Boolean Subalgebra: Interval of 2-Variable Function Lattice

The 3-Atom element \( x + y = xy' + xy + x'y \) is **ONE** of subalgebra

2-Atom elements are atoms of subalgebra

1-Atom element \( xy' \) is **ZERO** of subalgebra

A larger Subalgebra (Interval) of 2-Variable Function Lattice

**ONE** of algebra (4-Atoms) is also one of this subalgebra

2 levels means \( 2^3 = 8 \) elts

2-Atom elements are atoms of subalgebra

1-Atom element is **ZERO** of subalgebra

More On Counting

An algebra (or subalgebra) with \( n+1 \) levels has exactly \( 2^n \) elements, because \( n+1 \) levels implies \( n \) atoms
**Boolean Difference (Sensitivity)**

A Boolean function $f$ depends on $a$ if and only if $f_a \neq f_{a'}$. Thus

$$\frac{\partial f}{\partial a} = f_a \oplus f_{a'}$$

is called the Boolean Difference, or Sensitivity of $f$ with respect to $a$.

---

**Why Large Boolean Algebras?**

- When you design an optimal circuit, each gate must be optimized with respect to its Don’t Cares.
- Because of Don’t Cares, 4 functions of $(x, y)$ are equivalence preserving replacements for gate $g$.
- Optimal Design: pick best such replacement.

---

**Don’t Cares**

- $xy \in D^{Sat}$ if and only if $f^x = x$, $f^y = y \Rightarrow \partial z / \partial w = 0$ for all possible $x$.
- $xy \in D^{Obs}$ if and only if local input $xy$ never occurs.

The complete don’t care set for gate $g$ is

$$D^g = D^{Sat} + D^{Obs}$$

---

**Example:**

$$f = abc + a'bc$$

$$\frac{\partial f}{\partial a} = f_a \oplus f_{a'} = (bc) \oplus (bc) = 0$$

$$\frac{\partial f}{\partial b} = f_b \oplus f_{b'} = (ac) \oplus (a'c) = c$$

Note the formula depends on $a$, but the implied function does not.
**Don’t Cares**

For this circuit, global input combination 10 sets 
\[ t' = uv' = 1 \]
which makes \( z \) insensitive to \( w \). However, local input pair 10 \((xy')\) is **NOT** don’t care, since \( u'v' \) also gives \( xy' \), and in this case \( t_{u'v'} = 1 \).

**Computing ALL Don’t Cares**

\[ D^g = D^{Sat} + D^{Obs} = x'y' \]
\[ g = xy' + x'y' (+x'y') \rightarrow x' + y' \]

Thus the exclusive OR gate can be replaced by a NAND.

**Local inputs**

\[ x = u' + v', \quad y = v \]
\((y = 0) \Rightarrow (x = 1)\)

For this circuit, local input combinations \((xy')\) \((x = 0, y = 0)\) do not occur. That is, the local minterm \(x'y'\) is don’t care.

**Computing ALL Don’t Cares**

\[ D^g = D^{Sat} + D^{Obs} = x'y' \]
\[ g = xy' + x'y' (+x'y') \rightarrow x' + y' \]

Similarly \( x'y \in D^{Obs} \) if and only for every row \( u,v \) such that \( f^x(u,v) = x \) and \( f^y(u,v) = y \),
\[ \frac{\partial z}{\partial w} = z_w \oplus z_w' = 0 \]

Here 10 \((xy')\) is **NOT** don’t care since \( \frac{\partial z}{\partial w} = 1 \) in the first row.
**Incompletely Specified Functions**

Suppose we are given a Boolean Function $g$ and a don’t care set $D$. Then the triple 
\[(f,d,r)\]
where $f = gD'$, $d = D$, and $r = (f + D)'$ is called an **INCOMPLETELY SPECIFIED FUNCTION**.

Note $f + d + r = gD' + D + (g + D)' = 1$.

**Don’t Cares and Function Subalgebras**

Suppose we are given a Boolean Function $f$ and a don’t care set $D$. Then any function in the interval (subalgebra) 
\[[f_L, f_U] = [fD', f + D]\]
is an acceptable replacement for $f$ in the environment that produced $D$. Here $fD'$ is the $0$ of the subalgebra and $f + D$ is the $1$.

---

**Intervals and Don’t Cares**

\[D = D^{Obs} + D^{Sat} = \overline{x'y'}\]
\[L = g - D = gD' = (x'y + xy')(x + y) = (x'y + xy') = g\]
\[U = g + D = (x'y + xy') + x'y' = \overline{x'y}\]