

Linear and Integer Programming: an Introduction

CO@W Berlin

Martin Grötschel

21.9.2009

11:00 – 12:30

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>



Contents

CO@W

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



Contents

CO@W

1. **Linear, integer, nonlinear programming, optimization: What's that?**
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



typical optimization problems

$$\max f(x) \text{ or } \min f(x)$$

$$g_i(x) = 0, \quad i = 1, 2, \dots, k$$

$$h_j(x) \leq 0, \quad j = 1, 2, \dots, m$$

$$x \in \mathbf{R}^n \text{ (and } x \in S)$$

$$\min c^T x$$

$$Ax = a$$

$$Bx \leq b$$

$$x \geq 0$$

$$(x \in \mathbf{R}^{n^n})$$

$$(x \in \mathbf{k}^{n^n})$$

$$\min c^T x$$

$$Ax = a$$

$$Bx \leq b$$

$$x \geq 0$$

$$x \in \mathbf{Z}^n$$

$$(x \in \{0,1\}^n)$$

„general“
(nonlinear)
program
NLP

linear
program
LP

(linear)
integer
program
IP, MIP

program = optimization problem



Contents

1. Linear, integer, nonlinear programming, optimization: What's that?
2. **Historic roots**
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples

} will be covered by Bob Bixby
on Saturday



Linear Programming

CO@W

$$\max c_1 x_1 + c_2 x_2 + \dots + c_n x_n$$

subject to

$$a_{11} x_1 + a_{12} x_2 + \dots + a_{1n} x_n = b_1$$

$$a_{21} x_1 + a_{22} x_2 + \dots + a_{2n} x_n = b_2$$

.

$$a_{m1} x_1 + a_{m2} x_2 + \dots + a_{mn} x_n = b_m$$

$$x_1, x_2, \dots, x_n \geq 0$$

$$\max c^T x$$

$$Ax = b$$

$$x \geq 0$$

linear program
in standard form



Linear Programming: a very brief history

CO@W

- 1826/1827 Jean Baptiste Joseph Fourier (1786-1830): rudimentary 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 simplex algorithm

$$\max c^T x$$

$$Ax = b$$

$$x \geq 0$$

- Today: In my opinion and from an economic point of view, **linear programming is the most important development of mathematics in the 20th century.**



Optimal use of scarce resources

foundation and economic interpretation of LP

CO@W



Leonid V. Kantorovich Tjalling C. Koopmans
Nobel Prize for Economics 1975



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

CO@W

Min $x_1 + x_2$

costs

$2x_1 + x_2 \geq 3$

protein

$x_1 + 2x_2 \geq 3$

carbohydrates

$x_1 \geq 0$

potatoes

$x_2 \geq 0$

beans

minimizing the
cost of food



George J. Stigler
Nobel Prize in
economics 1982



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

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>



Milton Friedman on George J. Stigler

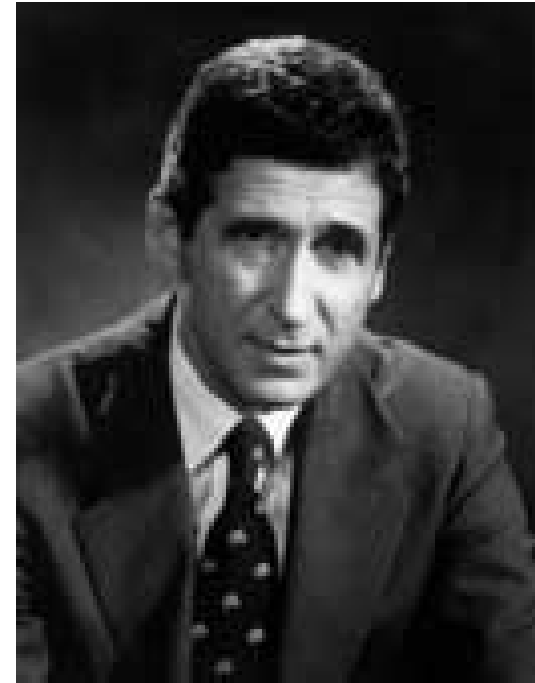
An early example of the latter is an article on "The Cost of Subsistence" (1945), which starts, "Elaborate investigations have been made of the adequacy of diets at various income levels, and a considerable number of 'low-cost,' 'moderate,' and 'expensive' diets have been recommended to consumers. Yet, so far as I know, no one has determined the minimum cost of obtaining the amounts of calories, proteins, minerals, and vitamins which these studies accept as adequate or optimum." George then set himself to determine the minimum cost diet, in the process producing one of the earliest formulations of a linear programming problem in economics, for which he found an approximate solution, explaining that "there does not appear to be any direct method of finding the minimum of a linear function subject to linear constraints." Two years later George Dantzig provided such a direct method, the simplex method, now widely used in many economic and industrial applications.

<http://www.nap.edu/readingroom.php?book=biomems&page=gstigler.html>



George Dantzig and Ralph Gomory

CO@W



„founding fathers“

~1950

linear programming

~1960

integer programming

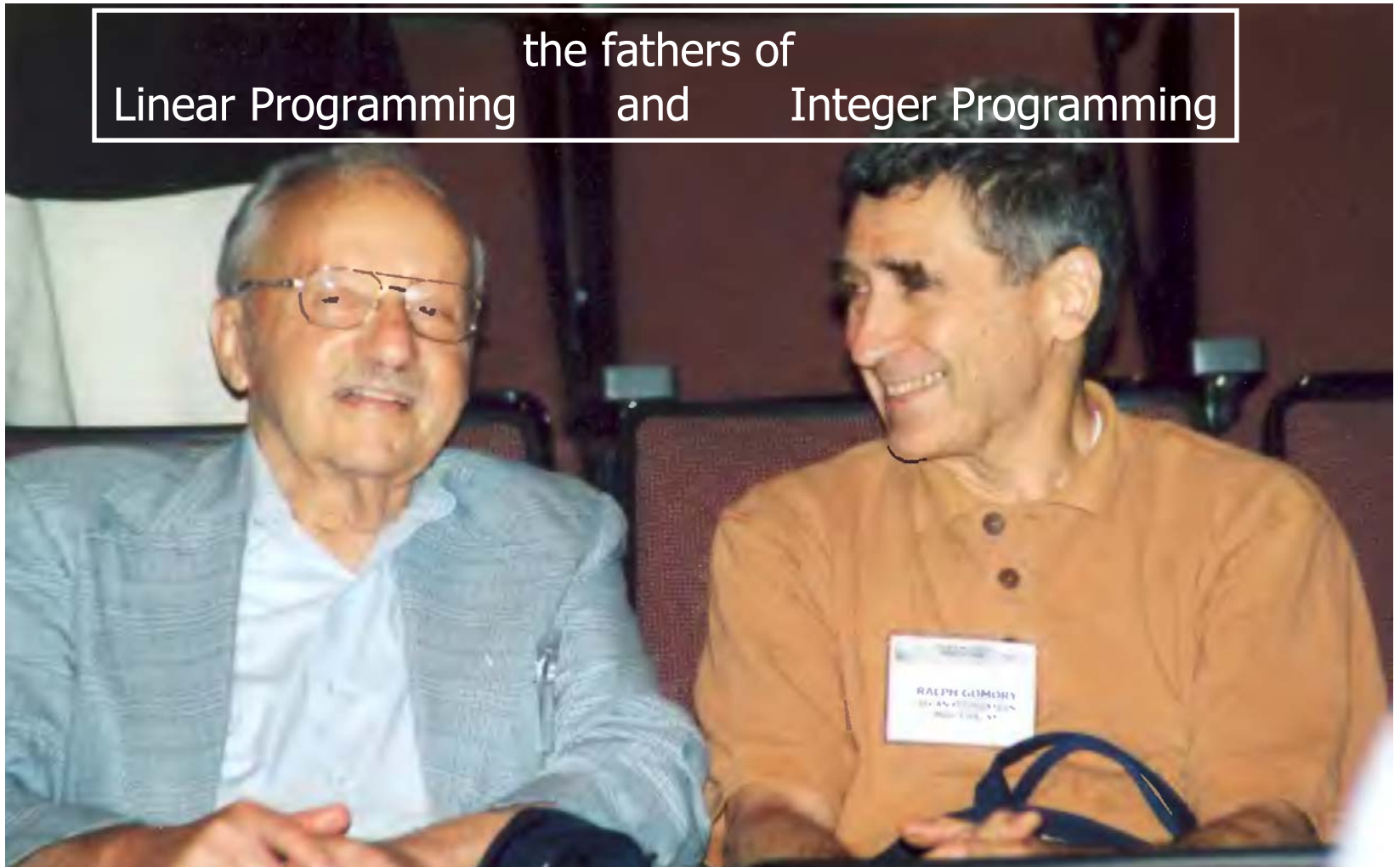


George Dantzig and Ralph Gomory

CO@W

ISMP Atlanta 2000

the fathers of
Linear Programming and Integer Programming



Dantzig and Bixby

CO@W



George Dantzig and
Bob Bixby

at the International
Symposium on Mathematical
Programming,

Atlanta, August 2000



Commercial software

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

The first commercial LP-Code was on the market in 1954 (i.e., ~55 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



Contents

CO@W

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. **LP-Theory**
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



Linear Programming

CO@W

$$\min c^T x$$

$$Ax = a$$

$$Bx \leq b$$

$$x \geq 0$$



$$\min c^T x$$

$$Ax = a$$

$$Bx \leq b$$

$$x \geq 0$$

$$\min c^T x$$

$$Ax = a$$

$$x \geq 0$$

$$\min c^T x$$

$$Bx \leq b$$

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



Optimizers' dream: Duality theorems

co@w

- Max-Flow Min-Cut Theorem

The maximal (s,t)-flow in a capacitated network is equal to the minimal capacity of an (s,t)-cut.

- The Duality Theorem of linear programming

$$\begin{array}{rcl} \max c^T x & = & \min y^T b \\ Ax \leq b & & y^T A \geq c^T \\ x \geq 0 & & y \geq 0 \end{array}$$



Optimizers' dream: Duality theorems for integer programming

co@w

- The **Max-Flow Min-Cut Theorem**
does not hold if several source-sink relations are given
(multicommodity flow).
- The **Duality Theorem of linear programming**
does not hold if integrality conditions are added

$$\begin{array}{ccc} \max c^T x & \leq & \min y^T b \\ Ax \leq b & & y^T A \geq c^T \\ x \geq 0 & & y \geq 0 \\ x \in \mathbf{Z}^n & & y \in \mathbf{Z}^m \end{array}$$



Important theorems

- complementary slackness theorems
- redundancy characterizations

CO@W



LP Solvability

co@w

- We assume in this course that the participants are somewhat familiar with complexity theory:
 - **Polynomial time solvability**, solvability in strongly polynomial time
 - Classes: P and NP , **NP -completeness**, 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?



LP Solvability: Generalizations

CO@W

Theorem (GLS 1979, 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 \mathbf{K} (e. g. polyhedra) if and only if, there exists a polynomial time algorithm that decides, for any given point \mathbf{x} , whether \mathbf{x} is in \mathbf{K} , and that, when \mathbf{x} is not in \mathbf{K} , finds a hyperplane that separates \mathbf{x} from \mathbf{K} .

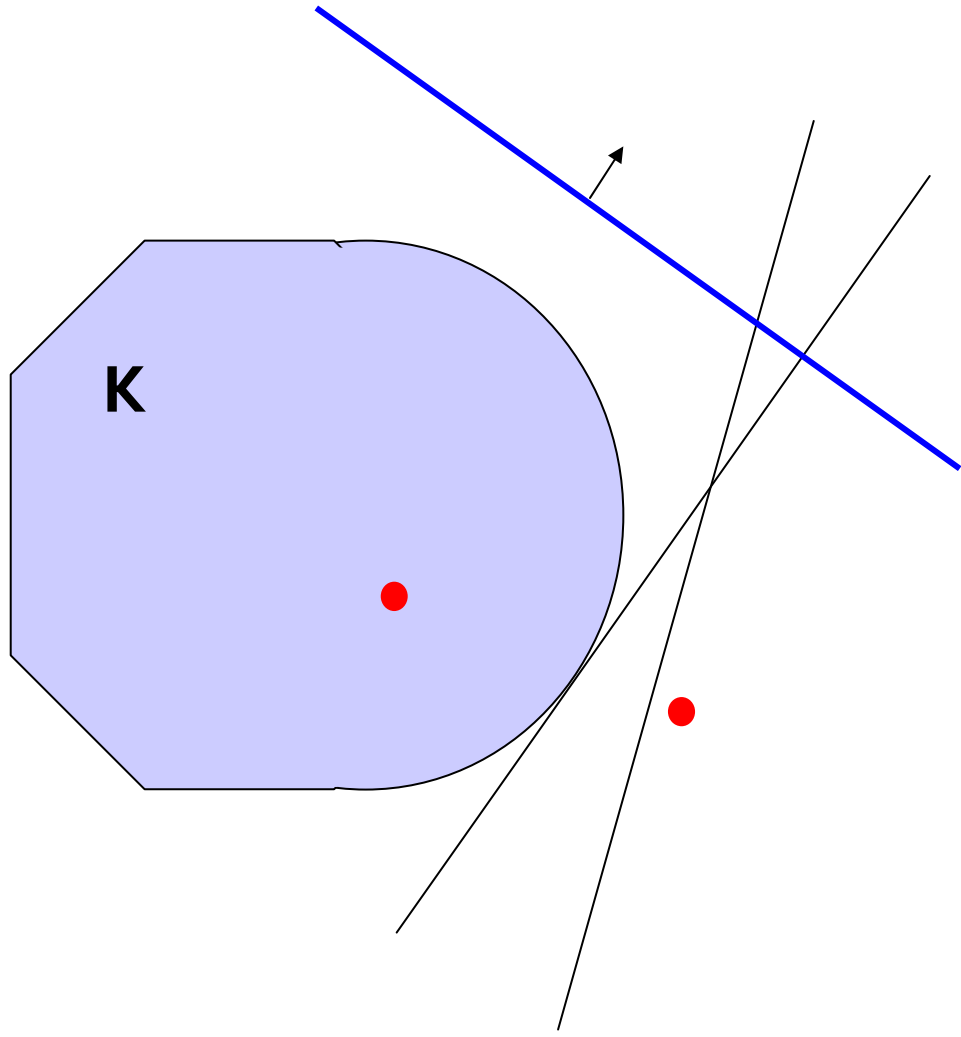
Short version: Optimization and Separation are polynomial-time equivalent.

Consequence: Theoretical Foundation of cutting plane algorithms.



Separation

CO@W



IP Solvability

CO@W

Theorem

Integer, 0/1, and mixed integer programming are NP-hard.

Consequences

- Investigation of special cases
- Exact problem specific special purpose algorithms
- Design of special purpose heuristics



Contents

CO@W

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. **Algorithms for the solution of linear programs**
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



Algorithms for nonlinear programming

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

$$x_{i+1} = x_i + \lambda_i d_i$$

$d_i \sim$ "descent direction"

$\lambda_i \sim$ "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.



Contents

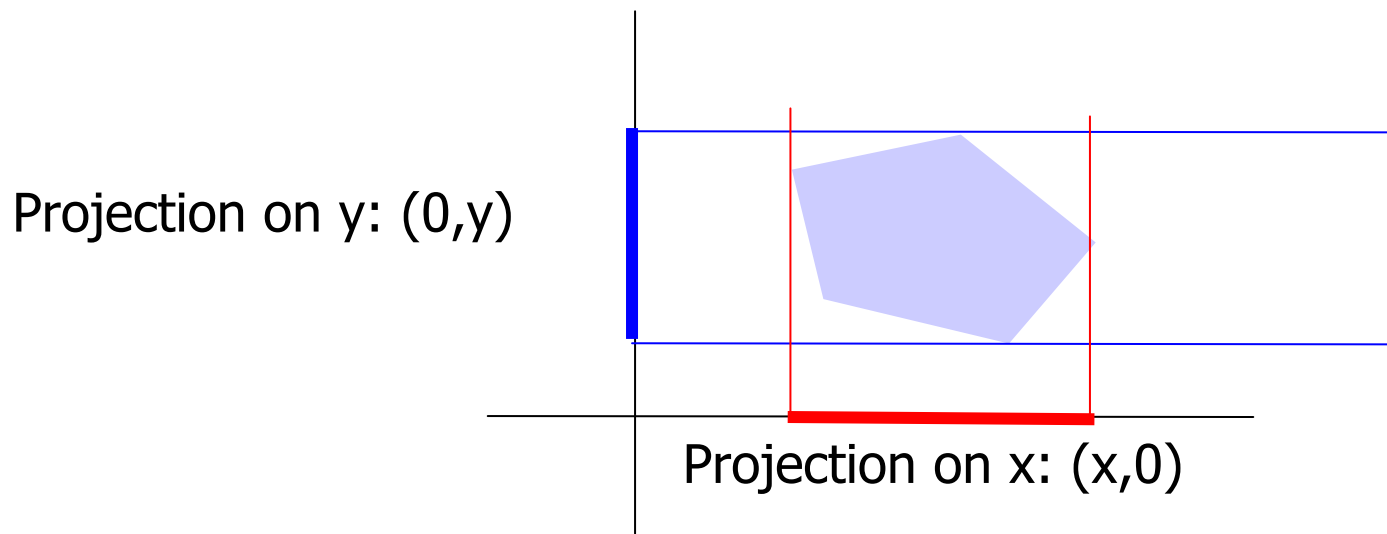
co@w

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) **Fourier-Motzkin Elimination**
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



Fourier-Motzkin Elimination

- Fourier, 1847
- 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.



Contents

co@w

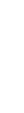
1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) **The Simplex Method**
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples

} will be covered by Bob Bixby
on Saturday



The Simplex Method

- Dantzig, 1947: primal Simplex Method
- Lemke, 1954; Beale, 1954: dual Simplex Method
- Dantzig, 1953: revised 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).



The Simplex Method: an example

CO@W

min/max $+ x_1 + 3x_2$

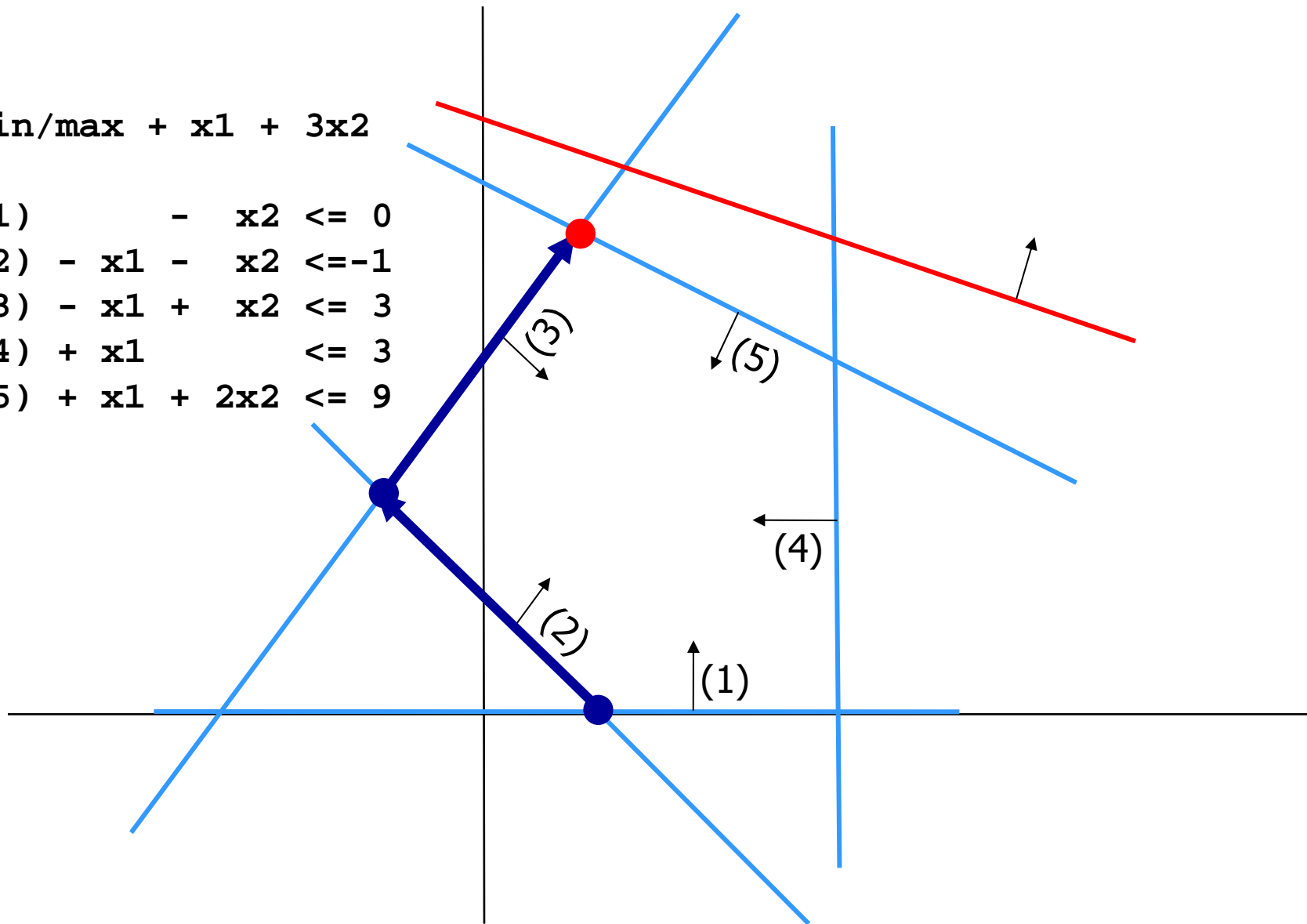
$$(1) \quad -x_2 \leq 0$$

$$(2) \quad -x_1 - x_2 \leq -1$$

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

$$(4) \quad +x_1 \leq 3$$

$$(5) \quad +x_1 + 2x_2 \leq 9$$



The Simplex Method: an example

CO@W

min/max $+ x_1 + 3x_2$

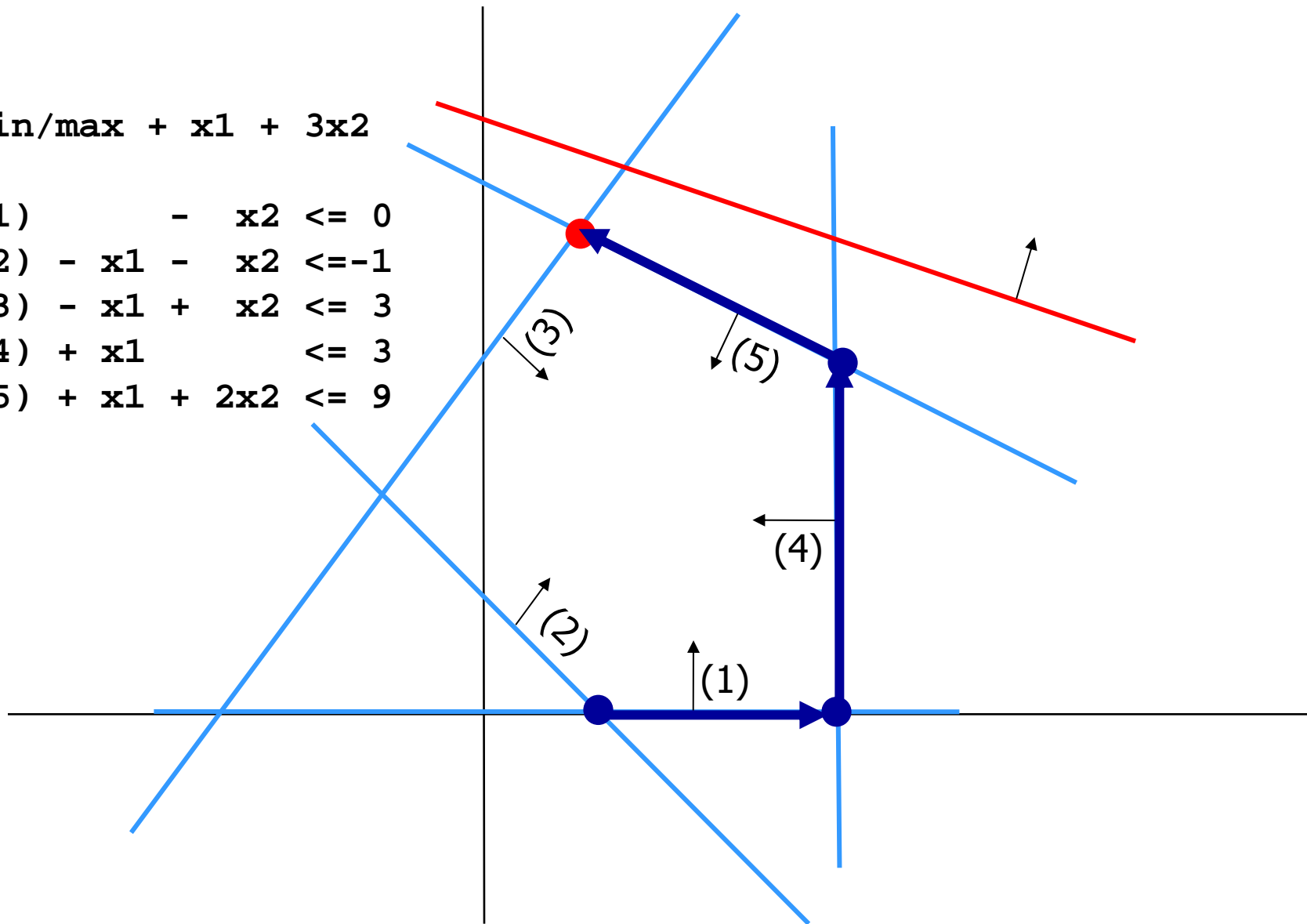
$$(1) \quad -x_2 \leq 0$$

$$(2) \quad -x_1 - x_2 \leq -1$$

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

$$(4) \quad +x_1 \leq 3$$

$$(5) \quad +x_1 + 2x_2 \leq 9$$



Hirsch Conjecture

CO@W

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).



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,

$$\max c^T x$$

$$Ax = b$$

$$x \geq 0$$

$$A =$$

B	N
---	---

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

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

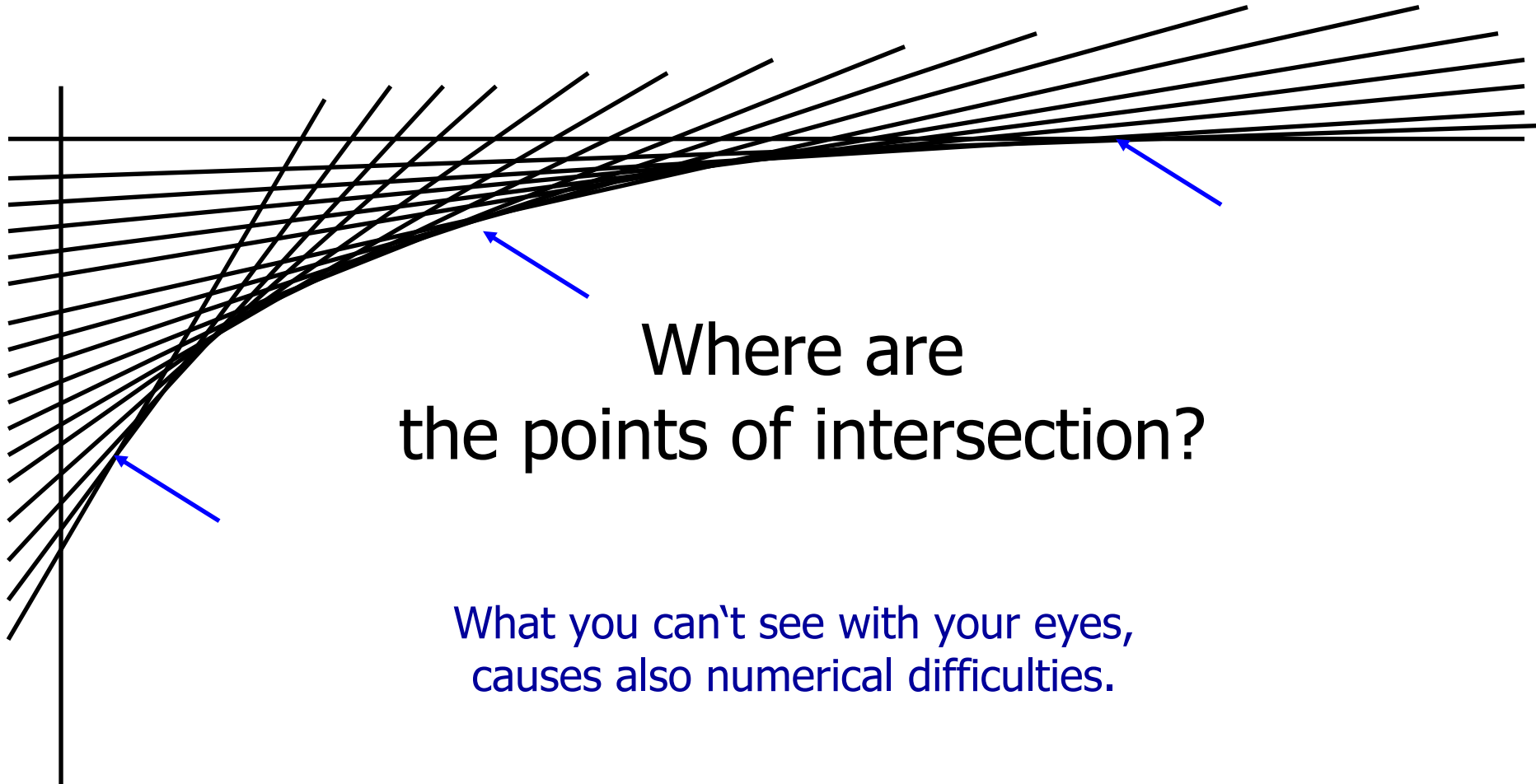
Many computational consequences:

Update-formulas, reduced cost calculations,
number of non-zeros of a vertex,...

You will hear a lot about that in other lectures.



Numerical trouble often has geometric reasons



Where are
the points of intersection?

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

Contents

CO@W

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) **The Ellipsoid Method**
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



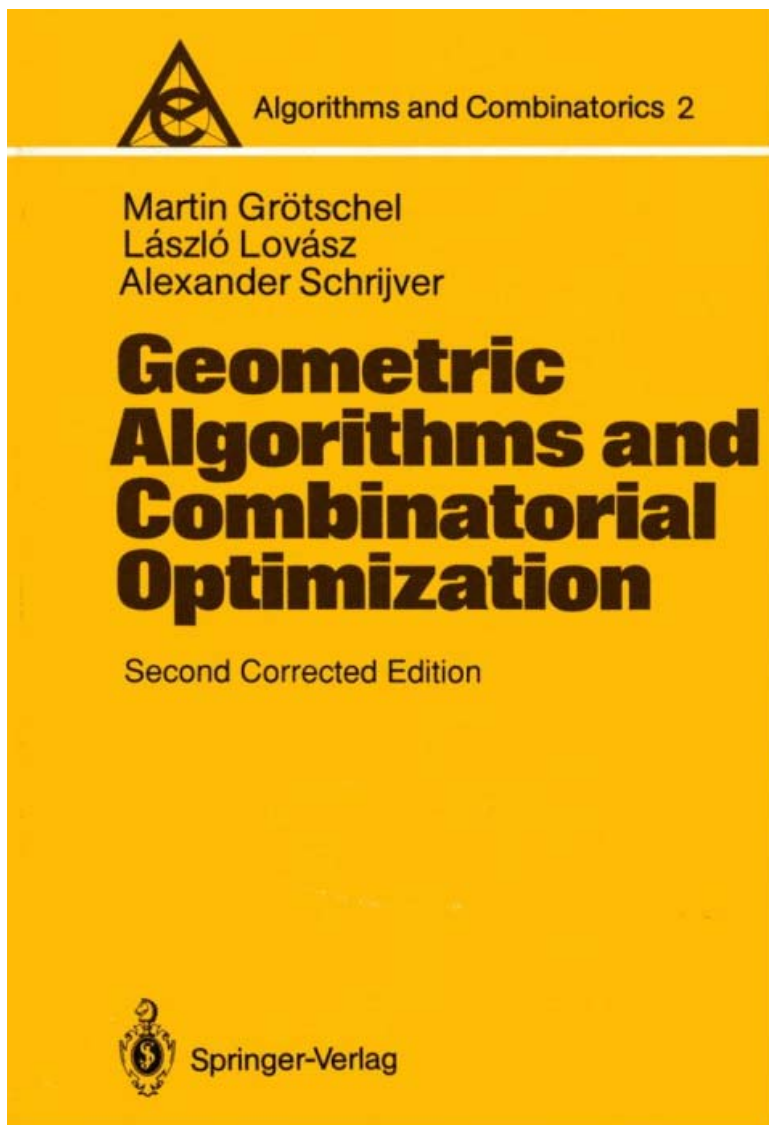
The Ellipsoid Method

- 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



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

CO@W

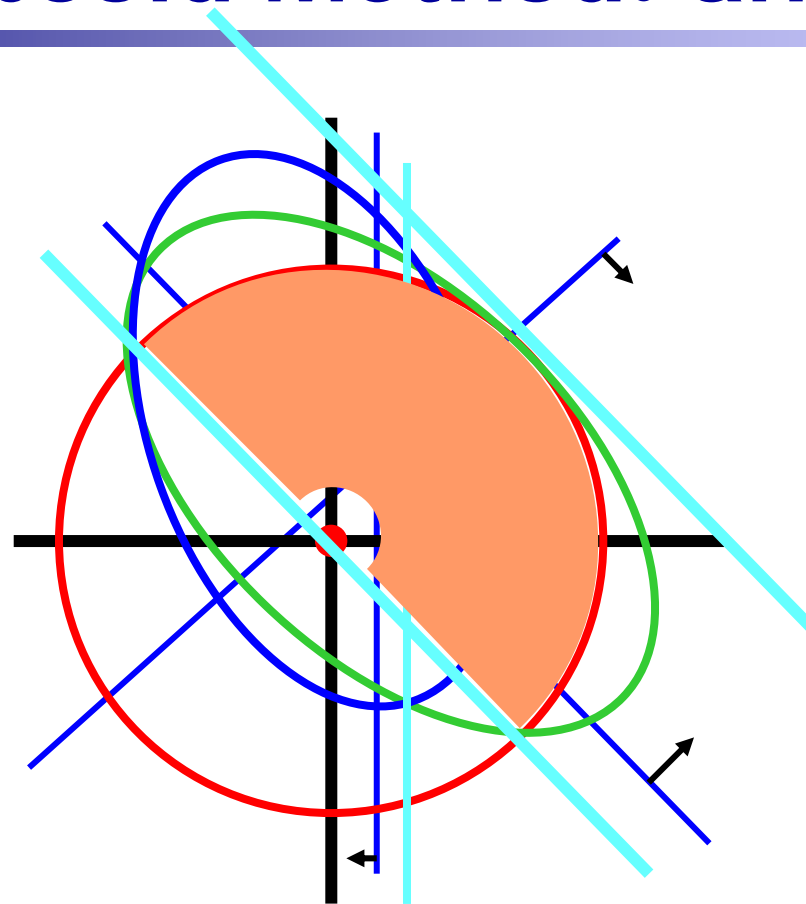


<http://www.zib.de/groetschel/pubnew/paper/groetschellovaszschrijver1988.pdf>



The Ellipsoid Method: an example

CO@W



$$k := 0,$$

$$N := 2n((2n + 1)\langle C \rangle + n\langle d \rangle - n^3)$$

$$A_0 := R^2 I \text{ with } R := \sqrt{n} 2^{\langle C, d \rangle - n^2}$$

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

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 $c^T x \leq \gamma$, of the system $Cx \leq d$ that is violated by a_k .

Cutting plane choice

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

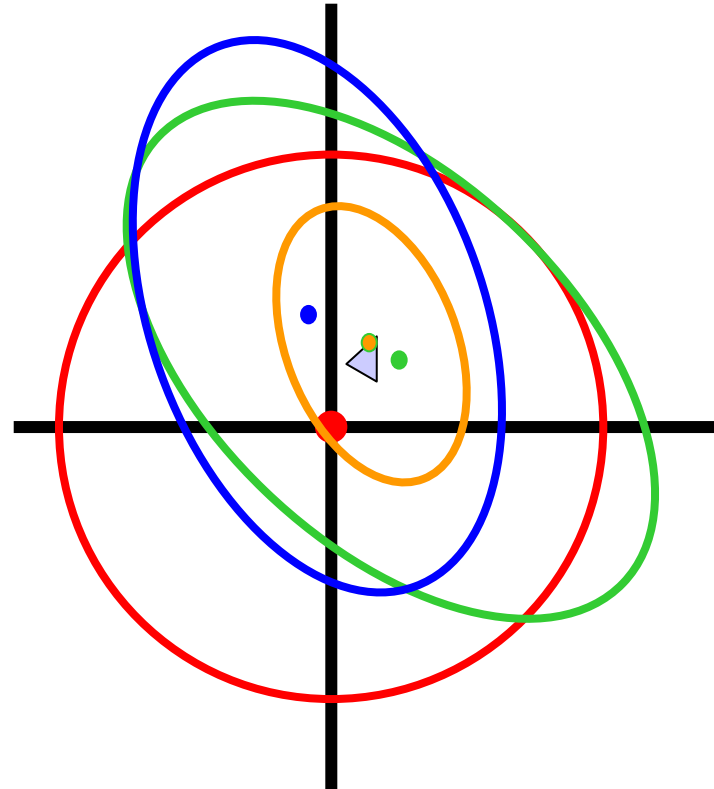
$$a_{k+1} := a_k - \frac{1}{n+1} b \quad \text{Update}$$

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

**The
Ellipsoid
Method**

Ellipsoid Method

CO@W

 $a(0)$ $a(1)$ $a(2)$ $a(7)$

feasible
solution
found



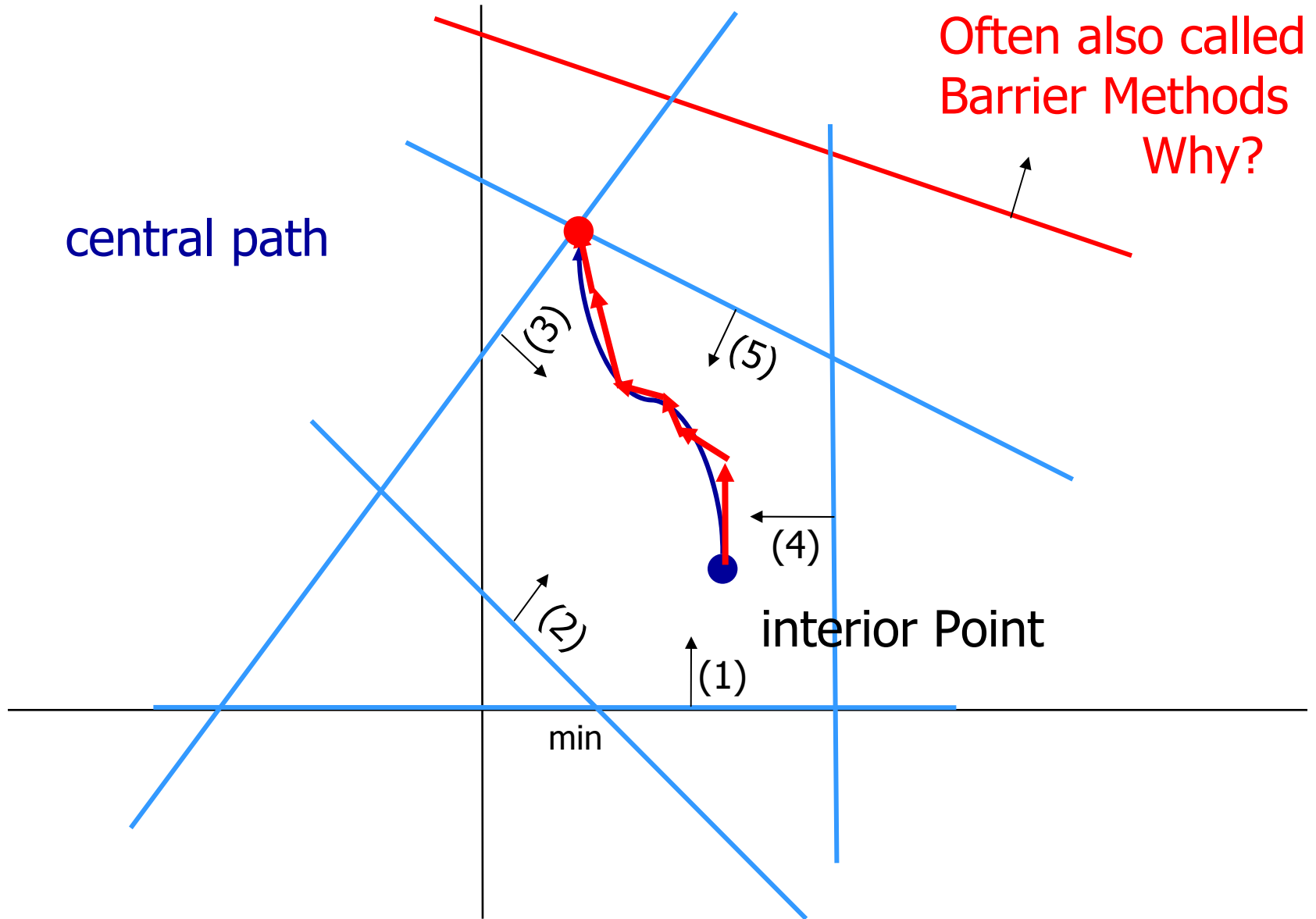
Contents

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



Interior-Point Methods: an example

CO@W



Other Approaches

- Lagrangean Relaxation
(for very large scale and structured LPs)
- plus
 - subgradient
 - bundle
 - bundle trust region

or any other nondifferentiable NLP method that looks promising

Lagrange relaxation, bundle methods, and their use in IP will be explained in detail in other lectures for special IPs.



Contents

co@w

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. **Algorithms for the solution of integer programs**
 - 1) Branch&Bound
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



special „simple“ combinatorial optimization problems

CO@W

Finding a

- minimum spanning tree
- shortest path
- maximum matching
- maximal flow through a network
- cost-minimal flow
- ...

solvable in polynomial time by special purpose algorithms



Dijkstra algorithm for shortest paths

co@w

```

1  function Dijkstra(Graph, source):
2      for each vertex v in Graph:           // Initializations
3          dist[v] := infinity              // Unknown distance function from source to v
4          previous[v] := undefined         // Previous node in optimal path from source
5      dist[source] := 0                    // Distance from source to source
6      Q := the set of all nodes in Graph
7      // All nodes in the graph are unoptimized - thus are in Q
8      while Q is not empty:                // The main loop
9          u := vertex in Q with smallest dist[]
10         if dist[u] = infinity:
11             break                        // all remaining vertices are inaccessible from source
12         remove u from Q
13         for each neighbor v of u:         // where v has not yet been removed from Q.
14             alt := dist[u] + dist_between(u, v)
15             if alt < dist[v]:             // Relax (u,v,a)
16                 dist[v] := alt
17                 previous[v] := u
18     return previous[]

```

http://en.wikipedia.org/wiki/Dijkstra's_algorithm

You will learn a lot more about shortest paths with side constraints in this course since such problems have to be solved in subroutines of cutting plane algorithms.



Special “hard” combinatorial optimization problems

CO@W

- travelling salesman problem
- location und routing
- set-packing, partitioning, -covering
- max-cut
- linear ordering
- scheduling (with a few exceptions)
- node and edge colouring
- ...

NP-hard (in the sense of complexity theory)

The most successful solution techniques employ linear programming.



Contents

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) **Branch&Bound**
 - 2) Cutting Planes
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



The Branch&Bound Technique: An Example

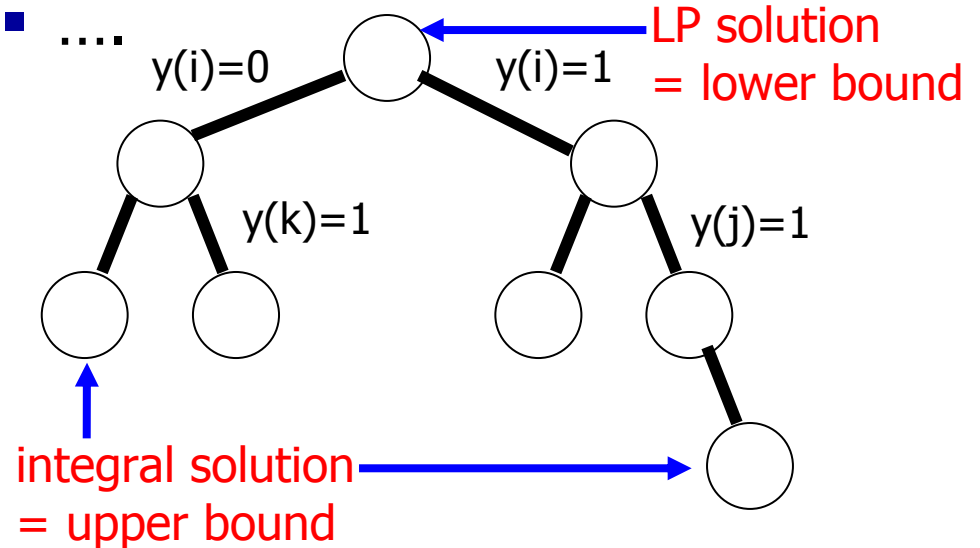
$$\begin{aligned} \min c^T x \\ Ax = a \\ Bx \leq b \\ x \geq 0 \\ x \in \{0,1\}^n \end{aligned}$$

0/1-
program

$$\begin{aligned} \min c^T x \\ Ax = a \\ Bx \leq b \\ x \geq 0 \\ \del{x \in \{0,1\}^n} \\ x \leq 1 \end{aligned}$$

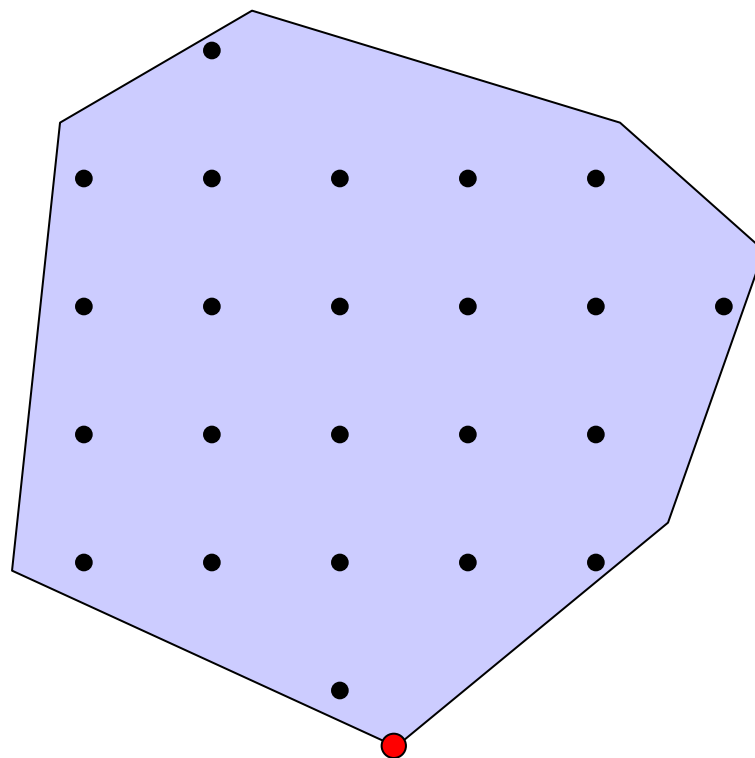
LP-
relaxation

- Solve the LP-relaxation and get optimal solution y . (lower bound)
- If y integral, DONE!
- Otherwise pick fractional component $y(i)$.
- Create two new subproblems by adding $y(i)=1$ and $y(i)=0$, resp.
-



Branching (in general)

CO@W

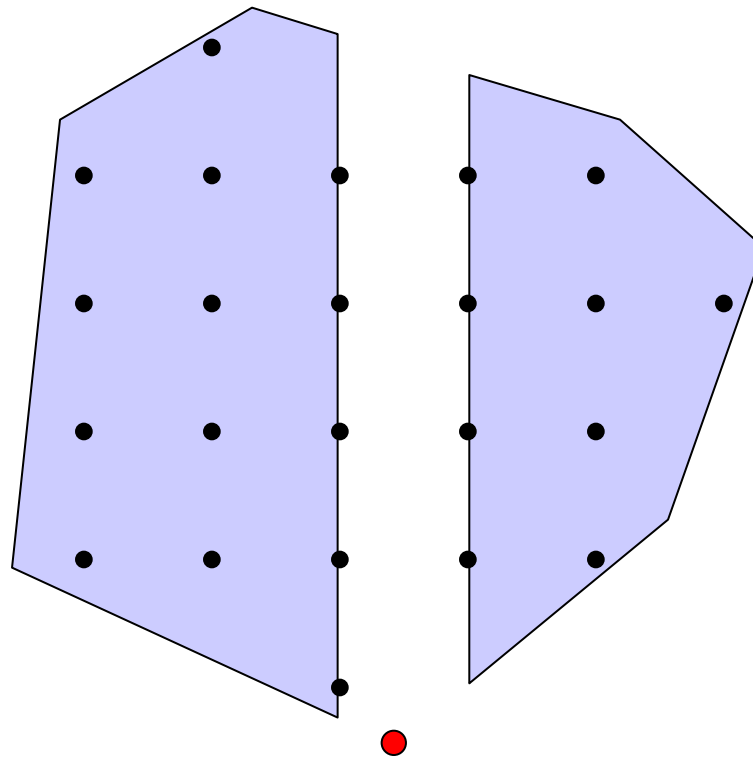


- Current solution is infeasible



Branching (in general)

- Rounding a fractional component up and down



- Decomposition into subproblems removes infeasible solution



A Branching Tree

co@w

Applegate

Bixby

Chvátal

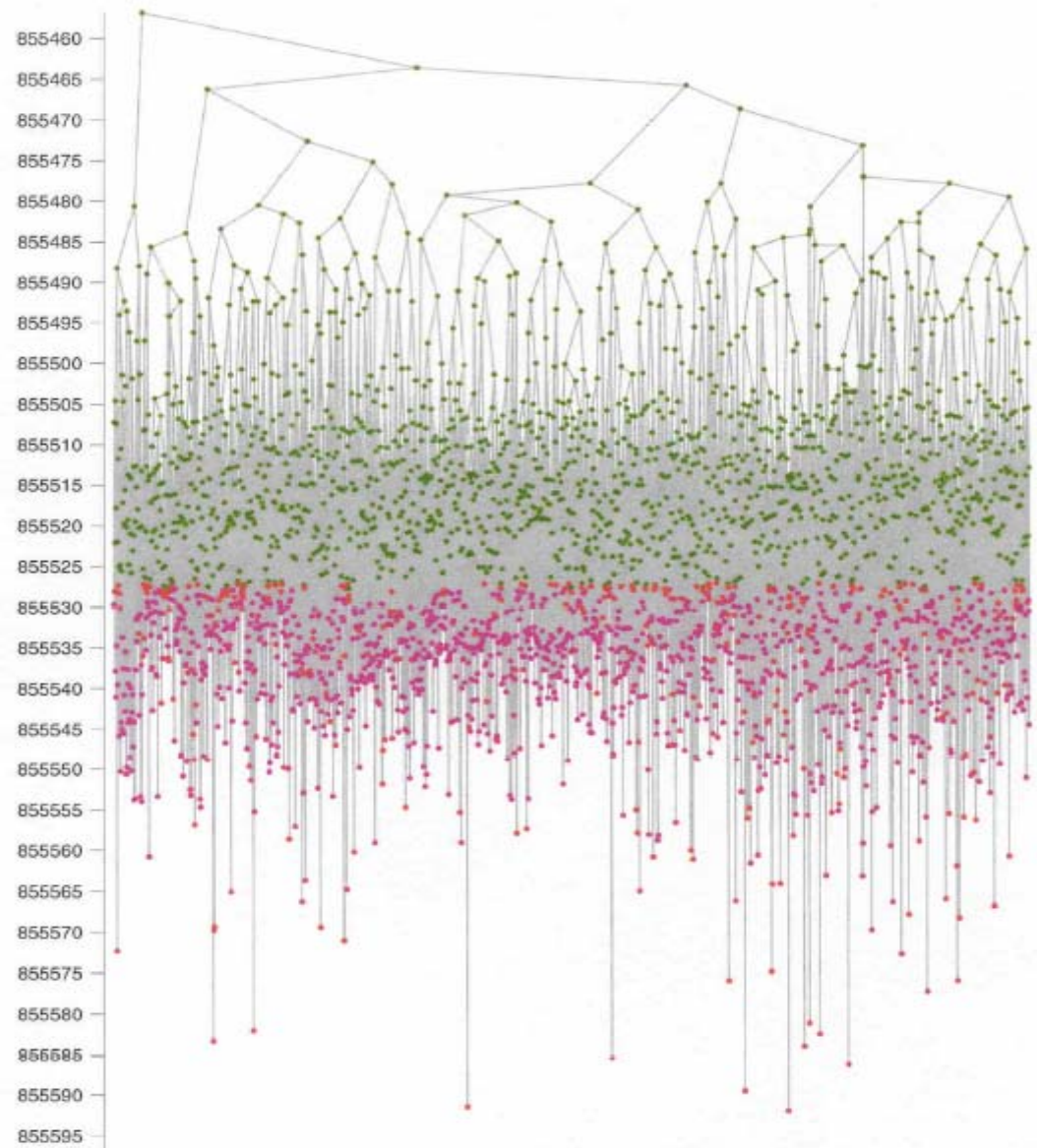
Cook

tree copied from

[www1.ctt.dtu.dk/ROUTE2003/
presentations/cook.pdf](http://www1.ctt.dtu.dk/ROUTE2003/presentations/cook.pdf)

sw24978 Branching Tree

Computation Carried out in Parallel at Georgia Tech, Princeton, Rice



Contents

CO@W

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) **Cutting Planes**
6. Where are we today?
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



Cutting plane technique for integer and mixed-integer programming

CO@W

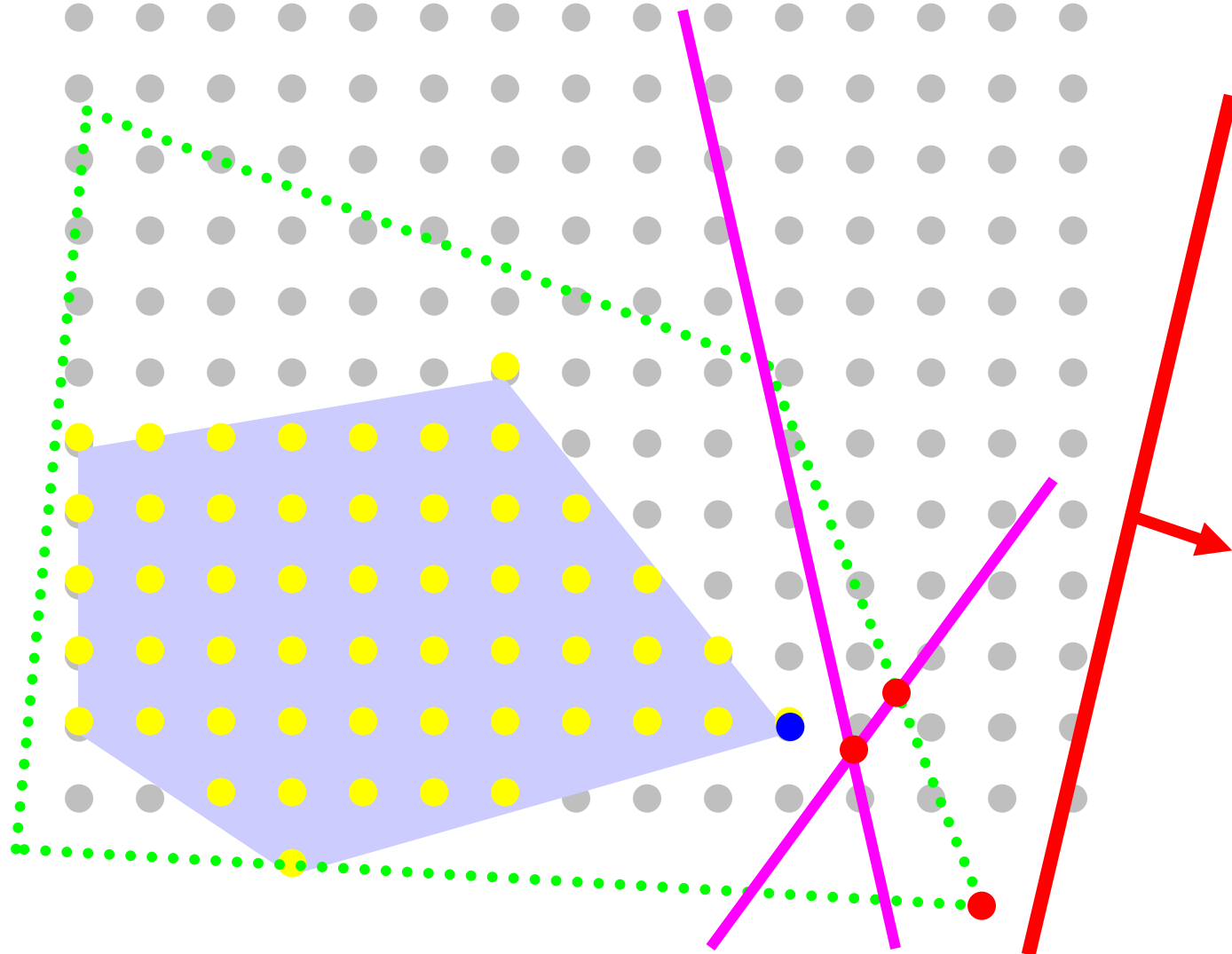
Feasible
integer
solutions

Objective
function

Convex
hull

LP-based
relaxation

Cutting
planes



Other Names

CO@W

- Branch & Cut
- Branch & Price
- Branch & Cut & Price
- etc.



Contents

CO@W

1. Linear, integer, nonlinear programming, optimization: What's that?
2. Historic roots
3. LP-Theory
4. Algorithms for the solution of linear programs
 - 1) Fourier-Motzkin Elimination
 - 2) The Simplex Method
 - 3) The Ellipsoid Method
 - 4) Interior-Point/Barrier Methods
5. Algorithms for the solution of integer programs
 - 1) Branch&Bound
 - 2) Cutting Planes
6. **Where are we today?**
 - 1) State of the art in LP
 - 2) State of the art in IP
 - 3) Examples



State of the art

CO@W

- You will learn the state of the art in this course by the world's topmost experts.
- It makes no sense to summarize the state now. Watch out for the lectures on SCIP,

Tu 22.09.

Introduction to the ZIB Optimization Suite

09:00-10:30 CR,AW

Porta and Polymake

11:00-12:30 IPG

Basic concepts of SCIP

14:00-15:30 IPG

The ZIB Optimization suite: SCIP, Soplex, Zimpl

16:00-16:30 IPG

Real world data

16:30-17:30 IPG

Refreshing C for SCIP programmers

- Bixby's lectures and the lectures on "commercial solvers day":



State of the art

CO@W

Th 24.09.

Advanced Linear Programming

09:00-10:30 BB

Solving LPs in practice

11:00-12:30 BB

14:00-15:30 BB

16:00-17:30 BB

Fr 25.09.

Advanced Mixed Integer Programming

09:00-10:30 BB

Solving MIPs in practice

11:00-12:30 BB

14:00-15:30 BB

16:00-17:30 BB

Tu 29.09.

(Commercial) Solvers: state-of-the-art

09:00-09:15 HM

Introduction

09:15-10:00 BB

Gurobi

10:00-10:45 TA

CPLEX

11:15-12:00 OB

Xpress

12:00-12:45 EA

MOSEK and Interior Point

14:15-15:00 TR

COIN-OR

15:30-16:15 EA

Conic Quadratic Optimization

16:30-17:30 MH

Integer Programming at Siemens



Linear and Integer Programming: an Introduction

CO@W Berlin

The End

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>

