Matrices in ACL2

Joe Hendrix
This talk introduces books for elementary matrix operations and theorems.

The current focus is on simplicity and creating good rewriting theorems.

Work in the immediate future is to prove the correctness of algorithms for inverting matrices, calculating determinants, and solving linear systems (i.e. solving for $x$ in $Ax = B$) using Gaussian-Jordan elimination.
Data Representation

- Matrices are represented as lists of lists with \texttt{Nil} denoting the \textit{empty matrix}.

- Although accessing a single element takes linear time instead of the constant time performance of an array-based implementation, the higher level operations should not perform asymptotically worse.

- Sample Matrix:

\[
\begin{pmatrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{pmatrix}
\]

is represented as

\[
\left( \begin{array}{ccc}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{array} \right)
\]
Basic Operations

The primary operations are implemented on top of a set of core constructors and destructors that build matrices one row or column at a time.

The Lisp definitions are immediately disabled. It would be useful if the expand hint could be modified to use logical definitions.

As there are essentially two ways of building matrices, by adding a new row via \texttt{row-cons} to the rows, or by adding a new column via \texttt{col-cons} to the columns, a number of theorems are proven relating the two constructors and the corresponding destructors \texttt{row-car}, \texttt{row-cdr}, \texttt{col-car}, \texttt{col-cdr}. 
Defined Operations

- The operations of matrix addition, subtraction, negation, transposition, and multiplication by a scalar, by a vector, and by another matrix have been defined.

- Functions for generating the identity matrix and zero matrix have also been defined.

- These operations are all implemented using the primitives described in the last slide.

- Can use guard checking to verify that matrices are of correct size in an expression.
Theorems

- Proved the basic ring properties
  - Matrix addition is associative and commutative.
  - Matrix multiplication is associative and distributes over addition.
  - Special properties of zero and identity matrices (e.g. $M + 0 = M$, $M \times 1 = M$, $1 \times M = M$, $M \times 0 = 0$, $0 \times M = 0$).
- Transpose distributes over addition and multiplication.
- Coerce expressions involving matrices into a cannonical form.
Future Work

- Gaussian-Jordan Elimination.
  - Used for solving systems of linear equations \((Ax = B)\), calculating determinates, and matrix inversion.
  - Required for any real application-level theorems.
- Research how to make definitions perform better - hopefully without making theorems more complicated outside the library itself.