Towards relating hyper- and epistemic-temporal logics

Matvey Soloviev[1], Vineet Rajani[2], and Musard Balliu[3]

[1] *Chalmers University*
[2] *University of Kent*
[3] *KTH Royal Institute of Technology*

Abstract

Hyperproperties specify relations between two or more executions of a program. Many classical security definitions are stated as hyperproperties. Specification logics like epistemic- and hyper-temporal logics provide two very different alternatives for specification of such hyperproperties. While epistemic-temporal logics (like KCTL*) use a knowledge modality to reason about all alternate unnamed paths in a coarse-grained, hyper-temporal logics (like HyperCTL*) take a more fine-grained approach by allowing explicit quantification over named paths. Prior work has shown that the two approaches are not equally expressive, there are examples of formulae expressible in one but not in the other. In this work, we show that is possible to subsume HyperCTL* in KCTL* with proposition quantification.

1 Introduction

Hyperproperties [2] specify relations between two or more executions of a program. Many classical security definitions are stated as hyperproperties. Specification logics like epistemic- and hyper-temporal logics provide two very different alternatives for specification of such hyperproperties. While epistemic-temporal logics (like KCTL*[1]) use a knowledge modality to reason about all alternate unnamed paths in a coarse-grained manner, hyper-temporal logics (like HyperCTL* [2]) take a more fine-grained approach by allowing explicit quantification over named paths. Prior work [1] has shown that the two logics are not equally expressive, as there are formulas expressible in one but not in the other.

There have been efforts to reconcile this divide by building a more expressive logic with a past-time operator [1] which subsumes both KCTL* and HyperCTL*. Another line of work [5] took a different approach, showing that addition of propositional quantification to HyperLTL suffices to subsume KLTL, without the addition of a past-time operator. However, it [5] says nothing about an embedding in the reverse direction. To the best of our knowledge, it is still an open question to investigate if epistemic-temporal logics with propositional quantification can subsume hyper-temporal logics. In this work, we answer this question in affirmative, by showing an embedding of HyperCTL* into KCTL* with propositional quantification.

2 The two logics: HyperCTL* and KPCTL*

HyperCTL* is an extension of CTL* with an ability to quantify over named paths. Doing so allows you to express properties defined over multiple executions (aka hyperproperties). The formulas of HyperCTL* are interpreted over a Kripke structure K[1], equipped with a path assignment Π mapping path variables to initial paths from K, a path variable y which refers to the current path and a positive natural number i which refers to the current position. The syntax and semantics of HyperCTL* (omitting the semantics of the boolean connectives) from [1] is described in Figure 1.

Unlike HyperCTL*, KCTL* does not introduce named path quantification, but instead uses a knowledge modality K_a (indexed with an agent a) to reason about alternate execution paths. The formulas of KCTL* are interpreted over extended Kripke structure denoted by $\Lambda = (K, Obs)$ (where an observation map $Obs : Agent \to 2^{AP}$, associates an agent with the set of observable propositions), an initial path π of K and a position i along π. The syntax and semantics of KCTL* (omitting the semantics of the boolean connectives) under the synchronous perfect recall setting [4, 3], from [1], is described in Figure 2.

We introduce a modest extension of KCTL* called KPCTL*. KPCTL* has an ability to quantify over KCTL* propositions. To do define the semantics of KPCTL* we add a mapping Φ of proposition variables to

[1] K is defined as a tuple $< S, S_0, E, V >$,. S is the set of states, S_0 is an initial state, E is a transition relation and $V : S \to 2^{AP}$ is valuation function, where AP is a finite set of atomic propositions.

$$
\begin{array}{lcl}
\varphi & ::= & \top \mid p[x] \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{X}\varphi \mid \varphi \mathsf{U} \varphi \mid \exists x.\varphi \\
\\
\Pi, y, i \vDash_K p[x] & \Leftrightarrow & p \in V(\Pi(x)(i)) \\
\Pi, y, i \vDash_K \mathsf{X}\varphi & \Leftrightarrow & \Pi, y, i+1 \vDash_K \varphi \\
\Pi, y, i \vDash_K \varphi_1 \mathsf{U} \varphi_2 & \Leftrightarrow & \text{for some } j \geq i\text{: } \Pi, y, j \vDash_K \varphi_2 \text{ and } \Pi, y, k \vDash_K \varphi_1 \text{ for all } k \in [i, j-1] \\
\Pi, y, i \vDash_K \exists x.\varphi & \Leftrightarrow & \Pi[x \mapsto \pi'], x, i \vDash_K \varphi \text{ for some initial path } \pi' \text{ of } K \text{ s.t. } \pi'[0,i] = \Pi(y)[0,i]
\end{array}
$$

Figure 1: Syntax and semantics of HyperCTL* from [1]

$$
\begin{array}{lcl}
\varphi & ::= & \top \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{X}\varphi \mid \varphi \mathsf{U} \varphi \mid \exists \varphi \mid \mathsf{K}_a \varphi \\
\\
\pi, i \vDash_\Lambda p & \Leftrightarrow & p \in V(\pi(i)) \\
\pi, i \vDash_\Lambda \mathsf{X}\varphi & \Leftrightarrow & \pi, i+1 \vDash_\Lambda \varphi \\
\pi, i \vDash_\Lambda \varphi_1 \mathsf{U} \varphi_2 & \Leftrightarrow & \text{for some } j \geq i\text{: } \pi, j \vDash_\Lambda \varphi_2 \text{ and } \pi, k \vDash_\Lambda \varphi_1 \text{ for all } k \in [i, j-1] \\
\pi, i \vDash_\Lambda \exists \varphi & \Leftrightarrow & \pi', i \vDash_\Lambda \varphi \text{ for some initial path } \pi' \text{ s.t.} \pi'[0,i] = \pi[0,i] \\
\pi, i \vDash_\Lambda \mathsf{K}_a \varphi & \Leftrightarrow & \text{for all initial paths } \pi' \text{ of } K \text{ s.t.} \\
 & & V(\pi[0,i]) \text{ and } V(\pi'[0,i]) \text{ are } Obs_a\text{-equivalent, } \pi', i \vDash_\Lambda \varphi.
\end{array}
$$

Figure 2: Syntax and semantics of KCTL* from [1]

KCTL* propositions. The semantics of proposition variables and proposition quantification is defined as follows (the remaining cases are directly lifted from the semantics of KCTL* in an expected way):

$$
\begin{array}{rcl}
\Phi, \pi, i \vDash_\Lambda P & \Leftrightarrow & \pi, i \vDash_\Lambda \Phi(P) \\
\Phi, \pi, i \vDash_\Lambda \exists P.\varphi & \Leftrightarrow & \exists \psi \in \text{KCTL*}\text{: } \Phi[P \mapsto \psi], \pi, i \vDash_\Lambda \varphi
\end{array}
$$

3 Embedding

Upfront it seems that in HyperCTL* we can define a formula relating valuations only along two named paths, but there is no mechanism to encode this behaviour in KCTL*, as it does not have an ability to isolate such named paths. Bozzelli et al. [1] concretise this intuition by describing a counterexample to show that HyperLTL (and hence HyperCTL*) cannot be embedded in KCTL*[2].

In this work, we show that it is possible to embed HyperCTL* into KPCTL*. The key cases of the embedding are described in Figure 3 (embedding of the boolean connectives is defined inductively in an expected way). The embedding uses a derived full-knowledge modality K^+ to represent the idea of "knowing everything". It is used for considering all paths identical to that of the current path, and can be seen as a specialisation of the knowledge modality whose observations are modelled using an identity map. Dually, the embedding also uses a derived zero-knowledge modality K^- to represent the idea of "knowing nothing". It is used for considering all paths of the same length as the current path, and can seen as a specialisation of the knowledge modality whose observations are modelled using a constant map. The key idea of our embedding is to use the characteristic formula to identify a unique path over which the translated HyperCTL* formula should hold. A characteristic formula for a path, φ, is the strongest formula that is true at every point of that path and nowhere else. Formally, a characteristic formula φ is defined using the predicate CHAR, $\text{CHAR}(\varphi) \triangleq \mathsf{FG}\varphi \wedge \forall\psi.\,(\mathsf{FG}\psi) \Rightarrow \mathsf{K}^-((\mathsf{FG}\varphi) \Rightarrow \mathsf{FG}\psi)$, where F (eventually) and G (always) are the standard CTL* modalities. The first conjunct ensures that φ is true everywhere along the path, and the second conjunct characterises its strength (required to uniquely identify a path).

$$
\begin{array}{lcl}
[\![X\varphi]\!] & \triangleq & X[\![\varphi]\!] \\
[\![\varphi_1 U \varphi_2]\!] & \triangleq & [\![\varphi_1]\!] U [\![\varphi_2]\!] \\
[\![\exists x.\varphi]\!] & \triangleq & \exists P_x.\, \neg\mathsf{K}^+\neg(\text{CHAR}(P_x) \wedge [\![\varphi]\!]) \\
[\![p[x]]\!] & \triangleq & \mathsf{K}^-(\mathsf{FG}P_x \Rightarrow p)
\end{array}
$$

Figure 3: Embedding of HyperCTL* into KPCTL* (key cases)

The embedding $([\![\cdot]\!])$ can be used to show that the model checking problem for HyperCTL* can be reduced to the model checking problem for KPCTL* by converting a HyperCTL* model into a KPCTL* model, and a HyperCTL* formula into a KPCTL* formula while preserving validity under the translated model. Both the model and formula conversion are computationally bounded.

Theorem 3.1. *Given a HyperCTL* model K in which all runs have characteristic formulae in KCTL*, we can construct a KPCTL* model $\Lambda(K)$ in polynomial time that has the same runs, such that there is a function*

[2] [1] also provides a counterexample to refute an embedding of KCTL* into HyperCTL*.

$[\![\cdot]\!]$ *that maps formulae of HyperCTL* to formulae of KPCTL*, is computable in linear time and satisfies that* $\vDash_K \varphi$ *iff* $\vDash_{\Lambda(K)} [\![\varphi]\!]$.

Interestingly, our embedding only uses the the linear time fragment of KPCTL*, understanding the full expressiveness of KPCTL* is an interesting direction of future work.

References

[1] Laura Bozzelli, Bastien Maubert, and Sophie Pinchinat. Unifying hyper and epistemic temporal logics. In *International Conference on Foundations of Software Science and Computation Structures*, pages 167–182. Springer, 2015.

[2] Michael R. Clarkson, Bernd Finkbeiner, Masoud Koleini, Kristopher K. Micinski, Markus N. Rabe, and César Sánchez. Temporal logics for hyperproperties. In *Principles of Security and Trust - International Conference, (POST)*, volume 8414, pages 265–284. Springer, 2014.

[3] Cătălin Dima. Revisiting satisfiability and model-checking for ctlk with synchrony and perfect recall. In *Computational Logic in Multi-Agent Systems*, pages 117–131. Springer Berlin Heidelberg, 2009.

[4] Joseph Y. Halpern, Ron van der Meyden, and Moshe Y. Vardi. Complete axiomatizations for reasoning about knowledge and time. *SIAM J. Comput.*, 33(3):674–703, 2004.

[5] Markus N. Rabe. *A Temporal Logic Approach to Information-flow Control.* PhD thesis, Saarland university, 2016.

Appendix

Theorem 3.2. *Given a HyperCTL model K in which all runs have characteristic formulae in KCTL*, we can construct a KPCTL* model $\Lambda(K)$ in polynomial time that has the same runs, such that there is a function $[\![\cdot]\!]$ that maps formulae of HyperCTL to formulae of KPCTL*, is computable in linear time and satisfies that $\vDash_K \varphi$ iff $\vDash_{\Lambda(K)} [\![\varphi]\!]$.*

Proof. The goal is to show that we can simulate path bindings in the context Π of HyperCTL* by binding characteristic formulae of paths in the formula context Φ. We use existential quantification over formulae in KPCTL* to pick out and bind characteristic formulae of paths sharing the prefix at the current point, which we will access using a specially prepared knowledge relation. Finally, to evaluate propositions at paths, we prepare another knowledge relation that lets us access *all* other paths in the model, and use the bound characteristic formula to pick out a particular one from among them.

We define the KPCTL* model $\Lambda(K)$ by introducing two observation maps: Obs_+, which is the identity map ("observes everything") and thus satisfies $Obs_+(V(\pi[0,i])) = Obs_+(V(\pi'[0,i]))$ iff $\pi[0,i] = \pi'[0,i]$, and Obs_-, which is a constant map ("observes nothing") and thus satisfies $Obs_-(V(\pi[0,i])) = Obs_-(V(\pi'[0,i]))$ for all π, π'. We write the associated modalities as K^+ and K^- respectively. It is not hard to verify that with an appropriate representation, these two observation maps can be defined in polynomial time in the size of K.

The translation $[\![\varphi]\!]$ is defined inductively on the structure of φ. For most constructors, we pass directly to the subformulae, e.g. $[\![\mathsf{X}\varphi]\!] \triangleq \mathsf{X}[\![\varphi]\!]$. The exceptional cases are the following:

- $[\![\exists x.\varphi]\!] \triangleq \exists P_x. \neg\mathsf{K}^+\neg(\mathrm{CHAR}(P_x) \wedge [\![\varphi]\!])$, where $\mathrm{CHAR}(\psi) \triangleq \mathsf{FG}\psi \wedge \forall\varphi.\, (\mathsf{FG}\varphi) \Rightarrow \mathsf{K}^-((\mathsf{FG}\psi) \Rightarrow \mathsf{FG}\varphi)$ is a formula that is true iff $\mathsf{FG}\psi$ is a characteristic formula for the current path.
- $[\![p[x]]\!] \triangleq \mathsf{K}^-(\mathsf{FG}P_x \Rightarrow p)$.

We note that this is linear-time.

Claim. $\mathrm{CHAR}(\psi)$ is true iff $\mathsf{FG}\psi$ is a characteristic formula for the current path.

Proof of claim. Suppose $\mathsf{FG}\psi$ is false somewhere on the current path π. Then it's easily checked that $\mathsf{FG}\psi$ is false everywhere on the current path, and hence the first conjunct fails. Suppose instead $\mathsf{FG}\psi$ is also true somewhere (and hence everywhere) on some path $\pi' \neq \pi$. Then the second conjunct fails with φ being the actual characteristic formula for φ: we have $\mathsf{FG}\varphi$, but $\mathsf{K}^-((\mathsf{FG}\psi) \Rightarrow \mathsf{FG}\varphi)$ requires in particular that $(\mathsf{FG}\psi) \Rightarrow \mathsf{FG}\varphi$ at the corresponding world in π', where however by assumption $\mathsf{FG}\psi$ but not $\mathsf{FG}\varphi$. ✓

Let X be the map from (maps from path variables to paths) to (maps from proposition variables to propositions) that, given Π, yields a map that maps P_x to a characteristic formula of $\Pi(x)$ for all $x \in \mathrm{dom}\,\Pi$. We can now prove by induction on the structure of φ that $X(\Pi), \pi, i \vDash_{\Lambda(K)} [\![\varphi]\!]$ iff $\Pi, \pi, i \vDash_K \varphi$.

- For all φ where $[\![\varphi]\!]$ just passes to subformulae, this is immediate by comparing semantics.
- For $\varphi = p[x]$, observe that $X(\Pi), \pi, i \vDash \mathsf{K}^-(\mathsf{FG}P_x \Rightarrow p)$ iff at all π' where $\mathsf{FG}P_x$, we have $p \in V(\pi'(i))$; the unique π' where $\mathsf{FG}P_x$ is $\Pi(x)$ by definition of X and characteristic formulae, so this is true iff $p \in V(\Pi(x)(i))$, which is $\Leftrightarrow \Pi, \pi, i \vDash_K p[x]$.
- For $\varphi = \exists x.\psi$, observe that $X(\Pi), \pi, i \vDash \exists P_x. \neg\mathsf{K}^+\neg(\mathrm{CHAR}(P_x) \wedge [\![\varphi]\!])$ iff there exists a ψ and a path π' accessed by K^+ such that ψ is the characteristic formula of π' (according to Claim) and $X(\Pi)[P_x \mapsto \psi], \pi', i \vDash_{\Lambda(K)} [\![\varphi]\!]$. By I.H., definition of K^+ and existence of characteristic formulae, this is true iff there exists a path π' such that $\pi[0, i] = \pi'[0, i]$ and $\Pi[x \mapsto \pi'], \pi', i \vDash_K \varphi$, which according to HyperCTL* semantics holds iff $\Pi, \pi, i \vDash_K \exists x. \varphi$.

This concludes the proof, as in particular $X(\varnothing) = \varnothing$ and hence $\varnothing, \pi, i_{\Lambda(K)} [\![\varphi]\!]$ iff $\varnothing, \pi, i \vDash_K \varphi$. □