Fast Hierarchical NPN Classification

Ana Petkovska, Mathias Soeken, Giovanni De Micheli, Paolo Ienne, and Alan Mishchenko

August 30, 2016
Lausanne, Switzerland
Negation-Permutation-Negation (NPN) Classification

\[ f(x, y, z) = x(y + z) \]

NPN equivalent
permute \( x \) and \( y \)
negate \( x \)

\[ g_1(x, y, z) = y(\bar{x} + z) \]
\[ g_2(x, y, z) = x(z + y) \]
\[ g_3(x, y, z) = \bar{x} + \bar{y}z \]
\[ g_4(x, y, z) = \bar{y} + xz \]

# functions > # NPN classes
For building compact libraries of circuit structures or cuts produced by different tools and benchmarks.

- W. Yang et al., “Lazy man’s logic synthesis”, ICCAD’12

For retrieving an optimal structure for a given Boolean function from a library.

- A. Mishchenko et al., “Combinational and sequential mapping with priority cuts”, ICCAD’07
- A. Mishchenko et al., “Technology mapping into general programmable cells”, FPGA’15

For matching Boolean functions of millions of enumerated structural cuts against a library of cells used to implement the design.
Algorithms for NPN Classification

Existing algorithms
Discard intermediate results

Our algorithm
Keep intermediate results as a hierarchy of classes
Experimental Results: Runtime Comparison

Classification of full-DSD functions

<table>
<thead>
<tr>
<th>#Inputs</th>
<th>#Func</th>
<th>State-of-the-art Heuristic</th>
<th>Hierarchical Approach (Heuristic)</th>
<th>Exhaustive Exact Algorithm</th>
<th>Hierarchical Approach (Exact)</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>1M</td>
<td>0.28 s</td>
<td>0.10 s</td>
<td>33 min</td>
<td>0.20 s</td>
</tr>
<tr>
<td>8</td>
<td>1M</td>
<td>0.80 s</td>
<td>0.22 s</td>
<td>&gt; 12 h</td>
<td>59.34 s</td>
</tr>
<tr>
<td>10</td>
<td>100K</td>
<td>0.19 s</td>
<td>0.09 s</td>
<td>&gt; 12 h</td>
<td>2.56 h</td>
</tr>
</tbody>
</table>

3.7x faster
max. 160 MB more memory

exact classification
for small functions
in seconds