IEEE Control Systems Letters paper presented at
2023 American Control Conference (ACC)
San Diego, CA, USA. May 31 - June 2, 2023

# Robust Multi-Agent Coordination from CaTL+ Specifications

Wenliang Liu[1], Kevin Leahy[2], Zachary Serlin[2], Calin Belta[1]

***Abstract*— We consider the problem of controlling a heterogeneous multi-agent system required to satisfy temporal logic requirements. Capability Temporal Logic (CaTL) was recently proposed to formalize such specifications for deploying a team of autonomous agents with different capabilities and cooperation requirements. In this paper, we extend CaTL to a new logic CaTL+, which is more expressive than CaTL and has semantics over a continuous workspace shared by all agents. We define a novel robustness metric for CaTL+, which is sound, differentiable almost everywhere and eliminates masking, which is one of the main limitations of existing traditional robustness metrics. We formulate a control synthesis problem to maximize CaTL+ robustness and propose a two-step optimization method to solve this problem. Simulation results are included to illustrate the increased expressivity of CaTL+ and the efficacy of the proposed control synthesis approach.**

## I. INTRODUCTION

Planning and controlling multi-agent systems from temporal logic specifications is a challenging problem that received a lot of attention in recent years. In [1], the authors used distributed formal synthesis to allocate global task specifications expressed as Linear Temporal Logic (LTL) formulas to locally interacting agents with finite dynamics. Related works [2]–[4] proposed extensions to more realistic models and scenarios. The authors of [5] used Signal Temporal Logic (STL) and Spatial Temporal Reach and Escape Logic (STREL) as specification languages to capture the connectivity constraints of a multi-robot team.

The above approaches do not scale for large teams. Very recent work has focused on development of logics and synthesis algorithms specifically tailored for such situations. The authors of [6] proposed Counting LTL (cLTL), which is used to define tasks for large groups of identical agents. *Counting constraints* specify the number of agents that need to achieve a certain goal. As long as enough agents achieve the goal, it does not matter which specific subgroup does that. Similar ideas are used in Capability Temporal Logic (CaTL) [7]. The atomic unit of a CaTL formula is a *task*, which specifies the number of agents with certain capability that need to reach some region and stay there for some time. Since agents can have different capabilities, CaTL is appropriate for heterogeneous teams. Unlike cLTL, CaTL is a fragment of STL, and allows for concrete time requirements.

The expressivities of both cLTL and CaTL are limited by the definition of *counting constraints* and *tasks*, respectively. For example, neither cLTL nor CaTL can require that “3 agents eventually reach region A” without requiring that the 3 agents reach region A at the same time. This can be restrictive. Consider, for example, a disaster relief scenario, where a team of robots needs to deliver supplies to an affected area. We require that enough supplies be delivered, i.e., enough robots eventually reach the affected area, rather than enough robots stay in the affected area at the same time. In fact, a robot is supposed to go on to the next task after drop off the supply without waiting for the other robots. To solve this problem, an extension of cLTL, called cLTL+, was proposed in [8], where a two-layer LTL structure was defined. However, cLTL+ can not specify concrete time requirements. To address this limitation, we extend CaTL to a novel logic called CaTL+, which has a two-layer structure similar to cLTL+. The authors of [9] extended STL with integral predicates, which also enables asynchronous satisfaction, but it can only specify how many times a service is needed. A CaTL+ task can specify the number of agents that need to satisfy an arbitrary STL formula. Another related work is CensusSTL [10], which is also a two-layer STL that refers to mutually exclusive subsets of a group, rather than capabilities of agents. Also, [10] focuses on inference of formulas from data, rather than control synthesis.

CaTL has both qualitative semantics, i.e., a specification is satisfied or violated, and quantitative semantics (known as robustness), which defines how much a specification is satisfied. The robustness of CaTL is an integer representing the minimum number of agents that can be removed from (added to) a given team in order to invalidate (satisfy) the given formula. Such a robustness metric is discontinuous and cannot represent how strongly each *task* is satisfied. In this paper, we define a novel quantitative semantics for CaTL+, called exponential robustness, which is continuous and measures how strongly a *task* is satisfied. A higher robustness indicates more agents reach the region and stay closer to the center for a longer time. The proposed exponential robustness also eliminates *masking* (i.e., only considering the most satisfying or violation points), which is a limitation of existing traditional robustness measures [11].

We also formulate and solve a centralized control synthesis problem from CaTL+. Control synthesis for cLTL [6], cLTL+ [8], CaTL [7] and STL with integral predicates [9] are solved in a graph environment using a (mixed) Integer

This work was partially supported by the NSF under grant IIS-2024606. DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited. This material is based upon work supported by the Under Secretary of Defense for Research and Engineering under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Under Secretary of Defense for Research and Engineering.

[1] Wenliang Liu and Calin Belta are with Boston University, Boston, MA, USA wliu97@bu.edu, cbelta@bu.edu

[2] Kevin Leahy and Zachary Serlin are with MIT Lincoln Laboratory, Lexington MA 02421, USA kevin.leahy@ll.mit.edu, zachary.serlin@ll.mit.edu

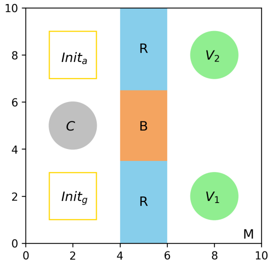


Fig. 1: An earthquake emergency response scenario. $M$ is the entire square workspace. $Init_a$ and $Init_g$ are the initial regions for the aerial and ground vehicles. $C$ is where the agents pick up supplies. $B$ is a bridge and the 2 rectangles $R$ correspond to the river. $V_1$ and $V_2$ are 2 affected villages.

Linear Program (ILP), where the controls are transitions between vertices in a graph. In this paper, we consider a continuous workspace shared by all agents. Each agent has its own discrete time dynamics with continuous state and control spaces. We propose a two-step optimization: a global optimization followed by a gradient-based local optimization to obtain the controls that maximize CaTL+ robustness.

## II. System Model and Notation

Let $|\mathcal{S}|$ be the cardinality of a set $\mathcal{S}$. We use bold symbols to represent trajectories and calligraphic symbols for sets. For $z \in \mathbb{R}$, we define $[z]_+ = \max\{0, z\}$ and $[z]_- = -[-z]_+$. Consider a team of agents labelled from a finite set $\mathcal{J}$. Let $Cap$ denote a finite set of agent capabilities. We assume that the agents operate in a continuous workspace $\mathcal{S} \subseteq \mathbb{R}^{n_s}$.

**Definition 1.** An *agent* $j \in \mathcal{J}$ is a tuple $A_j = \langle \mathcal{X}_j, x_j(0), Cap_j, \mathcal{U}_j, f_j, l_j \rangle$ where: $\mathcal{X}_j \subseteq \mathbb{R}^{n_{x,j}}$ is its state space; $x_j(0) \in \mathcal{X}_j$ is its initial state; $Cap_j \subseteq Cap$ is its finite set of capabilities; $\mathcal{U}_j \subseteq \mathbb{R}^{n_{u,j}}$ is its control space; $f_j : \mathcal{X}_j \times \mathcal{U}_j \to \mathcal{X}_j$ is a differentiable function giving the discrete time dynamics of agent $j$:

$$x_j(t+1) = f_j\big(x_j(t), u_j(t)\big), \quad t = 0, 1, \ldots, H-1, \quad (1)$$

where $x_j(t)$ and $u_j(t)$ are the state and control at time $t$, $H$ is a finite time horizon determined by the task (detailed later); $l_j : \mathcal{X}_j \to \mathcal{S}$ is a differentiable function that maps the state of agent $j$ to a point in the workspace shared by all agents (this enables heterogeneous state spaces):

$$s_j(t) = l_j\big(x_j(t)\big), \quad t = 0, 1, \ldots, H. \quad (2)$$

The trajectory of an agent $j$, called an *individual trajectory*, is a sequence $\mathbf{s}_j = s_j(0) \ldots s_j(H)$. We assume that $\cup_{j \in \mathcal{J}} Cap_j = Cap$.

Given a team of agents $\{A_j\}_{j \in \mathcal{J}}$, the *team trajectory* is defined as a set of pairs $\mathbf{S} = \{(\mathbf{s}_j, Cap_j)\}_{j \in \mathcal{J}}$, which captures all the *individual trajectories* with the corresponding capabilities. Let $\mathcal{J}_c = \{j \mid c \in Cap_j\}$ be the set of agent indices with capability $c$. Let $\mathbf{u}_j = u_j(0) \ldots u_j(H-1)$ be the sequence of controls for agent $j$.

**Example.** Consider an earthquake emergency response scenario. The workspace $\mathcal{S} \subset \mathbb{R}^2$ is shown in Fig. 1. There are 4 ground vehicles $j \in \{1,2,3,4\}$ and 2 aerial vehicles $j \in \{5,6\}$, totaling 6 robots indexed from $\mathcal{J} = \{1,2,3,4,5,6\}$ in the workspace. A river $R$ runs through this area and a bridge $B$ goes across the river. All ground vehicles are identical. The dynamics $f_j$, $j \in \{1,2,3,4\}$ are given by

$$\begin{aligned} p_{x,j}(t+1) &= p_{x,j}(t) + v_j(t)\cos\theta_j(t), \\ p_{y,j}(t+1) &= p_{y,j}(t) + v_j(t)\sin\theta_j(t), \\ \theta_j(t+1) &= \theta_j(t) + \omega_j(t), \end{aligned} \quad (3)$$

where the state $x_j$ is the 2D position and orientation $[p_{x,j}\ p_{y,j}\ \theta_j]$, the control $u_j$ is the forward and angular speed $[v_j\ \omega_j]$, the state space $\mathcal{X}_j = \mathcal{X}_g \subset \mathbb{R}^3$, the control space $\mathcal{U}_j = \mathcal{U}_g \subset \mathbb{R}^2$, the initial state $x_j(0)$ is a singleton randomly sampled in region $Init_g$ with randomly sampled orientation $\theta_j \in [\frac{1}{4}\pi, \frac{3}{4}\pi]$. The function $l_j(x_j) = [p_{x,j}\ p_{y,j}] = s_j$ maps the state of agent $j$ to a position in the workspace $\mathcal{S}$. The identical capabilities are given by $Cap_j = \{\text{“}Delivery\text{”}, \text{“}Ground\text{”}\}$, $j \in \{1,2,3,4\}$. All the aerial vehicles are identical. For $j \in \{5,6\}$, $f_j$ are given by

$$\begin{aligned} p_{x,j}(t+1) &= p_{x,j}(t) + v_{x,j}(t), \\ p_{y,j}(t+1) &= p_{y,j}(t) + v_{y,j}(t), \end{aligned} \quad (4)$$

where the state $x_j$ is the 2D position $[p_{x,j}\ p_{y,j}]$, the control $u_j$ is the speed $[v_{x,j}\ v_{y,j}]$, the state space $\mathcal{X}_j = \mathcal{X}_a \subset \mathbb{R}^2$, the control space $\mathcal{U}_j = \mathcal{U}_a \subset \mathbb{R}^2$, the initial state $x_j(0)$ is a singleton randomly sampled in the region $Init_a$, the identity mapping $l_j(x_j) = x_j = s_j$ maps the state of agent $j$ to a position in $\mathcal{S}$. The identical capabilities are given by $Cap_j = \{\text{“}Delivery\text{”}, \text{“}Inspection\text{”}\}$, $j \in \{5,6\}$. For this scenario, we assume the following set of requirements: (1) 6 agents with capability “*Delivery*” should pick up supplies from region $C$ within 8 time units; (2) 3 agents with capability “*Delivery*” should deliver supplies to the affected village $V_1$ within 25 time units, and 3 agents with capability “*Delivery*” should deliver supplies to the affected village $V_2$ within 25 time units; (3) the bridge might be affected by the earthquake so any agent with capability “*Ground*” cannot go over it until 2 agents with capability “*Inspection*” inspect it within 5 time units; (4) agents with capability “*Ground*” should always avoid entering the river $R$; (5) Since the load of the bridge is limited, at all times no more than 1 agent with capability “*Ground*” can be on $B$; (6) 6 agents with capability “*Delivery*” should always stay in region $M$.

## III. CaTL+ Syntax and Semantics

In this section we introduce *Capability Temporal Logic plus* (CaTL+), a logic used to specify requirements for multi-agent systems. CaTL+ has two layers: the *inner logic* and the *outer logic*. The proofs for all the results in this section are omitted and can be found in [12].

### A. Inner Logic

**Definition 2** (Syntax [13]). Given an individual trajectory s [1], the syntax of the inner logic can be defined as:

$$\varphi := True \mid \mu \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \mathbf{U}_{[a,b]} \varphi_2, \quad (5)$$

where $\varphi$, $\varphi_1$ and $\varphi_2$ are inner logic (STL) formulas, $\mu$ is a predicate in the form $h\big(s(t)\big) \geq 0$. We assume $h : \mathcal{S} \to \mathbb{R}$ is differentiable in this paper. $\neg$, $\wedge$, $\vee$ are the Boolean operators *negation*, *conjunction* and *disjunction* respectively. $U_{[a,b]}$ is the temporal operator *Until*, where $[a,b]$ is the time interval

[1] For simplicity we drop the subscript $j$ from $\mathbf{s}$.

containing all integers between $a$ and $b$ with $a, b \in \mathbb{Z}_{\geq 0}$. The temporal operators *Eventually* and *Always* can be defined as $\mathbf{F}_{[a,b]}\varphi = True\mathbf{U}_{[a,b]}\varphi$ and $\mathbf{G}_{[a,b]}\varphi = \neg\mathbf{F}_{[a,b]}\neg\varphi$.

The qualitative semantics of the inner logic, i.e., whether a formula $\varphi$ is satisfied by an individual trajectory s at time $t$ (denoted by $(\mathbf{s}, t) \models \varphi$) is same as STL [13]. In plain English, $\varphi_1\mathbf{U}_{[a,b]}\varphi_2$ means "$\varphi_2$ must be *True* at some time in $[a, b]$ and $\varphi_1$ must be $True$ at all time points before that". $\mathbf{F}_{[a,b]}\varphi$ states "$\varphi$ is *True* at some time point in $[a, b]$" and $\mathbf{G}_{[a,b]}\varphi$ states "$\varphi$ must be *True* at all time points in $[a, b]$".

### B. Outer Logic

The basic component of the outer logic, which with a slight abuse of terminology we will refer to as CaTL+, is a *task* $T = \langle\varphi, c, m\rangle$, where $\varphi$ is an inner logic formula, $c \in Cap$ is a capability, and $m$ is a positive integer. The syntax of CaTL+ is defined over a *team trajectory* $\mathbf{S}$ as:

$$\Phi := True \mid T \mid \neg\Phi \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \Phi_1\mathbf{U}_{[a,b]}\Phi_2, \quad (6)$$

where $\Phi$, $\Phi_1$ and $\Phi_2$ are CaTL+ formulas, $T = \langle\varphi, c, m\rangle$ is a task, and the other operators are the same as in the inner logic defined above. Before defining the qualitative semantics of CaTL+, we introduce a counting function $n(\mathbf{S}, c, \varphi, t)$:

$$n(\mathbf{S}, c, \varphi, t) = \sum_{j \in \mathcal{J}_c} I\big((\mathbf{s}_j, t) \models \varphi\big), \quad (7)$$

where $I$ is an indicator function, i.e., $I = 1$ if $(\mathbf{s}_j, t) \models \varphi$ and $I = 0$ otherwise. $n(\mathbf{S}, c, \varphi, t)$ captures how many *individual trajectories* $\mathbf{s}_j$ with capability $c$ in the *team trajectory* $\mathbf{S}$ satisfy an inner logic formula $\varphi$ at time $t$. The qualitative semantics of the *outer logic* is similar to the one of the *inner logic*, except for it involves *tasks* rather than predicates.

**Definition 3.** A *team trajectory* $\mathbf{S}$ satisfies a *task* $T = \langle\varphi, c, m\rangle$ at $t$, denoted by $(\mathbf{S}, t) \models T$, iff $n(\mathbf{S}, c, \varphi, t) \geq m$.

In words, a task $T = \langle\varphi, c, m\rangle$ is satisfied at time $t$ if and only if at least $m$ *individual trajectories* of agents with capability $c$ satisfy $\varphi$ at time $t$. The semantics for the other operators are identical with the ones in the *inner logic*. We denote the satisfaction of a CaTL+ formula $\Phi$ at time $t$ by a *team trajectory* $\mathbf{S}$ as $(\mathbf{S}, t) \models \Phi$. Note that specifying no more than $m$ agents with capability $c$ that could satisfy $\varphi$ can be formulated as $\Phi = \neg\langle\varphi, c, m + 1\rangle$. Let the *time horizon* of a CaTL+ formula $\Phi$, denoted by $hrz(\Phi)$, be the closest time point in the future that is needed to determine the satisfaction of $\Phi$. Note that cooperative inner logic is not allowed in CaTL+ since $\varphi$ is defined for one agent only.

**Example.** (continued) Reaching a circular or rectangular region can be formulated as an inner logic formula easily. For brevity, we use $s \in B$ (other regions are the same) to represent the inner logic formula of reaching region $B$. The requirements in the previous example can be formulated as CaTL+ formulas: (1) $\Phi_1 = \langle\mathbf{F}_{[0,8]}s \in C,$ "$Delivery$", $6\rangle$; (2) $\Phi_2 = \langle\mathbf{F}_{[0,25]}s \in V_1,$ "$Delivery$", $3\rangle \wedge \langle\mathbf{F}_{[0,25]}s \in V_2,$ "$Delivery$", $3\rangle$; (3) $\Phi_3 = \neg\langle s \in B,$ "$Ground$", $1\rangle\mathbf{U}_{[0,5]}\langle s \in B,$ "$Inspection$", $2\rangle$ ; (4) $\Phi_4 = \mathbf{G}_{[0,25]}\langle\neg(s \in R),$ "$Ground$", $4\rangle$; (5) $\Phi_5 = \mathbf{G}_{[0,25]}\neg\langle s \in B,$ "$Ground$", $2\rangle$; (6) $\Phi_6 = \mathbf{G}_{[0,25]}\langle s \in M,$ "$Delivery$", $6\rangle$. The overall specification for the system is $\Phi = \bigwedge_{i=1}^{6} \Phi_i$, with $hrz(\Phi) = 25$.

### C. CaTL+ Quantitative Semantics

The qualitative semantics defined above provides a *True* or *False* value, meaning that the CaTL+ formula is satisfied or not. In this section, we define its quantitative semantics (robustness), which is a real value that measures how much a formula is satisfied. We introduce a robustness metric for CaTL+, called exponential robustness. For the definition of an alternative robustness based on traditional robustness [11], we refer the reader to [12]. Using exponential robustness makes the optimization process for control synthesis easier.

For simplicity, we give the definition of CaTL+ robustness in a structured manner. We show that a robustness metric for CaTL+ can be captured by only the robustness for conjunction and *task*. According to the De Morgan law, disjunction can be replaced by conjunction and negation, i.e., $\Phi_1 \vee \Phi_2 = \neg(\neg\Phi_1 \wedge \neg\Phi_2)$. Meanwhile, the temporal operator "*always*" and "*eventually*" can be regarded as conjunction and disjunction evaluated over individual time steps. For any robustness metric, the robustness of a predicate $h\big(s(t)\big) \geq 0$ is $h\big(s(t)\big)$ and the robustness of $\neg\Phi$ is the negative of the robustness of $\Phi$ (see [14] for details).

We use $\eta(\mathbf{s}, \varphi, t)$ and $\eta(\mathbf{S}, \Phi, t)$ to denote the exponential robustness of the inner and outer logic. We begin our discuss from the definition of soundness:

**Definition 4** (Soundness)**.** A robustness metric $\eta(\mathbf{S}, \Phi, t)$ is *sound* if for any formula $\Phi$, $\eta(\mathbf{S}, \Phi, t) \geq 0$ iff $(\mathbf{S}, t) \models \Phi$.

Consider the conjunction over $M$ subformulas with robustness $\eta_1, \ldots, \eta_M$. Similar to [14], we first define an effective robustness measure, denoted by $\eta_i^{conj}$, $i = 1, \ldots, M$, for each subformula:

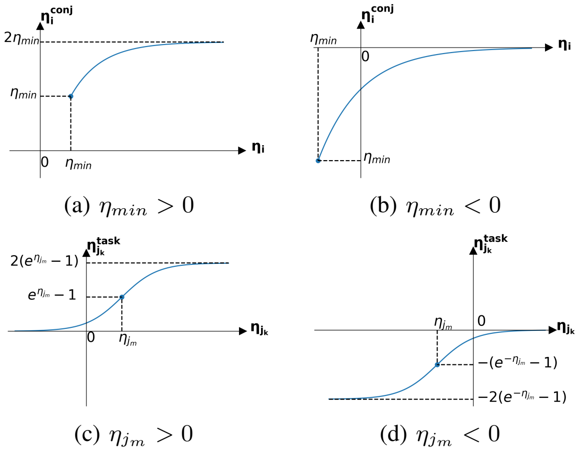
$$\eta_i^{conj} := \begin{cases} \eta_{min} e^{\frac{\eta_i - \eta_{min}}{\eta_{min}}} & \eta_{min} < 0 \\ \eta_{min}(2 - e^{\frac{\eta_{min} - \eta_i}{\eta_{min}}}) & \eta_{min} > 0 \\ 0 & \eta_{min} = 0 \end{cases} \quad (8)$$

where $\eta_{min} = \min(\eta_1, \ldots, \eta_M)$. The relation between $\eta_i^{conj}$ and $\eta_i$ is shown in Fig. 2a and Fig.2b. Intuitively, (8) ensures that $\eta_i^{conj}$ has the same sign with $\eta_{min}$, $\forall i = 1, \ldots, M$, and $\eta_i^{conj}$ increases monotonically with $\eta_i$. Note that $\eta_i^{conj} = \eta_i$ when $\eta_i = \eta_{min}$. We define the exponential robustness for conjunction as (9) which ensures soundness:

$$\mathcal{A}^{exp}(\eta_1, \ldots, \eta_M) = \beta\eta_{min} + (1 - \beta)\frac{1}{M}\sum_{i=1}^{M} \eta_i^{conj}, \quad (9)$$

where $\beta \in [0, 1]$ balances the contribution between $\eta_{min}$ and the mean of $\eta_i^{conj}$ (same sign as $\eta_{min}$). The exponential robustness turns to be the traditional robustness when $\beta = 1$.

Now consider a *task* $T = \langle\varphi, c, m\rangle$. For brevity, let $\eta_j = \eta(\mathbf{s}_j, \varphi, t)$ when $\varphi$ and $t$ are clear from the context. We reorder $\{\eta_j\}_{j \in \mathcal{J}_c}$ from the largest to the smallest, i.e., $\eta_{j_1} \geq \ldots \geq \eta_{j_m} \geq \ldots \geq \eta_{j_n}$, where $j_k \in \mathcal{J}$, $k = 1, \ldots, n$,



(a) $\eta_{min} > 0$

(b) $\eta_{min} < 0$

(c) $\eta_{j_m} > 0$

(d) $\eta_{j_m} < 0$

Fig. 2: (a), (b): Relation between $\eta_i^{conj}$ and $\eta_i$ while holding $\eta_{min}$ constant and $\eta_i \neq \eta_{min}$. (c), (d): Relation between $\eta_{j_k}^{task}$ and $\eta_{j_k}^{task}$ ($\alpha = 1$) while holding $\eta_{j_m}$ constant and $k \neq m$.

$n = |\mathcal{J}_c|$. Note that $\eta_{j_m}$ is the critical $m^{th}$ largest robustness. We define another effective robustness $\eta_{j_k}^{task}$ for each $j_k$, $k = 1, \ldots, |\mathcal{J}_c|$:

$$\eta_{j_k}^{task} := \begin{cases} \dfrac{2\alpha(e^{\eta_{j_m}}-1)}{1+e^{-\alpha(\eta_{j_k}-\eta_{j_m})}} & \eta_{j_m} > 0, \\ \dfrac{-2\alpha(e^{-\eta_{j_m}}-1)}{1+e^{\alpha(\eta_{j_k}-\eta_{j_m})}} & \eta_{j_m} \leq 0, \end{cases} \tag{10}$$

where $\alpha > 0$. The relation between $\eta_{j_k}^{task}$ and $\eta_{j_k}$ is shown in Fig. 2c and Fig. 2d ($\alpha = 1$). Similar to conjunction, (10) ensures that $\eta_{j_k}^{task}$ has the same sign with the critical $\eta_{j_m}$, $\forall k = 1, \ldots, |\mathcal{J}_c|$ and $\eta_{j_k}^{task}$ increases monotonically with $\eta_{j_k}$. Note that $\eta_{j_k}^{task} = sgn(\eta_{j_m})\alpha(e^{|\eta_{j_m}|}-1)$ when $k = m$. Again, we define the exponential robustness for a *task* as the mean of $\eta_{j_k}^{task}$ to ensure soundness:

$$\eta(\mathbf{S}, \langle\varphi, c, m\rangle, t) = \frac{1}{|\mathcal{J}_c|}\sum_{k=1}^{|\mathcal{J}_c|}\eta_{j_k}^{task}. \tag{11}$$

When $\alpha \to \infty$, the exponential robustness of a task only depends on $\eta_{j_m}$. The exponential robustness for CaTL+ is recursively constructed from (9) and (11). In the following, we discuss the properties of the exponential robustness.

**Proposition 1.** Exponential robustness for CaTL+ is sound.

In an optimal control problem, such as the one considered in Sec. IV, it is desirable to have a differentiable robustness allowing for gradient based optimization methods.

**Proposition 2.** The exponential robustness $\eta(\mathbf{S}, \Phi, t)$ is continuous everywhere and differentiable almost everywhere with respect to individual trajectories $\mathbf{s}_j$, $j \in \mathcal{J}$.

Although exponential robustness is not differentiable everywhere, in a numerical optimization process it rarely gets to the non-differentiable points. Moreover, these points are semi-differentiable, so even if they are met, we can still use the left or right derivative to keep the optimization running.

The most important advantage of the exponential robustness over traditional robustness is that it eliminates *masking*. In short, the traditional robustness only takes into account the minimum robustness in conjunction. All the other subformulas have no contribution to the overall robustness. For example, consider formula $\mathbf{F}_{[0,5]}(s > 3)$. Two trajectories $1, 1, 1, 1, 5$ and $1, 2, 3, 4, 5$ get the same traditional robustness score of 2, though it is obvious that the later is more robust under disturbances. The exponential robustness addresses this *masking* problem by taking into account the robustness for all subformulas, all time points, and all agents. As a result, the exponential robustness rewards the trajectories that satisfy the requirements at more time steps and with more agents. From an optimization point of view, our goal is to synthesize controls that maximize the CaTL+ robustness. If we use the traditional robustness, at each optimization step we can only modify the most satisfying or violating points. This may decelerate the optimization speed or even result in divergence. The exponential robustness makes the robustness-based control synthesis easier, and makes the results more robust. Formally, we have:

**Definition 5.** [mask-eliminating] The robustness of an operator $\mathcal{O}(\eta_1, \ldots, \eta_M)$ has the *mask-eliminating* property if it is differentiable almost everywhere, and wherever it is differentiable, it satisfies:

$$\frac{\partial\mathcal{O}(\eta_1, \ldots, \eta_M)}{\partial\eta_i} > 0, \quad \forall i = 1, \ldots, M, \tag{12}$$

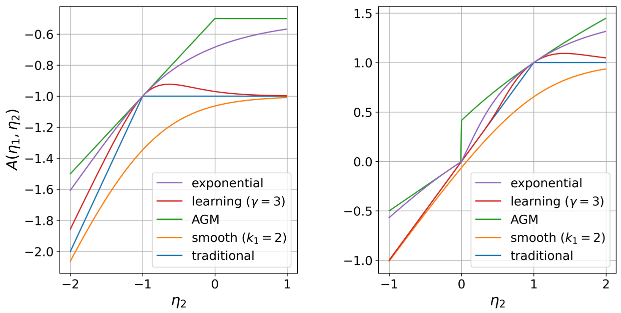
A robustness metric of CaTL+ has the mask-eliminating property if both conjunction and *task* satisfy (12).

**Proposition 3.** Exponential robustness of CaTL+ has the mask-eliminating property.

The *mask-eliminating* property of exponential robustness tells us that the increase of any component in conjunction or *task* results in the increase of the overall robustness.

**Remark 1.** By standard derivations, it can also be proved that, $\partial\mathcal{A}^{exp}/\partial\eta_{min} > \partial\mathcal{A}^{exp}/\partial\eta_i$, $\forall\eta_i \neq \eta_{min}$. The further $\eta_i$ is from $\eta_{min}$, the smaller the partial derivative will be. Meanwhile, $\partial\eta(\mathbf{S}, \langle\varphi, c, m\rangle, t)/\partial\eta_{j_m} \geq \partial\eta(\mathbf{S}, \langle\varphi, c, m\rangle, t)/\partial\eta_{j_k}$, $\forall k \neq m$. The further $\eta_{j_k}$ is from $\eta_{j_m}$, the smaller the partial derivative will be. This is helpful because $\eta_{min}$ and $\eta_{j_m}$ are the most critical components which decide the satisfaction of the formula.

**Remark 2.** The exponential robustness for conjunction itself forms a novel robustness of STL, which has desired properties including soundness and mask-eliminating (the conjunction satisfies (12)). Other robustness metrics including Arithmetic-Geometric Mean (AGM) robustness [15] and learning robustness [14] are also sound and partially solve the *masking* problem, but none of them satisfy the *mask-eliminating* property (as shown in Fig. 3). The smooth max-min robustness from [16], [17] has the mask-eliminating property, but loses soundness (in a necessary and sufficient sense). To the best of our knowledge, exponential robustness is the first that satisfies both of these two properties. Sample behaviors of $\mathcal{A}^{exp}(\eta_1, \eta_2)$ for different $\eta_1$ and $\eta_2$ are depicted in Fig. 3. We can see that traditional and AGM robustness have 0 partial derivatives and learning robustness



(a) $\eta_1 = -1$

(b) $\eta_1 = 1$

Fig. 3: Robustness of the conjunction $\mathcal{A}(\eta_1, \eta_2)$ for the traditional [11], AGM [15], learning [14], smooth max-min [17] and exponential ($\beta = 0$) metrics as a function of $\eta_2$ with fixed values of $\eta_1$.

has negative partial derivatives at some points, while smooth max-min robustness violates soundness.

### D. *Relationship between CaTL and CaTL+*

The syntax and qualitative semantics of CaTL are similar to the *outer logic* of CaTL+, with two main differences. The first is the definition of a *task*. In CaTL, a task is defined as a tuple $T' = \langle d,\ \pi,\ \{c_i, m_i\}_{i\in I_T}\rangle$, where $d$ is a duration of time, $\pi$ is an atomic proposition specifying a region, $c_i$ is a capability and $m_i$ is a positive integer. A CaTL *task* is satisfied if, between $[0, d]$, each of the regions labeled as $\pi$ contains at least $m_i$ agents with capability $c_i$ for all $\{c_i\}_{i\in I_T}$. There is no *inner logic* in CaTL. Second, the ILP encoding requires that CaTL formulas contain no negation, while CaTL+ may contain negations as in (5) and (6).

**Proposition 4.** With a given set of agents and the corresponding capabilities, specifications given by CaTL are a proper subset of specifications given by CaTL+.

Intuitively, a CaTL *task* can only specify the number of agents that should "always" exist in a region in a duration of time. All the other temporal and Boolean operators have to be outside the task. In contrast, CaTL+ is more expressive because a CaTL+ *task* can specify a full STL formula in the inner logic. E.g., a CaTL+ task can be $T = \langle \mathbf{F}_{[a,b]}\varphi_\pi,\ c,\ m\rangle$. This *task* is satisfied if $m$ agents satisfy $\varphi_\pi$ in $[a, b]$ synchronously or asynchronously. A CaTL formula can only specify $\mathbf{F}_{[a,b-1]}\langle 1, \pi, \{c, m\}\rangle$ which requires synchronous satisfaction. Note that by introducing new capability to each agent, the CaTL+ task $T = \langle \mathbf{F}_{[a,b]}\varphi_\pi,\ c,\ m\rangle$ can be transformed into an equivalent CaTL formula with combinatorially many conjunctions and disjunctions. However, the resulting CaTL formula will be very complex, which might make the control synthesis intractable.

Moreover, CaTL+ formula can contain negations, which is absent from CaTL. So CaTL+ can specify more specifications including "no more than $m$ agents could visit a region".

The robustness definitions of CaTL+ and CaTL are also very different. CaTL robustness represents the minimum number of agents that can be removed from a given team in order to invalidate the given formula. It is not directly related to the trajectory of each agent. It is *sound*, but without the *continuity*, *differentiability* and *mask-eliminating* properties of CaTL+ (exponential) robustness.

Another difference is that CaTL is defined on a discrete graph environment. The controls synthesized using ILP is a sequence of transitions between the vertices of the graph. In contrast, CaTL+ is defined on a continuous workspace. Each agent has its own dynamics and continuous control space.

## IV. Control Synthesis using CaTL+

In this section, we formulate and solve a CaTL+ control synthesis problem. In order to avoid unnecessary motions of the agents, we introduce a cost function over the controls. The overall optimization problem combines minimizing this cost with maximizing the CaTL+ (exponential) robustness.

**Problem 1.** Given a workspace $\mathcal{S}$, a set of agents $\{A_j \mid j \in \mathcal{J}\}$, a CaTL+ formula $\Phi$ with time horizon $H$, and a weighted cost function $C(\cdot) \geq 0$, find a control sequence $\mathbf{u}_j$ for each agent that maximizes the objective function:

$$\begin{aligned} \max_{\mathbf{u}_j, j\in\mathcal{J}} \quad & \eta(\mathbf{S}, \Phi, 0) - \frac{[\eta(\mathbf{S}, \Phi, 0)]_+}{\gamma} \cdot \sum_{j\in\mathcal{J}} C(\mathbf{u}_j) \\ \text{s.t.} \quad & x_j(t+1) = f_j(x_j(t), u_j(t)), \\ & u_j(t) \in \mathcal{U}_j,\ t = 0, \dots, H-1, \\ & l_j(x_j(t)) = s_j(t),\ j \in \mathcal{J},\ t = 0, \dots, H, \end{aligned} \tag{13}$$

where $\gamma$ is a parameter satisfying

$$\gamma \geq \sup_{\mathbf{u}_j\in\mathcal{U}_j, j\in\mathcal{J}} \sum_{j\in\mathcal{J}} C(\mathbf{u}_j). \tag{14}$$

Due to the soundness of the robustness, $\eta(\mathbf{S}, \Phi, 0) < 0$ means that the CaTL+ formula $\Phi$ is not satisfied. In such situations, we focus on increasing the robustness without considering the cost $C$. When $\Phi$ is satisfied, i.e., $\eta(\mathbf{S}, \Phi, 0) > 0$, we try to minimize the cost at the same time. But minimizing the cost will never override the priority of satisfying the specification, because when (14) is true, the cost cannot change the sign of the objective function. Hence, a positive objective ensures the satisfaction of the specification $\Phi$.

Next, we propose a solution to Pb. 1. The objective function in (13) is highly non-convex, which means that there might exist many local optima. To avoid getting stuck at such points, we apply a two-step optimization: a global optimization followed by a local optimization. The result of the global optimization provides a good initialization for the local search, so the local optimizer is able to reach a point near the global optimum (for a highly non-convex function obtaining the exact global optimum is very difficult). Specifically, for the global optimizer, we use Covariance Matrix Adaptation Evaluation Strategy (CMA-ES) [18], which is a derivative-free, evalution-based optimization approach. Note that CMA-ES is primarily a local optimization approach, but it has also been reported to be reliable for global optimization with large population size [19]. For the local optimizer, we apply gradient-based method Sequential Quadratic Programming (SQP) [20]. An important issue in gradient-based optimization is to compute the gradient efficiently. STLCG [21]

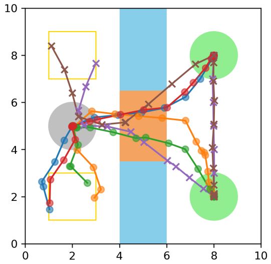


Fig. 4: Trajectories obtained solving Pb. 1. Brown and purple trajectories are for the aerial vehicles, the other trajectories are for the ground vehicles. Each agent's positions between adjacent time points are connected by straight lines.

provides a way to compute the gradient of STL robustness. We adapt STLCG such that it works for the exponential robustness of CaTL+. Hence, we can obtain the gradient of the objective in (13) automatically and analytically.

## V. Simulation Results

In this section we evaluate our algorithm on the earthquake emergency response scenario used as a running example throughout the paper. All algorithms were implemented in Python on a computer with 3.50GHz Core i7 CPU and 16GB RAM. For CMA-ES we used the pymoo [22] library. For SQP, we used the scipy package [23].

Consider the map shown in Fig. 1. The control constraints $\mathcal{U}_g$ and $\mathcal{U}_a$ for both ground and aerial vehicles are set to $[-1,1] \times [-1,1]$. We used $l^2$-norm as the cost function in (13), i.e., $C(\mathbf{u}_j) = \|\mathbf{u}_j\|_2$. Let $\gamma = 1000$, satisfying (14).

We applied CMA-ES followed by SQP to solve Pb. 1. Fig. 4 shows the resulting *individual trajectories* for each agent. The resulting *team trajectory* satisfies the specification $\Phi$. Note that the agents do not need to stay in a region at the same time to satisfy a *task* like $\Phi_1 = \langle \mathbf{F}_{[0,8]} s \in C,\ \text{“}Delivery\text{”},\ 6\rangle$ or $\Phi_2 = \langle \mathbf{F}_{[0,25]} s \in V_1,\ \text{“}Delivery\text{”},\ 3\rangle \wedge \langle \mathbf{F}_{[0,25]} s \in V_2,\ \text{“}Delivery\text{”},\ 3\rangle$. In fact, on the premise of satisfying $\Phi_1$, the two aerial vehicles depart from region $C$ to inspect the bridge before all the ground vehicles reach region $C$. Moreover, both villages $V_1$ and $V_2$ are visited by 4 agents though the requirement is 3, which makes the exponential robustness higher.

## VI. Conclusion and Future Work

We introduced a new logic called CaTL+, which is convenient to specify requirements for multi-agent systems over continuous workspace and is strictly more expressive than CaTL. We defined a quantitative semantics for CaTL+, called exponential robustness, which is sound, differentiable almost everywhere, and has the mask-eliminating property. A two-step optimization strategy was proposed for control synthesis from CaTL+ formula. The simulation results illustrate the efficacy of our approach. One limitation is that we do not consider inter-agent collisions and the behavior of the system between adjacent time points. In future work, we will investigate incorporating a lower level controller that ensures the correct dense-time behavior and collision avoidance. We also plan to employ more efficient optimizers.

## References

[1] Y. Chen, X. C. Ding, and C. Belta, “Synthesis of distributed control and communication schemes from global ltl specifications,” in *2011 50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, 2011, pp. 2718–2723.

[2] P. Schillinger, M. Bürger, and D. V. Dimarogonas, “Simultaneous task allocation and planning for temporal logic goals in heterogeneous multi-robot systems,” *The international journal of robotics research*, vol. 37, no. 7, pp. 818–838, 2018.

[3] Y. Kantaros and M. M. Zavlanos, “Stylus*: A temporal logic optimal control synthesis algorithm for large-scale multi-robot systems,” *The International Journal of Robotics Research*, vol. 39, no. 7, pp. 812–836, 2020.

[4] X. Luo, Y. Kantaros, and M. M. Zavlanos, “An abstraction-free method for multirobot temporal logic optimal control synthesis,” *IEEE Transactions on Robotics*, 2021.

[5] Z. Liu, B. Wu, J. Dai, and H. Lin, “Distributed communication-aware motion planning for networked mobile robots under formal specifications,” *IEEE Transactions on Control of Network Systems*, vol. 7, no. 4, pp. 1801–1811, 2020.

[6] Y. E. Sahin, P. Nilsson, and N. Ozay, “Provably-correct coordination of large collections of agents with counting temporal logic constraints,” in *2017 ACM/IEEE 8th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2017, pp. 249–258.

[7] K. Leahy, Z. Serlin, C.-I. Vasile, A. Schoer, A. M. Jones, R. Tron, and C. Belta, “Scalable and robust algorithms for task-based coordination from high-level specifications (scratches),” *IEEE Transactions on Robotics*, 2021.

[8] Y. E. Sahin, P. Nilsson, and N. Ozay, “Synchronous and asynchronous multi-agent coordination with cltl+ constraints,” in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 335–342.

[9] A. T. Buyukkocak, D. Aksaray, and Y. Yazıcıoğlu, “Planning of heterogeneous multi-agent systems under signal temporal logic specifications with integral predicates,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1375–1382, 2021.

[10] Z. Xu and A. A. Julius, “Census signal temporal logic inference for multiagent group behavior analysis,” *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 264–277, 2016.

[11] A. Donzé and O. Maler, “Robust satisfaction of temporal logic over real-valued signals,” in *International Conference on Formal Modeling and Analysis of Timed Systems*. Springer, 2010, pp. 92–106.

[12] W. Liu, K. Leahy, Z. Serlin, and C. Belta, “Robust multi-agent coordination from catl+ specifications,” *arXiv preprint arXiv:2210.01732*, 2022.

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

[14] P. Varnai and D. V. Dimarogonas, “On robustness metrics for learning stl tasks,” in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 5394–5399.

[15] N. Mehdipour, C.-I. Vasile, and C. Belta, “Arithmetic-geometric mean robustness for control from signal temporal logic specifications,” in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 1690–1695.

[16] Y. V. Pant, H. Abbas, and R. Mangharam, “Smooth operator: Control using the smooth robustness of temporal logic,” in *2017 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2017, pp. 1235–1240.

[17] Y. Gilpin, V. Kurtz, and H. Lin, “A smooth robustness measure of signal temporal logic for symbolic control,” *IEEE Control Systems Letters*, vol. 5, no. 1, pp. 241–246, 2020.

[18] N. Hansen and A. Ostermeier, “Completely derandomized self-adaptation in evolution strategies,” *Evolutionary computation*, vol. 9, no. 2, pp. 159–195, 2001.

[19] N. Hansen and S. Kern, “Evaluating the cma evolution strategy on multimodal test functions,” in *International Conference on Parallel Problem Solving from Nature*. Springer, 2004, pp. 282–291.

[20] D. P. Bertsekas, “Nonlinear programming,” *Journal of the Operational Research Society*, vol. 48, no. 3, pp. 334–334, 1997.

[21] K. Leung, N. Aréchiga, and M. Pavone, “Back-propagation through signal temporal logic specifications: Infusing logical structure into gradient-based methods,” in *Workshop on Algorithmic Foundations of Robotics*, 2020.

[22] J. Blank and K. Deb, “pymoo: Multi-objective optimization in python,” *IEEE Access*, vol. 8, pp. 89 497–89 509, 2020.

[23] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright *et al.*, “Scipy 1.0: fundamental algorithms for scientific computing in python,” *Nature methods*, vol. 17, no. 3, pp. 261–272, 2020.