Solving Linear Programs

Martin Grötschel

13. Februar 2013

Letzte Vorlesung

Algorithmische Diskrete Mathematik I

**„Einführung in die
Lineare und Kombinatorische Optimierung"**

WS 2012/2013, Institut für Mathematik, TU Berlin

Martin Grötschel
- Institut für Mathematik, Technische Universität Berlin (TUB)
- DFG-Forschungszentrum "Mathematik für Schlüsseltechnologien" (MATHEON)
- Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB)

groetschel@zib.de http://www.zib.de/groetschel

Einführung in die Lineare und Kombinatorische Optimierung

(Algorithmische Diskrete Mathematik I, kurz ADM I)

Skriptum zur Vorlesung im WS 2012/2013

Prof. Dr. Martin Grötschel
Institut für Mathematik
Technische Universität Berlin

Version vom 12. Februar 2013

Martin Grötschel

Inhaltsverzeichnis

Martin Grötschel

Martin Grötschel

Martin Grötschel

Contents

Martin Grötschel

Contents

Martin Grötschel

typical optimization problems

$$\max\ f(x)\ \ or\ \min\ f(x)$$
$$g_i(x) = 0, \quad i = 1, 2, \ldots, k$$
$$h_j(x) \leq 0, \quad j = 1, 2, \ldots, m$$
$$x \in \mathbb{R}^n \text{ (and } x \in S\text{)}$$

„general“
(nonlinear)
program
NLP

$$\min c^T x$$
$$Ax = a$$
$$Bx \leq b$$
$$x \geq 0$$
$$(x \in \mathbb{R}^n)$$
$$(x \in \mathbb{k}^n)$$

linear
program
LP

$$\min c^T x$$
$$Ax = a$$
$$Bx \leq b$$
$$x \geq 0$$
$$\text{some } x_j \in \mathbb{Z}$$
$$(x \in \{0,1\}^n)$$

(linear)
0/1-
mixed-
integer
program
IP, MIP

program = optimization problem

Martin Grötschel

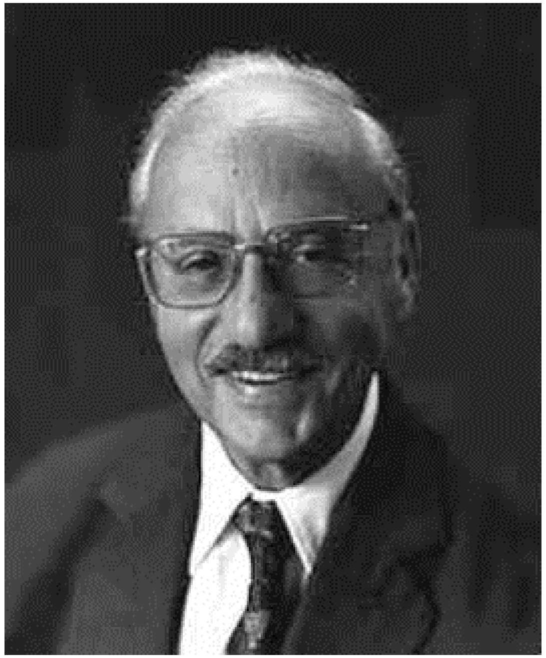
Contents

Martin Grötschel

Linear Programming: a very brief history

- 1826/1827 Jean Baptiste Joseph Fourier (1786-1830): rudimentory form of the simplex method in 3 dimensions.
- 1939 L. V. Kantorovitch (1912-1986): Foundations of linear programming (Nobel Prize 1975)
- 1947 G. B. Dantzig (1914-2005): Invention of the (primal) simplex algorithm
- 1954 C.E. Lemke: Dual simplex algorithm
- 1953 G.B. Dantzig, 1954 W. Orchard Hays, and 1954 G. B. Dantzig & W. Orchard Hays: Revised simplex algorithm

$$\max c^T x$$
$$Ax = b$$
$$x \geq 0$$



Martin Grötschel

Dantzig and Bixby

George Dantzig and
Bob Bixby
(founder of CPLEX and GUROBI)

at the International Symposium on Mathematical Programming,

Atlanta, August 2000

This lecture employs a lot of information I obtained from Bob and some of his slides.

Martin Grötschel

Optimal use of scarce ressources

foundation and economic interpretation of LP

Leonid V. Kantorovich Tjalling C. Koopmans

Nobel Prize for Economics 1975

Martin Grötschel

Stiglers „Diet Problem“: „The first linear program“

$\text{Min } x1 + x2$	costs
$2x1 + x2 \geq 3$	protein
$x1 + 2x2 \geq 3$	carbohydrates
$x1 \geq 0$	potatoes
$x2 \geq 0$	beans

minimizing the cost of food

George J. Stigler
Nobel Prize in economics 1982

Martin Grötschel

Sets n nutrients / calorie thousands , protein grams , calcium grams , iron milligrams vitamin-a thousand ius, vitamin-b1 milligrams, vitamin-b2 milligrams, niacin milligrams , vitamin-c milligrams /

f foods / wheat , cornmeal , cannedmilk, margarine , cheese , peanut-b , lard liver , porkroast, salmon , greenbeans, cabbage , onions , potatoes spinach, sweet-pot, peaches , prunes , limabeans, navybeans /

Parameter b(n) required daily allowances of nutrients / calorie 3, protein 70 , calcium .8 , iron 12 vitamin-a 5, vitamin-b1 1.8, vitamin-b2 2.7, niacin 18, vitamin-c 75 /

Table a(f,n) nutritive value of foods (per dollar spent)

	calorie (1000)	protein (g)	calcium (g)	iron (mg)	vitamin-a (1000iu)	vitamin-b1 (mg)	vitamin-b2 (mg)	niacin (mg)	vitamin-c (mg)
wheat	44.7	1411	2.0	365		55.4	33.3	441	
cornmeal	36	897	1.7	99	30.9	17.4	7.9	106	
cannedmilk	8.4	422	15.1	9	26	3	23.5	11	60
margarine	20.6	17	.6	6	55.8	.2			
cheese	7.4	448	16.4	19	28.1	.8	10.3	4	
peanut-b	15.7	661	1	48		9.6	8.1	471	
lard	41.7				.2		.5	5	
liver	2.2	333	.2	139	169.2	6.4	50.8	316	525
porkroast	4.4	249	.3	37		18.2	3.6	79	
salmon	5.8	705	6.8	45	3.5	1	4.9	209	
greenbeans	2.4	138	3.7	80	69	4.3	5.8	37	862
cabbage	2.6	125	4	36	7.2	9	4.5	26	5369
onions	5.8	166	3.8	59	16.6	4.7	5.9	21	1184
potatoes	14.3	336	1.8	118	6.7	29.4	7.1	198	2522
spinach	1.1	106		138	918.4	5.7	13.8	33	2755
sweet-pot	9.6	138	2.7	54	290.7	8.4	5.4	83	1912
peaches	8.5	87	1.7	173	86.8	1.2	4.3	55	57
prunes	12.8	99	2.5	154	85.7	3.9	4.3	65	257
limabeans	17.4	1055	3.7	459	5.1	26.9	38.2	93	
navybeans	26.9	1691	11.4	792		38.4	24.6	217	

Positive Variable x(f) dollars of food f to be purchased daily (dollars)

Free Variable cost total food bill (dollars)

Equations nb(n) nutrient balance (units), cb cost balance (dollars) ;

nb(n).. sum(f, a(f,n)*x(f)) =g= b(n); cb.. cost=e= sum(f, x(f));

Model diet stiglers diet problem / nb,cb /;

http://www.gams.com/modlib/libhtml/diet.htm

Solution of the Diet Problem

Goal: Find the cheapest combination of foods that will satisfy the daily requirements of a person!

The problem motivated by the army's desire to meet nutritional requirements of the soldiers at minimum cost.

Army's problem had 77 unknowns and 9 constraints.
Stigler solved problem using a heuristic: $39.93/year (1939)
Laderman (1947) used simplex: $39.69/year (1939 prices)

→ first "large-scale computation"
took 120 man days on hand operated desk calculators (10 human "computers")

http://www.mcs.anl.gov/home/otc/Guide/CaseStudies/diet/index.html

Martin Grötschel

Commercial software

William Orchard-Hayes (in the period 1953-1954)

The first commercial LP-Code was on the market in 1954 (i.e., 57 years ago) and available on an IBM CPC (card programmable calculator):

Code: Simplex Algorithm with explicit basis inverse, that was recomputed in each step.

Shortly after, Orchard-Hayes implemented a version with product form of the inverse (idea of A. Orden),
Record: 71 variables, 26 constraints, 8 h running time

About 1960: LP became commercially viable, used largely by oil companies.

Martin Grötschel

The Decade of the 70's: Theory

- V. Klee and G. J. Minty, „How good is the simplex algorithm?", in O. Shisha (ed.), Inequalities III, Academic Press, New York, 1972, 159-172
- K. H. Borgwardt, „Untersuchungen zur Asymptotik der mittleren Schrittzahl von Simplexverfahren in der linearen Optimierung", Dissertation, U Kaiserslautern, 1977
- L. G. Khachiyan, „A polynomial algorithm in linear programming", (Russian), Doklady Akademii Nauk SSR 244 (1979) 1093-1096

The Decade of the 70's: Practice

- Interest in optimization flowered
  - Large scale planning applications particularly popular
- Significant difficulties emerged
  - Building applications was very expensive and very risky
  - Technology just wasn't ready:
    - LP was slow and
    - Mixed Integer Programming was impossible.
- OR could not really "deliver" – with some exceptions, of course
- The ellipsoid method of 1979 was no practical success.

The Decade of the 80's and beyond

- Mid 80's:
  - There was perception was that LP software had progressed about as far as it could.
- There were several key developments
  - IBM PC introduced in 1981
    - Brought personal computing to business
  - Relational databases developed. ERP systems introduced.
  - 1984, major theoretical breakthrough in LP
    N. Karmarkar, "A new polynomial-time algorithm for linear programming", Combinatorica 4 (1984) 373-395
    (Interior Point Methods, front page New York Times)
- The last ~20 years: Remarkable progress
  - We now have three competitive algorithms:
    Primal & Dual Simplex, Barrier (interior points)

My opinion on Linear Programming

- From an commercial/economic point of view:

Linear programming is the most important development of mathematics in the 20th century.

Martin Grötschel

Contents

Martin Grötschel

Application of LP & MIP - I

- **Transportation-airlines**
  - Fleet assignment
  - Crew scheduling
  - Ground personnel scheduling
  - Yield management
  - Fuel allocation
  - Passenger mix
  - Booking control
  - Maintenance scheduling
  - Load balancing/freight packing
  - Airport traffic planning
  - Gate scheduling/assignment
  - Upset recover and management
- **Transportation-other**
  - Vehicle routing
  - Freight vehicle scheduling and assignment
  - Depot/warehouse location
  - Freight vehicle packing
  - Public transportation system operation
  - Rental car fleet management
- **Process industries**
  - Plant production scheduling and logistics
  - Capacity expansion planning
  - Pipeline transportation planning
  - Gasoline and chemical blending

Application of LP & MIP - II

- **Financial**
  - Portfolio selection and optimization
  - Cash management
  - Synthetic option development
  - Lease analysis
  - Capital budgeting and rationing
  - Bank financial planning
  - Accounting allocations
  - Securities industry surveillance
  - Audit staff planning
  - Assets/liabilities management
  - Unit costing
  - Financial valuation
  - Bank shift scheduling
  - Consumer credit delinquency management
  - Check clearing systems
  - Municipal bond bidding
  - Stock exchange operations
  - Debt financing
- **Manufacturing**
  - Product mix planning
  - Blending
  - Manufacturing scheduling
  - Inventory management
  - Job scheduling
  - Personnel scheduling
  - Maintenance scheduling and planning
  - Steel production scheduling
- **Coal Industry**
  - Coal sourcing/transportation logistics
  - Coal blending
  - Mining operations management
- **Forestry**
  - Forest land management
  - Forest valuation models
  - Planting and harvesting models

Martin Grötschel

Application of LP & MIP - III

- **Agriculture**
  - Production planning
  - Farm land management
  - Agricultural pricing models
  - Crop and product mix decision models
  - Product distribution
- **Public utilities and natural resources**
  - Electric power distribution
  - Power generator scheduling
  - Power tariff rate determination
  - Natural gas distribution planning
  - Natural gas pipeline transportation
  - Water resource management
  - Alternative water supply evaluation
  - Water reservoir management
  - Public water transportation models
  - Mining excavation models
- **Oil and gas exploration and production**
  - Oil and gas production scheduling
  - Natural gas transportation scheduling
- **Communications and computing**
  - Circuit board (VLSI) layout
  - Logical circuit design
  - Magnetic field design
  - Complex computer graphics
  - Curve fitting
  - Virtual reality systems
  - Computer system capacity planning
  - Office automation
  - Multiprocessor scheduling
  - Telecommunications scheduling
  - Telephone operator scheduling
  - Telemarketing site selection

Martin Grötschel

Application of LP & MIP - IV

- **Food processing**
  - Food blending
  - Recipe optimization
  - Food transportation logistics
  - Food manufacturing logistics and scheduling
- **Health care**
  - Hospital staff scheduling
  - Hospital layout
  - Health cost reimbursement
  - Ambulance scheduling
  - Radiation exposure models
- **Pulp and paper industry**
  - Inventory planning
  - Trim loss minimization
  - Waste water recycling
  - Transportation planning
- **Textile industry**
  - Pattern layout and cutting optimization
  - Production scheduling
- **Government and military**
  - Post office scheduling and planning
  - Military logistics
  - Target assignment
  - Missile detection
  - Manpower deployment
- **Miscellaneous applications**
  - Advertising mix/media scheduling
  - Pollution control models
  - Sales region definition
  - Sales force deployment

Martin Grötschel

Examples: ZIB & MATHEON-projects

http://www.zib.de/Optimization/Projects/index.en.html

Production and Logistics

In order to be able to organize the complex production processes of modern goods in a cost efficient way many logistic problems have to be solved. This refers both to the operational procedures inside a single factory as well as to the organization of manufacturing processes distributed over different places. At ZIB we have experience with several projects in the area of logistics. Due to their special requirements, some of these projects are listed under 'Online Optimization.'

[more]

Current Projects: MATHEON-B14: Comb-Log, MATHEON-D17: Chip Design Verification, RobotWelding
Past Projects: InterWarehouse

Online Optimization

In contrast to classical optimization, *Online Optimization* deals with situations where the input data of the problem is not known in advance. Therefore, decisions have to be made on the basis of incomplete knowledge. The decisions of the online algorithm are sometimes subject to further constraints. One important feature of practical online algorithms are *real-time requirements*: the algorithm has to present a feasible solution in a specified time frame.

[more]

Current Projects: KollmorgenElevator, ADAC-Dispatch, Online-Planning
Past Projects: MATHEON-C3: Online-Modular, MATHEON-C6: Online-Reopt, HTK-Survey, PrintSetup, Modelling, Shelf

Martin Grötschel

Examples: ZIB & MATHEON-projects

http://www.zib.de/Optimization/Projects/index.en.html

Telecommunication

Telecommunication technology is one of the key technologies in our society. Rapid progress in hardware development constantly enables new applications and services. The best example for this is UMTS. The tough competition in this sector forces network service providers to thouroughly control their investments and to implement potential savings. At the same time, the quality of service must be improved to get and keep new customers.

[more]

Current Projects: MATHEON-B3: MultiLayer, MATHEON-B4: UMTS, X-WiN, atesio, NEuK, ProSched, EIBONE, MLTN
Past Projects: FAP, G-WiN, OPADCO, OptNet, DynRoute, StatRoute, Momentum, STP, RoundRobin, BOSCH-PMP, B-WiN, Peakedness, DISCNET

Traffic

The application of mathematical optimization in public transport has been subject of successful research for many years, and more and more such methods are being put to practice. ZIB has been carrying out projects in this area for more than ten years by now, mainly in operative planning. e. g., vehicle or duty scheduling in public transport. During the last years, the application areas have widened.

[more]

Current Projects: MATHEON-B15: Service Design, Trassenbörse, Airline Crew Scheduling, Visualisation in Public Transit
Past Projects: MATHEON-B1: Strategic Planning, Telebus, BMBF-DS: Duty Scheduling, BMBF-VS: Vehicle Scheduling, BMBF-IS: Integrated Vehicle and Duty Scheduling

Martin Grötschel

Contents

Martin Grötschel

Linear Programming

$$\begin{aligned} &\min c^T x \\ &Ax = a \\ &Bx \le b \\ &x \ge 0 \end{aligned}$$

$$\begin{aligned} &\min c^T x \\ &Ax = a \\ &Bx \le b \\ &x \ge 0 \end{aligned}$$

$$\begin{aligned} &\min c^T x \\ &Ax = a \\ &x \ge 0 \end{aligned}$$

$$\begin{aligned} &\min c^T x \\ &Bx \le b \end{aligned}$$

Linear program in various forms.
They are all equivalent!
There are more versions!

Martin Grötschel

Optimizers‘ dream: Duality theorems

- Max-Flow Min-Cut Theorem
  The value of a maximal (s,t)-flow in a capacitated network is equal to the minimal capacity of an (s,t)-cut.
- The Farkas Lemma
- The Duality Theorem of Linear Programming

$$\max c^T x \quad = \quad \min y^T b$$
$$Ax \le b \qquad y^T A \ge c^T$$
$$x \ge 0 \qquad y \ge 0$$

Important theorems

- Complementary slackness theorems
- Redundancy characterizations
- Polyhedral theory

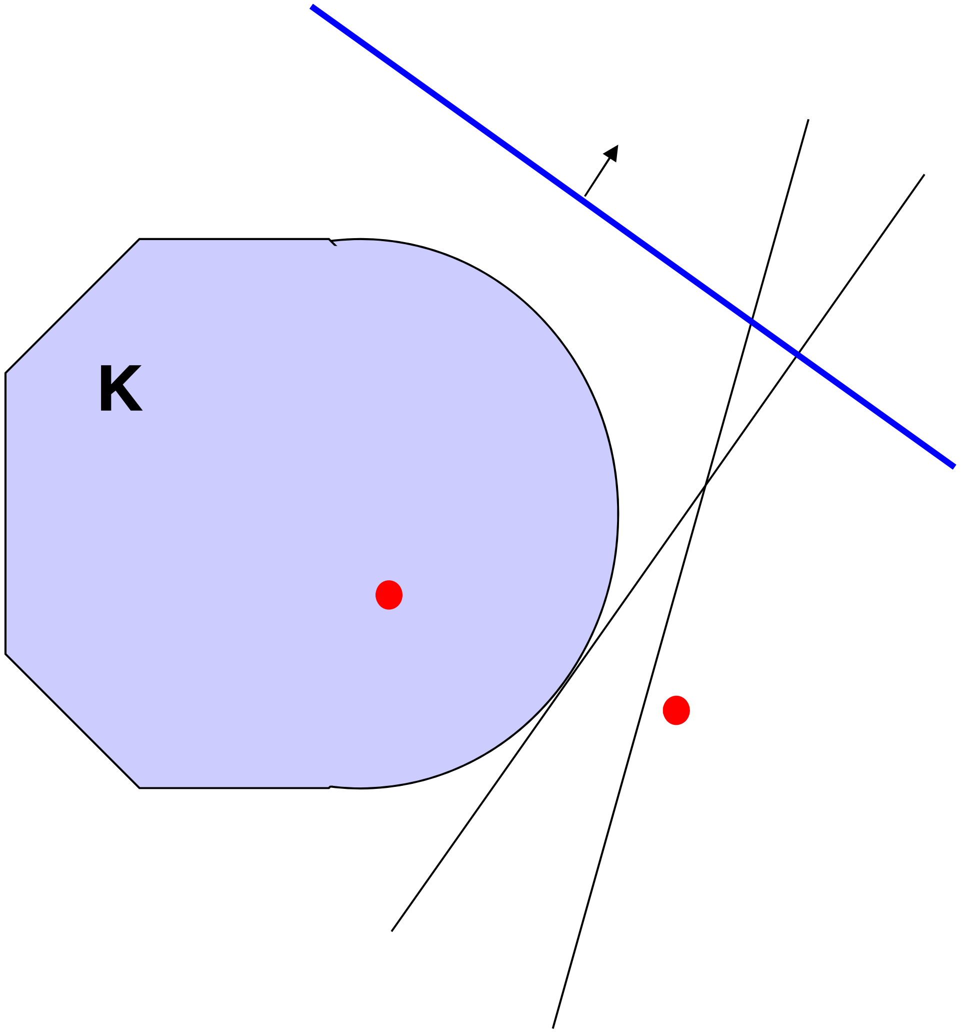
Martin
Grötschel

LP Solvability

- I assume that the audience is somewhat familiar with complexity theory:
  - Polynomial time solvability, solvability in strongly polynomial time
  - Classes: $\mathcal{P}$ and $\mathcal{NP}$, $\mathcal{NP}$-completeness , $\mathcal{NP}$-hardness , etc.
- Linear programs can be solved in polynomial time with
  - the Ellipsoid Method (Khachiyan, 1979)
  - Interior Points Methods (Karmarkar, 1984, and others)
- Open: Is there a strongly polynomial time algorithm for the solution of LPs?
- Certain variants of the Simplex Algorithm run – under certain conditions – in expected polynomial time (Borgwardt, 1977…)
- Open: Is there a polynomial time variant of the Simplex Algorithm?

Martin Grötschel

Separation



LP Solvability: Generalizations

Theorem (GLS 1981, 1988) (modulo technical details) : There exists a polynomial time algorithm to minimize convex functions (e.g., linear functions) over the elements of a class of convex bodies **K** (e. g. polyhedra) if and only if, there exists a polynomial time algorithm that decides, for any given point **x**, whether **x** is in **K**, and that, when **x** is not in **K**, finds a hyperplane that separates **x** from **K**.

Short version:

Optimization and Separation are polynomial-time equivalent.

Consequence: Theoretical Foundation of cutting plane algorithms.

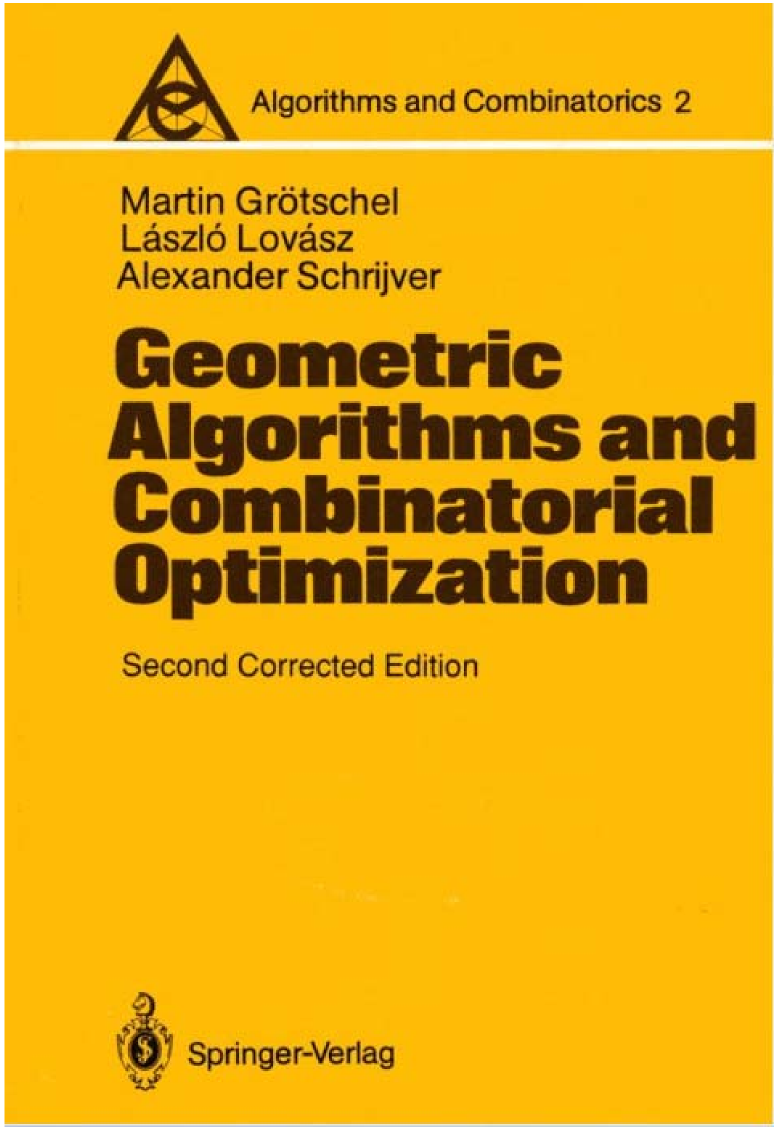
Particular special case: Polynomial time separation algorithm for the set of positive semi-definite matrices.

Consequences:

- Polynomial time algorithm for stable sets in perfect graphs.
- The beginning of semi-definite programming

Martin Grötschel

You can download this book from the publications list on my Web page.



http://www.zib.de/groetschel/pubnew/paper/groetschellovaszschrijver1988.pdf

Martin Grötschel

Contents

Martin Grötschel

Algorithms for nonlinear programming

- Iterative methods that solve the equation and inequality sytems representing the necessary local optimality conditions (e.g., KKT).

$$x_{i+1} = x_i + \lambda_i d_i$$
$$d_i \sim \text{"descent direction"}$$
$$\lambda_i \sim \text{"steplength"}$$

$$d_i = -\nabla f(x_i)$$ Steepest descent

$$d_i = -(H(x_i))^{-1} \nabla f(x_i)$$ Newton

(Quasi-Newton, conjugate-gradient-, SQP-, subgradient…methods)

- Sufficient conditions are rarely checked.

Martin Grötschel

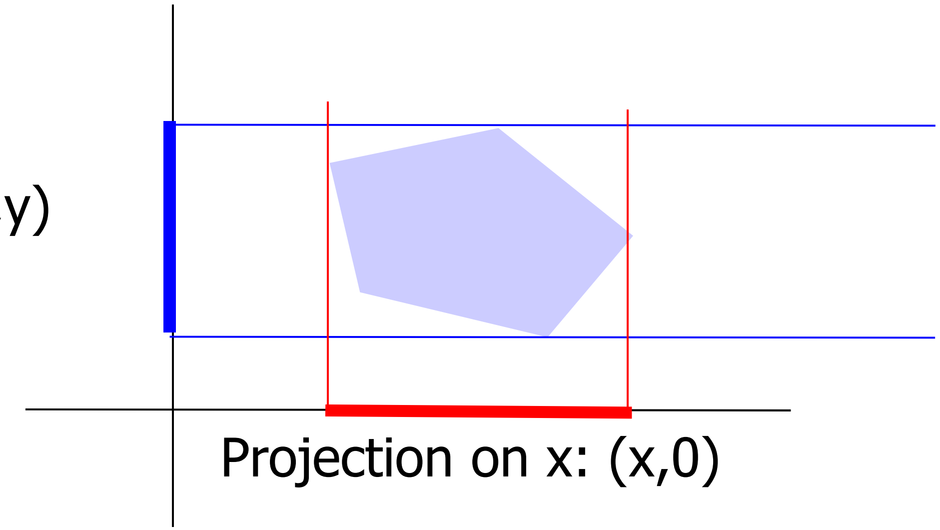
Contents

Martin Grötschel

Fourier-Motzkin Elimination

- Fourier, 1826/1827
- Motzkin, 1938
- **Method**: successive projection of a polyhedron in n-dimensional space into a vector space of dimension n-1 by elimination of one variable.

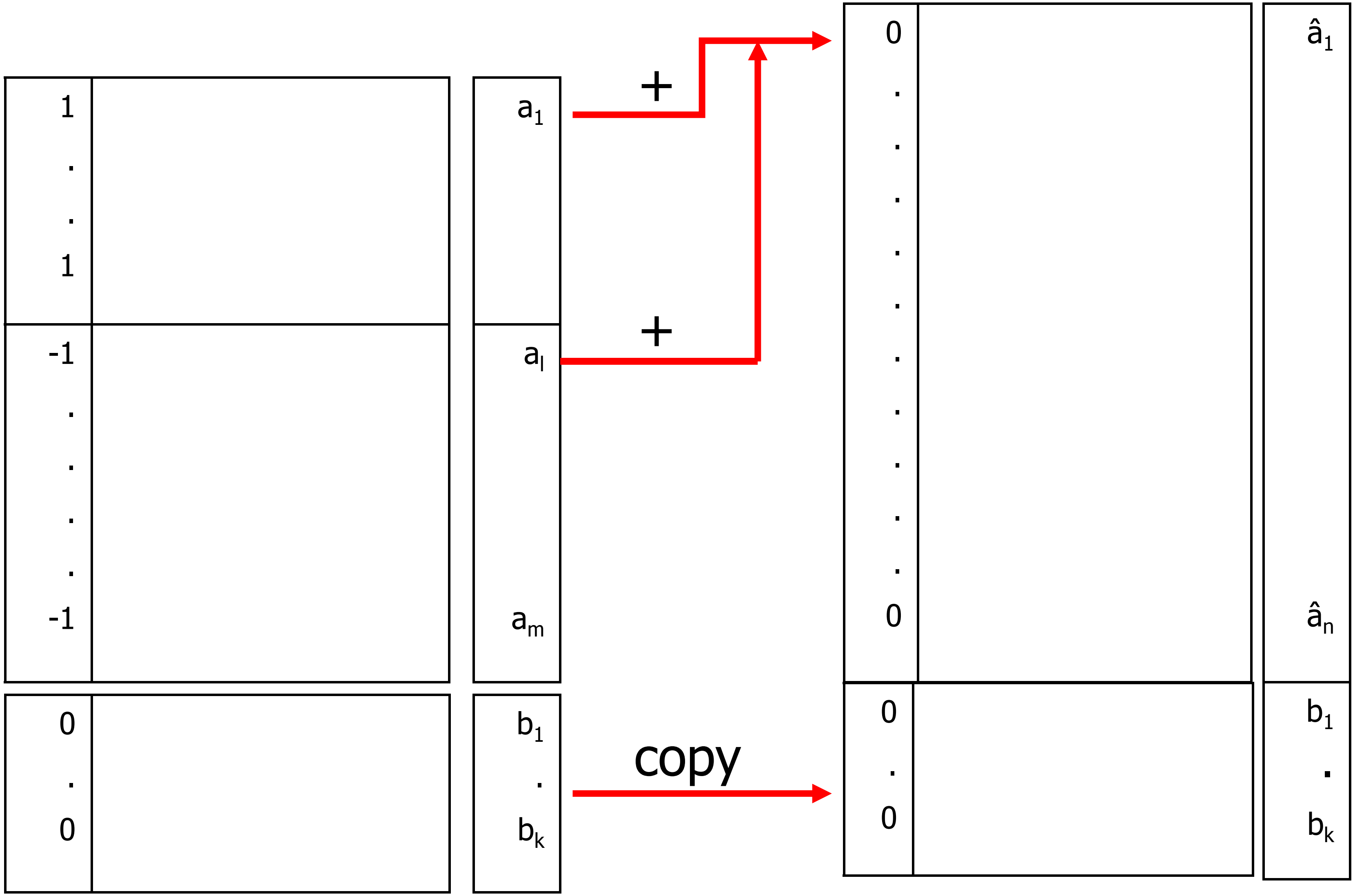
Projection on y: (0,y)



Projection on x: (x,0)

Martin Grötschel

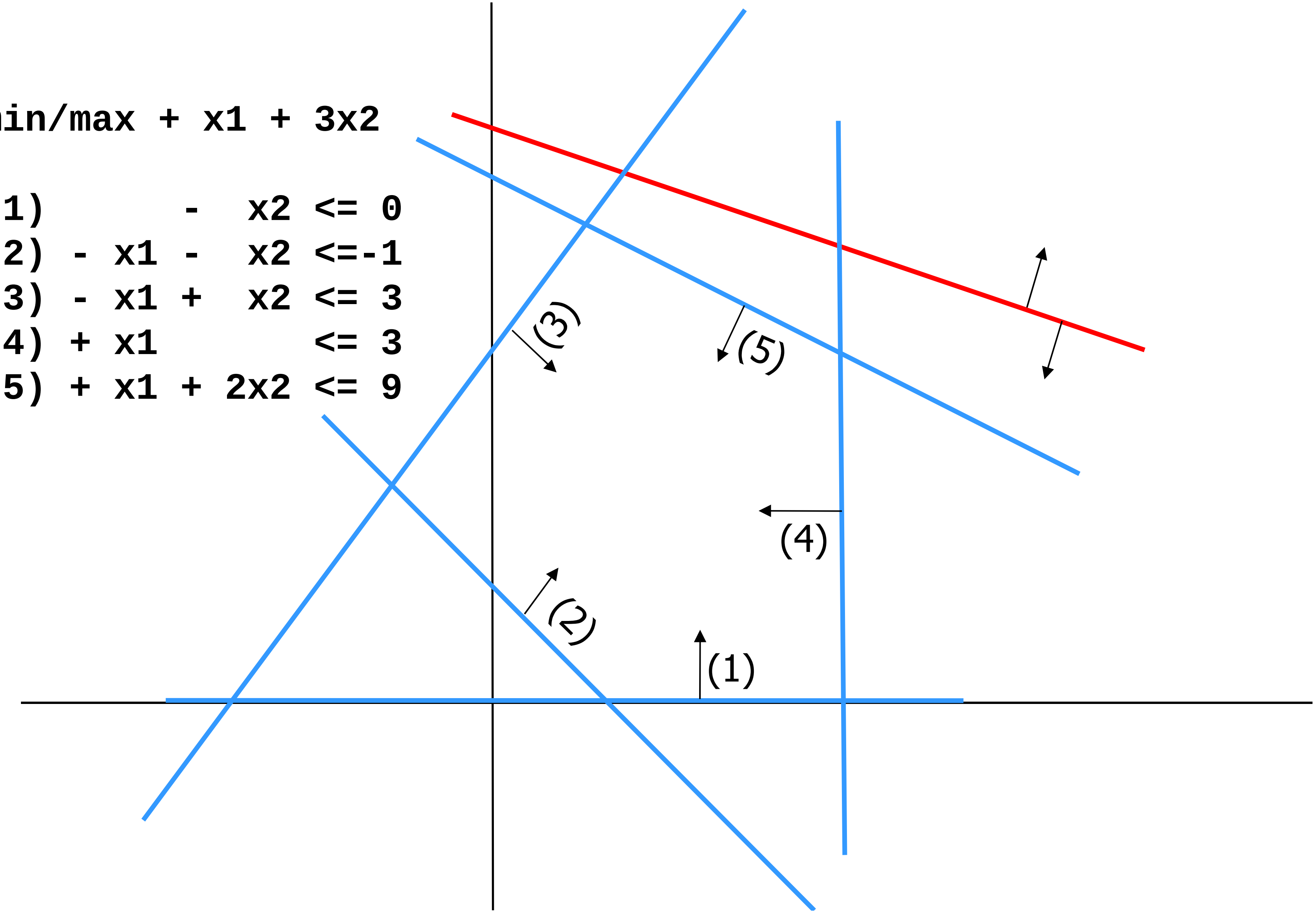
A Fourier-Motzkin step



Martin Grötschel

Fourier-Motzkin Elimination: an example

```
min/max + x1 + 3x2

(1)         -   x2 <= 0
(2) - x1 -   x2 <=-1
(3) - x1 +   x2 <= 3
(4) + x1          <= 3
(5) + x1 + 2x2 <= 9
```



Martin Grötschel

Fourier-Motzkin Elimination: an example, call of PORTA (Polymake)

```
min/max + x1 + 3x2

(1)       -   x2 <= 0
(2) - x1 -   x2 <=-1
(3) - x1 +   x2 <= 3
(4) + x1         <= 3
(5) + x1 + 2x2 <= 9
```

```
DIM = 3

INEQUALITIES_SECTION

(1)       -   x2        <= 0
(2) - x1 -   x2        <=-1
(3) - x1 +   x2        <= 3
(4) + x1               <= 3
(5) + x1 + 2x2         <= 9
(6) + x1 + 3x2 - x3 <= 0
(7) - x1 - 3x2 + x3 <= 0


ELIMINATION_ORDER
1 0 0
```

Martin Grötschel

Fourier-Motzkin Elimination: an example

```
DIM = 3

INEQUALITIES_SECTION

(1)   (1) - x2        <=  0
(2,4) (2) - x2        <=  2
(2,5) (3) + x2        <=  8
(2,6) (4) +2x2 - x3 <= -1
(3,4) (5) + x2        <=  6
(3,5) (6) + x2        <=  4
(3,6) (7) +4x2 - x3 <=  3
(7,4) (8) -3x2 + x3 <=  3
(7,5) (9) - x2 + x3 <=  9
(7,6)
```

```
DIM = 3

INEQUALITIES_SECTION

(1)        -  x2        <= 0
(2) - x1 -  x2        <=-1
(3) - x1 +  x2        <= 3
(4) + x1                <= 3
(5) + x1 + 2x2        <= 9
(6) + x1 + 3x2 - x3 <= 0
(7) - x1 - 3x2 + x3 <= 0

ELIMINATION_ORDER
1 0 0
```

Martin Grötschel

Fourier-Motzkin Elimination: an example

```
DIM = 3

INEQUALITIES_SECTION

(1)   (1) - x2        <=  0
(2,4) (2) - x2        <=  2
(2,5) (3) + x2        <=  8
(2,6) (4) +2x2 - x3 <= -1
(3,4) (5) + x2        <=  6
(3,5) (6) + x2        <=  4
(3,6) (7) +4x2 - x3 <=  3
(7,4) (8) -3x2 + x3 <=  3
(7,5) (9) - x2 + x3 <=  9
(7,6)

ELIMINATION_ORDER
0 1 0
```

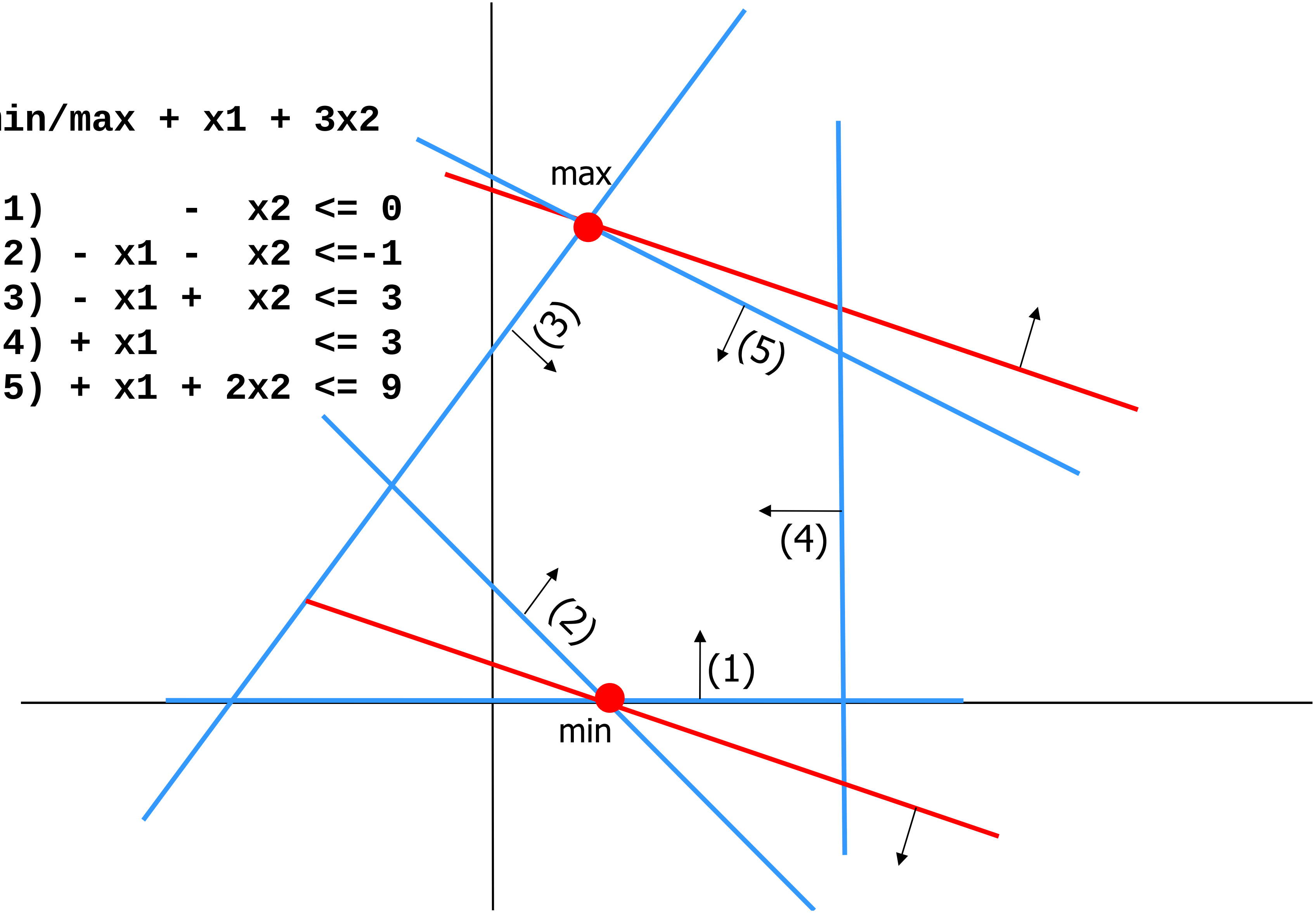
```
(1,4) (  1) -x3 <=  -1
(1,7) (  2) -x3 <=   3
(2,4) (  3) -x3 <=   3
(2,7) (  4) -x3 <=  11
(8,3) (  5) +x3 <=  27
(8,4) (  6) -x3 <=   3
(8,5) (  7) +x3 <=  21
(8,6) (  8) +x3 <=  15
(8,7) (  9) +x3 <=  21
(9,3) ( 10) +x3 <=  17
(9,4) ( 11) +x3 <=  17
(9,5) ( 12) +x3 <=  15
(9,6) ( 13) +x3 <=  13
(9,7) ( 14)+3x3 <=  39
```

```
min = 1 <= x3 <= 13 = max

x1 = 1          x1 = 1
x2 = 0          x2 = 4
```

Martin Grötschel

Fourier-Motzkin Elimination: an example

```
min/max + x1 + 3x2

(1)        -   x2 <= 0
(2) - x1 -   x2 <=-1
(3) - x1 +   x2 <= 3
(4) + x1         <= 3
(5) + x1 + 2x2 <= 9
```



Martin Grötschel

Fourier-Motzkin Elimination

- FME is a wonderful constructive proof method.
- Elimination of all variables of a given inequality system directly yields the Farkas Lemma:

$Ax \leq b$ has a solution or $y^T A = 0, y^T b < 0$ has a solution but not both.

- FME is computationally lousy.

Martin Grötschel

Contents

Martin Grötschel

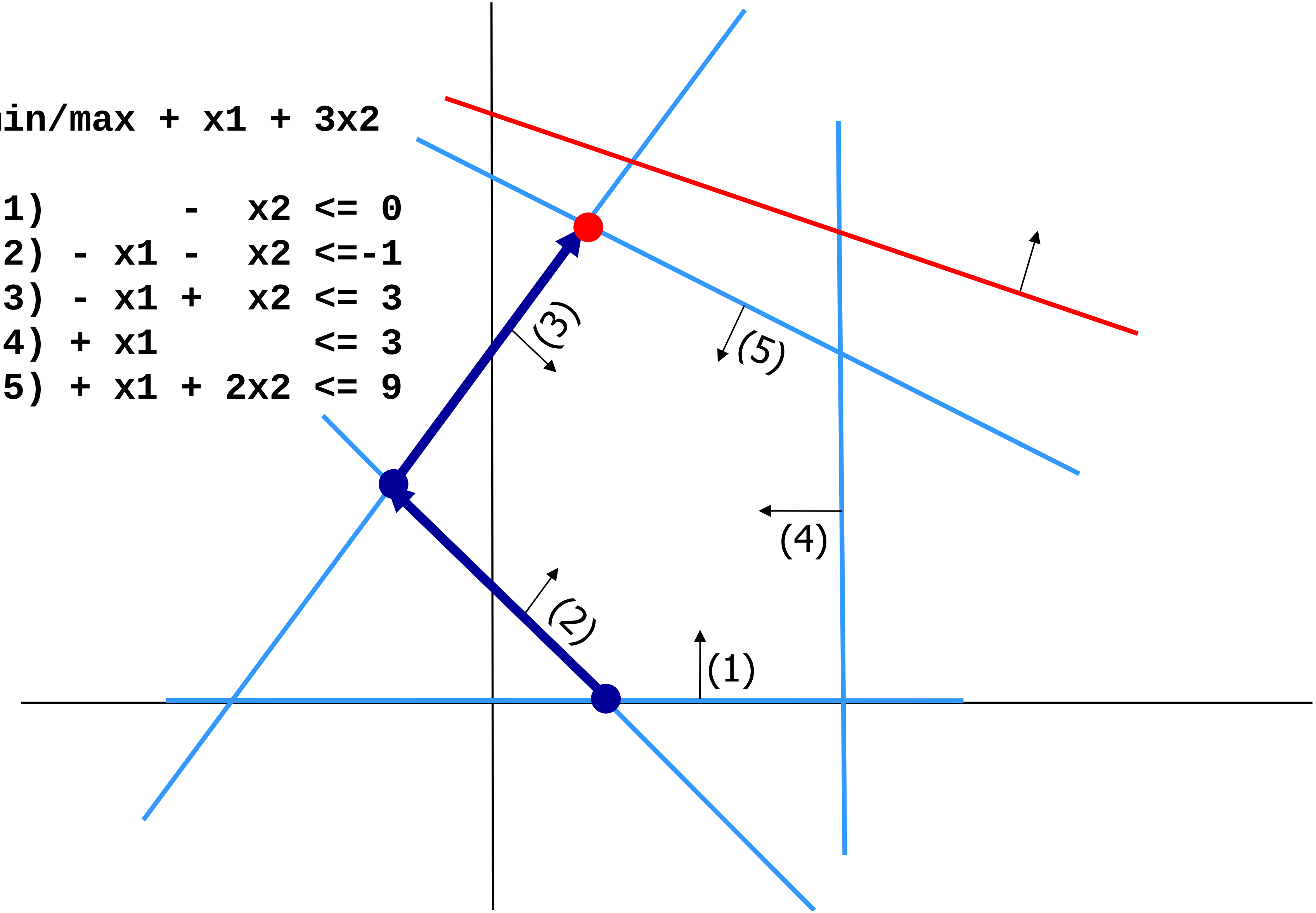
The Simplex Method

- Dantzig, 1947: primal Simplex Method
- Dantzig, 1953: revised Simplex Method
- Lemke, 1954; Beale, 1954: dual Simplex Method
- ….
- **Underlying Idea**: Find a vertex of the set of feasible LP solutions (polyhedron) and move to a better neighbouring vertex, if possible (Fourier‵s idea 1826/27).

Martin Grötschel

The Simplex Method: an example

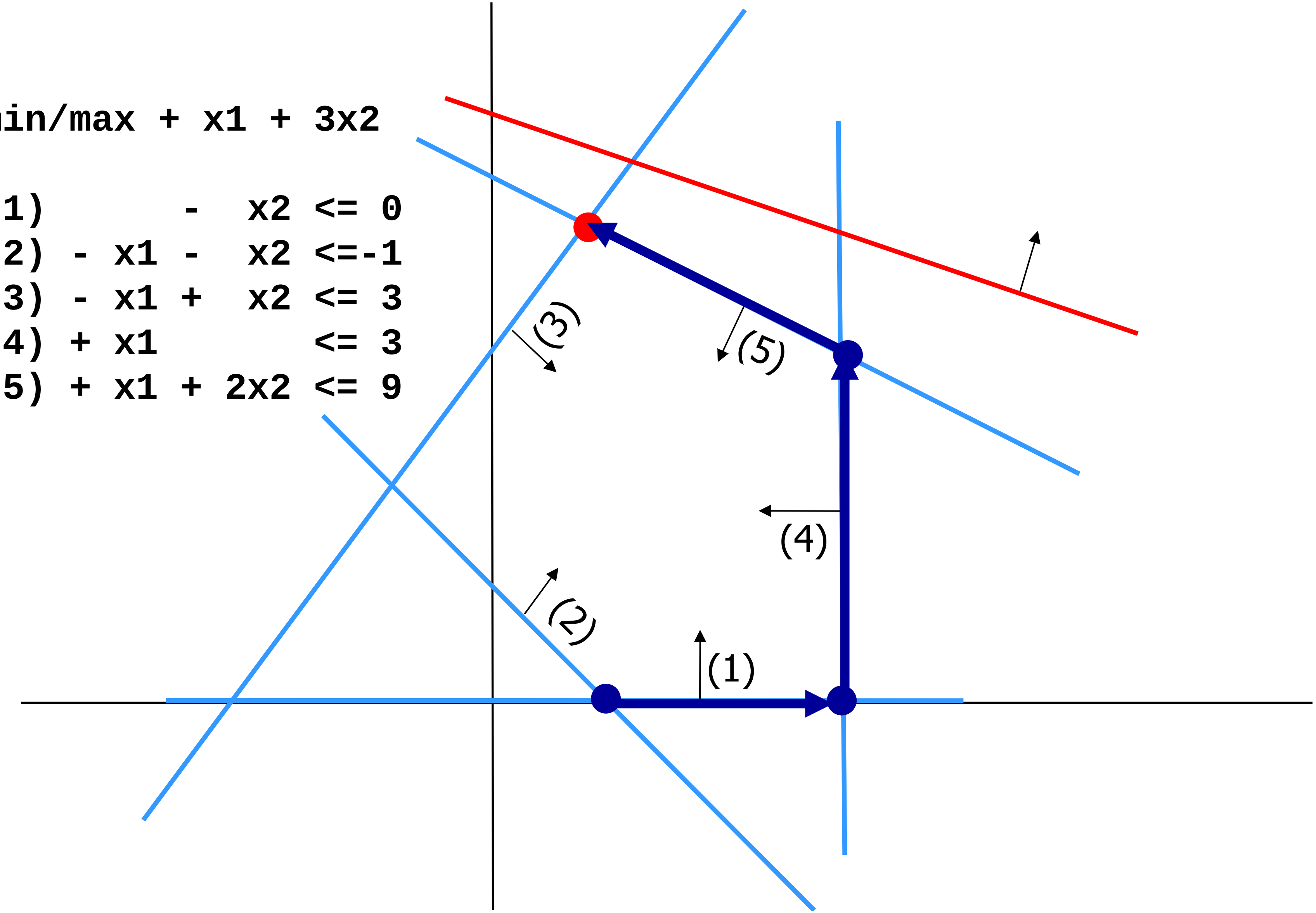
```
min/max + x1 + 3x2

(1)        -   x2 <= 0
(2) - x1 -   x2 <=-1
(3) - x1 +   x2 <= 3
(4) + x1        <= 3
(5) + x1 + 2x2 <= 9
```



Martin Grötschel

The Simplex Method: an example

```
min/max + x1 + 3x2

(1)        -   x2 <= 0
(2) - x1 -   x2 <=-1
(3) - x1 +   x2 <= 3
(4) + x1        <= 3
(5) + x1 + 2x2 <= 9
```



Martin Grötschel

Hirsch Conjecture

If P is a polytope of dimension n with m facets then every vertex of P can be reached from any other vertex of P on a path of length at most m-n.

In the example before: m=5, n=2 and m-n=3, conjecture is true.

At present, not even a polynomial bound on the path length is known.

Best upper bound: Kalai, Kleitman (1992): The diameter of the graph of an n-dimensional polyhedron with m facets is at most $m^{(\log n+1)}$.

Lower bound: Holt, Klee (1997): at least m-n (m, n large enough).

Martin Grötschel

arXiv:1006.2814v1 [math.CO] 14 Jun 2010

A counterexample to the Hirsch conjecture

Francisco Santos*

To Victor L. Klee (1925–2007), in memoriam†

Abstract

The Hirsch Conjecture (1957) stated that the graph of a d-dimensional polytope with n facets cannot have (combinatorial) diameter greater than $n-d$. That is, that any two vertices of the polytope can be connected to each other by a path of at most $n-d$ edges.

This paper presents the first counterexample to the conjecture. Our polytope has dimension 43 and 86 facets. It is obtained from a 5-dimensional polytope with 48 facets which violates a certain generalization of the d-step conjecture of Klee and Walkup.

To appear in STOC 2011 proceedings, June 2011

Subexponential lower bounds for
randomized pivoting rules for solving linear programs

Oliver Friedmann * *Thomas Dueholm Hansen* † *Uri Zwick* ‡

Abstract

The *simplex* algorithm is among the most widely used algorithms for solving *linear programs* in practice. Most *deterministic* pivoting rules are known, however, to need an exponential number of steps to solve some linear programs. No non-polynomial lower bounds were known, prior to this work, for *randomized* pivoting rules. We provide the first *subexponential* (i.e., of the form $2^{\Omega(n^\alpha)}$, for some $\alpha > 0$) lower bounds for the two most natural, and most studied, randomized pivoting rules suggested to date.

The first randomized pivoting rule we consider is RANDOM-EDGE, which among all improving pivoting steps (or *edges*) from the current basic feasible solution (or *vertex*) chooses one uniformly at random. The second randomized pivoting rule we consider is RANDOM-FACET, a more complicated randomized pivoting rule suggested by Matoušek, Sharir and Welzl [MSW96]. Our lower bound for the RANDOM-FACET pivoting rule essentially matches the subexponential upper bound of Matoušek *et al.* [MSW96]. Lower bounds for RANDOM-EDGE and RANDOM-FACET were known before only in *abstract* settings, and not for concrete linear programs.

Our lower bounds are obtained by utilizing connections between pivoting steps performed by simplex-based algorithms and *improving switches* performed by *policy iteration* algorithms for 1-player and 2-player games. We start by building 2-player *parity games* (PGs) on which suitable randomized policy iteration algorithms perform a subexponential number of iterations. We then transform these 2-player games into 1-player *Markov Decision Processes* (MDPs) which correspond almost immediately to concrete linear programs.

Martin Grötschel

Computationally important idea of the Simplex Method

Let a (m,n)-Matrix A with full row rank m, an m-vector b and an n-vector c with m<n be given. For every vertex y of the polyhedron of feasible solutions of the LP,

$$\max c^T x$$
$$Ax = b$$
$$x \geq 0$$

$$A = \left[\begin{array}{c|c} \quad B \quad & \qquad\qquad N \qquad\qquad \end{array}\right]$$

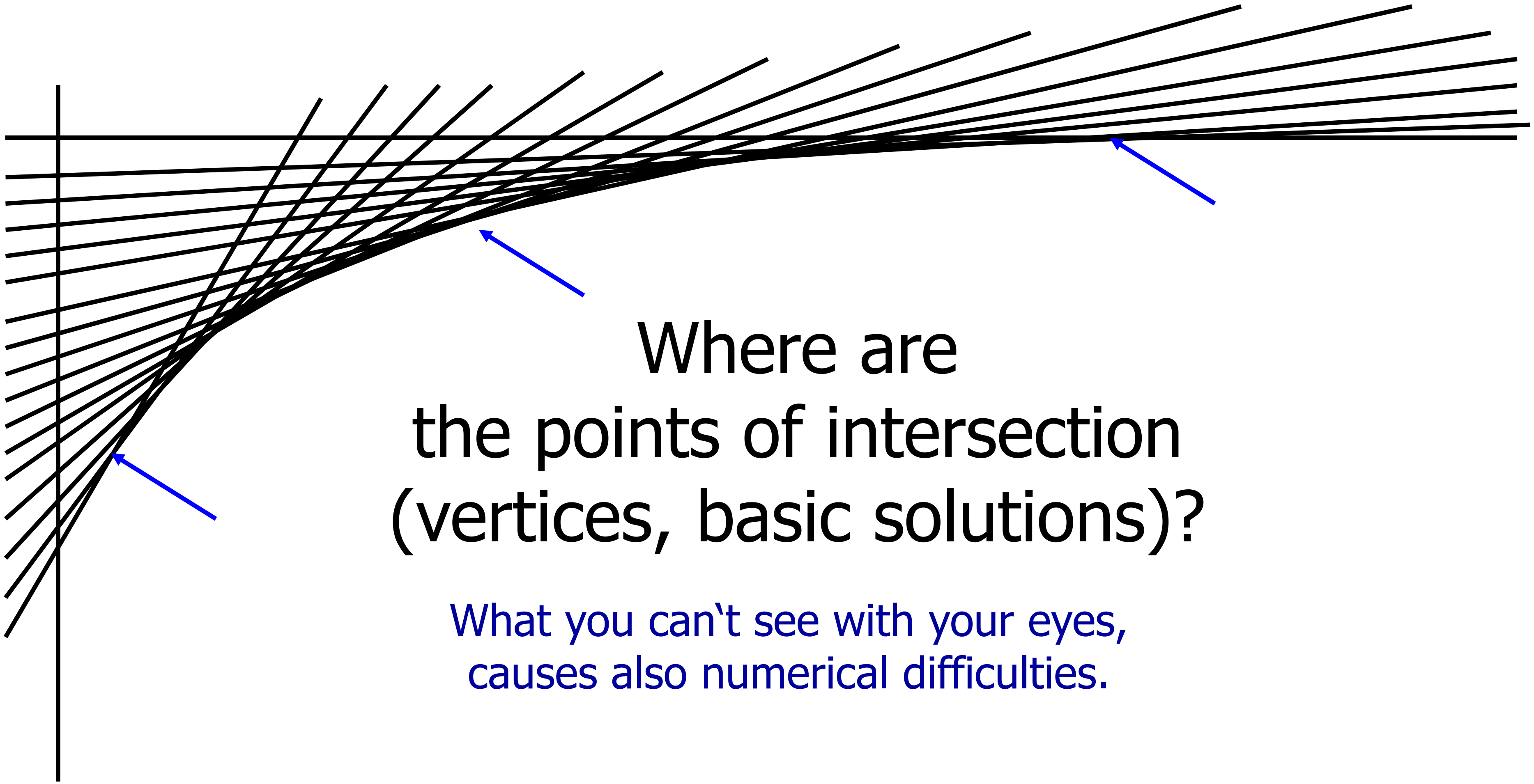
there is a non-singular (m,m)-submatrix B (called basis) of A representing the vertex y (basic solution) as follows

$$y_B = B^{-1}b, \quad y_N = 0$$

Many computational consequences:
Update-formulas, reduced cost calculations, number of non-zeros of a vertex,…

Martin Grötschel

Numerical trouble often has geometric reasons



Where are
the points of intersection
(vertices, basic solutions)?

What you can't see with your eyes,
causes also numerical difficulties.

Dual Simplex Method

- The Dual Simplex Method is the (Primal) Simplex Method applied to the dual of a given linear program.

Surprise in the mid-nineties:

- The Dual Simplex Method is faster than the Primal in practice.
  One key: Goldfarb’s steepest edge pivoting rule!
- A wonderful observation for the cutting plane methods of integer programming!

Martin Grötschel

Contents

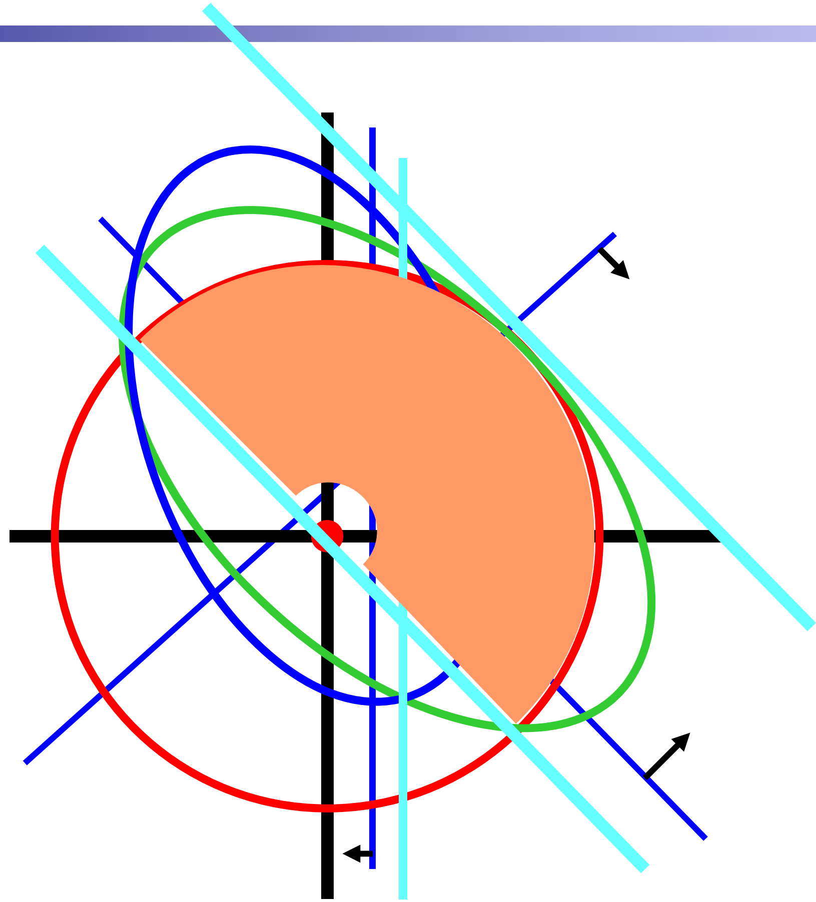
Martin Grötschel

The Ellipsoid Method

- Shor, 1970 - 1979
- Yudin & Nemirovskii, 1976
- Khachiyan, 1979
- M. Grötschel, L. Lovász, A. Schrijver,
  Geometric Algorithms and Combinatorial Optimization
  Algorithms and Combinatorics 2, Springer, 1988

Martin Grötschel

The Ellipsoid Method: an example



Martin
Grötschel

$$k := 0,$$
$$N := 2n((2n+1)\langle C\rangle + n\langle d\rangle - n^3)$$
$$A_0 := R^2 I \quad \text{with } R := \sqrt{n}\, 2^{\langle C,d\rangle - n^2}$$

$$P := \{x \mid Cx \le d\}$$

Initialization

$$a_0 := 0$$

If $k = N$, *STOP!* (Declare P empty.) — Stopping criterion

If $a_k \in P$, *STOP!* (A feasible solution is found.) — Feasibility check

If $a_k \notin P$, *then choose an inequality, say* $c^T x \le \gamma$, *of the system* $Cx \le d$ *that is violated by* a_k. — Cutting plane choice

$$b := \frac{1}{\sqrt{c^T A_k c}} A_k c$$

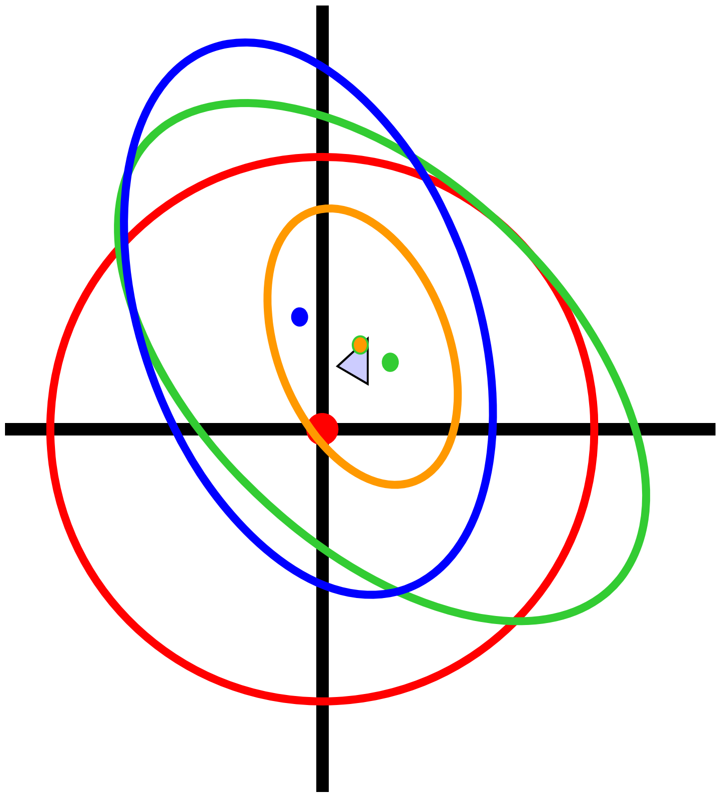
$$a_{k+1} := a_k - \frac{1}{n+1} b$$

Update

$$A_{k+1} := \frac{n^2}{n^2-1}\left(A_k - \frac{2}{n+1} b b^T\right)$$

The Ellipsoid Method

Ellipsoid Method



a(0)

a(1)

a(2)

a(7)

feasible solution found

Martin Grötschel

Доклады Академии наук СССР
1979. Том 244, № 5

УДК 519.95 *МАТЕМАТИКА*

Л. Г. ХАЧИЯН

ПОЛИНОМИАЛЬНЫЙ АЛГОРИТМ В ЛИНЕЙНОМ ПРОГРАММИРОВАНИИ

(Представлено академиком А. А. Дородницыным 4 X 1978)

Рассмотрим систему из $m \geqslant 2$ линейных неравенств относительно $n \geqslant 2$ вещественных переменных $x_1, \ldots, x_j, \ldots, x_n$

$$a_{i1}x_1 + \ldots + a_{in}x_n \leqslant b_i, \quad i = 1, 2, \ldots, m, \tag{1}$$

с целыми коэффициентами a_{ij}, b_i. Пусть

$$L = \left[\sum_{i,j=1}^{m,n} \log_2(|a_{ij}|+1) + \sum_{i=1}^{m} \log_2(|b_i|+1) + \log_2 nm \right] + 1 \tag{2}$$

есть длина входа системы, т. е. число символов 0 и 1, необходимых для записи (1) в двоичной системе счисления.

Martin Grötschel

There was an Oberwolfach meeting on Mathematical Programming in 9/1979 where I learned about Khachiyan's work

MATHEMATICAL PROGRAMMING STUDY 14

A PUBLICATION OF
THE MATHEMATICAL PROGRAMMING SOCIETY

Mathematical Programming at Oberwolfach

Edited by H. KÖNIG, B. KORTE and K. RITTER

January (1981)

Martin Grötschel

Mathematical Programming Study 14 (1981) 61–68.
North-Holland Publishing Company

KHACHIYAN'S ALGORITHM FOR LINEAR PROGRAMMING*

Peter GÁCS and Laszlo LOVÁSZ

Computer Science Department, Stanford University, Stanford, CA 94305, U.S.A.

Received 10 October 1979

L.G. Khachiyan's algorithm to check the solvability of a system of linear inequalities with integral coefficients is described. The running time of the algorithm is polynomial in the number of digits of the coefficients. It can be applied to solve linear programs in polynomial time.

Key Words: Linear Programming, Inequalities, Complexity, Polynomial Algorithms.

0. Introduction

L.G. Khachiyan [1, cf. also 2, 3] published a polynomial-bounded algorithm to solve linear programming. These are some notes on this paper. We have ignored his considerations which concern the precision of real computations in order to make the underlying idea clearer; on the other hand, proofs which are missing from his paper are given in Section 2. Let

$$a_i x < b_i \quad (i = 1, \dots, m,\ a_i \in Z^n,\ b_i \in Z) \tag{1}$$

be a system of *strict* linear inequalities with integral coefficients. We present an algorithm which decides whether or not (1) is solvable, and yields a solution if it is. Define

$$L = \sum_{i,j} \log(|a_{ij}| + 1) + \sum_i \log(|b_i| + 1) + \log nm + 1.$$

L is a lower bound on the space needed to state the problem.

MATHEMATIK

Schnelles Öl

Für Aufgaben mit schwer wägbaren Größen fand ein junger Sowjet-Forscher einen eleganten Lösungsweg. Kapitalistische Unternehmen wie auch die sozialistische Planwirtschaft könnten davon profitieren.

Seine Mutter ist Rentnerin. So erledigt der Junggeselle Leonid Genrichowitsch Kachijan, 27, der Bequemlichkeit halber das Tagewerk öfter mal zu Hause.

An seinem Arbeitsplatz in einem alten Moskauer Backsteinbau, dem Computer-Zentrum der Sowjetischen Akademie der Wissenschaften, erscheint der dunkelhaarige Armenier im Pullover.

Sein einziges Hobby — Karate — gab Kachijan des Studiums wegen auf. Vor fünf Jahren, mit 22, machte er Examen, letztes Jahr seinen Doktor; und nun wurde der Mathematik-Theoretiker, in Fachkreisen jedenfalls, auf einen Schlag weltberühmt.

Der Amerikaner George B. Dantzig etwa, der sich mit dem gleichen Aufgabengebiet beschäftigt wie Kachijan, wird geradezu „von Anrufen überschwemmt"; er soll die Bedeutung der Entdeckung seines sowjetischen Kollegen erklären. „Wirklich jedes Ministerium" der US-Regierung, berichtet der Stanford-Professor, habe sich schon danach erkundigt.

Die Leistung des jungen Moskauers:

hem Rechenaufwand von Computern.

Das verhältnismäßig junge Gebiet ist überdies, wie der „dtv-Atlas zur Mathematik" erläutert, „außerordentlich praxisnah": Die sogenannte lineare Optimierung wird zunehmend wichtig für weltweit operierende Unternehmen und für die Wirtschaftslenkung in den sozialistischen Staaten.

Im wesentlichen geht es darum, wie mit einer Vielzahl veränderlicher Größen innerhalb fester Rahmenbedingungen das günstigste Ergebnis zu erzielen ist. Wo aber früher Spürsinn des Managers oder Voraussicht des Planungs-

funktionärs ausreichen mußten (oder versagten), wird neuerdings scharf kalkuliert.

Im einfachsten Fall — zwei veränderliche Größen (Variable) — ginge es beispielsweise um die Frage, wie viele Farb- und wie viele Schwarzweißfernseher ein Fabrikant bauen müßte, um den größten Gewinn zu erzielen. Seine Rahmenbedingungen: Am Schwarzweißfernseher verdient er 150, am Farbfernseher 450 Mark; er kann wöchentlich höchstens 120 Schwarzweißgeräte, höchstens 70 Farbgeräte und insgesamt nicht mehr als 140 Fernseher herstellen lassen; außerdem sind in die Schwarzweißgeräte je ein, in die Farbgeräte je zwei Aggregate einzubauen, von denen wöchentlich nur 180 Stück zur Verfügung stehen.

Die optimale Lösung (40 Schwarzweiß- und 70 Farbgeräte, 37 500 Mark Gewinn) läßt sich noch durch graphische Darstellung finden. Bei drei Va-

Wahrscheinlich wird sich durch Kachijans Erkenntnis die dagegen doch recht umständliche Simplex-Methode weithin ersetzen lassen. Und womöglich erweist sie sich auch bei Problemen nichtlinearer Optimierung als tauglich, bei denen die Simplex-Methode versagt.

Seltsam mutet an, daß Kachijans Theorie, die nun alle Experten fasziniert, nur durch Flüsterpropaganda im Westen bekannt wurde. Veröffentlicht hatte er sie bereits vor einem Jahr.

Erst im Mai aber machten polnische Kollegen den Kölner Mathematik-Professor Rainer Burkard während einer Tagung auf eine Kurzfassung dieser Arbeit im Sowjet-Journal „Doklady" (Vorträge) aufmerksam. Burkard wiederum gab die Information an amerikanische Kollegen weiter, die freilich erst einen mathematikverständigen Übersetzer suchen mußten.

Dann allerdings machte der bis dahin unbekannte Name Kachijan die Runde und Furore. Mit seinem Lösungsschema gelang es etwa dem Ungarn László Lovász, zu Gast an der Stanford University, für Aufgaben mit sechs Ungleichungen und sechs Unbekannten auf Anhieb mit einem programmierbaren Taschenrechner das optimale Ergebnis zu finden. ◆

DER SPIEGEL, Nr. 50/1979

May 1980
New York Times, Nov. 7, 1979

A Soviet Discovery Rocks World of Mathematics
By MALCOLM W. BROWNE
New York Times (1857-Current file); Nov 7, 1979; ProQuest Historical Newspapers The New York Times (1851 - 2003)
pg. A1

A Soviet Discovery Rocks World of Mathematics

By MALCOLM W. BROWNE

A surprise discovery by an obscure Soviet mathematician has rocked the world of mathematics and computer analysis, and experts have begun exploring its practical applications.

Mathematicians describe the discovery by L.G. Khachian as a method by which computers can find guaranteed solutions to a class of very difficult problems that have hitherto been tackled on a kind of hit-or-miss basis.

Apart from its profound theoretical interest, the discovery may be applicable in weather prediction, complicated industrial processes, petroleum refining, the scheduling of workers at large factories, secret codes and many other things.

"I have been deluged with calls from virtually every department of government for an interpretation of the significance of this," a leading expert on computer methods, Dr. George B. Dantzig of Stanford University, said in an interview.

The solution of mathematical problems by computer must be broken down into a series of steps. One class of problem sometimes involves so many steps that it could take billions of years to compute.

The Russian discovery offers a way by which the number of steps in a solution can be dramatically reduced. It also offers the mathematician a way of learning quickly whether a problem has a solution or not, without having to complete the entire immense computation that may be required.

According to the American journal Sci-

Continued on Page A20, Column 3

May 1980
National Association of Science Writers

NEWSLETTER

NATIONAL ASSOCIATION OF SCIENCE WRITERS

Volume 28, Number 2 May 1980

Science Writers Rock World of Mathematics: Tales of the Traveling Salesman Problem

by Jonathan Weiner

Echoes of Sputnik. An obscure young Russian mathematician solves a key problem in linear programming, and American defense experts wring their hands worrying about its applications to secret codes, weather forecasting, and Kremlin-only-knows what else.

It was a pretty good story, as mathematics news goes, and it wound up on page one of *The New York Times* last November 7. It was run by *The Times* news service, and it was picked up far and wide as a nifty science novelty item. It had all the elements of a spy novel: the cold war, a valuable scientific formula, and sexual innuendo in the form of a traveling salesman. Who could ignore it?

National Association of Science Writers

Trouble was, it was wrong. The *Times* story said the Soviet mathematician had cracked the secret of the Traveling Salesman Problem — one of the most important in computer science. In fact, he didn't do it, and he never claimed he did. He didn't come close.

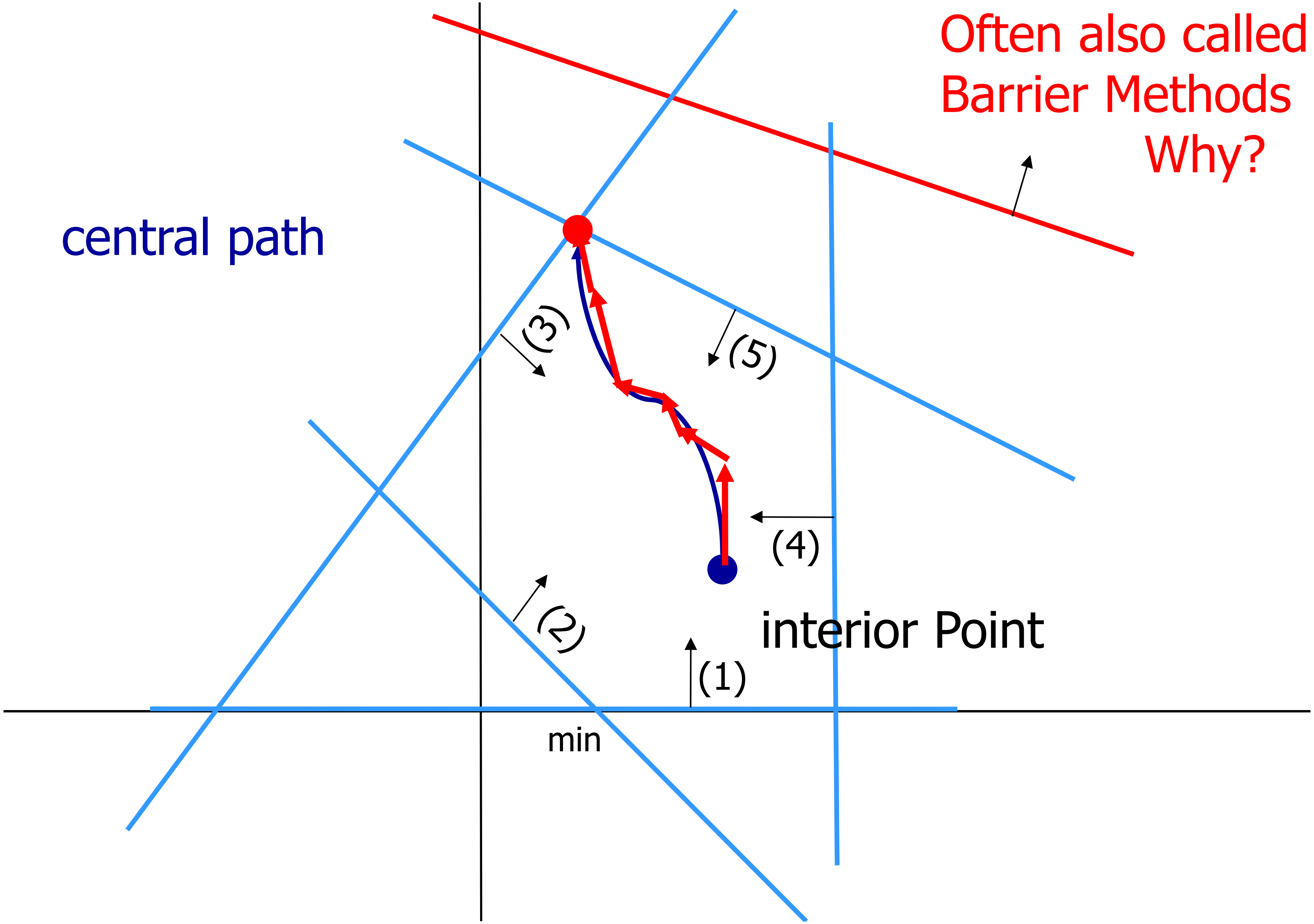
Bad enough. Mistakes happen. Everyone makes them, and even *The Times* doesn't claim it's immune. In this case, however, *The Times* enraged American computer scientists — who had to convince congressmen and others there was no 'salesman gap' — by seemingly refusing for months to admit it was wrong.

Martin Grötschel

Contents

Martin Grötschel

Interior-Point Methods: an example



Martin Grötschel

The Karmarkar Algorithm

(13.25) Der Karmarkar-Algorithmus.

Input: $A \in \mathbb{Q}^{(m,n)}$ *und* $c \in \mathbb{Q}^n$. *Zusätzlich wird vorausgesetzt, dass* $\frac{1}{n}A\mathbb{1} = 0$ *und* $c^T\mathbb{1} > 0$ *gilt.*

Output: *Ein Vektor* x *mit* $Ax = 0$, $\mathbb{1}^T x = 1$, $x \geq 0$ *und* $c^T x \leq 0$ *oder die Feststellung, dass kein derartiger Vektor existiert.*

(1) **Initialisierung.** *Setze*

$$\begin{aligned} x^0 &:= \tfrac{1}{n}\mathbb{1} \\ k &:= 0 \\ N &:= 3n(\langle A\rangle + 2\langle c\rangle - n) \end{aligned}$$

(2) **Abbruchkriterium.**

(2.a) Gilt $k = N$, *dann hat* $Ax = 0$, $\mathbb{1}^T x = 1$, $x \geq 0$, $c^T x \leq 0$ *keine Lösung, STOP!*

(2.b) Gilt $c^T x^k \leq 2^{-\langle A\rangle - \langle c\rangle}$, *dann ist eine Lösung gefunden. Falls* $c^T x^k \leq 0$, *dann ist* x^k *eine Lösung, andernfalls kann wie bei der Ellipsoidmethode (Satz (12.34)) aus* x^k *ein Vektor* $\overline{x}$ *konstruiert werden mit* $c^T\overline{x} \leq 0$ $A\overline{x} = 0$, $\mathbb{1}^T\overline{x} = 1$, $\overline{x} \geq 0$, *STOP!*

Update.

(3) (3.a) $D := \operatorname{diag}(x^k)$

(3.b) $\overline{\overline{c}} := (I - DA^T(AD^2A^T)^{-1}AD - \frac{1}{n}\mathbb{1}\mathbb{1}^T)Dc$

(3.c) $y^{k+1} := \frac{1}{n}\mathbb{1} - \frac{1}{2}\frac{1}{\sqrt{n(n-1)}}\frac{1}{\|\overline{\overline{c}}\|}\overline{\overline{c}}$

(3.d) $x^{k+1} := \frac{1}{\mathbb{1}^T D y^{k+1}} D y^{k+1}$

(3.e) $k := k + 1$

Gehe zu (2).

Martin Grötschel

Breakthrough in Problem Solving

By JAMES GLEICK

A 28-year-old mathematician at A.T.&T. Bell Laboratories has made a startling theoretical breakthrough in the solving of systems of equations that often grow too vast and complex for the most powerful computers.

The discovery, which is to be formally published next month, is already circulating rapidly through the mathematical world. It has also set off a deluge of inquiries from brokerage houses, oil companies and airlines, industries with millions of dollars at stake in problems known as linear programming.

Faster Solutions Seen

These problems are fiendishly complicated systems, often with thousands of variables. They arise in a variety of commercial and government applications, ranging from allocating time on a communications satellite to routing millions of telephone calls over long distances, or whenever a limited, expensive resource must be spread most efficiently among competing users. And investment companies use them in creating portfolios with the best mix of stocks and bonds.

The Bell Labs mathematician, Dr. Narendra Karmarkar, has devised a radically new procedure that may speed the routine handling of such problems by businesses and Government agencies and also make it possible to tackle problems that are now far out of reach.

"This is a path-breaking result," said Dr. Ronald L. Graham, director of mathematical sciences for Bell Labs in Murray Hill, N.J. "Science has its moments of great progress, and this may well be one of them."

Because problems in linear programming can have billions or more possible answers, even high-speed computers cannot check every one. So computers must use a special procedure, an algorithm, to examine as few answers as possible before finding the best one — typically the one that minimizes cost or maximizes efficiency.

A procedure devised in 1947, the simplex method, is now used for such problems,

Continued on Page A19, Column 1

THE NEW YORK TIMES, November 19, 1984

Karmarkar at Bell Labs: an equation to find a new way through the maze

Folding the Perfect Corner

A young Bell scientist makes a major math breakthrough

Every day 1,200 American Airlines jets crisscross the U.S., Mexico, Canada and the Caribbean, stopping in 110 cities and bearing over 80,000 passengers. More than 4,000 pilots, copilots, flight personnel, maintenance workers and baggage carriers are shuffled among the flights; a total of 3.6 million gal. of high-octane fuel is burned. Nuts, bolts, altimeters, landing gears and the like must be checked at each destination. And while performing these scheduling gymnastics, the company must keep a close eye on costs, projected revenue and profits.

Like American Airlines, thousands of companies must routinely untangle the myriad variables that complicate the efficient distribution of their resources. Solving such monstrous problems requires the use of an abstruse branch of mathematics known as linear programming. It is the kind of math that has frustrated theoreticians for years, and even the fastest and most powerful computers have had great difficulty juggling the bits and pieces of data. Now Narendra Karmarkar, a 28-year-old Indian-born mathematician at Bell Laboratories in Murray Hill, N.J., after only a years' work has cracked the puzzle of linear programming by devising a new algorithm, a step-by-step mathematical formula. He has translated the procedure into a program that should allow computers to track a greater combination of tasks than ever before and in a fraction of the time.

Unlike most advances in theoretical mathematics, Karmarkar's work will have an immediate and major impact on the real world. "Breakthrough is one of the most abused words in science," says Ronald Graham, director of mathematical sciences at Bell Labs. "But this is one situation where it is truly appropriate."

Before the Karmarkar method, linear equations could be solved only in a cumbersome fashion, ironically known as the simplex method, devised by Mathematician George Dantzig in 1947. Problems are conceived of as giant geodesic domes with thousands of sides. Each corner of a facet on the dome

TIME MAGAZINE, December 3, 1984

Milestones for Interior Point Methods (IPMs)

- 1984 Projective IPM: Karmarkar – efficient in practice!?
- 1989 $O(n^3L)$ for IPMs: Renegar – best complexity
- 1989 Primal–Dual IPMs: Kojima ... – dominant since then
- 1989 Self-Concordant Barrier: Nesterov–Nemirovskii – extensions to smooth convex optimozation
- 1992 Semi-Definite Optimization (SDO) and Second Order Conic Optimization (SOCO): Alizadeh, Nesterov–Nemirovskii –new applications, approximations, software
- 1998 Robust LO: Ben Tal–Nemirovskii

Martin Grötschel

Towards IPMs: The Primal–Dual Linear Optimization Problems

The primal-dual LO problems is given as:

$$\min \; c^T x \qquad\qquad Ax = b, \quad x \geq 0,$$

$$\max \; b^T y \qquad\qquad A^T y + s = c, \quad s \geq 0,$$

where $c, x, s \in \mathcal{R}^n$, $b, y \in \mathcal{R}^m$, $A \in \mathcal{R}^{m \times n}$, $\text{rank}(A) = m$.

Optimality conditions and the central path are given as:

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= 0, \end{aligned} \qquad\qquad \begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= \mu e, \end{aligned}$$

where $e = (1, \ldots, 1)^T \in \mathcal{R}^n$.

We assume that the Interior Point Condition holds.

following Tamás Terlaky

Towards IPMs: The Central Path

The central path and the Classical Newton direction:

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= \mu e. \end{aligned}$$

$$\begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= \mu e - xs, \end{aligned}$$

Scaled Newton direction:

$$\begin{aligned} \bar{A} p_x &= 0, \\ \bar{A}^T \Delta y + p_s &= 0, \\ p_x + p_s &= v^{-1} - v \end{aligned}$$

Proximity Functions:

$$\begin{aligned} \Psi(v) &= \sum_{i=1}^{n} \left(\frac{v_i^2 - 1}{2} - \log v_i \right) \\ \Psi(v) &= \frac{1}{2} \|v - v^{-1}\|^2. \end{aligned}$$

---

where $\bar{A} = \frac{1}{\mu} A V^{-1} X$, $V = \text{diag}(v)$, $X = \text{diag}(x)$ with

$$v := \sqrt{\frac{xs}{\mu}}, \quad v^{-1} := \sqrt{\frac{\mu}{xs}}, \quad p_x := \frac{v\Delta x}{x}, \quad p_s := \frac{v\Delta s}{s}.$$

Martin Grötschel

Self-Regular Functions

$\psi(t)$ **is Self-Regular (SR) if**

SR1: $\psi(t)$ is strongly convex,

global minimum: $\psi(1) = 0$,

$\exists, \nu_1, \nu_2 > 0$ and $p, q \geq 1$,

such that for $\forall t \in (0, +\infty)$

$\nu_1(t^{p-1} + t^{1-q}) \leq \psi''(t) \leq \nu_2(t^{p-1} + t^{1-q})$,

SR2: For $t_1, t_2 > 0$, $r \in [0, 1]$.

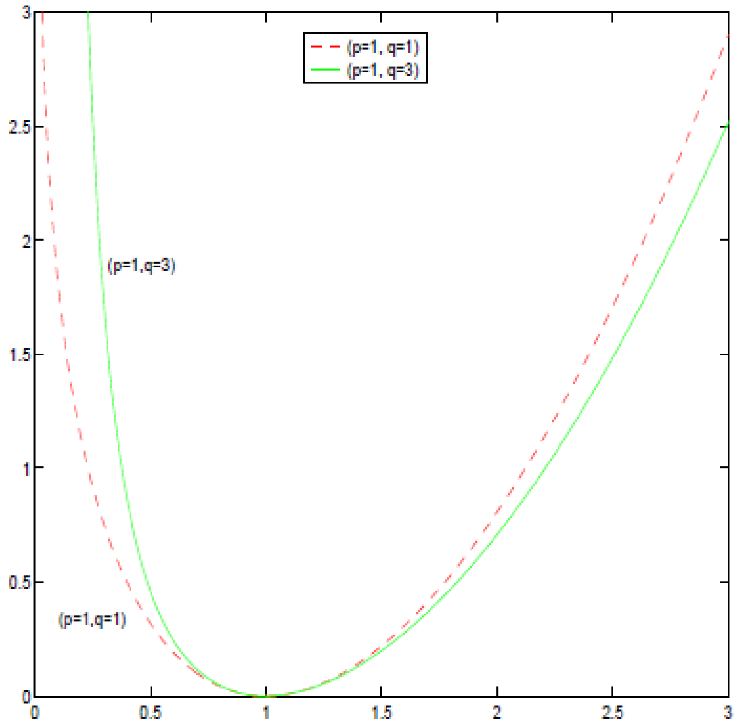
$\psi(t_1^r t_2^{1-r}) \leq r\psi(t_1) + (1 - r)\psi(t_2)$,

$SR2$: $\psi(\exp(\xi))$ is convex.

q : barrier degree;

p : growth degree.

SR is a non-closed convex cone.



$$\Psi(v) = \sum_{i=1}^{n} \psi(v_i)$$

Martin Grötschel

Primal-Dual Interior Point Methods with small and large updates

Input:

A proximity parameter τ; an accuracy parameter $\epsilon > 0$;
an update parameter $0 < \theta < 1$; a variable damping factor α;
(x^0, s^0), $\mu^0 = 1$ s.t. $\Psi(v^0) \le \tau$.

begin

$x := x^0$; $s := s^0$; $\mu := \mu^0$;

while $n\mu \ge \epsilon$ **do**

begin

$\mu := (1 - \theta)\mu$;

while $\Psi(v) \ge \tau$ **do**

begin

Do line search for $\Psi(v(\alpha))$;

$x := x + \alpha \Delta x$;

$s := s + \alpha \Delta s$;

end

end

end

Martin Grötschel

Complexity of Self-Regular IPMs

Method	Large update	Small update	
θ	$1 - 1/100$	$1/\sqrt{n}$	
Iter. bound	$\mathcal{O}(n \log \frac{n}{\epsilon})$	$\mathcal{O}(\sqrt{n} \log \frac{n}{\epsilon})$	
Performance	Efficient	Very poor	
SR-Method	SR-Large	SR-Small	SR-Large $q = \log n$
θ	$1 - 1/100$	$1/\sqrt{n}$	constant
Iter. bound	$\mathcal{O}(q\, n^{\frac{q+1}{2q}} \log \frac{n}{\epsilon})$	$\mathcal{O}(\sqrt{n} \log \frac{n}{\epsilon})$	$\mathcal{O}(\sqrt{n} \log n \log \frac{n}{\epsilon})$
Performance	Efficient	Very poor	Efficient

"Almost" constant (< 100) number of iterations in practice!

Martin Grötschel

Contents

Martin Grötschel

Lagrangean Relaxation & Non-differentiable Optimization

- Approach for very large scale and structured LPs
- Methods:
  - subgradient
  - bundle
  - bundle trust region

  or any other nondifferentiable NLP method that looks promising

Martin Grötschel

Lagrangian Relaxation

- Turning an LP into a nonlinear nondifferentiable optimization problem

$$\min c^T x$$
$$Ax = b$$
$$\left.\begin{array}{l} Dx \le d \\ x \ge 0 \end{array}\right\} =: Q$$

$$\max f(\lambda)$$
$$f(\lambda) := \min_{x \in Q} c^T x + \lambda^T (Ax - b)$$

(14.25) Satz. *Sei Q nicht leer und endlich und $f(\lambda) := \min_{x \in Q}(c^T x + \lambda^T(Ax - b))$, so gilt folgendes: Setzen wir für $\lambda_0 \in \mathbb{R}^m$, $L_0 := \{x_0 \in \mathbb{R}^m \mid f(\lambda_0) = c^T x_0 + \lambda_0^T(Ax_0 - b)\}$, so ist*

$$\partial f(\lambda_0) = \operatorname{conv}\{(Ax_0 - b) \mid x_0 \in L_0\} .$$

Martin Grötschel

Algorithms for nonlinear nondifferential programming

$$x_{i+1} = x_i + s_i d_i$$

d_i = subgradient (instead of gradient)

or element of ε-subdifferential (bundle)

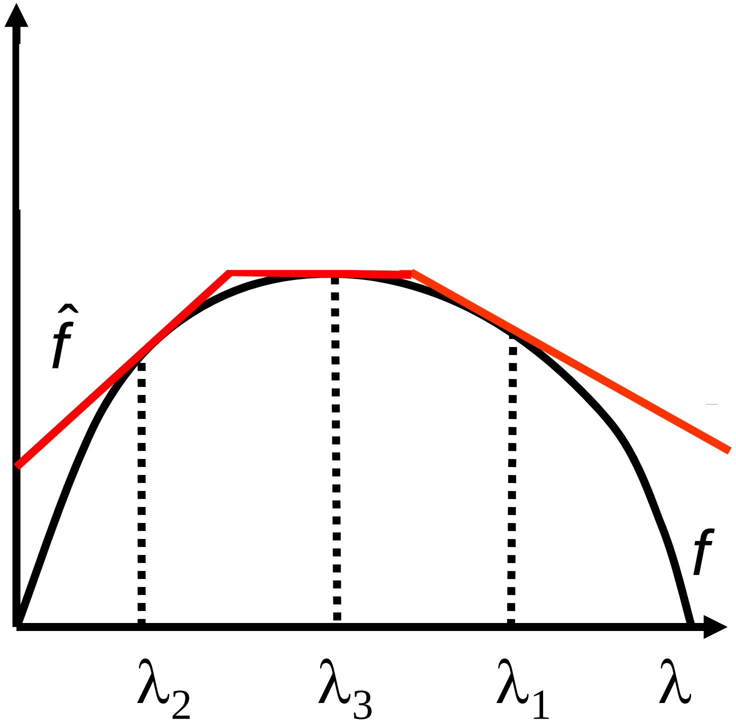
s_i = steplength

Martin Grötschel

Bundle Method

(Kiwiel [1990], Helmberg [2000])

- Max $f(\lambda) := \min_{x \in X} c^T x + \lambda^T (b - Ax)$
  X polyhedral (piecewise linear)



$$\bar{f}_\mu(\lambda) = c^T x_\mu + \lambda^T (b - Ax_\mu)$$

$$\hat{f}_k(\lambda) := \min_{\mu \in J_k} \bar{f}_\mu(\lambda)$$

$$\lambda_{k+1} = \operatorname{argmax}_{\lambda} \hat{f}_k(\lambda) - \frac{u_k}{2} \left\| \lambda - \hat{\lambda}_k \right\|^2$$

Ralf Borndörfer

Quadratic Subproblem

$$(1) \quad \max \hat{f}_k(\lambda) - \frac{u_k}{2}\left\|\lambda - \hat{\lambda}_k\right\|^2$$

$$\Leftrightarrow \quad (2) \quad \max \quad v - \frac{u_k}{2}\left\|\lambda - \hat{\lambda}^k\right\|^2$$

$$\text{s.t.} \quad v \le \bar{f}_\mu(\lambda), \text{ for all } \mu \in J_k$$

$$\Leftrightarrow \quad (3) \quad \max \quad \sum_{\mu \in J_k} \alpha_\mu \bar{f}_\mu(\hat{\lambda}) - \frac{1}{2u_k}\left\|\sum_{\mu \in J_k} \alpha_\mu (b - Ax_\mu)\right\|^2$$

$$\text{s.t.} \quad \sum_{\mu \in J_k} \alpha_\mu = 1$$

$$0 \le \alpha_\mu \le 1, \quad \text{for all } \mu \in J_k$$

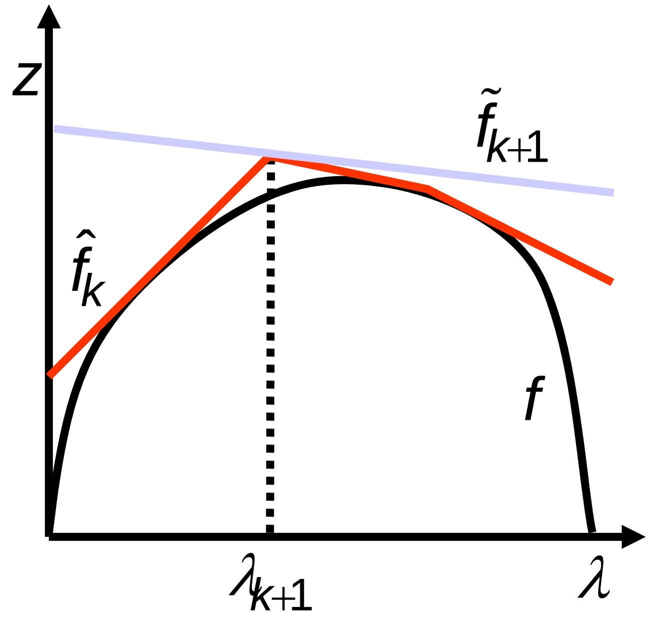
Ralf Borndörfer

Primal Approximation

$$\lambda_{k+1} = \hat{\lambda}_k + \frac{1}{u} \sum_{\mu \in J_k} \alpha_\mu (b - Ax_\mu)$$

$$\tilde{x}_{k+1} = \sum_{\mu \in J_k} \alpha_\mu x_\mu$$

$$\tilde{f}_k(\lambda) = c^T \tilde{x}_k + \lambda(b - A\tilde{x}_k)$$



- **Theorem**

$$\left\| b - A\tilde{x}_k \right\| \to 0 \ (k \to \infty)$$

$\Rightarrow (\tilde{x}_k)_{k \in N}$ converges to a point $\bar{x} \in \{x : Ax = b, x \in X\}$

Ralf Borndörfer

Where Bundle Wins

Ralf Borndörfer Andreas Löbel Steffen Weider

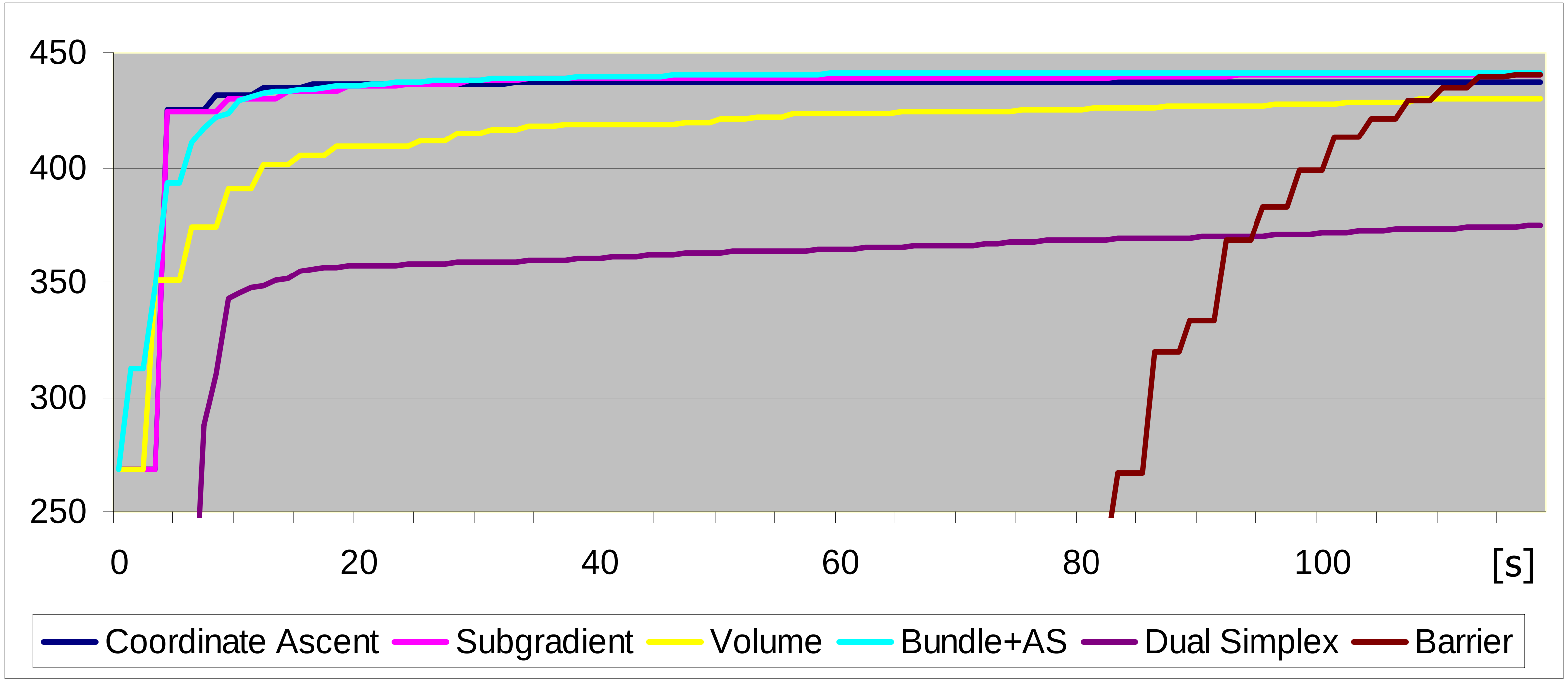
A Bundle Method for Integrated Multi-Depot Vehicle and Duty Scheduling in Public Transit

Coputational Results for a (Duty Scheduling) Set Partitioning Model

Duty Scheduling Problem Ivu41:

- 870 500 col
- 3 570 rows
- 10.5 non-zeroes per col

Coordinate Ascent:	Fast, low quality
Subgradient:	(Theoretical) Convergence
Volume:	Primal approximation
Bundle+AS:	Conv. + primal approx.
Dual Simplex:	Primal+dual optimal
Barrier:	Primal+dual optimal



Ralf Borndörfer

Contents

Martin Grötschel

Semi-algebraic Geometry
Real-algebraic Geometry

$f_i(x), i = 1, \ldots, l$ are polynomials in d real variables

$$S_{\geq} := \{x \in \mathbb{R}^d : f_1(x) \geq 0, \ldots, f_l(x) \geq 0\}$$

is called a basic closed semi-algebraic set

Martin Grötschel

Theorem of Bröcker(1991) & Scheiderer(1989) basic closed case

Every basic closed semi-algebraic set of the form

$$S = \{x \in \mathbb{R}^d : f_1(x) \geq 0, \dots, f_l(x) \geq 0\},$$

where $f_i \in \mathbb{R}[x_1, \dots, x_d], 1 \leq i \leq l,$ are polynomials,

can be represented by at most $d(d+1)/2$

polynomials, i.e., there exist polynomials

$p_1, \dots, p_{d(d+1)/2} \in \mathbb{R}[x_1, \dots, x_d]$ such that

$$S = \{x \in \mathbb{R}^d : p_1(x) \geq 0, \dots, p_{d(d+1)/2}(x) \geq 0\}$$

and d(d+1)/2 is best possible.

Martin Grötschel

Our first Theorem

Theorem Let $P \subset \mathbb{R}^n$ be a n-dimensional polytope given by an inequality representation. Then

$k \leq n^n$ polynomials $\mathfrak{p}_i \in \mathbb{R}[x_1, \ldots, x_n]$

can be constructed such that

$$P = \mathcal{P}(\mathfrak{p}_1, \ldots, \mathfrak{p}_k).$$

Martin Grötschel, Martin Henk:

The Representation of Polyhedra by Polynomial Inequalities

Discrete & Computational Geometry, 29:4 (2003) 485-504

Martin Grötschel

Our Main Theorem

Theorem Let $P \subset \mathbb{R}^n$ be a n-dimensional polytope given by an inequality representation. Then

2n polynomials $\mathfrak{p}_i \in \mathbb{R}[x_1,\ldots,x_n]$

can be constructed such that

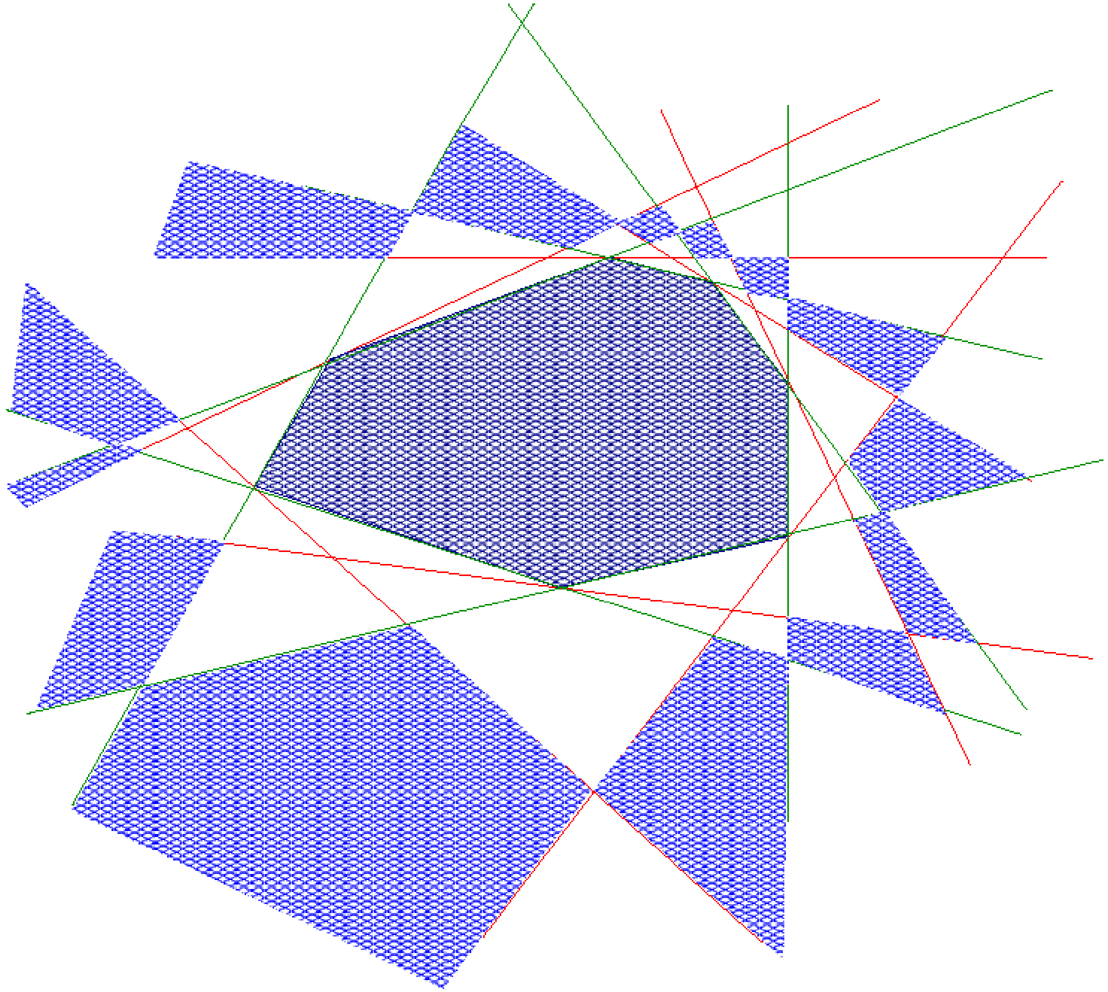
$$P = \mathcal{P}(\mathfrak{p}_1,\ldots,\mathfrak{p}_{2n}).$$

Hartwig Bosse, Martin Grötschel, Martin Henk:
Polynomial inequalities representing polyhedra
Mathematical Programming (2005)

http://www.springerlink.com/index/10.1007/s10107-004-0563-2

Martin
Grötschel

Homework: Try to describe an n-gon in 2-space by 2 polynomial inequalities



$$\{x \in \mathbb{R}^d : \mathfrak{p}_1(x) \geq 0 \text{ and } \mathfrak{p}_0(x) \geq 0\}$$

Martin Grötschel

Latest News

- Martin Henk, Gennadiy Averkov, *„Representing simple d-dimensional polytopes by d polynomials"*
  Mathematical Programming (A), **126**(2), 2011, 203-230;
  arXiv:0709.2099v1
- Bröcker, "Solution of the general case for polyhedra" ??, handwritten manuscript

My dream

- Make this result computationally useful!

- E.g., for every graph G with n nodes there are n polynomials, such that

$$STAB(G) = \mathcal{P}(\mathfrak{p}_1, \ldots, \mathfrak{p}_n).$$

Martin Grötschel

Contents

Martin Grötschel

Other Methods

There are many, see, e.g.,

- A. Schrijver (1986), *Theory of Linear and Integer Programming,* Wiley, Chichester, 1986
- M. J. Todd (2002) The many facets of linear programming. Math Program, Ser B 91:417–436
- M. Grötschel, L. Lovász, A. Schrijver, *Geometric Algorithms and Combinatorial Optimization,* Springer, 1988
- A LP-Newton method, based on the zonotope formulation and the minimum-norm-point algorithm of Wolfe (1976)

Optim Eng (2009) 10: 193–205
DOI 10.1007/s11081-008-9067-x

Zonotopes and the LP-Newton method

Satoru Fujishige · Takumi Hayashi ·
Kei Yamashita · Uwe Zimmermann

Contents

Martin Grötschel

Some LP/MIP Solvers

Solver	Version	URL
IBM CPLEX	12.2	www.cplex.com
Gurobi	3.0	www.gurobi.com
FICO XPress-MP	7	www.fico.com/en/Products/DMTools/Pages/FICO-Xpress-Optimization-Suite.aspx
…		
Lindo	6.1	www.lindo.com
Minto	3.1	coral.ie.lehigh.edu/~minto
SCIP	2.0	scip.zib.de
CBC	2.5	projects.coin-or.org/Cbc
Symphony	5.2	projects.coin-or.org/SYMPHONY
glpk	4.43	www.gnu.org/software/glpk/glpk.html
lp_solve	5.5	lpsolve.sourceforge.net
…		

OR / MS Today Surveys

OR/MS Today, June 2009

Linear Programming Survey
Table 3

Product	Platforms Supported											
	PC / Windows		PC / Linux		Unix			Other OS			Microprocessor Support	
	32-bit	64-bit	32-bit	64-bit	32-bit	64-bit	Specify flavor of Unix	32-bit	64-bit	Specify	Shared Memory	Distributed Memory
AIMMS, the modeling system	y	y	y	y							Parallel Solver Sessions (Windows/Linux)	
AMPL	y	y	y	y	y	y	Solaris, Mac OS X, AIX, HP-UX, IRIX					
BendX Stochastic Solver	y	y	y	y	y	y	Sun Solaris, HP-UX, AIX (Unix platforms are (C/C++/Java only)					
C-WHIZ	y	y	y									
CBC	y	y	y	y	y	y	AIX, Solaris			Can be ported to most systems	Linux,Unix, Windows (needs pthreads)	
CLP	y	y	y	y	y	y	AIX, Solaris			Can be ported to most systems		
CoinMP	y	y	y	y			Solaris, Mac OS X					
DATAFORM	y	y	y									
FICO Xpress	y	y	y	y	y	y	Solaris, AIX, HP-UX				All	
flopc++	y	y	y	y	y	y						
Frontier	y											Task splitting

Martin Grötschel

Which LP solvers are used in practice?

Preview summary

- Fourier-Motzkin: hopeless
- Ellipsoid Method: total failure
- primal Simplex Method: good
- dual Simplex Method: better
- Barrier Method: for large LPs frequently better
- For LP relaxations of IPs: dual Simplex Method

Martin Grötschel

http://www.netlib.org/lp/index.html

lp

Click here to see the number of accesses to this library.

```
lib     data
for     a set of test problems in MPS format.

lib     generators
for     programs that generate linear programming test problems

lib     infeas
for     infeasible linear programming test problems
```

Importance of test data!

Martin Grötschel

MIPLIB 1992/2010

MIPLIB - Mixed Integer Problem LIBrary

MIPLIB 2010

After its introduction, MIPLIB has become a standard test set used to compare the performance of mixed integer optimizers.

Since the first release in 1992 the MIPLIB has been updated several times. Now again 7 years have past since the last update in 2003. And again improvements in state-of-the-art optimizers, as well as improvements in computing machinery have made several instances too easy to be of further interest.

Last year a group of interested parties including participants from ASU, COIN, FICO, Gurobi, IBM, and MOSEK met at ZIB to discuss the guidelines for the 2010 release of the MIPLIB.

Involved people:

Tobias Achterberg (IBM)
Erling D. Andersen (Mosek)
Oliver Bastert (FICO)
Timo Berthold (ZIB, Matheon)
Robert Bixby (Gurobi)
Gerald Gamrath (ZIB)
Ambros Gleixner (ZIB)
Stefan Heinz (ZIB, Matheon)
Thorsten Koch (ZIB, Matheon)
Alexander Martin (TU Darmstadt)
Hans D. Mittelmann (Arizona State University)
Ted Ralphs (COIN-OR, Lehigh University)
Kati Wolter (ZIB)

We would be happy if you contribute to this library by sending us hard and/or real life instances. If you have any instances you would like to add to MIPLIB, please use the form below to submit it. The current deadline for instances is 10/1/2010!

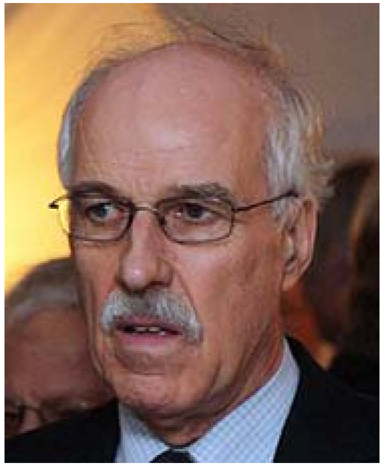
Independent Testing

Benchmarks for Optimization Software

by Hans Mittelmann (mittelmann at asu.edu)

The following are NEOS solvers we have installed.

BNBS, BPMPD, BPMPD-AMPL, Concorde, CONDOR, CSDP, DDSIP, FEASPUMP, FEASPUMP-AMPL, ICOS, NSIPS, PENBMI, PENSDP, QSOPT_EX, SCIP, SCIP-AMPL, SDPA, SDPLR, SDPT3, SeDuMi



http://plato.asu.edu/bench.html

LINEAR PROGRAMMING

- Benchmark of serial LP solvers (10-12-2010)
- Benchmark of parallel LP solvers (10-16-2010)
- Parallel CPLEX, GUROBI, and MOSEK on LP problems (7-18-2010)
- Large Network-LP Benchmark (commercial vs free) (10-16-2010)

MIXED INTEGER LINEAR PROGRAMMING

- MILP Benchmark - serial codes (10-15-2010)
- MILP Benchmark - parallel codes (10-14-2010)
- MILP cases that are difficult for some codes (10-8-2010)
- Feasibility Benchmark - Feaspump,CPLEX,SCIP,GUROBI (10-15-2010)
- Infeasibility Detection for MILP Problems (10-14-2010)

Martin Grötschel

LP survey

Robert E. Bixby, Solving Real-World Linear Programs: A Decade and More of Progress. *Operations Research* 50 (2002)3-15.

Bob on September 27, 2010 at his 65th birthday party

Progress in LP: 1988—2004

(<u>Operations Research</u>, Jan 2002, pp. 3—15, updated in 2004)

- Algorithms (*machine independent*):

  Primal *versus* best of Primal/Dual/Barrier 3,300x

- Machines (workstations →PCs): 1,600x
- NET: Algorithm × Machine 5,300,000x

(2 months/5300000 ~= 1 second)

Courtesy Bob Bixby

Progress in LP: 1988—2004

- Where are we today?
  - The good news
    - "LP is a solved problem in practice"
  - But, …. a word of warning
    - 2% of MIP models are blocked by linear programming
    - Little progress in LP computation since 2004
    - LP could become a serious bottleneck in the future

Courtesy Bob Bixby

The latest computational study: Ed Rothberg (Gurobi)

- Rothberg slides

- LP state of the art - according to Gurobi: as of September 28, 2010 (Bixby's 65th birthday conference in Erlangen, Germany)
- All software producer do computational studies permanently but rarely make them publicly available.

What can we solve today? "strange examples"

Example: Primal > Barrier > Dual

Problem name	: patrick1
Optimal objective	: 28609090
Variables	: 2,666,441 [Boxed: 2,656,781, Nneg: 9,660]
Objective nonzeros	: 684,145
Linear constraints	: 44,886 [Less: 8,173, Equal: 36,713]
Nonzeros	: 7,991,889
RHS nonzeros	: 41,808

Dual Simplex	: 488,900 iterations in 10,009 s (not finished)
Barrier+crossover	: 349 iterations in 3,111 s
Primal Simplex	: 3,268,455 (895,004) iterations in 1,900 s

Martin Grötschel

What can we solve today? "strange examples"

Example: Barrier > Primal > Dual

Problem name	: aflow_2000_50
Optimal objective	: 4720.3225806
Variables	: 3,996,000 [Boxed: 1,998,000, Nneg: 1,998,000]
Objective nonzeros	: 1,958,437
Linear constraints	: 2,001,998 [Less: 1,998,000, Equal: 3,998]
Nonzeros	: 9,988,972
RHS nonzeros	: 3,998

Dual Simplex	: 1,049,300 iterations in 10,054 sec (not finished)
Primal Simplex	: 2,321,540 (28277) iterations in 6,752 sec
Barrier + crossover	: 40 iterations in 1,704 sec (total 1,938 sec)
8 threads	: 430.03 sec

Martin Grötschel

What can we solve today? "strange examples"

Example: Primal > Dual > Barrier

Problem name	: ts.log-bundle-060831-162253
Optimal objective	: 5.69997.52369
Variables	: 218,776 [Boxed: 218,776]
Objective nonzeros	: 124,060
Linear constraints	: 1,102,735 [Less: 970,339, Greater: 11,590, Equal: 120,806]
Nonzeros	: 2,554,196
RHS nonzeros	: 981,241

Presolve generated explicit dual

Dual Simplex	: 132854 in 163 sec
Primal Simplex	: 96397 (0) in 31 sec
Barrier	: 53 iterations in 10069 sec (not finished)

Martin Grötschel

ZIB Instances

	Variables	Constraints	Non-zeros	Description
1	12,471,400	5,887,041	49,877,768	Group Channel Routing on a 3D Grid Graph (Chip-Bus-Routing)
2	37,709,944	9,049,868	146,280,582	Group Channel Routing on a 3D Grid Graph (different model, infeasible)
3	29,128,799	19,731,970	104,422,573	Steiner-Tree-Packing on a 3D Grid Graph
4	37,423	7,433,543	69,004,977	Integrated WLAN Transmitter Selection and Channel Assignment
5	9,253,265	9,808	349,424,637	Duty Scheduling with base constraints

LP can still be difficult

- **We were not able to compute a feasible basis for zib03 so far.**
- After 10 h we still do not even have a primal feasible solution. Furthermore, experiments with smaller instances suggest the model is very unfavorable for the simplex method, especially regarding warm starts. Unfortunately, it is an IP.

Algorithm	Time [h]	Result	Memory [GB]	Resident [GB]
Primal Simplex	>300	Infeasibility 2189	24	18
Dual Simplex	>300	Lower bound 8335	24	18
Bundle	13	Lower bound 5951	55	18
Interior Point	103 (32 threads)	Optimal 1.2228.148	256	175
Crossover	>300	unfinished		

Summary

You should be surprised
if a linear program could not be solved

Martin
Grötschel

Contents

Martin Grötschel

Advertisement: http://zibopt.zib.de/

ZIB Optimization Suite

Konrad-Zuse-Zentrum für Informationstechnik Berlin
Division Scientific Computing
Department Optimization

The ZIB Optimization Suite is a tool for generating and solving mixed integer programs. It consists of the following parts

ZIMPL	a mixed integer programming modeling language
SoPlex	a linear programming solver
SCIP	a mixed integer programming solver and constraint programming framework.

The user can easily generate linear programs and mixed integer programs with the modeling language ZIMPL. The resulting model can directly be loaded into SCIP and solved. In the solution process SCIP may use SoPlex as underlying LP solver.

Since all three tools are available in source code and free for academic use, they are an ideal tool for academic research purposes and for teaching integer programming.

See ZIB licences for more information.

SoPlex Sequential object-oriented simplex

SoPlex is an implementation of the revised simplex algorithm. It features primal and dual solving routines for linear programs and is implemented as a C++ class library that can be used with other programs.

Roland Wunderling,
Paralleler und Objektorientierter Simplex-Algorithmus,
Dissertation, TU Berlin,1997

Optimization Problems in Printed Circuit Board Assembly

Petra Bauer

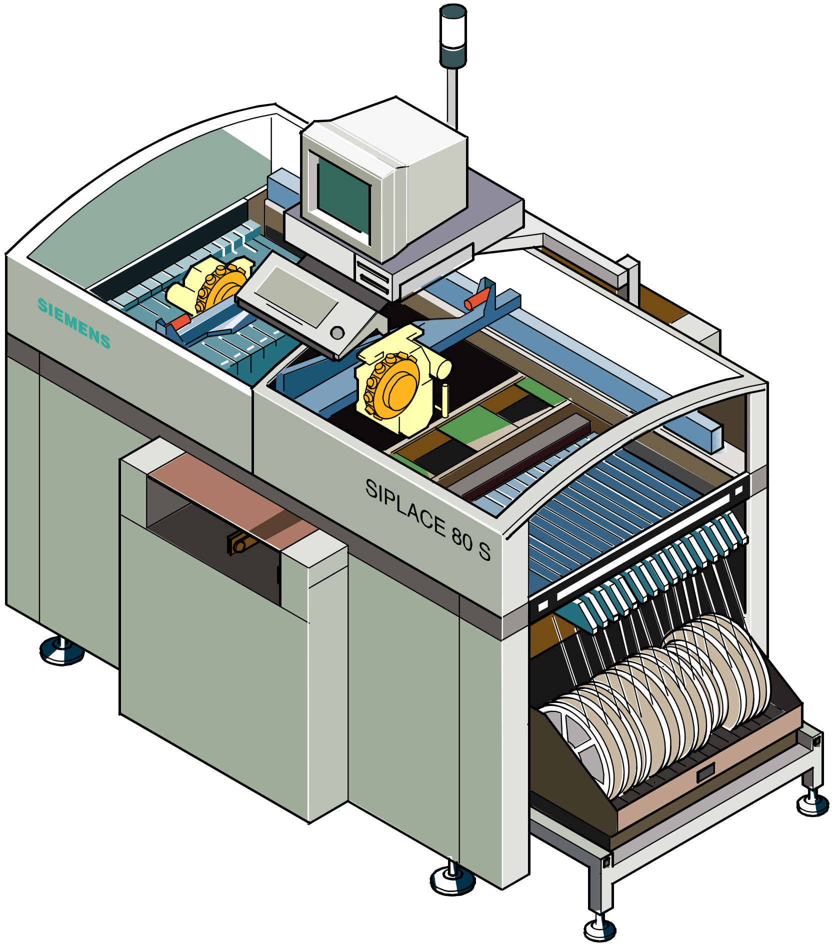
Siemens AG, Munich, Germany

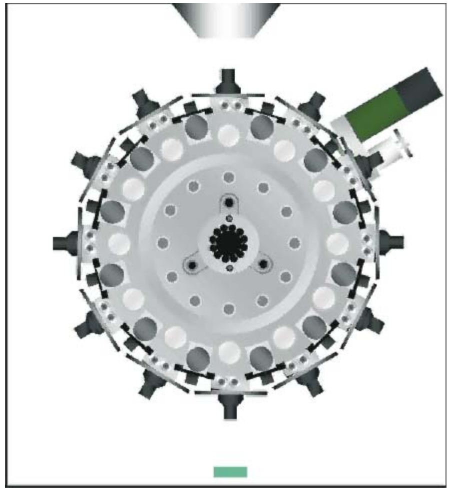
CORPORATE TECHNOLOGY

CT 6 S&E

Software & Engineering
Discrete Optimization

SIPLACE Placement Machines





Martin Grötschel

Zimpl

- Zimpl is a little language to translate the mathematical model of a problem into a linear or (mixed-) integer mathematical program expressed in .lp or .mps file format which can be read and (hopefully) solved by a LP or MIP solver.

- Thorsten Koch, *Rapid Mathematical Programming*, Dissertation, TU Berlin 2004
(awarded with the Dissertation Prize 2005 of the Gesellschaft für Operations Research)

Martin Grötschel

SCIP http://scip.zib.de/

Tobias Achterberg, Tobias, *Constraint Integer Programming*, Dissertation, TU Berlin, 2007

- Dissertation Prize 2008 of the Gesellschaft für Operations Research (GOR)
- George B. Dantzig Dissertation Award 2008 of the Institute of Operations Research and the Management Sciences (INFORMS), 2nd prize)
- Beale-Orchard-Hays Prize 2009 of the Mathematical Optimization Society for the paper: Tobias Achterberg, "SCIP: Solving constraint integer programs", Mathematical Programming Computation, 1 (2009), pp. 1-41.

Contents

Martin Grötschel

Improving the Simplex Numerics

On the factorization of simplex basis matrices

R. LUCE, J. DUINTJER TEBBENS, J. LIESEN and R. NABBEN
Technical University of Berlin
M. GRÖTSCHEL and T. KOCH
Zuse Institute Berlin
and
O. SCHENK
University of Basel

Reinhard Nabben

Martin Grötschel

Jörg Liesen

Emmy Noether-Programm
Deutsche Forschungsgemeinschaft DFG

MATHEON B17

Robert Luce
Research assistant with industry background

ILOG Changing the rules of business

Thorsten Koch

Jurjen Duintjer-Tebbens
Project researcher, from Czech Academy of Sciences

Olaf Schenk **Anshul Gupta**

ZIB-Report 09-24 (July 2009)

http://opus.kobv.de/zib/volltexte/2009/1188/

From the Abstract: The Findings in Brief

In the simplex algorithm, solving linear systems with the basis matrix and its transpose accounts for a large part of the total computation time. The most widely used solution technique is sparse LU factorization, paired with an updating scheme that allows to use the factors over several iterations. Clearly, small number of fill-in elements in the LU factors is critical for the overall performance.

Using a wide range of LPs we show numerically that after a simple permutation the non-triangular part of the basis matrix is so small, that the whole matrix can be factorized with (relative) fill-in close to the optimum. This permutation has been exploited by simplex practitioners for many years. But to our knowledge no systematic numerical study has been published that demonstrates the effective reduction to a surprisingly small non-triangular problem, even for large scale LPs.

For the factorization of the non-triangular part most existing simplex codes use some variant of dynamic Markowitz pivoting, which originated in the late 1950s. We also show numerically that, in terms of fill-in and in the simplex context, dynamic Markowitz is quite consistently superior to other, more recently developed techniques.

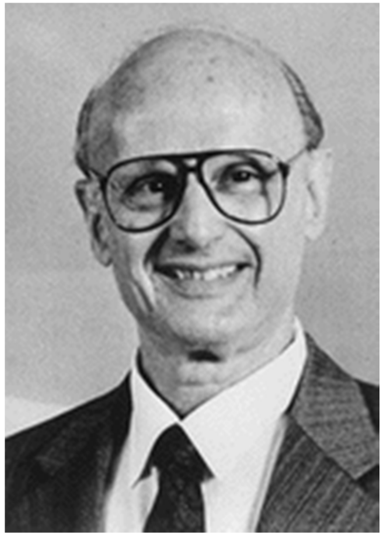
Martin Grötschel

A major question addressed in B17

- Most Simplex-based LP codes (such as ZIB´s **SoPlex**) use Dynamic Markowitz pivoting (Markowitz, 1957) to find fill-in reducing permutations.

- There have been dozens of new LU codes, particularly in recent years (see Davis´ list).

- *Is Markowitz still state of the art?*

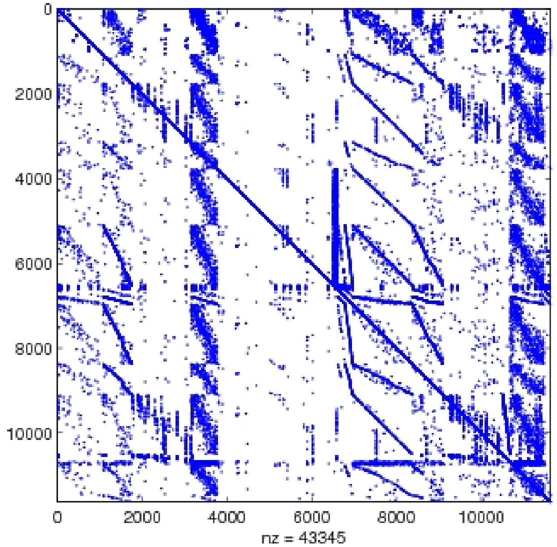


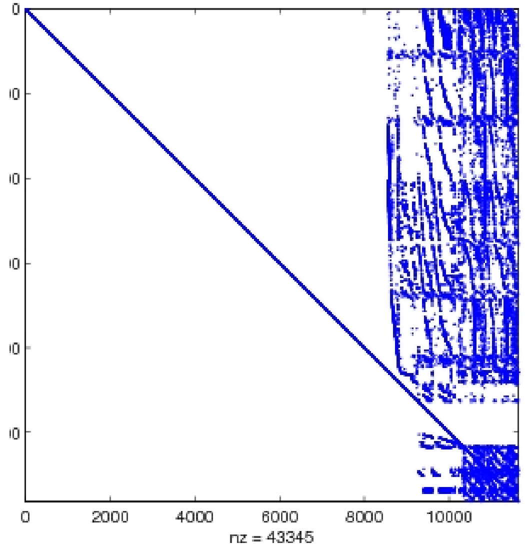
Harry M. Markowitz (*1928)
Economics Nobel Price Winner 1990

Results of B17: Typical structure of LP basis matrix

- Using 200 different LPs (392.701 computed LU factorizations) from various collections (MIPLIB, Mittelmann, Netlib, ZIB) we demonstrated that *typically*

$$PBQ = \begin{pmatrix} U & * & * \\ 0 & L & 0 \\ 0 & * & N \end{pmatrix} \text{ with } U, L \text{ triangular and } N \ \textbf{\textit{small}}.$$

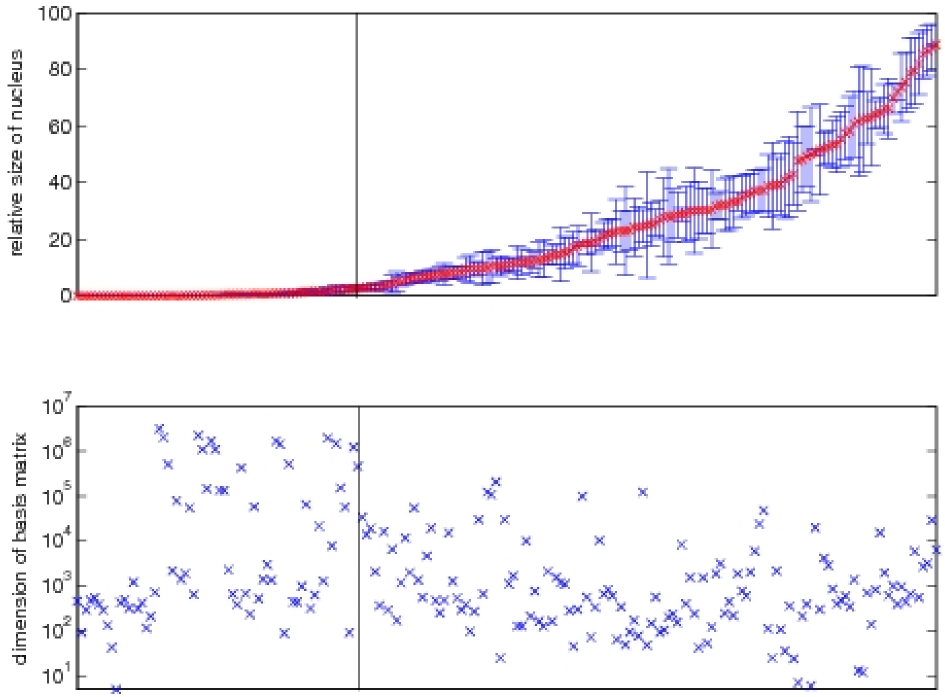




Martin Grötschel

Results of B17: Typical structure of LP basis matrix

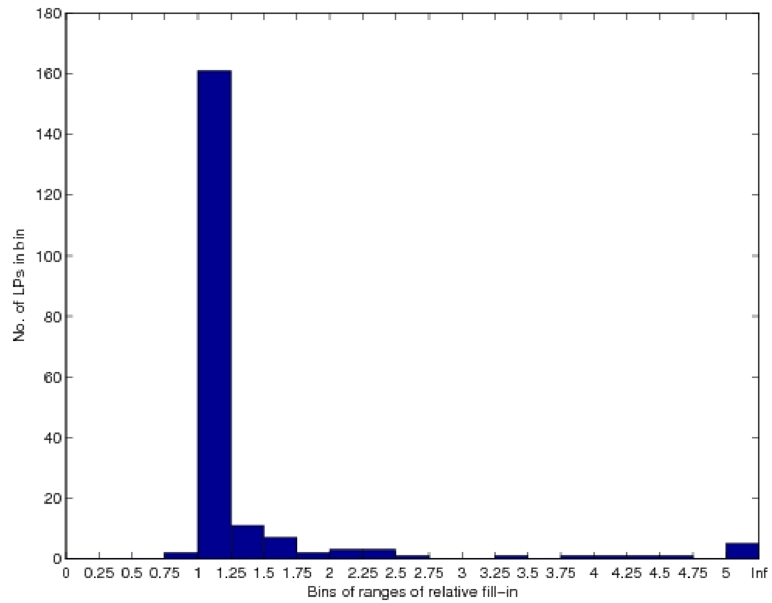
- Often Order(***N***) remains small throughout the Simplex run, particularly for large-scale LPs.
- This means ***B*** is almost reducible (decomposable)
  (for graph theorists: irreducible means digraph is strongly connected).



Martin Grötschel

Results of B17: Why Markowitz *globally* is so close to optimal

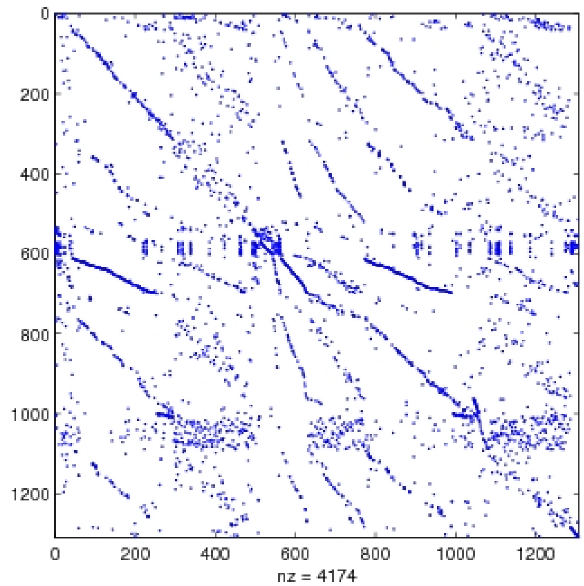
- Markowitz yields permutations that identify ***N***
  (main task: identify column/row singletons).
- Shown above: Often Order(***N)*** << Order(***B)***.
- Everything except ***N*** can be solved without fill-in,
  hence Markowitz is almost optimal in the LP context.
- In most examples:
  The Markowitz of **SoPlex**
  produces a relative fill-in
  in ***B*** close to the optimal 1.0.



Martin Grötschel

Results of B17: Markowitz is even the best for *N*

- A surprising result: Dynamic Markowitz even outperforms top-notch LU codes (PARDISO, UMFPACK, WSMP) in terms of fill-in when applied just to ***N***.
- Apparent reason: Most modern LU codes assume some type of structure (made for engineering applications).
- Typical ***N*** in the LP context: Extremely sparse with extremely low degree of structural symmetry.



Martin Grötschel

Concluding remarks

- Sparse GE involves many areas, including graph theory, numerical linear algebra, numerical analysis, scientific computing, …
- Many possibilities for further research.
- B17: Markowitz, though from the 50s, still is the method of choice in the LP context (fill-in close to optimal).
- Not addressed here: Numerical stability, efficient solution of $\boldsymbol{B}x=c$, interplay of Simplex and linear solver, …

Martin Grötschel

Contents

Martin Grötschel

Exact Linear Programming

Floating Point Computation for Linear Programming:

Advantages of Floating Point

- Fast computation
- Lower memory usage
- Sufficient for many applications
- Results are often very near optimal

Disadvantages

- Solutions are **not exact**
- Algorithms can **fail** for numerical reasons
- Correct solutions are not **guaranteed**

Slide from Dan Steffy

Example: sgpf5y6 from Mittelmann LP test set

LP Solver	Objective Value
Cplex 7.1 Primal	6398.71
Cplex 7.1 Dual	6484.44
Cplex 12.1 Primal	6425.87
Cplex 12.1 Dual	6484.47
Gurobi 2.0 Primal	6484.47
Gurobi 2.0 Dual	6484.47
XPress-20 Primal	6349.93
XPress-20 Dual	6408.02
QSopt Primal	6419.94
QSopt Dual	6480.33
CLP-1.12.0	6481.26
Soplex 1.2.2	6473.33
GLPK-4.44	6484.47
Exact Value (QSopt_ex)	$\frac{1621116398840608}{250000000000} \approx 6484.47$

(Some results reported by W. Cook, S. Dash, H. Mittelmann, D. Steffy)

Martin Grötschel

Slide from Dan Steffy

GNU Multiple Precision
Arithmetic Library

GMP
«Arithmetic without limitations»

http://gmplib.org

Applegate, Cook, Dash and Espinoza [2007] tested pure rational simplex implementation.

It was **hundreds or thousands** of times slower than floating-point code.

GNU Multiple Precision
Arithmetic Library

GMP
«Arithmetic without limitations»

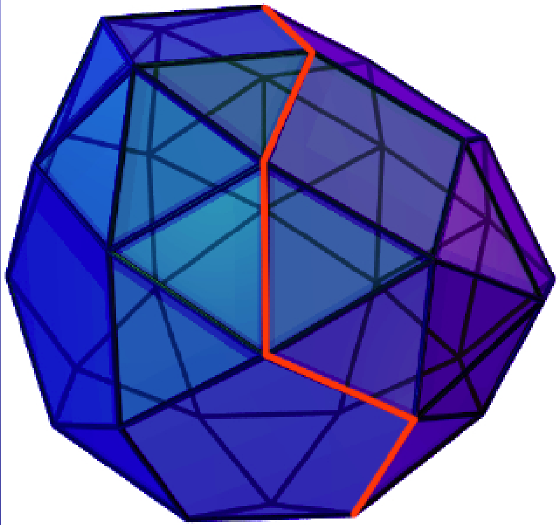
http://gmplib.org

Hybrid Symbolic-Numeric Computation

Use fast floating-point computation to assist in computing exact solutions.

Exact LP Solver

Simplex method terminates with basic solution
→ structural description of solution



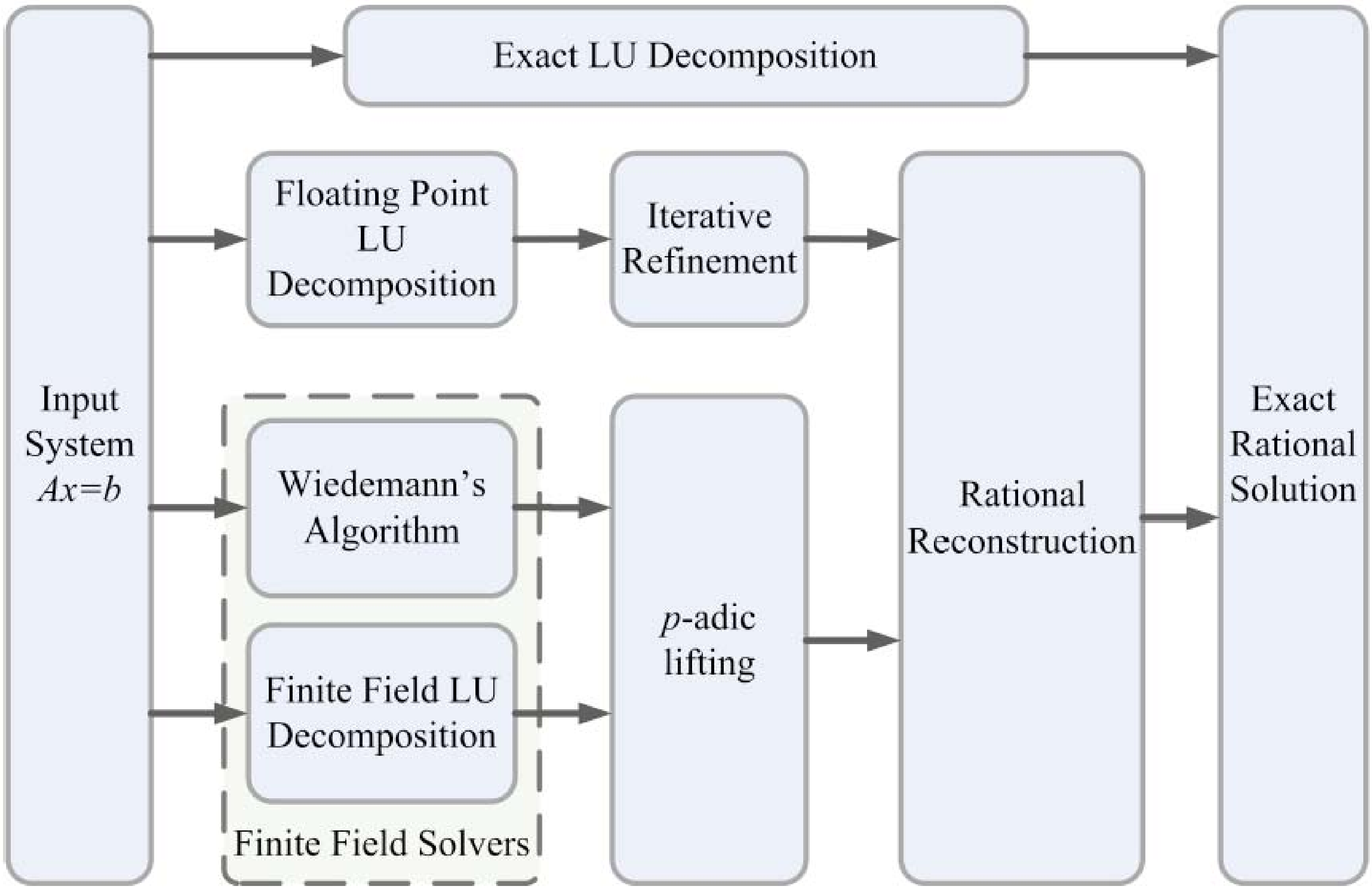
QSopt_ex: Exact Rational LP Solver[1]

- Simplex method performed numerically
- Final basic solution computed and checked exactly
- Precision is increased if necessary
- Roughly **two to five times slower** than floating-point LP solver

[1] Developed by Applegate, Cook, Dash and Espinoza [2007]

Martin Grötschel

Overview of Aproaches: Steffy & Cook



Martin Grötschel

Who is interested in exact solutions?

Thomas Hales' recent proof of the Kepler Conjecture relies on solving thousands of LPs

When are exact LP results necessary?

- Difficult feasibility problems
- Computer assisted proofs
- LP as a reliable subroutine
- When users demand it

Contents

Current issues: very^n large scale

- Very, very large scale LPs derived from IPs:
  billions of variables (verification of systems on chip, transportation, telecommunication,..),
  (column generation, cutting planes, Lagrange …)

- Modeling languages ZIMPL, OPL, AMPL

- "Effortless" solution of IPs using appropriate models via modeling languages

Martin Grötschel

Future Hardware Speed-Up

It is widely believed that in the future

- the speed of a single processor core will not substantially increase anymore
- the number of cores per processor will continuously increase
- GPUs and CPUs will merge again.

- Conclusion: If we want to continue to benefit from the development in hardware, LP and MIP solvers have to take advantage of parallel processing.

Parallelization

- The simplex algorithm can't be efficiently parallelized.
- There is much more hope for Barrier codes, and a lot of work is going on.

Martin Grötschel

Contents

Martin Grötschel

Bicriteria Optimization Model - Profit versus Robustness

$$
\begin{array}{llrll}
(BI-PCP) & \text{(i)} & \max \sum_{p\in P} w_p x_p & & \\
 & \text{(ii)} & \max \sum_{q\in Q} r_q y_q & & \\
 & \text{(iii)} & \sum_{p\in P_i} x_p & \leq 1, & \forall i \in I \\
 & \text{(iv)} & \sum_{q\in Q_j} y_q & \leq 1, & \forall j \in J \\
 & \text{(v)} & \sum_{a\in p\in P} x_p - \sum_{a\in q\in Q} y_q & \leq 0, & \forall a \in A_{LR} \\
 & \text{(vi)} & x_p, y_q & \in \{0,1\} & \forall p \in P,\ q \in Q
\end{array}
$$

Variables

- Path und config usage (request i uses path p, track j uses config q)

Constraints

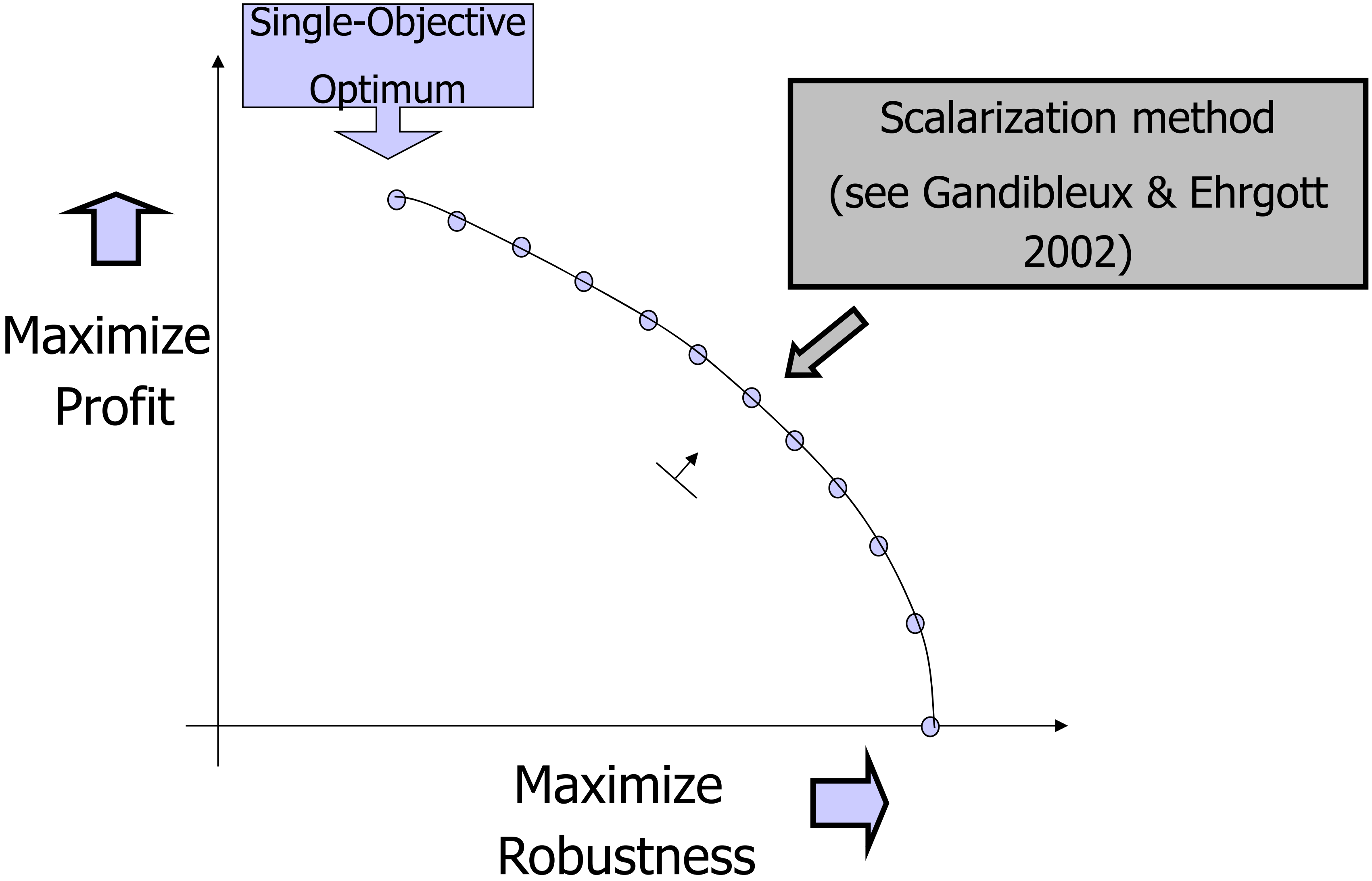
- Path and config choice
- Path-config-coupling (track capacity)

Objective Function

- Maximize proceedings and robustness

Thomas Schlechte

Price of Robustness (LP case)



Multi-objective LP

- Commercial software vendors offer scalarization.
- Computing the Pareto set is (in general) beyond what we can do.
- However, for small numbers of objective functions, special purpose methods may work.

Martin Grötschel

Contents

"Nonlinear" LPs

- Quadratic (convex) objective functions can be handled with the simplex method.
  Commercially available for the convex case.
- Additional quadratic (convex) constraints can be handled with interior point methods.

Martin Grötschel

Contents

Martin Grötschel

Current issues

- Very very large scale IPs:
  billions of variables (verification of systems on chip, transportation, telecommunication,..),
  (column generation, cutting planes, Lagrange …)

- Modeling languages ZIMPL, OPL, AMPL

- "Effortless" solution of IPs using appropriate models via modeling languages

Martin Grötschel

Solving Linear Programs

Martin Grötschel

13. Februar 2013

Thanks for your attention

Algorithmische Diskrete Mathematik I

„Einführung in die Lineare und Kombinatorische Optimierung“

WS 2012/2013, Institut für Mathematik, TU Berlin

Martin Grötschel
- Institut für Mathematik, Technische Universität Berlin (TUB)
- DFG-Forschungszentrum "Mathematik für Schlüsseltechnologien" (MATHEON)
- Konrad-Zuse-Zentrum für Informationstechnik Berlin (ZIB)

groetschel@zib.de http://www.zib.de/groetschel