Representation of Boolean Functions with ROBDDs

Aleš Časar
Staneta Rozmana 4
69000 Murska Sobota
Slovenia
e-mail: casar@uni-mb.si

Robert Meolic
Mali Bakovci 25
69000 Murska Sobota
Slovenia
e-mail: meolic@uni-mb.si

Mentor:
Dr. Zmago Brezočnik
Jezernikova 8
62342 Ruše
Slovenia
e-mail: brezocnik@uni-mb.si

University of Maribor
Faculty of technical sciences
Electrical engineering, computer science and information technology
Smetanova 17
62000 Maribor
Slovenia
Problem description:

- in computer aided design (CAD) and verification of circuits the system is mostly described in Boolean algebra,
- time and space complexity are exponential in the worst case,
- Reduced Ordered Binary Decision Diagram (ROBDD) is currently the best data structure for representation of Boolean functions.

Mathematical background:

- Shannon’s decomposition theorem: $f = x_i \cdot f|_{x_i=1} + \overline{x_i} \cdot f|_{x_i=0}$,
- If-Then-Else operator: $\text{ITE}(f, g, h) = f \cdot g + \overline{f} \cdot h$.

What have we done?

- creation of a ROBDD from a string of characters and its maintenance,
- logical operations on functions ($f <\text{op}> g$),
- equivalence testing ($f = g$),
- tautology checking ($f = 1$).
Different kinds of BDDs:

\[ f = a \cdot c + \overline{b} \cdot d + \overline{b} \cdot c + \overline{a} \cdot b \cdot \overline{c} \cdot \overline{d} \]

<table>
<thead>
<tr>
<th></th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>f</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ a \cdot c + \overline{b} \cdot d + \overline{b} \cdot c + \overline{a} \cdot b \cdot \overline{c} \cdot \overline{d} = \]

\[ = \overline{a} \cdot (\overline{b} \cdot d + \overline{b} \cdot c + b \cdot \overline{c} \cdot \overline{d}) + a \cdot (\overline{b} \cdot d + c) \]
Node structure:

Unique-table structure:

An example of a unique-table:

\[ f = a \cdot c + a \cdot d + \overline{b} \cdot d + \overline{a} \cdot b \cdot \overline{d} \]
Examples of simple functions:

\[
\begin{align*}
F &= 1 \\
F &= 0 \\
F &= a \\
F &= \overline{a}
\end{align*}
\]

Functions share ROBDD:

\[
\begin{align*}
E &= b \\
F &= a + b \\
G &= a + c \\
H &= (a + b) \cdot (a + c) = a + b \cdot c
\end{align*}
\]
Creation of ROBDD: \((a + b) \cdot (a + c)\)

1. create \(a\) \(\langle A \rangle\)
2. create \(b\) \(\langle B \rangle\)
3. \(F = a + b\) \(\langle \text{ite}(A, 1, B) \rangle\)
4. create \(a\) \(\langle \text{exist} \rangle\)
5. create \(c\) \(\langle C \rangle\)
6. \(G = a + c\) \(\langle \text{ite}(A, 1, C) \rangle\)
7. \(H = (a + b) \cdot (a + c)\) \(\langle \text{ite}(F, G, 0) \rangle\)
Garbage Collection:
<table>
<thead>
<tr>
<th>circuit name</th>
<th>res1</th>
<th>res2</th>
<th>res3</th>
<th>alphabetical</th>
<th>reordered</th>
</tr>
</thead>
<tbody>
<tr>
<td>add1</td>
<td>0.93</td>
<td>0.28</td>
<td>1.5</td>
<td>0.27</td>
<td>0.22</td>
</tr>
<tr>
<td>add2</td>
<td>1.40</td>
<td>1.97</td>
<td>2.2</td>
<td>0.94</td>
<td>0.79</td>
</tr>
<tr>
<td>add3</td>
<td>3.18</td>
<td>15.20</td>
<td>3.3</td>
<td>8.92</td>
<td>1.41</td>
</tr>
<tr>
<td>add4</td>
<td>7.14</td>
<td>11.50</td>
<td>5.5</td>
<td>1084.92</td>
<td>13.79</td>
</tr>
<tr>
<td>mul05</td>
<td>132.32</td>
<td>2.15</td>
<td>2.5</td>
<td>1.33</td>
<td>1.30</td>
</tr>
<tr>
<td>mul06</td>
<td>2521.26</td>
<td>7.52</td>
<td>4.9</td>
<td>4.63</td>
<td>4.63</td>
</tr>
<tr>
<td>mul07</td>
<td>......</td>
<td>28.30</td>
<td>12.3</td>
<td>15.64</td>
<td>15.26</td>
</tr>
<tr>
<td>mul08</td>
<td>......</td>
<td>88.90</td>
<td>40.0</td>
<td>59.68</td>
<td>60.59</td>
</tr>
<tr>
<td>alupla20</td>
<td>1.42</td>
<td>0.70</td>
<td>1.5</td>
<td>0.45</td>
<td>0.36</td>
</tr>
<tr>
<td>alupla21</td>
<td>1.87</td>
<td>1.61</td>
<td>1.8</td>
<td>0.74</td>
<td>0.65</td>
</tr>
<tr>
<td>alupla22</td>
<td>2.48</td>
<td>6.45</td>
<td>2.1</td>
<td>1.79</td>
<td>2.04</td>
</tr>
<tr>
<td>alupla23</td>
<td>2.00</td>
<td>15.00</td>
<td>2.1</td>
<td>3.03</td>
<td>2.82</td>
</tr>
<tr>
<td>alupla24</td>
<td>1.53</td>
<td>2.90</td>
<td>1.8</td>
<td>0.91</td>
<td>0.99</td>
</tr>
<tr>
<td>rip02</td>
<td>0.13</td>
<td>0.02</td>
<td>1.0</td>
<td>0.03</td>
<td>0.04</td>
</tr>
<tr>
<td>rip04</td>
<td>0.75</td>
<td>0.03</td>
<td>1.2</td>
<td>0.15</td>
<td>0.11</td>
</tr>
<tr>
<td>rip06</td>
<td>3.03</td>
<td>0.12</td>
<td>1.4</td>
<td>0.52</td>
<td>0.24</td>
</tr>
<tr>
<td>rip08</td>
<td>9.14</td>
<td>0.17</td>
<td>1.7</td>
<td>2.83</td>
<td>0.38</td>
</tr>
<tr>
<td>ztwaalf1</td>
<td>5.10</td>
<td>0.28</td>
<td>1.3</td>
<td>0.15</td>
<td>0.18</td>
</tr>
<tr>
<td>ztwaalf2</td>
<td>3.63</td>
<td>0.22</td>
<td>1.2</td>
<td>0.16</td>
<td>0.19</td>
</tr>
<tr>
<td>dk17</td>
<td>1.50</td>
<td>0.58</td>
<td>1.9</td>
<td>0.29</td>
<td>0.26</td>
</tr>
<tr>
<td>dk27</td>
<td>0.67</td>
<td>0.25</td>
<td>1.6</td>
<td>0.19</td>
<td>0.18</td>
</tr>
</tbody>
</table>

**res1**: directed weighted backtracking in LISP (*Frank Vlach*)\(^1\)

**res2**: BDDs in Prolog (*H. Simonis and T. Le Provost*)\(^1\)

**res3**: BDDs in C (*S. Minato, N. Ishiura and S. Yajima*)\(^1\)

**alphabetical**: our results – alphabetical order of variables

**reordered**: our results – variables reordered

---

\(^1\)These results were presented at *IMEC-IFIP International Workshop on Applied Formal Methods For Correct VLSI Design* in Houthalen, Belgium, November 1989.
Implementation:

- generally useful and efficient program package on VAX 8800 computer,
- 3500 lines of source Pascal program.

Our contributions to BDDs:

- an improved garbage collection,
- an algorithm for composition of functions,
- an influence of some parameters on efficiency: size of tables, when to start garbage collection, …

Application:

- program package is used in computer aided verification of combinational circuits,
- it will be used in design of circuits and analysis of distributed systems,
- this program package is intended to be a part of an integrated CAD tool for formal system design.