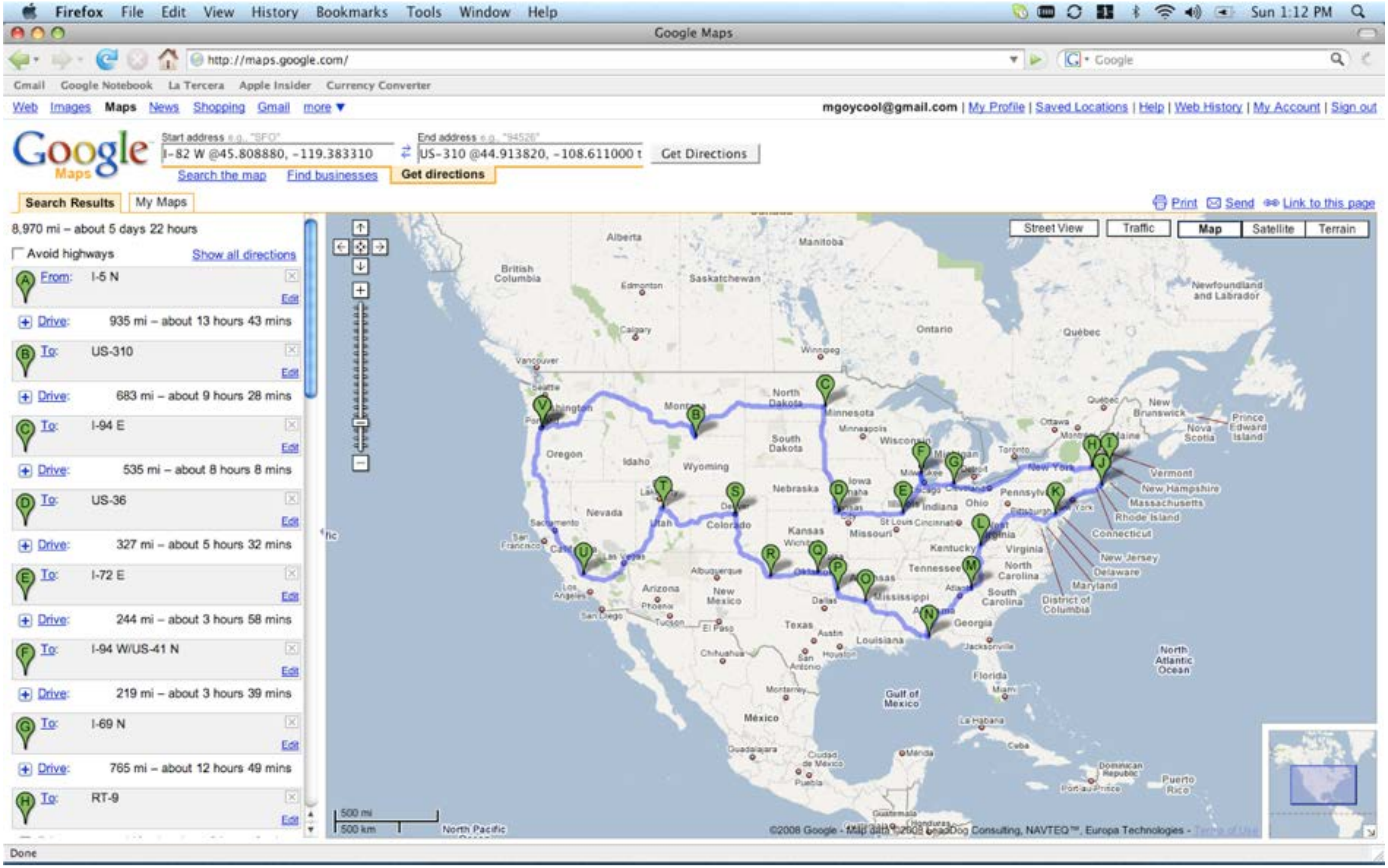
Advanced Mixed Integer Programming (MIP) Formulation Techniques

Juan Pablo Vielma

Massachusetts Institute of Technology

Center for Nonlinear Studies, Los Alamos National Laboratory.
Los Alamos, New Mexico, August, 2016.

Traveling Salesman Problem (TSP): Visit Cities Fast



Start address e.g., "SFO": I-82 W @45.808880, -119.383310

End address e.g., "94526": US-310 @44.913820, -108.611000 t

8,970 mi – about 5 days 22 hours

☐ Avoid highways

Show all directions

A From: I-5 N

Drive: 935 mi – about 13 hours 43 mins

B To: US-310

Drive: 683 mi – about 9 hours 28 mins

C To: I-94 E

Drive: 535 mi – about 8 hours 8 mins

D To: US-36

Drive: 327 mi – about 5 hours 32 mins

E To: I-72 E

Drive: 244 mi – about 3 hours 58 mins

F To: I-94 W/US-41 N

Drive: 219 mi – about 3 hours 39 mins

G To: I-69 N

Drive: 765 mi – about 12 hours 49 mins

H To: RT-9

MIP = Avoid Enumeration

- Number of tours for 49 cities $= 48!/2 \approx 10^{60}$
- Fastest supercomputer $\approx 10^{17}\text{flops}$
- Assuming one floating point operation per tour: $> 10^{35}$ years $\approx 10^{25}$ times the age of the universe!
- How long does it take on an iphone?
  - Less than a second!
  - 4 iterations of **cutting plane** method!
  - Dantzig, Fulkerson and Johnson 1954 did it by hand!
  - For more info see tutorial in ConcordeTSP app
  - **Cutting planes** are the key for effectively solving (even NP-hard) MIP problems in practice.

50+ Years of MIP = Significant Solver Speedups

- Algorithmic Improvements (Machine Independent):
  - CPLEX v1.2 (1991) – v11 (2007): 29,000x speedup
  - Gurobi v1 (2009) – v6.5 (2015): 48.7x speedup
  - Commercial, but free for academic use
- (Reasonably) effective free / open source solvers:
  - GLPK, CBC and SCIP (free only for non-commercial)
- Easy to use, fast and versatile modeling languages
  - Julia based JuMP modelling language
- Linear MIP solvers very mature and effective:
  - Convex nonlinear MIP getting there (quadratic nearly there)

What do YOU need to do to use MIP?

1. Use JuMP
2. Construct a MIP formulation of your problem

- This talk:
  - From non-convex constraints to linear MIP formulations
  - One illustrative example
- Beyond linear MIP:
  - Convex nonlinear MIP
  - See Miles talk on Thursday

Example: Experimental Design in Marketing

Think “Simulation-Based” Optimization

(Custom) Product Recommendations via CBCA

Feature	SX530	RX100
Zoom	50x	3.6x
Prize	$249.99	$399.99
Weight	15.68 ounces	7.5 ounces
Prefer	☑	☐

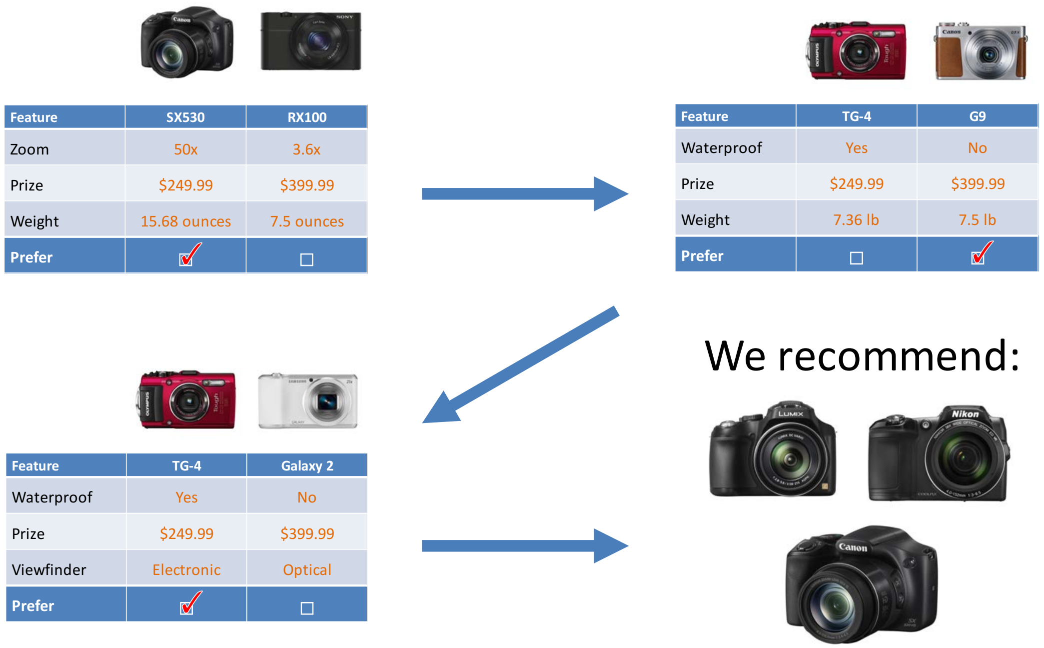
Feature	TG-4	G9
Waterproof	Yes	No
Prize	$249.99	$399.99
Weight	7.36 lb	7.5 lb
Prefer	☐	☑

Feature	TG-4	Galaxy 2
Waterproof	Yes	No
Prize	$249.99	$399.99
Viewfinder	Electronic	Optical
Prefer	☑	☐

We recommend:

Towards Optimal Product Recommendation

- Find enough information about preferences to recommend



Feature	SX530	RX100
Zoom	50x	3.6x
Prize	$249.99	$399.99
Weight	15.68 ounces	7.5 ounces
Prefer	☑	☐

Feature	TG-4	G9
Waterproof	Yes	No
Prize	$249.99	$399.99
Weight	7.36 lb	7.5 lb
Prefer	☐	☑

Feature	TG-4	Galaxy 2
Waterproof	Yes	No
Prize	$249.99	$399.99
Viewfinder	Electronic	Optical
Prefer	☑	☐

We recommend:

- How do I pick the **next (1^{st}) question** to obtain the largest reduction of uncertainty or “variance” on preferences

Choice-based Conjoint Analysis

Feature	Chewbacca	BB-8
Wookiee	Yes	No
Droid	No	Yes
Blaster	Yes	No
I would buy toy	☑	☐
Product Profile	x^1	x^2

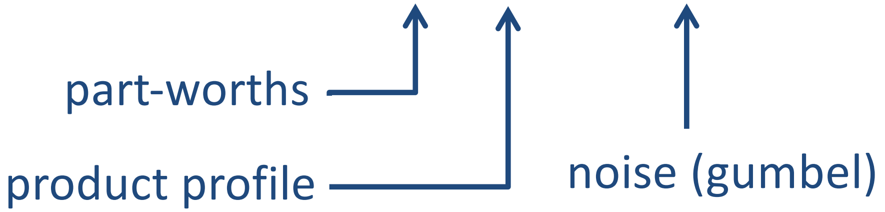
$$\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = x^2$$

MNL Preference Model

- Utilities for 2 products, n features (e.g. n = 12)

$$U_1 = \beta \cdot x^1 + \epsilon_1 = \sum\nolimits_{i=1}^{n} \beta_i x_i^1 + \epsilon_1$$

$$U_2 = \beta \cdot x^2 + \epsilon_2 = \sum\nolimits_{i=1}^{n} \beta_i x_i^2 + \epsilon_2$$



part-worths

product profile

noise (gumbel)

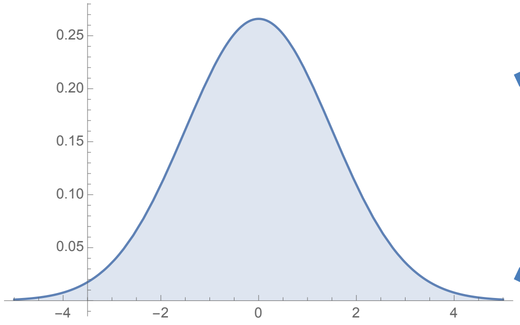
- Utility maximizing customer: $x^1 \succeq x^2 \Leftrightarrow U_1 \text{``}\geq\text{''} U_2$

- Noise can result in response error:

$$L\left(\beta \,|x^1 \;\succeq\; x^2\,\right) = \mathbb{P}\left(x^1 \;\succeq\; x^2 \mid \beta\right) = \frac{e^{\beta \cdot x^1}}{e^{\beta \cdot x^1} + e^{\beta \cdot x^2}}$$

Next Question To Reduce “Variance”: Bayesian

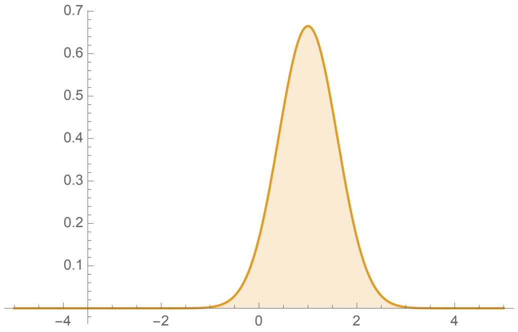
Prior Distribution of β



Feature	SX530	RX100
Zoom	50x	3.6x
Prize	$249.99	$399.99
Weight	15.68 ounces	7.5 ounces
Prefer	☑	☐

Bayesian Update

Posterior Distribution

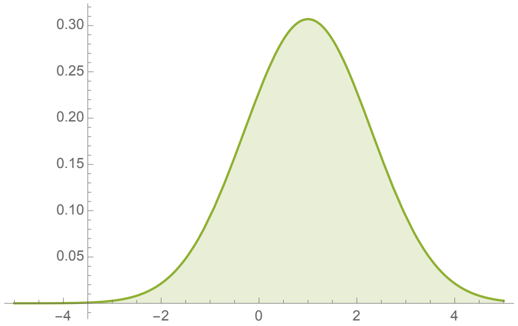


MCMC

Feature	TG-4	Galaxy 2
Waterproof	Yes	No
Prize	$249.99	$399.99
Viewfinder	Electronic	Optical
Prefer	☑	☐

Bayesian Update

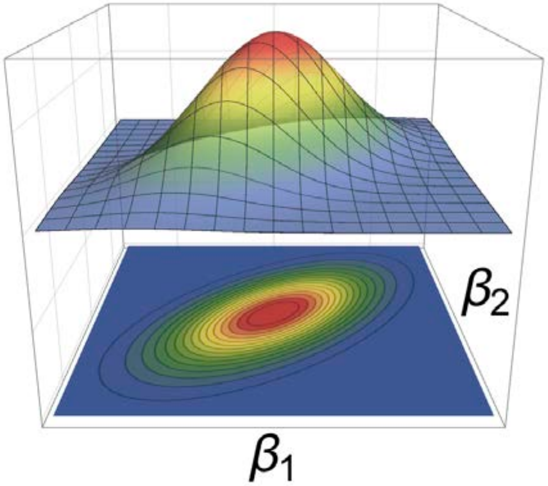
Posterior Distribution



- Black-box objective: Question Selection = Enumeration ☹
- Question selection by Mixed Integer Programming (MIP)

Bayesian Update and Geometric Updates

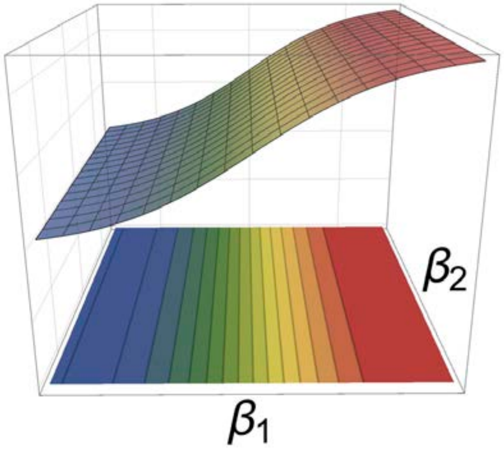
Prior distribution



$$\beta \sim N(\mu, \Sigma)$$

$$\phi(\beta; \mu, \Sigma)$$

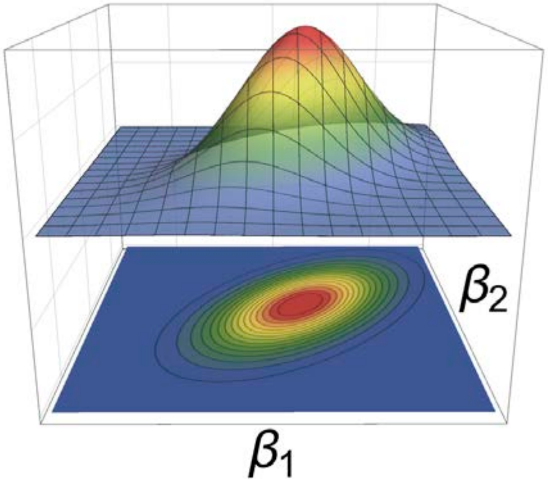
Answer likelihood



$$x^1 \succeq x^2$$

$$L\left(\beta \mid x^1 \succeq x^2\right)$$

Posterior distribution



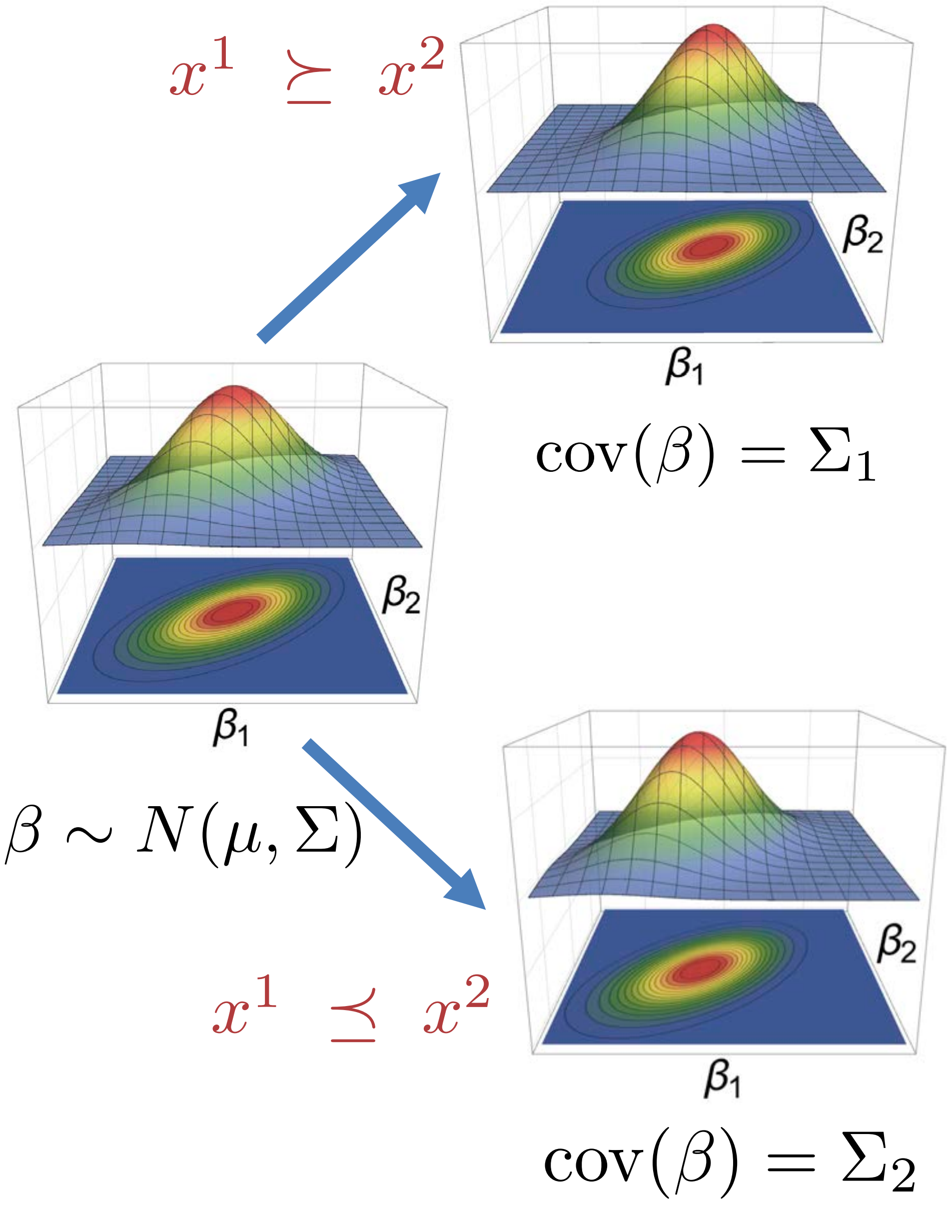
$$f\left(\beta \mid x^1 \succeq x^2\right)$$

Multidimensional Integration?

$$f\left(\beta \mid x^1 \succeq x^2\right) = \frac{\boxed{\phi\left(\beta\,;\ \mu, \Sigma\right) L\left(\beta \mid x^1 \succeq x^2\right)}}{\int_{\mathbb{R}} \phi\left(\beta\,;\ \mu, \Sigma\right) L\left(\beta \mid x^1 \succeq x^2\right) d\beta}$$

non-convex on $x^1, x^2 \in \{0, 1\}^n$

D-Efficiency and Posterior Covariance Matrix



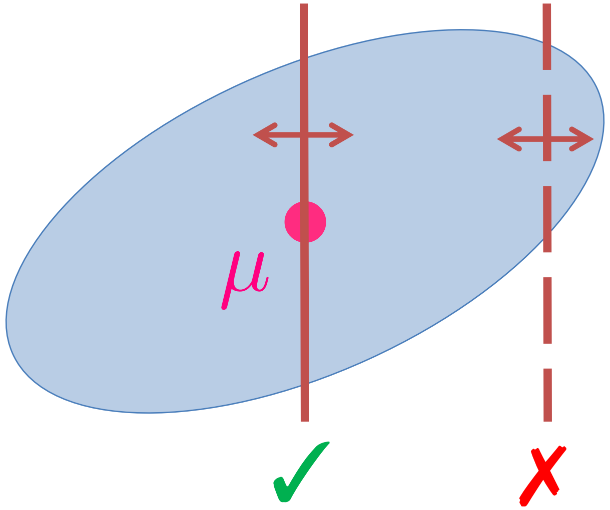
- “Variance” = D-Efficiency:
- $f\left(x^1, x^2\right) := \mathbb{E}_{\beta, x^1 \preceq/\succeq x^2}\left(\det(\Sigma_i)^{1/p}\right)$
- Non-convex function
- Even evaluating expected D-Efficiency for a question requires multidimensional integration

Standard Question Selection Criteria

$$(\beta - \mu)' \cdot \Sigma^{-1} \cdot (\beta - \mu) \leq r$$

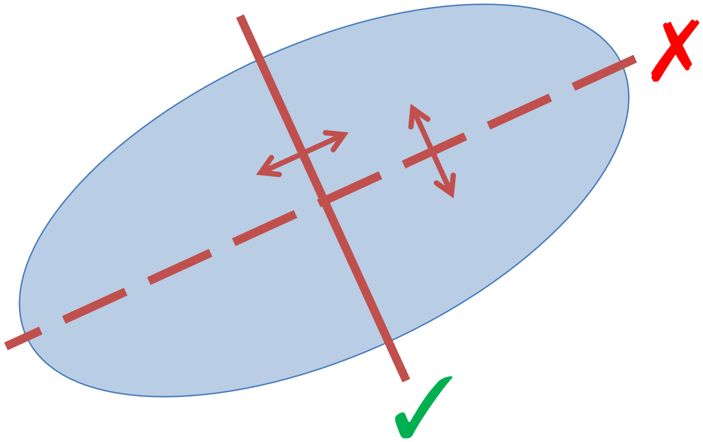
- Choice balance:
  - Minimize distance to center

$$\mu \cdot \left(x^1 - x^2\right)$$



- Postchoice symmetry:
  - Maximize variance of question

$$\left(x^1 - x^2\right)' \cdot \sum \cdot \left(x^1 - x^2\right)$$

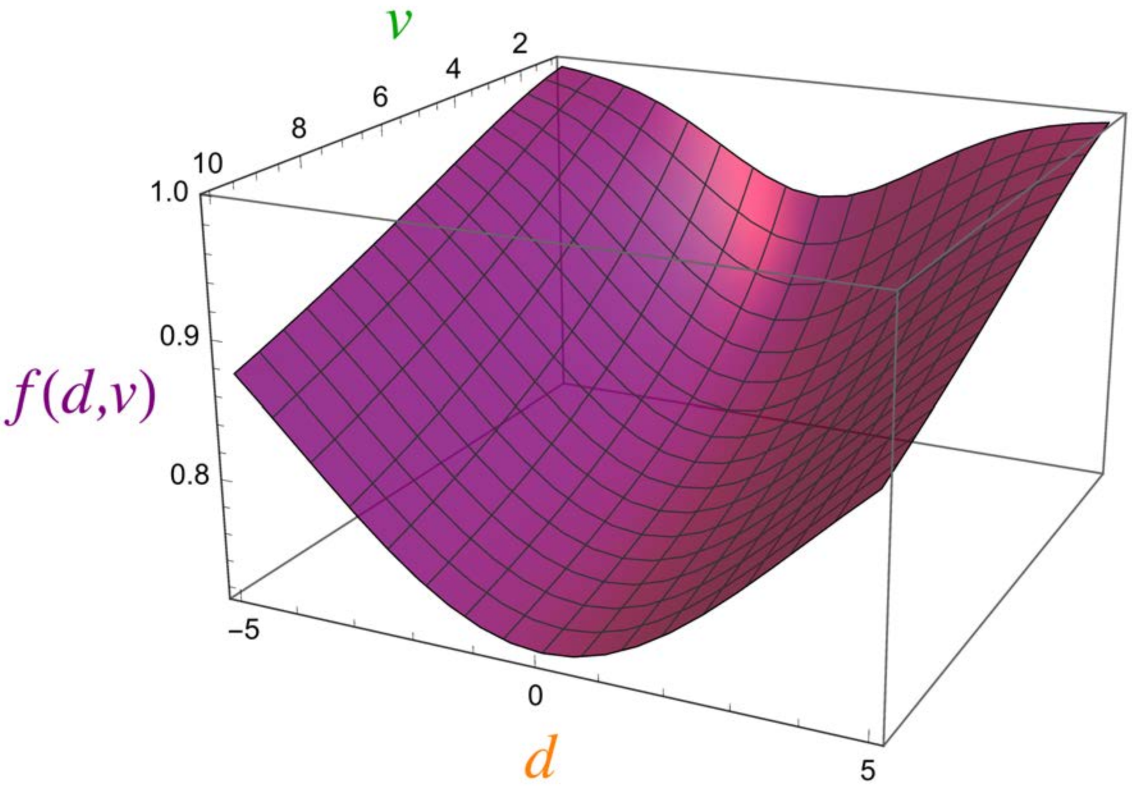


D-efficiency: Balance Question Trade-off

- D-efficiency = Non-convex function $f(d, v)$ of

  distance: $d := \mu \cdot (x^1 - x^2)$

  variance: $v := (x^1 - x^2)' \cdot \sum \cdot (x^1 - x^2)$



Can evaluate $f(d, v)$ with 1-dim integral 🙂

Optimization Model

$$\min \qquad f(d, v)$$ ✗

$$s.t.$$

$$\mu \cdot \left(x^1 - x^2\right) = d$$ ✓

$$\left(x^1 - x^2\right)' \cdot \sum \cdot \left(x^1 - x^2\right) = v$$ ✗

$$A^1 x^1 + A^2 x^2 \leq b$$ ✓

$$x^1 \neq x^2$$ ✗

$$x^1, x^2 \in \{0, 1\}^n$$

Technique 1: Binary Quadratic $x^1, x^2 \in \{0,1\}^n$

$$(x^1 - x^2)' \cdot \Sigma \cdot (x^1 - x^2) = v$$

$$X^l_{i,j} = x^l_i \cdot x^l_j \quad (l \in \{1,2\}, \quad i,j \in \{1,\ldots,n\}) :$$

$$X^l_{i,j} \leq x^l_i, \quad X^l_{i,j} \leq x^l_j, \quad X^l_{i,j} \geq x^l_i + x^l_j - 1, \quad X^l_{i,j} \geq 0$$

$$W_{i,j} = x^1_i \cdot x^2_j :$$

$$W_{i,j} \leq x^1_i, \quad W_{i,j} \leq x^2_j, \quad W_{i,j} \geq x^1_i + x^2_j - 1, \quad W_{i,j} \geq 0$$

$$\sum_{i,j=1}^{n} \left(X^1_{i,j} + X^2_{i,j} - W_{i,j} - W_{j,i}\right) \Sigma_{i,j} = v$$

Technique 1: Binary Quadratic $x^1, x^2 \in \{0,1\}^n$

$$x^1 \neq x^2 \quad \Leftrightarrow \quad \|x^1 - x^2\|_2^2 \geq 1$$

$$X^l_{i,j} = x^l_i \cdot x^l_j \quad (l \in \{1,2\}, \quad i,j \in \{1,\ldots,n\}):$$

$$X^l_{i,j} \leq x^l_i, \quad X^l_{i,j} \leq x^l_j, \quad X^l_{i,j} \geq x^l_i + x^l_j - 1, \quad X^l_{i,j} \geq 0$$

$$W_{i,j} = x^1_i \cdot x^2_j :$$

$$W_{i,j} \leq x^1_i, \quad W_{i,j} \leq x^2_j, \quad W_{i,j} \geq x^1_i + x^2_j - 1, \quad W_{i,j} \geq 0$$

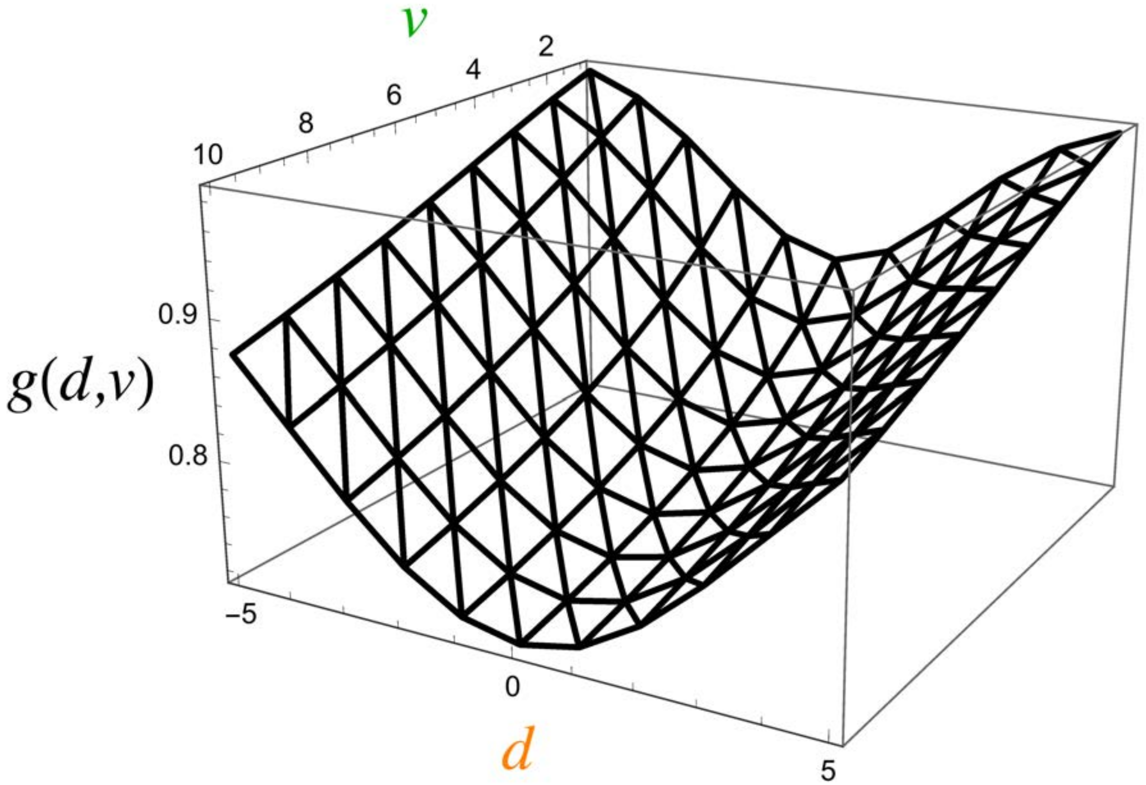
$$\sum_{i,j=1}^{n} \left(X^1_{i,j} + X^2_{i,j} - W_{i,j} - W_{j,i}\right) \geq 1$$

Technique 2: Piecewise Linear Functions

- D-efficiency = Non-convex function $f(d,v)$ of

distance: $d := \mu \cdot (x^1 - x^2)$

variance: $v := (x^1 - x^2)' \cdot \sum \cdot (x^1 - x^2)$



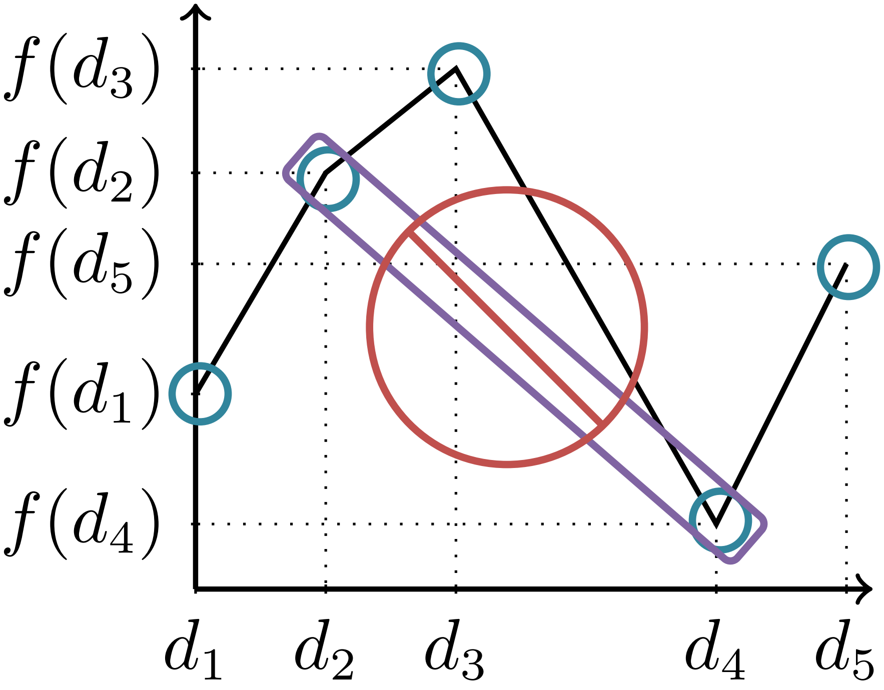
Can evaluate $f(d,v)$ with 1-dim integral 🙂

Piecewise Linear Interpolation

MIP formulation

Simple Formulation for Univariate Functions

$$z = f(x)$$



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

Non-Ideal: Fractional Extreme Points

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

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

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

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

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

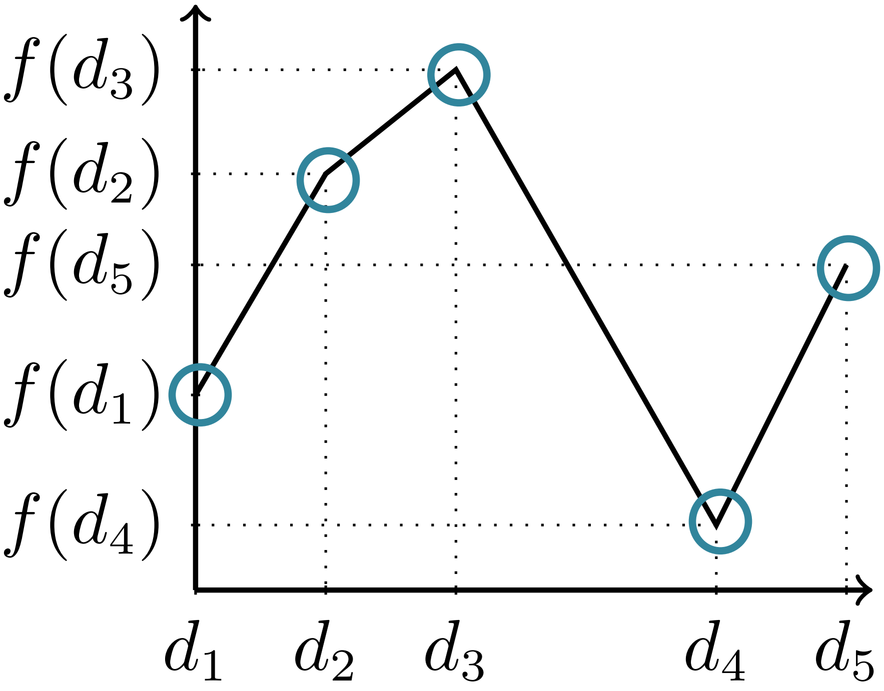
$$0 \leq \lambda_3 \leq y_2 + y_3$$

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

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

Advanced Formulation for Univariate Functions

$$z = f(x)$$



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

Ideal: Integral Extreme Points

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

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

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

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

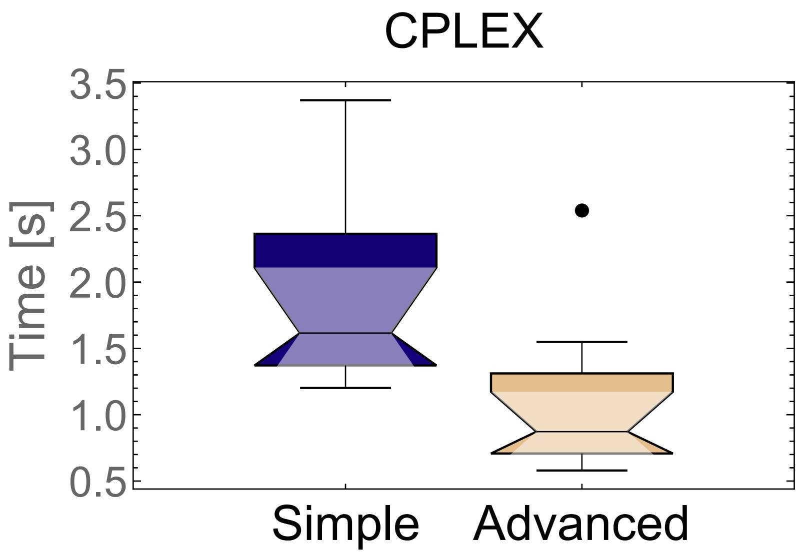
$$0 \leq \lambda_3 \leq y_1$$

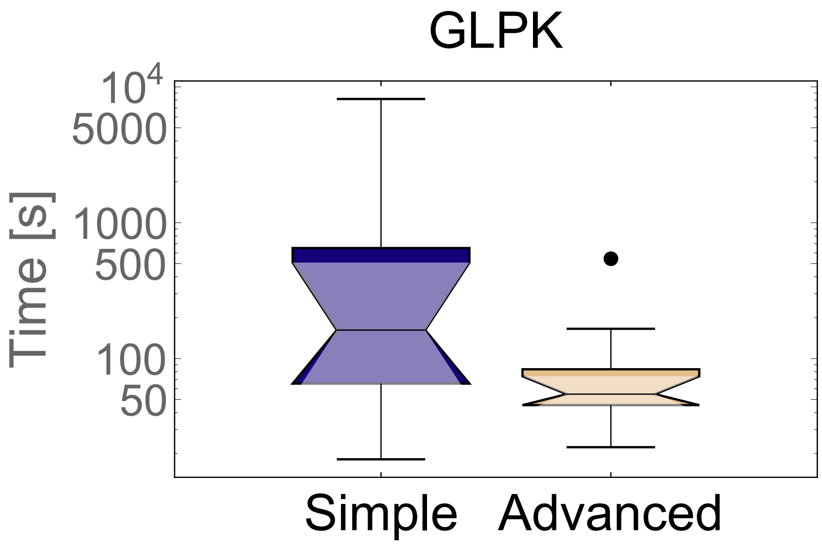
$$0 \leq \lambda_4 + \lambda_5 \leq 1 - y_2$$

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

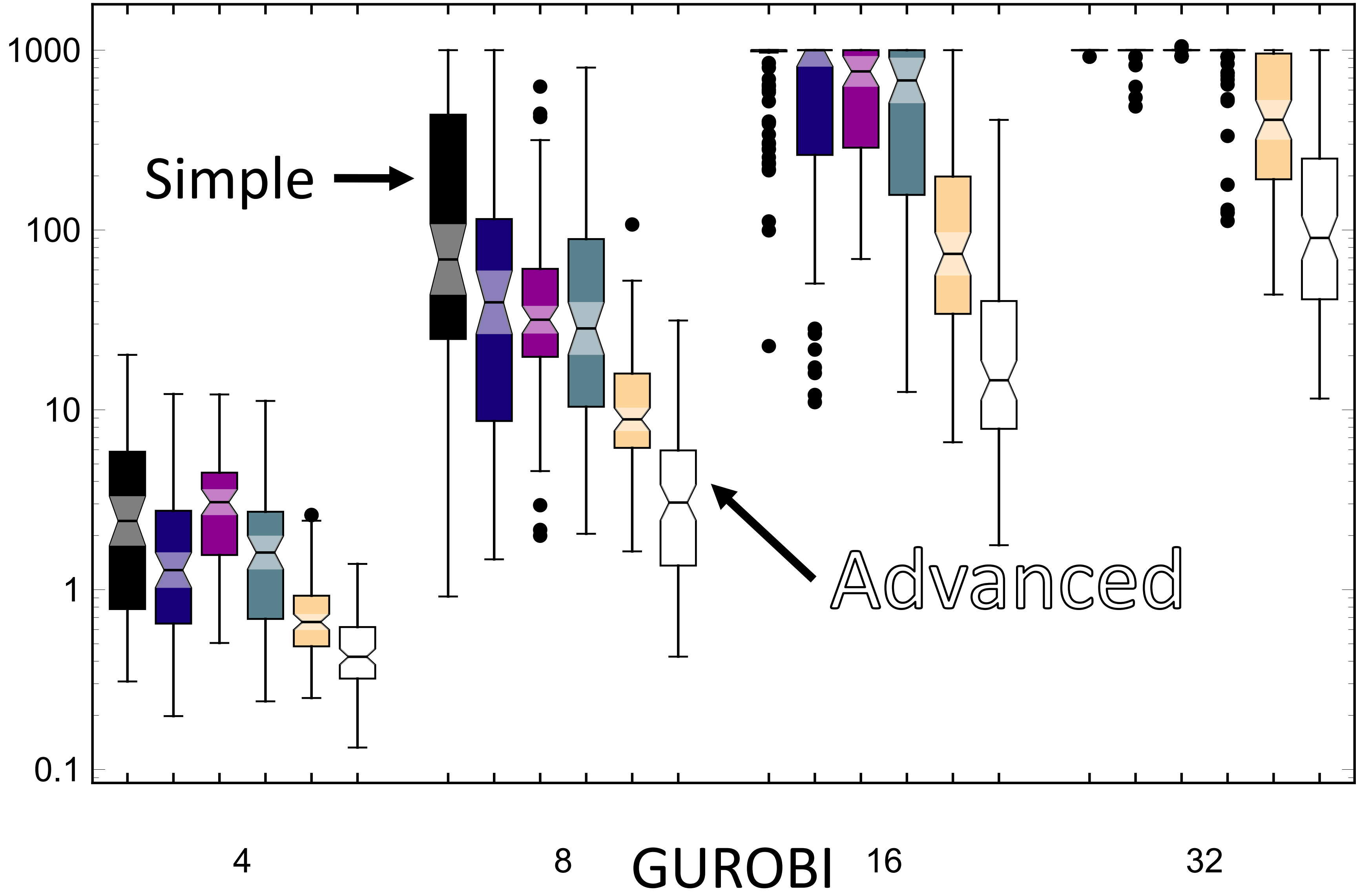
Computational Performance

- 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

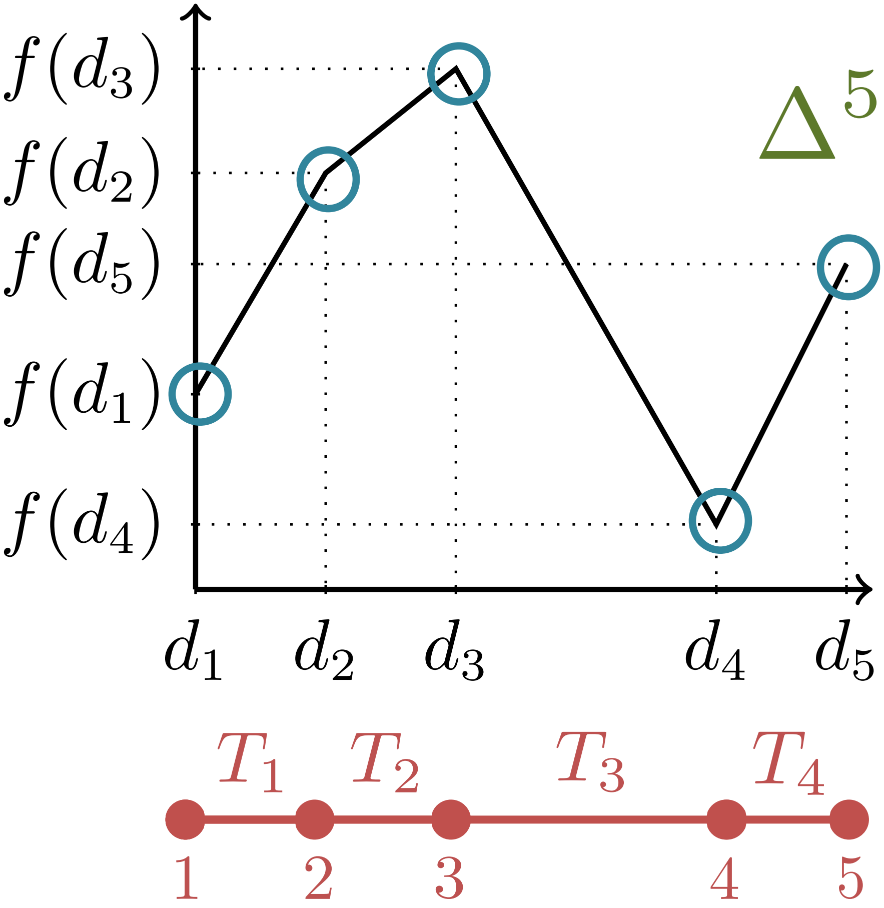
Abstracting Univariate Functions

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

$$z = f(x)$$

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

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

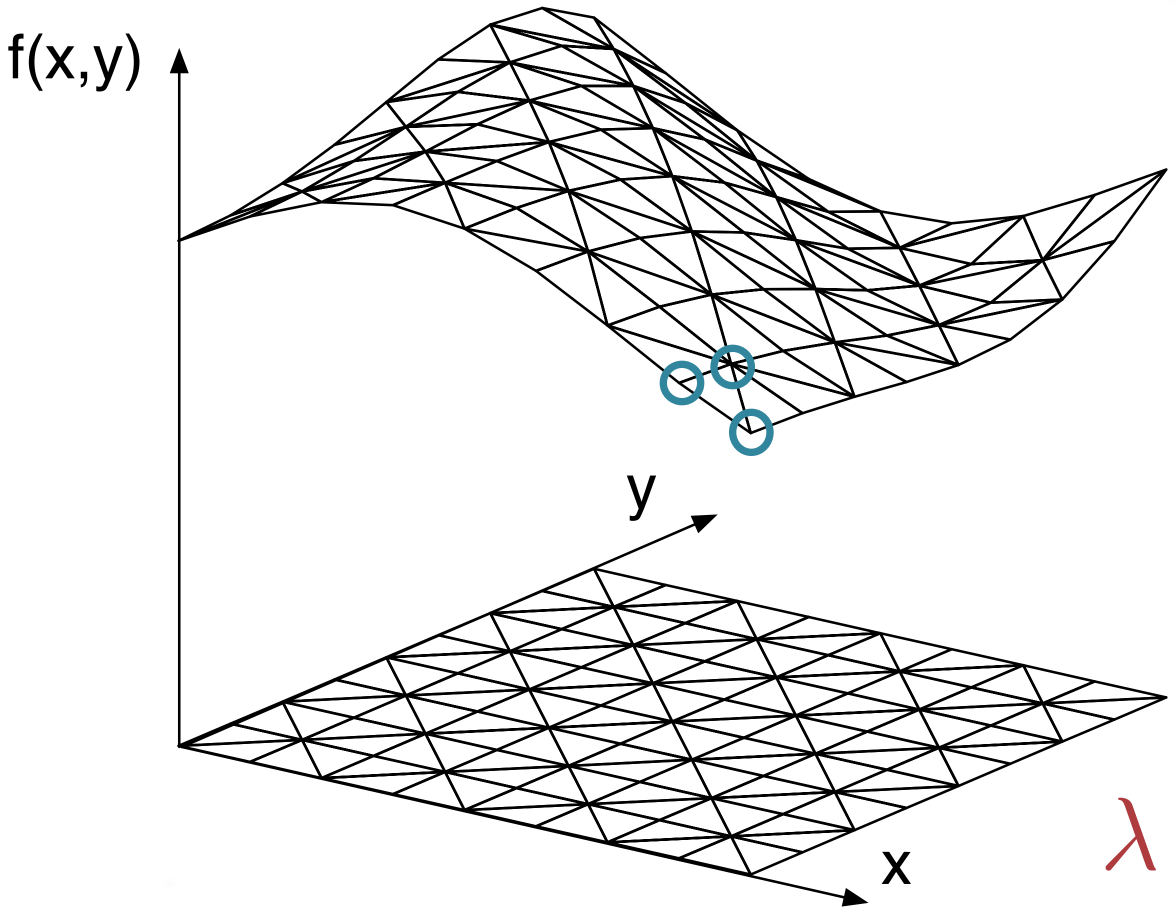


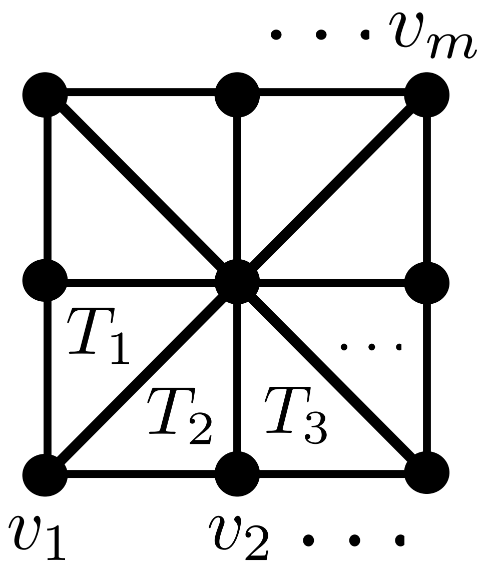
$$\Delta^5 = \boxed{1 = \sum_{j=1}^{5} \lambda_j , \quad \lambda_j \geq 0}$$

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

Abstraction Works for Multivariate Functions

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

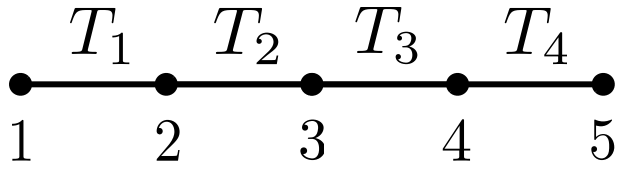


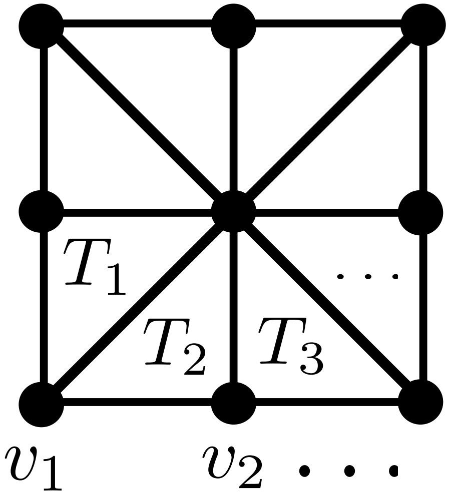


$$\lambda \in \bigcup_{i=1}^{n} P_i \subseteq \Delta^m$$

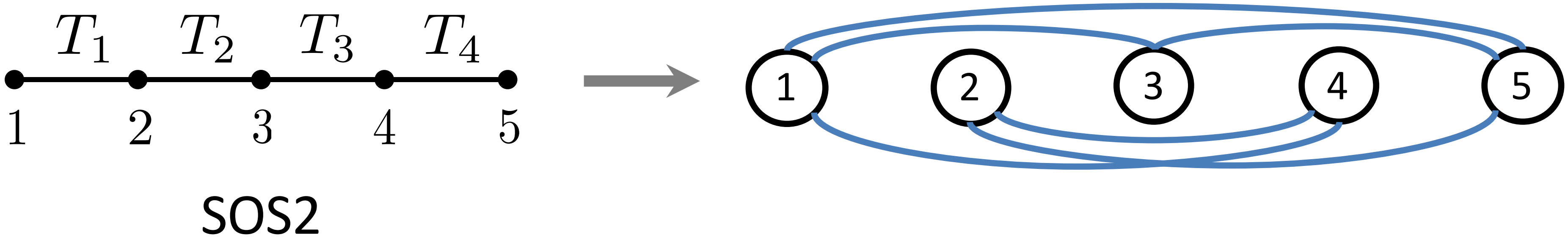
Complete Abstraction

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

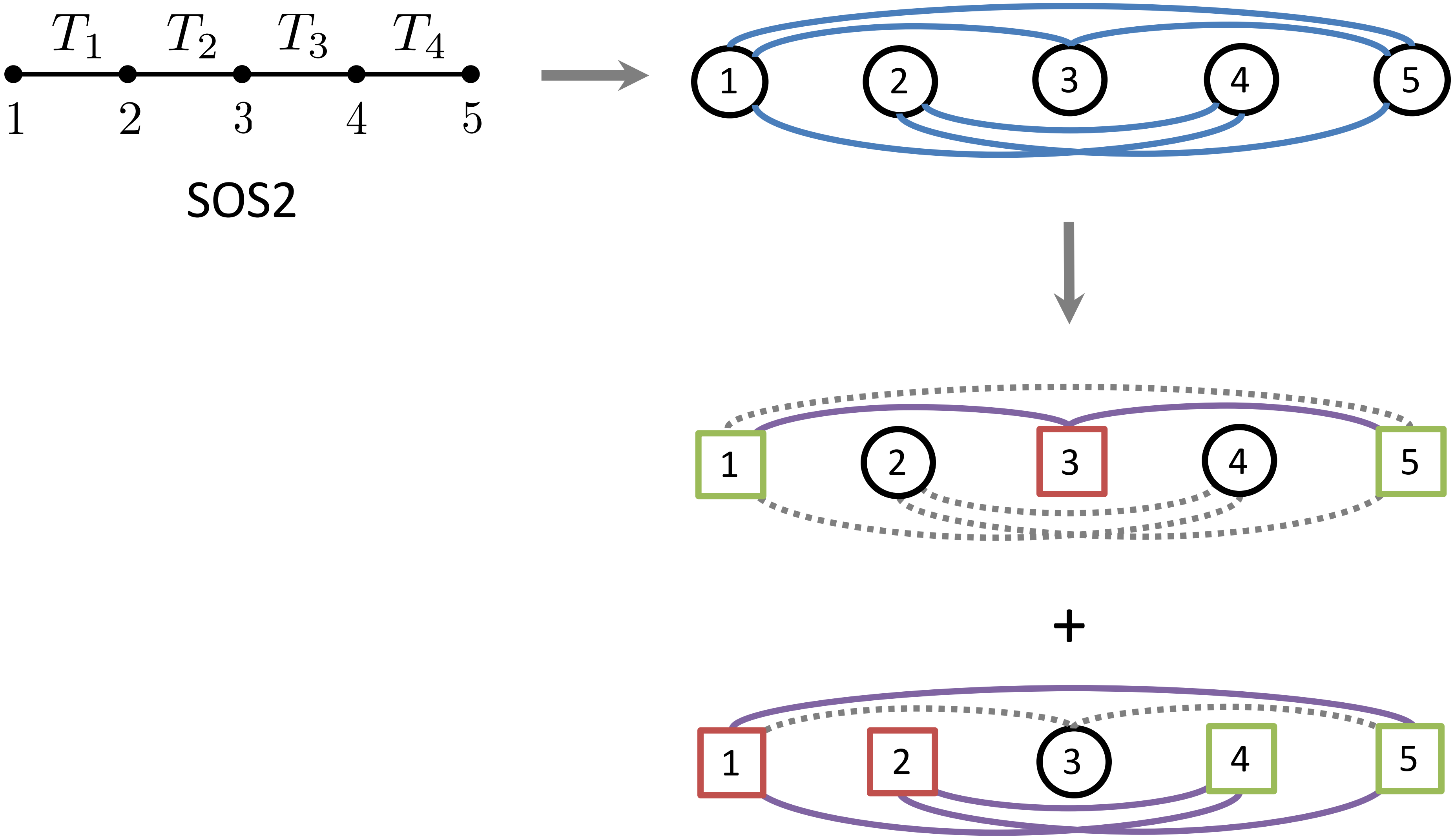




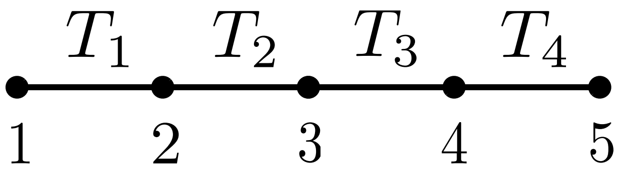
From Cliques to (Complement) Conflict Graph



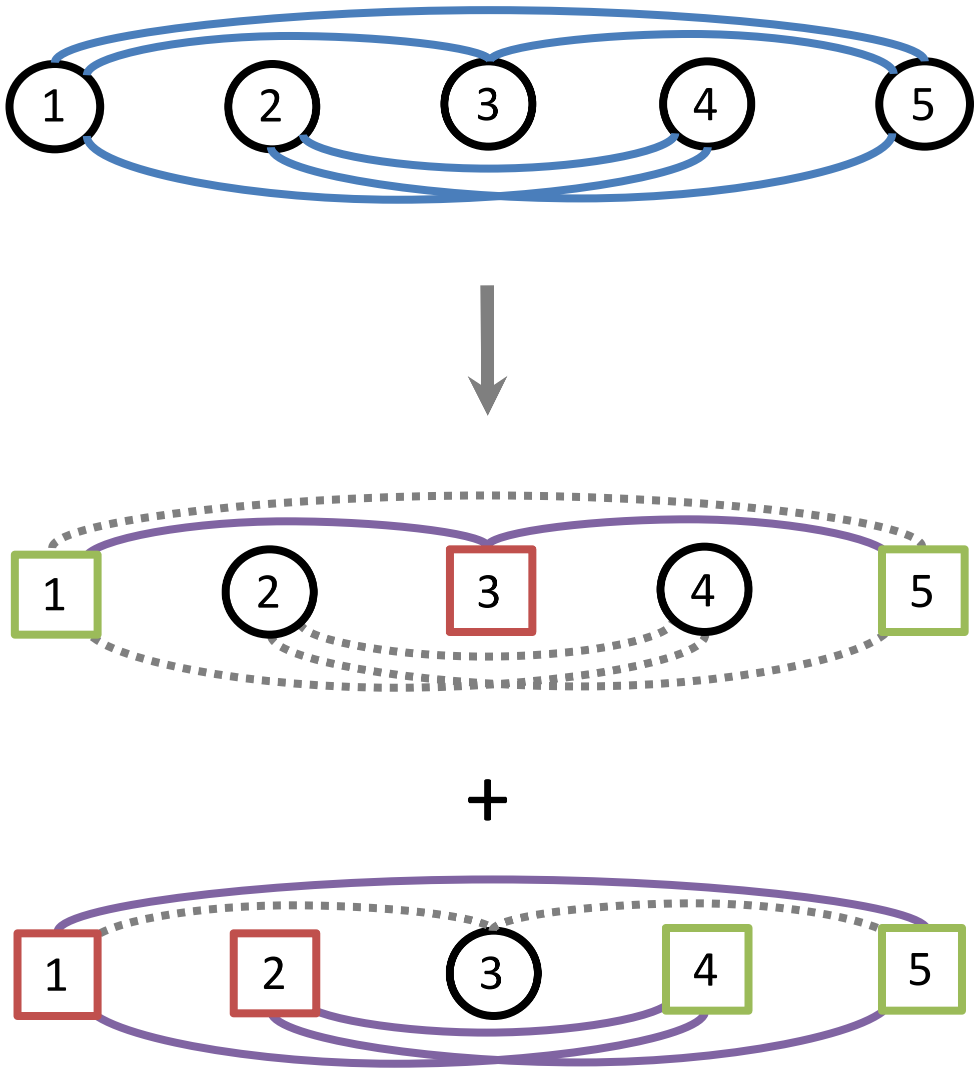
From Conflict Graph to Bi-clique Cover



From Bi-clique Cover to Formulation



SOS2



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

Ideal Formulation from Bi-clique Cover

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

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

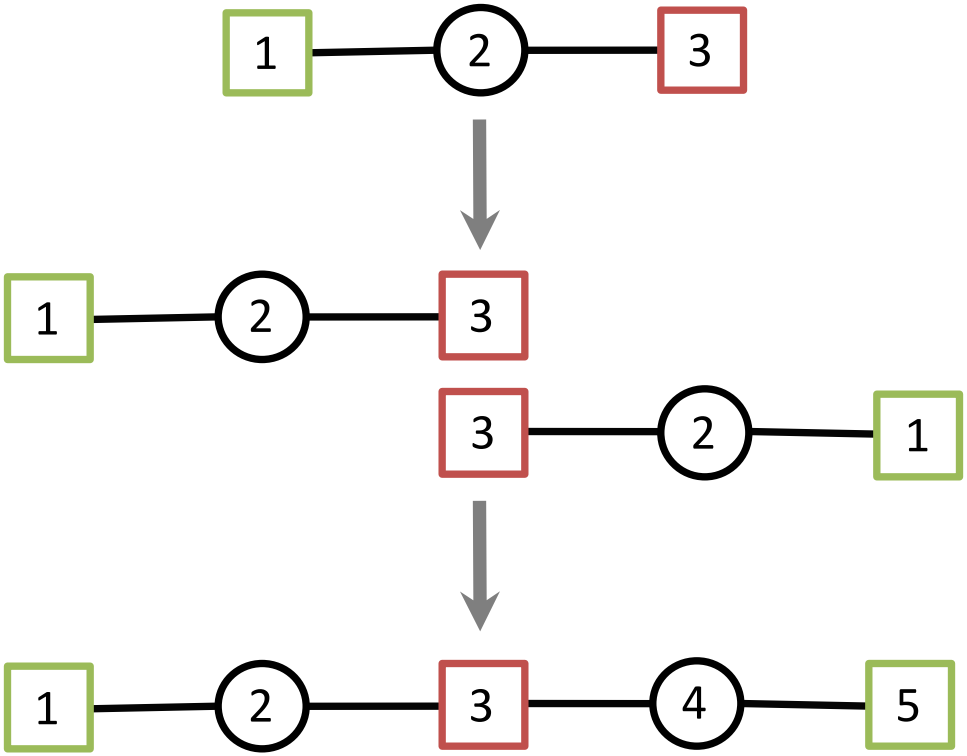
- Bi-clique cover $\left\{\left(A^j, B^j\right)\right\}_{j=1}^{t}, \quad A^j, B^j \subseteq V$

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

- Formulation

$$\sum_{v \in A^j} \lambda_v \leq 1 - y_j \quad \forall j \in [t]$$
$$\sum_{v \in B^j} \lambda_v \leq y_j \qquad \forall j \in [t]$$
$$y \in \{0, 1\}^t$$

Recursive Construction of Cover for SOS2, Step 1



Base case $n=2^1$:

Step 1 recursion :

Reflect Graph / Cover

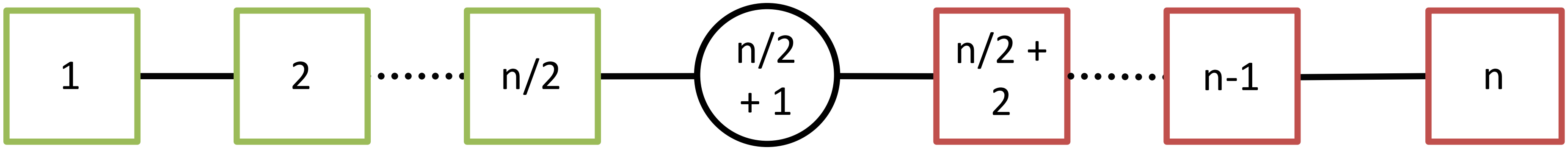
Stick Graph / Cover

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

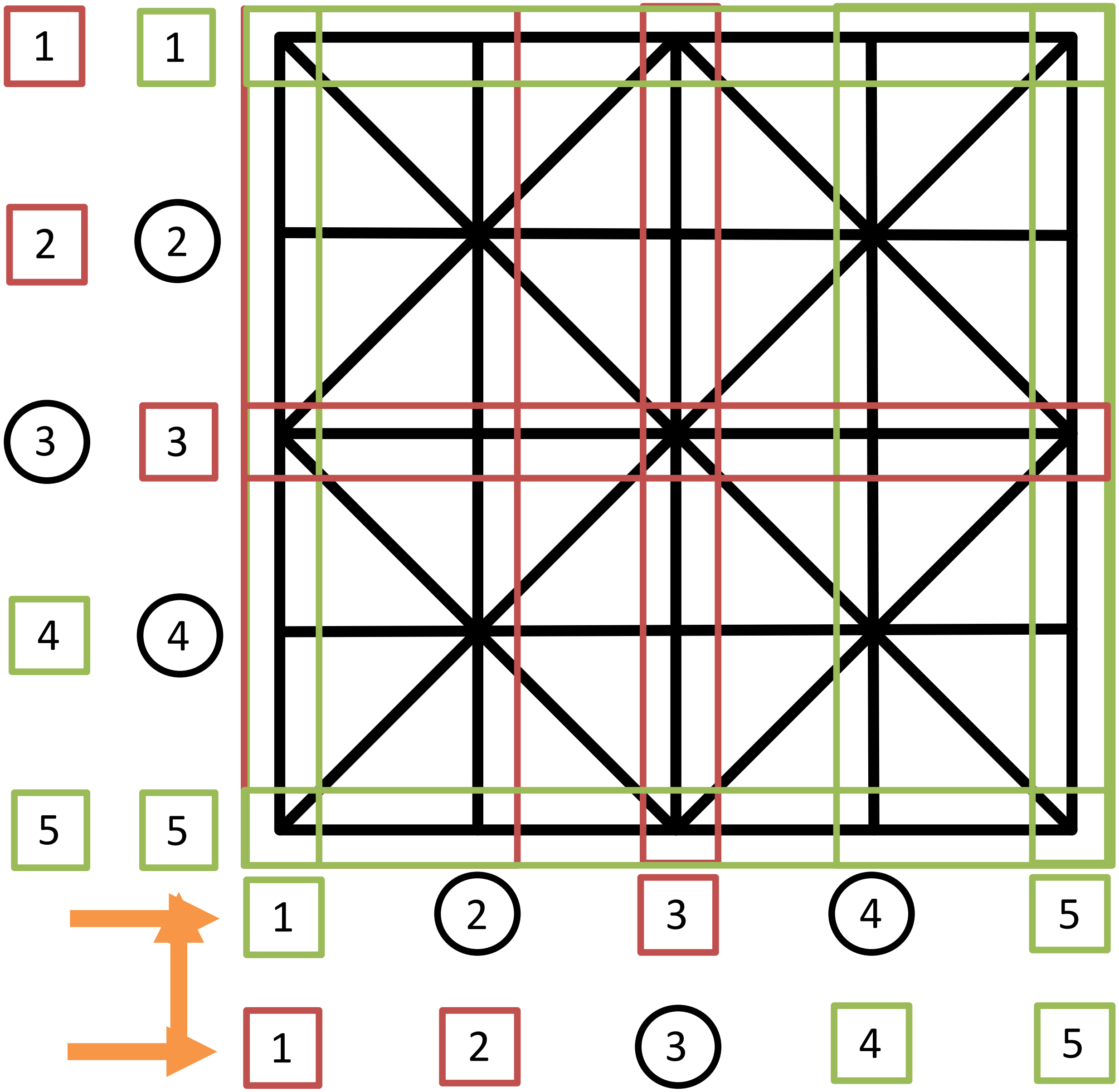


$$\text{Cover has } \log_2 n \text{ bi-cliques.}$$

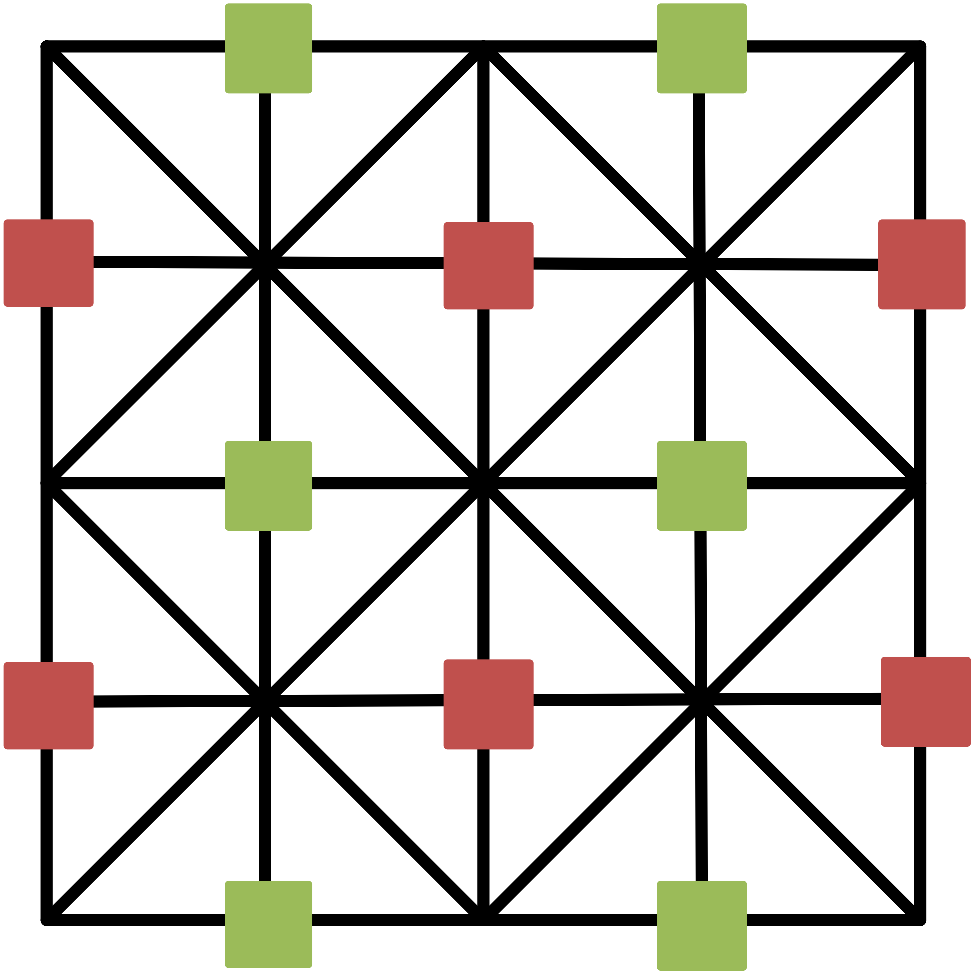
For non-power of two just delete extra nodes.

Grid Triangulations: Step 1 = SOS2 for Inter-Box

Covers all arcs between boxes



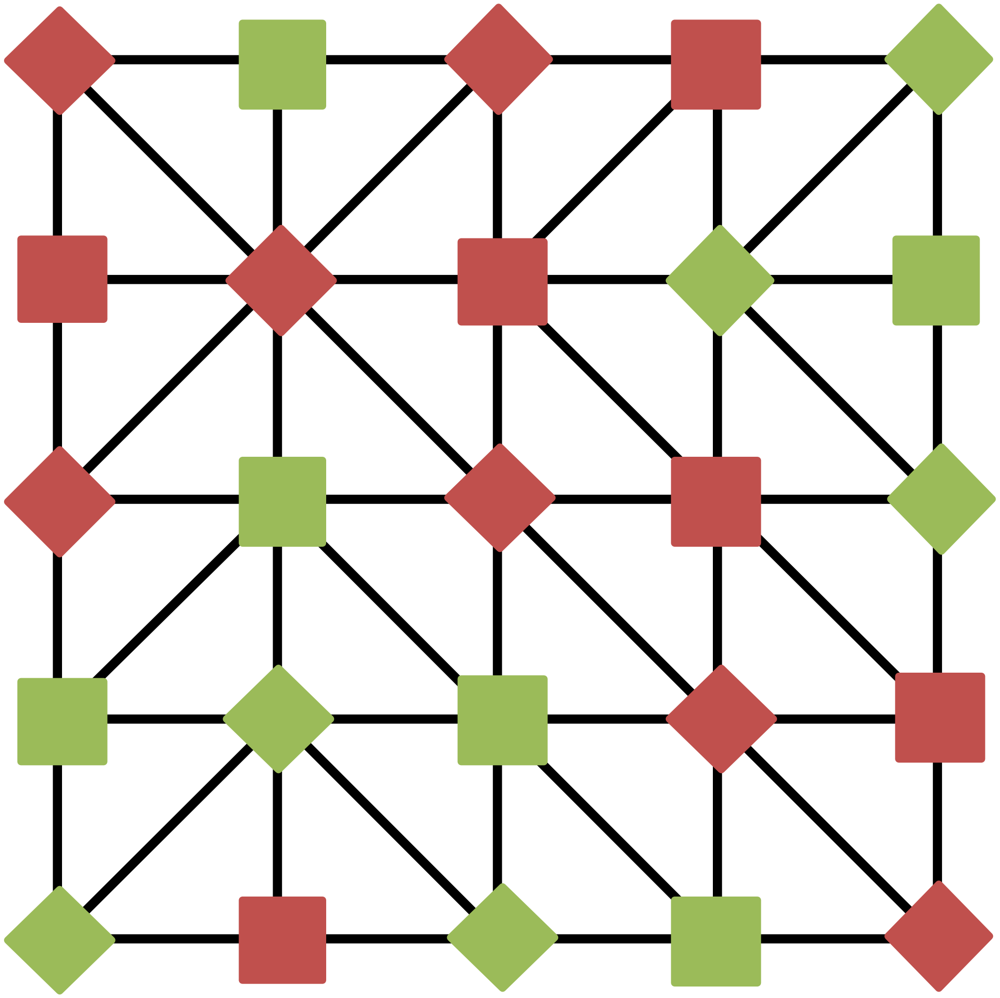
Grid Triangulations: Step 2 = Ad-hoc Intra-Box



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

Summary and Main Messages

- Always choose Chewbacca!

- MIP can solve very challenging problems in practice
- Commercial solvers best, but free solvers reasonable
  - Both easily accessible and integrated into complex systems through the JuMP
- Advanced formulations yield important speed-ups and are (relatively) easy to learn

More Information

- JuMP:
  - Ask Miles and https://github.com/JuliaOpt/JuMP.jl
- MIP Formulations:
  - Mixed integer linear programming formulation techniques. V. SIAM Review 57, 2015. pp. 3-57.
- Advanced Formulation:
  - Small independent branching formulations for unions of V-polyhedra. Joey Huchette and V. 2016. arXiv:1607.04803
- Marketing Application:
  - Ellipsoidal methods for adaptive choice-based conjoint analysis. Denis Saure and V. 2016. http://ssrn.com/abstract=2798984