Syntax of the Finite Model Property

Zachary Ernst

University of Missouri-Columbia

July 23, 2008

Preliminaries

A logic L has the finite model property if, for every formula α, if $L \nvdash \alpha$, then there is a model $\mathfrak{M}$ such that:

1. $\mathfrak{M}$ has only finitely many elements in its domain,
2. $\mathfrak{M}$ respects the rules of inference of L,
3. $\mathfrak{M}$ validates all the axioms of L, but
4. α is false in $\mathfrak{M}$.

In algebra, it seems that we talk instead about an algebra having 'non-trivial finite models'.

An Example (from last year)

Here is the simplest example I can think of:
The logic L has the following axioms:

1. $p \to p$
2. $(\Box p \to q) \to (p \to q)$
3. $(\Box p \to p) \to q$
4. Modus Ponens (from $p \to q$ and q, infer q) and Universal Substitution.

- Any finite model of L validates every formula whatsoever.
- All theorems L are either instances of the axioms, or $\Box^n p \to \Box^m p$, with $n \leq m$.
- So L smells like an algebra with no non-trivial finite models.
- A very simple argument, originating with Gödel, is used to prove this fact.

Makinson's Modal Logic

$$h(\Box^{n+1}p \wedge \neg\Box^{n+2}p) \neq 0,$$

so

$$h(\Box^{n+1}p) \neq h(\Box^{n+2}p),$$

and so since $*a \leq a$,

$$h(\Box^{n+2}p) < h(\Box^{n+1}p).$$

Thus by induction we have $h(\Box p) > h(\Box^2 p) > h(\Box^3 p) > \cdots$ and so each of these elements of A is distinct. Hence A has infinitely many elements.

THEOREM 2. *μ is not a thesis of* C.

PROOF. We construct an infinite relational model K = (K, R) and show that it validates all theses of C but does not validate μ. Let K be the set of all natural numbers 0, 1, 2, $\cdots$; and for all x and y in K put xRy iff $x \leq y + 1$. Note that this relation is reflexive over K, but neither transitive nor symmetric.

Dudek's Algebra (1)

- Dudek's algebra D has the identity: $(ex)y = x$. Although this system has non-trivial models, they are all infinite.
- Let $e^n = \overbrace{e(e(\cdots e(ee)\cdots))}^{n \text{ times}}$.
- We can show that for any $n \neq m$, if a model of D has $e^n = e^m$, then it has $ee = e$:

$$\begin{aligned} (ex)y &= ((ee)x)y \\ x &= ey \\ x &= (ee)y \\ x &= e \end{aligned}$$

- So any non-trivial model has to map each e^i onto a different element of its domain.

Dudek's Algebra (2)

$$xy = \begin{cases} 2^y & \text{if } x = 3 \\ i & \text{if } x = 2^i \text{ for some } i \\ x & \text{otherwise} \end{cases}$$

$$e = 3$$

- Recall that for every $n \neq m$, a nontrivial model must map $\overbrace{e(e(\cdots e(ee)))}^{n}$ and $\overbrace{e(e(\cdots e(ee)))}^{m}$ onto different elements.
- The model does this by raising 2 to higher powers:

$$\begin{aligned} ee &= 2^5 \\ e(ee) &= 2^{2^5} \text{ and so on...} \end{aligned}$$

- It is a model:

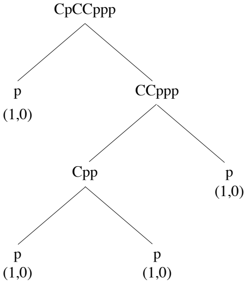
$$\begin{aligned} (ex)y &= \\ (3x)y &= \\ 2^x y &= x \end{aligned}$$

UCTA Evaluation

- This tree automaton moves up from the leaves of a tree to the root.
- Its state and counter change at each node, depending upon:
  - the symbol at the current node,
  - the states of the automaton at each child node, and
  - the counter values at each child node.
- Think of a model in which the elements of the domain are the possible states of the automaton.
- By having a counter that can take any of $|\mathbb{N}|$ values, such a model has an infinite domain.
- At least with respect to propositional logics, these can be discovered automatically.

Example Evaluation (1)

States: {1,2}
U–Formula: Cpp
Initial State: 1
Initial Counter: 0
+: Increment
–: Set to zero



Left Child < Right Child

	1	2
1	1	2
2	1	1

Left Child = Right Child

	1	2
1	1	2
2	2	2

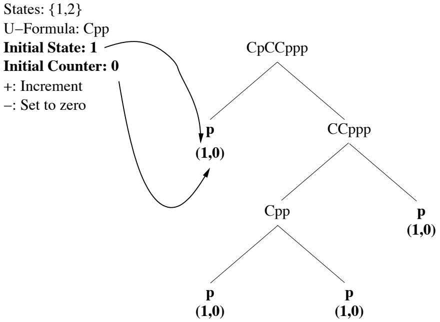
Left Child > Right Child

	1	2
1	2	2
2	1	1

Figure:

- The automaton starts at the leaves and moves up toward the root.

Example Evaluation (2)



Left Child < Right Child

	1	2
1	1	2
2	1	1

Left Child = Right Child

	1	2
1	1	2
2	2	2

Left Child > Right Child

	1	2
1	2	2
2	1	1

Figure:

Example Evaluation (3)

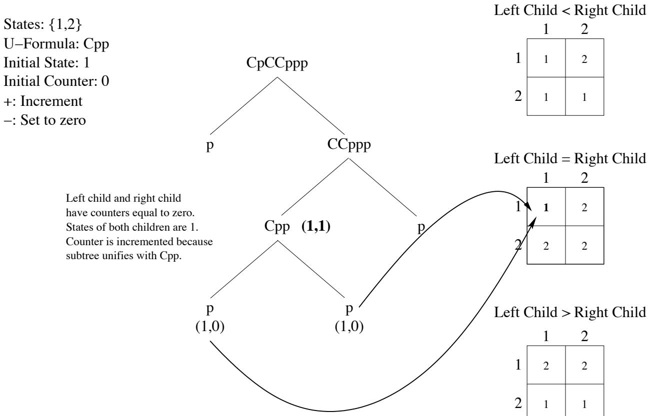


Figure:

Example Evaluation (4)

States: {1,2}
U–Formula: Cpp
Initial State: 1
Initial Counter: 0
+: Increment
–: Set to zero

Left child counter is greater than right child counter (1>0).

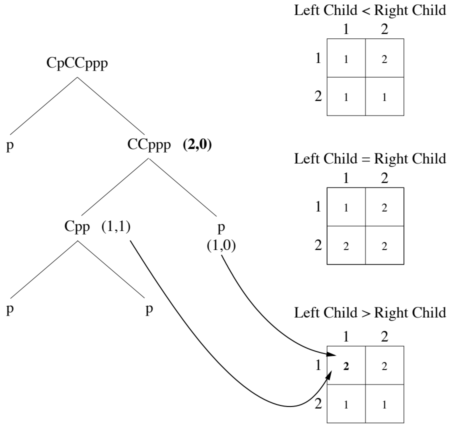


Figure:

Example Evaluation (5)

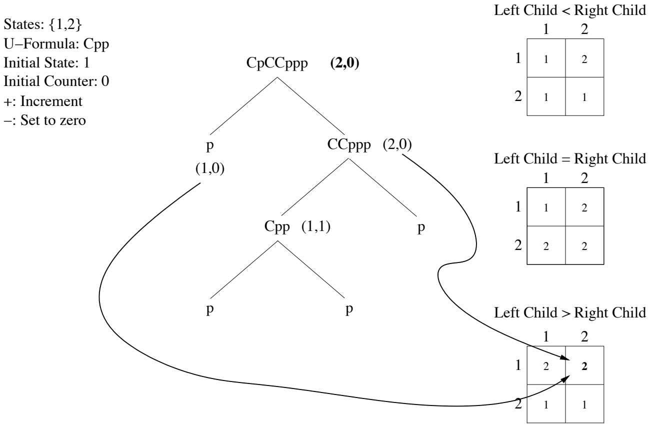


Figure:

Limits of UCTA (1)

- More problematic with algebra than propositional logic.
  - With propositional logics, whether a formula is true (provable, designated) on the model depends only on the state, not the counter.
  - In an algebra, we are interested in establishing equalities.
  - Equalities are intersubstitutable.
  - But since there are only a finite number of states, then whether two terms are equal cannot depend only upon the state.
  - So a UCTA for an algebra cannot be automatically discovered using a first-order model finder.

Limits of UCTA (2)

We might need more than one counter:

$$p \to p$$
$$(\Box p \to q) \to (p \to q)$$
$$(\Box p \to p) \to q$$
$$(\Diamond p \to q) \to (p \to q)$$
$$(\Diamond p \to p) \to q$$

Worse yet, we might need infinitely many counters.

Questions

1. Can we automate the search for algebraic models with equality?
2. When the 'provablity predicate' acts like equality, does this block UCTA countermodels?
3. Can we determine automatically whether a logic would require a UCTA with infinitely many counters?
4. Is there a relationship between regular languages and the finite model property?
5. Does there exist, for each finitely axiomatizable logic, a UCTA countermodel for any of its non-theorems?