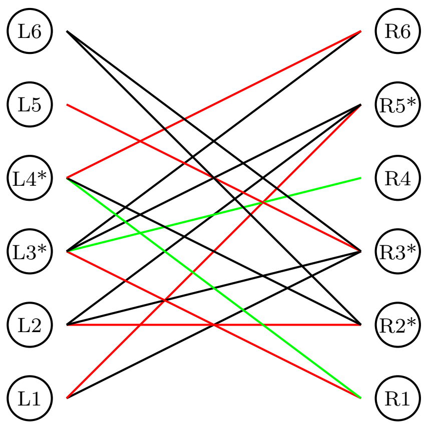
New $*$-zeros

$$A = \begin{bmatrix} 1 & 1 & 0 & 1 & 0^* & 1 \\ 1 & 0^* & 0 & 1 & 0 & 1 \\ 0^* & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0^* \\ 1 & 1 & 0^* & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$



Stage	lines	$'$-zero
2.1	C_1, C_2, C_3, C_5, C_6	a_{34}
2.2	R_3, C_2, C_3, C_5, C_6	a_{41}
2.3	R_3, C_2, C_3, C_5, R_4	none





The maximum independent set of zeros is $\{a_{15}, a_{22}, a_{31}, a_{46}, a_{53}\}$ and a minimum set of lines covering all the zeros is $\{C_2, C_3, C_5, R_3, R_4\}$.

In the graph, we have 5 vertices covering all edges and so we can at best get a matching of size 5. The 5 edges selected is a matching. Hence it is a maximum matching.

[23.3] **Corollary** Let A be a square matrix. Then the maximum number of independent zeros is the same as the minimum number of lines required to cover all the zeros of the matrix.

Proof.

Let the maximum number of independent zeros be k. Then by definition we would require at least k lines to cover all the zeros. By the bipartite matching algorithm, we can actually get k lines that cover all the zeros.■

[23.4] **Class workout** Find a maximum set of independent zeros in the following matrix.

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Answer.