Secure Linear Algebra over Finite Fields and over the Rationals

Frank Blom, Niek J. Bouman,
Berry Schoenmakers, Niels de Vreede

Technische Universiteit Eindhoven
University of Technology

Wednesday, May 30, 2018

Funded by EU H2020 SODA

Setting

- Secret-sharing based MPC
- Multi-party ($N_{\text{players}} \geq 3$) scenario

Setting

- Secret-sharing based MPC
- Multi-party ($N_{\text{players}} \geq 3$) scenario
- Protocols on top of abstract MPC "arithmetic black box"

Problem Sketch

Consider a matrix A and vector b with integral entries, secret-shared among the players

Task

- Compute vector x such that $Ax = b$.

Problem Sketch

Consider a matrix A and vector b with integral entries, secret-shared among the players

Task

- Compute vector x such that $Ax = b$.

Multiple Problem Variants

- Solution over a finite field $\mathbb{F}$ vs. over $\mathbb{Q}$

Problem Sketch

Consider a matrix A and vector b with integral entries, secret-shared among the players

Task

- Compute vector x such that $Ax = b$.

Multiple Problem Variants

- Solution over a finite field $\mathbb{F}$ vs. over $\mathbb{Q}$
- Size of A: Square vs. rectangular ("wide" or "tall")
- Rank of A: full-rank vs. singular, known vs. unknown.

Problem Sketch

Consider a matrix A and vector b with integral entries, secret-shared among the players

Task

- Compute vector x such that $Ax = b$.

Multiple Problem Variants

- Solution over a finite field $\mathbb{F}$ vs. over $\mathbb{Q}$
- Size of A: Square vs. rectangular ("wide" or "tall")
- Rank of A: full-rank vs. singular, known vs. unknown.
- Consistent vs. inconsistent

Problem Sketch

Consider a matrix A and vector b with integral entries, secret-shared among the players

Task

- Compute vector x such that $Ax = b$.

Multiple Problem Variants

- Solution over a finite field $\mathbb{F}$ vs. over $\mathbb{Q}$
- Size of A: Square vs. rectangular ("wide" or "tall")
- Rank of A: full-rank vs. singular, known vs. unknown.
- Consistent vs. inconsistent
- Finding least squared-error solution (over $\mathbb{Q}$):

$$x^* := \arg\min_x \|Ax - b\|_2$$

Talk Plan

1. Solution over $\mathbb{Q}$: A is square and has full rank,
2. Solution over a finite field $\mathbb{F}$ (A's rank unknown)
   2.1 Oblivious Elimination
   2.2 Block-Recursive Decomposition
3. Least-Squares Solution over $\mathbb{Q}$ (A's rank unknown)

Warmup: Solving Full-Rank System over $\mathbb{Q}$ (in MPC)

Motivation

Useful for privacy-preserving data processing / statistics / etc

Related Work: Secure Linear Algebra over $\mathbb{Q}$

Multi-party case

[Toft, 2009]

2-party case

Several results in the 2-party setting, like [Nikolaenko et al., 2013, Gascón et al., 2017, Joye, 2017, Giacomelli et al., 2017]

Nonetheless, we do not target the 2-party scenario in this work.

Solving $Ax = b$ over $\mathbb{Q}$ (A full rank)

- Let $A \in \mathbb{Z}^{n \times n}$
- Then, in general, $A^{-1} \in \mathbb{Q}^{n \times n}$.

Solving $Ax = b$ over $\mathbb{Q}$ (A full rank)

- Let $A \in \mathbb{Z}^{n \times n}$
- Then, in general, $A^{-1} \in \mathbb{Q}^{n \times n}$.
- Inverse of A can be written as follows:

$$A^{-1} = \frac{\operatorname{adj} A}{\det A}$$

where adj A is the adjugate of A

Solving $Ax = b$ over $\mathbb{Q}$ (A full rank)

- Let $A \in \mathbb{Z}^{n \times n}$
- Then, in general, $A^{-1} \in \mathbb{Q}^{n \times n}$.
- Inverse of A can be written as follows:

$$A^{-1} = \frac{\operatorname{adj} A}{\det A}$$

  where adj A is the adjugate of A
- adj A has integral entries

Solving $Ax = b$ over $\mathbb{Q}$ (A full rank)

- Let $A \in \mathbb{Z}^{n \times n}$
- Then, in general, $A^{-1} \in \mathbb{Q}^{n \times n}$.
- Inverse of A can be written as follows:

$$A^{-1} = \frac{\text{adj } A}{\det A}$$

  where adj A is the adjugate of A
- adj A has integral entries
- Solution x of the system $Ax = b$ can be represented as

$$\big(\text{adj}(A)b, \det(A)\big) \in \mathbb{Z}^n \times \mathbb{Z}$$

- Representation avoids occurrence of rational entries

Our Solution ($Ax = b$ over $\mathbb{Q}$, A full rank)

- We work over the finite field $\mathbb{F}_p := \mathbb{Z}/p\mathbb{Z}$, p prime
- A modification of protocol of [Cramer and Damgård, 2001] (which is based on [Bar-Ilan and Beaver, 1989])
- Modification: keep adjugate and determinant separate

Our Solution ($Ax = b$ over $\mathbb{Q}$, A full rank)

- We work over the finite field $\mathbb{F}_p := \mathbb{Z}/p\mathbb{Z}$, p prime
- A modification of protocol of [Cramer and Damgård, 2001] (which is based on [Bar-Ilan and Beaver, 1989])
- Modification: keep adjugate and determinant separate
- p must be large enough to represent $\det A$ and entries of $\mathrm{adj}(A)b$

Our Solution ($Ax = b$ over $\mathbb{Q}$, A full rank)

- We work over the finite field $\mathbb{F}_p := \mathbb{Z}/p\mathbb{Z}$, p prime
- A modification of protocol of [Cramer and Damgård, 2001] (which is based on [Bar-Ilan and Beaver, 1989])
- Modification: keep adjugate and determinant separate
- p must be large enough to represent $\det A$ and entries of $\mathrm{adj}(A)b$
- Bound on p follows essentially from Hadamard's inequality:

Lemma (Hadamard)

For any matrix $M \in [-B, B]^{n \times n}$

$$|\det M| \leq B^n n^{n/2}.$$

Computing $(\operatorname{adj} A, \det A)$ via Random Self-Reduction

1. Let $[\![A]\!]$ be Shamir-secret-shared over the field $\mathbb{F}_p$.

Computing $(\operatorname{adj} A, \det A)$ via Random Self-Reduction

1. Let $[\![A]\!]$ be Shamir-secret-shared over the field $\mathbb{F}_p$.
2. Sample lower triangular matrix $[\![L]\!] \in \mathbb{F}_p^{n \times n}$ having ones on its diagonal uniformly at random
3. Sample upper triangular matrix $[\![U]\!] \in \mathbb{F}_p^{n \times n}$ uniformly at random such that diagonal does not contain zeros.

Computing $(\operatorname{adj} A, \det A)$ via Random Self-Reduction

1. Let $[\![A]\!]$ be Shamir-secret-shared over the field $\mathbb{F}_p$.
2. Sample lower triangular matrix $[\![L]\!] \in \mathbb{F}_p^{n \times n}$ having ones on its diagonal uniformly at random
3. Sample upper triangular matrix $[\![U]\!] \in \mathbb{F}_p^{n \times n}$ uniformly at random such that diagonal does not contain zeros.
4. Compute $[\![R]\!] := [\![LU]\!]$ and $[\![d]\!] := [(\det R)] = (\prod_i \operatorname{diag}(U)_i)$

Computing $(\operatorname{adj} A, \det A)$ via Random Self-Reduction

1. Let $[\![A]\!]$ be Shamir-secret-shared over the field $\mathbb{F}_p$.
2. Sample lower triangular matrix $[\![L]\!] \in \mathbb{F}_p^{n \times n}$ having ones on its diagonal uniformly at random
3. Sample upper triangular matrix $[\![U]\!] \in \mathbb{F}_p^{n \times n}$ uniformly at random such that diagonal does not contain zeros.
4. Compute $[\![R]\!] := [\![LU]\!]$ and $[\![d]\!] := [(\det R)] = (\prod_i \operatorname{diag}(U)_i)$
5. Compute $[\![RA]\!]$ and reveal it
6. In the clear, compute $\operatorname{adj} RA$ and $\det RA$.

Computing $(\operatorname{adj} A, \det A)$ via Random Self-Reduction

1. Let $[\![A]\!]$ be Shamir-secret-shared over the field $\mathbb{F}_p$.
2. Sample lower triangular matrix $[\![L]\!] \in \mathbb{F}_p^{n \times n}$ having ones on its diagonal uniformly at random
3. Sample upper triangular matrix $[\![U]\!] \in \mathbb{F}_p^{n \times n}$ uniformly at random such that diagonal does not contain zeros.
4. Compute $[\![R]\!] := [\![LU]\!]$ and $[\![d]\!] := [(\det R)] = (\prod_i \operatorname{diag}(U)_i)$
5. Compute $[\![RA]\!]$ and reveal it
6. In the clear, compute $\operatorname{adj} RA$ and $\det RA$.
7. Compute $[\![\operatorname{adj} A]\!] := \operatorname{adj}(RA)[\![R]\!][\![d^{-1}]\!]$, $[\![\det A]\!] := \det(RA)[\![d^{-1}]\!]$

Computing $(\mathrm{adj}\, A, \det A)$ via Random Self-Reduction

1. Let $[\![A]\!]$ be Shamir-secret-shared over the field $\mathbb{F}_p$.
2. Sample lower triangular matrix $[\![L]\!] \in \mathbb{F}_p^{n \times n}$ having ones on its diagonal uniformly at random
3. Sample upper triangular matrix $[\![U]\!] \in \mathbb{F}_p^{n \times n}$ uniformly at random such that diagonal does not contain zeros.
4. Compute $[\![R]\!] := [\![LU]\!]$ and $[\![d]\!] := [(\det R)] = (\prod_i \mathrm{diag}(U)_i)$
5. Compute $[\![RA]\!]$ and reveal it
6. In the clear, compute adj RA and det RA.
7. Compute $[\![\mathrm{adj}\, A]\!] := \mathrm{adj}(RA)[\![R]\!][\![d^{-1}]\!]$, $[\![\det A]\!] := \det(RA)[\![d^{-1}]\!]$

L is uni-triangular: simplifies proof in [Cramer and Damgård, 2001] (and slightly fewer multiplications & saves 1 communication round)

Complexity

Solving $Ax = b$ securely over $\mathbb{Q}$, where A is square (n by n) and full rank.

Our work	# Rounds	# Secure Mults
Random Self-Reducibility	$O(1)$	$O(n^2)^*$

* Assuming "cheap" inner products (Shamir LSS)

Solution over $\mathbb{F}_p$, A's rank unknown

Oblivious Elimination

Related Work: Secure Linear Algebra over $\mathbb{F}_p$

Consider the linear system $Ax = b$, where A is an m by n matrix over finite field $\mathbb{F}_p$.

Reference	# Rounds	# Secure Mults
[Cramer and Damgård, 2001]	$O(1)$	$O(n^5)$*
[Cramer et al., 2007]	$O(1)$	$O(m^4 + n^2 m)$

* Assumption: $n \geq m$

Motivation (Solution over $\mathbb{F}_p$, Unknown-Rank Case)

- Existing constant-round-solutions have high computational complexity
- Trade-off: computational complexity vs. round complexity vs. communication complexity
- What can we get if we drop the constant-rounds property?

Oblivious Elimination

Given, $m \times n$ matrix A over $\mathbb{F}$ of unknown $\mathbb{F}$-rank and right-hand side $B \in \mathbb{F}^{m \times \ell}$

Basic idea

- Apply Integer-Preserving Gaussian Elim. [Bareiss, 1968]
- No pivoting (avoid expensive oblivious row/column swaps)
- Keep watching the diagonal elements (pivots), indicator for when we have "exhausted" the rank

Oblivious Elimination

Given, $m \times n$ matrix A over $\mathbb{F}$ of unknown $\mathbb{F}$-rank and right-hand side $B \in \mathbb{F}^{m \times \ell}$

Basic idea

- Apply Integer-Preserving Gaussian Elim. [Bareiss, 1968]
- No pivoting (avoid expensive oblivious row/column swaps)
- Keep watching the diagonal elements (pivots), indicator for when we have "exhausted" the rank

$$\begin{pmatrix} 36 & 30 & 22 & 45 \\ 49 & 39 & 33 & 53 \\ 67 & 51 & 49 & 62 \\ 45 & 39 & 25 & 63 \end{pmatrix}$$

Oblivious Elimination

Given, $m \times n$ matrix A over $\mathbb{F}$ of unknown $\mathbb{F}$-rank and right-hand side $B \in \mathbb{F}^{m \times \ell}$

Basic idea

- Apply Integer-Preserving Gaussian Elim. [Bareiss, 1968]
- No pivoting (avoid expensive oblivious row/column swaps)
- Keep watching the diagonal elements (pivots), indicator for when we have "exhausted" the rank

$$\begin{pmatrix} 36 & 30 & 22 & 45 \\ 0 & -66 & 110 & -297 \\ 0 & -174 & 290 & -783 \\ 0 & 54 & -90 & 243 \end{pmatrix}$$

Oblivious Elimination

Given, $m \times n$ matrix A over $\mathbb{F}$ of unknown $\mathbb{F}$-rank and right-hand side $B \in \mathbb{F}^{m \times \ell}$

Basic idea

- Apply Integer-Preserving Gaussian Elim. [Bareiss, 1968]
- No pivoting (avoid expensive oblivious row/column swaps)
- Keep watching the diagonal elements (pivots), indicator for when we have "exhausted" the rank

$$\begin{pmatrix} 36 & 0 & -4752 & 5940 \\ 0 & -66 & 110 & -297 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Oblivious Elimination

Given, $m \times n$ matrix A over $\mathbb{F}$ of unknown $\mathbb{F}$-rank and right-hand side $B \in \mathbb{F}^{m \times \ell}$

Basic idea

- Apply Integer-Preserving Gaussian Elim. [Bareiss, 1968]
- No pivoting (avoid expensive oblivious row/column swaps)
- Keep watching the diagonal elements (pivots), indicator for when we have "exhausted" the rank

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Oblivious Elimination

Given, $m \times n$ matrix A over $\mathbb{F}$ of unknown $\mathbb{F}$-rank and right-hand side $B \in \mathbb{F}^{m \times \ell}$

Basic idea

- Apply Integer-Preserving Gaussian Elim. [Bareiss, 1968]
- No pivoting (avoid expensive oblivious row/column swaps)
- Keep watching the diagonal elements (pivots), indicator for when we have "exhausted" the rank
- Upon exhausting the rank:
  - continue elimination with dummy operations (to avoid leaking the rank)

Oblivious Elimination

Given, $m \times n$ matrix A over $\mathbb{F}$ of unknown $\mathbb{F}$-rank and right-hand side $B \in \mathbb{F}^{m \times \ell}$

Basic idea

- Apply Integer-Preserving Gaussian Elim. [Bareiss, 1968]
- No pivoting (avoid expensive oblivious row/column swaps)
- Keep watching the diagonal elements (pivots), indicator for when we have "exhausted" the rank
- Upon exhausting the rank: …

Problem: Pivot-free GE fails for some matrices

- Success guaranteed iff A has generic rank profile: r leading principal minors of A are nonzero, where $r := \operatorname{rank} A$

Oblivious Elimination

Given, $m \times n$ matrix A over $\mathbb{F}$ of unknown $\mathbb{F}$-rank and right-hand side $B \in \mathbb{F}^{m \times \ell}$

Basic idea

- Apply Integer-Preserving Gaussian Elim. [Bareiss, 1968]
- No pivoting (avoid expensive oblivious row/column swaps)
- Keep watching the diagonal elements (pivots), indicator for when we have "exhausted" the rank
- Upon exhausting the rank: ...

Problem: Pivot-free GE fails for some matrices

- Success guaranteed iff A has generic rank profile: r leading principal minors of A are nonzero, where $r := \operatorname{rank} A$
- Can be achieved via Toeplitz preconditioning [Kaltofen and Saunders, 1991]

Kaltofen–Saunders lemma

Let $A \in \mathbb{F}^{n \times n}$ be arbitrary and let $r := \operatorname{rank} A$. Consider the matrix $A' := UAL$ with

$$U := \begin{bmatrix} 1 & u_2 & u_3 & \dots & u_n \\ & 1 & u_2 & \dots & u_{n-1} \\ & & 1 & \ddots & \vdots \\ & & & \ddots & u_2 \\ & & & & 1 \end{bmatrix}, \quad L := \begin{bmatrix} 1 & & & & \\ \ell_2 & 1 & & & \\ \ell_3 & \ell_2 & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ \ell_n & \ell_{n-1} & \dots & \ell_2 & 1 \end{bmatrix},$$

where u_i and ℓ_i for all $i \in \{2, \dots, n\}$ selected independently and uniformly at random from $\mathcal{S} \subseteq \mathbb{F}$.
Then,

$$\Pr(A' \text{ has generic rank profile}) \geq 1 - \frac{r(r+1)}{|\mathcal{S}|}.$$

Nullspace Computation & Consistency Check

- Apply elimination to augmented matrix

$$[\![C]\!] := \begin{pmatrix} U[\![A]\!]L & U[\![B]\!] \\ [\![I_n]\!] & 0^{n\times m} \end{pmatrix}$$

Nullspace Computation & Consistency Check

- Apply elimination to augmented matrix

$$[\![C]\!] := \begin{pmatrix} U[\![A]\!]L & U[\![B]\!] \\ [\![I_n]\!] & 0^{n\times m} \end{pmatrix}$$

Yields basis for the (right) nullspace of A

Nullspace Computation & Consistency Check

- Apply elimination to augmented matrix

$$[\![C]\!] := \begin{pmatrix} U[\![A]\!]L & U[\![B]\!] \\ [\![I_n]\!] & 0^{n\times m} \end{pmatrix}$$

- Column-wise consistency check by means of checking the candidate solution $\widetilde{X}_i$:

$$vA\widetilde{X}_i - vB_i \stackrel{?}{=} 0 \qquad \text{for a randomly chosen vector } v$$

Contributions: Solution to the $\mathbb{F}_p$-linear system

Consider the linear system $Ax = b$, where A is an m by n matrix over finite field $\mathbb{F}_p$.

Prior work	# Rounds	# Secure Mults
[Cramer and Damgård, 2001]	$O(1)$	$O(n^5)$
[Cramer et al., 2007]	$O(1)$	$O(m^4 + n^2 m)$
Our work	**# Rounds**	**# Secure Mults**
Oblivious Gaussian Elimination	$O(\min(m, n))$	$O(n^2 m)$

Can we use Obliv. GE to obtain solution over $\mathbb{Q}$? (Unknown-rank case)

- Like in the full-rank case, keep numerators and (common) denominator of the solution separated
- Coefficient-growth becomes important: Final values must not wrap around the modulus

Can we use Obliv. GE to obtain solution over $\mathbb{Q}$? (Unknown-rank case)

- Like in the full-rank case, keep numerators and (common) denominator of the solution separated
- Coefficient-growth becomes important: Final values must not wrap around the modulus
- Preconditioning becomes a problem:
  - Affects solution's numerators and common denominator
  - Precond. elements sampled from exponentially large set
  - Values in GE algorithm will quickly exceed modulus

Can we use Obliv. GE to obtain solution over $\mathbb{Q}$? (Unknown-rank case)

- Like in the full-rank case, keep numerators and (common) denominator of the solution separated
- Coefficient-growth becomes important: Final values must not wrap around the modulus
- Preconditioning becomes a problem:
  - Affects solution's numerators and common denominator
  - Precond. elements sampled from exponentially large set
  - Values in GE algorithm will quickly exceed modulus

Open Problem

How to apply pivoting efficiently in an MPC setting, or, how to perform generic-rank-profile preconditioning without introducing massive coefficient-growth?

Solution over $\mathbb{F}_p$, A's rank unknown

via Block-Recursive Decomposition

Block-Recursive Decomposition:

Some form of “divide-and-conquer” approach to (generalized) matrix inversion

Block-Recursive Decomposition:

Some form of "divide-and-conquer" approach to (generalized) matrix inversion

Full-rank matrices:

- [Strassen, 1969]: Computing matrix inverse has same asymptotic complexity as matrix multiplication
- [Bunch and Hopcroft, 1974]
- ...

Block-Recursive Decomposition:

Some form of "divide-and-conquer" approach to (generalized) matrix inversion

Full-rank matrices:

- [Strassen, 1969]: Computing matrix inverse has same asymptotic complexity as matrix multiplication
- [Bunch and Hopcroft, 1974]
- ...

Arbitrary-rank matrices:

- [Ibarra et al., 1982]
- Many others, see [Dumas et al., 2015] for overview
- [Malaschonok, 2010]: LEU decomposition
  Algorithm is a straight-line program (rank-insensitive time-complexity) and works over arbitrary field: suitable for MPC

Contributions: Solution to the $\mathbb{F}$-linear system

Consider the linear system $Ax = b$, where A is an m by n matrix over finite field $\mathbb{F}$.

Prior work	# Rounds	# Secure Mults
[Cramer and Damgård, 2001]	$O(1)$	$O(n^5)$
[Cramer et al., 2007]	$O(1)$	$O(m^4 + n^2 m)$
Our work	**# Rounds**	**# Secure Mults**
Oblivious Gaussian Elimination	$O(\min(m, n))$	$O(n^2 m)$
Block-Recursive Decomposition	$O(\max(m, n)^{1.59})$	$O(\max(m, n)^2)$

Least-Squares Solution over $\mathbb{Q}$,

A's rank unknown

Least-Squares Solution over $\mathbb{Q}$

Motivation

Plenty of applications, e.g.,:

- Fitting a line through data
- Solve a “noisy” system

Least-Squares Solution over $\mathbb{Q}$

Motivation

Plenty of applications, e.g.,:

- Fitting a line through data
- Solve a "noisy" system

Two Caveats

1. Non-standard scenario: Solution is revealed, followed by a rational reconstruction step "in the clear" [Wang, 1981]

Least-Squares Solution over $\mathbb{Q}$

Motivation

Plenty of applications, e.g.,:

- Fitting a line through data
- Solve a "noisy" system

Two Caveats

1. Non-standard scenario: Solution is revealed, followed by a rational reconstruction step "in the clear" [Wang, 1981]
   - Recover numerator r and denominator s via basis reduction in a 2D lattice (e.g., Lagrange–Gauss algorithm)

Least-Squares Solution over $\mathbb{Q}$

Motivation

Plenty of applications, e.g.,:

- Fitting a line through data
- Solve a "noisy" system

Two Caveats

1. Non-standard scenario: Solution is revealed, followed by a rational reconstruction step "in the clear" [Wang, 1981]
   - Recover numerator r and denominator s via basis reduction in a 2D lattice (e.g., Lagrange–Gauss algorithm)
   - Unique solution iff $|r|, |s| \leq \sqrt{p/2}$

Least-Squares Solution over $\mathbb{Q}$

Motivation

Plenty of applications, e.g.,:

- Fitting a line through data
- Solve a "noisy" system

Two Caveats

1. Non-standard scenario: Solution is revealed, followed by a rational reconstruction step "in the clear" [Wang, 1981]
   - Recover numerator r and denominator s via basis reduction in a 2D lattice (e.g., Lagrange–Gauss algorithm)
   - Unique solution iff $|r|, |s| \leq \sqrt{p/2}$
2. Non-standard assumption: the prime p of the finite field is chosen randomly from a large set, independently of values of matrix A and vector b.

   Makes sense against honest-but-curious adversary

A generalized Cramer's rule [Ben-Israel, 1982]

For $A \in \mathbb{C}^{m\times n}$ and $b \in \mathbb{C}^m$ consistent with A, solution given by:

$$x_j = \frac{\det \begin{bmatrix} A(j \to b) & U \\ V^\mathsf{T}(j \to 0) & 0 \end{bmatrix}}{\det \begin{bmatrix} A & U \\ V^\mathsf{T} & 0 \end{bmatrix}} \in \mathbb{C}, \qquad j \in [n],$$

where

- $U \in \mathbb{C}^{m\times m-r}$ is a basis for Ker A^T,
- $V \in \mathbb{C}^{n\times n-r}$ is a basis the Ker A,

A generalized Cramer's rule [Ben-Israel, 1982]

For $A \in \mathbb{C}^{m \times n}$ and $b \in \mathbb{C}^m$ consistent with A, solution given by:

$$x_j = \frac{\det \begin{bmatrix} A(j \to b) & U \\ V^\mathsf{T}(j \to 0) & 0 \end{bmatrix}}{\det \begin{bmatrix} A & U \\ V^\mathsf{T} & 0 \end{bmatrix}} \in \mathbb{C}, \qquad j \in [n],$$

where

- $U \in \mathbb{C}^{m \times m-r}$ is a basis for Ker A^T,
- $V \in \mathbb{C}^{n \times n-r}$ is a basis the Ker A,

[Verghese, 1982] proved that the same formula yields least-squares solution in inconsistent case

Using Ben-Israel/Verghese's Cramer's rule in MPC

High-Level Idea

- Apply Ben-Israel's Cramer's rule over $\mathbb{F}_p$
- Obtain solution over $\mathbb{Q}$ via rational reconstruction

Using Ben-Israel/Verghese's Cramer's rule in MPC

High-Level Idea

- Apply Ben-Israel's Cramer's rule over $\mathbb{F}_p$
- Obtain solution over $\mathbb{Q}$ via rational reconstruction
- Compute determinant in denominator via our random self-reducibility protocol

Using Ben-Israel/Verghese's Cramer's rule in MPC

High-Level Idea

- Apply Ben-Israel's Cramer's rule over $\mathbb{F}_p$
- Obtain solution over $\mathbb{Q}$ via rational reconstruction
- Compute determinant in denominator via our random self-reducibility protocol
- Determinant in the numerator(s) can be viewed as a rank-1 update of denominator:

Using Ben-Israel/Verghese's Cramer's rule in MPC

High-Level Idea

- Apply Ben-Israel's Cramer's rule over $\mathbb{F}_p$
- Obtain solution over $\mathbb{Q}$ via rational reconstruction
- Compute determinant in denominator via our random self-reducibility protocol
- Determinant in the numerator(s) can be viewed as a rank-1 update of denominator:

Lemma (Matrix Determinant Lemma)

Let $n \in \mathbb{N}$ be arbitrary. Let $M \in \mathbb{Z}^{n \times n}$ be a square matrix and let $u, v \in \mathbb{Z}^n$ be column vectors. Then, it holds that

$$\det(M + uv^\mathsf{T}) = \det(M) + v^\mathsf{T} \operatorname{adj}(M) u.$$

Using Ben-Israel/Verghese's Cramer's rule in MPC

Two problems

1. Matrices in numerator and denominator have rank-dependent dimensions

2. Matrices in numerator and denominator might not have full $\mathbb{F}_p$-rank

Using Ben-Israel/Verghese's Cramer's rule in MPC

Two problems

1. Matrices in numerator and denominator have rank-dependent dimensions (Easily dealt with by padding with ones on diagonal)
2. Matrices in numerator and denominator might not have full $\mathbb{F}_p$-rank

Using Ben-Israel/Verghese's Cramer's rule in MPC

Two problems

1. Matrices in numerator and denominator have rank-dependent dimensions (Easily dealt with by padding with ones on diagonal)
2. Matrices in numerator and denominator might not have full $\mathbb{F}_p$-rank
   - Diagonal preconditioning could avoid self-orthogonality with high-probability [Mulmuley, 1986, LaMacchia and Odlyzko, 1990, Diaz-Toca et al., 2005, Cramer et al., 2007]
   - Preconditioning "warps" the space, yields least-squares solution with respect to a "warped" distance measure

Using Ben-Israel/Verghese's Cramer's rule in MPC

"Way out"

- Omit (diagonal) preconditioning
- Assume: p chosen at random, independently of the elements of A and b, such that $p \gg \max(m, n)$
  $\implies$ probability of self-orthogonality is small

Protocol: LeastSq(A, b)

1: $([\![r]\!], [\![\begin{bmatrix} U & 0 \end{bmatrix}]\!], [\![\begin{bmatrix} V & 0 \end{bmatrix}]\!]) \leftarrow$ LRNullspace$([\![A]\!])$ $\quad \triangleright$ over $\mathbb{F}_p$

Protocol: LeastSq(A, b)

1: $(\llbracket r \rrbracket, \llbracket [U \quad 0] \rrbracket, \llbracket [V \quad 0] \rrbracket) \leftarrow$ LRNullspace$(\llbracket A \rrbracket)$ ▷ over $\mathbb{F}_p$
2: Form the matrix

$$\llbracket M \rrbracket := \begin{bmatrix} A & U & 0 \\ V^{\mathsf{T}} & 0 & 0 \\ 0 & 0 & I_{r\times r} \end{bmatrix} \in \mathbb{F}_p^{(n+m)\times(n+m)}.$$

Protocol: LeastSq(A, b)

1: $(\llbracket r \rrbracket, \llbracket \begin{bmatrix} U & 0 \end{bmatrix} \rrbracket, \llbracket \begin{bmatrix} V & 0 \end{bmatrix} \rrbracket) \leftarrow$ LRNullspace$(\llbracket A \rrbracket)$ ▷ over $\mathbb{F}_p$

2: Form the matrix

$$\llbracket M \rrbracket := \begin{bmatrix} A & U & 0 \\ V^{\mathsf{T}} & 0 & 0 \\ 0 & 0 & I_{r \times r} \end{bmatrix} \in \mathbb{F}_p^{(n+m)\times(n+m)}.$$

3: $(\llbracket \text{adj } M \rrbracket, \llbracket \det M \rrbracket) \leftarrow$ AdjDet$(\llbracket M \rrbracket)$

Protocol: LeastSq(A, b)

1: $(\llbracket r \rrbracket, \llbracket [U \quad 0] \rrbracket, \llbracket [V \quad 0] \rrbracket) \leftarrow$ LRNullspace$(\llbracket A \rrbracket)$ ▷ over $\mathbb{F}_p$

2: Form the matrix

$$\llbracket M \rrbracket := \begin{bmatrix} A & U & 0 \\ V^{\mathsf{T}} & 0 & 0 \\ 0 & 0 & I_{r \times r} \end{bmatrix} \in \mathbb{F}_p^{(n+m)\times(n+m)}.$$

3: $(\llbracket \text{adj } M \rrbracket, \llbracket \det M \rrbracket) \leftarrow$ AdjDet$(\llbracket M \rrbracket)$

4: Define $b_\circ$ as the column vector b padded with zeros up to length $n+m$.

For every $j \in [n]$:

5: Compute

$$\llbracket \tilde{x}_j \rrbracket := 1 + \llbracket (\det M)^{-1} \rrbracket \llbracket \text{Row}_j(\text{adj } M) \rrbracket \cdot \llbracket b_\circ - \text{Col}_j(M) \rrbracket$$

Protocol: $\mathsf{LeastSq}(A, b)$

1: $(\llbracket r \rrbracket, \llbracket [U \quad 0] \rrbracket, \llbracket [V \quad 0] \rrbracket) \leftarrow$ LRNullspace($\llbracket A \rrbracket$) ▷ over $\mathbb{F}_p$

2: Form the matrix

$$\llbracket M \rrbracket := \begin{bmatrix} A & U & 0 \\ V^\mathsf{T} & 0 & 0 \\ 0 & 0 & I_{r \times r} \end{bmatrix} \in \mathbb{F}_p^{(n+m)\times(n+m)}.$$

3: $(\llbracket \text{adj } M \rrbracket, \llbracket \det M \rrbracket) \leftarrow$ AdjDet($\llbracket M \rrbracket$)

4: Define $b_\circ$ as the column vector b padded with zeros up to length $n + m$.

For every $j \in [n]$:

5: Compute

$$\llbracket \tilde{x}_j \rrbracket := 1 + \llbracket (\det M)^{-1} \rrbracket \llbracket \text{Row}_j(\text{adj } M) \rrbracket \cdot \llbracket b_\circ - \text{Col}_j(M) \rrbracket$$

6: Reveal $\llbracket \tilde{x}_j \rrbracket$ to "output parties"

7: $x_j \leftarrow$ RationalReconstruct($\tilde{x}_j$)

Complexity

	# Rounds	# Secure Mults
Least-Squares	$R_{\text{nullspace}} + O(1)$	$M_{\text{nullspace}} + O(n^2)$

where $R_{\text{nullspace}}$ and $M_{\text{nullspace}}$ are the round and sec.-mult. complexities required for computing right and left nullspace of A over the finite field

Questions?

References I

Bar-Ilan, J. and Beaver, D. (1989).
Non-cryptographic fault-tolerant computing in constant number of rounds of interaction.
In *Proc. 8th Symp. on Princip. of Distr. Comp.*, pages 201–209, NY. ACM.

Bareiss, E. H. (1968).
Sylvester's identity and multistep integer-preserving gaussian elimination.
Mathematics of Computation, 22(103):565–578.

Ben-Israel, A. (1982).
A cramer rule for least-norm solutions of consistent linear equations.
Linear Algebra and its Applications, 43:223–226.

Bunch, J. R. and Hopcroft, J. E. (1974).
Triangular factorization and inversion by fast matrix multiplication.
Mathematics of Computation, 28(125):231–236.

Cramer, R. and Damgård, I. (2001).
Secure distributed linear algebra in a constant number of rounds.
In *Proc. CRYPTO 2001, Santa Barbara, USA*, pages 119–136. Springer.

References II

Cramer, R., Kiltz, E., and Padró, C. (2007).
A note on secure computation of the Moore–Penrose pseudoinverse and its application to secure linear algebra.
In *Proc. CRYPTO 2007, Santa Barbara, USA*, pages 613–630. Springer.

Diaz-Toca, G. M., Gonzalez-Vega, L., and Lombardi, H. (2005).
Generalizing cramer's rule: Solving uniformly linear systems of equations.
SIAM journal on matrix analysis and applications, 27(3):621–637.

Dumas, J.-G., Pernet, C., and Sultan, Z. (2015).
Computing the rank profile matrix.
In *Proceedings of the 2015 ACM on International Symposium on Symbolic and Algebraic Computation*, pages 149–156. ACM.

Gascón, A., Schoppmann, P., Balle, B., Raykova, M., Doerner, J., Zahur, S., and Evans, D. (2017).
Privacy-preserving distributed linear regression on high-dimensional data.
Proceedings on Privacy Enhancing Technologies, 2017(4):345–364.

Giacomelli, I., Jha, S., Joye, M., Page, C. D., and Yoon, K. (2017).
Privacy-preserving ridge regression over distributed data from LHE.
Cryptology ePrint Archive, Report 2017/979.

References III

Giesbrecht, M., Lobo, A., and Saunders, B. D. (1998).
Certifying inconsistency of sparse linear systems.
In *Proceedings of the 1998 international symposium on Symbolic and algebraic computation*, pages 113–119. ACM.

Ibarra, O. H., Moran, S., and Hui, R. (1982).
A generalization of the fast LUP matrix decomposition algorithm and applications.
Journal of Algorithms, 3(1):45–56.

Joye, M. (2017).
Privacy-preserving ridge regression without garbled circuits.
Cryptology ePrint Archive, Report 2017/732.

Kaltofen, E. and Saunders, B. D. (1991).
On Wiedemann's method of solving sparse linear systems.
In *International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, pages 29–38. Springer.

LaMacchia, B. A. and Odlyzko, A. M. (1990).
Solving large sparse linear systems over finite fields.
In *Conference on the Theory and Application of Cryptography*, pages 109–133. Springer.

References IV

Malaschonok, G. (2010).
Fast generalized Bruhat decomposition.
In *International Workshop on Computer Algebra in Scientific Computing*, pages 194–202. Springer.

Mulmuley, K. (1986).
A fast parallel algorithm to compute the rank of a matrix over an arbitrary field.
In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing*, STOC '86, pages 338–339, New York, NY, USA. ACM.

Nikolaenko, V., Weinsberg, U., Ioannidis, S., Joye, M., Boneh, D., and Taft, N. (2013).
Privacy-preserving ridge regression on hundreds of millions of records.
In *Proceedings of the 2013 IEEE Symposium on Security and Privacy*, SP '13, pages 334–348, Washington, DC, USA. IEEE Computer Society.

Strassen, V. (1969).
Gaussian elimination is not optimal.
Numerische mathematik, 13(4):354–356.

References V

Toft, T. (2009).
Solving linear programs using multiparty computation.
In *International Conference on Financial Cryptography and Data Security*, pages 90–107. Springer.

Verghese, G. C. (1982).
A "Cramer rule" for the least-norm, least-squared-error solution of inconsistent linear equations.
Linear Algebra and its Applications, 48:315–316.

Wang, P. S. (1981).
A p-adic algorithm for univariate partial fractions.
In *Proceedings of the Fourth ACM Symposium on Symbolic and Algebraic Computation*, SYMSAC '81, pages 212–217, NY, USA. ACM.