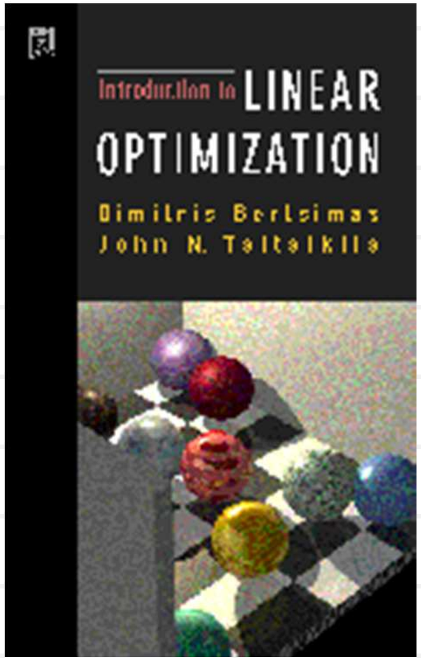
Lecture 8: Integer linear programming.

Source

Coursera course on Discrete Optimization



Bertsimas, Tsiliklis, *Introduction to Linear Optimization*, chapter 10, 11-2

http://www.sce.carleton.ca/faculty/chinneck/po.html chapter 12,13

Planning the optimal study plan

- You are given a list of courses and their schedule
- Each course has a certain attractiveness (can be negative)
- There are constraints on credits you have to get
- There are constraints on how many classes you can take in each term
- There are prerequisites

Binary integer linear program

Binary ILP is a systematic approach for modeling such problems:

i – courses, *j* – terms

$x_{i,j} \in \{0;1\}$ whether we take *i*th course in the *j*th term

$$\max_x \sum_{i,j} a_{i,j} x_{i,j}$$ maximize total attractiveness of the schedule

s.t. $$\sum_j x_{i,j} \leq 1$$ cannot take the same course twice

$$x_{i,j} \leq \sum_{k=1}^{j-1} x_{i',k}$$ *i'* is a prerequisite for *i*

$$\sum_{i,j} c_{i,j} x_{i,j} \geq C$$ enough credits to graduate

$$\sum_i l_i x_{i,j} \leq L_j$$ you have to survive it

General patterns in binary ILPs

$x_i \in \{0; 1\}$

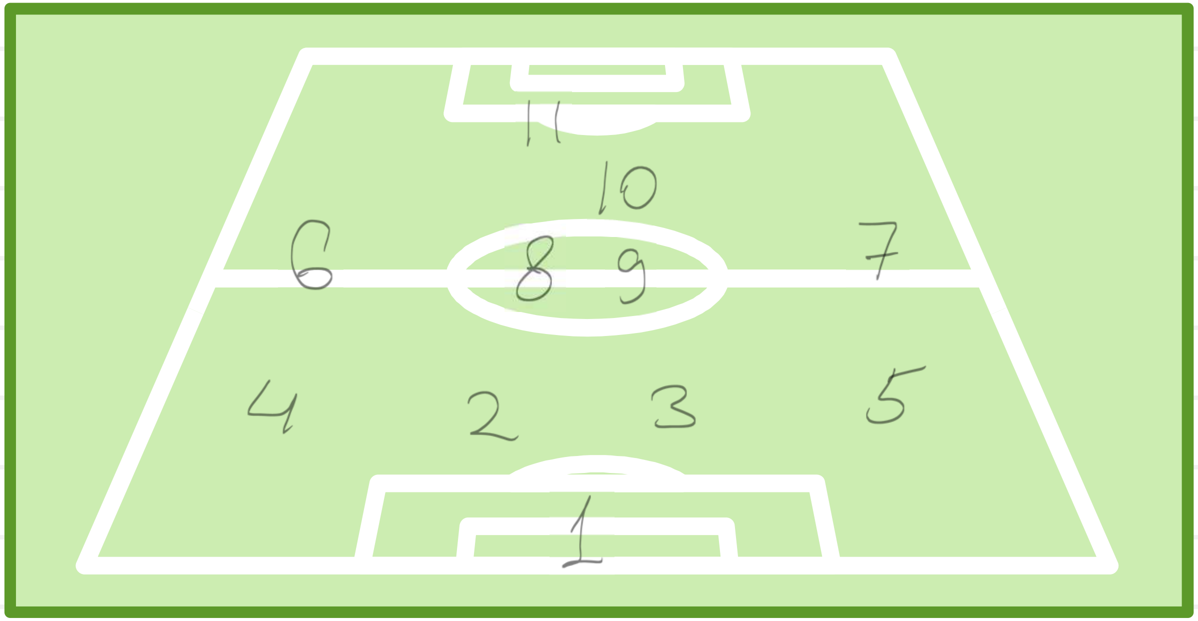
$\sum_{i \in I} x_i \leq 1$ "at most one is on"

$\sum_{i \in I} x_i \geq 1$ "at least one is on"

$\sum_{i \in I} x_i = 1$ "exactly one is on"

$x_i \leq x_j$ "*i* th variable can be on only when *j*th variable can be on"

Team selection



20 players

a_{ij} suitability of the *i*th player for *j*th position

How to pick the optimal starting team?

Team selection

a_{ij} suitability of the *i*th player for the *j*th position

x_{ij} whether the *i*th player is chosen for the *j*th position

...wait, but players 12 and 17 are terrific together!

$$\max \sum_{i=1}^{20} \sum_{j=1}^{11} a_{ij} x_{ij} + w \cdot z$$

$$\text{s.t.:} \quad \sum_{i=1}^{20} x_{ij} = 1$$

$$\sum_{j=1}^{11} x_{ij} \le 1 \qquad z \le \sum_{j=1}^{11} x_{12,j} \qquad z \le \sum_{j=1}^{11} x_{17,j}$$

$$x_{ij} \in \{0;1\} \qquad z \in \{0;1\}$$

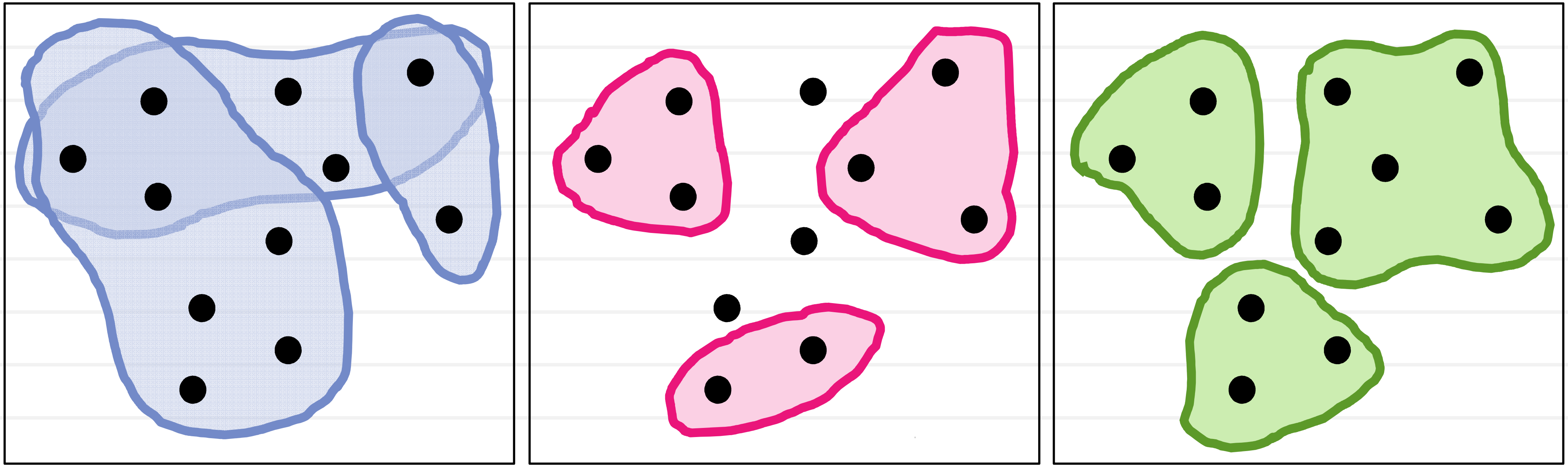
Set problems

$$\min \ \omega^T x$$

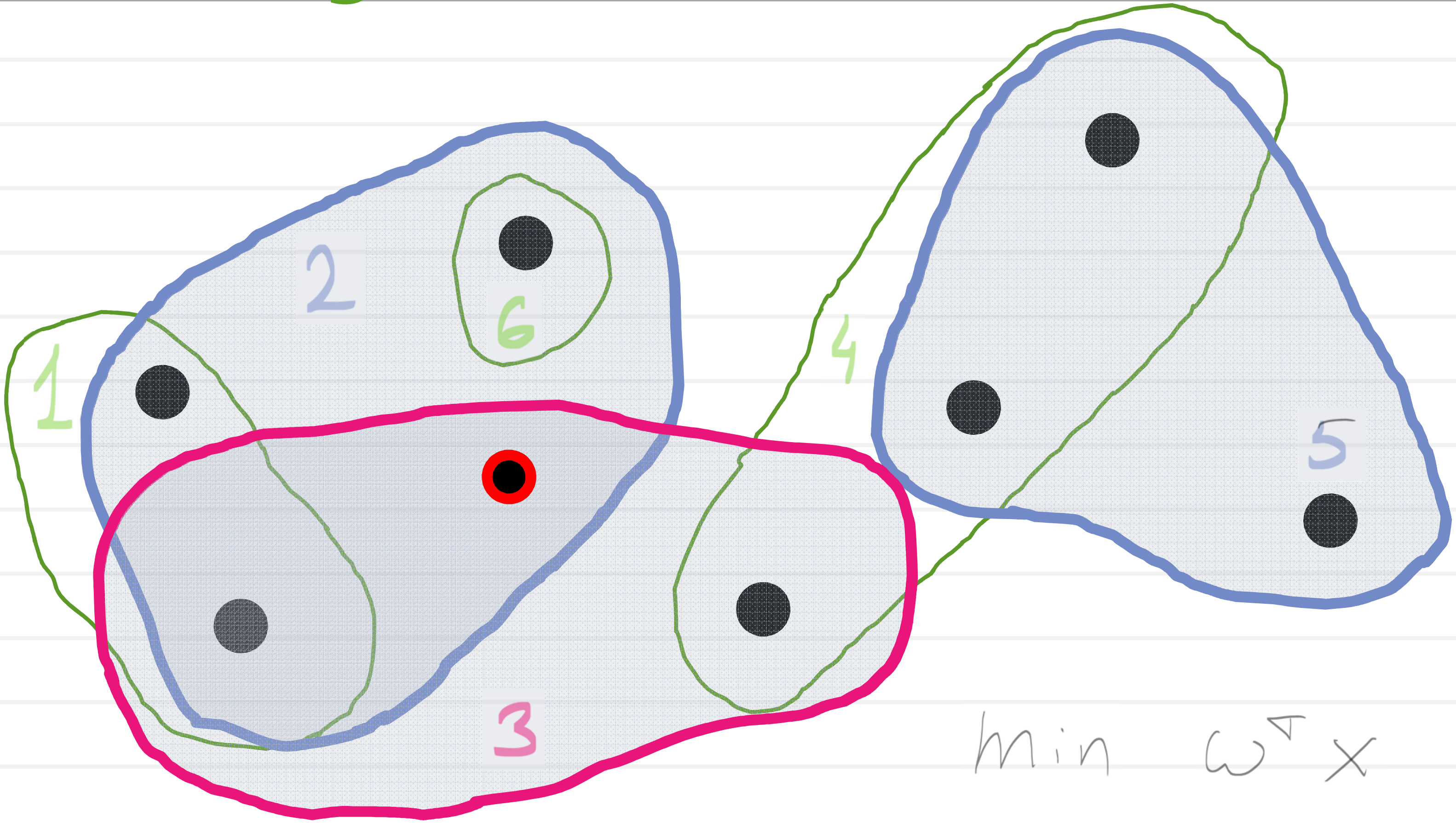
$$\text{s.t.:} \quad Ax \ \begin{cases} \geq 1 & \text{Cover} \\ \leq 1 & \text{Packing} \\ = 1 & \text{Partition} \end{cases}$$

$$x_i \in \{0; 1\}$$

$$A = \begin{array}{cl} \begin{bmatrix} 0 & 1 & 0 & \cdots & 1 \\ 0 & 0 & 1 & \cdots & 0 \\ & & \cdots & & \\ 1 & 1 & 0 & \cdots & 1 \end{bmatrix} & \textit{points} \\ \textit{sets} & \end{array}$$



Minimal weight set cover and ILP



$$x_2 + x_3 \geq 1$$

$$\min \ w^T x$$

$$\text{s.t.}: \forall j \sum_{i : v_j \in M_i} x_i \geq 1$$

$$x_i \in \{0; 1\}$$

Facility location

$$\min_{x,y} \sum_{i=1}^{F} f_i y_i + \sum_{i=1}^{F} \sum_{j=1}^{C} a_{ji} \cdot x_{ji}$$

$$x_i, y_i \in \{0; 1\} \qquad \forall j \sum_{i=1}^{F} x_{ji} = 1$$

$$\forall ij \quad x_{ji} \leq y_i$$

Travelling salesman problem

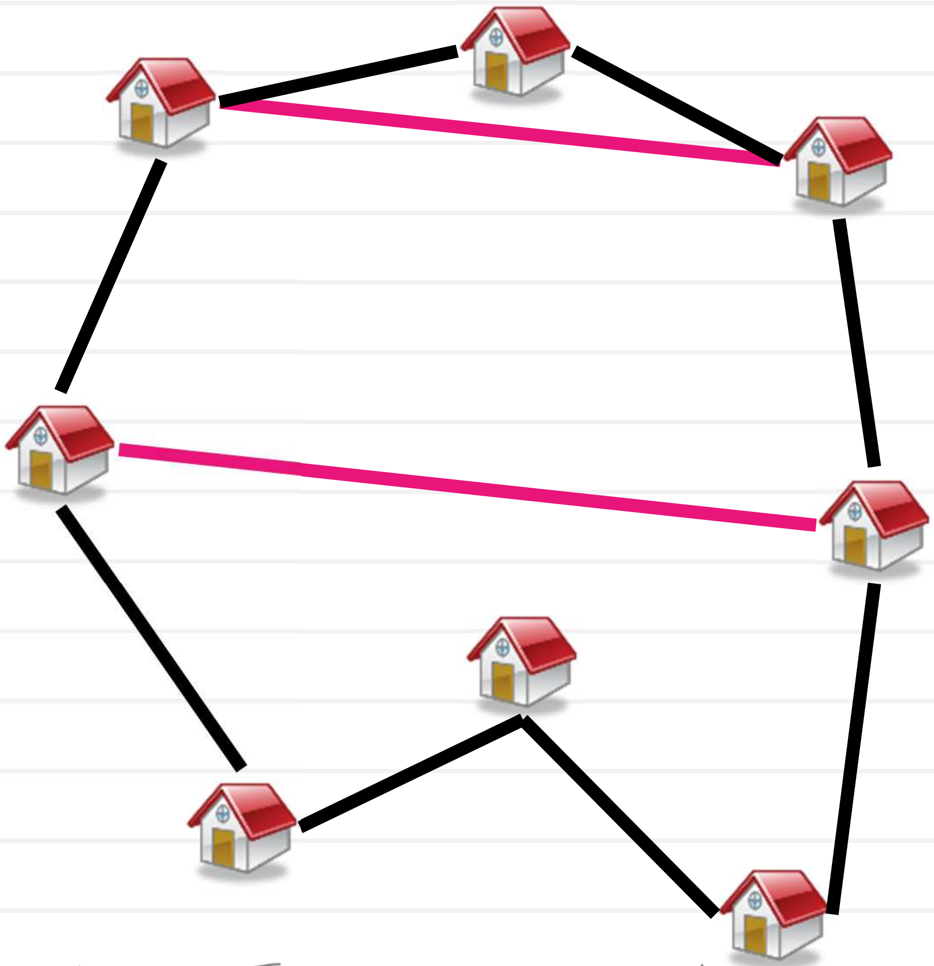
$$\min \; w_{ij} x_{ij}$$

$$\text{s.t.:} \; x_{ij} \in \{0;1\}$$

$$\forall i \; \sum_{j} x_{ij} = 2$$

$$\sum_{i,j} x_{ij} = N$$

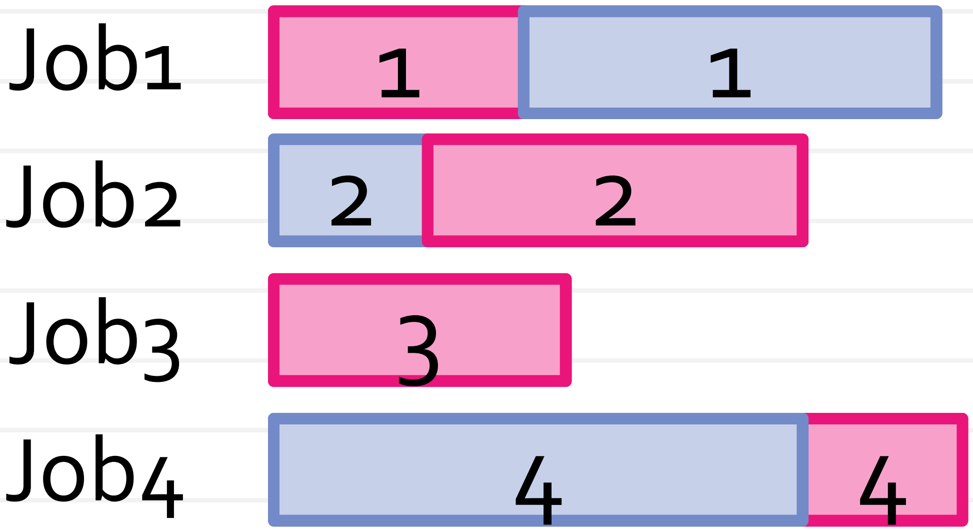
$$\forall C \subset \{1 \ldots N\} \; 0 < |C| < N: \sum_{\substack{i \in C \\ j \in C}} x_{ij} \leq |C| - 1$$



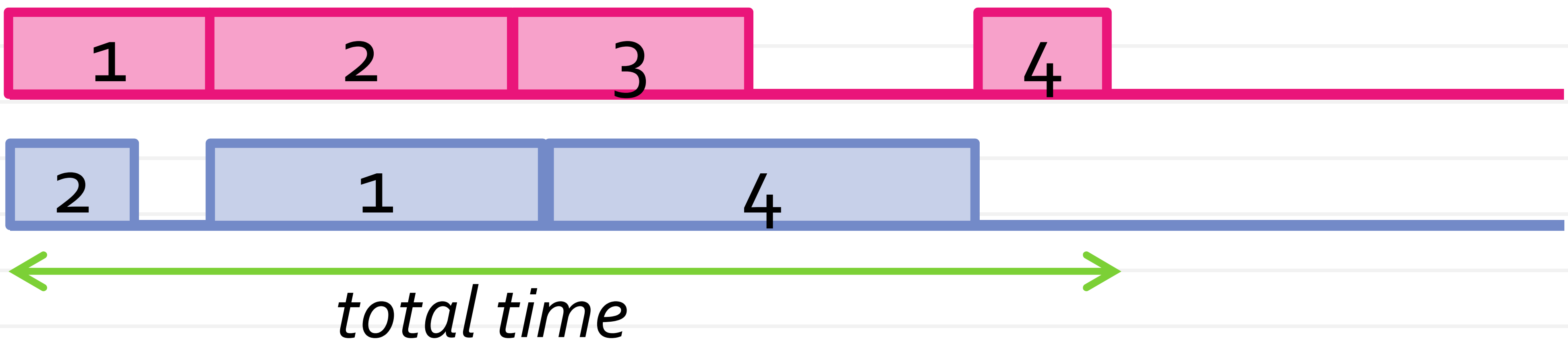
Exponentially many constraints!

Job shop scheduling

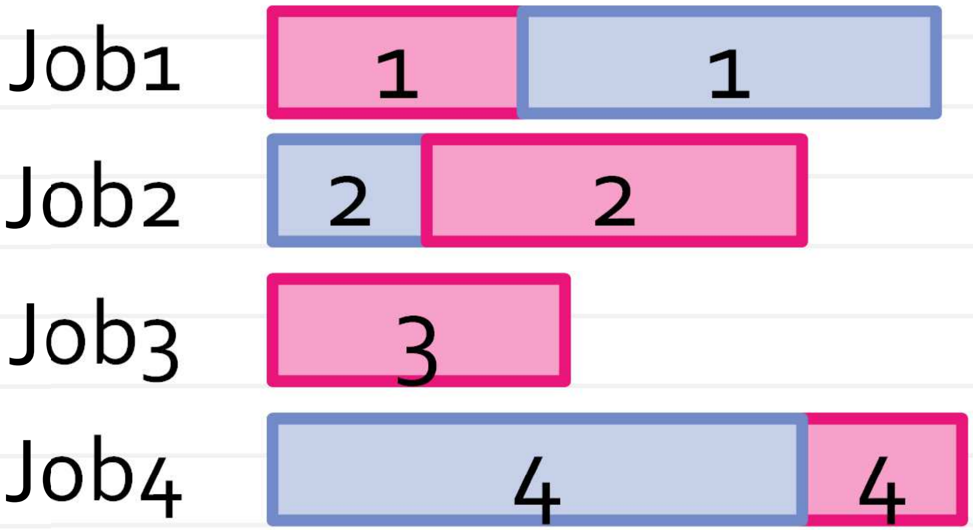
Job shop has two machines ("red" and "blue")



Possible schedule:



Job shop scheduling



$t_{1,1}\ d_{1,1}\ t_{1,2}\ d_{1,2}$

$t_{2,2}\ d_{2,2}\ t_{2,1}\ d_{2,1}$

$t_{3,1}\ d_{3,1}$

$t_{4,2}\ d_{4,2}\ t_{4,1}\ d_{4,1}$

$$\min_{t,z}\ z$$

s.t.:

$$z \geq t_{1,2} + d_{1,2} \qquad t_{1,2} \geq t_{1,1} + d_{1,1}$$

$$z \geq t_{2,1} + d_{2,1} \qquad t_{2,1} \geq t_{2,2} + d_{2,2}$$

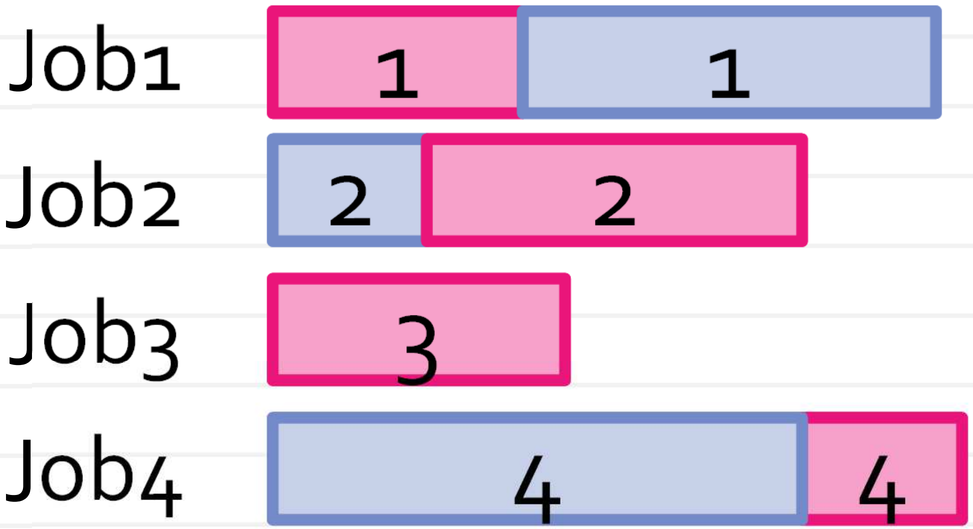
$$z \geq t_{3,1} + d_{3,1} \qquad t_{4,1} \geq t_{4,2} + d_{4,2}$$

$$z \geq t_{4,1} + d_{4,1}$$

$$t_{i,j} \geq 0$$

+ **different** jobs are competing for the same machines

Job shop scheduling



$$\min_{t,z} \; z$$

s.t.: $z \geq t_{1,2} + d_{1,2}$

$z \geq t_{2,1} + d_{2,1}$

$z \geq t_{3,1} + d_{3,1}$

$z \geq t_{4,1} + d_{4,1}$

$t_{i,j} \geq 0$

$$\left[\begin{array}{l} t_{2,1} \geq t_{1,1} + d_{1,1} \\ t_{1,1} \geq t_{2,1} + d_{2,1} \end{array}\right.$$

$$\left[\begin{array}{l} t_{2,2} \geq t_{1,2} + d_{1,2} \\ t_{1,2} \geq t_{2,2} + d_{2,2} \end{array}\right.$$

...

$t_{1,2} \geq t_{1,1} + d_{1,1}$

$t_{2,1} \geq t_{2,2} + d_{2,2}$

$t_{4,1} \geq t_{4,2} + d_{4,2}$

How to handle "either-or" constraints in this case?

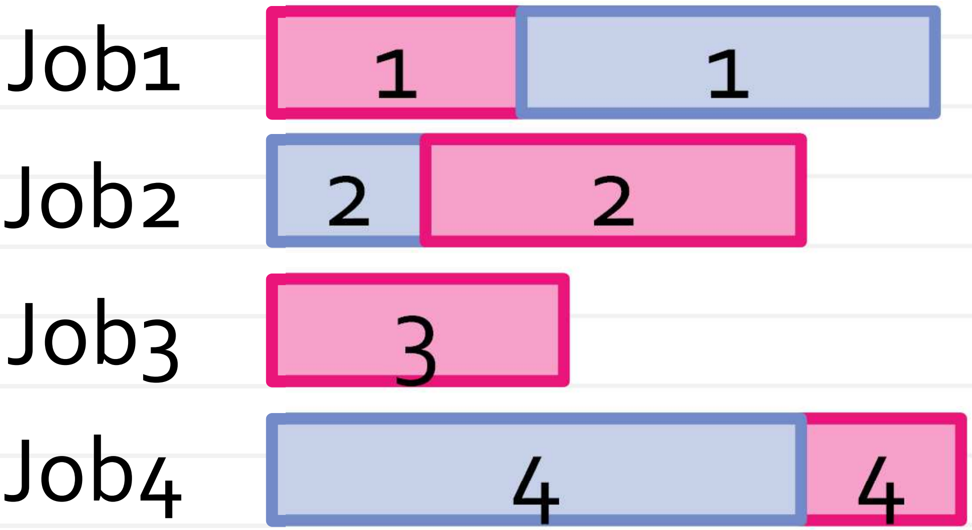
The "Big M" trick

$$\begin{cases} t_{2,1} \geq t_{1,1} + d_{1,1} \\ t_{1,1} \geq t_{2,1} + d_{2,1} \end{cases}$$

M is a big constant
(e.g. bigger than the time needed
for some poor schedule)

$$\begin{cases} M x_{1-2,1} + t_{2,1} \geq t_{1,1} + d_{1,1} \\ M(1 - x_{1-2,1}) + t_{1,1} \geq t_{2,1} + d_{2,1} \\ x_{1-2,1} \in \{0; 1\} \end{cases}$$

Job shop scheduling



$$\min_{t,z,x} \; z$$

$$\text{s.t.:}\quad z \geq t_{1,2} + d_{1,2}$$

$$z \geq t_{2,1} + d_{2,1}$$

$$z \geq t_{3,1} + d_{3,1}$$

$$z \geq t_{4,1} + d_{4,1}$$

$$t_{i,j} \geq 0$$

$$\left[\begin{array}{l} t_{2,1} \geq t_{1,1} + d_{1,1} \\ t_{1,1} \geq t_{2,1} + d_{2,1} \end{array}\right.$$

...

$$t_{1,2} \geq t_{1,1} + d_{1,1}$$

$$t_{2,1} \geq t_{2,2} + d_{2,2}$$

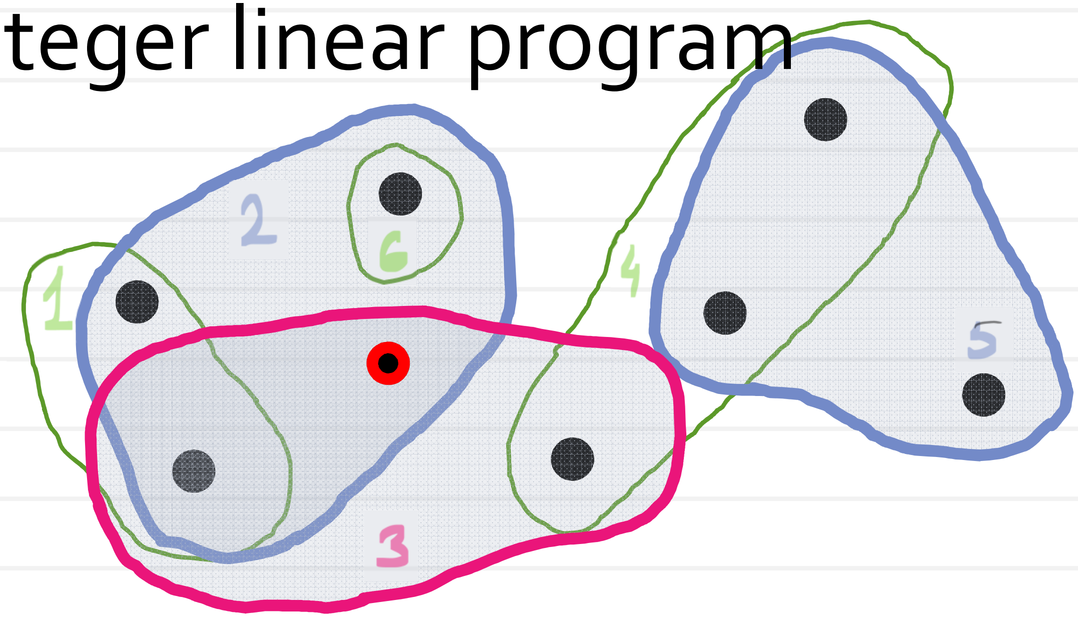
$$t_{4,1} \geq t_{4,2} + d_{4,2}$$

After expressing conflicts via the "Big M" trick, we get a *mixed-integer program* (combines integer and real variables)

LP relaxation

binary integer linear program

$$\min \ w^T x$$
$$\text{s.t.}: \forall j \sum_{i: V_j \in M_i} x_i \geq 1$$
$$x_i \in \{0; 1\}$$



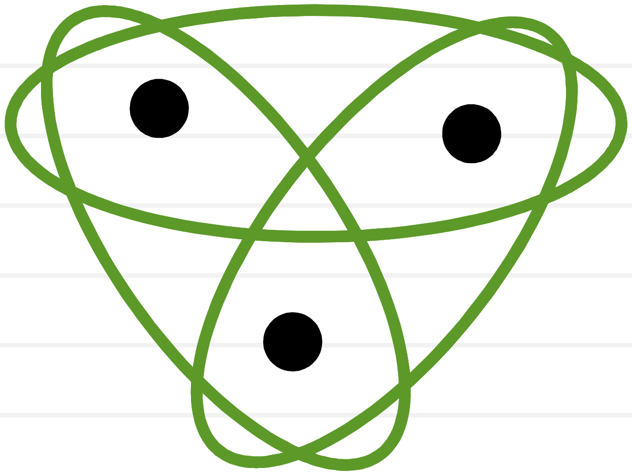
linear program (*relaxation*)

$$\min \ w^T x$$
$$\text{s.t.}: \forall j \sum_{i: V_j \in M_i} x_i \geq 1$$
$$x_i \geq 0$$
$$x_i \leq 1$$

(redundant if $w > 0$)

LP relaxation

- In some cases, the solution will be integer
- In many cases, the solution will be fractional



$$\min\ x_1 + x_2 + x_3$$

$$\text{s.t.:}\quad \begin{aligned} x_1 + x_2 &\geq 1 \\ x_2 + x_3 &\geq 1 \\ x_1 + x_3 &\geq 1 \\ x_1, x_2, x_3 &\geq 0 \end{aligned}$$

$$x_1 = x_2 = x_3 = \frac{1}{2}$$

- No free lunch: the harder the problem, the smaller are our chances to get an integer solution

Facility location: two ILPs

$$\min_{x,y} \sum_{i=1}^{F} f_i y_i + \sum_{i=1}^{F} \sum_{j=1}^{C} a_{ji} \cdot x_{ji}$$

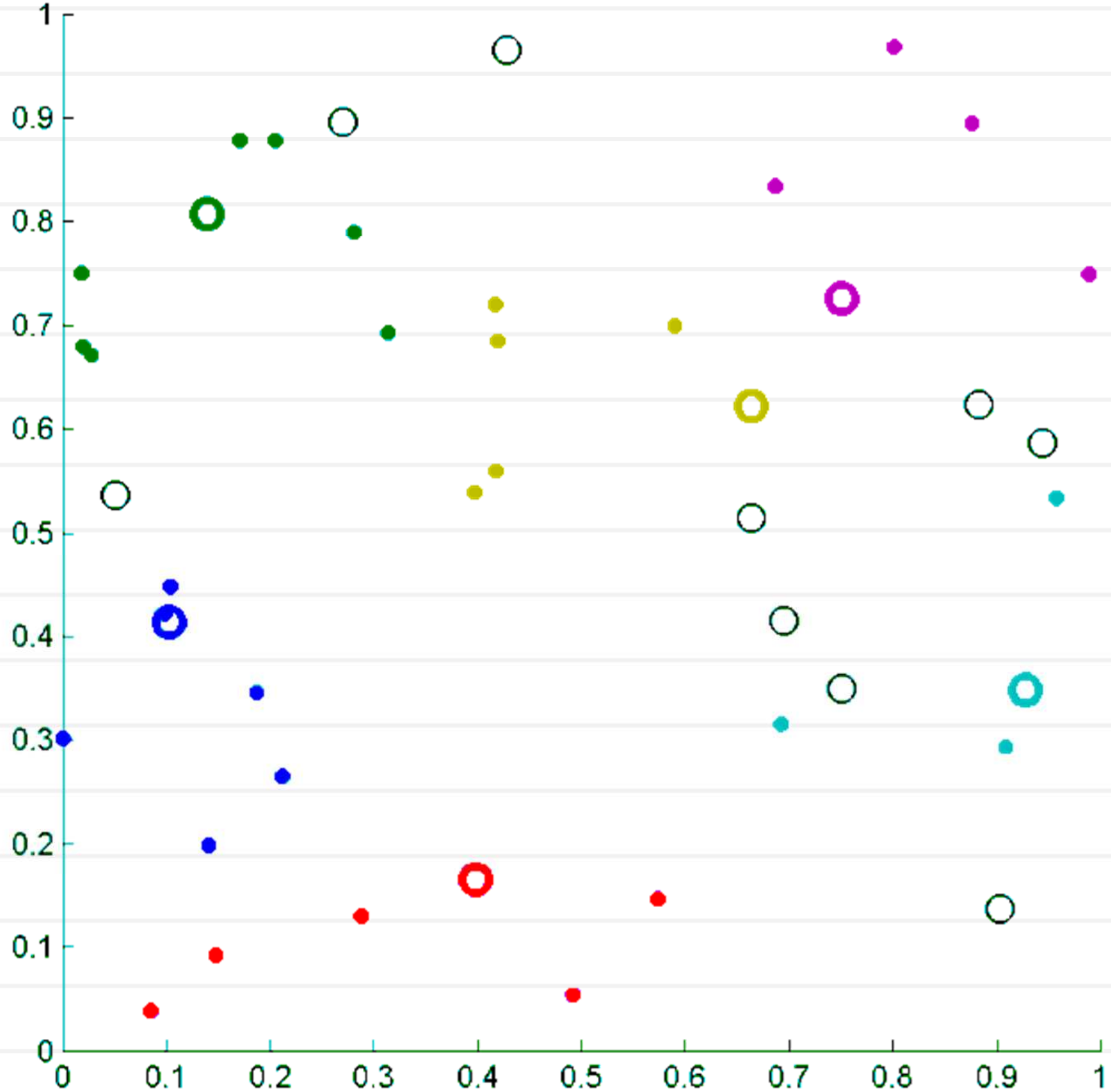
$$x_i, y_i \in \{0;1\} \quad \forall j \sum_{i=1}^{F} x_{ji} = 1$$

Two ways to define the consistency constraints:

$\forall i j \quad x_{ji} \le y_i$	$\sum_{j=1}^{C} x_{ji} \le C \cdot y_i$
Lots of (C×F) constraints	Fewer (F) constraints

Two equivalent formulations. Which one is preferable?

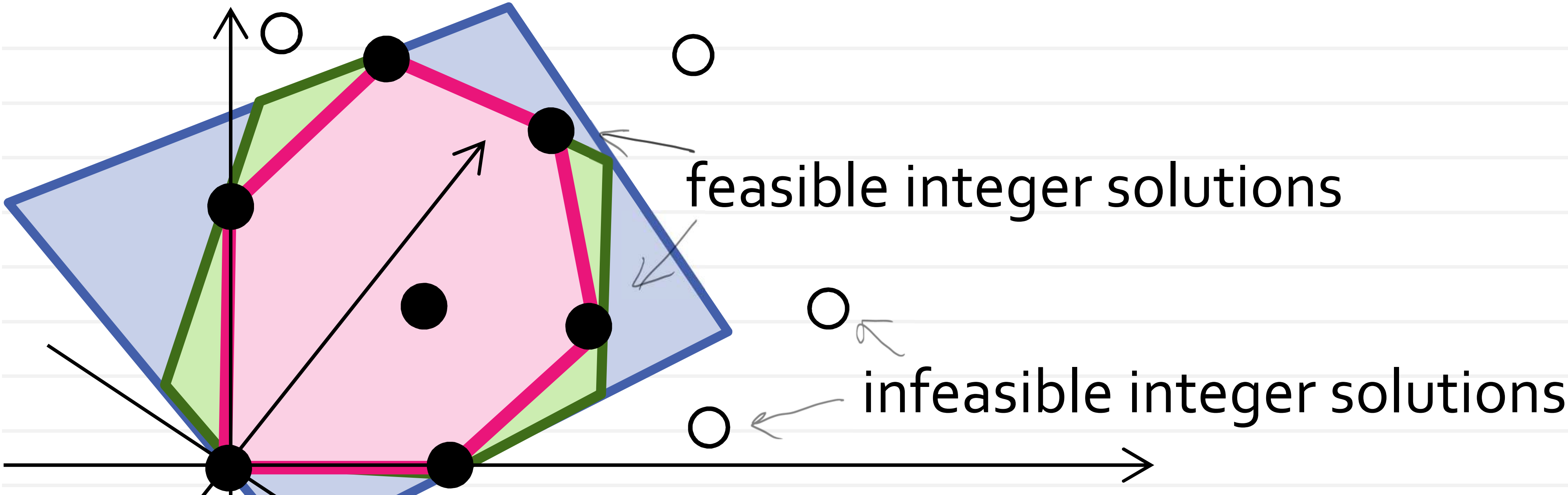
Comparison



Comparison

Integer program		Relaxation 1		Relaxation 2	
	1		1.0000		0.2000
	0		0.0000		0
	0		0.0000		0.0667
	0		0.0000		0.0667
	0		0.0000		0.0333
	0		0.0000		0
	1		1.0000		0.0667
	1		1.0000		0.1667
	1		1.0000		0.0333
	1		1.0000		0.1000
	0		0.0000		0.0333
	0		0.0000		0.0333
	0		0.0000		0.1333
	0		0.0000		0
1.6871	1	1.6871	1.0000	1.0105	0.0667

Geometric scheme



- Red is an ideal relaxation (convex hull)
- Green is an intermediate relaxation (some vertices are integer)
- Blue is a poor relaxation (all vertices are fractional)
- No non-feasible integer points can lie inside relaxations

When constructing of the ILP…

- Try to think of as many valid constraints as possible (as long as they are not redundant from **LP point of view**)

- Resolving fractional values is much harder then solving even very big LPs

- Even exponential number of constraints can be dealt with (more in the next lecture)