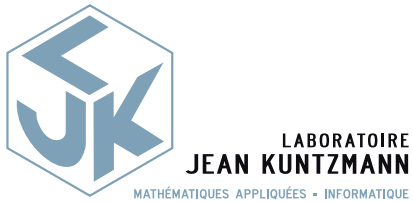
Certificates for Triangular Equivalences and Rank Profiles

Jean-Guillaume Dumas, **David Lucas**, Clément Pernet

Université Grenoble Alpes, Laboratoire Jean Kuntzmann, CNRS

ISSAC'17, Kaiserslautern, Germany
July 26, 2017

Supported by OpenDreamKit Horizon 2020 European Research Infrastructures project (#676541)



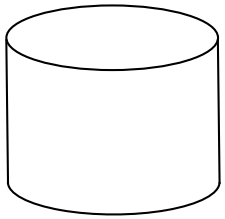
Outline

1. Introduction
2. Our certificates
3. Conclusion

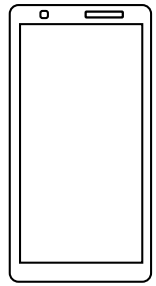
Plan

1. Introduction
2. Our certificates
3. Conclusion

Context: outsourced computation

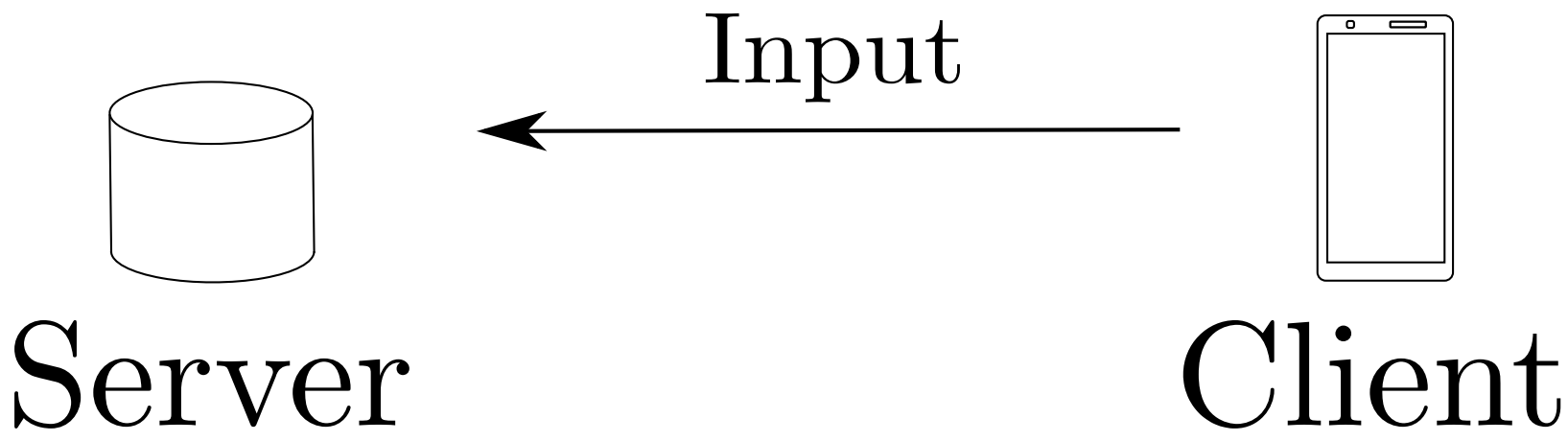


Server



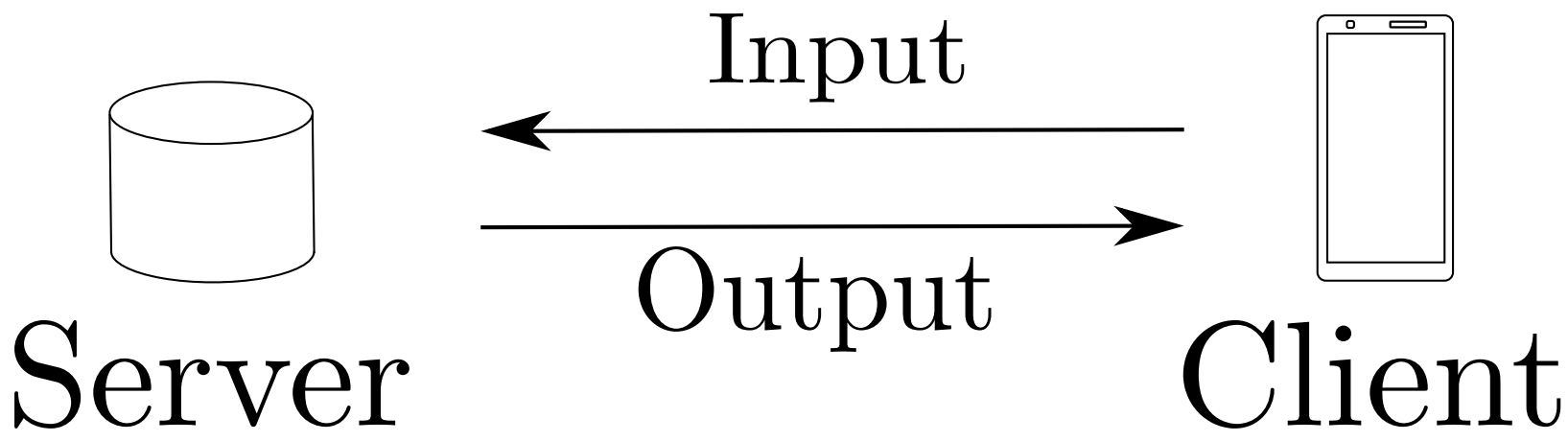
Client

Context: outsourced computation



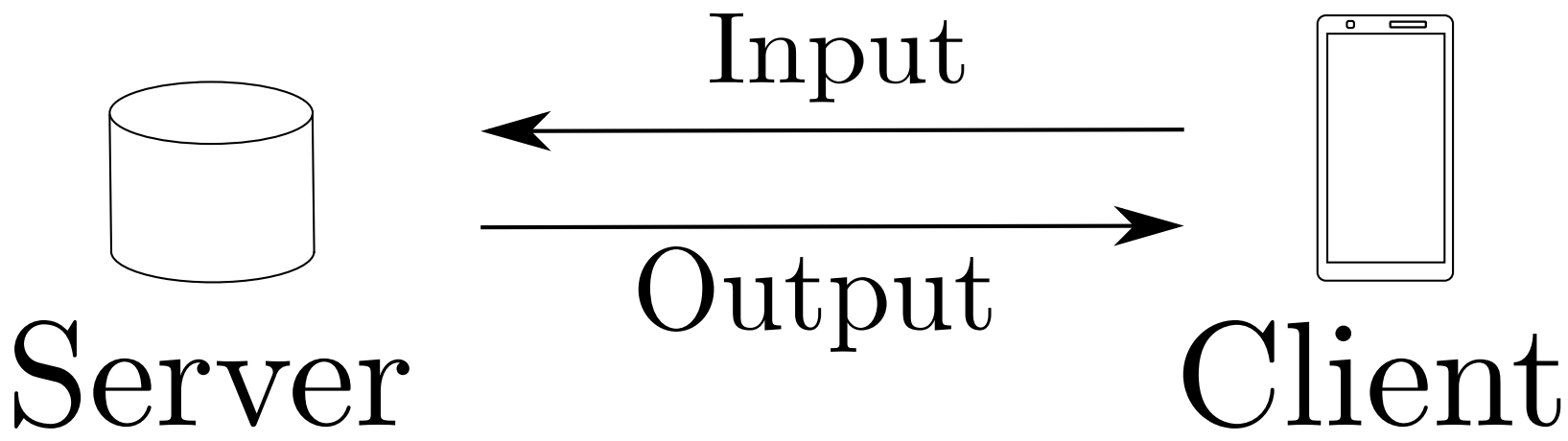
- The client wants to perform heavy computation

Context: outsourced computation



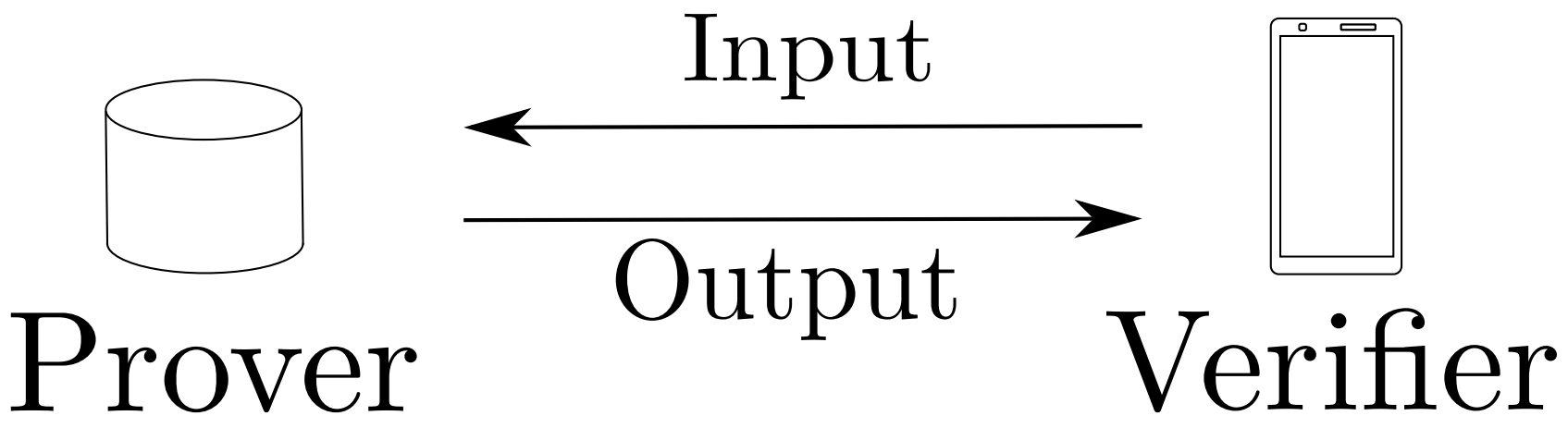
- The client wants to perform heavy computation
- Client and server exchange data

Context: outsourced computation



- The client wants to perform heavy computation
- Client and server exchange data
- How can the client verify the result?

Context: outsourced computation



- The client wants to perform heavy computation
- Client and server exchange data
- How can the client verify the result?

An example: Freivalds' certificate [F79]

Prover		Verifier
	$A, B \in \mathbf{F}^{n \times n}$	
$C = AB$	$\xrightarrow{C}$	Sample a random vector v
		$Cv \stackrel{?}{=} A(Bv)$

- **Probabilistic** check for matrix product
- Complexity measures

Prover	Comm.	Verifier
$O(n^\omega)$	$O(n^2)$	$O(n^2)$

An example: Freivalds' certificate [F79]

Prover		Verifier
	$A, B \in \mathbf{F}^{n \times n}$	
$C = AB$	$\xrightarrow{C}$	Sample a random vector v
		$Cv \stackrel{?}{=} A(Bv)$

- **Probabilistic** check for matrix product
- Complexity measures

Prover	Comm.	Verifier
$O(n^\omega)$	$O(n^2)$	$O(n^2)$

Rank profile matrix

Rank Profiles

Given a matrix A of rank r:

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$

Rank profile matrix

Rank Profiles

Given a matrix A of rank r:

- RRP (Row Rank Profile): first r linearly independant rows

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$

Rank profile matrix

Rank Profiles

Given a matrix A of rank r:

- RRP (Row Rank Profile): first r linearly independant rows
- CRP (Column Rank Profile): first r linearly independant columns

Example

$$A = \begin{pmatrix} 0 & 2 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$

Rank profile matrix

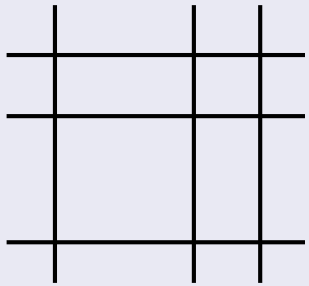
Rank Profiles

Given a matrix A of rank r:

- RRP (Row Rank Profile): first r linearly independant rows
- CRP (Column Rank Profile): first r linearly independant columns

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$



Rank profile matrix

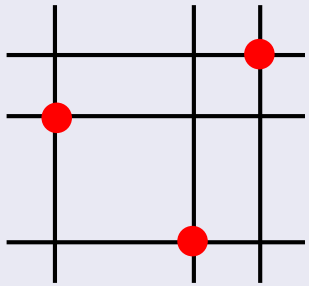
Rank Profiles

Given a matrix A of rank r:

- RRP (Row Rank Profile): first r linearly independant rows
- CRP (Column Rank Profile): first r linearly independant columns

Example

$$A = \begin{pmatrix} 0 & 0 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 5 & 0 & 0 & 0 \end{pmatrix}$$



RPM (Rank Profile Matrix)

The unique $\mathcal{R}_A$ such that any pair of (i, j)-leading sub-matrix of $\mathcal{R}_A$ and of A have the same rank.

Certificates approaches

- Several approaches for certificates
- Generic strategies based on crypto/non-crypto techniques
  - Often have at least a large constant overhead
  - [GKR08], [CFHKKNPZ15], ...

Certificates approaches

- Several approaches for certificates
- Generic strategies based on crypto/non-crypto techniques
  - Often have at least a large constant overhead
  - [GKR08], [CFHKKNPZ15], ...
- A strategy for linear algebra algorithms reducing to matrix multiplication
  - Soft factor on communication space and verifier time
  - [KNS11]

Certificates approaches

- Several approaches for certificates
- Generic strategies based on crypto/non-crypto techniques
  - Often have at least a large constant overhead
  - [GKR08], [CFHKKNPZ15], ...
- A strategy for linear algebra algorithms reducing to matrix multiplication
  - Soft factor on communication space and verifier time
  - [KNS11]
- Algorithm specific approach
  - Optimize global cost

Comparison with [KNS11] approach

Certificate	Algorithm	Prover time	Comm.	Verifier time
CRP/RRP/RPM	[KNS11]+[DPS16]	Best known	$\widetilde{O}(mn)$	$\widetilde{O}(mn)$

- $A \in \mathbf{F}^{m \times n}$ of rank r
- Examples of possible Provers :
  - $O(mnr^{\omega-2})$ (deterministic algorithm)
  - $\widetilde{O}(r^{\omega} + m + n + o(1))$ (probabilistic algorithm)

Comparison with [KNS11] approach

Certificate	Algorithm	Prover time	Comm.	Verifier time
CRP/RRP/RPM	[KNS11]+[DPS16]	Best known	$\widetilde{O}(mn)$	$\widetilde{O}(mn)$
CRP/RRP/RPM	This paper	Best known	$O(m+n)$	$O(mn)$

- $A \in \mathbf{F}^{m \times n}$ of rank r
- Examples of possible Provers :
  - $O(mnr^{\omega-2})$ (deterministic algorithm)
  - $\widetilde{O}(r^{\omega}+m+n+o(1))$ (probabilistic algorithm)

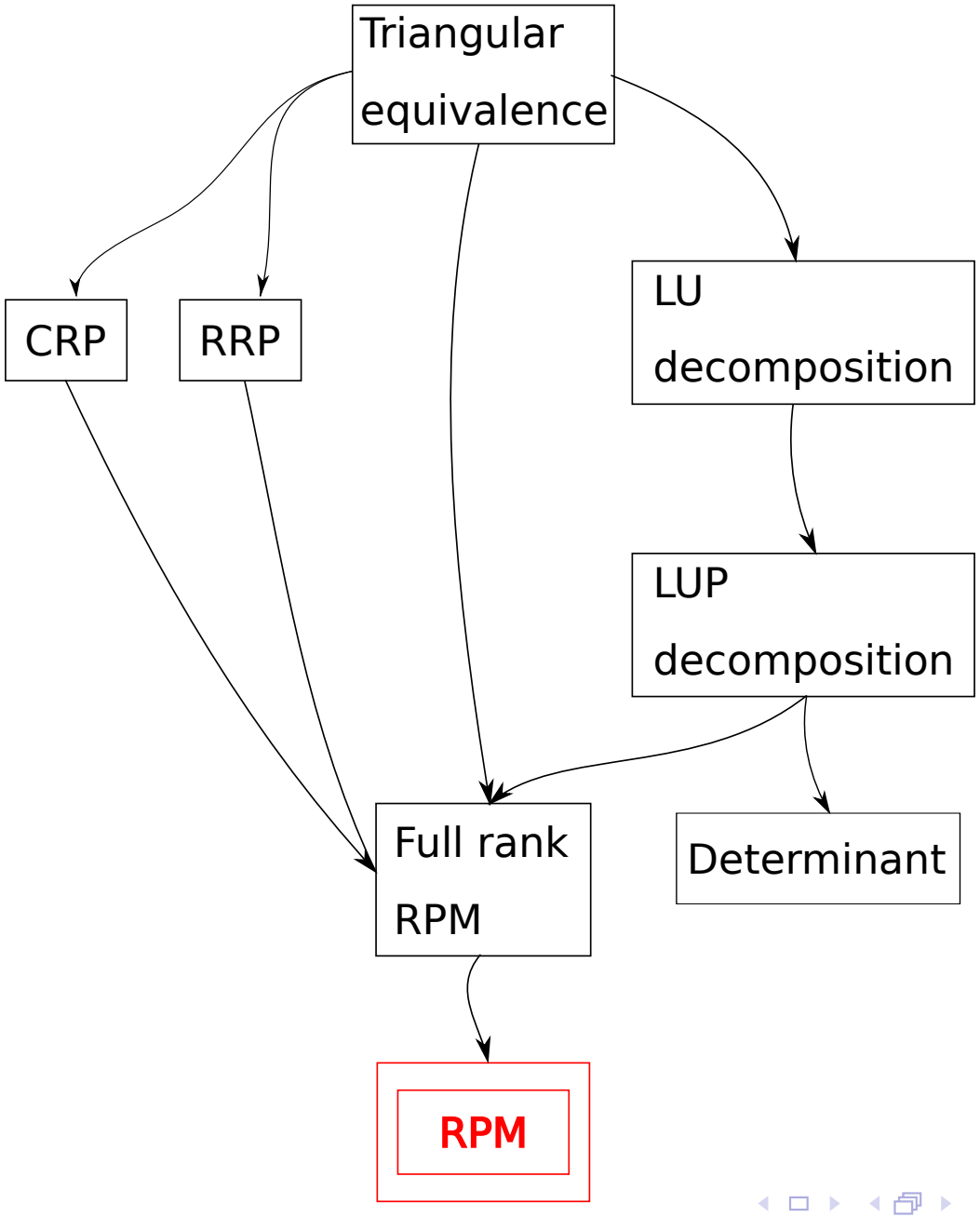
Plan

1 Introduction

2 Our certificates

3 Conclusion

Roadmap


Triangular equivalence
CRP
RRP
LU decomposition
LUP decomposition
Full rank RPM
Determinant
RPM

Certificate for the rank profile matrix

- $\widetilde{O}(r^\omega + m + n + o(1))$ probabilistic algorithm

---

Rank Profile Matrix [DPS16]

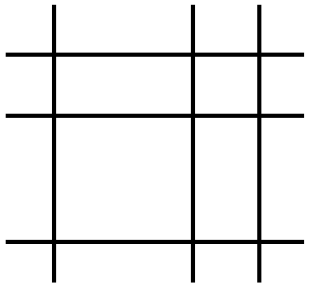
1: Compute the Row Rank Profile of A
2: Compute the Column Rank Profile of A

Certificate for the rank profile matrix

- $\widetilde{O}(r^\omega + m + n + o(1))$ probabilistic algorithm

---

Rank Profile Matrix [DPS16]

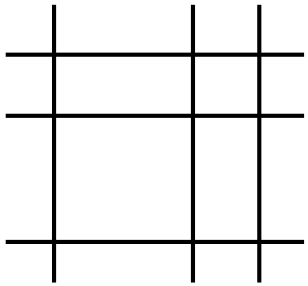
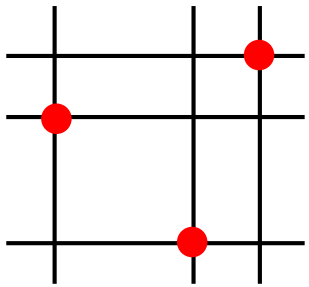
1: Compute the Row Rank Profile of A
2: Compute the Column Rank Profile of A
3: Extract the submatrix 

Certificate for the rank profile matrix

- $\widetilde{O}(r^\omega + m + n + o(1))$ probabilistic algorithm

---

Rank Profile Matrix [DPS16]

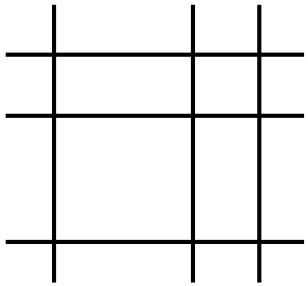
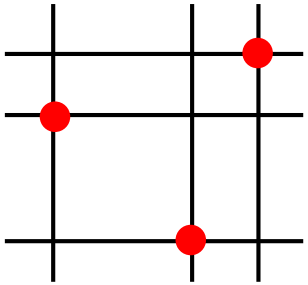
1: Compute the Row Rank Profile of A
2: Compute the Column Rank Profile of A
3: Extract the submatrix 
4: Compute the Rank Profile Matrix of the submatrix of A 

Certificate for the rank profile matrix

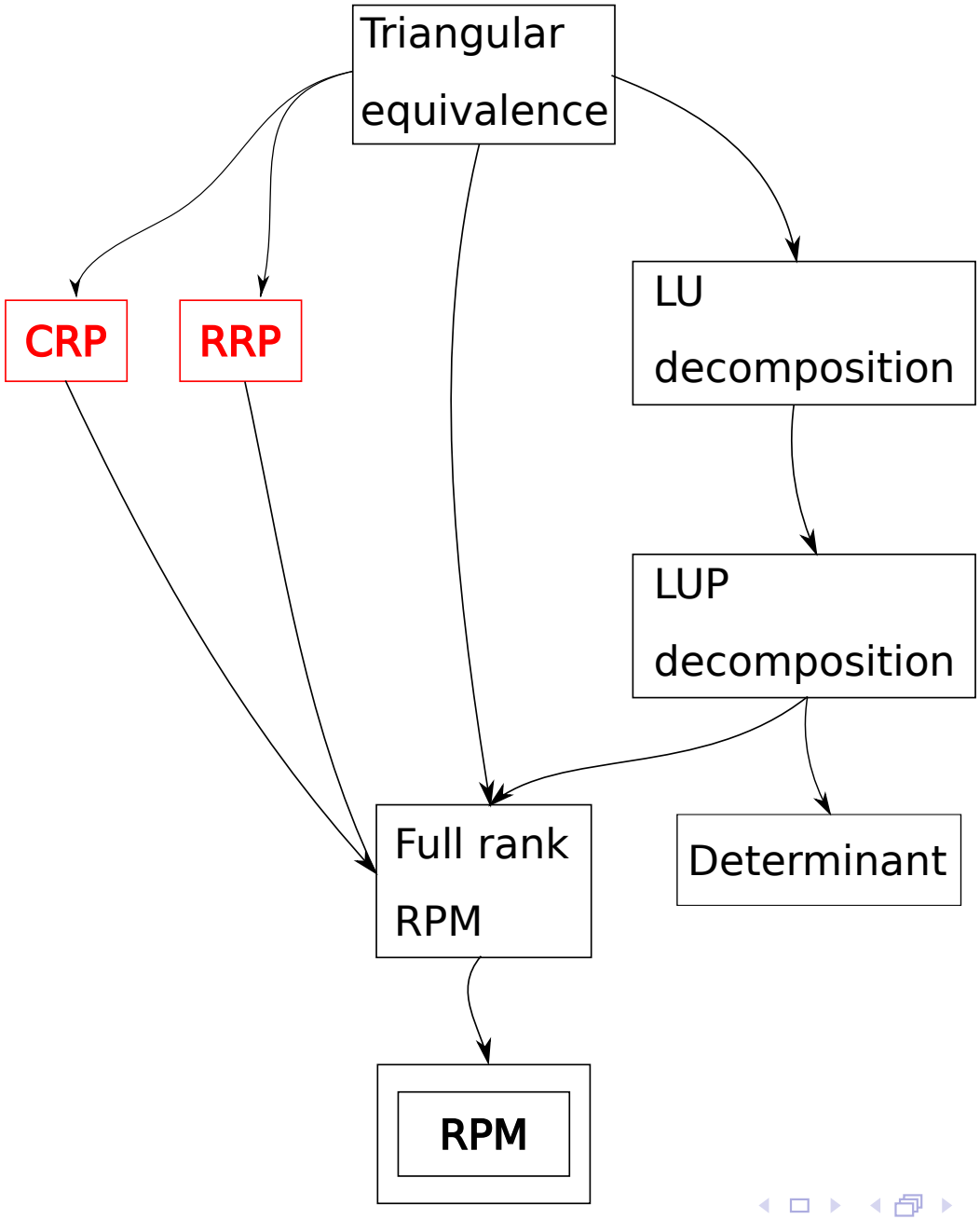
- $\widetilde{O}(r^\omega + m + n + o(1))$ probabilistic algorithm

---

Rank Profile Matrix [DPS16]

1: Verify the Row Rank Profile of A
2: Verify the Column Rank Profile of A
3: Extract the submatrix 
4: Verify the Rank Profile Matrix of the submatrix of A 

Roadmap

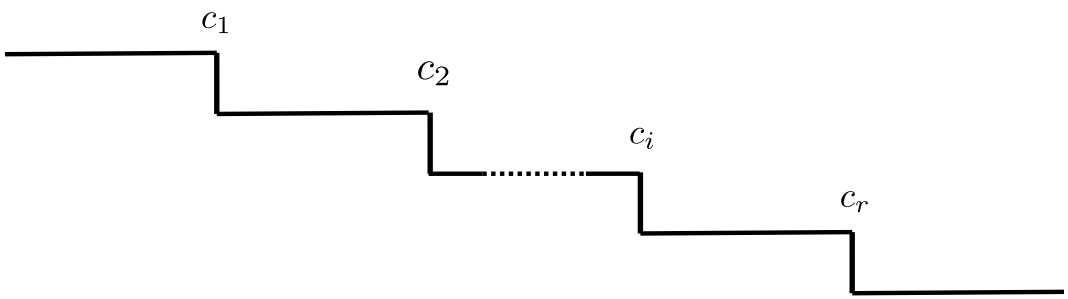

Triangular equivalence
CRP
RRP
LU decomposition
LUP decomposition
Full rank RPM
Determinant
RPM

Certificates for the column rank profile

- [DK14], [E15]: certificates for a bound on the rank:
- comm. $O(n)$
- Use it to verify the column rank profile?

$$\text{Prover: } CRP = \{c_1, \ldots, c_r\}$$

$$\text{Lin. independence: } rk(A) \overset{?}{\geq} r$$

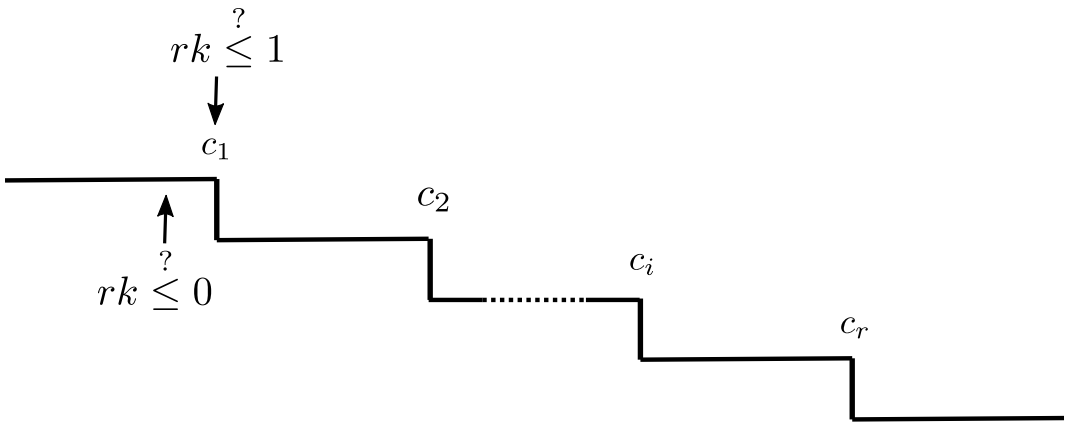


Certificates for the column rank profile

- [DK14], [E15]: certificates for a bound on the rank:
- comm. $O(n)$
- Use it to verify the column rank profile?

$$\text{Prover: } CRP = \{c_1, \ldots, c_r\}$$

$$\text{Lin. independence: } rk(A) \overset{?}{\geq} r$$

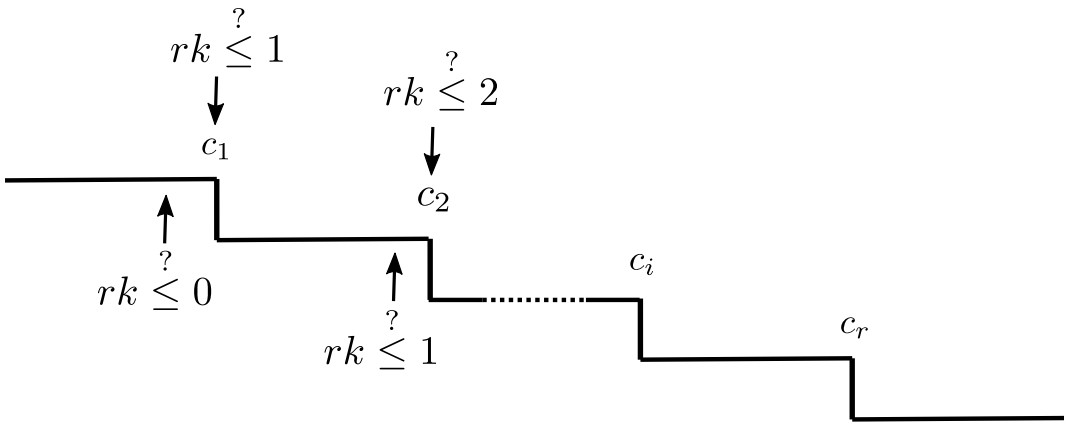


Certificates for the column rank profile

- [DK14], [E15]: certificates for a bound on the rank:
- comm. $O(n)$
- Use it to verify the column rank profile?

Prover: $CRP = \{c_1, \ldots, c_r\}$

Lin. independence: $rk(A) \overset{?}{\geq} r$

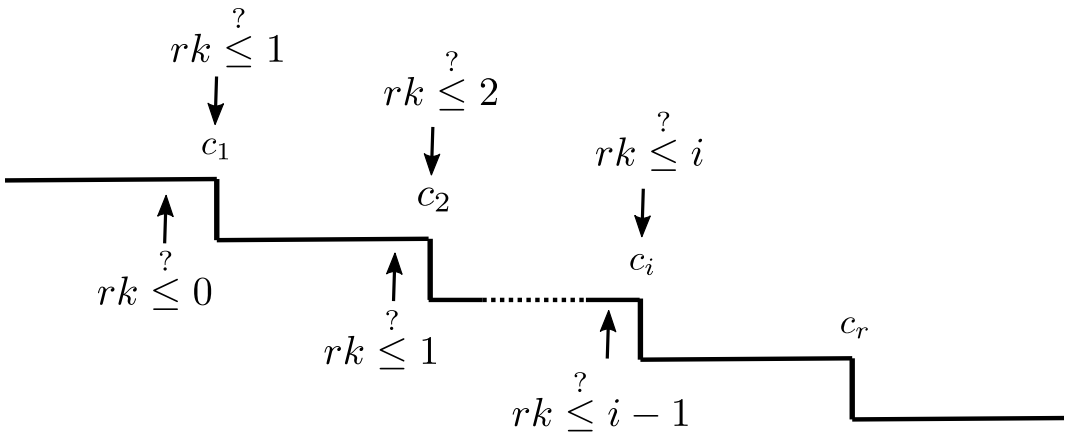


Certificates for the column rank profile

- [DK14], [E15]: certificates for a bound on the rank:
- comm. $O(n)$
- Use it to verify the column rank profile?

Prover: $CRP = \{c_1, \ldots, c_r\}$

Lin. independence: $rk(A) \overset{?}{\geq} r$

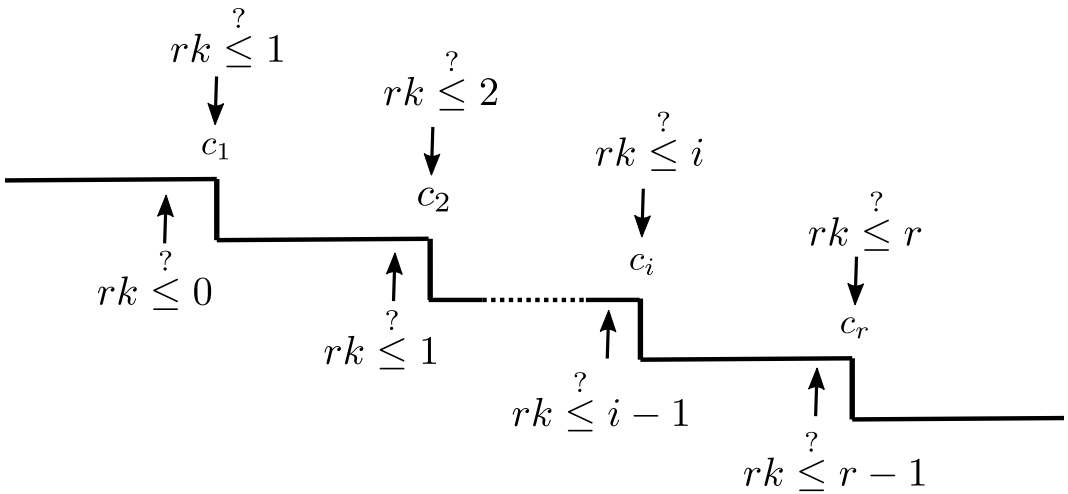


Certificates for the column rank profile

- [DK14], [E15]: certificates for a bound on the rank:
- comm. $O(n)$
- Use it to verify the column rank profile?
- Problem: $2rO(n)$ comm.

$$\text{Prover: } CRP = \{c_1, \ldots, c_r\}$$

$$\text{Lin. independence: } rk(A) \overset{?}{\geq} r$$



Certificates for the column rank profile

- A has rank r
- v is a random vector
- $\exists \gamma$ of size r s.t. $A_{*,c_1,\ldots,c_r}\gamma = Av$

Certificates for the column rank profile

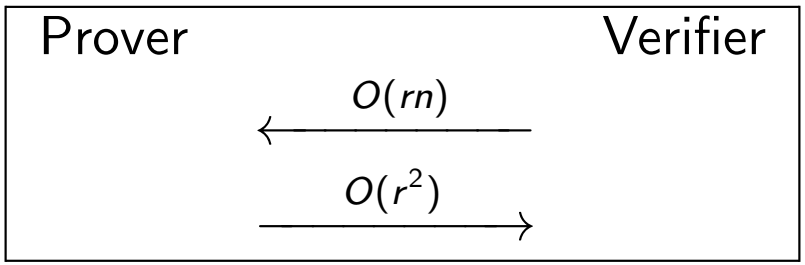
- A has rank r
- v is a random vector
- $\exists \gamma$ of size r s.t. $A_{*,c_1,\ldots,c_r}\gamma = Av$
- Previous operations can be rewritten as follows

$$A_{*,c_1,\ldots,c_r} \underbrace{\begin{bmatrix} \gamma_{11} & \gamma_{21} & \cdots & \gamma_{r1} \\ 0 & \gamma_{22} & & \gamma_{r2} \\ 0 & 0 & & \gamma_{r3} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \gamma_{rr} \end{bmatrix}}_{\Gamma} = A \begin{bmatrix} v_1 & w_1 & \cdots & \cdots & z_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{c_1-1} & \vdots & \vdots & \vdots & \vdots \\ 0 & w_{c_2-1} & \vdots & \vdots & \vdots \\ 0 & 0 & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & x_{c_r-1} & \vdots \\ 0 & 0 & 0 & 0 & z_n \end{bmatrix}$$

Certificates for the column rank profile

- A has rank r
- v is a random vector
- $\exists \gamma$ of size r s.t. $A_{*,c_1,\ldots,c_r}\gamma = Av$
- Previous operations can be rewritten as follows
- Still too much comm. space

$$A_{*,c_1,\ldots,c_r} \underbrace{\begin{bmatrix} \gamma_{11} & \gamma_{21} & \cdots & \gamma_{r1} \\ 0 & \gamma_{22} & & \gamma_{r2} \\ 0 & 0 & & \gamma_{r3} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \gamma_{rr} \end{bmatrix}}_{\Gamma} = A \begin{bmatrix} v_1 & w_1 & \cdots & \cdots & z_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{c_1-1} & \vdots & \vdots & \vdots & \vdots \\ 0 & w_{c_2-1} & \vdots & \vdots & \vdots \\ 0 & 0 & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & x_{c_r-1} & \vdots \\ 0 & 0 & 0 & 0 & z_n \end{bmatrix}$$



Certificates for the column rank profile

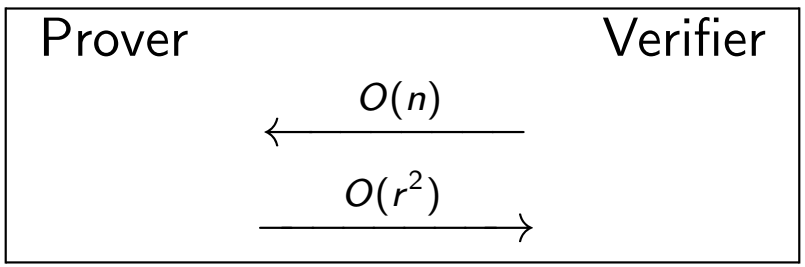
- Use vectors with the same prefix for all challenges

$$A_{*,c_1,\ldots,c_r} \underbrace{\begin{bmatrix} \gamma_{11} & \gamma_{21} & \cdots & \gamma_{r1} \\ 0 & \gamma_{22} & & \gamma_{r2} \\ 0 & 0 & & \gamma_{r3} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \gamma_{rr} \end{bmatrix}}_{\Gamma} = A \begin{bmatrix} v_1 & v_1 & \cdots & \cdots & v_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{c_1-1} & \vdots & \vdots & \vdots & \vdots \\ 0 & v_{c_2-1} & \vdots & \vdots & \vdots \\ 0 & 0 & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & v_{c_r-1} & \vdots \\ 0 & 0 & 0 & 0 & v_n \end{bmatrix}$$

Certificates for the column rank profile

- Use vectors with the same prefix for all challenges

$$A_{*,c_1,\ldots,c_r} \underbrace{\begin{bmatrix} \gamma_{11} & \gamma_{21} & \cdots & \gamma_{r1} \\ 0 & \gamma_{22} & & \gamma_{r2} \\ 0 & 0 & & \gamma_{r3} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \gamma_{rr} \end{bmatrix}}_{\Gamma} = A \begin{bmatrix} v_1 & v_1 & \cdots & \cdots & v_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{c_1-1} & \vdots & \vdots & \vdots & \vdots \\ 0 & v_{c_2-1} & \vdots & \vdots & \vdots \\ 0 & 0 & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & v_{c_r-1} & \vdots \\ 0 & 0 & 0 & 0 & v_n \end{bmatrix}$$

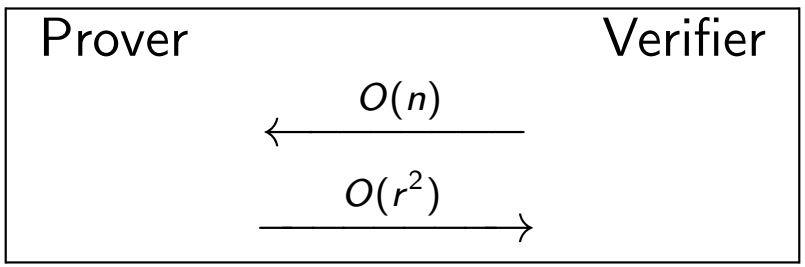


Certificates for the column rank profile

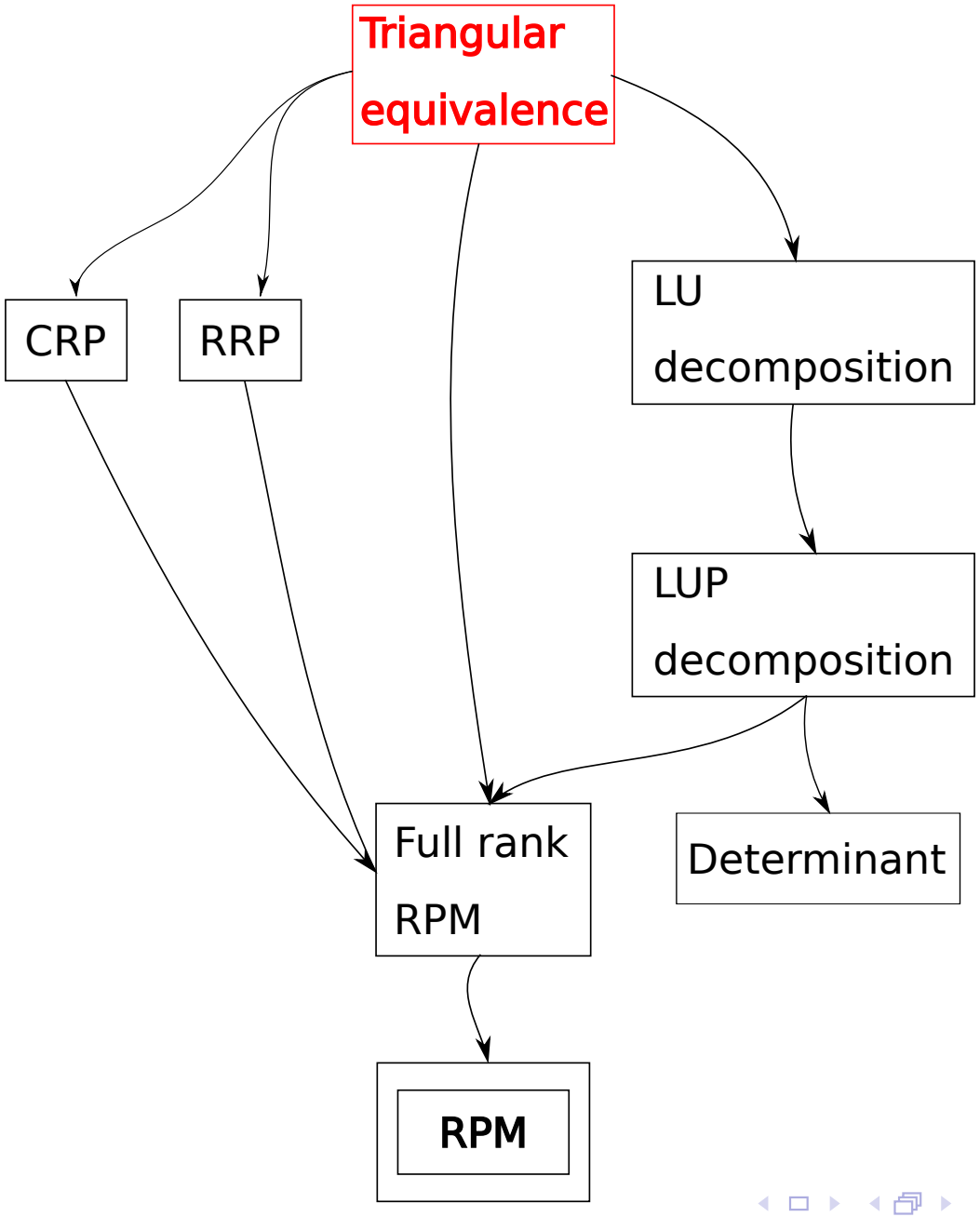
- Use vectors with the same prefix for all challenges
- Still quadratic for comm. space...
- Verify a projection of Γ

$\rightarrow$ Triangular equivalence

$$A_{*,c_1,\ldots,c_r} \underbrace{\begin{bmatrix} \gamma_{11} & \gamma_{21} & \cdots & \gamma_{r1} \\ 0 & \gamma_{22} & & \gamma_{r2} \\ 0 & 0 & & \gamma_{r3} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \gamma_{rr} \end{bmatrix}}_{\Gamma} = A \begin{bmatrix} v_1 & v_1 & \cdots & \cdots & v_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{c_1-1} & \vdots & \vdots & \vdots & \vdots \\ 0 & v_{c_2-1} & \vdots & \vdots & \vdots \\ 0 & 0 & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & v_{c_r-1} & \vdots \\ 0 & 0 & 0 & 0 & v_n \end{bmatrix}$$

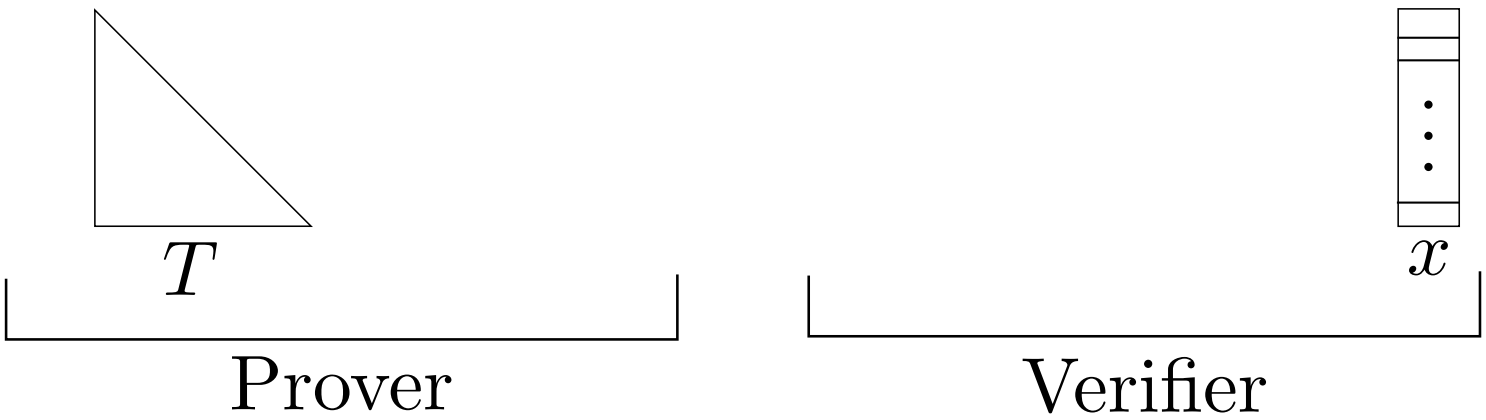


Roadmap


Triangular equivalence
CRP
RRP
LU decomposition
LUP decomposition
Full rank RPM
Determinant
RPM

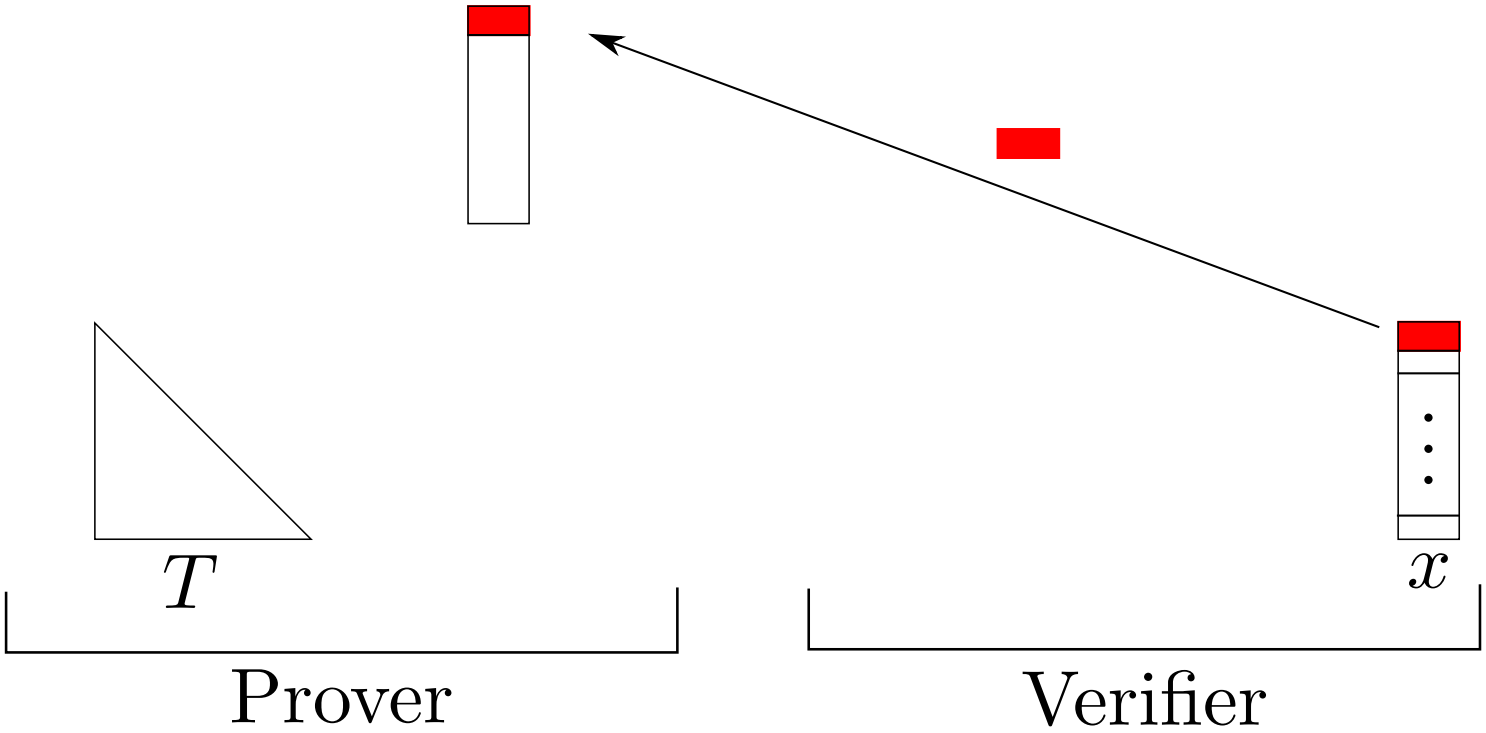
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



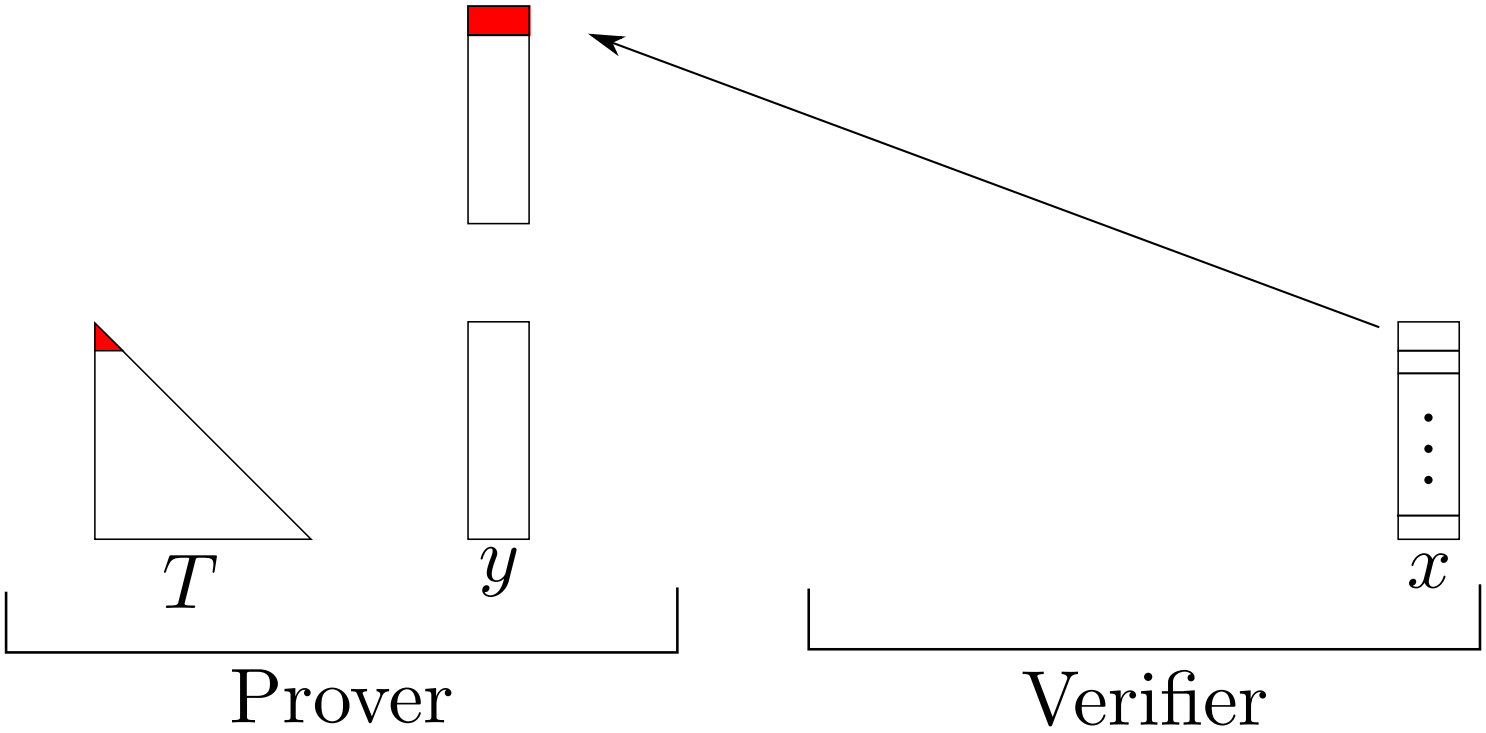
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



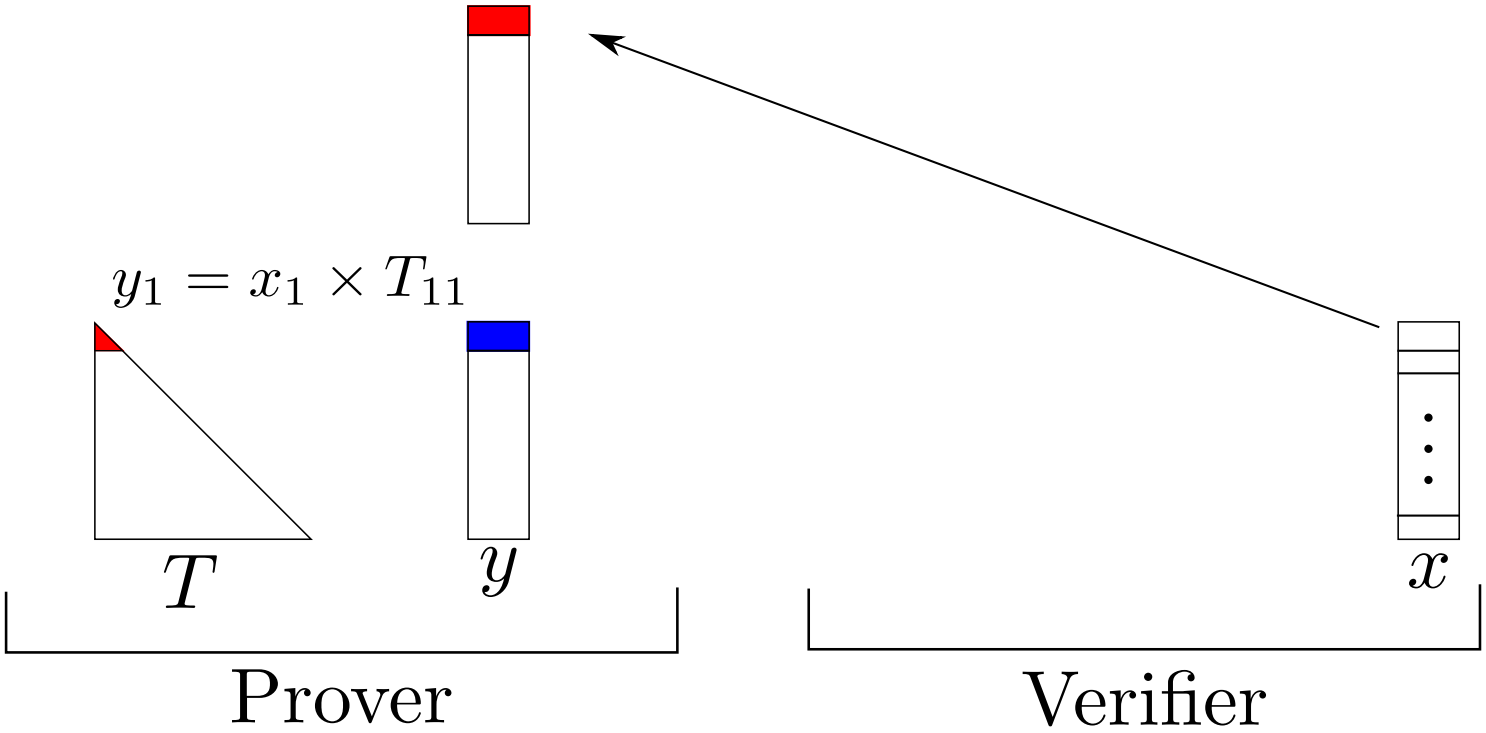
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



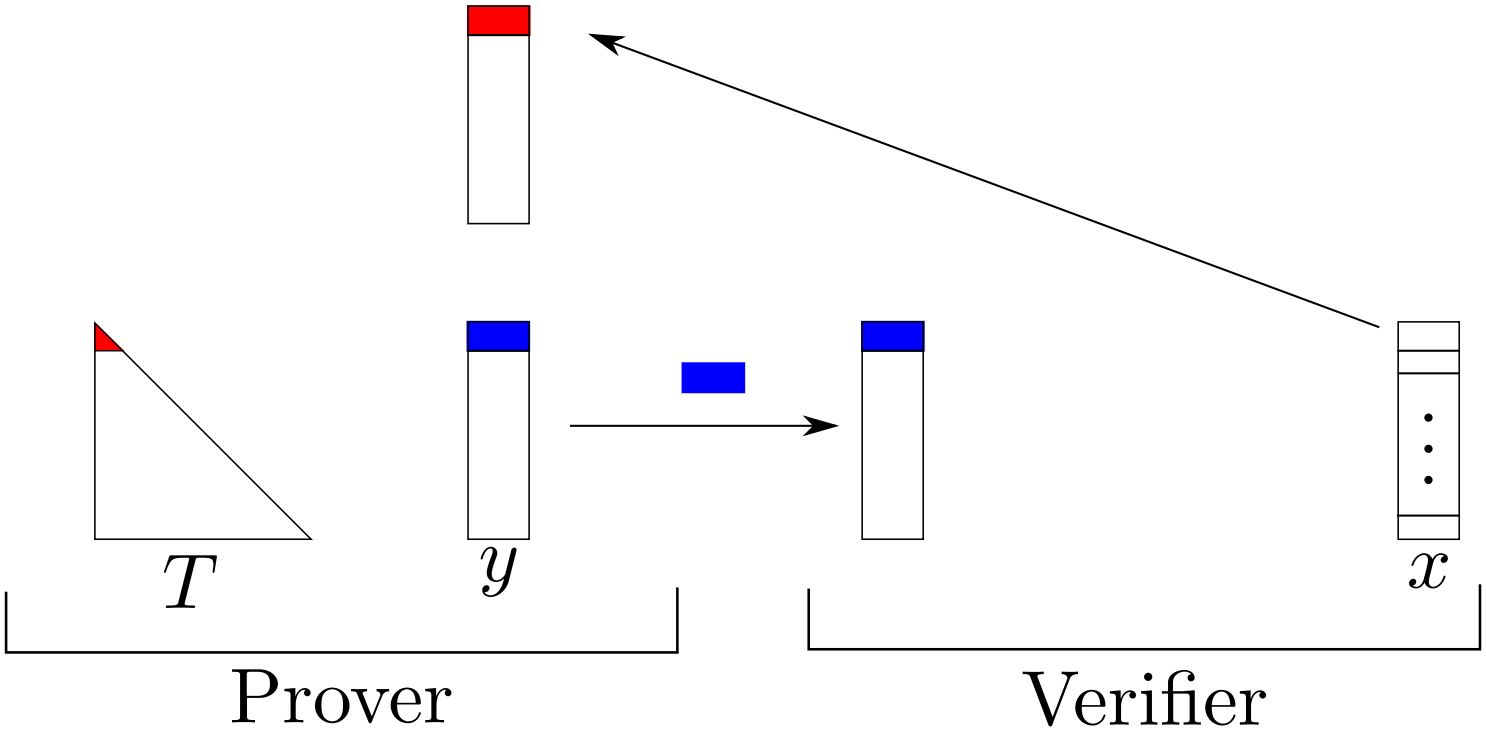
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



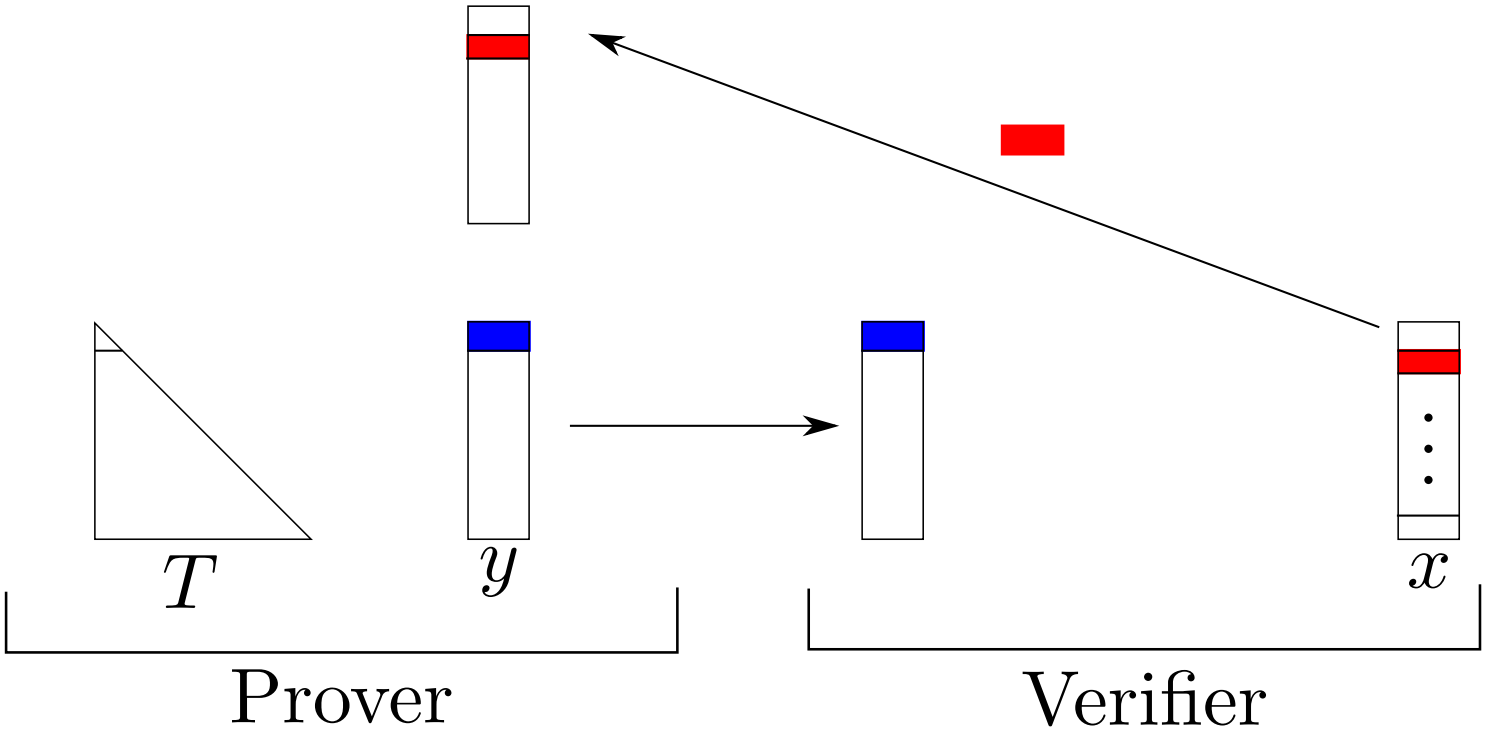
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



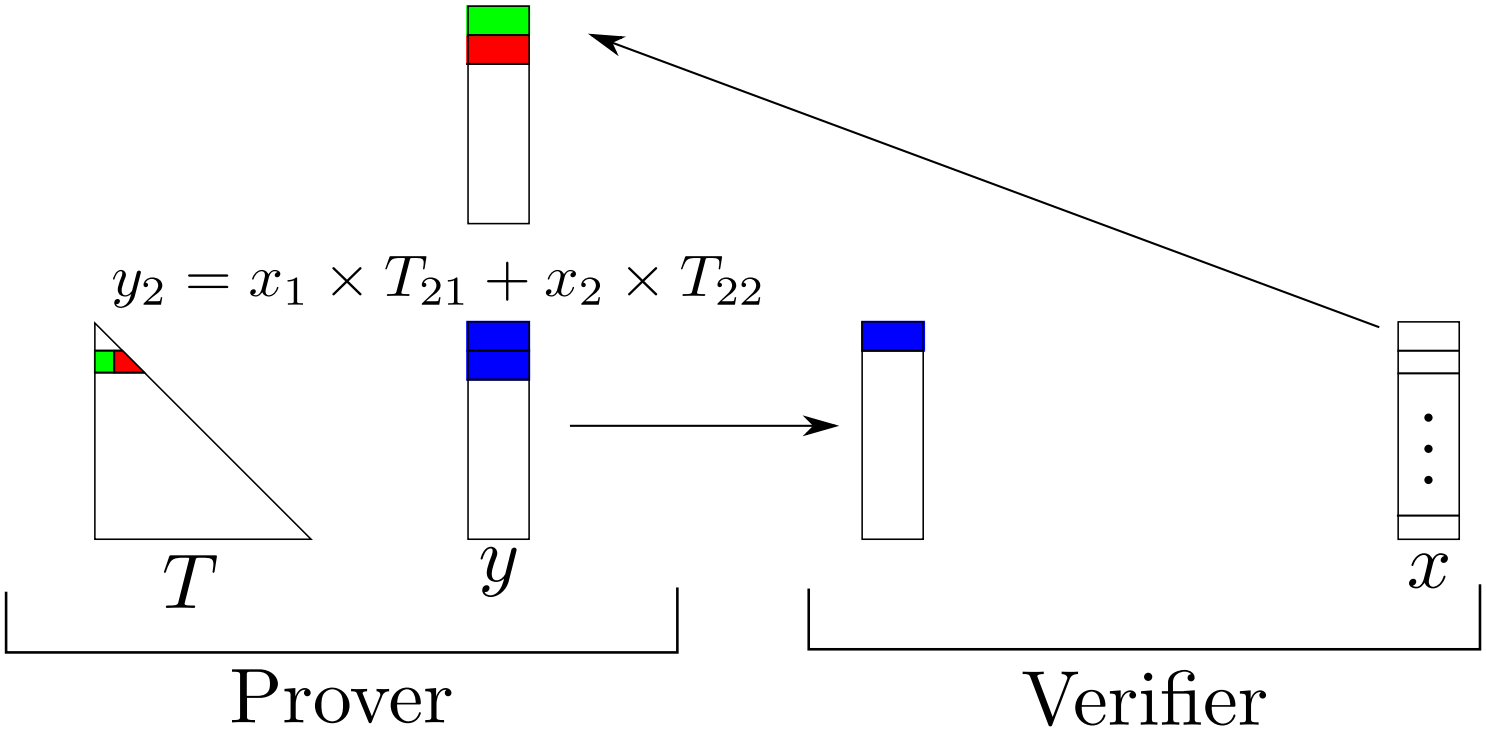
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



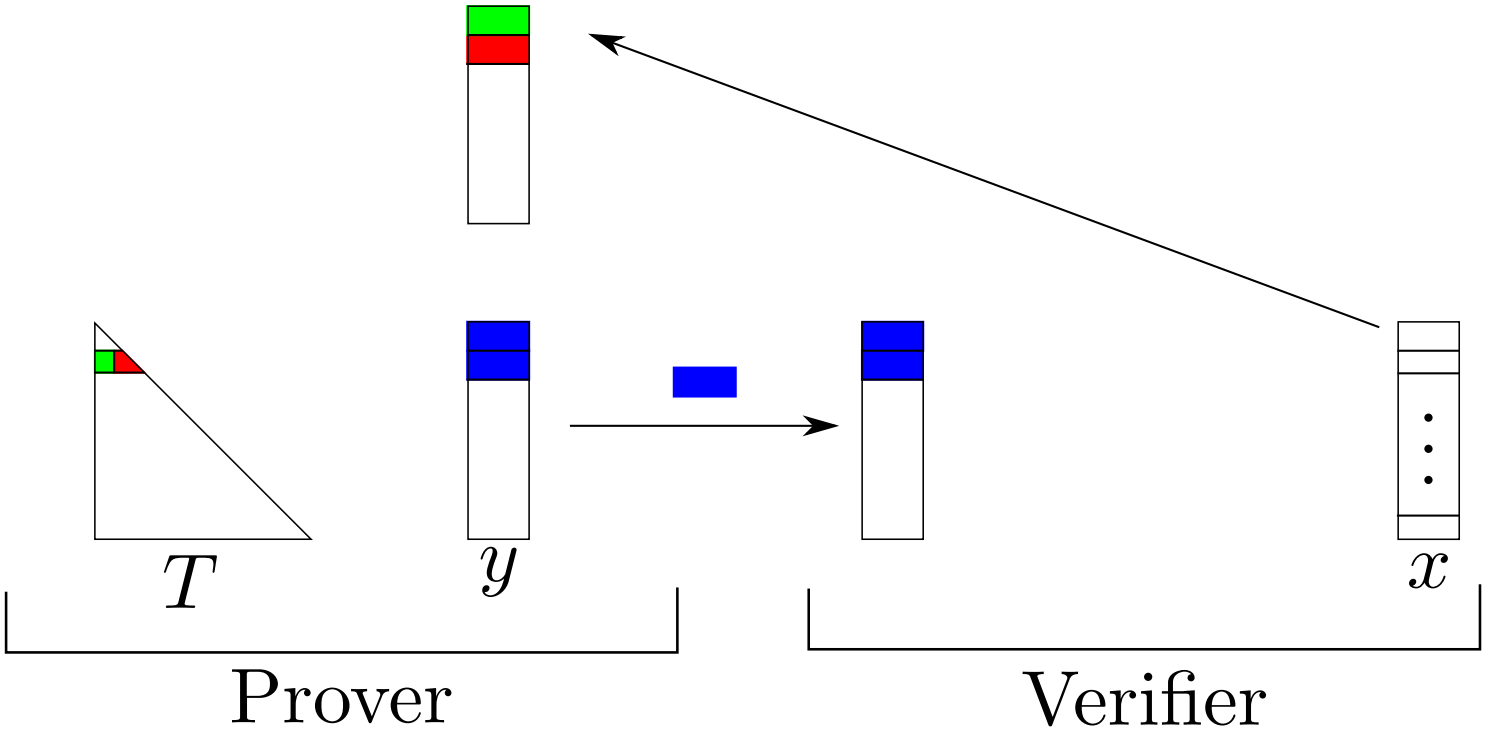
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



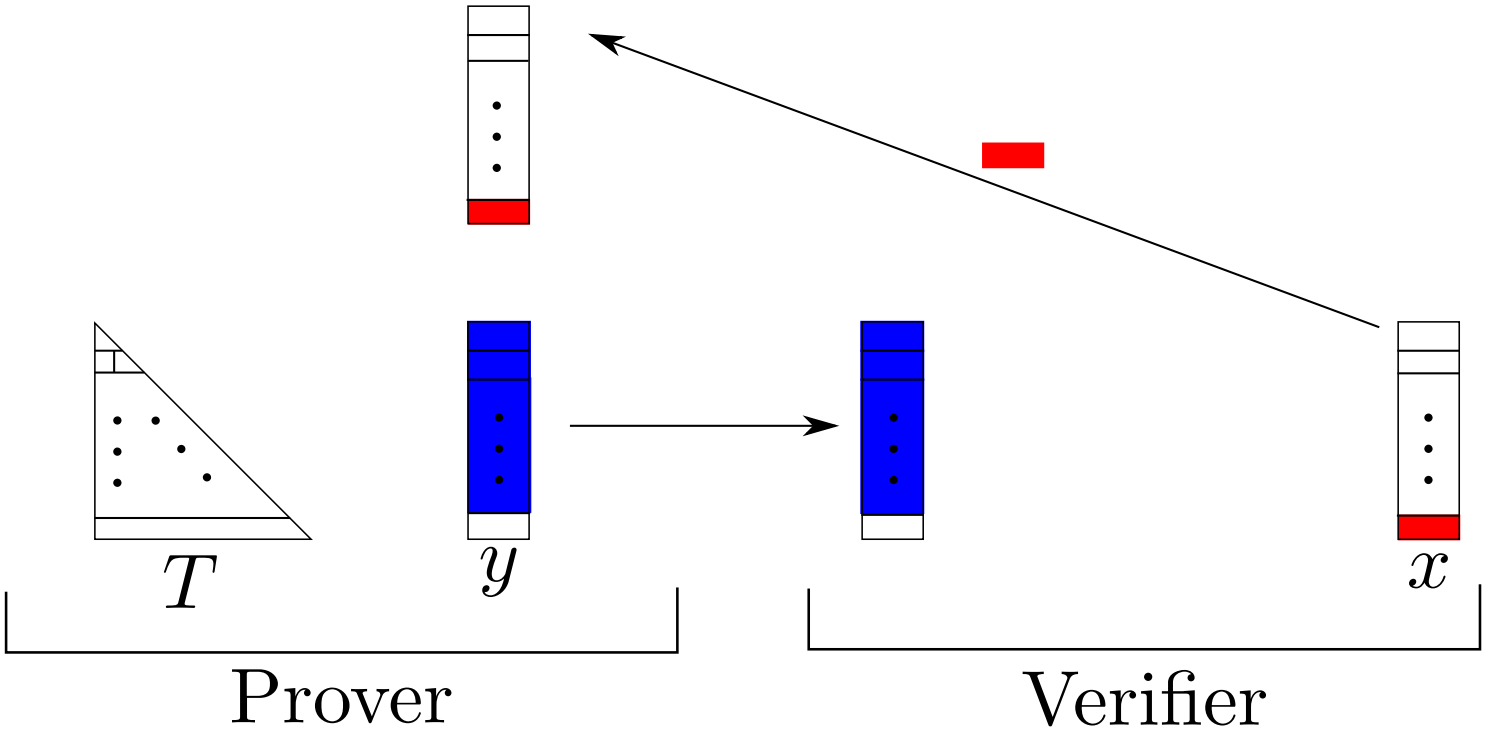
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



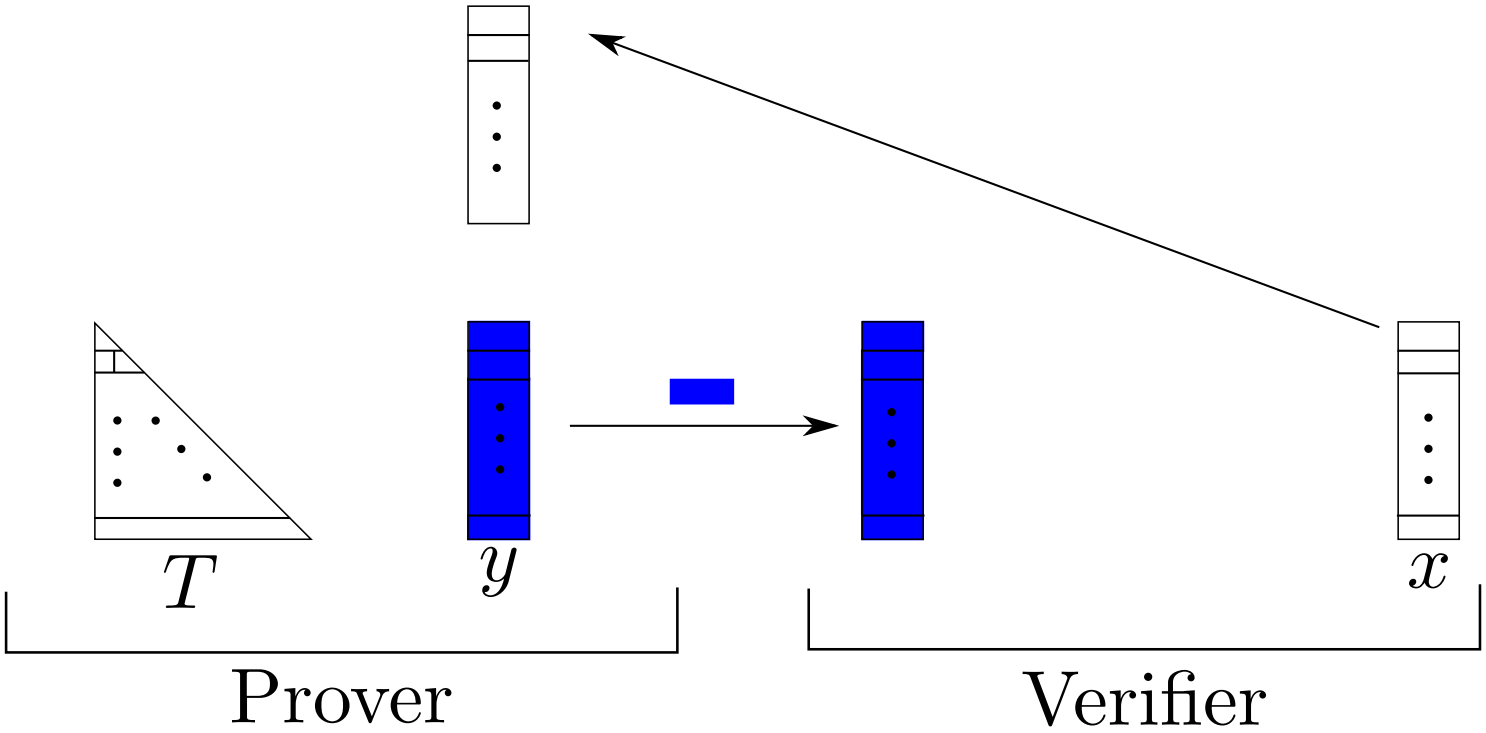
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



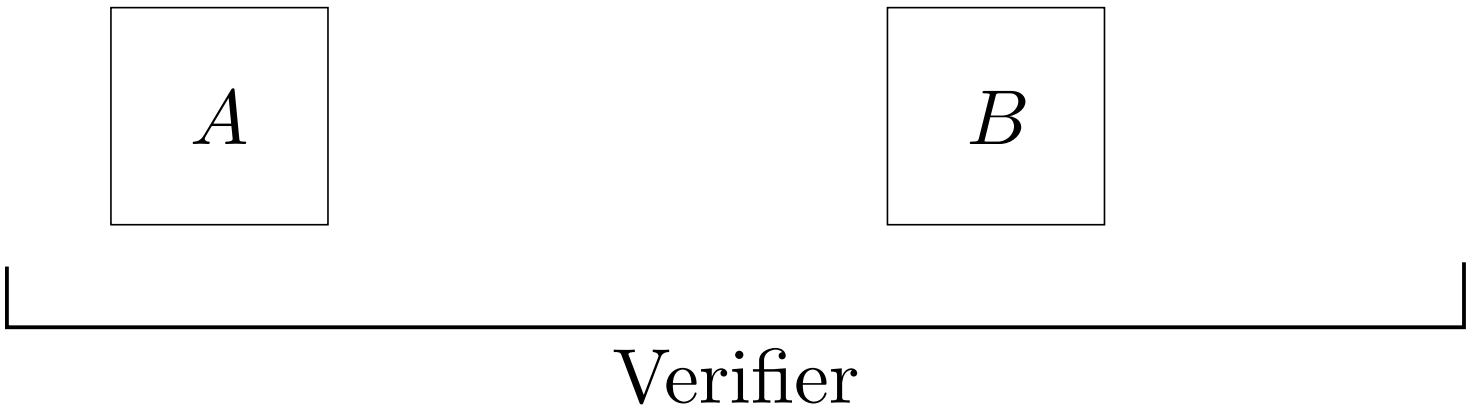
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



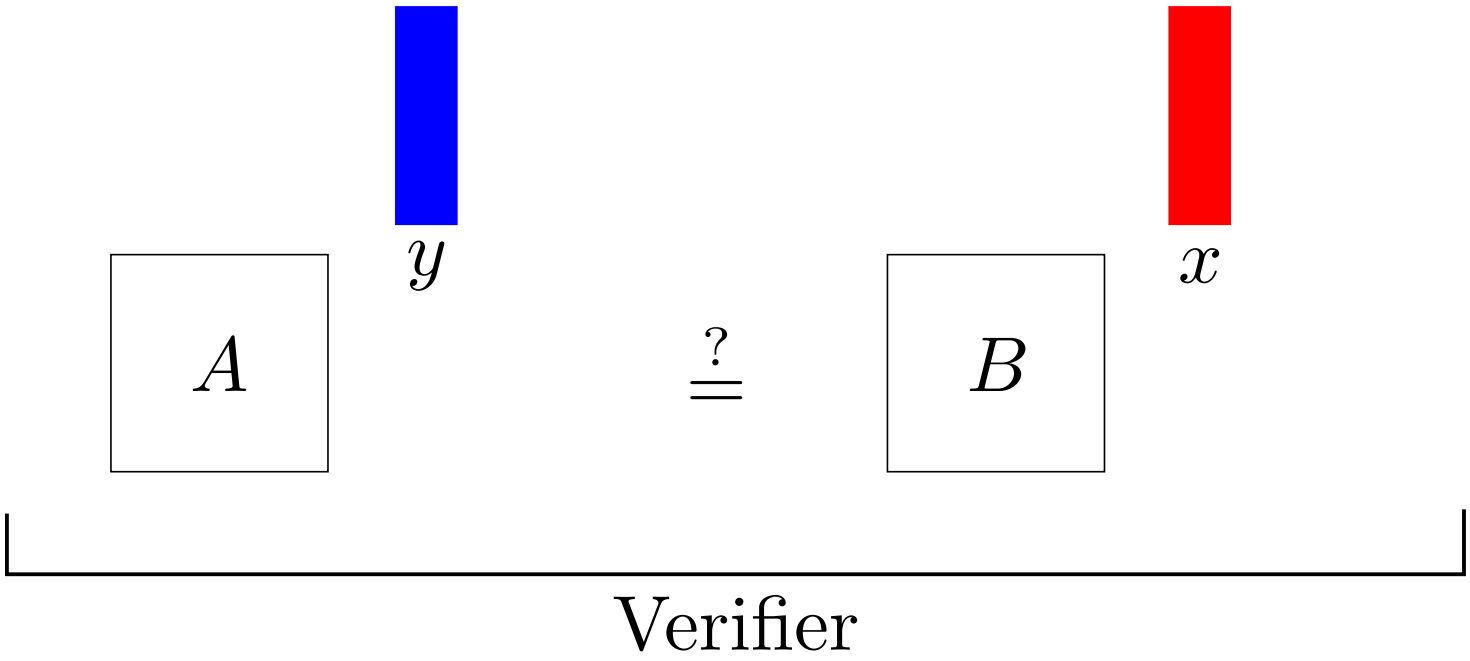
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



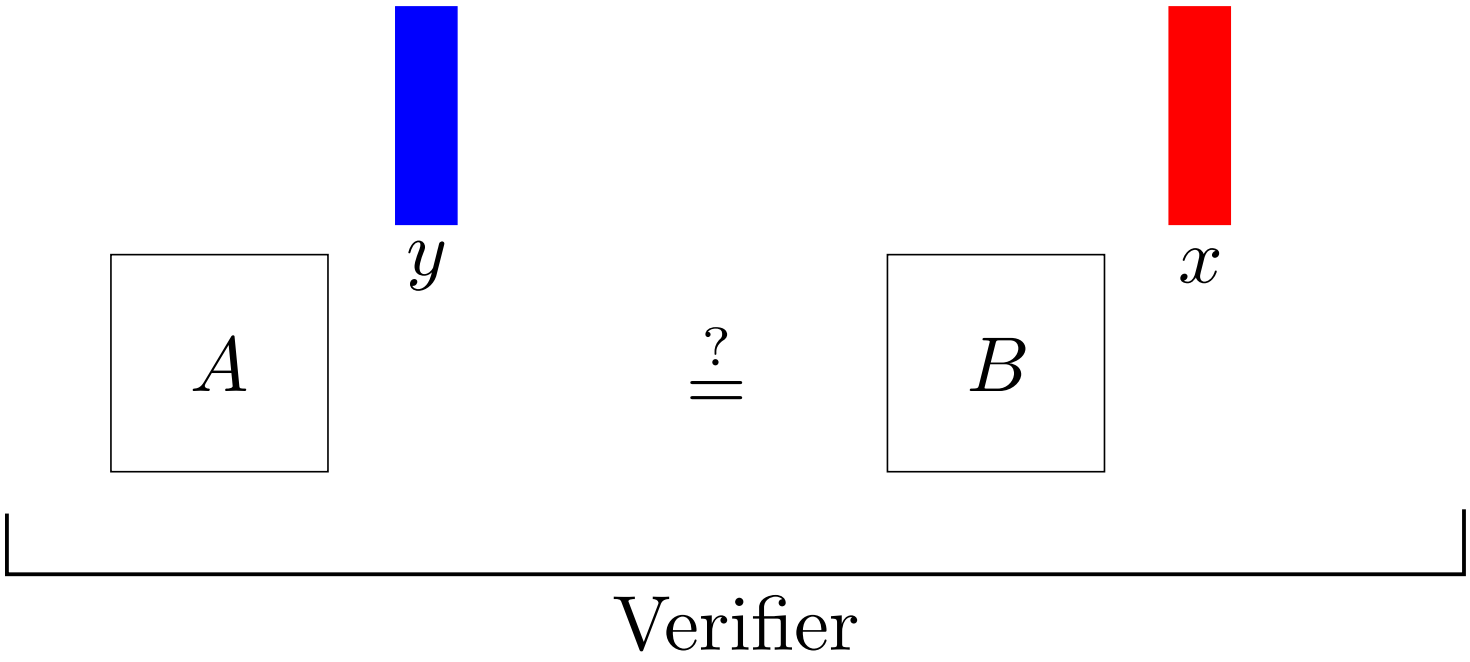
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



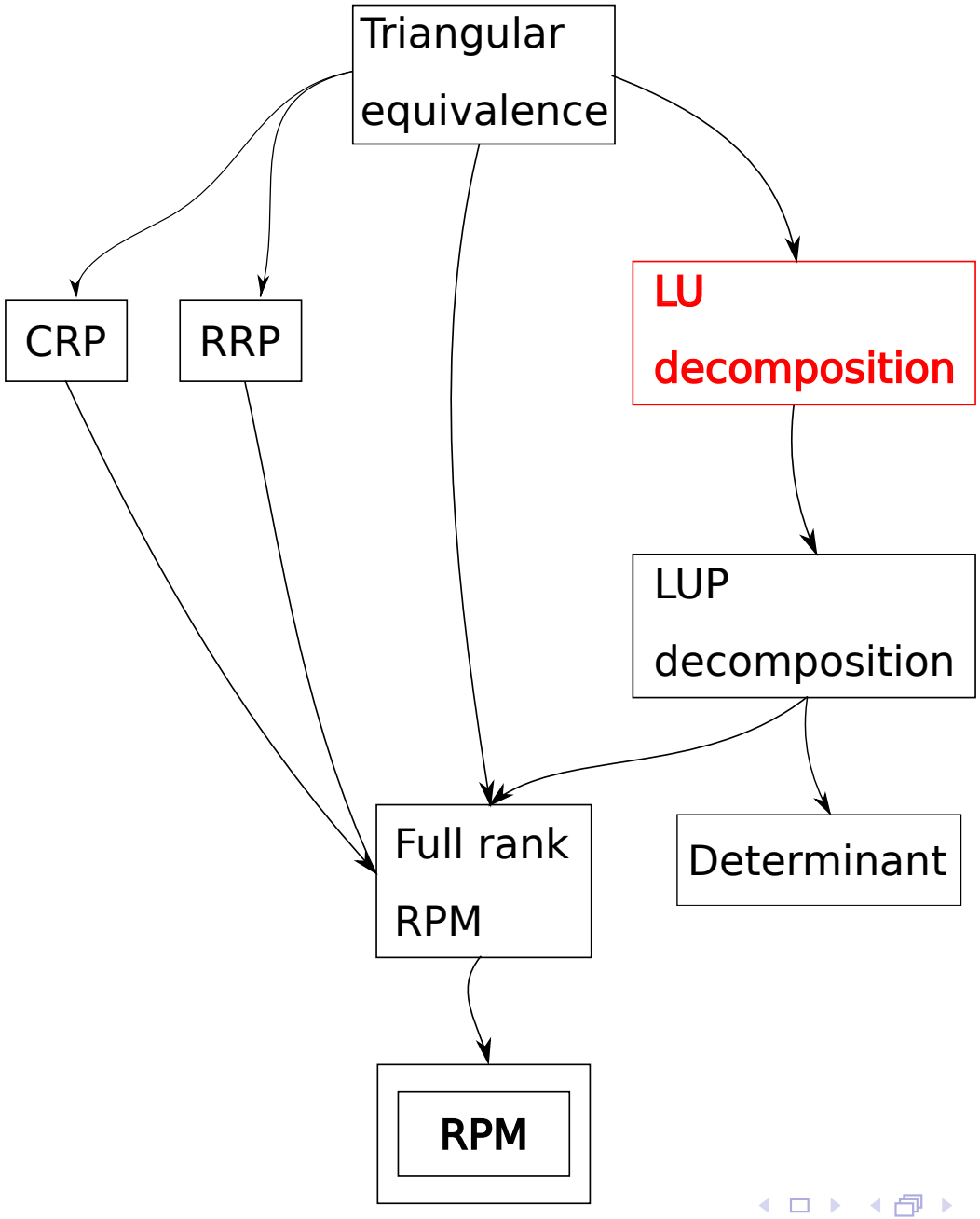
Triangular equivalence

Pb: Verify $\exists T$ s.t. $AT = B$ without sending T



Prover	Comm.	Verifier
$O(mn^{\omega-1})$	$2n$	$O(n^2)$

Roadmap


Triangular equivalence
CRP
RRP
LU decomposition
LUP decomposition
Full rank RPM
Determinant
RPM

Certificate for LU decomposition

- Certificate for the **existence** of $A = LU$
- L and U are not transmitted
- Use the triangular equivalence scheme

Certificate for LU decomposition

- Certificate for the **existence** of $A = LU$
- L and U are not transmitted
- Use the triangular equivalence scheme
- Certify with:
  - a right projection $x = U\phi$, and
  - a left projection $z^T = \lambda^T L$
  - and $\lambda^T A \phi = \underbrace{\lambda^T L}_{z^T} \underbrace{U\phi}_{x}$

Certificate for LU decomposition

- Certificate for the **existence** of $A = LU$
- L and U are not transmitted
- Use the triangular equivalence scheme
- Certify with:
  - a right projection $x = U\phi$, and
  - a left projection $z^T = \lambda^T L$
  - and $\lambda^T A \phi = \underbrace{\lambda^T L}_{z^T} \underbrace{U\phi}_{x}$
- In that case, not enough!
- Not enough constraints

$\rightarrow$ Use two right projections instead

Certificate for LU decomposition

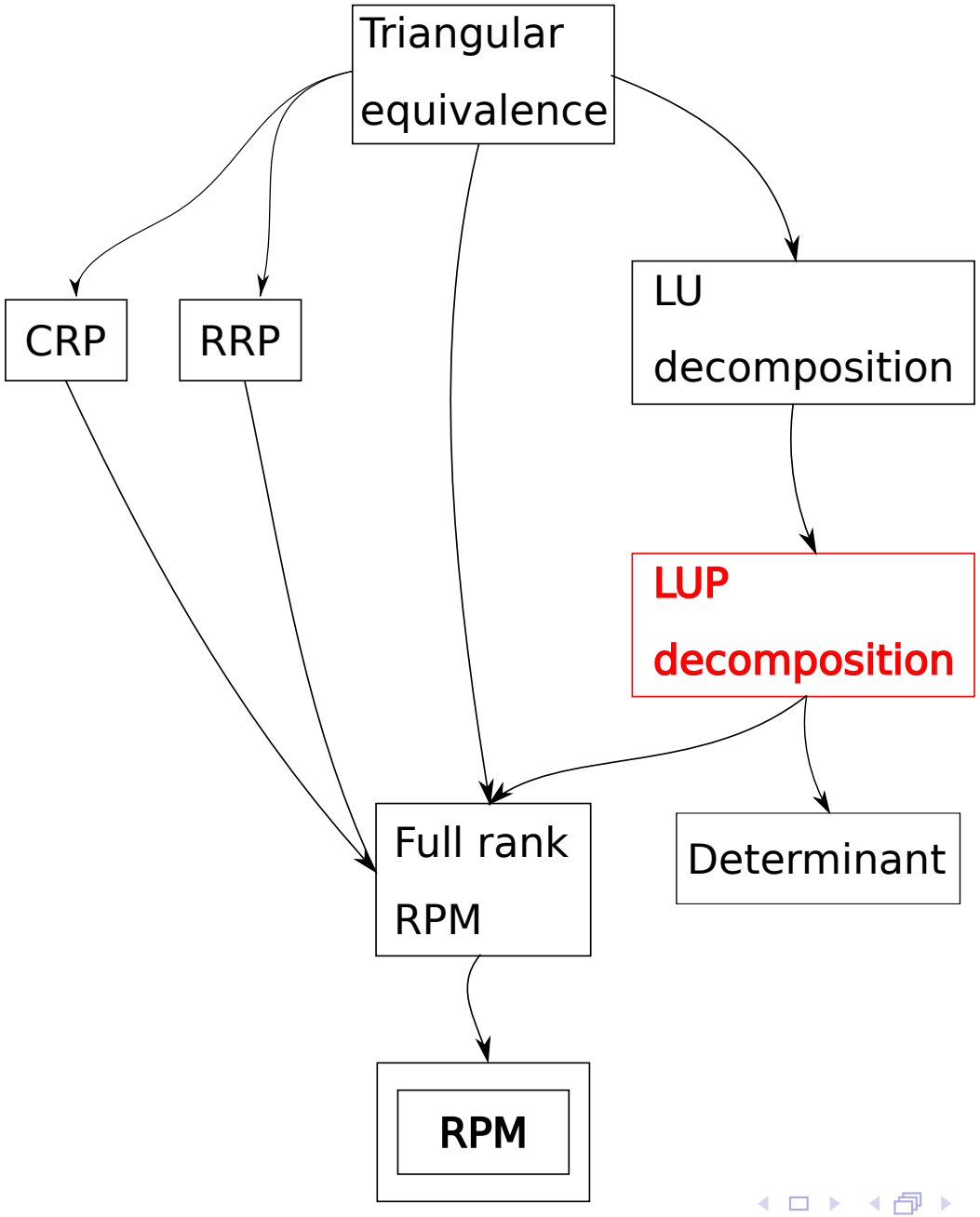
Prover		Verifier
	$A \in \mathbf{F}^{n \times n}$ non-singular	
$A = LU$	$\xrightarrow{A = LU}$	
	for i from n downto 1	
$\begin{bmatrix} x & y \end{bmatrix} = U \begin{bmatrix} \phi & \psi \end{bmatrix}$	$\xleftarrow{\phi_i, \psi_i}$ $\xrightarrow{x_i, y_i}$	(ϕ_i, ψ_i) random scalars
$z^T = \lambda^T L$	$\xleftarrow{\lambda_i}$ $\xrightarrow{z_i}$	λ_i random scalar
		$z^T \begin{bmatrix} x & y \end{bmatrix} \stackrel{?}{=} (\lambda^T A) \begin{bmatrix} \phi & \psi \end{bmatrix}$

Certificate for LU decomposition

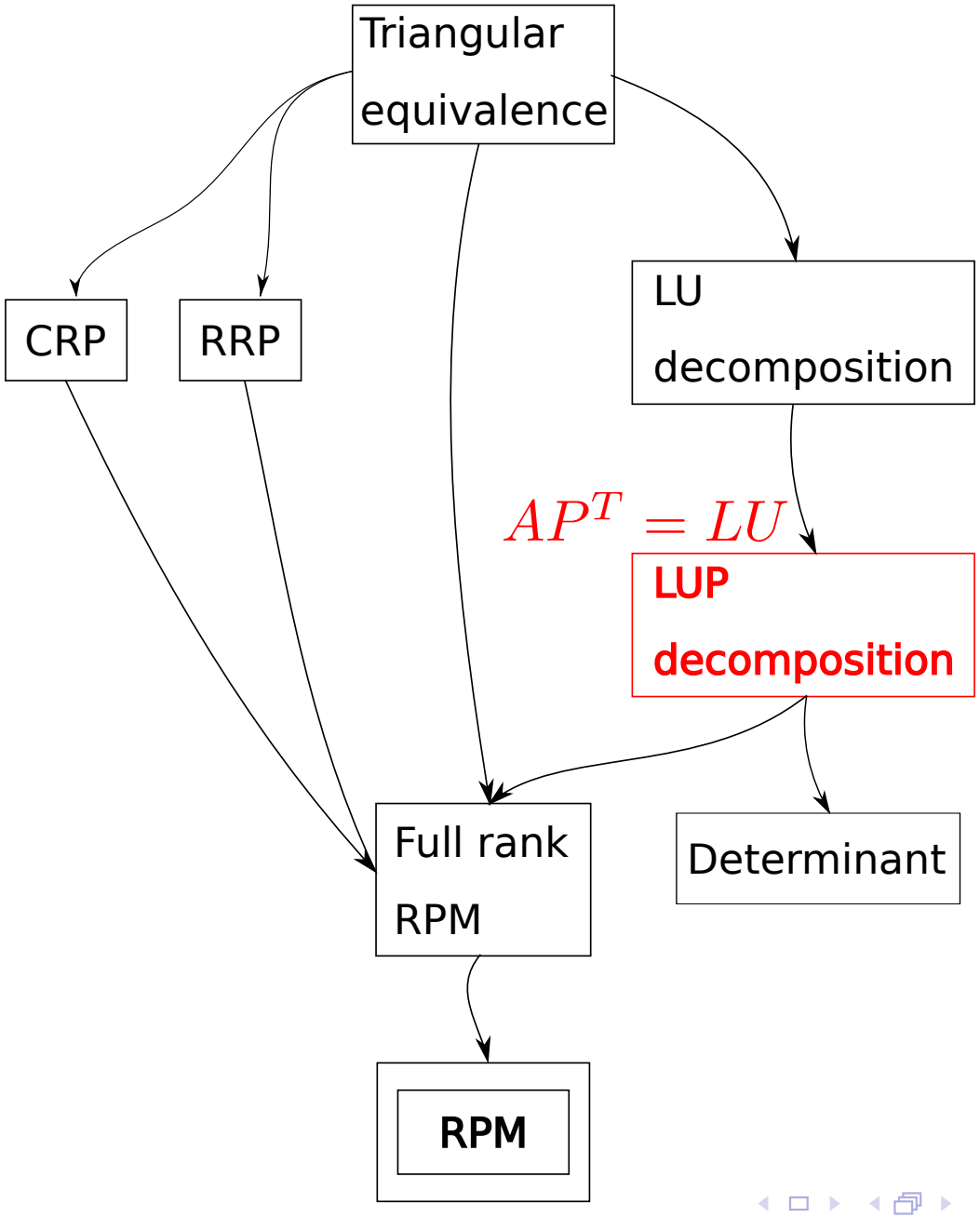
Prover		Verifier
	$A \in \mathbf{F}^{n \times n}$ non-singular	
$A = LU$	$\xrightarrow{A = LU}$	
	for i from n downto 1	
$\begin{bmatrix} x & y \end{bmatrix} = U \begin{bmatrix} \phi & \psi \end{bmatrix}$	$\xleftarrow{\phi_i, \psi_i}$ $\xrightarrow{x_i, y_i}$	(ϕ_i, ψ_i) random scalars
$z^T = \lambda^T L$	$\xleftarrow{\lambda_i}$ $\xrightarrow{z_i}$	λ_i random scalar
		$z^T \begin{bmatrix} x & y \end{bmatrix} \stackrel{?}{=} (\lambda^T A) \begin{bmatrix} \phi & \psi \end{bmatrix}$

Prover	Comm.	Verifier
$O(n^\omega)$	$3n$	$O(n^2)$

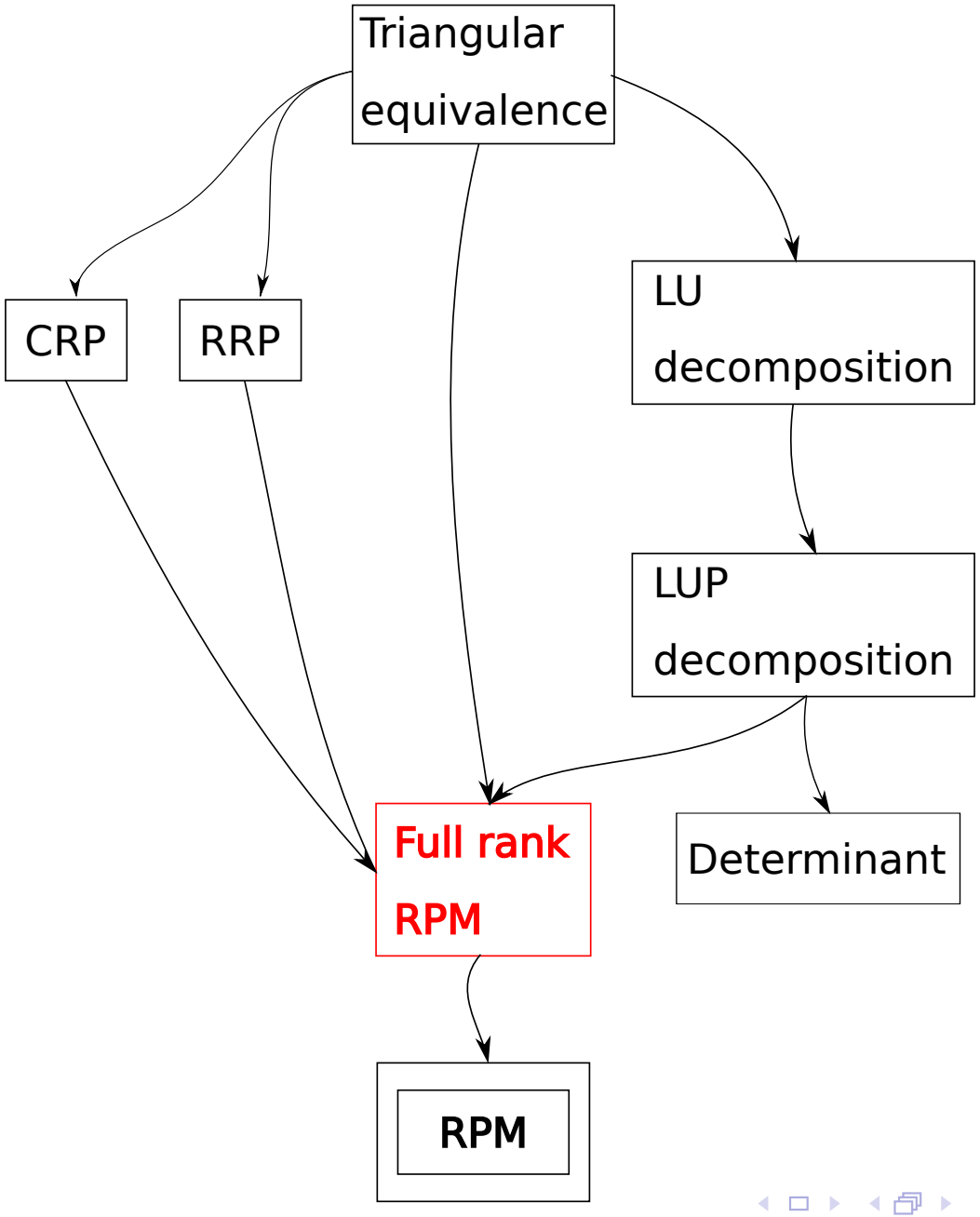
Roadmap


Triangular equivalence
CRP
RRP
LU decomposition
LUP decomposition
Full rank RPM
Determinant
RPM

Roadmap


Triangular equivalence
CRP
RRP
LU decomposition
$AP^T = LU$
LUP decomposition
Full rank RPM
Determinant
RPM

Roadmap


Triangular equivalence
CRP
RRP
LU decomposition
LUP decomposition
Full rank RPM
Determinant
RPM

Rank profile matrix for an invertible matrix

- LUP decomposition that reveals the RPM $\iff \exists \overline{U} = P^T UP$
  - U and $\overline{U}$ upper triangular
- Difficulty: neither U nor $\overline{U}$ will be transmitted

Rank profile matrix for an invertible matrix

Prover | Verifier

$A \in \mathbf{F}^{n \times n}$ invertible

$A = LUP$

$\xrightarrow{P}$

$\xleftarrow{e}$

Triangular equivalence

$f^T \leftarrow e^T \overline{U}$

$\xrightarrow{f}$

$AP^T \stackrel{?}{=} LU$

$$\begin{bmatrix} x & y \end{bmatrix} = U \begin{bmatrix} \phi & \psi \end{bmatrix}$$

$$z^t = \lambda^T L$$

$\xrightarrow{z^T}$

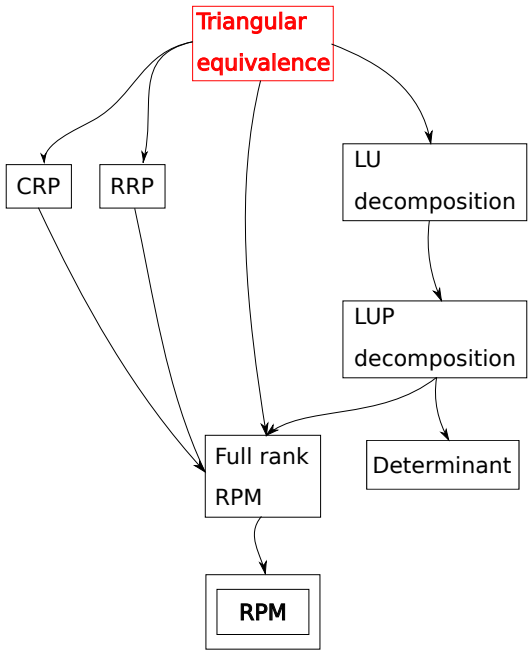
$\xrightarrow{x,y}$

$$z^T \begin{bmatrix} x & y \end{bmatrix} \stackrel{?}{=} \lambda^T A P^T \begin{bmatrix} \phi & \psi \end{bmatrix}$$

$e^T P^T x \stackrel{?}{=} f^T P^T \phi$

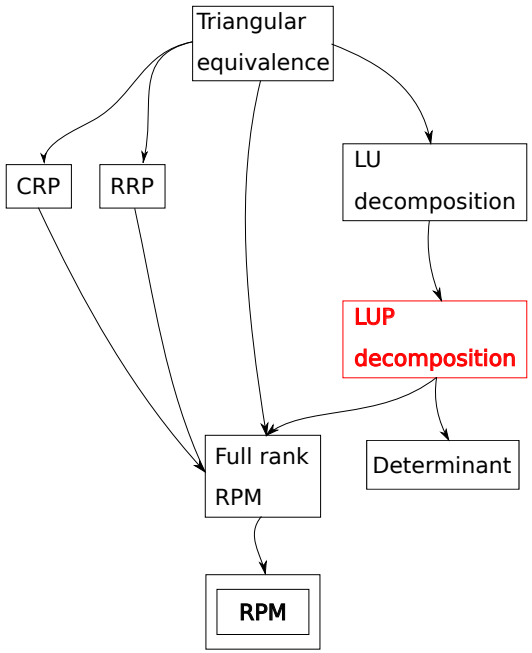
Summary

	Prover	Comm.	Verifier
Tri. eq.	Best known	$2n$	$O(n^2)$



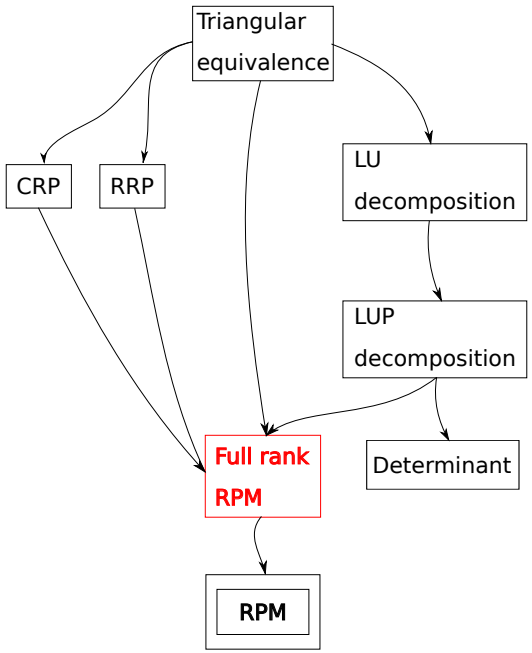
Summary

	Prover	Comm.	Verifier
Tri. eq.	Best known	$2n$	$O(n^2)$
LUP	Best known	$8n$	$O(n^2)$



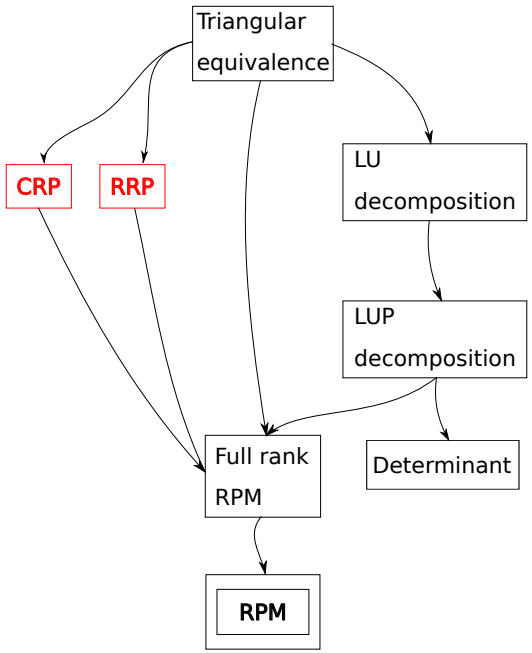
Summary

	Prover	Comm.	Verifier
Tri. eq.	Best known	$2n$	$O(n^2)$
LUP	Best known	$8n$	$O(n^2)$
Full Rank RPM	Best known	$10n$	$O(n^2)$



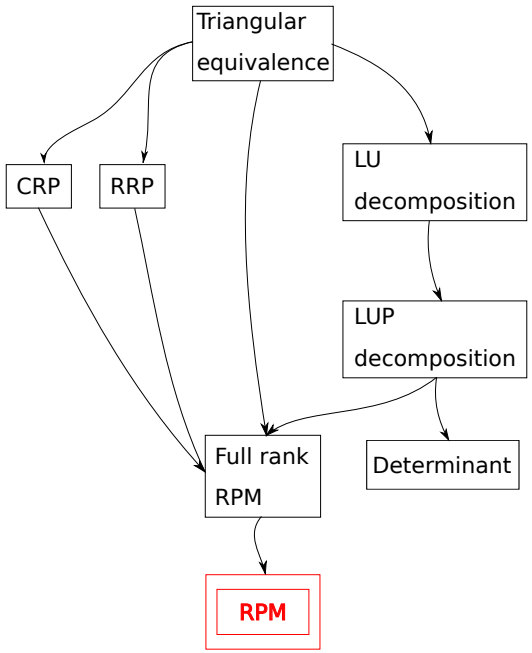
Summary

	Prover	Comm.	Verifier
Tri. eq.	Best known	$2n$	$O(n^2)$
LUP	Best known	$8n$	$O(n^2)$
Full Rank RPM	Best known	$10n$	$O(n^2)$
CRP	Best known	$2n + 4r$	$O(n^2)$



Summary

	Prover	Comm.	Verifier
Tri. eq.	Best known	$2n$	$O(n^2)$
LUP	Best known	$8n$	$O(n^2)$
Full Rank RPM	Best known	$10n$	$O(n^2)$
CRP	Best known	$2n + 4r$	$O(n^2)$
RPM	Best known	$3n + 16r$	$O(n^2)$



Plan

1. Introduction
2. Our certificates
3. Conclusion

Conclusion

- Application:
  - Dense determinant certificate from LUP instead of `CHARPOLY`
  - About 5 times more efficient for the prover than the dense case of [DKTV16]
- Perspectives:
  - LDL^T certificate and symmetric positive-definiteness

Determinant computation

	Prover	Comm.	Verifier
[F79]	Best known dense	$O(n^2)$	$O(n^2)$
[DKTV16]	Best known sparse	$5n$	$O(n^2)$
This paper	Best known dense	$8n$	$O(n^2)$

- Computation time measured using FFLAS-FFPACK
- Single core Intel Skylake @3.4GHz
- Communication over Ethernet Cat. 6, @1Gb/s cable
- 64 bit words, random matrices modulo 131071

Determinant computation

	Prover	Comm.	Verifier
[F79]	Best known dense	$O(n^2)$	$O(n^2)$
[DKTV16]	Best known sparse	$5n$	$O(n^2)$
This paper	Best known dense	$8n$	$O(n^2)$

- Computation time measured using FFLAS-FFPACK
- Single core Intel Skylake @3.4GHz
- Communication over Ethernet Cat. 6, @1Gb/s cable
- 64 bit words, random matrices modulo 131071

Dimension	$2k$	$10k$	$50k$
PLUQ	0.28s	17.99s	1448.16s
CharPoly	1.96s	100.37s	8047.56s
Linear comm.	0.50s	0.50s	0.50s
Quadratic comm.	1.50s	7.50s	222.68s
`fgemv`	0.0013s	0.038s	1.03s