FLABloM: Functional Linear Algebra with Block Matrices

Adam Sandberg Eriksson Patrik Jansson

Chalmers University of Technology, Sweden
{saadam,patrikj}@chalmers.se

June 13, 2016

Functional linear algebra with block matrices

- Inspired by work on parallel parsing by Bernardy & Jansson
- Matrices in Agda
- Reflexive, transitive closure of matrices

$$\begin{bmatrix} \begin{bmatrix} 1 & \begin{bmatrix} 0 & 1 \end{bmatrix} \\ \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \end{bmatrix} & \begin{bmatrix} 0 & \begin{bmatrix} 1 & 2 \end{bmatrix} \\ \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 1 & 7 \\ 3 & 8 \end{bmatrix} \end{bmatrix} \\ \begin{bmatrix} 2 & \begin{bmatrix} 1 & 0 \end{bmatrix} \\ \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \begin{bmatrix} 9 & 8 \\ 7 & 6 \end{bmatrix} \end{bmatrix} & \begin{bmatrix} 1 & \begin{bmatrix} 0 & 1 \end{bmatrix} \\ \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Matrices

Desirable:

- Easy to program with
- Easy to write proofs with

Matrices

Desirable:

- Easy to program with
- Easy to write proofs with

Possibilities:

- Vectors of vectors: $\mathit{Vec}\ (\mathit{Vec}\ a\ n)\ m$
- Functions from indices: $\mathit{Fin}\ m \to \mathit{Fin}\ n \to a$
- ...

Matrices: shapes

A type for shapes:

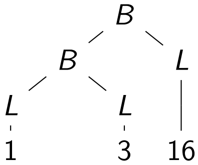
$$
\begin{array}{l}
\textbf{data}\ \mathit{Shape}\ :\ \mathit{Set}\ \textbf{where} \\
\quad L\ :\ \mathit{Shape} \\
\quad B\ :\ (s_1\ s_2\ :\ \mathit{Shape}) \rightarrow \mathit{Shape}
\end{array}
$$

Matrices: shapes

A type for shapes:

```
data Shape : Set where
  L : Shape
  B : (s1 s2 : Shape) → Shape
```

Shapes for one dimension: (a vector/row matrix)



Matrices: building blocks

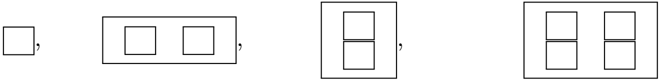
Matrices are indexed by two shapes:

$$\textbf{data}\ M\ (a : Set) : (rows\ cols : Shape) \to Set$$

Matrices: building blocks

Matrices are indexed by two shapes:

$$\textbf{data}\ M\ (a : Set) : (rows\ cols : Shape) \rightarrow Set$$



$M\ a\ L\ L$ $\quad M\ a\ L\ (B\ c_1\ c_2)$ $\quad M\ a\ (B\ r_1\ r_2)\ L$ $\quad M\ a\ (B\ r_1\ r_2)\ (B\ c_1\ c_2)$

Matrices: a datatype

$$
\begin{array}{l}
\textbf{data}\ M\ (a : Set) : (rows\ cols : Shape) \rightarrow Set\ \textbf{where} \\
\quad \begin{array}{lll}
One & : & a \rightarrow M\ a\ L\ L \\
Col & : & \{r_1\ r_2 : Shape\} \rightarrow \\
 & & M\ a\ r_1\ L \rightarrow M\ a\ r_2\ L \rightarrow M\ a\ (B\ r_1\ r_2)\ L \\
Row & : & \{c_1\ c_2 : Shape\} \rightarrow \\
 & & M\ a\ L\ c_1 \rightarrow M\ a\ L\ c_2 \rightarrow M\ a\ L\ (B\ c_1\ c_2) \\
Q & : & \{r_1\ r_2\ c_1\ c_2 : Shape\} \rightarrow \\
 & & M\ a\ r_1\ c_1 \rightarrow M\ a\ r_1\ c_2 \rightarrow \\
 & & M\ a\ r_2\ c_1 \rightarrow M\ a\ r_2\ c_2 \rightarrow \\
 & & M\ a\ (B\ r_1\ r_2)\ (B\ c_1\ c_2)
\end{array}
\end{array}
$$

Rings

A hierarchy of rings as Agda records:

- *SemiNearRing*
  $\simeq$, $+$, $\cdot$, *0* (+ is associative and commutes, 0 identity of $+$ and zero of $\cdot$, $\cdot$ distributes over $+$)
- *SemiRing*
  1 (1 identity of $\cdot$, $\cdot$ is associative)
- *ClosedSemiRing*
  an operation * with $w^* \simeq 1 + w \cdot w^*$.

Lifting matrices

We take a semi-(near)-ring and lift it to square matrices.
A lifting function *Square* for each *Shape* and ring structure.

$$\begin{aligned}
Square &: Shape \to SemiNearRing &\to SemiNearRing \\
Square' &: Shape \to SemiRing &\to SemiRing \\
Square'' &: Shape \to ClosedSemiRing &\to ClosedSemiRing
\end{aligned}$$

Lifting matrices

(Parts of) lifted equivalence:

$$
\begin{array}{lllll}
\_{\simeq_S}\_ & : \forall \{r\ c\} \to M\ s\ r\ c \to M\ s\ r\ c \to Set \\
(One\ x) & \simeq_S & (One\ x_1) & = & x \simeq_s x_1 \\
(Row\ m\ m_1) & \simeq_S & (Row\ n\ n_1) & = & (m \simeq_S n) \times (m_1 \simeq_S n_1)
\end{array}
$$

(Parts of) lifted multiplication:

$$
\begin{array}{lllll}
\_{\cdot_S}\_ & : \forall \{r\ m\ c\} \to M\ s\ r\ m \to M\ s\ m\ c \to M\ s\ r\ c \\
One\ x & \cdot_S & One\ y & = & One\ (x \cdot_s y) \\
Row\ m_0\ m_1 & \cdot_S & Col\ n_0\ n_1 & = & m_0 \cdot_S n_0 +_S m_1 \cdot_S n_1
\end{array}
$$

Proofs: reflexivity

$$
\begin{array}{lcl}
\mathit{reflS} \ : \ \forall \, \{ r \; c \} \rightarrow (X \ : \ M \; s \; r \; c) \rightarrow X \ \simeq_S \ X & & \\
\mathit{reflS} \; (\mathit{One} \; x) & = & \mathit{refl}_s \; \{x\} \\
\mathit{reflS} \; (\mathit{Row} \; X \; X_1) & = & \mathit{reflS} \; X \; , \; \mathit{reflS} \; X_1 \\
\mathit{reflS} \; (\mathit{Col} \; X \; X_1) & = & \mathit{reflS} \; X \; , \; \mathit{reflS} \; X_1 \\
\mathit{reflS} \; (Q \; X \; X_1 \; X_2 \; X_3) & = & \mathit{reflS} \; X \; , \; \mathit{reflS} \; X_1 \; , \\
 & & \mathit{reflS} \; X_2 \; , \; \mathit{reflS} \; X_3
\end{array}
$$

Closure for matrices

Computing the reflexive, transitive closure:

$$[a]^* = [a^*]$$

$$\boxed{\begin{matrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{matrix}}^* = \boxed{\begin{matrix} A_{11}^* + A_{11}^* \cdot A_{12} \cdot \Delta^* \cdot A_{21} \cdot A_{11}^* & A_{11}^* \cdot A_{12} \cdot \Delta^* \\ \Delta^* \cdot A_{21} \cdot A_{11}^* & \Delta^* \end{matrix}}$$

(with $\Delta = A_{22} + A_{21} \cdot A_{11}^* \cdot A_{12}$)

with proof that it satisfies $w^* \simeq 1 + w \cdot w^*$

Reachability example

$$\begin{tikzcd} 1 & 2 \arrow[d] \\ 3 \arrow[ur] & 4 \end{tikzcd}$$

$$\boxed{\begin{matrix} \boxed{\begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix}} & \boxed{\begin{matrix} 0 & 0 \\ 0 & 1 \end{matrix}} \\ \boxed{\begin{matrix} 0 & 1 \\ 0 & 0 \end{matrix}} & \boxed{\begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix}} \end{matrix}}^{*}$$

Reachability example

$$\begin{tikzcd}
1 & 2 \arrow[d] \\
3 \arrow[ur] & 4
\end{tikzcd}$$

$$\begin{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\ \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{bmatrix}^* = \begin{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\ \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \end{bmatrix}$$

$$\begin{tikzcd}
1 \arrow[loop above] & 2 \arrow[loop above] \arrow[d] \\
3 \arrow[loop below] \arrow[ur] \arrow[r] & 4 \arrow[loop below]
\end{tikzcd}$$

Wrapping up

Conclusions, further work, et.c.

- This matrix definition is useable...
- A more flexible matrix definition: sparse? fewer constructors?
- Automation (of proofs)!
- Generalisation to closed semi-near-ring for parsing applications.
- Agda development available at `https://github.com/DSLsofMath/FLABloM`.