Towards Certificates for Integer Programming Computations

SNL ASCR Discrete Math/optimization research program

Robert Carr, Sandia National Laboratories
Harvey Greenberg, U. Colordo, Denver
Ojas Parekh, Sandia National Laboratories
Cynthia Phillips, Sandia National Laboratories

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation,
A wholly owned subsidiary of Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under contract DE-AC04-94AL85000.

(Mixed) Integer Programming (IP)

Min $c^T x$

Subject to: $Ax \geq b$

$\ell \leq x \leq u$

$x = (x_I, x_C)$

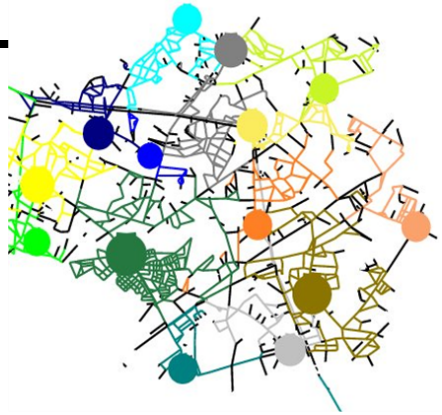
$x_I \in Z^n$ (integer values)

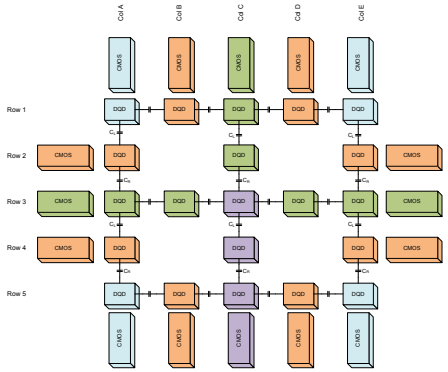
$x_C \in Q^n$ (rational values)

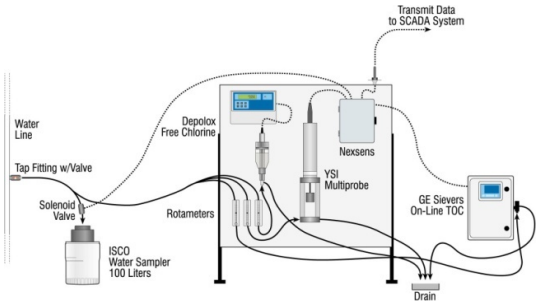
- (Easily) express NP-complete combinatorial optimization problems
  - Resource allocation, study of natural systems
- PICO: SNL's massively parallel solver
- Instance-specific proofs of quality
- Benchmarking heuristics, study structure

MIP Applications (Sample)

- Sensor placement (municipal water systems, roadways)
- Network Interdiction (vulnerability analysis)
- Scheduling quantum error correction (quantum computing architecture)
- Management of unattended ground sensors
  - Volcanoes, subway tunnels, building integrity
- Bioinformatics: protein structure prediction/ comparison, protein-protein docking, protein folding
- Meshing (for simulating physical systems)
- Space-filling curves – preprocessor for fast heuristic node allocator for MP machines
- Energy system and energy/water planning
- DOE enterprise transformation
- Conference scheduling, reviewing allocation







Sandia National Laboratories

MIP Certificates

- Making irrevocable, expensive and/or critical decisions
- Confidence (proof) computation is correct
  - Implementation errors
  - Numerical issues
- Certificate
  - External program can check correctness (within tolerance)
  - Faster and simpler to check than original computation
- TSP example: Applegate, Bixby, Chvatal, Cook, Espinoza, Goycoolea, Helsgaun (ORL, 2009)
- MIPLIB2010 exact solution checker

Linear programming (LP) relaxation of an IP

Min $c^T x$

Subject to:

$$Ax = b$$

$$\ell \le x \le u$$

$$x = (x_I, x_C)$$

$$\cancel{x_I \in Z^n} \quad \text{(integer values)}$$

$$x_C \in Q^n \quad \text{(rational values)}$$

- LP can be solved efficiently (in theory and practice)
- LP optimal gives lower bound

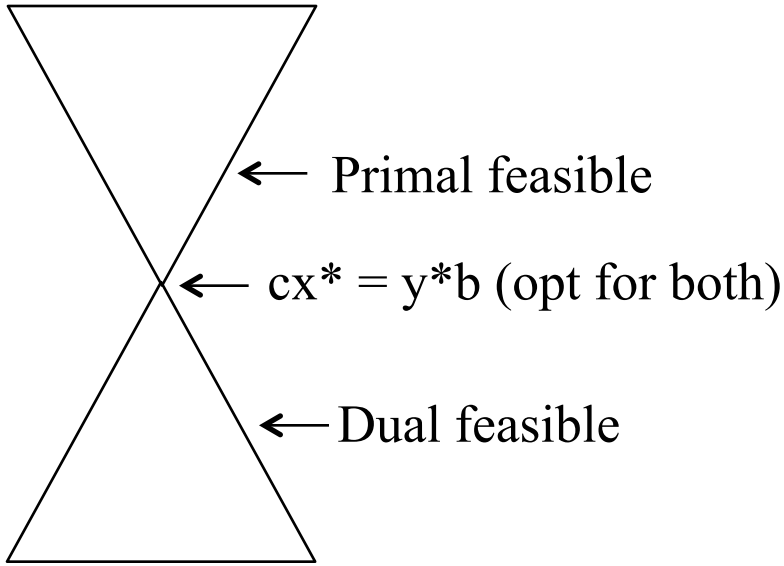
Linear programming Certificate

Primal LP:

$$\min cx$$
$$\text{subject to } Ax \geq b$$
$$x \geq 0$$

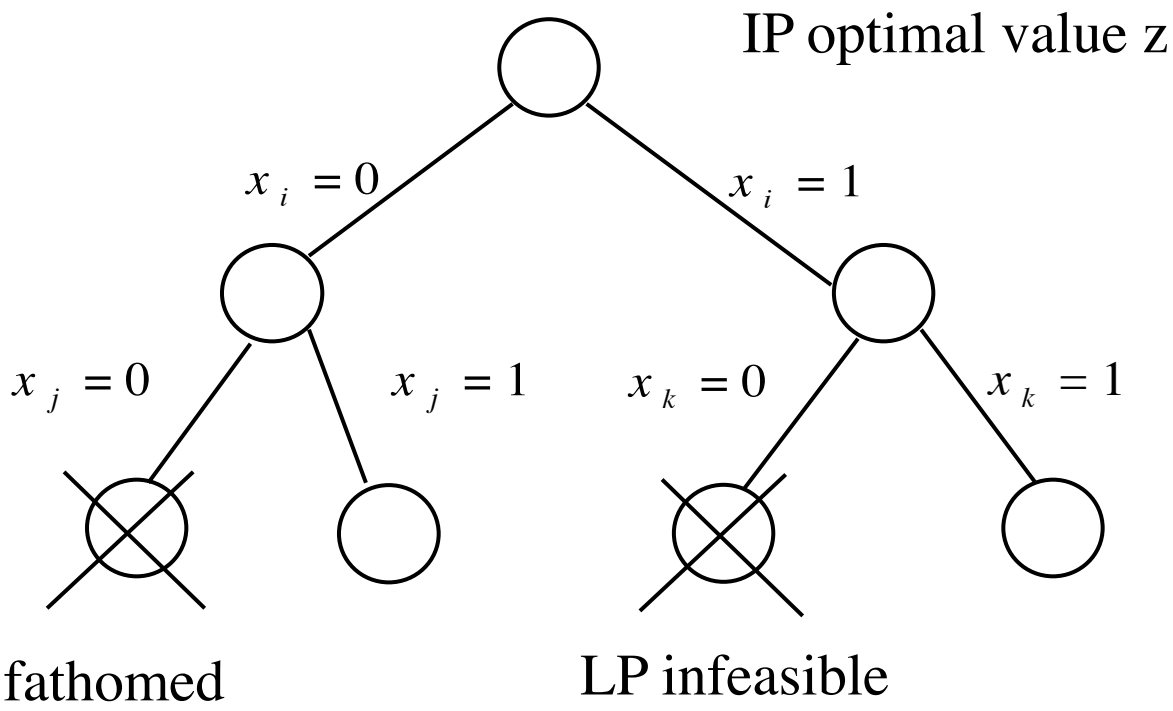
LP Dual:

$$\max yb$$
$$\text{subject to } yA \leq c$$
$$y \geq 0$$



At optimality cx* = y*b (if both primal and dual feasible)
(x*,y*) is a certificate for the LP. Fast check.

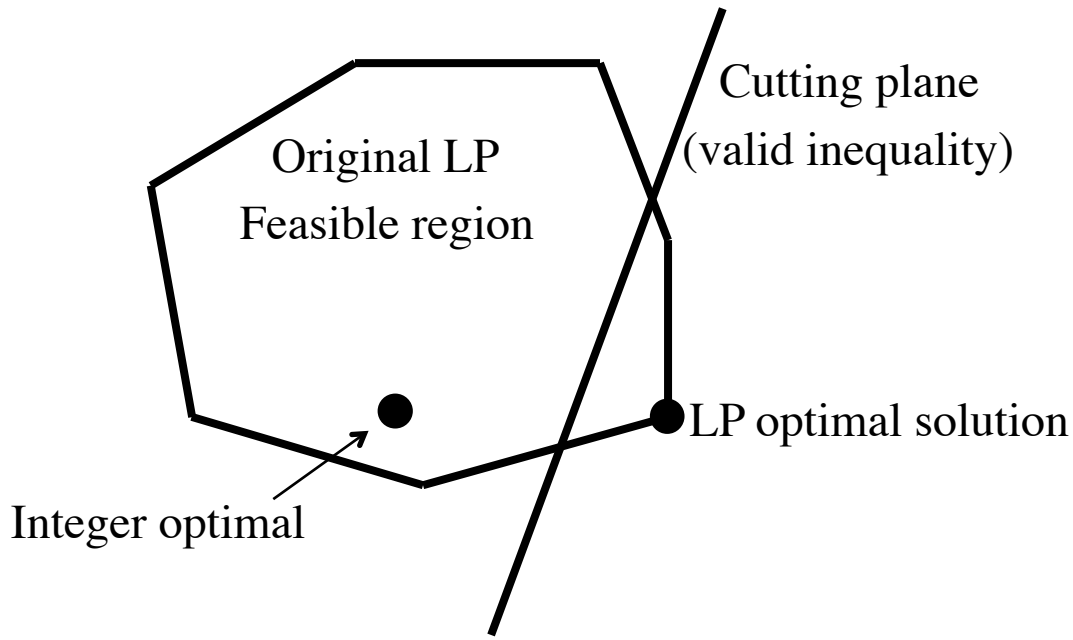
Certificate from Branch and Bound Tree



Dual feasible y st
yb > z
OR LP opt is integer feasible

Consider
min x_k for parent
Give yb > 0

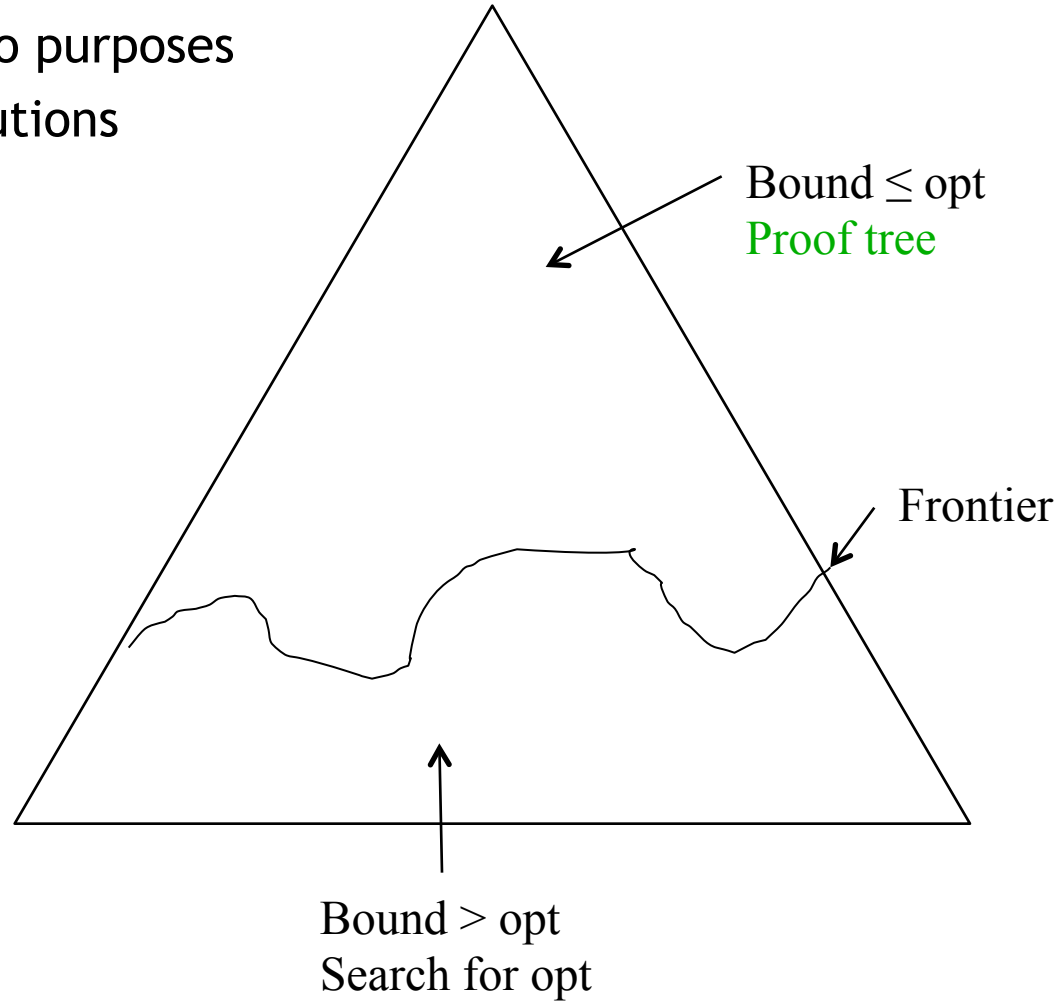
Strengthen LP Bound with Cutting Planes



- Make LP polytope closer to integer polytope
- Use families of constraints too large to explicitly list
- Separation algorithm: efficiently determine if all constraints are satisfied, or return a violated inequality
- Make progress while minimizing/delaying branching

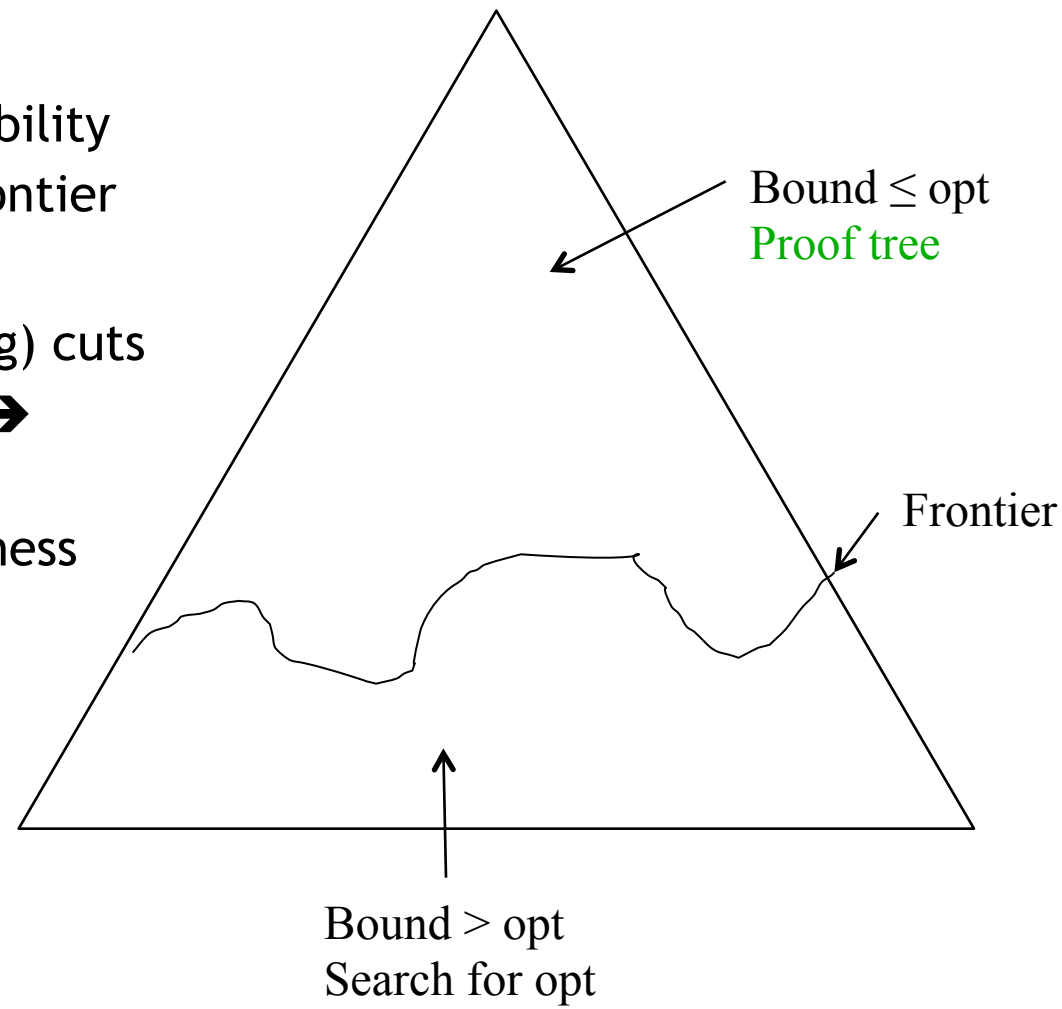
Branch and Bound for Integer programming

- Branch and bound has two purposes
  - Find the optimal solutions
    - Feasible leaf
    - Heuristics
  - Prove optimality



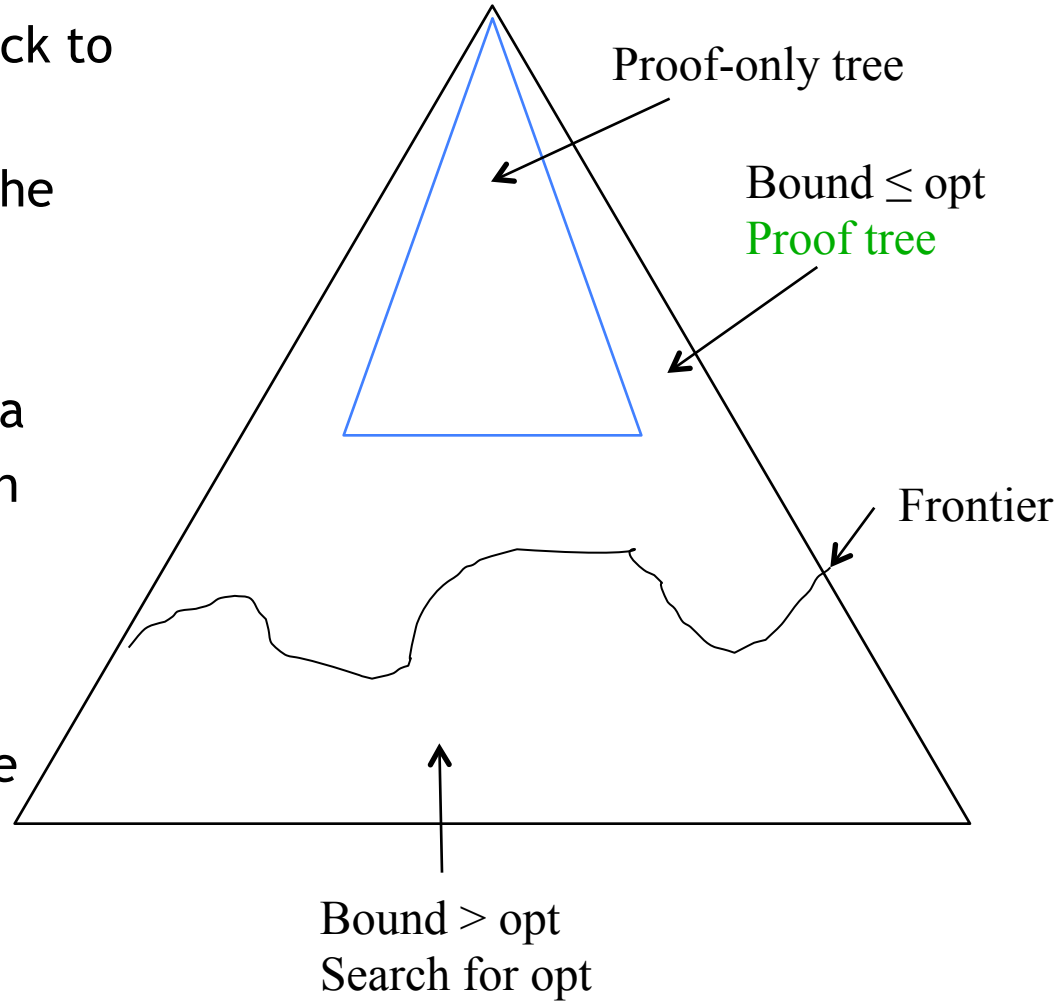
Brute Force Certificate

- Optimal Solution
  - Verifier checks feasibility
- For every node on the frontier
  - Branching decisions
  - Set of active (binding) cuts
    - Removing 1 cut ➔ bound too low
  - Proof of cut correctness
  - LP certificate



Brute Force Certificate - Which Frontier?

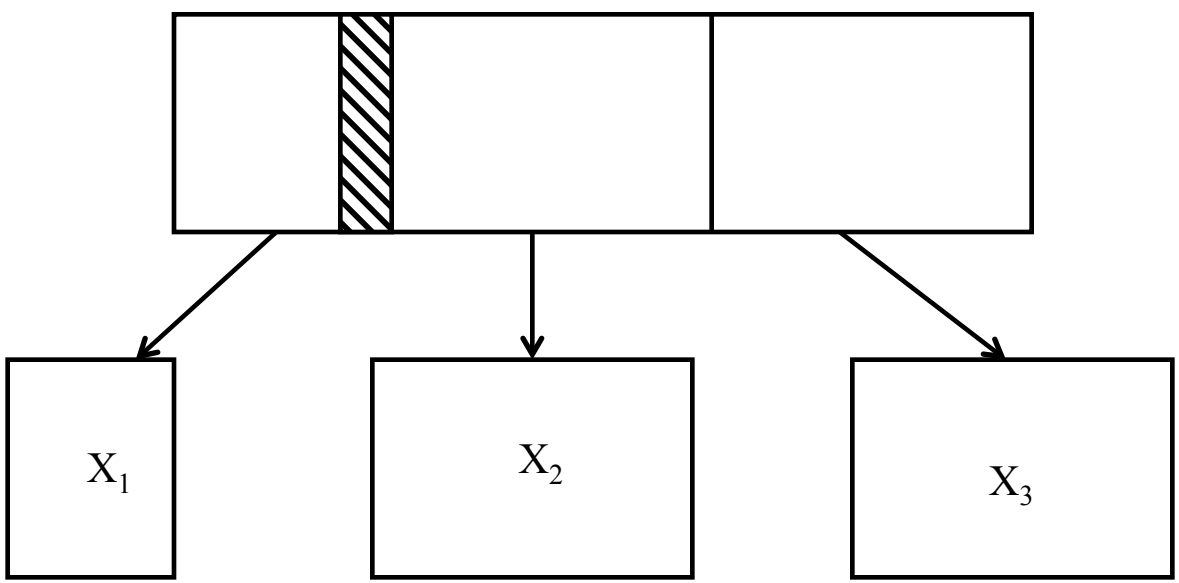
- Save B&B tree and cut back to frontier
- Rerun B&B pruning with the optimal value
  - May be be smaller
    - Old miplib enigma
    - 43% avg reduction
    - 2.3% to 96.8% Preliminary!
  - Can bias search
    - Solution structure
    - Gradients
  - Can do in parallel



Sandia National Laboratories

Brute Force: Proof of coverage

- Subproblems must cover the feasible space
- For simple variable bounds (x=0, x=1), give branching tree
  - In principle can let the verifier check from frontier bounds
- Simple branch on constraint: $ax \leq b$ and $ax \geq b$
- User defined branching
- In general can prove uncovered regions empty



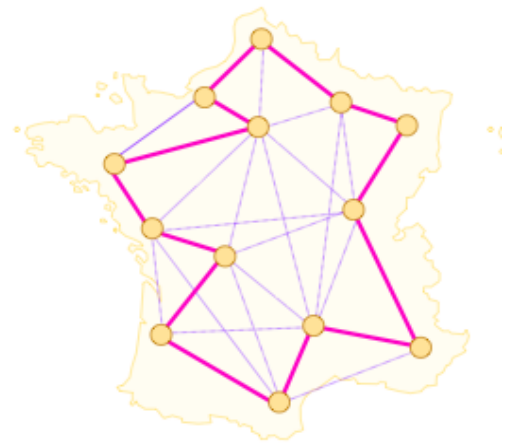
Cut Certificates

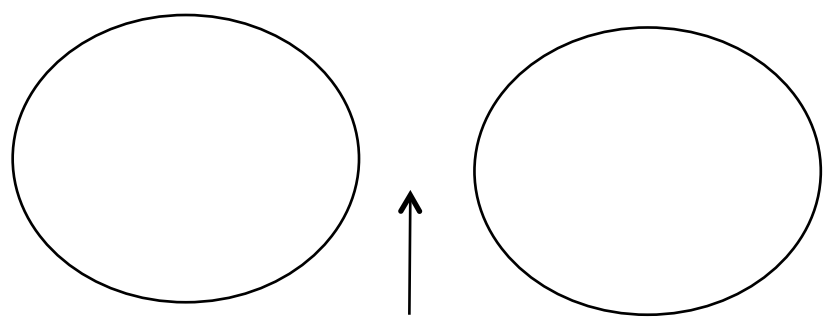
- Best case: class-specific cuts
  - Relies on theorems
- Example: TSP

$$\min \sum c_{ij} x_{ij}$$

- Subject to: $\sum_i x_{ij} = 2 \quad \forall j$

$$x_{ij} \in \{0,1\}$$





Disconnected subtours

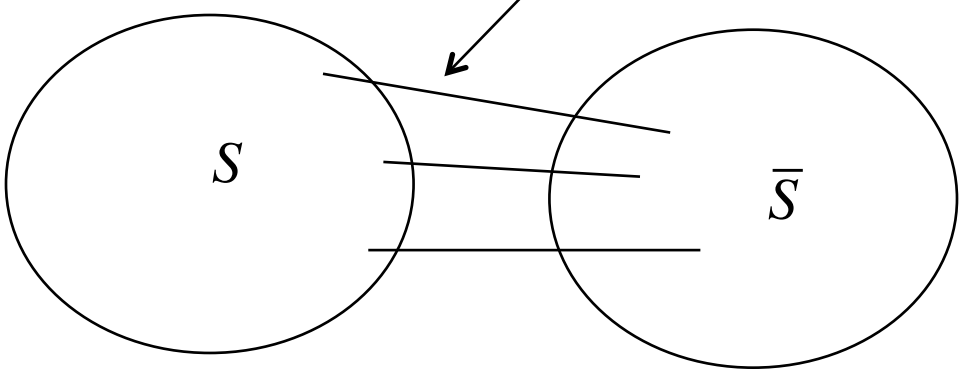
Should have at least
2 edges between each
Partition of nodes

Example: Traveling Salesman Problem

subtour elimination contraint:

$$\sum_{\substack{(i,j)\in E \\ i\in S, j\in \bar{S}}} x_{ij} \geq 2$$

Useful if LP for relaxation edge weights x_{ij}^* for this cut < 2

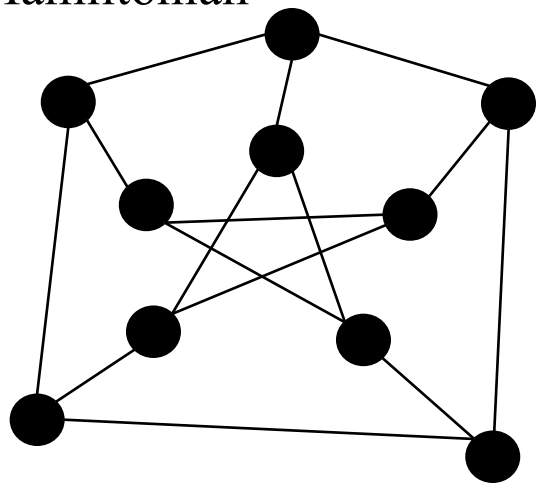


- Separate with min cut (efficient using LP values)
- Certificate is the cut itself or S

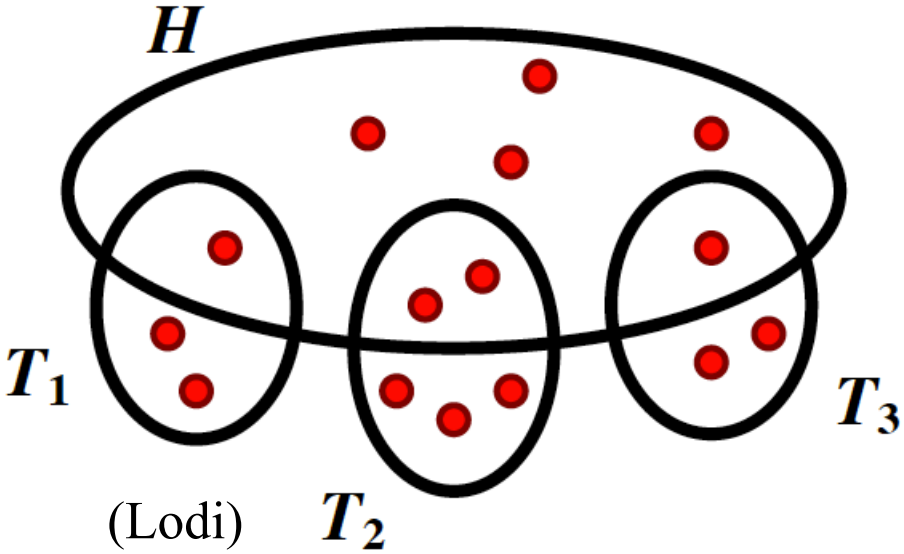
Traveling Salesman: Cut examples

- TSP has lots of cuts that are structural
  - Relies on theorems
- Proof of correctness is proving structure: graph isomorphism
  - Map nodes to roles

Hypo-Hamiltonian



Comb



(Lodi)

Example: Gomory Cuts

$$\text{Tableau row: } x_B + \sum a_i x_i = a_0$$

$$\text{Gomory cut: } \Rightarrow \sum_{f_i \le f_0} f_i x_i + \sum_{f_i > f_0} \frac{f_0(1-f_i)}{1-f_0} x_i \ge f_0,$$

$$\text{where } f_i = a_i - \lfloor a_i \rfloor \text{ and } f_0 = a_0 - \lfloor a_0 \rfloor$$

- Standard proof of validity is nontrivial
- Naïve certificate:
  - Certificate: tableau row
  - Verification: check if in Gomory cut form and apply standard proof/derivation to tableau row

Sandia National Laboratories

A Better Certificate

- Every valid ineq generated as positive combination of facets:

$$\text{Input: } ax \geq \gamma \text{ valid for } \{x : Ax \geq b\}$$

$$\text{Certificate: } \Rightarrow \exists \lambda \geq 0 \text{ s.t. } \lambda^T Ax \geq \lambda^T b \text{ implies } ax \geq \gamma$$

$$\text{Verification: } a \geq \lambda^T A \text{ and } \lambda^T b \geq \gamma$$

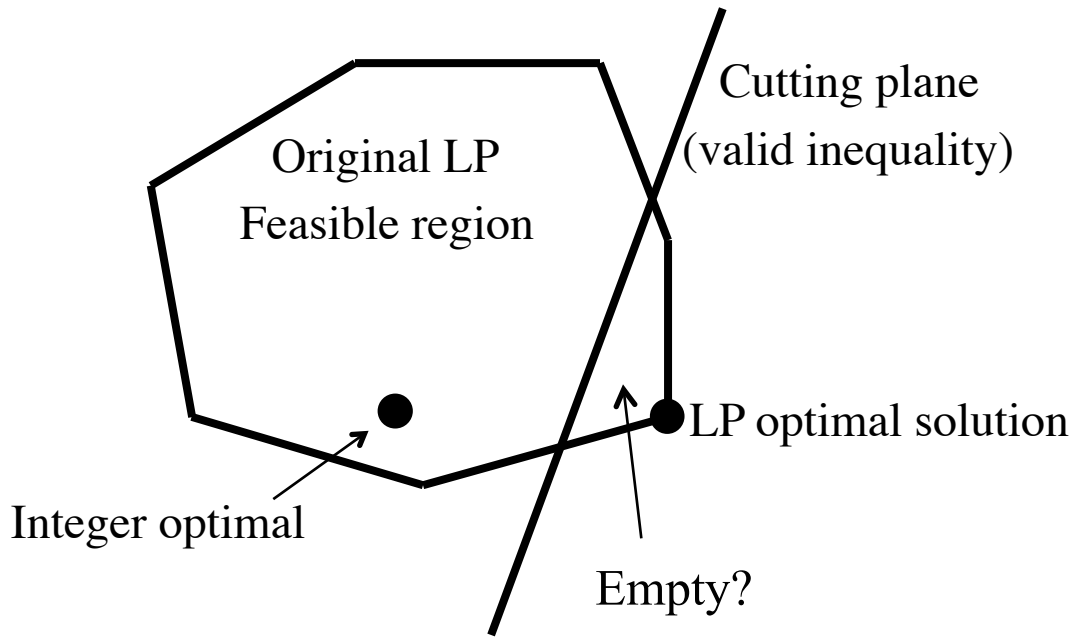
- Can strengthen valid ineq by rounding (assuming x_i integer):

$$\sum a_i x_i \geq f \Rightarrow \sum \lceil a_i \rceil x_i \geq f \Rightarrow \sum \lceil a_i \rceil x_i \geq \lceil f \rceil$$

e.g. $0.9x_1 + 0.8x_2 \geq 1.1 \Rightarrow x_1 + x_2 \geq 2$, which cuts off $(0.6, 0.7)$

Sandia National Laboratories

Proofs for arbitrary cuts



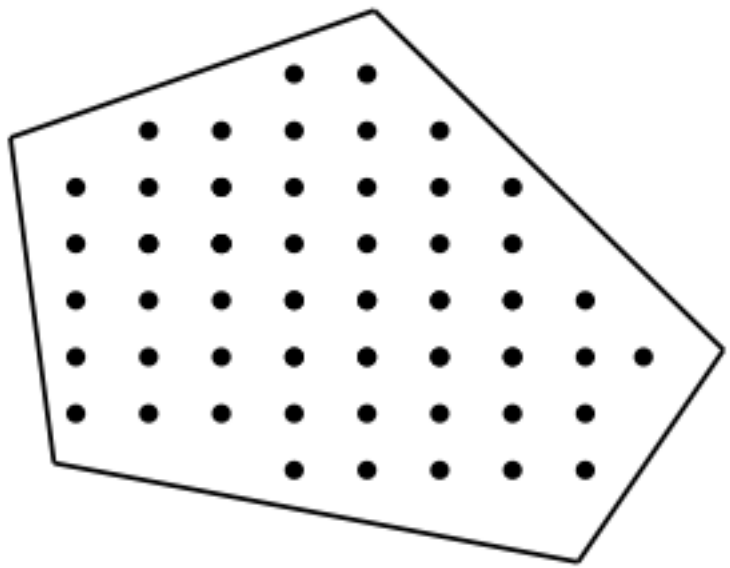
- Prove region "cut off" (complemented cut) has no better integer points
- Can optimize with cut as the objective
- Generally recursive (but should be simpler)
- Can use previously proven cuts for this node if they help

Integer Infeasibility Computations

- Useful in 3 possible ways
  - Cut correctness
  - Branching correctness
  - The full proof (objective cut from opt on original problem)
- Are these easier to solve for cut complements?
  - Benefit if LP optimal has small violation?
  - Benefit from “knowing” region is empty?
- Note: Can remove redundant constraints
- For our research, creating EmptyLib, a library of infeasible IP problems generated by certificate computations

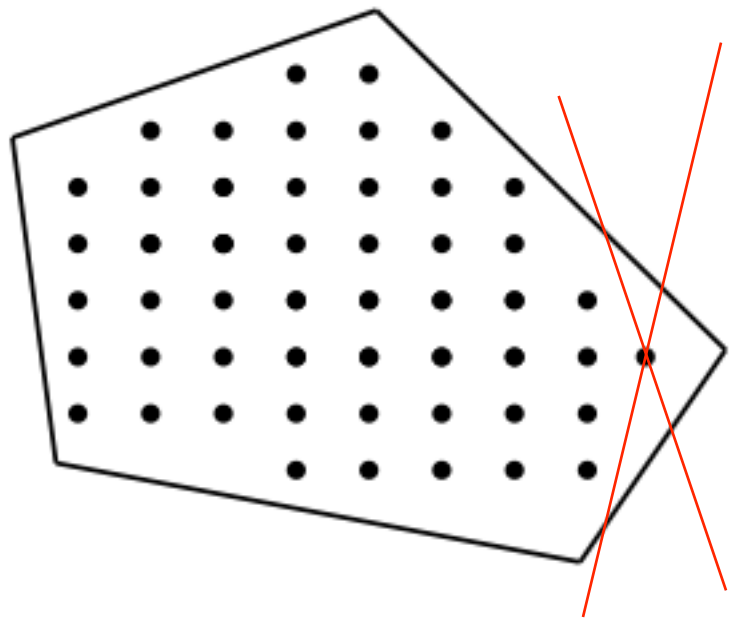
Gomory Cuts at Integer Optimal

- In principle at most n cuts at the integer optimal to make LP integer feasible



Gomory Cuts at Integer Optimal

- In principle at most n cuts at the integer optimal to make LP integer feasible
- "Just" prove these cuts are correct
  - Hermite Normal Form may be relevant

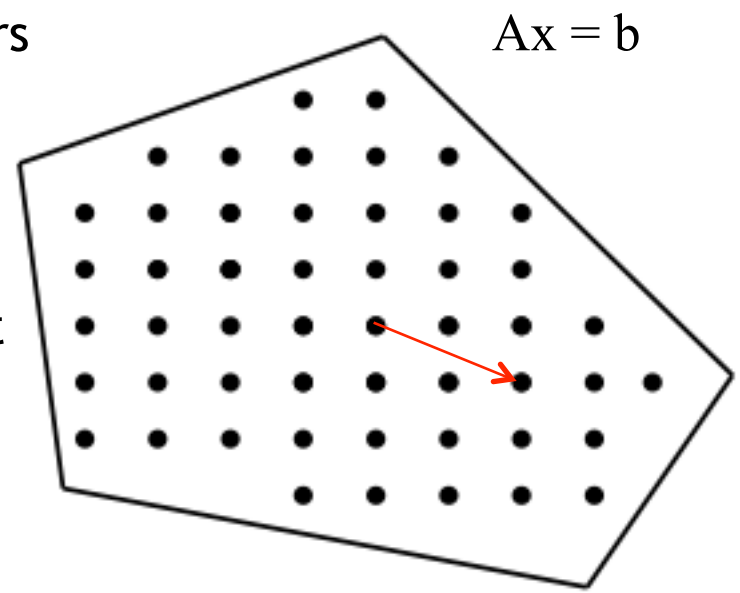


Numerical Issues

- Inexact arithmetic requires tolerances: constraint feasibility, integrality, gap, etc
  - Every IP or LP solver defines differently
- Exact arithmetic (e.g. Applegate, Cook, Dash, Espinoza)
  - Limited only by memory
    - Can be huge (Kramer's rule, value of determinants)
    - Exact solvers start with approximation of floating point
- Can do some things to help in practice during computations (fathoming)
  - Fix integer values and resolve LP with tighter tolerances
  - If same basis, tighten more (to $(|\text{basis determinant}|)^{-1}$)

Groebner Basis

- Given an integer feasible solution, improving step
  - In nullspace of contraint matrix A
  - Takes integer point to integer point
- Groebner basis: basis for all such vectors
  - Can be exponentially large
  - Considered algorithmically
  - We consider as test set
    - Requires provably complete set



Hilbert's Nullstellensatz

- Certifies infeasibility of a system of polynomial equations

$$f_i(x) = 0 \quad \forall i = 1,\ldots,m$$

$$\text{is infeasible iff } \exists \beta_i \text{ s.t.}$$

$$1 = \sum \beta_i f_i$$

- (0-1) IP feasibility can be formulated as polynomial system:

$$Ax - b = 0$$

$$x_i(x_i - 1) = 0 \quad \forall i = 1,\ldots,n$$

- At most m polys in certificate, but polys may require large encoding size (e.g. large degree and dense)

Sandia National Laboratories

Nullstellensatz Certificates

- Goal: find small certificate of infeasibility if it exists
- Can find certificate of fixed small degree if it exists
  - Exhaustive search by incrementing degree
    [2010 ICS Prize: De Loera et al., 2008 & 2009]
- Certificate (size) depends on formulation
  - Adding redundant polynomials may help [De Loera et al.]
- More inspired formulations for specific combinatorial problems, e.g. using roots of unity as discrete choices
  - Interesting nontrivial formulations arising from our context?
  - Automatic generation of redundant equations
    (analogue of cut generation)

Sandia National Laboratories