Counterexample-Guided Interval Weakening

Ben M. Andrew, Marie Farrell, Louise A. Dennis, and Michael Fisher

Department of Computer Science, University of Manchester, UK
{first.last}@manchester.ac.uk

Abstract. Systems that are deployed in unpredictable environments for long periods of time are prone to sensor and component degradation. These degradations impact the tasks that the system, including its software, can accomplish. When designing critical systems for hazardous environments, we often specify and verify that they obey temporal logic properties. With degradation, these properties may not hold perfectly throughout long-term deployments. However, *weakened* versions of these properties may still hold and be useful to provide system guarantees and assurance under degradation. For example, we may want to verify the property that an elevator will always arrive within 30 seconds of calling it, but due to motor degradation we may only be able to guarantee arrival within 60 seconds. Although weaker, this property is still useful and allows the system to maintain a reasonable level of operation. In this paper we present CEGIW, an iterative, counterexample-guided algorithm for automatically weakening the intervals of timing properties in Metric Temporal Logic such that the properties hold in the degraded system. We prove the correctness and optimality of CEGIW, and conduct an empirical evaluation to demonstrate the practicality of interval weakening using formalised requirements from a number of real-world case-studies. Using a model checker to produce counterexamples, CEGIW either weakens an interval of the property so that it holds on the counterexamples or detects whether no weakening exists.

Keywords: System degradation · Specification weakening · Formal methods · Metric temporal logic

1 Introduction

Temporal properties of systems are often specified using logics such as Metric Temporal Logic [27] (MTL), and these properties can be verified to hold using model-checking [12]. However, in the real world, system failures or degradation can invalidate these proofs by breaking their assumptions, in which case the desired properties may no longer hold. Yet, under degradation the system may still have some useful capabilities for reduced operation, and so *logically weaker* versions of these properties may hold.

Given a degraded system $\mathcal{M}$, and an ideal MTL property ϕ that doesn't hold in $\mathcal{M}$, we aim to derive ϕ', the strongest possible *weakening* of ϕ, such that $\phi \Rightarrow \phi'$. To constrain the search space, we focus on modifying the intervals of

MTL formulae. For example, we may want an elevator to always arrive at least 30 seconds after calling it, represented by

$$\Box(\texttt{callElevator} \to \Diamond_{[0,30]}\texttt{elevatorArrives}) \tag{1}$$

(where $\Box$ is the *always* operator and $\Diamond_{[0,30]}$ is the *eventually* operator bounded between zero and thirty time units). However, if the main motor breaks, a weaker backup motor may start, slowing the system down. In this case the ideal property may not hold, and we may only be able to guarantee that the elevator will arrive within 60 seconds, represented by

$$\Box(\texttt{callElevator} \to \Diamond_{[0,60]}\texttt{elevatorArrives}). \tag{2}$$

This property is logically weaker than the original, but still guarantees a useful level of functionality. We would like to be able to derive this new property automatically from the system model and the original property.

Related work. Many works consider *unrealisable* set of requirements — where conflicts mean that no satisfying implementation exists — solving the problem by weakening specifications. Some use counterstrategies to strengthen assumptions [13, 6, 29] in the LTL fragment $GR(1)$, while others use heuristic-guided genetic algorithms to mutate assumptions and guarantees towards realisability [11]. However, this is different to the problem of weakening specifications relative to an existing implementation, which we are concerned with. We use a counterexample-guided approach, which has been applied to a large variety of problems including abstraction refinement [15, 1, 25], program synthesis [4], and learning assumptions for compositional verification [17], but not yet to the problem of specification repair in the presence of an existing implementation. This has been explored using techniques from the field of program repair [21], typically heuristically-guided *generate-and-validate* approaches like mutation-based repairs [14] and dynamic invariant detection [2]. We, however, are concerned with correct-by-construction, *semantics-driven* approaches, which have only been explored in the case of propositional logic specifications [7].

Contribution. We present our Counterexample-Guided Interval Weakening (CEGIW) algorithm that, given a degraded system and a desired MTL property ϕ that does not hold on the system, produces a new optimal MTL property ϕ' that both is weakening of ϕ and holds in the degraded system. We use a counterexample-guided approach, generating counterexamples with the nuXmv model checker [12], weakening the property to hold on the counterexamples, and iteratively weakening in this way until the property holds in the system. This approach is aimed at engineers in the design phase of safety-critical systems, who are trying to understand how resilient the timing properties of their system are to various proposed degradations, and how the system's formal guarantees are thus impacted.

The paper is organised as follows: Section 2 sets up the weakening of MTL formulae within contexts, Section 3 describes CEGIW and proves its correctness and optimality, Section 4 demonstrates CEGIW on an example and considers its usefulness in real-world case-studies, and Section 5 concludes and outlines future work.

2 Weakening Within Contexts

We briefly state the syntax and semantics of Metric Temporal Logic [27] (MTL). Let $\mathcal{P}$ be a set of propositional variables. Well-formed MTL formulae are formed according to the rule:

$$\phi := p \mid \top \mid \neg\phi \mid \phi \wedge \phi \mid \phi\, \mathcal{U}_I\, \phi \mid \phi\, \mathcal{R}_I\, \phi \tag{3}$$

where $p \in \mathcal{P}$ and I is an interval of the form $[a, b]$ for $a \in \mathbb{N}$ and $b \in \mathbb{N} \cup \{\infty\}$ where $a \leq b$. Other constructs can be defined in the standard way, e.g. $\Diamond_I \phi = \top\, \mathcal{U}_I\, \phi$. While the release operator $\mathcal{R}$ can be defined in terms of the until operator $\mathcal{U}$, this would leave a negation as the outermost operator, complicating our weakening algorithm. We consider MTL formulae with a pointwise semantics over the natural numbers [5], defined according to a trace π which is an infinite sequence of states in which atomic propositions can hold, and an index of the trace $t \in \mathbb{N}$. The set of atomic propositions that hold in the tth state is denoted by $\pi(t)$. A trace π satisfies an MTL formula ϕ, denoted by $\pi \vDash \phi$, if and only if $\pi, 0 \vDash \phi$.

$$\begin{array}{lll}
\pi, t \vDash p & \text{iff} & p \in \pi(t) \\
\pi, t \vDash \neg\phi & \text{iff} & \pi, t \nvDash \phi \\
\pi, t \vDash \phi_1 \wedge \phi_2 & \text{iff} & \pi, t \vDash \phi_1 \text{ and } \pi, t \vDash \phi_2 \\
\pi, t \vDash \phi_1\, \mathcal{U}_I\, \phi_2 & \text{iff} & \exists i \in I.\, ((\pi, t+i \vDash \phi_2) \wedge \forall j \in [0, i) \cap I.\, (\pi, t+j \vDash \phi_1)) \\
\pi, t \vDash \phi_1\, \mathcal{R}_I\, \phi_2 & \text{iff} & \forall i \in I.\, (\pi, t+i \vDash \phi_1) \\
& & \vee\, \exists i \in I.\, (\pi, t+i \vDash \phi_2 \wedge \forall j \in [0, i] \cap I.\, (\pi, t+j \vDash \phi_1))
\end{array}$$

CEGIW weakens a constituent subformula of a larger formula. We show, using the notion of *contexts*, that a weakening of a subformula implies a weakening of the larger formula.

Definition 1 (Contexts). *MTL Contexts are like MTL formulae with a single hole $[-]$, and are formed according to the rule:*

$$C ::= [-] \mid C \wedge \phi \mid \phi \wedge C \mid C \vee \phi \mid \phi \vee C \mid C\, \mathcal{U}_I\, \phi \mid \phi\, \mathcal{U}_I\, C \mid C\, \mathcal{R}_I\, \phi \mid \phi\, \mathcal{R}_I\, C \tag{4}$$

where ϕ is an MTL formula and I is an interval. Note that our definition of contexts does not allow negations on the path to the hole $[-]$, but allows them in adjacent MTL subformulae.

We define the notion of *context substitution*, where an MTL formula ψ is substituted into the hole of a context C to produce an MTL formula $C[\psi]$.

$$\begin{array}{rlrl}
[-][\psi] &= \psi & & \\
(C \wedge \phi)[\psi] &= C[\psi] \wedge \phi & (C\, \mathcal{U}_I\, \phi)[\psi] &= C[\psi]\, \mathcal{U}_I\, \phi \\
(\phi \wedge C)[\psi] &= \phi \wedge C[\psi] & (\phi\, \mathcal{U}_I\, C)[\psi] &= \phi\, \mathcal{U}_I\, C[\psi] \\
(C \vee \phi)[\psi] &= C[\psi] \vee \phi & (C\, \mathcal{R}_I\, \phi)[\psi] &= C[\psi]\, \mathcal{R}_I\, \phi \\
(\phi \vee C)[\psi] &= \phi \vee C[\psi] & (\phi\, \mathcal{R}_I\, C)[\psi] &= \phi\, \mathcal{R}_I\, C[\psi]
\end{array} \tag{5}$$

Note that, by pushing negations inwards, an MTL formula ϕ can always be transformed into a context C and subformula ψ where ϕ is logically equivalent to $C[\psi]$.

Definition 2 (Weakening and strengthening of MTL formulae). *Let ϕ and ϕ' be MTL formulae. ϕ' is a weakening of ϕ, denoted*

$$\phi \sqsubseteq \phi' \tag{6}$$

if and only if, for all traces π and time-points t, if $\pi, t \vDash \phi$, then $\pi, t \vDash \phi'$. In this case, symmetrically, ϕ is a strengthening of ϕ'. Note that an MTL formula ϕ is always both a strengthening and a weakening of itself, i.e. $\phi \sqsubseteq \phi$.

Theorem 3 (Weakening of contexts). *Let C be a context and ψ and ψ' be MTL formulae. If $\psi \sqsubseteq \psi'$, then $C[\psi] \sqsubseteq C[\psi']$.*

Proof. We do an induction proof over the grammar of contexts using the induction hypothesis $P(C)$, that $C[\psi] \sqsubseteq C[\psi']$. The non-temporal inductive cases are omitted for brevity.

Base case $[-]$: We assume that $\psi \sqsubseteq \psi'$, and by the definition of context substitution we have that $[-][\psi] \sqsubseteq [-][\psi']$ and thus $P([-])$.

Inductive case $C\, \mathcal{U}_I\, \phi$: Assuming $P(C)$, we take an arbitrary trace π and time-point t, assume $\pi, t \vDash C[\psi]\, \mathcal{U}_I\, \phi$, and want to prove $\pi, t \vDash C[\psi']\, \mathcal{U}_I\, \phi$. We know that there exists an $i \in I$ such that $\pi, t+i \vDash \phi$, and that for all $j \in [0, i) \cap I$ we have $\pi, t+j \vDash C[\psi]$. Taking arbitrary i and j, by the induction hypothesis we have that $\pi, t+j \vDash C[\psi']$, and so by the semantics $\pi, t \vDash C[\psi']\, \mathcal{U}_I\, \phi$. Thus, we have $P(C\, \mathcal{U}_I\, \phi)$.

Inductive case $\phi\, \mathcal{U}_I\, C$: Similar to the above case.

Inductive case $C\, \mathcal{R}_I\, \phi$: Assuming $P(C)$, we take an arbitrary trace π and time-point t, assume $\pi, t \vDash C[\psi]\, \mathcal{R}_I\, \phi$, and want to prove $\pi, t \vDash C[\psi']\, \mathcal{R}_I\, \phi$. By the semantics of $\mathcal{R}$ there are two cases:

1. For all $i \in I$ we have $\pi, t+i \vDash \phi$, thus we have $\pi, t+i \vDash C[\psi']$, and so we have $\pi, t \vDash C[\psi']\, \mathcal{R}_I\, \phi$.
2. There exists an $i \in I$ such that $\pi, t+i \vDash C[\psi]$, and that for all $j \in [0, i] \cap I$ we have $\pi, t+j \vDash \phi$. Taking arbitrary i and j, by the assumptions we have that $\pi, t+i \vDash C[\psi']$ and $\pi, t+j \vDash \phi$, and then by the semantics we have $\pi, t \vDash C[\psi']\, \mathcal{R}_I\, \phi$.

Thus, in both cases we have $P(C\, \mathcal{R}_I\, \phi)$.

Inductive case $\phi\, \mathcal{R}_I\, C$: Similar to the above case. □

We show that, depending on which temporal operator is used, by expanding or contracting its interval we can weaken or strengthen the surrounding formula.

Definition 4 (Right-modifications of intervals). *Let $I = [a, b]$ be an interval. For any $i \in \mathbb{N}$, a right-modification of I is either a right-extension $[a, b+i]$, or, provided $i \leq b-a$, a right-contraction $[a, b-i]$. A right-modification is* strict *if $i > 0$.*

Lemma 5 (Weakening of $\mathcal{U}$ interval). *Let ϕ and ψ be MTL formulae, and I and I' be intervals, where I' is a right-extension of I. Then, $\phi\,\mathcal{U}_I\,\psi \sqsubseteq \phi\,\mathcal{U}_{I'}\,\psi$.*

Proof. We assume that I' is a right-extension of I, and so, taking an arbitrary trace π and time-point t, we assume $\pi, t \vDash \phi\ \mathcal{U}_I\,\psi$ and want to prove $\pi, t \vDash \phi\,\mathcal{U}_{I'}\,\psi$. We know that there exists an $i \in I$ such that $\pi, t+i \vDash \psi$, and that for all $j \in [0,i) \cap I$ we have $\pi, t+j \vDash \phi$. Taking arbitrary i and j, we have that $i, j \in I'$, and so $\pi, t \vDash \phi\,\mathcal{U}_{I'}\,\psi$. Thus, $\phi\,\mathcal{U}_I\,\psi \sqsubseteq \phi\,\mathcal{U}_{I'}\,\psi$. □

Lemma 6 (Weakening of $\mathcal{R}$ interval). *Let ϕ and ψ be MTL formulae, and I and I' be intervals, where I' is a right-contraction of I. Then, $\phi\,\mathcal{R}_I\,\psi \sqsubseteq \phi\,\mathcal{R}_{I'}\,\psi$.*

Proof. We assume that I' is a right-contraction of I, and so, taking an arbitrary trace π and time-point t, we assume $\pi, t \vDash \phi\ \mathcal{R}_I\,\psi$ and want to prove $\pi, t \vDash \phi\,\mathcal{R}_{I'}\,\psi$. By the semantics of $\mathcal{R}$ there are two cases:

1. For all $t' \in I$ we have $\pi, t+t' \vDash \psi$. Then, as $I' \subseteq I$, we know that for all $t'' \in I'$ we have $\pi, t+t' \vDash \psi$, and so $\pi, t \vDash \phi\,\mathcal{R}_{I'}\,\psi$.
2. There exists a $t' \in I$ such that $\pi, t+t' \vDash \phi$ and for all $t'' \in I \cap [0,t']$, we have $\pi, t+t'' \vDash \psi$. As I' is a right-contraction of I, there are two further cases:
   (a) If $t' \in I'$, then we still have that $\pi, t+t' \vDash \phi$ and for all $t'' \in I' \cap [0,t']$, we have $\pi, t+t'' \vDash \psi$, and so $\pi, t \vDash \phi\,\mathcal{R}_{I'}\,\psi$.
   (b) If $t' \notin I'$, then $I' \cap [0,t'] = I'$ and so we know that for all $t'' \in I'$, we have $\pi, t+t'' \vDash \psi$, and so $\pi, t \vDash \phi\,\mathcal{R}_{I'}\,\psi$.

Thus in all cases we have that $\phi\,\mathcal{R}_I\,\psi \sqsubseteq \phi\,\mathcal{R}_{I'}\,\psi$. □

Often in CEGIW, recursive calls will generate a set of intervals from which either the strongest or weakest must be chosen. We show that there is a total order of implication over the set of right-modifications of an interval, which allows us to make that choice.

Lemma 7 (Extension-weakening order of right-modifications). *The set of right-modifications of an interval I, denoted $\mathcal{R}(I)$, has a total order $\supseteq$, where for all MTL contexts C, MTL formulae ϕ and ϕ', and all $I', I'' \in \mathcal{R}(I)$, if $I'' \supseteq I'$ then we have $C[\phi\,\mathcal{U}_{I'}\,\phi'] \sqsubseteq C[\phi\,\mathcal{U}_{I''}\,\phi']$.*

Proof. For any pair of intervals in $\mathcal{R}(I)$ we can order the resulting subformulae by applying Lemma 5 to get $\phi\,\mathcal{U}_{I'}\,\phi' \sqsubseteq \phi\,\mathcal{U}_{I''}\,\phi'$ (or the reverse), and then order the full formulae with their contexts by applying Theorem 3 to get $C[\phi\,\mathcal{U}_{I'}\,\phi'] \sqsubseteq C[\phi\,\mathcal{U}_{I''}\,\phi']$ (or the reverse). □

Lemma 8 (Contraction-weakening order of right-modifications). *The set of right-modifications of an interval I, denoted $\mathcal{R}(I)$, has a total order $\subseteq$, where for all MTL contexts C, MTL formulae ϕ and ϕ', and all $I', I'' \in \mathcal{R}(I)$, if $I'' \subseteq I'$ then we have $C[\phi\,\mathcal{R}_{I'}\,\phi'] \sqsubseteq C[\phi\,\mathcal{R}_{I''}\,\phi']$.*

Proof. Similar to the proof of Lemma 7, but uses Lemma 6 to order $\phi\ \mathcal{R}_{I'}\,\phi'$ and $\phi\,\mathcal{R}_{I''}\,\phi'$. □

3 Algorithm for Interval Weakening

CEGIW is split into two levels. First, there is a function *weaken* that, given an MTL formula ϕ, an interval $I = [a, b]$ in ϕ to weaken, and a counterexample trace π, weakens I such that the new formula ϕ' holds on π (Section 3.1). However, this does not guarantee that ϕ' holds on the model itself, and so we need to repeat the process, finding a new counterexample trace for ϕ' and weakening the interval again. The second part of CEGIW is this iterative process that finds counterexample traces by model checking (Section 3.2).

3.1 Weakening on a Counterexample

Model checkers generally produce a specific type of infinite counterexample trace, called a *lasso trace.*

Definition 9 (Lasso traces). *A trace π is* lasso *if it can be separated into a finite prefix π_{pre} and an infinitely repeating finite suffix π_{suf}, forming*

$$\pi = \pi_{\text{pre}}(\pi_{\text{suf}})^{\omega}. \tag{7}$$

This restricts us to a subset of infinite traces that can be finitely represented. The finite length of a lasso trace is then defined as $|\pi| = |\pi_{\text{pre}}| + |\pi_{\text{suf}}|$.

We show that we can prove properties of an entire infinite lasso trace using only a finite *covering interval.* Without this, we may need to iterate over the entire infinite trace, impacting completeness.

Definition 10 (Covering intervals). *The suffix-covering interval of π, defined with respect to an interval $[a, b]$, is*

$$\text{cov}_{\pi}([a, b]) = [a, \min(b, \text{end}_{\pi}(a))] \tag{8}$$

where

$$\text{end}_{\pi}(a) = \begin{cases} |\pi| & \textit{if } a < |\pi_{\text{pre}}| \\ a + |\pi_{\text{suf}}| - 1 & \textit{otherwise.} \end{cases} \tag{9}$$

Lemma 11 (Lasso trace coverage). *Let ϕ be an MTL formula, π be a lasso trace, and $a \in \mathbb{N}$. If for all $t \in [a, \text{end}_{\pi}(a)]$ we have $\pi, t \vDash \phi$, then for all $t' \in \mathbb{N}$ with $t' \geq a$ we have $\pi, t' \vDash \phi$.*

Proof. We assume that for all $t \in [a, \text{end}_{\pi}(a)]$ we have $\pi, t \vDash \phi$, and, taking an arbitrary $t' \in \mathbb{N}$ with $t' \geq a$ want to prove that $\pi, t' \vDash \phi$. There are two cases. Firstly, if $t' < |\pi|$ then we know that this is within the $[a, \text{end}_{\pi}(a)]$ range and so we have $\pi, t' \vDash \phi$. Otherwise, if $t' \geq |\pi|$, we split π into its prefix π_{pre} and infinitely repeating suffix π_{suf}, and want to prove that $\pi_{\text{pre}}(\pi_{\text{suf}})^{\omega}, t' \vDash \phi$. As $t' \geq |\pi|$, we can split it into $t' = |\pi_{\text{pre}}| + n \cdot |\pi_{\text{suf}}| + m$ for some $n, m \in \mathbb{N}$ with

Algorithm 1: Weakening within a context C

```
function Weaken(C, $\psi \triangle_{I_{\mathrm{orig}}} \psi'$, $\pi$)
    return $WeakenRec(C, 0)$
function WeakenRec(C, t)
    if $C = [-]$ then
        if $\triangle = \mathcal{U}$ then
            return $Weaken\mathcal{U}Direct(\psi, \psi', t)$
        else // $\triangle = \mathcal{R}$
            return $Weaken\mathcal{R}Direct(\psi, \psi', t)$
    ...
    else if $C = C_l \,\mathcal{U}_I\, \phi_r$ then
        return $Weaken\mathcal{U}Left(C_l, \phi_r, I, t)$
    else if $C = \phi_l \,\mathcal{U}_I\, C_r$ then
        return $Weaken\mathcal{U}Right(\phi_l, C_r, I, t)$
    else if $C = C_l \,\mathcal{R}_I\, \phi_r$ then
        return $Weaken\mathcal{R}Left(C_l, \phi_r, I, t)$
    else if $C = \phi_l \,\mathcal{R}_I\, C_r$ then
        return $Weaken\mathcal{R}Right(\phi_l, C_r, I, t)$
```

$n \geq 1$ and $m < |\pi_{\mathrm{suf}}|$.

$$\begin{aligned} & \pi_{\mathrm{pre}}(\pi_{\mathrm{suf}})^{\omega}, |\pi_{\mathrm{pre}}| + n \cdot |\pi_{\mathrm{suf}}| + m \vDash \phi \\ \Longrightarrow\ & (\pi_{\mathrm{suf}})^{\omega}, n \cdot |\pi_{\mathrm{suf}}| + m \vDash \phi \\ \Longrightarrow\ & (\pi_{\mathrm{suf}})^{\omega}, m \vDash \phi \\ \Longrightarrow\ & \pi_{\mathrm{pre}}(\pi_{\mathrm{suf}})^{\omega}, |\pi_{\mathrm{pre}}| + m \vDash \phi \end{aligned} \tag{10}$$

We know that $|\pi_{\mathrm{pre}}| + m$ is in the $[a, \mathrm{end}_{\pi}(a)]$ interval, so we have $\pi, t' \vDash \phi$. □

We also define the *optimality* of weakenings, used to prove that CEGIW will not produce an interval weakening that is any stronger than it needs to be.

Definition 12 (Optimality of right-extensions and -contractions). *An interval I' is an optimal right-extension (resp. -contraction) of an interval I with respect to a context C, MTL formulae ψ and ψ', a temporal operator $\triangle \in \{\mathcal{U}, \mathcal{R}\}$, trace π, and time-step t, if*

$$\pi, t \vDash C[\psi \triangle_{I'} \psi'] \tag{11}$$

and either (a) $I = I'$, or (b) there exists no strict right-contraction (resp. -extension) I'' of I' such that $\pi, t \vDash C[\psi \triangle_{I''} \psi']$.

The proof of correctness and optimality follows the inductive structure of CEGIW, with base cases for directly weakening the intervals of $\mathcal{U}_I$ and $\mathcal{R}_I$ (Lemmas 13 and 14), and inductive cases for weakening within both operators on either side (Lemmas 15 to 18). Note that the temporal subformula $\psi \triangle_{I_{\mathrm{orig}}} \psi'$ with original interval I_{orig} is globally visible in all of the recursive function calls.

Algorithm 2: Directly weakening interval of $\mathcal{U}$

```
1 function $WeakenUDirect(\psi_l,\ \psi_r,\ [a,b],\ t)$
2     for $i \leftarrow a$ to $\mathrm{end}_\pi(a)$ do
3         if $\pi, t+i \vDash \psi_r$ then
4             return $[a, \max(b, i)]$
5         if $\pi, t+i \nvDash \psi_l$ then
6             break
7     return $None$
```

Lemma 13 ($\mathcal{U}$ base case). *Let I be an interval, ψ_l and ψ_r MTL formulae, and π a lasso trace. Then, for all timepoints $t \in \mathbb{N}$ with*

$$I' = WeakenUDirect(\psi_l, \psi_r, I, t), \tag{12}$$

either I' is an optimal right-extension of I such that $\pi, t \vDash \psi_l \, \mathcal{U}_{I'} \, \psi_r$, or $I' = None$, in which case there exists no such interval.

Proof. We take an arbitrary t. Our proof for Algorithm 2 uses the loop invariant that $\forall j \in [a, \mathrm{end}_\pi(a)]$ with $j < i$ (where $I = [a, b]$), we have that $\pi, t+j \nvDash \psi_r$ and $\pi, t+j \vDash \psi_l$. On first entry to the loop there is no such j, so this is trivially true. On reaching the end of the loop body, we know that $\pi, t+i \nvDash \psi_r$ and $\pi, t+i \vDash \psi_l$, and so in combination with the loop invariant we know that $\forall j \in I$ where $j \leq i$, we have $\pi, t+j \nvDash \psi_r$ and $\pi, t+j \vDash \psi_l$. Thus the loop invariant is preserved. Suppose at the start of iteration i the loop invariant holds. If $\pi, t+j \vDash \psi_r$ on Line 3 then we return $I' = [a, \max(b, i)]$. This is an optimal right-extension of of I and we have that $\pi, t \vDash \psi_l \, \mathcal{U}_{I'} \, \psi_r$.

If $None$ is returned, then either we broke out of the loop early because for some $i \in [a, \mathrm{end}_\pi(a)]$ we have $\pi, t+i \nvDash \psi_l$ at Line 5, or we ran the loop to completion. In the first case, we know that $\pi, t+i \nvDash \psi_r$ as this is checked before at Line 3, and so combining with the loop invariant we know that ψ_r never held up until ψ_l stopped holding, and so there is no right-extension I' for which $\pi, t \vDash \psi_l \, \mathcal{U}_{I'} \, \psi_r$.

In the second case, by the loop invariant we have that for all $i \in [a, \mathrm{end}_\pi(a)]$ we have $\pi, t+i \vDash \psi_l$ and $\pi, t+i \nvDash \psi_r$. By Lemma 11 we then have the same for all $i \in \mathbb{N}$ with $i \geq a$, and so there exists no right-extension I' of I that satisfies $\pi, t \vDash \psi_l \, \mathcal{U}_{I'} \, \psi_r$. □

Lemma 14 ($\mathcal{R}$ base case). *Let I be an interval, ψ_l and ψ_r MTL formulae, and π a lasso trace. Then, for all timepoints $t \in \mathbb{N}$ with*

$$I' = WeakenRDirect(\psi_l, \psi_r, I, t), \tag{13}$$

either I' is an optimal right-contraction of I such that $\pi, t \vDash \psi_l \, \mathcal{R}_{I'} \, \psi_r$, or $I' = None$, in which case there exists no such interval.

Proof. Proof in Appendix A. □

Algorithm 3: Weakening within $\mathcal{U}$ on the left

```
1  function WeakenULeft($C$, $\phi$, $[a,b]$, $t$)
2      $b_{\text{fin}} \leftarrow \min(b, \text{end}_\pi(a))$
3      $intervals \leftarrow [\,]$
4      for $i \leftarrow a$ to $b_{\text{fin}}$ do
5          if $\pi, t+i \vDash \phi$ then
6              if $i = a$ then
7                  return $I_{\text{orig}}$
8              return interval in $intervals$ with maximal absolute difference to
                 $I_{\text{orig}}$
9          $I \leftarrow WeakenRec(C, t+i)$
10         if $I = None$ then
11             return $None$
12         append $I$ to $intervals$
13     return $None$
```

We prove the inductive cases with an MTL context C, an interval I, MTL formulae ψ and ψ', a temporal operator $\triangle \in \{\mathcal{U}, \mathcal{R}\}$, and a lasso trace π. We use the induction hypothesis $P(C)$, that for all timepoints $t \in \mathbb{N}$ with $I' = WeakenRec(C, \psi \triangle_I \psi', t)$, if I' is an interval then $\pi, t \vDash C[\psi \triangle_I \psi']$, and

1. If $\triangle = \mathcal{U}$, then I' is an optimal right-extension of I;
2. If $\triangle = \mathcal{R}$, then I' is an optimal right-contraction of I.

If $I' = None$, then there exists no such interval in each case.

Lemma 15 ($\mathcal{U}$-left inductive case). *Let C be an MTL context, I and J intervals, ϕ, ψ, and ψ' MTL formulae, $\triangle \in \{\mathcal{U}, \mathcal{R}\}$ a temporal operator, and π a lasso trace. If $P(C)$ holds, then so does $P(C\, \mathcal{U}_J\, \phi)$.*

Proof. For Algorithm 3 we assume the inductive hypothesis $P(C)$ and want to prove $P(C\, \mathcal{U}_J\, \phi)$. We take an arbitrary t and distinguish two cases, according to whether $\triangle$ is $\mathcal{U}$ or $\mathcal{R}$. In either case, by the induction hypothesis each recursive call evaluates to either $None$ or an optimal interval I' related to I by the corresponding relation (right-extension or -contraction respectively) such that $\pi, t+t' \vDash C[\psi \triangle_{I'} \psi']$.

We use the loop invariant that, for all $j \in \text{cov}_\pi(J)$ with $j < i$, we have that $\pi, t+j \nvDash \phi$ and that $I' = WeakenRec(C, \psi \triangle_I \psi', t+j)$ is an interval such that $\pi, t+j \vDash C[\psi \triangle_{I'} \psi']$. On first entry to the loop there is no such j, so this is trivially true. On reaching the end of the loop body, we know that $WeakenRec(C, \psi \triangle_I \psi', t+i) \neq None$ from Line 10, and so by the induction hypothesis the recursive call must have produced a suitable interval I'. As we also know that $\pi, t+i \nvDash \phi$ from Line 5, the loop invariant is thus preserved for $j \leq i$. Suppose at the start of iteration i the loop invariant holds. If $WeakenRec(C, \psi \triangle_I \psi', t+i) = None$ at Line 10 then by the induction hypothesis we know that there is no suitable interval I'' for which $\pi, t+i \vDash C[\psi \triangle_{I''} \psi']$, and by the loop invariant that there is no $j < i$ for which $\pi, t + j \vDash \phi$. Thus, there is no suitable interval I'' for which $\pi, t \vDash (C\,\mathcal{U}_J\,\phi)[\psi \triangle_{I''} \psi']$. If $\pi, t+j \vDash \phi$ at Line 5 then we split on whether it is our first iteration or not. If $i = a$ (where $J = [a, b]$) then we know that $\pi, t + a \vDash \phi$, and so any interval will work. We simply return the original interval I_{orig}.

Otherwise, by the loop invariant we know that for all $j \in \text{cov}_\pi(J)$ with $j < i$ — of which there must be at least one as $i > a$ — we have an interval I' such that $\pi, t + j \vDash C[\psi \triangle_{I'} \psi']$. Applying Lemma 7 if $\triangle = \mathcal{U}$, or Lemma 8 if $\triangle = \mathcal{R}$, we obtain a maximum interval I'' such that for all $j \in \text{cov}_\pi(J)$ with $j < i$ we have $\pi, t + j \vDash C[\psi \triangle_{I''} \psi']$. If $\text{cov}_\pi(J) = J$ then we have

$$\pi, t \vDash (C\,\mathcal{U}_J\,\phi)[\psi \triangle_{I''} \psi']. \tag{14}$$

Otherwise, if $\text{cov}_\pi(J) = [a, \text{end}_\pi(a)]$, then by Lemma 11 for all $k \in \mathbb{N}$ with $k \geq a$ we have $\pi, t + k \vDash C[\psi \triangle_{I''} \psi']$, and so the above holds here too. □

Lemma 16 ($\mathcal{U}$-right inductive case). *Let C be an MTL context, I and J intervals, ϕ, ψ, and ψ' MTL formulae, $\triangle \in \{\mathcal{U}, \mathcal{R}\}$ a temporal operator, and π a lasso trace. If $P(C)$ holds, then so does $P(\phi\,\mathcal{U}_J\,C)$.*

Lemma 17 ($\mathcal{R}$-left inductive case). *Let C be an MTL context, I and J intervals, ϕ, ψ, and ψ' MTL formulae, $\triangle \in \{\mathcal{U}, \mathcal{R}\}$ a temporal operator, and π a lasso trace. If $P(C)$ holds, then so does $P(C\,\mathcal{R}_J\,\phi)$.*

Lemma 18 ($\mathcal{R}$-right inductive case). *Let C be an MTL context, I and J intervals, ϕ, ψ, and ψ' MTL formulae, $\triangle \in \{\mathcal{U}, \mathcal{R}\}$ a temporal operator, and π a lasso trace. If $P(C)$ holds, then so does $P(\phi\,\mathcal{R}_J\,C)$.*

Proofs for Lemmas 16 to 18 are located in Appendix A. We use these supporting lemmas to prove the correctness and optimality of CEGIW.

Theorem 19 (Correctness for weakening). *Let C be an MTL context, I and J intervals, ϕ, ψ, and ψ' MTL formulae, $\triangle \in \{\mathcal{U}, \mathcal{R}\}$ a temporal operator, π a lasso trace, and $t \in \mathbb{N}$ be a timepoint. Let $I' = WeakenRec(C, \psi \triangle_I \psi', t)$. If I' is an interval then $\pi \vDash C[\psi \triangle_{I'} \psi']$, and*

1. *If $\triangle = \mathcal{U}$, then I' is an optimal right-extension of I;*
2. *If $\triangle = \mathcal{R}$, then I' is an optimal right-contraction of I.*

If $I' = None$, then there exists no such interval in each case.

Proof. We use the same induction hypothesis $P(C)$ defined for the preceding inductive lemmas. The non-temporal inductive cases are omitted for brevity.

Base case $[-]$: By Lemma 13 if $\triangle = \mathcal{U}$, and Lemma 14 if $\triangle = \mathcal{R}$, we have $P([-])$.
Inductive case $C\,\mathcal{U}_J\,\phi$: Assuming $P(C)$, by Lemma 15 we have $P(C\,\mathcal{U}_J\,\phi)$.
Inductive case $\phi\,\mathcal{U}_J\,C$: Assuming $P(C)$, by Lemma 16 we have $P(\phi\,\mathcal{U}_J\,C)$.
Inductive case $C\,\mathcal{R}_J\,\phi$: Assuming $P(C)$, by Lemma 17 we have $P(C\,\mathcal{R}_J\,\phi)$.
Inductive case $\phi\,\mathcal{R}_J\,C$: Assuming $P(C)$, by Lemma 18 we have $P(\phi\,\mathcal{R}_J\,C)$.
□

The time complexity of Algorithm 1 is $O(|\pi|^{\texttt{td}(\phi)})$. where π is the counterexample trace and $\texttt{td}(\phi)$ is the *temporal depth* of the MTL formula ϕ, i.e. the maximum number of nested temporal operators along any path in the syntax tree.

3.2 Iterative Weakening

Assume that we have an MTL formula ϕ that has a temporal subformula $\psi \triangle_I \psi'$ with an interval I that we want to weaken. ϕ can be split into $\psi \triangle_I \psi'$ and the surrounding MTL context C, such that ϕ is logically equivalent to $C[\psi \triangle_I \psi']$. Assume we also have a transition system $\mathcal{M}$. Using a model checker, we check whether ϕ holds on $\mathcal{M}$. If it holds then we are done, but if not, we will receive a counterexample trace π through $\mathcal{M}$ for which $\pi \nvDash C[\psi \triangle_I \psi']$. We can then weaken on this counterexample with

$$I' = Weaken(C, \psi \triangle_I \psi', \pi). \tag{15}$$

By Theorem 19, if $I' = None$, then there exists no weakening I'' of I such that $\pi \vDash C[\psi \triangle_{I''} \psi']$, and so the same holds for the model $\mathcal{M}$. Otherwise, I' is an interval such that $\pi \vDash C[\psi \triangle_{I'} \psi']$. However, $C[\psi \triangle_{I'} \psi']$ does not necessarily hold on $\mathcal{M}$, and so we model check again, creating an iterative loop that ends when we either produce an interval I'' that is a weakening of I such that $C[\psi \triangle_{I''} \psi']$ holds on $\mathcal{M}$, or show that no such weakening exists.

4 Evaluation

We evaluate how *effective* interval weakening is in understanding the temporal behaviour of specifications, and how applicable it is to real-world requirements. To this end, we investigate the following research questions:

RQ1: To what extent is CEGIW useful during the system design phase? (Section 4.1)

RQ2: To what extent can CEGIW be used in real-world requirements? (Section 4.2)

Choosing a model checker. There are no industrial-strength model checkers for MTL with pointwise semantics [10, 3], yet many efficient tools exist for linear temporal logic [12, 24] (LTL). Thus, we translate MTL formulae into LTL using the *next* (X) operator [8, Remark 5.15] and use an LTL model checker. During preliminary investigation, it was found that symbolic LTL model checkers such as NUXMV [12] and SPIN [24] typically generate minimal counterexample traces. Weakening intervals with these usually only increments or decrements the bound rather than modifying it by a larger amount, which increases the number of calls made to the model checker dramatically. Our implementation[1] uses NUXMV in bounded model checking (BMC) mode, producing multiple counterexamples for a specific bound length, finding the optimal interval for each of them and returning the weakest, making it more likely that we make fewer calls to the model checker. For our implementation we require the user to choose the bound, but theoretical completeness can be preserved as completeness thresholds do exist for BMC [16].

[1] https://github.com/benmandrew/CEGIW

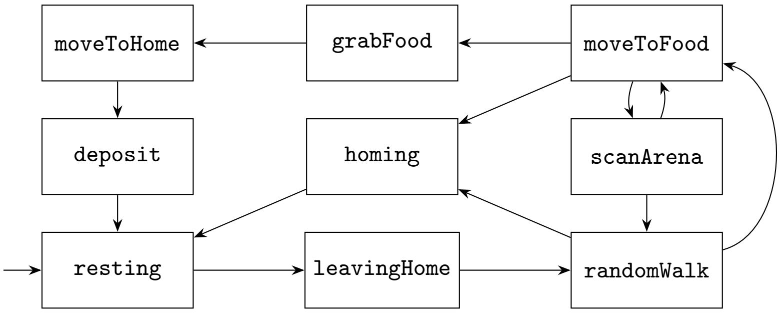


Fig. 1: Abstract state transition system for the robot's foraging behaviour. The robot begins in a resting state, then searches for, collects, and deposits food.

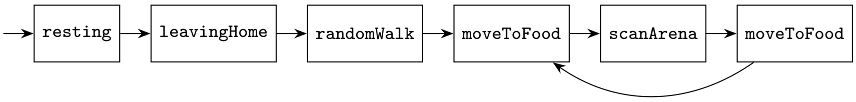


Fig. 2: Infinite lasso counterexample trace for the property in Eq. (16).

4.1 Demonstration of CEGIW (RQ1)

To address **RQ1**, we use an example based on a model of a foraging robot swarm [28]. Robots are located in an arena and do a random walk to find food, which they then carry back to their home. Here they recharge and then repeat their foraging task. We model a single robot with a state machine, depicted abstractly in Fig. 1. In the concrete transition system $\mathcal{M}$ the robot can remain in a given state for a configurable amount of time before it is forced to move to a next state. The transition system is specified concretely using SMV, the language of the nuXmv model checker [12]. We would like to prove that, after leaving the `resting` state, the robot will return to `resting` in at most 3 time units, represented by

$$\mathcal{M} \vDash \Box(\texttt{resting} \rightarrow \Diamond_{[1,3]}\texttt{resting}) \tag{16}$$

and translated from MTL to LTL as

$$\mathcal{M} \vDash \Box(\texttt{resting} \rightarrow X(\texttt{resting} \vee X(\texttt{resting} \vee X(\texttt{resting})))). \tag{17}$$

If this does not hold in the transition system, we would like to weaken the interval to produce a new, weaker property that does hold. Using CEGIW, we find in the first iteration that no suitable weakening of the interval exists based on the counterexample in Fig. 2, which shows an infinite loop between the `scanArena` and `moveToFood` states. This suggests a mistake in the modelling of the system, as in the real world the robot's battery would run out of charge. We amend the design by including in the requirements the notion of a battery that decreases as transitions are taken. While we are in `randomWalk`, `scanArena`, or `moveToFood` — in other words, searching for food — we monitor the battery level, and if it

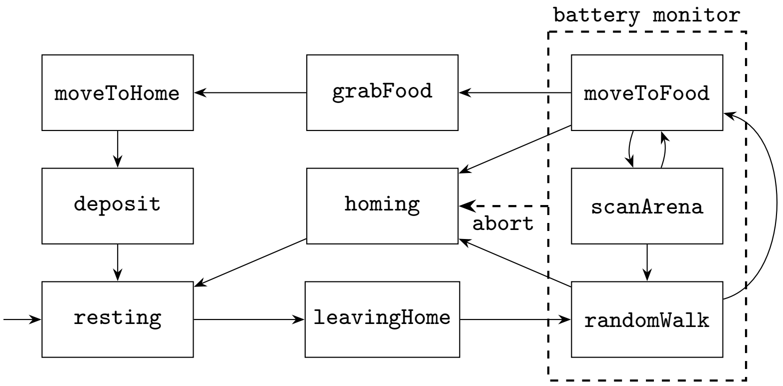
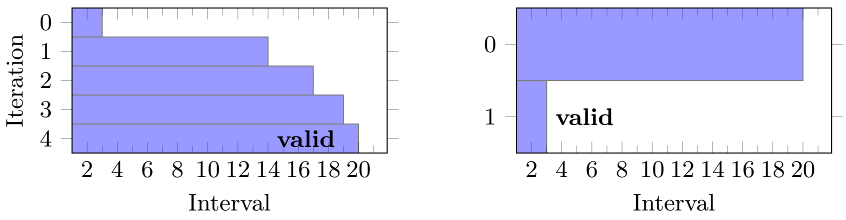


Fig. 3: Abstract state transition system for the robot's modified foraging behaviour. The battery monitor is represented by the dashed section on the right.



(a) Interval extension for Eq. (16).

(b) Interval contraction for Eq. (19).

Fig. 4: Iterative interval weakening to generate optimal, valid intervals.

decreases below a certain threshold we abort and return home to recharge. The modified state transition system is depicted in Fig. 3. We check our desired property (Eq. (16)) against our amended model, and can see in Fig. 4a that in four iterations of CEGIW we extended the interval, and ended with the optimal interval which was then verified to hold in the system. So, the optimal property that holds in our amended system is

$$\mathcal{M} \models \Box(\texttt{resting} \rightarrow \Diamond_{[1,20]}\texttt{resting}). \tag{18}$$

Another property we are interested in is not the maximum time that the robot can spend away from home, but the *minimum*. We wish the robot to spend at least 20 time units away from home, formalised as

$$\mathcal{M} \models \Box((\texttt{resting} \land \Diamond_{[1,1]}\neg\texttt{resting}) \rightarrow \Box_{[1,20]}\neg\texttt{resting}). \tag{19}$$

Again, we check this against our modified model and can see in Fig. 4b that it took only one iteration to contract the interval, reaching the optimal interval which was then verified to hold in the system as

$$\mathcal{M} \models \Box((\texttt{resting} \land \Diamond_{[1,1]}\neg\texttt{resting}) \rightarrow \Box_{[1,3]}\neg\texttt{resting}). \tag{20}$$

Table 1: Interval-weakenable requirements in FRET case studies.

Case study	**Total requirements**	**Weakenable requirements**
Mechanical lung ventilator [18]	121	63
Autonomous drone [32]	62	19
Lift-plus-cruise aircraft [31]	49	30
Aircraft engine controller [19]	42	0
Inspection rover [9]	15	2
Grasping for debris removal [20]	20	0
Robotic patterns [33]	28	9
LMCP challenges [30]	73	7
Total	**410**	**130**

```
upon ControlLoopStart System shall
within 12 milliseconds satisfy
ControlLoopFinish
```

(a) Autonomous drone requirement REQ018 describing the maximum time the control loop can take to complete.

```
if powerFailure System shall for
120 minutes satisfy !off
```

(b) Mechanical lung ventilator requirement FUN37 describing how long the system must stay on after power failure.

Fig. 5: Example FRETISH requirements from the case studies. Both are taken from systems that are fully implemented and operational in real-world settings.

By using CEGIW, we first identified that the original specification had a design flaw that allowed unwanted infinite loops. After modifying the specification, we then deduced both the maximum time that the robot can stay away from home, as well as the minimum time.

4.2 Feasibility of Approach in Real-World Case Studies (RQ2)

We analyse the requirement sets from a number of case studies, which are formalised using the Formal Requirements Elicitation Tool (FRET) [22]. Requirements are written in FRETISH, a structured natural language that can be translated to MTL [23]. FRETISH requirements can have a `timing` field, on which we can use interval weakening to weaken the requirement itself. The number of requirements that can be weakened using interval weakening per case study is shown in Table 1. As an example, the requirement in Fig. 5a from the autonomous drone case study [32] uses the `within 12 milliseconds` timing which specifies that if the condition holds in one state, then the consequent must hold within the next twelve states (assuming that state transitions correspond to a millisecond of time passing). The MTL translation of this timing corresponds to the MTL temporal operator $\Diamond_{[0,12]}$, and so the requirement corresponds to

$$\Box(\texttt{ControlLoopStart} \rightarrow \Diamond_{[0,12]}(\texttt{ControlLoopFinish})). \tag{21}$$

Under system degradation, for example if the onboard communications network is degraded so that commands take longer to reach control surfaces, we may not be able to guarantee this and so would have to weaken the property by extending the interval, giving more time for the system to run its control loop, with an example weakening in

$$\Box(\texttt{ControlLoopStart} \rightarrow \Diamond_{[0,24]}(\texttt{ControlLoopFinish})). \tag{22}$$

An example requirement from the mechanical lung ventilator case study [18] is shown in Fig. 5b, and as the FRETISH timing `for 120 minutes` corresponds to the MTL temporal operator $\Box_{[1,120]}$, the corresponding MTL property is

$$\Box(\texttt{powerFailure} \rightarrow \Box_{[1,120]}(\neg\texttt{off})). \tag{23}$$

This is a regulatory requirement [26] and so, if it does not hold in the degraded system, it is critical to know by exactly how much it is violated. We may only be able to guarantee that the ventilator will stay on for at most 90 minutes after `powerFailure`, producing the weakening

$$\Box(\texttt{powerFailure} \rightarrow \Box_{[1,90]}(\neg\texttt{off})). \tag{24}$$

Several case studies in Table 1 have few or no requirements that can be weakened with interval weakening. These requirements are typically liveness properties specified with the `eventually` timing, which cannot be weakened further, or safety properties specified with the `always` timing, for which interval weakening would not be appropriate. For example, from the grasping for debris removal case study [20],

$$\texttt{SV shall always satisfy !collide(SV, TGT).} \tag{25}$$

To answer **RQ2**, we have shown that interval weakening can be applied to the requirements of a number of real-world systems, as nearly a third of the analysed requirements could be weakened. We also used example requirements to demonstrate how these weakenings would be useful in domain-specific settings for understanding how safety-critical guarantees change under system degradation. We also answered **RQ1** by using CEGIW to identify problems in a specification, and then deduce useful timing properties in the fixed system.

5 Conclusion

We present CEGIW, a novel algorithm for weakening intervals in MTL properties of degraded systems, and prove its correctness and optimality. We demonstrate how CEGIW can be used during the design phase to understand system limitations under degradation, and explore how the formalised requirements of a number of real-world systems may be weakened against real implementations. This shows the applicability of CEGIW in the design of safety-critical systems for understanding the impacts of system degradation.

Future work. A current limitation is that we only weaken on the right-hand-side of intervals, when both left- and right-modifications can produce valid weakenings. Restricting to only right-modifications creates a total order over the search space, so there is always a single optimum when multiple choices exist. Expanding to both left- and right-modifications creates a partial order over generated intervals, and so choosing between intervals is much less obvious.

References

[1] Fides Aarts, Faranak Heidarian, Harco Kuppens, Petur Olsen, and Frits Vaandrager. "Automata Learning through Counterexample Guided Abstraction Refinement". In: *Formal Methods.* 2012, pp. 10–27. DOI: 10.1007/978-3-642-32759-9_4.

[2] Alexandre Abreu, Nuno Macedo, and Alexandra Mendes. "Exploring Automatic Specification Repair in Dafny Programs". In: *IEEE/ACM International Conference on Automated Software Engineering Workshops (ASEW).* 2023, pp. 105–112. DOI: 10.1109/ASEW60602.2023.00019.

[3] S. Akshay, Prerak Contractor, Paul Gastin, R. Govind, and B. Srivathsan. *Efficient Verification of Metric Temporal Properties with Past in Pointwise Semantics.* https://arxiv.org/abs/2510.14699v1. 2025.

[4] Rajeev Alur, Rastislav Bodik, Garvit Juniwal, Milo M. K. Martin, Mukund Raghothaman, Sanjit A. Seshia, Rishabh Singh, Armando Solar-Lezama, Emina Torlak, and Abhishek Udupa. "Syntax-Guided Synthesis". In: *Formal Methods in Computer-Aided Design.* 2013, pp. 1–8. DOI: 10.1109/FMCAD.2013.6679385.

[5] Rajeev Alur and Thomas A. Henzinger. "Real-Time Logics: Complexity and Expressiveness". In: *Information and Computation* 104.1 (1993), pp. 35–77. DOI: 10.1006/inco.1993.1025.

[6] Rajeev Alur, Salar Moarref, and Ufuk Topcu. "Counter-Strategy Guided Refinement of GR(1) Temporal Logic Specifications". In: *Formal Methods in Computer-Aided Design.* 2013, pp. 26–33. DOI: 10.1109/FMCAD.2013.6679387.

[7] Ben M. Andrew. "Weakening Goals in Logical Specifications". In: *Rigorous State-Based Methods.* 2026, pp. 349–353. DOI: 10.1007/978-3-031-94533-5_22.

[8] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking.* 2008.

[9] Hamza Bourbouh, Marie Farrell, Anastasia Mavridou, Irfan Sljivo, Guillaume Brat, Louise A. Dennis, and Michael Fisher. "Integrating Formal Verification and Assurance: An Inspection Rover Case Study". In: *NASA Formal Methods.* Vol. 12673. 2021, pp. 53–71. DOI: 10.1007/978-3-030-76384-8_4.

[10] Thomas Brihaye, Gilles Geeraerts, Hsi-Ming Ho, Arthur Milchior, and Benjamin Monmege. "Efficient Algorithms and Tools for MITL Model-Checking and Synthesis". In: *2018 23rd International Conference on Engineering of Complex Computer Systems (ICECCS).* 2018, pp. 180–184. DOI: 10.1109/ICECCS2018.2018.00027.

[11] Matías Brizzio, Maxime Cordy, Mike Papadakis, César Sánchez, Nazareno Aguirre, and Renzo Degiovanni. "Automated Repair of Unrealisable LTL Specifications Guided by Model Counting". In: *Proceedings of the Genetic and Evolutionary Computation Conference.* GECCO '23. 2023, pp. 1499–1507. DOI: 10.1145/3583131.3590454.

[12] Roberto Cavada, Alessandro Cimatti, Michele Dorigatti, Alberto Griggio, Alessandro Mariotti, Andrea Micheli, Sergio Mover, Marco Roveri, and Stefano Tonetta. “The NUXMV Symbolic Model Checker”. In: *Computer Aided Verification.* 2014, pp. 334–342. DOI: 10.1007/978-3-319-08867-9_22.

[13] Davide G. Cavezza, Dalal Alrajeh, and András György. “Minimal Assumptions Refinement for Realizable Specifications”. In: *Proceedings of the 8th International Conference on Formal Methods in Software Engineering.* FormaliSE ’20. 2020, pp. 66–76. DOI: 10.1145/3372020.3391557.

[14] Jorge Cerqueira, Alcino Cunha, and Nuno Macedo. “Timely Specification Repair for Alloy 6”. In: *Software Engineering and Formal Methods.* 2022, pp. 288–303. DOI: 10.1007/978-3-031-17108-6_18.

[15] Edmund Clarke, Orna Grumberg, Somesh Jha, Yuan Lu, and Helmut Veith. “Counterexample-Guided Abstraction Refinement”. In: *Computer Aided Verification.* 2000, pp. 154–169. DOI: 10.1007/10722167_15.

[16] Edmund Clarke, Daniel Kroening, Joël Ouaknine, and Ofer Strichman. “Completeness and Complexity of Bounded Model Checking”. In: *Verification, Model Checking, and Abstract Interpretation.* Vol. 2937. 2004, pp. 85–96. DOI: 10.1007/978-3-540-24622-0_9.

[17] Jamieson M. Cobleigh, Dimitra Giannakopoulou, and Corina S. Păsăreanu. “Learning Assumptions for Compositional Verification”. In: *Tools and Algorithms for the Construction and Analysis of Systems.* Vol. 2619. 2003, pp. 331–346. DOI: 10.1007/3-540-36577-X_24.

[18] Marie Farrell, Matt Luckcuck, Rosemary Monahan, Conor Reynolds, and Oisín Sheridan. “FRETting and Formal Modelling: A Mechanical Lung Ventilator”. In: *Rigorous State-Based Methods.* 2024, pp. 360–383. DOI: 10.1007/978-3-031-63790-2_28.

[19] Marie Farrell, Matt Luckcuck, Oisín Sheridan, and Rosemary Monahan. “FRETting About Requirements: Formalised Requirements for an Aircraft Engine Controller”. In: *Requirements Engineering: Foundation for Software Quality.* 2022, pp. 96–111. DOI: 10.1007/978-3-030-98464-9_9.

[20] Marie Farrell, Nikos Mavrakis, Angelo Ferrando, Clare Dixon, and Yang Gao. “Formal Modelling and Runtime Verification of Autonomous Grasping for Active Debris Removal”. In: *Frontiers in Robotics and AI* 8 (2022). DOI: 10.3389/frobt.2021.639282.

[21] Luca Gazzola, Daniela Micucci, and Leonardo Mariani. “Automatic Software Repair: A Survey”. In: *Proceedings of the 40th International Conference on Software Engineering.* ICSE ’18. 2018, p. 1219. DOI: 10.1145/3180155.3182526.

[22] Dimitra Giannakopoulou, Thomas Pressburger, Anastasia Mavridou, Julian Rhein, Johann Schumann, and Nija Shi. “Formal Requirements Elicitation with FRET”. In: *International Working Conference on Requirements Engineering: Foundation for Software Quality* (2020).

[23] Dimitra Giannakopoulou, Thomas Pressburger, Anastasia Mavridou, and Johann Schumann. “Automated Formalization of Structured Natural Lan-

guage Requirements". In: *Information and Software Technology* 137 (2021), p. 106590. DOI: 10.1016/j.infsof.2021.106590.

[24] Gerard J. Holzmann. "The Model Checker SPIN". In: *IEEE Transactions on Software Engineering* 23.5 (1997), pp. 279–295. DOI: 10.1109/32.588521.

[25] Falk Howar, Bernhard Steffen, and Maik Merten. "Automata Learning with Automated Alphabet Abstraction Refinement". In: *Verification, Model Checking, and Abstract Interpretation.* 2011, pp. 263–277. DOI: 10.1007/978-3-642-18275-4_19.

[26] ISO. *Particular Requirements for Basic Safety and Essential Performance of Critical Care Ventilators.* 80601-2-12. 2023.

[27] Ron Koymans. "Specifying Real-Time Properties with Metric Temporal Logic". In: *Real-Time Systems* 2.4 (1990), pp. 255–299. DOI: 10.1007/BF01995674.

[28] Wenguo Liu and Alan F. T. Winfield. "Modeling and Optimization of Adaptive Foraging in Swarm Robotic Systems". In: *The International Journal of Robotics Research* 29.14 (2010), pp. 1743–1760. DOI: 10.1177/0278364910375139.

[29] Shahar Maoz, Jan Oliver Ringert, and Rafi Shalom. "Symbolic Repairs for GR(1) Specifications". In: *IEEE/ACM 41st International Conference on Software Engineering (ICSE).* 2019, pp. 1016–1026. DOI: 10.1109/ICSE.2019.00106.

[30] Anastasia Mavridou, Hamza Bourbouh, Dimitra Giannakopoulou, Thomas Pressburger, Mohammad Hejase, Pierre-Loïc Garoche, and Johann Schumann. "The Ten Lockheed Martin Cyber-Physical Challenges: Formalized, Analyzed, and Explained". In: *IEEE 28th International Requirements Engineering Conference (RE).* 2020, pp. 300–310. DOI: 10.1109/RE48521.2020.00040.

[31] Tom Pressburger, Andreas Katis, Aaron Dutle, and Anastasia Mavridou. "Authoring, Analyzing, and Monitoring Requirements for a Lift-Plus-Cruise Aircraft". In: *Requirements Engineering: Foundation for Software Quality.* 2023, pp. 295–308. DOI: 10.1007/978-3-031-29786-1_21.

[32] Oisín Sheridan, Leandro Buss Becker, Marie Farrell, Matt Luckcuck, and Rosemary Monahan. "Sharper Specs for Smarter Drones: Formalising Requirements with FRET". In: *Requirements Engineering: Foundation for Software Quality.* 2025, pp. 350–362. DOI: 10.1007/978-3-031-88531-0_25.

[33] Gricel Vázquez, Anastasia Mavridou, Marie Farrell, Tom Pressburger, and Radu Calinescu. "Robotics: A New Mission for FRET Requirements". In: *NASA Formal Methods.* 2024, pp. 359–376. DOI: 10.1007/978-3-031-60698-4_22.

A Remaining Algorithm and Proofs

Proof of Lemma 14, directly weakening the interval of $\mathcal{R}$.

Proof. We take an arbitrary t. Our proof for Algorithm 4 uses the loop invariant that for all $j \in \mathrm{cov}_\pi(I)$ with $j < i$, we have that $\pi, t + j \vDash \psi_r$ and $\pi, t + j \nvDash \psi_l$.

Algorithm 4: Directly weakening interval of $\mathcal{R}$

```
function $WeakenRDirect(\psi_l, \psi_r, [a,b], t)$
    $b_{\text{fin}} \leftarrow \min(b, \text{end}_\pi(a))$
    for $i \leftarrow a$ to $b_{\text{fin}}$ do
        if $\pi, t+i \nvDash \psi_r$ then
            if $i = a$ then
                return $None$
            return $[a, i-1]$
        if $\pi, t+i \vDash \psi_l$ then
            return $[a, b]$
    return $[a, b]$
```

On first entry to the loop there is no such j, so this is trivially true. On reaching the end of the loop body, we know that $\pi, t+i \nvDash \psi_r$ and $\pi, t+i \vDash \psi_l$, and so in combination with the loop invariant we know that for all $j \in \text{cov}_\pi(I)$ with $j \leq i$, we have that $\pi, t+j \nvDash \psi_r$ and $\pi, t+j \vDash \psi_l$. Thus the loop invariant is preserved. Suppose at the start of iteration i the loop invariant holds. If $\pi, t+i \nvDash \psi_r$ then we have two cases:

1. If $i = a$ we have that $\pi, t+a \nvDash \psi_r$, so for all possible right-contractions I' of I we have that $\pi, t \nvDash \psi_l \; \mathcal{R}_{I'} \, \psi_r$. Thus, there is no suitable interval and we return $None$.
2. Otherwise, we return $I' = [a, i-1]$, which is an optimal right contraction of I. By the loop invariant we know for all $j \in \text{cov}_\pi(I)$ with $j < i$ that $\pi, t+j \vDash \psi_r$, so we can conclude that $\pi, t \vDash \psi_l \; \mathcal{R}_{I'} \, \psi_r$.

If during this iteration i we have that $\pi, t+i \vDash \psi_l$, then as this is after the above case is checked for we know that $\pi, t+i \vDash \psi_r$, and in combination with the loop invariant we have that $\pi, t \vDash \psi_l \; \mathcal{R}_I \, \psi_r$. If we run the loop to completion, then by the loop invariant we know that for all $i \in \text{cov}_\pi(I)$ we have $\pi, t+j \vDash \psi_r$. If $\text{cov}_\pi(I) = I$ then we have

$$\pi, t \vDash \psi_l \; \mathcal{R}_I \, \psi_r$$

If $\text{cov}_\pi(I) = [a, \text{end}_\pi(a)]$ for some $a \in \mathbb{N}$, then by Lemma 11 we have that for all $i \in \mathbb{N}$ with $i \geq a$ we have $\pi, t+j \vDash \psi_r$, and so the above holds here too. □

Proof of Lemma 16, weakening within $\mathcal{U}$ on the right.

Proof. We assume the inductive hypothesis $P(C)$, and want to prove $P(\phi \, \mathcal{U}_J \, C)$. We take an arbitrary t. We distinguish two cases, according to whether $\triangle$ is $\mathcal{U}$ or $\mathcal{R}$. In either case, by the induction hypothesis each recursive call evaluates to either $None$ or an interval I' related to I by the corresponding relation (right-extension or -contraction respectively) such that $\pi, t+t' \vDash C[\psi \, \triangle_{I'} \psi']$.

We use the loop invariant that for all $j \in \text{cov}_\pi(J)$ with $j < i$, we have that $\pi, t+j \vDash \phi$. On first entry to the loop there is no such j, so this is trivially

Algorithm 5: Weakening within $\mathcal{U}$ on the right

```
function WeakenURight(C, $\phi$, $[a, b]$, t)
    $b_{\text{fin}} \leftarrow \min(b, \text{end}_\pi(a))$
    $intervals \leftarrow [\,]$
    for $i \leftarrow a$ to $b_{\text{fin}}$ do
        $I \leftarrow WeakenRec(C, t + i)$
        if $I \neq None$ then
            append $I$ to $intervals$
        if $\pi, t + i \nvDash \phi$ then
            break
    if intervals is empty then
        return None
    return interval in $intervals$ with minimal absolute difference to $I_{\text{orig}}$
```

Algorithm 6: Weakening within $\mathcal{U}$ on the right

```
function WeakenRRight(C, $\phi$, $[a, b]$, t)
    $b_{\text{fin}} \leftarrow \min(b, \text{end}_\pi(a))$
    $intervals \leftarrow [\,]$
    for $i \leftarrow a$ to $b_{\text{fin}}$ do
        $I \leftarrow WeakenRec(C, t + i)$
        if $I = None$ then
            return None
        append $I$ to $intervals$
        if $\pi, t + i \vDash \phi$ then
            break
    return interval in $intervals$ with maximal absolute difference to $I_{\text{orig}}$
```

true. On reaching the end of the loop body, we know for all $j \in \text{cov}_\pi(J)$ with $j \leq i$ that $\pi, t + j \vDash \phi$, and so the loop invariant is preserved. Suppose at the start of iteration i the loop invariant holds. If $\pi, t + i \nvDash \phi$ we exit the loop, and if the list of intervals is empty then by the loop invariant we know that there are no appropriate intervals I' for any $j \in \text{cov}_\pi(J)$ with $j < i$ for which $\pi, t + j \vDash C[\psi \triangle_{I'} \psi']$, and so the same holds for $\pi, t \vDash (\phi\, \mathcal{U}_J\, C)[\psi \triangle_{I'} \psi']$.

Otherwise, by the loop invariant we know that for all $j \in \text{cov}_\pi(J)$ with $j < i$ we have $\pi, t + j \vDash \phi$ and, as we have passed the check above, that for at least one of these j we have an interval I' such that $\pi, t + j \vDash C[\psi \triangle_{I'} \psi']$. Thus, we have

$$\pi, t \vDash (\phi\, \mathcal{U}_J\, C)[\psi \triangle_{I'} \psi']$$

□

Proof of Lemma 18, weakening within $\mathcal{R}$ on the right.

Algorithm 7: Weakening within $\mathcal{R}$ on the left

```
function WeakenRLeft(C, $\phi$, $[a, b]$, t)
    $b_{\mathrm{fin}} \leftarrow \min(b, \mathrm{end}_\pi(a))$
    intervals $\leftarrow$ [ ]
    for $i \leftarrow a$ to $b_{\mathrm{fin}}$ do
        if $\pi, t + i \nvDash \phi$ then
            break
        $I \leftarrow WeakenRec(C, t + i)$
        if $I \neq None$ then
            append $I$ to intervals
    if intervals is empty then
        return None
    return interval in intervals with minimal absolute difference to $I_{\mathrm{orig}}$
```

Proof. We assume the inductive hypothesis $P(C)$, and want to prove $P(C\ \mathcal{R}_J\ \phi)$. We take an arbitrary t. We distinguish two cases, according to whether $\triangle$ is $\mathcal{U}$ or $\mathcal{R}$. In either case, by the induction hypothesis each recursive call evaluates to either $None$ or an interval I' related to I by the corresponding relation (right-extension or -contraction respectively) such that $\pi, t + t' \vDash C[\psi \triangle_{I'} \psi']$.

We use the loop invariant that for all $j \in \mathrm{cov}_\pi(J)$ with $j < i$, we have that $\pi, t + j \vDash \phi$. On first entry to the loop there is no such j, so this is trivially true. On reaching the end of the loop body, we know that for all $j \in \mathrm{cov}_\pi(J)$ with $j \leq i$ that $\pi, t + j \vDash \phi$, so the loop invariant holds. Suppose at the start of iteration i the loop invariant holds. If $\pi, t + i \nvDash \phi$ we exit the loop, and if the list of intervals is empty then we know that there are no appropriate intervals I' for any $j \in \mathrm{cov}_\pi(J)$ with $j < i$ for which $\pi, t + j \vDash C[\psi \triangle_{I'} \psi']$, and so the same holds for $\pi, t \vDash (C\ \mathcal{R}_J\ \phi)[\psi \triangle_{I'} \psi']$.

Otherwise, by the loop invariant we know that for all $j \in \mathrm{cov}_\pi(J)$ with $j < i$ we have $\pi, t + j \vDash \phi$, and, as we have passed the check above, that for at least one of these j we have an interval I' such that $\pi, t + j \vDash C[\psi \triangle_{I'} \psi']$. Thus, we have

$$\pi, t \vDash (C\ \mathcal{R}_J\ \phi)[\psi \triangle_{I'} \psi']$$

□

Proof of Lemma 18, weakening within $\mathcal{R}$ on the right.

Proof. We assume the inductive hypothesis $P(C)$, and want to prove $P(\phi\ \mathcal{R}_J\ C)$. We take an arbitrary t. We distinguish two cases, according to whether $\triangle$ is $\mathcal{U}$ or $\mathcal{R}$. In either case, by the induction hypothesis each recursive call evaluates to either $None$ or an interval I' related to I by the corresponding relation (right-extension or -contraction respectively) such that $\pi, t + t' \vDash C[\psi \triangle_{I'} \psi']$.

We use the loop invariant that for all $j \in \mathrm{cov}_\pi(J)$ with $j < i$, we have that $\pi, t + j \nvDash \phi$ and that $I' = WeakenRec(C, \psi \triangle_I \psi', t + j)$ is an interval such that $\pi, t + j \vDash C[\psi \triangle_{I'} \psi']$. On first entry to the loop there is no such j, so this is trivially true. On reaching the end of the loop body, we know that $\mathit{WeakenRec}(C, \psi \triangle_I \psi', t + i) \neq \mathit{None}$, and so by the induction hypothesis the recursive call must have produced a suitable interval I'. As we also know that $\pi, t + i \nvDash \phi$, the loop invariant is thus preserved for $j \in \operatorname{cov}_\pi(J)$ with $j \leq i$. Suppose at the start of iteration i the loop invariant holds. If $I' = \mathit{WeakenRec}(C, \psi \triangle_I \psi', t+i) = \mathit{None}$ then by the induction hypothesis we know that there is no suitable interval I'' for which $\pi, t + i \vDash C[\psi \triangle_{I''} \psi']$, and by the loop invariant that there is no $j \in \operatorname{cov}_\pi(J)$ with $j < i$ for which $\pi, t + j \vDash \phi$. Thus, there is no interval I'' for which $\pi, t \vDash (C \,\mathcal{U}_J\, \phi)[\psi \triangle_{I''} \psi']$.

If $\pi, t+i \vDash \phi$, then we exit the loop. As this check occurred after the recursive call to $\mathit{WeakenRec}$, we know that we have found at least one suitable interval. By the loop invariant we know that for all $j \in \operatorname{cov}_\pi(J)$ with $j < i$ we have an interval I' such that $\pi, t + j \vDash C[\psi \triangle_{I'} \psi']$. Applying Lemma 7 if $\triangle = \mathcal{U}$, or Lemma 8 if $\triangle = \mathcal{R}$, we obtain a maximum interval I'' such that for all $j \in \operatorname{cov}_\pi(J)$ with $j \leq i$ we have $\pi, t + j \vDash C[\psi \triangle_{I''} \psi']$, and so

$$\pi, t \vDash (\phi \,\mathcal{R}_J\, C)[\psi \triangle_{I''} \psi']$$

If we run the loop to completion, then by the loop invariant for each $i \in \operatorname{cov}_\pi(J)$ we have a suitable interval I' where $\pi, t + j \vDash C[\psi \triangle_{I'} \psi']$. If $\operatorname{cov}_\pi(J) = J$ then by the same reasoning as above we have a suitable maximum interval. If $\operatorname{cov}_\pi(J) = [a, \operatorname{end}_\pi(a)]$ for some a then by Lemma 11 this property holds for all $i \in \mathbb{N}$ with $i \geq a$, and so in this case we also have a maximum interval. □