2010 17th International Symposium on Temporal Representation and Reasoning

CTL-Like Fragments of a Temporal Logic of Robustness

John C. M^{c}Cabe-Dansted
School of Computer Science & Software Engineering
The University of Western Australia
Perth, Australia
Email: john@csse.uwa.edu.au

Clare Dixon
Department of Computer Science
University of Liverpool
Liverpool, UK
Email: cldixon@liverpool.ac.uk

***Abstract*—The logic RoCTL* is an extension of the branching time temporal logic CTL* to represent robustness of systems to transient failures such as loss of data packets. New operators are introduced dealing with obligation (where no failures occur) and robustness (where at most one additional failure occurs). The only known decision procedures for the temporal logic of robustness RoCTL* are non-elementary. Here we propose two CTL-like restrictions of RoCTL*, Pair-RoCTL and State-RoCTL. We investigate whether it is possible to translate these fragments into CTL showing whilst this is not in general possible for Pair-RoCTL it is for State-RoCTL. We obtain a satisfiability preserving translation for State-RoCTL into CTL showing that the complexity of satisfiability of State-RoCTL is EXPTIME-complete. We also show that these fragments of RoCTL* are useful in specifying systems.**

I. Introduction

The RoCTL* logic [1] is an extension of the branching-time temporal logic CTL* [2]. It was introduced to represent issues relating to robustness and reliability in systems. It does this by explicitly representing success and failure relations in the underlying model structures and using these to define an *obligatory* operator and a *robustly* operator. The obligatory operator specifies how the systems *should* behave by quantifying over paths in which no failures occur. The robustly operator specifies that something must be true on the current path and on all paths that deviate from the current path that have at most one more failure than the current path. This notation allows phrases such as "even with n additional failures" to be built up by chaining n simple unary operators together. One of the strengths of RoCTL* is its ability to express contrary-to-duty [3] obligations, which can be difficult for some other Deontic logics.

Unfortunately the only known decision procedures for RoCTL* are non-elementary to decide. We can reduce RoCTL* to QCTL* [1], which is non-elementary to decide. We can also translate RoCTL* into CTL* [4], but it is known that no truth preserving translation into CTL* or tree-automata can be elementary in the length of the formula [4], closing down what was the most promising avenue for finding an elementary decision procedure for RoCTL*. The tableau for a bundled variant of RoCTL* presented in [5] is also non-elementary. We therefore study CTL-like restrictions, termed Pair-RoCTL (RoCTL$^{\mathrm{P}}$) and State-RoCTL (RoCTL$^{\mathrm{S}}$) which may be both useful in expressing systems and more efficient to decide than RoCTL*. CTL [6] is a fragment of the logic CTL* where every path operator is paired with a temporal operators, the complexity of satisfiability is known to be EXPTIME-complete [7] and has a number of decision procedures (e.g. [7], [8], [9]).

RoCTL$^{\mathrm{P}}$ can express any statement that can be expressed in CTL, and can additionally use a *robustly* or *obligatory* operator (or their duals) in place of the *all paths* or *some paths* operators. RoCTL$^{\mathrm{S}}$ requires the robustly/prone operators to be in the scope of a CTL path operator or obligatory operator and RoCTL$^{\mathrm{S}}$ formulae are therefore state formulae.

We provide a linear translation of CTL* into RoCTL$^{\mathrm{P}}$. This shows that the complexity of satisfiability of RoCTL$^{\mathrm{P}}$ is at least as hard as CTL* (double EXPTIME-complete [10], [11]). However, we can provide a linear translation from RoCTL$^{\mathrm{S}}$ into CTL showing that the complexity of satisfiability for RoCTL$^{\mathrm{S}}$ is EXPTIME-complete (and model-checking is P-complete [12]). Further, via this translation using implemented theorem provers for CTL, for example [13], [9] we obtain tools for deciding RoCTL$^{\mathrm{S}}$ formulae.

This paper is structured as follows. In Section II we present the syntax and semantics of RoCTL* and the logics CTL* and CTL. In Section III we define RoCTL$^{\mathrm{P}}$ and show that there is an efficient satisfiability-preserving translation from CTL* into RoCTL$^{\mathrm{P}}$. In Section IV we define RoCTL$^{\mathrm{S}}$ and provide a translation from RoCTL$^{\mathrm{S}}$ into CTL. In Section V we provide an example of a problem in RoCTL$^{\mathrm{S}}$ and, via the translation into CTL, apply a CTL prover to this problem. We provide concluding remarks and mention related work in Section VI.

II. RoCTL* and CTL*

Here we introduce the logics RoCTL* and CTL* and provide their syntax and semantics.

A. RoCTL*

First we introduce the syntax and semantics of RoCTL*. This follows the presentation in [1]. The underlying models

This Project is supported by the Australian Government's International Science Linkages program and the University of Western Australia's Convocation Postgraduate Research Travel Award

1530-1311/10 $26.00 © 2010 IEEE
DOI 10.1109/TIME.2010.7

of RoCTL* have two types of relations between states, success and failure relations. The failure transition generally represents some undesired and hopefully unlikely event which we wish to handle gracefully. For example, when modelling network protocols, a failure transition may represent a packet sent being lost or corrupted. In this case we may wish to prove that a single or small number of such failures do not cause the failure of an application using this data. For a more detailed example of modelling failures in RoCTL*, see the examples in Section V and [12].

RoCTL* extends CTL* by adding the following path operators: $\mathbf{O}\phi$ (obligatory): a deontic operator, denoting that ϕ holds on every failure-free path; $\mathbf{P}\phi$ (permissible): a deontic operator, denoting that ϕ holds on some failure free path; $\blacktriangle\phi$ (robust): denoting that ϕ holds on the current path and on any path that differs from this path by a single deviating event; $\triangle\phi$ (prone): denoting that ϕ holds on the current path or on a path that differs from this path by a single deviating event; to the CTL* path operators: $\mathbf{A}\phi$ (all paths): denoting that ϕ holds on every path; and $\mathbf{E}\phi$ (some path): denoting that ϕ holds on some path.

To better understand the meaning of $\blacktriangle$ (and $\triangle$) consider a branching tree of possible future states joined by failure transitions and success transitions. Given a path σ, a path π which differs from σ by a single deviating event means that π follows σ initially to some state s_i. At s_i a transition which may be a failure or a success transition is taken to some state s_i' (possibly not on σ) i.e. the path deviates or diverges here from σ. After this point all the transitions on π must be success transitions. Since there is only one failure in π not in σ there is at most one more failure in π than σ and so $\blacktriangle$ can be read as "even if an additional failure occurs".

We can combine $\mathbf{O}$ and $\blacktriangle$ to quantify over paths with a fixed number of errors. For example, the statement $\mathbf{O}\blacktriangle\blacktriangle\phi$, indicates that all paths with a total of two or less failure transitions satisfy ϕ.

to the CTL* path operators: $\mathbf{A}\phi$ (all paths): denoting that ϕ holds on every path; and $\mathbf{E}\phi$ (some path): denoting that ϕ holds on some path. Formulae are constructed from a set $\text{PROP} = \{p, q, r, \ldots\}$ of *primitive propositions*. As well as the path operators described above, the language of RoCTL* contains **true** and **false** and the standard propositional connectives $\neg$ (not), $\vee$ (or), $\wedge$ (and) and $\Rightarrow$ (implies). For the temporal dimension we take the usual [14] set of future-time temporal connectives $\bigcirc$ (*next*), $\Diamond$ (*sometime* or *eventually*), $\Box$ (*always*), $\mathcal{U}$ *until* and $\mathcal{W}$ *unless* or *weak until*. In the following some of these operators will be introduced as abbreviations.

The set of well-formed formulae of RoCTL*, WFF_R, is defined as follows:

- **false** and any element of PROP is in WFF_R;
- if ϕ and ψ are in WFF and $\mathbf{H} \in \{\mathbf{A}, \mathbf{O}, \blacktriangle\}$ then the following are in WFF_R: $\neg\phi$; $\phi \wedge \psi$; $\mathbf{H}\phi$; $\bigcirc\phi$; $\phi\,\mathcal{U}\,\psi$.

A CTL* structure, M, is a 3-tuple (S, R, L) such that S is a set of states; R is a serial ($\forall x \exists y$: xRy) binary relation on S; L is a valuation (a map from S to the powerset of propositional variables).

A *fullpath* is an infinite sequence of states $\sigma = \langle w_0, w_1, w_2, \ldots\rangle$ such that for all $i \geqslant 0$ $(w_i, w_{i+1}) \in R$. Let $\sigma_{\geqslant i}$ be the fullpath $w_i, w_{i+1}, \ldots$, let σ_i be w_i and $\sigma_{\leqslant i}$ be $w_0, \ldots w_i$.

Definition 1. *We introduce a special proposition* $\mathbf{v}$ *to indicate that the last transition was a failure*[1]*. A fullpath is* failure-free *if and only if for all* $i \geq 1$ *we have* $\mathbf{v} \notin L(w_i)$.

Let $\boldsymbol{ap}(w)$ *be the set of fullpaths in* M *starting at state* w *and* $\boldsymbol{sp}(w)$ *be the set of all failure-free fullpaths in* M *starting with* w.

Definition 2. *For two fullpaths* σ *and* π, π *is an* i-deviation *from* σ *if and only if* $\sigma_{\leqslant i} = \pi_{\leqslant i}$ *and* $\pi_{\geqslant i+1} \in \boldsymbol{sp}(\pi_{i+1})$. π *is a deviation from* σ *if there exists a non-negative integer* i *such that,* π *is an* i*-deviation from* σ*. For a fullpath* σ*, we let* $\boldsymbol{dp}(\sigma)$ *be the set of fullpaths which are deviations from* σ.

We call a CTL* structure a RoCTL* structure iff $\mathbf{sp}(w)$ is non-empty for all $w \in S$. The semantics of RoCTL* formulae are defined on a fullpath $\sigma = \langle w_0, w_1, \ldots\rangle$ in a RoCTL* structure M as follows. Recall $\sigma_i = w_i$ so $\sigma_0 = w_0$.

$$
\begin{array}{rcl}
M, \sigma \models \bigcirc\phi & \text{iff} & M, \sigma_{\geqslant 1} \models \phi \\
M, \sigma \models \phi\,\mathcal{U}\,\psi & \text{iff} & \exists i \in \mathbf{N} \text{ s.t. } M, \sigma_{\geqslant i} \models \psi \text{ and} \\
 & & \forall j \in \mathbf{N} \text{ s.t. } j < i,\ M, \sigma_{\geqslant j} \models \phi \\
M, \sigma \models \mathbf{A}\phi & \text{iff} & \forall \pi \in \mathbf{ap}(\sigma_0)\ M, \pi \models \phi \\
M, \sigma \models \mathbf{O}\phi & \text{iff} & \forall \pi \in \mathbf{sp}(\sigma_0)\ M, \pi \models \phi \\
M, \sigma \models \blacktriangle\phi & \text{iff} & \forall \pi \in \mathbf{dp}(\sigma)\ M, \pi \models \phi \text{ and } M, \sigma \models \phi
\end{array}
$$

The definitions for propositions, and Boolean operators are as we would expect from classical logic. The semantics of other operators can be derived via equivalent formulae where $\mathbf{E}\phi \equiv \neg\mathbf{A}\neg\phi$, $\mathbf{P}\phi \equiv \neg\mathbf{O}\neg\phi$, $\triangle\phi \equiv \neg\blacktriangle\neg\phi$, $\Diamond\phi \equiv \mathbf{true}\,\mathcal{U}\,\phi$, $\Box\phi \equiv \neg\Diamond\neg\phi$, $\phi\,\mathcal{W}\,\psi \equiv \phi\,\mathcal{U}\,\psi \vee \Box\phi$.

B. CTL* and CTL

Well formed formulae of CTL* [2] are constructed from the same elements as RoCTL* but without the operators $\mathbf{O}$, $\mathbf{P}$, $\blacktriangle$ and $\triangle$. The set of well-formed formulae of CTL*, WFF_C, is defined as follows:

- **false** and any element of PROP is in WFF_C;
- if ϕ and ψ are in WFF_C then the following are in WFF_C: $\neg\phi$; $\phi \wedge \psi$; $\mathbf{A}\phi$ $\bigcirc\phi$; $\phi\,\mathcal{U}\,\psi$.

CTL* formulae are evaluated over CTL* structures and do not have the O or $\blacktriangle$ operator, otherwise the semantics

[1]The original definition of RoCTL* had two accessibility relations, a success and a failure relation. Here we find the definition of [15], which defines RoCTL* structures as special CTL* structures, more convenient. These definitions are known to be equivalent [12].

of CTL* are the same as the semantics for RoCTL* defined above. State formulae are defined as follows:-

- **false** and any element of PROP are state formulae;
- If ϕ and ψ are state formulae and θ is any well-formed CTL* formula then the following are also state formulae: $\neg\phi$; $\phi \wedge \psi$; $\mathbf{A}\theta$; $\mathbf{E}\theta$;

CTL [6] is the fragment of CTL* such that every path operator is paired with a temporal operators.

- **false** and any element of PROP is in WFF$_{CTL}$;
- if ϕ and ψ are in WFF$_{CTL}$ then the following are in WFF$_{CTL}$: $\neg\phi$; $\phi \vee \psi$; $\mathbf{A}\phi\,\mathcal{U}\,\psi$; $\mathbf{E}\phi\,\mathcal{U}\,\psi$; $\mathbf{E}\bigcirc\phi$

where the other pairs of operators can be defined through standard equivalences, see for example [6].

C. Terminology and Notation

We say that a RoCTL* (CTL*) formula ϕ is satisfiable if and only if for some RoCTL* (CTL*) structure M and some path σ, $M, \sigma \models \phi$.

Given two formulae ϕ and ψ if ϕ is a subformula of ψ we denote this as $\phi \in sub(\psi)$.

The length of any formula ϕ, denoted $|\phi|$ is the number of occurrences of symbols other than "(" and ")" in ϕ.

If ϕ is a formula, let $\phi[\psi/p]$ be ϕ with every occurrence of ψ replaced with p.

A formula in Negated Normal Form contains no implications and has negations occurring only in front of propositions.

III. PAIR-ROCTL

The formulae of RoCTL$^{\mathrm{P}}$ are limited syntactically so that every path operator is paired with a temporal operator. The set of well-formed formulae of RoCTL$^{\mathrm{P}}$, WFF$_{PR}$, is defined as follows:

- **false** and any element of PROP is in WFF$_{PR}$;
- if ϕ and ψ are in WFF$_{PR}$ and $\mathbf{H} \in \{\mathbf{A}, \mathbf{O}, \blacktriangle\}$ then the following are in WFF$_{PR}$: $\neg\phi$; $\phi \wedge \psi$; $\mathbf{H}\bigcirc\phi$; $\mathbf{H}(\phi\,\mathcal{U}\,\psi)$; $\mathbf{H}(\phi\,\mathcal{W}\,\psi)$.

Next we provide a translation from CTL* into RoCTL$^{\mathrm{P}}$. This shows that the complexity of satisfiability of RoCTL$^{\mathrm{P}}$ is at least as hard as CTL* i.e. that it is harder than CTL to decide. We will create a new proposition f such that $\bigcirc f \Rightarrow \bigcirc\mathbf{v}$ and the subset of paths where f is true form a CTL* model; as f is not a special variable it can appear in RoCTL$^{\mathrm{P}}$ formulae. Note that when we are evaluating over entirely failing paths there are no deviations as every path has an infinite number of failures, and every deviation has a failure-free suffix. Then $\triangle\phi \equiv \phi$. This means that:

1) we can ensure that we are being evaluated over a fully failing path (one that satisfies $\Box f$) by use of $\triangle\Box f$;
2) we can represent any temporal operator (e.g F) in RoCTL$^{\mathrm{P}}$ by prefixing it with $\triangle$ (e.g. $F\phi \equiv \triangle F\phi$).

Let ϕ be a CTL* formula in negation normal form. We will now define κ_f, τ_y, $\mathbf{prev}$, τ_X, τ_1, and τ_p which we will use to define a translation function τ such that $\tau(\phi)$ is satisfiable iff ϕ is satisfiable. The formula κ_f ensures that the f variable remains false once it becomes false, that f is only true if the last transition was a failure transition and that the subset of the states S that satisfy f true is serial.

$$\begin{aligned}\kappa_f \;=\; & \mathbf{A}\,\Box(\neg f \Rightarrow \mathbf{A}\bigcirc\neg f)\wedge \\ & \mathbf{A}\,\Box\mathbf{O}\bigcirc\neg f \wedge \mathbf{A}\,\Box(f \Rightarrow \mathbf{E}\bigcirc f).\end{aligned}$$

The formula $\tau_y(\phi)$, defined below, is used to encode the state-formulae of ϕ into variables. It ensures that each variable of the form $y_{\mathbf{A}\psi}$ is only true at those states that satisfy $\mathbf{A}\psi$. Likewise each variable of the form $y_{\mathbf{E}\psi}$ is only true at those states that satisfy $\mathbf{E}\psi$.

$$\begin{aligned}\tau_y(\phi) \;=\; & \textstyle\bigwedge_{\mathbf{A}\psi \le \phi} \mathbf{A}\,\Box\left((y_{\mathbf{A}\psi} \wedge (\triangle\,\Box f)) \Rightarrow \tau_1(\psi)\right) \wedge \\ & \textstyle\bigwedge_{\mathbf{E}\psi \le \phi} \mathbf{A}\,\Box\left(y_{\mathbf{E}\psi} \Rightarrow \mathbf{E}\bigcirc\mathbf{prev}\left(\tau_1(\psi)\right)\right).\end{aligned}$$

We now define the function $\mathbf{prev}$ from formulae to formulae with the intention that $\bigcirc\mathbf{prev}(\psi) \iff \psi$ on all paths through our structure and all relevant formulae ψ.

$$\mathbf{prev}(\psi) \;=\; \psi[p/p'] \qquad \text{for all } p \in sub(\psi)$$

The translation above replaces each occurrence of p with p', relying on p' being true exactly when p was true at the last state. We define the function τ_X below from formulae to formulae for the purpose of ensuring that this holds for each variable of the form p' that occurs in some formula ψ.

$$\begin{aligned}\tau_X(\psi) \;=\; & \textstyle\bigwedge_{p' \in sub(\psi)} (p \Rightarrow \mathbf{A}\bigcirc p') \wedge \\ & \textstyle\bigwedge_{p' \in sub(\psi)} (\neg p \Rightarrow \mathbf{A}\bigcirc\neg p').\end{aligned}$$

We can now define τ_p, which is used to add three different types of proposition, and ensures that they are true only at the desired states.

$$\tau_p(\phi) \;=\; \kappa_f \wedge \tau_y(\phi) \wedge \tau_X(\tau_y(\phi)).$$

We now define the τ_1 translation such that $\tau_1(\phi)$ is satisfiable on the class of RoCTL* structures that satisfy $\tau_p(\phi)$ iff ϕ is satisfiable on the class of all CTL-structures.

$$\begin{aligned}\tau_1(\mathbf{A}\psi) &= y_{\mathbf{A}\psi} \wedge \triangle\,\Box f \\ \tau_1(\mathbf{E}\psi) &= y_{\mathbf{E}\psi} \wedge \triangle\,\Box f \\ \tau_1(\bigcirc\psi) &= (\triangle\bigcirc\tau_1(\psi)) \\ \tau_1(\psi * \phi) &= \triangle[\tau_1(\psi) * \tau_1(\phi)], * \in \mathcal{U}, \mathcal{W} \\ \tau_1(p) &= (p \wedge \triangle\,\Box f) \\ \tau_1(\neg p) &= (\neg p \wedge \triangle\,\Box f)\end{aligned}$$

Finally we define $\tau(\phi)$ itself as follows:

$$\tau(\phi) \;=\; \tau_1(\phi) \wedge \tau_p(\phi).$$

Translating a CTL* Model into a RoCTL^P Model: Given a model $M = (S, R, L)$ for CTL* formula, we construct a RoCTL* model structure $M^{\mathfrak{R}} = \left(S^{\mathfrak{R}}, R^{\mathfrak{R}}, L^{\mathfrak{R}}\right)$ from M as follows:

- We add a new "success" state s so that $S^{\mathfrak{R}} = S \cup \{s\}$.
- The accessibility relation $R^{\mathfrak{R}}$ is the least relation that satisfies $R \subseteq R^{\mathfrak{R}}$ and $\langle w, s\rangle \in R^{\mathfrak{R}}$ for all $w \in S^{\mathfrak{R}}$.

- The valuation $L^{\mathfrak{R}}$ satisfies the following:
  - For every proposition p in the original formula ϕ and $w \in S$ we have $p \in L^{\mathfrak{R}}(w)$ iff $p \in L(w)$.
  - The failure proposition f and violation proposition $\mathbf{v}$ is true at every state except the success state. Formally, $f \in L^{\mathfrak{R}}(w)$ iff $w \neq s$ and $\mathbf{v} \in L^{\mathfrak{R}}(w)$ iff $w \neq s$.
  - For every proposition of the form y_ψ in $\tau_1(\phi)$ and every state $w \in S$, $y_\psi \in L^{\mathfrak{R}}(w)$ iff the formulae ψ holds at the state w.

We will now define a function h to add the p' propositions into the model.

Definition 3. *We define a function h from RoCTL* structures to RoCTL* structures such that for any RoCTL* structure $M = (S, R, L)$ we have $h(M) = \left(S^h, R^h, L^h\right)$ where:*

1) $S^h = R$, so every state in $h(M)$ is of the form (w, v) where w and v are in S (i.e. states of M). Being in state (w, v) means roughly "we are currently at state v but were at state w previously".
2) For any pair of states (w, v) and (x, y) in S^h we have $(w, v) R^h (x, y)$ iff $x = v$ and $(x, y) \in R$.
3) For all $p \in$ PROP, it is the case that $p \in L^h(\langle w, v\rangle) \iff p \in L(v)$ and $p' \in L^h(\langle w, v\rangle) \iff p \in L(w)$.

We will use $\langle ?, w\rangle$ to represent $\langle v, w\rangle$ for some arbitrary v, when we do not care about truth values of the p' variables at this state. For convenience we extend the definition of h such that $h(\sigma) = \langle ?, \sigma_0\rangle, \langle \sigma_0, \sigma_1\rangle, \langle \sigma_1, \sigma_2\rangle \ldots$, and $h(M, \sigma) = h(M), h(\sigma)$.

Lemma 1. *For all RoCTL* structures M, fullpaths σ through M and RoCTL* formulae ϕ (not including propositions of the form p') we have*

$$\begin{aligned} M, \sigma \models \phi &\iff h(M, \sigma) \models \phi \\ &\iff h(M, \sigma) \models \bigcirc \mathbf{prev}(\phi). \end{aligned}$$

Proof: From the definition of the function h. ■

Note that it is possible to add the p' variables without defining the function h if the model is an infinite tree. Here we provide the function h to allow finite models.

Next we show that the translation preserves satisfiability. Without loss of generality we can assume that each structure has a state w_0 such that every other state is reachable from that state, so for example $M, w_0 \models \mathbf{A} \Box p$ means that p is true at all states in M.

Lemma 2. *For any RoCTL* structure M that satisfies $M, w_0 \models \mathbf{A} \Box \mathbf{O} \bigcirc (\neg f)$, it is the case that if $M, \pi \models \triangle \Box f$ then $M, \pi \models \Box f$ and if $M, \pi \models \triangle \tau_1(\psi)$ then $M, \pi \models \tau_1(\psi)$ for all paths π through M and formulae ψ.*

Proof: As $M, \pi \models \triangle \Box f$ either $M, \pi \models \Box f$ or there exists a deviation σ from π that satisfies $\Box f$. Any deviation σ from π has a failure-free suffix. That is there exists i such that $\sigma_{\geq i}$ is failure-free. As $M, w_0 \models \mathbf{A} \Box \mathbf{O} \bigcirc \neg f$, it is the case that $M, \sigma_{\geq i} \models \mathbf{O} \bigcirc \neg f$, and as $\sigma_{\geq i}$ is failure-free, $M, \sigma_{\geq i+1} \models \neg f$ and so $M, \sigma \not\models \Box f$. Hence, by contradiction, $M, \pi \models \Box f$.

As every deviation σ has a failure-free suffix, $M, \sigma \not\models ((\neg) p \wedge \triangle \Box f)$ and by recursion $M, \sigma \not\models \tau_1(\psi)$ for any ψ. It follows that if $M, \pi \models \triangle \tau_1(\psi)$ then $M, \pi \models \tau_1(\psi)$. ■

Lemma 3. *Let $M^{\mathfrak{R}}, \sigma \models \tau_1(\psi)$ for some path σ through M^R then $s \neq \sigma_i$ for any $i \geqslant 0$.*

Proof: For any fullpath π such that $M^R, \pi \not\models \triangle \Box f$ we see that for ψ of the form $\mathbf{A}\theta$, $\mathbf{E}\theta$, p or $\neg p$ it is the case that $M^R, \pi \not\models \tau_1(\psi)$. If there exists an integer i such that $\sigma_i = s$ we see that $M^R, \sigma_j \not\models \Box f$ for any $j \in \mathbf{N}$, and from Lemma 2 we have $M^R, \sigma_j \not\models \triangle \Box f$. By induction, we see that $M^R, \sigma \not\models \tau_1(\psi)$, for any formulae ψ. ■

Lemma 4. *For any CTL* formula ϕ, $h\left(M^{\mathfrak{R}}, \pi\right) \models \tau(\phi)$ iff $M, \pi \models \phi$.*

Proof: ($\Longleftarrow$) It is easy to see that for every path σ through M we have $M^{\mathfrak{R}}, \sigma \models \triangle \Box f$ and so for every $p \in$ PROP, it is the case that if $M, \sigma \models (\neg) p$ then $M^{\mathfrak{R}}, \sigma \models ((\neg) p \wedge \triangle \Box f)$. Thus for all formulae ψ that consist of a single (possibly negated) proposition it is the case that if $M, \sigma \models \psi$ then $M^{\mathfrak{R}}, \sigma \models \tau_1(\psi)$ and by Lemma 1, $h\left(M^{\mathfrak{R}}, \sigma\right) \models \tau_1(\psi)$. Assume that if $M, \sigma \models \psi$ then $h\left(M^{\mathfrak{R}}, \sigma\right) \models \tau_1(\psi)$ for all ψ of length less than n. Now consider the case where $|\psi| = n + 1$. We provide the cases for $\mathcal{U}$ and $\mathbf{E}$. The other cases are similar.

$\psi = \alpha U \beta$ For all σ it is the case that if $M, \sigma \models \alpha$ then $M^{\mathfrak{R}}, \sigma \models \tau_1(\alpha)$ and if $M, \sigma \models \beta$ then $M^{\mathfrak{R}}, \sigma \models \tau_1(\beta)$. If $M, \sigma \models \alpha U \beta$ then there exists $i \in \mathbf{N}$ such that $M, \sigma_{\geq i} \models \beta$ and for all $j \in \mathbf{N}$ such that $j < i$, $M, \sigma_{\geq j} \models \alpha$. Thus $M^{\mathfrak{R}}, \sigma_{\geq i} \models \tau_1(\beta)$ and for all $j \in \mathbf{N}$ such that $j < i$, $M^{\mathfrak{R}}, \sigma_{\geq j} \models \tau_1(\alpha)$. Thus $M^{\mathfrak{R}}, \sigma \models \tau_1(\alpha) U \tau_1(\beta)$, so $M^{\mathfrak{R}}, \sigma \models \triangle(\tau_1(\alpha) U \tau_1(\beta)) = \tau_1(\psi)$.

$\psi = \mathbf{E}\alpha$ From the construction of $M^{\mathfrak{R}}$, $M, \sigma \models \mathbf{E}\alpha$ iff $M^{\mathfrak{R}}, \sigma \models y_{\mathbf{E}\alpha}$. As σ is a path through M, it does not contain the success state and so $M^{\mathfrak{R}}, \sigma \models \Box f$ and so if $M, \sigma \models \mathbf{E}\alpha$ then $M^{\mathfrak{R}}, \sigma \models y_{\mathbf{E}\alpha} \wedge \triangle \Box f = \tau_1(\psi)$.

So by induction that $M^{\mathfrak{R}}, \sigma \models \tau_1(\phi)$ and by Lemma 1, $h\left(M^{\mathfrak{R}}, \sigma\right) \models \tau_1(\phi)$. Note that cases for $\mathbf{A}\alpha$ and $\mathbf{E}\alpha$ are trivial. However, further conditions relating to the newly introduced propositions $y_{\mathbf{A}\alpha}$ and $y_{\mathbf{E}\alpha}$ are required in the definition of $\tau_y(\phi)$. Now $y_{\mathbf{E}\psi}$ occurs exactly on those σ_0 where $M, \sigma \models \mathbf{E}\psi$. Thus $M^{\mathfrak{R}}, \sigma \models \mathbf{E}\tau_1(\psi)$ and $h\left(M^{\mathfrak{R}}, \sigma\right) \models \mathbf{E}\tau_1(\psi)$, thus by Lemma 1, $h\left(M^{\mathfrak{R}}, \sigma\right) \models \mathbf{E} \bigcirc \mathbf{prev}(\tau_1(\psi))$. Likewise $y_{\mathbf{A}\psi}$ occurs exactly on those σ_0 where $M, \sigma \models \mathbf{A}\psi$, that is, for every path σ' through M such that $\sigma'_0 = \sigma_0$ it is the case that $M, \sigma' \models \psi$ and

so $M^{\mathfrak{R}}, \sigma' \models \tau_1(\psi)$ and $M^{\mathfrak{R}}, \sigma' \models (y_{\mathbf{A}\phi} \wedge (\triangle \Box f)) \Rightarrow \tau_1(\psi)$. If the path σ is not through M, it contains the success state s, and so $\sigma \not\models \triangle \Box f$ and so again $M^{\mathfrak{R}}, \sigma \models (y_{\mathbf{A}\phi} \wedge (\triangle \Box f)) \Rightarrow \tau_1(\psi)$.

It is now easy to show that the way $M^{\mathfrak{R}}$ is constructed ensures that $h\left(M^{\mathfrak{R}}, \sigma\right) \models \tau_p(\phi)$, and since $h\left(M^{\mathfrak{R}}, \sigma\right) \models \tau_1(\phi)$, it follows that $h\left(M^{\mathfrak{R}}, \sigma\right) \models \tau(\phi)$.

($\Longrightarrow$) If $\psi = (\neg) p$ then $\tau_1(\psi) = ((\neg) p \wedge \triangle \Box f)$. Clearly if $M^{\mathfrak{R}}, \sigma \models ((\neg) p \wedge \triangle \Box f)$ then $M^{\mathfrak{R}}, \sigma \models (\neg) p$ and $M, \sigma \models (\neg) p = \psi$.

Assume that for all paths σ through M and ψ with $|\psi| \leq n$, it is the case that if $M^{\mathfrak{R}}, \sigma \models \tau_1(\psi)$ then $M, \sigma \models \psi$. Now consider the case where $|\psi| = n+1$. We provide the cases for $\mathcal{U}$ and $\mathbf{E}$. The others are similar.

- $\psi = \alpha U \beta$ For all σ it is the case that if $M^{\mathfrak{R}}, \sigma \models \tau_1(\alpha)$ then $M, \sigma \models \alpha$ and if $M^{\mathfrak{R}}, \sigma \models \tau_1(\beta)$ then $M, \sigma \models \beta$. Let $M^{\mathfrak{R}}, \sigma \models \tau_1(\alpha U \beta) = \triangle[\tau_1(\alpha) U \tau_1(\beta)]$ then from Lemma 2 it follows that $M^{\mathfrak{R}}, \sigma \models \tau_1(\alpha) U \tau_1(\beta)$ and so there exists $i \in \mathbf{N}$ such that $M, \sigma_{\geq i} \models \beta$ and for all $j < i$ $M, \sigma_{\geq j} \models \alpha$. Thus $M^{\mathfrak{R}}, \sigma_{\geq i} \models \tau_1(\alpha)$ and for all $j < i$ $M^{\mathfrak{R}}, \sigma_{\geq j} \models \tau_1(\beta)$. Thus $M, \sigma \models \alpha U \beta$.
- $\psi = \mathbf{E}\alpha$ By definition, $M, \sigma \models \mathbf{E}\alpha$ iff $M^{\mathfrak{R}}, \sigma \models y_{\mathbf{E}\alpha} = \tau_1(\psi)$. Clearly if $M^{\mathfrak{R}}, \sigma \models y_{\mathbf{E}\alpha} \wedge \triangle \Box f$ then $M^{\mathfrak{R}}, \sigma \models y_{\mathbf{E}\alpha}$ and so $M, \sigma \models \mathbf{E}\alpha$.

By Lemma 1, if $h(M^{\mathfrak{R}}, \sigma) \models \tau_1(\psi)$ then $M^{\mathfrak{R}}, \sigma \models \tau_1(\psi)$ and by induction if $M^{\mathfrak{R}}, \sigma \models \tau_1(\psi)$ then $M, \sigma \models \psi$ for all ψ so if $h(M^{\mathfrak{R}}, \sigma) \models \tau_1(\psi)$ then $M, \sigma \models \psi$. ■

Given any CTL* structure M we can construct the RoCTL* structure $M^{\mathfrak{R}}$, and so from Lemma 4 it is clear that $\tau(\phi)$ is satisfiable if ϕ is satisfiable. We have not yet shown that for any RoCTL* structure we can construct an equivalent CTL* structure; we will do so in Lemma 5 below and so it is clear that ϕ is satisfiable, then $\tau(\phi)$ is satisfiable. By combining these we get Theorem 1.

Definition 4. *Given a model M for an RoCTL^P formula $\tau(\phi)$ we construct a model M^C for the CTL* formula ϕ as follows: remove all states where f is false. Optionally, also remove y_ψ, p' and f.*

Lemma 5. *For any RoCTL* structure M, if $M, w_0 \models \tau(\phi)$ then $M^C, w_0 \models \phi$ where M^C is the translation of M from the Definition 4; i.e. M^C is M with the states that do not satisfy f removed.*

Proof: This lemma is very similar to the ($\Rightarrow$) direction of the previous lemma.

As $M \models \tau(\phi)$, clearly $M \models \tau_p(\phi)$. Since $M \models \tau_p(\phi)$ we see that $M, w_0 \models \mathbf{A} \Box \mathbf{O} \bigcirc (\neg f)$ from Lemma 2 and again it is the case that if $M, \pi \models \triangle \Box f$ then $M, \pi \models \Box f$ and if $M, \pi \models \triangle \tau_1(\psi)$ then $M, \pi \models \tau_1(\psi)$ for all CTL* formulae ψ and fullpaths π through M.

Assume that for all paths σ through M and ψ with $|\psi| \leq n$ for some integer n, it is the case that if $M, \sigma \models \tau_1(\psi)$ then $M, \sigma \models \psi$. Now consider the case where $|\psi| = n+1$. Where ψ is of the form $(\neg) p$, $\alpha \mathcal{W} \beta$, $\alpha U \beta$ or $\bigcirc \alpha$ we see that $M, \sigma \models \tau_1(\psi) \Rightarrow M, \sigma \models \psi$ using the same arguments made in Lemma 4.

Now assume that ψ is of the form $\mathbf{A}\theta$, and that $M, \sigma \models \tau_1(\psi) = y_{\mathbf{A}\theta} \wedge \triangle \Box f$. From τ_y we know that $\mathbf{A} \Box((y_{\mathbf{A}\theta} \wedge (\triangle \Box f)) \Rightarrow \tau_1(\theta))$. Consider a path π through M^C such that $\pi_0 = \sigma_0$, i.e. $\pi \in \mathbf{ap}(\sigma_0)$. We know that $M^C, \pi 0 \models \Box f$ as f is true at every state in M^C, so likewise $M, \pi \models \Box f$ and $M, \pi \models \triangle \Box f$. From $\tau_1(\phi)$ we know that $M, \pi \models y_{\mathbf{A}\theta}$ and from τ_y we know that $\mathbf{A} \Box((y_{\mathbf{A}\theta} \wedge (\triangle \Box f)) \Rightarrow \tau_1(\theta))$, hence $M, \pi \models \tau_1(\theta)$. Since $|\theta| \leqslant n$ it follows that $M^C, \pi \models \theta$, for all $\pi \in \mathbf{ap}(\sigma_0)$. Thus $M^C, \sigma \models \mathbf{A}\theta$. Assume that ψ is of the form $\mathbf{E}\theta$ and $M, \sigma \models \tau_1(\psi) = y_{\mathbf{E}\theta} \wedge \triangle \Box f$. As $M, \sigma \models y_{\mathbf{E}\theta}$, from τ_y we know that $M, \sigma \models \mathbf{E} \bigcirc \mathbf{prev}(\tau_1(\theta))$. As $M, \sigma \models \tau_X(\tau_y(\phi))$, we see that $M, \sigma \models \mathbf{E}(\tau_1(\theta))$. Finally, since $|\theta| \leqslant n$ we know that $M, \sigma \models \mathbf{E}\theta$. ■

Theorem 1. *$\tau(\phi)$ is satisfiable in RoCTL^P* iff ϕ is satisfiable in CTL*.

IV. State-RoCTL

The formulae of RoCTL$^{\mathrm{S}}$ are state formulae because all temporal operators and robustly or prone operators must be in the scope of either a CTL path quantifier or obligatory or permissible operator. To define the set of well-formed formulae of RoCTL$^{\mathrm{S}}$, WFF$_{SR}$, we define both state and path formulae as follows:

- **false** and any element of PROP is in the set of state formulae;
- if α and β are state formulae and θ is a path formula then the following are state formulae $\neg\alpha$; $\alpha \wedge \beta$; $\mathbf{P} \bigcirc \alpha$; $\mathbf{E} \bigcirc \alpha$; $\mathbf{O}\theta$; $\mathbf{P}\theta$; $\mathbf{A}\theta$; $\mathbf{E}\theta$;
- if α and β are state formulae and θ is a path formula then the following are path formulae $\blacktriangle\theta$; $\triangle\theta$; $\alpha \mathcal{U} \beta$.

The set of well-formed formulae of RoCTL$^{\mathrm{S}}$, WFF$_{SR}$, is defined as the set of state formulae.

Note that as well as the usual abbreviations, in RoCTL$^{\mathrm{S}}$ we treat $\blacktriangle \bigcirc \alpha$ and $\triangle \bigcirc \alpha$ as abbreviations as follows $\blacktriangle \bigcirc \alpha \equiv \mathbf{A} \bigcirc \alpha$ and $\triangle \bigcirc \alpha \equiv \mathbf{E} \bigcirc \alpha$. Further we treat $\mathbf{H}_1 \ldots \mathbf{H}_2 \bigcirc \alpha$ as an abbreviation for $\mathbf{H}_2 \bigcirc \alpha$ when α is a state formula, "…" is a sequence of operators in $\{\triangle, \blacktriangle\}$ and $\mathbf{H}_1, \mathbf{H}_2 \in \{\mathbf{A}, \mathbf{E}, \mathbf{O}, \mathbf{P}\}$. In this section (as above) we use α and β to denote state formulae, θ to denote path formulae and ϕ and ψ to denote either.

We will now define a translation from RoCTL$^{\mathrm{S}}$ to CTL. To understand how the translation to CTL works, consider the formula $\triangle(\alpha \mathcal{U} \beta)$. If $M, \sigma \models \triangle(\alpha \mathcal{U} \beta)$ then either $M, \sigma \models \alpha \mathcal{U} \beta$ or there exists an i-deviation from σ, say π, such that $M, \pi_{\geqslant i+1} \models \alpha \mathcal{U} \beta$. Since π is an i-deviation $\pi_{\geqslant i+1}$ is failure-free and so $\pi_{i+1} \models \mathbf{P}(\alpha \mathcal{U} \beta)$, thus the

state formula $\mathbf{E}\bigcirc\mathbf{P}(\alpha\,\mathcal{U}\,\beta)$ holds at σ_i. Thus along the path σ the state-formula α holds until $\mathbf{E}\bigcirc\mathbf{P}(\alpha\,\mathcal{U}\,\beta)$ holds together with α. The translation will remove $\triangle$ operators by replacing $\triangle(\alpha\,\mathcal{U}\,\beta)$ with $\alpha\,\mathcal{U}\,(\beta\vee(\alpha\wedge\mathbf{E}\bigcirc\mathbf{P}(\alpha\,\mathcal{U}\,\beta)))$. The $\blacktriangle$ operator is the dual of the $\triangle$, so it will be handled similarly.

We now define a translation function τ from RoCTL$^{\mathrm{S}}$ state and path formulae to CTL state and path formulae respectively. For any variable p we let $\tau(p)=p$. For any RoCTL$^{\mathrm{S}}$ state formulae α and β define τ as follows:

$$\begin{array}{rcl}
\tau(\mathbf{E}\bigcirc\alpha) & = & \mathbf{E}\bigcirc\tau(\alpha)\\
\tau(\mathbf{P}\bigcirc\alpha) & = & \mathbf{E}\bigcirc(\tau(\alpha)\wedge\neg\mathbf{v})\\
\tau(\neg\alpha) & = & \neg\tau(\alpha)\\
\tau(\alpha\wedge\beta) & = & \tau(\alpha)\wedge\tau(\beta)\\
\tau(\alpha\,\mathcal{U}\,\beta) & = & \tau(\alpha)\,\mathcal{U}\,\tau(\beta)\\
\tau(\blacktriangle(\alpha\,\mathcal{U}\,\beta)) & = & \tau(\alpha\wedge\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta))\,\mathcal{U}\,\tau(\beta)\\
\tau(\triangle(\alpha\,\mathcal{U}\,\beta)) & = & \tau(\alpha)\,\mathcal{U}\,\tau(\beta\vee(\alpha\wedge\mathbf{E}\bigcirc\mathbf{P}(\alpha\,\mathcal{U}\,\beta))).
\end{array}$$

For any RoCTL$^{\mathrm{S}}$ path-formula θ :

$$\begin{array}{rcl}
\tau(\mathbf{A}\theta) & = & \mathbf{A}\tau(\theta)\\
\tau(\mathbf{E}\theta) & = & \mathbf{E}\tau(\theta).
\end{array}$$

Next considering the operators $\mathbf{O}$ and $\mathbf{P}$ we will first consider the case where $\alpha\,\mathcal{U}\,\beta$ is nested directly inside the $\mathbf{O}$ or $\mathbf{P}$ operators:

$$\begin{array}{rcl}
\tau(\mathbf{O}(\alpha\,\mathcal{U}\,\beta)) & = & \tau(\beta)\vee(\tau(\alpha)\wedge\\
 & & \mathbf{A}\bigcirc\mathbf{A}(\tau(\alpha)\,\mathcal{U}\,(\tau(\beta)\vee\mathbf{v})))\\
\tau(\mathbf{P}(\alpha\,\mathcal{U}\,\beta)) & = & \tau(\beta)\vee(\tau(\alpha)\wedge\\
 & & \mathbf{E}\bigcirc\mathbf{E}((\tau(\alpha)\wedge\neg\mathbf{v})\,\mathcal{U}\,(\tau(\beta)\wedge\neg\mathbf{v})).
\end{array}$$

We handle the case where $\blacktriangle$ or $\triangle$ occur inside $\mathbf{O}$ or $\mathbf{P}$ by specifying that for all path-formulae θ not of the form $\alpha\,\mathcal{U}\,\beta$:

$$\begin{array}{rcl}
\tau(\mathbf{O}\theta) & = & \tau(\mathbf{O}\tau(\theta))\\
\tau(\mathbf{P}\theta) & = & \tau(\mathbf{P}\tau(\theta))
\end{array}$$

and similarly for $\blacktriangle$ and $\triangle$:

$$\begin{array}{rcl}
\tau(\blacktriangle\theta) & = & \tau(\blacktriangle\tau(\theta))\\
\tau(\triangle\theta) & = & \tau(\triangle\tau(\theta)).
\end{array}$$

Definition 5. *We define a partial ordering $<$ on RoCTL* formulae such that $\phi<\psi$ if ϕ has less $\blacktriangle$ (or $\triangle$) operators than ψ, and $\phi<\psi$ if ϕ and ψ have the same number of $\blacktriangle$ or $\triangle$ operators and $|\phi|<|\psi|$. Otherwise $\phi\not<\psi$ and $\psi\not<\phi$.*

Lemma 6. *For all ϕ in the domain of τ we have $\phi\equiv\tau(\phi)$.*

Proof: Assume that ϕ is the simplest counter example in the sense that $\phi\not\equiv\tau(\phi)$ and $\psi\equiv\tau(\psi)$ for all $\psi<\phi$. Let M be some arbitrary RoCTL* structure M and σ be an arbitrary path through M. Assume that ϕ is of the form $\blacktriangle(\alpha\,\mathcal{U}\,\beta)$.

($\Longrightarrow$) Assume that $M,\sigma\models\blacktriangle(\alpha\,\mathcal{U}\,\beta)$ and $M,\sigma\not\models\tau(\blacktriangle(\alpha\,\mathcal{U}\,\beta))$. Thus

$$M,\sigma\not\models\tau(\alpha\wedge\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta))\,\mathcal{U}\,\tau(\beta)$$

and since ϕ is the simplest counter example

$$M,\sigma\not\models(\alpha\wedge\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta))\,\mathcal{U}\,\beta.$$

As we have chosen the simplest counter example there exists some k such that $M,\sigma_k\models\beta$ and there must exist an integer i such that for all $j\leqslant i<k$ we have $M,\sigma_j\not\models\beta$, and

$$M,\sigma_i\not\models(\alpha\wedge\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta)).$$

Assume that $M,\sigma_i\not\models\alpha$, then we see that $M,\sigma\not\models\alpha\,\mathcal{U}\,\beta$ and so $M,\sigma\not\models\blacktriangle(\alpha\,\mathcal{U}\,\beta)$. By contradiction $M,\sigma_i\models\alpha$ and hence

$$M,\sigma_i\not\models\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta).$$

The sequence of operators $\mathbf{A}\bigcirc\mathbf{O}$ quantifies over exactly those fullpaths that are failure-free after the first step. Thus there exists an i-deviation π from σ such that $M,\pi_{\geqslant i}\not\models\bigcirc\alpha\,\mathcal{U}\,\beta$ and equivalently $M,\pi_{\geqslant i+1}\not\models\alpha\,\mathcal{U}\,\beta$. Recall that $M,\sigma_j\not\models\beta$ for all $j\leqslant i$; since β is a state-formula and $\sigma_{\leqslant i}=\pi_{\leqslant i}$ it follows that $M,\pi_j\not\models\beta$. Combining this with the fact that $M,\pi_{\geqslant i+1}\not\models\alpha\,\mathcal{U}\,\beta$ we find that $M,\pi\not\models\alpha\,\mathcal{U}\,\beta$. Since π is a deviation from σ we find that $M,\sigma\not\models\blacktriangle(\alpha\,\mathcal{U}\,\beta)$, which contradicts our original assumption.

($\Longleftarrow$) Assume that $M,\sigma\not\models\blacktriangle(\alpha\,\mathcal{U}\,\beta)$ and $M,\sigma\models\tau(\blacktriangle(\alpha\,\mathcal{U}\,\beta))$. Since $M,\sigma\not\models\blacktriangle(\alpha\,\mathcal{U}\,\beta)$ either $M,\sigma\not\models\alpha\,\mathcal{U}\,\beta$ or there exists a deviation π from σ such that $M,\pi\not\models\alpha\,\mathcal{U}\,\beta$. If $M,\sigma\not\models\alpha\,\mathcal{U}\,\beta$ then clearly $M,\sigma\not\models(\alpha\wedge\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta))\,\mathcal{U}\,\beta$, by contradiction there must exist an i-deviation π from σ such that $M,\pi\not\models\alpha\,\mathcal{U}\,\beta$.

Since $M,\sigma\models(\alpha\wedge\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta))\,\mathcal{U}\,\beta$ we see that there exists n such that $M,\sigma_n\models\beta$ and for all $m<n$ it is the case that

$$M,\sigma_m\models(\alpha\wedge\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta)).$$

Assume that $n\leqslant i$. Since $M,\sigma_m\models\alpha$ for all $m<n$ and $M,\sigma_n\models\beta$, we can see that as $\pi_{\leqslant i}=\sigma_{\leqslant i}$ it must also be the case that $M,\pi\models(\alpha\,\mathcal{U}\,\beta)$.

However, recall that we chose the path π such that $M,\pi\not\models\alpha\,\mathcal{U}\,\beta$. By contradiction we know that $n>i$.

Since $n>i$ we know that for all $j\leqslant i$ it is the case that $M,\sigma_j\models\alpha$ and $\sigma_j=\pi_j$. From this and the fact that $M,\pi\not\models\alpha\,\mathcal{U}\,\beta$ it follows that $M,\pi_{\geqslant i+1}\not\models\alpha\,\mathcal{U}\,\beta$. Since $\pi_{\geqslant i+1}$ is failure-free we see that $M,\pi_{i+1}\not\models\mathbf{O}(\alpha\,\mathcal{U}\,\beta)$ and

$$M,\pi_i\not\models\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta).$$

Since $\pi_i=\sigma_i$ we also have $M,\sigma_i\not\models\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta)$. However, recall that

$$M,\sigma_m\models(\alpha\wedge\mathbf{A}\bigcirc\mathbf{O}(\alpha\,\mathcal{U}\,\beta)),$$

for all $m<n$. By contradiction we see that the smallest counter-example ϕ cannot be of the form $\blacktriangle(\alpha\,\mathcal{U}\,\beta)$.

The proof for the case where ϕ is of the form $\triangle(\alpha\,\mathcal{U}\,\beta)$ is similar. Let ϕ be of the form $\triangle(\alpha\,\mathcal{U}\,\beta)$ then recall that

$$\tau(\triangle(\alpha\,\mathcal{U}\,\beta))=\tau(\alpha)\,\mathcal{U}\,\tau(\beta\vee(\alpha\wedge\mathbf{E}\bigcirc\mathbf{P}(\alpha\,\mathcal{U}\,\beta))).$$

If $M, \sigma \models \triangle(\alpha \mathcal{U} \beta)$ and $M, \sigma \not\models \tau(\triangle(\alpha \mathcal{U} \beta))$, then

$$M, \sigma \not\models \alpha \mathcal{U} (\beta \vee (\alpha \wedge \mathbf{E}\bigcirc\mathbf{P}(\alpha \mathcal{U} \beta))),$$

and clearly $M, \sigma \not\models \alpha \mathcal{U} \beta$. Thus there exists an i-deviation π from σ such that $M, \pi \models \alpha \mathcal{U} \beta$, for some $i \in \mathbf{N}$. Since $M, \sigma \not\models \alpha \mathcal{U} \beta$ we see that $M, \sigma_j \not\models \beta$ for any $j \leqslant i$. Thus $M, \pi_{\geqslant i+1} \models \alpha \mathcal{U} \beta$, and since $\pi_{\geqslant i+1}$ is failure-free we know that $M, \pi_{i+1} \models \mathbf{P}(\alpha \mathcal{U} \beta)$. Thus $M, \sigma_i \models \mathbf{E}\bigcirc\mathbf{P}(\alpha \mathcal{U} \beta)$. Since $M, \pi \models \alpha \mathcal{U} \beta$ and β is not satisfied before π deviates from σ we know that for each $j \leqslant i$ we have $M, \sigma_j \models \alpha$. Hence $M, \sigma \models (\alpha \mathcal{U} \alpha \wedge \mathbf{E}\bigcirc\mathbf{P}(\alpha \mathcal{U} \beta))$ and so $M, \sigma \models \tau(\triangle(\alpha \mathcal{U} \beta))$.

There are many other possible forms for ϕ, but we see that the translations for all of these are trivial. For example, consider the case where $\phi = \mathbf{A}\psi$. Recall that $\tau(\mathbf{A}\psi) = \mathbf{A}\tau(\psi)$, and since $\psi < \phi$ it is obvious that

$$\begin{array}{rl} M, \sigma \models \mathbf{A}\tau(\psi) & \iff \forall \pi \in \mathbf{ap}(\sigma_0)(M, \pi \models \psi) \\ & \iff M, \sigma \models \mathbf{A}\psi. \end{array}$$

Likewise $M, \sigma \models \tau(\phi) \iff M, \sigma \models \phi$ for the other forms of ϕ. By contradiction the lemma holds. ∎

We have shown that τ is a truth-preserving translation from RoCTL$^{\mathrm{S}}$ to CTL. To produce a satisfaction preserving translation we can add the cause $\mathbf{A}\,\Box\mathbf{E}\bigcirc\neg\mathbf{v}$, which ensures the translated formulae are only satisfied by RoCTL* structures. Strictly speaking the translation into CTL isn't linear. For example, consider $\tau(\triangle(\alpha \mathcal{U} \beta)) = \tau(\alpha) \mathcal{U} \tau(\beta \vee (\alpha \wedge \mathbf{E}\bigcirc\mathbf{P}(\alpha \mathcal{U} \beta)))$. See that α and β occur several times on the RHS. However, since α and β are state formulae, it is well known that we can replace α and β with new propositions p_α and p_β and a clause requiring that

$$\mathbf{A}\,\Box(p_\alpha \iff \tau(\alpha) \wedge p_\beta \iff \tau(\beta)).$$

Theorem 2. *There exists a satisfiability preserving linear translation τ from RoCTL^S formulae into CTL formulae.*

Theorem 3. *Like CTL [EH82], the satisfiability decision problem for RoCTL^S is EXPTIME-complete.*

V. Examples

A number of examples using RoCTL* to specify problems are provided in [1]. We reformulate one of these, the well-known coordinated attack problem in RoCTL$^{\mathrm{S}}$.

In the coordinated attack problem we have two generals X and Y. General X wants to organize an attack with Y. A protocol will be presented such that a coordinated attack will occur if no more than one message is lost. In the following let s_i denote that General i sends a message; r_i denote that General i receives a message; and f_i denote that General i commits to an attack.

$\mathbf{A}\,\Box\,(s_X \Rightarrow \mathbf{O}\bigcirc r_Y)$: If X sends a message, Y should receive it at the next step.

$\mathbf{A}\,\Box\,(s_X \Rightarrow \mathbf{O}\bigcirc r_Y)$: If X sends a message, Y should receive it at the next step.

$\mathbf{A}\,\Box\,(\neg s_X \Rightarrow \neg\mathbf{E}\bigcirc r_Y)$: If X does not send a message now, Y will not receive a message at the next step.

$\mathbf{A}\,\Box(f_X \Rightarrow \mathbf{A}\,\Box f_X)$: If X commits to an attack, X cannot withdraw.

$\mathbf{A}\,\Box(f_X \Rightarrow \neg s_X)$: If X has committed to an attack, it is too late to send messages.

$\mathbf{A}\,(\neg f_X \,\mathcal{W}\, r_X)$: X cannot commit to an attack until X has received plans (from Y)

$\mathbf{A}(\neg r_X \,\mathcal{W}\, s_Y)$: X cannot receive a message until Y sends one.

Similar constraints to the above also apply to Y. Below we add a constraint requiring X to be the general planning the attack.

$\mathbf{A}\,(\neg s_Y \,\mathcal{W}\, r_Y)$: General Y will not send a message until Y has received a message.

No protocol satisfies the original coordination problem, since an unbounded number of messages can be lost. Here we only attempt to ensure correct behaviour if one or fewer messages are lost.

$\mathbf{A}\,(s_X \,\mathcal{U}\, r_X)$: General X will send plans until a response is received.

$\mathbf{A}\,\Box\,(r_X \Rightarrow f_X)$: Once general X receives a response, X will commit to an attack.

$\mathbf{A}\,(\neg r_Y \,\mathcal{W}\, (r_Y \wedge (s_Y \wedge \mathbf{A}\bigcirc s_Y \wedge \mathbf{A}\bigcirc\mathbf{A}\bigcirc f_Y)))$: Once general Y receives plans, Y will send two messages to X and then commit to an attack.

Let the conjunction of the formulae in the specification above be $\hat{\phi}$. We want to show that both generals attack at the same time (i.e. a coordinated attack) will occur if no more than one message is lost, i.e. $\mathbf{O}\blacktriangle\diamondsuit(f_X \wedge f_Y)$. We can show that $\hat{\phi}$ is satisfiable and that $\hat{\phi} \Rightarrow \mathbf{O}\blacktriangle\diamondsuit(f_X \wedge f_Y)$ is valid.

We see that the above example is in RoCTL$^{\mathrm{S}}$ by using the following abbreviations where ϕ is a state formula.

$$\begin{array}{rcl} \mathbf{A}\,\Box\phi & \equiv & \neg\mathbf{E}(\mathbf{true}\,\mathcal{U}\,\neg\phi) \\ \mathbf{O}\bigcirc\phi & \equiv & \neg\mathbf{P}\bigcirc\neg\phi \\ \mathbf{A}\bigcirc\phi & \equiv & \neg\mathbf{E}\bigcirc\neg\phi \\ \mathbf{A}\phi_1\,\mathcal{W}\,\phi_2 & \equiv & \neg(\mathbf{E}(\neg\phi_2)\,\mathcal{U}\,(\neg\phi_1 \wedge \neg\phi_2)) \end{array}$$

Whilst there is insufficient space here to present the translation of the above into CTL we have performed this translation and used a CTL theorem prover CTL-RP [9] to show that $\tau(\hat{\phi})$ is satisfiable and that $\tau(\hat{\phi} \Rightarrow \mathbf{O}\blacktriangle\diamondsuit(f_X \wedge f_Y))$ is valid. This required 0.7 seconds and 1.7 seconds of CPU time respectively.

VI. Related Work and Conclusions

RoCTL* is a useful logic for reasoning about robustness and obligation. As well as other approaches to deontic logics and robustness using temporal logics, for example [16], [17], related work includes proof methods for CTL [18], [7], [19], [8], [9], implemented provers for CTL, CTL-RP [9] and the Tableau Workbench [13] and calculi for bundled CTL* [20] and CTL* [21]. An earlier study of Pair-RoCTL is at [22].

However, whilst RoCTL* is useful in representing systems it is complex to decide; all known decision procedures for RoCTL* have non-elementary complexity and there is no elementary translation into tree-automata so we do not expect any elementary decision procedure to be found [15]. We investigated two fragments of RoCTL*, namely Pair-RoCTL and State-RoCTL aiming to identify sublogics that are simpler than full RoCTL* to decide. We show that we can translate CTL* into Pair-RoCTL so that Pair-RoCTL is at least as hard as CTL* to decide. However, we provide a linear satisfiability preserving translation from State-RoCTL into CTL. As State-RoCTL has CTL as a sublogic this shows that the complexity of satisfiability of State-RoCTL is EXPTIME-complete. It is possible to use a similar translation to demonstrate that the model checking problem for State-RoCTL also has the same order of complexity as that of CTL [12]. We provide an example of a problem that can be specified in State-RoCTL and apply a CTL prover to the resulting formulae.

REFERENCES

[1] T. French, J. M^{c}Cabe-Dansted, and M. Reynolds, "Temporal Logic of Robustness," in *Proceedings of the 6th International Symposium of the Frontiers of Combining Systems*, ser. Lecture Notes in Artificial Intelligence, B. Konev and F. Wolter, Eds., vol. 4720. Springer, 2007, pp. 193–205.

[2] E. A. Emerson and J. Y. Halpern, ""Sometimes" and "Not Never" Revisited: On Branching Versus Linear Time," in *Proceedings of the 10th ACM Symposium on Principles of Programming Languages*, 1983, pp. 127–140.

[3] J. Forrester, "Gentle murder, or the adverbial samaritan," *The Journal of Philosophy*, vol. 81, no. 4, pp. 193–7, April 1984.

[4] J. C. M^{c}Cabe-Dansted, T. French, M. Reynolds, and S. Pinchinat, "On the expressivity of RoCTL*," in *TIME*, C. Lutz and J.-F. Raskin, Eds. IEEE Computer Society, 2009, pp. 37–44.

[5] J. C. M^{c}Cabe-Dansted, "A tableau for RoBCTL*," in *JELIA*, ser. Lecture Notes in Computer Science, S. Hölldobler, C. Lutz, and H. Wansing, Eds., vol. 5293. Springer, 2008, pp. 298–310.

[6] E. A. Emerson and E. M. Clarke, "Using Branching Time Temporal Logic to Synthesize Synchronization Skeletons," *Science of Computer Programming*, vol. 2, no. 3, pp. 241–266, 1982.

[7] E. A. Emerson and J. Y. Halpern, "Decision Procedures and Expressiveness in the Temporal Logic of Branching Time," *Journal of Computer and System Sciences*, vol. 30, no. 1, pp. 1–24, Feb. 1985.

[8] P. Abate, R. Goré, and F. Widmann, "One-Pass Tableaux for Computation Tree Logic," in *Logic for Programming, Artificial Intelligence, and Reasoning*, ser. LNCS, vol. 4790. Springer, 2007, pp. 32–46.

[9] L. Zhang, U. Hustadt, and C. Dixon, "A Refined Resolution Calculus for CTL," in *Automated Deduction—CADE-22*, ser. LNAI. Springer, 2009, pp. 245–260.

[10] M. Vardi and L. Stockmeyer, "Improved upper and lower bounds for modal logics of programs," in *17th ACM Symp. on Theory of Computing, Proceedings*. ACM, 1985, pp. 240–251.

[11] E. Emerson and C. Jutla, "Complexity of Tree Automata and Modal Logics of Programs," *SIAM Journal of Computing*, vol. 29, no. 1, pp. 132–158, 2000.

[12] J. C. M^{c}Cabe-Dansted, "A temporal logic of robustness," Ph.D. dissertation, The University of Western Australia, 2010, in preparation, draft available at http://dansted.co.cc/papers/Thesis_RoCTL.pdf.

[13] P. Abate and R.Goré, "The Tableaux Workbench," in *Automated Reasoning with Analytic Tableaux and Related Methods*, ser. LNCS, vol. 2796. Springer, 2003, pp. 230–236.

[14] D. Gabbay, A. Pnueli, S. Shelah, and J. Stavi, "The Temporal Analysis of Fairness," in *Proceedings of the Seventh ACM Symposium on the Principles of Programming Languages*, Las Vegas, Nevada, January 1980, pp. 163–173.

[15] J. C. McCabe-Dansted, T. French, M. Reynolds, and S. Pinchinat, "On the expressivity of RoCTL*," in *TIME*, C. Lutz and J.-F. Raskin, Eds. IEEE Computer Society, 2009, pp. 37–44.

[16] H. Hansson and B. Jonsson, "A Logic for Reasoning about Time and Reliability," *Formal Aspects of Computing*, vol. 6, no. 5, pp. 512–535, 1994.

[17] J. Broersen, F. Dignum, V. Dignum, and J.-J.Ch.Meyer, "Designing a Deontic Logic of Deadlines," in *DEON*, ser. LNCS, A. Lomuscio and D. Nute, Eds., vol. 3065. Springer, 2004, pp. 43–56.

[18] E. A. Emerson and J. Srinivasan, "Branching Time Temporal Logic," *LNCS*, vol. 354, pp. 123–172, 1988.

[19] A. Bolotov, "Clausal Resolution for Branching-Time Temporal Logic," Ph.D. dissertation, Dept. of Computing and Mathematics, Manchester Metropolitan University, 2000.

[20] M. Reynolds, "A Tableau for Bundled CTL*," *J Logic Computation*, vol. 17, no. 1, pp. 117–132, 2007.

[21] ——, "A tableau for CTL*," in *FM 2009: Formal Methods*, vol. 5850. Springer, 2009, pp. 403–418.

[22] C. Dixon and J. M^{c}Cabe-Dansted, "Resolution for a temporal logic of robustness (extended version)," University of Liverpool, Department of Computer Science, Tech. Rep. ULCS-08-002, 2008, `www.csc.liv.ac.uk/research/techreports/`.

[23] C. Lutz and J.-F. Raskin, Eds., *TIME 2009, 16th International Symposium on Temporal Representation and Reasoning, Bressanone-Brixen, Italy, 23-25 July 2009, Proceedings*. IEEE Computer Society, 2009.