

Monadic Second Order Logic as the Model Companion of Temporal Logic; a clearer version

Anatole Leterrier

April 14, 2022

This is nothing more than another version of the 2016 paper by Ghilardi and van Gool. Since the proofs in [1] are mostly developed in the appendix, rewriting them completely in a more readable way is a quite useful exercise. Doing so, my goal is to understand deeply how the constructions work, in order to produce my own version for finite trees with counters by the end of my internship. If I explicit some point more than it was in the original paper, or if I change the order of a few definitions, it is because my understanding benefits from it. Conversely, if I move quickly on a topic, it is because it is already clear in my head.

1 Syntax and Semantics of fair CTL

Fair CTL, which is an extension of the well-known computational tree logic, is defined intuitively so it can simulate the termination condition for executions of tree automata. For example, the definition of EG is close to what we expect for a fair execution, ie. a branch on which a formula is true infinitely often (or a final state visited infinitely often).

We will define a *syntax* for the logic CTL^f , which will allow to interpret formulas as elements of a particular type of Boolean algebra, and to use results on duality for those algebras.

On the other hand, we will define *semantics*, which will allow to interpret such formulas on models, ie. trees whose nodes are labeled by sets of propositional variables supposed to hold on each node. Next section will then be dedicated to prove that these semantics are *complete* with respect to the syntax, in other words, that for every formula φ that holds in every model, we can syntactically deduce that φ is equivalent to the \top formula of the logic.

Syntax of fair CTL

Definition 1.1. Let $\bar{p} = \{p_1, \dots, p_n\}$ be a finite set of propositional variables. We define inductively a CTL^f -formula φ to be of the shape:

- \perp
- $p \in \bar{p}$
- $\neg\varphi$
- $\varphi \vee \psi$
- $\diamond\varphi$
- $EU(\varphi, \psi)$
- $EG(\varphi, \psi)$

For convenience, we then also define the De Morgan duals of our binary operations, respectively \wedge for \vee ; AR for EU; AF for EG. For the unary operator, we set $\Box\varphi = \neg\diamond(\neg\varphi)$; Finally, we set \top to be the negation of \perp , naturally.

Definition 1.2. We define the quasi-equational theory CTL^f by requiring the formulas to satisfy the following axioms:

1. Boolean algebra axioms for \vee, \neg, \perp ;
2. Unary operator \diamond preserves finite joins, including the empty joint \perp , making CTL^f into a modal logic;
3. $\diamond\top = \top$
4. Binary operators EU and EG satisfy the following fixpoint axioms for all a, b, c :

- $a \vee (b \wedge \diamond EU(a, b)) \leq EU(a, b)$
- $a \vee (b \wedge \diamond c) \leq c \implies EU(a, b) \leq c$
- $EG(a, b) \leq a \wedge \diamond EU(b \wedge EG(a, b), a)$
- $EG(a, b) \leq a \wedge \diamond EU(b \wedge c, a) \implies c \leq EG(a, b)$

In other words, $EU(a, b)$ is the least pre-fixpoint of the function $x \mapsto a \vee (b \wedge \diamond x)$. And $EG(a, b)$ is the greatest post-fixpoint of the function $x \mapsto a \wedge \diamond EU(b \wedge x, a)$.

Definition 1.3. Given what is said just above, we can deduce the definition of a CTL^f -algebra, which is a tuple $\mathbb{A} = (A, \perp, \neg, \vee, \diamond, EU, EG)$ which verifies every axiom of CTL^f quasi-equational theory.

Importantly, we remark that for all a, b , $AR(a, b)$ is the greatest post-fixpoint of $x \mapsto a \wedge (b \vee \Box x)$, while $AF(a, b)$ is the least pre-fixpoint of $x \mapsto a \vee \Box AR(b \vee c, a)$.

Also, the way EU and EG formulas are defined reminds the fixpoint operators of the μ -calculus. For example, for every a, b , $EU(a, b)$ could have been defined as $\mu x.(a \vee (b \wedge \Diamond x))$, the least fixpoint of this monotone function (it is enough to say the least *pre-fixpoint*). As for EG, it is a greatest (*post*-) fixpoint, so it could have been defined with a ν operator, where $\nu f = \neg \mu(\neg f)$. We can see that such a definition of EG makes apparent a nesting of a ν and a μ operator. This is what makes the logic hard to simplify, compared to LTL which only contains one such fixpoint operator.

Definition 1.4. *For any finite set of propositional variables \bar{p} and any CTL^f -algebra \mathbb{A} , we can define a valuation $V : \bar{p} \rightarrow A$. Then any CTL^f -formula φ with variables in \bar{p} can be interpreted as a term in A (the interpretation of φ under V , $\varphi^A(V)$).*

Then, we can say the equality $\varphi(\bar{p}) = \psi(\bar{p})$ is valid if it interprets to \top under every valuation to every \mathbb{A} .

φ and ψ are said to be equivalent if the equation $\varphi = \psi$ is valid. φ is called a tautology if it is equivalent to \top , and is said to be consistent if it is not equivalent to \perp .

Finally, we say φ entails ψ if $\neg \varphi \vee \psi$ is a tautology, which we will note $\varphi \vdash \psi$ or $\varphi \leq \psi$.

Semantics of fair CTL

Definition 1.5. *We first define the notion of transition system, ie. a pair (S, R) , where S is a set, and R a binary relation on S . Then, a R -path is a (possibly infinite) sequence of nodes in S such that $s_i R s_{i+1}$ for all i . Finally, for a finite set \bar{p} of variables, define a \bar{p} -coloring $\sigma : S \rightarrow \mathcal{P}(\bar{p})$.*

We can define the forcing relation \models between nodes in S and formulas φ with variables in \bar{p} , by induction on φ :

- $s \not\models \perp$;
- $s \models p$ if $p \in \sigma(s)$;
- $s \models \neg \varphi$ if not $s \models \varphi$;
- $s \models \varphi \vee \psi$ if $s \models \varphi$ or if $s \models \psi$;
- $s \models \Diamond \varphi$ if there exists s' such that $s R s'$ and $s' \models \varphi$;
- $s \models EU(\varphi, \psi)$ if there exists $n \geq 0$ and a path $s_0 R s_1 R \dots R s_n$ ($s_0 = s$) such that $s_k \models \psi$ for every $k < n$ and $s_n \models \varphi$;

- $s \models EG(\varphi, \psi)$ if there exists an infinite path $s_0Rs_1R\ldots$ ($s_0 = s$) such that $s_k \models \varphi$ for every k and $s_j \models \psi$ infinitely often on the path.

Remark that as a consequence, $s \models \Box\varphi$ if and only if for every successor s' of s by R , $s' \models \varphi$ holds. Also, $s \models AR(\varphi, \psi)$ if and only if for every $n \geq 0$ and every path $s = s_0Rs_1R\ldots Rs_n$, either $s_k \models \psi$ for some $k < n$, or $s_n \models \varphi$.

Similarly, $s \models AF(\varphi, \psi)$ if and only if for all infinite paths $(s)_k$ starting from s_0 , if $s_j \not\models \psi$ infinitely often, then $s_k \models \varphi$ for some k .

Also, as a convention, we will consider the transition systems to be *serial*, ie. that every node has a successor. Syntactically, this is represented by the axiom $\Diamond\top = \top$.

Now, in order to prove completeness, we introduce the notion of *complex algebra* of a transition system. This can be seen as a canonical algebra for the semantics of our logic. Then, we will be able to say that a formula holds semantically if, and only if, it can be derived in the complex algebra.

Definition 1.6. *Given a transition system (S, R) , its complex algebra is the tuple*

$$\mathbb{P}_{(S,R)} = (\mathcal{P}(S), \emptyset, S \setminus (-), \cup, \Diamond_R, EU_R, EG_R)$$

which is the classical Boolean algebra over the powerset of S , along with operators

$$\Diamond_R(a) = R^{-1}[a] = \{s \in S \mid \exists t \in a, sRt\}$$

and EU_R and EG_R defined as fixpoints (see above) so that we finally have a CTL^f -algebra.

Lemma 1.7. *For every a, b in the complex algebra of (S, R) it holds that*

$$EU(a, b) = \bigcup_{i=1}^{\infty} D_n(a, b)$$

where D_n is defined inductively as

$$\begin{cases} D_0(a, b) = \emptyset \\ D_{n+1}(a, b) = a \cup (b \cap \Diamond_R D_n(a, b)) \end{cases}$$

Proof. \supseteq By induction on n we show $D_n(a, b) \subseteq EU(a, b)$.

For $n = 0$ this is obvious. For the induction step, let $s \in D_{n+1}(a, b)$. By the pre-fixpoint definition of EU we have $a \leq EU(a, b)$ and $(b \cap \Diamond EU(a, b)) \leq EU(a, b)$. It follows that if s is in a , it is directly in $EU(a, b)$; and in the other case, there exists s' in $D_n(a, b)$ such that sRs' . By hypothesis $s' \in EU(a, b)$. We have exactly $s \in b \cap \Diamond EU(a, b)$, from which we conclude $s \in EU(a, b)$.

\subseteq We will show that $D(a, b) := \bigcup_{i=1}^{\infty} D_i(a, b)$ is a pre-fixpoint of $x \mapsto a \cup (b \cap \diamond x)$. From the definition of *EU* the inclusion will directly follow.

Let $s \in a \cup (b \cap \diamond D(a, b))$. Note that $D_1(a, b) = a$ so if $s \in a$ it is in $D(a, b)$. And if $s \in (b \cap \diamond D_k(a, b))$ for some k (\diamond preserves countable unions), it is in $D_{k+1}(a, b)$ by definition. We can conclude here. \square

Note that $D_n(a, b)$ is the set

$$\{s \in S \mid \exists s_1, \dots, s_n, s = s0Rs_1R\dots Rs_n, s_k \in a \forall k < n \text{ and } s_n \in b\}$$

We can now state the corresponding lemma for *EG*:

Lemma 1.8.

Proposition 1.9. *Consider a transition system (S, R) along with a \bar{p} -coloring $\sigma : S \rightarrow \mathcal{P}(\bar{p})$. This coloring corresponds exactly to a valuation $V_\sigma : \bar{p} \rightarrow \mathcal{P}(S)$, as we in fact have a bijection between \bar{p} -colorings and such valuations.*

Then, for every CTL^f -formula φ with variables in \bar{p} and for every $s \in S$, we have:

$$s \models \varphi \iff s \in \mathbb{P}^{(S, R)}(V_\sigma(\bar{p}))$$

Proof. First, we show there is a bijection between \bar{p} -colorings $\sigma : S \rightarrow \mathcal{P}(\bar{p})$ and valuations $V : \bar{p} \rightarrow \mathcal{P}(S)$. Given σ , define $V_\sigma : \bar{p} \mapsto \{s \in S \mid p \in \sigma(s)\}$. And conversely, given V , we can define $\sigma_V : v \mapsto \{p \in \bar{p} \mid v \in V(p)\}$.

Second, we show the equivalence, by induction on the structure of φ . Looking at the definitions, it should be obvious for all cases up to diamond; as for *EU* and *EG*, the wanted equivalence is a consequence of lemmas 1.7 and 1.8. \square

2 Completeness proof

Definition 2.1.

References

- [1] Silvio Ghilardi and Samuel van Gool. Monadic second order logic as the model companion of temporal logic. In *Proceedings of the Thirty first*

Annual IEEE Symposium on Logic in Computer Science (LICS 2016),
2016.