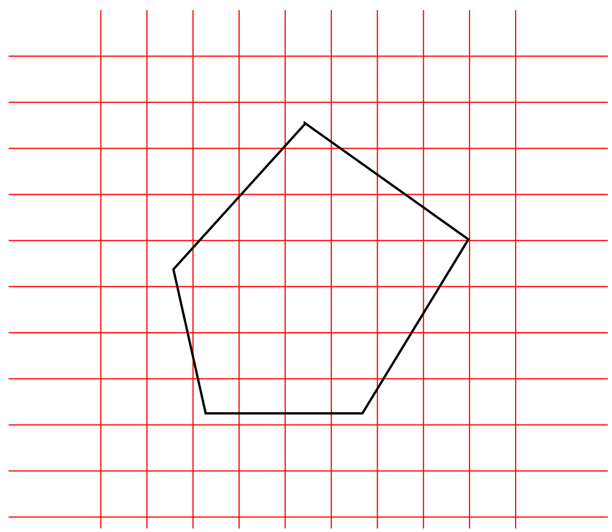
Generating Functions Algorithms in Integer Optimization

LECTURE I

Generating functions and Lattice Points

Jesús Antonio De Loera
University of California, Davis



Lattice Point Problems

Given a subset X of $\mathbb{R}^d$, there are a number of basic problems about lattice points:

- Decide whether $X \cap \mathbb{Z}^d$ is non empty.
- If X is bounded, count how many lattice points are in X.
- Given a norm, such as the l_∞ or l_p norms, find the shortest lattice vector of X.
- Given a linear functional $c \cdot x$ we wish to optimize it over the lattice points of X, i.e. find the lattice point in X that maximizes (minimizes) cx.

- Given a polynomial $f(x) \in \mathbb{Z}[x_1, \ldots, x_d]$, find $y \in X \cap \mathbb{Z}^d$ which maximizes the value $f(y)$.
- How to generate a lattice point in X uniformly at random?
- Find a Hilbert bases for a polyhedral cone X.

Discrete Optimization needs all the math help available

We present a non-traditional algebraic-analytic point of view:

GENERATING FUNCTIONS!!

ENCODING LATTICE POINTS

The Generating Function Encoding

Given $K \subset \mathbb{R}^d$ we **WANT** to compute the generating function

$$f(K) = \sum_{\alpha \in K \cap \mathbb{Z}^d} z_1^{\alpha_1} z_2^{\alpha_2} \dots z_n^{\alpha_n}.$$

Think of the lattice points as monomials!!! EXAMPLE: $(7, 4, -3)$ is $z_1^7 z_2^4 z_3^{-3}$.

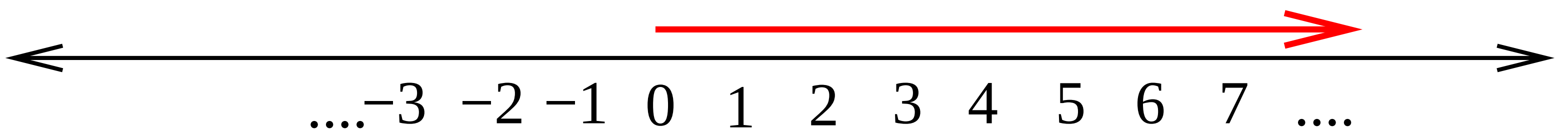
$f(K)$ has inside **all lattice points** of K. But it is too long! In fact, this is an infinite formal power series if K is not bounded, but if K is a polytope it is a (Laurent) polynomial.

We need a SHORT REPRESENTATION!!!

BARVINOK's ANSWER:

When K is a rational convex polyhedron, i.e. $K = \{x \in \mathbb{R}^n | Ax = b,\ Bx \leq b'\}$, where A, B are integral matrices and b, b' are integral vectors, The generating function $f(K)$, and thus ALL the lattice points of the polyhedron K, can be encoded in a "short" sum of rational functions!!!

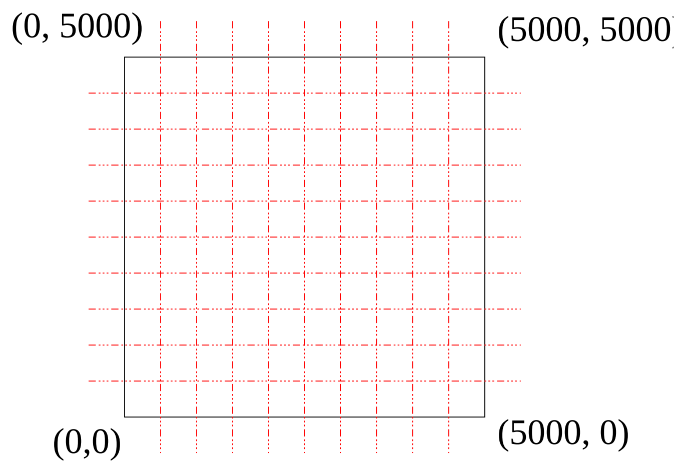
EXAMPLE 1: Suppose my polyhedron is the infinite half-line $P = \{x | x \geq 0\}$



$$f(P) = 1 + z + z^2 + z^3 + \ldots = \frac{1}{1-z}.$$

Example 2

Let P be the square with vertices $V_1 = (0,0)$, $V_2 = (5000,0)$, $V_3 = (5000,5000)$, and $V_4 = (0,5000)$.



The generating function $f(P)$ has over 25,000,000 monomials, $f(P) = 1 + z_1 + z_2 + z_1^1 z_2^2 + z_1^2 z_2 + \cdots + z_1^{5000} z_2^{5000}$, But it has only four rational functions in its Barvinok's encoding.

$$\frac{1}{(1-z_1)(1-z_2)}+\frac{{z_1}^{5000}}{(1-{z_1}^{-1})(1-z_2)}+\frac{{z_2}^{5000}}{(1-{z_2}^{-1})(1-z_1)}+\frac{{z_1}^{5000}{z_2}^{5000}}{(1-{z_1}^{-1})(1-{z_2}^{-1})}$$

Barvinok's Original Algorithm (1993 Barvinok)

Assume the dimension d is fixed. Let P be a rational convex d-dimensional polytope. Then, in polynomial time on the size of the input, we can write the generating function $f(P) = \sum_{\alpha \in P \cap \mathbb{Z}^d} z^\alpha$. as a polynomial-size sum of rational functions of the form:

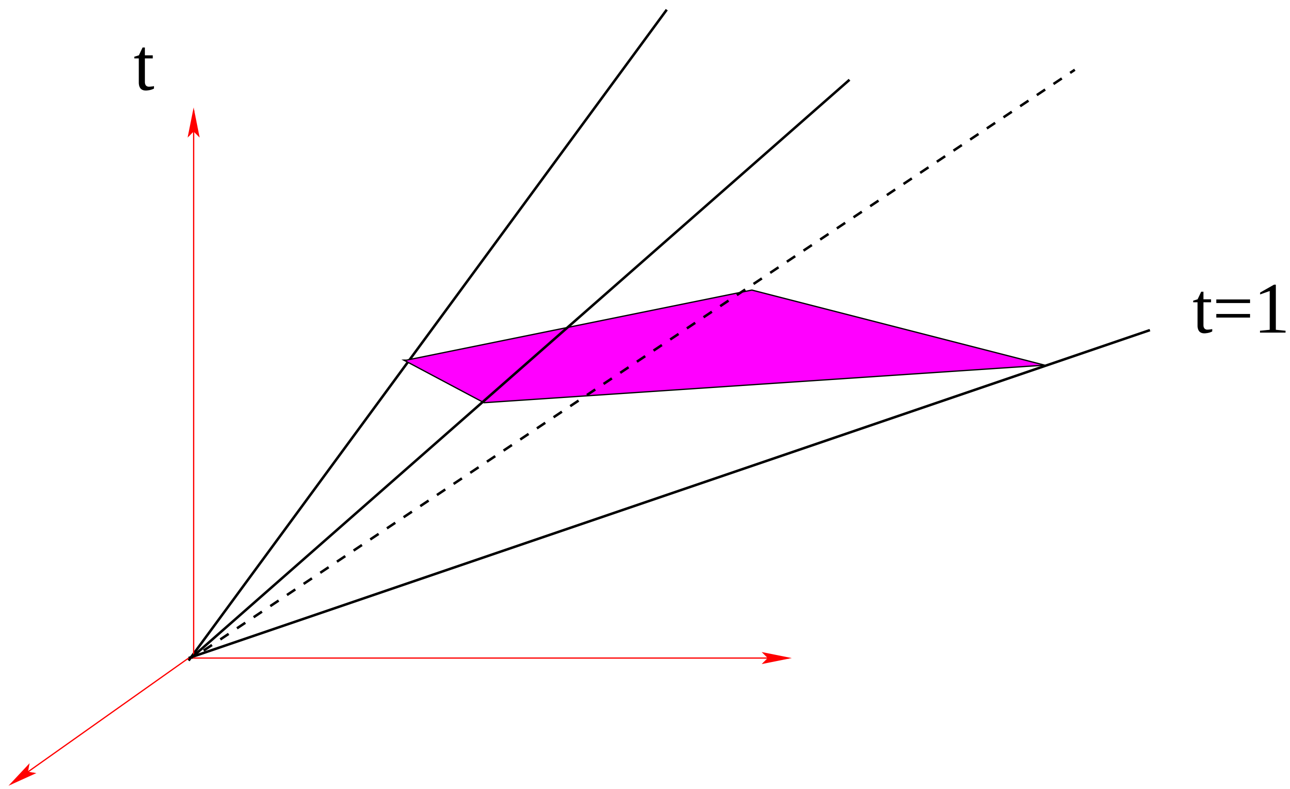
$$\sum_{i \in I} E_i \frac{z^{u_i}}{\prod_{j=1}^{d} (1 - z^{v_{ij}})}, \tag{1}$$

where I is a polynomial-size indexing set, and where $E_i \in \{1, -1\}$ and $u_i, v_{ij} \in \mathbb{Z}^d$ for all i and j.

We present an improved algorithm (2002 De Loera et al.)

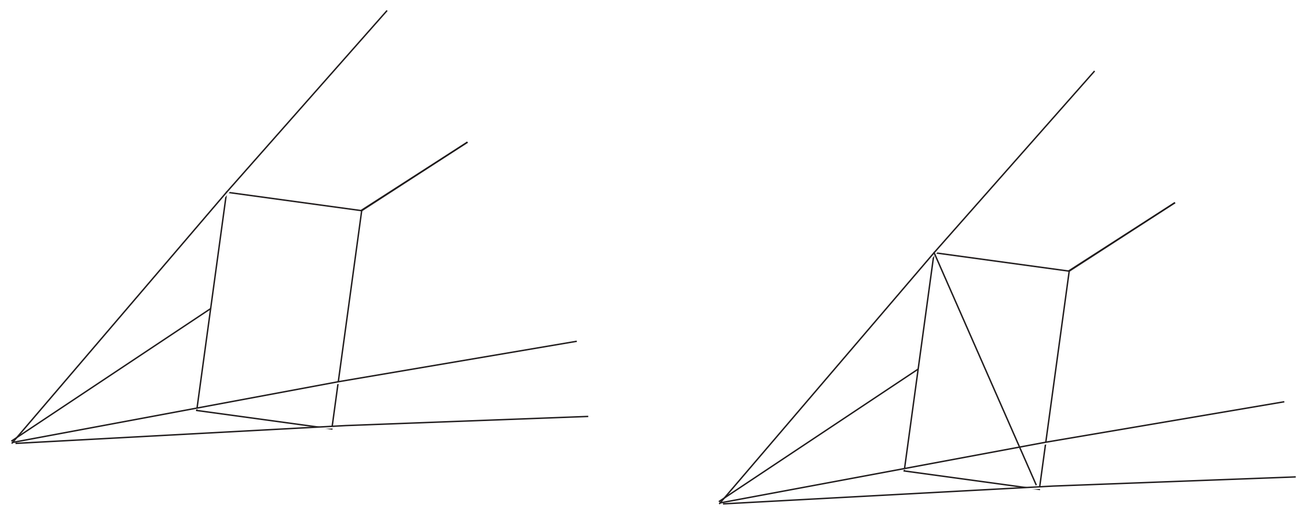
Enough to do it for CONES

Set your polytope P inside the hyperplane $t = 1$. What we want is the generating function of the lattice points in the cone.



Enough to do it for SIMPLE CONES

By the INCLUSION-EXCLUSION principle, we can just add the generating functions of the simplicial pieces!

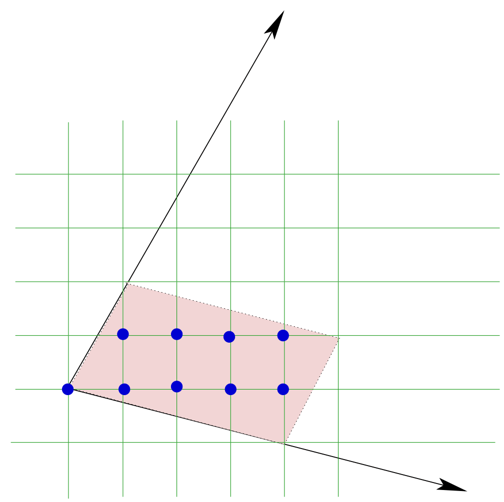


Simple Cones are Easy

For a simple cone $K \subset \mathbb{R}^d$,

$$f(K) = \frac{\sum_{u \in \Pi \cap \mathbb{Z}^d} z^u}{(1 - z^{c_1})(1 - z^{c_2}) \dots (1 - z^{c_d})}$$

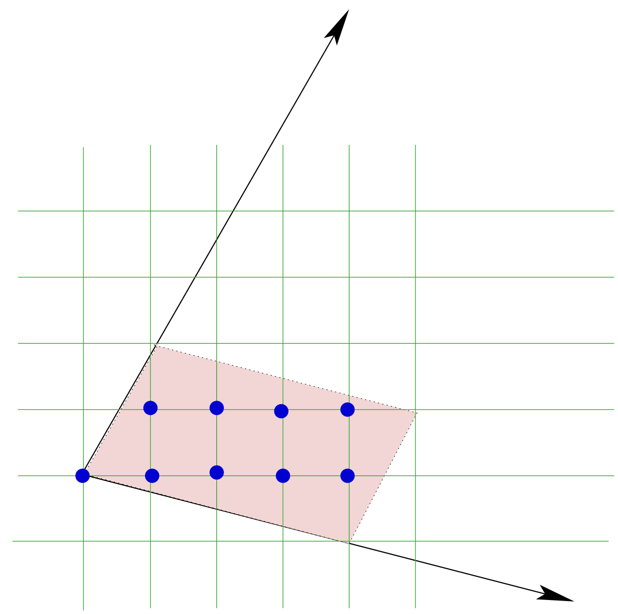
Π is the half open parallelepiped $\{x | x = \alpha_1 c_1 + \cdots + \alpha_d c_d,\ 0 \leq \alpha_i < 1\}$.



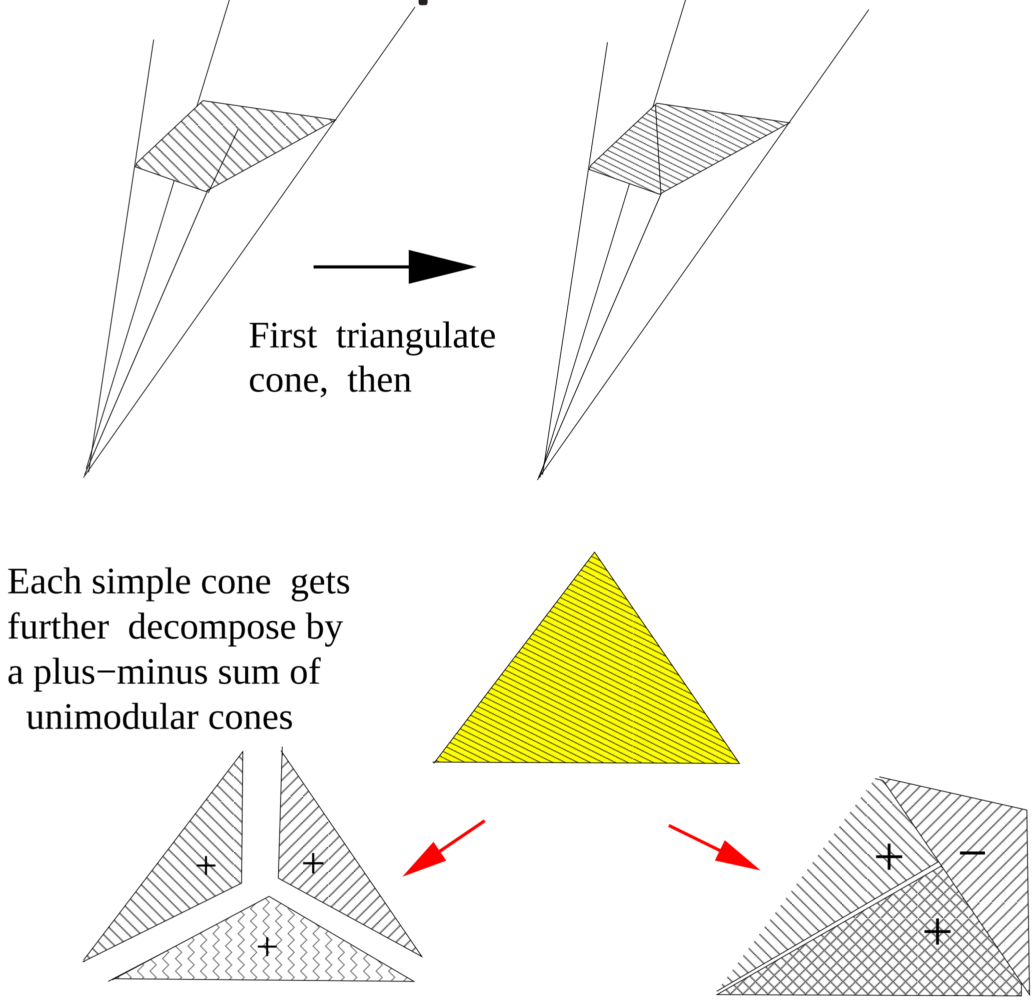
Example

In this case, we have $d = 2$ and $c_1 = (1, 2)$, $c_2 = (4, -1)$. We have:

$$f(K) = \frac{z_1^4 z_2 + z_1^3 z_2 + z_1^2 z_2 + z_1 z_2 + z_1^4 + z_1^3 + z_1^2 + z_1 + 1}{(1 - z_1 z_2^2)(1 - z_1^4 z_2^{-1})}.$$



If a cone K is not simple UNIMODULAR...break it



Barvinok's cone decomposition lemma

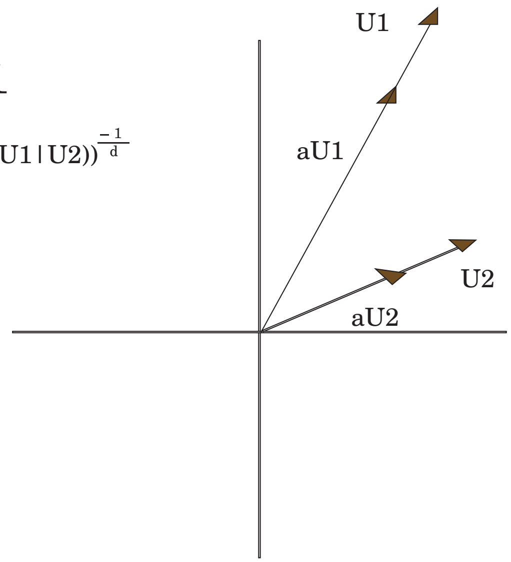
Theorem [Barvinok] Fix the dimension d. Then there exists a polynomial time algorithm which decomposes a rational polyhedral cone $K \subset \Re^d$ into unimodular cones K_i with numbers $\epsilon_i \in \{-1, 1\}$ such that

$$f(K) = \sum_{i \in I} \epsilon_i f(K_i),\ |I| < \infty.$$

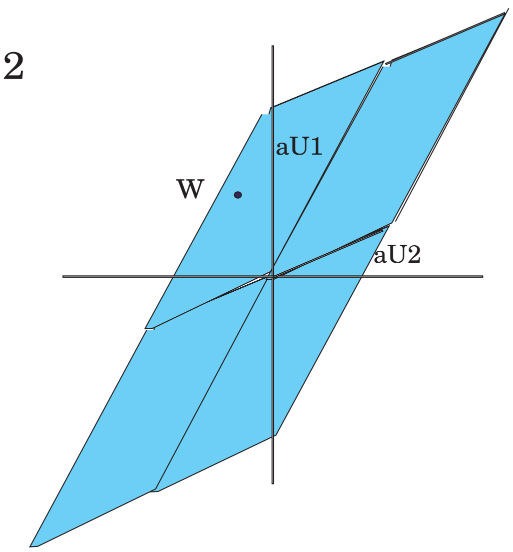
Main idea Triangulation is TOO expensive, allow simplicial cones's rays to be outside the original cone. Rays are short integer vectors inside a convex body, apply Minkowski's theorem!

Step 1

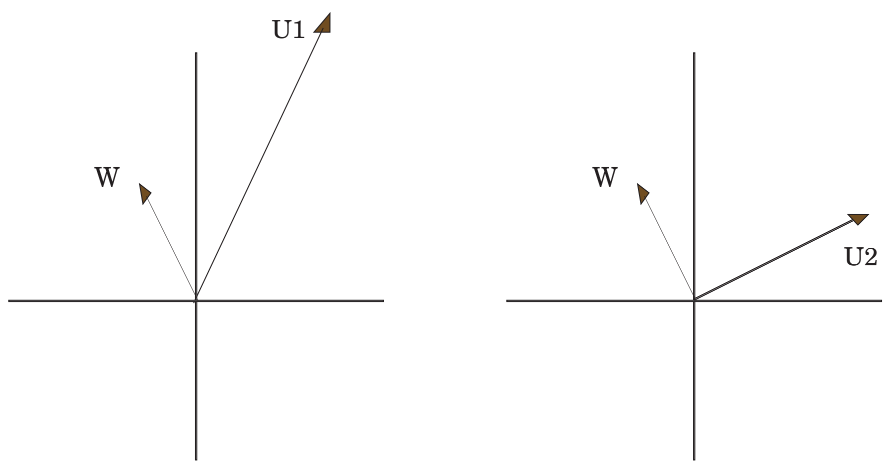
$a = (\det(U1 \mid U2))^{\frac{-1}{d}}$



Step 2



Step 3



SUMMARY of Homogenized Barvinok Algorithm.

Input is a full-dimensional convex rational convex polytope P in $\mathbb{R}^d$ specified by linear inequalities and linear equations.

1. Place the polytope P into the hyperplane defined by $x_{d+1} = 1$ in $\mathbb{R}^{d+1}$. Let K be the $d+1$-dimensional cone over P, that is, $K = cone(\{(p,1) : p \in P\})$.

2. We can triangulate K and reduce everything to simple cones $\sigma_1, \sigma_2, \ldots, \sigma_r$. Apply Barvinok's decomposition of σ_i into unimodular cones. We get a **signed** unimodular cone decomposition of K.

3. Retrieve a signed sum of multivariate rational functions, one per cone, which represents the series $\sum_{a \in K \cap \mathbb{Z}^n} x^a$.

4. If we call the variable $x_{d+1} = t$ then we obtain the expression of the generating function of $\sum_{n=0}^{\infty} \left(\sum_{\alpha \in nP \cap \mathbb{Z}^d} z^\alpha \right) \ t^n$,

EXAMPLE

For the triangle σ with vertices $V_0 = (-1,-1)$, $V_1 = (2,-1)$, and $V_2 = (-1,2)$ we have

$$(1-x)^{-1}(1-y)^{-1}\left(1-\frac{t}{xy}\right)^{-1} + \left(1-x^{-1}\right)^{-1}\left(1-\frac{y}{x}\right)^{-1}\left(1-\frac{x^2t}{y}\right)^{-1}$$
$$+ \left(1-y^{-1}\right)^{-1}\left(1-\frac{x}{y}\right)^{-1}\left(1-\frac{y^2t}{x}\right)^{-1}$$

Counting Lattice Points FAST!

LEMMA: The number of lattice points in P is the limit when the vector $(x_1, \ldots, x_n)$ goes to $(1, 1, \ldots, 1)$.

TROUBLE: The vector $(1, 1, \ldots, 1)$ is a pole in all the rational functions, a singularity, because the Barvinok rational functions are

$$\frac{z^a}{\prod_{i=1}^{k}(1 - z_i^v)}$$

HOW TO COMPUTE THIS LIMIT????

Shall I expand into monomials???

The singularity gets resolved that way...right?

NO WAY!

Never fully expand the rational

functions into ALL monomials!

USE NUMERICAL COMPLEX ANALYSIS 101

TO EVALUATE THE RATIONAL FUNCTIONS!!

Reduce everything to ONE variable

$f(P, z)$ has the form:

$$\sum_{i \in I} E[i] \frac{z^{A_i}}{\prod(1 - z^{B_{ij}})},$$

an integer vector λ and making the substitution $z_i \to t^{\lambda_i}$. λ must be such that dot product of λ with a B_{ij} cannot be zero. We have

$$\sum_{i=1}^{\#\ of\ cones} E[i] \frac{t^{num_i}}{\prod(1 - t^{den_{ij}})}$$

The Laurent expansion of these expressions at $t = 1$ must equal to the finite number $\sum_{\alpha \in P \cap \mathbb{Z}^d} 1$. Thus we need the constant coefficient!

Computation of Residues for rational functions

This reduces to computing a residue at a pole z_0.

If $f(z) = \sum_{k=-m}^{\infty} a_n (z - z_0)^k$, the residue is defined as

$$Res(f(z_0)) = a_{-1}.$$

Given a rational function $f(z) = \frac{p(z)}{q(z)}$, and a pole z_0 we use

THEOREM *Henrici's Algorithm for the residue:* If $p(z), q(z)$ have degree no more than d, then residue at z_0 can be computed in no more than $0(d^2)$ arithmetic operations.

Algorithm

(CASE 1) If z_0 is a simple pole is TRIVIAL, then $Resf(z_0) = \frac{p(z_0)}{q'(z_0)}$.

(CASE 2) Else z_0 is a pole of order $m > 1$,

(A) Write $f(z) = \frac{p(z)}{(z-z_0)^m q_1(z)}$.

(B) Expand p, q_1 in powers of $(z - z_0)$

$p(z) = a_0 + a_1(z - z_0) + a_2(z - z_0)^2 + \ldots$ $q_1(z) = b_0 + b_1(z - z_0) + b_2(z - z_0)^2 + \ldots$

(C) The Taylor expansion of $p(z)/q_1(z)$ at z_0 is $c_0 + c_1(z - z_0) + c_2(z - z_0)^2 + c_3(z - z_0)^3 + ...$ where

$c_0 = \frac{a_0}{b_0}$, and $c_k = \frac{1}{b_0}(a_k - b_1 c_{k-1} - b_2 c_{k-2} - \cdots - b_k c_0)$

(D) OUTPUT $Res(f(z_0)) = c_{m-1}$.

Example

(A triangle). Let us consider three vertice $V_1 = (0,1)$, $V_2 = (1,0)$, and $V_3 = (0,0)$. In this case we have the rational functions:

$$f(K_1) = \frac{y}{(1-y^{-1})(1-xy^{-1})}, f(K_2) = \frac{x}{(1-x^{-1})(1-x^{-1}y)},$$

$$f(K_3) = \frac{1}{(1-x)(1-y)}.$$

We choose a vector λ such that the inner products of λ and the generators of K_i are not equal to zero. We choose $\lambda = (1,-1)$ in this example. have (after cosmetic surgery!):

$$f(K_1) = \frac{1}{(1-t)(1-t^2)}, f(K_2,t) = \frac{t^5}{(1-t^1)(1-t^2)}, f(K_3,t) = \frac{-t^2}{(1-t)(1-t)}.$$

Simplify to the form $\frac{Q_1(s)}{(1-z)^m q(s)}$. Apply Henrici recurrence.

Monomial Substitution

Lemma: Let us fix k, the number of binomials in the denominator of a rational function. Given a rational function sum g of the form

$$g(x) = \sum_{i \in I} \alpha_i \frac{x^{u_i}}{\prod_{j=1}^{k} (1 - x^{v_{ij}})},$$

where u_i, v_{ij} are integral d-dimensional vectors, and a monomial map $\psi : \mathbb{C}^n \longrightarrow \mathbb{C}^d$ given by the variable change $x_i \rightarrow z_1^{l_{i1}} z_2^{l_{i2}} \dots z_n^{l_{in}}$ whose image does not lie entirely in the set of poles of $g(x)$, then there exists a polynomial time algorithm which, computes the function $g(\psi(z))$ as a sum of rational functions of the same shape as $g(z)$.

Corollary: Random Generation of Lattice Points

How to pick a random lattice point? Markov chain methods have been around for some time, but they work on some "roundness" assumptions!! Not working well for all polytopes! (work by Dyer, Frieze, Kannan, Lovasz, Simonovits and others)

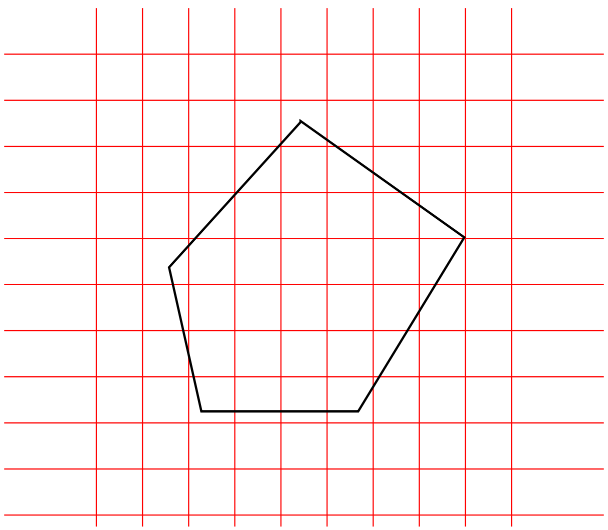
THEOREM (Barvinok-Pak) Let P be a convex rational polytope in $\mathbb{R}^d$. Then using $O(d^2 \log(size(P)))$ calls to Barvinok's counting algorithm, one can in polynomial time can sample uniformly from set $P \cap \mathbb{Z}^d$.

Generating Functions Algorithms in Integer Optimization

LECTURE II

Contributions to Integer Linear Programming

Jesús Antonio De Loera
University of California, Davis



Integer Linear Programming

Given the a convex polyhedron $X = P \cap \mathbb{Z}^d$, and a linear functional $c \cdot x$ we wish to optimize it over the lattice points of X, i.e. find the lattice point in X that maximizes (minimizes) cx.

We take the point of view: GENERATING FUNCTIONS.

Recall: Barvinok's Theorem

Assume the dimension d is fixed. Let P be a rational convex d-dimensional polytope. Then, in polynomial time on the size of the input, we can write the generating function $f(P) = \sum_{\alpha \in P \cap \mathbb{Z}^d} z^{\alpha}$. as a polynomial-size sum of rational functions of the form:

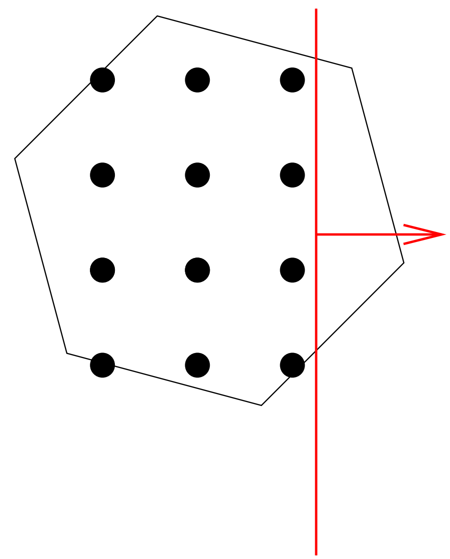
$$\sum_{i \in I} E_i \frac{z^{u_i}}{\prod_{j=1}^{d} (1 - z^{v_{ij}})}, \tag{2}$$

where I is a polynomial-size indexing set, and where $E_i \in \{1, -1\}$ and $u_i, v_{ij} \in \mathbb{Z}^d$ for all i and j.

ALGORITHM: Barvinok + Binary Search

Input: $A \in \mathbb{Z}^{m \times d}$, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^d$.

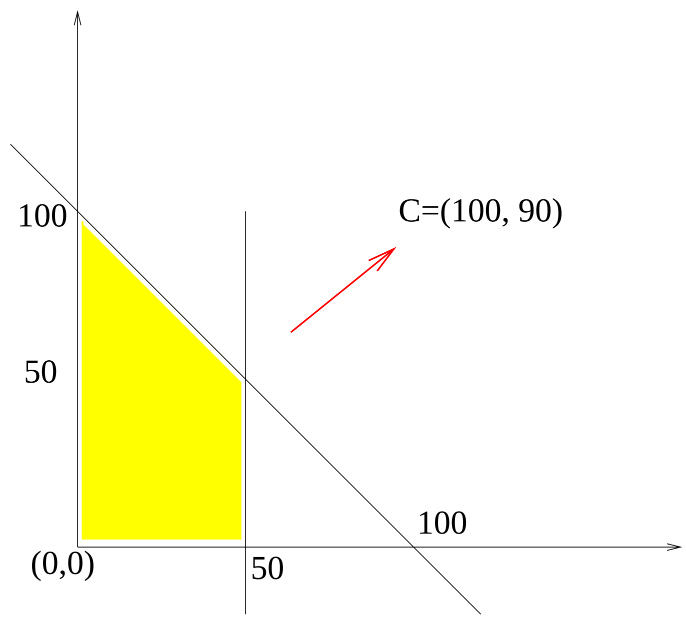
Output: The optimal value of maximize $\{c \cdot x : Ax \leq b, x \geq 0, x \in \mathbb{Z}^d\}$.



For fixed d, this algorithm runs in polynomial time (on the input size) by using the polynomiality of Barvinok's counting algorithm.

Toward a more Direct Algorithm:

Barvinok's algorithm computes the function $f(P,z) = \sum_{\alpha \in P \cap \mathbb{Z}^d} z^\alpha$, in the form: $f(P,z) = \sum_{i \in I} \epsilon_i \frac{z^{u_i}}{\prod_{j=1}^{n}(1-z^{v_{ij}})}$.



$$f(P,z) = \frac{1}{(1-z_1)(1-z_2)} + \frac{z_1^{50}}{(1-z_1^{-1})(1-z_2)} + \frac{z_2^{100}}{(1-z_1^{-1})(1-z_2)} + \frac{z_1^{50} z_2^{50}}{(1-z_1^{-1})(1-z_1^{-1} z_2)}.$$

Changing Variables is IMPORTANT!!

$f(P,z) = \sum_{\alpha \in P \cap \mathbb{Z}^d} z^\alpha$, in the form:

$$f(P,z) = \sum_{i \in I} \epsilon_i \frac{z^{u_i}}{\prod_{j=1}^{n}(1 - z^{v_{ij}})}.$$

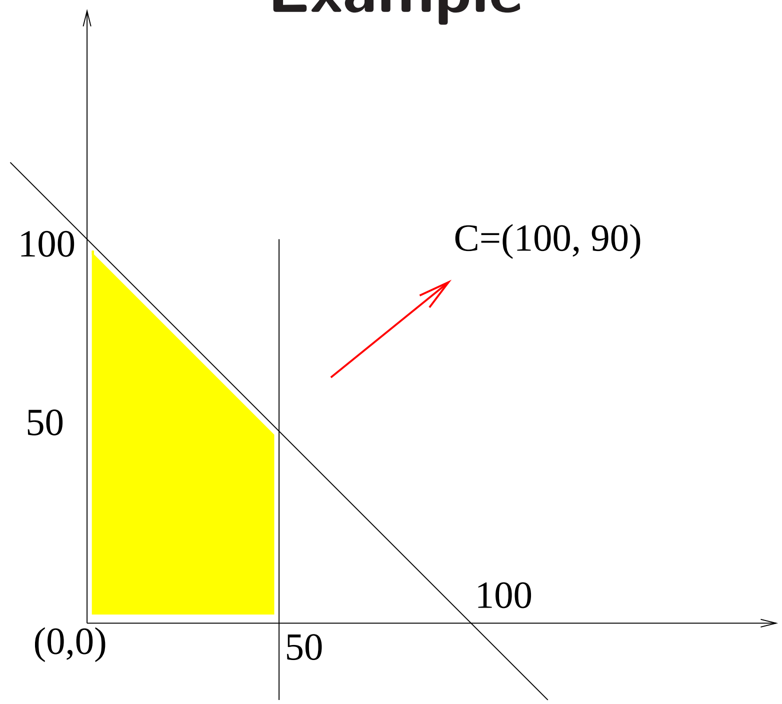
If we make the substitutions $z_i \to t^{c_i}$, then we have $z^\alpha \to t^{c \cdot \alpha}$,

$$f(P,z) \to \sum_{\alpha \in P \cap \mathbb{Z}^d} t^{c \cdot \alpha}$$

$$= t^M + (\text{lower degree terms in } t)$$

M is the optimal value of the integer linear programming problem!

Example



$$f(P,z) = \frac{1}{(1-z_1)(1-z_2)} + \frac{z_1^{50}}{(1-z_1^{-1})(1-z_2)} + \frac{z_2^{100}}{(1-z_1^{-1})(1-z_2)} + \frac{z_1^{50} z_2^{50}}{(1-z_1^{-1})(1-z_1^{-1} z_2)}.$$

Substitute $z_1 \to t^{100}$ and $z_2 \to t^{90}$, then we have t^{9500} + lower degree terms in t.

Monomial Substitution

Lemma: (Barvinok-Woods) Let us fix k, the number of binomials in the denominator of a rational function. Given a rational function sum g of the form

$$g(x) = \sum_{i \in I} \alpha_i \frac{x^{u_i}}{\prod_{j=1}^{k} (1 - x^{v_{ij}})},$$

where u_i, v_{ij} are integral d-dimensional vectors, and a monomial map $\psi : \mathbb{C}^n \longrightarrow \mathbb{C}^d$ given by the variable change $x_i \to z_1^{l_{i1}} z_2^{l_{i2}} \ldots z_n^{l_{in}}$ whose image does not lie entirely in the set of poles of $g(x)$, then there exists a polynomial time algorithm which, computes the function $g(\psi(z))$ as a sum of rational functions of the same shape as $g(z)$.

A Reformulation of Integer Linear Programming:

GOAL: Given $A \in \mathbb{Z}^{m \times d}$, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^d$, and assume that number of variables d is fixed. Wish to solve the integer programming problem

maximize $(c \cdot x)$ subject to $x \in \{x | Ax \leq b, x \geq 0, \; x_i \in \mathbb{Z} \; \}$,

In our setting this is

DETECTING THE HIGHEST DEGREE COEFFICIENT OF A POLYNOMIAL!

THE POLYNOMIAL IS GIVEN AS A SUM OF RATIONAL FUNCTIONS.

Several different ways to do this!

Digging Algorithm: Laurent Series Expansion

Input: $A \in \mathbb{Z}^{m \times d}$, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^d$.

Output: The optimal value of maximize $\{c \cdot x : Ax \leq b, x \geq 0, x \in \mathbb{Z}^d\}$.

(A) Using Barvinok's algorithm and monomial substitution compute the rational function expression

$$\sum_{i \in I} E_i \frac{t^{c \cdot u_i}}{\prod_{j=1}^{d} (1 - t^{c \cdot v_{ij}})}. \tag{3}$$

(B) Use the identity

$$\frac{1}{1 - t^{c \cdot v_{ij}}} = \frac{-t^{-c \cdot v_{ij}}}{1 - t^{-c \cdot v_{ij}}}$$

as necessary to enforce that all v_{ij} in (3) satisfy $c \cdot v_{ij} < 0$. So now the terms of the series are given in decreasing order with respect to the degree of t.

(3) For each of the rational functions in the sum compute a Laurent series expansion of the form

$$E_i \, t^{c \cdot u_i} \prod_{j=1}^{d} (1 + t^{c \cdot v_{ij}} + (t^{c \cdot v_{ij}})^2 + \ldots).$$

multiply out the factors and add the terms, group together those of the same degree in t. Thus we find a term expansion. Proceed in decreasing order with respect to the degree of t.

(4) Continue until a degree n of t is found such that for some the coefficient is non-zero in the expansion. Return n as the optimal value.

Number of roots of a polynomial

We have a black box polynomial $p(z)$. Assume coefficients are all 1 or 0. This happens for generic cost vectors!!

LEMMA: (Argument principle) Let C be a simple closed curve in the complex plane that contains no root of $p(z)$ itself. Then

$$\text{number of roots of } p(z) \text{ (with multiplicity) inside} C = \frac{1}{2\pi i}\left(\int_C \frac{p'(z)}{p(z)}dz\right).$$

(A) Find an upper bound M on the absolute value of the roots. $M = 2$ sufficient by a result of Cauchy.

(B) Let C be the square centered at the origin of size $2M$

(C) Perform the integration numerically. High accuracy not necessary because the answer is an integer.

Example

$$f(z) = z^{25} + z^{24} + z^{23} + z^{22} + z^{21} + z^{20} + z^{19} + z^{18} + z^{17} + z^{16} + z^{15} + z^{14} + z^{13} + z^{12} + z^{11} + z^{10} + z^9 + z^8 + z^7 + z^6 + z^5 + z^4 + z^3 + z^2 + z$$

```
p:=unapply(D(f)(z)/f(z),z);

Digits:=14:

evalf(Int(p(t-2*I), t=-2..2,method=_CCquad)+ Int(p(2+I*t)*I, t=-2..
     -Int(p(t+2*I), t=-2..2, method=_CCquad)-Int(p(-2+t*I)*I, t=-2

                                   -13
INTEGRALS EQUAL            0.6 10    + 157.07963267950 I
                                                            -14
> evalf(%/(2*Pi*I)) ->   25.000000000002 - 0.95492965855137 10    I
```

Boolean operations on rational functions

Lemma: Let S_1, S_2 be finite subsets of $\mathbb{Z}^n$ and let $f(S_1, x)$ and $f(S_2, x)$ be the corresponding generating functions, represented as short rational functions with at most k binomials in each denominator. Then there exist a polynomial time algorithm, which, given $f(S_i, x)$, computes

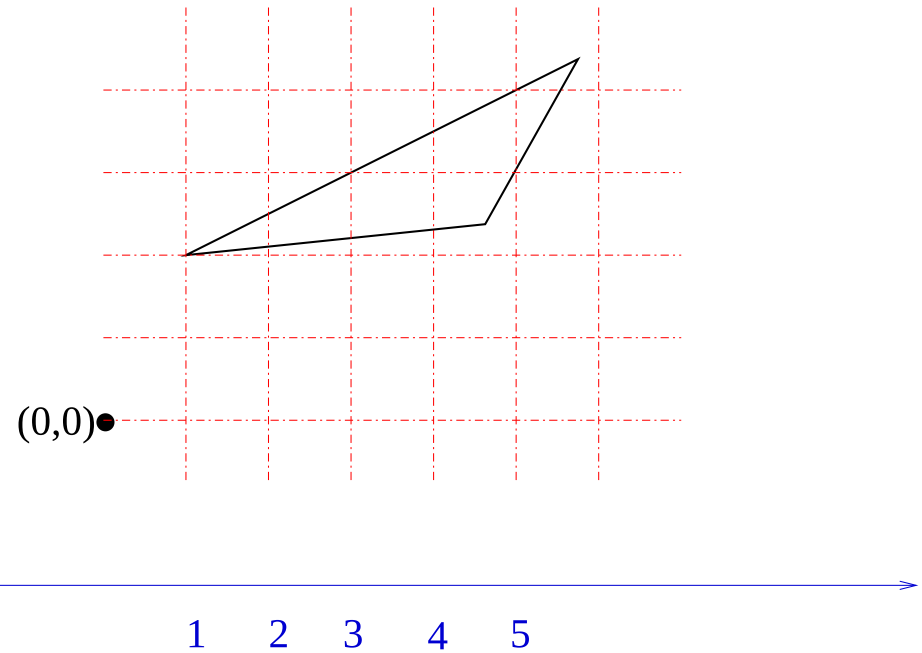
$$f(S_1 \cap S_2, x) \quad = \quad \sum_{i \in I} \gamma_i \frac{x^{u_i}}{(1 - x^{v_{i1}}) \dots (1 - x^{v_{is}})}$$

with $s \leq 2k$ and γ_i rational numbers, u_i, v_{ij} nonzero integers.

Same with finite unions or complements!

The Projection Lemma

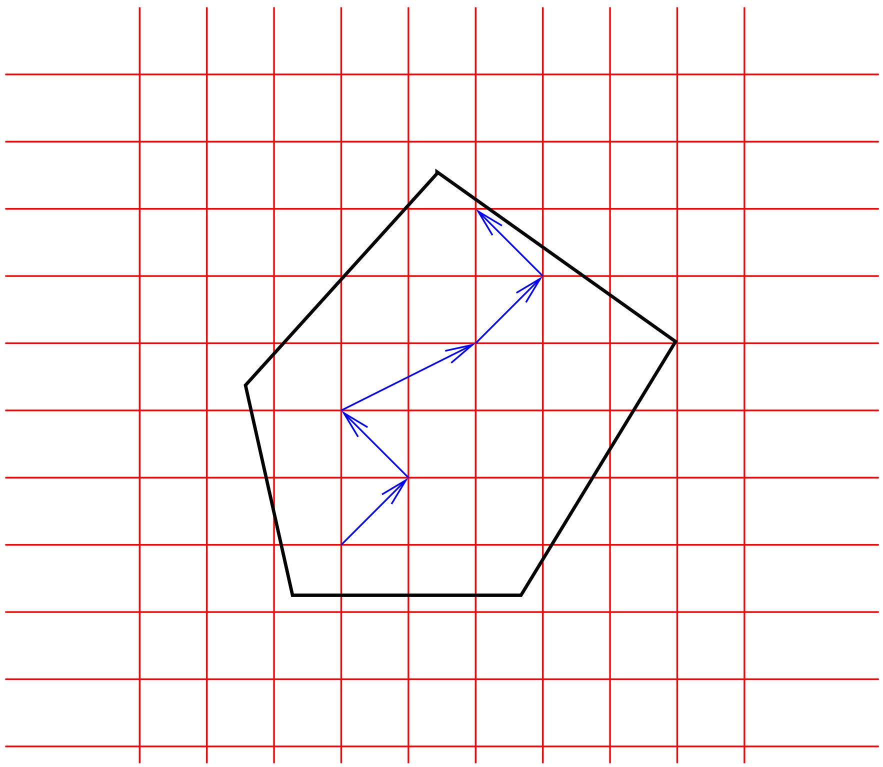
Lemma Consider a rational polytope $P \subset R^n$ and a linear map $T : \mathbb{Z}^n \to \mathbb{Z}^k$. There is a polynomial time algorithm which computes a short representation of the generating function $f\big(T(P \cap \mathbb{Z}^n), x\big)$.



$$z_1 z_2^2 + z_1^3 z_2^3 + z_1^4 z_2^3 + z_1^5 z_2^3 + z_1^5 z_2^4 \quad \text{projects to} \quad z_1 + z_1^3 + z_1^4 + z_1^5.$$

TEST SETS

A TEST SET is a finite collection of integral vectors with the property that every feasible non-optimal solution of an integer program in the can be improved by adding a vector in the test set.



Examples of test sets and augmentation methods: Graver and Gröbner bases, Hilbert bases, integral basis method. Work by Hemmecke, Graver, Scarf, Thomas, Sturmfels, Weismantel et al. and others.

TROUBLE Test sets can be exponentially large even in fixed dimension!

THEOREM: (Barvinok-Woods 2003) When the dimension is fixed, Barvinok's rational functions can compute Hilbert bases or Graver bases in polynomial time (on the size of the input), as rational functions.

THEOREM: (LattE team 2004 + Sturmfels) When the dimension is fixed, Barvinok's rational functions can compute reduced Gröbner bases in polynomial time (on the size of the input), as rational functions.

LattE

- Our goal was to implement and develop algebraic-analytic algorithms. Members: J. De Loera, R. Hemmecke, R. Yoshida, D. Haws, P. Huggins, J. Tauzer.
- First implementation of Barvinok's encoding algorithm. Software implemented in C++.
- We used also libraries from **CDD, NTL**.
- We use BOTH geometric computing AND symbolic-algebraic manipulations!!

Experimental Results

The cost vector c, we choose the first d components of the vector

$$(213, -1928, -11111, -2345, 9123, -12834, -123, 122331, 0, 0).$$

Problem	a										b
prob1	25067	49300	4 9717	62124	87608	88025	11 3673	119169			33367336
prob2	11948	23330	30635	44197	92754	123389	136951	140745			14215207
prob3	39559	61679	79625	99658	133404	137071	159757	173977			58424800
prob4	48709	55893	62177	65919	86271	87692	102881	109765			60575666
prob5	28637	48198	80330	91980	102221	135518	165564	176049			62442885
prob6	20601	40429	40429	45415	53725	61919	64470	69340	78539	95043	22382775
prob7	18902	26720	34538	34868	49201	49531	65167	66800	84069	137179	27267752
prob8	17035	45529	48317	48506	86120	100178	112464	115819	125128	129688	21733991
prob9	3719	20289	29067	60517	64354	65633	76969	102024	106036	119930	13385100
prob10	45276	70778	86911	92634	97839	125941	134269	141033	147279	153525	106925262

Table 1: We implemented the *BBS algorithm* and the *digging algorithm* in `LattE`. We solved several challenging knapsack problems by Aardal, Lenstra, and Lenstra

	Runtime for digging (Original)	Runtime for digging (S. Cone)	Runtime for BBS	Runtime for CPLEX 6.6
cuww1	0.4 sec.	0.17 sec.	414 sec.	$>$ 1.5 h (OM)
cuww2	$>$ 2 h	$>$2 h	1.8 h	$>$ 0.75 h (OM)
cuww3	1.4 sec.	0.24 sec.	1.7 h	$>$ 0.75 h (OM)
cuww4	$>$ 2 h	$>$ 2 h	$>$ 2 h	$>$ 0.75 h (OM)
cuww5	$>$ 2 h	147.63 sec.	$>$ 2 h	$>$ 0.75 h (OM)
prob1	51.4 sec.	18.55 sec.	$>$ 2 h	$>$ 1 h (OM)
prob2	24.8 sec.	6.07 sec	$>$ 2 h	$>$ 0.75 h (OM)
prob3	48.2 sec.	9.03 sec.	$>$ 2 h	$>$ 1.5 h (OM)
prob4	34.2 sec.	9.61 sec.	$>$ 2 h	$>$ 1.5 h (OM)
prob5	34.5 sec.	9.94 sec.	$>$ 2 h	$>$ 1.5 h (OM)
prob6	143.2 sec.	19.21 sec.	$>$ 2 h	$>$ 2 h (OM)
prob7	142.3 sec.	12.84 sec.	$>$ 2 h	$>$ 1 h (OM)
prob8	469.9 sec.	49.21 sec.	$>$ 2 h	$>$ 2.5 h (OM)
prob9	0.39 h	283.34 sec.	$>$ 2 h	4.7 sec.
prob10	250.6 sec.	29.28 sec.	$>$ 2 h	$>$ 1 h (OM)

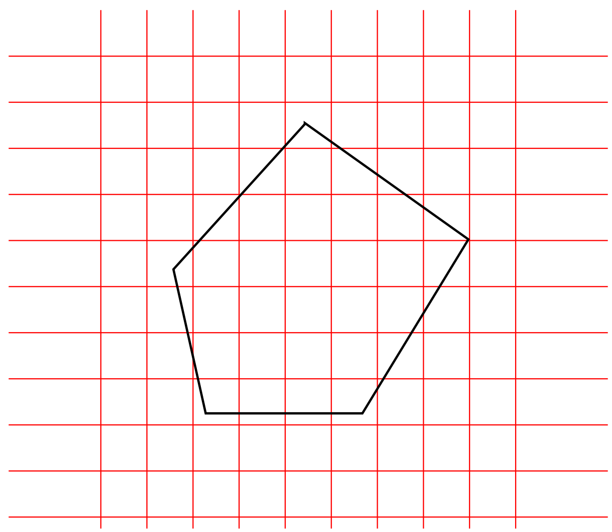
Table 2: Optimal values, optimal solutions, and running times for each problem. Symbol $> x$ h (OM), means CPLEX ran out of memory after x hours. The symbol $>$ 2 h means that the problem was not solved before two hours of computation.

Generating Functions Algorithms in Integer Optimization

LECTURE III

Contributions to Integer Non-Linear Programming

Jesús Antonio De Loera
University of California, Davis



Integer Semi-algebraic Optimization

Problem: Let f, g_i are d-variate polynomials with integral coefficients.

$$\boxed{\text{maximize } f(x_1, \ldots, x_d) \text{ subject to } g_i(x_1, \ldots, x_d) \geq 0,\ x \in \mathbb{Z}^d.}$$

Also called **Integer Semialgebraic Optimization.**

Question: **What happens if we assume the number of variables is fixed?**

Positive Notes: problem contains *Integer Linear Programming*, Lenstra's Algorithm guarantees is solvable in polynomial time for fixed dimension. Also, *Integer Semidefinite Programming* runs in polynomial time in fixed dimension by Khachiyan and Porkolab's work.

Negative Notes: continuous polynomial optimization over polytopes, without fixed dimension, is NP-hard and no FPTAS is possible! the max-cut problem can be modeled as minimizing a quadratic form over the cube $[-1, 1]^d$.

The whole picture

Table 3: Computational complexity of polynomial integer problems in fixed dimension.

	Type of objective function				
Type of constraints	linear		convex polynomial		arbitrary polynomial
Linear constraints,	polytime $(*)$	$\Leftarrow$	polytime $(**)$		NP-hard (a)
	$\Uparrow$		$\Uparrow$		$\Downarrow$
Convex semialgebraic constraints,	polytime $(**)$	$\Leftarrow$	polytime $(**)$		NP-hard (c)
Arbitrary polynomial constraints,	undecidable (b)	$\Rightarrow$	undecidable (d)	$\Rightarrow$	undecidable (e)

Integer Polynomial Optimization over a Polytope

Problem: Let f be a d-variate polynomial with integral coefficients. Now the $g_i(x)$ are *linear inequalities.*

maximize $f(x_1, \ldots, x_d)$ subject to $g_i(x_1, \ldots, x_d) \geq 0, \; x \in \mathbb{Z}^d$.

Example: Consider this problem from *MINLPLIB* library

$$\begin{aligned} \max \quad & 100 \left(\frac{1}{2} + i_2 - \left(\frac{3}{5} + i_1 \right)^2 \right)^2 + \left(\frac{2}{5} - i_1 \right)^2 \\ \text{s.t.} \quad & i_1, i_2 \in [0, 200] \cap \mathbb{Z}. \end{aligned} \tag{4}$$

Its optimal solution is $i_1 = 1$, $i_2 = 2$ with an objective value of 0.72.

Integer Polynomial Optimization over a Polytope

Theorem (D,Hemmecke,Koeppe,Weismantel) Let the number of variables d be fixed. Let $f(x_1, \ldots, x_d)$ be a polynomial of maximum total degree D with integer coefficients, and let P be a convex rational polytope defined by linear inequalities in d variables.

(1) We can construct an increasing sequence of lower bounds $\{L_k\}$ and a decreasing sequence of upper bounds $\{U_k\}$ to the optimal value

$$f^* = \text{maximize } f(x_1, x_2, \ldots, x_d) \text{ subject to } x \in P \cap \mathbb{Z}^d. \tag{5}$$

The bounds L_k, U_k can be computed in time polynomial in k, the input size of P and f, and the maximum total degree D and they satisfy the inequality $U_k - L_k \leq f^* \cdot (\sqrt[k]{|P \cap \mathbb{Z}^d|} - 1)$.

(2) Moreover, if f is positive semidefinite over the polytope (i.e. $f(x) \geq 0$ for all $x \in P$), there exists a fully polynomial-time approximation scheme (FPTAS) for the optimization problem (5).

The construction of the bounds and algorithm uses **Barvinok's rational functions.**

Polynomial Evaluation Lemma

Lemma: Given a Barvinok rational function $f(S)$, representing a finite set of lattice points S, and a polynomial g with integer coefficients we can compute, in time polynomial on the input size a Barvinok rational function for the generating function

$$f(S, g, z) = \sum_{a \in S} g(a) z^a.$$

NOTE: This is *independent* of the degree of g.

Differential Operators give the coefficients:

We can define the basic differential operator associated to $f(x) = x_r$

$$z_r \frac{\partial}{\partial z_r} \cdot \sum_{\alpha \in P \cap \mathbb{Z}^d} z^\alpha = \sum_{\alpha \in P \cap \mathbb{Z}^d} z_r \frac{\partial}{\partial z_r} z^\alpha = \sum_{\alpha \in P \cap \mathbb{Z}^d} \alpha_r z^\alpha.$$

Next if $f(z) = c \cdot z_1^{\beta_1} \cdot \ldots \cdot z_d^{\beta_d}$, then we can compute again a rational function representation of $g_{P,f}(z)$ by repeated application of basic differential operators:

$$c \left(z_1 \frac{\partial}{\partial z_1} \right)^{\beta_1} \cdot \ldots \cdot \left(z_d \frac{\partial}{\partial z_d} \right)^{\beta_d} \cdot g_P(z) = \sum_{\alpha \in P \cap \mathbb{Z}^d} c \cdot \alpha^\beta z^\alpha.$$

Sketch of proof of theorem/algorithm

Input: A rational convex polytope $P \subset \mathbb{R}^d$, a polynomial objective $f \in \mathbb{Z}[x_1, \ldots, x_d]$ of maximum total degree D.

Output: An increasing sequence of lower bounds L_k, and a decreasing sequence of upper bounds U_k reaching the maximal function value f^* of f over all lattice points of P.

W.l.o.g: We can assume f is positive semidefinite. Else translate it!

Via Barvinok's algorithm compute a short rational function expression for the generating function $g_P(z) = \sum_{\alpha \in P \cap \mathbb{Z}^d} z^\alpha$. From $g_P(z)$ compute the number $|P \cap \mathbb{Z}^d| = g_P(1)$ of lattice points in P. Can be done in polynomial time.

How to define such sequences to approximate the maximum?

Lemma: For a collection $S = \{s_1, \ldots, s_r\}$ of non-negative real numbers, $\mathrm{maximum}\{s_i | s_i \in S\}$ equals $\lim_{k\to\infty} \sqrt[k]{\sum_{j=1}^{r} s_j^k}$.

From the rational function representation $g_P(z)$ of the generating function $\sum_{\alpha\in P\cap\mathbb{Z}^d} z^\alpha$ we can compute the rational function representation of $g_{P,f^k}(z)$ of $\sum_{\alpha\in P\cap\mathbb{Z}^d} f^k(\alpha) z^\alpha$ in polynomial time by application of the Polynomial Evaluation Lemma.

define

$$L_k := \sqrt[k]{g_{P,f^k}(1)/g_{P,f^0}(1)} \quad \text{and} \quad U_k := \sqrt[k]{g_{P,f^k}(1)}.$$

If you want the optimal value compute the sequences L_k, U_k until $\lfloor U_k \rfloor - \lceil L_k \rceil < 1$ stop and return $\lceil L_k \rceil = \lfloor U_k \rfloor$ as the optimal value.

Example

maximize x^3y subject to

$$\{(x,y)|3991 \le 3996\,x - 4\,y \le 3993,\ 1/2 \le x \le 5/2,\ \text{integer}\}.$$

Region contains only 2 lattice points. The sum of rational functions encoding the lattice points is

$$x^2y^{1000}\left(1-\frac{1}{xy^{999}}\right)^{-1}\left(1-y^{-1}\right)^{-1}+xy\left(1-xy^{999}\right)^{-1}\left(1-y^{-1}\right)^{-1}+$$
$$\frac{xy}{\left(1-xy^{999}\right)\left(1-y\right)}+x^2y^{1000}\left(1-\frac{1}{xy^{999}}\right)^{-1}\left(1-y\right)^{-1}.$$

The true optimal value is 8000. Here are a few iterations:

Iteration	Lower bound	Upper bound
1	4000.500000	8001.
2	5656.854295	8000.000063
3	6349.604210	8000.000000
4	6727.171325	8000.000000
5	6964.404510	8000.000000
6	7127.189745	8000.000000
7	7245.789315	8000.000000
8	7336.032345	8000.000000
9	7406.997700	8000.000000
10	7464.263930	8000.000000
11	7511.447285	8000.000000
12	7550.994500	8000.000000
13	7584.620115	8000.000000
14	7613.561225	8000.000000
15	7638.732830	8000.000000

Example

Recall

$$\begin{aligned} \max \quad & 100\left(\frac{1}{2}+i_2-\left(\frac{3}{5}+i_1\right)^2\right)^2+\left(\frac{2}{5}-i_1\right)^2 \\ \text{s. t.} \quad & i_1, i_2 \in [0,200] \cap \mathbb{Z} \end{aligned} \tag{6}$$

Using the bounds on i_1 and i_2 we obtain an upper bound of $165 \cdot 10^9$ for the objective function. Use it to convert the problem into one where all feasible points have a non-negative objective value.

Expanding the new objective function and translating it into a differential operator yields

$$\frac{4124999999947}{25}\mathrm{Id} - 28 z_2 \frac{\partial}{\partial z_2} + \frac{172}{5} z_1 \frac{\partial}{\partial z_1} - 117 \left(z_1 \frac{\partial}{\partial z_1} \right)^{(2)} - 100 \left(z_2 \frac{\partial}{\partial z_2} \right)^{(2)}$$

$$+ 240 \left(z_2 \frac{\partial}{\partial z_2} \right) \left(z_1 \frac{\partial}{\partial z_1} \right) + 200 \left(z_2 \frac{\partial}{\partial z_2} \right) \left(z_1 \frac{\partial}{\partial z_1} \right)^{(2)} - 240 \left(z_1 \frac{\partial}{\partial z_1} \right)^{(3)} - 100 \left(z\right.$$

The short generating function can be written as $g(z_1, z_2) = \left(\frac{1}{1-z_1} - \frac{z_1^{201}}{1-z_1} \right) \left(\frac{1}{1-z_2} - \frac{z_2^{201}}{1-z_2} \right)$.

In this example, the number of lattice points is $|P \cap \mathbb{Z}^2| = 40401$. The first bounds are $L_1 = 139463892042.292155534$, $U_1 = 28032242300500.723262442$. After 30 iterations the bounds become $L_{30} = 164999998845.993553019$ and $U_{30} = 165000000475.892451381$. The new optimal objective value is 164999999999.28.

Mixed Integer Case

What is the computational complexity, of the *non-linear* mixed integer problem?

$$\max \quad f(x_1, \ldots, x_{d_1}, z_1, \ldots, z_{d_2}) : \tag{7a}$$

$$Ax + Bz \leq b \tag{7b}$$

$$x_i \in \mathbb{R} \qquad \text{for } i = 1, \ldots, d_1, \tag{7c}$$

$$z_i \in \mathbb{Z} \qquad \text{for } i = 1, \ldots, d_2, \tag{7d}$$

where f is a polynomial function of maximum total degree D with rational coefficients, and $A \in \mathbb{Z}^{m \times d_1}$, $B \in \mathbb{Z}^{m \times d_2}$, $b \in \mathbb{Z}^m$ (here we assume that $Ax + Bz \leq b$ describes a convex polytope, which we denote by P).

Theorem Let the dimension $d = d_1 + d_2$ be fixed.

There exists a fully polynomial time approximation scheme (FPTAS) for the mixed integer polynomial optimization problem for all polynomial functions $f \in \mathbb{Q}[x_1, \ldots, x_{d_1}, z_1, \ldots, z_{d_2}]$ that are non-negative on the feasible region.

Moreover, the restriction to non-negative polynomials is necessary, as there does not even exist a polynomial time approximation scheme (PTAS) for the maximization of *arbitrary* polynomials over mixed-integer sets in polytopes, even for fixed dimension $d \geq 2$.

OTHER RATIONAL FUNCTION TECHNIQUES

THE PROBLEMS

Given a $d \times n$ integral matrix A and integral d-vectors c, b. Solve:

- **General**

$$\text{maximize } cx \text{ subject to } x \in P = \{x | Ax = b,\ x \geq 0\} \cap \mathbb{Z}^d,$$

- **Binary**

$$\text{maximize } cx \text{ subject to } x \in P = \{x | Ax = b,\ x_i \in \{0, 1\}\},$$

Two Non-linear Models

LEMMA Let A_i denote the columns of the matrix A.

$$\prod_{j=1}^{n}(1 + z^{A_j} t^{c_j}) = \sum_{\text{over feasible } b} \left(\sum_{\alpha \in P} t^{c\alpha} \right) z^b.$$

There is a monomial $t^{\beta} x^b$ in expansion if and only if there is a $0/1$ vertex of $P = \{x | Ax = b, 0 \leq x \leq 1, \}$ of cost value β.

LEMMA Let A_i denote the columns of the matrix A.

$$\frac{1}{\prod_{j=1}^{n}(1 - z^{A_j} t^{c_j})} = \sum t^{\beta} z^b.$$

There is a monomial $t^{\beta} x^b$ if and only if $P = \{x | Ax = b, x \geq 0, integer\}$ has a lattice point of cost value β.

Example

maximize $x + 2y + z$

subject to $\{(x, y, z) \in \mathbb{Z}^3 \mid 3x + 5y + 17z = b,\ x \geq 0, y \geq 0, z \geq 0\}$

It is encoded, for *any* right-hand-side into

$$\left(1 + x^3 t\right)\left(1 + x^5 t^2\right)\left(1 + x^{17} t\right)$$

This is a COMPACT representation of any optimal value. In expanded form

$$1 + x^{17}t + x^5t^2 + x^{22}t^3 + x^3t + x^{20}t^2 + x^8t^3 + x^{25}t^4.$$

Suppose now it is not bounded,

$$\frac{1}{(1 - x^3 t)(1 - x^5 t^2)(1 - x^{17} t)}.$$

Its Multivariate Taylor Series expansion is

$$1 + +x^3t + x^5t^2 + x^6t^2 + x^8t^3 + x^9t^3 + x^{10}t^4 + x^{11}t^4 + x^{12}t^4 + x^{13}t^5 + x^{14}t^5 + \left(t^6 + t^5\right)x^{15} + x^{16}t^6 + \left(t^6 + t\right)x^{17} + \left(t^7 + t^6\right)x^{18} + x^{19}t^7 + \left(t^8 + t^7 + t^2\right)x^{20} + \left(t^8 + t^7\right)x^{21} + \left(t^8 + t^3\right)x^{22} + \left(t^9 + t^8 + t^3\right)x^{23} + \left(t^9 + t^8\right)x^{24} + \left(t^{10} + t^9 + t^4\right)x^{25} + + \left(t^{10} + t^9 + t^4\right)x^{26} + \left(t^{10} + t^9 + t^5\right)x^{27} + \left(t^{11} + t^{10} + t^5\right)x^{28} + \ldots ...$$

Note that we have $t^{10}x^{28}$ because we have ONE Knapsack solution $x = 6, y = 2, z = 0$

IMPORTANT Note that if $t = 1$, we COUNT lattice points.

SELECTING A COEFFICIENT!!

Let $\phi_A(b)$ be the coefficient of $z^b := z_1^{b_1} \cdots z_m^{b_m}$ of the function

$$f(z) = \frac{1}{(1 - z^{A_1}) \cdots (1 - z^{\mathcal{A}_d})}$$

expanded as a power series centered at $z = 0$.

Fast Fourier Transforms

THEOREM The coefficients of the product of n polynomials of degree d_i can be computed by FFT in $O(dln(d)ln(n))$ arithmetic operations where $d = \sum d_i$.

IDEA:

- Polynomials represented as monomials are wasteful!
- *Represent* polynomials $p_1(x), p_2(x)$ of degree d sets of $2d + 1$ points $(y_j, p_i(y_j))$.
- *Pointwise multiply* these values to get $p_1(x)p_2(x)$. We get the point representation of $p_1(x)p_2(x)$. Takes $O(d)$
- Choose the values of evaluation cleverly y_i comes from roots of unity! This can be done in $O(d \log(d))$ using the **Fast Fourier Transform**.

- *Interpolate* to create the coefficient representation of the polynomial $p_1(x)p_2(x)$ through an application of FFT transform, again takes $O(d\log(d))$ operations.

- VERY fast code available now, parallelizable.

- Idea goes back at least to the 1970's when Statisticians used it for enumerating contingency tables with given margins. See Diaconis-Gangolli 1995 survey.

A Case Study: $0/1$-Knapsack Problems

THEOREM Using dynamic programming one can solve the knapsack problem

maximize cx **subject to** $a_1x_1 + a_2x_2 + \cdots + a_nx_n = b \; x_i \in \{0, 1\}$,

in $O(nb)$ steps.

THEOREM (Nesterov 2004) The same knapsack problem, using Fast Fourier Transforms, can be solved in $O(b \log^2(n))$ steps.

QUESTION: How does this idea behave in practice? Can one improve the complexity further?

Multivariate Complex Analysis view

Work by Beck et al., Lasserre et al., Pemantle et al.

$$\phi_A(b) = \frac{1}{(2\pi i)^m} \int_{|z_1|=\epsilon_1} \cdots \int_{|z_m|=\epsilon_m} \frac{z_1^{-b_1-1} \cdots z_m^{-b_m-1}}{(1 - z^{A_1}) \cdots (1 - z^{A_d})} \, dz \ .$$

Here $0 < \epsilon_1, \ldots, \epsilon_m < 1$ are different numbers such that we can expand all the $\dfrac{1}{1 - z^{M_k}}$ into the power series about 0.

VISIT:
www.math.ucdavis.edu/~latte

www.math.ucdavis.edu/~totalresidue

with lots of nice stuff about lattice points on polytopes...

THANK YOU!