Integer Relations among Real Numbers

Darya Romanova

Saint-Petersburg State University

Joint Advanced Student School 2007

Outline

Outline

Starting Examples

Let X be a mathematical expression, that can be approximated numerically. (For example a definite integral.)

Starting Examples

Let X be a mathematical expression, that can be approximated numerically. (For example a definite integral.)

Suppose we know, that X is *rational*.

Starting Examples

Let X be a mathematical expression, that can be approximated numerically. (For example a definite integral.)

Suppose we know, that X is *rational*.

- $X \approx 2.33333333333333333\ldots \Rightarrow X =$

Starting Examples

Let X be a mathematical expression, that can be approximated numerically. (For example a definite integral.)

Suppose we know, that X is *rational*.

- $X \approx 2.3333333333333333\ldots \Rightarrow X = 7/3.$

Starting Examples

Let X be a mathematical expression, that can be approximated numerically. (For example a definite integral.)

Suppose we know, that X is *rational*.

- $X \approx 2.3333333333333333\ldots \Rightarrow X = 7/3.$
- $X \approx 0.1412742382271468144044321\ldots \Rightarrow X =$

Starting Examples

Let X be a mathematical expression, that can be approximated numerically. (For example a definite integral.)

Suppose we know, that X is *rational*.

- $X \approx 2.3333333333333333\ldots \Rightarrow X = 7/3.$
- $X \approx 0.1412742382271468144044321\ldots \Rightarrow X = ?$

Continuous Fractions

Let's make X into continuous fraction:

Continuous Fractions

Let's make X into continuous fraction:

$1/0.1412742382271468144 \approx 7.0784313725490196080$

$1/0.0784313725490196078431 39 \approx 12.7499999999999999975$

$1/0.7499999999999999975 \approx 1.3333333333333333778$

$1/0.3333333333333333778 \approx 2.9999999999999995998$

$1/0.9999999999999995998 \approx 1.0000000000000004002$

$1/0.0000000000000004002 \approx 2498750624687656 1,719$

Continuous Fractions

$$0.14127423822714681440\ldots \approx \cfrac{1}{7+\cfrac{1}{12+\cfrac{1}{1+\cfrac{1}{2+\cfrac{1}{1+\cfrac{1}{24987506246876561,719}}}}}} \Rightarrow$$

Continuous Fractions

$$0.14127423822714681440\ldots \approx \cfrac{1}{7+\cfrac{1}{12+\cfrac{1}{1+\cfrac{1}{2+\cfrac{1}{1+\cfrac{1}{24987506246876561,719}}}}}} \Rightarrow$$

$$X = \cfrac{1}{7+\cfrac{1}{12+\cfrac{1}{1+\cfrac{1}{2+\cfrac{1}{1}}}}} = \frac{51}{361}.$$

Continuous Fractions

$$0.14127423822714681440\ldots \approx \cfrac{1}{7+\cfrac{1}{12+\cfrac{1}{1+\cfrac{1}{2+\cfrac{1}{1+\cfrac{1}{24987506246876561,719}}}}}} \Rightarrow$$

$$X = \cfrac{1}{7+\cfrac{1}{12+\cfrac{1}{1+\cfrac{1}{2+\cfrac{1}{1}}}}} = \frac{51}{361}.$$

(Actually the period of $\dfrac{51}{361}$ is 342.)

Quadratic Irrationalities

Now suppose we don't know if X is rational. But we do know that it is a *quadratic irrationality*. (That is a root of an equation $ax^2 + bx + c = 0$ with a, b, c rational.)

Quadratic Irrationalities

Now suppose we don't know if X is rational. But we do know that it is a *quadratic irrationality*. (That is a root of an equation $ax^2 + bx + c = 0$ with a, b, c rational.)

Theorem(Lagrange)

X is a quadratic irrationality $\Leftrightarrow$ its continuos fraction is periodic.

Quadratic Irrationalities

Example: $\sqrt{3}$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1)$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1) = 1 + \dfrac{2}{\sqrt{3} + 1}$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1) = 1 + \dfrac{2}{\sqrt{3} + 1} = 1 + \dfrac{1}{\dfrac{\sqrt{3} + 1}{2}}$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1) = 1 + \dfrac{2}{\sqrt{3}+1} = 1 + \dfrac{1}{\dfrac{\sqrt{3}+1}{2}}$

$$= 1 + \cfrac{1}{\dfrac{2 + (\sqrt{3} - 1)}{2}}$$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1) = 1 + \dfrac{2}{\sqrt{3}+1} = 1 + \dfrac{1}{\dfrac{\sqrt{3}+1}{2}}$

$$= 1 + \frac{1}{\dfrac{2 + (\sqrt{3} - 1)}{2}} = 1 + \frac{1}{1 + \dfrac{\sqrt{3}-1}{2}}$$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1) = 1 + \dfrac{2}{\sqrt{3}+1} = 1 + \dfrac{1}{\dfrac{\sqrt{3}+1}{2}}$

$$= 1 + \cfrac{1}{\cfrac{2 + (\sqrt{3} - 1)}{2}} = 1 + \cfrac{1}{1 + \cfrac{\sqrt{3}-1}{2}}$$

$$= 1 + \cfrac{1}{1 + \cfrac{2}{2(\sqrt{3}+1)}}$$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1) = 1 + \dfrac{2}{\sqrt{3}+1} = 1 + \dfrac{1}{\dfrac{\sqrt{3}+1}{2}}$

$$= 1 + \frac{1}{\dfrac{2 + (\sqrt{3} - 1)}{2}} = 1 + \frac{1}{1 + \dfrac{\sqrt{3}-1}{2}}$$

$$= 1 + \frac{1}{1 + \dfrac{2}{2(\sqrt{3}+1)}} = 1 + \frac{1}{1 + \dfrac{1}{\sqrt{3}+1}}$$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1) = 1 + \dfrac{2}{\sqrt{3}+1} = 1 + \dfrac{1}{\dfrac{\sqrt{3}+1}{2}}$

$$= 1 + \cfrac{1}{\cfrac{2 + (\sqrt{3}-1)}{2}} = 1 + \cfrac{1}{1 + \cfrac{\sqrt{3}-1}{2}}$$

$$= 1 + \cfrac{1}{1 + \cfrac{2}{2(\sqrt{3}+1)}} = 1 + \cfrac{1}{1 + \cfrac{1}{\sqrt{3}+1}}$$

$$= 1 + \cfrac{1}{1 + \cfrac{1}{2 + (\sqrt{3}-1)}}$$

Quadratic Irrationalities

Example: $\sqrt{3} = 1 + (\sqrt{3} - 1) = 1 + \dfrac{2}{\sqrt{3}+1} = 1 + \dfrac{1}{\dfrac{\sqrt{3}+1}{2}}$

$$= 1 + \cfrac{1}{\cfrac{2 + (\sqrt{3}-1)}{2}} = 1 + \cfrac{1}{1 + \cfrac{\sqrt{3}-1}{2}}$$

$$= 1 + \cfrac{1}{1 + \cfrac{2}{2(\sqrt{3}+1)}} = 1 + \cfrac{1}{1 + \cfrac{1}{\sqrt{3}+1}}$$

$$= 1 + \cfrac{1}{1 + \cfrac{1}{2 + (\sqrt{3}-1)}} = 1 + \cfrac{1}{1 + \cfrac{1}{2 + \cfrac{1}{1 + \cfrac{1}{2 + \dots}}}}.$$

Integer Relations

Let's generalize the problem.

Integer Relations

Let's generalize the problem.

Definition

α is an *algebraic number* if there exist $a_0, \ldots, a_n \in \mathbb{Z}$ such that $a_n\alpha^n + \ldots + a_1\alpha + a_0 = 0$ and $a_n \neq 0$. The *degree* of α is the smallest of such n.

Integer Relations

Let's generalize the problem.

Definition

α is an *algebraic number* if there exist $a_0, \ldots, a_n \in \mathbb{Z}$ such that $a_n\alpha^n + \ldots + a_1\alpha + a_0 = 0$ and $a_n \neq 0$. The *degree* of α is the smallest of such n.

Remark

α is algebraic of degree $\leq n \Leftrightarrow (1, \alpha, \alpha^2, \ldots, \alpha^n)$ posess an *integer relation* [see below].

Integer Relations

Let's generalize the problem.

Definition

α is an *algebraic number* if there exist $a_0, \ldots, a_n \in \mathbb{Z}$ such that $a_n\alpha^n + \ldots + a_1\alpha + a_0 = 0$ and $a_n \neq 0$. The *degree* of α is the smallest of such n.

Remark

α is algebraic of degree $\leq n \Leftrightarrow (1, \alpha, \alpha^2, \ldots, \alpha^n)$ posess an *integer relation* [see below].

Definition

An *integer relation* for n-tuple $(x_1, \ldots, x_n) \in \mathbb{R}^n$ is an n-tuple $0 \neq (a_1, \ldots, a_n) \in \mathbb{Z}^n$ such that $a_1x_1 + \ldots + a_nx_n = 0$.

Algorithms for Finding Integral Relations

The problem of finding an integer relation for two numbers (x_1, x_2) can be solved by applying the Euclidian algorithm to x_1, x_2, or, equivalently, by computing the continued fraction expansion of x_1/x_2.

Algorithms for Finding Integral Relations

The problem of finding an integer relation for two numbers (x_1, x_2) can be solved by applying the Euclidian algorithm to x_1, x_2, or, equivalently, by computing the continued fraction expansion of x_1/x_2.

The generalization for $n \geq 3$ was attempted by Euler, Jacobi, Minkowski, Perron, Bernstein, among others.

Algorithms for Finding Integral Relations

The problem of finding an integer relation for two numbers (x_1, x_2) can be solved by applying the Euclidian algorithm to x_1, x_2, or, equivalently, by computing the continued fraction expansion of x_1/x_2.

The generalization for $n \geq 3$ was attempted by Euler, Jacobi, Minkowski, Perron, Bernstein, among others.

The best known and most used algorithms at the present time are either algorithms based on lattice basis reduction algorithm by Lenstra, Lenstra, Jr. and Lovász (**LLL**) or **PSLQ algorithm** based on ideas of Ferguson, Forcade and Bergman. (Both discovered in 1970-s –1980-s.)

Outline

Some Reminders from Linear Algebra

- Let $\mathbb{R}^n$ be the n -dimensional real vector space ($n > 1$) with inner product: $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{j=1}^{n} x_j y_j$.

Some Reminders from Linear Algebra

- Let $\mathbb{R}^n$ be the n -dimensional real vector space ($n > 1$) with inner product: $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{j=1}^{n} x_j y_j$.
- $\|\mathbf{y}\| = \sqrt{\langle \mathbf{y}, \mathbf{y} \rangle}$ is the length of $\mathbf{y} \in \mathbb{R}^n$.

Some Reminders from Linear Algebra

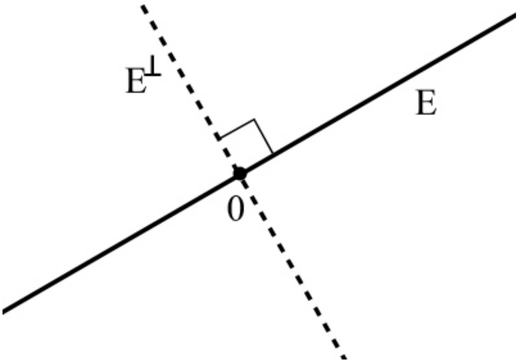
- Let $\mathbb{R}^n$ be the n -dimensional real vector space ($n > 1$) with inner product: $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{j=1}^{n} x_j y_j$.
- $\|\mathbf{y}\| = \sqrt{\langle \mathbf{y}, \mathbf{y} \rangle}$ is the length of $\mathbf{y} \in \mathbb{R}^n$.
- $\mathbf{x}$ and $\mathbf{y}$ are *orthogonal* $\Leftrightarrow \langle \mathbf{x}, \mathbf{y} \rangle = 0$.

Some Reminders from Linear Algebra

- Let $\mathbb{R}^n$ be the n -dimensional real vector space ($n > 1$) with inner product: $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{j=1}^{n} x_j y_j$.
- $\|\mathbf{y}\| = \sqrt{\langle \mathbf{y}, \mathbf{y} \rangle}$ is the length of $\mathbf{y} \in \mathbb{R}^n$.
- $\mathbf{x}$ and $\mathbf{y}$ are *orthogonal* $\Leftrightarrow \langle \mathbf{x}, \mathbf{y} \rangle = 0$.
- For a linear subspace $E \subset \mathbb{R}^n$ we denote by $E^\perp \subset \mathbb{R}^n$ the *orthogonal complement* of E (i.e., the subspace consisting of all vectors that are orthogonal to E).

Some Reminders from Linear Algebra

E and $E^{\perp}$:



Some Reminders from Linear Algebra

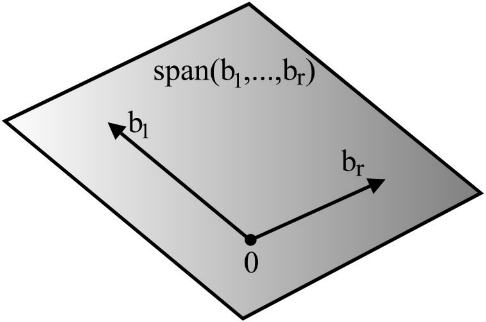
- The transpose of matrix A is A^T.

Some Reminders from Linear Algebra

- The transpose of matrix A is A^T.
- If $\mathbf{b}_1, \ldots, \mathbf{b}_r \in \mathbb{R}^n$ then $[\mathbf{b}_1, \ldots, \mathbf{b}_r]$ will denote $n \times r$ matrix which has $\mathbf{b}_1, \ldots, \mathbf{b}_r \in \mathbb{R}^n$ as columns.

Some Reminders from Linear Algebra

- The transpose of matrix A is A^T.
- If $\mathbf{b}_1, \ldots, \mathbf{b}_r \in \mathbb{R}^n$ then $[\mathbf{b}_1, \ldots, \mathbf{b}_r]$ will denote $n \times r$ matrix which has $\mathbf{b}_1, \ldots, \mathbf{b}_r \in \mathbb{R}^n$ as columns.
- $\mathrm{span}(\mathbf{b}_1, \ldots, \mathbf{b}_r)$ is the linear space, spanned on $\mathbf{b}_1, \ldots, \mathbf{b}_r$:
$$\mathrm{span}(\mathbf{b}_1, \ldots, \mathbf{b}_r) = \left\{ \sum_{j=1}^{j=r} a_j \mathbf{b}_j \,\middle|\, a_j \in \mathbb{R} \right\}.$$



Gram-Schmidt orthogonalization

With $\mathbf{b}_0 = \mathbf{x},\ \mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{R}^n$ we associate the orthogonal system $\mathbf{b}_0^*, \ldots, \mathbf{b}_n^*$ that are defined inductively:

Gram-Schmidt orthogonalization

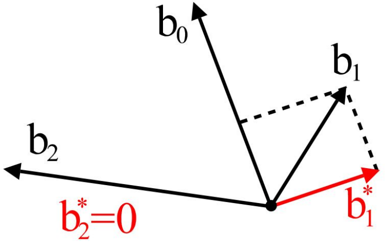
With $\mathbf{b}_0 = \mathbf{x},\ \mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{R}^n$ we associate the orthogonal system $\mathbf{b}_0^*, \ldots, \mathbf{b}_n^*$ that are defined inductively:

- $\mathbf{b}_0^* = \mathbf{x}$,
- $\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=0}^{i-1} \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \mathbf{b}_j^*,\ i = 1, \ldots, n.$

Gram-Schmidt orthogonalization

With $\mathbf{b}_0 = \mathbf{x}$, $\mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{R}^n$ we associate the orthogonal system $\mathbf{b}_0^*, \ldots, \mathbf{b}_n^*$ that are defined inductively:

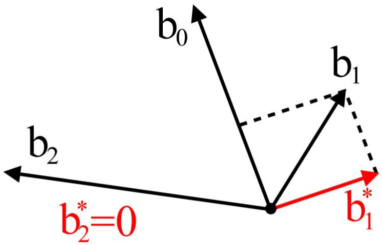
- $\mathbf{b}_0^* = \mathbf{x}$,
- $\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=0}^{i-1} \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \mathbf{b}_j^*$, $i = 1, \ldots, n$.



Gram-Schmidt orthogonalization

With $\mathbf{b}_0 = \mathbf{x},\ \mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{R}^n$ we associate the orthogonal system $\mathbf{b}_0^*, \ldots, \mathbf{b}_n^*$ that are defined inductively:

- $\mathbf{b}_0^* = \mathbf{x}$,
- $\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=0}^{i-1} \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \mathbf{b}_j^*,\ i = 1, \ldots, n.$



Note: $\mathbf{b}_i^*$ is orthogonal to $\mathrm{span}(\mathbf{b}_0^*, \ldots, \mathbf{b}_{i-1}^*) = \mathrm{span}(\mathbf{b}_0, \ldots, \mathbf{b}_{i-1})$.

Lattices

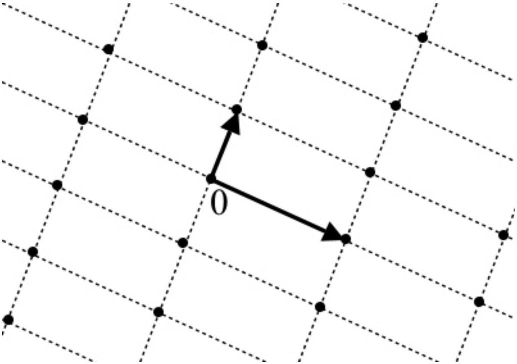
Definition

A *lattice* $L \subset \mathbb{R}^n$ is an additive closure of some linear independent $\mathbf{b}_1, \ldots, \mathbf{b}_r \in \mathbb{R}^n$, i.e. $L = \left\{ \sum_{i=1}^{r} m_i \mathbf{b}_i \,|\, m_i \in \mathbb{Z} \right\}$.

Lattices

Definition

A *lattice* $L \subset \mathbb{R}^n$ is an additive closure of some linear independent $\mathbf{b}_1, \ldots, \mathbf{b}_r \in \mathbb{R}^n$, i.e. $L = \left\{ \sum_{i=1}^{r} m_i \mathbf{b}_i \,|\, m_i \in \mathbb{Z} \right\}$.

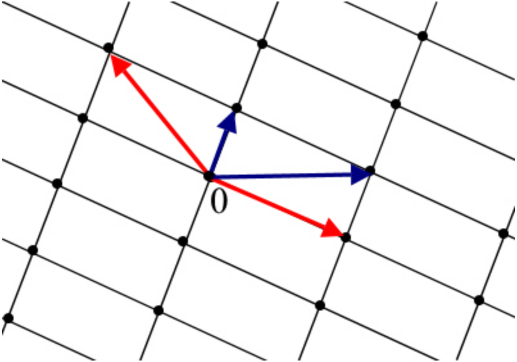


Lattices

Such $\mathbf{b}_1, \ldots, \mathbf{b}_r$ are called the *basis* of L. Of course they are not defined uniquely.

Lattices

Such $\mathbf{b}_1, \ldots, \mathbf{b}_r$ are called the *basis* of L. Of course they are not defined uniquely.

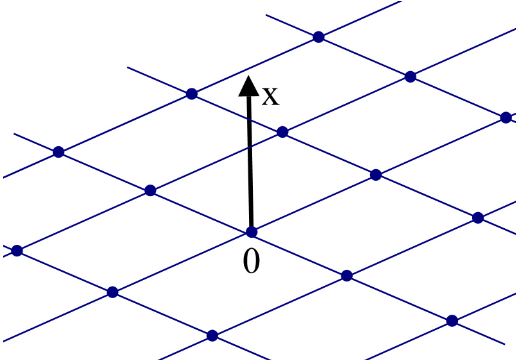


Lattices

An important example: the lattice $L_{\mathbf{x}} \subset \mathbb{Z}^n$ of all integer relations for $\mathbf{x}$ together with $\mathbf{0}$: $L_{\mathbf{x}} := \{\mathbf{m} \in \mathbb{Z}^n \,|\, \langle \mathbf{x}, \mathbf{m} \rangle = 0\}$.

Lattices

An important example: the lattice $L_{\mathbf{x}} \subset \mathbb{Z}^n$ of all integer relations for $\mathbf{x}$ together with $\mathbf{0}$: $L_{\mathbf{x}} := \{\mathbf{m} \in \mathbb{Z}^n \,|\, \langle \mathbf{x}, \mathbf{m} \rangle = 0\}$.



Bases

We will perform two types of elementary basis exchange operations on a current basis $\mathbf{b}_1, \ldots, \mathbf{b}_n$ of a given lattice:

Bases

We will perform two types of elementary basis exchange operations on a current basis $\mathbf{b}_1, \ldots, \mathbf{b}_n$ of a given lattice:

- *Exchange steps*: Interchange $\mathbf{b}_i$ and $\mathbf{b}_{i+1}$ for some i.

Bases

We will perform two types of elementary basis exchange operations on a current basis $\mathbf{b}_1, \ldots, \mathbf{b}_n$ of a given lattice:

- *Exchange steps*: Interchange $\mathbf{b}_i$ and $\mathbf{b}_{i+1}$ for some i.
- *Size-reduction steps*: Replace $\mathbf{b}_i$ with $\mathbf{b}_i - p\mathbf{b}_j$ where $p \in \mathbb{Z}$ for some $1 \leq j < i$.

Bases

We will perform two types of elementary basis exchange operations on a current basis $\mathbf{b}_1, \ldots, \mathbf{b}_n$ of a given lattice:

- *Exchange steps*: Interchange $\mathbf{b}_i$ and $\mathbf{b}_{i+1}$ for some i.
- *Size-reduction steps*: Replace $\mathbf{b}_i$ with $\mathbf{b}_i - p\mathbf{b}_j$ where $p \in \mathbb{Z}$ for some $1 \leq j < i$.

With every basis $\mathbf{b}_1, \ldots, \mathbf{b}_n$ there is the *dual basis* $\mathbf{c}_1, \ldots, \mathbf{c}_n$:

$$[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1} \Leftrightarrow [\mathbf{c}_1, \ldots, \mathbf{c}_n]^T [\mathbf{b}_1, \ldots, \mathbf{b}_n] = \mathrm{Id}$$

$$\Leftrightarrow \langle \mathbf{b}_j, \mathbf{c}_k \rangle = \delta_{jk} = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases} .$$

Bases

We will perform two types of elementary basis exchange operations on a current basis $\mathbf{b}_1, \ldots, \mathbf{b}_n$ of a given lattice:

- *Exchange steps*: Interchange $\mathbf{b}_i$ and $\mathbf{b}_{i+1}$ for some i.
- *Size-reduction steps*: Replace $\mathbf{b}_i$ with $\mathbf{b}_i - p\mathbf{b}_j$ where $p \in \mathbb{Z}$ for some $1 \leq j < i$.

With every basis $\mathbf{b}_1, \ldots, \mathbf{b}_n$ there is the *dual basis* $\mathbf{c}_1, \ldots, \mathbf{c}_n$:
$[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1} \Leftrightarrow [\mathbf{c}_1, \ldots, \mathbf{c}_n]^T [\mathbf{b}_1, \ldots, \mathbf{b}_n] = \mathrm{Id}$
$\Leftrightarrow \langle \mathbf{b}_j, \mathbf{c}_k \rangle = \delta_{jk} = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases}.$

Note: $\mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{Z}^n$ and
$B = [\mathbf{b}_1, \ldots, \mathbf{b}_n]$ unimodular ($\det B = \pm 1$) $\Rightarrow \mathbf{c}_1, \ldots, \mathbf{c}_n \in \mathbb{Z}^n$.

HJLS: Source; Model of Computation; Notation

Source: J. Hastad, B. Just, J. C. Lagarias, and C. P. Schnorr. Polynomial Time Algorithms for Finding Integer Relations among Real Numbers.
SIAM J. Comput., Vol.18, 1989, pp.859-881.

HJLS: Source; Model of Computation; Notation

Source: J. Hastad, B. Just, J. C. Lagarias, and C. P. Schnorr.
Polynomial Time Algorithms for Finding Integer Relations among Real Numbers.
SIAM J. Comput., Vol.18, 1989, pp.859-881.

Model of Computation:

- Computation with real numbers.
- Operations: addition, subtraction, multiplication, division, comparison, the nearest integer($\lceil\rfloor$) — at unit cost.

HJLS: Source; Model of Computation; Notation

Source: J. Hastad, B. Just, J. C. Lagarias, and C. P. Schnorr. Polynomial Time Algorithms for Finding Integer Relations among Real Numbers.
SIAM J. Comput., Vol.18, 1989, pp.859-881.

Model of Computation:

- Computation with real numbers.
- Operations: addition, subtraction, multiplication, division, comparison, the nearest integer($\lceil \rfloor$) — at unit cost.

Notation:

- $\dfrac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} =: \mu_{ij}.$

HJLS: Source; Model of Computation; Notation

Source: J. Hastad, B. Just, J. C. Lagarias, and C. P. Schnorr.
Polynomial Time Algorithms for Finding Integer Relations among Real Numbers.
SIAM J. Comput., Vol.18, 1989, pp.859-881.

Model of Computation:

- Computation with real numbers.
- Operations: addition, subtraction, multiplication, division, comparison, the nearest integer($\lceil \rfloor$) — at unit cost.

Notation:

- $\dfrac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} =: \mu_{ij}$.
- $\lambda(\mathbf{x}) :=$ the length of the shortest integer relation for $\mathbf{x}$. If there are no relations then $\lambda(\mathbf{x}) := \infty$.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n$, $k \in \mathbb{N}$.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n, k \in \mathbb{N}$.
(1) Initiation: $\mathbf{b}_0 := \mathbf{x}$; $\mathbf{b}_1, \ldots, \mathbf{b}_n :=$ standard basis of $\mathbb{Z}^n$.
Compute μ_{ij} and $\|\mathbf{b}_i^*\|^2 = \langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle$.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n, k \in \mathbb{N}$.
(1) Initiation: $\mathbf{b}_0 := \mathbf{x}$; $\mathbf{b}_1, \ldots, \mathbf{b}_n :=$ standard basis of $\mathbb{Z}^n$.
Compute μ_{ij} and $\|\mathbf{b}_i^*\|^2 = \langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle$.
(2) Termination test:
If $\|\mathbf{b}_n^*\| \neq 0$ then an integer relation is found.
Compute $[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1}$ and output the integer relation $\mathbf{c}_n$. Stop.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n, k \in \mathbb{N}$.
(1) Initiation: $\mathbf{b}_0 := \mathbf{x}$; $\mathbf{b}_1, \ldots, \mathbf{b}_n :=$ standard basis of $\mathbb{Z}^n$.
Compute μ_{ij} and $\|\mathbf{b}_i^*\|^2 = \langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle$.
(2) Termination test:
If $\|\mathbf{b}_n^*\| \neq 0$ then an integer relation is found.
Compute $[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1}$ and output the integer relation $\mathbf{c}_n$. Stop.
If $\left\|\mathbf{b}_j^*\right\| \leq 1/2^k,\ 1 \leq j \leq n$ then no small integer relation exist.
Output "$\lambda(\mathbf{x}) \geq 2^k$" and stop.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n, k \in \mathbb{N}$.
(1) Initiation: $\mathbf{b}_0 := \mathbf{x}$; $\mathbf{b}_1, \ldots, \mathbf{b}_n :=$ standard basis of $\mathbb{Z}^n$.
Compute μ_{ij} and $\|\mathbf{b}_\imath^*\|^2 = \langle \mathbf{b}_\imath^*, \mathbf{b}_\imath^* \rangle$.
(2) Termination test:
If $\|\mathbf{b}_n^*\| \neq 0$ then an integer relation is found.
Compute $[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1}$ and output the integer relation $\mathbf{c}_n$. Stop.
If $\left\|\mathbf{b}_j^*\right\| \leq 1/2^k,\ 1 \leq j \leq n$ then no small integer relation exist.
Output "$\lambda(\mathbf{x}) \geq 2^k$" and stop.
(3) Exchange step:
Choose from $1 \leq \imath \leq n$ that $\imath$ that maximizes $2^\imath \|\mathbf{b}_\imath^*\|$.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n, k \in \mathbb{N}$.
(1) <u>Initiation</u>: $\mathbf{b}_0 := \mathbf{x}$; $\mathbf{b}_1, \ldots, \mathbf{b}_n :=$ standard basis of $\mathbb{Z}^n$.
Compute μ_{ij} and $\|\mathbf{b}_i^*\|^2 = \langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle$.
(2) <u>Termination test</u>:
If $\|\mathbf{b}_n^*\| \neq 0$ then an integer relation is found.
Compute $[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1}$ and output the integer relation $\mathbf{c}_n$. Stop.
If $\left\|\mathbf{b}_j^*\right\| \leq 1/2^k,\ 1 \leq j \leq n$ then no small integer relation exist.
Output "$\lambda(\mathbf{x}) \geq 2^k$" and stop.
(3) <u>Exchange step</u>:
Choose from $1 \leq i \leq n$ that i that maximizes $2^i \|\mathbf{b}_i^*\|$.
Size-reduce $\mathbf{b}_{i+1}$: $\mathbf{b}_{i+1} := \mathbf{b}_{i+1} - \lceil \mu_{i+1,i} \rfloor \mathbf{b}_i$.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n, k \in \mathbb{N}$.
(1) Initiation: $\mathbf{b}_0 := \mathbf{x}$; $\mathbf{b}_1, \ldots, \mathbf{b}_n :=$ standard basis of $\mathbb{Z}^n$.
Compute μ_{ij} and $\|\mathbf{b}_i^*\|^2 = \langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle$.
(2) Termination test:
If $\|\mathbf{b}_n^*\| \neq 0$ then an integer relation is found.
Compute $[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1}$ and output the integer relation $\mathbf{c}_n$. Stop.
If $\left\|\mathbf{b}_j^*\right\| \leq 1/2^k, \ 1 \leq j \leq n$ then no small integer relation exist.
Output "$\lambda(\mathbf{x}) \geq 2^k$" and stop.
(3) Exchange step:
Choose from $1 \leq i \leq n$ that i that maximizes $2^i \|\mathbf{b}_i^*\|$.
Size-reduce $\mathbf{b}_{i+1}$: $\mathbf{b}_{i+1} := \mathbf{b}_{i+1} - \lceil \mu_{i+1,i} \rfloor \mathbf{b}_i$.
Update $\mu_{i+1,j}$ for $j = 1, \ldots, i$.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n, k \in \mathbb{N}$.
(1) <u>Initiation</u>: $\mathbf{b}_0 := \mathbf{x}$; $\mathbf{b}_1, \ldots, \mathbf{b}_n :=$ standard basis of $\mathbb{Z}^n$.
Compute μ_{ij} and $\|\mathbf{b}_i^*\|^2 = \langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle$.
(2) <u>Termination test</u>:
If $\|\mathbf{b}_n^*\| \neq 0$ then an integer relation is found.
Compute $[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1}$ and output the integer relation $\mathbf{c}_n$. Stop.
If $\left\|\mathbf{b}_j^*\right\| \leq 1/2^k, \ 1 \leq j \leq n$ then no small integer relation exist.
Output "$\lambda(\mathbf{x}) \geq 2^k$" and stop.
(3) <u>Exchange step</u>:
Choose from $1 \leq i \leq n$ that i that maximizes $2^i \|\mathbf{b}_i^*\|$.
Size-reduce $\mathbf{b}_{i+1}$: $\mathbf{b}_{i+1} := \mathbf{b}_{i+1} - \lceil \mu_{i+1,i} \rfloor \mathbf{b}_i$.
Update $\mu_{i+1,j}$ for $j = 1, \ldots, i$.
Exchange $\mathbf{b}_i$ and $\mathbf{b}_{i+1}$.

HJLS: the Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n, k \in \mathbb{N}$.
(1) Initiation: $\mathbf{b}_0 := \mathbf{x}$; $\mathbf{b}_1, \ldots, \mathbf{b}_n :=$ standard basis of $\mathbb{Z}^n$.
Compute μ_{ij} and $\|\mathbf{b}_i^*\|^2 = \langle \mathbf{b}_i^*, \mathbf{b}_i^* \rangle$.
(2) Termination test:
If $\|\mathbf{b}_n^*\| \neq 0$ then an integer relation is found.
Compute $[\mathbf{c}_1, \ldots, \mathbf{c}_n]^T = [\mathbf{b}_1, \ldots, \mathbf{b}_n]^{-1}$ and output the integer relation $\mathbf{c}_n$. Stop.
If $\|\mathbf{b}_j^*\| \leq 1/2^k, \; 1 \leq j \leq n$ then no small integer relation exist.
Output "$\lambda(\mathbf{x}) \geq 2^k$" and stop.
(3) Exchange step:
Choose from $1 \leq i \leq n$ that i that maximizes $2^i \|\mathbf{b}_i^*\|$.
Size-reduce $\mathbf{b}_{i+1}$: $\mathbf{b}_{i+1} := \mathbf{b}_{i+1} - \lceil \mu_{i+1,i} \rfloor \mathbf{b}_i$.
Update $\mu_{i+1,j}$ for $j = 1, \ldots, i$.
Exchange $\mathbf{b}_i$ and $\mathbf{b}_{i+1}$.
Update $\|\mathbf{b}_\nu^*\|^2, \mu_{\nu j}, \; \mu_{j\nu}$ for $\nu = i, i+1, \; 1 \leq j \leq n$. Go to *(2)*.

HJLS: the Algorithm

Note: The matrix $[\mathbf{c}_1, \ldots, \mathbf{c}_n]$ can be computed incrementally:

- Initially $[\mathbf{c}_1, \ldots, \mathbf{c}_n] = \mathrm{Id}_n$.
- $\mathbf{b}_{i+1} := \mathbf{b}_{i+1} - \lceil \mu_{i+1,i} \rfloor \mathbf{b}_i \Rightarrow \mathbf{c}_i := \mathbf{c}_i + \lceil \mu_{i+1,i} \rfloor \mathbf{c}_{i+1}$.
- $\mathbf{b}_i \leftrightarrow \mathbf{b}_{i+1} \Rightarrow \mathbf{c}_i \leftrightarrow \mathbf{c}_{i+1}$.

HJLS: Correctedness and Polynomial Time

Theorem

- *The output $\boldsymbol{c}_n$ is an integer relation for $\boldsymbol{x}$.*

HJLS: Correctedness and Polynomial Time

Theorem

- *The output* $\boldsymbol{c}_n$ *is an integer relation for* $\boldsymbol{x}$.
- *For every basis* $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ *of* $\mathbb{Z}^n$ $\lambda(\boldsymbol{x}) \geq \frac{1}{\max \|\boldsymbol{b}_j^*\|}$. *So the algorithm claims "*$\lambda(\boldsymbol{x}) \geq 2^k$*" correctly.*

HJLS: Correctedness and Polynomial Time

Theorem

- *The output* $\boldsymbol{c}_n$ *is an integer relation for* $\boldsymbol{x}$.
- *For every basis* $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ *of* $\mathbb{Z}^n$ $\lambda(\boldsymbol{x}) \geq \frac{1}{\max\|\boldsymbol{b}_j^*\|}$. *So the algorithm claims "*$\lambda(\boldsymbol{x}) \geq 2^k$*" correctly.*
- *The output* $\boldsymbol{c}_n$ *satisfies* $\|\boldsymbol{c}_n\|^2 \leq 2^{n-2} \min\left\{\lambda(\boldsymbol{x})^2, 2^{2k}\right\}$.

HJLS: Correctedness and Polynomial Time

Theorem

- *The output* $\boldsymbol{c}_n$ *is an integer relation for* $\boldsymbol{x}$.
- *For every basis* $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ *of* $\mathbb{Z}^n$ $\lambda(\boldsymbol{x}) \geq \frac{1}{\max \|\boldsymbol{b}_j^*\|}$. *So the algorithm claims "*$\lambda(\boldsymbol{x}) \geq 2^k$*" correctly.*
- *The output* $\boldsymbol{c}_n$ *satisfies* $\|\boldsymbol{c}_n\|^2 \leq 2^{n-2} \min\left\{\lambda(\boldsymbol{x})^2, 2^{2k}\right\}$.
- *The algorithm halts after at most* $O(n^3(k+n))$ *arithmetic steps on real numbers.*

HJLS: a Partial Proof

Proof.

- $\mathbf{b}_n^* \neq \mathbf{0} \Rightarrow \exists i$ s.t. $\mathbf{b}_i^* = \mathbf{0}$.

HJLS: a Partial Proof

Proof.

- $\mathbf{b}_n^* \neq \mathbf{0} \Rightarrow \exists i$ s.t. $\mathbf{b}_i^* = \mathbf{0}$.

  Then $\mathbf{0} = \mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=0}^{i-1} \frac{\left\langle \mathbf{b}_i, \mathbf{b}_j^* \right\rangle}{\left\langle \mathbf{b}_j^*, \mathbf{b}_j^* \right\rangle} \mathbf{b}_j^*$.

HJLS: a Partial Proof

Proof.

- $\mathbf{b}_n^* \neq \mathbf{0} \Rightarrow \exists i$ s.t. $\mathbf{b}_i^* = \mathbf{0}$.

  Then $\mathbf{0} = \mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=0}^{i-1} \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \mathbf{b}_j^*$.

  Then $\exists a_j$ s.t. $\sum_{j=0}^{j=i} a_j \mathbf{b}_j = \mathbf{0}$.

HJLS: a Partial Proof

Proof.

- $\mathbf{b}_n^* \neq \mathbf{0} \Rightarrow \exists i$ s.t. $\mathbf{b}_i^* = \mathbf{0}$.

  Then $\mathbf{0} = \mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=0}^{i-1} \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \mathbf{b}_j^*$.

  Then $\exists a_j$ s.t. $\sum_{j=0}^{j=i} a_j \mathbf{b}_j = \mathbf{0}$.

  $\mathbf{b}_1, \ldots, \mathbf{b}_i$ are linearly independent $\Rightarrow a_0 \neq 0$

  $\Rightarrow \mathbf{x} = \mathbf{b}_0 = \sum_{j=1}^{j=i} \frac{a_j}{a_0} \mathbf{b}_j$.

HJLS: a Partial Proof

Proof.

- $\mathbf{b}_n^* \neq \mathbf{0} \Rightarrow \exists i$ s.t. $\mathbf{b}_i^* = \mathbf{0}$.

  Then $\mathbf{0} = \mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=0}^{i-1} \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\langle \mathbf{b}_j^*, \mathbf{b}_j^* \rangle} \mathbf{b}_j^*$.

  Then $\exists a_j$ s.t. $\sum_{j=0}^{j=i} a_j \mathbf{b}_j = \mathbf{0}$.

  $\mathbf{b}_1, \ldots, \mathbf{b}_i$ are linearly independent $\Rightarrow a_0 \neq 0$

  $\Rightarrow \mathbf{x} = \mathbf{b}_0 = \sum_{j=1}^{j=i} \frac{a_j}{a_0} \mathbf{b}_j$.

  Since $\langle \mathbf{b}_j, \mathbf{c}_k \rangle = 0 \ \forall k > j$ we have $\langle \mathbf{x}, \mathbf{c}_k \rangle = 0 \ \forall k > i$, in particular $\langle \mathbf{x}, \mathbf{c}_n \rangle = 0$. $\square$

HJLS: a Partial Proof

Proof.

- Let $\mathbf{m}$ be any integer relation for $\mathbf{x}$.

HJLS: a Partial Proof

Proof.

- Let $\mathbf{m}$ be any integer relation for $\mathbf{x}$.

  Since $\mathbf{m} \in (\mathbf{x}\mathbb{R})^\perp = \mathrm{span}(\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*)$

HJLS: a Partial Proof

Proof.

- Let **m** be any integer relation for **x**.

  Since $\mathbf{m} \in (\mathbf{x}\mathbb{R})^{\perp} = \mathrm{span}(\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*)$
  there exists i s.t. $\langle \mathbf{m}, \mathbf{b}_i^* \rangle \neq 0$.

HJLS: a Partial Proof

Proof.

- Let **m** be any integer relation for **x**.

  Since $\mathbf{m} \in (\mathbf{x}\mathbb{R})^{\perp} = \mathrm{span}(\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*)$
  there exists i s.t. $\langle \mathbf{m}, \mathbf{b}_i^* \rangle \neq 0$.
  For the smallest such i we have

  $$\langle \mathbf{m}, \mathbf{b}_i^* \rangle = \left\langle \mathbf{m}, \mathbf{b}_i - \sum_{j=0}^{i-1} \mu_{ij} \mathbf{b}_j^* \right\rangle = \langle \mathbf{m}, \mathbf{b}_i \rangle - \sum_{j=0}^{i-1} \mu_{ij} \left\langle \mathbf{m}, \mathbf{b}_j^* \right\rangle =$$

HJLS: a Partial Proof

Proof.

- Let **m** be any integer relation for **x**.

  Since $\mathbf{m} \in (\mathbf{x}\mathbb{R})^{\perp} = \mathrm{span}(\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*)$
  there exists i s.t. $\langle \mathbf{m}, \mathbf{b}_i^* \rangle \neq 0$.
  For the smallest such i we have

  $$\langle \mathbf{m}, \mathbf{b}_i^* \rangle = \left\langle \mathbf{m}, \mathbf{b}_i - \sum_{j=0}^{i-1} \mu_{ij} \mathbf{b}_j^* \right\rangle = \langle \mathbf{m}, \mathbf{b}_i \rangle - \sum_{j=0}^{i-1} \mu_{ij} \left\langle \mathbf{m}, \mathbf{b}_j^* \right\rangle =$$

  $= \langle \mathbf{m}, \mathbf{b}_i \rangle \in \mathbb{Z}$, and hence $|\langle \mathbf{m}, \mathbf{b}_i^* \rangle| \geq 1$.

HJLS: a Partial Proof

Proof.

- Let $\mathbf{m}$ be any integer relation for $\mathbf{x}$.

  Since $\mathbf{m} \in (\mathbf{x}\mathbb{R})^{\perp} = \operatorname{span}(\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*)$
  there exists i s.t. $\langle \mathbf{m}, \mathbf{b}_i^* \rangle \neq 0$.
  For the smallest such i we have

  $$\langle \mathbf{m}, \mathbf{b}_i^* \rangle = \left\langle \mathbf{m}, \mathbf{b}_i - \sum_{j=0}^{i-1} \mu_{ij} \mathbf{b}_j^* \right\rangle = \langle \mathbf{m}, \mathbf{b}_i \rangle - \sum_{j=0}^{i-1} \mu_{ij} \left\langle \mathbf{m}, \mathbf{b}_j^* \right\rangle =$$

  $= \langle \mathbf{m}, \mathbf{b}_i \rangle \in \mathbb{Z}$, and hence $|\langle \mathbf{m}, \mathbf{b}_i^* \rangle| \geq 1$.

  But $|\langle \mathbf{m}, \mathbf{b}_i^* \rangle| \leq \|\mathbf{m}\| \, \|\mathbf{b}_i^*\|$.

HJLS: a Partial Proof

Proof.

- Let $\mathbf{m}$ be any integer relation for $\mathbf{x}$.

  Since $\mathbf{m} \in (\mathbf{x}\mathbb{R})^{\perp} = \operatorname{span}(\mathbf{b}_1^*, \ldots, \mathbf{b}_n^*)$
  there exists i s.t. $\langle \mathbf{m}, \mathbf{b}_i^* \rangle \neq 0$.
  For the smallest such i we have

  $$\langle \mathbf{m}, \mathbf{b}_i^* \rangle = \left\langle \mathbf{m}, \mathbf{b}_i - \sum_{j=0}^{i-1} \mu_{ij} \mathbf{b}_j^* \right\rangle = \langle \mathbf{m}, \mathbf{b}_i \rangle - \sum_{j=0}^{i-1} \mu_{ij} \left\langle \mathbf{m}, \mathbf{b}_j^* \right\rangle =$$

  $= \langle \mathbf{m}, \mathbf{b}_i \rangle \in \mathbb{Z}$, and hence $|\langle \mathbf{m}, \mathbf{b}_i^* \rangle| \geq 1$.

  But $|\langle \mathbf{m}, \mathbf{b}_i^* \rangle| \leq \|\mathbf{m}\| \, \|\mathbf{b}_i^*\|$.

  So $\|\mathbf{m}\| \geq \dfrac{1}{\|\mathbf{b}_i^*\|}$. $\square$

Outline

PSLQ: Source; Model of Computation

The name "PSLQ" comes from partial sums of squares and LQ (lower-diagonal—orthogonal) matrix decomposition.

PSLQ: Source; Model of Computation

The name "PSLQ" comes from partial sums of squares and LQ (lower-diagonal—orthogonal) matrix decomposition.

Source:
H. R. P. Ferguson, D. H. Bailey, S. Arno.
Analysis of PSLQ, an Integer Finding Algorithm.
Mathematics of Computation, Vol.68, 1999, pp.351-369.

PSLQ: Source; Model of Computation

The name "PSLQ" comes from partial sums of squares and LQ (lower-diagonal—orthogonal) matrix decomposition.

Source:
H. R. P. Ferguson, D. H. Bailey, S. Arno.
Analysis of PSLQ, an Integer Finding Algorithm.
Mathematics of Computation, Vol.68, 1999, pp.351-369.

Model of Computation:

- Computation with real numbers.
- Operations: addition, subtraction, multiplication, division, comparison, the nearest integer($\lceil \rfloor$) — at unit cost.

Definitions

$\mathbf{x} = (x_1, \ldots, x_n), \|\mathbf{x}\| = 1, x_j \neq 0.$

Definitions

$\mathbf{x} = (x_1, \ldots, x_n)$, $\|\mathbf{x}\| = 1$, $x_j \neq 0$.

Definition

Let for $1 \leq j \leq n$ $\; s_j^2 := \sum_{k=j}^{k=n} x_k^2$.

Definitions

$\mathbf{x} = (x_1, \ldots, x_n)$, $\|\mathbf{x}\| = 1$, $x_j \neq 0$.

Definition

Let for $1 \leq j \leq n$ $\; s_j^2 := \sum_{k=j}^{k=n} x_k^2$.

Definition

Let $H_{\mathbf{x}} = (h_{i,j})$ be $n \times (n-1)$ lower-trapezoidal matrix defined by:

$$h_{i,j} := \begin{cases} 0 & 1 \leq i < j \leq n-1 \\ s_{i+1}/s_i & 1 \leq i = j \leq n-1 \\ -x_j^2/(s_j s_{j+1}) & 1 \leq j < i \leq n-1. \end{cases}$$

PSLQ: The Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n$; $\gamma \geq \sqrt{4/3}$.

PSLQ: The Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n; \gamma \geq \sqrt{4/3}$.

(1)Initiation: $\mathbf{s} := (s_1, \ldots, s_n);\ \mathbf{y} := \mathbf{x}/s_1;\ H := H_{\mathbf{x}};\ B := \mathrm{Id}_n$.

PSLQ: The Algorithm

Input: $\mathbf{x} \in \mathbb{R}^n; \gamma \geq \sqrt{4/3}$.
(1)<u>*Initiation:*</u> $\mathbf{s} := (s_1, \ldots, s_n)$; $\mathbf{y} := \mathbf{x}/s_1$; $H := H_{\mathbf{x}}$; $B := \mathrm{Id}_n$.
Reduce H:

for $i := 2$ **to** n
    for $j := i - 1$ **to** 1 **step** -1
        $t := \lceil h_{ij}/h_{jj} \rfloor$
        $y_j := y_j + t y_i$
        for $k := 1$ **to** j
            $h_{ik} := h_{ik} - t h_{jk}$
        endfor
        for $k := 1$ **to** n
            $b_{kj} := b_{kj} + t b_{ki}$
        endfor
    endfor
endfor

PSLQ: The Algorithm

(2)Exchange step:

Choose r that maximizes $\gamma^r \left| h_{rr} \right|$.

PSLQ: The Algorithm

(2)Exchange step:

Choose r that maximizes $\gamma^r |h_{rr}|$.

Exchange $y_r \leftrightarrow y_{r+1}$, corresponding rows of H and corresponding columns of B.

PSLQ: The Algorithm

(2)Exchange step:

Choose r that maximizes $\gamma^r\,|h_{rr}|$.

Exchange $y_r \leftrightarrow y_{r+1}$, corresponding rows of H and corresponding columns of B.

(3)Corner:

$\delta := \sqrt{h_{rr}^2 + h_{r,r+1}^2};\ \alpha := h_{rr}/\delta;\ \beta := h_{r,r+1}/\delta.$

if $r \leq n-2$ **then**

  for $i := r$ **to** n

    $h_0 := h_{ir};\ h_1 := h_{i,r+1};$

    $h_{ir} := \alpha h_0 + \beta h_1;\ h_{i,r+1} := -\beta h_0 + \alpha h_1$

  endfor

endif

PSLQ: The Algorithm

(2)Exchange step:
Choose r that maximizes $\gamma^r\,|h_{rr}|$.
Exchange $y_r \leftrightarrow y_{r+1}$, corresponding rows of H and corresponding columns of B.
(3)Corner:
$\delta := \sqrt{h_{rr}^2 + h_{r,r+1}^2};\ \alpha := h_{rr}/\delta;\ \beta := h_{r,r+1}/\delta.$
if $r \le n-2$ **then**
  for $i := r$ **to** n
    $h_0 := h_{ir};\ h_1 := h_{i,r+1};$
    $h_{ir} := \alpha h_0 + \beta h_1;\ h_{i,r+1} := -\beta h_0 + \alpha h_1$
  endfor
endif
(4)Reduce H.

PSLQ: The Algorithm

(2)Exchange step:
Choose r that maximizes $\gamma^r |h_{rr}|$.
Exchange $y_r \leftrightarrow y_{r+1}$, corresponding rows of H and corresponding columns of B.
(3)Corner:
$\delta := \sqrt{h_{rr}^2 + h_{r,r+1}^2}$; $\alpha := h_{rr}/\delta$; $\beta := h_{r,r+1}/\delta$.
if $r \leq n - 2$ **then**
  for $i := r$ **to** n
    $h_0 := h_{ir}$; $h_1 := h_{i,r+1}$;
    $h_{ir} := \alpha h_0 + \beta h_1$; $h_{i,r+1} := -\beta h_0 + \alpha h_1$
  endfor
endif
(4)Reduce H.
(5)Norm bound: Compute $M := 1/\max_{1 \leq j \leq n} h_{jj}$. Then $\lambda(\mathbf{x}) \geq M$.

PSLQ: The Algorithm

(2)Exchange step:
Choose r that maximizes $\gamma^r |h_{rr}|$.
Exchange $y_r \leftrightarrow y_{r+1}$, corresponding rows of H and corresponding columns of B.
(3)Corner:
$\delta := \sqrt{h_{rr}^2 + h_{r,r+1}^2};\ \alpha := h_{rr}/\delta;\ \beta := h_{r,r+1}/\delta.$
if $r \leq n-2$ **then**
  for $i := r$ **to** n
    $h_0 := h_{ir};\ h_1 := h_{i,r+1};$
    $h_{ir} := \alpha h_0 + \beta h_1;\ h_{i,r+1} := -\beta h_0 + \alpha h_1$
  endfor
endif
(4)Reduce H.
(5)Norm bound: Compute $M := 1/\max\limits_{1\leq j\leq n} h_{jj}$. Then $\lambda(\mathbf{x}) \geq M$.
(6)Termination: Goto *(2)* unless $y_j = 0$ for some $1 \leq j \leq n$ or $h_{ii} = 0$ for some $1 \leq i \leq n-1$.

PSLQ: Correctedness and Polynomial Time

Theorem

- *The integer relation* $\mathbf{m}$ *for* $\boldsymbol{x}$ *appears as one of the columns of* B.

PSLQ: Correctedness and Polynomial Time

Theorem

- *The integer relation* $\mathbf{m}$ *for* $\boldsymbol{x}$ *appears as one of the columns of* B.
- $\lambda(\boldsymbol{x}) \geq 1/\max_{1\leq j\leq n} h_{jj}$.

PSLQ: Correctedness and Polynomial Time

Theorem

- *The integer relation* $\mathbf{m}$ *for* $\boldsymbol{x}$ *appears as one of the columns of* B.
- $\lambda(\boldsymbol{x}) \geq 1 / \max\limits_{1 \leq j \leq n} h_{jj}$.
- $\|\mathbf{m}\| \leq \gamma^{n-2} \lambda(\boldsymbol{x})$.

PSLQ: Correctedness and Polynomial Time

Theorem

- *The integer relation* $\mathbf{m}$ *for* $\boldsymbol{x}$ *appears as one of the columns of* B.
- $\lambda(\boldsymbol{x}) \geq 1/\max_{1\leq j\leq n} h_{jj}$.
- $\|\mathbf{m}\| \leq \gamma^{n-2}\lambda(\boldsymbol{x})$.
- *The algorithm halts after at most* $O(n^4 + n^3 \log \lambda(\boldsymbol{x}))$ *arithmetic steps on real numbers.*

Outline

Usage

Note: Proving relations is a separate matter.

Usage

Note: Proving relations is a separate matter.

Precision:
As a rule of thumb if $\mathbf{x}$ has n entries and D is the maximal number of digits in the relation we hope to find then we should work with nD digits precision.

Usage

Note: Proving relations is a separate matter.

Precision:
As a rule of thumb if $\mathbf{x}$ has n entries and D is the maximal number of digits in the relation we hope to find then we should work with nD digits precision.

LLL or PSLQ?
LLL-based algorithms are available in almost any computer algebra system (Maple, Mathematica).
PSLQ implementation are less directly available.

Usage

Note: Proving relations is a separate matter.

Precision:
As a rule of thumb if $\mathbf{x}$ has n entries and D is the maximal number of digits in the relation we hope to find then we should work with nD digits precision.

LLL or PSLQ?
LLL-based algorithms are available in almost any computer algebra system (Maple, Mathematica).
PSLQ implementation are less directly available.

PSLQ is more stable, because it uses a stable matrix reduction procedure. Unfortunately, HJLS is not stable.

An Example

Consider $\mathbf{x} = (11, 27, 31)$.

An Example

Consider $\mathbf{x} = (11, 27, 31)$.

PSLQ with $\gamma = \sqrt{2}$ for successive iterations $N = 0, 1, 2, 3, 4$ yields the five matrices:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 3 & 8 & 1 \\ -3 & -7 & -1 \end{pmatrix}, \begin{pmatrix} -2 & 1 & 0 \\ 2 & 3 & 1 \\ -1 & -3 & -1 \end{pmatrix},$$

$$\begin{pmatrix} 3 & -2 & 0 \\ 1 & 2 & 1 \\ -2 & -1 & -1 \end{pmatrix}, \begin{pmatrix} \boxed{\begin{matrix} -1 \\ 5 \\ -4 \end{matrix}} & \boxed{\begin{matrix} -8 \\ 9 \\ -5 \end{matrix}} & \begin{matrix} -2 \\ 2 \\ -1 \end{matrix} \end{pmatrix}.$$

An Example

Consider $\mathbf{x} = (11, 27, 31)$.

PSLQ with $\gamma = \sqrt{2}$ for successive iterations $N = 0, 1, 2, 3, 4$ yields the five matrices:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 3 & 8 & 1 \\ -3 & -7 & -1 \end{pmatrix}, \begin{pmatrix} -2 & 1 & 0 \\ 2 & 3 & 1 \\ -1 & -3 & -1 \end{pmatrix},$$

$$\begin{pmatrix} 3 & -2 & 0 \\ 1 & 2 & 1 \\ -2 & -1 & -1 \end{pmatrix}, \begin{pmatrix} \boxed{\begin{matrix} -1 \\ 5 \\ -4 \end{matrix}} & \boxed{\begin{matrix} -8 \\ 9 \\ -5 \end{matrix}} & \begin{matrix} -2 \\ 2 \\ -1 \end{matrix} \end{pmatrix}.$$

It found 2 relations:
$-11 + 5 \cdot 27 - 4 \cdot 31 = -11 + 135 - 124 = 0;$
$-8 \cdot 11 + 9 \cdot 27 - 5 \cdot 31 = -88 + 243 - 155 = 0.$

An Example

HJLS for successive iterations $N = 0, 1, 2, 3, 4, 5, 6$ yields the seven matrices:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & -1 \end{pmatrix},$$

$$\begin{pmatrix} 1 & -2 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & -2 \\ 0 & 1 & 2 \\ 0 & -1 & -1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & -2 \\ 1 & 3 & 2 \\ -1 & -3 & -1 \end{pmatrix},$$

$$\left(\begin{array}{ccc} 0 & -2 & \boxed{\begin{array}{c} -1 \\ 5 \\ -4 \end{array}} \\ 1 & 2 & \\ -1 & -1 & \end{array}\right).$$

An Example

HJLS for successive iterations $N = 0, 1, 2, 3, 4, 5, 6$ yields the seven matrices:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & -1 \end{pmatrix},$$

$$\begin{pmatrix} 1 & -2 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & -2 \\ 0 & 1 & 2 \\ 0 & -1 & -1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & -2 \\ 1 & 3 & 2 \\ -1 & -3 & -1 \end{pmatrix},$$

$$\begin{pmatrix} 0 & -2 & \boxed{\begin{matrix} -1 \\ 5 \\ -4 \end{matrix}} \\ 1 & 2 & \\ -1 & -1 & \end{pmatrix}.$$

It found 1 relation.

Outline

"BBP" formula for π

"BBP" formula for π

Perhaps one of the best known applications of PSLQ is the 1995 discovery, by means of PSLQ computation, of the "BBP" (Bailey, Borwein, Plouffe) formula for π:

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right).$$

"BBP" formula for π

Perhaps one of the best known applications of PSLQ is the 1995 discovery, by means of PSLQ computation, of the "BBP" (Bailey, Borwein, Plouffe) formula for π:

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right).$$

This formula permits one to compute directly hexademical digits of π without computing previous ones.

"BBP" formula for π

Perhaps one of the best known applications of PSLQ is the 1995 discovery, by means of PSLQ computation, of the "BBP" (Bailey, Borwein, Plouffe) formula for π:

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right).$$

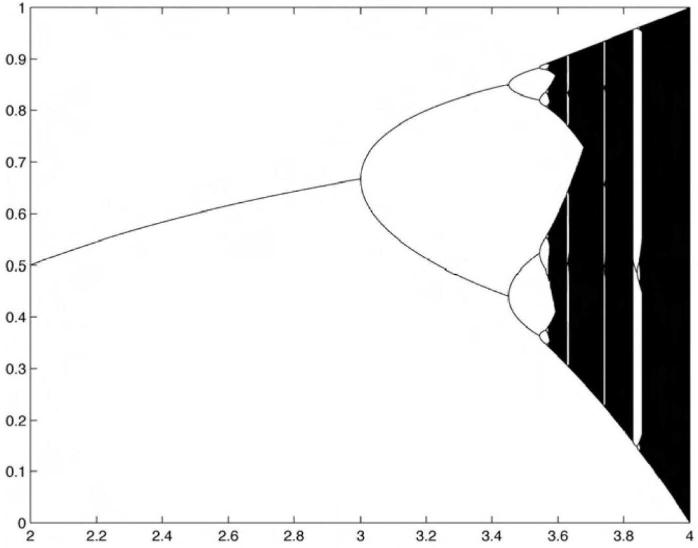
This formula permits one to compute directly hexademical digits of π without computing previous ones.

The formula was found by applying PSLQ to $(X_1, \ldots, X_n, \pi)$ where

$$X_j = \sum_{k=0}^{\infty} \frac{1}{16^k(8k+j)}.$$

Bifurcation Points in Chaos Theory

The chaotic iteration $x_{n+1} = rx_n(1 - x_n)$ ("logistic iteration"):



Bifurcation Points in Chaos Theory

$1 < r < B_1 = 3$: one limit point.
$B_1 < r < B_2 = 1 + \sqrt{6} = 3.449489\ldots$: two distinct limit points.
$B_2 < r < B_3$: four distinct limit points.
$B_3 < r < B_4$: eight distinct limit points.
And so on.

Bifurcation Points in Chaos Theory

$1 < r < B_1 = 3$: one limit point.
$B_1 < r < B_2 = 1 + \sqrt{6} = 3.449489\ldots$: two distinct limit points.
$B_2 < r < B_3$: four distinct limit points.
$B_3 < r < B_4$: eight distinct limit points.
And so on.

Using PSLQ with $n = 13$ we get that B_3 satisfies:
$r^{12} - 12r^{11} + 48r^{10} - 40r^9 - 193r^8 + 392r^7 + 44r^6 + 8r^5 - 977r^4 - 604r^3 + 2108r^2 + 4913 = 0.$

Bifurcation Points in Chaos Theory

The much more difficult problem for finding B_4 was studied in

D. H. Bailey and D. J. Broadhurst. Parallel integer relation detection: techniques and applications. Mathematics of Computation, Vol.70, 2000, pp.1719-1736.

Bifurcation Points in Chaos Theory

The much more difficult problem for finding B_4 was studied in

D. H. Bailey and D. J. Broadhurst. Parallel integer relation detection: techniques and applications. Mathematics of Computation, Vol.70, 2000, pp.1719-1736.

It was conjectured that B_4 might satisfy a 240-degree polynomial, and, in addition, $\alpha = -B_4(B_4 - 2)$ might satisfy a 120-degree polynomial.
Then an advanced PSLQ implementation was employed, and a relation with coefficients descending from 257^{30} to 1 was found.

Bifurcation Points in Chaos Theory

The much more difficult problem for finding B_4 was studied in

D. H. Bailey and D. J. Broadhurst. Parallel integer relation detection: techniques and applications. Mathematics of Computation, Vol.70, 2000, pp.1719-1736.

It was conjectured that B_4 might satisfy a 240-degree polynomial, and, in addition, $\alpha = -B_4(B_4 - 2)$ might satisfy a 120-degree polynomial.
Then an advanced PSLQ implementation was employed, and a relation with coefficients descending from 257^{30} to 1 was found.

4 year later the result was confirmed in large symbolic computation in

I. Kotsireas and K. Karamanos. Exact computation of the bifurcation point b4 of the logistic map and the Bailey-Broadhurst conjectures. Internat. J. Bifurcation and Chaos, Vol.14, 2004, pp.2417-2423.

Outline

Outline

Further Reading

A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász.
Factoring Polynomials with Rational Coefficients.
Math. Ann., Vol.261, 1982, pp.515-534.

Further Reading

- A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász.
  Factoring Polynomials with Rational Coefficients.
  Math. Ann., Vol.261, 1982, pp.515-534.
- J. Hastad, B. Just, J. C. Lagarias, and C. P. Schnorr.
  Polynomial Time Algorithms for Finding Integer Relations among Real Numbers.
  SIAM J. Comput., Vol.18, 1989, pp.859-881.

Further Reading

- A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász.
  Factoring Polynomials with Rational Coefficients.
  Math. Ann., Vol.261, 1982, pp.515-534.
- J. Hastad, B. Just, J. C. Lagarias, and C. P. Schnorr.
  Polynomial Time Algorithms for Finding Integer Relations among Real Numbers.
  SIAM J. Comput., Vol.18, 1989, pp.859-881.
- H. R. P. Ferguson, D. H. Bailey, S. Arno.
  Analysis of PSLQ, an Integer Finding Algorithm.
  Mathematics of Computation, Vol.68, 1999, pp.351-369.

Further Reading

A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász.
Factoring Polynomials with Rational Coefficients.
Math. Ann., Vol.261, 1982, pp.515-534.

J. Hastad, B. Just, J. C. Lagarias, and C. P. Schnorr.
Polynomial Time Algorithms for Finding Integer Relations among Real Numbers.
SIAM J. Comput., Vol.18, 1989, pp.859-881.

H. R. P. Ferguson, D. H. Bailey, S. Arno.
Analysis of PSLQ, an Integer Finding Algorithm.
Mathematics of Computation, Vol.68, 1999, pp.351-369.

D. H. Bailey, J. M. Borwein, N. J. Calkin, R. Girgensohn, D. R. Luke, and V. H. Moll. "Integer Relation Detection":
§2.2 in Experimental Mathematics in Action.
Natick, MA: A. K. Peters, pp. 29-31, 2006.