# 1. The Robbins Conjecture—A Brief History

- In 1933, Huntington gave this 3-axiom basis for Boolean algebra:

| | |
|---|---|
| $x \oplus y = y \oplus x$ | [Commutativity of $\oplus$] |
| $(x \oplus y) \oplus z = x \oplus (y \oplus z)$ | [Associativity of $\oplus$] |
| $\overline{\overline{x} \oplus y} \oplus \overline{\overline{x} \oplus \overline{y}} = x$ | [Huntington equation] |

- Shortly thereafter, Robbins conjectured that if the Huntington equation were replaced with the following equation (which, in the context of Boolean algebra, is *equivalent* to the Huntington equation):

$$\overline{\overline{x \oplus y} \oplus \overline{x \oplus \overline{y}}} = x \qquad \text{[Robbins equation]}$$

then the resulting axiomatization would *also* be characteristic of Boolean algebra. This is *the Robbins conjecture*. Robbins and Huntington could not find a proof of it. The problem was later studied by Tarski and his students (they couldn't prove it either).

- In 1992, almost 60 years later, Winker (with the help of Argonne's automated theorem provers) showed that the surprisingly weak:

$$(\exists C)(\exists D)\,[\overline{C \oplus D} = \overline{C}] \qquad \text{[Winker condition]}$$

is sufficient to make any Robbins algebra Boolean.

- In early October, 1996, Bill McCune and his Argonne Lab team found a proof of the Winker condition from the Robbins axioms, using their automated theorem prover **EQP**. This finally established the truth of the Robbins conjecture. Unfortunately, the 16–line **EQP** proof object is not terribly illuminating. It is highly complex and dense, and quite difficult to follow. Indeed, it is virtually impossible to work through the proof entirely "by hand". That's where *Mathematica* comes in…

# 2. The Puzzling EQP Proof Object

- There are two main problems with discovering the inner workings of the **EQP** proof. (1) The **EQP** proof object doesn't give us any information about *precisely which* substitution strategies were used during the complicated derivation (it only says which previous *lines* were used to derive each line), and (2) **EQP**'s 1-dimensional prefix notation for the "negation" operator "$n(\bullet)$" is quite difficult to parse for complex, nested formulas of the kind seen in the proof.

- There isn't much we can do about (1). In fact, our challenge is to *figure out* which substitution strategies were involved in the computer proof. As for problem (2), this is where *Mathematica* comes in handy. *Mathematica*'s 2-dimensional overbar notation $\overline{\bullet}$ is much more informative, concerning the structure of complex nested formulas.

- To get a feel for how the 2-D *Mathematica* 3 notation improves our ability to see patterns in complex expressions, consider the first line of the **EQP** proof object (*i.e.*, the Robbins equation) and the second line of the **EQP** proof object — in **EQP** notation *vs*. *Mathematica* 3 notation.

Line 1 (**EQP**). $n\,(n(x \oplus y) \oplus n(x \oplus n(y))) = x$

Line 2 (**EQP**). $n\,(y \oplus n(y \oplus n(x) \oplus n(x \oplus y))) = n(x \oplus y)$

Line 1 (*Mathematica* 3). $\overline{\overline{x \oplus y} \oplus \overline{x \oplus \overline{y}}} = x$

Line 2 (*Mathematica* 3). $\overline{y \oplus \overline{y \oplus \overline{x} \oplus \overline{x \oplus y}}} = \overline{x \oplus y}$

Notice how much more visually informative the *Mathematica* 3 versions are. This improved clarity afforded by the *Mathematica* 3 notation is just the beginning of *Mathematica*'s usefulness…

# 3. *Mathematica* to the Rescue: Part I—Encoding the Robbins Axioms in *Mathematica* 3

- I explain below how I figured out the first step of the **EQP** proof using *Mathematica* (for the other 14 steps, consult my *Robbins Algebra Web Page*, see page 6 below for the URL). This first step is the most important step in the proof, and it paves the way for the proof's completion. First, I introduce three useful *Mathematica* constructs.

- The key to cracking the computer proof is effectively encoding the three Robbins axioms in *Mathematica* 3. To this end, we evaluate the following three *Mathematica* 3 expressions:

*In[1]:=* 

$$\texttt{Attributes[CirclePlus] = \{Orderless, Flat\};}$$

$$\texttt{RF[x\_, y\_] :=}\ \overline{\overline{x \oplus y} \oplus \overline{x \oplus \overline{y}}};$$

$$\texttt{RR :=}\ \left\{\overline{\overline{x\_\_ \oplus y\_\_} \oplus \overline{x\_\_ \oplus \overline{y\_\_}}} \to x\right\};$$

- By telling *Mathematica* that ⊕ is **Orderless** and **Flat**, we insure that *Mathematica* will *automatically* take into account the commutativity and associativity of ⊕ in all of its symbolic operations. This allows us to focus on the more interesting role that the Robbins equation plays in the proof. (**EQP** also makes *implicit* use of commutativity and associativity, without explaining how they are used.)

- Encoding the Robbins equation is more important, and not as easy. The function **RF[x,y]** (which is equivalent to **x**, for all **y**) lets us generate arbitrary Robbins-equivalents of any formula **x**. And, the replacement rule **RR** automatically simplifies (some) expressions containing instances of the Robbins pattern (see page 5 for some subtleties in defining **RR**). These simple *Mathematica* constructs prove to be very powerful tools for aiding our comprehension of the **EQP** proof.

# 4. *Mathematica* to the Rescue: Part II —Deciphering the First Step of the Proof

- Line 2 of the proof says that $\overline{x \oplus y} = \overline{y \oplus \overline{y \oplus \bar{x} \oplus \overline{x \oplus y}}}$. We know this must follow from the Robbins equation (*i.e.*, Line 1). The only way to go from the Robbins equation to Line 2 is by creating a Robbins–equivalent of $\overline{x \oplus y}$ using **RF**[$\overline{\mathbf{x} \oplus \mathbf{y}}$, **y**], for *some* **y**. The question is *which* **y** is such that **RF**[$\overline{\mathbf{x} \oplus \mathbf{y}}$, **y**] yields $\overline{y \oplus \overline{y \oplus \bar{x} \oplus \overline{x \oplus y}}}$?

- As with any good proof-theoretic question of this kind, the answer involves a neat trick. In this case, the relevant trick is very difficult to figure out (or even *see*) without *Mathematica*'s help.

*In[2]:=* **RF**[$\overline{\mathbf{x} \oplus \mathbf{y}}$, $\bar{\mathbf{x}} \oplus \mathbf{y}$]

*Out[2]=* $\overline{\overline{\overline{x \oplus y} \oplus \overline{y \oplus \bar{x}}} \oplus \overline{y \oplus \bar{x} \oplus \overline{x \oplus y}}}$

- We can clearly see that **RF**[$\overline{\mathbf{x} \oplus \mathbf{y}}$, $\bar{\mathbf{x}} \oplus \mathbf{y}$] contains *two* distinct instances of the Robbins pattern. By definition, the *entire* formula is an instance of the Robbins pattern (equivalent to $\overline{x \oplus y}$). But, so is the *sub-expression* $\overline{\overline{x \oplus y} \oplus \overline{y \oplus \bar{x}}}$, which is equivalent to $y$. Using **RR**, we can *automatically* simplify this sub-expression, yielding the desired result.

*In[3]:=* **RF**[$\overline{\mathbf{x} \oplus \mathbf{y}}$, $\bar{\mathbf{x}} \oplus \mathbf{y}$] /. **RR**

*Out[3]=* $\overline{y \oplus \overline{y \oplus \bar{x} \oplus \overline{x \oplus y}}}$

- (Where's Waldo?) Can you to find the two distinct instances of the Robbins pattern, in the **EQP** rendition of **RF**[$\overline{\mathbf{x} \oplus \mathbf{y}}$, $\bar{\mathbf{x}} \oplus \mathbf{y}$]?

*In[4]:=* **TraditionalForm**[**RF**[$\overline{\mathbf{x} \oplus \mathbf{y}}$, $\bar{\mathbf{x}} \oplus \mathbf{y}$] //. {$\overline{\mathbf{x\_\_}}$ → **n**[**x**]}]

*Out[4]//TraditionalForm=*

$$n(n(n(x \oplus y) \oplus n(y \oplus n(x))) \oplus n(y \oplus n(x) \oplus n(x \oplus y)))$$

# 5. Some Implementation Notes on RR

- **RR** is useful for illustrating the first step of the **EQP** proof. But, one should *not* think of **RR** as a "general inverse" of **RF**[**x, •**]. In fact, **RR** only works effectively for *certain kinds* of instances of the Robbins pattern. To wit (can you explain **RR**'s behavior in these cases?):

```
In[5]:= RF[a, b] /. RR // TraditionalForm
        RF[a ⊕ b, c] /. RR
        RF[c, a ⊕ b] /. RR // TraditionalForm
        RF[a ⊕ b, c ⊕ d] /. RR // TraditionalForm
```

Out[5]//TraditionalForm=

$$a$$

Out[6]= Sequence[a, b]

Out[7]//TraditionalForm=

$$\overline{\overline{c \oplus \overline{a \oplus b}} \oplus \overline{a \oplus b \oplus c}}$$

Out[8]//TraditionalForm=

$$\overline{\overline{a \oplus b \oplus \overline{c \oplus d}} \oplus \overline{a \oplus b \oplus c \oplus d}}$$

- **RR** can only be counted on to correctly simplify **RF**[**x, y**] when **x** and **y** are both of length 1 (as in the first case above). The following table summarizes the behavior of the four possible variants of **RR**. Together, these 4 variants provide a *general* "inverse" of **RF[x,•]**.

Out[11]//TableForm=

| | | | | |
|---|---|---|---|---|
| | RR | $\{\overline{\overline{x\_ \oplus y\_\_} \oplus \overline{x\_ \oplus \overline{y\_\_}}} \to x\}$ | $\{\overline{\overline{x\_\_ \oplus y\_} \oplus \overline{x\_\_ \oplus \overline{y\_}}} \to x\}$ | $\{\overline{\overline{x\_ \oplus y\_} \oplus \overline{x\_ \oplus \overline{y\_}}} \to x\}$ |
| RF[a,b] | a | $\overline{\overline{a \oplus b} \oplus \overline{a \oplus \overline{b}}}$ | CirclePlus[a] | $\overline{\overline{a \oplus b} \oplus \overline{a \oplus \overline{b}}}$ |
| RF[c,a⊕b] | $\overline{\overline{c \oplus \overline{a \oplus b}} \oplus \overline{a \oplus b \oplus c}}$ | c | $\overline{\overline{c \oplus \overline{a \oplus b}} \oplus \overline{a \oplus b \oplus c}}$ | CirclePlus[c] |
| RF[a⊕b,c] | {a, b} | $\overline{\overline{a \oplus b \oplus c} \oplus \overline{a \oplus b \oplus \overline{c}}}$ | a⊕b | $\overline{\overline{a \oplus b \oplus c} \oplus \overline{a \oplus b \oplus \overline{c}}}$ |
| RF[a⊕b,c⊕d] | $\overline{\overline{a \oplus b \oplus \overline{c \oplus d}} \oplus \overline{a \oplus b \oplus c \oplus d}}$ | {a, b} | $\overline{\overline{a \oplus b \oplus \overline{c \oplus d}} \oplus \overline{a \oplus b \oplus c \oplus d}}$ | a⊕b |

# 6. Getting More Information about Automated Reasoning, the Robbins Problem, and my *Mathematica* 3 Reconstruction

## ■ My *Robbins Algebra Web Page*

● Check-out my *Robbins Algebra Page*. It contains a *Mathematica* 3 notebook explaining my entire reconstruction of the **EQP** proof of the Winker condition. The best way to figure out how the proof works is to fiddle around with my *Mathematica* notebook yourself! You'll also find links there to Bill McCune's highly informative automated reasoning web pages at Argonne National Laboratory:

**http://polyglot.lss.wisc.edu/philosophy/PEOPLE/TA/BF/ROBBINS.HTM**

## ■ Bibliography

Huntington, E.V., 1933a, "New sets of independent postulates for the algebra of logic", *Transactions of the American Mathematical Society* **35**, pages 274–304

Huntington, E.V., 1933b, "Boolean algebra: A correction", *Transactions of the American Mathematical Society* **35**, pages 557–558.

Kolata, G., 1996, "Computer Math Proof Shows Reasoning Power", *The New York Times*, December 10, 1996.

McCune, W., 1997, "Solution of the Robbins Problem," forthcoming in the *Journal of Automated Reasoning*.

Tarski, A., *et al*, 1971, *Cylindric Algebras: Part I*, North-Holland.

Winker, S., 1992, "Absorption and idempotency criteria for a problem in near-Boolean algebras", *Journal of Algebra* **153**(2), pages 414–423.

Wos, L., *et al*, 1992, *Automated Reasoning: Introduction and Applications*, *Second Edition*, McGraw Hill.