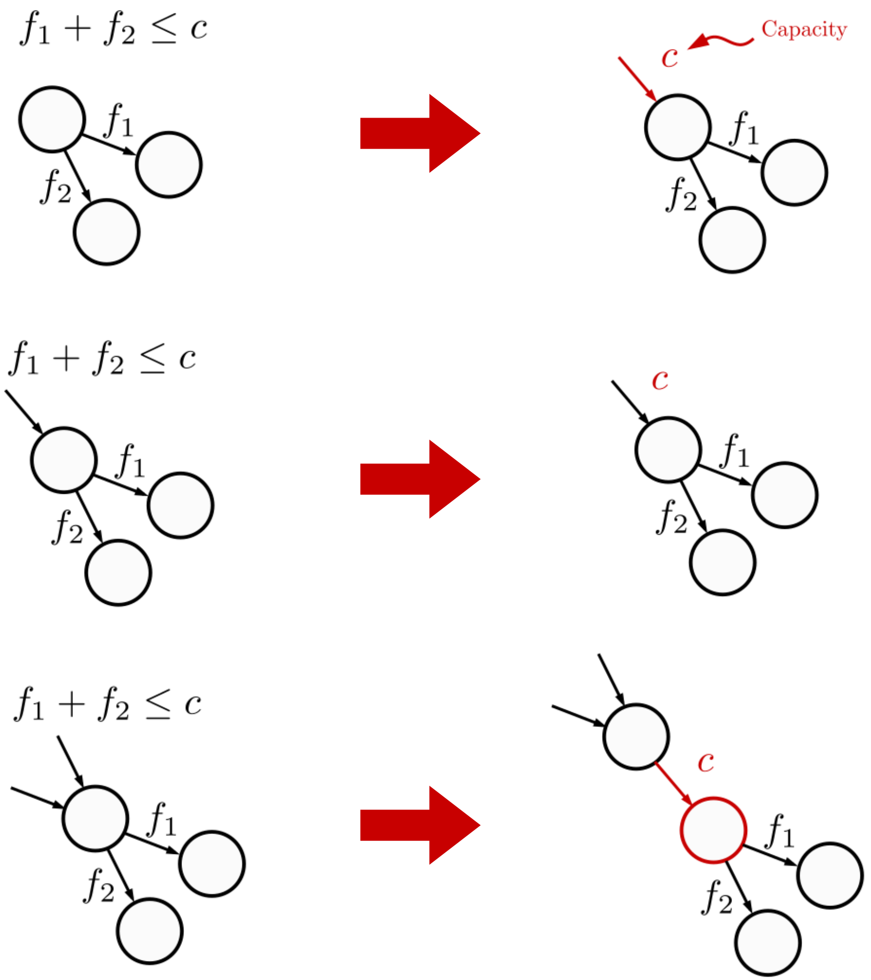
网络流习题课

2021-12-30

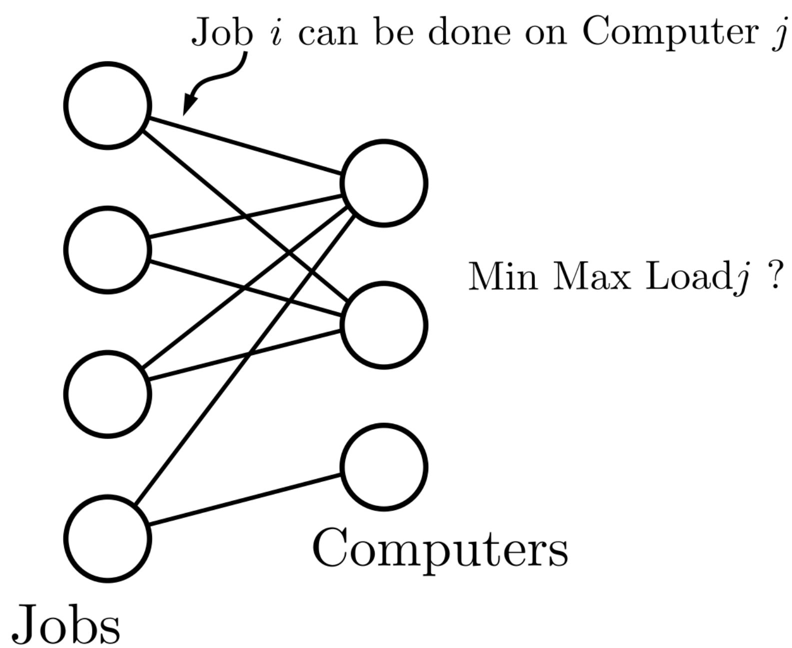
- Techniques
  - From *graph problem with constrains* to **network flow problem**



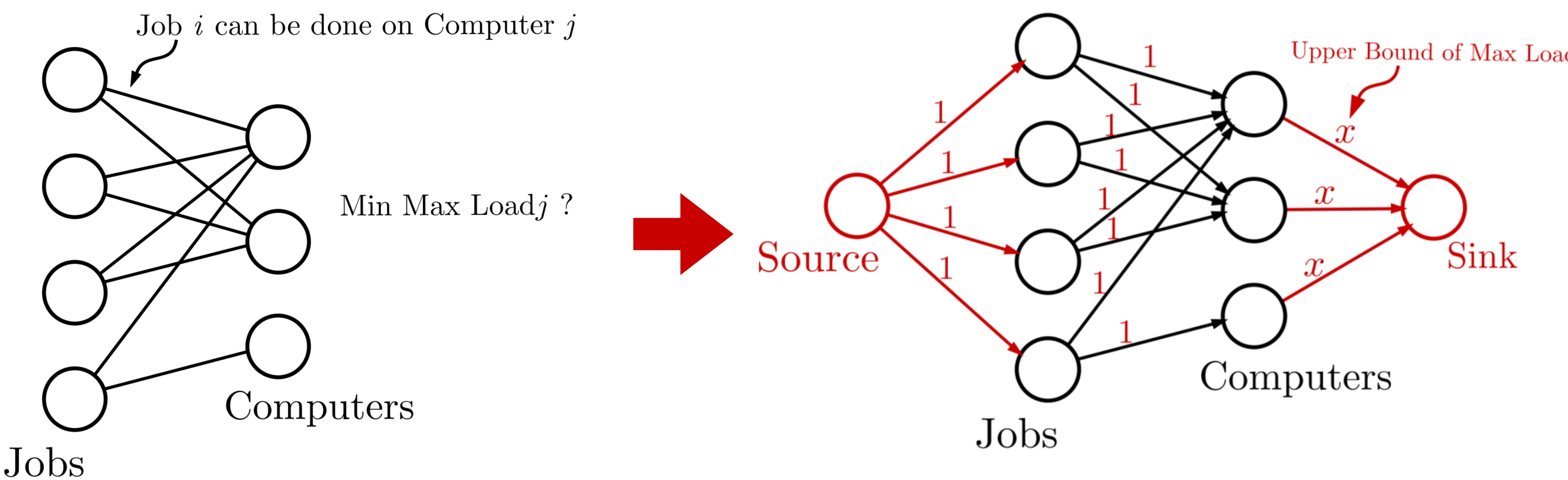
1 Load balance

You have some different computers and jobs. For each job, it can only be done on one of two specified computers. The load of a computer is the number of jobs which have been done on the computer. Give the number of jobs and two computer ID for each job. You task is to minimize the max load.

(hint: binary search)



- Set Max Load as **x** & Convert to Max Flow Problem
- Binary Search on **x** in range [1, #Jobs]
  - Until when x, there is a feasible solution
    and when x-1, no feasible solution



2 Matrix

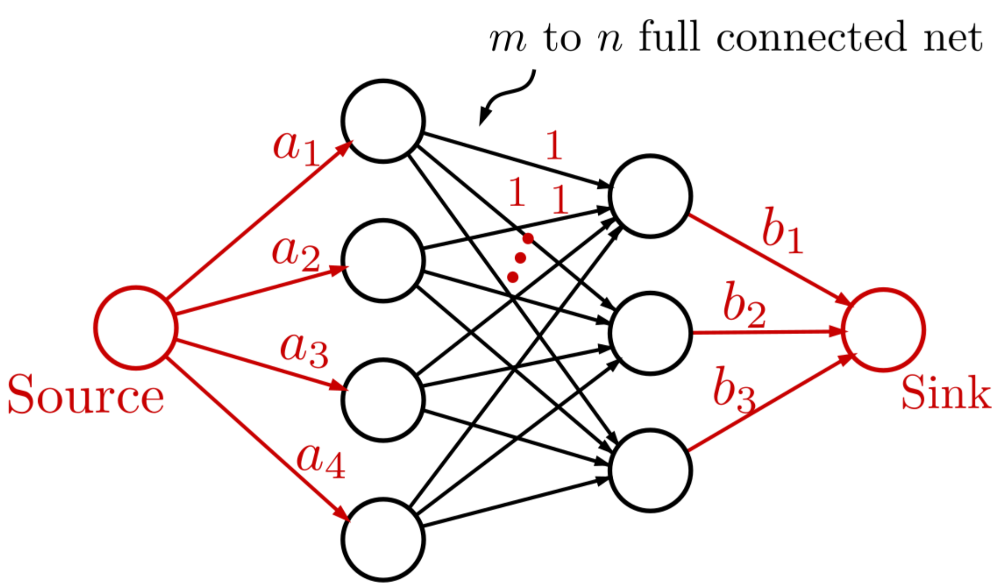
For a matrix filled with 0 and 1, you know the sum of every row and column. You are asked to give such a matrix which satisfys the conditions.

$$
\begin{array}{rcc}
 & & \text{Sum of Rows} \\
X = & \begin{pmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & x_{m3} & \cdots & x_{mn} \end{pmatrix} & \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} \\
\text{Sum of Columns} & \begin{pmatrix} a_1 & a_2 & a_3 & \cdots & a_n \end{pmatrix} &
\end{array}
$$

find X

- Convert to Max Flow Problem

$$
\begin{array}{rcc}
 & & \text{Sum of Rows} \\
X = & \begin{pmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & x_{m3} & \cdots & x_{mn} \end{pmatrix} & \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} \\
\text{Sum of Columns} & \begin{pmatrix} a_1 & a_2 & a_3 & \cdots & a_n \end{pmatrix} &
\end{array}
$$



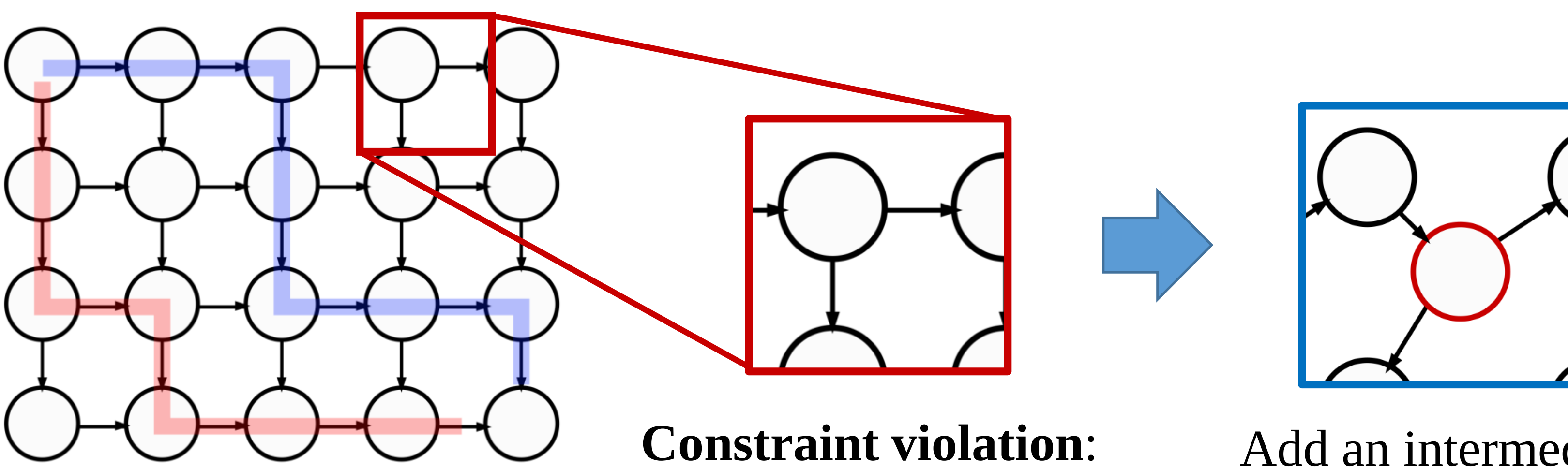
3 Problem Reduction

There is a matrix with numbers which means the cost when you walk through this point. you are asked to walk through the matrix from the top left point to the right bottom point and then return to the top left point with the minimal cost. Note that when you walk from the top to the bottom you can just walk to the right or bottom point and when you return, you can just walk to the top or left point. And each point CAN NOT be walked through more than once.

$$\begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} \\ c_{31} & c_{32} & c_{33} & c_{34} & c_{35} \\ c_{41} & c_{42} & c_{43} & c_{44} & c_{45} \end{pmatrix}$$

- Try to Convert to Network Flow Problem
  - 1. Direction is not important
  - 2. Need to consider Only once walk constrain

$$\begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} \\ c_{31} & c_{32} & c_{33} & c_{34} & c_{35} \\ c_{41} & c_{42} & c_{43} & c_{44} & c_{45} \end{pmatrix}$$



Constraint violation:
At each position only one way

Add an intermediate node!

- Convert to Net Flow Problem
  - Then treat it as Max-Flow-Min-Cost Problem

$$\text{Cost}_{\text{edge}} = \begin{cases} c_{ij} & \text{if edge going to black node}_{ij} \\ 0 & \text{Otherwise (edge going to new red node)} \end{cases}$$

$$\text{Capacity}_{\text{edge}} = \begin{cases} 2 & \text{if edge is outgoing edge of source} \\ 2 & \text{if edge is incoming edge of sink} \\ 1 & \text{Otherwise} \end{cases}$$