New Trends in Combinatorial Optimization

by Achim Bachem, Bonn

Combinatorial optimization and decision problems

E (finite) $I \subseteq 2^E$ $f : I \to \mathbb{R}$ objective functions, $k \in \mathbb{R}$

Input: (I, f)	*Input:* (I, f, k)
task: find $S \in I$ with $f(S) = \max\{f(T) \mid T \in I\}$ or conclude that no such T exists.	*task:* decide: $\exists S \in I$ $f(S) > k$? yes or no

e.g. The travelling salesman problem (TSP)

Input: graph $G = (V, E)$ edge lengths c_e $(e \in E)$

find a shortest hamilton cycle in G.	Does G have a hamilton cycle of length > k ?

Theory of computational complexity

Problem: = Set of instances (E,f) + task

Instance: = (finite set E (feasible solutions), objective function $f : E \to \mathbb{R}$)

task: = find $x \in$ s.t. $f(x) = \max\{f(y) \mid y \in E\}$ or conclude that no such x exists.

Algorithm: = Turing machine (or recursive function)

Efficiency measures:

a) time complexity

b) space complexity

input: = instance (E,f)

input length: = binary encoding of instance (F,f)

time complexity of a problem $P = T$: <=>

$\exists$ algorithm A $\forall$ instances $(F,f) \in P$ time of A applied to $(F,f) \leq T$(input length of (F,f))

Complexity classes: *(for decision problems)*

$\mathbb{P}$: = {problems p | p has polynomial complexity}

EXP : = {problems p | p has exponential time complexity}

measures of hardness:

a) solve: say yes or no

b) verify: if yes, verify "yes is true"

NP: = {p | "yes" can be verified in polynomial time}

co-NP: = {p | "no" can be verified in polynomial time}

Theorem $\mathbb{P} \subseteq \mathbb{NP} \subseteq \text{EXP}$

$\quad\quad\quad\quad \neq? \quad \neq?$

$\mathbb{P} \stackrel{?}{=} \text{NP}$	$\text{NP} \stackrel{?}{=} \text{co-NP}$

$P, Q \in \mathbb{NP}$ P "$\leq$" Q <=> Q has time complexity T

=> P has time complexity

$g(T)$ with polynomial g.

Theorem $\forall P \in \mathbb{NP}$ P "$\leq$" TSP

$\mathbb{NP}$C: = {p $\in$ NP | P "=" TSP}

Criticism

(1) input length as a criterion

e.g. knapsack problem $\in$ NPC

$\min\{cx \mid ax \leq b, \quad x \in \{0,1\}^n\}$

$\exists$ algorithm with time $\leq$ poly(b), but b = exp(encoding of b) , i.e. time $\geq$ exp(log(b))

number and *non-number* problems

(2) worst case analysis?

(3) polynomial time complexity = "good" = "efficient"?

counter example: Khachian's algorithm but classification of problems

P is easy <=> $P \in \mathbb{P}$

P is hard <=> $P \in$ NPC

P is intractable <=> $P \in$ EXP$\setminus\mathbb{P}$

How many problems are classified?

1975 : $\geq$ 2000 problems in combinatorial optimization are identified NP - hard

1983 : $\geq$?

e.g. Scheduling Problems

easy	416	9%
open	390	9%
hard	3730	82%
total:	*4536*	

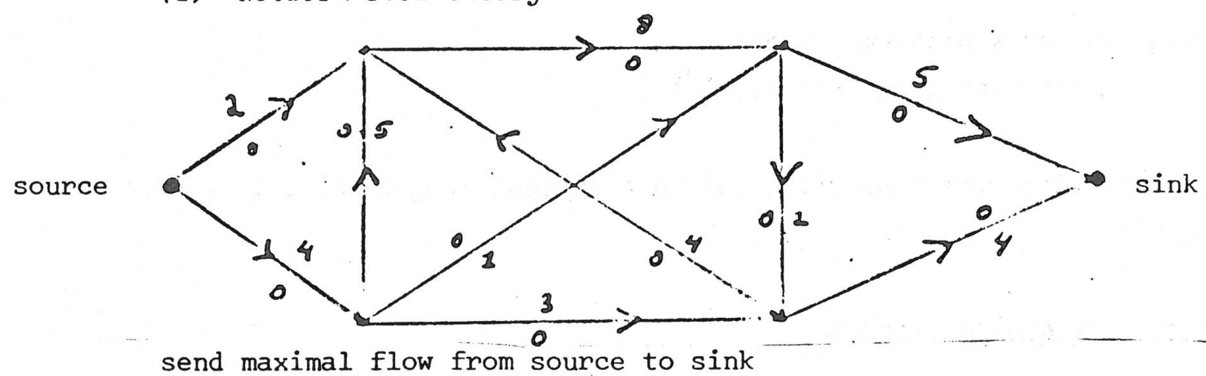
Theorem

The problem:

"determine the minimum number of research results that would completely resolve the status of all remaining open problems" is an NP-complete problem.

Some "easy" problems

(1) Network Flow theory



send maximal flow from source to sink

send minimum cost flow from source to sink

$P = \{x \mid Nx = 0 \; , \; l \le x \le u\}$ (integer data)

$$(1) \qquad \begin{array}{l} \max \; cx \\ x \in P \\ x \text{ integer} \end{array}$$

(network flow problem)

Th.:

(1) <=> $\max\{cx \mid x \in P\}$

Th.:

conv (P) has only integer vertices

Why? N is totally unimodular

Def.: An (m,n)-matrix is *totally unimodular* <=> every subdeterminant is 1, -1 or 0.

Def.: A polyhedron $P \subseteq \mathbb{R}^n$ is an *integer polyhedron* if every face ($\neq \emptyset$) contains an integer point.

Th.: If A is totally unimodular => $\forall b \in \mathbb{R}^m$ $Ax \leq b$ is an integer polyhedron

Which matrices are totally unimodular?

(1) Incidence matrices of digraphs

and else?

e.g. R_{10} $\begin{pmatrix} \quad \end{pmatrix}$

(Hoffman and Kruskal)

Seymour: R_{10} is the only additional example because:

Th.: If M is a regular (totally unimodular) matroid (not 1,2-, or 3-separable), then M is graphic, cographic or isomorphic to R_{10}.

(Edmonds: testing total unimodularity can be done in polynomial time)

When is $Ax \leq b$ an integer polyhedron for some special b ?

e.g. set packing polyhedron $Ax \leq 1$ for some 0 - 1 - matrix.

Def.: A is *perfect* <=> the graph with cliques incidence matrix A is a perfect graph.

Th.: If A is perfect then $Ax \leq 1$ is an integer polyhedron.

total dual integrality (TDI)

$Ax \leq b$ is TDI $\Leftrightarrow \forall c \in \mathbb{R}^n$ the dual $\min\{ub \mid uA = c, u \geq 0\}$ has an integer optimal solution

Th.: $Ax \leq b$ TDI, $b \in \mathbb{Z}^m \Rightarrow Ax \leq b$ integer polyhedron

Def.: $Ax \leq b$ has the integer round-up property (IRU) $\Leftrightarrow \min\{cx \mid Ax \leq b, x \in \mathbb{Z}^n\} = \lceil \min\{cx \mid Ax \leq b\} \rceil$

Th.: $Ax \leq b$ has IRU $\Leftrightarrow y(A,b) \leq 0$ is TDI

Which systems are TDI ?

$G = (V,E)$ digraph

$F \subseteq 2^E$ a crossing family (i.e. $S,T \in F$, $\emptyset \neq S \cap T$, $S \cup T \neq V \Rightarrow S \cap T, S \cup T \in F$)

$f : F \to \mathbb{R}$ submodular (i.e. $f(S \cap T) + f(S \cup T) \leq f(S) + f(T)$)

$S \subseteq V$ $\delta(S)$ = all arcs leaving S

Theorem (Edmonds/Giles)

The system $\sum_{e \in \delta(S)} x_e - \sum_{e \in \delta(V \setminus S)} x_e \leq f(S) \quad \forall\ S \in F$ is TDI .

=>

- max flow-min cut theorem
- Edmonds polymatroid intersection theorem
- min-max theorems for direct cut k-packings, k-covers
- blocking results for matroid intersection polyhedra
- ...
- ...

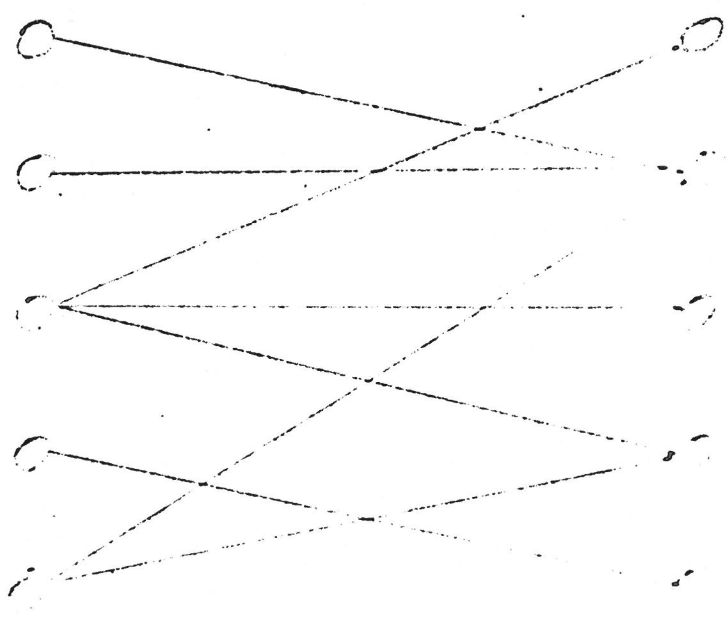
(2) Matching Problems

Def.: (b-matching problem)

given: graph $G = (V,E)$, $b \in \mathbb{N}^V$ "weights" c_{ij} $(i, j) \in E$
find: $S \subseteq E$ *(a b-matching)* s.t. every node $v \in V$ is incident with at most b_v edges and $\sum_{e \in S} c_e$ is maximal.

e.g. 1-matchings in bipartite graphs with weights $c_{ij} = 1$. (≡ *marriage problem)*

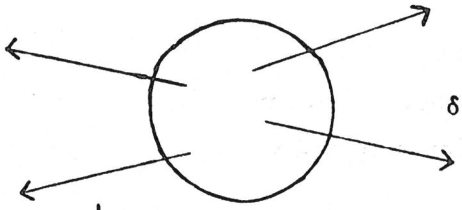
A has 5 daughters B has 5 sons



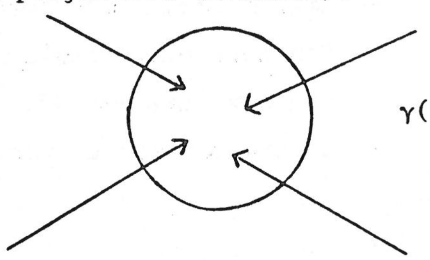
b-matching problems

- assignment problem (perfect weighted 1-matchings in bipartite graphs)
- perfect Gaussian elimination
- scheduling two processors
- heuristic for the travelling salesman polyhedron.

A linear characterization for the b-matching polyhedron (Edmonds):



$\delta(S)$



$\gamma(S)$

$$\sum_{e \in \delta(v)} X_e \leq b_v \qquad \forall\, v \in V$$

(*)

$$\sum_{e \in \gamma(S)} X_e \leq \Big(\sum_{e \in S} b_e - 1\Big)/2 \quad \forall\, S \quad V \text{ with } \sum_{e \in S} b_e \text{ odd} \quad X \geq 0$$

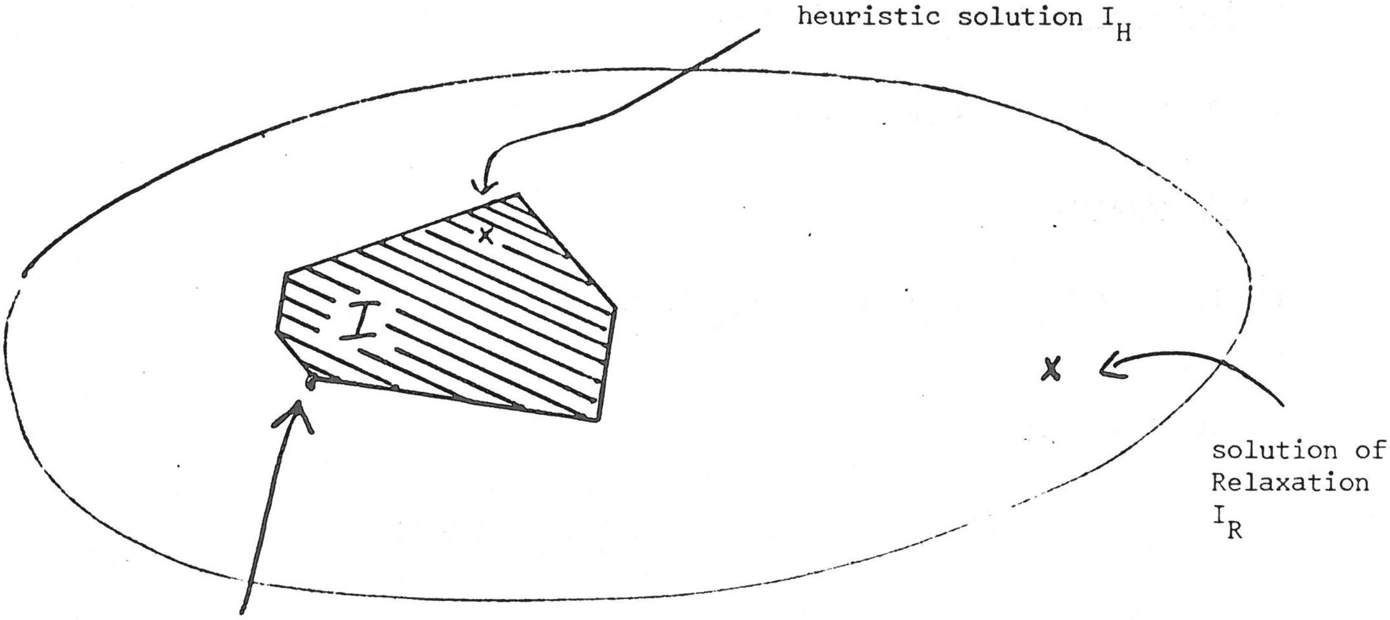
$$x \geq 0$$

(*) is an irredundant linear description of the b-matching polyhedron.

For $b = 1$ (*) is TDI

Solving hard combinatorial problems

$$\max \{\sum_{e \in S} C(e) \mid S \in I\}$$



Relaxation gives: $UB = C(I_R)$
Heuristic gives: $LB = C(I_H)$
thus

$LB \leq C(I_o) \leq UB$ and

relative Errors is $\left| \frac{C(I_H) - C(I_o)}{C(I_o)} \right| \leq \frac{UB - LB}{LB}$

Getting lower bounds

1) using heuristics
2) using approximative algorithms

and performance quarantee via

a) worst case analysis
b) probabilistic analysis

Heuristic algorithms

Most combinatorial problems can be (re-)formulated as an optimization problem over an independence system $I \subseteq 2^E$, i.e.

$I \neq \emptyset$ and $S \subseteq T \in I \Rightarrow S \in I$

Greedy algorithm

1) Let $E = \{e_1,\dots,e_n\}$ s.t. $C(e_1) \geq C(e_2) \geq \dots \geq C(e_n) \geq 0$
2) set $S = \emptyset$
3) Do $i = 1$ TO n

   IF $S \cup \{i\} \in I$ THEN $S := S \cup \{i\}$

   END.

Worst case analysis

$B \in I$ basis $\Leftrightarrow$ B has maximal cardinality

$$r^*(S) = \max\{|B| \mid S \subseteq B,\ B \text{ basis}\}$$

$$r_*(S) = \min\{|B| \mid S \subseteq B,\ B \text{ basis}\}$$

$$q(I) := \min_{S \subseteq E} \frac{r(S)}{r(S)} \qquad \text{rankquotient}$$

Theorem Let $C(I_g)$ be the objectivvalue of the Greedysolution I_g and $C(I_o)$ the value of the optimal solution I_o, then:

$$\boxed{C(I_o) \geq C(I_g) \geq q(I)C(I_o)}$$

Theorem

$\varphi(I) = 1 \iff I$ is a matroid

(e.g. spanning tree problem)

ε-approximation algorithms

Def.: If H is a heuristic for which always

$$\frac{C(I_H) - C(I_o)}{C(I_o)} \leq \varepsilon$$

(and $C(I_o) > 0$) holds, then H is called an ε-approximation algorithm

We then have: $C(I_H) \geq (1 - \varepsilon)C(I_o)$

Thus: Greedy -alg. = $[1 - \varphi(I)]$ -approximation alg.

e.g. uncapacitated location problem: greedy gives an e-approx.alg., i.e.
error ≤ 37%
minimum problems ≠ maximum problems e.g.

Theorem There exists a polynomial time ε-approx. alg. for the symmetric travelling salesman problem <=> P = NP
(true for all $0 < \varepsilon < 1$)

Theorem For every euclidean symmetric travelling salesman problem the following holds

$$\frac{\text{Length of Christofides tour}}{\text{length of optimal tour}} < 1{,}5$$

-approximation schemes

for a given input I and $\varepsilon > 0$ the alg. produces a heuristic solution with

$$\frac{C(I_H) - C(I_o)}{C(I_o)} \leq \varepsilon$$

If the running time is bounded by a polynomial in the length L of the input and in $1/\varepsilon$ we have a

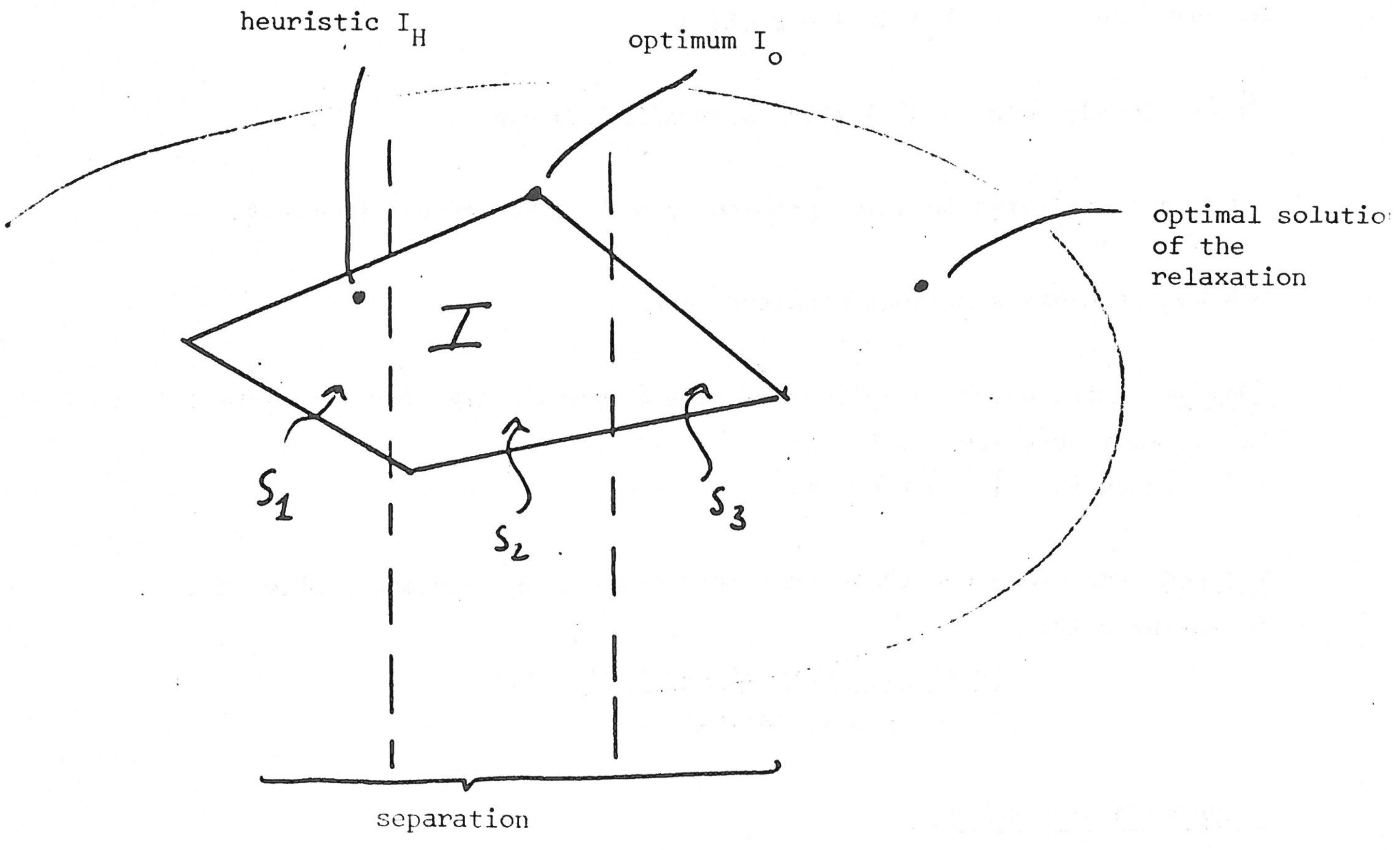
fully polynomial approximation scheme (FPAS)

e.g. : FPAS ex. for knapsack problems

Def.: A problem is a number problem , if there is no polynomial "poly" with $C(I_o) \leq poly(L)$

Theorem. If a non-number problem is NP-complete it has a FPAS <=> P = NP.

Getting upper bounds



Branch & Bound with

a) Lagrangean Relaxation (using subgradient techniques)
b) cutting plane algorithms

Lagrangean Relaxation

$$\text{IP} \quad \left| \begin{array}{l} \min\ cx \\ Ax = b \\ Dx \leq d \\ x \geq 0,\ \text{integer} \end{array} \right.$$

$$\text{LR}_u \quad \left| \begin{array}{l} f(u) : \min cx + u(Ax - b) \\ \qquad\qquad Dx \leq d \\ \qquad\qquad x \geq 0\ ,\ \text{integer} \end{array} \right.$$

Problem : choose optimal relaxation $f(u^*) = \max\limits_{u \in \mathbb{R}^n} f(u)$

thus choose u^* s.t. $0 \in \partial f(u^*)$

$\partial f\ (u) = \{\pi \in \mathbb{R}^m \mid \pi = \Sigma\mu_t(Ax^t - b)\ ,\ \Sigma\mu_t = 1\ ,\ \mu_t \geq 0\}$

Alg:

$u^o = 0\ ,\ i = 0$

$0 \in \partial f(u^i) \rightarrow$ stop u^i is optimal

choose $\pi^i \in \partial f(u^i)$

choose $t^i \in \mathbb{R}$ and set $u^{i+1} := u^i + t^i \pi^i$

$i = i + 1$

Th.: If $t_i > 0 \quad t_i \rightarrow 0 \quad \sum\limits_1^{\infty} t_i = \infty$ then $f(u^i) \quad \max\limits_{u \in \mathbb{R}^m} f(u)$

Beispiel: (Symmetrisches Travelling Salesman P) geschickte Formulierung des STSP als ganzzahliges Programm liefert:

$$\min \sum_{1 \leq i < j \leq n} C_{ij} x_{ij}$$

$$\boxed{\sum_{i<j} x_{ij} + \sum_{j<i} x_{ij} = 2 \; ; \; i = 2,\ldots,n} \qquad Ax = b$$

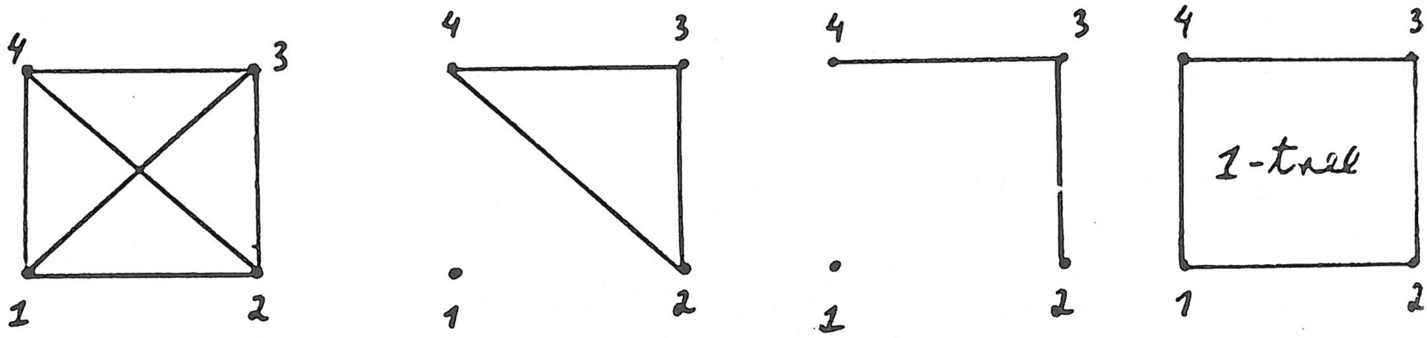
$$\boxed{\begin{array}{l} \sum_{j=2} x_{ij} = 2 \; , \quad \sum_{1 \leq i < j \leq n} x_{ij} = n \\ \sum_{ij E(W)} x_{ij} \leq |W| - 1 \quad \forall \, W \subseteq \{2,\ldots,n\} \\ x_{ij} \leq 1 \; , \; x_{ij} \geq 0 \; , \; x_{ij} \text{ ganzz.} (1 \leq i < j \leq n) \end{array}} \qquad \begin{array}{l} Dx \leq e \\ x \geq 0 \\ x \text{ ganzz.} \end{array}$$

min cx

$Dx \leq e$ <=> Bestimme einen minimalen 1-tree in K_n (Prim-Dijkstra Methode)

$x \geq 0$

x ganzzahlig



also Bestimmung von $x^t (t \in \{1,\ldots,T\})$, so daß $Ax^t - b \in \partial f(u)$ für $t \in eq(u)$ ist "einfach"

Schrittweite für Subgradientenverfahren: $t := \dfrac{\lambda(UB - LB)}{\sum_{k=2}^{n} (d_k - 2)^2}$

mit $0 < \lambda \leq 2$ und d_k = Knotengrad des k-ten Knoten des letzten optimalen 1-tree.

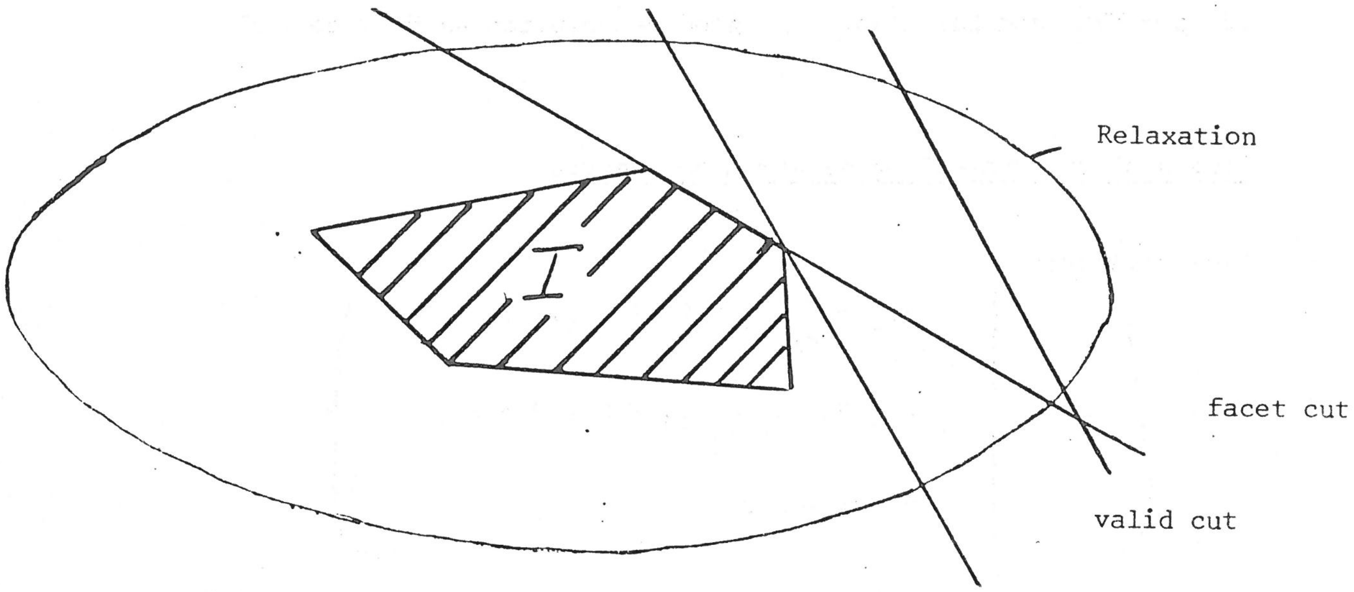
Held & Karp (1970)

Helbig Hansen & Krarup (1974)

Smith & Thompson (1977)

Volgenant & Ionker (1980)

Cutting plane algorithms



(1) classic cutting plane algorithms (Gomory...)

- using only valid inequalities (thus very slow convergence)

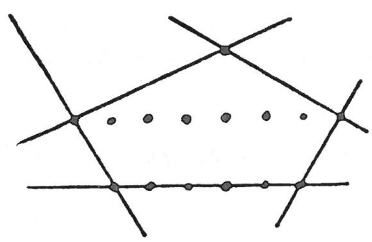
(2) advanced cutting plane algorithms

- using facets as cuts

Problem: a) characterization of the facets

b) separation using facets

Linear characterization



$I \subseteq 2^E$ $\quad E = \{1,\dots,n\}$

$I^1 := \{x \in \{0,1\}^n \mid \text{support}(x) \in I\}$

$P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is a *linear characterization* of I $\Leftrightarrow$ $\text{conv}(I^1) = P$

I has a *good* description: $\Leftrightarrow$ given: $dx \leq d_o$ "$\forall x \in I^1\ dx \leq d_o$" $\in NP$

Theorem (Karp/ Papadimitriou (1980))

If $p \in NPC$ and has always a good description $\Rightarrow NP = co-NP$

Facets of the travelling salesman polyhedron

TSP-polyhedron $P =$

$$\operatorname{conv}\left\{ x \;\middle|\; \begin{array}{l} \sum_{i<j} x_{ij} + \sum_{j<i} x_{ij} = 2 \quad i = 2,\dots,n \\ \sum_{2} x_{1j} = 2 \;,\; \sum_{i<j} x_{ij} = n \;,\; 0 \leq x_{ij} \leq 1 \\ x_{ij} \text{ integer} \end{array} \right\}$$

Th.: $\dim P = 1/2(n-3)n$ (STSP)

How many facets are known for the n-city STSP?

for $n = 120$ more than 10^{179} !

e.g. "easy" facets: comb inequalities

for $n \geq 6$ let $W_o, W_1, \dots, W_k \subseteq V$ (the cities)

s.t. (a) $|W_o \cap W_i| \geq 1 \quad i = 1,\dots,k$

(b) $|W_i \setminus W_o| \geq 1 \quad i = 1,\dots,k$

(c) $|W_i \cap W_j| = \emptyset \quad 1 \leq i < j \leq k$

(d) $k \geq 3$, odd

then $\sum_{i=0}^{k} \sum_{e \in E(W_i)} x_e \leq |W_o| + \sum_{i=1}^{k} (|W_i| - 1) - \lfloor \frac{k}{2} \rfloor$ is a facet of the STSP .

For $n = 59$ there are more than $2 \cdot 10^{74}$ comb facets.

e.g. "hard" facets

$G = (V,E)$ graph

G is traceable <=> G has a hamiltonian cycle

G is hypotraceable <=> (a_1) G is not traceable

(a_2) $\forall v \in V$ $G - v$ is traceable

G is maximal hypotraceable <=> (a_1) G is hypotraceable

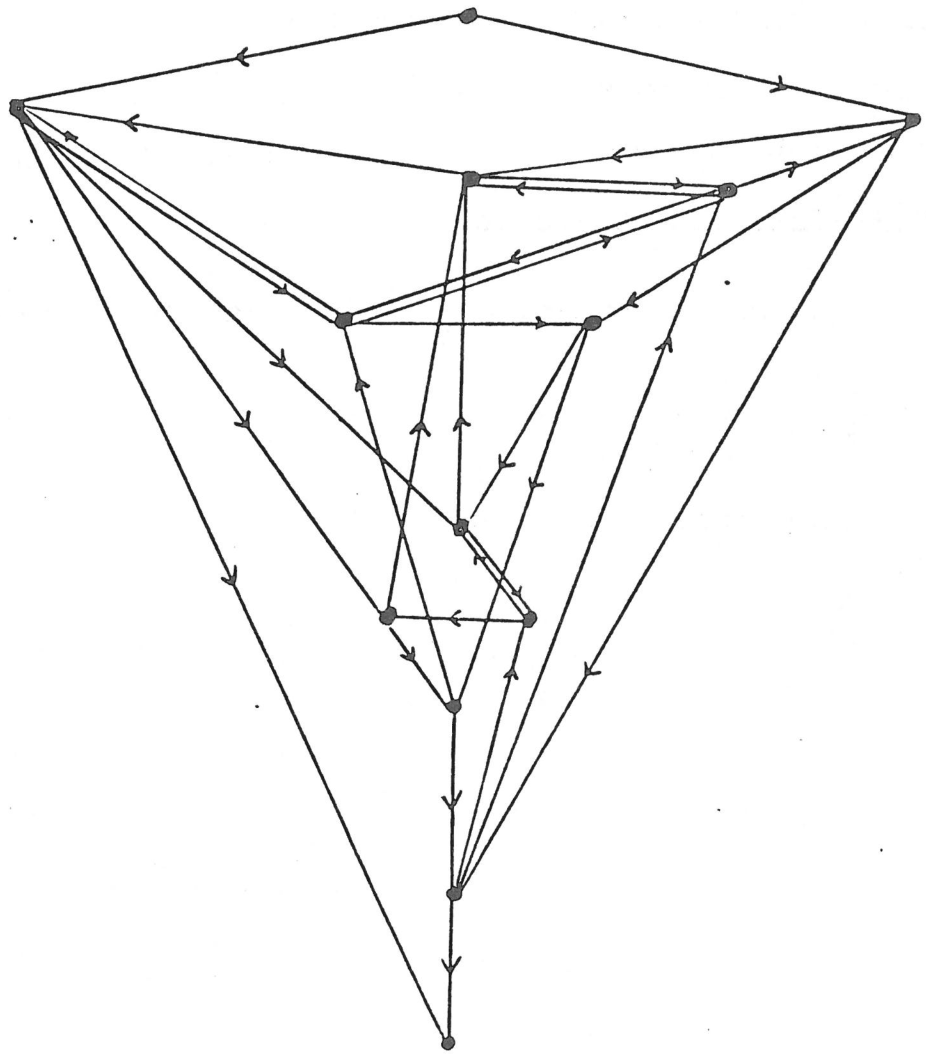
(a_2) $\forall e \notin E$ $G + e$ is traceable

Theorem

Let $G = (V,E)$ be a maximal hypotraceable graph with $|V| = k$. Then

$$\sum_{e \in E} x_e \leq k - 2$$ is a facet of all STSP with $n \geq k$ "cities".

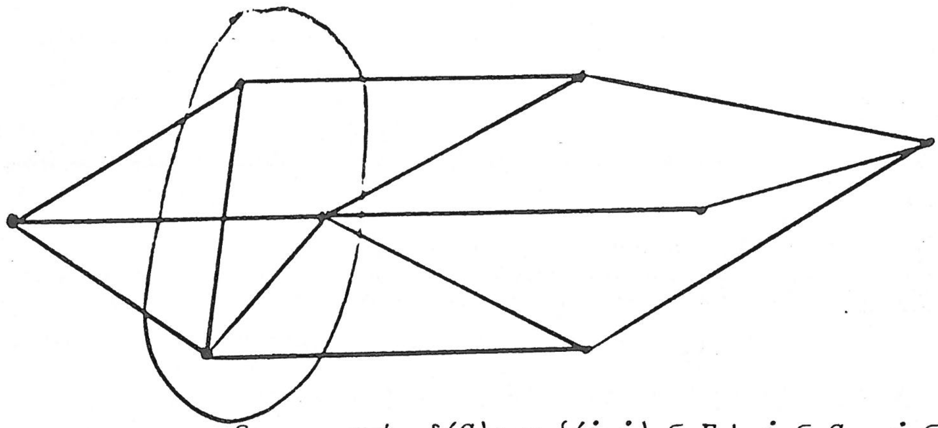
R.: The smallest hypotraceable graph known is of order 34.



Γ_{12}

The Max-Cut Problem

$G = (V,E)$



S   *cut* $\delta(S) := \{(i,j) \in E \mid i \in S,\ j \in V \setminus S\}$

$c_e > 0 \quad (e \in E)$ edge weights

Problem: find a cut $\delta(S)$ in G such that $c(\delta(S)) = \sum_{e \in \delta(S)} c_e$ is

as small as possible	as large as possible
$\in \mathbb{P}$	$\mathbb{P}$-*complete*

$\delta(S)$ cut $\Rightarrow$ $(V, \delta(S))$ bipartite subgraph of G

$F \subseteq E \quad \Leftrightarrow \quad$ 0 - 1 - vector $x^F \in \{0,1\}^E$

$P_B(G)$ = all incidence vectors of the edge sets of bipartite subgraph of G

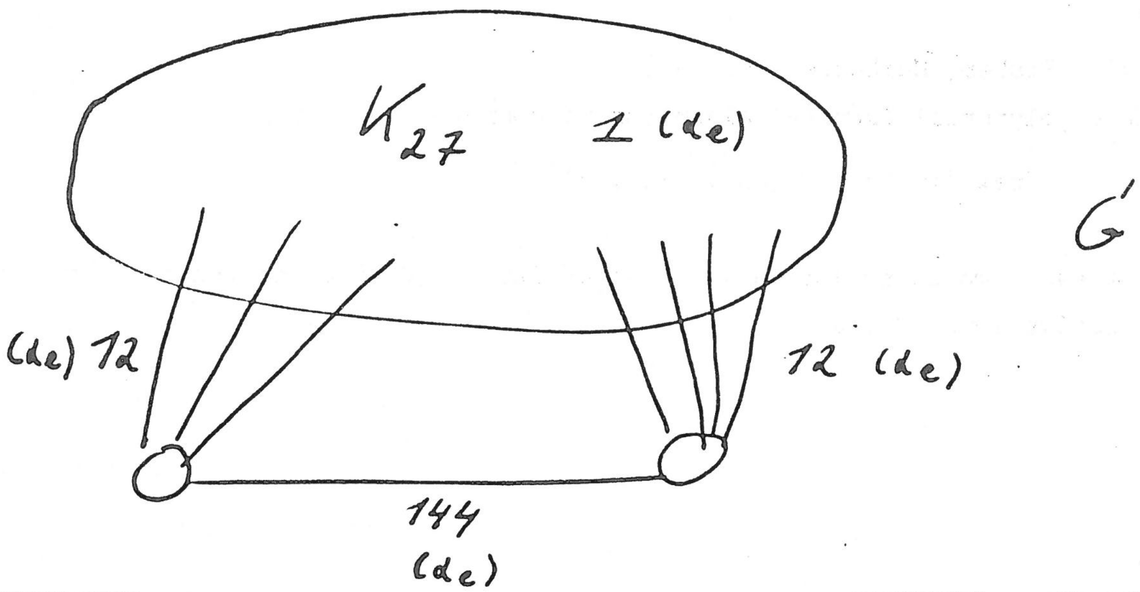
= *bipartite subgraph polytope*

Prop.: For positive edge weights $c_e (e \in E)$, every optimum basic solution of the LP

$$\max cx$$
$$x \in P_B(G)$$

is the incidence vector of a cut of G.

A facet for $P_B(G)$ $G = K_{29}$



$$\sum_{e \in E(G^1)} x_e \leq 2k(k+1) \qquad 2k+1 \mid V(G^1) \mid$$

Submodular setfunctions in combinatorial optimization

Discrete optimization	Continuous optimization
submodular	*convex*
supermodular setfunct.	*concav* *function*
modular	*linear*

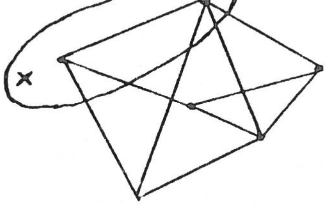
Def.: $f : 2^E \to \mathbb{R}$ is submodular <=>

$$f(X \cup Y) + f(X \cap Y) \leq f(X) + f(Y)$$

Examples

(a) $r(X)$ rank of a matrix formed by columns in X

(b)



$\delta(X)$ = # of edges entering X

(c) $A = (A_{ij})$ $X \subseteq \{1,\dots,n\}$ $\varphi(X) = \sum_{i=1}^{n} \max_{j \in X} A_{ij}$

"plant location problem"

(3) (Fisher, Nemhauser, Wolsey)

$\exists$ a polynomial *(simple)* algorithm to compute

$$\max \{\varphi(X) \mid X \subseteq S, |X| \leqq k\}$$

for every monotone increasing, submodular, $f(\emptyset) = 0$ setfunction f, with relative error $< 1/e$.

Submodular<~>convex

(1) $\varphi : 2^S \to \mathbb{R}$, $x \geq 0$ $\hat{\varphi}(x) = \Sigma\, \lambda_i \varphi(S_i)$

where $S_o \subset S_1 \subset \ldots \subset S_k \subseteq S$ $\lambda_i < 0$

$x = \Sigma\, \lambda_i$ (incidence vector of S_i)

Then $\hat{\varphi}$ is convex <=> φ is submodular

(2) $f : \mathbb{R}_+ \to \mathbb{R}$, S finite , $\check{f}(X) = f(|X|)$ for $X \subseteq S$

Then $\check{f}$ is submodular <=> f is concave

Results

(1) (Grötschel, Lovasz, Schrijver)
∃ polynomial time algorithm to minimize a submodular setfunction!

(2) (Frank)
(a) Let $f, g : 2^S \to \mathbb{R}$, $f \leq g$,

f supermodular

g submodular

$\Rightarrow\ \exists\, h : 2^S \to \mathbb{R}$, $f \leq h \leq g$

h modular

(b) If f,g are integral valued, then ∃ integral valued h.

The Ellipsoid Method

problem: find a polynomial time algorithm for solving the linear programming problem:

$$\max cX$$
$$Ax \leq b$$

Borgwardt (1977) : Simplex algorithm has average complexity $O(n^4 m)$ pivot operation

Klee-Minty (1972) : Simplex algorithm needs $O(2^n)$ pivot operations in the worst case

feasibility test <=> optimization

find $\bar{x} \in \mathbb{R}^n$ s.t. $A\bar{x} \leq b$ <=> find $\bar{x} \in \mathbb{R}^n$ $c\bar{x} = \max\{cx \mid Ax \leq b\}$

because : duality theorem of linear programming

$$x \text{ optimal} \iff \begin{cases} A\bar{x} \leq b \; \} \text{ primal feasible} \\ \left.\begin{array}{l} \bar{u}^T A = c^T \\ \bar{u} \geq 0 \end{array}\right\} \text{ dual feasible} \\ c\bar{x} \leq \bar{u}b \; \} \text{ inv. weak duality} \end{cases}$$

Relaxation	*Subgradient*
Motzkin, Schoenberg (1954)	Shor (1962)
$0 < \lambda \leq 2$	$f(x) := \max(A_i x - b_i)$
step 0 : $x^o = 0$, $i = 0$	step 0 : $x^o = 0$, $i = 0$
step 1 : $Ax^i \leq b$ stop	step 1 : $0 \in f(x^i)$ stop
step 2 : sei $A_k x^i > b_k$	step 2 : sei $g^i \in f(x^i)$, $\lambda_i \geq 0$
step 3 : $x^{i+1} = x^i - \lambda \frac{[A_k x^i - b_k] A_k}{\lVert A_k \rVert}$	step 3 : $x^{i+1} = x^i - \frac{\lambda_i g^i}{\lVert g^i \rVert}$
step 4 : $i = i + 1$, go to 1	step 4 : $i = i + 1$, go to 1

($\lambda = 1$ Projektion von x^i auf $\{x \in \mathbb{R}^n \mid A_k x = b_k\}$)

Δ geometrische Konvergenz $\qquad$ Δ lineare Konvergenz

<u>Satz</u> (Jeroslow (1977))

Gilt $\forall x \in \mathbb{R}^n (Ax \leq b \Rightarrow \lVert x \rVert \leq D)$, dann findet die Relaxation method mit $\lambda = 1$ ein $\bar{x}$ mit $A\bar{x} \leq b + \varepsilon$ in $O(D^2/\varepsilon^2)$ Iterationen.

$$= \exp \underbrace{(\log \lfloor \varepsilon \rfloor + 1)}_{= \text{ Kodierungslänge von } \varepsilon}$$

Levin's Methode (Levin (1965))

K (kompakt, konvex) $\subseteq \mathbb{R}^n$ $f : K \to \mathbb{R}$ stetig

$\min \{ f(x) \mid x \in K \}$

step 0 : $K_o := K$, $i = 0$

step 1 : berechne den Schwerpunkt x^i von K_i

step 2 : falls grad $f(x^i) = 0$ oder $K_i = \{x^i\}$ stop , sonst

step 3 : $K_{i+1} = K_i \cap \{x \in \mathbb{R}^n \mid (x - x^i)^T \text{ grad } f(x^i) \leq 0\}$

step 4 : $i = i + 1$, go to 1

Satz: (Mitjagin (1969))

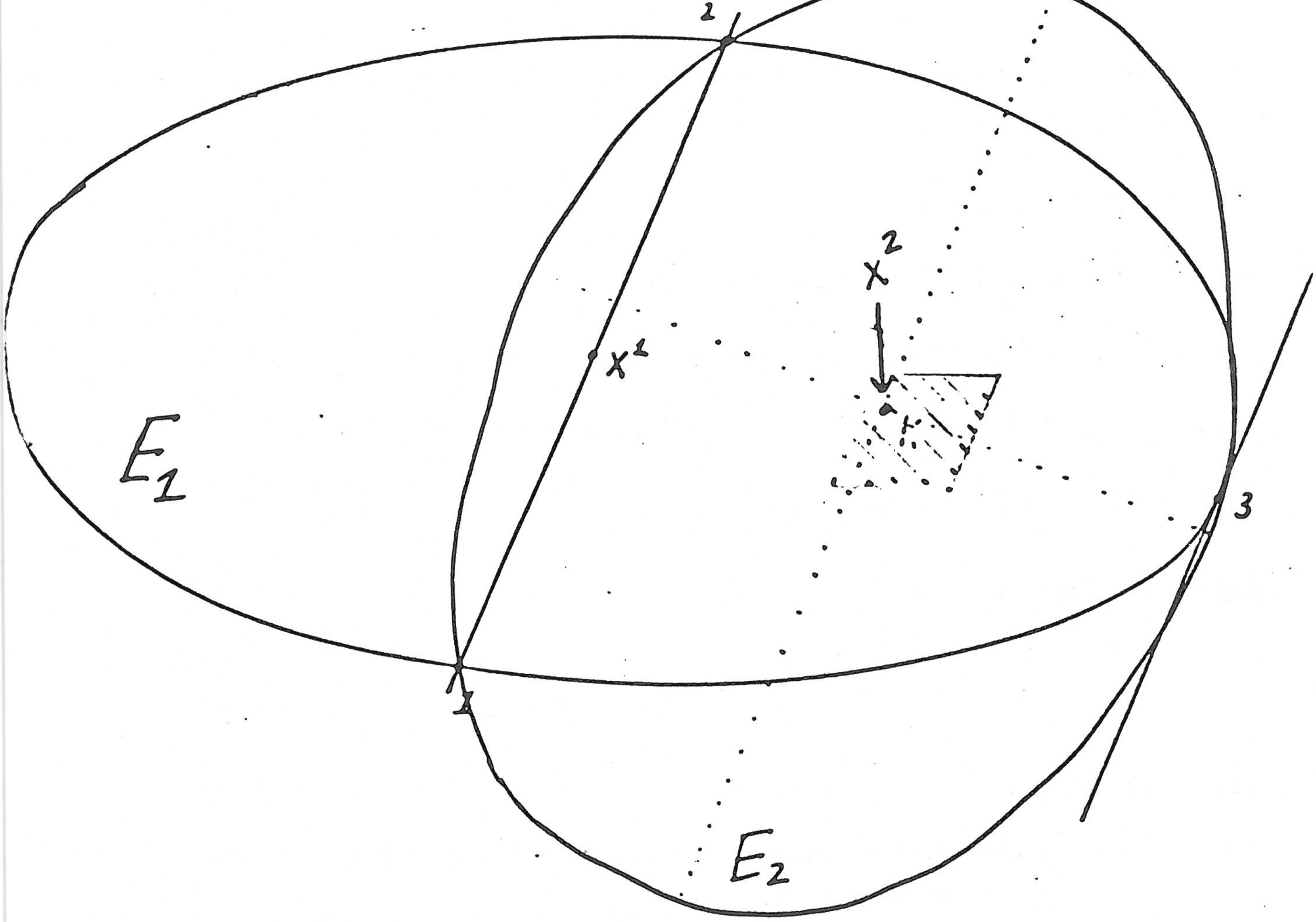
$$\text{vol}(K_i) < (1 - e^{-1})^i \text{vol}(K_o)$$

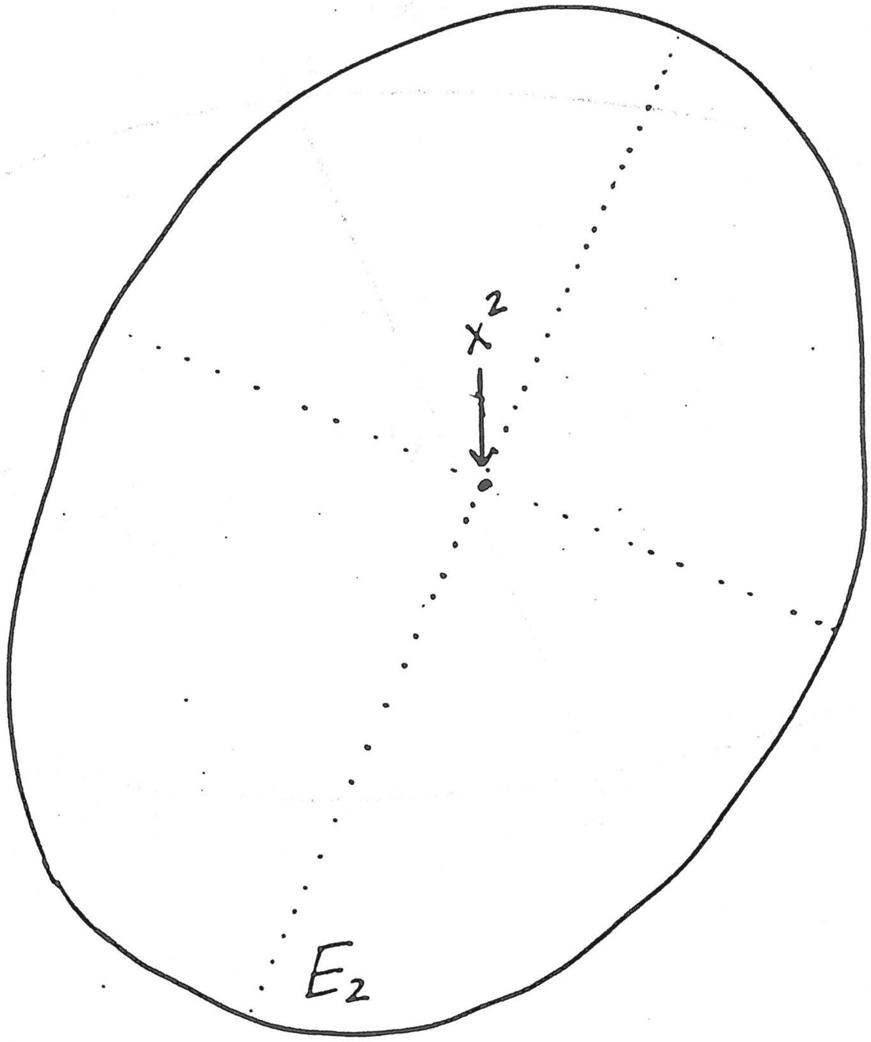
Ellipsoid Methode

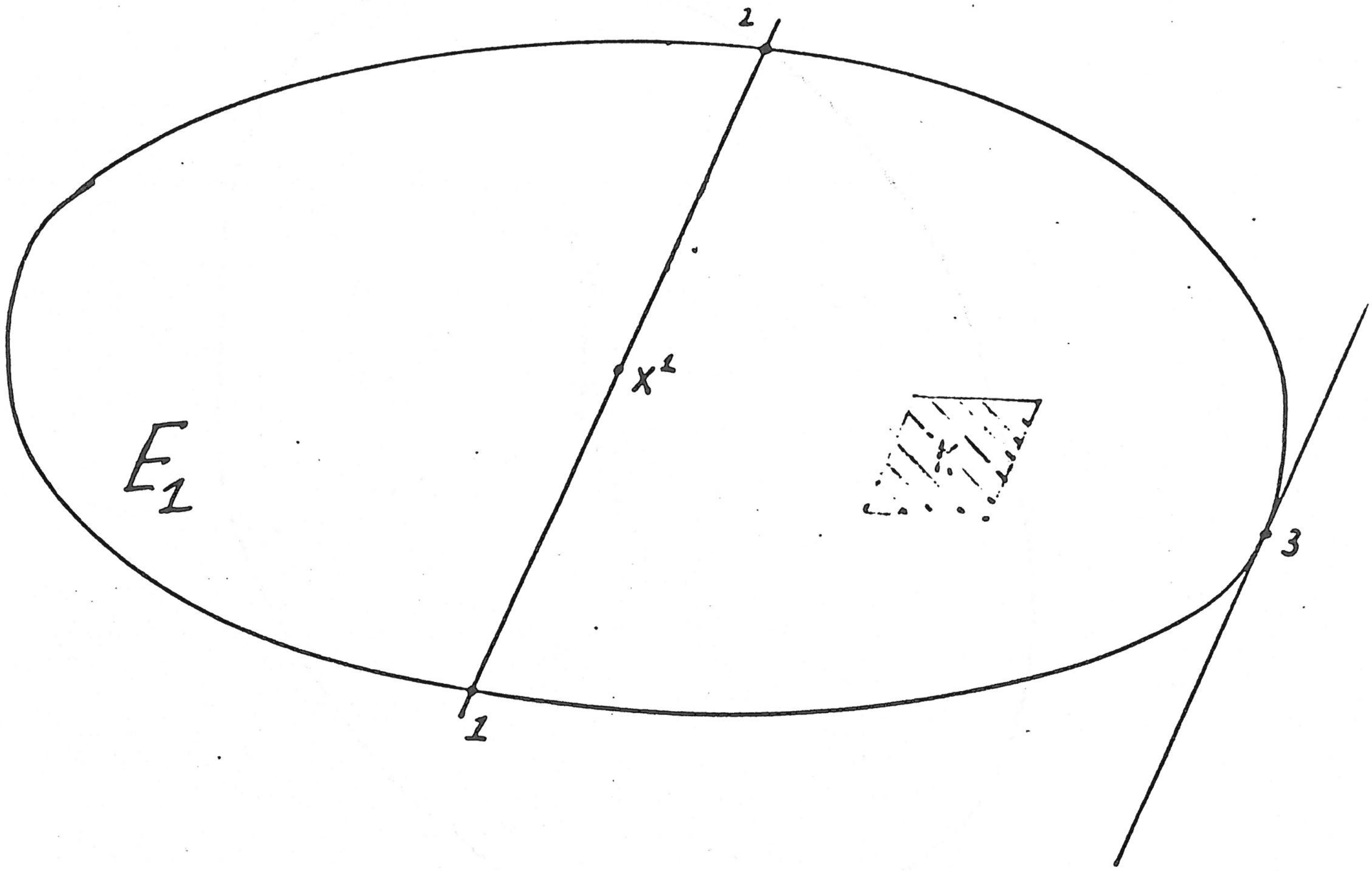
Shor (1970) : Subgradient relaxation + Störung des Subgradienten mit pos. definiter Matrix B.

Judin und Nemirowskii (1977) : Shor's Algorithmus = Levin's Methode mit K als Ellipsoid.

Ellipsoidalgorithmus




x^2
E_2



2
x^1
E_1
1
3

The weak optimization problem

given : convex body $K \subseteq \mathbb{R}^n$, $c \in \mathbb{Q}^n$, $\varepsilon \in \mathbb{Q}_+$

find : a "solution" vector $y \in \mathbb{Q}^n$ such that $d(y,K) \leq \varepsilon$ and

$\forall x \in K \quad cx \leq cy + \varepsilon$

(y is "almost" in K and "almost" maximizes cx over K)

Assumptions

K is a convex body and

(a) $n \in \mathbb{N}$ with $K \subseteq \mathbb{R}^n$

(b) $r, R \in Q$ with $0 < t \leq R$ and $a_o \in Q^n$ with

$\text{ball}(a_o, r) \subseteq K \subseteq \text{ball}(a_o, R)$ are givens as inputs.

How is the convex body K given?

(c) *weak separation oracle*

oracle input: $y \in Q^n$, $\delta \in Q_+$

oracle output: Either: yes, $d(y,K) \leq \delta$

or: the vector $c \in Q^n$ with $\|c\| \geq 1$ defines

an "almost" separating hyperplane, i.e.

$\underbrace{\forall x \in K \quad cx \leq cy + \delta,}$

$:=$ *subroutine* *Sep(K,y,δ)*

Ellipsoidalalgorithmus (Grötschel + Lovasz + Schrijver)

Input: $a_o, n, r, R, \varepsilon$ (so daß $S(a_o, r)$ K $S(a_o, R)$

Output: $y \in \mathbb{Q}^n$ mit $d(y,K) \leq \varepsilon \wedge cx \leq cy + \varepsilon \quad \forall x \in K$

Step 0: Setze $N := 4n^2 \lceil \log(2R^2 \|C\| / + \varepsilon) \rceil$

$\delta := R \cdot 4^{-N/2} / 300n$

$p := 2N + \lceil \log(8(n+1) / R^2)$

$x_o := a_o$, $A_o := R^2 I_n$ (n×n Einheitsmatrix)

Lemma : $\xi_k := \max\{cx_j \mid 0 \leq j \leq k , j$ zulässiger Ind.$\}$

$K_k := K \cap \{x \in \mathbb{R}^n \mid cx \geq \xi_k\}$

dann gilt : $K_k \subseteq E_k$

Satz Sei $0 \leq j < N$ mit $cx_j = \max_{k=0}^{N-1} (cx_k \mid k$ zul. Ind.)

dann gilt : $_j \geq \max\{cx \mid x \in K\} - \mathcal{E}$.

d.h. x_j löst das (schwache) Optimierungsproblem.

Theorem The ellipsoid algorithm stops after polynomial many (binary) operations with respect to the length of the (binary) encoded input a_o , n, r, R, E provided the weak separation oracle is solvable in polynomial time.

Application to LP

$\max\ cx$ $Ax \leq b$	$\max\ cx$ $x \in \text{conv}(v^1,\dots,v^t)$
separation: easy optimization: ?	optimization: easy separation: ?

Separation ≡ Optimization

Theorem Let K be a class of convex bodies.
There exists a polynomial algorithm for solving the weak optimization problem for all $P \in K$

<=>

There exists a polynomial algorithm for solving the weak separation problem for all $P \in K$.

Step 1 : FOR $k = 0$ TO $N - 1$ DO ;

(1) call Sep (K, x_k, δ)

gilt $d(x_k, K) \leq \delta$ so heißt k <u>zulässiger Index</u> ; setze <u>$a := a$</u>

liefert Sep (K, x_k, δ) $\quad d \in \mathbb{Q}^n$ mit $dx \leq dx_k + \delta$

$\forall x \in K$, so heißt k <u>unzulässiger Index</u> ; setze <u>$a := -d$</u>

(2) $b_k := A_k a / \sqrt{a^T A_k a}$

$x_k^* := x_k + (1/n + 1) b_k$

$A_k^* := \frac{2n^2+3}{2n^2} (A_k - \frac{2}{n+1} b_k b_k^T)$

$x_{k+1} :\approx x_k^* \qquad A_{k+1} :\approx A_k^*$

Rundung nach p Binärstellen

END ;

$\underline{E_k} := \{x \in \mathbb{R}^n \mid (x - x_k) A_k^{-1} (x - x_k) \leq 1\}$

<u>Lemma</u> :

(1) A_k $\quad k = 0,\ldots,N$ sind pos.

(2) $\|x_k\| \leq \|a_o\| + R \cdot 2^k$

(3) $\|A_k\| \leq R^2 \cdot 2^k$ $\qquad$ für $k = 0,\ldots,N$

(4) $\|A_k^{-1}\| \in R^{-2} \cdot 4^k$

<u>Lemma</u> : $\quad \mathrm{vol}(E_{k+1}) < e^{-1/4n} \mathrm{vol}(E_k)$

Khachiyan's idea

How can we use the ellipsoid method to solve the LP : $\max\{cx \mid Ax \leq b\}$ in polynomial time?

Input of LP : $L := \Sigma \lfloor \log A_{ij} \rfloor + 1 + \ldots$

Input of ellipsoid method : a_o, n, R, with $\text{ball}(a_o, r) \subseteq \{x \mid Ax \leq b\} \subseteq \text{ball}(a_o R)$

time of ellipsoid method is polynomial in
$\log |a_o| + \log |n| + \log |r| + \log |R| + \log |\varepsilon|$

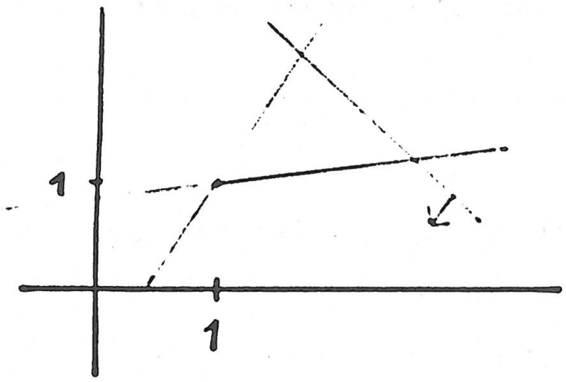
How do we get r, R, with $\log |r| + \log |R| = \text{poly}(L)$?

Theorem Let $P = \{x \mid Ax \leq b\}$ be a polytope and α the largest entry (in absolute value) of A, c and b . Then

$\text{ball}(a_o, t) \subseteq P \subseteq \text{ball}(a_o, R)$ holds with

$$r = \frac{1}{\sqrt{n}} \alpha^{-n^4 - 2n^2} \qquad R = 2\sqrt{n}\,\alpha^n .$$

Numerische Probleme



$$\mu x_1 + (\mu + 1)x_2 \leq -1$$

$$-\mu x_1 + (\mu - 1)x_2 \leq -1$$

$$\mu x_1 + \mu x_2 \leq 2\mu + 1$$

$\mu = 2^{16}$ L = 0,12E3

Iteration	x_1	x_2	min EW	max EW
10	-0,18E18	-0,18E18	0,16E29	0,15E36
20	-0,24E16	-0,24E16	0,29E23	0,24E32
30	-0,92E13	-0,92E13	0,59E18	0,73E27
40	0,11E12	0,11E12	0,96E13	0,33E23
50	-0,35E9	-0,35E9	0,49E9	0,5E18
60	-0,13E7	-0,13E7	0,13E5	0,15E14
70	0,17E5	0,17E5	0,21	0,73E9
80	0,5E2	0,5E2	0,11E-4	0,11E5
100	0,1E1	0,1E1	0,48E-14	0,16E-4
113	1	1	0,12E-18	o,8E-10

A new class of polynomial time solvable problems

e.g. (1) stable set problem
(2) clique problem
(3) colouring problem
(4) clique covering problem
in perfect graphs.

$G = [V,E]$ graph, $W_v \in \mathbb{Z}_+$ $(v \in V)$ "weights" $\alpha_w(G)$ = max weight of a stable set in $\rho_w(G)$ = min weight of a clique covering in G. G is perfect $\Leftrightarrow$ $\forall w \in \mathbb{Z}_+$ $\alpha_w(G) = \rho_w(G)$

Th.: The problem :
given: a graph $G = (V,E)$ and weights $w_v \in \mathbb{Z}_+$ $(v \in V)$ and $k \in \mathbb{Z}_+$
decide: $\alpha_w(G) > k$?
is NP-complete.

Minimizing submodular functions

E (finite) $F \subseteq 2^E$ a crossing family, i.e.

$\emptyset \neq S \cap T$, $S \cup T \neq E \Rightarrow S \cap T$, $S \cup T \in F$

$f : F \to \mathbb{Z}$ submodular , i.e.

$f(S \cup T) + f(S \cap T) \leq f(S) + F(T) \quad \forall S,T \in F$

Theorem If $f : F \to \mathbb{Z}$ is submodular, then $\min\{f(S) \mid S \in F\}$ is solvable in polynomial time using the ellipsoid method,
provided :
(1) $\exists B \in \mathbb{Z}$ $\forall S \in F$ $|f(S)| \leq B$ and $f(S)$ can be evaluated in polynomial time with respect to $|E|$ and $\log |B|$.

$B(G) := \{B = (b_{ij}) \mid B$ symm.pos.def. (n,n) - matrix, trace $(B) = 1$,

$b_{ij} = 0 <=> (i,j) \in E(G)\}$

$V(G) := \max \{\sum_{i,j} b_{ij} \mid B \in B(G)\}$

<u>Theorem</u>

G is perfect $<=> \forall G' \subseteq G \quad \alpha(G') = V(G')$

<u>Theorem</u>

Let $_n := 1/nI_n \in B(G)$, then

(1) $B(G)$ is a convex body

(2) $\text{ball}(\beta_n, 1/n^2\sqrt{n}) \subseteq B(G) \subseteq \text{ball}(\beta_n, \sqrt{\frac{n-1}{n}})$

<u>Thus:</u> Ellipsoid method can be used to compute $V(G)$

separation oracle for $B(G)$

"<=>" test of positive definitenen of an (n,n) - matrix

(polynomial time algorithm with Gauss-elim. + Cholesky decomp.)