Advanced Mixed Integer Programming Formulations

Juan Pablo Vielma

Massachusetts Institute of Technology

Ingeniería UC,

Agosto, 2017.

(Linear) Mixed Integer Programming Formulation

- Let
  - $S \subseteq \mathbb{R}^n$,
  - $n_1 + n_2 = n,\ p_1 + p_2 = p,\ A \in \mathbb{Q}^{m \times n},\ D \in \mathbb{Q}^{m \times p},\ b \in \mathbb{Q}^m$
  - $P := \{(x, w) \in \mathbb{R}^n \times \mathbb{R}^p \ : \ Ax + Dw \leq b\}$
  - $P_I := P \cap (\mathbb{R}^{n_1} \times \mathbb{Z}^{n_2} \times \mathbb{R}^{p_1} \times \mathbb{Z}^{p_2})$
- P_I is a MIP formulation of S iff

$$S = \operatorname{Proj}_x (P_I)$$

- A formulation is *integral* or *ideal* iff

$$\operatorname{ext}(P) \subseteq (\mathbb{R}^{n_1} \times \mathbb{Z}^{n_2} \times \mathbb{R}^{p_1} \times \mathbb{Z}^{p_2})$$

Advantage of Integral Formulations

- If P_I is a formulation of S then:

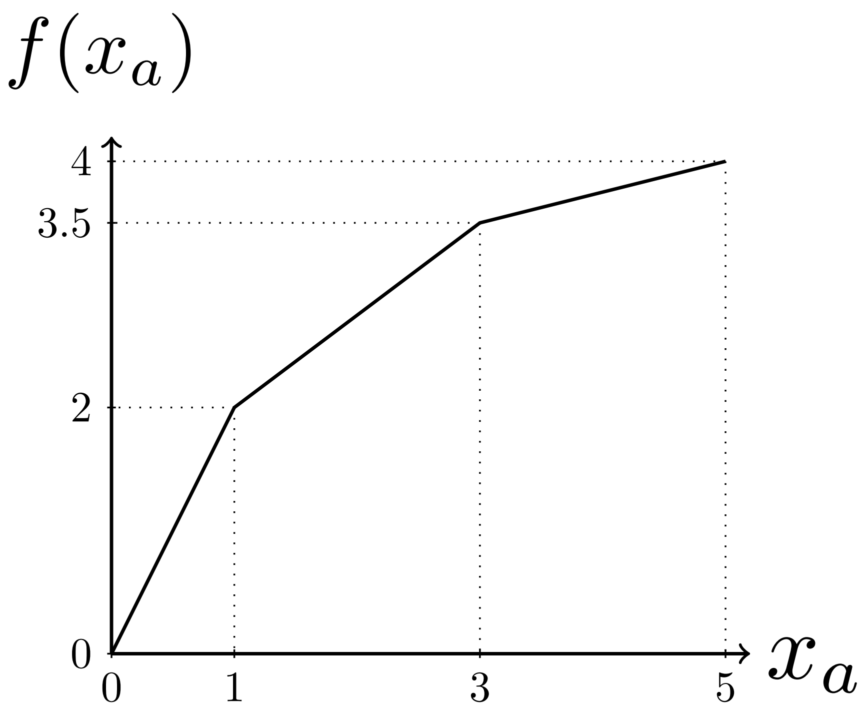
$$\max_{x} \left(c \cdot x \; : \; x \in S \right) = \max_{x,w} \left(c \cdot x \; : \; (x, w) \in P_I \right)$$

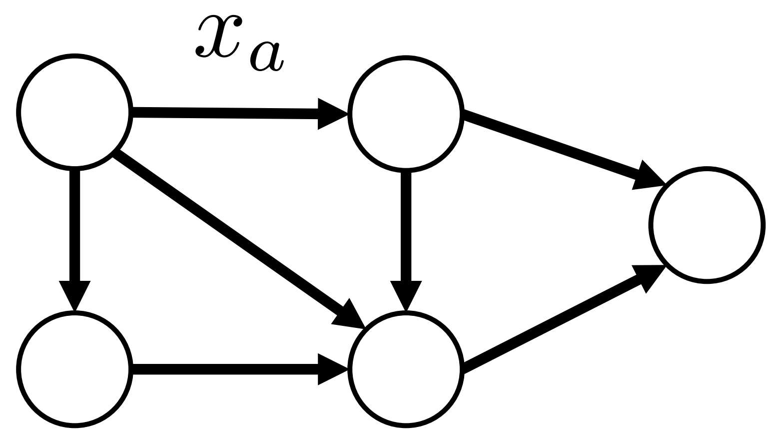
- If P_I is an ideal formulation of S then:

$$\max_{x,w} \left(c \cdot x \; : \; (x, w) \in P_I \right) = \max_{x,w} \left(c \cdot x \; : \; (x, w) \in P \right)$$

- In practice, S is one of many constraints:
  - Ideal (or strong) formulations tend to be more effective

Example: Piecewise Linear Network Flow





$$gr(f_i) := \{(x_i, z_i) \; : \; f_i(x_i) = z_i\}$$

- Network flow or transportation problem
- Economies of scale for transportation costs

Constructing a MIP Formulation

$$\min \quad \sum_{i=1}^{n} f_i(x_i)$$

$$s.t.$$

$$Ex \le h$$
$$0 \le x_i \le u \quad \forall i \in [n]$$

$$\min \quad \sum_{i=1}^{n} z_i$$

$$s.t.$$

$$\begin{aligned} Ex &\le h \\ 0 \le x_i &\le u & \forall i \in [n] \\ (x_i, z_i) &\in gr(f_i) & \forall i \in [n]. \end{aligned}$$

$$\min \quad \sum_{i=1}^{n} z_i$$

$$s.t.$$

$$\begin{aligned} Ex &\le h \\ 0 \le x_i &\le u & \forall i \in [n] \\ A^i \begin{pmatrix} x_i \\ z_i \end{pmatrix} + B^i \lambda^i + D^i y^i &\le b^i & \forall i \in [n] \\ y^i &\in \mathbb{Z}^k & \forall i \in [n] \end{aligned}$$

MIP Formulation

$$gr(f_i) := \{(x_i, z_i) \; : \; f_i(x_i) = z_i\}$$

Strong, but not Necessarily Ideal

$$\min \quad \sum_{i=1}^{n} z_i$$

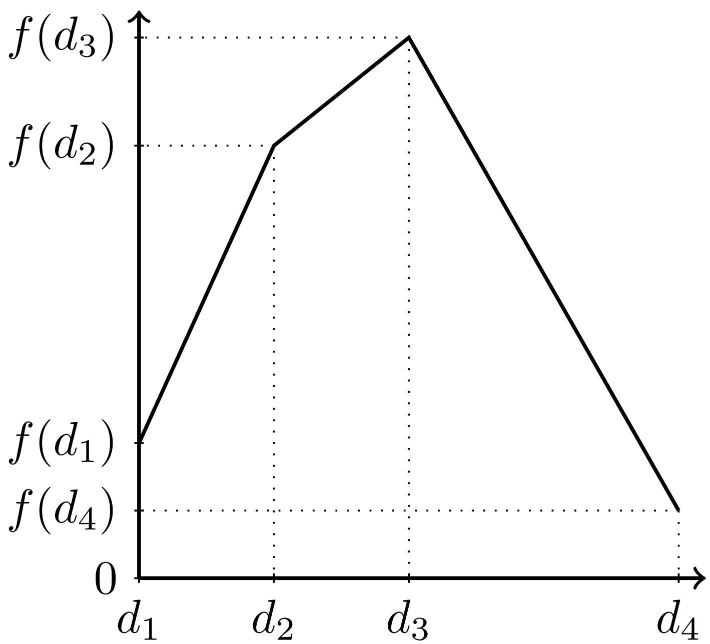
$$s.t.$$

$$\begin{aligned} Ex &\le h \\ 0 \le x_i &\le u && \forall i \in [n] \\ A^i \begin{pmatrix} x_i \\ z_i \end{pmatrix} + B^i \lambda^i + D^i y^i &\le b^i && \forall i \in [n] \\ y^i &\in \mathbb{Z}^k && \forall i \in [n] \end{aligned}$$

Ideal for each i

Not necessarily ideal for complete problem

Naïve Formulation for Piecewise Linear Functions



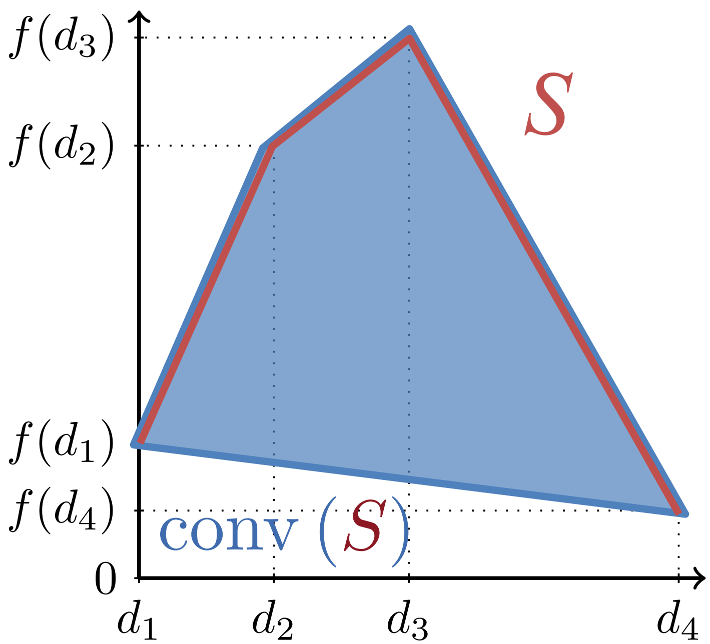
$$\sum_{i=1}^{k} y_i = 1$$

$$y \in \{0,1\}^k$$

$$f(x) = \begin{cases} m_1 x + c_1 & x \in [d_1, d_2] \\ & \vdots \\ m_k x + c_k & x \in [d_k, d_{k+1}] \end{cases}$$

Not integral and very weak

Better Formulation (CC)



$$\sum_{i=1}^{4} d_i \lambda_i = x, \qquad \sum_{i=1}^{4} f(d_i)\lambda_i = z$$

$$\sum_{i=1}^{4} \lambda_i = 1, \qquad \lambda_i \geq 0$$

$$\sum_{i=1}^{3} y_i = 1, \qquad y_i \in \{0,1\}$$

$$\lambda_1 \leq y_1, \qquad \lambda_2 \leq y_1 + y_2$$

$$\lambda_3 \leq y_2 + y_3, \qquad \lambda_4 \leq y_3$$

$$f(x) = \begin{cases} m_1 x + c_1 & x \in [d_1, d_2] \\ & \vdots \\ m_k x + c_k & x \in [d_k, d_{k+1}] \end{cases}$$

Still not integral, but strong in a restricted sense

Sharp Formulations

- A MIP formulation P_I of S is *sharp or convex hull* iff

$$\operatorname{conv}(S) = \operatorname{Proj}_x(P)$$

- If P_I is a *sharp* formulation of S then:

$$\max_{x,w}\left(c\cdot x \;:\; (x,w)\in P_I\right) = \max_{x,w}\left(c\cdot x \;:\; (x,w)\in P\right)$$

- CC is a sharp formulation for piecewise linear functions, but Big-M is not sharp.
  – Exercise: Show for $x=2$ and

$$f(x) = \begin{cases} 1-x & x\in[0,1] \\ 2x-2 & x\in[1,2] \\ 6-2x & x\in[2,3] \\ x-3 & x\in[3,4] \end{cases}$$

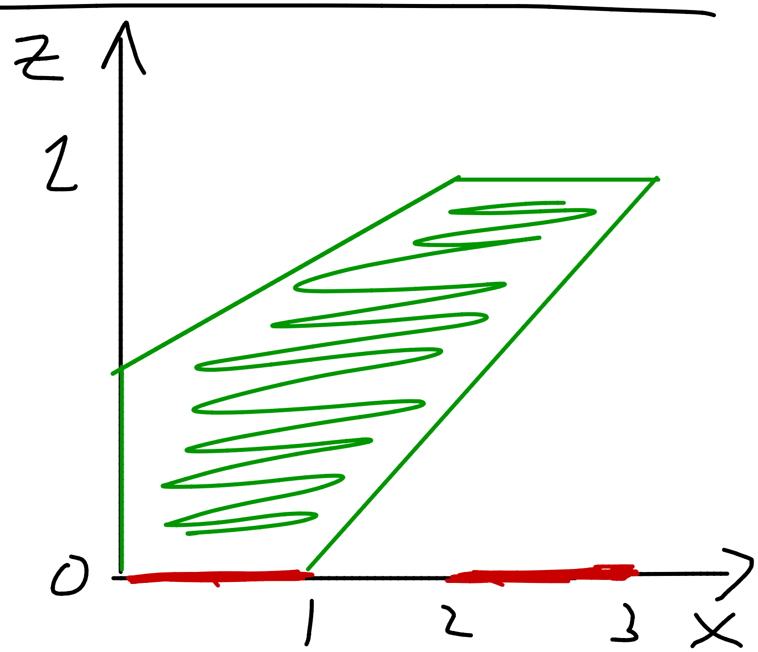
Ideal and Sharp Formulations

- Ideal formulations are sharp
- If p=0 (no auxiliary variables) then sharp formulations are ideal
- Example of non-ideal sharp formulation:

Example:

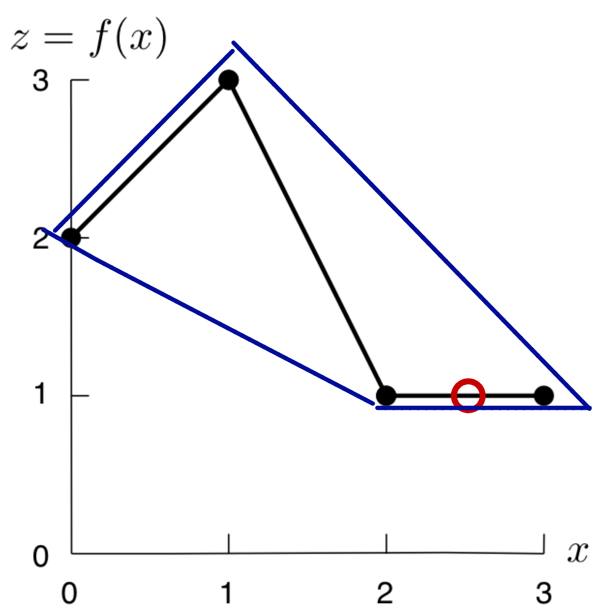
$$4z - x \le 2,\quad x \ge 0$$
$$x - 2z \le 1,\quad z \in \{0,1\}$$

is a sharp but not integral formulation of $S = [0,1] \cup [2,3]$



Remember: CC is not Ideal

Example: $S = \{(x,z) : f(x) = z\}$



$$
\begin{aligned}
0\lambda_0 + 1\lambda_1 + 2\lambda_2 + 3\lambda_3 &= x \\
2\lambda_0 + 3\lambda_1 + 1\lambda_2 + 1\lambda_3 &= z \\
\lambda_0 + \lambda_1 + \lambda_2 + \lambda_3 &= 1 \\
\lambda_0 &\leq y_1 \\
\lambda_1 &\leq y_1 + y_2 \\
\lambda_2 &\leq y_2 + y_3 \\
\lambda_3 &\leq y_3 \\
y_1 + y_2 + y_3 &= 1 \\
\lambda_i &\geq 0 & \forall i \in \{0, \ldots, 3\} \\
0 \leq y_i &\leq 1 & \forall i \in \{1, 2, 3\} \\
y_i &\in \mathbb{Z} & \forall i \in \{1, 2, 3\}.
\end{aligned}
$$

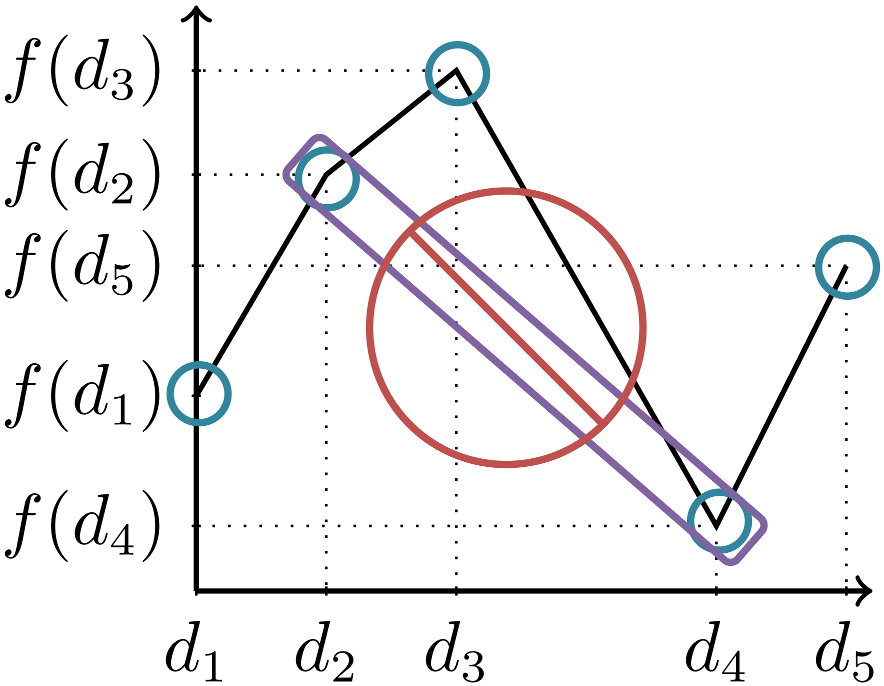
Sharp

Extreme point: $\lambda_2 = \lambda_3 = 1/2$, $\lambda_0 = \lambda_1 = 0$

$x = 2.5$, $z = 1$ $\quad y_1 = y_3 = 1/2$, $y_2 = 0$

Simple Formulation for Univariate Functions

$$z = f(x)$$



$$\text{Size} = O(\#\text{ of segments})$$

Non-Ideal: Fractional Extreme Points

$$\begin{pmatrix} x \\ z \end{pmatrix} = \sum_{j=1}^{5} \begin{pmatrix} d_j \\ f(d_j) \end{pmatrix} \lambda_j$$

$$1 = \sum_{j=1}^{5} \lambda_j, \quad \lambda_j \geq 0$$

$$y \in \{0,1\}^4, \quad \sum_{i=1}^{4} y_i = 1$$

$$0 \leq \lambda_1 \leq y_1$$

$$0 \leq \lambda_2 \leq y_1 + y_2$$

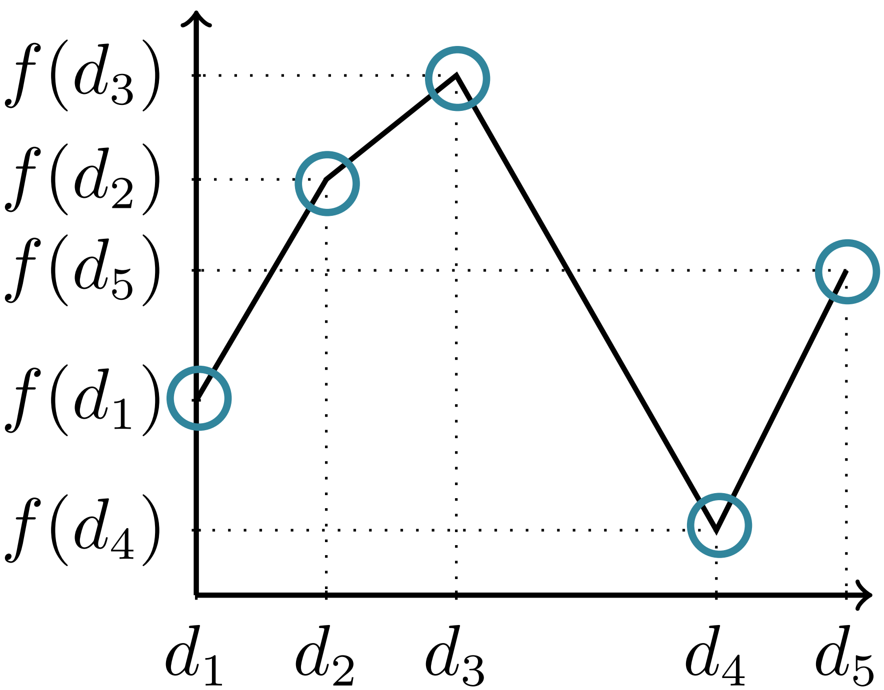
$$0 \leq \lambda_3 \leq y_2 + y_3$$

$$0 \leq \lambda_4 \leq y_3 + y_4$$

$$0 \leq \lambda_5 \leq y_4$$

Advanced Formulation for Univariate Functions

$$z = f(x)$$



$$\binom{x}{z} = \sum_{j=1}^{5} \binom{d_j}{f(d_j)} \lambda_j$$

$$1 = \sum_{j=1}^{5} \lambda_j, \quad \lambda_j \geq 0$$

$$y \in \{0,1\}^2$$

$$0 \leq \lambda_1 + \lambda_5 \leq 1 - y_1$$

$$0 \leq \lambda_3 \leq y_1$$

$$0 \leq \lambda_4 + \lambda_5 \leq 1 - y_2$$

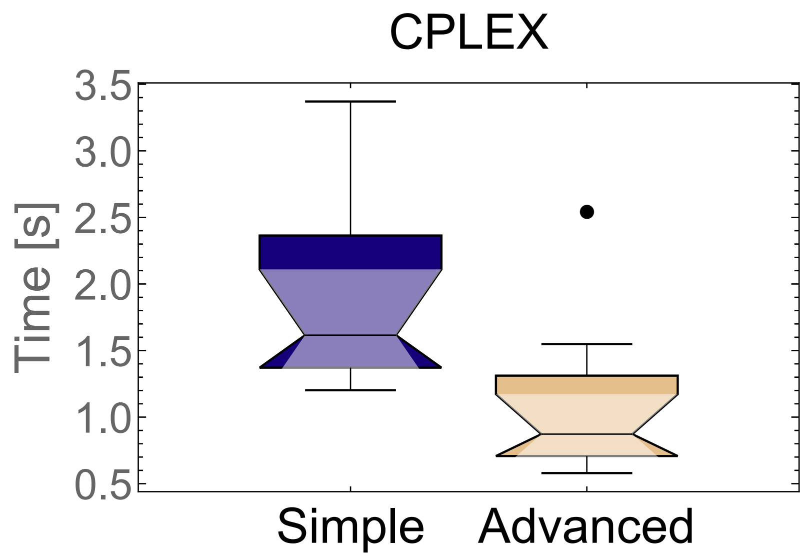
$$0 \leq \lambda_1 + \lambda_2 \leq y_2$$

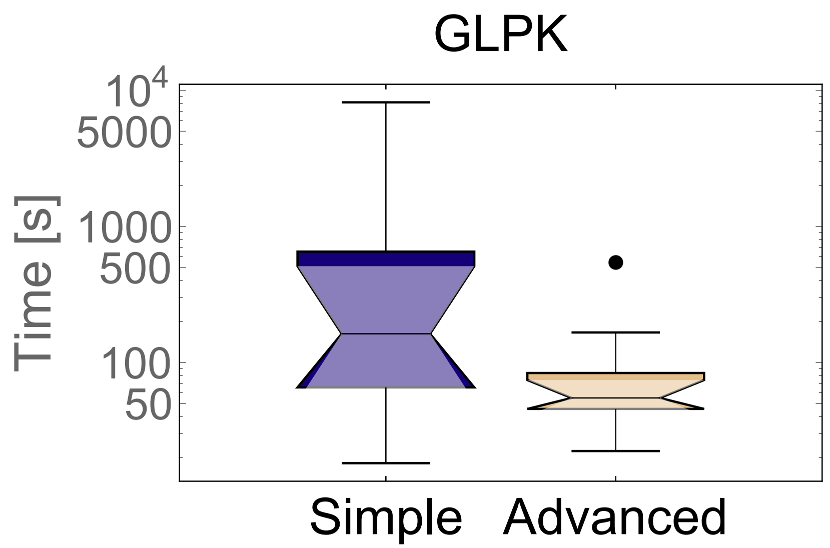
$$\text{Size} = O(\log_2 \# \text{ of segments})$$

Ideal: Integral Extreme Points

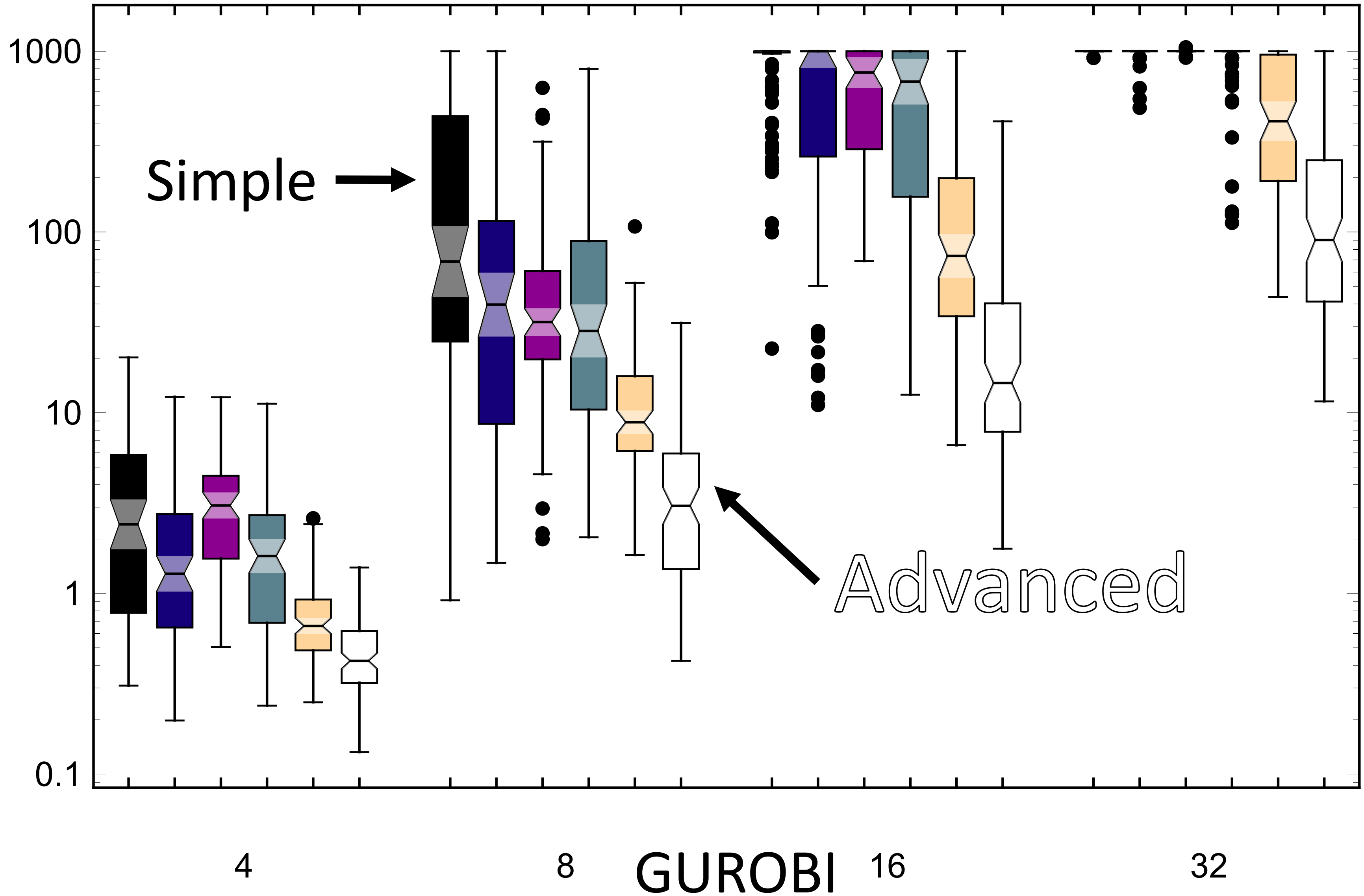
Computational Performance

- Advanced formulations provide an computational advantage
- Advantage is significantly more important for free solvers
- State of the art commercial solvers can be significantly better that free solvers
- Still, free is free!





Formulation Improvements can be Significant



Constructing Advanced Formulations

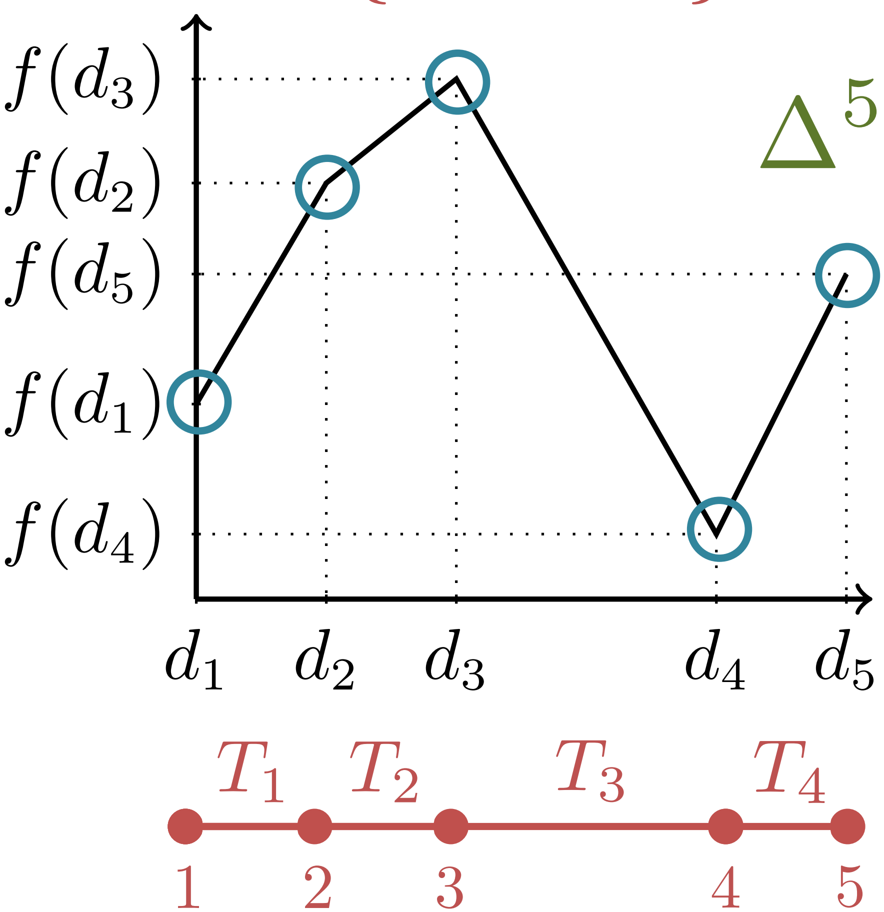
Abstracting Univariate Functions

$$P_i := \{\lambda \in \Delta^5 : \lambda_j = 0 \quad \forall j \notin T_i\}$$

$$z = f(x)$$

$$\begin{pmatrix} x \\ z \end{pmatrix} = \sum_{j=1}^{5} \begin{pmatrix} d_j \\ f(d_j) \end{pmatrix} \lambda_j$$

$$T_i := \{i, i+1\} \quad i \in \{1, \ldots, 4\}$$

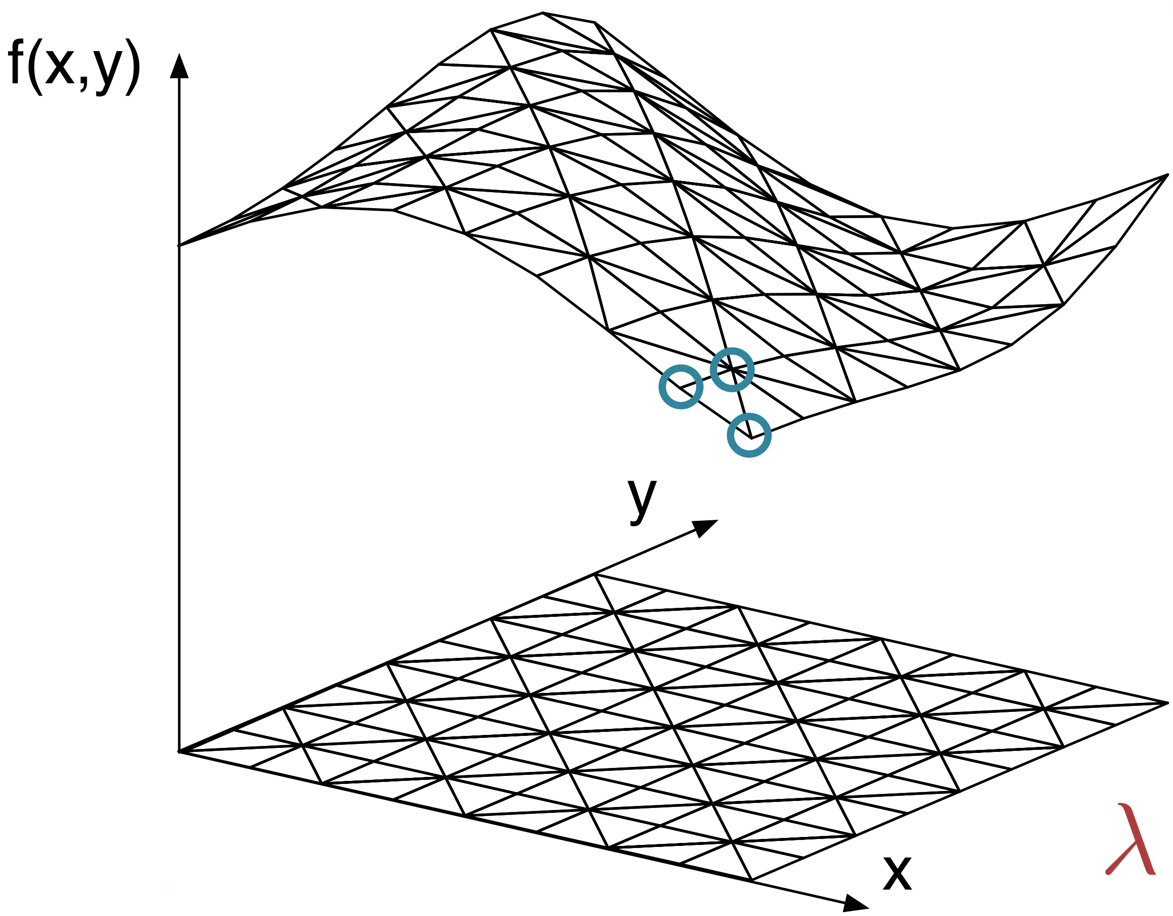


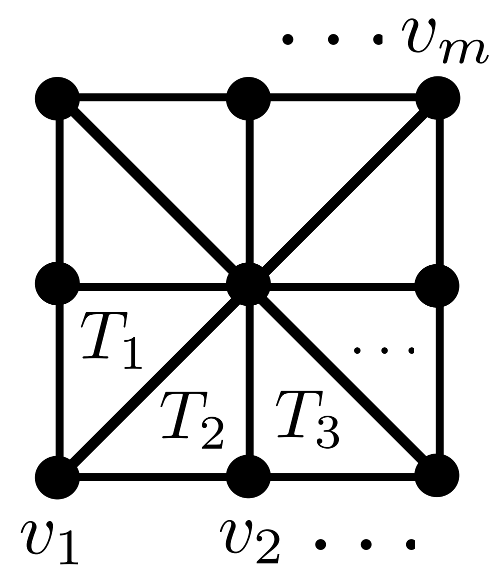
$$\Delta^5 = \left\{ 1 = \sum_{j=1}^{5} \lambda_j, \quad \lambda_j \geq 0 \right\}$$

$$\lambda \in \bigcup_{i=1}^{4} P_i \subseteq \Delta^5$$

Abstraction Works for Multivariate Functions

$$P_i := \{\lambda \in \Delta^m : \lambda_j = 0 \quad \forall v_j \notin T_i\}$$

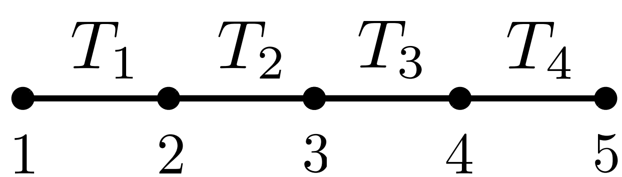


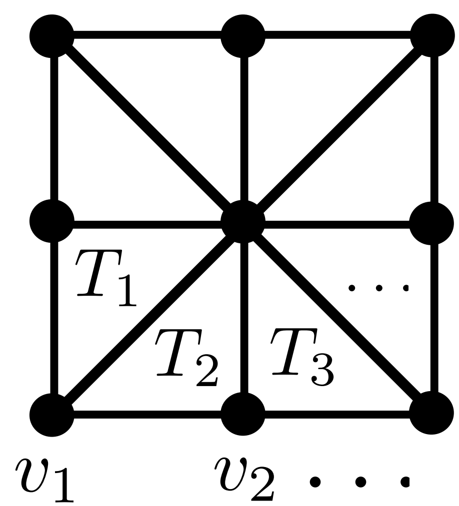


$$\lambda \in \bigcup_{i=1}^{n} P_i \subseteq \Delta^m$$

Complete Abstraction

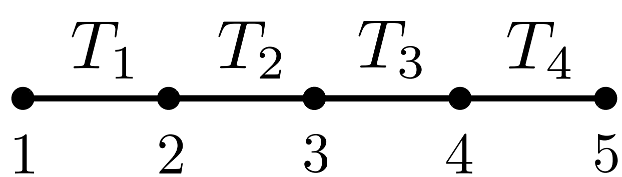
- $\Delta^V := \left\{ \lambda \in \mathbb{R}^V_+ : \sum_{v \in V} \lambda_v = 1 \right\}$,
- $P_i = \left\{ \lambda \in \Delta^V : \lambda_v = 0 \quad \forall v \notin T_i \right\}$
- $\lambda \in \bigcup_{i=1}^{n} P_i$
- $T_i =$ cliques of a graph

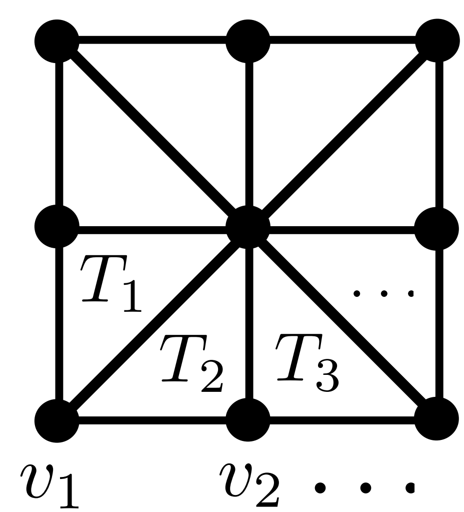




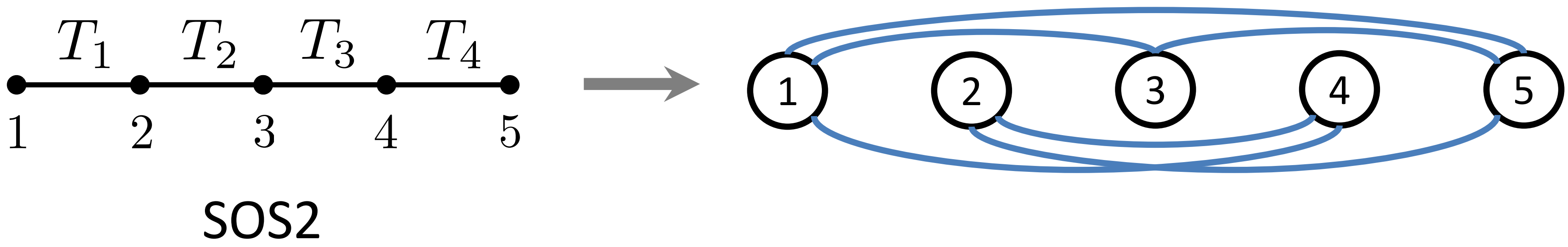
Complete Abstraction

- $\Delta^V := \left\{ \lambda \in \mathbb{R}^V_+ : \sum_{v \in V} \lambda_v = 1 \right\}$,
- $P_i = \left\{ \lambda \in \Delta^V : \lambda_v = 0 \quad \forall v \notin T_i \right\}$
- $\lambda \in \bigcup_{i=1}^n P_i$
- $T_i =$ cliques of a graph

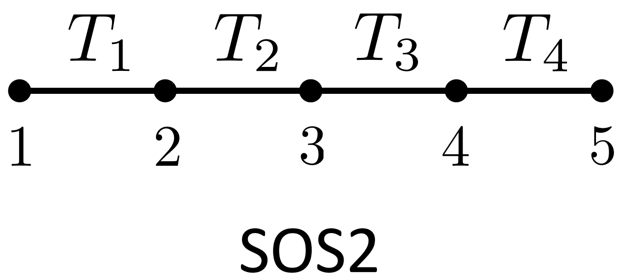


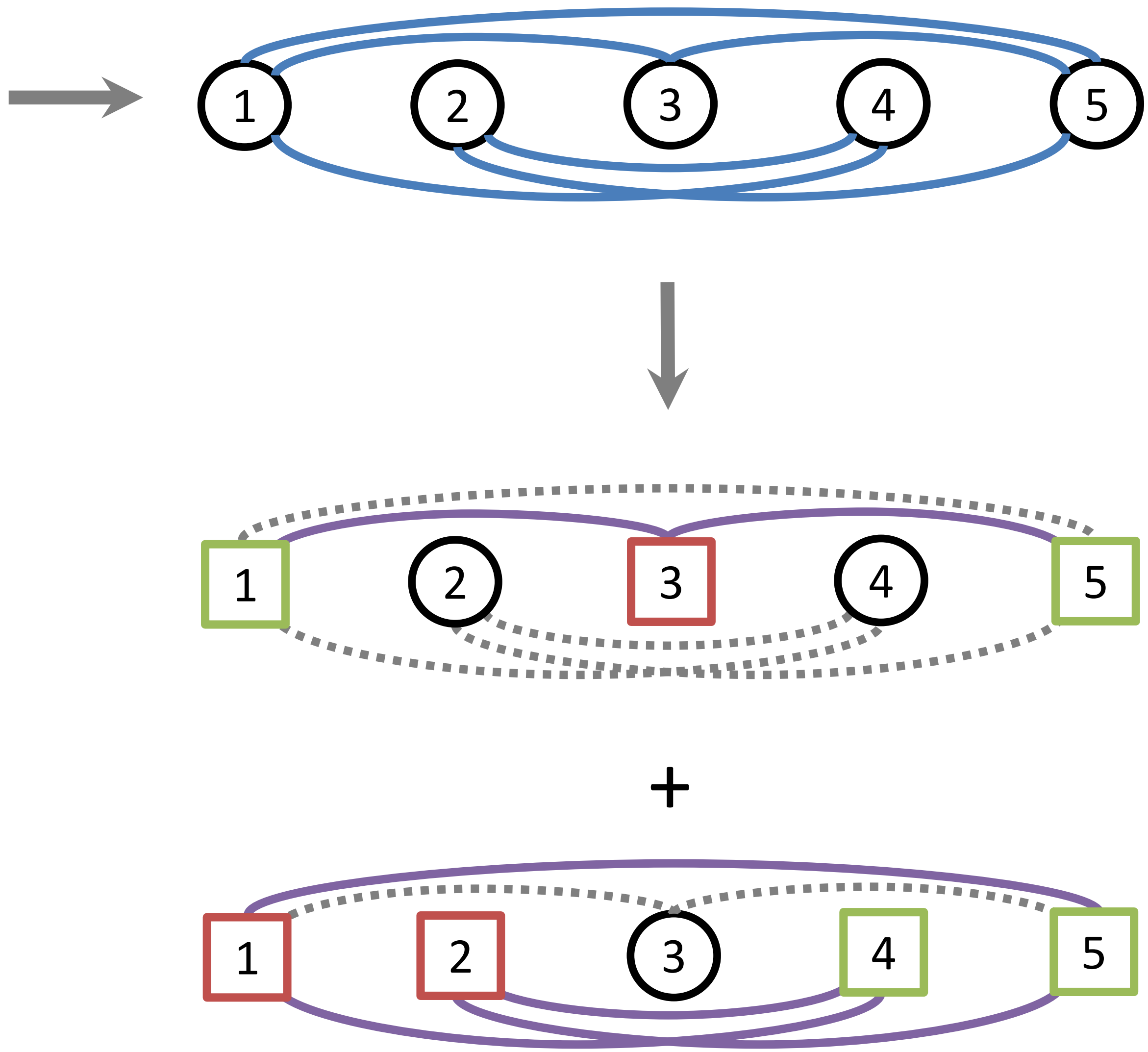


From Cliques to (Complement) Conflict Graph

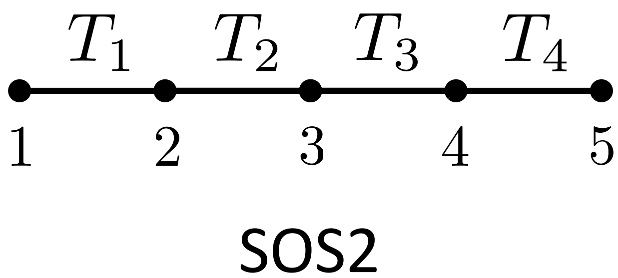


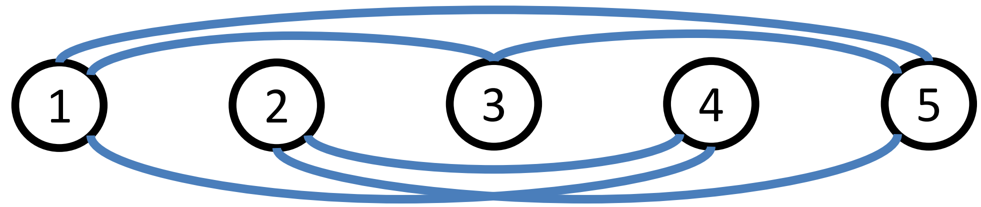
From Conflict Graph to Bi-clique Cover



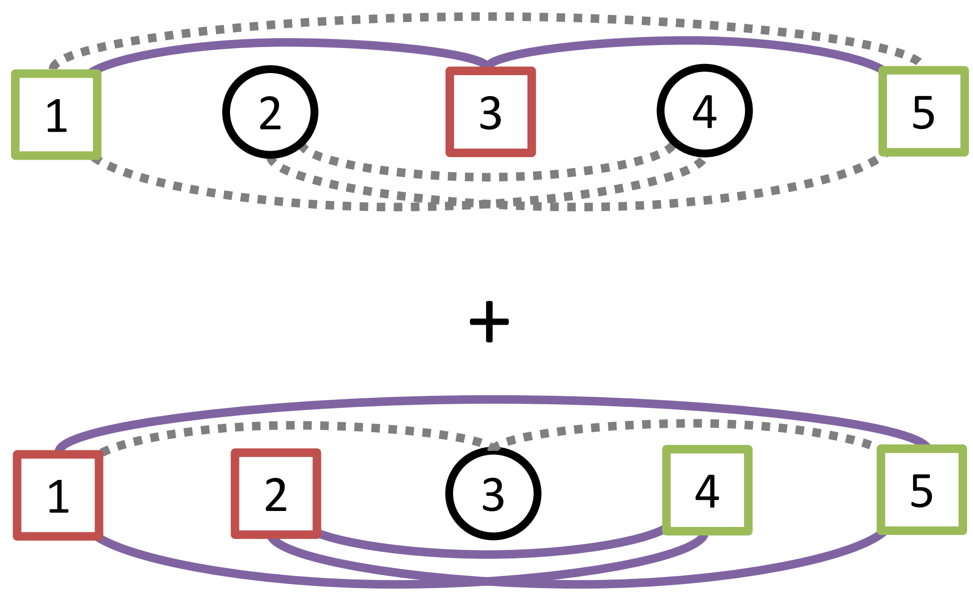


From Bi-clique Cover to Formulation





$$
\begin{aligned}
0 &\leq \lambda_1 + \lambda_5 \leq 1 - y_1 \\
0 &\leq \lambda_3 \leq y_1 \\
0 &\leq \lambda_4 + \lambda_5 \leq 1 - y_2 \\
0 &\leq \lambda_1 + \lambda_2 \leq y_2
\end{aligned}
$$



Ideal Formulation from Bi-clique Cover

- Conflict Graph $G = (V, E)$

$$E = \{(u, v) \, : \, u, v \in V, \; u \neq v, \quad \nexists i \text{ s.t. } u, v \in T_i\}$$

- Bi-clique cover $\{(A^j, B^j)\}_{j=1}^{t}, \quad A^j, B^j \subseteq V$

$$\forall \{u, v\} \in E \quad \exists j \text{ s.t. } u \in A^j \; \wedge v \in B^j$$

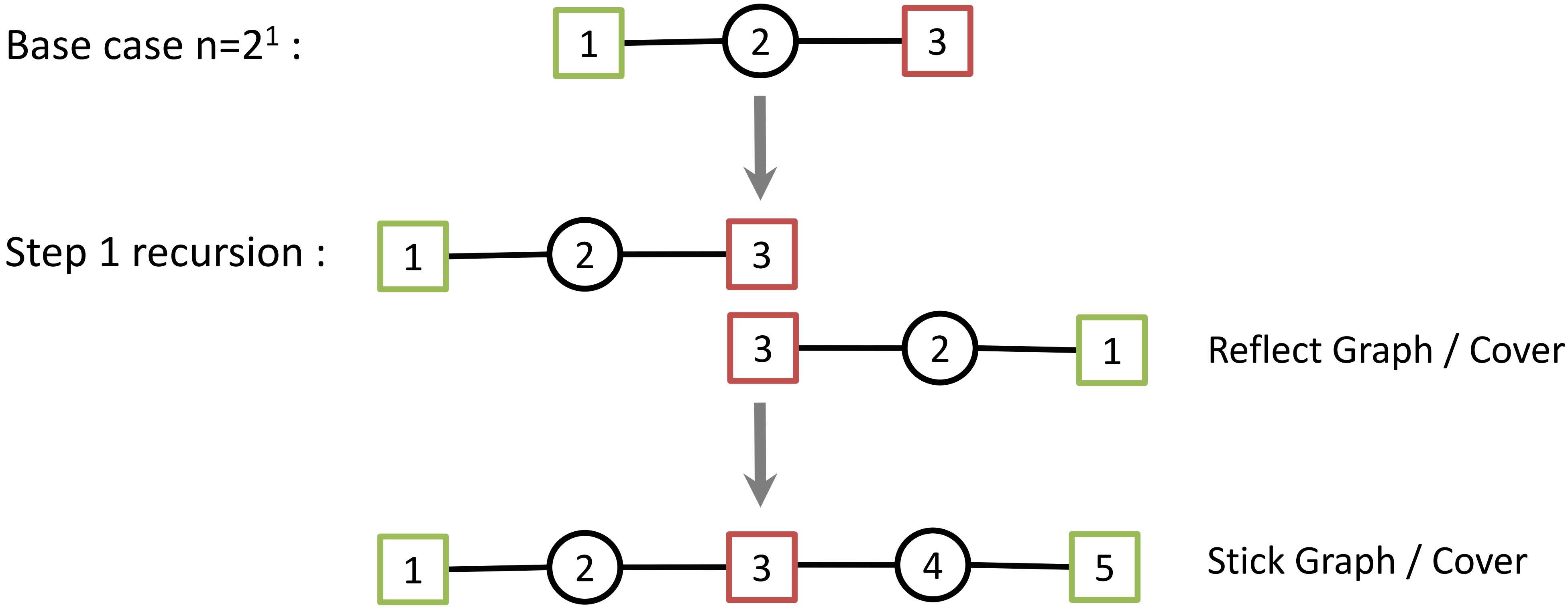
- Formulation

$$\sum_{v \in A^j} \lambda_v \leq 1 - y_j \quad \forall j \in [t]$$

$$\sum_{v \in B^j} \lambda_v \leq y_j \quad \forall j \in [t]$$

$$y \in \{0, 1\}^t$$

Recursive Construction of Cover for SOS2, Step 1

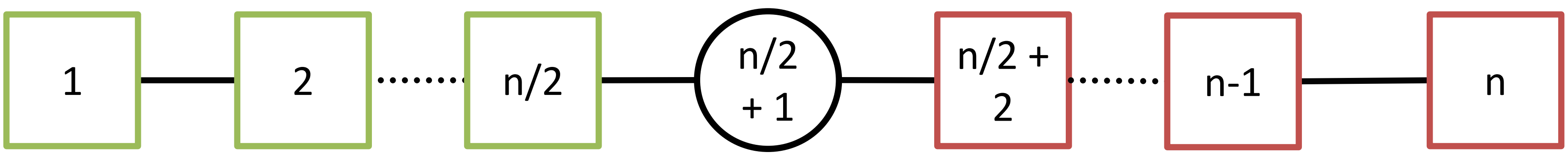


Repeat for all bi-cliques from 2^{k-1} to cover all edges within first and last half of conflict graph

Recursive Construction of Cover for SOS2, Step 2

Only edges missing are those between
first and last half of conflict graph

Step 2 : Add one more bi-clique

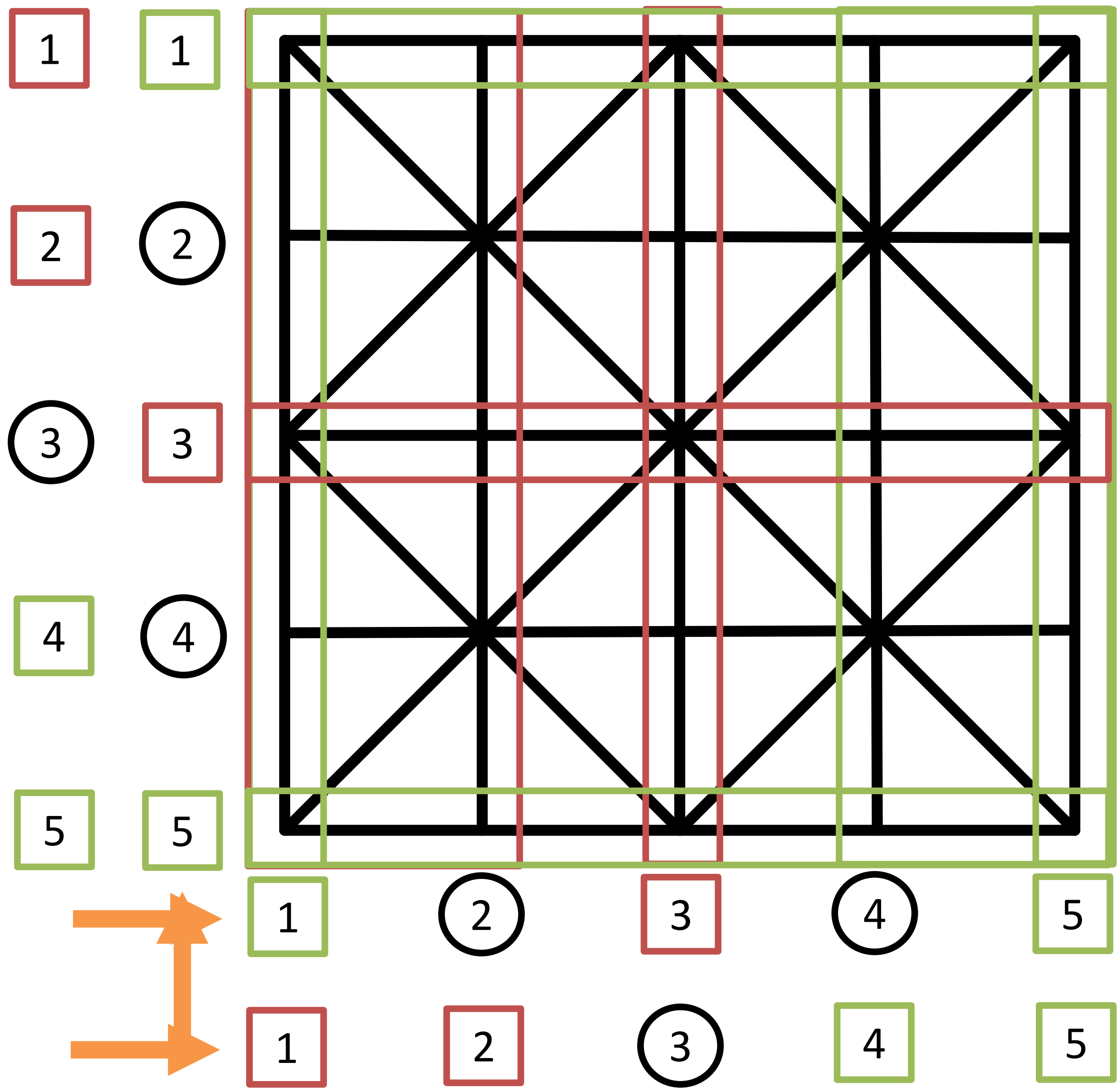


Cover has $\log_2 n$ bi-cliques.

For non-power of two just delete extra nodes.

Grid Triangulations: Step 1 = SOS2 for Inter-Box

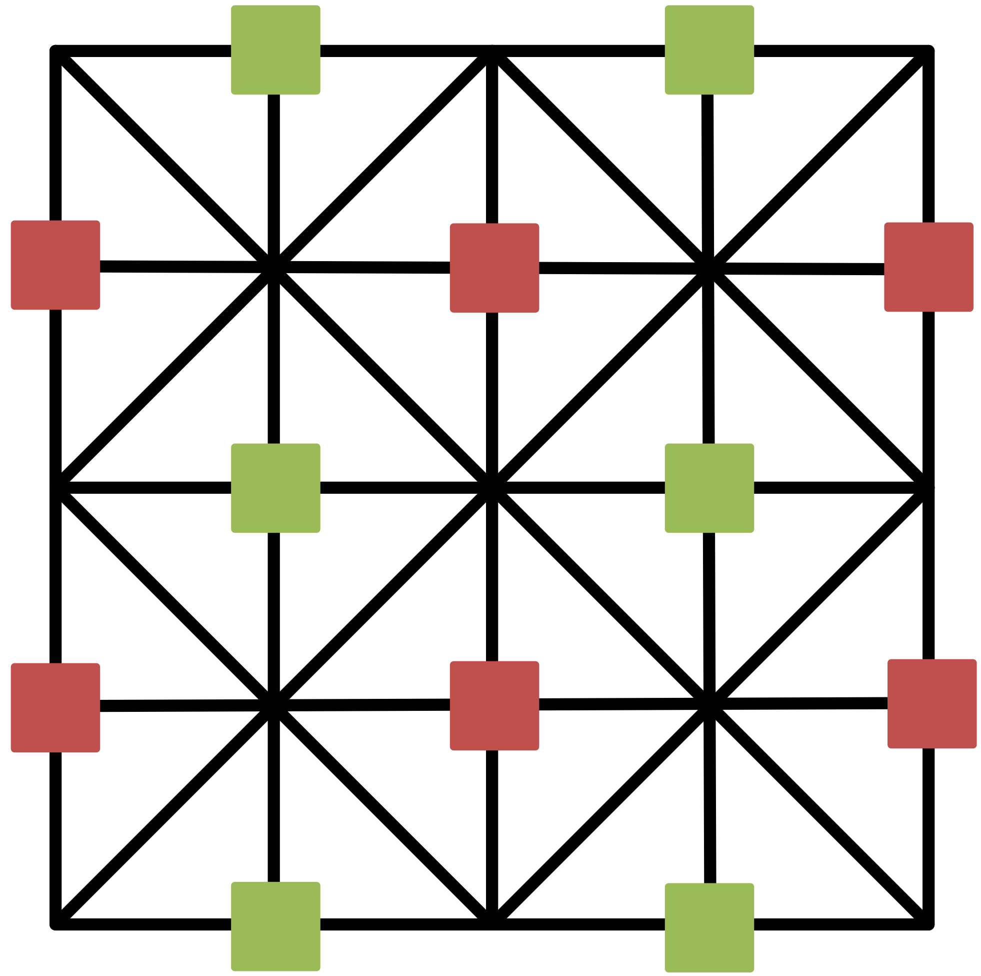
Covers all arcs between boxes



Grid Triangulations: Step 2 = Ad-hoc Intra-Box

Covers all arcs within boxes

Sometimes 1 additional cover

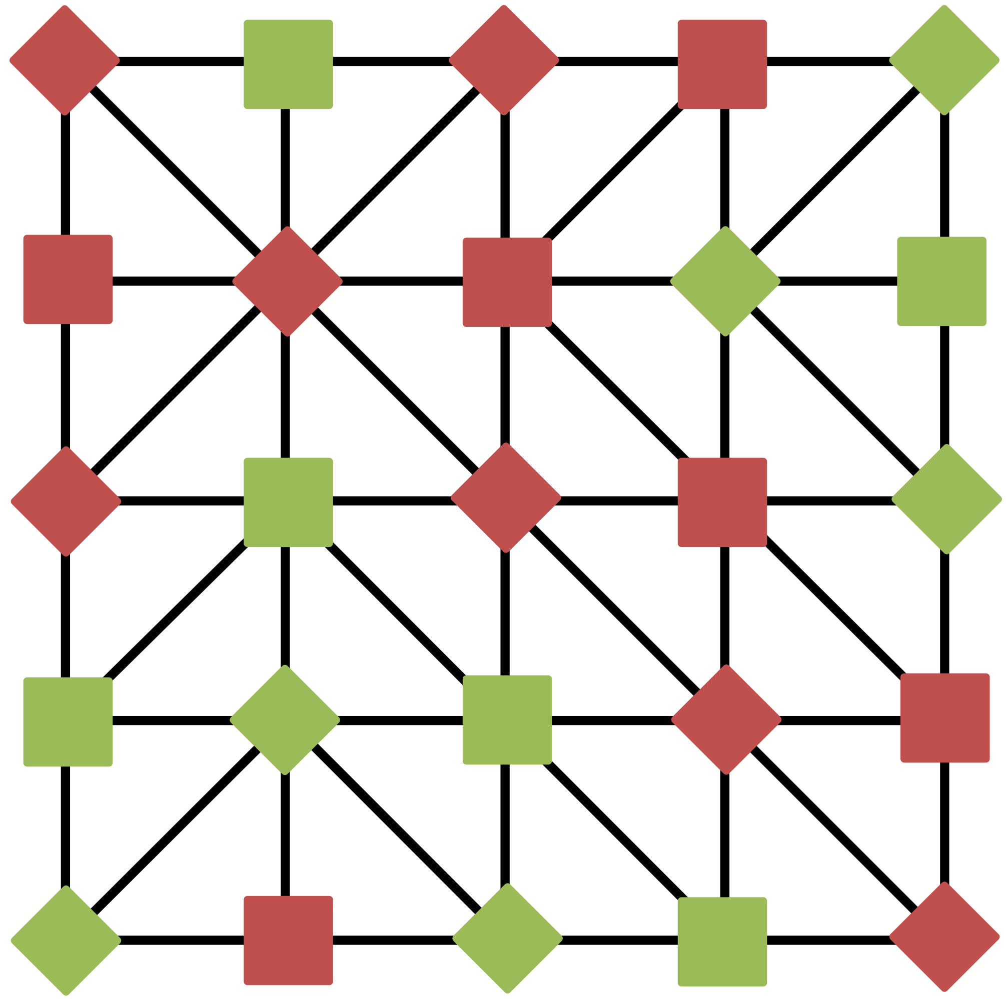


Grid Triangulations: Step 2 = Ad-hoc Intra-Box

Sometimes **2** additional covers

Sometimes more, but always less than **9**

Simple rules to get (near) optimal in Fall '16



More elaborate: SOS3(26)

SOS2 on Blocks of 3

Cover arcs between adjacent blocks of 3