Refutation of the hoop and pocrim in Heyting algebras

Abstract: We evaluate 14 equations for the hoop and pocrim in Heyting algebras as *not* tautologous. The methodology of using proof assistants Prover9 and Mace8 is also refuted. These artifacts form a *non* tautologous fragment of the universal logic VŁ4.

We assume the method and apparatus of Meth8/VŁ4 with Tautology as the designated proof value, **F** as contradiction, N as truthity (non-contingency), and C as falsity (contingency). The 16-valued truth table is row-major and horizontal, or repeating fragments of 128-tables, sometimes with table counts, for more variables. (See ersatz-systems.com.)

LET ~ Not, ¬ ; + Or, ∨, ∪, ⊔ ; - Not Or; & And, ∧, ∩, ⊓ , · , ⊗ ; \ Not And;
> Imply, greater than, →, ⇒ , ↦, ≻, ⊃, ↠ ; < Not Imply, less than, ∈, ≺, ⊂, ⊬, ⊭, ↞, ≲ ;
= Equivalent, ≡, :=, ⇔, ↔, ≜, ≈, ≃ ; @ Not Equivalent, ≠, ⊕;
% possibility, for one or some, ∃, ◊, M; # necessity, for every or all, ∀, □, L;
(z=z) T as tautology, ⊤, ordinal 3; (z@z) **F** as contradiction, Ø, Null, ⊥ , zero;
(%z>#z) N as non-contingency, Δ, ordinal 1; (%z<#z) C as contingency, ∇, ordinal 2;
~(y < x) (x ≤ y), (x ⊆ y), (x ⊑ y); (A=B) (A~B).
Note for clarity, we usually distribute quantifiers onto each designated variable.

From: Arthan, R.; Oliva, P. (2019). Studying algebraic structures using Prover9 and Mace4. arxiv.org/pdf/1908.06479.pdf p.oliva@qmul.ac.uk, rda@lemma-one.com

Abstract … The specific tools in our case study are Prover9 and Mace4; the algebraic structures are generalisations of Heyting algebras known as hoops. We will see how this approach helped us to discover new theorems and to find new or improved proofs of known results. …

1.1 Using Prover9 and Mace4 In a semilattice, one defines a relation ≥ by

$x \geq y \Leftrightarrow x \cup y = x.$ (1.1.1.1)

LET p, q, r: x, y, z.

~(q>p)=((p+q)=p) ; **FFFF FFFF FFFF FFFF** (1.1.1.2)

Remark 1.1.1.2: Eq. 1.1.1.2 is *no*t tautologous, refuting the definition of ≥ in a semilattice.

1.2 Investigating the algebraic structure of hoops Hoops are a generalisation of Heyting algebras (used in the study of intuitionistic logic [In a Heyting algebra one normally uses $x \rightarrow y$ for $y \ominus x$, and $x \wedge y$ for $x \oplus y$.]). ... A hoop [Strictly speaking this is a bounded hoop: an unbounded hoop omits the constant 1 and axiom (8).] is a structure (H, 0, 1, ⊕, ⊖) satisfying the following axioms:

$x \ominus x = 0$ (1.2.4.1)

LET p, q, r: x, y, z.

(p>p)=(s@s) ; **FFFF FFFF FFFF FFFF** (1.2.4.2)

$0 \ominus x = 0$ (1.2.7.1)

p>(s@s)=(s@s) ; **FFFF FFFF FFFF FFFF** (1.2.7.2)

$x \ominus 1 = 0$ (1.2.8.1)

((s=s)>p)=(s@s) ; **FFFF FFFF FFFF FFFF** (1.2.8.2)

Remark 1.2.8.2: Eqs. 1.2.4.2-..8.2 are *not* tautologous, are contradictions, and are equivalent.

[A] semilattice structure induces an ordering on a hoop which turns out to be equivalent to defining $x \geq y$ to hold when10 $y \ominus x = 0$. (1.2.9.1)

((p>q)=(s@s))>~(q>p) ; T**F**TT T**F**TT T**F**TT T**F**TT (1.2.9.2)

… [T]he conjecture that for any x, y and z, $z \geq x \ominus y$ iff $z \oplus y \geq x$. (1.2.10.1)

(#r&~(#p>#q))>~((q>p)>#r) ;TTTT TCTT TTTT TCTT (1.2.10.2)

[is] an analogue of one of the laws for manipulating inequalities in an ordered commutative group, is known as the *residuation* property and is quickly proved by Prover[9].. .

Remark 1.2.10.2: Eq 1.2.10.2 is not tautologous, refuting the residuation property.

A structure for the signature $(0, 1, \oplus, \ominus, \geq)$ such that $(0, \oplus, \geq)$ is an ordered commutative monoid with least element 0, greatest element 1 and satisfying the residuation axiom [known as a (bounded) pocrim]:

$z \geq x \ominus y \Leftrightarrow z \oplus y \geq x$. (1.2.11.1)

~(q>p)>(p<(q&(q>p))) ; TT**F**T TT**F**T TT**F**T TT**F**T (1.2.11.2)

Remark 1.2.11.2: Eq. 1.2.11.2 is *not* tautologous, refuting the definition of a pocrim.

One might conjecture that any pocrim is a hoop. However this conjecture is false … . Inspection of the operation tables reveals the weakness: in a hoop, if $x \geq y$ then $x = y \oplus (x \ominus y)$, but in a pocrim, even when $x \geq y$, we can have $x < y \oplus (x \ominus y)$: (1.2.12.0)

Remark 1.2.12.0: We write Eq. 1.2.12.1 to read: "in a hoop, if $x \geq y$ then $x = y \oplus (x \ominus y)$, but in a pocrim, even when $x \geq y$, we can have $x < y \oplus (x \ominus y)$, *which is different*:" (1.2.12.1)

(~(q>q)>(p=(q&(q>p))))@(~(q>q)>(p<(q&(q>p)))) ;
FFFF FFFF FFFF FFFF (1.2.12.2)

Remark 1.2.12.2: Eq. 1.2.12.2 is *not* tautologous, but a contradiction, meaning that the hoop and pocrim as rendered are equivalent, the opposite point of what the writers intended.

2 Analysing larger proofs

2.2 Discovering derived operations and their basic properties

This led us to introduce new operations so that multiple steps in the proof could be understood as properties of these new operations. In total we found, apart from $x \cup y$, three further new derived operations:

Remark 2.0: We count four further new derived operations, but introduction of the "\" connective as "difference" in Table 3 Nomenclature is unclear to us (although we suspect XOR), so we stop evaluation after two equations.

$x \cup y \equiv x \oplus (y \ominus x)$ (2.0.1.1)

(p+q)=(p&(q>p)) ; TT**F**T TT**F**T TT**F**T TT**F**T (2.0.1.2)

Remark 2.2.1.2: Eq. 2.0.1.2 is the equivalent truth table value result as Eq. 1.2.11.2.

$x \cap y \equiv x \ominus (x \ominus y)$ (20.2.1)

(p&q)=((q>p)>p) ; T**FF**T T**FF**T T**FF**T T**FF**T (2.0.2.2)

When identifying these operations we also used our knowledge of the correspondence between hoops and Heyting algebras. For instance, $x \cap y$ in logical terms corresponds to $(y \rightarrow x) \rightarrow x$, which generalises double negation and in theoretical computer science is known as the continuation monad.. . (2.0.5.1)

(p&q)=((q>p)>p) ; T**FF**T T**FF**T T**FF**T T**FF**T (2.0.5.2)

Remark 2.0.5.2: Eq. 2.0.5.2 as rendered is equivalent to Eq. 2.0.2.2.

So, according to … our methodology, we looked first for basic properties of these new operations, or of their relation with the primitive operations. We come up with six simple properties (listed in the following lemma) that we then added as axioms, and rerun the proof search.

Lemma 2.1 The following hold in all hoops:

Remark 2.1.0: We evaluate two of the less obvious of the six properties to avoid the "\" connective.

(v) $z \cap (y \ominus x) \geq (z \cap y) \ominus (z \cap x)$ (2.1.v.1)

~(((r&p)>(r&q)) >(r&(p>q))) = (s=s) ;
TTTT **FFFF** TTTT **FFFF** (2.1.v.2)

(vi) $x \ominus (x \cap y) = x \ominus y$ (2.1.vi.1)

((p&q)>p)=(q>p) ; TT**F**T TT**F**T TT**F**T TT**F**T (2.1.vi.2)

Remark 2.1.vi.2: Eq. 2.1.vi.2 is the equivalent truth table value result as 1.2.11.2. and 2.0.1.2. Eqs. ..v.2 and ..vi.2 as axioms are *not* tautologous.

2.5 Tackling the harder conjecture

Lemma 2.11 (MPS) x = (x ∩ y) ⊕ (x ⊖ y) (2.11.1)

p=(p&q)&(q>p) ; T**F**TT T**F**TT T**F**TT T**F**TT (2.11.2)

Remark 2.11.2: Eq. 2.11.2 is *not* tautologous and is the equivalent truth table value result as 1.2.9.2.

We evaluated 14 expressions as definitions, axioms, and lemmas for which none is tautologous and four are contradictions. This refutes hoops and pocrims as stated for Heyting algebras and the methodology adopted for Prover9 and Mace8 proof assistants.

We include the chapter conclusion of the writers to complete this artifact.

3 Concluding Remarks In Section 1 we have attempted to introduce the tools and methods we have been using by examples at the level of an undergraduate project. We hope this is of interest to educators and advocate introduction of tools such as Prover9 and Mace4 into mathematical curricula. At a more advanced level, we have discussed our own research using Prover9 and Mace4 to investigate algebraic structures. It is possible to demonstrate the provability of properties like duality, commutativity or homomorphism properties by model-theoretic methods but these methods are not constructive, whereas the methods discussed in Section 2 construct explicit equational proofs. … Tools such as Prover9 automate the process of discovering a proof, but at first glance, the proofs that are discovered seem inaccessible to a human reader. We take this as an intellectual challenge in its own right and claim that with human effort, judiciously applied, we can "mine" explanative and systematic human-oriented proofs from machine-generated ones, potentially leading to new insights into the problem domain. … Some automated support for refactoring the machine-generated proofs could be very helpful. The refactoring steps of interest would include separating out lemmas and retrofitting derived notations. It is certainly of interest to speculate on possibilities for fully automating extraction of human-readable proofs from machine-generated proofs, but we view this as a hard challenge for Artificial Intelligence.

Remark 3.0: The comments above stand on their face because we show that Prover9, a non-bivalent vector space, can itself be coerced into the appearance of bivalency, such as:
for (◊p&◊q)>◊(p&q), read (◊p&◊~p)>◊(p&~p).