Specifying User Preferences Using Weighted Signal Temporal Logic

Noushin Mehdipour, *Graduate Student Member, IEEE*, Cristian-Ioan Vasile, *Member, IEEE*, and Calin Belta, *Fellow, IEEE*

***Abstract*—We extend Signal Temporal Logic (STL) to enable the specification of importance and priorities. The extension, called Weighted STL (wSTL), has the same qualitative (Boolean) semantics as STL, but additionally defines weights associated with Boolean and temporal operators that modulate its quantitative semantics (robustness). We show that the robustness of wSTL can be defined as weighted generalizations of all known compatible robustness functionals (i.e., robustness scores that are recursively defined over formulae) that can take into account the weights in wSTL formulae. We utilize this weighted robustness to distinguish signals with respect to a desired wSTL formula that has subformulae with different importance or priorities and time preferences, and demonstrate its usefulness in problems with conflicting tasks where satisfaction of all tasks cannot be achieved. We also employ wSTL robustness in an optimization framework to synthesize controllers that maximize satisfaction of a specification with user specified preferences.**

***Index Terms*—Autonomous systems, robotics, hybrid systems.**

I. Introduction

TEMPORAL logics, such as Linear Temporal Logic (LTL) and Computation Tree logic (CTL) [1] are formal specification languages that enable expressing temporal and Boolean properties of system executions. Recently, temporal logics have been used to formalize specifications for complex monitoring and control problems in cyber-physical systems. A variety of tools has been developed for analysis and control of many systems from such specifications [2], [3], [4].

Signal Temporal Logic (STL) [5] specifies signal characteristics over time. Its quantitative semantics, known as robustness, provides a measure of satisfaction or violation of the desired temporal specification, with larger robustness indicating more satisfaction. The quantitative semantics enables formulating STL satisfaction as an optimization problem with robustness as the objective function. This problem has been solved using heuristics, mixed-integer programming or gradient methods [6], [7], [8], [9], [10].

Multiple functionals have been proposed to capture the STL quantitative robustness. The traditional robustness introduced in [11] uses min and max functions over temporal and logical formulae, resulting in a *sound*, non-convex and non-smooth robustness function. For linear systems with linear costs and STL formulae defined over linear predicates, traditional robustness optimization approaches commonly encoded Boolean and temporal operators as linear constraints over continuous and integer variables [7], [8]. However, the resulting Mixed Integer Linear Programs (MILPs) scaled poorly with the size and horizon of the specifications (i.e., they require a large number of integer variables). Later works employed smooth approximations for max and min to achieve a differentiable robustness and use scalable gradient-based optimization methods applicable to general nonlinear systems. However, the *soundness* property was lost due to the approximation errors [9].

Several works have tackled the issue of defining sound robustness functionals with regularity properties (i.e., continuity and smoothness) [12], [13], [14], [15]. In [16], the limitations of traditional robustness (induced by the min and max functions) in optimization were categorized as *locality* and *masking*. *Locality* means that robustness depends only on the value of signal at a single time instant, while *masking* indicates that the satisfaction of parts of the formulae different from the most "extreme" part does not contribute to the robustness. References [16], [17] employed additive and multiplicative smoothing and eliminated the locality and masking effects to enhance optimization. Later works [14], [15] defined parametric approximations for max and min that enabled adjustment of the locality and masking to a desired level. A similar issue was studied in LTL specifications, where a counting method was used to distinguish between small and large satisfactions (or violations) of a LTL formula [18].

All these works have focused on the run-time performance of the planning or verification with temporal logic specifications. However, little attention has been devoted to the problem of capturing user preferences in satisfying temporal logic properties with timing constraints. In LTL, specifying the preferences of multiple temporal properties was addressed in *minimum-violation* [19], [20] or *maximum realizability* [21] problems, i.e., if multiple specifications are not realizable for

Manuscript received September 14, 2020; revised November 12, 2020; accepted December 8, 2020. Date of publication December 25, 2020; date of current version March 22, 2021. This work was partially supported by NSF at Boston University under Grant IIS-1723995 and Grant IIS-2024606. Recommended by Senior Editor S. Tarbouriech. *(Corresponding author: Noushin Mehdipour.)*

Noushin Mehdipour and Calin Belta are with the Division of Systems Engineering, Boston University, Boston, MA 02446 USA (e-mail: noushinm@bu.edu; cbelta@bu.edu).

Cristian-Ioan Vasile is with the Department of Mechanical Engineering and Mechanics, Lehigh University, Bethlehem, PA 18015 USA (e-mail: cvasile@lehigh.edu).

Digital Object Identifier 10.1109/LCSYS.2020.3047362

a system, it is preferable to synthesize a minimally violating or maximally realizing system. These problems were formulated by assigning priority-based positive numerical weights (weight functions) to the LTL formulae [21] or corresponding deterministic transition systems [19]. The idea of using priority functions was also studied in [22] in order to prioritize optimization of specific parameters in a mining problem with parametric temporal logic properties. Time Window Temporal Logic (TWTL) proposed in [23] enabled specifying preferences on the deadlines through temporal (deadline) relaxations and formulation of time delays.

However, for STL specifications, the problem of capturing user preferences, i.e., *importance* or *priorities* of different specifications or the timing of satisfaction is not well understood. The contributions of this letter are: (1) we extend STL to *Weighted Signal Temporal Logic* (*wSTL*) to formally capture importance and priorities of tasks or timing of satisfaction via weights; (2) we show that the extended quantitative semantics can be defined as a weighted generalization of a recursively defined STL robustness functional, (3) we propose adapted evaluation and control frameworks that use wSTL to reason about a system behavior with incompatible (infeasible) tasks or with performance preferences.

II. Preliminaries

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a real function. We define $[f]_+ = \begin{cases} f & f > 0 \\ 0 & \text{otherwise} \end{cases}$ and $[f]_- = -[-f]_+$, where $f = [f]_+ + [f]_-$. The sign function is denoted by $\text{sign} : \mathbb{R} \to \{-1, 0, 1\}$.

A. Signal Temporal Logic (STL)

Signal Temporal Logic (STL) was introduced in [5] to monitor temporal properties of real-valued signals. A discrete-time n-dimensional signal is denoted as $S = S[0], S[1], \ldots$ where $S[t] \in \mathbb{R}^n$, $t \in \mathbb{Z}_{\geq 0}$. We denote $I = [a, b] = \{k \in \mathbb{Z}_{\geq 0} \mid a \leq k \leq b;\ a, b \in \mathbb{Z}_{\geq 0}\}$ and $t + I$ as the interval $[t + a, t + b]$. In this letter, we consider a fragment of STL with syntax defined and interpreted over S as follows:

$$\varphi := \top \mid \mu \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \mathbf{G}_I\varphi \mid \mathbf{F}_I\varphi, \tag{1}$$

where φ, φ_1, φ_2 are STL formulae, $\top$ is logical *True*, $\mu := (l(S[t]) \geq 0)$ is a *predicate* where $l : \mathbb{R}^n \to \mathbb{R}$ is a Lipschitz continuous function defined over values of S, and $\neg$, $\wedge$ and $\vee$ are the Boolean *negation*, *conjunction* and *disjunction* operators. The Boolean constant *False* can be defined from $\top$ and $\neg$ in the usual way. The temporal operator *eventually* $\mathbf{F}_I\varphi$ is satisfied at time t if "φ is True at some time in $t + I$"; while *always* $\mathbf{G}_I\varphi$ is satisfied at time t if "φ is True at all times in $t + I$". For example, formula $\varphi = \mathbf{G}_{[0,7]}\mathbf{F}_{[0,3]}(S > 0)$ evaluated at time 0 specifies that for all times between 0 and 7, within the next 3 time units, signal S becomes positive. STL qualitative semantics determines whether S satisfies φ at time t ($S \models^t \varphi$) or violates it ($S \not\models^t \varphi$). Its quantitative semantics, or *robustness*, measures *how much* a signal satisfies or violates a specification.

Definition 1 (Traditional Robustness [11]): Given a specification φ and a signal S, the traditional robustness $\rho(\varphi, S, t)$ at time t is recursively defined as follows [11]:

$$\begin{aligned}
\rho(\mu, S, t) &:= l(S[t]),\\
\rho(\neg\varphi, S, t) &:= -\rho(\varphi, S, t),\\
\rho(\varphi_1 \wedge \varphi_2, S, t) &:= \min(\rho(\varphi_1, S, t), \rho(\varphi_2, S, t)),\\
\rho(\varphi_1 \vee \varphi_2, S, t) &:= \max(\rho(\varphi_1, S, t), \rho(\varphi_2, S, t)),\\
\rho(\mathbf{G}_I\varphi, S, t) &:= \inf_{t' \in t+I} \rho(\varphi, S, t'),\\
\rho(\mathbf{F}_I\varphi, S, t) &:= \sup_{t' \in t+I} \rho(\varphi, S, t').
\end{aligned} \tag{2}$$

Unless otherwise mentioned, throughout the letter, signals start at time 0 and we evaluate their robustness at time 0.

Theorem 1 (Soundness [11]): The traditional robustness is sound, i.e., $\rho(\varphi, S, t) > 0$ implies $S \models^t \varphi$, and $\rho(\varphi, S, t) < 0$ implies $S \not\models^t \varphi$.

If a formula (or subformula) φ is written as $\varphi = \bigwedge_i \varphi_i$, then each subformulae φ_i is called *obligatory* for φ. Similarly, if $\varphi = \bigvee_i \varphi_i$, then each φ_i is called *alternative* for φ.

B. Smooth Approximations

The max and min functions can be approximated by:

$$\begin{aligned}
\widetilde{\min}\{x_1, \ldots, x_d\} &\approx -\frac{1}{\beta}\ln\Big(\sum_{i=1}^{d} e^{-\beta x_i}\Big),\\
\widetilde{\max}\{x_1, \ldots, x_d\} &\approx \frac{\sum_{i=1}^{d} x_i e^{\beta x_i}}{\sum_{i=1}^{d} e^{\beta x_i}},
\end{aligned} \tag{3}$$

where $\beta > 0$ is an adjustable parameter determining an approximation of the true minimum and maximum [15].

III. Weighted Signal Temporal Logic (wSTL)

In many applications, a high-level temporal logic specification may consist of obligatory or alternative sub-specifications or timings with different importance or priorities. The expressivity of traditional STL does not allow for specifying these preferences. For instance, consider $\varphi = \mathbf{F}_{[0,5]}(S > 0)$, which is satisfied if S becomes greater than 0 within 5 time units, and assume that satisfaction at earlier times within this deadline is more desirable. Traditional robustness $\rho(\varphi, S, 0)$ for signals $S_1 = 0, 1, 0, 0, 0, 0, \ldots$ and $S_2 = 0, 0, 0, 0, 0, 1, \ldots$ is the same. Similar argument holds for approximation and Arithmetic-Geometric Mean (AGM) robustness scores in the literature [14], [15], [16]. However, it would be natural to assign a higher robustness to S_1 due to satisfaction of φ at an earlier time. Imposing importance and priorities of satisfaction especially becomes important when a formula has conflicting obligatory sub-formulae:

Example 1: Consider a car driving on the two-lane road shown in Fig. 1 [19]. The car starts from an initial point at $t = 0$ and has to reach *Green*. Meanwhile, it has to always drive in lane, and avoid *Blocked* area on the road. Assuming the duration of the overall task is bounded by 7, we formally define this specification as: $\varphi = \varphi_1 \wedge \varphi_2 \wedge \varphi_3$ where $\varphi_1 = \mathbf{F}_{[0,7]}Green$, $\varphi_2 = \mathbf{G}_{[0,7]}\neg Blocked$, $\varphi_3 = \mathbf{G}_{[0,7]}Lane$. As shown in Fig. 1, in order to reach *Green*, the car must either pass through the blocked area ($c_1 \models \varphi_1$,φ_3 but $c_1 \not\models \varphi_2$) or violate the lane requirement ($c_2 \models \varphi_1, \varphi_2$ but $c_2 \not\models \varphi_3$). In this example, a trajectory that satisfies φ does not exist. The minimally violating trajectory is dependent on the satisfaction importance of the obligatory tasks φ_2 and φ_3.

We introduce an extension of STL, called wSTL, to enable the definition of user preferences (priorities and importance).

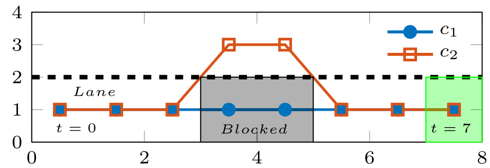


Fig. 1. Trajectories c_1 and c_2 from Example 1: the dots represent the positions at discrete times $t = 0, 1, \ldots, 7$ (the continuous interpolation is shown for visualization).

Definition 2 (wSTL Syntax): The syntax of wSTL is an extension of the STL syntax, and is defined as:

$$\varphi := \top \mid \mu \mid \neg\varphi \mid \bigwedge_{i=1:N}^{w} \varphi_i \mid \bigvee_{i=1:N}^{w} \varphi_i \mid \mathbf{G}_I^w \varphi \mid \mathbf{F}_I^w \varphi, \quad (4)$$

where the logical *True* (and *False*) value, the predicate μ, and all the Boolean and temporal operators have the same interpretation as in STL. The weight $w = [w_i]_{i=1:N} \in \mathbb{R}^N_{>0}$ assigns a positive weight w_i to each subformula i of the N subformulae of the Boolean operators, and $w = [w_k]_{k \in I} \in \mathbb{R}^{|I|}_{>0}$ assigns a positive weight w_k to time k in the interval I of the temporal operators and $|I|$ denotes cardinality of I.

The weights w capture the *importance* of obligatory specifications for conjunctions or *priorities* of alternatives for disjunctions. Similarly, w capture the *importance* of satisfaction times for temporal *always* or *priorities* of satisfaction times for temporal *eventually* over the interval I (see Examples 2, 3, 4). Throughout the letter, if the weight w associated with an operator (Boolean or temporal) in a wSTL formula is constant 1, we drop it from the notation. Thus, STL formulae are wSTL formulae with all weights equal to 1.

The Boolean (qualitative) semantics of a wSTL formula is the same as the associated STL formula without the weight functions, i.e., $S \models^t \varphi \Leftrightarrow S \models^t \hat{\varphi}$, where $\hat{\varphi}$ is the unweighted version of a wSTL formula φ.

Definition 3 (wSTL Robustness): Given a wSTL specification φ and a signal S, the weighted robustness score $r^w(\varphi, S, t)$ at time t is recursively defined as:

$$\begin{aligned}
&r^w(\mu, S, t) := l(S[t]),\\
&r^w(\neg\varphi, S, t) := -r^w(\varphi, S, t),\\
&r^w\left(\bigwedge_{i=1:N}^{w} \varphi_i, S, t\right) := \otimes^{\wedge}(w, [r^w(\varphi_i, S, t)]_{i=1:N}),\\
&r^w\left(\bigvee_{i=1:N}^{w} \varphi_i, S, t\right) := \oplus^{\vee}(w, [r^w(\varphi_i, S, t)]_{i=1:N}),\\
&r^w\left(\mathbf{G}_I^w \varphi, S, t\right) := \otimes^{\mathbf{G}}(w, [r^w(\varphi, S, t')]_{t' \in t+I}),\\
&r^w\left(\mathbf{F}_I^w \varphi, S, t\right) := \oplus^{\mathbf{F}}(w, [r^w(\varphi, S, t')]_{t' \in t+I}), \qquad (5)
\end{aligned}$$

where $\otimes^{\wedge} : \mathbb{R}^N_{>0} \times \mathbb{R}^N \to \mathbb{R}$, $\oplus^{\vee} : \mathbb{R}^N_{>0} \times \mathbb{R}^N \to \mathbb{R}$, $\otimes^{\mathbf{G}} : \mathbb{R}^{|I|}_{>0} \times \mathbb{R}^{|I|} \to \mathbb{R}$, and $\oplus^{\mathbf{F}} : \mathbb{R}^{|I|}_{>0} \times \mathbb{R}^{|I|} \to \mathbb{R}$ are aggregation functions associated with the $\wedge$, $\vee$, $\mathbf{G}$ and $\mathbf{F}$ operators, respectively.

The aggregation functions $\Upsilon = (\otimes^{\wedge}, \oplus^{\vee}, \otimes^{\mathbf{G}}, \oplus^{\mathbf{F}})$ are said to be *sign-consistent* if for all $w \in \mathbb{R}^d_{>0}$ and $\mathbf{r} \in \mathbb{R}^d$, we have $\min\{\mathbf{r}\} \cdot \otimes^{\wedge}(w, \mathbf{r}) > 0$ if $\min\{\mathbf{r}\} \neq 0$, $\max\{\mathbf{r}\} \cdot \oplus^{\vee}(w, \mathbf{r}) > 0$ if $\max\{\mathbf{r}\} \neq 0$, $\min\{\mathbf{r}\} \cdot \otimes^{\mathbf{G}}(w, \mathbf{r}) > 0$ if $\min\{\mathbf{r}\} \neq 0$, and $\max\{\mathbf{r}\} \cdot \oplus^{\mathbf{F}}(w, \mathbf{r}) > 0$ if $\max\{\mathbf{r}\} \neq 0$, where $\mathbf{r} = [r^w(\varphi_i, S, t)]_{i=1:N}$ or $\mathbf{r} = [r^w(\varphi, S, t')]_{t' \in t+I}$, and d is either N or $|I|$ for Boolean and temporal operators, respectively.

Theorem 2 [wSTL Soundness]: The weighted robustness score r^w given by Def. 3 is sound iff Υ is sign-consistent:

$$\begin{aligned}
&r^w(\varphi, S, t) > 0 \Leftrightarrow \rho(\hat{\varphi}, S, t) > 0 \to S \models^t \varphi,\\
&r^w(\varphi, S, t) < 0 \Leftrightarrow \rho(\hat{\varphi}, S, t) < 0 \to S \not\models^t \varphi \qquad (6)
\end{aligned}$$

Proof [Sketch]: A formal proof is omitted due to space constraints. Informally, soundness can be viewed as a sign consistency between the weighted robustness r^w and the (unweighted) traditional robustness ρ. The proof follows by structural induction and holds trivially for the base case corresponding to predicates. The induction step also follows easily from the induction hypothesis and the sign-consistency of Υ. Thus, the sign of the aggregation result in our (5) correctly captures the satisfaction and violation of composite formulae connected via Boolean and temporal operators.

The aggregation functions Υ are said to be *DeMorgan* if $\oplus^{\vee}(w, \mathbf{r}) = -\otimes^{\wedge}(w, -\mathbf{r})$, $\otimes^{\mathbf{G}}(w, \mathbf{r}) = \otimes^{\wedge}(w, \mathbf{r})$, and $\oplus^{\mathbf{F}}(w, \mathbf{r}) = \oplus^{\vee}(w, \mathbf{r})$ for all $w, \mathbf{r}$ of appropriate dimension.

The next proposition shows how the constraints for sign-consistency can be simplified if Υ is DeMorgan.

Proposition 1: If Υ is DeMorgan and $\min\{\mathbf{r}\} \cdot \otimes^{\wedge}(w, \mathbf{r}) > 0$ for all $w \in \mathbb{R}^d_{>0}$ and $\mathbf{r} \in \mathbb{R}^d$ with $\min\{\mathbf{r}\} \neq 0$, $d \geq 1$, then Υ is sign-consistent.

Proof: For brevity, we only show sign consistency for disjunction. Let $\mathbf{r} \in \mathbb{R}^d$ with $\max\{\mathbf{r}\} \neq 0$, and $w \in \mathbb{R}^d_{>0}$. We have $\max\{\mathbf{r}\} \cdot \oplus^{\vee}(w, \mathbf{r}) = \max\{\mathbf{r}\} \cdot (-\otimes^{\wedge}(w, -\mathbf{r})) = \min\{-\mathbf{r}\} \cdot \otimes^{\wedge}(w, -\mathbf{r}) > 0$, where the last inequality holds because $\min\{-\mathbf{r}\} = -\max\{\mathbf{r}\} \neq 0$. Proofs for the other operators follow similarly.

In the following, we show that desirable properties of STL robustness can be obtained in wSTL as well.

Proposition 2: Let r^w be a weighted robustness defined by Def. 3 using aggregation functions Υ. The functional r^w satisfies DeMorgan's law iff Υ is DeMorgan. Double negation holds unconditionally.

Proof: The double negation property follows trivially from Def. 3, i.e., $r^w(\neg\neg\varphi, S, t) = -r^w(\neg\varphi, S, t) = r^w(\varphi, S, t)$. Since Υ is DeMorgan, it follows immediately by structural induction that r^w satisfies DeMorgan's law.

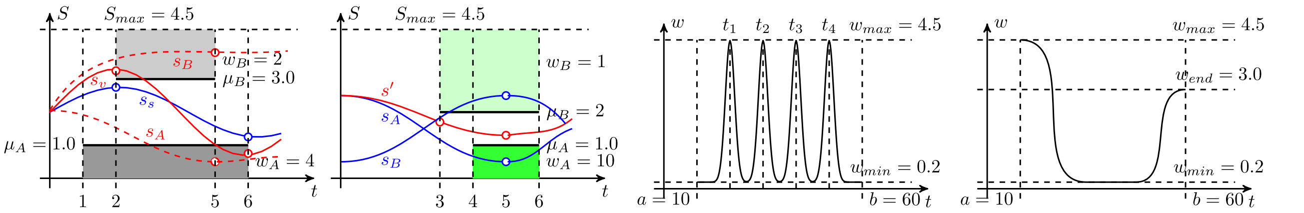
Function $\otimes^{\wedge} \in \Upsilon$ is (a) commutative if $\otimes^{\wedge}(Pw, P\mathbf{r}) = \otimes^{\wedge}(w, \mathbf{r})$, (b) idempotent if $\otimes^{\wedge}(w, c\mathbf{1}) = c$, (c) monotonic if $\otimes^{\wedge}(w, \mathbf{r}) \leq \otimes^{\wedge}(w, \mathbf{r} + h)$, (d) absolutely scalable if $\otimes^{\wedge}(w, \alpha\mathbf{r}) = \alpha \cdot \otimes^{\wedge}(w, \mathbf{r})$, for all permutation matrices $P \in \mathbb{R}^d \times \mathbb{R}^d$, $c \in \mathbb{R}$, $h \in \mathbb{R}^d_{\geq 0}$, $\alpha \in \mathbb{R}_{\geq 0}$, where $\mathbf{1}$ is a vector of all 1's with appropriate dimension [14], [16], [17]. These definitions hold for the other functions in Υ similarly.

Let r^w be a weighted robustness defined by Def. 3 using aggregation functions Υ. The functional r^w is said to be commutative, idempotent, monotonic, and absolutely scalable, if the properties hold for all functions in Υ, respectively.

Proposition 3: If Υ is DeMorgan and $\otimes^{\wedge}$ is commutative (idempotent, monotonic, absolutely scalable), then so is r^w.

Proof: We first show that the properties hold for $\oplus^{\vee}$ as well. For brevity, we only show monotonicity: we have $\oplus^{\vee}(w, \mathbf{r} + h) = -\otimes^{\wedge}(w, -\mathbf{r} - h) \geq -\otimes^{\wedge}(w, -\mathbf{r}) = \oplus^{\vee}(w, \mathbf{r})$, where $h \geq 0$. The proof for the temporal operators follow similarly from the Boolean ones. Since the properties hold for all functions in Υ, by definition they hold for r^w as well.

In the following, we discuss some examples to illustrate the expressivity of wSTL. We define weighted generalizations of the traditional [11] and AGM [16] robustness in a way that they measure how much a wSTL specification is satisfied or



(a) Importance of obligatory tasks (b) Priorities of alternatives (c) Preferences over time (d) Preferences over time

Fig. 2. wSTL performance for different specifications. Signals and weights are considered at discrete times $t \in \mathbb{Z}_{\geq 0}$, the continuous interpolation is shown for visualization.

violated considering the importance or priorities of its subformulae and time. The wSTL robustness for other compatible recursive STL robustness measures [9], [12], [13], [14], [15] can be defined similarly (see Section V). For brevity, we denote all weighted robustness scores $r^w(\varphi, S, 0)$ by $r^w(\varphi, S)$. Weighted robustness for predicates μ and the negation operator are defined as in Def. 3.

A. Weighted Traditional Robustness

We propose a weighted generalization of the traditional robustness in (2) denoted by ρ^w by defining a DeMorgan Υ^ρ with conjunction aggregation as

$$\otimes^\wedge (w, \mathbf{r}) = \min_{i=1:N} \left\{ \left((\frac{1}{2} - \bar{w}_i) \operatorname{sign}(r_i) + \frac{1}{2} \right) \cdot r_i \right\}, \quad (7)$$

where $\mathbf{r} = [\rho^w(\varphi_i, S, t)]_{i=1:N}$, and $r_i = \rho^w(\varphi_i, S, t)$ for brevity, and $\bar{w}_i = \frac{w_i}{\mathbf{1}^T w}$ are normalized weights for the conjunction operator, therefore, $1 - \bar{w}_i = \sum_{j=1:N, j\neq i} \bar{w}_j$.

In the following, we explain how formulation of weights in (7) aligns with the interpretation of maximum satisfaction or minimum violation of $\varphi = \bigwedge_{i=1:N}^{w} \varphi_i$ with respect to the subformulae importance. Let $\bar{w}_h = \max_{i=1:N}\{\bar{w}_i\}$, i.e., φ_h be the most important subformula for φ. We consider the following cases. Assume $\forall i : r_i = c > 0$, we have $(\frac{1}{2} - \bar{w}_i) \operatorname{sign}(r_i) + \frac{1}{2} = 1 - \bar{w}_i$ and $\otimes^\wedge(w, \mathbf{r}) = \min_i\{(1 - \bar{w}_i)c\} = (1 - \bar{w}_h)c$ which holds since $1 - \bar{w}_h = \min_{i=1:N}\{1 - \bar{w}_i\}$. Thus, in the case where all φ_is are satisfied with a same score, ρ^w captures satisfaction of the most important subformula φ_h, and maximizing ρ^w for control synthesis leads to improving satisfaction of φ_h (see Section IV). Next assume $\forall i : r_i = c < 0$, we have $(\frac{1}{2} - \bar{w}_i) \operatorname{sign}(r_i) + \frac{1}{2} = \bar{w}_i$, and $\otimes^\wedge(w, \mathbf{r}) = \min_i\{\bar{w}_i c\} = \bar{w}_h c$ meaning that in the case where all φ_is are violated with a same negative score, ρ^w is imposed by violation of φ_h, and maximizing ρ^w for control synthesis leads to reducing violation of φ_h. In all other cases, ρ^w of φ is determined based on the trade-off between level of satisfaction or violation (robustness values) and assigned importance of subformulae φ_is (see Example 2).

Proposition 4: The robustness ρ^w defined by Def. 3 with Υ^ρ is sound, commutative, monotonic, and absolutely scalable.

Proof: The properties are inherited by ρ^w from $\otimes^\wedge$ in (7) and the DeMorgan property of Υ^ρ. For brevity, we only show soundness. Let $c = \min\{\mathbf{r}\} \neq 0$, $v_i = (\frac{1}{2} - \bar{w}_i) \operatorname{sign}(r_i) + \frac{1}{2}$, and $v_\ell r_\ell = \min_i\{v_i r_i\}$. Note that $v_i > 0, \forall i$ since $\bar{w}_i \in (0, 1)$. From (7), we have $\min\{\mathbf{r}\} \cdot \otimes^\wedge(w, \mathbf{r}) = c v_\ell r_\ell \geq v_\ell c^2 > 0$. The soundness of ρ^w follows from the sign-consistency of Υ^ρ given by Proposition 1. All other properties follow by Proposition 3.

Note that ρ^w defined by Υ^ρ from (7) is not idempotent.

Example 2 (Importance of Obligatory Tasks): Consider $\varphi = \varphi_A \wedge^w \varphi_B = \mathbf{G}_{[1,6]}(S \geq 1) \wedge^w \mathbf{G}_{[2,5]}(S \leq 3)$ with $w_A = 4$ and $w_B = 2$, and signals in Fig. 2(a). The weights associated with the subformulae of the conjunction operator specify that it is twice as important to stay above 1 between time $t = 1$ to $t = 6$ than to stay below 3 from time $t = 2$ to $t = 5$. From (7), $\rho^w(\varphi, s_s) = \min\left((1 - \frac{4}{6}) \times 0.25, (1 - \frac{2}{6}) \times 0.25\right) = 0.08$ and $\rho^w(\varphi, s_v) = \min(\frac{4}{6} \times -0.25, \frac{2}{6} \times -0.25) = -0.16$, highlighting importance of φ_A (it is more important for s_s to satisfy φ_A, and violation of φ_A by s_v is considered worse than violation of φ_B). Next, consider s_A which violates φ_A but satisfies φ_B, and s_B which violates φ_B but satisfies φ_A. We have $\rho^w(\varphi, s_A) = \frac{4}{6} \times -0.5 = -0.33$ and $\rho^w(\varphi, s_B) = \frac{2}{6} \times -0.75 = -0.25$. Thus, based on the violation scores and importance w_A and w_B, s_A is considered more violating than s_B.

Example 3 (Priorities of Alternative Tasks): Consider $\varphi = \varphi_A \vee^w \varphi_B = \mathbf{F}_{[4,6]}(S \leq 1) \vee^w \mathbf{F}_{[3,6]}(S \geq 2)$ with $w_A = 10$, $w_B = 1$ and signals in Fig. 2(b). From (7) and using DeMorgan Υ^ρ, we have $\oplus^\vee(w, \mathbf{r}) = \max_{i=1:N}\{(-(\frac{1}{2} - \bar{w}_i) \operatorname{sign}(r_i) + \frac{1}{2}) \cdot r_i\}$. Thus, $\rho^w(\varphi, s_A) = \max(\frac{10}{11} \times 0.5, (1 - \frac{1}{11}) \times -0.8) = 0.45$, while $\rho^w(\varphi, s_B) = \max((1 - \frac{10}{11}) \times -1.3, \frac{1}{11} \times 0.5) = 0.045$. Although both signals satisfy φ, s_A is preferred to s_B, i.e., has a higher robustness, because it visits the higher priority region (induced by φ_A) while s_B visits the lower priority region (induced by φ_B). Similarly, ρ^w for s' is imposed by φ_A (distance to the higher priority region at $t = 5$), rather than by φ_B (for which s' has the same distance to the lower priority region at $t = 3$). This formulation results in moving s' towards satisfying φ_A when maximizing ρ^w for control synthesis (see Section IV, Section V).

Example 4 (Preferences Over Time): Consider the formulae $\varphi_F = \mathbf{F}^w_{[10,60]}\varphi$ and $\varphi_G = \mathbf{G}^w_{[10,60]}\varphi$, and weights in Fig. 2(c) and 2(d). For *eventually*, φ_F with weight w from Fig. 2(c) specifies that the task φ should be done within $[10, 60]$ with higher priorities at one of the times $\{t_1, t_2, t_3, t_4\}$; while the weight w in Fig. 2(d) gives priorities to satisfaction at the endpoints especially at the start. For *always*, φ_G with weight w from Fig. 2(c) specifies that φ must hold at all times within $[10, 60]$, more importantly at times $\{t_1, t_2, t_3, t_4\}$; while the w in Fig. 2(d) specifies a higher importance at the end of the interval and the highest importance at the start.

B. Weighted AGM Robustness

We adapt the AGM robustness [16] to a weighted AGM robustness denoted by η^w. The weighted AGM captures satisfaction of all subformulae and times, as well as their importance and priorities. For example, for $\varphi = \mathbf{F}^w_{[0,5]}(S > 0)$, if satisfaction at earlier times is desirable, η^w for $S_1 = 0, 1, 0, 0, 0, 0, \ldots$ is higher than $S_2 = 0, 0, 0, 0, 0, 1, \ldots$ but

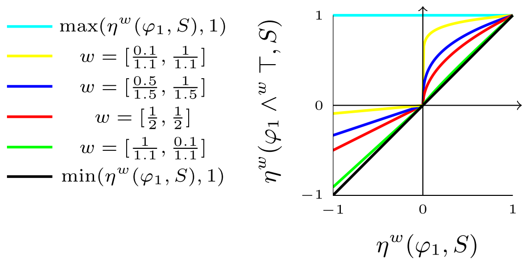


Fig. 3. Effect of w in $\eta^w(\varphi_1 \wedge^w \top, S)$, legends correspond to the signals from top to bottom.

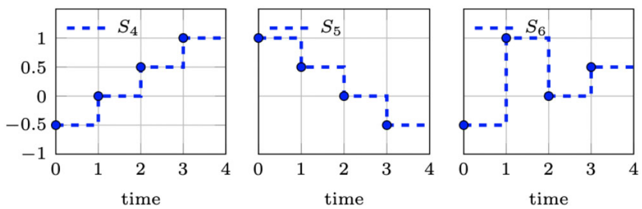


Fig. 4. Discrete-time signals in Example 6.

lower than $S_3 = 0, 1, 1, 1, 0, 0, \ldots$ since S_3 satisfies φ as early as S_1 but also at more time points. Notice that weighted traditional robustness ρ^w cannot distinguish between S_1 and S_3. The aggregation functions Υ^η in η^w can be defined using weighted arithmetic- and geometric- means. We define Υ^η to be DeMorgan with the conjunction aggregation as

$$\otimes^\wedge (w, \mathbf{r}) = \begin{cases} \prod_i r_i^{\bar{w}_i} & \text{if } \min\{\mathbf{r}\} > 0, \\ \sum_i \bar{w}_i [r_i]_- & \text{otherwise}, \end{cases} \tag{8}$$

where $\mathbf{r} = [\eta^w(\varphi_i, S, t)]_{i=1:N}$, $r_i = \eta^w(\varphi_i, S, t)$ for brevity, and $\bar{w}_i = \frac{w_i}{\mathbf{1}^T w}$ are normalized weights.

Proposition 5: The robustness η^w defined as in Def. 3 with Υ^η is sound, commutative, idempotent, monotonic, and absolutely scalable.

Proof: Again, the properties are inherited from $\otimes^\wedge$ in (8), and the DeMorgan property of Υ^η. From (8), we have $\min\{\mathbf{r}\} \cdot \otimes^\wedge(w, \mathbf{r}) = \min\{\mathbf{r}\} \cdot \prod_i r_i^{\bar{w}_i} > 0$ if $\min\{\mathbf{r}\} > 0$, and $\min\{\mathbf{r}\} \cdot \otimes^\wedge(w, \mathbf{r}) = \min\{\mathbf{r}\} \cdot \sum_i \bar{w}_i [r_i]_- > 0$ if $\min\{\mathbf{r}\} < 0$. Thus, Υ^η is sign-consistent by Proposition 1, and η^w is sound. All other properties follow in the same way via Proposition 3.

Example 5: We demonstrate how the conjunction function changes for different normalized weights w for $\eta^w(\varphi_1 \wedge^w \top, S)$, where $\eta^w(\top, S) = 1$ is fixed, and $\eta^w(\varphi_1, S) \in [-1, 1]$. As illustrated in Fig. 3, by assigning a higher importance to $\top$, $\eta^w(\varphi_1 \wedge^w \top, S)$ is closer to 1, and for a higher importance to φ_1, robustness is closer to $\eta^w(\varphi_1, S)$. Moreover, similar to the AGM robustness, $\eta^w(\varphi, S, t)$ is sound and monotone, and for $\eta^w(\varphi_1, S) = 1$, we have $\eta^w(\varphi_1 \wedge^w \top, S) = 1$, $\forall w$.

Example 6: Consider signals S_4, S_5, S_6 shown in Fig. 4 and $\varphi = \mathbf{F}^w{}_{[0,3]}(S \geq 0)$. We can choose the weights (priorities) as $w_k = \gamma^{(k-1)}$ with the discount rate $\gamma > 0$ to reward satisfaction of the formula at earlier times within the deadline. For larger γ (closer to 1), satisfaction at different times has similar priorities, and by decreasing γ, satisfaction at earlier times results in a higher η^w, as seen in Table I. Note that unweighted robustness of these signals are equal.

Example 7: Reconsider Example 1. Assuming $w_k = 1$ in the interval $I = [0, 7]$, we have $\eta^w(\varphi_2, c_1) = \eta^w(\varphi_3, c_2) = -\frac{2}{8}$. For the overall specification φ, from (8) we have

TABLE I
UNWEIGHTED ρ AND WEIGHTED η^W ROBUSTNESS SCORES FOR DIFFERENT VALUES OF γ IN $w_k = \gamma^{(k-1)}$ FOR $\varphi = \mathbf{F}^w_{[0,3]}(S \geq 0)$

Signal	ρ	η	$\eta^w, \gamma = 0.9$	$\eta^w, \gamma = 0.5$	$\eta^w, \gamma = 0.1$
S_4	1	0.375	0.330	0.133	0.005
S_5	1	0.375	0.420	0.666	0.945
S_6	1	0.375	0.367	0.300	0.090

$\eta^w(\varphi, c_1) = \sum_{i=1}^3 \bar{w}_i [\eta^w(\varphi_i, c_1)]_- = \bar{w}_2\, \eta^w(\varphi_2, c_1)$ and $\eta^w(\varphi, c_2) = \bar{w}_3\, \eta^w(\varphi_3, c_2)$. Assume avoiding *Blocked* is more important than staying in the lane. By choosing $w_2 > w_3$, we have $\eta^w(\varphi, c_1) < \eta^w(\varphi, c_2) < 0$, and c_2 is chosen as the minimally violating trajectory (compared to c_1).

Remark 1: wSTL syntax and quantitative semantics in Def. 2 and Def. 3 can be extended to continuous-time. For temporal operators over continuous time, the weights w become functions of time $w(t)$, and the weighted robustness can be defined by replacing max, min, $\sum, \prod$ in Υ with sup, inf, integral and product integral, respectively [14], [17].

IV. SYNTHESIS USING WEIGHTED ROBUSTNESS

Consider a discrete-time dynamical system given by:

$$q[t+1] = f(q[t], u[t]), q[0] = q_0, \tag{9}$$

where $q[t] \in \mathcal{Q} \subseteq \mathbb{R}^n$ is the state of the system and $u[t] \in \mathcal{U} \subseteq \mathbb{R}^m$ is the control input at time t, $q_0 \in \mathcal{Q}$ is the initial state and $f{:}\mathcal{Q} \times \mathcal{U} \to \mathcal{Q}$ is a Lipschitz continuous function. We denote the system trajectory generated by applying control input $\mathbf{u} = u[0], \ldots, u[T-1]$ to (9) over a finite time T starting from the initial state q_0 by $\mathbf{q}(q_0, \mathbf{u})$. Consider a cost function $J(u[t], q[t+1])$ representing the cost of ending up at state $q[t+1]$ by applying control input $u[t]$ at time t. The desired specification is given by wSTL formula φ over the system's trajectories. The control synthesis problem is defined as finding an optimal control policy $\mathbf{u}^*$ that minimizes the cost and maximizes the weighted robustness r^w such that the resulting $\mathbf{q}(q_0, \mathbf{u}^*)$ satisfies φ with r^w greater than ϵ:

$$\begin{aligned} \mathbf{u}^* = \operatorname{argmax}_{u[t]\in\mathcal{U}}\ & r^w(\varphi, \mathbf{q}(q_0, \mathbf{u})) - \lambda \sum_{t=0}^{T-1} J(u[t], q[t+1]) \\ \text{s.t.} \quad & \text{dynamics (9) are satisfied}, \\ & r^w(\varphi, \mathbf{q}(q_0, \mathbf{u})) > \epsilon, \end{aligned} \tag{10}$$

where $\epsilon \geq 0$ is the satisfaction margin (soundness threshold) [9] and λ penalizes the trade-off between maximizing r^w and minimizing the cost. Since weighted robustness is defined as a generalization of an unweighted STL robustness, previous robustness optimization frameworks including MILPs [8] and gradient-based methods [9] can be adapted to solve (10). In this letter, we use a gradient-based optimization where the objective function in (10) is iteratively maximized by updating control inputs $u[t]$ proportional to the gradient of objective function [24]. A similar synthesis approach can be applied to continuous time with zeroth-order hold input [17].

V. CASE STUDY

Consider a discrete-time nonlinear dynamical system as:

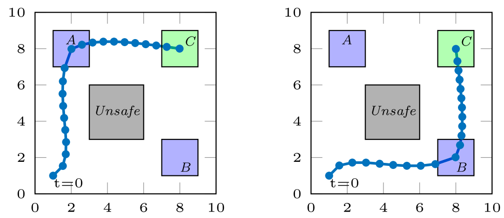


Fig. 5. Trajectories from the synthesized control $\mathbf{u}^*$ satisfy φ and minimize the cost. $w_A > w_B$ (left), $w_A < w_B$ (right).

$$\begin{aligned} x[t+1] &= x[t] + \cos\theta[t]v[t], \\ y[t+1] &= y[t] + \sin\theta[t]v[t], \\ \theta[t+1] &= \theta[t] + v[t]w[t], \end{aligned} \tag{11}$$

and a task "Eventually visit A or B within $[1, 10]$ and eventually visit C within $[11, 20]$ and Always avoid Unsafe and Always stay inside Boundary" given by wSTL formula:

$$\begin{aligned} \varphi = (\mathbf{F}_{[1,10]}\ (A \vee^w B)) \ \wedge (\mathbf{F}_{[11,20]}\ C) \\ \wedge (\mathbf{G}_{[1,20]}\ \neg \mathit{Unsafe}) \ \wedge (\mathbf{G}_{[1,20]}\ \mathit{Boundary}), \end{aligned} \tag{12}$$

where $A = [7, 9] \times [1, 3]$ or $B = [1, 3] \times [7, 9]$ and $C = [7, 9]^2$ are regions to be sequentially visited within the deadlines, $\mathit{Unsafe} = [3, 6]^2$ and $\mathit{Boundary} = [0, 10]^2$. $q = [x, y, \theta]$ is state vector with initial state $q_0 = [1, 1, \pi/4]$, $u = [v, w]$ is the input vector with $\mathcal{U} = [-2, 2]^2$, and cost function is $\frac{1}{2}\sum_{k=0}^{T-1} \|u[t]\|^2$ with $T = 20$, $\lambda = 0.05$ in (10). We define $\otimes^\wedge(w, \mathbf{r}) = \widetilde{\min}_i\{((\frac{1}{2} - \bar{w}_i)\,\mathrm{sign}(r_i) + \frac{1}{2}) \cdot r_i\}$, $\oplus^\vee(w, \mathbf{r}) = \widetilde{\max}_i\{(-(\frac{1}{2} - \bar{w}_i)\,\mathrm{sign}(r_i) + \frac{1}{2}) \cdot r_i\}$ using smooth approximations in (3) (similar for $\otimes^{\mathbf{G}}$ and $\oplus^{\mathbf{F}}$) and approximate $\widetilde{\mathrm{sign}}(r_i) \approx \tanh(\beta r_i)$, $\beta > 0$ to obtain a weighted smooth robustness $\tilde{\rho}^w$ of [15]. Note that $(\frac{1}{2} - \bar{w}_i)\widetilde{\mathrm{sign}}(r_i) + \frac{1}{2} > 0, \forall i$ (similarly $-(\frac{1}{2} - \bar{w}_i)\widetilde{\mathrm{sign}}(r_i) + \frac{1}{2} > 0$) since $\bar{w}_i \in (0, 1)$, $-1 \leq \widetilde{\mathrm{sign}}(r_i) \leq 1$, so soundness of $\tilde{\rho}^w$ can be interpreted similar to [15], i.e., if $\tilde{\rho}^w(\varphi, \mathbf{q}(q_0, \mathbf{u})) > 0$, then φ is satisfied. Fig. 5 shows trajectories obtained from optimizing (10) with weighted robustness $\tilde{\rho}^w$ achieved for $\beta = 10$ up to a same termination criteria with different priorities for visiting A or B as $w_A = \frac{2}{3}$, $w_B = \frac{1}{3}$ (left), and $w_A = \frac{1}{3}$, $w_B = \frac{2}{3}$ (right). All other weights dropped from the notation are constant 1. The optimization is implemented in MATLAB using the SQP optimizer and takes about 1.2 seconds on a Mac with 2.5 GHz Core i7 CPU 16GB RAM. In the given symmetrical configuration and initial state, optimizing the weighted robustness ensures that the optimal trajectory visits the higher priority region A or B as chosen by the disjunction priorities w.

VI. Conclusion and Future Work

We presented an extension of STL to improve its expressivity by encoding the importance or priorities of subformulae and time in a formula. The new formalism, called wSTL, is advantageous in problems where satisfaction of a formula is not feasible, so the less important subformulae or time are preferred to be violated to guarantee more important ones are satisfied. The weighted robustness associated with wSTL improved the optimal behavior in a control synthesis problem where prioritized tasks were critical. Future work investigates learning frameworks for systematic design of weights to capture hierarchies in wSTL formulae.

References

[1] C. Baier and J. Katoen, *Principles of Model Checking*. Cambridge, MA, USA: MIT Press, 2008.

[2] C. Belta, B. Yordanov, and E. A. Göl, *Formal Methods for Discrete-Time Dynamical Systems*, vol. 89. Cham, Switzerland: Springer, 2017.

[3] P. Tabuada, *Verification and Control of Hybrid Systems: A Symbolic Approach*, Boston, MA, USA: Springer, 2009.

[4] R. Alur, T. A. Henzinger, and E. D. Sontag, *Hybrid Systems III: Verification and Control*, vol. 3. Heidelberg, Germany: Springer, 1996.

[5] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*. Heidelberg, Germany: Springer, 2004, pp. 152–166.

[6] N. Mehdipour, D. Briers, I. Haghighi, C. M. Glen, M. L. Kemp, and C. Belta, "Spatial-temporal pattern synthesis in a network of locally interacting cells," in *Proc. IEEE Conf. Decis. Control*, Miami Beach, FL, USA, 2018, pp. 3516–3521.

[7] S. Saha and A. A. Julius, "An MILP approach for real-time optimal controller synthesis with metric temporal logic specifications," in *Proc. IEEE Amer. Control Conf. (ACC)*, Boston, MA, USA, 2016, pp. 1105–1110.

[8] V. Raman, A. Donzé, M. Maasoumy, R. M. Murray, A. Sangiovanni-Vincentelli, and S. A. Seshia, "Model predictive control with signal temporal logic specifications," in *Proc. 53rd IEEE Conf. Decis. Control*, Los Angeles, CA, USA, 2014, pp. 81–87.

[9] Y. V. Pant, H. Abbas, and R. Mangharam, "Smooth operator: Control using the smooth robustness of temporal logic," in *Proc. IEEE Conf. Control Technol. Appl.*, Mauna Lani, HI, USA, 2017, pp. 1235–1240.

[10] C. Belta and S. Sadraddini, "Formal methods for control synthesis: An optimization perspective," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 2, pp. 115–140, May 2019.

[11] A. Donzé and O. Maler, "Robust satisfaction of temporal logic over real-valued signals," in *Proc. Int. Conf. Formal Model. Anal. Timed Syst.*, 2010, pp. 92–106.

[12] L. Lindemann and D. V. Dimarogonas, "Robust control for signal temporal logic specifications using discrete average space robustness," *Automatica*, vol. 101, pp. 377–387, Mar. 2019.

[13] I. Haghighi, N. Mehdipour, E. Bartocci, and C. Belta, "Control from signal temporal logic specifications with smooth cumulative quantitative semantics," in *Proc. IEEE Conf. Decis. Control*, Nice, France, 2019, pp. 4361–4366.

[14] P. Varnai and D. V. Dimarogonas, "On robustness metrics for learning STL tasks," in *Proc. Amer. Control Conf.*, Denver, CO, USA, 2020, pp. 5394–5399.

[15] Y. Gilpin, V. Kurtz, and H. Lin, "A smooth robustness measure of signal temporal logic for symbolic control," *IEEE Control Syst. Lett.*, vol. 5, no. 1, pp. 241–246, Jan. 2021, doi: 10.1109/LCSYS.2020.3001875.

[16] N. Mehdipour, C.-I. Vasile, and C. Belta, "Arithmetic-geometric mean robustness for control from signal temporal logic specifications," in *Proc. Amer. Control Conf. (ACC)*, Philadelphia, PA, USA, 2019, pp. 1690–1695.

[17] N. Mehdipour, C. Vasile, and C. Belta, "Average-based robustness for continuous-time signal temporal logic," in *Proc. IEEE 58th Conf. Decis. Control (CDC)*, Nice, France, 2019, pp. 5312–5317.

[18] P. Tabuada and D. Neider, "Robust linear temporal logic," in *Proc. EACSL Annu. Conf. Comput. Sci. Logic*, vol. 62, 2016, pp. 1–21.

[19] C.-I. Vasile, J. Tumova, S. Karaman, C. Belta, and D. Rus, "Minimum-violation scLTL motion planning for mobility-on-demand," in *Proc. Int. Conf. Robot. Autom.*, Singapore, 2017, pp. 1481–1488.

[20] J. Tůmová, L. I. R. Castro, S. Karaman, E. Frazzoli, and D. Rus, "Minimum-violation LTL planning with conflicting specifications," in *Proc. IEEE Amer. Control Conf.*, Washington, DC, USA, 2013, pp. 200–205.

[21] R. Dimitrova, M. Ghasemi, and U. Topcu, "Maximum realizability for linear temporal logic specifications," in *Proc. Int. Symp. Autom. Technol. Verification Anal.*, 2018, pp. 458–475.

[22] B. Hoxha, A. Dokhanchi, and G. Fainekos, "Mining parametric temporal logic properties in model-based design for cyber-physical systems," *Int. J. Softw. Tools Technol. Transf.*, vol. 20, no. 1, pp. 79–93, 2018.

[23] C.-I. Vasile, D. Aksaray, and C. Belta, "Time window temporal logic," *Theor. Comput. Sci.*, vol. 691, pp. 27–54, Aug. 2017.

[24] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Sci., 1999.