Lattice-based linear solver and number field computations

Paul Kirchner

April 9, 2025

Solving linear systems

Original problem

Given an invertible sparse $\mathbf{A}$ of dimension n over the Euclidean ring $\mathcal{R}$, solve $\mathbf{A}x = y$ over the fraction field. The coordinates of x are fractions with denominator $\det \mathbf{A}$.

Solving linear systems

Original problem

Given an invertible sparse $\mathbf{A}$ of dimension n over the Euclidean ring $\mathcal{R}$, solve $\mathbf{A}x = y$ over the fraction field. The coordinates of x are fractions with denominator $\det \mathbf{A}$.

New problem

Given an invertible sparse $\mathbf{A}$ over the Euclidean ring $\mathcal{R}$, solve over $\mathcal{R}$ the equation $\mathbf{A}x + \mathbf{B}y = z \in \mathcal{R}^n$. The random matrix $\mathbf{B}$ has $m \approx \sqrt{n}$ columns; coordinates of x, y have $\sqrt{n}$ bits.

Solving linear systems

Original problem

Given an invertible sparse $\mathbf{A}$ of dimension n over the Euclidean ring $\mathcal{R}$, solve $\mathbf{A}x = y$ over the fraction field. The coordinates of x are fractions with denominator $\det \mathbf{A}$.

New problem

Given an invertible sparse $\mathbf{A}$ over the Euclidean ring $\mathcal{R}$, solve over $\mathcal{R}$ the equation $\mathbf{A}x + \mathbf{B}y = z \in \mathcal{R}^n$. The random matrix $\mathbf{B}$ has $m \approx \sqrt{n}$ columns; coordinates of x, y have $\sqrt{n}$ bits.

Objective

Find the class group of $\mathbb{Q}[\sqrt{\Delta}]$ in time:

$$\exp\left((1 + o(1))\sqrt{\ln(|\Delta|) \ln \ln |\Delta|}\right).$$

Summary

The overall method is to solve $\mathbf{A}x = y$ with high precision and then recover the exact solution.

- Ursic-Patarra '76: solve $x = \mathbf{A}^{-1}y$ over reals, use continued fractions on each coordinate

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 65 \\ 35 \end{pmatrix} \approx \begin{pmatrix} 83.1818 \\ -13.1818 \end{pmatrix} \implies x = \begin{pmatrix} 83 + 2/11 \\ -13 - 2/11 \end{pmatrix}$$

Summary

The overall method is to solve $\mathbf{A}x = y$ with high precision and then recover the exact solution.

- Ursic-Patarra '76: solve $x = \mathbf{A}^{-1}y$ over reals, use continued fractions on each coordinate

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 65 \\ 35 \end{pmatrix} \approx \begin{pmatrix} 83.1818 \\ -13.1818 \end{pmatrix} \implies x = \begin{pmatrix} 83 + 2/11 \\ -13 - 2/11 \end{pmatrix}$$

- Moenck-Carter '79: over X-adics/p-adics, solve $\mathbf{A}^{-1}y \bmod p^k$ one p-adic digit after another

Summary

The overall method is to solve $\mathbf{A}x = y$ with high precision and then recover the exact solution.

- Ursic-Patarra '76: solve $x = \mathbf{A}^{-1}y$ over reals, use continued fractions on each coordinate

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 65 \\ 35 \end{pmatrix} \approx \begin{pmatrix} 83.1818 \\ -13.1818 \end{pmatrix} \implies x = \begin{pmatrix} 83 + 2/11 \\ -13 - 2/11 \end{pmatrix}$$

- Moenck-Carter '79: over X-adics/p-adics, solve $\mathbf{A}^{-1}y \bmod p^k$ one p-adic digit after another
- Dixon '82: same over p-adics but with correct complexity: $\mathrm{O}\left(n^3\right)$

$$\mathbf{A}^{-1} = \begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & -1 \\ -2 & 3 \end{pmatrix} \bmod 13$$

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix}^{-1} \begin{pmatrix} 65 \\ 35 \end{pmatrix} = \begin{pmatrix} 4 + 12 \cdot 13 + 9 \cdot 13^2 + 4 \cdot 13^3 \\ 1 + 6 \cdot 13 + 3 \cdot 13^2 + 8 \cdot 13^3 \end{pmatrix} \bmod 13^4$$

Summary

The overall method is to solve $\mathbf{A}x = y$ with high precision and then recover the exact solution.

- Ursic-Patarra '76: solve $x = \mathbf{A}^{-1}y$ over reals, use continued fractions on each coordinate

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 65 \\ 35 \end{pmatrix} \approx \begin{pmatrix} 83.1818 \\ -13.1818 \end{pmatrix} \implies x = \begin{pmatrix} 83 + 2/11 \\ -13 - 2/11 \end{pmatrix}$$

- Moenck-Carter '79: over X-adics/p-adics, solve $\mathbf{A}^{-1}y \bmod p^k$ one p-adic digit after another
- Dixon '82: same over p-adics but with correct complexity: $\mathrm{O}\left(n^3\right)$

$$\mathbf{A}^{-1} = \begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & -1 \\ -2 & 3 \end{pmatrix} \bmod 13$$

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix}^{-1} \begin{pmatrix} 65 \\ 35 \end{pmatrix} = \begin{pmatrix} 4 + 12 \cdot 13 + 9 \cdot 13^2 + 4 \cdot 13^3 \\ 1 + 6 \cdot 13 + 3 \cdot 13^2 + 8 \cdot 13^3 \end{pmatrix} \bmod 13^4$$

- Storjohann '00: solve n systems in parallel, $\tilde{\mathrm{O}}\left(n^\omega\right)$

Table of Contents

Old iterative algorithms

Sparse matrix

We consider $\mathbf{A}$ sparse so that computing $\mathbf{A}x$ has a complexity of $\tilde{\mathrm{O}}(n)$.

Old iterative algorithms

Sparse matrix

We consider $\mathbf{A}$ sparse so that computing $\mathbf{A}x$ has a complexity of $\tilde{\mathrm{O}}(n)$.

Iterative refinement (Hensel, Wilkinson, ...)

If we can find x' such that $\mathbf{A}x' \approx y$, the solution is $x = x' + \mathbf{A}^{-1}(y - \mathbf{A}x')$.

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix} \cdot \begin{pmatrix} 83 \\ -13 \end{pmatrix} = \begin{pmatrix} 65 \\ 35 \end{pmatrix} + \begin{pmatrix} 2 \\ 14 \end{pmatrix}$$

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix} \cdot \begin{pmatrix} 18 \\ -18 \end{pmatrix} = \begin{pmatrix} -200 \\ -1400 \end{pmatrix} + \begin{pmatrix} 2 \\ 14 \end{pmatrix}$$

Old iterative algorithms

Sparse matrix

We consider $\mathbf{A}$ sparse so that computing $\mathbf{A}x$ has a complexity of $\tilde{O}(n)$.

Iterative refinement (Hensel, Wilkinson, …)

If we can find x' such that $\mathbf{A}x' \approx y$, the solution is $x = x' + \mathbf{A}^{-1}(y - \mathbf{A}x')$.

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix} \cdot \begin{pmatrix} 83 \\ -13 \end{pmatrix} = \begin{pmatrix} 65 \\ 35 \end{pmatrix} + \begin{pmatrix} 2 \\ 14 \end{pmatrix}$$

$$\begin{pmatrix} 3 & 14 \\ 15 & 92 \end{pmatrix} \cdot \begin{pmatrix} 18 \\ -18 \end{pmatrix} = \begin{pmatrix} -200 \\ -1400 \end{pmatrix} + \begin{pmatrix} 2 \\ 14 \end{pmatrix}$$

Jacobi's solver

If $\mathbf{A} \approx \mathbf{D}$, select $x' = \mathbf{D}^{-1}y$. If $\forall i, |\mathbf{A}_{i,i}| \geq K \sum_{j \neq i} |\mathbf{A}_{i,j}|$, the error is divided by K.

Conjugate gradient

Conjugate gradient, Lanczos 50s

For $\mathbf{G} > 0$ (positive symmetric definite), let $\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}$; we fix $\lambda_{\min} = 1$.
Accelerated gradient descent/Conjugate gradient: $\sqrt{\kappa}\log(\epsilon^{-1})$ matrix-vector products.

Proof: appendix.

Our algorithm, Cholesky decomposition step

Norm equation

If we have $\mathbf{A}^t\mathbf{A}x = \mathbf{A}^t y$ then $\mathbf{A}x = y$. Split the matrix $\mathbf{A}$ vertically into two matrices $\begin{pmatrix}\mathbf{E} & \mathbf{F}\end{pmatrix}$ with $\mathbf{F}$ having $K \ll n$ columns.

Our algorithm, Cholesky decomposition step

Norm equation

If we have $\mathbf{A}^t\mathbf{A}x = \mathbf{A}^t y$ then $\mathbf{A}x = y$. Split the matrix $\mathbf{A}$ vertically into two matrices $\begin{pmatrix}\mathbf{E} & \mathbf{F}\end{pmatrix}$ with $\mathbf{F}$ having $K \ll n$ columns.

LDL decomposition of $\mathbf{A}^t\mathbf{A}$

With $\mathbf{D}_0 = \mathbf{E}^t\mathbf{E}$, the Schur complement $\mathbf{D}_1 = \mathbf{F}^t\mathbf{F} - (\mathbf{F}^t\mathbf{E})\mathbf{D}_0^{-1}(\mathbf{F}^t\mathbf{E})^t$ and $\mathbf{L} = \mathbf{F}^t\mathbf{E}\mathbf{D}_0^{-1}$ of dimension $K \times (n-K)$, the block LDL decomposition of $\mathbf{A}^t\mathbf{A}$ is

$$\mathbf{A}^t\mathbf{A} = \begin{pmatrix}\mathbf{Id}_{n-K} & 0 \\ \mathbf{L} & \mathbf{Id}_K\end{pmatrix}\begin{pmatrix}\mathbf{D}_0 & 0 \\ 0 & \mathbf{D}_1\end{pmatrix}\begin{pmatrix}\mathbf{Id}_{n-K} & \mathbf{L}^t \\ 0 & \mathbf{Id}_K\end{pmatrix}$$

with $\mathbf{D}_1$ a $K \times K$ matrix.

Cholesky decomposition

With $\mathbf{L} = \mathbf{F}^t \mathbf{E} \mathbf{D}_0^{-1}$:

Inverse

$$(\mathbf{A}^t\mathbf{A})^{-1} = \begin{pmatrix} \mathbf{Id}_{n-K} & -\mathbf{L}^t \\ 0 & \mathbf{Id}_K \end{pmatrix} \begin{pmatrix} \mathbf{D}_0^{-1} & 0 \\ 0 & \mathbf{D}_1^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{Id}_{n-K} & 0 \\ -\mathbf{L} & \mathbf{Id}_K \end{pmatrix}.$$

Two multiplications by $\mathbf{D}_0^{-1}$, one by $\mathbf{D}_1^{-1}$, plus negligible ($\mathbf{E}, \mathbf{F}$ sparse).

Our fast random matrix solver

Conditioning

We solve by $\mathbf{D}_0 = \mathbf{E}^t\mathbf{E}$ using the conjugate gradient. It is believed that $\kappa \approx \frac{n^2}{K^2}$, since $\mathbf{E}$ is sparse, the complexity is $\tilde{\mathrm{O}}\left(\sqrt{\kappa} \cdot n\right) = \tilde{\mathrm{O}}\left(n^2/K\right)$.

Our fast random matrix solver

Conditioning

We solve by $\mathbf{D}_0 = \mathbf{E}^t\mathbf{E}$ using the conjugate gradient. It is believed that $\kappa \approx \frac{n^2}{K^2}$, since $\mathbf{E}$ is sparse, the complexity is $\tilde{\mathrm{O}}\left(\sqrt{\kappa} \cdot n\right) = \tilde{\mathrm{O}}\left(n^2/K\right)$.

Precomputation step of our algorithm

Precompute the dense $K \times K$ matrix $\mathbf{D}_1 = \mathbf{F}^t\mathbf{F} - (\mathbf{F}^t\mathbf{E})\mathbf{D}_0^{-1}(\mathbf{F}^t\mathbf{E})^t$ with K calls to $\mathbf{D}_0^{-1}$ for a cost of $\tilde{\mathrm{O}}\left(K \cdot n^2/K\right) = \tilde{\mathrm{O}}\left(n^2\right)$. Precompute the inverse $\mathbf{D}_1^{-1}$, with complexity $\tilde{\mathrm{O}}\left(K^\omega\right)$.

Our fast random matrix solver

Conditioning

We solve by $\mathbf{D}_0 = \mathbf{E}^t\mathbf{E}$ using the conjugate gradient. It is believed that $\kappa \approx \frac{n^2}{K^2}$, since $\mathbf{E}$ is sparse, the complexity is $\tilde{\mathrm{O}}\left(\sqrt{\kappa} \cdot n\right) = \tilde{\mathrm{O}}\left(n^2/K\right)$.

Precomputation step of our algorithm

Precompute the dense $K \times K$ matrix $\mathbf{D}_1 = \mathbf{F}^t\mathbf{F} - (\mathbf{F}^t\mathbf{E})\mathbf{D}_0^{-1}(\mathbf{F}^t\mathbf{E})^t$ with K calls to $\mathbf{D}_0^{-1}$ for a cost of $\tilde{\mathrm{O}}\left(K \cdot n^2/K\right) = \tilde{\mathrm{O}}\left(n^2\right)$. Precompute the inverse $\mathbf{D}_1^{-1}$, with complexity $\tilde{\mathrm{O}}\left(K^\omega\right)$.

$\mathbf{D}_1^{-1}$

Use standard multiplication for $\mathbf{D}_1^{-1}$, with complexity $\mathrm{O}\left(K^2\right)$.
It is faster for numerous vectors at the same time, $\mathbf{A}\mathbf{X} = \mathbf{Y}$.

Our results

Diagonally-dominant case

If $\mathbf{A}$ is a diagonally-dominant matrix, we have $\tilde{\mathrm{O}}(n)$ per system and bit.

Our results

Diagonally-dominant case

If $\mathbf{A}$ is a diagonally-dominant matrix, we have $\tilde{\mathrm{O}}\left(n\right)$ per system and bit.

$\omega = 3$

Take $K = n^{2/3}$ and after a precomputation of complexity $\tilde{\mathrm{O}}\left(n^2\right)$ we have a complexity in $\tilde{\mathrm{O}}\left(n^2/K + K^2\right) = \tilde{\mathrm{O}}\left(n^{4/3}\right)$ per system and bit.

Our results

Diagonally-dominant case

If **A** is a diagonally-dominant matrix, we have $\tilde{\mathrm{O}}(n)$ per system and bit.

$\omega = 3$

Take $K = n^{2/3}$ and after a precomputation of complexity $\tilde{\mathrm{O}}(n^2)$ we have a complexity in $\tilde{\mathrm{O}}(n^2/K + K^2) = \tilde{\mathrm{O}}(n^{4/3})$ per system and bit.

Full inverse

We want to solve n linear systems with $\sqrt{n}$ bits.
Use Storjohann's high-order lifting: we solve $n^{1.3}$ systems with precision $n^{0.2}$ bits.
Take $K = n^{0.87}$, use fast rectangular matrix multiplication, and we obtain $n^{1.13}$ per system and bit.

Table of Contents

Introduction

Problem

Given an invertible $\mathbf{A}$ over the Euclidean ring $\mathcal{R}$, solve $\mathbf{A}x + \mathbf{B}y = z \in \mathcal{R}^n$, namely $x + \mathbf{A}^{-1}\mathbf{B}y = \mathbf{A}^{-1}z$.

Since $x \in \mathcal{R}^n$ and $y \in \mathcal{R}^m$, this is a lattice problem of dimension $n + m$.

Precomputations use $z = 0$.

Introduction

Problem

Given an invertible $\mathbf{A}$ over the Euclidean ring $\mathcal{R}$, solve $\mathbf{A}x + \mathbf{B}y = z \in \mathcal{R}^n$, namely $x + \mathbf{A}^{-1}\mathbf{B}y = \mathbf{A}^{-1}z$.
Since $x \in \mathcal{R}^n$ and $y \in \mathcal{R}^m$, this is a lattice problem of dimension $n + m$.
Precomputations use $z = 0$.

Projection

We sample a matrix $\mathbf{P}^t$ in $\mathcal{R}$, and solve instead $\mathbf{P}^t\mathbf{A}x + \mathbf{P}^t\mathbf{B}y = \mathbf{P}^t z$, or $x' + \mathbf{P}^t\mathbf{A}^{-1}\mathbf{B}y = \mathbf{P}^t\mathbf{A}^{-1}z$.
With m columns, we now have a lattice of dimension $2m \approx 2\sqrt{n}$.

Precomputation

Our algorithm, precomputation of NTRU lattice

We compute $\tilde{\mathbf{C}} \approx \mathbf{C} = \mathbf{P}^t \mathbf{A}^{-1} \mathbf{B}$. Then with $\epsilon = \|\tilde{\mathbf{C}} - \mathbf{C}\|$ we reduce the lattice

$$\begin{pmatrix} \mathbf{Id}_m & \tilde{\mathbf{C}} \\ 0 & \epsilon \mathbf{Id}_m \end{pmatrix} \mathcal{R}^{2m}.$$

Precomputation

Our algorithm, precomputation of NTRU lattice

We compute $\tilde{\mathbf{C}} \approx \mathbf{C} = \mathbf{P}^t\mathbf{A}^{-1}\mathbf{B}$. Then with $\epsilon = \|\tilde{\mathbf{C}} - \mathbf{C}\|$ we reduce the lattice

$$\begin{pmatrix} \mathbf{Id}_m & \tilde{\mathbf{C}} \\ 0 & \epsilon\mathbf{Id}_m \end{pmatrix} \mathcal{R}^{2m}.$$

Property

Suppose the columns of $\mathbf{B}$ generate $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$. Then the lattice of $x + \mathbf{A}^{-1}\mathbf{B}y = 0$ has volume $\mathrm{vol}(\mathbf{A}) = |\det \mathbf{A}|$.

We expect to find the sublattice of all y in the NTRU lattice:

$$\begin{pmatrix} (\tilde{\mathbf{C}} - \mathbf{C})y \\ \epsilon y \end{pmatrix}.$$

Proof next slide.

Proof of precomputation success

Assumptions

Linear combinations $\mathbf{B}y$ cover $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$ with $\|y\| \leq M$; same for $\mathbf{P}$ and $\mathbf{A}^t$.

Proof of precomputation success

Assumptions

Linear combinations $\mathbf{B}y$ cover $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$ with $\|y\| \leq M$; same for $\mathbf{P}$ and $\mathbf{A}^t$.

Proof of assumption by the Gaussian technique, from Hildebrand

Only if the rank of $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$ is less than $m-2$.
Sample y according to a discrete Gaussian with standard deviation $\approx \mathrm{vol}(\mathbf{A})^{1/m} \leq \|\mathbf{A}\|^{n/m}$. Then if $\mathbf{B}$ mod $\mathbf{A}$ is sampled uniformly, $\mathbf{B}y$ is uniform modulo $\mathbf{A}$.

Proof of precomputation success

Assumptions

Linear combinations $\mathbf{B}y$ cover $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$ with $\|y\| \leq M$; same for $\mathbf{P}$ and $\mathbf{A}^t$.

Proof of assumption by the Gaussian technique, from Hildebrand

Only if the rank of $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$ is less than $m-2$.
Sample y according to a discrete Gaussian with standard deviation $\approx \mathrm{vol}(\mathbf{A})^{1/m} \leq \|\mathbf{A}\|^{n/m}$. Then if $\mathbf{B}$ mod $\mathbf{A}$ is sampled uniformly, $\mathbf{B}y$ is uniform modulo $\mathbf{A}$.

Lattice property

Let $\|\epsilon y\|, \|x' + \mathbf{C}y\| \leq 1/(3nM\|\mathbf{A}\|)$. Consider $\mathbf{P}z = e_i + \mathbf{A}^t w$, $\|z\| \leq M$.
Then $(\mathbf{A}^{-1}\mathbf{B}y)_i = e_i^t\mathbf{A}^{-1}\mathbf{B}y = (z^t\mathbf{P}^t - w^t\mathbf{A})\mathbf{A}^{-1}\mathbf{B}y$ which is

$$z^t\mathbf{C}y - w^t\mathbf{B}y = z^t(x' + \mathbf{C}y) \bmod 1$$

but $\mathbf{B}y = \mathbf{A}(\mathbf{A}^{-1}\mathbf{B}y) \in \mathcal{R}^n$ so $\mathbf{A}^{-1}\mathbf{B}y \in \mathcal{R}^n$.

Our fast solver

Precomputation complexity

We have m systems to be solved with precision $\|\mathbf{A}\|^{-n/m}$. If $\mathbf{A}$ is diagonally-dominant this takes $\tilde{\mathrm{O}}\left(n^2\right)$.

Reducing a dimension $2m$ lattice with precision n/m takes time $\tilde{\mathrm{O}}\left(m^\omega \cdot n/m\right)$.

Our fast solver

Precomputation complexity

We have m systems to be solved with precision $\|\mathbf{A}\|^{-n/m}$. If $\mathbf{A}$ is diagonally-dominant this takes $\tilde{\mathrm{O}}\left(n^2\right)$.
Reducing a dimension $2m$ lattice with precision n/m takes time $\tilde{\mathrm{O}}\left(m^\omega \cdot n/m\right)$.

Extracting a solution

Suppose there is a solution $\mathbf{A}x + \mathbf{B}y = z$. This leads to a solution of the form $x' + \mathbf{C}y = \mathbf{P}^t\mathbf{A}^{-1}z$, given by a lattice point close to

$$\begin{pmatrix} \mathbf{P}^t\mathbf{A}^{-1}z \\ 0 \end{pmatrix} \approx \begin{pmatrix} x' + \tilde{\mathbf{C}}y \\ \epsilon y \end{pmatrix}.$$

Then $x = \mathbf{A}^{-1}(z - \mathbf{B}y)$.
We need to solve a system with precision $\sqrt{n}$.

Miscellaneous

Invariant factors

If $\mathbf{B}$ generates $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$, we obtain the invariant factors from lattice reduction, in particular $|\det \mathbf{A}|$.

Miscellaneous

Invariant factors

If $\mathbf{B}$ generates $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$, we obtain the invariant factors from lattice reduction, in particular $\left|\det \mathbf{A}\right|$.

Block Wiedemann (Coppersmith, Villard, Kaltofen)

Take $\mathcal{R} = \mathbb{K}[X]$, for example $\mathbb{K} = \mathbb{F}_q$. For solving $-\mathbf{E}x = y$, choose $\mathbf{A} = X\mathbf{Id}_n - \mathbf{E}$ and reduce modulo X the solution. $\mathbf{A}$ is diagonally-dominant and $\mathbf{A}^{-1} = \sum_{i=0}^{\infty} X^{-i-1}\mathbf{E}^i$. We obtain Eberly *et al.*'s 2007 solver with complexity $\tilde{\mathrm{O}}\left(n^{1.5}\right)$.

Miscellaneous

Invariant factors

If $\mathbf{B}$ generates $\mathcal{R}^n/\mathbf{A}\mathcal{R}^n$, we obtain the invariant factors from lattice reduction, in particular $\left|\det \mathbf{A}\right|$.

Block Wiedemann (Coppersmith, Villard, Kaltofen)

Take $\mathcal{R} = \mathbb{K}[X]$, for example $\mathbb{K} = \mathbb{F}_q$. For solving $-\mathbf{E}x = y$, choose $\mathbf{A} = X\mathbf{Id}_n - \mathbf{E}$ and reduce modulo X the solution. $\mathbf{A}$ is diagonally-dominant and $\mathbf{A}^{-1} = \sum_{i=0}^{\infty} X^{-i-1}\mathbf{E}^i$. We obtain Eberly *et al.*'s 2007 solver with complexity $\tilde{\mathrm{O}}\left(n^{1.5}\right)$.

Peng-Vempala, Nie

With $\mathbb{K} = \mathbb{Q}$, $\mathcal{R} = \mathbb{Q}[X]$, $\mathbf{B}, \mathbf{P}$ sampled according to Gaussians, polynomial lattice reduction is well-conditioned. We can compute $\det \mathbf{E}$ within 1.1 in time $\mathrm{O}\left(n^{2.34}\right)$.

Table of Contents

Definition

Problems

Consider $\mathbb{K} = \mathbb{Q}[x]/f(x)$ a number field, with $\mathcal{O}_{\mathbb{K}}$ its maximal order of integers.

- Find ideals $\mathfrak{g}_1, \dots, \mathfrak{g}_k$ generating the class group and their order o_k in the class group with $o_1 \mid \cdots \mid o_k$ such that the class group is isomorphic to $\prod_{i=1}^{k} \langle \mathfrak{g}_i \rangle$
- Given a basis of an ideal $\mathfrak{a}$, find a decomposition of it, which is the class group exponents $a_1, \dots, a_k$ and a generator $g \in \mathbb{K}$ where $\mathfrak{a} = (g) \prod_{i=1}^{k} \mathfrak{g}_i^{a_i}$
- Find the $r_{\mathbb{R}} + r_{\mathbb{C}} - 1$ generators u_i of $\mathcal{O}^{\times}$, the unit group

Definition

Problems

Consider $\mathbb{K} = \mathbb{Q}[x]/f(x)$ a number field, with $\mathcal{O}_{\mathbb{K}}$ its maximal order of integers.

- Find ideals $\mathfrak{g}_1, \ldots, \mathfrak{g}_k$ generating the class group and their order o_k in the class group with $o_1 \mid \cdots \mid o_k$ such that the class group is isomorphic to $\prod_{i=1}^{k} \langle \mathfrak{g}_i \rangle$
- Given a basis of an ideal $\mathfrak{a}$, find a decomposition of it, which is the class group exponents $a_1, \ldots, a_k$ and a generator $g \in \mathbb{K}$ where $\mathfrak{a} = (g) \prod_{i=1}^{k} \mathfrak{g}_i^{a_i}$
- Find the $r_{\mathbb{R}} + r_{\mathbb{C}} - 1$ generators u_i of $\mathcal{O}^{\times}$, the unit group

We consider a field with small degree.

Cryptanalysis

Decomposition, or finding the group order cryptanalyze various systems (RSA without trusted setup).

Ideal reduction

Classical algorithm (Minkowski)

We are given $\mathfrak{a}$ (a basis), compute $v \in \mathfrak{a}^{-1}$. Then $v\mathfrak{a} \subset \mathcal{O}_{\mathbb{K}}$ and of index (norm) $\approx \sqrt{|\Delta_{\mathbb{K}}|}$.

Ideal reduction

Classical algorithm (Minkowski)

We are given $\mathfrak{a}$ (a basis), compute $v \in \mathfrak{a}^{-1}$. Then $v\mathfrak{a} \subset \mathcal{O}_{\mathbb{K}}$ and of index (norm) $\approx \sqrt{|\Delta_{\mathbb{K}}|}$.

Smoothness

Call $\mathfrak{p}_i \forall i \leq B$ the ideals of small norm the *factor basis*, the cardinal is

$$B = \exp\left((\frac{1}{2} + o(1)) \sqrt{\ln |\Delta_{\mathbb{K}}| \ln \ln |\Delta_{\mathbb{K}}|} \right).$$

A random ideal of norm $\sqrt{|\Delta_{\mathbb{K}}|}$ has probability $1/B$ of factoring over the base.

Descent

Generalized Riemann Hypothesis

Prime ideals $\mathfrak{p}$ with norm below $\approx \log^2(|\Delta_{\mathbb{K}}|)$ generate the class group.

Descent

Generalized Riemann Hypothesis

Prime ideals $\mathfrak{p}$ with norm below $\approx \log^2(|\Delta_{\mathbb{K}}|)$ generate the class group.

Descent

Given $\mathfrak{a}$, we want $v\mathfrak{a} = \prod_i \mathfrak{p}_i^{e_i}$.

We sample $e_i' \in \mathbb{Z}$, reduce $\mathfrak{a} \prod_i \mathfrak{p}_i^{e_i'}$ and detect the smoothness of the norm.

Improvements

Linear algebra

A relation is of the form $v\mathcal{O}_{\mathbb{K}} = \prod_i \mathfrak{p}_i^{e_i}$. Put the exponents in the columns of **A** and **B**, remove the first $\log^2$ prime ideals. Any $\mathbf{A}x + \mathbf{B}y = 0$ leads to a relation on the generators. A Smith Normal Form algorithm computes the structure of an abelian group from relations.

Improvements

Linear algebra

A relation is of the form $v\mathcal{O}_{\mathbb{K}} = \prod_i \mathfrak{p}_i^{e_i}$. Put the exponents in the columns of $\mathbf{A}$ and $\mathbf{B}$, remove the first $\log^2$ prime ideals. Any $\mathbf{A}x + \mathbf{B}y = 0$ leads to a relation on the generators. A Smith Normal Form algorithm computes the structure of an abelian group from relations.

Generator (PIP)

Assume a descent on $\mathfrak{a}$ leads to $v\mathfrak{a} = \prod_i \mathfrak{p}_i^{z_i}$.
Then we solve $\mathbf{A}x + \mathbf{B}y = z$.

Relation vs. descents

Descent

A descent on $\mathfrak{p}_i^D$ guarantees the generation of a diagonally-dominant matrix of relations $\mathbf{A}$.

Relation vs. descents

Descent

A descent on $\mathfrak{p}_i^D$ guarantees the generation of a diagonally-dominant matrix of relations $\mathbf{A}$.

Special fields

Special fields have for example $\|f\|$ small so that $ax + b$ are much more likely to be smooth. $\mathbf{A}$ is then not diagonally-dominant.

Conclusion

- New preconditioning algorithm for random sparse matrices

Conclusion

- New preconditioning algorithm for random sparse matrices
- The Block Wiedemann algorithm was generalized to integers

Conclusion

- New preconditioning algorithm for random sparse matrices
- The Block Wiedemann algorithm was generalized to integers
- A big and stretched NTRU problem $\mathbf{F}^{-1}\mathbf{G}$ can be projected to

$$\mathbf{P}^t\mathbf{F}^{-1}\mathbf{G}$$

Conclusion

- New preconditioning algorithm for random sparse matrices
- The Block Wiedemann algorithm was generalized to integers
- A big and stretched NTRU problem $\mathbf{F}^{-1}\mathbf{G}$ can be projected to

$$\mathbf{P}^t\mathbf{F}^{-1}\mathbf{G}$$

- Computing a class group now has the asymptotic complexity which corresponds to relation finding

Conclusion

- New preconditioning algorithm for random sparse matrices
- The Block Wiedemann algorithm was generalized to integers
- A big and stretched NTRU problem $\mathbf{F}^{-1}\mathbf{G}$ can be projected to

$$\mathbf{P}^t\mathbf{F}^{-1}\mathbf{G}$$

- Computing a class group now has the asymptotic complexity which corresponds to relation finding
- Slightly improved discrete logarithm for hyperelliptic curves in genus $g \geq 3$

Accelerated gradient descent

Proof.

We take $x' = P(\mathbf{G})y$ for $P \in \mathbb{R}[X]$ of degree $d+1$, the error is $\|y - \mathbf{G}(P(\mathbf{G})y)\| = \|(1 - XP)(\mathbf{G})(y)\|$. Using the spectral theorem, the relative error is $\leq \sum_\lambda |(1 - XP)(\lambda)|$. For T a Chebyshev polynomial, i.e. $T = \frac{1}{2}(X - \sqrt{X^2 - 1})^d + \frac{1}{2}(X - \sqrt{X^2 - 1})^{-d}$, we choose

$$1 - XP = \frac{T(\frac{\kappa+1-2X}{\kappa-1})}{T(\frac{\kappa+1}{\kappa-1})}$$

so that the error made is $\leq T(\frac{\kappa+1}{\kappa-1})^{-1} \approx (1 - 1/\sqrt{\kappa})^d$. $\square$