An Implementation of the Barvinok–Woods Integer Projection Algorithm

Matthias Köppe
(with Sven Verdoolaege and Kevin M. Woods)

University of California, Davis
Department of Mathematics

Outline

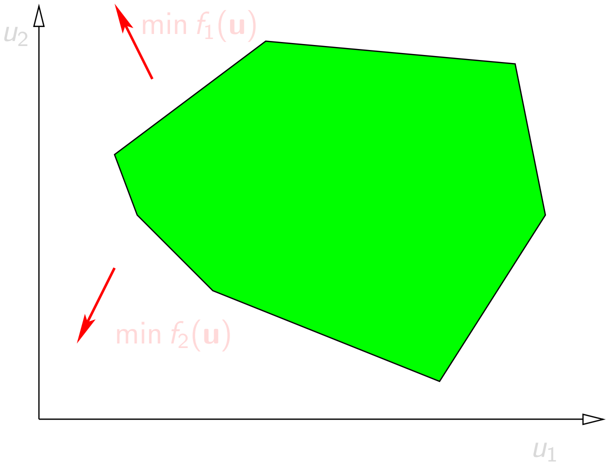
Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \leq \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \ldots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{aligned} \text{vmin} \quad & \big(f_1(\mathbf{u}), \ldots, f_k(\mathbf{u})\big) \\ \text{subject to} \quad & A\mathbf{u} \leq \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{aligned}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = \big(f_1(\mathbf{u}), \ldots, f_k(\mathbf{u})\big)$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \leq f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.



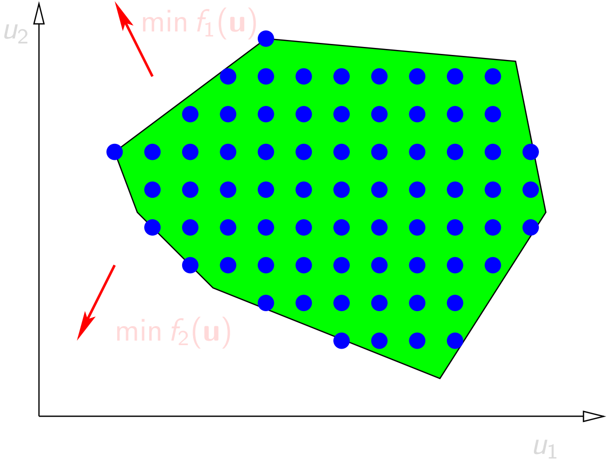
Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \le \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \ldots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{aligned} \text{vmin} \quad & (f_1(\mathbf{u}), \ldots, f_k(\mathbf{u})) \\ \text{subject to} \quad & A\mathbf{u} \le \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{aligned}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), \ldots, f_k(\mathbf{u}))$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \le f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.



Multicriterion integer linear programming problems

De Loera, Hemmecke, K., 2007

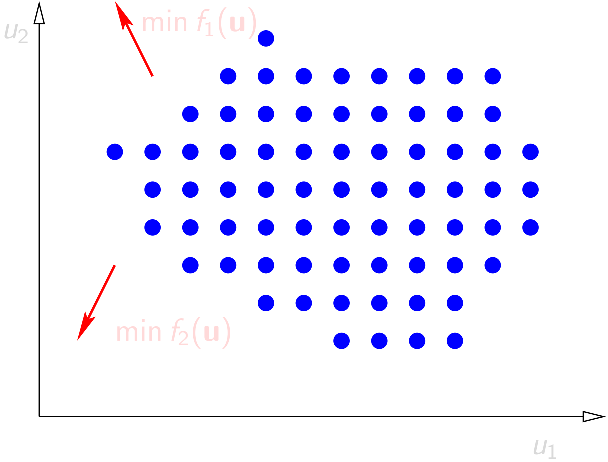
Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \leq \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \dots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{aligned} \text{vmin} \quad & (f_1(\mathbf{u}), \dots, f_k(\mathbf{u})) \\ \text{subject to} \quad & A\mathbf{u} \leq \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{aligned}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), \dots, f_k(\mathbf{u}))$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \leq f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.



Multicriterion integer linear programming problems

De Loera, Hemmecke, K., 2007

Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \leq \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \dots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{aligned} \text{vmin} \quad & \big(f_1(\mathbf{u}), \dots, f_k(\mathbf{u})\big) \\ \text{subject to} \quad & A\mathbf{u} \leq \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{aligned}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = \big(f_1(\mathbf{u}), \dots, f_k(\mathbf{u})\big)$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \leq f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.

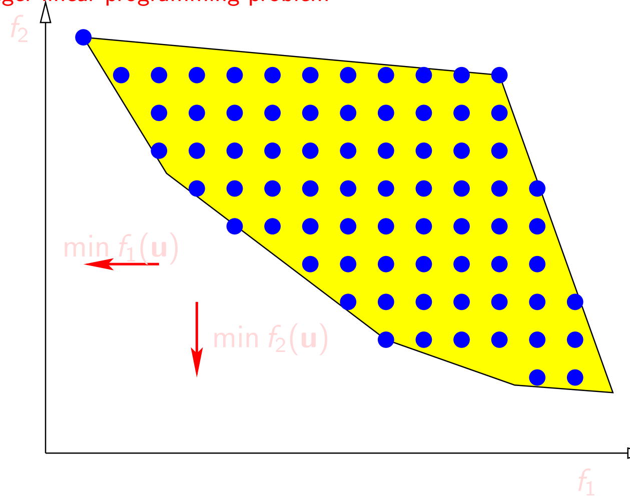
Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \leq \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \dots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{aligned} \text{vmin} \quad & (f_1(\mathbf{u}), \dots, f_k(\mathbf{u})) \\ \text{subject to} \quad & A\mathbf{u} \leq \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{aligned}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), \dots, f_k(\mathbf{u}))$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \leq f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.



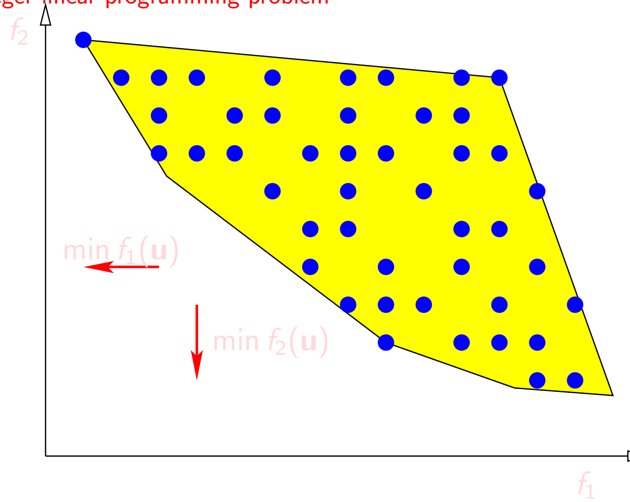
Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \leq \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \ldots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{aligned} \text{vmin} \quad & (f_1(\mathbf{u}), \ldots, f_k(\mathbf{u})) \\ \text{subject to} \quad & A\mathbf{u} \leq \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{aligned}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), \ldots, f_k(\mathbf{u}))$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \leq f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.



Multicriterion integer linear programming problems

De Loera, Hemmecke, K., 2007

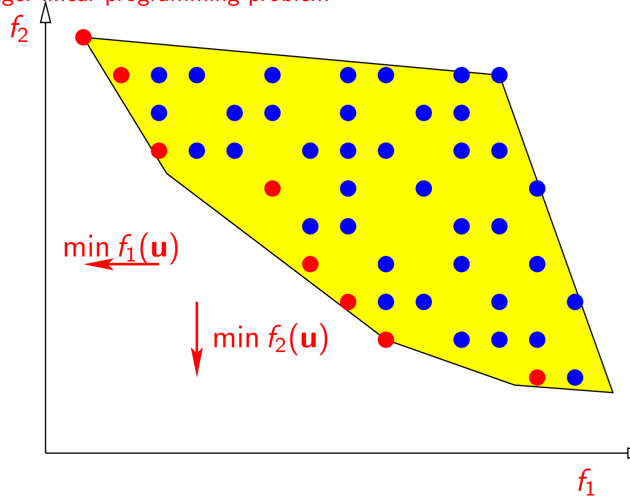
Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \leq \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \ldots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{aligned} \text{vmin} \quad & (f_1(\mathbf{u}), \ldots, f_k(\mathbf{u})) \\ \text{subject to} \quad & A\mathbf{u} \leq \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{aligned}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), \ldots, f_k(\mathbf{u}))$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \leq f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.



Multicriterion integer linear programming problems

De Loera, Hemmecke, K., 2007

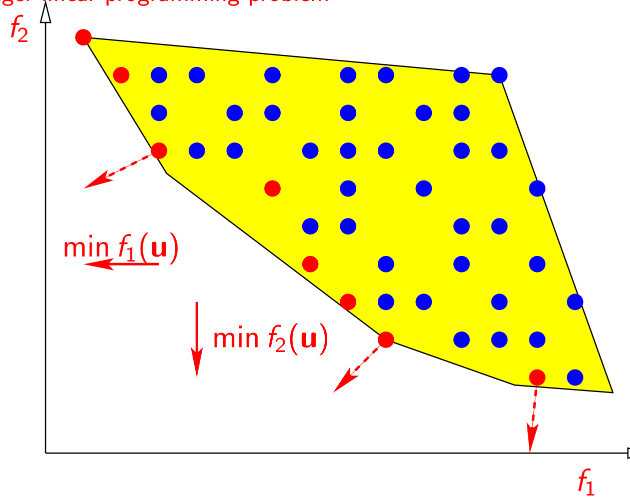
Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \leq \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \dots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{array}{rl} \text{vmin} & (f_1(\mathbf{u}), \dots, f_k(\mathbf{u})) \\ \text{subject to} & A\mathbf{u} \leq \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{array}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), \dots, f_k(\mathbf{u}))$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \leq f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.



Multicriterion integer linear programming problems

De Loera, Hemmecke, K., 2007

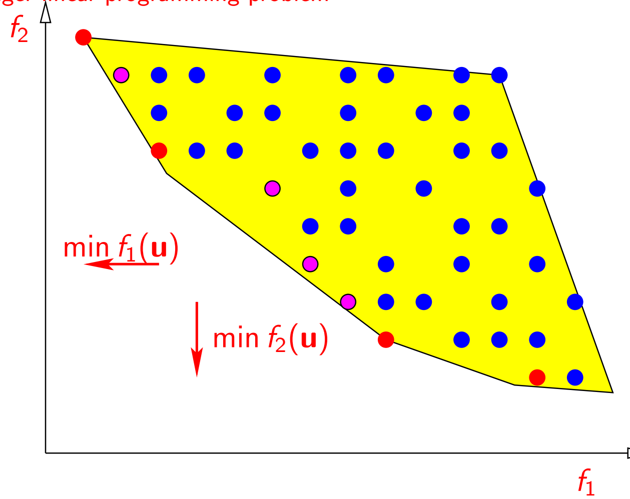
Let $A = (a_{ij})$ be an integral $m \times n$-matrix and $\mathbf{b} \in \mathbf{Z}^m$ such that the convex polyhedron $P = \{\, \mathbf{u} \in \mathbf{R}^n : A\mathbf{u} \le \mathbf{b} \,\}$ is bounded. Given k linear functionals $f_1, f_2, \dots, f_k \in \mathbf{Z}^n$, we consider the multicriterion integer linear programming problem

$$\begin{aligned} \text{vmin} \quad & (f_1(\mathbf{u}), \dots, f_k(\mathbf{u})) \\ \text{subject to} \quad & A\mathbf{u} \le \mathbf{b} \\ & \mathbf{u} \in \mathbf{Z}^n \end{aligned}$$

An outcome vector

$$\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), \dots, f_k(\mathbf{u}))$$

is a Pareto optimum if and only if there is no other feasible point $\tilde{\mathbf{u}}$ such that $f_i(\tilde{\mathbf{u}}) \le f_i(\mathbf{u})$ for all i and $f_j(\tilde{\mathbf{u}}) < f_j(\mathbf{u})$ for at least one index j.



Bilevel integer linear optimization

K., Queyranne, Ryan, 2008

Bilevel integer linear programming problem

Consider the bilevel integer linear programming problem

$$\begin{aligned} \min \quad & f(\mathbf{x}) + h(\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{z} \in Z \subseteq \mathbf{R}^d \\ & \mathbf{x} \in \operatorname{Argmin}\{\, \psi^\top \mathbf{x} : A\mathbf{x} \leq \pi(\mathbf{z}),\ \mathbf{x} \in \mathbf{Z}^n \,\}, \end{aligned}$$

where Z is a polytope, $\psi \in \mathbf{Q}^n$, and $\pi\colon \mathbf{R}^d \to \mathbf{R}^m$, $f\colon \mathbf{R}^n \to \mathbf{R}$, and $h\colon \mathbf{R}^d \to \mathbf{R}$ are affine-linear functions. We assume that all sets $\{\, \mathbf{x} \in \mathbf{Z}^n : A\mathbf{x} \leq \pi(\mathbf{z}) \,\}$ are contained in a polytope.

Hardness

Not even in NP.

Bilevel integer linear programming problem

Consider the bilevel integer linear programming problem

$$\begin{aligned} \min \quad & f(\mathbf{x}) + h(\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{z} \in Z \subseteq \mathbf{R}^d \\ & \mathbf{x} \in \operatorname{Argmin}\{\, \psi^\top \mathbf{x} : A\mathbf{x} \leq \pi(\mathbf{z}),\ \mathbf{x} \in \mathbf{Z}^n \,\}, \end{aligned}$$

where Z is a polytope, $\psi \in \mathbf{Q}^n$, and $\pi\colon \mathbf{R}^d \to \mathbf{R}^m$, $f\colon \mathbf{R}^n \to \mathbf{R}$, and $h\colon \mathbf{R}^d \to \mathbf{R}$ are affine-linear functions. We assume that all sets $\{\, \mathbf{x} \in \mathbf{Z}^n : A\mathbf{x} \leq \pi(\mathbf{z}) \,\}$ are contained in a polytope.

Hardness

Not even in NP.

Consider the set of pure k-player Nash equilibria, for instance in an integer-splittable weighted network congestion game.
Let the strategy of each player $i \in \{1, \dots, k\}$ be described by a vector $\mathbf{u}_i \in U_i \cap \mathbf{Z}^{d_i}$, let $d = d_1 + \cdots + d_k$, where $U_i \subseteq \mathbf{R}^{d_i}$ is a polytope.
Let $F = U_1 \times \cdots \times U_k$.
Let the payoff function of each player i be a piecewise linear concave function $\psi_i(\mathbf{u}_1, \dots, \mathbf{u}_k)$, which the player seeks to maximize.
A pure Nash equilibrium $\bar{\mathbf{u}} = (\bar{\mathbf{u}}_1, \dots, \bar{\mathbf{u}}_k)$ is characterized by the following conditions.

(i) $\bar{\mathbf{u}}$ is a feasible strategy combination, i.e.,

$$\bar{\mathbf{u}} = (\bar{\mathbf{u}}_1, \dots, \bar{\mathbf{u}}_k) \in F \tag{1}$$

(ii) For all $i = 1, \dots, k$,

$$\nexists \mathbf{u}_i \in \mathbf{Z}^{d_i} \text{ with } \tilde{\mathbf{u}}^{(i)} := (\bar{\mathbf{u}}_1, \dots, \bar{\mathbf{u}}_{i-1}, \mathbf{u}_i, \bar{\mathbf{u}}_{i+1}, \dots, \bar{\mathbf{u}}_k) \in F \quad \text{and } \psi_i(\tilde{\mathbf{u}}^{(i)}) > \psi_i(\bar{\mathbf{u}}) \tag{2}$$

Program Transformations in Optimizing Compilers

Verdoolaege et al., 2004–2007

Perform program transformations to

- reduce total memory requirements or working set
- parallelize

Need to count

- number of memory elements, communication volume, ...

Example: Count array elements accessed in a nested loop

```
for (j = 1; j <= p; ++j)
    for (i = 1; i <= 8; ++i)
        a[6*i+9*j-7] = a[6*i+9*j-7] + 5;
```

Equal to the number of elements in

$$S_p = \{\, \ell \in \mathbf{Z} \mid \exists i, j \in \mathbf{Z} : \ell = 6i + 9j - 7,\ 1 \leq j \leq p,\ 1 \leq i \leq 8 \,\}$$

$$\#S_p = \begin{cases} 8 & \text{if } p = 1 \\ 3p + 10 & \text{if } p \geq 2 \end{cases}$$

Program Transformations in Optimizing Compilers

Verdoolaege et al., 2004–2007

Perform program transformations to

- reduce total memory requirements or working set
- parallelize

Need to count

- number of memory elements, communication volume, . . .

Example: Count array elements accessed in a nested loop

```
for (j = 1; j <= p; ++j)
    for (i = 1; i <= 8; ++i)
        a[6*i+9*j-7] = a[6*i+9*j-7] + 5;
```

Equal to the number of elements in

$$S_p = \{\, \ell \in \mathbf{Z} \mid \exists i, j \in \mathbf{Z} : \ell = 6i + 9j - 7,\ 1 \le j \le p,\ 1 \le i \le 8 \,\}$$

$$\#S_p = \begin{cases} 8 & \text{if } p = 1 \\ 3p + 10 & \text{if } p \ge 2 \end{cases}$$

Program Transformations in Optimizing Compilers

Verdoolaege et al., 2004–2007

Perform program transformations to

- reduce total memory requirements or working set
- parallelize

Need to count

- number of memory elements, communication volume, ...

Example: Count array elements accessed in a nested loop

```
for (j = 1; j <= p; ++j)
    for (i = 1; i <= 8; ++i)
        a[6*i+9*j-7] = a[6*i+9*j-7] + 5;
```

Equal to the number of elements in

$$S_p = \{\, \ell \in \mathbf{Z} \mid \exists i, j \in \mathbf{Z} : \ell = 6i + 9j - 7,\ 1 \leq j \leq p,\ 1 \leq i \leq 8 \,\}$$

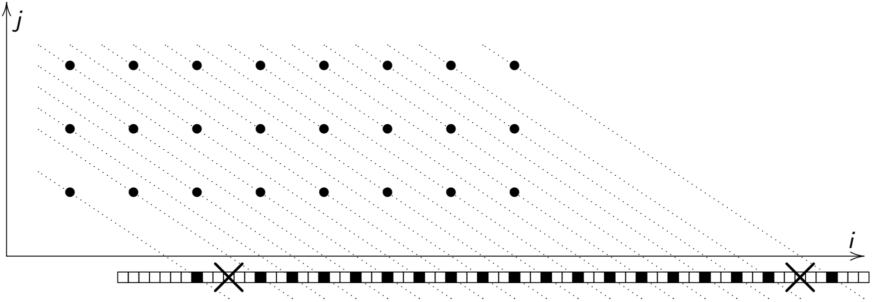
$$\#S_p = \begin{cases} 8 & \text{if } p = 1 \\ 3p + 10 & \text{if } p \geq 2 \end{cases}$$

Counting Example

```
for (j = 1; j <= p; ++j)
    for (i = 1; i <= 8; ++i)
        a[6*i+9*j-7] = a[6*i+9*j-7] + 5;
```

Array reference: `a[6*i+9*j-7]`

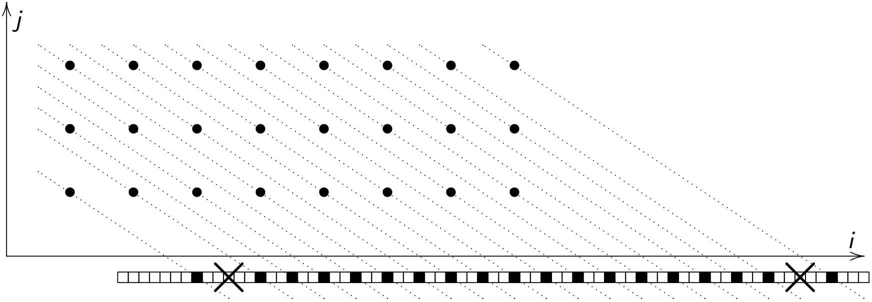
Array elements accessed for $p = 3$:



Counting Example

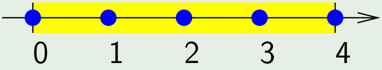
```
for (j = 1; j <= p; ++j)
    for (i = 1; i <= 8; ++i)
        a[6*i+9*j-7] = a[6*i+9*j-7] + 5;
```

Array reference: `a[6*i+9*j-7]`
Array elements accessed for $p = 3$:



Barvinok's short rational generating functions

Generating functions



$$g_P(z) = z^0 + z^1 + z^2 + z^3 + z^4$$

Theorem (Alexander Barvinok, 1994)

Let the dimension d be fixed. There is a polynomial-time algorithm for computing a representation of the generating function

$$g_P(z_1, \dots, z_d) = \sum_{(\alpha_1,\dots,\alpha_d)\in P\cap\mathbf{Z}^d} z_1^{\alpha_1} \cdots z_d^{\alpha_d} = \sum_{\alpha\in P\cap\mathbf{Z}^d} \mathbf{z}^{\alpha}$$

of the integer points $P \cap \mathbf{Z}^d$ of a polyhedron $P \subset \mathbf{R}^d$ (given by rational inequalities) in the form of a rational function.

Corollary

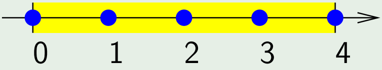
In particular,

$$N = |P \cap \mathbf{Z}^d| = g_P(\mathbf{1})$$

can be computed in polynomial time (in fixed dimension).

Barvinok's short rational generating functions

Generating functions



$$g_P(z) = z^0 + z^1 + z^2 + z^3 + z^4 = \frac{1 - z^5}{1 - z} \quad \text{for } z \neq 1$$

Theorem (Alexander Barvinok, 1994)

Let the dimension d be fixed. There is a polynomial-time algorithm for computing a representation of the generating function

$$g_P(z_1, \dots, z_d) = \sum_{(\alpha_1,\dots,\alpha_d) \in P \cap \mathbf{Z}^d} z_1^{\alpha_1} \cdots z_d^{\alpha_d} = \sum_{\alpha \in P \cap \mathbf{Z}^d} \mathbf{z}^{\alpha}$$

of the integer points $P \cap \mathbf{Z}^d$ of a polyhedron $P \subset \mathbf{R}^d$ (given by rational inequalities) in the form of a rational function.

Corollary

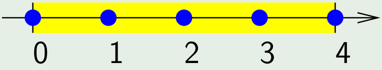
In particular,

$$N = |P \cap \mathbf{Z}^d| = g_P(\mathbf{1})$$

can be computed in polynomial time (in fixed dimension).

Barvinok's short rational generating functions

Generating functions



$$g_P(z) = z^0 + z^1 + z^2 + z^3 + z^4 = \frac{1 - z^5}{1 - z} \quad \text{for } z \neq 1$$

Theorem (Alexander Barvinok, 1994)

Let the dimension d be fixed. There is a polynomial-time algorithm for computing a representation of the generating function

$$g_P(z_1, \dots, z_d) = \sum_{(\alpha_1,\dots,\alpha_d) \in P \cap \mathbf{Z}^d} z_1^{\alpha_1} \cdots z_d^{\alpha_d} = \sum_{\boldsymbol{\alpha} \in P \cap \mathbf{Z}^d} \mathbf{z}^{\boldsymbol{\alpha}}$$

of the integer points $P \cap \mathbf{Z}^d$ of a polyhedron $P \subset \mathbf{R}^d$ (given by rational inequalities) in the form of a rational function.

Corollary

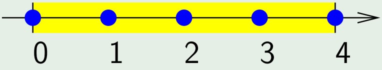
In particular,

$$N = |P \cap \mathbf{Z}^d| = g_P(\mathbf{1})$$

can be computed in polynomial time (in fixed dimension).

Barvinok's short rational generating functions

Generating functions



$$g_P(z) = z^0 + z^1 + z^2 + z^3 + z^4 = \frac{1 - z^5}{1 - z} \quad \text{for } z \neq 1$$

Theorem (Alexander Barvinok, 1994)

Let the dimension d be fixed. There is a polynomial-time algorithm for computing a representation of the generating function

$$g_P(z_1, \ldots, z_d) = \sum_{(\alpha_1,\ldots,\alpha_d) \in P \cap \mathbf{Z}^d} z_1^{\alpha_1} \cdots z_d^{\alpha_d} = \sum_{\boldsymbol{\alpha} \in P \cap \mathbf{Z}^d} \mathbf{z}^{\boldsymbol{\alpha}}$$

of the integer points $P \cap \mathbf{Z}^d$ of a polyhedron $P \subset \mathbf{R}^d$ (given by rational inequalities) in the form of a rational function.

Corollary

In particular,

$$N = |P \cap \mathbf{Z}^d| = g_P(\mathbf{1})$$

can be computed in polynomial time (in fixed dimension).

First Practical Implementation: LattE

From the website http://www.math.ucdavis.edu/~latte/

LattE is a computer software dedicated to the problems of counting and detecting lattice points inside convex polytopes, and the solution of integer programs. LattE contains the first ever implementation of Barvinok's algorithm. LattE stands for Lattice point Enumeration.

Developed 2002–3 by:

Jesús De Loera
David Haws
Raymond Hemmecke
Peter Huggins
Ruriko Yoshida
Jeremy Tauzer

Capabilities

- Count lattice points
- Compute multivariate generating functions
- Compute Ehrhart series
- Experimental integer programming algorithms

First Practical Implementation: LattE

From the website `http://www.math.ucdavis.edu/~latte/`

LattE is a computer software dedicated to the problems of counting and detecting lattice points inside convex polytopes, and the solution of integer programs. LattE contains the first ever implementation of Barvinok's algorithm. LattE stands for Lattice point Enumeration.

Developed 2002–3 by:

Jesús De Loera
David Haws
Raymond Hemmecke
Peter Huggins
Ruriko Yoshida
Jeremy Tauzer

Capabilities

- Count lattice points
- Compute multivariate generating functions
- Compute Ehrhart series
- Experimental integer programming algorithms

First Practical Implementation: LattE

From the website http://www.math.ucdavis.edu/~latte/

LattE is a computer software dedicated to the problems of counting and detecting lattice points inside convex polytopes, and the solution of integer programs. LattE contains the first ever implementation of Barvinok's algorithm. LattE stands for Lattice point Enumeration.

Developed 2002–3 by:

Jesús De Loera
David Haws
Raymond Hemmecke
Peter Huggins
Ruriko Yoshida
Jeremy Tauzer

Capabilities

- Count lattice points
- Compute multivariate generating functions
- Compute Ehrhart series
- Experimental integer programming algorithms

First Practical Implementation: LattE

From the website http://www.math.ucdavis.edu/~latte/

LattE is a computer software dedicated to the problems of counting and detecting lattice points inside convex polytopes, and the solution of integer programs. LattE contains the first ever implementation of Barvinok's algorithm. LattE stands for Lattice point Enumeration.

Developed 2002–3 by:

Jesús De Loera
David Haws
Raymond Hemmecke
Peter Huggins
Ruriko Yoshida
Jeremy Tauzer

Capabilities

- Count lattice points
- Compute multivariate generating functions
- Compute Ehrhart series
- Experimental integer programming algorithms

S. Verdoolaege's `barvinok` library (2005–)

Parametric counting problems

LattE macchiato – an improved version of LattE (K., 2006–)

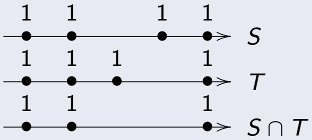
LattE macchiato – an improved version of LattE (K., 2006–)

Set Intersection

$$f(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})\,\mathbf{x}^{\mathbf{s}} \qquad g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$

Hadamard product:

$$f(\mathbf{x}) \star g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$

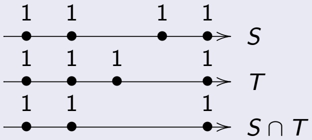


Set Intersection

$$f(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})\,\mathbf{x}^{\mathbf{s}} \qquad g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$

Hadamard product:

$$f(\mathbf{x}) \star g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$



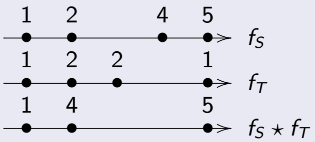
Boolean Operations: Products of Generating Functions

Set Intersection

$$f(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})\,\mathbf{x}^{\mathbf{s}} \qquad g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$

Hadamard product:

$$f(\mathbf{x}) \star g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$



Multi-sets: counts pairs of occurrences

Union and set difference (not for multisets)

$$[\mathbf{s} \in S \setminus T] = [\mathbf{s} \in S] - [\mathbf{s} \in S \cap T]$$
$$[\mathbf{s} \in S \cup T] = [\mathbf{s} \in S] + [\mathbf{s} \in T] - [\mathbf{s} \in S \cap T]$$

Theorem (Barvinok and Woods (2003))

The rational generating function of arbitrary fixed-length Boolean combinations of sets given by Barvinok-style rational generating functions can be computed in polynomial time in fixed dimension.

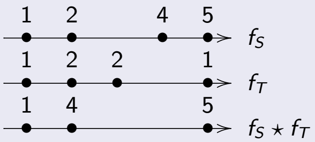
Boolean Operations: Products of Generating Functions

Set Intersection

$$f(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})\,\mathbf{x}^{\mathbf{s}} \qquad g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$

Hadamard product:

$$f(\mathbf{x}) \star g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$



Multi-sets: counts pairs of occurrences

Union and set difference (not for multisets)

$$[\mathbf{s} \in S \setminus T] = [\mathbf{s} \in S] - [\mathbf{s} \in S \cap T]$$
$$[\mathbf{s} \in S \cup T] = [\mathbf{s} \in S] + [\mathbf{s} \in T] - [\mathbf{s} \in S \cap T]$$

Theorem (Barvinok and Woods (2003))

The rational generating function of arbitrary fixed-length Boolean combinations of sets given by Barvinok-style rational generating functions can be computed in polynomial time in fixed dimension.

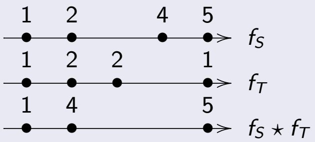
Boolean Operations: Products of Generating Functions

Set Intersection

$$f(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})\,\mathbf{x}^{\mathbf{s}} \qquad g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$

Hadamard product:

$$f(\mathbf{x}) \star g(\mathbf{x}) = \sum_{\mathbf{s}\in\mathbf{Z}^d} a(\mathbf{s})b(\mathbf{s})\,\mathbf{x}^{\mathbf{s}}$$



Multi-sets: counts pairs of occurrences

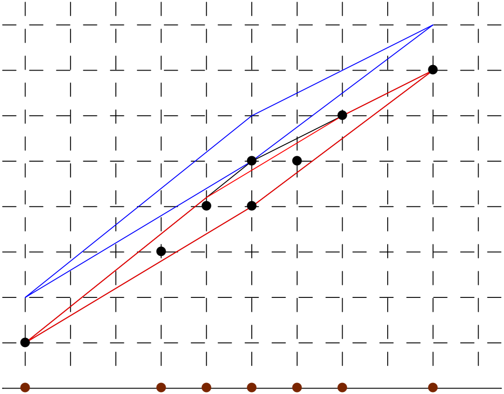
Union and set difference (not for multisets)

$$[\mathbf{s} \in S \setminus T] = [\mathbf{s} \in S] - [\mathbf{s} \in S \cap T]$$
$$[\mathbf{s} \in S \cup T] = [\mathbf{s} \in S] + [\mathbf{s} \in T] - [\mathbf{s} \in S \cap T]$$

Theorem (Barvinok and Woods (2003))

The rational generating function of arbitrary fixed-length Boolean combinations of sets given by Barvinok-style rational generating functions can be computed in polynomial time in fixed dimension.

Integer Projection: The Codimension-1 Case



P

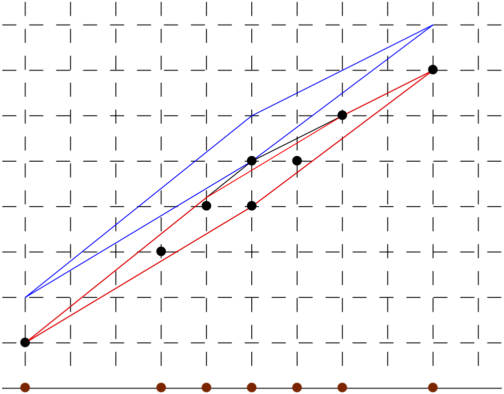
$P + \mathbf{e}_{n+d+1}$

$S' = P \setminus (P + \mathbf{e}_{n+d+1})$

$S = \pi_{n+d} S'$

(one-to-one projection)

Integer Projection: The Codimension-1 Case



P

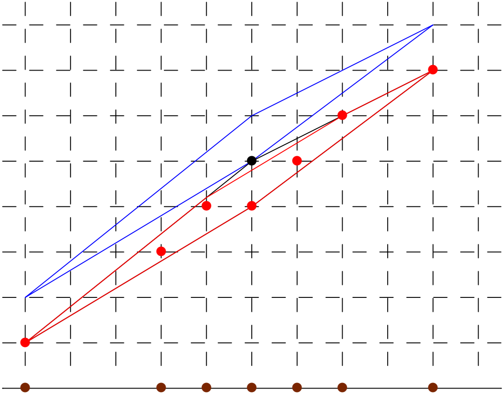
$P + \mathbf{e}_{n+d+1}$

$S' = P \setminus (P + \mathbf{e}_{n+d+1})$

$S = \pi_{n+d} S'$

(one-to-one projection)

Integer Projection: The Codimension-1 Case



P

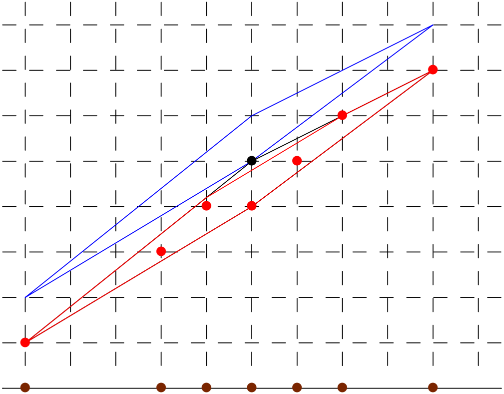
$P + \mathbf{e}_{n+d+1}$

$S' = P \setminus (P + \mathbf{e}_{n+d+1})$

$S = \pi_{n+d} S'$

(one-to-one projection)

Integer Projection: The Codimension-1 Case



P

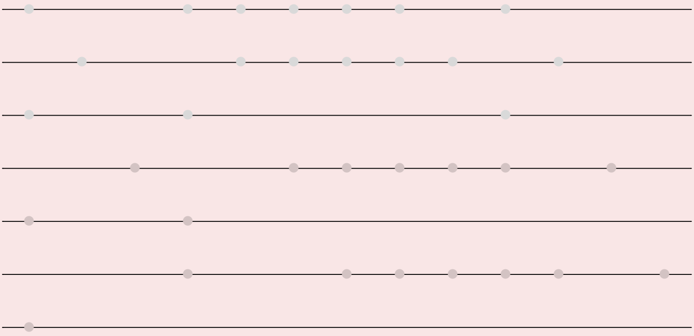
$P + \mathbf{e}_{n+d+1}$

$S' = P \setminus (P + \mathbf{e}_{n+d+1})$

$S = \pi_{n+d} S'$

(one-to-one projection)

Shifting once in second direction is not enough



h

$z_1\, h$

$h - h \star z_1\, h$

$z_1^2\, h$

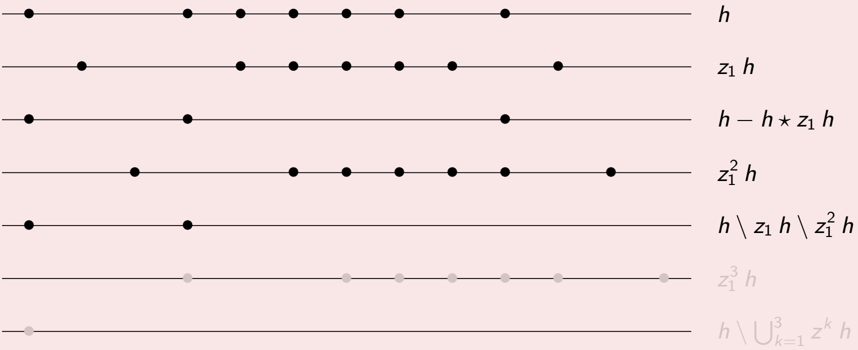
$h \setminus z_1\, h \setminus z_1^2\, h$

$z_1^3\, h$

$h \setminus \bigcup_{k=1}^{3} z^k\, h$

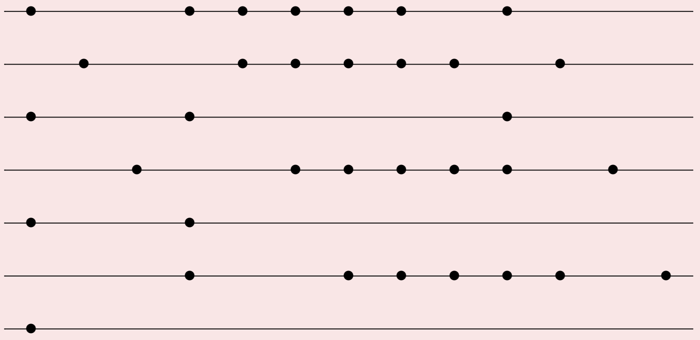
We need to subtract as many shifted copies as the size of the biggest gap

Shifting once in second direction is not enough



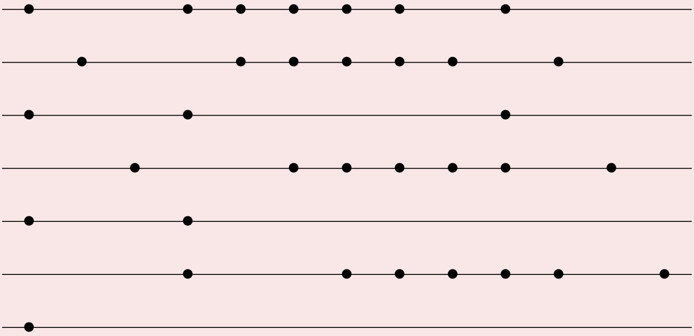
We need to subtract as many shifted copies as the size of the biggest gap

Shifting once in second direction is not enough



We need to subtract as many shifted copies as the size of the biggest gap

Shifting once in second direction is not enough



h

$z_1\, h$

$h - h \star z_1\, h$

$z_1^2\, h$

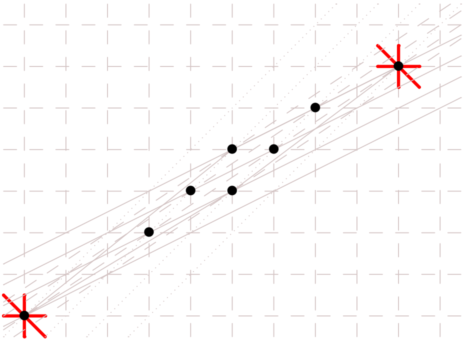
$h \setminus z_1\, h \setminus z_1^2\, h$

$z_1^3\, h$

$h \setminus \bigcup_{k=1}^{3} z^k\, h$

We need to subtract as many shifted copies as the size of the biggest gap

Lattice Widths



width$_{(1,0)}$ $P = 9$

width$_{(0,1)}$ $P = 6$

width$_{(1,1)}$ $P = 15$

width$_{(-1,2)}$ $P = 3$

width$_{(-2,3)}$ $P = 3$

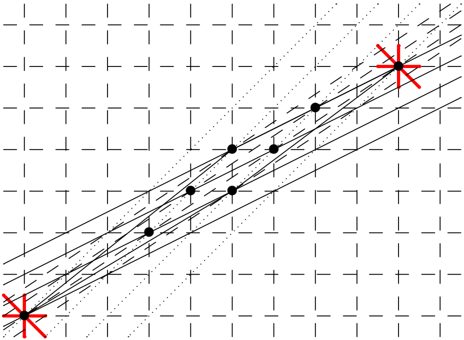
width$_{(1,-1)}$ $P = 3$

Width in direction $\mathbf{c}$:

$$\operatorname{width}_{\mathbf{c}} P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x}\rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x}\rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$

Lattice width of P

$$\operatorname{width} P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \operatorname{width}_{\mathbf{c}} P(\mathbf{p})$$



$\text{width}_{(1,0)}\, P = 9$

$\text{width}_{(0,1)}\, P = 6$

$\text{width}_{(1,1)}\, P = 15$

$\text{width}_{(-1,2)}\, P = 3$

$\text{width}_{(-2,3)}\, P = 3$

$\text{width}_{(1,-1)}\, P = 3$

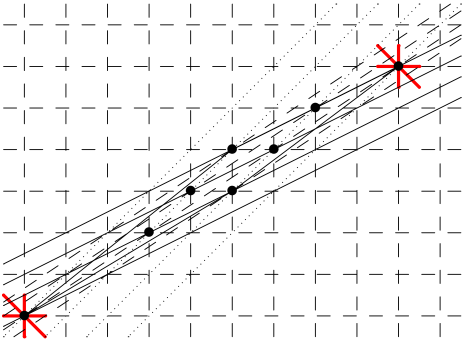
Width in direction **c**:

$$\text{width}_{\mathbf{c}}\, P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p})\,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p})\,\}$$

Lattice width of P

$$\text{width}\, P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \text{width}_{\mathbf{c}}\, P(\mathbf{p})$$

Lattice Widths



$\mathrm{width}_{(1,0)}\, P = 9$

$\mathrm{width}_{(0,1)}\, P = 6$

$\mathrm{width}_{(1,1)}\, P = 15$

$\mathrm{width}_{(-1,2)}\, P = 3$

$\mathrm{width}_{(-2,3)}\, P = 3$

$\mathrm{width}_{(1,-1)}\, P = 3$

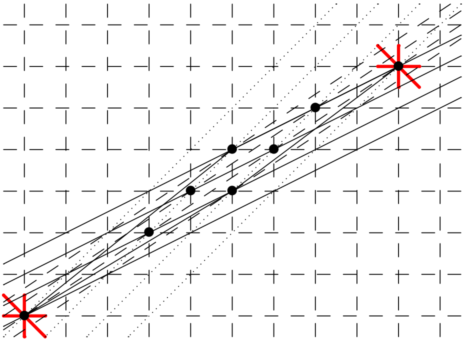
Width in direction **c**:

$$\mathrm{width}_{\mathbf{c}}\, P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x}\rangle \mid \mathbf{x} \in P(\mathbf{p})\,\} - \min\{\, \langle \mathbf{c}, \mathbf{x}\rangle \mid \mathbf{x} \in P(\mathbf{p})\,\}$$

Lattice width of P

$$\mathrm{width}\, P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \mathrm{width}_{\mathbf{c}}\, P(\mathbf{p})$$

Lattice Widths



$\mathrm{width}_{(1,0)}\, P = 9$

$\mathrm{width}_{(0,1)}\, P = 6$

$\mathrm{width}_{(1,1)}\, P = 15$

$\mathrm{width}_{(-1,2)}\, P = 3$

$\mathrm{width}_{(-2,3)}\, P = 3$

$\mathrm{width}_{(1,-1)}\, P = 3$

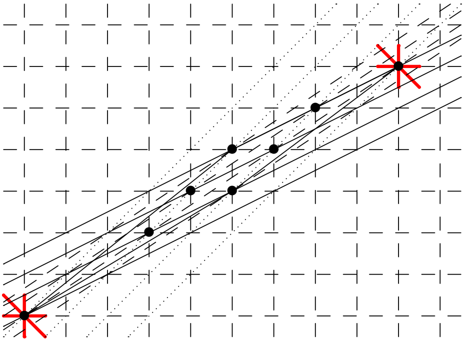
Width in direction **c**:

$$\mathrm{width}_{\mathbf{c}}\, P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$

Lattice width of P

$$\mathrm{width}\, P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \mathrm{width}_{\mathbf{c}}\, P(\mathbf{p})$$

Lattice Widths



$\text{width}_{(1,0)}\, P = 9$

$\text{width}_{(0,1)}\, P = 6$

$\text{width}_{(1,1)}\, P = 15$

$\text{width}_{(-1,2)}\, P = 3$

$\text{width}_{(-2,3)}\, P = 3$

$\text{width}_{(1,-1)}\, P = 3$

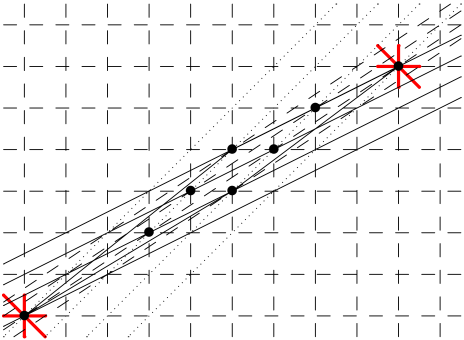
Width in direction $\mathbf{c}$:

$$\text{width}_{\mathbf{c}}\, P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$

Lattice width of P

$$\text{width}\, P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \text{width}_{\mathbf{c}}\, P(\mathbf{p})$$

Lattice Widths



width$_{(1,0)}$ $P = 9$

width$_{(0,1)}$ $P = 6$

width$_{(1,1)}$ $P = 15$

width$_{(-1,2)}$ $P = 3$

width$_{(-2,3)}$ $P = 3$

width$_{(1,-1)}$ $P = 3$

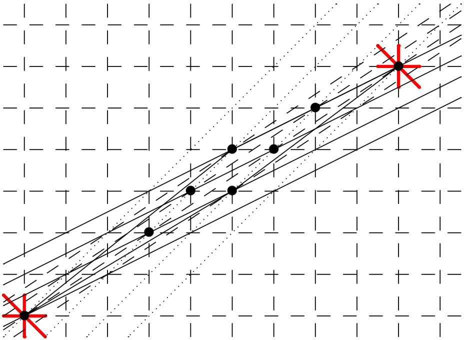
Width in direction **c**:

$$\operatorname{width}_{\mathbf{c}} P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$

Lattice width of P

$$\operatorname{width} P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \operatorname{width}_{\mathbf{c}} P(\mathbf{p})$$

Lattice Widths



$$\mathrm{width}_{(1,0)}\, P = 9$$
$$\mathrm{width}_{(0,1)}\, P = 6$$
$$\mathrm{width}_{(1,1)}\, P = 15$$
$$\mathrm{width}_{(-1,2)}\, P = 3$$
$$\mathrm{width}_{(-2,3)}\, P = 3$$
$$\mathrm{width}_{(1,-1)}\, P = 3$$

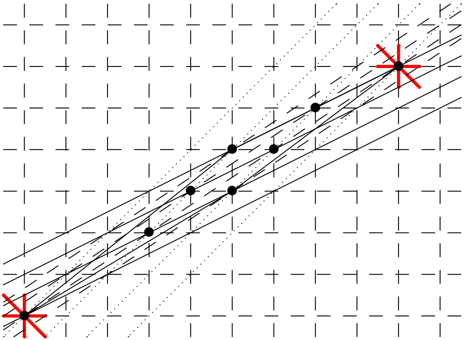
Width in direction **c**:

$$\mathrm{width}_{\mathbf{c}}\, P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$

Lattice width of P

$$\mathrm{width}\, P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \mathrm{width}_{\mathbf{c}}\, P(\mathbf{p})$$

Lattice Widths



$\text{width}_{(1,0)} P = 9$

$\text{width}_{(0,1)} P = 6$

$\text{width}_{(1,1)} P = 15$

$\text{width}_{(-1,2)} P = 3$

$\text{width}_{(-2,3)} P = 3$

$\text{width}_{(1,-1)} P = 3$

Width in direction $\mathbf{c}$:

$$\text{width}_{\mathbf{c}} P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$

Lattice width of P

$$\text{width}\, P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \text{width}_{\mathbf{c}} P(\mathbf{p})$$

Flatness Theory and Small Gaps

As a consequence of flatness theory for lattice-point-free convex bodies, it is possible to construct directions of small gaps for the parametric polytopes.

Theorem (Small-gaps theorem; cf. Barvinok and Woods (2003))

Let $\kappa \geq 1$ and let $\mathbf{c} \in \mathbf{Z}^m$ be a κ-approximative lattice width direction, i.e.,

$$\operatorname{width}_{\mathbf{c}} P_{\mathbf{s},\mathbf{t}} \leq \kappa \cdot \operatorname{width} P_{\mathbf{s},\mathbf{t}}.$$

Then the image $Y = \{ \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P_{\mathbf{s},\mathbf{t}} \cap \mathbf{Z}^d \}$ does not have gaps larger than $\kappa \cdot \omega(m)$.

Here $\omega(m)$ is the flatness constant; it only depends on the dimension m (Lagarias et al., 1990; Barvinok, 2002; Banaszczyk et al., 1999).

Theorem (Kannan (1992))

Let the total dimension $n + d + m$ be fixed. Then there exists a polynomial-time algorithm for the following problem. Given as input, in binary encoding,

- (I_1) *inequalities describing a rational polytope $P \subset \mathbf{R}^{n+d+m}$,*

output, in binary encoding,

- (O_1) *inequality systems describing partially open polyhedra $\tilde{Q}_1, \dots, \tilde{Q}_M \subset \mathbf{R}^{n+d}$ that form a partition of the projection, Q, of P onto the first $n + d$ coordinates,*
- (O_2) *integer vectors $\mathbf{c}_1, \dots, \mathbf{c}_M \in \mathbf{R}^m$,*

such that $\mathbf{c}_i$ is a 2-approximative lattice width direction for every polytope $P_{\mathbf{s},\mathbf{t}}$ when $(\mathbf{s}, \mathbf{t}) \in \tilde{Q}_i$.

Flatness Theory and Small Gaps: Strengthening

As a consequence of flatness theory for lattice-point-free convex bodies, it is possible to construct directions of small gaps for the parametric polytopes.

Theorem (Small-gaps theorem; cf. Barvinok and Woods (2003))

Let $\kappa \geq 1$ and let $\mathbf{c} \in \mathbf{Z}^m$ be a κ-approximative lattice width direction, i.e.,

$$\operatorname{width}_{\mathbf{c}} P_{\mathbf{s},\mathbf{t}} \leq \kappa \cdot \operatorname{width} P_{\mathbf{s},\mathbf{t}}.$$

Then the image $Y = \{ \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P_{\mathbf{s},\mathbf{t}} \cap \mathbf{Z}^d \}$ does not have gaps larger than $\kappa \cdot \omega(m)$.

Here $\omega(m)$ is the flatness constant; it only depends on the dimension m (Lagarias et al., 1990; Barvinok, 2002; Banaszczyk et al., 1999).

Theorem (Eisenbrand–Shmonin, 2008)

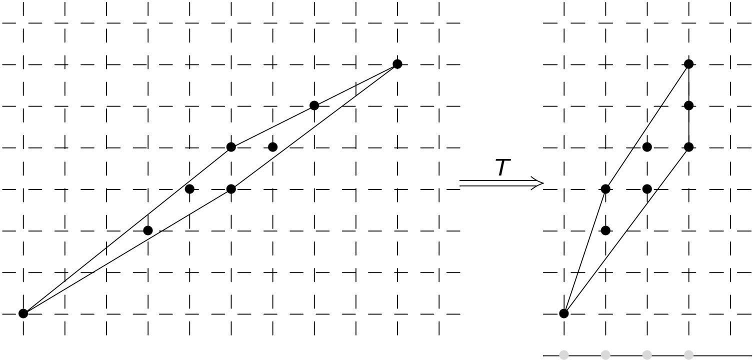
Let the dimension $n + d$ be fixed; m is allowed to vary. Then there exists a polynomial-time algorithm for the following problem. Given as input, in binary encoding,

- (I_1) *inequalities describing a rational polytope $P \subset \mathbf{R}^{n+d+m}$,*

output, in binary encoding,

- (O_1) *inequality systems describing partially open polyhedra $\tilde{Q}_1, \dots, \tilde{Q}_M \subset \mathbf{R}^{n+d}$ that form a partition of the projection, Q, of P onto the first $n + d$ coordinates,*
- (O_2) *integer vectors $\mathbf{c}_1, \dots, \mathbf{c}_M \in \mathbf{R}^m$,*

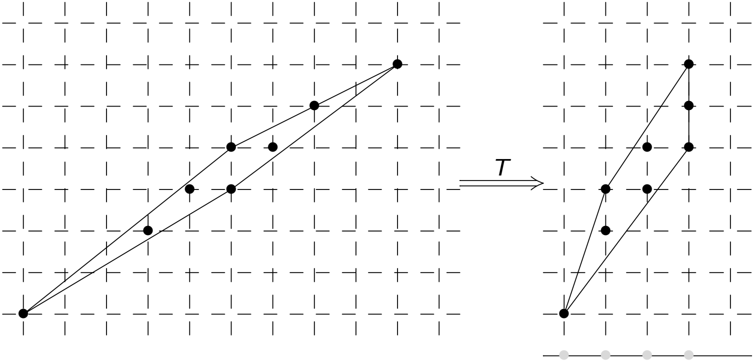
such that $\mathbf{c}_i$ is an exact lattice width direction for every polytope $P_{\mathbf{s},\mathbf{t}}$ when $(\mathbf{s}, \mathbf{t}) \in \tilde{Q}_i$.



Apply unimodular transformation

$$T = \begin{bmatrix} -1 & 2 \\ 0 & 1 \end{bmatrix}$$ lattice width direction

$\Rightarrow$ small gaps ($= 1$ if $m = 2$)

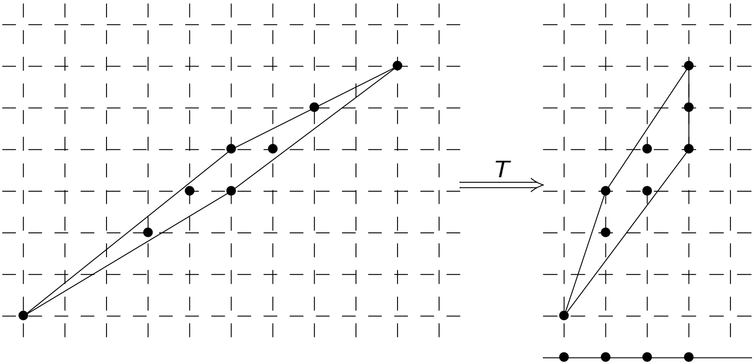


Apply unimodular transformation

$$T = \begin{bmatrix} -1 & 2 \\ 0 & 1 \end{bmatrix}$$ lattice width direction

$\Rightarrow$ small gaps ($= 1$ if $m = 2$)

Small Gaps in Lattice Width Direction



Apply unimodular transformation

$$T = \begin{bmatrix} -1 & 2 \\ 0 & 1 \end{bmatrix}$$ lattice width direction

$\Rightarrow$ small gaps ($= 1$ if $m = 2$)

Lattice Widths Computation: The Eisenbrand–Shmonin Method

Width in direction $\mathbf{c} \in \mathbf{Z}^d$:

$$\operatorname{width}_{\mathbf{c}} P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$

Lattice width of P

$$\operatorname{width} P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \operatorname{width}_{\mathbf{c}} P(\mathbf{p})$$

min and max occur at **vertices** (extremal points of polytope)

- $\Rightarrow$ Consider all pairs of (parametric) vertices
- $\Rightarrow$ Compute candidate width directions for each pair
- $\Rightarrow$ Compute smallest overall width

Lattice Widths Computation: The Eisenbrand–Shmonin Method

Width in direction $\mathbf{c} \in \mathbf{Z}^d$:

$$\text{width}_{\mathbf{c}} P(\mathbf{p}) = \max\{ \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \} - \min\{ \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \}$$

Lattice width of P

$$\text{width } P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \text{width}_{\mathbf{c}} P(\mathbf{p})$$

min and max occur at **vertices** (extremal points of polytope)

- $\Rightarrow$ Consider all pairs of (parametric) vertices
- $\Rightarrow$ Compute candidate width directions for each pair
- $\Rightarrow$ Compute smallest overall width

Lattice Widths Computation: The Eisenbrand–Shmonin Method

Width in direction $\mathbf{c} \in \mathbf{Z}^d$:

$$\operatorname{width}_{\mathbf{c}} P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$

Lattice width of P

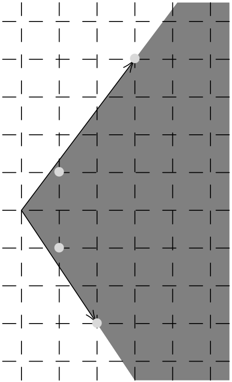
$$\operatorname{width} P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \operatorname{width}_{\mathbf{c}} P(\mathbf{p})$$

min and max occur at **vertices** (extremal points of polytope)

- $\Rightarrow$ Consider all pairs of (parametric) vertices
- $\Rightarrow$ Compute candidate width directions for each pair
- $\Rightarrow$ Compute smallest overall width

Lattice Widths Computation: The Eisenbrand–Shmonin Method II

$$\text{width}_{\mathbf{c}}\, P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$



Candidates for vertex pair $(\mathbf{v}_1(\mathbf{p}), \mathbf{v}_2(\mathbf{p}))$

$$\begin{aligned} \text{width}_{\mathbf{c}}\, P(\mathbf{p}) &= \langle \mathbf{c}, \mathbf{v}_2(\mathbf{p}) \rangle - \langle \mathbf{c}, \mathbf{v}_1(\mathbf{p}) \rangle \\ &= \langle \mathbf{c}, \mathbf{v}_2(\mathbf{p}) - \mathbf{v}_1(\mathbf{p}) \rangle \end{aligned}$$

$$\text{width}\, P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \text{width}_{\mathbf{c}}\, P(\mathbf{p})$$

$\Rightarrow \mathbf{c} \in \left(C_1^* \cap -C_2^* \cap \mathbf{Z}^d\right) \setminus \{\mathbf{0}\}$

$\Rightarrow$ vertices of integer hull of $(C_1^* \cap -C_2^*) \setminus \{\mathbf{0}\}$

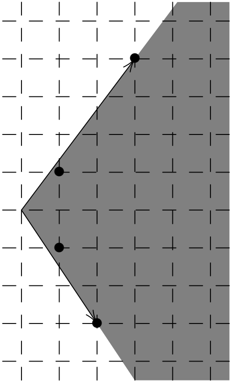
Polynomial-time due to Cook et al. (1992)

Practical implementation

- Compute the convex hull based on a linear optimization oracle
  - implemented using binary search and Generalized Basis Reduction
  - implement using CPLEX?
- Use Hilbert basis computation?

Lattice Widths Computation: The Eisenbrand–Shmonin Method II

$$\operatorname{width}_{\mathbf{c}} P(\mathbf{p}) = \max\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\} - \min\{\, \langle \mathbf{c}, \mathbf{x} \rangle \mid \mathbf{x} \in P(\mathbf{p}) \,\}$$



Candidates for vertex pair $(\mathbf{v}_1(\mathbf{p}), \mathbf{v}_2(\mathbf{p}))$

$$\begin{aligned} \operatorname{width}_{\mathbf{c}} P(\mathbf{p}) &= \langle \mathbf{c}, \mathbf{v}_2(\mathbf{p}) \rangle - \langle \mathbf{c}, \mathbf{v}_1(\mathbf{p}) \rangle \\ &= \langle \mathbf{c}, \mathbf{v}_2(\mathbf{p}) - \mathbf{v}_1(\mathbf{p}) \rangle \end{aligned}$$

$$\operatorname{width} P(\mathbf{p}) = \min_{\mathbf{c} \in \mathbf{Z}^d \setminus \{\mathbf{0}\}} \operatorname{width}_{\mathbf{c}} P(\mathbf{p})$$

$\Rightarrow \mathbf{c} \in \left(C_1^* \cap -C_2^* \cap \mathbf{Z}^d \right) \setminus \{\mathbf{0}\}$

$\Rightarrow$ vertices of integer hull of $(C_1^* \cap -C_2^*) \setminus \{\mathbf{0}\}$

Polynomial-time due to Cook et al. (1992)

Practical implementation

- Compute the convex hull based on a linear optimization oracle
  - implemented using binary search and Generalized Basis Reduction
  - implement using CPLEX?
- Use Hilbert basis computation?

Computational Results

		Problem				
		ex1	*woods*	*pugh*	*p-pugh*	*scarf1*
Parameter variables **s**	n	0	1	0	1	2
Variables **t**	d	0	0	0	0	0
Existentially quant. variables **u**	m	2	2	2	2	2
Inequalities		4	4	4	4	5
Parametric Vertices		4	4	4	4	6
Width directions		7	3	8	8	6
Distinct width directions		4	2	7	7	4
Chambers		1	2	1	6	2
LPs solved in gen. basis red.		8		49	43	4
Without exploiting small gaps in dimension 2						
Time (CPU seconds)		0.11	29.2	0.09	797	126
Exploiting small gaps in dimension 2						
Time (CPU seconds)		0.08	2.7		18.0	1.1

Wojciech Banaszczyk, Alexander E. Litvak, A. Pajor, and S. J. Szarek. The flatness theorem for nonsymmetric convex bodies via the local theory of Banach spaces. Mathematics of Operations Research, 24(3):728–750, August 1999.

Alexander Barvinok. A Course in Convexity, volume 54 of Graduate Studies in Mathematics. American Mathematical Society, Providence, RI, 2002. ISBN 0-8218-2968-8.

Alexander I. Barvinok and Kevin Woods. Short rational generating functions for lattice point problems. Journal of the AMS, 16(4):957–979, 2003.

W. J. Cook, M. E. Hartmann, R. Kannan, and C. McDiarmid. On integer points in polyhedra. Combinatorica, 12(1):27–37, 1992.

R. Kannan. Lattice translates of a polytope and the Frobenius problem. Combinatorica, 12(2):161–177, 1992.

J. C. Lagarias, Hendrik W. Lenstra, Jr., and Claus-Peter Schnorr. Korkin-Zolotarev bases and successive minima of a lattice and its reciprocal lattice. Combinatorica, 10(4): 333–348, 1990.