The common fragment of CTL and LTL needs existential modalities

Mikołaj Bojańczyk
Warsaw University

Abstract

The language "all paths belong to $(ab)^*a(ab)^*c^\omega$" is definable in CTL, but not in ACTL, which is the fragment of CTL that only uses universal modalities.

CTL [1] is the temporal branching time logic that uses the modalities: $\mathsf{A}\varphi\mathsf{U}\psi$ (on every path, there is a node with ψ, and all preceding nodes satisfy φ), $\mathsf{AX}\varphi$ (every successor position satisfies φ) and $\mathsf{AG}\varphi$ (on every path, every position satisfies φ). ACTL is the fragment of CTL, which uses the above modalities and does not allow negation. (Here we use mutually exclusive atomic propositions; if they are are not exclusive then negation is allowed next to atomic propositions.) Clearly, ACTL is a proper fragment of CTL; for instance, the CTL property "some node has label a" is not definable in ACTL.

A *universal path* property is one of the form "all paths belong to L", for some infinite word language L. In [2], Maidl investigated the universal path properties expressible in CTL. However, the decidability results from [2] actually concerned not all of CTL, but only ACTL, and the following question was raised: is a CTL-definable universal path language necessarily ACTL-definable? A positive answer to this question would be consistent with the intuition that only universal path modalities are needed to express universal path properties. In this note, we show that this is not the case:

Theorem 0.1
The language L = "all paths belong to $(ab)^*a(ab)^*c^\omega$" is definable in CTL, but not ACTL.

This example shows that ideas significantly different from those in [2] are needed to understand the universal path properties definable in CTL.

Here, we view CTL as a logic that describes properties of infinite trees, where both infinite paths, and infinite node outdegree are allowed. The same results would apply for transition systems, and also finite trees.

Before proceeding with the proof, we would like to remark the similarity of this "paradox" with a result for first-order logic over finite binary trees. In [4], Potthof showed that the language "all paths belong to $(aa)^*$" is definable in first-order logic over finite binary trees, even though the word language $(aa)^*$ is not definable in first-order logic over words. His technique was similar to the one invoked below, in that it used properties of "maximal" nodes.

We will now prove Theorem 0.1.

Lemma 0.2 The language L is definable in CTL.

Proof
It is easy to show that the language "all paths belong to $(ab)^*c^\omega$" is definable in CTL. Let φ_a be such a formula; we will use it below. Likewise we will use a formula φ_b for the language "all paths belong to $b(ab)^*c^\omega$".

The formula for the language L is a conjunction of several properties. First, we have to manage the way c's are used. Formula (1) says that every path contains some c, and every time c appears, all subsequent nodes are c's; finally, only b nodes can have a c successor:

$$\mathsf{AF}c \ \wedge\ \mathsf{AG}(c \Rightarrow \mathsf{AG}c) \ \wedge\ \mathsf{AG}(a \Rightarrow \mathsf{AX}(a \vee b)) \ . \qquad (1)$$

Formula (2) says that the tree does not contain two consecutive b's:

$$\mathsf{AG}(b \Rightarrow \mathsf{AX}(a \vee c)) \ . \qquad (2)$$

Formula (3) says that on every path, two consecutive a's can be found at most once:

$$\neg\mathsf{EF}(a \wedge \mathsf{EX}(a \wedge (\mathsf{EF}(a \wedge \mathsf{EX}a))) \ . \qquad (3)$$

So far, we have stayed within ACTL. The above three properties guarantee that every path in the tree is either in $(ab)^*c^\omega$ or in $(ab)^*a(ab)^*c^\omega$, as long as the root has label a. We now need to eliminate the paths of the first type. First, we enforce the root label, and say that at least one path is not in $(ab)^*c^\omega$

$$a \quad \wedge \quad \neg\varphi_a \ . \qquad (4)$$

Note that already here, we go beyond ACTL, since φ_a is negated. The more imporant property, however, says there is no node x such that: some path beginning in x has two consecutive a's, and some successor of x satisfies φ_a or φ_b, depending on the label of x. This expressed by the formula:

$$\neg\mathsf{EF}(\mathsf{F}(a \wedge \mathsf{EX}a) \wedge (a \Rightarrow \mathsf{EX}\varphi_b) \wedge (b \Rightarrow \mathsf{EX}\varphi_a)) \qquad (5)$$

Here again we go beyond ACTL. We claim that a tree belongs to L if and only if it satisfies the conjunction of formulas (1)-(5).

The left-to-right implication is proved as follows. Let t be a tree in L. Clearly (1)-(4) have to be satisfied. For (5), we need to show that every node x fails the property:

$$\mathsf{EF}(a \wedge \mathsf{EX}a) \wedge (a \Rightarrow \mathsf{EX}\varphi_b) \wedge (b \Rightarrow \mathsf{EX}\varphi_a) \; .$$

We only consider the case when the node x has label a, the other is done in a similar way. Let then x be a node with label a that satisfies $\mathsf{EF}(a \wedge \mathsf{EX}a)$. By (2)-(4), the path leading up to x must belong to $(ab)^*$. In particular, no successor of x can be the beginning of a path in $(ab)^*c^\omega$, not to mention having all paths of this form (which is what $\mathsf{EX}\varphi_a$ says).

We now take the right-to-left implication. Let then t be a tree that satisfies formulas (1)-(4). We claim that if t is outside L, then the formula (5) fails. By (1)-(4), the tree has paths of the form $(ab)^*c^\omega$, and of the form $(ab)^*a(ab)^*c^\omega$. Let X be the set of nodes, which lie on the intersection of some path of the form $(ab)^*a(ab)^*c^\omega$ and some other path of the form $(ab)^*c^\omega$. By $\mathsf{AG}c$, the prefix-closed set X does not contain an infinite path, therefore it has some maximal element, i.e. a node $x \in X$ without proper descendants in X. (Since nodes may have infinite outdegree, there may be no common bound on the depth of nodes from X, but this is not a problem here, since we only need one maximal node.) We claim that the maximal node x witnesses the failure of (5), i. e. x satisfies

$$\mathsf{EF}(a \wedge \mathsf{EX}a) \wedge (a \Rightarrow \mathsf{EX}\varphi_b) \wedge (b \Rightarrow \mathsf{EX}\varphi_a) \; .$$

We only consider the case where x has label b. By maximality of x, there must be a successor y such that all paths that pass through y belong to $(ab)^*c^\omega$; otherwise y would belong to X. The node y witnesses $\mathsf{EX}\varphi_a$ (note that y may have label c, and only c's in its subtree). Since the path leading to x belongs to $(ab)^*$ and x is on some path in $(ab)^*a(ab)^*c^\omega$, there must be two consecutive a's on some path that begins in x. □

We now show that the language L is not definable in ACTL. We will use a characterization of Maidl from [2], slightly restated:

Theorem 0.3
Let $L \subseteq A^\omega$ be a language of infinite words. The following are equivalent:

- *The tree language "all paths belong to L" is definable in ACTL;*
- *The complement of L is a finite union of languages of the form*

$$A_1^* a_1 A_2^* a_2 \cdots A_n^* a_n A_{n+1}^\omega \qquad a_1,\ldots,a_n \in A,\; A_1,\ldots,A_{n+1} \subseteq A \; . \tag{6}$$

Lemma 0.4 The language L is not definable in ACTL.

Proof
Towards a contradiction with Theorem 0.3, assume that the complement of L is defined by an finite union of expressions as in (6). Let n be the size of the largest of these expressions. Since $(ab)^{n+1}c^\omega$ is outside L, it must be captured by one of the expressions:

$$(ab)^{n+1}c^\omega \in A_1^* a_1 A_2^* a_2 \cdots A_m^* a_m A_{m+1}^\omega \; .$$

Since $m \leq n$, it is fairly easy to see that the word $(ab)^j a(ab)^{n+1-j}c^\omega \in L$ will also be captured by the same expression, a contradiction. □

We remark that a third equivalent condition can be added to Theorem 0.3: the word language L is defined by a Π_2 formula, i.e. a first order formula with a quantifier prefix $\forall^*\exists^*$, where the signature allows label tests, and the linear order on word positions. Since it is decidable if a language $L \subseteq A^*$ of finite words is definable by a Π_2 formula, it follows that it is decidable if a regular property of finite trees is expressible in ACTL. A similar result might hold for infinite trees, but would require an adaptation of techniques from [5] or [3], from the finite to the infinite, which is left open.

References

[1] E. M. Clarke and E. A. Emerson. Design and synthesis of synchronization skeletons using branching time temporal logic. In *Logics of Programs: Workshop*, volume 131 of *Lecture Notes in Computer Science*, pages 52–71, 1981.

[2] M. Maidl. The common fragment of CTL and LTL. In *Foundations of Computer Science*, pages 643–652, 2000.

[3] J.-É. Pin and P. Weil. Polynomial closure and unambiguous product. *Theory Comput. Systems*, 30:1–30, 1997.

[4] A. Potthoff. First-order logic on finite trees. In *Theory and Practice of Software Development*, volume 915 of *Lecture Notes in Computer Science*, pages 125–139, 1995.

[5] J. Stern. Complexity of some problems from the theory of automata. *Information and Control*, 66:163–176, 1985.