Emulation of Biological Networks in Reconfigurable Hardware


Aakash Chugh


Ratin Malkud


Natasa Miskov-Zivanov


Diana Marculescu

Developing computational models of cell signaling networks is critical for their understanding. However, the complexity of these models increases rapidly with the size of the network. Moreover, simulations of such models are usually executed sequentially on general purpose CPUs. The main goal of this work is to automate the generation of synthesizable Verilog code that allows for emulations of discrete, logical models of biological networks (Fig. 1). This tool will provide orders of magnitude speed-up in analyzing biological networks and gaining new insights into their organization, function and behavior.

The first stage can automatically translate description of the circuitry of a biological network to the DATA_PATH module (Fig. 2). The textual description of the network is translated using the parser that was created in Java. The next step automates the generation of the remaining files to allow hardware emulation of the model. Properties of the Boolean model such as size of the network and initial network state are used to automate the generation of the control path, library files, the data path and the testbench. Previous work that emulated models in an FPGA shown a slight discrepancy compared to software simulations. To explore causes for this discrepancy, we implemented two different random number generators (RNGs), Linear Feedback Shift Register (LFSR) and Cellular Automata Shift Register (CASR), and compared results for the two implementations (Fig. 2). We have also implemented a graphical user interface to select the random number generator at runtime. The initial seed for the random number generators are derived from the network at hand, thus ensuring a unique initial seed for each network. Future work will include automating the generation of synthesizable Verilog code for multi-level Boolean models of biological networks, and connecting this automation tool with the model design automation within a single framework.

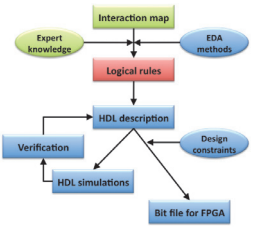

Fig. 1. Methodology.

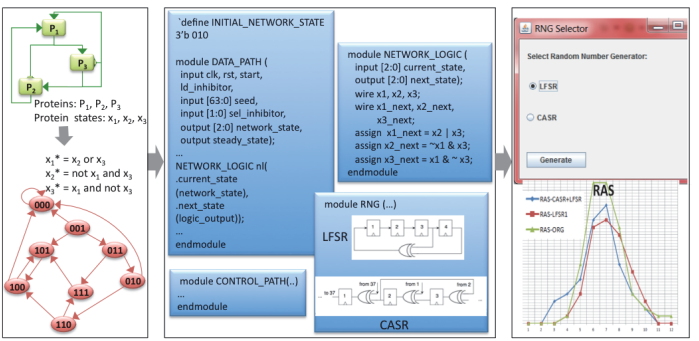

Fig. 2. Example of interaction map and logical model, translation into Verilog, and GUI snapshot.