Synthesis Benchmarks for Matrix Multiplication

Romain Brenguier
Université Libre de Bruxelles – Brussels, Belgium
romain.brenguier@ulb.ac.be

***Abstract*—We present benchmarks for matrix multiplication. A first category consists in a static problem, where the environment provides matrices and the controller has to correctly perform the multiplication. In the second category, the result of the previous step is kept and multiplied by a matrices which is splited between controllable and uncontrollable columns.**

I. Introduction

The original AIGER format [1] is a safety property specification used for model checking. Its extended version was introduced in 2014 by Jacobs [2] to represent safety objectives for controller synthesis. It was first used in the Synthesis Competition 2014, a satellite event of the 26th International Conference on Computer Aided Verification 2014.

The AIGER files describe synchronous circuit, with inputs that are partitionned between the controllable and uncontrollable ones, and one output that signal an error. The goal of synthesis is to control the controllable inputs, so that the error output is never set to true.

Matrix multiplication is a basic operation which has many applications in mathematics, physics, and engineering. Implementing this operation with a logical circuit of small size is important to produce hardware at small cost.

II. Static multiplication

In the first set of benchmarks, the goal is to obtain a circuit that provides a correct implementation of Boolean matrix multiplication. We consider the set of Booleans $\mathbb{B} = \{0, 1\}$ and multiplication of matrices in the Boolean ring $\langle \mathbb{B}, \vee, \wedge, 0, 1\rangle$. Formally, inputs provide matrices $A \in \mathbb{B}^{m\times n}$, $B \in \mathbb{B}^{n\times o}$, and $C \in \mathbb{B}^{m\times o}$, with the inputs for C being controllable. The controllers has to perform a correct multiplication, that is at each step the circuits checks that $A \cdot B = C$. The error output is set to true if $A \cdot B \neq C$.

In the benchmark package, the AIGER file `mult_bool_matrix_`m`_`n`_`o`.aag` encodes the mutiplication of a matrice A of dimension $m \times n$ and a matrice B of dimension $n \times o$.

III. Dynamic multiplication

We give a set of benchmarks which is an example of a dynamic system whose transition relation is determined by a matrix multiplication. In this set of benchmarks, $A_0 \in \mathbb{B}^{m\times n}$ is initialized to an abitrary matrix, then at step i, the next valuation A_{i+1} is given by the product $A_{i+1} = A_i \left(B \mid C\right)$ where $B \in \mathbb{B}^{n\times \frac{n}{2}}$ and $C \in \mathbb{B}^{n\times \frac{n}{2}}$. The valuation of B and C at each step is determined by inputs, these inputs are uncontrollable for B and controllable for C. The controller has to choose valuations such that no column of A is composed of only 0's or only 1's.

Below is an example of an execution that ends with an error (i.e. controller loses):

$$\begin{array}{ll}
A_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & B, C = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\
A_1 = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} & B, C = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\
A_2 = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} & B, C = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\
A_3 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} & \\
 & \rightarrow \text{ error: the first column of } A \\
 & \quad \text{is only composed of 0's}
\end{array}$$

The AIGER files for these benchmarks have been generated from an OCaml program that can be downloaded from the following address: https://github.com/romainbrenguier/Speculoos.

References

[1] A. Biere. Aiger format and toolbox. [Online]. Available: http://fmv.jku.at/aiger/

[2] S. Jacobs. (2014, February) Extended aiger format for synthesis (v0.1). [Online]. Available: http://www.syntcomp.org/wp-content/uploads/2014/02/Format.pdf