2-16-2017

Relaxed decidability and the robust semantics of Metric Temporal Logic

Houssam Abbas
University of Pennsylvania, habbas@seas.upenn.edu

Matthew O'Kelly
University of Pennsylvania, mokelly@seas.upenn.edu

Rahul Mangharam
University of Pennsylvania, rahulm@seas.upenn.edu

Relaxed decidability and the robust semantics of Metric Temporal Logic

Abstract

Relaxed notions of decidability widen the scope of automatic verification of hybrid systems. In quasi-decidability and δ-decidability, the fundamental compromise is that if we are willing to accept a slight error in the algorithm's answer, or a slight restriction on the class of problems we verify, then it is possible to obtain practically useful answers. This paper explores the connections between relaxed decidability and the robust semantics of Metric Temporal Logic formulas. It establishes a formal equivalence between the robustness degree of MTL specifications, and the imprecision parameter δ used in δ-decidability when it is used to verify MTL properties. We present an application of this result in the form of an algorithm that generates new constraints to the δ-decision procedure from falsification runs, which speeds up the verification run. We then establish new conditions under which robust testing, based on the robust semantics of MTL, is in fact a quasi-semidecision procedure. These results allow us to delimit what is possible with fast, robustness-based methods, accelerate (near-)exhaustive verification, and further bridge the gap between verification and simulation.

Keywords

Logic and verification, Cyber-Physical Systems, Reachability, Falsification, Robustness, Hybrid Systems

Disciplines

Computer Engineering | Electrical and Computer Engineering

Relaxed Decidability and the Robust Semantics of Metric Temporal Logic

Houssam Abbas
Electrical and Systems Engineering Department
The University of Pennsylvania
Philadelphia, Pennsylvania
habbas@seas.upenn.edu

Matthew O'Kelly
Electrical and Systems Engineering Department
The University of Pennsylvania
Philadelphia, Pennsylvania
mokelly@seas.upenn.edu

Rahul Mangharam
Electrical and Systems Engineering Department
The University of Pennsylvania
Philadelphia, Pennsylvania
rahulm@seas.upenn.edu

ABSTRACT

Relaxed notions of decidability widen the scope of automatic verification of hybrid systems. In quasi-decidability and δ-decidability, the fundamental compromise is that if we are willing to accept a slight error in the algorithm's answer, or a slight restriction on the class of problems we verify, then it is possible to obtain practically useful answers. This paper explores the connections between relaxed decidability and the robust semantics of Metric Temporal Logic formulas. It establishes a formal equivalence between the robustness degree of MTL specifications, and the imprecision parameter δ used in δ-decidability when it is used to verify MTL properties. We present an application of this result in the form of an algorithm that generates new constraints to the δ-decision procedure from falsification runs, which can speed up the verification run. We then establish new conditions under which robust testing, based on the robust semantics of MTL, is in fact a quasi-semidecision procedure. These results allow us to delimit what is possible with fast, robustness-based methods, accelerate (near-)exhaustive verification, and further bridge the gap between verification and simulation.

Keywords

Hybrid systems; robust semantics; falsification; metric temporal logic; δ-decidability; quasi-decidability

1. INTRODUCTION

The formal analysis of hybrid dynamical systems initially focused on decidability considerations. Studies such as [23, 4, 22] analyzed classes of hybrid systems for which questions like reachability could be decided. The commonly accepted lesson of these initial investigations was that most hybrid systems are undecidable, with the decidable class being rather special and placing strong limitations on what we can model and verify automatically.

Relaxed decidability. Partially as a result of this conclusion, two independent trends emerged, which we view as trying to bridge the gap between exhaustive verification (expensive, complete and sound) and testing (inexpensive, incomplete and sound).[1] See Fig. 1 (A)-(B). The first trend defined and applied *relaxed* notions of decidability to the analysis of hybrid systems [17, 10, 15, 31, 16, 20, 19, 21]. Broadly speaking, these works re-formulated the *safety* problem for hybrid systems as a first-order formula over real constraints: does there exist an initial point x_0 such that a system trajectory starting from x_0 reaches the unsafe set while respecting the system dynamics? In quasi-decidability [15, 31, 16], the (quasi-)decision procedure always returns a correct YES/NO answer to this question, except for 'pathological' cases on which it might run forever. The argument then is that such pathological cases are of little interest in practical system design. In δ-decidability [20, 19, 21], the (δ-complete) decision procedure always halts and returns either a correct NO answer (the formula is not true, i.e. the system is safe) or an approximate δ-YES answer (the formula may be false but a small δ-sized perturbation of it is true, i.e. the system is δ-close to being unsafe). The argument, then, is that a system which is δ-close to being unsafe should be, for all practical purposes, considered unsafe. Thus, this research thrust relaxes exhaustive verification to make it more widely applicable, at the cost of small errors in the answer or the arguably small likelihood of never getting an answer. These approaches were implemented in software tools (iSAT, dReach and HSolver).

Robustness-guided methods. Separately from the above efforts, the second line of research [12, 8, 24, 2] sought to put *falsification* (a.k.a. testing) on a more rigorous footing, thus bringing it closer to exhaustive verification. See Fig. 1 (C)-(D). This was done to leverage falsification's ability to handle any system, including black boxes, and any specification, not just safety. An additional benefit is that it only uses relatively inexpensive simulations. This enhancement to falsification was accomplished by defining a real-valued *robust satisfaction degree* of the formal specification expressed in Metric Temporal Logic (MTL). This robustness was used as an objective function to perform *robustness-guided falsification*: [1, 29, 2]. By minimizing the robustness over the set of initial conditions, we reach a system trajectory that violates the specification. It can also be used in *robust testing* [24], in which a finite number of simulations could cover the entire set of system behaviors. Thus, this approach provides stronger guarantees on the outcome of

HSCC'17, April 18-20, 2017, Pittsburgh, PA, USA

DOI: http://dx.doi.org/10.1145/3049797.3049813

[1] Other approaches exist, of course, but they are not in the scope of this paper.

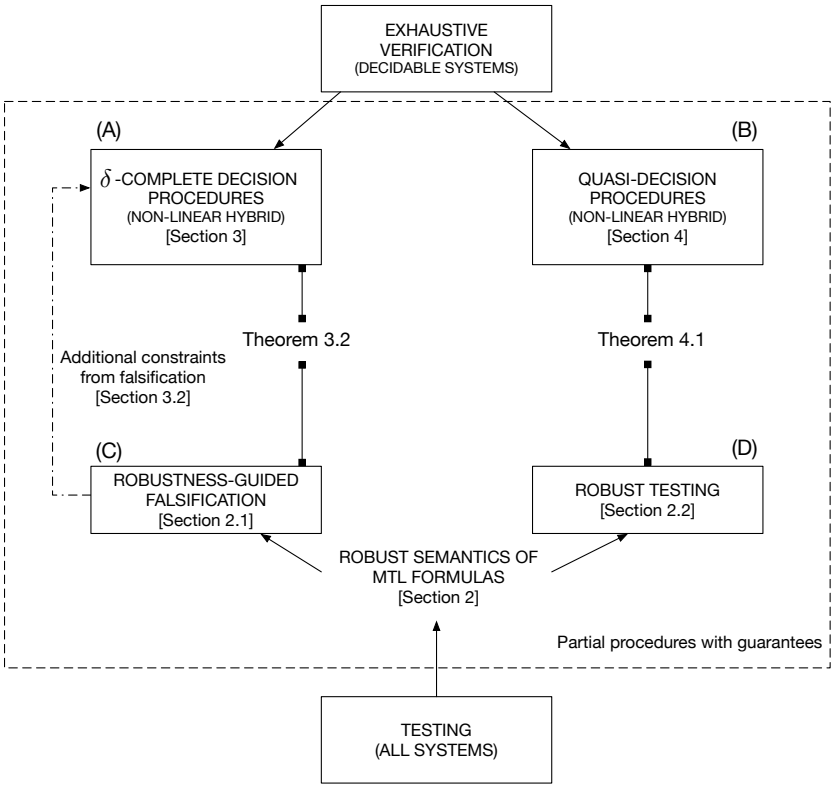

Figure 1: The rapprochement between formal verification and testing. Robustness-guided falsification is integrated with δ-complete decision procedures and Robust testing is a quasi-semidecision procedure.

testing and seeks better-performing falsification algorithms applicable to a broad class of problems.

While the connection between the relaxed decidability notions was previously observed [16], *the connection between the robust semantics and relaxed decidability has not been explored*. Notions of robustness are fundamental to both approaches, so it is tempting to study robustness-guided methods in the light of relaxed decidability. Our motivation is both theoretical and practical: we take testing as our starting point, and want to provide rigorous ways in which robustness-guided testing can accelerate relaxed decision procedures, and to delimit what is theoretically possible with robustness-guided methods.

Contributions. We establish a formal equivalence between the robustness degree of MTL specifications, and the imprecision parameter δ used in δ-decidability (Section 3). Informally, we find that δ-decidability is computing the robust semantics of the MTL formula and deciding whether it is negative (formula is False) or larger than $-\delta$ (formula is δ-True). We present an application of this connection by using the results of falsification to further constrain the operation of a δ-decidability tool (Section 3.3). Empirical evidence obtained using our approach demonsrates runtime savings for the δ-decider, which we expect will improve its scalability to larger systems. The paper then turns to the relation between robust testing and quasi-decidability. We establish a relation between the robust semantics of MTL and the notion of quasi-robustness. We then give new relaxed conditions under which Robust Testing terminates for (almost everywhere) robustly correct systems (Section 4). In the process, we delimit the class of MTL formulas for which there can be an arbitrary difference between the exact robustness degree of an MTL formula and the robust semantics used to approximate it (Section 4.1.2). The plan of the paper is given in Fig. 1.

This study opens the way to a principled integration of falsification and exhaustive verification, where inexpensive but robust simulations are *an integral part* of (relaxed) exhaustive verification algorithms, rather than an independent accessory in the verification process.

All proofs appear in the online technical report [3].

2. ROBUSTNESS OF MTL FORMULAS

A *falsification algorithm* searches a system's set of initial conditions $X_0 \subset \mathbb{R}^n$ for a point x_0 from which the system exhibits a trajectory that falsifies (i.e., violates) the system's specification. When the specification is expressed in Metric Temporal Logic (MTL) [28], then the *robustness degree* of the specification can be used to guide the search. We now define the robustness degree of an MTL formula and describe how it's approximated by the robust semantics of MTL formulas. The formal connections between these concepts and relaxed decidability are established in the next sections.

Notation. The word *signal* will refer to a function from some bounded time domain $\mathbb{T} \subset \mathbb{R}$ to the bounded state space $X \subset \mathbb{R}^n$. The set of all signals $\mathbf{x} : \mathbb{T} \to X$ is $X^{\mathbb{T}}$. Signals are denoted by the letters $\mathbf{x}, \mathbf{y}$, etc. The value of signal $\mathbf{x}$ at time t is x_t. All time intervals $I \subset \mathbb{R}$ that

appear in what follows should be interpreted as meaning $I \cap \mathbb{T}$. Given $t \in \mathbb{R}$ and $I \subset \mathbb{R}$, $t +_{\mathbb{T}} I = \mathbb{T} \cap \{t + t' | t' \in I\}$. The symbols $\sqcup$ and $\sqcap$ denote the sup and inf opereators, respectively. A *trajectory* is a signal generated by a hybrid system. A trajectory starting from x_0 is denoted $\mathbf{y}_{x_0}$. $\mathcal{P}(X)$ is the set of all subset of X, and $\mathrm{cl}(X)$ is its closure. The positive reals are $\mathbb{R}_+ := (0, \infty)$, and the negative reals are $\mathbb{R}_- = (-\infty, 0)$.

Let (A, d) be a metric space; that is, the distance function $d : A \times A \to \mathbb{R}_+$ is non-negative, symmetric, respects the triangle inequality and is 0 iff its arguments are equal. Let $Y \subset A$ be a subset of A, and let $\mathrm{cl}(Y)$ denote its closure in the metric topology. Then we define:

$$\mathbf{dist}_d(x, Y) := \inf_{y \in \mathrm{cl}(Y)} d(x, y) \quad (1)$$

$$\mathbf{depth}_d(x, Y) := \mathbf{dist}_d(x, A \setminus Y) \quad (2)$$

$$\mathbf{Dist}_d(x, Y) := \begin{cases} -\mathbf{dist}_d(x, Y), & x \notin Y \\ \mathbf{depth}_d(x, Y), & x \in Y \end{cases} \quad (3)$$

$$B_d(x, r) := \{y \in A \mid d(x, y) < r\} \quad (4)$$

For example, if $A = X$ is the state space and $d(a, b) = |a - b|$ is the Euclidian distance between points in X, then the above define, respectively, the distance of a point to a subset $Y \subset X$, the depth of a point in a set Y, the signed distance of point x to set Y (with positive value indicating the point a is in Y, and a negative value indicating otherwise), and the open ball of radius r centered on x. As another important example, $A = X^{\mathbb{T}}$ can be the signal space and $d(\mathbf{x}, \mathbf{y}) = \rho(\mathbf{x}, \mathbf{y}) := \sup_{t \in \mathbb{T}} |x_t - y_t|$ is the sup norm of the difference between the signals $\mathbf{x}$ and $\mathbf{y}$.

Let AP be a set of atomic propositions, and let φ be a formula in MTL$^+$, the set of MTL formulas in Negative Normal Form (so only atomic propositions can have a $\neg$ preceding them):

$$\varphi := \top | p | \neg p | \varphi_1 \vee \varphi_2 | \varphi_1 \wedge \varphi_2 | \varphi_1 \mathcal{U}_I \varphi_2 | \varphi_1 \mathcal{R}_I \varphi_2$$

Let $\mathcal{O} : AP \to \mathcal{P}(X)$ be an observation map for the atomic propositions. That is, for every $p \in AP$, $\mathcal{O}(p) = \{x \in X \mid x \models p\}$.

Assumption 2.1. *Unless otherwise indicated, signals are continuous-time. The set X is a bounded box in $\mathbb{R}^n$: $X = [a_1, b_1] \times \ldots \times [a_n, b_n]$, and is not included in any $\mathcal{O}(p)$. All formulas have a bounded horizon (all of their temporal intervals are bounded). Therefore, we may also assume that all trajectories have fininte length, that is no longer than the formula's horizon. When we need to compute the robustness (defined below) of a system trajectory, we assume a rigorous simulator is used, and a lower bound on the rigorous simulation's robustness is computed, as explained in [3, Section 6].*

Let $t \in \mathbb{T}$ be a time instant and φ be an MTL$^+$ formula. $\mathcal{L}_t(\varphi)$ is the set of signals in $X^{\mathbb{T}}$ that satisfy φ at time t, that is, $\mathcal{L}_t(\varphi) = \{\mathbf{x} \in X^{\mathbb{T}} \mid (\mathbf{x}, t) \models_{\mathcal{O}} \varphi\}$.

Definition 2.1. *[12] Define the distance $\rho : X^{\mathbb{T}} \times X^{\mathbb{T}} \to \mathbb{R}_+$ by $\rho(\mathbf{x}, \mathbf{y}) = \sup_{t \in \mathbb{T}} d(x_t, y_t)$. The* robustness degree *of signal $\mathbf{x}$ at time t relative to formula φ under observation $\mathcal{O}$ is $\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))$.*

By definition, if for two signals $\mathbf{x}, \mathbf{y}$ it holds that $\rho(\mathbf{x}, \mathbf{y}) < |\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))|$ then either both signals are in $\mathcal{L}_t(\varphi)$ or both are outside it, and so they both have the same truth value relative to φ. The robustness degree therefore defines a *level of perturbation* to $\mathbf{x}$ which will not change its truth value relative to φ. The perturbation is measured using the distance function ρ.

The robustness degree, in general, cannot be computed directly because the set $\mathcal{L}_t(\varphi)$ cannot be characterized. However, it can be conservatively approximated by the *robustness estimate*, defined using the following semantics of MTL formulas.

Definition 2.2 (Robust semantics [12])**.** *The* robust semantics *of φ are denoted by $[\![\varphi, \mathcal{O}]\!](\mathbf{x}, t)$ and are defined as*

$$\begin{aligned}
[\![\top, \mathcal{O}]\!](\mathbf{x}, t) &= +\infty \\
\forall p \in AP,\ [\![p, \mathcal{O}]\!](\mathbf{x}, t) &= \mathbf{Dist}_d(x_t, \mathcal{O}(p)) \\
[\![\neg\varphi, \mathcal{O}]\!](\mathbf{x}, t) &= -[\![\varphi, \mathcal{O}]\!](\mathbf{x}, t) \\
[\![\varphi_1 \vee \varphi_2, \mathcal{O}]\!](\mathbf{x}, t) &= [\![\varphi_1, \mathcal{O}]\!](\mathbf{x}, t) \sqcup [\![\varphi_2, \mathcal{O}]\!](\mathbf{x}, t) \\
[\![\varphi_1 \wedge \varphi_2, \mathcal{O}]\!](\mathbf{x}, t) &= [\![\varphi_1, \mathcal{O}]\!](\mathbf{x}, t) \sqcap [\![\varphi_2, \mathcal{O}]\!](\mathbf{x}, t) \\
[\![\varphi_1 \mathcal{U}_I \varphi_2, \mathcal{O}]\!](\mathbf{x}, t) &= \sqcup_{t' \in t +_{\mathbb{T}} I} \Big([\![\varphi_2, \mathcal{O}]\!](\mathbf{x}, t') \bigsqcap \\
&\quad \sqcap_{t'' \in [t, t')} [\![\varphi_1, \mathcal{O}]\!](\mathbf{x}, t'') \Big) \\
[\![\varphi_1 \mathcal{R}_I \varphi_2, \mathcal{O}]\!](\mathbf{x}, t) &= \sqcap_{t' \in t +_{\mathbb{T}} I} \Big([\![\varphi_2, \mathcal{O}]\!](\mathbf{x}, t') \bigsqcup \\
&\quad \sqcup_{t'' \in [t, t')} [\![\varphi_1, \mathcal{O}]\!](\mathbf{x}, t'') \Big)
\end{aligned}$$

The robustness estimate of signal $\mathbf{x}$ relative to φ at time t under observation $\mathcal{O}$ is $[\![\varphi, \mathcal{O}]\!](\mathbf{x}, t)$.

A *determistic* hybrid system produces a unique trajectory $\mathbf{y}_{x_0}$ from any initial point x_0. Therefore we will speak interchangeably of the robustness estimate of $\mathbf{y}_{x_0}$ and the robustness of the initial point x_0. The following establishes that the robustness estimate is a conservative bound on the robustness degree [12]

Theorem 2.1. *For any $\mathbf{x} \in X^{\mathbb{T}}$ and MTL$^+$ formula φ, the following hold*

1. $-\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) \leq [\![\varphi, \mathcal{O}]\!](\mathbf{x}, t) \leq \mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))$
2. *If $r = [\![\varphi, \mathcal{O}]\!](\mathbf{x}, t) < 0$ then $\mathbf{x}$ falsifies the spec φ, and if $r > 0$ then $\mathbf{x}$ satisfies φ. The case $r = 0$ is inconclusive.*
3. *Any signal in $B_\rho(\mathbf{x}, |r|)$ has the same truth value relative to φ as $\mathbf{x}$.*

2.1 Robustness-guided falsification

We now present two applications of the robust semantics, starting with robustness-guided falsification. Using Thm. 2.1, a *robustness-guided falsification* algorithm searches for falsifying trajectories by minimizing the robustness estimate over X_0, the set of initial conditons of the system.

$$\min_{x_0 \in X_0} [\![\varphi, \mathcal{O}]\!](\mathbf{y}_{x_0}, t)$$

where $\mathbf{y}_{x_0}$ is the system trajectory starting from x_0. If the found minimum is negative then this means the corresponding minimizer $\mathbf{y}^*_{x_0}$ falsifies φ. Falsification uses relatively fast simulations and only requires the ability to simulate the system.

2.2 Robust testing

A second application of the robust semantics is *robust testing* [13, 24]. It proceeds as shown in Algorithm 1: it iteratively samples the search space X_0 to yield a sequence of

Algorithm 1: Robust Testing

Data: An MTL formula φ, a system $\mathcal{H}$ with initial set $X_0 \subset \mathbb{R}^n$ and bisimulation $V : X \times X \to [0, \infty)$

1 Set $i = 0, X_r = X_0$;
`/* Sample while the balls have not yet covered` X_0 `*/`
2 **while** $X_r \neq \emptyset$ **do**
3 Sample x_i in the interior of X_r;
4 Compute $r_i = [\![\varphi, \mathcal{O}]\!](\mathbf{y}_{x_i}, t)$;
5 **if** $r_i < 0$ **then**
6 Return False ; `/* Found a falsifier */`
7 **else if** $r_i > 0$ **then**
`/* Compute a 'robustness ball'` B_i `around the initial point` x_i `*/`
8 Compute $c_i > 0$ s.t. $\forall\, z \in X_0$, $d(x_i, z) < c_i \implies d(\mathbf{y}_{x_i}(t), \mathbf{y}_z(t)) < r\, \forall\, t \in \mathbb{T}$;
9 Set $X_r = X_0 \setminus B_d(x_i; c_i)$
10 **else**
`/* No robustness ball - keep sampling */`
11 **end**
12 $i = i + 1$
13 **end**
14 Return True. `/* Covered` X_0 `with the balls` B_i `*/`

samples $x_0, x_1, \ldots$. If a new sample x_i yields a trajectory $\mathbf{y}_{x_i}$ with negative robustness $r_i < 0$, the algorithm returns False (Line 6). Otherwise, if $r_i > 0$, we know that any signal $\mathbf{x}$ within $B_\rho(\mathbf{y}_{x_i}, r_i)$ also satisfies φ. So we wish to exclude any points in X_0 that produce trajectories that stay in $B_\rho(\mathbf{y}_{x_i}; r_i)$ to avoid searching in them. We compute such a set of points $B_d(x_i; c_i)$ in Line 8, e.g., using bisimulations[2] [13]. The ball $B_d(x_i; c_i)$ is then excluded from X_0, and the sampling continues in the rest of the search space (Line 9). If X_0 is fully covered by the union of balls $\cup_i B_d(x_i; c_i)$ at some point, the algorithm halts and returns True. See Fig. 2.

Note that Robust Testing, as presented here, might not terminate. For example, if the sampler gets stuck sampling points of 0 robustness (Line 10), then it will run forever. Or, if the balls B_i become infinitesimally smaller, as shown in Fig. 2 (right), their union will never cover X_0.

Previous work has shown that Robust Testing terminates if the minimum robustness estimate of any system trajectory is *positive* [13, Thm. 21]. In essence, this guarantees that the '**if** $r_i > 0$' branch (Line 7) always executes, so every new sample reduces the residual search space X_r by a minimum amount r, $0 < r \leq r_i$. In Section 4 we establish a stronger result that extends the limits of what is achievable with Robust Testing.

3. ROBUSTNESS AND δ-DECIDABILITY

So-called δ-Complete Decision Procedures (δ-CDP) have been used to verify the safety of a large variety of hybrid systems. For examples, see the website of the tool dReach [26]. The approach to the problem is to write the reachability question

Do there exist initial conditions $x_0 \in X_0 \subset \mathbb{R}^n$ from which the system enters the unsafe set $U \subset \mathbb{R}^n$?

[2]Bisimulations are outside the scope of this paper. The reader is referred to [13] for details.

as a first-order formula over the reals. As an example, for a (non-hybrid) dynamical system $\dot{y}(t) = g(y(t, x_0))$ with $y(0, x_0) = x_0 \in X_0$ and bounded state-space X, the reachability question above is formulated as

$$\exists^{X_0} x_0 \exists^{[0,T]} t.\, f(y(t, x_0)) \geq 0$$

Here, the unsafe set is $U = \{u \in X \mid f(u) \geq 0\}$.

This is an example of the more general *bounded* $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-sentence. Let $\mathcal{F}$ be a set of Type 2 computable functions[3] which contains at least the constant 0, unary negation, addition and the absolute value. It is also closed under bounded minimization and maximization [25]. Let $\vec{v} = (v_1, \ldots, v_n)$ be a vector of variables. An $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-term f is either a variable or a computable function of a term: $f := v | g(f(\vec{v}))$ for some $g \in \mathcal{F}$. Let $Q_i \in \{\forall, \exists\}$. A bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-sentence is

$$Q_1^{V_1} v_1 Q_2^{V_2} v_2 \ldots Q_n^{V_n} v_n \psi[f_i(\vec{v}) \geq 0, f_j(\vec{v}) > 0] \qquad (5)$$

The constraint sets $V_\ell \subset \mathbb{R}$ are bounded intervals and the f_i, f_j's are $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-terms, with $i \in \{1, \ldots, k\}$ and $j \in \{k+1, \ldots, m\}$. ψ is a first-order, quantifier-free formula (a 'matrix') on the predicates $f_i \geq 0, f_j > 0$. See [20, 21].

Bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$ sentences have a notion of robustness that comes from relaxing or tightening the constraints in the matrix ψ.

Definition 3.1 (δ-variants and δ-robustness [20])**.** *Let $\delta \in \mathbb{Q}_+ \cup \{0\}$ and S be a bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-sentence as in* (5)*. The* δ-weakening *of S is obtained by replacing each atom $f_i(\vec{v}) \geq 0$ by $f_i(\vec{v}) \geq -\delta$ and $f_i(\vec{v}) > 0$ by $f_i(\vec{v}) > -\delta$:*

$$S^{-\delta} = Q_1^{V_1} v_1 Q_2^{V_2} v_2 \ldots Q_n^{V_n} v_n \psi[f_i(\vec{v}) \geq -\delta, f_j(\vec{v}) > -\delta]$$

The δ-strenghtening *of S is analogously defined:*

$$S^{+\delta} = Q_1^{V_1} v_1 Q_2^{V_2} v_2 \ldots Q_n^{V_n} v_n \psi[f_i(\vec{v}) \geq \delta, f_j(\vec{v}) > \delta]$$

We say S is robust to δ-weakening *if $S^{-\delta} \implies S$, and is* robust to δ-strenghtening *if $S \implies S^{+\delta}$.*

Because $S^{-\delta} \implies S$ is equivalent to $\neg S^{-\delta} \vee S$, if a sentence is robust to δ-weakening, this means that either it is true, or it is 'robustly' false, so that even a δ-relaxation of it won't make it true. Similarly for δ-strengthening. We refer to these notions as 'δ-robustness'.

3.1 Bounding δ for trajectories

We will now define a natural translation from an MTL formula φ to a bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-formula $sen(\varphi)$. The translation allows us to connect the robustness degree of φ to the δ-robustness of $sen(\varphi)$.

In the following definition, given a boolean operator $\boxdot \in \{\vee, \wedge\}$ and two bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$ formulas S_1, S_2, we construct $S_1 \boxdot S_2$ in prenex normal form (i.e. all the quantifiers are pushed to the left and only a quantifier-free matrix ψ is used. New variable names are used to avoid conflicting quantifications on the same variable).

Definition 3.2. *Define the map*

$$sen : MTL^+ \times \mathcal{P}(Y)^{AP} \to (X^{\mathbb{T}} \times \mathbb{T} \to \mathcal{L}_{\mathbb{R}_\mathcal{F}} \textit{formulas})$$

$$sen(\varphi, \mathcal{O})(\mathbf{x}, t) = \mathcal{L}_{\mathbb{R}_\mathcal{F}} \textit{ formula}$$

[3]Intuitively, a function g is Type 2 computable if $g(x)$ can be computed with arbitrary precision given an arbitrarily precise approximation of x. See [25] or [20].

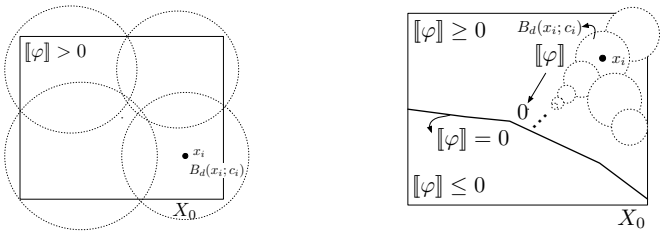

Figure 2: Robust testing terminates on robustly correct systems (left), but not necessarily on non-robust systems where the robustness vanishes gradually (right).

$$sen(\top,\mathcal{O})(\mathbf{x},t) = 0 \geq 0$$
$$sen(p,\mathcal{O})(\mathbf{x},t) = \begin{cases} \mathbf{Dist}_d(x_t,\mathcal{O}(p))\geq 0 & ,\mathcal{O}(p)\ closed \\ \mathbf{Dist}_d(x_t,\mathcal{O}(p))>0 & ,\mathcal{O}(p)\ open \end{cases}$$
$$sen(\neg p,\mathcal{O})(\mathbf{x},t), \begin{cases} -\mathbf{Dist}_d(x_t,\mathcal{O}(p))\geq 0, & \mathcal{O}(p)\ open \\ -\mathbf{Dist}_d(x_t,\mathcal{O}(p))>0, & \mathcal{O}(p)\ closed \end{cases}$$
$$sen(\varphi_1 \vee \varphi_2,\mathcal{O})(\mathbf{x},t) = sen(\varphi_1,\mathcal{O})(\mathbf{x},t) \vee sen(\varphi_2,\mathcal{O})(\mathbf{x},t)$$
$$sen(\varphi_1 \wedge \varphi_2,\mathcal{O})(\mathbf{x},t) = sen(\varphi_1,\mathcal{O})(\mathbf{x},t) \wedge sen(\varphi_2,\mathcal{O})(\mathbf{x},t)$$
$$sen(\varphi_1 \mathcal{U}_I \varphi_2,\mathcal{O})(\mathbf{x},t) = \exists^{t+_{\mathbb{T}}I} t' \forall^{(t,t')} t'' \left(sen(\varphi_1,\mathcal{O})(x,t'') \wedge sen(\varphi_2,\mathcal{O})(x,t')\right)$$
$$sen(\varphi_1 \mathcal{R}_I \varphi_2,\mathcal{O})(\mathbf{x},t) = \forall^{t+_{\mathbb{T}}I} t' \exists^{(t,t')} t'' \left(sen(\varphi_1,\mathcal{O})(x,t'') \vee sen(\varphi_2,\mathcal{O})(x,t')\right)$$

where (recall) $t +_{\mathbb{T}} I := \mathbb{T} \cap (t + I)$ *and all time intervals* I *are interpreted as* $I \cap \mathbb{T}$.

E.g. if $\varphi = \Box_{[0,3]}\Diamond_{[1,2]} x > 5$ then $sen(\varphi,\mathcal{O})(\mathbf{x},0) = \forall^{[0,3]}t' \exists^{t'+[1,2]} t''\ x_{t''} > 5$.

Lemma 3.1. *Consider a bounded-time MTL$^+$ formula* φ *and a signal* $\mathbf{x}$. *If every set* $\mathcal{O}(p)$, $p \in AP$, *is given by*

$$\mathcal{O}(p) = \{(y_1,\ldots,y_n) \in \mathbb{R}^n \mid y_i \in [u_i, v_i], 1 \leq i \leq n\}$$

where u_i, v_i *are* $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$*-terms that do not involve* y_i, *then it holds that* $sen(\varphi,\mathcal{O})(\mathbf{x},t)$ *is a bounded* $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$ *formula.*

A simple special case of Lemma. 3.1 is when each $\mathcal{O}(p)$ is a box with constant endpoints, e.g. $\mathcal{O}(p) = \{(y_1,y_2) \in \mathbb{R}^2 \mid 1 \leq y_1 \leq 2, -3 \leq y_2 \leq -0.5\}$.

The following lemma about the boolean truth value of φ and its $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$ translation is easily established by induction on the structure of φ:

Lemma 3.2. *Consider the MTL$^+$ formula* φ, *the signal* $\mathbf{x} \in X^{\mathbb{T}}$, *the observation map* $\mathcal{O}$ *and* $t \in \mathbb{T}$. *Then*

$$(\mathbf{x},t) \models_\mathcal{O} \varphi \Leftrightarrow sen(\varphi,\mathcal{O})(\mathbf{x},t)\ \textit{is True} \tag{6}$$

The next lemma connects the robustness degree of φ and the robustness to δ-weakening/strengthening of $sen(\varphi)$.

Lemma 3.3. *Let* $r = \mathbf{Dist}_\rho(\mathbf{x},\mathcal{L}_t(\varphi))$. *Under the hypothesis that* $r \neq 0$, *it holds that for any rational* $0 \leq \delta < |r|$, $sen(\varphi,\mathcal{O})(\mathbf{x},t)$ *is both robust to* δ*-strengthening and robust to* δ*-weakening.*

If $r = 0$ is allowed, then φ merely implies $r \geq 0$, which doesn't leave enough 'room' for any δ-strengthening. Note that since $|[\![\varphi,\mathcal{O}]\!](\mathbf{x},t)| \leq |\mathbf{Dist}_\rho(\mathbf{x},\mathcal{L}_t(\varphi))|$, the result above holds also for all $\delta < |[\![\varphi,\mathcal{O}]\!](\mathbf{x},t)|$.

3.2 Bounding δ for systems

We are now ready to lift the results of Section 3.1 to systems. We connect the minimum robustness of a *system* relative to an MTL spec to the δ-robustness of the corresponding $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-sentence. Let $\mathcal{H}$ be an ODE system. A *trajectory* of the system is a solution to its dynamical equations from some initial point. We will mostly be concerned with $\mathcal{L}_0$, the set of $\mathcal{H}$ trajectories with initial point chosen from the bounded set X_0 and of duration $T > 0$: $\mathcal{L}_0 = \{\mathbf{y}_{x_0} \mid x_0 \in X_0, \sup \mathrm{dom} y = T\}$. All ODE solution functions are assumed to be in $\mathcal{F}$.

Let $\varphi \in \mathrm{MTL}^+$. With abuse of notation, define the robustness degree and estimate of a system w.r.t. an MTL formula to be, respectively:

$$\begin{aligned} \mathbf{Dist}_\rho(\mathcal{L}_0,\mathcal{L}_t(\varphi)) &= \inf\{\mathbf{Dist}_\rho(\mathbf{x},\mathcal{L}_t(\varphi)) \mid \mathbf{x} \in \mathcal{L}_0\} \\ [\![\varphi,\mathcal{O}]\!](\mathcal{L}_0,t) &= \inf\{[\![\varphi,\mathcal{O}]\!](\mathbf{x},t) \mid \mathbf{x} \in \mathcal{L}_0\} \end{aligned} \tag{7}$$

When any of the quantities $\mathcal{O}, t, \mathcal{L}_0$ are clear from the context we may drop them from the notation. Define

$$ssen : MTL^+ \times \mathcal{P}(Y)^{AP} \to (\mathcal{P}(Y)^{AP} \times \mathbb{T} \to \text{bounded } \mathcal{L}_{\mathbb{R}_\mathcal{F}})$$

by

$$ssen(\varphi,\mathcal{O})(X_0,t) = \forall^{X_0} x_0 . sen(\varphi,\mathcal{O})(\mathbf{y}_{x_0},t) \tag{8}$$

E.g. if $\varphi = \Box_{[0,3]}\Diamond_{[1,2]} x > 5$ and $X_0 = [-2,-1]$, then

$$ssen(\varphi,\mathcal{O})(X_0,0) = \forall^{[-2,-1]}x_0\ \forall^{[0,3]}t' \exists^{t'+[1,2]} t''\ y(t'',x_0) > 5$$

For ease of reference later, we will define two flavors of δ-complete decision procedures [20]:

Definition 3.3. *Let* B *be a set of* $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$ *formulas and* $\delta \in \mathbb{Q}_+$. *We say an algorithm* A^- *is an* optimistic δ-CDP *for* B *if for any formula* S *in* B, A^- *returns correctly one of these two answers:*

- S *is false*
- $S^{-\delta}$ *is true*

If the two cases overlap, either one is returned. We say an algorithm A^+ *is a* pessimistic δ-CDP *for* B *if for any* $S \in \mathsf{B}$, A^+ *returns correctly one of these two answers:*

- S *is true*
- $S^{+\delta}$ *is false*

If the two cases overlap, either one is returned.

Informally, if A^- returns δ-true, this means that the sentence $ssen(\varphi)$ may be false, but a small (δ-sized) relaxation of it makes it true. The main result of this section follows.

Theorem 3.1. *Consider the MTL$^+$ formula* φ, *the ODE system* $\mathcal{H}$ *with behavior* $\mathcal{L}_0$, *and the observation map* $\mathcal{O}$. *Let* $r = \mathbf{Dist}_\rho(\mathcal{L}_0,\mathcal{L}_t(\varphi))$. *Then it holds that:*

Table 1: Summary of Thm. 3.1

$A^+\rightarrow$ / $A^-\downarrow$	True	$(ssen(\varphi))^{+\delta}$ False
False	- - - -	$r \leq 0$
$(ssen(\varphi))^{-\delta}$ True	$r \geq 0$	$-\delta \leq r \leq \delta$

1. *If $r \neq 0$, then $ssen(\varphi)$ is robust to δ-strengthening and to δ-weakening for all $\delta < |r|$.*
2. *If a pessimistic A^+ returns $(ssen(\varphi))^{+\delta}$ False, then $r \leq \delta$.*
3. *If an optimistic A^- returns $(ssen(\varphi))^{-\delta}$ True, then $r \geq -\delta$.*

The last two results are summarized in Table 1. Each box indicates what we can infer about the system robustness, based on what is returned by A^+ and A^- when running on $ssen(\varphi)$.

This table asserts that *a δ-complete decision procedure can be used to bound the robustness degree.*

Thus we may consider that *any δ-CDP is actually a procedure for computing the robustness degree r of the system: it halts once it establishes either that $r \leq \delta$ (for a pessimistic procedure) or that $r \geq -\delta$ (for an optimistic procedure).*

3.3 Robustness-Guided Verification

There is a number of ways in which Thm. 3.1 can be exploited. The basic idea is that simulation provides system trajectories whose (MTL) robustness values are easily evaluated. These robustness values provide an upper bound on the δ with which the sentence $ssen(\varphi)$ is δ-robust. Therefore we may use them to guide a δ-CDP, either by suggesting choices of δ, areas of X_0 to be explored or ignored, or simulation times at which simulation gives way to verification. We present one such application here: we use robustness-guided falsification to accelerate a δ-CDP. The δ-decision problems for $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-sentences with ODEs are PSPACE-complete [20]. The practical runtimes of current tools can be exhorbitant (e.g., see the benhmarks for [27] for an idea of the runtimes), and they are sensitive to how the problem is encoded. Scaling these tools is therefore an important challenge. Due to current tool limiations, we restrict ourselves to safety specs in this section: $\varphi = \Box_{[0,T]}(\neg p)$. Let $\mathcal{H}$ be a system with initial set of conditions X_0. We ask a δ-CDP whether there exists a trajectory satisfying $\neg\varphi$:

$$\exists^{X_0} x_0 \exists^{X} x \exists^{[0,T]} t . x = y(t, x_0) \wedge \mathbf{Dist}_d(x_t, \mathcal{O}(p)) \geq 0 \quad (9)$$

Now if a trajectory $\mathbf{z}$ has robustness estimate $[\![\varphi, \mathcal{O}]\!](\mathbf{z}, t) = r > 0$, then $sen(\neg\varphi)(\mathbf{z})$ couldn't be δ-SAT for any $\delta < r$ by Lemma 3.3. But only δ-SAT initial conditions can be returned by an optimistic δ-CDP to indicate (δ)-unsafe behavior. Therefore, *we can use this robust trajectory to provide extra constraints to the δ-CDP, telling it to ignore such trajectories.*

Specifically, given a desired precision $\delta > 0$ and a trajectory $\mathbf{z}$ of robustness estimate $r = [\![\varphi, \mathcal{O}]\!](\mathbf{z}, t) > \delta$, we pass the following to the δ-CDP instead of (9):

$$\exists^{X_0} x_0 \; \exists^{X} x \; \exists^{[0,T]} t \; \exists^{X} x' \; \exists^{[0,T]} t' . \; x = y(t, x_0) \wedge x' = y(t', x_0)$$
$$\wedge \underbrace{(|z_{t'} - x'| \geq r - \delta)}_{\rho(\mathbf{z}, \mathbf{y}_{x_0}) \geq r - \delta} \wedge \mathbf{Dist}_d(x_t, \mathcal{O}(p)) \geq 0 \quad (10)$$

Table 2: Falsification-Guided Verification: dReach running on an Intel(R) Core i7(R) 2.2GHz CPU and 16 GB memory

Benchmark	No constraint	With added constraint
Insulin	6secs	3secs
Afib1	5mins 27secs	2mins 3secs
Afib2	17mins 37secs	14mins 3secs

The extra constraint constrains the solver to look for those trajectories that are *not* robust to δ-weakening, i.e. it eliminates from consideration trajectories that are robustly false relative to $sen(\neg\varphi)$, and so robustly true relative to $sen(\varphi)$. Section 6 of [3] gives the detailed theoretical justification for why this works, and addresses the need for rigorous simulation.

Computational savings. The computational savings from adding this constraint can be substantial. dReach [26] implements a δ-CDP by integrating Interval Constraint Propagation with ODE solving. It uses a prune-and-split approach, where the prune step shrinks the constraint intervals [21]. By adding constraints to the formula, we are increasing the amount of pruning that is performed, and thus *reducing* the sizes of the sets that have to be propagated backward through the ODE dynamics at every iteration. Backward propagation is the most expensive step of the procedure, thus reducing its runtime can save substantial runtime.

Sample results. We did an initial exploration of these ideas using S-TaLiRo [5] to compute robustness of trajectories and dReach [26] to perform δ-complete reachability analysis. The implementation is crude, and we haven't attempted to optimize the choice of trajectories $\mathbf{z}$. Our goal is to show achievable savings on some simple benchmarks. Future work will optimize the approach in several ways.

We first ran this on a 3-dimensional ODE model of insulin processing by the body presented in [14]. The result is in Table 2. We also ran this on 4-dimensional hybrid models of atrial fibrillation (see [18]). Table 2 shows two examples of the obtained results (afib1 and afib2): in both cases, the added constraint caused a meaningful reduction in runtime. Note that these results were obtained with just *one* additional constraint. In general, we can add several constraints, coming from different trajectories returned by falsification, further pruning the search space.

Discussion. While promising, the above results are not conclusive. In general, the runtime savings will depend on the interplay between the components of the δ-CDP, in particular, how the new constraint affects the heuristics used by the SAT solver. The above results were obtained by adding a self-transition to each mode of the hybrid system, to capture the event $|z_{t'} - x'| > r - \delta$ from Eq. (10). These extra transitions could negatively affect the runtime and the overall savings will depend on how much is saved by adding the constraint. Future work will explore these issues in greater depth and seek more direct ways to encode the new constraint.

3.3.1 *Difference with robust testing.*

This approach has advantages over robust testing (Section 2.2). First, robust testing requires finding an approximate bisimulation of the system, which may not be possible for nonlinear systems. Secondly, computing a ball $B(x_i; c_i)$ using the bisimulation requires a costly bilevel optimization,

whose solution may be very conservative depending on the particular bisimulation. The proposed approach also differs from simply removing B_i from X_0 and running dReach on $X \setminus B_i$. That's because the back-propagation step in dReach is more costly than forward simulation. By removing sets from X_0 we save runtime in forward propagation. By imposing an extra constraint we save runtime in backward propagation, achieving greater computational savings.

4. QUASI-SEMIDECISION PROCEDURES AND ROBUST TESTING

We now connect the robust semantics of MTL to quasi-semidecidability. This closes the loop on the relation between verification and testing by the means of relaxing decidability and robustifying testing. See Fig. 1 (B), (D).

Recall the Robust Testing algorithm Alg. 1. Robust Testing halts when it finds a falsifier to the MTL formula and returns False, or when X_0 has been covered by the balls $B_d(x_i; c_i) \equiv B_i$ and returns True. In both cases, the answer it returns is evidently correct. If neither of these things happens, then it will run forever.

Intuitively, robust testing might run forever on 'non-robust' instances of the problem: instances where the system has trajectories of vanishingly small positive robustness, leading to vanishingly small balls (Alg. 1). It might also run forever if the system can generate falsifying trajectories, since we have no deterministic guarantee, in general, that they will be sampled. This suggests that robust testing is a *quasi-semidecision procedure.*

Definition 4.1 (Quasi-(semi)decision procedure[16])**.** *A* quasi-semidecision procedure P *for some class* B *of formulas is an algorithm that returns True for any formula* S *in* B *which is True and robust, but might otherwise run forever.*

A quasi-decision procedure P *for* B *is an algorithm that terminates and returns a correct answer for any robust formula* S *in* B *(whether it's true of false), but might otherwise run forever.*

The notion of robustness used in Def. 4.1 refers to a distance function d_R *between formulas*, which we define for the class $\mathsf{B} = \mathcal{L}_{\mathbb{R}_\mathcal{F}}$. For a vector $x \in \mathbb{R}^n$, its norm is $|x| = \max\{|x_1|, |x_2|, \ldots, |x_n|\}$.

Definition 4.2 (Quasi-robustness[16])**.** *Two* $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$*-sentences* S *and* S' *are said to have* the same structure *iff one can be obtained from the other by only exchanging terms. (I.e., they have the same Boolean and quantification structure, same bounds on quantified variables, and the same predicate symbols).*

Define the distance function $d_R(S, S')$ *as follows: if* S *and* S' *have different structure then* $d_R(S, S') = \infty$*. Else let* $\{f_i\}$ *be the terms of* S *and* $\{f'_i\}$ *be the corresponding terms of* S'*. Their common domain* Ω_i *is given by the quantification of all variables. Then* $d_R(S, S') = \max_i \|f_i - f'_i\|_\infty$*,* $\|f_i - f'_i\|_\infty := \sup_{\vec{v}\in\Omega_i} |f_i(\vec{v}) - f'_i(\vec{v})|$*. A sentence* S *is* ε-quasi-robust *if for any sentence* S' *that satisfies* $d_R(S, S') < \varepsilon$*, both* S *and* S' *have the same truth value.*

4.1 Robust testing as a quasi-semidecision procedure

The notion of quasi-robustness presented in Def. 4.2 is related to the robust semantics of MTL, as established in the following Lemma.

Lemma 4.1 (Quasi-robustness implies MTL^+ robustness)**.** *Let* φ *be an* MTL^+ *formula and* $\varepsilon \in \mathbb{R}_+$*.*

$$ssen(\varphi, \mathcal{O})(X_0, t) \text{ is } \varepsilon\text{-quasi-robust and True} \implies [\![\varphi, \mathcal{O}]\!](\mathcal{L}_0, t) \geq \varepsilon$$

Lemma 4.1 is a one-sided result: it requires that $ssen(\varphi)$ be True. Even if $ssen(\varphi)$ is robustly false, this only implies that there exist trajectories that falsify the formula robustly, but says nothing about whether there exist trajectories that falsify it non-robustly. Thus we cannot bound $[\![\varphi, \mathcal{O}]\!](\mathcal{L}_0, \mathcal{L}_t(\varphi))$ from above away from zero.

When the robustness estimate of the system $[\![\varphi, \mathcal{O}]\!](\mathcal{L}_0, t)$ is positive, then X_0 can be covered with a finite number of balls by Robust Testing - see Fig. 2. Thus, from Lemma 4.1, it immediately follows that *Robust Testing is a quasi-semidecision procedure*: that is, it terminates and returns True if the system satisfies the spec and is ε-quasi-robust, but otherwise it might run forever. A similar theoretical result was proved for safety in [30].

We now generalize this result in two directions. First, we allow for 'small' sets of initial points that have zero robustness. Secondly, instead of requiring that the robustness estimate be positive, we only require that the robustness degree $\mathbf{Dist}_\rho(\mathcal{L}_0, t)$ be positive.

4.1.1 Almost-everywhere robust systems

For the first strengthening, we will need the sampler used in Line 3 of Alg. 1 to satisfy the following *coverage condition.*

(CC) *Let* $Z \subset X_0$ *have measure 0 in* $\mathbb{R}^n$*. Let* $w_k \geq 0$ *be the number of samples that belong to* Z *in the first* k *samples* $x_0, x_1, \ldots, x_k$*. Then*

$$\lim_{k\to\infty} \frac{w_k}{k} = 0$$

We call this a coverage criterion because it implies that the sampler will never get stuck in 'small' sets (of measure 0). For every w_k samples in a small set Z, the sampler will produce, in the long run, significantly more samples outside it. Any stochastic sampler, like Hit-and-Run, obeys (CC), since sets of measure 0 have probability 0. A deterministic sampler would have to be extremely unlucky to violate (CC). Note however that in higher dimensions, getting good coverage becomes harder. See [9] for a promising approach.

We now give the main result of this sub-section. It states that Robust Testing will terminate for a system even if it exhibits trajectories of 0 robustness, as long as there are only 'few' of them. We call this an *almost-everywhere robust system.*

Theorem 4.1. *Consider a hybrid system* $\mathcal{H}$ *with initial set* X_0*. Let* $R_0 := \{x_0 \in X_0 \mid [\![\varphi, \mathcal{O}]\!](\mathbf{y}_{x_0}, t) = 0\}$ *be the set of initial points of robustness 0, and set* $R_1 = X_0 \setminus R_0$*. If* R_0 *has measure 0 in* $\mathbb{R}^n$*,* $\inf_{x_0\in R_1} [\![\varphi, \mathcal{O}]\!](\mathbf{y}_{x_0}, t) := r_* > 0$*, and the sampling strategy obeys the coverage criterion* (CC)*, then Robust Testing will terminate and return True.*

It is possible to bound the measure of R_0 rather than compute it exactly. For instance, if a region of X_0 is enclosed by sequences of $B(x_i; c_i)$ of vanishing radius, this can conservatively upper-bound the size of the set of zero robustness.

4.1.2 Robustness Degree Testing

The robustness estimate, which was used in Thm. 4.1, is a *lower* bound on the true robustness degree $\mathbf{Dist}_\rho(\mathcal{L}_0, \mathcal{L}_t(\varphi))$. Thus there may be systems that are indeed robust, in the sense that $\mathbf{Dist}_\rho(\mathcal{L}_0, \mathcal{L}_t(\varphi)) > 0$, but Robust Testing will not terminate for them because it looks at the system's robustness estimate, which could be 0. As an example of this phenomenon, consider the identically zero signal $\mathbf{x} \equiv 0$ and $\varphi = (x \geq 0 \vee x < 0)$, for which $\mathbf{Dist}_\rho(\mathbf{x}, 0) = \infty$ but $[\![\varphi, \mathcal{O}]\!](\mathbf{x}, 0) = 0$.

In this section, we give a technical condition under which this does not happen. Specifically, under this condition, a positive lower (negative upper) bound on the robustness degree implies a positive lower (negative upper) bound on the robustness estimate. As will be seen, the condition we give is not easy to check - we cannot presently think of an algorithm that might test it for a given system. Nonetheless, the theoretical interest of the link between robust testing and quasi-decidability is in its potential to suggest new ways to bridge the gap between verification and testing, *and to draw the limits of what can be done with robust testing and similar robustness-guided algorithms.* This is not affected by the hardness of this condition.

Lemma 4.2. *Consider a discrete-time system $\mathcal{H}$ with trajectory space $\mathcal{L}_0$, and let $t \in \mathbb{T}$ be a time instant. Consider the bounded-time MTL formula φ, and let S_φ be the set of all its sub-formulas. Given $L \subset X^{\mathbb{T}}$, $\overline{L} := X^{\mathbb{T}} \setminus L$. Define the set $\mathcal{D}_\varphi \subset X^{\mathbb{T}}$ as follows.*

- *For every $\psi_1 \vee \psi_2 \in \mathsf{S}_\varphi$ and $\psi_1 \wedge \psi_2 \in \mathsf{S}_\varphi$, $\mathcal{D}_\varphi$ contains $\mathcal{L}_t(\psi_i)$ and $\overline{\mathcal{L}_t(\psi_i)}$, $i = 1, 2$.*
- *for every $\psi_1 \mathcal{U}_I \psi_2 \in \mathsf{S}_\varphi$ and every $t' \in t + I$, $t'' \in (t, t')$, $\mathcal{D}_\varphi$ contains $\mathcal{L}_{t'}(\psi_2)$, $\mathcal{L}_{t''}(\psi_1)$, $\cap_{t'' \in (t,t')} \mathcal{L}_{t''}(\psi_1)$, and $\mathcal{L}_{t'}(\psi_2) \bigcap \cap_{t'' \in (t,t')} \mathcal{L}_{t''}(\psi_1)$.*

If

$$\text{for all } \mathbf{x} \in \mathcal{L}_0,\ \mathbf{x} \notin \bigcap_{A \in \mathcal{D}_\varphi} \mathrm{cl}(A) \tag{11}$$

then $\mathbf{Dist}_\rho(\mathcal{L}_0, t) > 0 \implies [\![\varphi, \mathcal{O}]\!](\mathcal{L}_0, t) > 0$.

The example we gave at the outset violates the Lemma's conditions since $\mathbf{x}$ is in the intersection of the closures $\mathrm{cl}(\mathcal{L}_0(x \geq 0))$ and $\mathrm{cl}(\mathcal{L}_0(x < 0))$. In fact, the Lemma establishes that *this is the prototypical example of this phenomenon:* namely, the only cases where we get $[\![\varphi, \mathcal{O}]\!](\mathbf{x}) = 0 < \mathbf{Dist}_\rho(\mathbf{x}, 0)$ is when $\mathbf{x}$ lives on the boundaries of all the sets $A \in \mathcal{D}_\varphi$.

Combining Lemma 4.2 and Thm. 4.1, we immediately get:

Theorem 4.2. *Let $\mathcal{H}$, R_0 and R_1 be as in Thm. 4.1. Assume the hypotheses of Lemma 4.2. If R_0 has measure 0 in $\mathbb{R}^n$, $\inf_{x_0 \in R_1} \mathbf{Dist}_\rho(\mathbf{y}_{x_0}, \mathcal{L}_t(\varphi)) > 0$, and the sampling strategy obeys the coverage criterion* (CC)*, then Robust Testing will terminate and return True.*

5. CONCLUSION

By exploring the connections between relaxed decidability and the robust semantics of MTL formulas, we improve near-exhausitve verification methods by the results of robust simulations, and delimit what is possible with robustness-guided testing. Future work will integrate robust simulations into a δ-complete decidability tool, to examine the achievable runtime savings on benchamrks of various sizes. In particular, we will explore the efficiency of different encodings of the additional constraints obtained from robust simulation. We will also pursue generalizations of Robust Testing in which a bisimulation is not needed, to tackle a broader range of systems which contain a mixture of robustly correct and robustly incorrect behavior.

6. PROOFS

6.1 Proofs and details of Section 3.1

6.1.1 Proof of Lemma. 3.1

For $sen(\varphi, \mathcal{O})(\mathbf{x}, t)$ to be a valid bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$ formula, the distance functions $\mathbf{dist}_d(\cdot, \mathcal{O}(p))$ and $\mathbf{dist}_d(\cdot, X \setminus \mathcal{O}(p))$ must be in $\mathcal{F}$ for a given $\mathcal{O}(p)$, and all quantifications must be bounded. Write $y = (y_1, \ldots, y_n)$. We need the fact that $d(x_t, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$ is computable, being the composition of computable functions. (All the facts we invoke in this proof can be looked up in [32]). Write

$$\mathbf{dist}_d(x_t, \mathcal{O}(p)) = \min\{d(x_t, y) \mid y_i \in [u_i, v_i], 1 \leq i \leq n\}$$

Bounded minimization preserves computability, so $\mathbf{dist}_d(x_t, \mathcal{O}(p))$ is computable. Now recall that $X = [a_1, b_1] \times \ldots \times [a_n, b_n]$ where a_i, b_i are constants. Then $\mathbf{dist}_d(x_t, X \setminus \mathcal{O}(p))$ evaluates to

$$\begin{aligned} & \min\{d(x_t, y) \mid y \in \mathrm{cl}(X \setminus \mathcal{O}(p))\} \\ = \ & \min\{d(x_t, y) \mid y_i \in [a_i, u_i] \vee y_i \in [v_i, b_i], 1 \leq i \leq n\} \\ = \ & \min\{d(x_t, y) \mid y_i \in [a_i, u_i], 1 \leq i \leq n\} \\ & \qquad \sqcap \min\{d(x_t, y) \mid y_i \in [v_i, b_i], 1 \leq i \leq n\} \end{aligned}$$

n-ary minimization also preserves computability, and this concludes the proof.

6.1.2 Proof of lemma. 3.2

($\implies$) Assume that $(\mathbf{x}, t) \models_\mathcal{O} \varphi$. Proof is by structural induction on φ.

$\varphi = p \in AP$. $(x, t) \models_\mathcal{O} p \Leftrightarrow x_t \in \mathcal{O}(p) \Leftrightarrow \mathbf{Dist}_\rho(x_t, \mathcal{O}(p)) \geq 0$ (assume the set is closed) $\Leftrightarrow sen(p, \mathcal{O})(x, t)$ is True. Similarly if $\mathcal{O}(p)$ is open.

$\varphi = \neg p$ $(x, t) \not\models_\mathcal{O} p \Leftrightarrow x_t \notin \mathcal{O}(p) \Leftrightarrow \mathbf{Dist}_\rho(x_t, \mathcal{O}(p)) < 0$ (assume the set is closed) $\Leftrightarrow -\mathbf{Dist}_\rho(x_t, \mathcal{O}(p)) > 0 \Leftrightarrow sen(\neg p, \mathcal{O})(x, t)$ is True. Similarly if $\mathcal{O}(p)$ is open.

$\varphi = \varphi_1 \vee \varphi_2$. $(x, t) \models_\mathcal{O} \varphi \Leftrightarrow sen(\varphi_1, \mathcal{O})(x, t)$ True or $sen(\varphi_2, \mathcal{O})(x, t)$ True by the induction hypothesis. This is equivalent to $sen(\varphi_1, \mathcal{O})(x, t) \vee sen(\varphi_2, \mathcal{O})(x, t)$ True by definition of the latter.

The cases for $\wedge, \mathcal{U}_I$ and $\mathcal{R}_I$ are similar. The converse is obtained in a similar manner.

6.1.3 Proof of lemma. 3.3

To establish this lemma, we need to avoid the $-\mathbf{Dist}_d$ terms in $sen(\varphi)$. Thus we need to transform our MTL$^+$ formulas to equivalent formulas that don't have negation in them. This is done by using the construction in [11, Section 3.1]: introduce an *extended* set of atomic propositions $AP^e = AP \cup AP'$ where $AP' = \{p' \mid p \in AP\}$. Then given an MTL$^+$ formula φ, map it to $\mathbf{pos}(\varphi)$ which replaces every occurence of $\neg p$ by the corresponding p'. This gets rid of the negations. To keep the semantics, we extend the observation map as well: define $\mathcal{O}^e : AP^e \to \mathcal{P}(X)$ by $\mathcal{O}^e(p) = \mathcal{O}(p)$ for every $p \in AP$ and $\mathcal{O}^e(p') = X \setminus \mathcal{O}(p)$ if $p' \in AP'$. By [11, Lemma 3.1.2], $\mathbf{pos}(\varphi)$ and φ have the same truth value.

In the rest of this proof we work implicitly with $\mathbf{pos}(\varphi)$ and $\mathcal{O}^e$. Given the map $\mathcal{O}$, define the observation map $\mathcal{O}_\delta$, which maps an atomic proposition p to a δ-contraction of $\mathcal{O}(p)$. I.e. $\mathcal{O}_\delta(p) = \{a \in \mathcal{O}(p) \mid \mathbf{dist}_d(a, X \setminus \mathcal{O}(p)) \geq \delta\}$. Write

$$\bar{\varphi} = sen(\varphi, \mathcal{O})(\mathbf{x}, t) = Q_1^{V_1} v_1 \dots Q_k^{V_k} v_k \psi[f_i(\vec{v}) \geq 0, f_j(\vec{v}) > 0]$$

Then $\bar{\varphi}^{+\delta}$ equals

$$\begin{aligned}
&Q_1^{I_1} t_1 \dots Q_n^{I_n} t_n \psi[f_i(\vec{t}) \geq \delta, f_j(\vec{t}) > \delta] \\
&= Q_1^{I_1} t_1 \dots Q_n^{I_n} t_n \psi[\mathbf{Dist}_d(x_{t_i}, \mathcal{O}(p)) \geq \delta, \mathbf{Dist}_d(x_{t_j}, \mathcal{O}(p)) > \delta] \\
&= Q_1^{I_1} t_1 \dots Q_n^{I_n} t_n \psi[\mathbf{Dist}_\rho(x_{t_i}, \mathcal{O}_\delta(p)) \geq 0, \mathbf{Dist}_\rho(x_{t_j}, \mathcal{O}_\delta(p)) > 0]
\end{aligned}$$

It can be seen that

$$\bar{\varphi}^{+\delta} = sen(\varphi, \mathcal{O}_\delta)(\mathbf{x}, t) \tag{12}$$

Now $\bar{\varphi} \implies \varphi$ (by lemma. 3.2) $\implies r > 0$ (by assumption, $r \neq 0$) $\implies (\mathbf{x}, t) \models_{\mathcal{O}_\delta} \varphi\ \forall \delta < r$ (by Lemma 6.1 below) $\implies sen(\varphi, \mathcal{O}_\delta)$ True for all $\delta < r$ by Lemma 3.2 $\implies \bar{\varphi}^{+\delta}$ True for all $\delta < r$ by (12). Thus $\bar{\varphi}$ is robust to δ-strengthening for all $\delta < r$. The second case (robust to δ-weakening) is similarly proved.

Lemma 6.1. $\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) = r > 0$ *implies that* $(\mathbf{x}, t) \models_{\mathcal{O}_\delta} \varphi$ *for all* $\delta < r$.

Proof. We argue by contradiction. Suppose that for some $\delta < r$, $(\mathbf{x}, t) \not\models_{\mathcal{O}_\delta} \varphi$. This implies that there exists a set of time instants, $\mathbb{T}' \subset \mathbb{T}$, at which the points of trajectory $\mathbf{x}$ switch from being inside some atomic set to not being in it, or vice versa. That is, at every $t \in \mathbb{T}'$, there exists $p_t \in AP$ s.t. $x_t \in \mathcal{O}(p_t)$ iff $x_t \notin \mathcal{O}_\delta(p_t)$. Otherwise, if no such time set $\mathbb{T}'$ exists, the truth value of the formula would not have changed.

Now note that for $t \in \mathbb{T}'$, and by definition of the contracted map $\mathcal{O}_\delta$, $|\mathbf{Dist}_d(x_t, \mathcal{O}(p_t))| \leq \delta$.

For every x_t, let $\mathrm{pr}(x_t, A)$ be the projection of x_t onto the set $A \subset X$. Thus it is the point of A that is nearest x_t. Now define the signal $\mathbf{y}$ as follows:

$$y_t = \begin{cases} x_t, & t \in \mathbb{T} \setminus \mathbb{T}' \\ \mathrm{pr}(x_t, X \setminus \mathcal{O}(p_t)), & t \in \mathbb{T}' \wedge x_t \in \mathcal{O}(p_t) \\ \mathrm{pr}(x_t, \mathcal{O}(p_t)), & t \in \mathbb{T}' \wedge x_t \notin \mathcal{O}(p_t) \end{cases}$$

By construction, $d(x_t, y_t) \leq \delta$ for all $t \in \mathbb{T}$, thus $\rho(\mathbf{x}, \mathbf{y}) = \sup_t d(x_t, y_t) \leq \delta < r$.

On the other hand, by definition of $\mathbb{T}'$, y does not satisfy φ at t: $(y, t) \not\models_{\mathcal{O}} \varphi$. Thus $\rho(\mathbf{x}, \mathbf{y}) \geq r$ - a contradiction. □

6.1.4 Proof of Thm. 3.1

Let $r = \mathbf{Dist}_\rho(\mathcal{L}_0, \mathcal{L}_t(\varphi))$.

1. Suppose that $r > 0$. Write $\bar{\varphi} = sen(\varphi)$ and $\mathcal{S}\bar{\varphi} = ssen(\varphi)$. Then, by lemma 3.3, for all $\mathbf{x} \in \mathcal{L}_0$, $\bar{\varphi}$ is robust to δ-strengthening for any $\delta < r$, i.e. $\forall^{X_0} x_0 . (\bar{\varphi})^{+\delta}$ is True. Since $(\mathcal{S}\bar{\varphi})^\delta = \forall^{X_0} x_0 (\bar{\varphi})^\delta$, this is equivalent to $(\mathcal{S}\bar{\varphi})^\delta$ is True $\Leftrightarrow \mathcal{S}\bar{\varphi}$ is robust to δ-strengthening for all $\delta < r$.

Similarly, if $r < 0$, then $(\mathcal{S}\bar{\varphi})^{-\delta}$ is False for all $\delta < |r|$.

2. Suppose A^- (which returns False or δ-True) found that $(\mathcal{S}\bar{\varphi})^{-\delta}$ is True.

Case 1: Some trajectory $\mathbf{x} \in \mathcal{L}_0$ violates the spec $\varphi \Leftrightarrow \mathcal{S}\bar{\varphi}$ is False. Then $(\mathcal{S}\bar{\varphi})^{-\delta'}$ is False for all $\delta' < |r|$ by part 1, so $\delta \geq |r| \Leftrightarrow -\delta \leq -|r| = r < 0$.

Case 2: All trajectories satisfy the spec. No conclusion about robustness (other than being non-negative) can be made.

Now suppose A^+ (which returns True or δ-False) has returned δ-False.

Case 1: All trajectories $\mathbf{x}$ satisfy the spec $\Leftrightarrow \mathcal{S}\bar{\varphi}$ is True. Then $(\mathcal{S}\bar{\varphi})^{+\delta'}$ is True for all $\delta' < r$ by part 1, thus $\delta \geq r$.

Case 2: $\mathcal{S}\bar{\varphi}$ is False. No conclusion about robustness (other than being non-positive) can be made.

6.1.5 Technical details for Falsification-guided verification

For the application presented in Section 3.3 to work, we will need the following result. It states that for each trajectory of positive robustness, there is always a trajectory at the edge of its tube of robustness which has robustness 0.

The fragment $MTL^+(AP, \wedge, \Box)$ only allows conjunction and Always. Safety is expressible in this fragment as $\Box_I p$.

Theorem 6.1 (Tube of robustness)**.** *Consider a bounded-time $MTL^+(AP, \wedge, \Box)$ formula φ. Let $\mathbf{x} \in X^{\mathbb{T}}$ be a system trajectory with positive robustness estimate $r = [\![\varphi, \mathcal{O}]\!](\mathbf{x}, t) > 0$, and let $\mathcal{T} = \{\mathbf{y} \in X^{\mathbb{T}} \mid \rho(\mathbf{x}, \mathbf{y}) < r\}$ be its robustness tube. Then cl($\mathcal{T}$) contains at least one signal $\mathbf{x}_*$ with robustness degree 0 and r-distant from $\mathbf{x}$: $\mathbf{Dist}_\rho(\mathbf{x}_*, \mathcal{L}_t(\varphi)) = 0, \rho(\mathbf{x}, \mathbf{x}_*) = r$.*

Proof. Because $\varphi \in MTL^+(AP, \wedge, \Box)$ and $[\![\varphi, \mathcal{O}]\!](\mathbf{x}, t) > 0$, it holds that the robustness estimate is actually an exact computation of the robustness degree [12, Prop. 19]:

$$\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) = r$$

Now $\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) = \mathbf{dist}_\rho(\mathbf{x}, X^{\mathbb{T}} \setminus \mathcal{L}_t(\varphi))$. The existence of a nearest point $\mathbf{x}_* \in \mathrm{cl}(X^{\mathbb{T}} \setminus \mathcal{L}_t(\varphi))$ to $\mathbf{x}$ follows from general results on the existence of nearest points in reflexive Banach spaces [6]. Because x^* is on the boundary of $\mathcal{L}_t(\varphi)$, it can be concluded that x^* has robustness 0 and has distance r to $\mathbf{x}$. □

6.1.6 Robustness intervals for a rigorous simulation

We end this section with a point of practical importance: In all situations where a simulation is called for, if we use a numerical integrator like ode45, we will obtain a trajectory with some nunmerical errors, so the robustness we compute using that trajectory may be incorrect. Instead, we should use a guaranteed integrator. A guaranteed integrator returns a time-stamped sequence of boxes $Z = (R_i, t_i)$. Each R_i is a subset of the state space X, $z_{t_i} \in R_i$, and for all $t \in [t_{i-1}, t_i)$, $z_t \in hull(R_{i-1}, R_i)$. Instead of a single robustness value for a trajectory $\mathbf{z}$, we now need to compute an interval enclosure of the robustness of all trajectories in Z. That is, we need to compute $[\underline{\rho}, \overline{\rho}]$ such that for any trajectory $\mathbf{y}$ enclosed by Z, it holds that $\underline{\rho} \leq [\![\varphi, \mathcal{O}]\!](\mathbf{y}, t) \leq \overline{\rho}$. We can calculate this interval by adopting the work in [7], and modifying it to allow for time-varying enclosures. We omit the details. Given the interval $[\underline{\rho}, \overline{\rho}]$, we use $\underline{\rho}$ in the additional constraint (10).

6.2 Proofs of Section 4

6.2.1 Proof of Lemma 4.1

The basic idea is that the disturbances allowed by quasi-robustness are a superset of those required for MTL robustness. Thus quasi-robustness implies MTL robustness.

As before, $Q_k \in \{\exists, \forall\}$ is a quantifier, v_k denotes a variable and V_k a real bounded interval. For any bounded $\mathcal{L}_{\mathbb{R}_\mathcal{F}}$-

sentence S

$$S = Q_1^{V_1} v_1 \dots Q_n^{V_n} v_n \psi[f_i(\vec{v}) \geq 0, f_j(\vec{v}) > 0] \quad (13)$$

$1 \leq i \leq m, m+1 \leq j \leq n$, we define its *$\varepsilon$-level perturbation* to be the sentence

$$S_\varepsilon = Q_1^{V_1} v_1 \dots Q_n^{V_n} v_n \psi[f_i(\vec{v}) - \varepsilon \geq 0, f_j(\vec{v}) - \varepsilon > 0] \quad (14)$$

Clearly, $d_R(S, S_\varepsilon) = \varepsilon$.

Now fix the MTL$^+$ formula φ, the scalar $\varepsilon > 0$, and let $\vec{v} = (x_0, t_1, \dots, t_n)$ be the vector of variables. The sentence $S = ssen(\varphi, \mathcal{O})(X_0, t)$ obtained from φ by Def. 8 takes the form

$$S = \mathbf{Q}^{\vec{V}} \vec{v}\, \psi[\mathbf{Dist}_d(x_{t_i}, \mathcal{O}(p_i)) \geq 0, \mathbf{Dist}_d(x_{t_j}, \mathcal{O}(p_j)) > 0]$$

By hypothesis, S is true. The proof now proceeds by structural induction on the MTL formula φ.

$\underline{\varphi = p.}$ Assume that $\mathcal{O}(p)$ is closed. Then the perturbation $S_{\varepsilon'} = \forall^{X_0} x_0 \mathbf{Dist}_d(x_t, \mathcal{O}(p)) - \varepsilon' \geq 0 \Leftrightarrow \forall^{X_0} x_0 [\![p, \mathcal{O}]\!](\mathbf{y}_{x_0}, t) \geq \varepsilon'$. Since $\varepsilon' < \varepsilon$ is arbitrary, it comes that $[\![p, \mathcal{O}]\!](X_0, t) \geq \varepsilon$, which is what we set out to prove. An identical reasoning applies when $\mathcal{O}(p)$ is open.

$\underline{\varphi = \neg p.}$ An argument identical to the previous case applies here.

$\underline{\varphi = \varphi_1 \wedge \varphi_2}$. Set $S_i = ssen(\varphi_i), i = 1, 2$. Let S', S'_i, $i = 1, 2$, be such that $d_R(S, S') < \varepsilon, d_R(S_i, S'_i) < \varepsilon$. It is easy to see that $d_R(S, S'_1 \wedge S'_2) < \varepsilon$, and is therefore true by ε-quasi-robustness of S. Thus S'_1 and S'_2 are both also true, and S_1 and S_2 are ε-quasi-robust. By the induction hypothesis, $[\![\varphi_i]\!] \geq \varepsilon$. Therefore $[\![\varphi]\!] = [\![\varphi_1]\!] \sqcap [\![\varphi_2]\!] \geq \varepsilon$.

$\underline{\varphi = \varphi_1 \vee \varphi_2}$. Given an arbitrary sentence S of the form (13), consider the ε-level perturbation S_ε given in Eq. (14). We show that if S_ε is true, this implies that S is in fact ε-quasi-robust. Indeed, take any $\varepsilon' \leq \varepsilon$, and any S' s.t. $d_R(S_\varepsilon, S) = \varepsilon'$. Let f'_k be the terms of S'. By definition, $\|f_k(\vec{v}) - f'_k(\vec{v})\| \leq \varepsilon'$ for all $1 \leq k \leq n$ and all $\vec{v} \in \Omega_k$, which implies $f'_k(\vec{v}) \geq f_k(\vec{v}) - \varepsilon' \geq f_k(\vec{v}) - \varepsilon$. But $f_k(\vec{v}) - \varepsilon$ is a term of S_ε. Recalling that S_ε is True and comparing the matrices of S_ε and S', it follows immediately that S' is true. Since ε' was chosen arbitrary less than ε, it follows that S is ε-quasi-robust.

Set $S_i = ssen(\varphi_i), i = 1, 2$. Now let $\bar{\varepsilon}$ be the least upper bound of all ε' such that $S_{1,\varepsilon'}$ is true. By the above result applied to S_1, this implies that S_1 is ν-quasi-robust for all $0 \leq \nu < \bar{\varepsilon}$, and by Induction Hypothesis (I.H.), $[\![\varphi_1]\!] \geq \nu$ for all $\nu < \bar{\varepsilon}$, therefore$[\![\varphi_1]\!] \geq \bar{\varepsilon}$. If $\bar{\varepsilon} = \varepsilon$, then we may conclude that $[\![\varphi]\!] = [\![\varphi_1]\!] \sqcup [\![\varphi_2]\!] \geq \varepsilon$ and we're done. Otherwise for all ν s.t. $\bar{\varepsilon} \leq \nu < \varepsilon$, $S_{2,\nu}$ is true because $S_\nu = S_{1,\nu} \vee S_{2,\nu}$ is true. This implies that S_2 is ν-quasi-robust for all $\bar{\varepsilon} \leq \nu < \varepsilon$ and by the I.H., $[\![\varphi_2]\!] \geq \nu$, therefore $[\![\varphi_2]\!] \geq \varepsilon$. Then $[\![\varphi]\!] \geq \varepsilon$ and we're done.

$\underline{\varphi = \varphi_1 \mathcal{U} \varphi_2 \text{ and } \varphi = \varphi_1 \mathcal{R} \varphi_2}$. The above cases can be combined to yield a proof for both the Until and Release operators.

6.2.2 Proof of Thm. 4.1

In this proof, the word 'robustness' means robustness estimate. Recall that d is the distance function on the state space, and that $\rho(\mathbf{x}, \mathbf{y}) = \sup_t d(x_t, y_t)$.

Every trajectory $\mathbf{x}$ that starts in $R_1 := X_0 \setminus R_0$ has robustness at least $r_* > 0$. The bisimulation function V has the property that, for any two points $x_0, y_0 \in X_0$, if $V(x_0, y_0) < r_*$, then $d(x_t, y_t) \leq V(x_t, y_t) \leq r_*$ for all t in $\mathbb{T}$. Therefore we can choose $c_i = r_* > 0$ for all x_i in R_1. Since the new samples are always chosen outside the already-covered area, it is possible to cover X_0 with a finite number of balls $B_d(x_i; c_i) \equiv B_i$. Let K be the maximum number of balls needed to cover X_0, where the maximization is over all possible choices of ball sequences $B_0, B_1, B_2, \dots$. Then $K < \infty$.

Because the sampler obeys the coverage criterion (CC), there exists an integer k_* such that the number of samples in R_1 exceeds K: $|\{x_i, i = 0, \dots, k_*\} \cap R_1| \geq K$. Thus after at most k_* samples, X_0 has been fully covered by balls, and returns true.

6.2.3 Proof of Lemma 4.2

We first sketch the proof, then give the details. Let (M, d) be a metric space, $Y \subset M$ be bounded, and let $\mathcal{Y}$ be a finite family of bounded subsets of M. Then the following holds: for every $x \in Y$ there exists a constant $m_x \geq 1$ s.t. for any two sets $A, B \in \mathcal{Y}$ it holds that $\mathbf{dist}_d(x, A \cap B) \leq m_x(\mathbf{dist}_d(x, A) \sqcup \mathbf{dist}_d(x, B))$.

By applying this to the metric space $(X^{\mathbb{T}}, \rho)$ and the subset of signals $\mathcal{Y} = \mathcal{L}_0$, we show that for any $\mathbf{x} \in \mathcal{L}_0$ and MTL formula φ it holds that

$$\frac{1}{m_{\mathbf{x}}^{n_\varphi}} |\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))| \leq |[\![\varphi, \mathcal{O}]\!](\mathbf{x}, t)| \leq |\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))|$$

where $n_\varphi \in \mathbb{N}$ is a positive integer that depends on the structure of φ and the endpoints of its temporal intervals.

The conclusion follows: if $\inf_{\mathbf{x} \in \mathcal{L}_0} |\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))| > 0$ then $\inf_{\mathbf{x} \in \mathcal{L}_0} |[\![\varphi, \mathcal{O}]\!](\mathbf{x}, t)| > 0$. The technical condition of this lemma is required to eliminate the possibility that m_x might be infinite.

We now give the detailed proof. Let (M, d) be a metric space, $Y \subset M$ be bounded, and let $\mathcal{Y}$ be a finite family of bounded subsets of M.

Lemma 6.2. *For all $x \in Y$, there exists a real number $m_x \geq 1$ such that for all $A, B \in \mathcal{Y}$,*

$$\mathbf{dist}_d(x, A \cap B) \leq m_x(\mathbf{dist}_d(x, A) \sqcup \mathbf{dist}_d(x, B))$$

Proof. Given two sets $A, B \in \mathcal{Y}$ and a point $x \in M$, let $\alpha(x, A, B) = \mathbf{dist}_d(x, A) \sqcup \mathbf{dist}_d(x, B)$. Define

$$m_x = \max\{\frac{\mathbf{dist}_d(x, A \cap B)}{\alpha(x, A, B)} : A, B \in \mathcal{Y} \text{ s.t. } \alpha(x, A, B) \neq 0\}$$

$\underline{\text{Case 1: } x \in A \cap B}$. Then $\mathbf{dist}_d(x, A \cap B) \leq \alpha(x, A, B) = 0$ so any m_x will do.

$\underline{\text{Case 2: } x \in A \setminus B}$. Then $\mathbf{dist}_d(x, A \cap B)/\alpha(x, A, B) \leq m_x$ by definition of m_x.

The remaining cases ($x \in B \setminus A$ and $x \notin A \cup B$) also follow immediately from the definition of m_x. □

By induction, this implies the following lemma:

Lemma 6.3. *For any $x \in Y$, there exists a constant m_x s.t. for any $n \geq 2$ subsets $A_1, A_2, \dots, A_n$ in $\mathcal{Y}$, $\mathbf{dist}_d(x, \cap_{1 \leq i \leq n} A_i) \leq m_x^n \sqcup_{1 \leq i \leq n} \mathbf{dist}_d(x, A_i)$*

We apply the preceding two lemmas to the metric space $(X^{\mathbb{T}}, \rho)$, the family $\mathcal{Y} = \mathcal{D}_\varphi$ of bounded subsets of $X^{\mathbb{T}}$ and the bounded set $Y = \mathcal{L}_0$.

Lemma 6.4. *For all $\mathbf{x} \in \mathcal{L}_0$, the following implications hold:*

$$(\mathbf{x},t) \models \varphi \implies \mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) \leq m_{\mathbf{x}}^{n_\varphi} [\![\varphi, \mathcal{O}]\!](\mathbf{x},t)$$

$$(\mathbf{x},t) \not\models \varphi \implies m_{\mathbf{x}}^{n_\varphi} [\![\varphi, \mathcal{O}]\!](\mathbf{x},t) \leq -\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))$$

where $n_\varphi = \#$boolean operators in φ + $\#$Until operators + $\sum |I_r|$. Here, I_r is the right end-point of interval I, and the sum is over all Until operators in the formula.

Proof. By structural induction on φ. Firs, note that the metric space $(X^{\mathbb{T}}, \rho)$ and the set $\mathcal{D}_\varphi$ satisfy the hypotheses of Lemma 6.2.

$\underline{\varphi = p}$. Here $n_\varphi = 0$. In the SAT case (i.e., $\mathbf{x} \models \varphi$), $\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) = [\![\varphi, \mathcal{O}]\!]$, while in the UNSAT case (i.e. $\mathbf{x} \not\models \varphi$), $-\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) = [\![\varphi, \mathcal{O}]\!]$.

$\underline{\varphi = \neg\psi}$. In the SAT case, $n_\varphi = n_\psi + 1$ and

$$\begin{aligned}
[\![\neg\psi]\!] &= -[\![\psi]\!] \geq (1/m_{\mathbf{x}}^{n_\psi})\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\psi)) \text{ (by the I.H.)} \\
&\geq (1/m_{\mathbf{x}}^{n_\psi})\mathbf{depth}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\psi)}) \\
&= (1/m_{\mathbf{x}}^{n_\psi})\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\neg\psi)) \\
&\geq (1/m_{\mathbf{x}}^{n_\varphi})\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))
\end{aligned}$$

In the UNSAT case,

$$\begin{aligned}
[\![\neg\psi]\!] &= -[\![\psi]\!] \leq -(1/m_{\mathbf{x}}^{n_\psi})\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\psi)) \text{ (by the I.H.)} \\
&= -(1/m_{\mathbf{x}}^{n_\psi})\mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\psi)}) \\
&= -(1/m_{\mathbf{x}}^{n_\psi})\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\neg\psi)) \\
&\leq -(1/m_{\mathbf{x}}^{n_\varphi})\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))
\end{aligned}$$

$\underline{\text{Case 3: } \varphi = \varphi_1 \vee \varphi_2}$: $n_\varphi = n_{\varphi_1} + n_{\varphi_2} + 1$. In the SAT case, if $\mathbf{x}$ satisfies both φ_1 and φ_2, then

$$\begin{aligned}
\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) &= \mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\varphi)}) = \mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\varphi_1)} \cap \overline{\mathcal{L}_t(\varphi_2)}) \\
&\leq m_{\mathbf{x}}(\mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\varphi_1)}) \sqcup \mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\varphi_2)})) \\
&\quad \text{by Claim 1 and the fact that } \overline{\mathcal{L}_t(\varphi_1)}, \overline{\mathcal{L}_t(\varphi_2)} \in \mathcal{D}_\varphi \\
&= m_{\mathbf{x}}(\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi_1)) \sqcup \mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi_2))) \\
&= m_{\mathbf{x}}(m_{\mathbf{x}}^{n_{\varphi_1}} [\![\varphi_1]\!] \sqcup m_{\mathbf{x}}^{n_{\varphi_2}} [\![\varphi_2]\!]) \text{ by I.H.} \\
&= m_{\mathbf{x}}^{1+n_{\varphi_1}} [\![\varphi_1]\!] \sqcup m_{\mathbf{x}}^{1+n_{\varphi_2}} [\![\varphi_2]\!] \\
&\leq m_{\mathbf{x}}^{1+n_{\varphi_1}+n_{\varphi_2}} ([\![\varphi_1]\!] \sqcup [\![\varphi_2]\!]) = m_{\mathbf{x}}^{n_\varphi} [\![\varphi]\!]
\end{aligned}$$

If $\mathbf{x}$ satisfies φ_1 but not φ_2, then $\mathbf{x} \in \overline{\mathcal{L}_t(\varphi_2)} \setminus \overline{\mathcal{L}_t(\varphi_1)}$, so that it holds that $\mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\varphi_1)} \cap \overline{\mathcal{L}_t(\varphi_2)}) = \mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\varphi_1)})$, therefore

$$\begin{aligned}
\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) &= \mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_t(\varphi_1)} \cap \overline{\mathcal{L}_t(\varphi_2)}) \\
&= \mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi_1)) \\
&\leq m_{\mathbf{x}}^{n_{\varphi_1}} [\![\varphi_1]\!] \leq m_{\mathbf{x}}^{n_\varphi} [\![\varphi]\!]
\end{aligned}$$

In the UNSAT case,

$$\mathbf{dist}_\rho(\mathcal{L}_t(\varphi)) = \mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi_1)) \sqcap \mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi_2))$$

Then

$$\begin{aligned}
-\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) &= -\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi_1)) \sqcup -\mathbf{dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi_2)) \\
&\geq m_{\mathbf{x}}^{n_{\varphi_1}} [\![\varphi_1]\!] \sqcup m_{\mathbf{x}}^{n_{\varphi_2}} [\![\varphi_2]\!] \text{ by I.H.} \\
&\geq m_{\mathbf{x}}^{n_\varphi} ([\![\varphi_1]\!] \sqcup [\![\varphi]\!]) = m_{\mathbf{x}}^{n_\varphi} [\![\varphi]\!]
\end{aligned}$$

$\underline{\text{Case 4: } \varphi = \varphi_1 \wedge \varphi_2}$. Similar arguments to the previous case apply here, and we skip the details.

$\underline{\text{Case 5: } \varphi = \varphi_1 \mathcal{U}_I \varphi_2}$. In the SAT case,

$$\begin{aligned}
\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) &= \mathbf{depth}_\rho(\mathbf{x}, \bigsqcup_{t' \in t+I} \mathcal{L}_{t'}\varphi_2 \bigsqcap \sqcap_{t'' \in (t,t')} \mathcal{L}_{t''}\varphi_2) \\
&= \mathbf{dist}_\rho(\mathbf{x}, \bigsqcap_{t' \in t+I} \overline{\mathcal{L}_{t'}\varphi_2 \bigsqcap \sqcap_{t'' \in (t,t')} \mathcal{L}_{t''}\varphi_2})
\end{aligned}$$

Recall that we are working in discrete-time so that the interval I is finite and has $|I|$ points in it. By noting that $\overline{\mathcal{L}_{t'}\varphi_2 \bigsqcap \sqcap_{t'' \in (t,t')} \mathcal{L}_{t''}\varphi_2}$ is in $\mathcal{D}_\varphi$, we can invoke Lemma 6.3 to continue

$$\begin{aligned}
&\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) \\
&\leq m_{\mathbf{x}}^{|I|} \bigsqcup_{t' \in t+I} \mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_{t'}\varphi_2 \bigsqcap \sqcap_{t'' \in (t,t')} \mathcal{L}_{t''}\varphi_1}) \\
&= m_{\mathbf{x}}^{|I|} \bigsqcup_{t' \in t+I} \mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_{t'}\varphi_2} \bigsqcup \overline{\sqcap_{t'' \in (t,t')} \mathcal{L}_{t''}\varphi_1}) \\
&= m_{\mathbf{x}}^{|I|} \bigsqcup_{t' \in t+I} \mathbf{dist}_\rho(\mathbf{x}, \overline{\mathcal{L}_{t'}\varphi_2}) \sqcap \mathbf{dist}_\rho(\mathbf{x}, \overline{\sqcap_{t'' \in (t,t')} \mathcal{L}_{t''}\varphi_1}) \\
&= m_{\mathbf{x}}^{|I|} \bigsqcup_{t' \in t+I} \underbrace{\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_{t'}\varphi_2) \sqcap \mathbf{depth}_\rho(\mathbf{x}, \sqcap_{t'' \in (t,t')} \mathcal{L}_{t''}\varphi_1)}_{B}
\end{aligned}$$

Let $P = \{t' \in t+I \mid (\mathbf{x},t') \models \varphi_2 \wedge \exists t'' \in (t,t') \text{ s.t. } (\mathbf{x},t'') \models \varphi_1\}$. This is the set of times that witness satisfaction of φ by $\mathbf{x}$.

Then $\bigsqcup_{t' \in t+I}(B) = \bigsqcup_{t' \in P} B$ since on $t + I \setminus P$, B is negative. So

$$\begin{aligned}
&\mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) \\
&\leq m_{\mathbf{x}}^{|I|} \bigsqcup_{t' \in P} \mathbf{depth}_\rho(\mathbf{x}, \mathcal{L}_{t'}\varphi_2) \bigsqcap \mathbf{depth}_\rho(\mathbf{x}, \sqcap_{t'' \in (t,t')} \mathcal{L}_{t''}\varphi_1) \\
&\leq m_{\mathbf{x}}^{|I|} \bigsqcup_{t' \in P} m_{\mathbf{x}}^{n_{\varphi_2}} [\![\varphi_2]\!](t') \bigsqcap \sqcap_{t''} m_{\mathbf{x}}^{n_{\varphi_1}} [\![\varphi_1]\!](t'')) \\
&\leq m_{\mathbf{x}}^{|I|} m_{\mathbf{x}}^{1+n_{\varphi_1}+n_{\varphi_2}} \bigsqcup_{t' \in P} [\![\varphi_2]\!](t') \bigsqcap \sqcap_{t''} [\![\varphi_1]\!](t'')) \\
&= m_{\mathbf{x}}^{n_\varphi} [\![\varphi]\!], n_\varphi = |I| + 1 + n_{\varphi_1} + n_{\varphi_2}
\end{aligned}$$

The UNSAT case is treated similarly and we skip the details. The only notable difference is that, in the UNSAT case, each Until operator adds $1 + I_r$ to the exponent n_φ.

Since $|I| \leq I_r$, each Until operator adds, at the most, $1+I_r$ to the exponent. □

Lemma 6.5. *Let $\mathcal{D}_\varphi$ be as defined in Lemma 4.2. If the system is robustly correct, $\inf\{\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi)) \mid \mathbf{x} \in \mathcal{L}_0\} = r > 0$, then $\inf\{[\![\varphi, \mathcal{O}]\!](\mathbf{x},t) \mid \mathbf{x} \in \mathcal{L}_0\} > 0$.*

Proof. Lemma 6.4 implies that for all $\mathbf{x}$ in $\mathcal{L}_0$, there exists an $m_{\mathbf{x}} \geq 1$ s.t.

$$\frac{1}{m_{\mathbf{x}}^{n_\varphi}} |\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))| \leq [\![\varphi, \mathcal{O}]\!](\mathbf{x},t) \leq |\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))|$$

So we must show that

$$\inf \left\{ \frac{1}{m_{\mathbf{x}}^{n_\varphi}} |\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))| \mid \mathbf{x} \in \mathcal{L}_0 \right\} > 0$$

Since

$$\inf_{\mathbf{x}} \frac{|\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))|}{m_{\mathbf{x}}^{n_\varphi}} \geq \frac{\inf_{\mathbf{x}} |\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))|}{\sup_{\mathbf{x}} m_{\mathbf{x}}^{n_\varphi}} = \frac{r}{\sup_{\mathbf{x}} m_{\mathbf{x}}^{n_\varphi}}$$

it suffices to show that $\sup_{\mathbf{x}} m_{\mathbf{x}}^{n_\varphi} < \infty \Leftrightarrow \sup_{\mathbf{x}} m_{\mathbf{x}} < \infty$

Recall that

$$m_{\mathbf{x}} = \max \left\{ \frac{\mathbf{dist}_\rho(\mathbf{x}, A \cap B)}{\alpha(\mathbf{x}, A, B)}, A, B \in \mathcal{D}_\varphi \text{ s.t. } \alpha(\mathbf{x}, A, B) \neq 0 \right\}$$

where $\alpha(\mathbf{x}, A, B) = \mathbf{dist}_\rho(\mathbf{x}, A) \sqcup \mathbf{dist}_\rho(\mathbf{x}, B)$. One way in which $m_{\mathbf{x}}$ could be infinite is if $\alpha(\mathbf{x}, A, B) = 0$ for all A, B in $\mathcal{D}_\varphi$. However, this case is eliminated by condition (11) of Lemma 4.2.

The other cases that can lead to an infinite $m_{\mathbf{x}}$ is if $\mathbf{dist}_\rho(\mathbf{x}, A \cap B) \to \infty$ faster than α, or $\alpha \to 0$ faster than $\mathbf{dist}_\rho(\mathbf{x}, A \cap B)$.

For any $\mathbf{x} \in \mathcal{L}_0$, let $a_{\mathbf{x}}$ be the projection of $\mathbf{x}$ on A, $b_{\mathbf{x}}$ its projection on B, and $y_{\mathbf{x}}$ its projection on $A \cap B$. Since

$$\rho(\mathbf{x}, y_{\mathbf{x}}) \leq \rho(\mathbf{x}, a_{\mathbf{x}}) + \rho(a_{\mathbf{x}}, y_{\mathbf{x}})$$

$$\rho(\mathbf{x}, y_{\mathbf{x}}) \leq \rho(\mathbf{x}, b_{\mathbf{x}}) + \rho(b_{\mathbf{x}}, y_{\mathbf{x}})$$

It comes ($diam(A)$ is the diameter of set A)

$$\mathbf{dist}_\rho(\mathbf{x}, A \cap B) \leq \alpha(\mathbf{x}, A, B) + \underbrace{diam(A) \sqcup diam(B)}_{\text{constant } c}$$

$$\implies \alpha(\mathbf{x}, A, B) \leq \mathbf{dist}_\rho(\mathbf{x}, A \cap B) \leq \alpha(\mathbf{x}, A, B) + c$$

Thus they both have the same growth rate, and their ratio is upper-bounded by some β, $0 < \beta < \infty$.

Therefore, $\sup_{\mathbf{x}} m_{\mathbf{x}} < \infty$ and finally

$$\inf_{\mathbf{x}} \frac{|\mathbf{Dist}_\rho(\mathbf{x}, \mathcal{L}_t(\varphi))|}{m_{\mathbf{x}}^{n_\varphi}} \geq \frac{r}{\beta^{n_\varphi}} > 0 \implies [\![\varphi, \mathcal{O}]\!](\mathbf{x}, t) > 0$$

□

7. REFERENCES

[1] H. Abbas and G. Fainekos. Linear hybrid system falsification through local search. In *Automated Technology for Verification and Analysis*, volume 6996 of *LNCS*, pages 503–510. Springer, 2011.

[2] H. Abbas, G. E. Fainekos, S. Sankaranarayanan, F. Ivancic, and A. Gupta. Probabilistic temporal logic falsification of cyber-physical systems. *ACM Transactions on Embedded Computing Systems*, 12(s2), May 2013.

[3] H. Abbas, M. O'Kelly, and R. Mangharam. Relaxed decidability and the robust semantics of metric temporal logic: Technical report. University of Pennsylvania Scholarly Commons, 2017.

[4] R. Alur, C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138(1):3–34, 1995.

[5] Y. S. R. Annapureddy, C. Liu, G. E. Fainekos, and S. Sankaranarayanan. S-taliro: A tool for temporal logic falsification for hybrid systems. In *Tools and algorithms for the construction and analysis of systems*, volume 6605 of *LNCS*, pages 254–257. Springer, 2011.

[6] J. Borwein and S. Fitzpatrick. Existence of nearest points in Banach spaces. *Can. J. Math.*, XLI(4):702–720, 1989.

[7] J. V. Deshmukh, A. Donzé, S. Ghosh, X. Jin, G. Juniwal, and S. A. Seshia. *Robust Online Monitoring of Signal Temporal Logic*, pages 55–70. Springer International Publishing, Cham, 2015.

[8] A. Donze. Breach, a toolbox for verification and parameter synthesis of hybrid systems. In *Computer Aided Verification*, volume 6174 of *LNCS*, pages 167–170. Springer, 2010.

[9] T. Dreossi, T. Dang, A. Donze, J. Kapinski, X. Jin, and J. V. Deshmukh. A trajectory splicing approach to concretizing counterexamples for hybrid systems. In *NASA Symposium on Formal Methods*, 2015.

[10] A. Eggers, M. Fränzle, and C. Herde. *SAT Modulo ODE: A Direct SAT Approach to Hybrid Systems*, pages 171–185. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[11] G. Fainekos. *Robustness of Temporal Logic Specifications*. PhD thesis, University of Pennsylvania, 2008.

[12] G. Fainekos and G. Pappas. Robustness of temporal logic specifications for continuous-time signals. *Theoretical Computer Science*, 410(42):4262–4291, September 2009.

[13] G. E. Fainekos, A. Girard, and G. J. Pappas. Temporal logic verification using simulation. In E. Asarin and P. Bouyer, editors, *FORMATS*, volume 4202 of *LNCS*, pages 171–186. Springer, 2006.

[14] M. Fisher. A semiclosed-loop algorithm for the control of blood glucose levels in diabetics. *Biomedical Engineering, IEEE Transactions on*, 38(1):57–61, 1991.

[15] P. Franek, S. Ratschan, and P. Zgliczynski. *Satisfiability of Systems of Equations of Real Analytic Functions Is Quasi-decidable*, pages 315–326. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[16] P. Franek, S. Ratschan, and P. Zgliczynski. Quasi-decidability of a fragment of the first-order theory of real numbers. *Journal of Automated Reasoning*, 57(2):157–185, 2016.

[17] M. Fränzle. Analysis of hybrid systems: An ounce of realism can save an infinity of states. In *Proceedings of the 13th International Workshop and 8th Annual Conference of the EACSL on Computer Science Logic (CSL)*, pages 126–140, London, UK, 1999. Springer-Verlag.

[18] S. Gao. Atrial fibrillation model. accessed 09/30/2016, 2016.

[19] S. Gao, J. Avigad, and E. M. Clarke. δ-complete decision procedures for satisfiability over the reals. In *Proceedings of the 6th International Joint Conference on Automated Reasoning*, IJCAR'12, pages 286–300, Berlin, Heidelberg, 2012. Springer-Verlag.

[20] S. Gao, J. Avigad, and E. M. Clarke. δ-decidability over the reals. In *Proceedings of the 2012 27th Annual IEEE/ACM Symposium on Logic in Computer Science*, LICS '12, pages 305–314, Washington, DC, USA, 2012. IEEE Computer Society.

[21] S. Gao, S. Kong, and E. M. Clarke. Satisfiability modulo ODEs. In *FMCAD*, pages 105–112, 2013.

[22] T. A. Henzinger. *The Theory of Hybrid Automata*, pages 265–292. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000.

[23] T. A. Henzinger, P. W. Kopke, A. Puri, and P. Varaiya. What's decidable about hybrid automata? *Journal of Computer and System Sciences*, 57(1):94 – 124, 1998.

[24] A. A. Julius, G. Fainekos, M. Anand, I. Lee, and G. Pappas. Robust test generation and coverage for hybrid systems. In *Hybrid Systems: Computation and Control*, volume 4416 of *LNCS*, pages 329–342. Springer-Verlag Berlin Heidelberg, 2007.

[25] K.-I. Ko. *Complexity theory of real functions*. Birkhauser, 1991.

[26] S. Kong, S. Gao, W. Chen, and E. Clarke. *dReach: δ-Reachability Analysis for Hybrid Systems*, pages 200–205. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[27] S. Kong, S. Gao, W. Chen, and E. Clarke. dreach: delta-reachability analysis for hybrid systems. In C. Baier and C. Tinelli, editors, *TACAS*, volume 9035 of *Lecture Notes in Computer Science*. 2015.

[28] R. Koymans. Specifying real-time properties with metric temporal logic. *Real-Time Systems*, 2(4):255–299, 1990.

[29] T. Nghiem, S. Sankaranarayanan, G. Fainekos, F. Ivancic, A. Gupta, and G. Pappas. Monte-carlo techniques for falsification of temporal properties of non-linear hybrid systems. In *Hybrid Systems: Computation and Control*, 2010.

[30] S. Ratschan. *Safety Verification of Non-linear Hybrid Systems Is Quasi-Semidecidable*, pages 397–408. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[31] S. Ratschan. Safety verification of non-linear hybrid systems is quasi-decidable. *Formal Methods in System Design*, 44(1):71–90, 2014.

[32] K. Weihrauch. *Computable analysis: an introduction*. Springer, 2000.