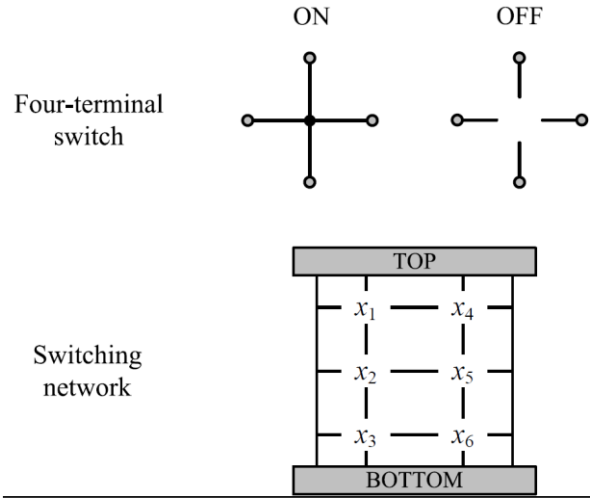
# OPTIMAL SYNTHESIS TOOLS

The tools synthesize Boolean functions with optimal size of arrays of four-terminal switches. A four-terminal switch shown in the top part of the below figure. The four terminals of the switch are all either mutually connected (ON) or disconnected (OFF). We consider arrays of four-terminal switches arranged in rectangular array lattices with N rows and M columns (area: N×M). An example is shown in the bottom part of the figure. Each switch is controlled by a Boolean literal. If the literal takes the value 1 (0) then corresponding switch is ON (OFF). The Boolean function for the lattice evaluates to 1 iff there is a closed path between the top and bottom edges of the lattice. The Boolean function is computed by taking the sum of the products of the literals along each path. These products are $x_1x_2x_3$, $x_1x_2x_5x_6$, $x_4x_5x_2x_3$, and $x_4x_5x_6$. We conclude that this lattice of four-terminal switches implements the Boolean function $x_1x_2x_3 + x_1x_2x_5x_6 + x_2x_3x_4x_5 + x_4x_5x_6$.




We have two optimal synthesis tools Tool-1 and Tool-2 developed in Matlab and Python, respectively. While Tool-1 is completely new, Tool-2 is the highly improved version of the tool previously presented in the literature (Gange, G., Søndergaard, H., & Stuckey, P. J. (2014). Synthesizing optimal switching lattices. ACM Transactions on Design Automation of Electronic Systems (TODAES), 20(1), 6) .

Both tools aim to synthesize a given target Boolean functions with an optimal array/lattice size. It means that below the found sizes there is no synthesis solution. For example, if tools find sizes N=4 and M=4 (area 16), then synthesizing the given function with sizes N=3 and M=5 (area 15) impossible.

Tools are given in separate files having readme files that give brief information of the tools as well as running instructions.