Strengthening model formulations a priori

Presolving

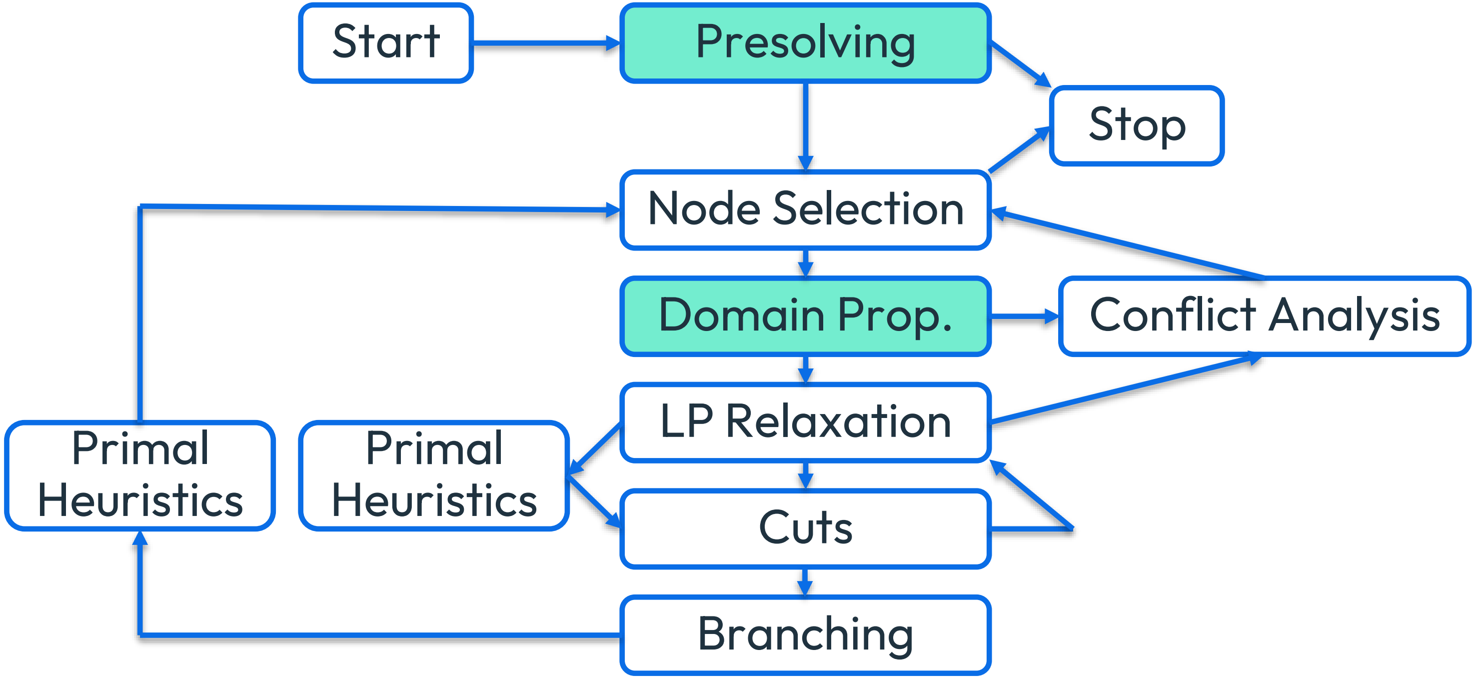
PD Dr. Timo Berthold

Private Lecturer @ TU Berlin
Director, Mixed-Integer Optimization @ FICO

Motivation

- Original MIP formulation can almost always be improved
  - Smaller difference between space of feasible continuous and feasible integer solutions
  - aka “tight relaxation”
- Two techniques:
  - Presolving: Logic reductions of the model before the main search starts
  - Cutting planes: Generating additional constraints that tighten the formulation
- Three principles occur at many places in cutting and presolving:
  - Rounding: Integer multiples of integer variables take integer values
  - Lifting: Fixing a variable at a bound can make constraints infeasible or redundant
  - Disjunction: Binary variable must take one of two values

MIP Solver Flowchart



Presolving

LP Presolving

Goals:

- Reduce problem size
  - Speed up linear algebra during the solution process
  - And virtually every loop over rows and columns
- Improve the numerics of the problem (e.g., by scaling)
- Keep the ability to postsolve primal & dual solutions and optimal basis
  - Preserve duality

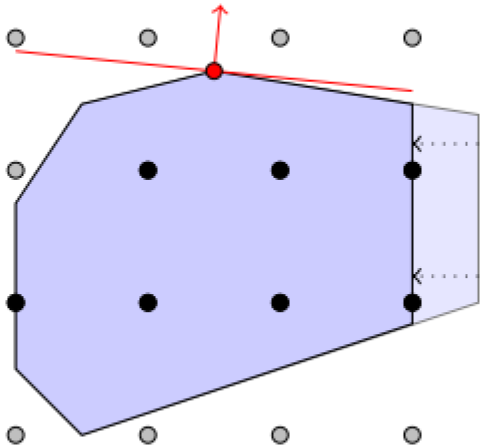
- Primal Reductions:
  - based on feasibility reasoning
  - no feasible solution is cut off

- Dual Reductions:
  - consider objective function
  - at least one optimal solution remains

MIP Presolving

Additional techniques:

- Exploit integrality
  - To strengthen the LP relaxation
  - To reduce search space size
- Identify problem sub-structures
  - Cliques, implied bounds, networks, connected components, …



- No need to preserve duality
  - We only need to be able to postsolve primal solutions

- Same consideration for primal and dual reductions

Trivial Stuff

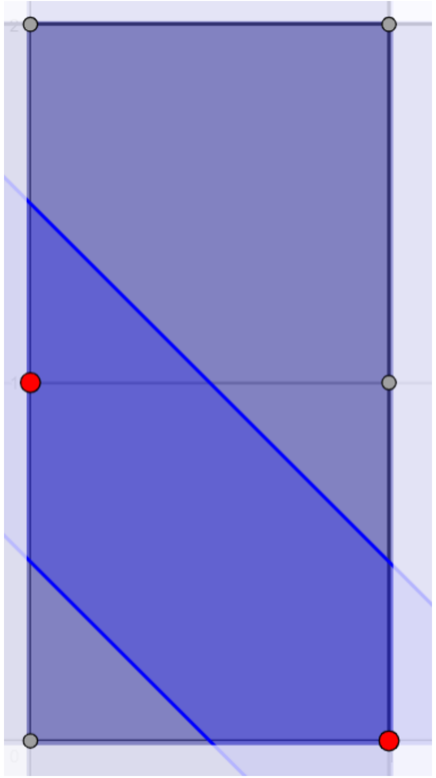
- Remove empty rows, columns
  - E.g., $0^T x \leq b_i < 0 \Rightarrow$ infeasible
- Tighten fractional bounds of integer variables
- Substitute fixed variables $x_j = c$ and aggregated variables $x_j = ax_k + c$
- Boundshifting of general integers: Replace $x_i \in \{N, N+1\}$ by binary variable
- Replace singleton rows
  - E.g., $ax_j \leq b_i, \mathrm{a} < 0 \Rightarrow x_j \geq \frac{b_i}{a} \Rightarrow$ new lower bound on x_j
- Normalize constraints
  - E.g., if all coefficients are integral, divide by greatest common divisor
- Classify constraints

Important!

- Problem instances are often automatically generated and contain many artifacts
- Often, the first modeling attempt is trivially infeasible or unbounded
  - Want to recognize this quickly
- Software that cannot recognize trivial things does not look trustworthy
- Trivial reductions can result from prior, non-trivial reductions

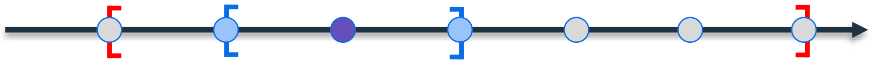
Our Working Example for the Next Four Slides

$$1 \leq 2x_1 + 2x_2 \leq 3$$
$$x_1 \in \{0,1\}$$
$$x_2 \in \mathbb{Z}_{\geq 0}, \mathrm{x}_2 \leq 2$$



Linear Presolving

- Important concept: minimal and maximal activities (Brearly et al 1975)
- Let a linear constraint $a^T x \le b$ and bounds $l \le x \le u$ be given
  - $\alpha_{min} := \min\{a^T x : l \le x \le u\} = \sum_{j,a_j>0} a_j l_j + \sum_{j,a_j<0} a_j u_j$ is called minimal activity
  - $\alpha_{max} := \max\{a^T x : l \le x \le u\} = \sum_{j,a_j>0} a_j u_j + \sum_{j,a_j<0} a_j l_j$ is called maximal activity
  - Example: $1 \le 2x_1 + 2x_2 \le 3, x_1 \in \{0,1\}, x_2 \in \{0,1,2\}$



- First observation:
  - $\alpha_{max} \le b \Rightarrow$ constraint is redundant
    - example: $2x_1 + 2x_2 \le 7, x_1 \in \{0,1\}, x_2 \in \{0,1,2\}$
  - $\alpha_{min} > b \Rightarrow$ problem is infeasible
    - example: $2x_1 + 2x_2 \le -1, x_1 \in \{0,1\}, x_2 \in \{0,1,2\}$

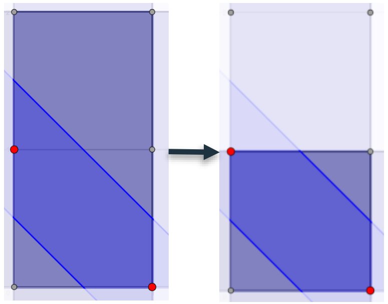
Bound Strengthening

- Important concept: minimal and maximal activities
- Let a linear constraint $a^T x \leq b$ and bounds $l \leq x \leq u$ be given
  - $\alpha_{min} \coloneqq \min\{a^T x : l \leq x \leq u\} = \sum_{j,a_j>0} a_j l_j + \sum_{j,a_j<0} a_j u_j$ is called minimal activity
  - $\alpha_{max} \coloneqq \max\{a^T x : l \leq x \leq u\} = \sum_{j,a_j>0} a_j u_j + \sum_{j,a_j<0} a_j l_j$ is called maximal activity

Example: $1 \leq 2x_1 + 2x_2 \leq 3, x_1 \in \{0,1\}, x_2 \in \{0,1,2\}$

$$x_2 \leq \left\lfloor \frac{3-0+2\cdot 0}{2} \right\rfloor = 1$$

- Second observation:
  - Let $a_i > 0$
  - $a^T x - a_i x_i + a_i x_i \leq b \;\Leftrightarrow\; x_i \leq \frac{b-(a^T x - a_i x_i)}{a_i} \quad \Rightarrow x_i \leq \frac{b-\alpha_{min}+a_i l_i}{a_i}$
  - For integer variables: $x_i \leq \left\lfloor \frac{b-\alpha_{min}+a_i l_i}{a_i} \right\rfloor$
  - Analogous for lower bound and max activity



FICO

Coefficient Tightening

- Important concept: minimal and maximal activities
- Let a linear constraint $a^T x \leq b$ and bounds $l \leq x \leq u$ be given
  - $\alpha_{min} := \min\{a^T x : l \leq x \leq u\} = \sum_{j,a_j>0} a_j l_j + \sum_{j,a_j<0} a_j u_j$ is called minimal activity
  - $\alpha_{max} := \max\{a^T x : l \leq x \leq u\} = \sum_{j,a_j>0} a_j u_j + \sum_{j,a_j<0} a_j l_j$ is called maximal activity

- Third observation:
  - Let $a_i > 0, x_i \in \{0,1\}$ and $a_{max} - a_i < b$
  - Then $a_i x_i + \sum_{j \neq i} a_j x_j \leq b$ can be reformulated as

$$(\alpha_{max} - b)x_i + \sum_{j \neq i} a_j x_j \leq (\alpha_{max} - a_i)$$

  - Proof: Check for 0 and 1. Again, other cases analogous

Coefficient Tightening

- Let $a_i > 0, x_i \in \{0,1\}$ and $\alpha_{max} - a_i < b$
- Then $a_i x_i + \sum_{j \neq i} a_j x_j \leq b$ can be reformulated as

$$(\alpha_{max} - b)x_i + \sum_{j \neq i} a_j x_j \leq (\alpha_{max} - a_i)$$

Example: $1 \leq 2x_1 + 2x_2 \leq 3, x_1, x_2 \in \{0,1\}, a_{max} = 4$

$$2x_1 + 2x_2 \leq 3$$

$$(4-3)x_1 + 2x_2 \leq (4-2)$$

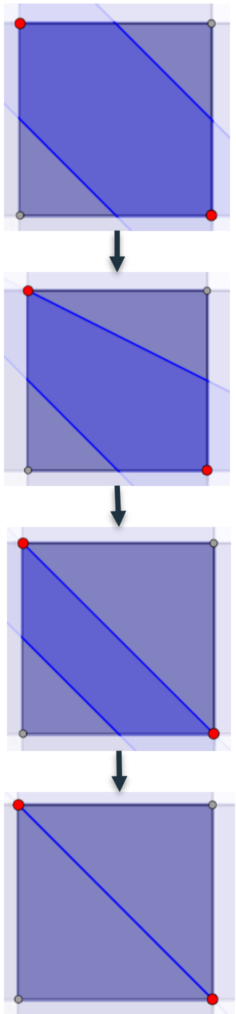
$$x_1 + 2x_2 \leq 2, a_{max} = 3$$

$$x_1 + (3-2)x_2 \leq (3-2)$$

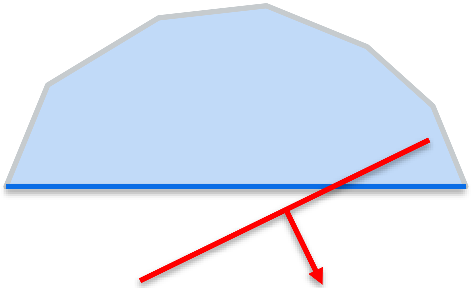
$$x_1 + x_2 \leq 1$$

Same argument for left-hand side to get $x_1 + x_2 = 1$

Hence $x_1 = 1 - x_2$. Eliminate variable and constraint



Dual Reductions



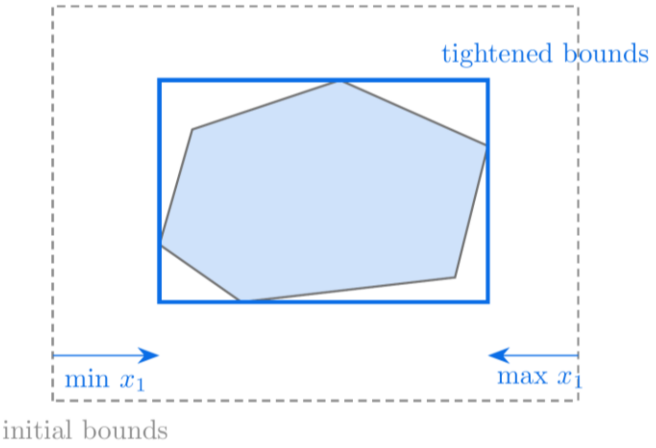
- Ensure that you keep at least one optimal solution
- Most trivial example: Fixing variables that appear in no constraint
- Dual fixing: If variable x_j appears in no equation, only with nonnegative coefficients in $\leq$-constraints, with nonpositive coefficients in $\geq$-constraints and has a nonnegative objective (minimization), then x_j can be fixed to its lower bound (recall: down-locks)
- Dual aggregation: Assume there is exactly one constraint violating the above, say x_j appears only in $\leq$-constraints, $c_i > 0$ and a_{ij} is its only negative coefficient.
  We can use $x_j = \frac{b_i}{a_{ij}} - \frac{1}{a_{ij}} \sum a_k x_k$. Note: x_j continuous or aggregation must imply integrality
- Dual bound reduction: Strengthen bounds of variables to the tightest value for which all its constraints are redundant

Probing (Savelsbergh 1994)

- „Strong branching without LP“, only applying bound strengthening
- Usually only for binary variables, various working limits apply
  - Sequence-dependent
- If $x_1 = 0 \Rightarrow$ infeasible and $x_1 = 1 \Rightarrow$ infeasible, then the problem is infeasible
- If $x_1 = 0 \Rightarrow$ infeasible, then fix $x_1 = 1$
- If $x_1 = 0 \Rightarrow x_2 = a$ and $x_1 = 1 \Rightarrow x_2 = b > a$, aggregate $x_2 = a + (b - a)x_1$
- If $x_1 = 0 \Rightarrow x_2 \leq a$ and $x_1 = 1 \Rightarrow x_2 \leq b$, then add bound $x_2 \leq \max(a, b)$
- If $x_1 = 0 \Rightarrow x_2 \leq a$, store information in implication graph, use for heuristics, lifting, …

OBBT(Optimization-Based Bound-Tightening) [Quesada&Grossmann 1993, Zamora&Grossmann 1999]

- For each variable x_j: solve two LPs over the current relaxation, $\min x_j$ and $\max x_j$
  - Tighten if the optimal value improves the bound
  - Clear relation to strong branching
    - For binaries: Same possible reduction, but less efficient algorithm (primal vs dual simplex)
- Speedup techniques [Gleixner et al. 2017]
  - Filtering: skip LPs that cannot improve a bound
  - Ordering heuristics: exploit simplex warm starts across the LP sequence
  - Lagrangian variable bounds: cheaper surrogate from the LP dual instead of a full LP solve



Multi-Row/Column Reductions

- Parallel rows/columns
  - Search for pairs of rows such that coefficient vectors are parallel to each other
    - Hashing plus sorting algorithm
  - Discard the dominated row, or merge two inequalities into an equation
- Dominated rows/columns
  - Pairwise comparison, heuristic selection of pairs
- Clique merging:
  - Merge multiple cliques into one larger clique:
  - $x_1 + x_2 \leq 1, x_2 + x_3 \leq 1, x_1 + x_3 \leq 1 \Rightarrow x_1 + x_2 + x_3 \leq 1$

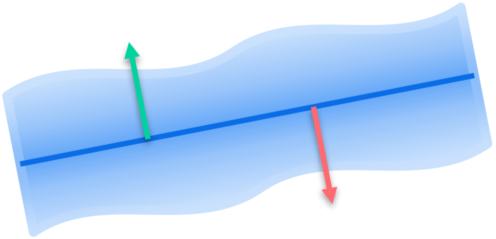
Column-Based Coefficient Eliminations aka. Sparsification

- Reduce the number of non-zero coefficients by combining rows:
  - If $\tilde{a}x = \tilde{b}$, then $\bar{a}x \leq \bar{b}$ can be replaced by $(\bar{a} + \delta\tilde{a})x \leq \bar{b} + \delta\tilde{b}$
  - Find weight δ that minimizes the number of non-zero coefficients
- We can also combine columns to reduce the number of non-zero coefficients:
  - For a **free** variables x_1 and a variable x_2, introduce a new variable $z = x_1 - \delta x_2$, or $x_1 = z + \delta x_2$
  - Eliminating x_1 replaces each sub-expression $a_{i1}x_1 + a_{i2}x_2$ of a row i by $a_{i1}z + (a_{i2} + \delta a_{i1})x_2$
  - Find weight δ that minimizes the number of non-zero coefficients
  - Can even be applied to integer variables with $\delta \in \{-1, +1\}$

$$\overset{+\delta}{\curvearrowright}$$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots \\ \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots \end{bmatrix} \Rightarrow \begin{bmatrix} a_{11} & a_{12} + \delta a_{11} & \cdots \\ \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} + \delta a_{m1} & \cdots \end{bmatrix}$$

Presolve Fixing of Penalty Variables



- Structure: $\sum_{1}^{n} a_k x_k + y_1 - y_2 = r$ $\quad y_1, y_2 \geq 0$

  y_1, y_2 have non-zero objective

- Interpretation: y_1, y_2 penalize violation of $\sum_{1}^{n} a_k x = r$

  y_1: violation in negative direction

  y_2: violation in positive direction

- Reduction (depending on actual objective coefficient values)
  - $\sum_{1}^{n} a_k x_k \geq r$ for all x_k implies $y_1 = 0$
  - $\sum_{1}^{n} a_k x_k \leq r$ for all x_k implies $y_2 = 0$
  - otherwise tighten upper bounds on y_1, y_2

Performance impact:

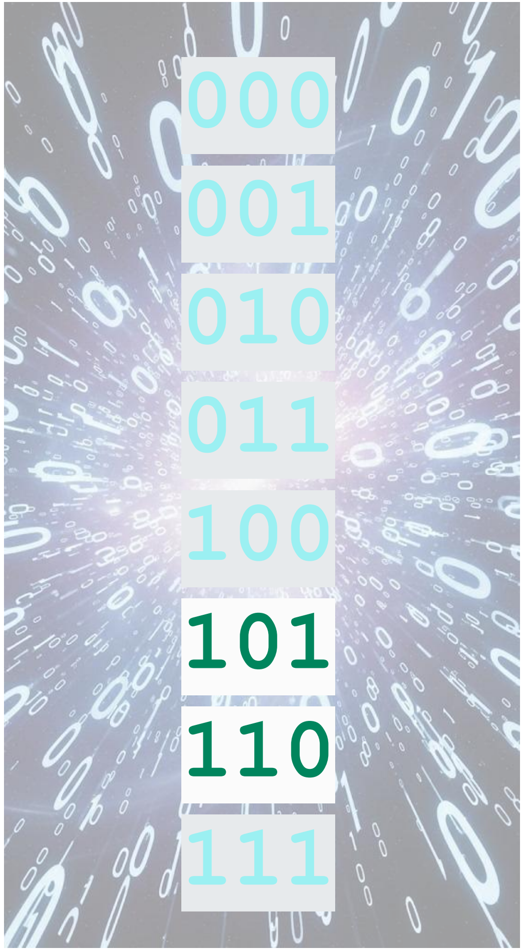
- Overall: 1%
- >100s: 2%
- affected: 10%

Many More... (e.g., Achterberg et al. 2020)

- Implied integer detection
  - $\sum a_j x_j + y = b$, x_j integer variables $a_j \in \mathbb{Z}\ \forall j$ and $b \in \mathbb{Z}$, then y integer
- GCD reduction
  - Let gcd be the GCD of all coefficients a_j in an all-integer row
  - $\sum \frac{a_j}{gcd} x_j \leq \left\lfloor \frac{b}{gcd} \right\rfloor$
- Disconnected component detection
  - DFS on matrix A
  - If there is an independent component $\tilde{A}\tilde{x} \leq \tilde{b}$ that is not connected to the rest of $Ax \leq b$, solve it as auxiliary MIP (if it is small enough)
- Analytic center presolving (Berthold et al 2017)
  - Fix variables that are at one of their bounds in the analytic center

Solution Enumeration

- Enumerate all the solutions of short binary constraints
  - E.g., $x_1 + x_2 + x_3 \geq 2$
    - (allows 011, 101, 110 , 111 and eliminates 000,001,010,100)
- Check them to see if they are feasible for the other constraints
  - E.g., $x_2 + x_3 + x_{42} = 1$ eliminates 111 and 011
- Fix common variable values, find linear relations, perform eliminations
  - E.g., we can now deduce $x_1 = 1$
  - And $x_2 = 1 - x_3$
  - Hence $x_{42} = 0$
- Strengthen the constraint bound to touch the solutions
- Lift coefficients so that the constraint is tight for more solutions
- In Xpress: applied to rows of length up to 12
  - Still very fast thanks to an optimized implementation and shortcuts



Configuration Presolving

- Consider a binary row with only few different coefficients:

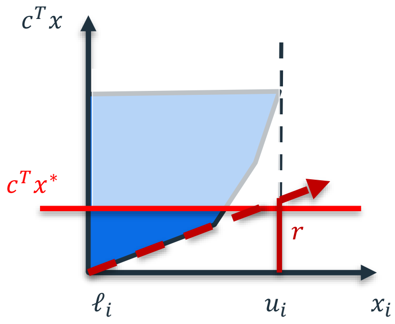
$$7\,x_1 + 7\,x_2 + 7\,x_3 + 7\,x_4 + 13\,x_5 + 13\,x_6 + 13\,x_7 \leq 21, \quad x_1, \dots, x_7 \in \{0,1\}$$

- Every feasible solution can have
  1. At most three variables out of x_1, x_2, x_3, x_4 set to 1 **OR**
  2. One of x_5, x_6, x_7 set to 1 and at most one of the others
- We can introduce new 0-1 variables z_1, z_2 that describe the two *configurations* above:

$$z_1 + z_2 = 1$$
$$x_1 + x_2 + x_3 + x_4 \leq 3\,z_1 + 1\,z_2$$
$$x_5 + x_6 + x_7 \leq 0\,z_1 + 1\,z_2$$

- This reformulation renders the original row redundant
- Can be considered an implicit enumeration of the feasible region (modulo symmetry)

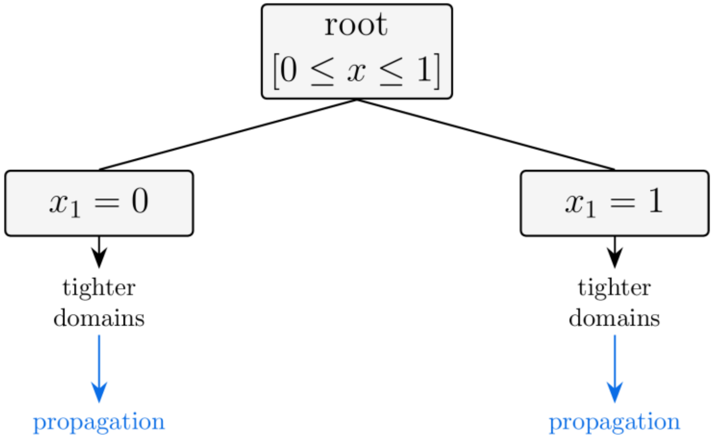
Reduced Cost Fixing



- Not yet a cut, not presolving anymore
- Reduced costs: $\mathrm{r} \coloneqq c - A^T y$ for optimal dual solution y
  - Underestimator of the amount by which LP value would change, if we shifted solution towards other bound
  - Zero for basic variables, nonnegative for nonbasic variables at lower bound, nonpositive for nonbasic variables at upper bound
- Reduced costs can be used to tighten variable bounds of nonbasic variables
  - For binary x_i (at 0 in LP optimum) and LB the dual bound, UB the primal bound of an optimization problem, fix $x_i = 0$, if $r_i \geq UB - LB$.
- Apply locally with current LP solution
- Globally, store best reduced cost per variable from any global LP optimum (cut loop!)
  - Reconsider every time when UB changes

Node Presolving / Domain Propagation

Presolving Does Not Stop at the Root



Root Presolving

- Works with global domains
- Structural transformations: aggregation, sparsification, probing, coefficient tightening, …
- Performed relatively aggressively

Node Presolving / Propagation

- Works with local domains
- Branching decisions create additional information
    - Repeatedly tighten bounds and detects infeasibility
- Must be very cheap
- Reductions are undone when backtracking

- Bound changes, constraint (de)activation
    - but no matrix coefficient change, reformulation, …

A constraint that does not propagate at the root may become powerful after branching

Domain Propagation

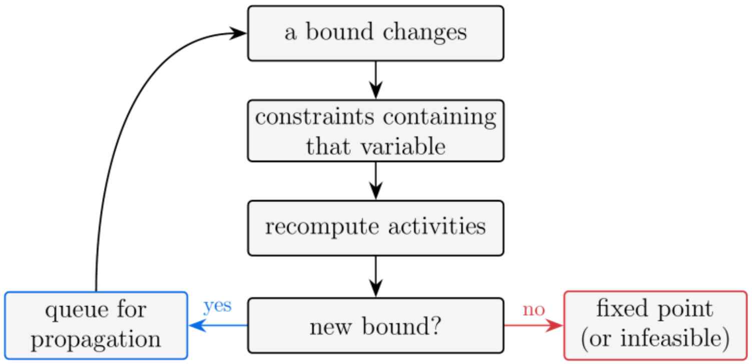
Bound Tightening Reloaded

$$a^T x \leq b, \qquad l^N \leq x \leq u^N$$

$$u_i^N \leftarrow \min\left\{ u_i^N, \left\lfloor \frac{b - \alpha_{\min}^N + a_i l_i^N}{a_i} \right\rfloor \right\} \quad (a_i > 0)$$

During presolving: l, u are global bounds.

At a node: l^N, u^N contain all branching decisions and local deductions.



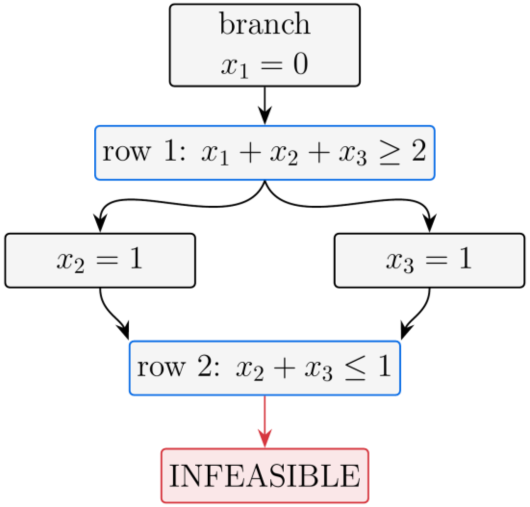
Not only linear rows:

cliques / set-packing constraints, implications / variable bounds, indicators, SOS, knapsacks, conflict constraints, specialized constraint handlers

Propagation Can Solve a Node Without Solving an LP

$x_1 + x_2 + x_3 \geq 2, \;\; x_2 + x_3 \leq 1$
$x_1, x_2, x_3 \in \{0,1\}$

- For the global domain, bound tightening does nothing
  - forget for the moment that there are other presolving routines ;-)

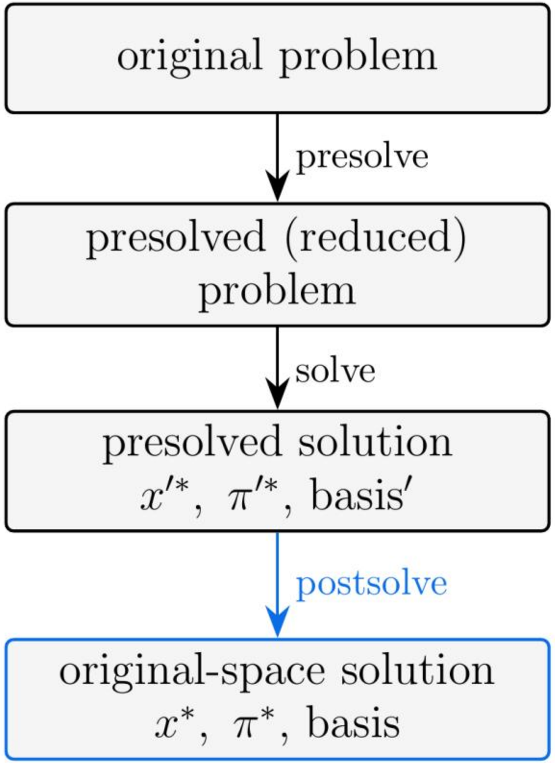


Remember probing?

Probing = temporarily create such a node during presolving and run propagation.

- **Domain propagation:** we are actually in the node x_1=0, propagate, and cut it off if infeasible
- **Probing:** we only pretend to be in that node at the root, propagate, see infeasibility, and therefore fix x_1=1 globally

Postsolving



Postsolving reconstructs:

- Primal solution
- Dual values & reduced costs (LP only)
  - needed for sensitivity analysis, cut generation, …
  - Correct duals require dual-safe presolving
- Optimal basis (LP only): needed for warm-starts

Mechanism:

- Undo fixings, aggregations, substitutions, scaling
- Reductions are stored and undone in reverse order
- Just need to reconstruct ONE solution, not a particular
  - `postsolve(presolve(`x^*`))` is not necessarily x^*

FICO

Quiz Time

- Presolving
  a) Must not cut off any feasible solution
  b) May cut off feasible, but must not cut off optimal solutions
  c) May cut off optimal solutions
- The maximum activity of $x_1 - x_2 + 2x_3 \leq 5, x_1, x_2, x_3 \in \{1,2\}$ is
  a) 2
  b) 5
  c) 6
- Which of the following is not a goal of presolving?
  a) Shrink the problem size
  b) Find an initial solution
  c) Strengthen the LP relaxation

Quiz Time

- Presolving
  a) Must not cut off any feasible solution
  b) May cut off feasible, but must not cut off optimal solutions
  c) May cut off optimal solutions
- The maximum activity of $x_1 - x_2 + 2x_3 \leq 5, x_1, x_2, x_3 \in \{1,2\}$ is
  a) 2
  b) 5
  c) 6
- Which of the following is not a goal of presolving?
  a) Shrink the problem size
  b) Find an initial solution
  c) Strengthen the LP relaxation

Quiz Time

- Presolving
  a) Must not cut off any feasible solution
  b) May cut off feasible, but must not cut off optimal solutions
  c) **May cut off optimal solutions**
- The maximum activity of $x_1 - x_2 + 2x_3 \leq 5, x_1, x_2, x_3 \in \{1,2\}$ is
  a) 2
  b) **5**
  c) 6
- Which of the following is not a goal of presolving?
  a) Shrink the problem size
  b) Find an initial solution
  c) Strengthen the LP relaxation

Quiz Time

- Presolving
  a) Must not cut off any feasible solution
  b) May cut off feasible, but must not cut off optimal solutions
  c) May cut off optimal solutions
- The maximum activity of $x_1 - x_2 + 2x_3 \leq 5, x_1, x_2, x_3 \in \{1,2\}$ is
  a) 2
  b) 5
  c) 6
- Which of the following is not a goal of presolving?
  a) Shrink the problem size
  b) Find an initial solution
  c) Strengthen the LP relaxation

Thank You!

PD Dr. Timo Berthold
Private Lecturer @ TU Berlin
Director, Mixed-Integer Optimization @ FICO

Bound Strengthening: Little Exercise

- Consider the linear inequality $3x_1 + 8x_2 + 5x_3 \leq 12$
  with integer variables $x_1 \geq 1, x_2 \geq 0, x_3 \geq -1, x \in \mathbb{Z}^3$
  - Compute the minimum activity of the constraint
  - Use bound strengthening to compute upper bounds on all variables

Bound Strengthening: Little Exercise

- Consider the linear inequality $3x_1 + 8x_2 + 5x_3 \leq 12$
  with integer variables $x_1 \geq 1, x_2 \geq 0, x_3 \geq -1, x \in \mathbb{Z}^3$
  - Compute the minimum activity of the constraint
  - Use bound strengthening to compute upper bounds on all variables

  - The minimum activity is $\alpha_{min} = 3 \cdot 1 + 0 + 5 \cdot (-1) = -2$
  - $x_1 \leq \left\lfloor \frac{12-(-2)+3}{3} \right\rfloor = 5, \quad x_2 \leq \left\lfloor \frac{12-(-2)+0}{8} \right\rfloor = 1, \quad x_3 \leq \left\lfloor \frac{12-(-2)-5}{5} \right\rfloor = 1$

Bound Strengthening: Little Exercise

- Consider the linear inequality $3x_1 + 8x_2 + 5x_3 \leq 12$
  with integer variables $x_1 \geq 1, x_2 \geq 0,\ x_3 \geq -1, x \in \mathbb{Z}^3$
  - Compute the minimum activity of the constraint
  - Use bound strengthening to compute upper bounds on all variables

  - The minimum activity is $\alpha_{min} = 3 \cdot 1 + 0 + 5 \cdot (-1) = -2$
  - $x_1 \leq \left\lfloor \frac{12-(-2)+3}{3} \right\rfloor = 5, \quad x_2 \leq \left\lfloor \frac{12-(-2)+0}{8} \right\rfloor = 1, \quad x_3 \leq \left\lfloor \frac{12-(-2)-5}{5} \right\rfloor = 1$