Technische Informatik: Diskussion von 3.2

22. November 2013

Discussion of the exercise 3.2 (Air conditioner)

The description of the automatic part of the system contained four parts, all of which had to be satisfied simultaneously (so there is an AND between each requirement).

- Die Klimaanlage ist an, wenn der Temperatursensor an ist, ausser der Windsensor ist an und der Feuchtigkeitssensor ist aus.
- Die Klimaanlage ist auch an, wenn der Feuchtigkeitssensor an ist, ausser der Temperatursensor ist aus und der Windsensor ist an.
- Die Klimaanlage ist aus, wenn der Windsensor an ist, ausser der Temperatursensor und der Feuchtigkeitssensor sind beide gleichzeitig an.
- In allen anderen Faellen ist die Klimaanlage aus.

We used the following rule to translate the word "ausser" into boolean expressions: "x ausser y" has been translated into "$x \wedge \neg y$". In the next step of the interpretation, we paraphrase sentences of type "b, wenn a" into "Wenn a dann b". For variables t (Temperature), h (Humidity), w (Wind), a (Automatic part of the air conditioner control system) this gives us:

- Wenn $t \wedge \neg(w \wedge \neg h) = t\bar{w} + th =: e_1$ dann a.
- Wenn $h \wedge \neg(\neg t \wedge w) = ht + h\bar{w} =: e_2$ dann a.
- Wenn $w \wedge \neg(t \wedge h) = w\bar{t} + w\bar{h} =: e_3$ dann $\neg a$.
- Wenn $\neg e_1 \wedge \neg e_2 \wedge \neg e_3$ dann $\neg a$.

Now we transform "Wenn ... dann ..." sentences into logical implications:

- $e_1 \to a$
- $e_2 \to a$
- $e_3 \to \neg a$
- $\neg e_1 \wedge \neg e_2 \wedge \neg e_3 \to \neg a$

We simplify the first two clauses using the following rule for the implications:

$$(a \to z) \wedge (b \to z) = (\bar{a} + z)(\bar{b} + z) = \bar{a}\bar{b} + \bar{a}z + \bar{b}z + z = \neg(a \vee b) + z = (a \vee b) \to z$$

into $e_{12} := th + t\bar{w} + h\bar{w}$ and obtain:

- $e_{12} \to a$
- $e_3 \to \neg a$
- $\neg e_{12} \wedge \neg e_3 \to \neg a$

We now make the observation that e_{12} and e_3 are never both 1 at the same time: $e_{12} \wedge e_3 = 0$ (pointwise, as functions $\mathbb{B}^3 \to \mathbb{B}$). Let us for a moment call such functions *nonconflicting*. That means that even if the implications of e_{12} and e_3 are opposite, we still can construct a function a which fulfills all these implications:

h	t	w	e_{12}	e_3
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	1	0
1	1	1	1	0

In particular, this means that there is at least no ambiguity about the precedence of the rules: the rules do not conflict, and therefore their order is irrelevant. How can we combine these implications into one single function a?

Lemma Let $x, y, a : \mathbb{B}^n \to \mathbb{B}$ be some boolean functions with x, y nonconflicting (i.e. $x \wedge y = 0$) and

$$(x \to a) \wedge (y \to \neg a) \wedge ((\neg x \wedge \neg y) \to \neg a).$$

Claim: a = x

Proof: We show that $(x \to a)$ and $(a \to x)$, from which the equality easily follows:

$$(a \to x) \wedge (x \to a) = (\bar{a} + x)(\bar{x} + a) = \bar{a}\bar{x} + 0 + 0 + ax = a \leftrightarrow x.$$

The $(x \to a)$ clause is already assumed, there is nothing to show. For $(a \to x)$ we turn the two other assumed applications around, negating the arguments (using de Morgan in second case):

$$a \to \neg y$$
$$a \to (x \vee y)$$

Now we can combine the right sides of these two implications:

$$\bar{y}(x + y) = \bar{y}x = \bar{y}x + 0 \overset{\text{nonconflicting}}{=} \bar{y}x + xy = x$$

obtaining $(a \to x)$. Thus the equality holds.

With this lemma, it is immediately obvious that the implementation of the automatic part of the air conditioner is just $e_{12} = th + t\bar{w} + h\bar{w}$, the e_3 OFF-part can be ignored alltogether, at least as long as it is nonconflicting with the ON-part.

Notice the $\bar{y}x = x \wedge \neg y$ expression in the last line of the proof. This expression occurred in our original formulation, back then we simply did not notice that the conjunction with the $\bar{y}$ from the $(y \to \neg a)$ requirement has no effect, and calculated everything by brute force[1]:

[1]Unfortunately, we used a “\not” instead of a “\neg” in our TeX, this is why there was a cancelled parenthesis instead of a negation sign in the first line of the calculation.

$$
\begin{aligned}
a &= ((t \wedge \neg(w \wedge \neg h)) \vee (h \wedge \neg(\neg t \wedge w))) \wedge \not(w \wedge \neg(t \wedge h)) \\
&\overset{DeMorgan}{=} ((t \wedge (\neg w \vee h)) \vee (h \wedge (t \vee \neg w))) \wedge (\neg w \vee (t \wedge h)) \\
&\overset{distr,idempot}{=} (t\bar{w} + th + ht + h\bar{w})(\bar{w} + th) \\
&\overset{distr,assoc}{=} t\bar{w} + th\bar{w} + h\bar{w} + th\bar{w} + th + th\bar{w} \\
&\overset{absorp}{=} t\bar{w} + th + h\bar{w}
\end{aligned}
$$

This is exactly the function represented by the table:

t	h	w	e_{12}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

However, if we replace the AND by OR in $e_{12} \wedge \neg e_3$, we obtain a different function:

t	h	w	$e_3 \vee \neg e_3 = th + \bar{w}$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

The difference is in the 0-th row: it's exactly the case where none of the rules 1-3 matches. By implementing it with an OR, we change the default from OFF to ON, in contradiction to the fourth requirement about the default state. Therefore, we cannot just flip AND to OR.

Cancellation of non-epimorphisms.

A word of caution on cancelling non-surjective[1] functions. Let $f_1, f_2, f_3 : \mathbb{B}^3 \to \mathbb{B}$ be some boolean functions. From the unequality of $\phi_\wedge : (A, B, C) \mapsto (A \vee B) \wedge C$ and $\phi_\vee : (A, B, C) \mapsto (A \vee B) \vee C$ it does *not* follow that

$$\phi_\vee(f_1, f_2, f_3) \neq \phi_\wedge(f_1, f_2, f_3).$$

One still has to look carefully at the function f.

Example from the first exercise sheet, using the consensus theorem: $f_1(x, y, z) = xy$, $f_2(x, y, z) = \bar{x}z$, $f_3(x, y, z) = xy + \bar{x}z + yz$. Althought $\phi_\vee \neq \phi_\wedge$, it still holds: $\phi_\vee \circ f = \phi_\wedge \circ f$. This happens every time when the range of f and the set $\{x : \phi_\wedge(x) \neq \phi_\vee(x)\}$ are disjoint. The function f just happens not to take any values where $\phi_\vee$ and $\phi_\wedge$ differ:

[1] It was of course *surjective*, not *injective*, always mix them up, sorry.

x	y	z	f_1	f_2	f_3	$\phi_\wedge$	$\phi_\vee$	$\phi_\wedge \circ f$	$\phi_\vee \circ f$
0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	1	0	1	1	1
0	1	0	0	0	0	0	1	0	0
0	1	1	0	1	1	1	1	1	1
1	0	0	0	0	0	0	1	0	0
1	0	1	0	0	0	1	1	0	0
1	1	0	1	0	1	0	1	1	1
1	1	1	1	0	1	1	1	1	1

All this, as it turned out, had no connection to the actual problem, because in the case of the air conditioner, flipping AND to OR yielded a different function.