Certification of Matrix Interpretations in Coq

Adam Koprowski
(joint work with Hans Zantema)

Eindhoven University of Technology
Department of Mathematics and Computer Science

11 May 2007
CoLoR workshop

Outline

1. CoLoR
   - Motivation
2. Formalization of matrix interpretations
   - Introduction to matrix interpretations
   - Monotone algebras
   - Matrices
   - Matrix interpretations
   - Practicalities
3. CoLoR
   - Overview
   - Proof format

Outline

Outline

Outline

1. CoLoR
   - Motivation
2. Formalization of matrix interpretations
   - Introduction to matrix interpretations
   - Monotone algebras
   - Matrices
   - Matrix interpretations
   - Practicalities
3. CoLoR
   - Overview
   - Proof format

Outline

Outline

Outline

Outline

Outline

1. CoLoR
   - Motivation
2. Formalization of matrix interpretations
3. CoLoR

Motivation:

- Enhanced trust in tools' results.
- Common proof format – all tools speaking the same language!
    - common tools (proof presentation, manipulation, . . .),
    - easier integration of the tools [Waldmann],
    - categories for single technique in the competition [Middeldorp],

Motivation:

- Enhanced trust in tools' results.
- Common proof format – all tools speaking the same language!
  - common tools (proof presentation, manipulation, . . .),
  - easier integration of the tools [Waldmann],
  - categories for single technique in the competition [Middeldorp],

Motivation:

- Enhanced trust in tools' results.
- Common proof format – all tools speaking the same language!
  - common tools (proof presentation, manipulation, . . .),
  - easier integration of the tools [Waldmann],
  - categories for single technique in the competition [Middeldorp],

Motivation:

- Enhanced trust in tools' results.
- Common proof format – all tools speaking the same language!
  - common tools (proof presentation, manipulation, . . .),
  - easier integration of the tools [Waldmann],
  - categories for single technique in the competition [Middeldorp],

Motivation:

- Enhanced trust in tools' results.
- Common proof format – all tools speaking the same language!
  - common tools (proof presentation, manipulation, . . .),
  - easier integration of the tools [Waldmann],
  - categories for single technique in the competition [Middeldorp],

Outline

z086.trs

$$a(a(x)) \to c(b(x)), \quad b(b(x)) \to c(a(x)), \quad c(c(x)) \to b(a(x))$$

Matrix interpretation for z086.trs

$$a(x) = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$

$$b(x) = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$c(x) = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

z086.trs

$$a(a(x)) \to c(b(x)), \quad b(b(x)) \to c(a(x)), \quad c(c(x)) \to b(a(x))$$

Matrix interpretation for z086.trs

$$a(x) = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$

$$b(x) = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$c(x) = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Termination proof for z086.trs

$$a(a(x)) = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix} \right) + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$

$$c(b(x)) = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \right) + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

Termination proof for z086.trs

$$a(a(x)) = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix} \right) + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$

$$c(b(x)) = \begin{bmatrix} 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \left(\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \right) + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \end{bmatrix}$$

Outline

Definition (Monotonicity)

An operation $[f] : A \times \cdots \times A \to A$ is *monotone* with respect to a binary relation $\rhd$ on A if
$a_i \rhd a_i' \implies [f](a_1, \ldots, a_i, \ldots a_n) \rhd [f](a_1, \ldots, a_i', \ldots, a_n)$.

Definition

Given a relation $\rhd$ on A we define its extension to a relation on terms as:

$$s \rhd_{\mathcal{T}} t \equiv \forall \alpha : \mathcal{X} \to A, [s, \alpha] \rhd [t, \alpha]$$

Definition (Monotonicity)

An operation $[f] : A \times \cdots \times A \rightarrow A$ is *monotone* with respect to a binary relation $\rhd$ on A if
$a_i \rhd a_i' \implies [f](a_1, \ldots, a_i, \ldots a_n) \rhd [f](a_1, \ldots, a_i', \ldots, a_n)$.

Definition

Given a relation $\rhd$ on A we define its extension to a relation on terms as:

$$s \rhd_{\mathcal{T}} t \equiv \forall \alpha : \mathcal{X} \rightarrow A, [s, \alpha] \rhd [t, \alpha]$$

Definition (A weakly monotone Σ-algebra)

A *weakly monotone Σ-algebra* $(A, [\cdot], >, \gtrsim)$ is a Σ-algebra $(A, [\cdot])$ equipped with two binary relations $>, \gtrsim$ on A such that

- $>$ is well-founded;
- $> \cdot \gtrsim \subseteq >$;
- for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $\gtrsim$.

Definition (An *extended monotone Σ-algebra*)

An *extended monotone Σ-algebra* $(A, [\cdot], >, \gtrsim)$ is a weakly monotone Σ-algebra $(A, [\cdot], >, \gtrsim)$ in which moreover for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $>$.

Definition (A weakly monotone Σ-algebra)

A *weakly monotone Σ-algebra* $(A, [\cdot], >, \gtrsim)$ is a Σ-algebra $(A, [\cdot])$ equipped with two binary relations $>$, $\gtrsim$ on A such that

- $>$ is well-founded;
- $> \cdot \gtrsim \subseteq >$;
- for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $\gtrsim$.

Definition (An *extended monotone Σ-algebra*)

An *extended monotone Σ-algebra* $(A, [\cdot], >, \gtrsim)$ is a weakly monotone Σ-algebra $(A, [\cdot], >, \gtrsim)$ in which moreover for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $>$.

Definition (A weakly monotone Σ-algebra)

A *weakly monotone Σ-algebra* $(A, [\cdot], >, \gtrsim)$ is a Σ-algebra $(A, [\cdot])$ equipped with two binary relations $>$, $\gtrsim$ on A such that

- $>$ is well-founded;
- $> \cdot \gtrsim \subseteq >$;
- for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $\gtrsim$.

Definition (An *extended monotone Σ-algebra*)

An *extended monotone Σ-algebra* $(A, [\cdot], >, \gtrsim)$ is a weakly monotone Σ-algebra $(A, [\cdot], >, \gtrsim)$ in which moreover for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $>$.

Definition (A weakly monotone Σ-algebra)

A *weakly monotone Σ-algebra* $(A, [\cdot], >, \gtrsim)$ is a Σ-algebra $(A, [\cdot])$ equipped with two binary relations $>$, $\gtrsim$ on A such that

- $>$ is well-founded;
- $> \cdot \gtrsim \subseteq >$;
- for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $\gtrsim$.

Definition (An *extended monotone Σ-algebra*)

An *extended monotone Σ-algebra* $(A, [\cdot], >, \gtrsim)$ is a weakly monotone Σ-algebra $(A, [\cdot], >, \gtrsim)$ in which moreover for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $>$.

Definition (A weakly monotone Σ-algebra)

A *weakly monotone* Σ*-algebra* $(A, [\cdot], >, \gtrsim)$ is a Σ-algebra $(A, [\cdot])$ equipped with two binary relations $>, \gtrsim$ on A such that

- $>$ is well-founded;
- $> \cdot \gtrsim \subseteq >$;
- for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $\gtrsim$.

Definition (An *extended monotone* Σ*-algebra*)

An *extended monotone* Σ*-algebra* $(A, [\cdot], >, \gtrsim)$ is a weakly monotone Σ-algebra $(A, [\cdot], >, \gtrsim)$ in which moreover for every $f \in \Sigma$ the operation $[f]$ is monotone with respect to $>$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, $(A, [\cdot], >, \gtrsim)$ be an extended monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R' \cup S'$

Then $\mathrm{SN}(R/S)$ *implies* $\mathrm{SN}(R \cup R' \ / \ S \cup S')$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, let $(A, [\cdot], >, \gtrsim)$ be a weakly monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* R',

Then $\mathrm{SN}(R_{\mathrm{top}}/S)$ *implies* $\mathrm{SN}((R \cup R')_{\mathrm{top}}/S)$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, $(A, [\cdot], >, \gtrsim)$ be an extended monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{T}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{T}} r$ *for every rule* $\ell \to r$ *in* $R' \cup S'$

Then $\mathrm{SN}(R/S)$ *implies* $\mathrm{SN}(R \cup R' \ / \ S \cup S')$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, let $(A, [\cdot], >, \gtrsim)$ be a weakly monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{T}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{T}} r$ *for every rule* $\ell \to r$ *in* R',

Then $\mathrm{SN}(R_{\mathrm{top}}/S)$ *implies* $\mathrm{SN}((R \cup R')_{\mathrm{top}}/S)$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, $(A, [\cdot], >, \gtrsim)$ be an extended monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{T}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{T}} r$ *for every rule* $\ell \to r$ *in* $R' \cup S'$

Then $\mathrm{SN}(R/S)$ *implies* $\mathrm{SN}(R \cup R' \;/\; S \cup S')$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, let $(A, [\cdot], >, \gtrsim)$ be a weakly monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{T}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{T}} r$ *for every rule* $\ell \to r$ *in* R',

Then $\mathrm{SN}(R_{\mathrm{top}}/S)$ *implies* $\mathrm{SN}((R \cup R')_{\mathrm{top}}/S)$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, $(A, [\cdot], >, \gtrsim)$ be an extended monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R' \cup S'$

Then $\mathrm{SN}(R/S)$ *implies* $\mathrm{SN}(R \cup R' \ / \ S \cup S')$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, let $(A, [\cdot], >, \gtrsim)$ be a weakly monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* R',

Then $\mathrm{SN}(R_{\mathrm{top}}/S)$ *implies* $\mathrm{SN}((R \cup R')_{\mathrm{top}}/S)$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, $(A, [\cdot], >, \gtrsim)$ be an extended monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R' \cup S'$

Then $\mathrm{SN}(R/S)$ *implies* $\mathrm{SN}(R \cup R' \ / \ S \cup S')$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, let $(A, [\cdot], >, \gtrsim)$ be a weakly monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* R',

Then $\mathrm{SN}(R_{\text{top}}/S)$ *implies* $\mathrm{SN}((R \cup R')_{\text{top}}/S)$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, $(A, [\cdot], >, \gtrsim)$ be an extended monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R' \cup S'$

Then $\mathrm{SN}(R/S)$ *implies* $\mathrm{SN}(R \cup R' \ / \ S \cup S')$.

Theorem

Let R, R', S, S' be TRSs over a signature Σ, let $(A, [\cdot], >, \gtrsim)$ be a weakly monotone Σ-algebra such that:

- $\ell \gtrsim_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* $R \cup S$ *and*
- $\ell >_{\mathcal{I}} r$ *for every rule* $\ell \to r$ *in* R'*,*

Then $\mathrm{SN}(R_{\mathrm{top}}/S)$ *implies* $\mathrm{SN}((R \cup R')_{\mathrm{top}}/S)$.

- Monotone algebras are formalized as a functor.
- Apart for the aforementioned requirements there is one additional required to deal with concrete examples: $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ must be decidable.
- More precisely the requirement is to provide a relation $\gg$, such that
  - $\gg \subseteq >_{\mathcal{T}}$ and
  - $\gg$ is decidable
  - similarly for $\gtrsim$.
- The structure returned by the functor contains all the machinery required to prove (relative)-(top)-termination in Coq.

- Monotone algebras are formalized as a functor.
- Apart for the aforementioned requirements there is one additional required to deal with concrete examples: $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ must be decidable.
- More precisely the requirement is to provide a relation $\gg$, such that
  - $\gg \subseteq >_{\mathcal{T}}$ and
  - $\gg$ is decidable
  - similarly for $\gtrsim$.
- The structure returned by the functor contains all the machinery required to prove (relative)-(top)-termination in Coq.

- Monotone algebras are formalized as a functor.
- Apart for the aforementioned requirements there is one additional required to deal with concrete examples: $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ must be decidable.
- More precisely the requirement is to provide a relation $\gg$, such that
  - $\gg \subseteq >_{\mathcal{T}}$ and
  - $\gg$ is decidable
  - similarly for $\gtrsim$.
- The structure returned by the functor contains all the machinery required to prove (relative)-(top)-termination in Coq.

- Monotone algebras are formalized as a functor.
- Apart for the aforementioned requirements there is one additional required to deal with concrete examples: $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ must be decidable.
- More precisely the requirement is to provide a relation $\gg$, such that
  - $\gg \subseteq >_{\mathcal{T}}$ and
  - $\gg$ is decidable
  - similarly for $\gtrsim$.
- The structure returned by the functor contains all the machinery required to prove (relative)-(top)-termination in Coq.

- Monotone algebras are formalized as a functor.
- Apart for the aforementioned requirements there is one additional required to deal with concrete examples: $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ must be decidable.
- More precisely the requirement is to provide a relation $\gg$, such that
  - $\gg \subseteq >_{\mathcal{T}}$ and
  - $\gg$ is decidable
  - similarly for $\gtrsim$.
- The structure returned by the functor contains all the machinery required to prove (relative)-(top)-termination in Coq.

- Monotone algebras are formalized as a functor.
- Apart for the aforementioned requirements there is one additional required to deal with concrete examples: $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ must be decidable.
- More precisely the requirement is to provide a relation $\gg$, such that
  - $\gg \subseteq >_{\mathcal{T}}$ and
  - $\gg$ is decidable
  - similarly for $\gtrsim$.
- The structure returned by the functor contains all the machinery required to prove (relative)-(top)-termination in Coq.

- Monotone algebras are formalized as a functor.
- Apart for the aforementioned requirements there is one additional required to deal with concrete examples: $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ must be decidable.
- More precisely the requirement is to provide a relation $\gg$, such that
  - $\gg \subseteq >_{\mathcal{T}}$ and
  - $\gg$ is decidable
  - similarly for $\gtrsim$.
- The structure returned by the functor contains all the machinery required to prove (relative)-(top)-termination in Coq.

Outline

- Matrices are formalized as a functor taking as an argument the semi-ring of coefficients $\mathcal{R}$ and providing a structure of matrices of arbitrary sizes with coefficients in $\mathcal{R}$ and
- a number of basic operations over matrices such as:

$$[\cdot], \quad M_{i,j}, \quad M + N, \quad M * N, \quad M^T, \ \ldots$$

- and a number of basic properties such as:
  - $M + N = N + M$,
  - $M * (N * P) = (M * N) * P$
  - monotonicity of $*$
  - ...

- Matrices are formalized as a functor taking as an argument the semi-ring of coefficients $\mathcal{R}$ and providing a structure of matrices of arbitrary sizes with coefficients in $\mathcal{R}$ and
- a number of basic operations over matrices such as:

$$[\cdot], \quad M_{i,j}, \quad M + N, \quad M * N, \quad M^T, \ \ldots$$

- and a number of basic properties such as:
  - $M + N = N + M$,
  - $M * (N * P) = (M * N) * P$
  - monotonicity of $*$
  - $\ldots$

- Matrices are formalized as a functor taking as an argument the semi-ring of coefficients $\mathcal{R}$ and providing a structure of matrices of arbitrary sizes with coefficients in $\mathcal{R}$ and
- a number of basic operations over matrices such as:

$$[\cdot], \quad M_{i,j}, \quad M+N, \quad M*N, \quad M^T, \ \ldots$$

- and a number of basic properties such as:
  - $M+N=N+M$,
  - $M*(N*P)=(M*N)*P$
  - monotonicity of $*$
  - ...

- Matrices are formalized as a functor taking as an argument the semi-ring of coefficients $\mathcal{R}$ and providing a structure of matrices of arbitrary sizes with coefficients in $\mathcal{R}$ and
- a number of basic operations over matrices such as:

$$[\cdot], \quad M_{i,j}, \quad M + N, \quad M * N, \quad M^T, \ \ldots$$

- and a number of basic properties such as:
  - $M + N = N + M$,
  - $M * (N * P) = (M * N) * P$
  - monotonicity of $*$
  - ...

- Matrices are formalized as a functor taking as an argument the semi-ring of coefficients $\mathcal{R}$ and providing a structure of matrices of arbitrary sizes with coefficients in $\mathcal{R}$ and
- a number of basic operations over matrices such as:

$$[\cdot], \quad M_{i,j}, \quad M + N, \quad M * N, \quad M^T, \; \ldots$$

- and a number of basic properties such as:
  - $M + N = N + M$,
  - $M * (N * P) = (M * N) * P$
  - monotonicity of $*$
  - $\ldots$

- Matrices are formalized as a functor taking as an argument the semi-ring of coefficients $\mathcal{R}$ and providing a structure of matrices of arbitrary sizes with coefficients in $\mathcal{R}$ and
- a number of basic operations over matrices such as:

$$[\cdot], \quad M_{i,j}, \quad M + N, \quad M * N, \quad M^T, \; \ldots$$

- and a number of basic properties such as:
  - $M + N = N + M$,
  - $M * (N * P) = (M * N) * P$
  - monotonicity of $*$
  - ...

- Matrices are formalized as a functor taking as an argument the semi-ring of coefficients $\mathcal{R}$ and providing a structure of matrices of arbitrary sizes with coefficients in $\mathcal{R}$ and
- a number of basic operations over matrices such as:

$$[\cdot], \quad M_{i,j}, \quad M + N, \quad M * N, \quad M^T, \; \ldots$$

- and a number of basic properties such as:
  - $M + N = N + M$,
  - $M * (N * P) = (M * N) * P$
  - monotonicity of $*$
  - ...

Outline

1. CoLoR
2. Formalization of matrix interpretations
   - Introduction to matrix interpretations
   - Monotone algebras
   - Matrices
   - Matrix interpretations
   - Practicalities
3. CoLoR

Polynomial interpretations in the setting of monotone algebras:

- $A = \mathbb{Z}$,
- $> \, = \, >_{\mathbb{Z}}$, $\gtrsim = \geq_{\mathbb{Z}}$,
- interpretations represented by polynomials $[f(x_1, \ldots, x_n)] = P_{\mathbb{Z}}(x_1, \ldots, x_n)$,
- $>_{\mathcal{T}}$ not decidable (positiveness of polynomial) — heuristics required.

Polynomial interpretations in the setting of monotone algebras:

- $A = \mathbb{Z}$,
- $> \, = \, >_{\mathbb{Z}}$, $\gtrsim = \geq_{\mathbb{Z}}$,
- interpretations represented by polynomials $[f(x_1, \ldots, x_n)] = P_{\mathbb{Z}}(x_1, \ldots, x_n)$,
- $>_{\mathcal{T}}$ not decidable (positiveness of polynomial) — heuristics required.

Polynomial interpretations in the setting of monotone algebras:

- $A = \mathbb{Z}$,
- $> = >_{\mathbb{Z}}$, $\gtrsim = \geq_{\mathbb{Z}}$,
- interpretations represented by polynomials $[f(x_1, \ldots, x_n)] = P_{\mathbb{Z}}(x_1, \ldots, x_n)$,
- $>_{\mathcal{T}}$ not decidable (positiveness of polynomial) — heuristics required.

Polynomial interpretations in the setting of monotone algebras:

- $A = \mathbb{Z}$,
- $> \; = \; >_{\mathbb{Z}}$, $\gtrsim = \geq_{\mathbb{Z}}$,
- interpretations represented by polynomials $[f(x_1, \ldots, x_n)] = P_{\mathbb{Z}}(x_1, \ldots, x_n)$,
- $>_{\mathcal{T}}$ not decidable (positiveness of polynomial) — heuristics required.

Matrix interpretations in the setting of monotone algebras:

- fix a dimension d,
- $A = \mathbb{N}^d$,
- $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d)$ iff $\forall i, u_i \geq_{\mathbb{N}} v_i$,
- $(u_1, \ldots, u_d) > (v_1, \ldots, v_d)$ iff $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d) \wedge u_1 >_{\mathbb{N}} v_1$,
- interpretations represented as: $[f(x_1, \ldots, x_n)] = M_1 x_1 + \ldots + M_n x_n + v$ where $M_i \in \mathbb{N}^{d \times d}$, $v \in \mathbb{N}^d$,
- $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ are decidable in this case but thanks to introducing $\gg$ we do not need to prove completeness of their characterization.

Matrix interpretations in the setting of monotone algebras:

- fix a dimension d,
- $A = \mathbb{N}^d$,
- $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d)$ iff $\forall i, u_i \geq_{\mathbb{N}} v_i$,
- $(u_1, \ldots, u_d) > (v_1, \ldots, v_d)$ iff $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d) \land u_1 >_{\mathbb{N}} v_1$,
- interpretations represented as: $[f(x_1, \ldots, x_n)] = M_1 x_1 + \ldots + M_n x_n + v$ where $M_i \in \mathbb{N}^{d \times d}$, $v \in \mathbb{N}^d$,
- $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ are decidable in this case but thanks to introducing $\gg$ we do not need to prove completeness of their characterization.

Matrix interpretations in the setting of monotone algebras:

- fix a dimension d,
- $A = \mathbb{N}^d$,
- $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d)$ iff $\forall i, u_i \geq_{\mathbb{N}} v_i$,
- $(u_1, \ldots, u_d) > (v_1, \ldots, v_d)$ iff $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d) \wedge u_1 >_{\mathbb{N}} v_1$,
- interpretations represented as: $[f(x_1, \ldots, x_n)] = M_1 x_1 + \ldots + M_n x_n + v$ where $M_i \in \mathbb{N}^{d \times d}$, $v \in \mathbb{N}^d$,
- $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ are decidable in this case but thanks to introducing $\gg$ we do not need to prove completeness of their characterization.

Matrix interpretations in the setting of monotone algebras:

- fix a dimension d,
- $A = \mathbb{N}^d$,
- $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d)$ iff $\forall i, u_i \geq_{\mathbb{N}} v_i$,
- $(u_1, \ldots, u_d) > (v_1, \ldots, v_d)$ iff $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d) \wedge u_1 >_{\mathbb{N}} v_1$,
- interpretations represented as: $[f(x_1, \ldots, x_n)] = M_1 x_1 + \ldots + M_n x_n + v$ where $M_i \in \mathbb{N}^{d \times d}$, $v \in \mathbb{N}^d$,
- $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ are decidable in this case but thanks to introducing $\gg$ we do not need to prove completeness of their characterization.

Matrix interpretations in the setting of monotone algebras:

- fix a dimension d,
- $A = \mathbb{N}^d$,
- $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d)$ iff $\forall i, u_i \geq_{\mathbb{N}} v_i$,
- $(u_1, \ldots, u_d) > (v_1, \ldots, v_d)$ iff $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d) \wedge u_1 >_{\mathbb{N}} v_1$,
- interpretations represented as: $[f(x_1, \ldots, x_n)] = M_1 x_1 + \ldots + M_n x_n + v$ where $M_i \in \mathbb{N}^{d \times d}$, $v \in \mathbb{N}^d$,
- $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ are decidable in this case but thanks to introducing $\gg$ we do not need to prove completeness of their characterization.

Matrix interpretations in the setting of monotone algebras:

- fix a dimension d,
- $A = \mathbb{N}^d$,
- $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d)$ iff $\forall i, u_i \geq_{\mathbb{N}} v_i$,
- $(u_1, \ldots, u_d) > (v_1, \ldots, v_d)$ iff $(u_1, \ldots, u_d) \gtrsim (v_1, \ldots, v_d) \wedge u_1 >_{\mathbb{N}} v_1$,
- interpretations represented as: $[f(x_1, \ldots, x_n)] = M_1 x_1 + \ldots + M_n x_n + v$ where $M_i \in \mathbb{N}^{d \times d}$, $v \in \mathbb{N}^d$,
- $>_{\mathcal{T}}$ and $\gtrsim_{\mathcal{T}}$ are decidable in this case but thanks to introducing $\gg$ we do not need to prove completeness of their characterization.

Outline

Formalization size (LOC):

• Monotone algebras:	351
• Matrices:	642
• Matrix interpretations:	673
• Polynomial interpretations in MA setting:	116

Formalization size (LOC):

• Monotone algebras:	351
• Matrices:	642
• Matrix interpretations:	673
• Polynomial interpretations in MA setting:	116

Formalization size (LOC):

Monotone algebras:	351
Matrices:	642
Matrix interpretations:	673
Polynomial interpretations in MA setting:	116

Formalization size (LOC):

Monotone algebras:	351
Matrices:	642
Matrix interpretations:	673
Polynomial interpretations in MA setting:	116

Evaluation of TPA + Rainbow on TPDB 3.2 (864 TRSs):

- polynomial interpretations: 167
- matrix interpretations: 237
- polynomial and matrix interpretations: 275
  - Verification time: AVG: 5sec. MAX: 75sec.
  - Certificate size: AVG: 25kB. MAX: 437kB
  - Proof steps: AVG: 5 MAX: 29

Evaluation of TPA + Rainbow on TPDB 3.2 (864 TRSs):

- polynomial interpretations: 167
- matrix interpretations: 237
- polynomial and matrix interpretations: 275
  - Verification time: AVG: 5sec. MAX: 75sec.
  - Certificate size: AVG: 25kB. MAX: 437kB
  - Proof steps: AVG: 5 MAX: 29

Evaluation of TPA + Rainbow on TPDB 3.2 (864 TRSs):

- polynomial interpretations: 167
- matrix interpretations: 237
- polynomial and matrix interpretations: 275
  - Verification time: AVG: 5sec. MAX: 75sec.
  - Certificate size: AVG: 25kB. MAX: 437kB
  - Proof steps: AVG: 5 MAX: 29

Evaluation of TPA + Rainbow on TPDB 3.2 (864 TRSs):

- polynomial interpretations: 167
- matrix interpretations: 237
- polynomial and matrix interpretations: 275
  - Verification time: AVG: 5sec. MAX: 75sec.
  - Certificate size: AVG: 25kB. MAX: 437kB
  - Proof steps: AVG: 5 MAX: 29

Evaluation of TPA + Rainbow on TPDB 3.2 (864 TRSs):

- polynomial interpretations: 167
- matrix interpretations: 237
- polynomial and matrix interpretations: 275
  - Verification time: AVG: 5sec. MAX: 75sec.
  - Certificate size: AVG: 25kB. MAX: 437kB
  - Proof steps: AVG: 5 MAX: 29

Evaluation of TPA + Rainbow on TPDB 3.2 (864 TRSs):

- polynomial interpretations: 167
- matrix interpretations: 237
- polynomial and matrix interpretations: 275
  - Verification time: AVG: 5sec. MAX: 75sec.
  - Certificate size: AVG: 25kB. MAX: 437kB
  - Proof steps: AVG: 5 MAX: 29

Evaluation of TPA + Rainbow on TPDB 3.2 (864 TRSs):

- polynomial interpretations: 167
- matrix interpretations: 237
- polynomial and matrix interpretations: 275
  - Verification time: AVG: 5sec. MAX: 75sec.
  - Certificate size: AVG: 25kB. MAX: 437kB
  - Proof steps: AVG: 5 MAX: 29

"demo"

Outline

1. CoLoR
2. Formalization of matrix interpretations
3. CoLoR
   - Overview
   - Proof format

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR.

- Termination criteria:
  - **matrix interpretations** [Koprowski, Zantema]
  - dependency graph cycles [Blanqui]
  - higher-order recursive path ordering [Koprowski]
  - recursive path ordering [Coupet-Grimal, Delobel]
  - multiset ordering [Koprowski]
  - **polynomial interpretations** [Hinderer]
- Transformation techniques:
  - dependency pairs [Blanqui]
  - rule elimination [Blanqui]
  - arguments filtering [Blanqui]
  - conversion from algebraic to varyadic terms [Blanqui]

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

Content of CoLoR (ctd.)

- General libraries:
  - matrices [Koprowski]
  - simply typed lambda-terms [Koprowski]
  - finite multisets [Koprowski]
  - varyadic terms [Blanqui]
  - algebraic terms with symbols of fixed arity [Hinderer, Blanqui]
  - integer polynomials with multiple variables [Hinderer]
  - vectors [Hinderer, Blanqui]
  - lists, relations, etc.

- Lines of code

Data-types	11.990	28.69%
Terms	18.474	44.21%
Math	4.007	9.59%
Term. techniques	7.312	17.49%
Total	41.783	100.00%

- Coq constructs

Inductive definitions	38
Fixpoint definitions	115
Definitions	554
Lemmas	2.160

- Lines of code

Data-types	11.990	28.69%
Terms	18.474	44.21%
Math	4.007	9.59%
Term. techniques	7.312	17.49%
Total	41.783	100.00%

- Coq constructs

Inductive definitions	38
Fixpoint definitions	115
Definitions	554
Lemmas	2.160

Outline

1. CoLoR
2. Formalization of matrix interpretations
3. CoLoR
    - Overview
    - Proof format

```
type vector = int list

type matrix = vector list

type monom = int list

type polynom = (int * monom) list

type poly_int = polynom FMap.t

type mi_fun = { mi_const: vector; mi_args: matrix list }

type matrix_int = { mi_dim: int; mi_int: mi_fun FMap.t }

type red_ord =
  | PolyInt of poly_int
  | MatrixInt of matrix_int

type proof =
  | MannaNess of red_ord * proof
  | Trivial
```