Logic for Timed Agent Network Topologies

Clovis Eberhart[1,2], James Haydon[1], Jérémy Dubut[1,2], Ahmet Cetinkaya[1], and Sasinee Pruekprasert[1]

***Abstract*— We define μTGL, a spatio-temporal logic with fixed points and first-order agent quantification whose expressive power allows the definition of topological properties of networks of communicating agents. The existence of temporal operators and fixed points requires particular care when defining its semantics. We also give a streaming monitoring algorithm for μTGL formulas. Finally, we demonstrate the logic's usefulness on an example, where we monitor a complex property that ensures resilient consensus.**

I. INTRODUCTION

With more multi-agent systems being used in practice, the need to study topological properties of automated multi-agent networks has come to light. The key challenge in studying multi-agent networks arises from complex interactions among the agents, especially when many agents are involved. Efficiently describing desired properties for such systems is a non-trivial task that requires mathematical formalism.

Temporal logics, which can express system properties dependent on time, have been proved useful for monitoring specifications. In particular, Linear Temporal Logic (LTL) and Signal Temporal Logic (STL) have been extensively used to monitor cyber-physical systems [1]. However, those logics are not meant to model spatial relations between agents, which are crucial to monitoring some interesting properties. For instance, the distances between the different agents can influence the overall behaviour of the network.

Several previous studies proposed extensions of well-known temporal logics to model spatio-temporal behaviours of multi-agent systems. Signal Spatio-Temporal Logic (SSTL) [2], [3] extends STL with spatial modalities, and imposes bounds on spatial and temporal operators. Spatial-Temporal Logic (SpaTeL) [4] extends STL with spatial component based on Tree Spatial Superposition Logic [5], resulting in a logic suitable for robotic swarm control [6]. Counting linear temporal logic (cLTL) [7] can specify desired behaviours for multi-robot systems with constraints on the number of agents satisfying given LTL properties. To specify topological properties of networks, we also need a fixed-point operator (such as that from the μ-calculus [8]), which none of these logics have.

The authors are supported by ERATO HASUO Metamathematics for Systems Design Project (No. JPMJER1603), JST. J. D. is also supported by Grant-in-aid No. 19K20215, JSPS. A. C. is also supported by Grant-in-aid No. 20K14771, JSPS. S. P. is also supported by Grant-in-aid No. 21K14191, JSPS.

[1]The authors are with the National Institute of Informatics, Hitotsubashi 2-1-2, Tokyo 101-8430, Japan {eberhart, jhaydon, dubut, cetinkaya, sasinee}@nii.ac.jp.

[2]C. E. and J. D. are also affiliated with the Japanese-French Laboratory for Informatics, IRL 3527.

This paper proposes a new logic μTGL (Temporal Graph Logic with fixed points). It has several features needed to express interesting properties of communicating agent networks. It has temporal operators, as well as a "connectivity" operator that has a strong spatial flavour. It also contains first-order agent quantification, which is necessary to express the properties of agent networks with an unknown number of agents. Finally, our logic contains fixed points, which allows the definition of recursive properties that cannot be expressed by spatio-temporal logics proposed in the previous studies. Using these constructs, we can express properties such as the connectivity of the communication graph of the network or the existence of communication chains between agents.

The presence of fixed points and temporal operators in the logic implies that we may need to look at an infinite portion of the agents' trajectories to be able to decide whether a property is satisfied or not. This leads us to come up with a novel semantics that takes into account how much of the signal we are allowed to see. This semantics allows us to define a streaming monitoring algorithm for μTGL.

Our logic draws inspiration from many other logics. The temporal part is inherited from continuous-time temporal logics such as Metric Temporal Logic (MTL) [9], Metric Interval Temporal Logic (MITL) [10], and STL [11]. The diamond modality is inspired by spatial operators from spatio-temporal logics [2], [3]. The fixed point operator is inspired from fixed-point logics, and in particular the μ-calculus [8]. Since μTGL features a diamond modality corresponding to connectivity of agents, our logic is also related to temporal graph logics [12]. However, such logics describe graph rewriting properties rather than agent networks, so they are quite different in practice. Moreover, our semantics uses three-valued logic [13], more precisely, Kleene's logic of indeterminacy. It enables us to handle indeterminacy in the monitoring algorithm that we propose.

The rest of the paper is structured as follows. In Section II, we define the syntax and semantics of μTGL, give examples of especially useful formulas, and describe a monitoring algorithm for it. In Section III, we give several examples of formulas that showcase the expressivity of μTGL. In Section IV, we give a monitoring algorithm for μTGL formulas. In Section V, we design a μTGL formula that describes a property that ensures resilient consensus in an agent network, and apply our monitoring algorithm on a specific scenario of mobile robots surveilling a perimeter.

Notations: We denote by $\mathbb{N}$ the set of natural numbers, by $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, $\overline{\mathbb{R}}$, $\overline{\mathbb{R}_{\geq 0}}$ the sets of real numbers, non-negative real numbers, and their extensions with $\pm\infty$ and $+\infty$ respectively. We denote by $\mathbb{2}$ the set $\{\top, \bot\}$ of booleans. We denote by

3 the set of booleans augmented with an *indeterminate* $\mathtt{I}$, and equipped with the order $\bot < \mathtt{I} < \top$. It is equipped with the usual functions on 2: $\vee$ and $\wedge$ are respectively $\min$ and $\max$, and negation maps $\mathtt{I}$ to $\mathtt{I}$. *Uncertain sets* are functions $X \to 3$. They are equipped with the same order and operations as 3, defined pointwise. We sometimes use set notation $\{x \in X \mid \varphi(x)\}$ to denote an uncertain set, by which we mean that no element is mapped to $\mathtt{I}$.

II. μTGL

In this section, we define μTGL, its syntax and semantics, give examples of its uses, and describe a monitoring algorithm for it.

We denote by A the set of agents, which we assume finite. Agents move in a metric space (M, d) – usually $\mathbb{R}^n$ equipped with the Euclidean distance.

A. Syntax

We start by defining the syntax of μTGL, that is the set of μTGL formulas. From now on, we assume given sets $\mathbb{X}$ of *fixed-point variables* and $\mathbb{A}$ of *agent variables*, both infinite countable. We also fix a set P of predicates on M, i.e. functions from M to 2.

Definition 1 (Agent formulas)**.** Agent formulas *are given by the following grammar:*

$$\begin{aligned}
\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid & \quad \textit{(propositional logic)} \\
\Diamond_D\, \varphi \mid F_T\, \varphi \mid & \quad \textit{(spatio-temporal logic)} \\
X \mid \mu X.\, \varphi \mid H_h\, \varphi \mid & \quad \textit{(fixed points)} \\
a \mid \exists a.\, \varphi & \quad \textit{(quantification),}
\end{aligned}$$

where $a \in \mathbb{A}$ is an agent variable, $p \in P$ is a predicate, T and D are intervals of $\mathbb{R}_{\geq 0}$, $h \in \mathbb{R}_{\geq 0}$ is a non-negative real, and $X \in \mathbb{X}$ is a fixed-point variable. We impose the usual condition that, for $\mu X.\, \varphi$ to be well-formed, all occurrences of X must be positive *in φ (i.e., they occur under an even number of negations).*

Intuitively, agent formulas hold *for some agents*, or equivalently represent a set of agents (those agents for which the formula holds). This intuition is made precise in Definition 6. The propositional logic constructs are self-explanatory. The $F_T\, \varphi$ construct is the temporal *eventually* modality: it holds for agent a if there is a point in time $t \in T$ when φ holds for a. The $\Diamond_D\, \varphi$ construct is the *diamond* modality, an equivalent to eventually for agent connectivity: it holds for a if there is b "connected" to it (i.e., within a certain distance from it) for which φ holds. The $\mu X.\, \varphi$ construct is the *least fixed point* construct. The only unusual construct is the *horizon* construct H_T, whose use we will explain in Section II-B.

We use the usual syntactic sugar for propositional logic $\bot$, $\varphi \vee \psi$, $\varphi \to \psi$, as well as $G_T\, \varphi \equiv \neg F_T \neg\varphi$ and $\Box_D\, \varphi \equiv \neg \Diamond_D \neg\varphi$ for spatio-temporal operators, $\forall a.\, \varphi \equiv \neg\exists a.\, \neg\varphi$ for universal quantification, and $\nu X.\, \varphi \equiv \neg\mu X.\, \neg\varphi[X \to \neg X]$ for greatest fixed point. We also use $F\varphi$ and $G\,\varphi$ when the time interval is $[0, +\infty)$.

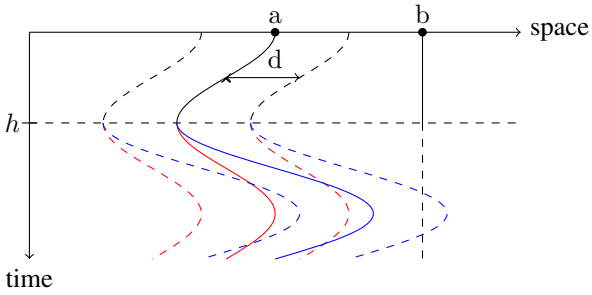


Fig. 1. Known and possible trajectories of agents

Free and bound variables are defined as usual. A formula is *closed* if all $X \in \mathbb{X}$ in φ are bound.

Definition 2 (Truth formulas)**.** Truth formulas *are given by the following grammar:*

$$\Phi ::= \top \mid \varphi \leq \varphi \mid \neg\Phi \mid \Phi \wedge \Phi \mid F_T\, \Phi \mid \exists a.\, \Phi,$$

where φ and ψ are closed agent formulas in $\varphi \leq \psi$, and the rest is as in Definition 1.

Truth formulas use the same constructs as agent formulas except fixed points and diamond, and also allows for comparison of agent formulas through $\varphi \leq \psi$.

We also introduce some syntactic sugar for equality on agent formulas: $\varphi = \psi \equiv (\varphi \leq \psi) \wedge (\psi \leq \varphi)$, $a \neq b \equiv \neg(a = b)$.

Example 3 (Connectedness of communication graph)**.** *We consider a network of communicating agents where agents can communicate up to some distance* d*. The* communication graph *G_t at time t of such a network is a graph whose nodes are agents, and there is an edge between* a *and* b *if they are within communication range at time* t*. The following formula expresses connectivity of the communication graphs between agents at all times:*

$$G\, \forall a.\, (\top \leq \mu X.\, (a \vee \Diamond_{[0,\mathrm{d}]}\, X)).$$

The subformula $\mu X.\, (a \vee \Diamond_{[0,\mathrm{d}]}\, X)$ represents the set of agents that can be reached in the communication graph from the agent represented by a. Therefore, the whole formula expresses the fact that, at all times, all agents are reachable from any other agent, so the communication graph is connected. Another way to state the same property is as follows:

$$\Phi_{\text{con}} \;=\; G\, \forall a, b.\, a \leq \mu X.\, (b \vee \Diamond_{[0,\mathrm{d}]}\, X).$$

We will show later in Example 8 that this formula indeed has its intended meaning.

B. Semantics

In this section, we give a semantics to agent and truth formulas. In general, a semantics takes a formula and a model (in our case, a signal of all agent positions at all times), and returns a boolean: whether the formula is true or false in this model. In our case, however, we need to be slightly more subtle: because of the presence of fixed points and temporal operators, we may need to look arbitrarily far into the future to prove or disprove a formula. Let us show an example.

Example 4 (Communcation chain)**.** *Let us consider the same setting as in Example 3, but with agents re-broadcasting messages sent by other agents for duration t. The following formula expresses the existence, at all times and for all pairs (a, b) of agents, of a* communication chain *from a to b, that is a list of agents that can communicate within time t to transmit a message from a to b:*

$$\Phi_{\text{ch}} \;=\; G\,\forall a, b.\, a \leq \mu X.\,(b \vee F_{[0,t]}\,\Diamond_{[0,\mathrm{d}]}\,X). \qquad (1)$$

However, deciding whether Φ_{ch} holds may require knowing an arbitrarily long part of the signal (even if we restrict to $G_{[0,T]}$). Let us assume that the whole network consists of two agents a *and* b *that move on a line, as in Fig. 1. If we look at the signal up to time h, we can neither prove nor disprove the existence of a communication chain from* a *to* b*: if* a *follows the red trajectory in the future, there will be no such chain, but there will be one if it follows the blue one (namely* $\mathrm{a} \to \mathrm{a} \to \ldots \to \mathrm{a} \to \mathrm{b}$*).*

There are two problems with this. First, we may need infinite time before being able to compute the semantics, which is not feasible in practice. Second, and more importantly, we may want to express properties such as "there exists a communication chain from a to b such that they can communicate within time t", which cannot be expressed with the current examples.

As a first step towards fixing this, we use a three-valued logic, rather than a boolean one. The semantics of a formula can thus also be indeterminate (I), meaning that we do not have enough information to determine its actual value.

This single change does not however solve the problem. Indeed, in the case we describe in Example 4, the semantics of Φ_{ch} would be I unless we look at the whole time interval. We therefore introduce the *horizon* construct, whose purpose is to "give up" on trying to prove that a formula holds: if the semantics of φ is undetermined and "enough time has passed", then we "stop searching" and say it is false.

Example 5 (Bounded-time communication)**.** *Building on Example 4, we express the fact that all agents can communicate with one another within time t. This property can be expressed by*

$$G\,\forall a, b.\, a \leq H_t\,\mu X.\,(b \vee F_{[0,t]}\,\Diamond_{[0,\mathrm{d}]}\,X). \qquad (2)$$

This is the same formula as (1)*, with an added* horizon *construct H_t. The intuitive meaning of this construct is that it forbids chains that are longer than t.*

The semantics of a formula is relative to a model. Our models here are *signals*. Given a set A of agents, a signal is a function $\sigma\colon (\mathsf{A} \times \mathbb{R}_{\geq 0}) \to M$, which represents the fact that agents evolve through time in space M.

The final piece of structure we need before we can define the semantics, is contexts. A *context* is a partial function that maps $\mathbb{X}$ to functions $(\mathbb{R}_{\geq 0} \times \mathbb{R}) \to \mathsf{A} \to \mathsf{3}$ and $\mathbb{A}$ to A.

Definition 6 (Semantics)**.** *The semantics $[\![\varphi]\!]_\rho(\sigma, t, h)$ of an agent formula φ on signal σ, at time $t \in \mathbb{R}_{\geq 0}$, with horizon $h \in \overline{\mathbb{R}}$, and in context ρ is an uncertain set of agents defined inductively as in Fig. 2. The semantics $[\![\Phi]\!]_\rho(\sigma, t, h) \in \mathsf{3}$ of truth formulas is also given in Fig. 2.*

The existence of fixed points is ensured by the fact that variables appear positively in fixed-point formulas. When writing $[\![\varphi]\!]_\rho(\sigma, t, h)$, we assume that ρ is defined on all free variables of φ. All of the definitions are rather standard, except for a few points that we describe here.

The first unusual point is the presence of the *horizon* variable h, which represents how much into the future we are allowed to see. This variable is untouched by most constructs. However, when constructs make use of positions (such as $\Diamond_D\,\varphi$), if the horizon h is negative (i.e., we do not know the value of the signal at that point in time), then the semantics uses I. When constructs do not use positions (such as a), then it does not matter whether the horizon is negative or not. Moreover, the semantics of $F_T\,\varphi$ shifts the horizon variable to reflect the fact that we evaluate the semantics of φ at a different point in time.

Secondly, the semantics of the diamond modality is rather long, but $[\![\Diamond_D\,\varphi]\!]_\rho(\sigma, t, h)(\mathrm{a})$ is $\top$ if $h \geq 0$ and there is an agent b whose distance from a at time t is in D and $[\![\varphi]\!]_\rho(\sigma, t, h)(\mathrm{b})$ is $\top$, which corresponds to the usual semantics of $\Diamond_D$. It is $\bot$ in two cases: first, if $h \geq 0$ and for all agents b, either their distance to a at time t is not in D or $[\![\varphi]\!]_\rho(\sigma, t, h)(\mathrm{b})$ is $\bot$; second, if $h < 0$ and for all agents b, $[\![\varphi]\!]_\rho(\sigma, t, h)(\mathrm{b})$ is $\bot$. In the first case, we have access to positions and can refute the formula if the distance between agents is not in D. In the second one, do not have access to positions, and must therefore refute φ for all agents to be sure that $\Diamond_D\,\varphi$ is $\bot$. It is I otherwise: either $h < 0$ and there is an agent b such that $[\![\varphi]\!]_\rho(\sigma, t, h)(\mathrm{b})$ is not $\bot$, or $h \geq 0$ and there is an agent b whose distance to a at time t is in D and such that $[\![\varphi]\!]_\rho(\sigma, t, h)(\mathrm{b})$ is I.

Remark 7 (Diamond modality, space, and connectivity)**.** *In this paper, the diamond modality has a spatial flavour, since we chose its interpretation to be about the distance between agents. In this sense, our logic is close to spatio-temporal logics. However, from a theoretical point of view, the diamond modality is an agent modality, rather than a spatial one, and we could similarly define a semantics on a graph, rather than on trajectories (μ*TGL *stands for* Temporal Graph Logic with Fixed Points*). We stick to this interpretation of the diamond modality in this paper because it is general enough to express the properties we are interested in.*

Finally, the only non-standard construct is $H_t\,\varphi$. Intuitively, it asks whether φ is true by looking only at horizon h'. To do so, it computes the semantics of φ with (the potentially shorter) horizon $\min(h, h')$ and returns it. However, when $h \geq h'$, we have a long enough horizon to give a boolean answer, and therefore I is changed to $\bot$ (we were unable to prove the property with horizon h'). This helps give formulas a boolean value in finite time, since it gets rid of I values when enough time has passed.

Example 8 (Semantics of Φ_{con})**.** *For any formula φ where X*

$$[\![\top]\!]_\rho(\sigma,t,h) = \top \qquad [\![p]\!]_\rho(\sigma,t,h)(\mathrm{a}) = \begin{cases} p(\sigma(\mathrm{a},t)) & \text{if } h \geq 0 \\ \mathtt{I} & \text{otherwise} \end{cases} \qquad [\![\neg\varphi]\!]_\rho(\sigma,t,h) = \neg\,[\![\varphi]\!]_\rho(\sigma,t,h)$$

$$[\![\varphi \wedge \psi]\!]_\rho(\sigma,t,h) = [\![\varphi]\!]_\rho(\sigma,t,h) \wedge [\![\psi]\!]_\rho(\sigma,t,h) \qquad [\![F_T\,\varphi]\!]_\rho(\sigma,t,h) = \bigvee_{t' \in T} [\![\varphi]\!]_\rho(\sigma,t+t',h-t')$$

$$[\![\Diamond_D\,\varphi]\!]_\rho(\sigma,t,h)(\mathrm{a}) = \begin{cases} \bigvee_{\mathrm{b}\in\mathsf{A}}([\![\varphi]\!]_\rho(\sigma,t,h)(\mathrm{b}) \wedge d(\sigma(\mathrm{a},t),\sigma(\mathrm{b},t) \in D) & \text{if } h \geq 0 \\ (\bigvee_{\mathrm{b}\in\mathsf{A}} [\![\varphi]\!]_\rho(\sigma,t,h)(\mathrm{b})) \wedge \mathtt{I} & \text{otherwise} \end{cases} \qquad [\![a]\!]_\rho(\sigma,t,h) = \{\rho(a)\}$$

$$[\![\exists a.\,\varphi]\!]_\rho(\sigma,t,h) = \bigvee_{\mathrm{b}\in\mathsf{A}} [\![\varphi]\!]_{\rho[a\to\mathrm{b}]}(\sigma,t,h) \qquad [\![X]\!]_\rho(\sigma,t,h) = \rho(X)(t,h) \qquad [\![\mu X.\,\varphi]\!]_\rho(\sigma,t,h) = \mathrm{lfp}(f \mapsto [\![\varphi]\!]_{\rho[X\to f]}(\sigma,-,-))(t,h)$$

$$[\![H_{h'}\,\varphi]\!]_\rho(\sigma,t,h)(\mathrm{a}) = \begin{cases} \bot & \text{if } [\![\varphi]\!]_\rho(\sigma,t,h')(\mathrm{a}) = \mathtt{I} \text{ and } h \geq h' \\ [\![\varphi]\!]_\rho(\sigma,t,\min(h,h'))(\mathrm{a}) & \text{otherwise} \end{cases}$$

$$[\![\varphi \leq \psi]\!]_\rho(\sigma,t,h) = \bigwedge_{\mathrm{a}\in\mathsf{A}} ([\![\varphi]\!]_\rho(\sigma,t,h)(\mathrm{a}) \to [\![\psi]\!]_\rho(\sigma,t,h)(\mathrm{a}))$$

Fig. 2. Definition of the semantics

only appears positively, by the Kleene fixed-point theorem, the sequence $f_0(t,h) = \bot$, $f_{n+1}(t,h) = [\![\varphi]\!]_{\rho[X\to f_n]}(\sigma,t,h)$, reaches a fixed point, and $[\![\varphi]\!]_\rho(\sigma,t,h) = \sup_{n\in\mathbb{N}} f_n(t,h)$.

We apply this to Φ_{con} from Example 3. Let us assume $h \geq 0$, then we have

$$[\![b \vee \Diamond_{[0,\mathrm{d}]}\, X]\!]_\rho(\sigma,t,h)(a) = \{\rho(b)\} \vee \bigvee_{c\in\mathsf{A}} (\rho(X)(t,h) \wedge d(\sigma(a,t),\sigma(c,t)) \leq \mathrm{d}).$$

If we denote by G_t the graph of agents, where a and a' are connected if and only if $d(\sigma(\mathrm{a},t),\sigma(\mathrm{a}',t)) \leq \mathrm{d}$, then we get that $f_{n+1}(t,h)$ is the set of agents reachable from $\rho(b)$ by a path of length at most n in G_t. Therefore $[\![\mu X.\,(b \vee \Diamond_{[0,\mathrm{d}]}\, X)]\!]_\rho(\sigma,t,h)$ is the set of agents reachable from $\rho(b)$ in G_t. Note in particular that it is a set, in the sense that it does not map any agent to $\mathtt{I}$. Therefore, the whole formula states that, at all times, for all agents a and b, a is reachable from b at time t, so the communication graph at time t is connected.

C. Properties of horizons

Here, we discuss several properties of μTGL relative to horizons. First, we want to show that $\mathtt{I}$ indeed corresponds to the fact that the current data is not enough to conclude whether the semantics is true or false. To represent information gain, we define the *information order* $\preceq$ on $\mathsf{3}$ generated by $\mathtt{I} \prec \top$ and $\mathtt{I} \prec \bot$. First, we should show that having a longer horizon can only lead to more information.

Lemma 9 (Monotonicity of semantics)**.** *For all H-free formulas φ, $t \in \mathbb{R}_{\geq 0}$, $h \leq h' \in \overline{\mathbb{R}}$, signals σ, and monotone contexts ρ (i.e., such that $\rho(X)$ is $\preceq$-monotone in h for all X), $[\![\varphi]\!]_\rho(\sigma,t,h) \preceq [\![\varphi]\!]_\rho(\sigma,t,h')$.*

This allows us to give meaning to the following notion:

Definition 10 (h-determination)**.** *We say that φ is h-*determined *for $h \in \overline{\mathbb{R}_{\geq 0}}$, if for all times t, signals σ, and contexts ρ, $[\![\varphi]\!]_\rho(\sigma,t,h) \neq \mathtt{I}$. We write $\sigma,t \vDash_\rho \varphi$ to denote the fact that $[\![\varphi]\!]_\rho(\sigma,t,+\infty)$ is $\top$ and $\sigma,t \nvDash_\rho \varphi$ when it is $\bot$.*

By Lemma 9, if $[\![\varphi]\!]_\rho(\sigma,t,h) = \top$ for some $h \in \mathbb{R}$, then $\sigma,t \vDash_\rho \varphi$, and similarly for $\bot$. By Example 4, we know that there are H-free formulas such that $[\![\varphi]\!]_\rho(\sigma,t,h) = \mathtt{I}$ for all $h \in \mathbb{R}$. The following lemma shows that this behaviour does not happen with infinite horizons, which is expected: having information about the whole signal is enough to decide the semantics.

Lemma 11 ($+\infty$-determination)**.** *For all formulas φ, $t \in \mathbb{R}_{\geq 0}$, signals σ, and contexts ρ, either $\sigma,t \vDash_\rho \varphi$ or $\sigma,t \nvDash_\rho \varphi$.*

The following lemma gives a concrete meaning to the intuition that $\mathtt{I}$ corresponds to a lack of information about the signal.

Lemma 12 (Horizon lemma)**.** *For all H-free formulas φ, $t \in \mathbb{R}_{\geq 0}$, $h \in \overline{\mathbb{R}}$, signals σ, and monotone contexts ρ:*

- *if $[\![\varphi]\!]_\rho(\sigma,t,h) = \top$, then for all σ' such that $\sigma'_{|[t,t+h]} = \sigma_{|[t,t+h]}$, $\sigma',t \vDash_\rho \varphi$,*
- *if $[\![\varphi]\!]_\rho(\sigma,t,h) = \bot$, then for all σ' such that $\sigma'_{|[t,t+h]} = \sigma_{|[t,t+h]}$, $\sigma',t \nvDash_\rho \varphi$,*

where $\sigma_{|t,t'}$ is the restriction of σ to $[t,t']$.

As an immediate corollary, we get that, if there are signals σ_1 and σ_2, with $\sigma_{i|[t,t+h]} = \sigma_{|[t,t+h]}$, $\sigma_1,t \vDash_\rho \varphi$ and $\sigma_2,t \nvDash_\rho \varphi$, then $[\![\varphi]\!]_\rho(\sigma,t,h) = \mathtt{I}$.

By Lemma 12, the semantics of an h-determined formula only depends on a window of size h of the signal. A streaming algorithm is therefore able to yield a result for time t after it has received the input signal up to time $t+h$.

For any formula φ, we can statically compute a horizon

$\overline{\varphi}$ (possibly infinite) as follows:

$$\overline{\top} := 0, \qquad \overline{f(a) > 0} := 0,$$
$$\overline{\neg\varphi} := \overline{\varphi}, \qquad \overline{\varphi \wedge \psi} := \max(\overline{\varphi}, \overline{\psi}),$$
$$\overline{\Diamond_D\, \varphi} := \overline{\varphi}, \qquad \overline{F_{[s,t]}\, \varphi} := \overline{\varphi} + t,$$
$$\overline{X} := 0, \qquad \overline{\mu X.\, \varphi} := \begin{cases} 0 & \text{if } \overline{\varphi} = 0 \\ \infty & \text{otherwise,} \end{cases}$$
$$\overline{a} := 0, \qquad \overline{\exists a.\, \varphi} := \overline{\varphi},$$
$$\overline{H_t\, \varphi} := \min(t, \overline{\varphi}), \qquad \overline{\varphi \leq \psi} := \max(\overline{\varphi}, \overline{\psi}).$$

Lemma 13 (Bounded horizon)**.** *All formulas φ are $\overline{\varphi}$-determined.*

D. Notable Formulas

Here, we give some typical agent formulas that are useful when specifying properties, as well as their semantics.

1) Reachability: The first class of formulas we are interested in are formulas that express *reachability* properties. In our context, this corresponds to the possibility to reach an agent that satisfies some property ψ. One such formula is

$$\Phi_{\text{reach},\psi} \;=\; \mu X.\, (\psi \vee \Diamond_D\, X). \tag{3}$$

Note that this is the shape of Φ_{con} from Example 3. Its semantics is basically the same as the one in Example 8, but with a general formula ψ.

A variation on (3) is $\varphi = \mu X.\, (\psi \vee F_T\, \Diamond_D\, X)$, which adds a temporal aspect to the formula. Its semantics $[\![\varphi]\!]_\rho\, (\sigma, t, h)$ maps a to:

- $\top$ if there exist agents $\mathrm{a}_1, \ldots, \mathrm{a}_n$ and times $t_1, \ldots, t_{n-1}$ such that $\mathrm{a}_1 = \mathrm{a}$, for all $i < n$, $d(\sigma(\mathrm{a}_i, t_i), \sigma(\mathrm{a}_{i+1}, t_i)) \in D$, for all $i < n-1$, $t_{i+1} - t_i \in T$, $t_{n-1} \leq h$, and $[\![\psi]\!]_\rho\, (\sigma, t + t_n, h - t_n)(\mathrm{a}_n) = \top$,
- $\mathtt{I}$ if there exist agents $\mathrm{a}_1, \ldots, \mathrm{a}_n$ and times $t_1, \ldots, t_{n-1}$ such that $\mathrm{a}_1 = \mathrm{a}$, , for all $i < n$, $d(\sigma(\mathrm{a}_i, t_i), \sigma(\mathrm{a}_{i+1}, t_i)) \in D$, for all $i < n-1$, $t_{i+1} - t_i \in T$, and $t_{n-1} > h$ or $[\![\psi]\!]_\rho\, (\sigma, t + t_n, h - t_n)(\mathrm{a}_n) = \mathtt{I}$ (and we are not in the previous case)
- $\bot$ otherwise.

Note that this is the shape of Φ_{ch} from Example 4 and that the semantics above is a formal definition of what a communication chain from a to some agent that satisfies ψ is. More precisely, the semantics is $\top$ if there is a chain within horizon h, $\mathtt{I}$ if there may be a chain after horizon h (but we do not have enough information to know yet), and $\bot$ if we know there is no such chain (no matter the horizon).

2) Safety: Another class of formulas of particular interest is that of *safety* formulas, which in our case correspond to formulas that we can never reach an agent that satisfies some formula ψ. The typical formula that expresses this is

$$\Phi_{\text{safe},\psi} \;=\; \nu X.\, (\neg\psi \wedge \Box_D\, X). \tag{4}$$

Note that it is the negation of $\Phi_{\text{reach},\psi}$ (up to double negations), and therefore its semantics is also negated. It therefore exactly describes the impossibility to reach any agent that satisfies ψ.

As in the previous section, we can modify (4) to obtain $\varphi = \mu X.\, (\psi \vee F_T\, \Diamond_D\, X)$, which expresses the inexistence of communication chains to any agent that satisfied ψ.

E. Expressive Power

We want to briefly discuss the expressive power of μTGL. First, let us consider the variant of MITL [10] with syntax:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid F_T\, \varphi.$$

In order to compare μTGL to MITL, they need to use the same predicates, and we need to extend the set of MITL predicates to contain some spatial operators. We thus assume that MITL has a predicate p_{a} for each predicate p of μTGL and agent a which holds when p holds for agent a. We also add a predicate $d_{\mathrm{a},\mathrm{b},D}$ to MITL predicates for each distance interval $D \subseteq \overline{\mathbb{R}_{\geq 0}}$ and agents a and b. It holds when the distance between a and b is in D. Note that, in order to do this, we need to know in advance all agents in the network, so MITL is limited compared to μTGL on that point. Nevertheless, to be able to compare the two logics, we assume that the number of agents is known in advance.

Lemma 14. MITL *is strictly less expressive than* μTGL.

The translation of MITL into μTGL is straightforward. Conversely, formulas such as Φ_{ch} from Example 4 cannot be expressed in MITL.

We can also express basic *counting* properties in μTGL [7], by which we mean that we can express properties such as "at least n agents that satisfy φ".

Given a constant integer n and a formula φ with an open agent variable a, we can define the following syntactic sugar:

$$(|\varphi| \geq n) = \exists a_1, \ldots, a_n.\, (\bigvee_i \bigvee_{j \neq i} (a_i \neq a_j)) \wedge (\bigwedge_i \varphi[a \to a_i]),$$
$$(|\varphi| \leq n) = \exists a_1, \ldots, a_n.\, \forall a.\, \varphi \to a \leq \bigvee_i a_i,$$
$$(|\varphi| = n) = (|\varphi| \geq n) \wedge (|\varphi| \leq n).$$

These formulas have the obvious meaning, e.g., the first one means "at least n agents satisfy φ".

III. EXAMPLES

In this section, to showcase the expressive power of μTGL, we discuss scenarios and formulas that express interesting properties of these scenarios.

A. Communication

We start by expressing the existence of certain types of communication chains in networks of communicating agents. In all the following examples, we assume that we want to model a similar situation as in Example 4, where agents can communicate up to distance d.

Example 15 (Communication time)**.** *A possible refinement of Example 5 is to model the communication as taking a fixed amount of time ε. We can then slightly modify* (2) *to obtain*

$$G\, \forall a, b.\, a \leq H_t\, \mu X.\, (b \vee F_{[\varepsilon,t]}\, \Diamond_{[0,\mathrm{d}]}\, X).$$

Note that communication chains may no longer be of arbitrary length, since at least ε has to pass when communicating to another agent.

Example 16 (Message storing)**.** *Building on Example 15, we can also model a scenario where agents only store the messages they receive for a finite amount of time s. The existence of a communication chain in this scenario can be expressed by*

$$H_t\, G\, \forall a, b.\, a \subseteq \mu X.\, (b \vee F_{[\varepsilon,s]}\, \Diamond_{[0,\mathrm{d}]}\, X).$$

Example 17 (Chains of bounded length)**.** *While our logic is not designed to, properties such as the existence of communication chains of bounded length in the number of agents can also be expressed by "unrolling" fixed points in the previous formulas. For example, the fact that, at all times, any two agents can communicate by passing through at most one other agent can be expressed as*

$$G\, \forall a, b.\, a \subseteq b \vee \Diamond_{[0,\mathrm{d}]}\, (b \vee \Diamond_{[0,\mathrm{d}]}\, b).$$

We can also model scenarios where communication is impeded, either in some specific zones or by malicious agents jamming signals.

Example 18 (Communication-forbidding zones)**.** *Assume that there is a zone $Z \subseteq M$ where communication is impossible. We can define a predicate z such that $z(x)$ holds if and only if $x \in Z$. The existence of communication chains between all agents can be expressed by*

$$G\, \forall a, b.\, a \subseteq \mu X.\, (\neg z \wedge (b \vee F_{[0,t]}\, \Diamond_{[0,\mathrm{d}]}\, X)).$$

Adding the $\neg z$ subformula ensures that all agents must be outside Z in any communication chain.

Example 19 (Jamming agents)**.** *We can model a scenario where malicious agents can jam the communication between two agents if they are within distance* j *of either the sender or the receiver of the message. If agents evolve in space M, we can model them as evolving in $M' = \{0,1\} \times M$, with d' inherited from d ($d'((i,x),(i,y)) = d(x,y)$ and $d'((i,x),(1-i,y)) = 1 + d(x,y)$), where malicious agents are labelled* 1*. We can define a predicate j that holds on points of the form $(1,x)$, i.e., malicious agents.*

In such a scenario, the existence of communication chains between all agents can be expressed by

$$G\, \forall a, b.\, ((a \vee b \leq \neg j) \wedge (a \neq b)) \rightarrow \\ (a \leq \mu X.\, (\neg\, \Diamond_{[0,\mathrm{j}+1]}\, j \wedge (b \vee F_{[0,t]}\, \Diamond_{[0,\mathrm{d}]}\, X)))$$

B. Network Topologies

Our logic can also express the existence of specific network topologies, and not only connectedness. In this section, we will assume for simplicity that $D = [0, \mathrm{d}]$.

Example 20 (Star topology)**.** *The simplest topology to express in our logic is the* star *topology, in which there is a single agent* a *that is linked to all other agents, and no other agents are linked to one another. This topology can be expressed by*

$$\Phi_\star \;=\; \exists a.\, \forall b.\, (b \subseteq \Diamond_D\, a) \wedge (a \neq b \rightarrow \Diamond_D\, b \subseteq a \vee b)$$

While $\Phi_\star$ holds when agents are linked through a star topology, it may also be useful to know when there is a star communication topology, possibly with more connections, which is simply expressed by

$$\widetilde{\Phi_\star} \;=\; \exists a.\, \forall b.\, b \subseteq \Diamond_D\, a.$$

Example 21 (Linear topology)**.** *We can also express the fact that agents are linked through a linear topology, i.e., there is a line of agents connecting all agents together. This can be expressed by*

$$\Phi_- \;=\; \Phi_{\mathrm{con}} \wedge \exists a, b.\, |\Diamond_D\, a| = 2 \wedge |\Diamond_D\, b| = 2 \wedge \\ \forall c.\, (c \wedge (a \vee b) \subseteq \varnothing) \rightarrow |\Diamond_D\, c| = 3.$$

Example 22 (Ring topology)**.** *Finally, we can also express the fact that the communication graph between the agents forms a ring by*

$$\Phi_\circ \;=\; \Phi_{\mathrm{con}} \wedge \forall a.\, |\Diamond_D\, a| = 2.$$

C. MITL Properties

As expressed by Lemma 14, MITL properties (of agents) can be expressed in μTGL. More precisely, any MITL property φ can be translated to some μTGL formula φ' that describes the set of agents that satisfy φ. This translation can be used to model certain kinds of scenarios.

The first typical case is when all agents must satisfy some MITL property (for example a safety property, stating that agents must stay in some safe zone). That all agents satisfy this property is expressed in μTGL as $\top \leq \varphi'$. The second typical case is when at least one agent must satisfy some property (for example a progress property, stating that the agent much reach some zone). That at least an agent satisfies this property is directly expressed as $\neg(\varphi' \leq \bot)$.

Example 23 (MITL goals and connectivity)**.** *Let us consider a scenario where agents must "link" a starting zone (expressed as some predicate s) to a goal zone (stated as some predicate g). Formally, there are three requirements: 1) that some agent stays in the start zone at all times (expressed as the MITL property $G\, s$), 2) that some agent eventually reaches goal (MITL property $F\, g$), 3) that the communication graph stays connected at all times (μ*TGL *property $G\, \Phi_{\mathrm{con}}$). Then the formula that models the desired property is $G\, \Phi_{\mathrm{con}} \wedge (\top \leq G\, s) \wedge \neg(F g \leq \bot)$.*

IV. MONITORING

A. Monitoring Algorithm

In this section we describe an algorithm for monitoring a μTGL formula, see Algorithm 1.

1) Data structures: The algorithm manipulates values representing piecewise constant functions $\mathbb{R}_{\geq t_0} \to X$, for some set X. Such a function is represented by a sequence of *samples*, pairs

$$x = ((t_0, x_0), \ldots, (t_n, x_n))$$

where $t_i \in \mathbb{R}_{\geq 0}$ and $x_i \in X$. This represents the signal $\bar{x}\colon \mathbb{R}_{\geq t_0} \to X$ such that

$$\bar{x}(t) = \begin{cases} x_i & \text{when } t_i \leq t < t_{i+1}, \text{ for some } 0 \leq i < n, \\ x_n & \text{when } t_n \leq t. \end{cases}$$

The time t_0 is the *start time* and denoted $\mathsf{start}(x)$. When the signal to be represented is only defined over some finite interval, there is also a specified end time $\mathsf{end}(x)$, in which case $\bar{x}(t) = x_n$ for all $t_n \leq t \leq \mathsf{end}(x)$.

The main operation we need on signals represented in this way is to zip them. This is a procedure which takes a signal x of values of type X, a signal y of values of type Y, and produces a signal $x \times y$ of values of type $X \times Y$, such that

$$\overline{x \times y}(t) = (\bar{x}(t), \bar{y}(t))$$

for all t for which both $\bar{x}(t)$ and $\bar{y}(t)$ are defined. This is achieved by unifying the time-indices in x and y.

Uncertain sets, that is, functions $\mathsf{A} \to \mathsf{3}$ in the semantics, are represented as A-*intervals* $[U, V]$ ($U, V \in \mathcal{P}(\mathsf{A})$) indicating that the uncertain set could be any $X \in \mathcal{P}(A)$ such that $U \subseteq X \subseteq V$, that is:

$$[U, V](a) = \begin{cases} \top & \text{if } a \in U, \\ \mathtt{I} & \text{if } a \notin U \text{ and } a \in V, \\ \bot & \text{if } a \notin V. \end{cases}$$

The intuition here is that what is known about the uncertain set is a lower bound U (elements that are surely members), and an upper-bound V (all possible members). We denote by $\bot$ the A-interval $[\varnothing, \varnothing]$, which represents the uncertain set $a \mapsto \bot$.

The uncertain sets manipulated in the algorithm also depend on a horizon parameter $h \in \mathbb{R}_{\geq 0}$, representing how much future lookahead the information in the uncertain set depends on. To model this we use signals of A-intervals which always start from time index 0. The signal which is constantly $\bot$ is also denoted $\bot$.

All in all, a semantic value $\bar{w}\colon \mathbb{R}_{\geq t_0} \times \mathbb{R}_{\geq 0} \times \mathsf{A} \to \mathsf{3}$ is represented by a signal of signals of A-intervals:

$$w = \left(\begin{array}{c} (t_0, ((h_{0,0}, [U_{0,0}, V_{0,0}]), \ldots, (h_{0,m_0}, [U_{0,m_0}, V_{0,m_0}]))), \\ \vdots \\ (t_n, ((h_{n,0}, [U_{n,0}, V_{n,0}]), \ldots, (h_{n,m_n}, [U_{n,m_n}, V_{n,m_n}]))) \end{array} \right)$$

such that

$$\bar{w}(t, h, a) = \begin{cases} \top & \text{if } a \in U_{i,j}, \\ \mathtt{I} & \text{if } a \notin U_{i,j} \text{ and } a \in V_{i,j}, \\ \bot & \text{if } a \notin V_{i,j}. \end{cases}$$

whenever

$$t_i \leq t < t_{i+1} \quad \text{and} \quad h_{i,j} \leq h - (t - t_i) < h_{i,j+1}.$$

2) Computing semantics: We first describe computing the semantics of a formula φ on a finite signal w with start time t_0 and end time $\mathsf{end}(w) = t_{\mathsf{end}}$. The computation proceeds by induction on the structure of φ. Most of the cases are simple, so we will only describe the cases $\varphi \vee \psi$, $F_{[a,b]}\,\varphi$ and $\mu X.\, \varphi$.

- For $\varphi \vee \psi$, the semantics are computed for φ and ψ individually, producing two output signals w_φ and w_ψ. Zipping these two signals produces $w_\varphi \times w_\psi$, a signal of pairs of signals of A-intervals. These pairs are themselves zipped, thus producing a signal of signals of pairs of A-intervals. Finally these pairs of A-intervals are joined using the union operation lifted to A-intervals:

  $$[U, V] \vee [U', V'] := [U \cup U', V \cup V'],$$

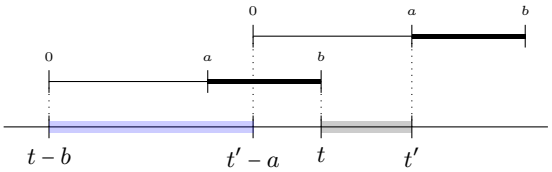
  thereby producing the desired result $w_{\varphi \vee \psi}$. Here we use that $[U \cup U', V \cup V'](a) = [U, V](a) \vee [U', V'](a)$. Most of the other logical operations are computed in a similar point-wise fashion. The operation

  $$w_\varphi \vee w_\psi := w_{\psi \vee \psi}$$

  just described will also be used in the semantics of $F_{[a,b]}\,\varphi$.
- The semantics of $F_{[a,b]}\varphi$ is computed by first computing the semantics w_φ of φ. Then, for each sample $(t, w_{\varphi,t}) \in w_\varphi$ that lasts till $t' \in \mathbb{R}_{\geq 0}$, we construct the signal of signals

  $$F(t, t', w_\varphi) := ((t_0, \bot), (t - b, \mathsf{delay}^b(w_{\varphi,t})), (t' - a, \bot))$$

  with end time t_{end}. This represents the information of w_φ available in $[t, t']$ given an allowed future lookahead of some duration $d \in [a, b]$. To understand this consider the following diagram:

  The blue region represents the non-$\bot$ portion of $F(t, t', w_\varphi)$. Here, a *delay* is needed to account for the fact that this lookahead has used up some of the horizon. This is defined by

  $$\overline{\mathsf{delay}^b(x)}(s) = \begin{cases} \bot & \text{when } 0 \leq s < b, \\ \bar{x}(s - b) & \text{when } b \leq s. \end{cases}$$

  for all signals x. The semantics of $F_{[a,b]}\varphi$ is then obtained by

  $$\bigvee F(t, t', w_\varphi).$$

  where the disjunction is taken over all the piecewise constant parts of w_φ described above.
- The semantics of $\mu X.\, \varphi$ is computed by repeatedly computing the semantics of φ while using increasing values for resolving the variable X, starting with $X = \bot$. Since X appears only positively in φ, successive values

Algorithm 1: Monitoring algorithm.

Input: a μTGL formula φ, a stream S
Output: a stream of $\{\bot, \top\}$

initialise the μ-cache to $\bot$ for all top-level μ's of φ;
$h_{\mathsf{max}} \leftarrow \overline{\varphi}$;
if $h_{\mathsf{max}} = \infty$ **then**
    fail;
end
$w \leftarrow$ ingest a window of size $\geq h_{\mathsf{max}}$ of samples from S;
$t \leftarrow 0$;
repeat
    $o \leftarrow$ `Semantics(`φ`,`w`)`;
    yield $\bar{o}(t, h_{\mathsf{max}})$;
    drop the first sample from w and all items in the μ-cache;
    $t \leftarrow \mathsf{start}(w)$;
    $s \leftarrow$ ingest new samples from S until w has size $\geq h_{\mathsf{max}}$;
    Add a $\bot$ sample to the end of all items in the cache;
until *end of input stream* S;

of X must be increasing. Since the number of agents is finite, we are guaranteed to reach a fixed point in finite time.

3) Streaming algorithm: We are now able to describe a streaming monitoring algorithm. This is achieved by computing an upper-bound h_{max} for the horizon of the formula and then operating on a window of size h_{max}. Over such a window the semantics are computed as described above. Most forms are not computationally expensive, the exceptions are the μ-forms which may require a large number of iterations before reaching a fixed point. Therefore, we re-use the work done for computing the μ-forms of the previous window. When monitoring a formula φ, all toplevel μ-forms of φ, that is, all subformulas $\mu X.\psi$ of φ that are not themselves sub-formulas of another μ-form, form the index of the *μ-cache*. This is a mapping of such subformulas to semantic values over the current time window. When computing the semantics of a toplevel μ-form $\mu X.\psi$ over the current window, instead of starting the iteration at $X = \bot$, the current value of the cache is used instead.

The streaming algorithm is described in Algorithm 1. The use of the μ-cache produces a correct result thanks to the following two lemmas. The first ensures that to compute $\mu X.\psi$, the algorithm can start iterating from any element we know to be smaller than the least fixed point, rather than always starting at $\bot$:

Lemma 24. *Assume L is a dcpo[1] and that $f\colon L \to L$ is a Scott-continuous function with least fixed point M. If $N \leq M$, then the supremum of the ascending Kleene chain starting at N is M.*

[1]directed-complete partial order

The second ensures that when computing $\mu X.\psi$ of a prefix of a signal, the result obtained is smaller than the result for the full signal. In fact this is true for any positive formula:

Lemma 25. *If σ' is a prefix of σ, and φ is positive, then*

$$[\![\varphi]\!]_\rho(\sigma', t, h)(a) \leq [\![\varphi]\!]_\rho(\sigma, t, h)(a).$$

V. EXTENDED EXAMPLE: RESILIENT CONSENSUS

In this section, we prove the usefulness of our logic and monitoring algorithm on the example of resilient consensus. In Section V-A we give context and the existing relevant results on resilient consensus. In Section V-B, we extend μTGL and give a formula that ensures resilient consensus. In Section V-C, we discuss causality in consensus and derive another formula for resilient consensus. Finally, in Section V-D, we discuss a specific example and test our monitoring algorithm on it.

A. A Property for Resilient Consensus

We consider the *resilient consensus problem* in the presence of malicious/faulty agents. This problem involves a number of *cooperative agents* that try to reach agreement on a scalar value by exchanging data according to a consensus protocol and *non-cooperative agents* that do not follow the protocol either intentionally or due to a fault. In particular, we consider scenarios where the cooperative agents do not know which other agents are cooperative and thus may receive and use malicious/faulty data transmitted from non-cooperative agents.

It was shown in [14], [15] that consensus among cooperative agents is possible if the number of non-cooperative agents is sufficiently small and the graph representing the inter-agent communication topology has sufficiently large connectivity. In those articles, the connectivity of a graph was characterized through the notion of *r-robustness*. For a given a graph G with node set V, edge set $E \subseteq V \times V$, and neighbor sets $N_i = \{j \mid (i,j) \in E\}$, $i \in V$, this notion is defined as follows.

Definition 26 (r-robustness [14])**.** *Given a positive integer r, a graph G is* r-robust*, if $\forall S_1, S_2 \in 2^V \setminus \{\varnothing\}$ such that $S_1 \cap S_2 = \varnothing$, we have $\{i \in S_1 \mid |N_i \setminus S_1| \geq r\} \neq \varnothing$ or $\{i \in S_2 \mid |N_i \setminus S_2| \geq r\} \neq \varnothing$ or both.*

An intuitive meaning of r-robustness is that, for any two non-empty sets, if we choose any $r-1$ nodes in the graph, then there is always a path between these two sets that avoids these $r-1$ nodes.

Recently, the notion of r-robustness was extended in [16] to handle the cases where the inter-agent communication topology is time-varying. In the setting of [16], the communication topology at each time $t \in \mathbb{N}$ is represented by a graph $G[t]$ with node set V, edge set $E[t]$, and neighbor sets $N_i[t] = \{j \mid (i,j) \in E[t]\}$, $i \in V$. The notation $\bigcup_{\tau=0}^{T} G[t-\tau]$ represents the union graph with node set V, and edge set $\bigcup_{\tau=0}^{T} E[t-\tau]$. For a given time-varying graph G, the notion of (T,r)-robustness is defined as follows.

Definition 27 ((T,r)-robustness [16])**.** *Given positive integers T and r, a time-varying graph G is* (T,r)-robust *if $\bigcup_{\tau=0}^{T} G[t-\tau]$ is r-robust for every $t \geq T$.*

It was shown by [16] that consensus among cooperative agents can be achieved if there are at most F non-cooperative agents in the network and the time-varying graph representing the communication topology is $(T, 2F+1)$-robust.

Time-varying communication topologies and the notion of (T,r)-robustness are particularly useful for scenarios where inter-agent data transmissions require the agents to be physically close to each other. In such scenarios there is a communication radius and two agents can exchange data only when their distance is smaller than it.

The techniques that we introduced in earlier sections allow us to develop monitoring algorithms for assessing (T,r)-*robustness* in scenarios with time-varying communication topologies by using the history of the physical locations of agents and the communication radius.

B. Extending μTGL

In order to express (T,r)-robustness, we extend μTGL with a second-order agent quantification construct $\exists s.\, \varphi$ with the following semantics:

$$[\![\exists s.\, \varphi]\!]_\rho(\sigma,t,h) = \bigvee_{S \subseteq \mathsf{A}} [\![\varphi]\!]_{\rho[s \to S]}(\sigma,t,h).$$

This extension comes with syntactic sugar $a \in S \equiv a \leq s$, $a \notin s$, and $|s| \geq n$ and such defined as in Section II-E. We also allow truth formulas inside agent formulas. For example, we allow formulas such as $\mu X.\, \Phi \wedge (b \vee \Diamond_D\, X)$, which is the connected component of $\rho(b)$ if Φ holds, and empty either. Formally, the semantics of a truth formula seen as an agent formula is defined as $[\![\Phi]\!]_\rho(\sigma,t,h)(\mathrm{a}) = [\![\Phi]\!]_\rho(\sigma,t,h)$ (where Φ is seen as a truth formula on the right). Extending the monitoring algorithm is straightforward.

With this extension of μTGL, we can directly encode the definition of (T,r)-robustness.

Lemma 28 ((T,r)-robustness in μTGL)**.** *Given a signal σ, a distance interval $D \subseteq \overline{\mathbb{R}_{\geq 0}}$, $T \in \mathbb{R}_{\geq 0}$, and $r \in \mathbb{N}$, the communication graph is (T,r)-robust if and only if σ satisfies*

$$G\ \forall s, s'.\, (\exists a.\, a \in s \wedge a \in s') \vee \mathrm{reach}_{T,r}(s) \vee \mathrm{reach}_{T,r}(s'),$$

where $\mathrm{reach}_{T,r}(s)$ *is*

$$\exists a, a_1, \ldots, a_r.\, a \in s \wedge \left(\textstyle\bigwedge_{i=1}^{r} \bigwedge_{j=i+1}^{r} a_i \neq a_j\right) \wedge \left(\textstyle\bigwedge_{i=1}^{r} a_i \notin s \wedge a_i \leq F_{[0,T]}\, a_i\right).$$

C. Robustness and Causality

As we explained, an intuition behind r-robustness is that it guarantees the existence of paths that avoid any fixed $r-1$ agents. However, the notion of (T,r)-robustness does not have the same intuition, because it not causal: the path in $\bigcup_{\tau=0}^{T} G[t-\tau]$ may not be a causal chain, as illustrated in the next example.

Example 29 ((T,r)-robustness and causality)**.** *Let us consider a scenario with three agents* a, b, *and* c, *moving on a line, as according to Section V-C, and which are all cooperating. This network is $(T,1)$-robust, but at time $t = 0$,* c *cannot send a message to* a *within time T.*

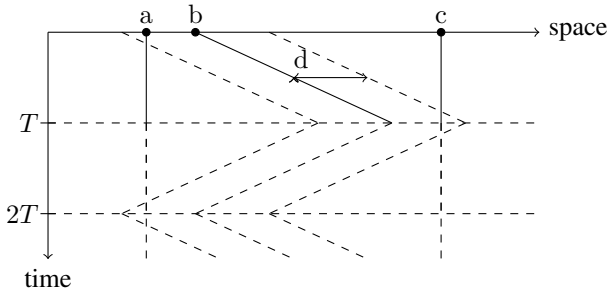


Fig. 3. A $(T,1)$-robust network. However, c cannot send a message to a within time T at $t = 0$

However, since agents can only act based on the messages they have received, whether there is consensus or not does not depend on non-causal properties. Therefore, we may refine (T,r)-robustness into a causal property.

Definition 30 (Causal (T,r)-robustness)**.** *Given $T \in \mathbb{R}_{\geq 0}$ and $r \in \mathbb{N}$, a time-varying graph G is* causally (T,r)-robust *if, for every $t \in \mathbb{R}_{\geq 0}$, $\forall S_1, S_2 \in 2^V \setminus \{\varnothing\}$ such that $S_1 \cap S_2 = \varnothing$, for any $i_1, \ldots, i_{r-1}$, there is a causal chain that starts from some $i \in S_1$ at time t, ends at some $j \in S_2$ at time $t+T$, and avoids all i_k's.*

Lemma 31 (Causal and non-causal (T,r)-robustnesses)**.** *Causal (T,r)-robustness implies (T,r)-robustness. Conversely, if the network contains n agents, (T,r)-robustness implies causal $((n-r)(n-r+1)T/2, r)$-robustness.*

In particular, monitoring causal (T,r)-robustness is enough to guarantee consensus, and it may be faster: monitoring (T,r)-robustness may be in some cases as costly monitoring $((n-r)(n-r+1)T/2, r)$-robustness. The following formula describes causal (T,r)-robustness in μTGL.

Lemma 32 (Causal (T,r)-robustness in μTGL)**.** *Given $T \in \overline{\mathbb{R}_{\geq 0}}$, $r \in \mathbb{N}$, and $D \subseteq \overline{\mathbb{R}_{\geq 0}}$ the communication graph is causally (T,r)-robust if and only if σ satisfies*

$$G\ \forall a, b, x_1, \ldots, x_r.\ \left(\bigwedge_{i=1}^{r} \neg(x_i \leq (a \vee b))\right) \rightarrow a \leq H_T\, \mu X.\, \left(\neg\left(\bigvee_{i=1}^{r} x_i\right) \wedge (b \vee F_{[0,T]}\, \Diamond_D\, X)\right).$$

D. Monitoring for Resilient Consensus

We experimented on a perimeter surveillance scenario [16]. In this scenario, mobile robots circle around a point, with possibly different radii and velocities, and some of them may be faulty or malicious. The robots in the network try to reach consensus to agree on a perimeter to circle. We monitored causal (T,r)-robustness, as expressed in Lemma 32, which ensures that there is consensus.

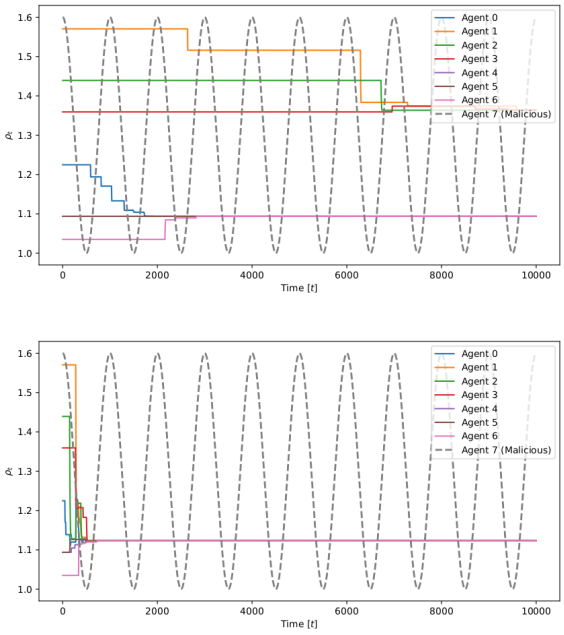


Fig. 4. Signals in the perimeter surveillance scenario

We created two sets of data with eight agents (among which one malicious agent), where agents act according to the control policy described in [16]. The only difference between the two data sets is that, in the first case, the communication radius is 0.2, while it is 0.5 in the second. The radius of the perimeter circled by each agent over time in the two data sets is plotted in Section V-D. It can be seen that cooperative agents reach consensus in the second case, but not in the first one (there are two clusters).

We ran our monitoring algorithm on the two data sets for causal (T, r)-robustness for $r = 3$ (which corresponds to 1 malicious agent). For the first example, we monitored causal robustness for $T = 120$, and it did not hold, which means that (non-causal) robustness is false for $T = 8$. For the second example, we monitored causal robustness for $T = 35$, and it held, ensuring consensus.

VI. CONCLUSION

We defined μTGL, showed its usefulness in expressing interesting properties of agent networks, designed a monitoring algorithm for it, and showed its validity on an example to prove resilient consensus.

There is much left to do in the future. First, while this exposition of μTGL has a strong spatial flavour, it is actually closer to a graph logic, and we want to study it from that angle too. From a practical point of view, it may be interesting to look at general n-ary relations (rather than binary $\Diamond$), since they can be used to model interesting properties, such as more general jamming possibilities. To shorten the horizon used by the monitoring algorithm, we want to refine the definition of the horizon formula, which will require some work to detect tautologies in the logic. Finally, for μTGL to be usable in advanced testing techniques such as falsification, we need to define a robust (or quantitative) semantics, which will prove challenging, since the logic is not boolean. Another direction is to look further into causal robustness, its relationship to non-causal robustness, and whether it can be of algorithmic interest.

REFERENCES

[1] E. Bartocci, J. Deshmukh, A. Donzé, G. Fainekos, O. Maler, D. Ničković, and S. Sankaranarayanan, "Specification-based monitoring of cyber-physical systems: a survey on theory, tools and applications," in *Lectures on Runtime Verification*. Springer, 2018, pp. 135–175.

[2] L. Bortolussi and L. Nenzi, "Specifying and monitoring properties of stochastic spatio-temporal systems in signal temporal logic," in *Proceedings of the 8th International Conference on Performance Evaluation Methodologies and Tools*, 2014, pp. 66–73.

[3] L. Nenzi, L. Bortolussi, V. Ciancia, M. Loreti, and M. Massink, "Qualitative and quantitative monitoring of spatio-temporal properties," in *Runtime Verification*. Springer, 2015, pp. 21–37.

[4] I. Haghighi, A. Jones, Z. Kong, E. Bartocci, R. Gros, and C. Belta, "Spatel: a novel spatial-temporal logic and its applications to networked systems," in *Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control*, 2015, pp. 189–198.

[5] E. A. Gol, E. Bartocci, and C. Belta, "A formal methods approach to pattern synthesis in reaction diffusion systems," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 108–113.

[6] I. Haghighi, S. Sadraddini, and C. Belta, "Robotic swarm control from spatio-temporal specifications," in *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 2016, pp. 5708–5713.

[7] Y. E. Sahin, P. Nilsson, and N. Ozay, "Multirobot coordination with counting temporal logics," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1189–1206, 2019.

[8] D. Kozen, "Results on the propositional mu-calculus," *Theor. Comput. Sci.*, vol. 27, pp. 333–354, 1983. [Online]. Available: https://doi.org/10.1016/0304-3975(82)90125-6

[9] R. Koymans, "Specifying Real-Time Properties with Metric Temporal Logic," *Real Time Syst.*, vol. 2, no. 4, pp. 255–299, 1990. [Online]. Available: https://doi.org/10.1007/BF01995674

[10] R. Alur, T. Feder, and T. A. Henzinger, "The Benefits of Relaxing Punctuality," *J. ACM*, vol. 43, no. 1, pp. 116–146, 1996. [Online]. Available: https://doi.org/10.1145/227595.227602

[11] O. Maler and D. Nickovic, "Monitoring Temporal Properties of Continuous Signals," in *FORMATS 2004*, ser. Lecture Notes in Computer Science, Y. Lakhnech and S. Yovine, Eds., vol. 3253. Springer, 2004, pp. 152–166. [Online]. Available: https://doi.org/10.1007/978-3-540-30206-3_12

[12] P. Baldan, A. Corradini, B. König, and A. Lluch-Lafuente, "A temporal graph logic for verification of graph transformation systems," in *WADT 2006*, ser. Lecture Notes in Computer Science, J. L. Fiadeiro and P. Schobbens, Eds., vol. 4409. Springer, 2006, pp. 1–20. [Online]. Available: https://doi.org/10.1007/978-3-540-71998-4_1

[13] M. Bergmann, *An Introduction to Many-Valued and Fuzzy Logic: Semantics, Algebras, and Derivation Systems*. Cambridge University Press, 2008.

[14] H. J. LeBlanc, H. Zhang, X. Koutsoukos, and S. Sundaram, "Resilient asymptotic consensus in robust networks," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 4, pp. 766–781, 2013.

[15] H. Zhang, E. Fata, and S. Sundaram, "A notion of robustness in complex networks," *IEEE Transactions on Control of Network Systems*, vol. 2, no. 3, pp. 310–320, 2015.

[16] D. Saldana, A. Prorok, S. Sundaram, M. F. Campos, and V. Kumar, "Resilient consensus for time-varying networks of dynamic agents," in *Proc. American control conference*, 2017, pp. 252–258.