New heuristic techniques for general mixed-integer programs

Robert Carr, Sandia National Laboratories
Jonathan Eckstein, Rutgers
Cynthia Phillips, Sandia National Laboratories
Jean-Paul Watson, Sandia National Laboratories

Sandia is a multiprogram laboratory operated by Sandia Corporation, a 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:

$$Ax = b$$

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

$$x = (x_I, x_C)$$

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

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

- Can also have inequalities in either direction (slack variables):

$$a_i^T x \le b_i \Rightarrow a_i^T x + s_i = b_i \ , \ s_i \ge 0$$

- IP (easily) expresses any NP-complete problem

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

DOE/Science MIP Applications (Small Sample)

Defense program applications:

- Logistics
  - Capacity planning, scheduling, workforce planning, constrained vehicle routing, fleet planning
- Site security
- Tools for high-performance computing (scheduling, node allocation, domain decomposition, meshing)

Science

- Bioinformatics: protein structure prediction/comparison
- Wireless sensor management
- New applications (with Ali Pinar, LBNL)
  - Scheduling telescope time (eg. For supernovae observations)
  - Groundwater monitoring
  - Analysis of particle behaviors in supercolliders

Simple Example: Scheduling telescope

- A number of projects are sharing a telescope
  - Looking for different types of objects
  - Sky regions observed multiple times
  - Quality of a pair of observations depends on time gap

- x_{ij} = 1 if observe region *i* on night j
- Z_{ijkp} = 1 if project p uses an observation of region *i* on nights *j* and *k*

- v_{gp} = value to project p for observing with a gap of g
- n = # of observations/night
- V_p = minimum value for project p

Sandia National Laboratories

Simple example: Scheduling telescope

$$\max \sum_{ijkp} v_{k-j,p} z_{ijkp}$$

$$st$$

$$\sum_{ij} x_{ij} \leq n \qquad \text{maximum observations/night}$$

$$z_{ikjp} \leq x_{ik} \forall i,k,j,p$$

$$z_{ikjp} \leq x_{jk} \forall i,k,j,p$$

$$\sum_{ijk} v_{k-j,p} z_{ijkp} \geq V_p \forall p \ \ \text{miminum quality}$$

$$\sum_{(\mathrm{i,j,k}) \in \mathrm{F}} \mathrm{z}_{\mathrm{ijkp}} \leq 1 \ \forall p, \text{ overlap sets } F \text{ (no overlaping intervals)}$$

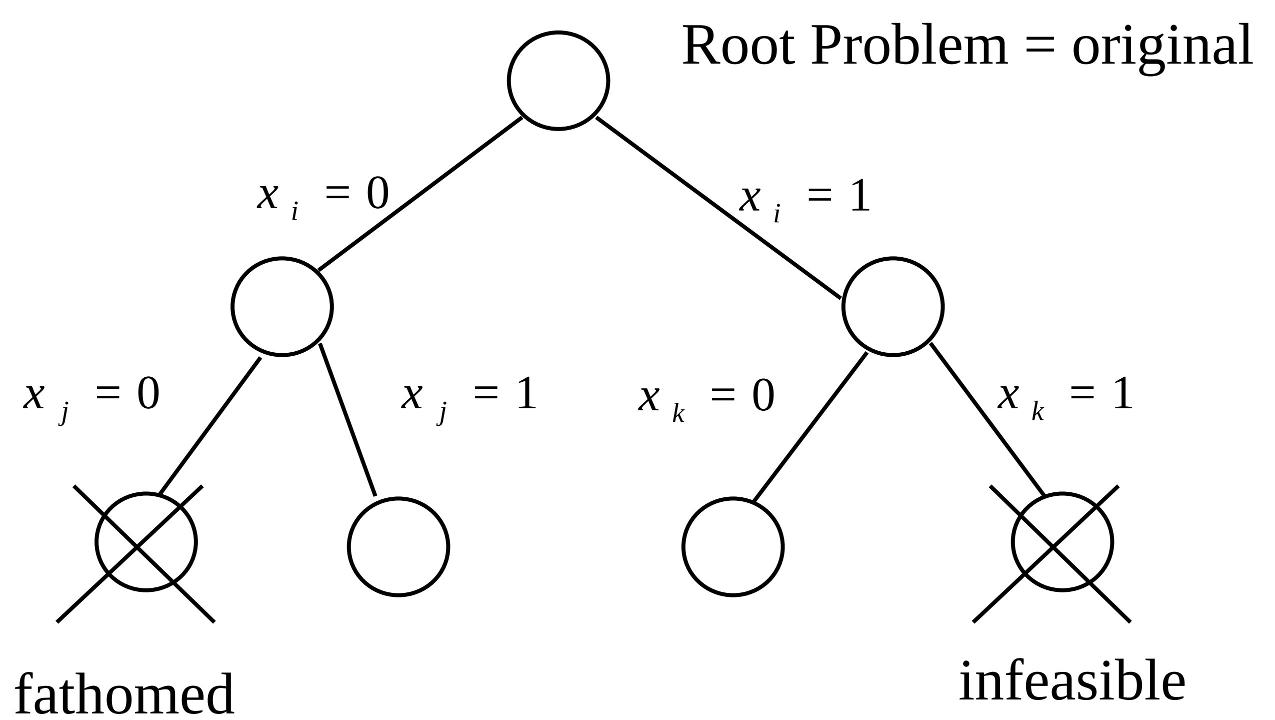
Solution Options for Integer Programming

- Commercial codes (ILOG's cplex)
  - Good and getting better
  - Expensive
  - Serial (or modest SMP)
- Free serial codes (ABACUS, MINTO, BCP)
- Modest-level parallel codes (Symphony)
- Grid parallelism (FATCOP)
- In development: ALPS/BiCePs/BLIS

- Massive parallelism: PICO (Parallel Integer and Combinatorial Optimizer)

Note: Parallel B&B for simple bounding: PUBB, BoB/BOB++, PPBB-lib, Mallba, Zram

Solving Integer Programs: Branch and Bound

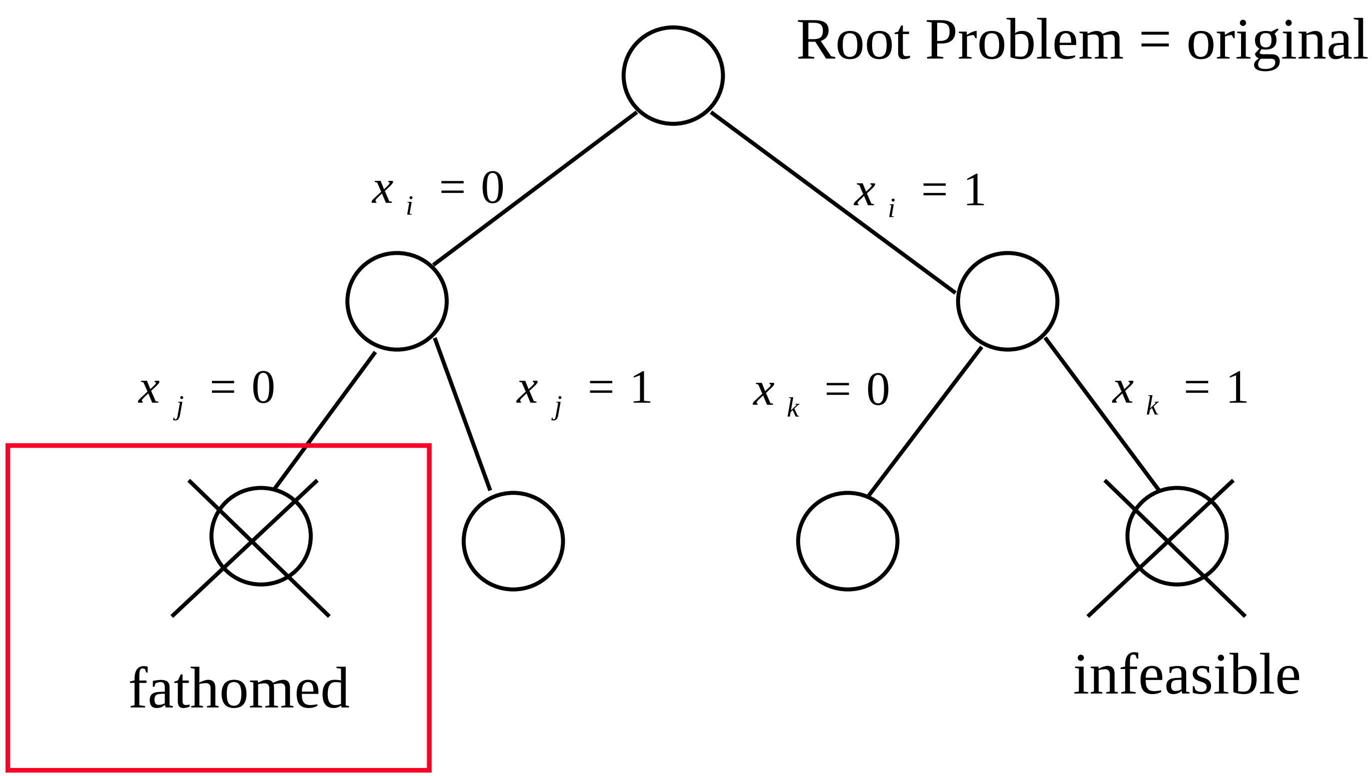


- Lower bound: LP relaxation

PICO Parallel IP Solver: Two Phases

- Parallel subproblem phase
  - There are plenty of subproblems compared to # processors
- Ramp up (eg. 1 subproblem, 10000 processors)
  - Parallel processing of single problem
    - Gradients
    - Cuts
    - LP bounds
  - Incumbent heuristics (looking for a good feasible solution)

Value of a Good Feasible Solution Found Early



- Faster pruning
- Having something to say if the computation stops early

General-Purpose Incumbent Heurstics

- Randomized rounding
- Feasibility pump
- Nediak-Eckstein
- Fractional Decomposition Tree

Randomized Rounding

Binary decision variables. LP relaxation x*

- Simplest form: treat LP relaxation $0 \leq x^* \leq 1$ probability
- Select each x* independently with probability x*
- For parallel IP, in early computation, many processors can do this independently.
- Resulting vector x is
  - Integer by construction
  - Almost certainly infeasible for linear constraints $Ax = b$.
    - Exception: covering problems [Raghavan, Thompson]
- Fast way to find something when (almost) everything is feasible

Sandia National Laboratories

Feasibility Pump (Fischetti, Glover, Lodi)

Basic algorithm:

1. Solve LP to obtain x*
2. Round (arithmetically) x* to $\tilde{x}$
3. While $\tilde{x}$ is not feasible obtain new x* from this LP:

$$\min \sum_i y_i$$
$$s.t$$
$$y_i \geq x_i - \tilde{x}_i$$
$$y_i \geq \tilde{x}_i - x_i$$
$$Ax = b$$

   and round x to $\tilde{x}$ again

Feasibility Pump Improvements

- Gap is $\dfrac{\text{value of first feasible solution}}{\text{optimal (or best known)}}$

- Improvement is relative to initial feasibility pump
- Tested with problems from miplib2003

- Round x* multiple times, take best (most feasible) of k trials
  - 23.7% gap improvement for k = 30
  - Running time increase factor of k, but fully parallelizable

- Iterated local search: perturb and redo the feasibility pump
  - For k=30 iterations, gap improvement of 31.2%
  - Runtime increase of k, not parallelizable individually, but can do multiple independent iterated searches.
- Good idea to round randomly for x* components near .5

Eckstein-Nediak Heuristic

- Parallelizable, General 0-1 MIPs

Uses a **merit function** $\psi(x)$

- motivated by Løkkentangen and Glover, 1998
- $\psi(x) = 0$ if vector x is integer feasible
- $\psi(x) > 0$ if an integer variable is fractional
- $\psi(x)$ is differentiable and strictly concave
  - Important properties, not enforced by Løkkentangen and Glover

Goals:

- Reduce $\psi(x)$ to 0
- Obey linear constraints ($Ax \leq b$) and variable bounds
- Minimize increase in MIP objective ($c^T x$)

Parallel MIP Heuristic Merit Function

We define a separate merit function $\phi_j(x_j)$ for each binary variable x_j

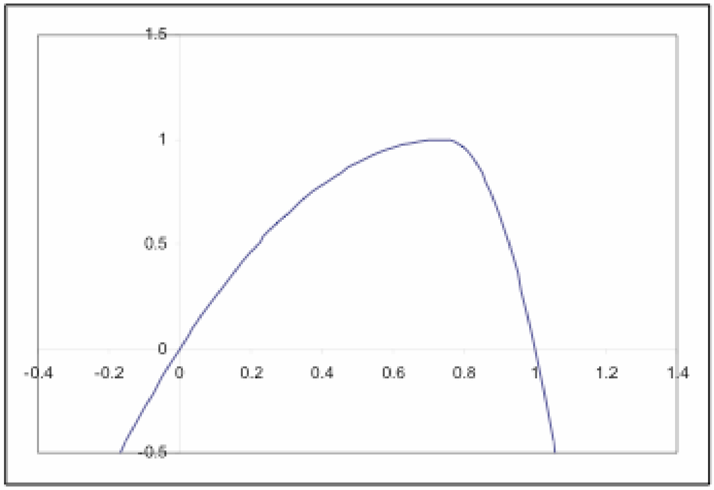
Same properties:

- $\phi_j(0) = \phi_j(1) = 0$
- $\phi_j(x) > 0$ for $0 < x < 1$
- Differentiable, strictly convex

Total merit is the sum of the individual merits (retains properties)

$$\psi(x) = \sum_{j \in I} \phi_j(x_j)$$

Merit Function for a variable x_j



$\mathcal{C}^1$ quadratic spline defined by

- $\phi(0) = 0$
- $\phi(\alpha) = 1$
- $\phi'(\alpha) = 0$
- $\phi(1) = 0$

$\alpha = 0.75$ shown

Specifically, for

$$\alpha \in (0,1)$$

$$\phi_\alpha(x) = 1 - \begin{cases} \left(\dfrac{x-\alpha}{\alpha}\right)^2 & \text{for } x \le \alpha \\ \left(\dfrac{x-\alpha}{1-\alpha}\right)^2 & \text{for } x > \alpha \end{cases}$$

Nediak-Eckstein MIP heuristic

New objective function $\nabla\psi(x^*) + wc$, where

- x^* is the current point (such as LP optimal)
- c is the original IP objective function
- w is a weighting factor (IP objective vs. integrality)

This is the Sum/Frank-Wolfe approach

Use normal LP simplex pivots to improve the new objective

- Adjust the objective at each step (for new x^*)
- Provably finds a local optimum (via concavity)

If the local optimum x has $\psi(x) > 0$, can add Gomory cuts and continue.

Sandia National Laboratories

Nediak-Eckstein MIP Heuristic

- Processors can use different merit functions
  - Random values of α for each variable
- Processors can also fix one fractional variable
  - For example, if binary variable x_j is .4. Set to 0 or 1 in heuristic.
- Combinations of the two types of variation
  - Fixing variables that have a good history of improving integrality

LP-Relaxation-Based Approximation for IP

- Compute LP relaxation (lower bound).
- Common technique:
  - Use structural information from LP solution to find feasible IP solution (use parallelism if possible)
  - Bound quality using LP bound
- Integrality gap = $\max_I(\text{IP}(I))/(\text{LP}(I))$
  - Taken over all instances I (settings of class parameters: c,b)
  - Integrality gap is unbounded (infinite) if
    - LP(I) = 0 or
    - IP is infeasible when LP isn't
- This technique cannot prove anything better than integrality gap

Sandia National Laboratories

Finding an Approximate solution: Convex Decomposition

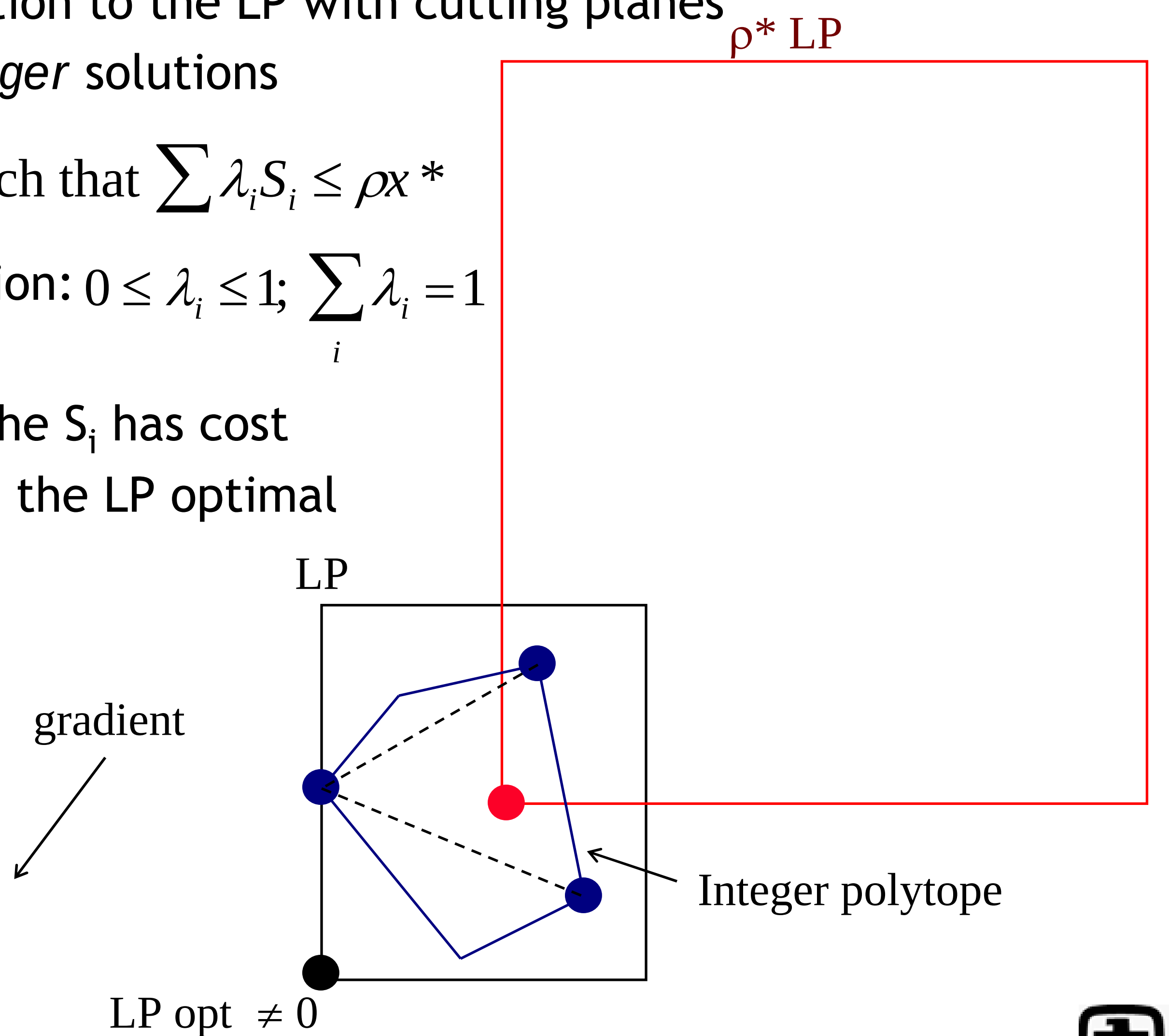
x* = feasible solution to the LP with cutting planes

Find feasible *integer* solutions

$$S_0, S_1, \cdots, S_m \text{ such that } \sum \lambda_i S_i \leq \rho x^*$$

Convex combination: $0 \leq \lambda_i \leq 1; \sum_i \lambda_i = 1$

- Implies one of the S_i has cost at most ρ times the LP optimal



Key Theorem (Carr, Vempala)

- Recall integrality gap = $\dfrac{\text{value of best integer solution}}{\text{value of LP relaxation}}$

- Let x^* be the optimal LP solution to the LP relaxation for an IP. There exists a convex decomposition dominated by ρx^* if and only if the integrality gap is ρ for finite ρ.

$$S_0, S_1, \cdots, S_m \text{ such that } \sum \lambda_i S_i \leq \rho x^*$$

$$0 \leq \lambda_i \leq 1; \ \sum_i \lambda_i = 1$$

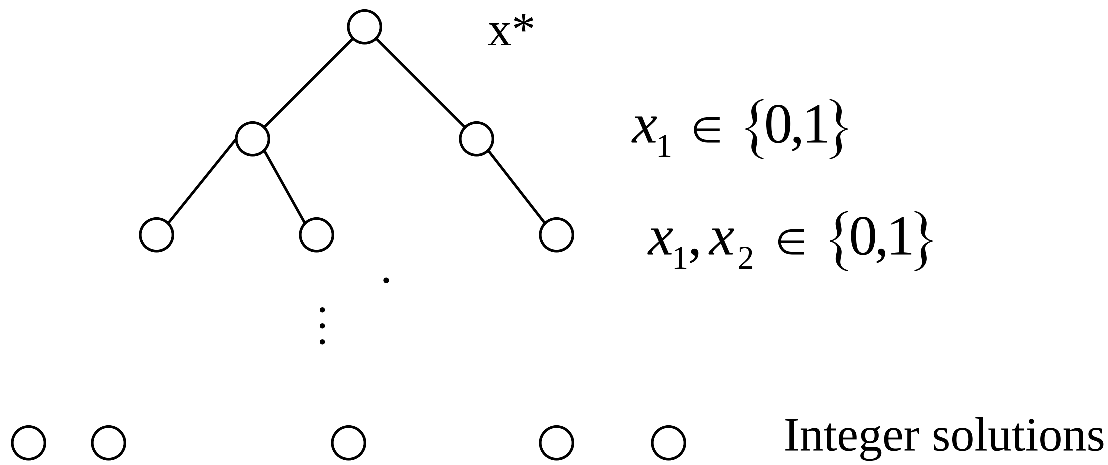
Fractional Decomposition Tree - Overview

- Previous decomposition results were problem-specific

The FDT method applies decomposition to any integer program.

- Will succeed if the problem class has finite integrality gap!
  - Success = find feasible solution
  - No quality guarantee
- Grows a tree-like branch and bound (B&B) except
  - Preserves structure of LP relaxation (vs. preserving objective function in B&B)
  - Limits the tree to polynomial size (vs. exponential for B&B)

FDTs for 0-1 IPs



- Order the variables that are fractional in the LP optimal *x**
- At each level of the tree, one more variable is forced integral
- Use LP to pack the children into the parent optimally
  - Preserve structure of the solution

LP to create the children

- To create children of the root from *x** (LPC):

$$\begin{aligned}
&\max \lambda_0 + \lambda_1 \\
&\text{st } Ay^0 \geq b\lambda_0 \\
&\quad Ay^1 \geq b\lambda_1 \\
&\quad 0 \leq y^0 \leq \lambda_0 \bullet 1 \\
&\quad 0 \leq y^1 \leq \lambda_1 \bullet 1 \\
&\quad y_1^0 = 0;\ y_1^1 = \lambda_1 \\
&\quad y^0 + y^1 \leq x^* \\
&\quad \lambda_0, \lambda_1 \geq 0
\end{aligned}$$

- Children of the root have solutions:

$$x^{0*} = \frac{y^0}{\lambda_0} \text{ and } x^{1*} = \frac{y^1}{\lambda_1}$$

- Solutions are feasible, have first variable integral, and decompose x* with value $\rho = \dfrac{1}{\lambda_0 + \lambda_1}$.

LPC is feasible in general

- For finite integrality gap, there exists

$$S_0, S_1, \cdots, S_m \text{ such that } \sum \lambda_i S_i \le \rho x^*$$

- Let $S_i^{(1)}$ be the members of S_i with x_1=1
  $S_i^{(0)}$ be the members of S_i with x_1=0

$$\sum \lambda_i S_i^{(1)} \le \rho x^*$$

$$\sum \lambda_j S_j^{(0)} \le \rho x^*$$

Pruning the tree

- Let n be the number of fractional variables in *x**
- If any level of the tree has more than n nodes, we prune the tree, keeping only the best n partially integral solutions.
- This LP (LPP) picks the n survivors that best pack into the root solution x* and calculates the convex combination parameters.

$$\max \sum_{i} \lambda^i$$

$$\text{st} \quad \sum_{i} \lambda^i x^i \le x^*$$

$$0 \le \lambda^i \le 1$$

- Has only n nonzeros because there are only n constraints

Sandia National Laboratories

FDT heuristic

- Some of the decompositions will have only one child.
- If any of the x^i are integral, no further decomposition. They can participate in LPP (travel to next “level” logically).
- If this were to run n levels, all leaves would be feasible integral solutions.
- Running to the end level could be very expensive
  - Combine this with randomized rounding or other heuristics

Sandia National Laboratories

Parallelizing FDT

- Child decompositions on each level are independent

- Alternatively, can “dive” through the FDT
  - Do a child decomposition
  - Pick a single child
  - Travel a single path to a leaf
  - This can fail even when the full computation would not
- Each processor can dive independently

Final remarks

- For important applications, customization is best
  - PICO provides tools for each addition of custom incumbent heuristics
  - If using ampl modeling language, ampl variables are available directly within PICO
- Expect FDT will be the sledgehammer for when nothing else works.

- Key challenge: managing parallel heuristics