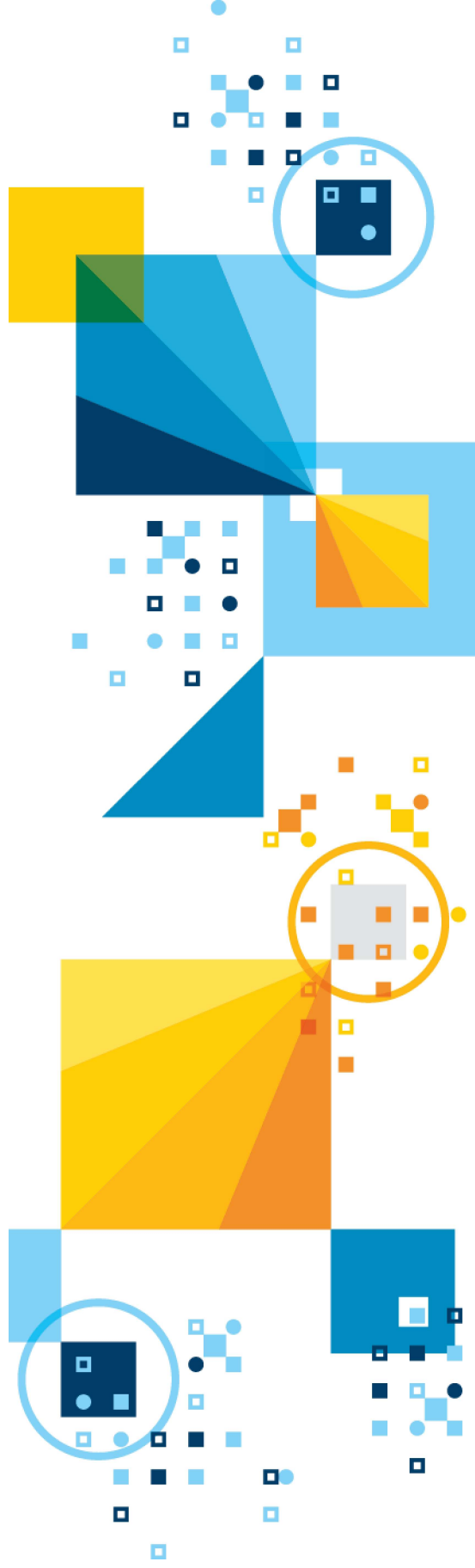


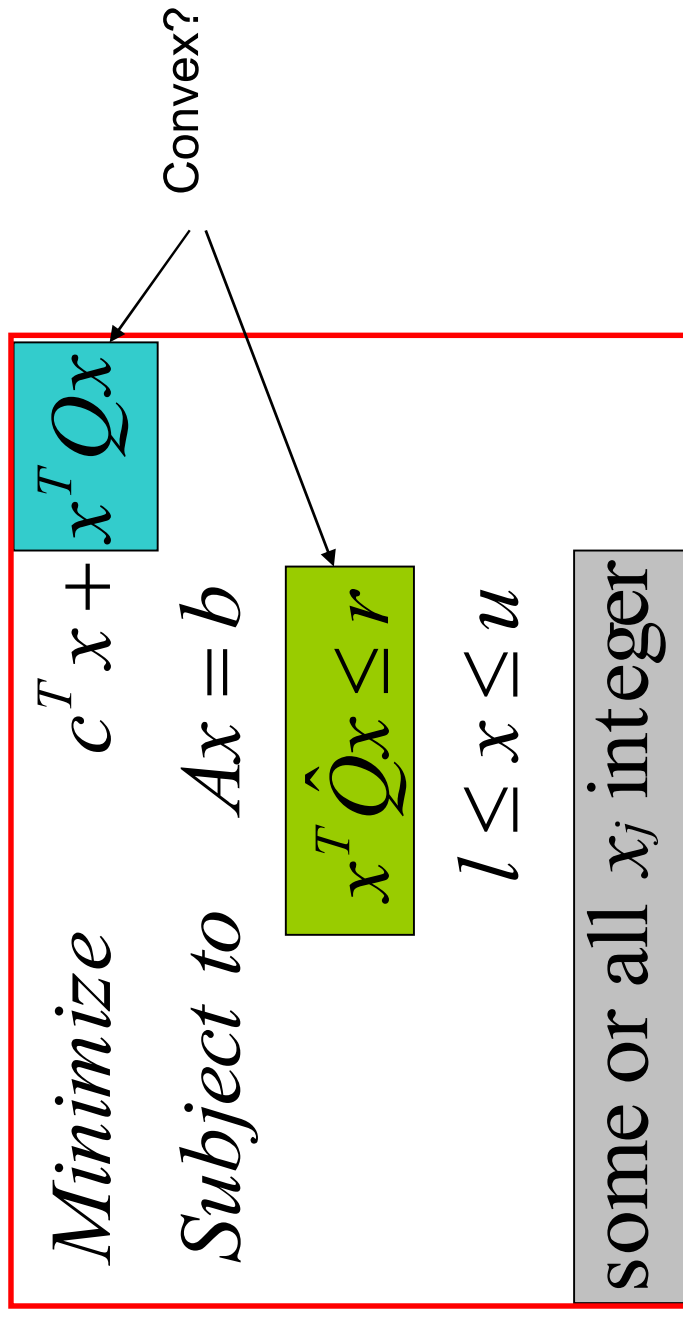
Industrial Careers in Mathematical Programming



Careers in Math Programming

- Problems
- Algorithms
- Capabilities and Limitations
- Industries
- Mathematics
- Summary

Math Programming



The diagram shows a linear programming problem within a red rectangular frame. The objective function is $\text{Minimize } c^T x + x^T Q x$, with $x^T Q x$ highlighted in a blue box. The constraints are $\text{Subject to } Ax = b$ and $x^T \hat{Q} x \leq r$, with the latter highlighted in a green box. Below these is the bound $l \leq x \leq u$. A grey box at the bottom states "some or all x_j integer". An arrow points from the text "Convex?" to the blue box, and another arrow points from the same text to the green box.

$$\begin{array}{ll} \text{Minimize} & c^T x + x^T Q x \\ \text{Subject to} & Ax = b \\ & x^T \hat{Q} x \leq r \\ & l \leq x \leq u \\ & \text{some or all } x_j \text{ integer} \end{array}$$

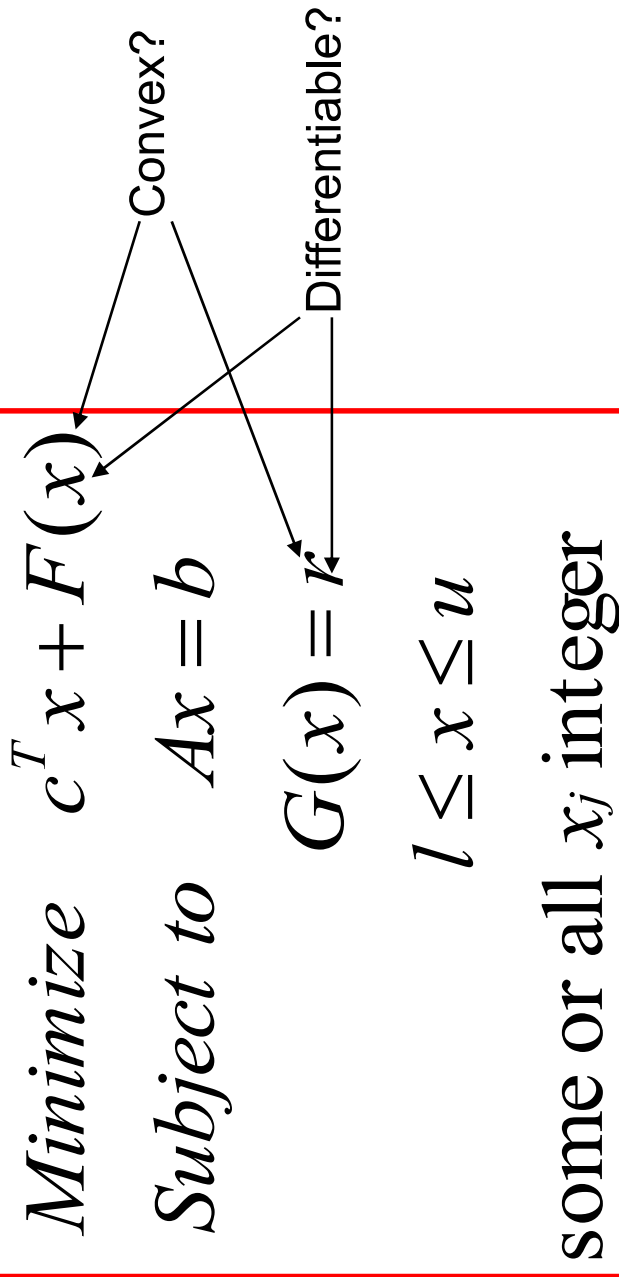
Convex?

Math Programming

$$\begin{array}{ll} \text{Minimize} & c^T x + F(x) \\ \text{Subject to} & Ax = b \\ & G(x) = k \\ & l \leq x \leq u \\ & \text{some or all } x_j \text{ integer} \end{array}$$

Convex?

Differentiable?



Algorithms for Linear Programs

$$\begin{array}{ll} \text{Maximize} & 3x_1 + 2x_2 + 2x_3 \\ \text{Subject to} & x_1 + \quad x_3 \leq 8 \\ & x_1 + \quad x_2 \leq 7 \\ & x_1 + 2x_2 \leq 12 \\ & x_1, x_2, x_3 \geq 0 \end{array}$$

Algorithms for LP, QP: The Simplex Algorithm

Current Basis

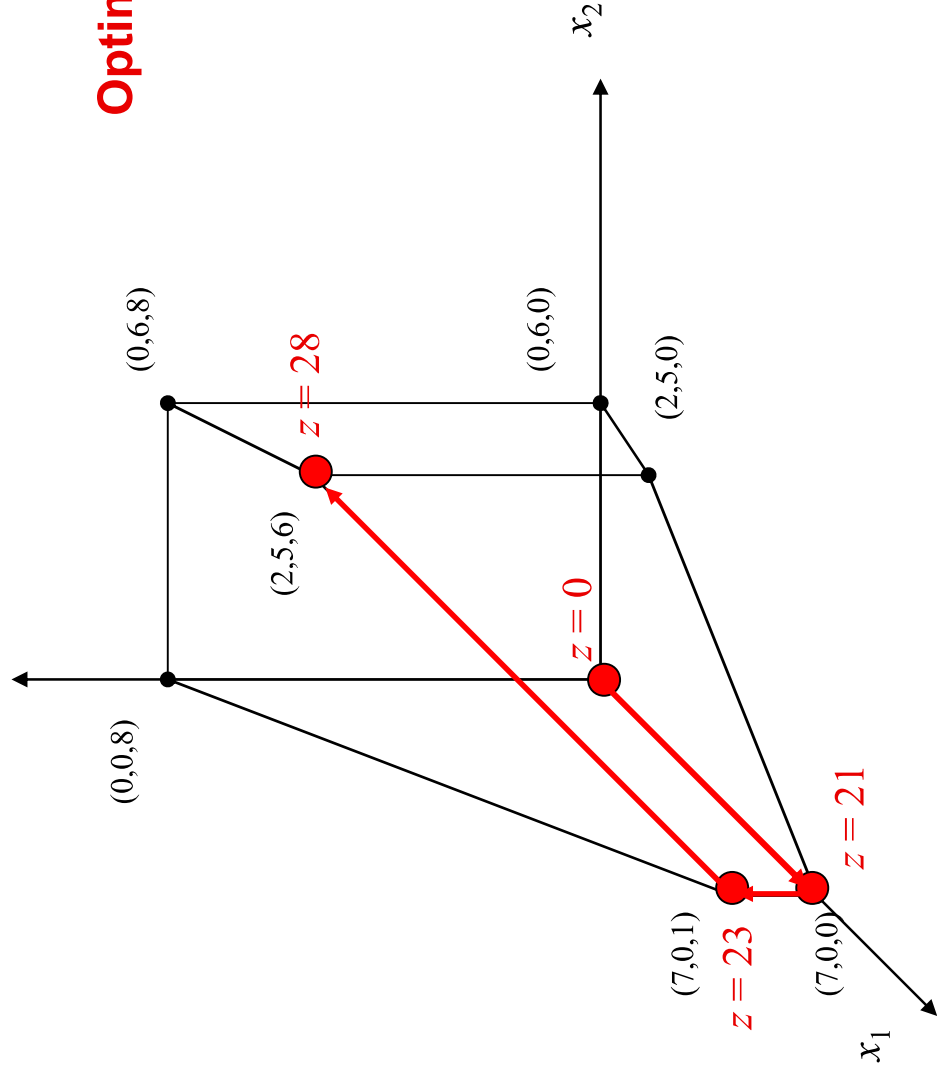
~~(x_4, x_5, x_6)~~

~~(x_4, x_1, x_6)~~

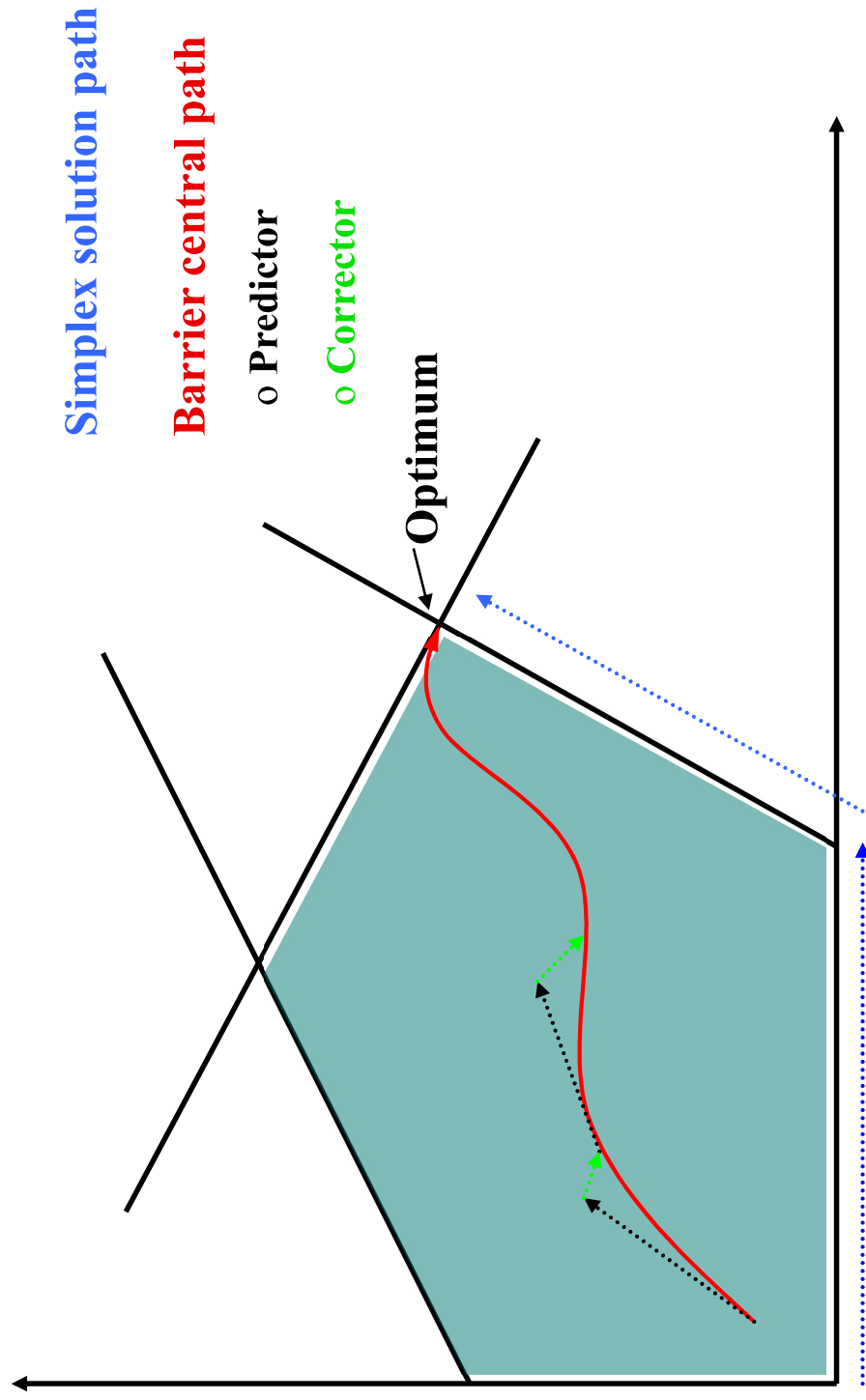
~~(x_3, x_1, x_6)~~

(x_3, x_1, x_2)

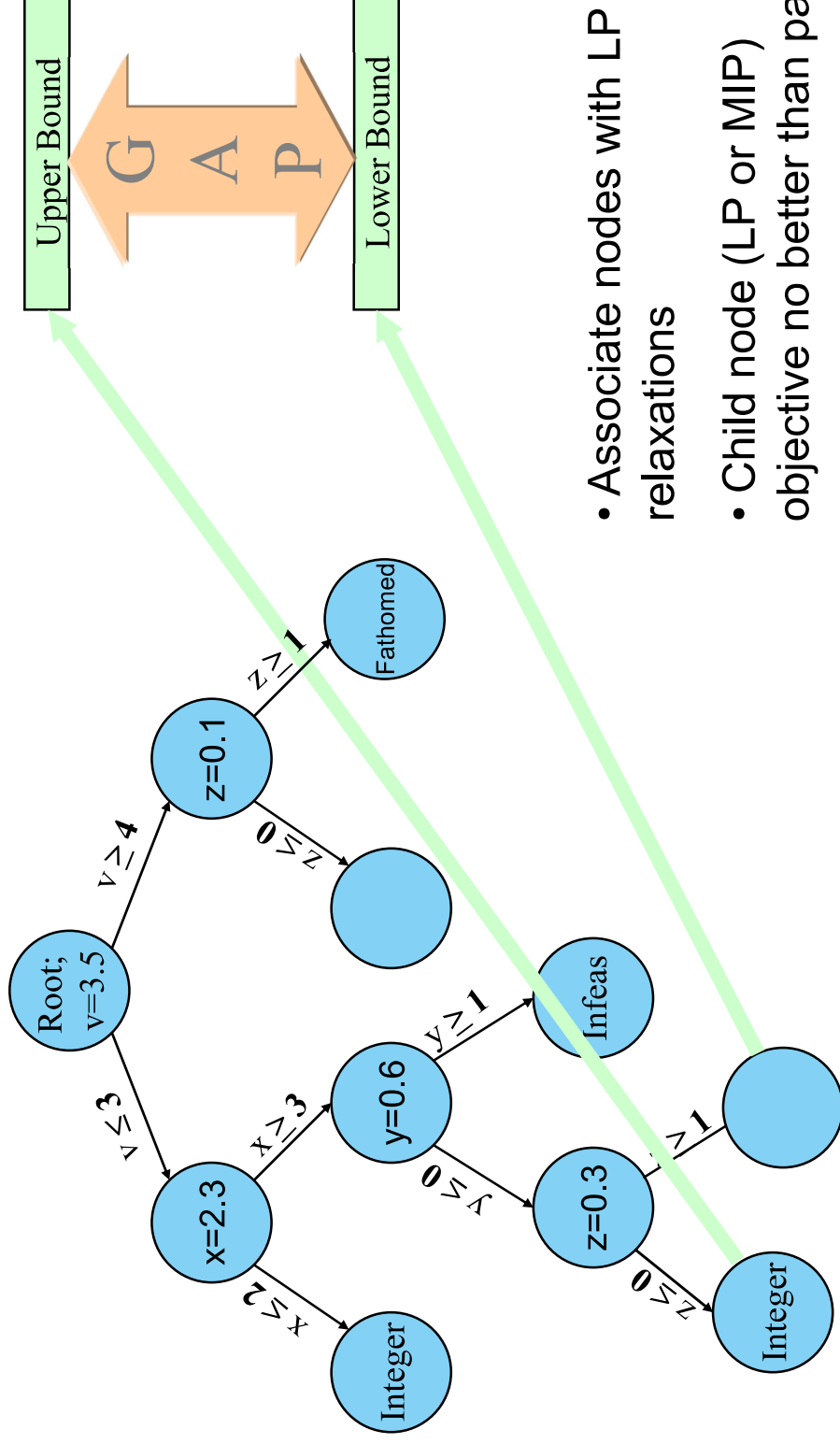
$$\text{Maximize } z = 3x_1 + 2x_2 + 2x_3$$



Algorithms for LP, QP, QCP: The Barrier Algorithm



Algorithms for MILP, MIQP, MIQCP

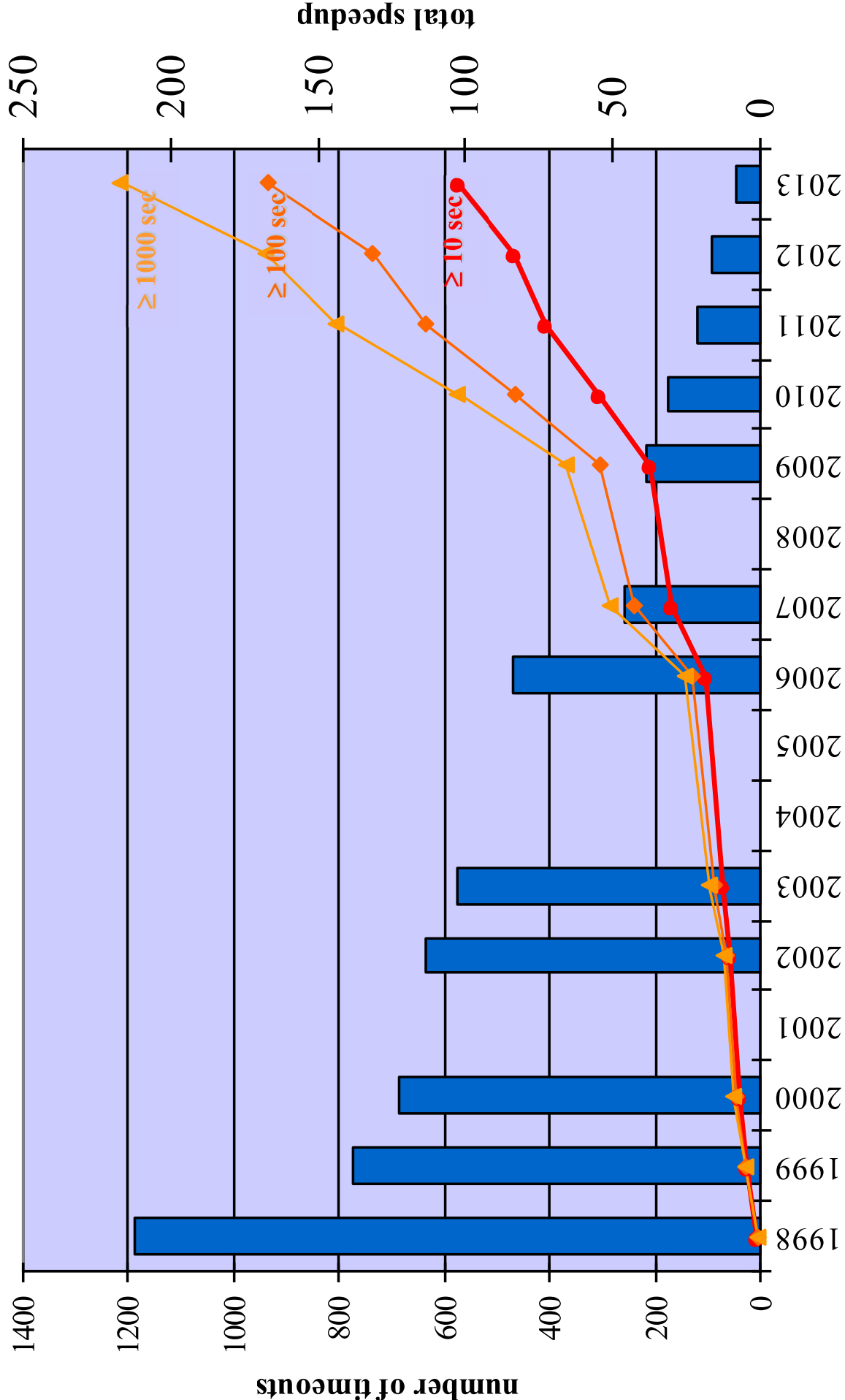


LP Progress 1988 - 2004

([Operations Research](#), Jan 2002, pp. 3—15, updated in 2004)

- **Algorithms** (*machine independent*):
Primal *versus* best of Primal/Dual/Barrier 3300x
- **Machines** (workstations → PCs): 1600x
- NET: Algorithm × Machine 5 300 000x
(2 months/5300000 ≈ 1 second)

As with any statistics, interpretation requires care



Date: 28 September 2013
Testset: 3147 models (1792 in ≥ 10 sec, 1554 in ≥ 100 sec, 1384 in ≥ 1000 sec)
Machine: Intel X5650 @ 2.67GHz, 24 GB RAM, 12 threads (deterministic since CPLEX 11.0)
Timelimit: 10,000 sec

Progress

- Synergy between algorithmic improvements in software and improvements in hardware
 - Chip speed
 - There is a free lunch; the same code just runs faster
 - More memory
 - Enables implementation of algorithmic ideas that were previously too expensive
 - More flexible data structures that require more memory
 - Enables more accurate computation
 - No longer need to store floating point numbers in 32 bits
 - Can even store in 128 bits as need arises
 - Enables use of multiple threads, parallel computation
 - Barrier, branch and bound parallelize better than simplex
 - Users solve larger, more challenging models
 - New insights into performance improvements

Current Capabilities

- Linear Programs
 - Allow 1 gigabyte of memory per million constraints
 - # of variables much less influential
- Quadratic Programs
 - Add 1 gigabyte per million variables with quadratic terms
 - Less precise than LP estimate
- Quadratically Constrained Programs
 - Add 1 gigabyte per million (cumulative # of rows in all quadratic constraint matrices)
- Mixed Integer Program
 - LP/QP/QCP requirement provides a weak lower bound
 - Depends on size of branch and bound tree
 - Recent versions of CPLEX can efficiently swap B&B tree to disk
- More memory needed as number of parallel threads in use increases

Current Capabilities

- Linear Programs
 - CPLEX has solved models with 10-50 million constraints
 - Can solve models with essentially unlimited number of variables with decomposition techniques like column generation
- Quadratic Programs
 - CPLEX capable of solving models with over a million constraints and variables
 - # of nonzeros in quadratic objective matrix affects memory usage
- Quadratically Constrained Programs
 - CPLEX capable of solving models with 1000-10000 quadratic constraints
 - More if only a few variables per quadratic constraint

Current Capabilities

- Mixed Integer Programs
 - Problem size typically is not the issue
 - Large MIPs can be easy to solve
 - Small MIPs can be extremely hard to solve
 - Tight versus Weak Formulations

Industrial Applications

- **Inventory Optimization**
- **Supply Chain Network Design**
- **Production Planning**
- **Detailed Scheduling**
- **Shipment Planning**
- **Truck Loading**
 - **3 dimensional bin packing**
- **Maintenance Scheduling**
- **Machine Learning/Big Data Analytics**

Industrial Applications

- **Depot/warehouse location**
- **Fleet Assignment**
 - **Set partitioning**
- **Crew scheduling**
 - **Set covering**
- **Network design**
- **Vehicle & container loading**
- **Vehicle routing & delivery scheduling**
 - **Travelling salesman problem**
- **Yard, Crew, Driver & Maintenance scheduling**

Industrial Applications

- **Fleet Assignment, Crew Scheduling**

$$S = \{1, \dots, n\}$$

$$P_j \subseteq S, j=1, \dots, m$$

Set partition (fleet assignment): Find $j_1, \dots, j_k$ such that

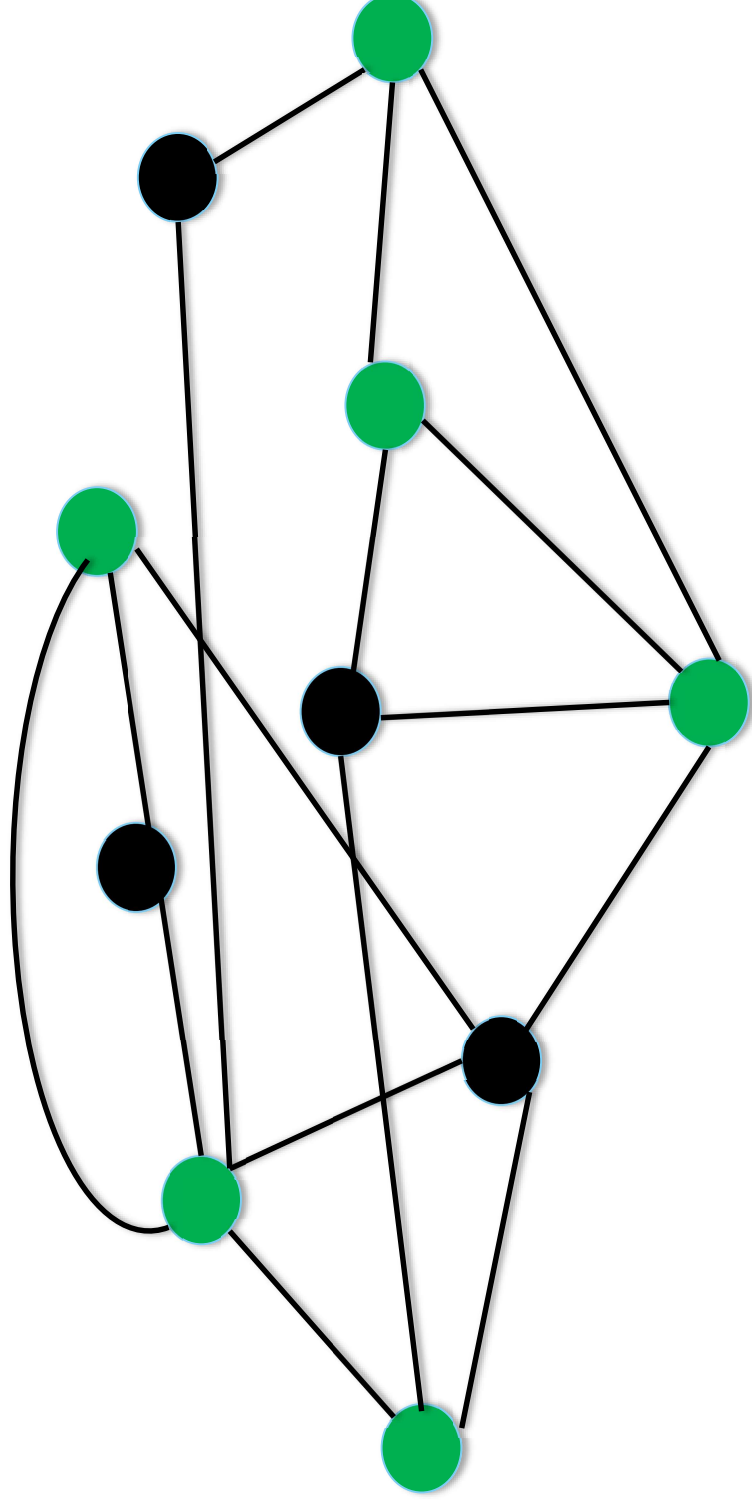
$P_{j_2} \cup P_{j_2} \cup \dots \cup P_{j_k}$ has exactly one occurrence of $1, \dots, n$

Set covering (crew scheduling): Find $j_1, \dots, j_k$ such that

$P_{j_1} \cup P_{j_2} \cup \dots \cup P_{j_k}$ has at least one occurrence of $1, \dots, n$

Industrial Applications

- **Vehicle routing & delivery scheduling**
- ### Travelling Salesman Problem



Industrial Applications

- **Portfolio Optimization and rebalancing**
- **Portfolio indexing**
- **Trade crossing**
- **Loan pooling**
- **Product/Price recommendations**

Industrial Applications

- **Supply portfolio planning**
- **Power generation scheduling**
- **Distribution planning**
- **Reservoir management**
- **Mine operations**
- **Timber harvesting**

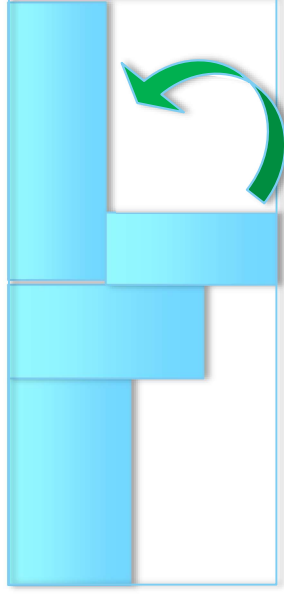
Industrial Applications

- **Network capacity planning**
- **Routing**
- **Network configuration**
- **Antenna and concentrator location**
- **Equipment and service configuration**
- **Semiconductor design**

Industrial Applications

- **Semiconductor design**
- 2 dimensional bin packing problem**

Given a collection of m rectangles with widths w_i and heights h_i , place them in a 2 dimensional grid with no overlap so as to minimize the area of the smallest rectangle containing all of them.



Industrial Applications

- **Equipment and service configuration (radiation therapy for tumors)**

Integer Matrix

Given an integer matrix C of positive integer values and a collection of m binary matrices $B_1, \dots, B_m$, find the integer linear combination of the binary matrices that comes closest to C

Minimize $|C - \sum_{i=1}^m \lambda_i B_i|$, λ integer

Industrial Applications

- **Equipment and service configuration
(radiation therapy for tumors)**



Industrial Applications

- **Workforce scheduling**
- **Advertising scheduling**
- **Marketing campaign optimization**
- **Shelf/Layout optimization**
- **Revenue/Yield Management**
- **Appointment/Field service scheduling**
- **Combinatorial Auctions for Procurement**

Math Programming Software Providers

- **Optimizers**
 - IBM CPLEX/CP Optimizer
 - SAS Institute
 - Fair Isaacs (XPRESS-MP)
 - Gurobi Optimization, Inc.
 - **Stanford Systems Optimization Lab (MINOS, other general nonlinear solvers)**
 - **Artelys (KNITRO)**
 - **COIN (open source)**
 - **Zuse Institute Berlin (SCIP, open source)**

Math Programming Software Providers

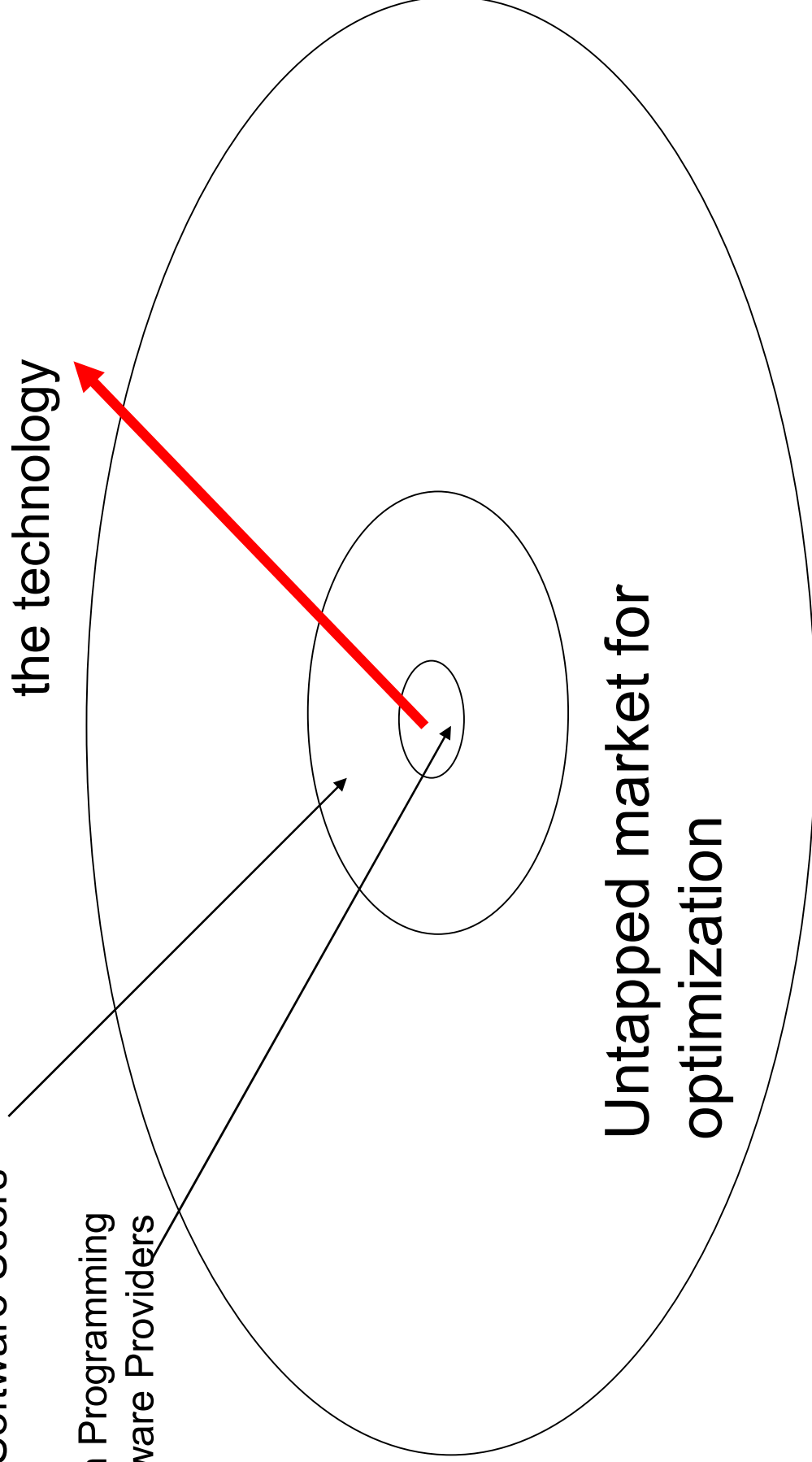
- **Modeling Language Vendors**
 - **IBM (OPL)**
 - **Fair Isaac Company(MOSEL)**
 - **Maximal Software (MPL)**
 - **AMPL Optimization LLC (AMPL)**
 - **GAMS Development Corp. (GAMS)**
 - <http://www.gams.com> (click on solvers link) has an extensive list of solvers
- **Paragon Software (AIMMS)**

The Big Picture

Math Programming
Software Users

Math Programming
Software Providers

Effort required to
communicate the benefits of
the technology



A Vehicle and Container Loading example

“ ...working out the most efficient way to pack up and down the trucks, since saving one truck ... could save something in the region of \$100,000.”

Everybody wants to manage costs

“One of the most important aspects of these rehearsals was working out the most efficient way to pack up and down the trucks, since saving one truck for one year of touring could save something in the region of \$100,000.”

-Inside Out, A Personal History of Pink Floyd

Nick Mason

Mathematics: The Simplex Algorithm

- The given problem: (A an $m \times n$ matrix)
Minimize $c^T x$
Subject to $Ax = b$
 $x \geq 0$
- Starting point: A nonsingular $m \times m$ basis matrix A_B
where $x_B^* = A_B^{-1} b \geq 0$, i.e.,
 A_B is feasible.

Algorithms for LP, QP

Current Basis

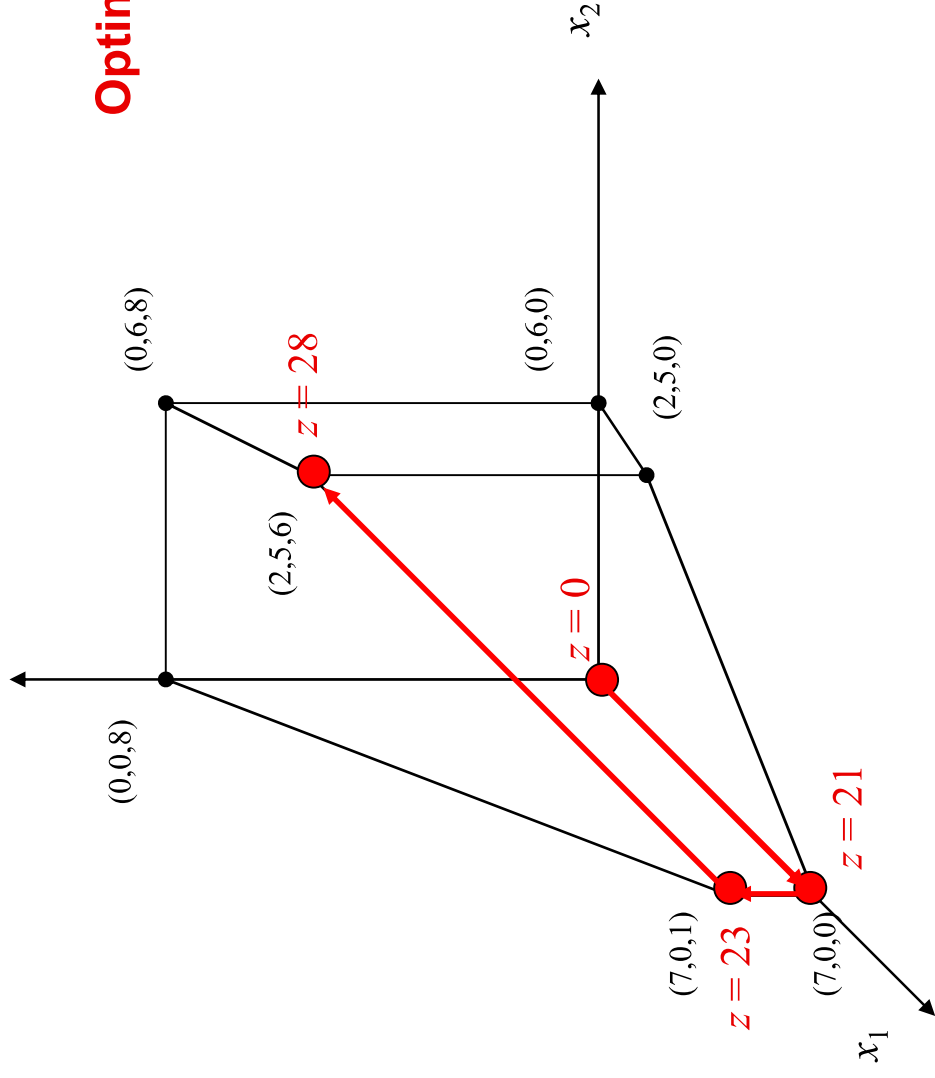
~~(x_4, x_5, x_6)~~

~~(x_4, x_1, x_6)~~

~~(x_3, x_1, x_6)~~

(x_3, x_1, x_2)

Maximize $z = 3x_1 + 2x_2 + 2x_3$

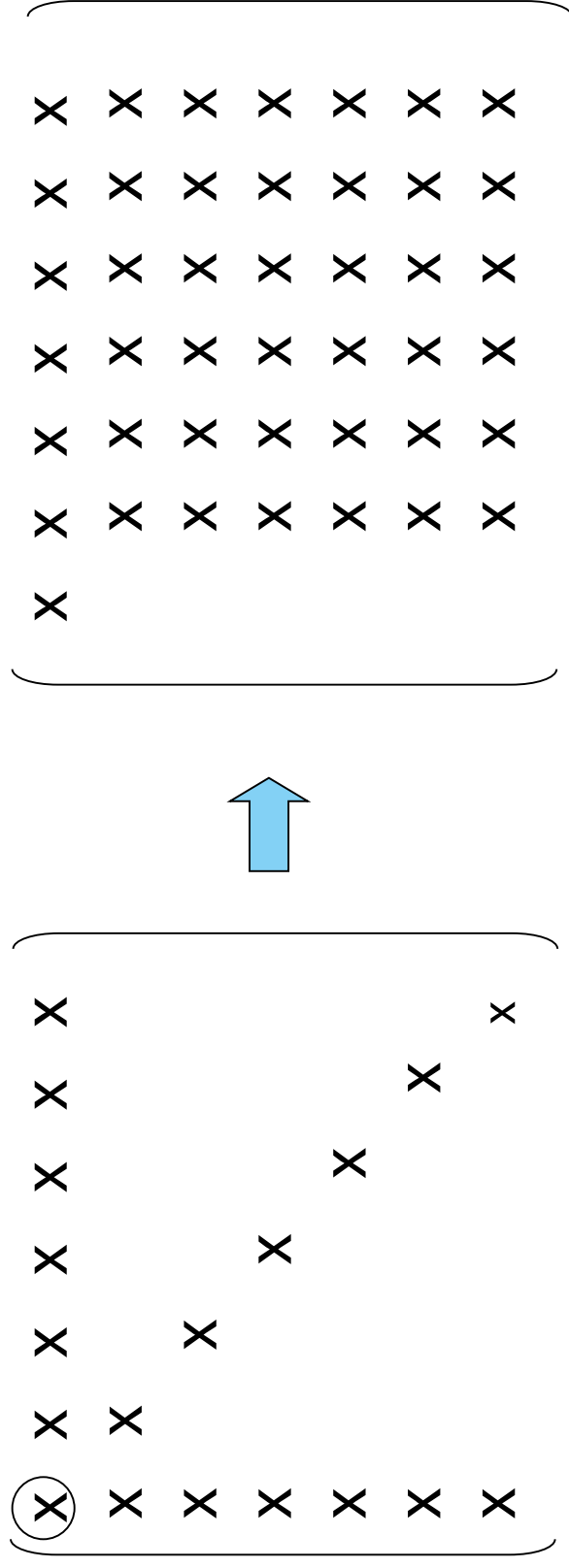


Mathematics: The Simplex Algorithm

- **Factorization:** Every 200 iterations compute $A_B = LU$
- **BTRAN:** Solve $y^T A_B = c_B^T$.
- **Pricing:** Compute the *reduced costs* $d_N^T = c_N^T - y^T A_N$. If $d_N \geq 0 \Rightarrow$ optimal, else pick $d_j < 0$ (x_j = entering variable).
- **FTRAN:** Solve $A_B z = A_j$.
- **Ratio Test:** Determine step length; conclude unbounded or find *leaving variable*.
- **Update:** Update B and the factorization of A_B .

Mathematics: The Simplex Algorithm

Compute $A_B = LU$



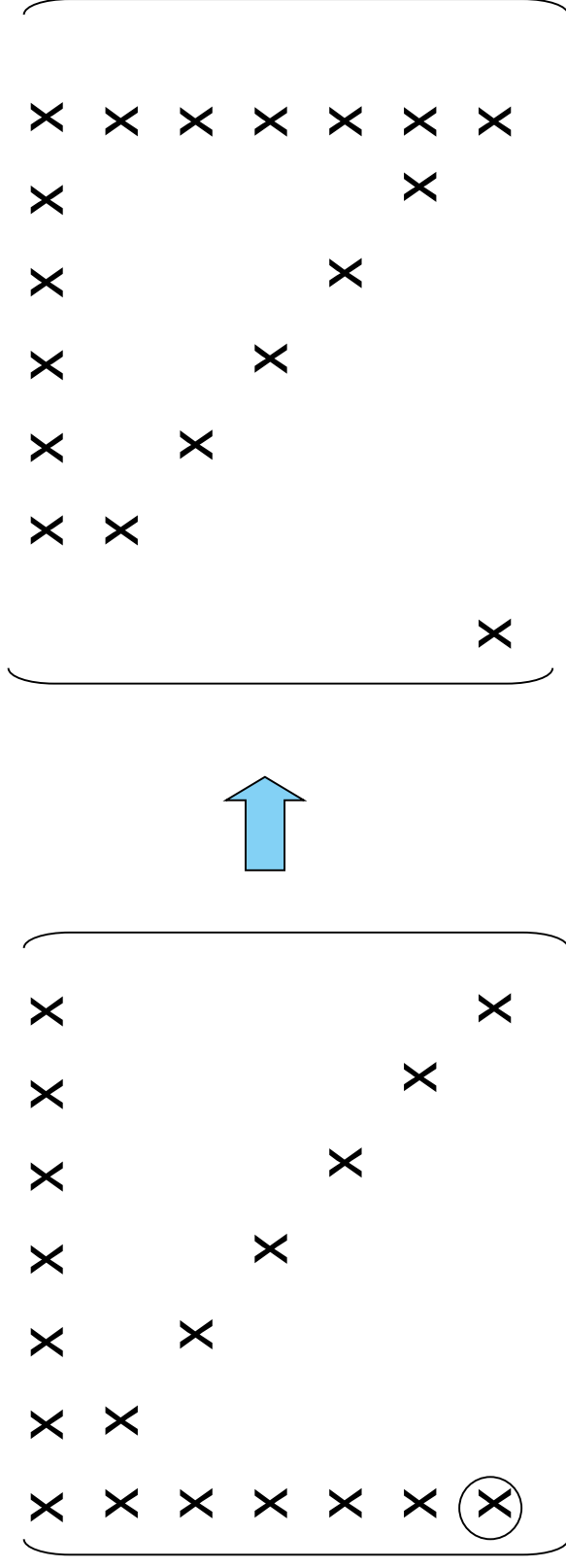
Mathematics: The Simplex Algorithm

Solve $y^T A_B = y^T L U = c_B^T$; solve $A_B z = L U z = A_j$.

$$\begin{pmatrix} \times & & & & & & & \\ & \times & & & & & & \\ & & \times & & & & & \\ & & & \times & & & & \\ & & & & \times & & & \\ & & & & & \times & & \\ & & & & & & \times & \\ & & & & & & & \times \end{pmatrix} \begin{pmatrix} \times & \times & \times & \times & \times & \times & \times & \times \\ & \times & \times & \times & \times & \times & \times & \times \\ & & \times & \times & \times & \times & \times & \times \\ & & & \times & \times & \times & \times & \times \\ & & & & \times & \times & \times & \times \\ & & & & & \times & \times & \times \\ & & & & & & \times & \times \\ & & & & & & & \times \end{pmatrix}$$

Mathematics: The Simplex Algorithm

Compute $A_B = LU$



Mathematics: The Simplex Algorithm

Solve $y^T A_B = y^T L U = c_B^T$; solve $A_B z = L U z = A_j$.

L

$$\begin{pmatrix} x & & & & & & \\ & x & & & & & \\ & & x & & & & \\ & & & x & & & \\ & & & & x & & \\ & & & & & x & \\ & & & & & & x \end{pmatrix}$$

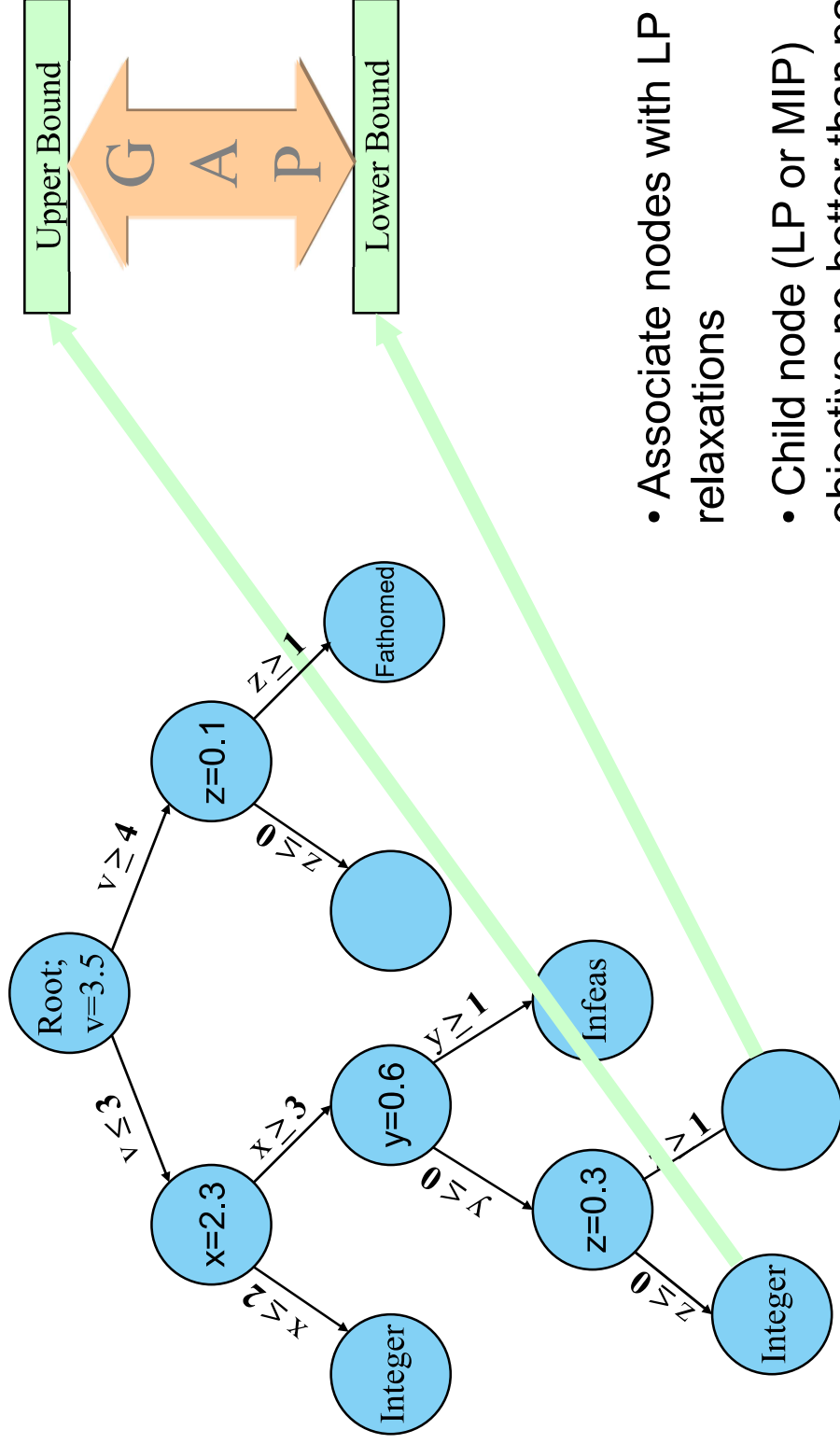
U

$$\begin{pmatrix} x & & & & & & \\ & x & & & & & \\ & & x & & & & \\ & & & x & & & \\ & & & & x & & \\ & & & & & x & \\ & & & & & & x \end{pmatrix}$$

Mathematics: The Simplex Algorithm

- Ordering in factorization affects Ftran and Btran times
- Ftran and Btran times can comprise 50% or more of run time for many LPs
- Computing optimal ordering is a combinatorial problem that is NP-complete
- Effective methods to compute better orderings will speed up the simplex method
 - Discrete math needed despite all continuous variables
- Ordering needed to ensure limited round off error in solves (numerical linear algebra)
- Ordering for factorizations also plays a (somewhat different) role in the barrier method
- Additional challenges to exploit parallel processing

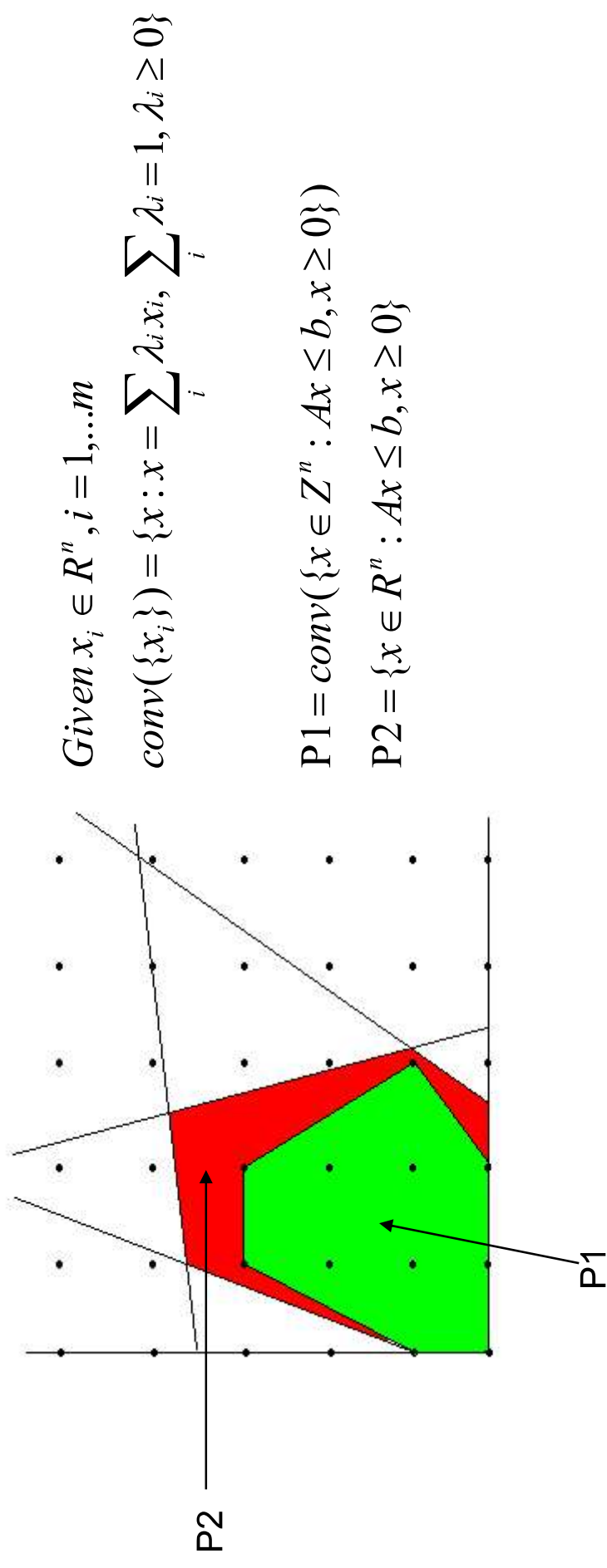
Mathematics: Branch and Bound for MIP



- Associate nodes with LP relaxations
- Child node (LP or MIP) objective no better than parent

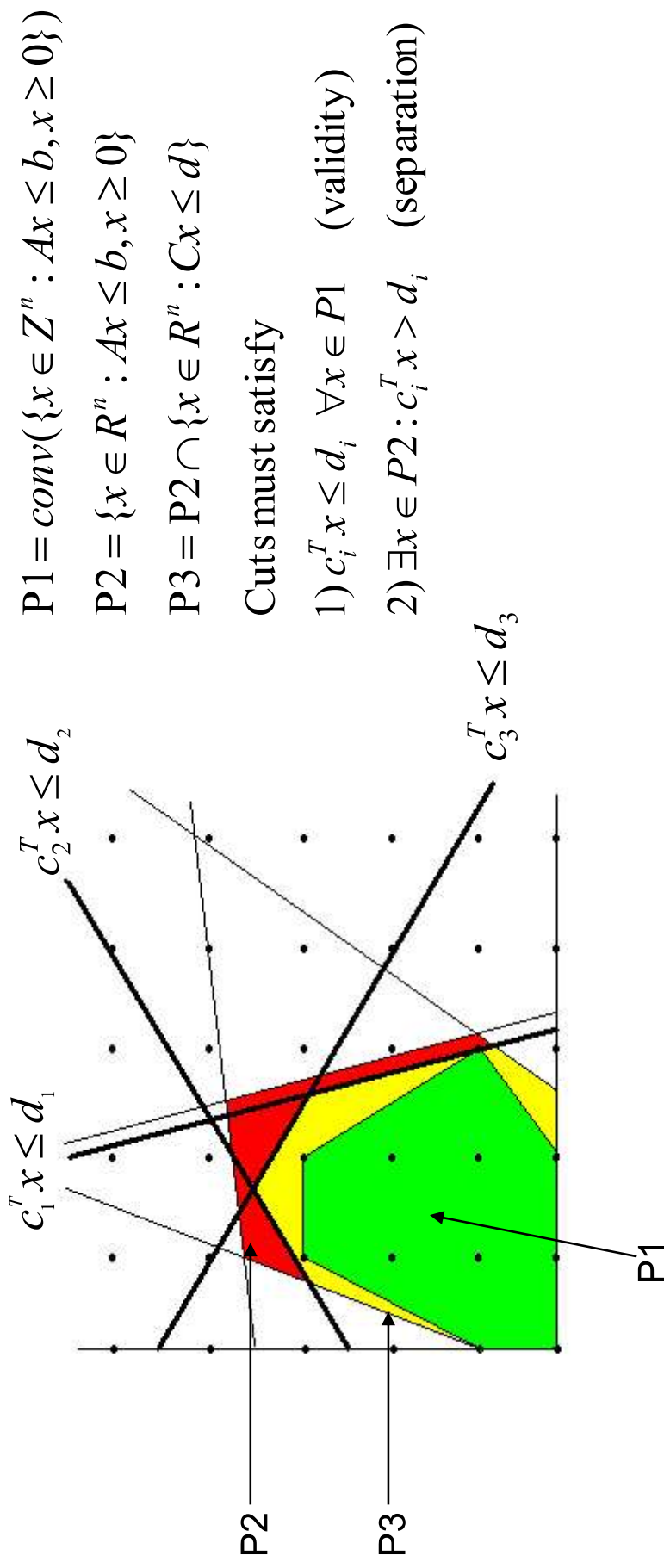
Mathematics: Branch and Bound for MIP

- Add constraints to strengthen formulation



Mathematics: Branch and Bound for MIP

- Add constraints to strengthen formulation



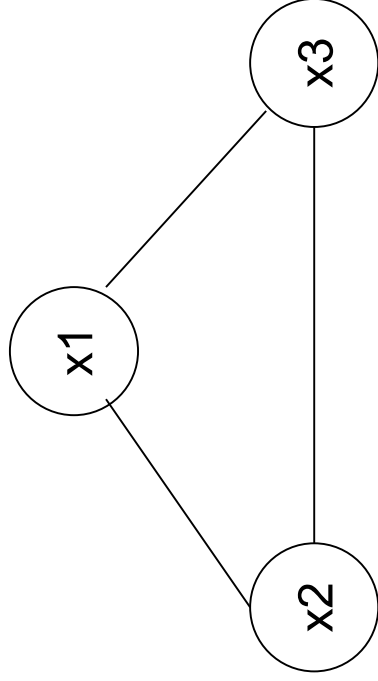
Mathematics: Branch and Bound for MIP

- Clique cuts (conflict graph)

$$P2 = \{x \in R_+^3 : x1 + x2 \leq 1,$$

$$x1 + x3 \leq 1,$$

$$x2 + x3 \leq 1, \quad 0 \leq x_i \leq 1\}$$

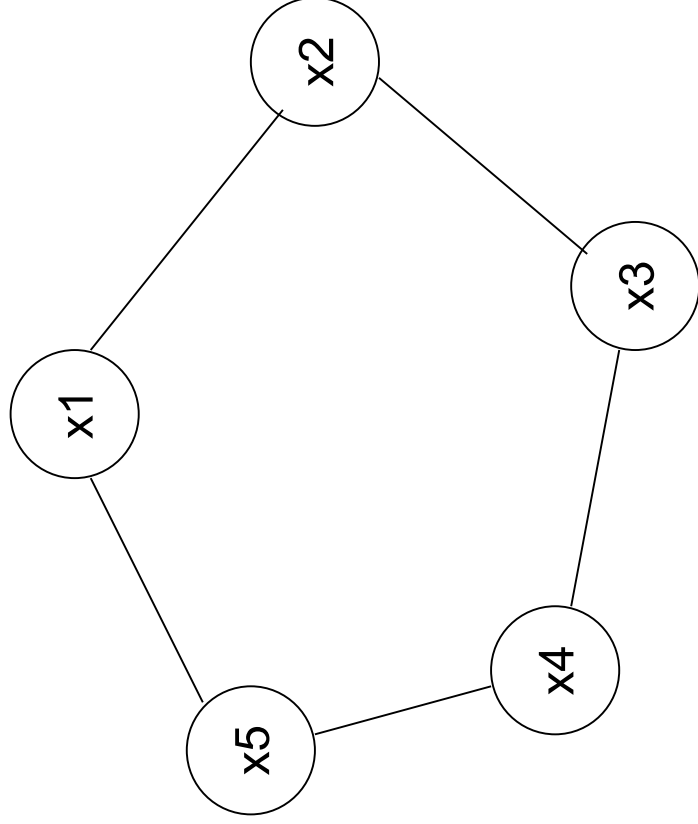


Clique Cut : $x1 + x2 + x3 \leq 1$

- 1) Valid? Suppose $q \in P2$ is integral but violates the cut. Then $q1 + q2 + q3 \geq 2$, but $q \in P2$ implies $q1 + q2 + q3 \leq 1.5$
- 2) Separation? Yes; consider $(.5, .5, .5)$.

Mathematics: Branch and Bound for MIP

- More with the conflict graph



$$P2 = \{x \in R_+^5 : x1 + x2 \leq 1, x2 + x3 \leq 1,$$

$$x3 + x4 \leq 1, x4 + x5 \leq 1$$

$$x5 + x1 \leq 1, 0 \leq x_i \leq 1\}$$

Odd Cycle Cut : $x1 + x2 + x3 + x4 + x5 \leq 2$

1) Valid? Use same argument as clique cut, or note that max.

cardinality matching = 2

2) Separation? Yes; consider $(.5, .5, .5, .5, .5)$.

Mathematics: Branch and Bound for MIP

- Mixed integer rounding cut

$$P = \{x \in Z_+^3 : 4x_1 + 3x_2 + 5x_3 = 10\}$$

$$P = \{x \in Z_+^3 : x_1 + \frac{3}{4}x_2 + \frac{5}{4}x_3 = \frac{5}{2}\}$$

$$P = \{x \in Z_+^3 : \underbrace{x_1 + x_2 + x_3}_{t} - \frac{1}{4}x_2 + \frac{1}{4}x_3 = 2 + \frac{1}{2} = 3 - \frac{1}{2}\}$$

$$t \leq 2 \Rightarrow -\frac{1}{4}x_2 + \frac{1}{4}x_3 \geq \frac{1}{2} \Rightarrow x_3 \geq 2$$

$$t \geq 3 \Rightarrow -\frac{1}{4}x_2 + \frac{1}{4}x_3 \leq -\frac{1}{2} \Rightarrow x_2 \geq 2$$

Combining, $x_2 + x_3 \geq 2$; cuts off $(2, \frac{1}{3}, \frac{1}{5})$

Mathematics: Branch and Bound for MIP

- Given $y, x_j \in \mathbb{Z}_+$, and

$$y + \sum a_j x_j = d = \lfloor d \rfloor + f, \quad f > 0$$
- **Rounding:** Where $a_j = \lfloor a_j \rfloor + f_j$, define

$$t = y + \sum (\lfloor a_j \rfloor x_j: f_j \leq f) + \sum (\lceil a_j \rceil x_j: f_j > f) \in \mathbb{Z}$$
- Then
$$\begin{aligned} \sum (f_j x_j: f_j \leq f) + \sum ((f_j - 1) x_j: f_j > f) &= d - t = (\lfloor d \rfloor - t) + f &> 0 \\ &= (\lceil d \rceil - t) + f - 1 &\leq 0 \end{aligned}$$
- **Disjunction:**

$$\begin{aligned} t \leq \lfloor d \rfloor &\Rightarrow \sum (f_j x_j: f_j \leq f) \geq f \\ t \geq \lceil d \rceil &\Rightarrow \sum ((1 - f_j) x_j: f_j > f) \geq 1 - f \end{aligned}$$
- Combining:

$$\sum ((f_j/f) x_j: f_j \leq f) + \sum (((1 - f_j)/(1 - f)) x_j: f_j > f) \geq 1$$

Mathematics: Branch and Bound for MIP

- Cuts on set partitioning constraints
 - Sets: $\{1,2\}$, $\{1,3\}$, $\{2,3\}$, $\{4\}$, $\{5\}$, $\{6\}$

$$P2 = \{x \in R_+^6 : x1 + x2 + x4 = 1,$$

$$x1 + x3 + x5 = 1,$$

$$x2 + x3 + x6 = 1,$$

$$0 \leq x_i \leq 1\}$$

Add them up : $2x1 + 2x2 + 2x3 + x4 + x5 + x6 = 3$

Integrality implies $x4 + x5 + x6 \geq 1$

Validity : $x4 + x5 + x6 \leq 0 \Rightarrow 2x1 + 2x2 + 2x3 = 3$

Separation : Consider $(.5, .5, .5, 0, 0, 0)$

Mathematics: Branch and Bound for MIP

- Many other types of cuts exist
- CPLEX effectively finds most “generic” cuts that are common to many different types of models
- User knowledge about specific models can yield additional cuts that optimizers like CPLEX probably won’t find
- Anything goes
 - Graph Theory
 - Number Theory
 - Algebra
 - Polyhedral Theory
 - Satisfiability/Logic

Looking forward

- We will solve even bigger problems in the future
 - Computers may continue to increase capacity and speed even if no more algorithmic improvements occur
 - Availability of more memory enables implementations of performance improvements that were previously impossible
 - More memory and speed enables study of larger problems, leading to new insights
 - Computer speed improvements more likely to come from more cores/CPU's than faster individual chip speeds
 - Programming in parallel poses additional challenges
- Don't dismiss LP and MIP algorithms of today on models what were too difficult 10 or more years ago
- 10 years from now, don't dismiss LP and MIP algorithms on models that are too difficult today

Looking forward

- Big data analytics problems often involve huge problem sizes
 - Often too big for current state of the art math programming solvers
 - But less intelligent algorithms that parallelize better have their limitations as well.
 - Understanding of decomposition algorithms in linear and integer programming can provide insights for other algorithms

Summary

- Many different types of interesting mathematics and computer science challenges
- Good programmers with math background will have many opportunities (and vice versa)
- Many different types of industries make use of optimization
 - But perhaps more opportunities in those that don't
- Graduate degree is helpful
- Interesting, useful work that pays reasonably well (or better)
 - Get paid to solve interesting math puzzles

Additional Reading

- Wolsey, L. Integer Programming
- Wright, S. Primal-Dual Interior Point Methods
- Bixby, R. Solving Real-World Linear Programs: A Decade and More of Progress
(www.caam.rice.edu/tech_reports/2001/TR01-20.pdf)
- Woolsey, R. Real World Operations Research: The Woolsey Papers (Seminar in Math. Modeling)
- Williams, H.P. Model Building in Mathematical Programming (Seminar in Math. Modeling)



Backup



Legal Disclaimer

- © IBM Corporation 2015. All Rights Reserved.
- The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.
- References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.
- If the text contains performance statistics or references to benchmarks, insert the following language; otherwise delete:
Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.
- If the text includes any customer examples, please confirm we have prior written approval from such customer and insert the following language; otherwise delete:
All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.
- Please review text for proper trademark attribution of IBM products. At first use, each product name must be the full name and include appropriate trademark symbols (e.g., IBM Lotus® Sametime® Unyte™). Subsequent references can drop "IBM" but should include the proper branding (e.g., Lotus Sametime Gateway, or WebSphere Application Server). Please refer to <http://www.ibm.com/legal/copytrade.shtml> for guidance on which trademarks require the ® or ™ symbol. Do not use abbreviations for IBM product names in your presentation. All product names must be used as adjectives rather than nouns. Please list all of the trademarks that you use in your presentation as follows; delete any not included in your presentation. IBM, the IBM logo, Lotus, Lotus Notes, Notes, Domino, Quickr, Sametime, WebSphere, UC2, PartnerWorld and Lotusphere are trademarks of International Business Machines Corporation in the United States; other countries, or both. Unyte is a trademark of WebDialogs, Inc., in the United States; other countries, or both.
- If you reference Adobe® in the text, please mark the first use and include the following; otherwise delete:
Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
- If you reference Java™ in the text, please mark the first use and include the following; otherwise delete:
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- If you reference Microsoft® and/or Windows® in the text, please mark the first use and include the following, as applicable; otherwise delete:
Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.
- If you reference Intel® and/or any of the following Intel products in the text, please mark the first use and include those that you use as follows; otherwise delete:
Intel, Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
- If you reference UNIX® in the text, please mark the first use and include the following; otherwise delete:
UNIX is a registered trademark of The Open Group in the United States and other countries.
- If you reference Linux® in your presentation, please mark the first use and include the following; otherwise delete:
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both. Other company, product, or service names may be trademarks or service marks of others.
- If the text/graphics include screenshots, no actual IBM employee names may be used (even your own), if your screenshots include fictitious company names (e.g., Renovations, Zeta Bank, Acme) please update and insert the following; otherwise delete: All references to [insert fictitious company name] refer to a fictitious company and are used for illustration purposes only.