An STL Formulation for Intent-Expressive Motion Planning and Intent Estimation With Output Feedback

Elikplim Gah and Sze Zheng Yong, *Member, IEEE*

***Abstract*—This letter presents a tractable signal temporal logic (STL) approach for designing set-based intent-expressive trajectory planning and intent estimation algorithms with output feedback for multi-agent teams. These algorithms allow an observed agent to implicitly convey intent to observer agents while guaranteeing that the agent robustly satisfies state and input constraints, avoids obstacles and achieves its intended STL task specification under worst-case realizations of uncertainties. Specifically, the intent-expressive trajectory planning algorithm encodes intent information by ensuring that the output reachable sets (i.e., all possible measured outputs by the observer agents) for satisfying the intended STL task specifications are disjoint from each other, while the intent estimation algorithm enables the observer agents to decode the intent by eliminating all intent models that are incompatible with noisy run-time observations.**

***Index Terms*—Autonomous systems, optimal control, robust control, estimation, model validation.**

I. Introduction

MULTI-AGENT systems rely on the transmission of data and intent to allow for effective communication between team members to facilitate safe, optimal decision-making and task accomplishment. However, any number of factors ranging from lossy transmission to noisy sensors can bring system operations to a halt. Thus, intent-expressive motion planning, which removes the need for explicit communication between agents while facilitating effective collaboration, is a useful tool for implementing such systems.

Literature Review. Intent-expressive motion planning has been primarily studied in the context of human-robot interactions [1], [2], where the goal is to ensure that the robot moves in a way that the human understands and trusts. The use of gestures or other implicit forms of communication (not unlike non-verbal cues in humans) are commonplace [1]. However, much of the advances made do not apply when the model uncertainty or stochastic distributions are unknown.

Manuscript received 8 March 2024; revised 18 May 2024; accepted 5 June 2024. Date of publication 24 June 2024; date of current version 31 July 2024. This work was supported in part by ONR under Grant N00014-23-1-2093, and in part by NSF under Grant CNS-2313814. Recommended by Senior Editor C. Manzie. *(Corresponding author: Sze Zheng Yong.)*

The authors are with the Mechanical and Industrial Engineering Department, Northeastern University, Boston, MA 02115 USA (e-mail: gah.e@northeastern.edu; s.yong@northeastern.edu).

Digital Object Identifier 10.1109/LCSYS.2024.3418313

In our previous work [3], [4], we presented open- and closed-loop set-based methods for solving this intent-expressive motion planning problem inspired by the active model discrimination (AMD) problem [5], [6]. AMD involves designing a small input that guarantees distinction of the observable output set of the intent model from any other comparable models. Similarly, we designed an input to ensure that trajectories generated for each intent model are distinct. However, these approaches involved solving bilevel optimization algorithms that are computationally expensive and often do not generalize to larger problems. In contrast, this letter will consider an alternative formulation based on signal temporal logic (STL) specifications, e.g., [7], which we found to be computationally more advantageous.

For intent estimation, existing approaches often use a probabilistic framework, e.g., [1], [2]. In contrast, we will consider a set-based approach similar to [3] that leverages a passive model discrimination approach based on model invalidation [8], [9], which uses only output data and knowledge of system dynamics to determine which intent models are incompatible.

Contribution: This letter introduces an improved closed-loop set-based intent-expressive motion planning algorithm based on an STL formulation. As in [4], our approach leverages tube-based predictive control to obtain a closed-loop control law instead of open-loop input sequences in [3]. However, unlike the solution based on bilevel optimization in [4], our proposed STL formulation, which is then recast as mixed-integer linear constraints, achieves the same results with lower computational complexity, and is found in simulations to be significantly faster and to scale better with longer planning horizons. This new formulation also generalizes the setting in [4] to allow more general STL task specifications as intents. In addition, we present an intent estimation algorithm as in [3]. The former algorithm enables an observed agent to convey intent through its motion without explicit communication by ensuring that the output reachable sets for all STL task specifications are disjoint, while the latter algorithm allows an observer agent to decode the intended STL task from only noisy output observations. Together, these algorithms allow multi-agent teams to convey intent without explicit communication.

II. Preliminaries

Notations: Let $x \in \mathbb{R}^n$ be a vector with its norm denoted by $\|x\|_i$ with $i \in \{1, \infty\}$, and $M \in \mathbb{R}^{n\times m}$ a matrix, with transpose $M^\intercal$ and $M \geq 0$ denotes element-wise non-negativity. $\mathbf{0}$, $\mathbb{1}$ and $\mathbb{I}$ represent the vector/matrix of zeros, the vector of ones and the identity matrix of appropriate dimensions. Further, $\epsilon > 0$ is a small positive constant and $\{0, 1\}^n$ denotes a set of n-dimensional binary vectors. The diag and vec operators are defined for a set of matrices $\{M_i\}_{i=m}^n$ and matrix M as:

$$\mathrm{diag}_{k=\{i,j\}}\{M_k\} = \begin{bmatrix} M_i & \mathbf{0} \\ \mathbf{0} & M_j \end{bmatrix}, \mathrm{vec}_{k=\{i,j\}}\{M_k\} = \begin{bmatrix} M_i \\ M_j \end{bmatrix},$$

$$\mathrm{diag}_{i=m}^n\{M_i\} = \begin{bmatrix} M_m & & \\ & \ddots & \\ & & M_n \end{bmatrix}, \mathrm{vec}_{i=m}^n\{M_i\} = \begin{bmatrix} M_m \\ \vdots \\ M_n \end{bmatrix},$$

$$\mathrm{diag}_N\{M\} = \mathbb{I}_N \otimes M, \mathrm{vec}_N\{M\} = \mathbb{1}_N \otimes M,$$

where $\otimes$ is the Kronecker product. The sets of positive and non-negative integers up to n are denoted by $\mathbb{Z}_n^+$ and $\mathbb{Z}_n^0$, respectively. Given two sets $\mathcal{U}, \mathcal{V} \in \mathbb{R}^n$, the Minkowski sum is defined as $\mathcal{U} \oplus \mathcal{V} \triangleq \{u + v \mid u \in \mathcal{U}, v \in \mathcal{V}\}$, the Pontryagin set difference $\mathcal{U} \ominus \mathcal{V} \triangleq \{u \mid u \oplus \mathcal{V} \subseteq \mathcal{U}\}$, and the set minus operation $\mathcal{U} \setminus \mathcal{V} \triangleq \{u \in \mathcal{U} : u \notin \mathcal{V}\}$. For a given matrix $K \in \mathbb{R}^{m\times n}$, $K\mathcal{U} \triangleq \{Ku \mid u \in \mathcal{U}\}$.

Definition 1 (SOS-1 Set): A special ordered set of degree 1 (SOS-1) constraint is a set of integer, continuous or mixed-integer scalar variables for which at most one variable in the set may take a value other than zero, denoted as SOS-1: $\{v_1, \dots, v_N\}$. For instance, if $v_i \neq 0$, then this constraint imposes that $v_j = 0$ for all $j \neq i$.

Further, we overload this constraint for vectors, where for a pair of vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$, SOS-1:$\{\mathbf{u}, \mathbf{v}\}$ denotes element-wise constraints SOS-1:$\{u_i, v_i\}$ for all i.

Note that SOS-1 constraints can be incorporated into most off-the-shelf optimization solvers, e.g., Gurobi [11].

A. Signal Temporal Logic (STL)

Let Σ be a finite set of predicates. The syntax of signal temporal logic (STL) formulas over Σ is given by:

$$\varphi ::= \top \mid p^\mu \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \mathcal{U}_{[t_1,t_2]} \varphi_2, \quad (1)$$

where $\top$ is the *true* predicate, $p^\mu \in \Sigma$ is a predicate whose truth value is determined by the sign of its underlying predicate function $\mu : \mathbb{R}^n \to \mathbb{R}^m$ and it is true if $\mu(x) \geq 0$ and false otherwise, while $\neg$, $\vee$, and $\mathcal{U}_{[t_1,t_2]}$ are the negation, disjunction, and time-constrained until operators, respectively, and $[t_1, t_2] \subset [0, \infty)$ is an interval of reals. Applying the grammar given in (1), we can also define next ($\bigcirc$; for discrete-time systems), conjunction ($\wedge$), implication ($\Rightarrow$), eventually in $[t_1, t_2]$ ($\Diamond_{[t_1,t_2]}$), and always in $[t_1, t_2]$ ($\Box_{[t_1,t_2]}$) as $\bigcirc\varphi = \top\mathcal{U}_{[0,1]}\varphi$, $\varphi_1 \wedge \varphi_2 = \neg(\neg\varphi_1 \vee \neg\varphi_2)$, $\varphi_1 \Rightarrow \varphi_2 = \neg\varphi_1 \vee \varphi_2$, $\Diamond_{[t_1,t_2]}\varphi = \top\mathcal{U}_{[t_1,t_2]}\varphi = \bigvee_{\tau=t_1}^{t_2} \bigcirc^\tau\varphi$, and $\Box_{[t_1,t_2]}\varphi = \neg\Diamond_{[t_1,t_2]}\neg\varphi = \bigwedge_{\tau=t_1}^{t_2} \bigcirc^\tau\varphi$, respectively. Further, we abbreviate $\mathcal{U}_{[0,\infty)}$, $\Diamond_{[0,\infty)}$, $\Box_{[0,\infty)}$ as $\mathcal{U}$, $\Diamond$, $\Box$.

Definition 2 (STL Semantics): Let $\boldsymbol{x}$ be an ω-word of real-valued vector signal over $\mathcal{X} \subseteq \mathbb{R}^n$, i.e., $\boldsymbol{x} \in \mathcal{X}^\omega$ and let $\boldsymbol{\sigma}$ be the ω-word over Σ corresponding to the predicate of an STL formula φ, i.e., $\boldsymbol{\sigma} \in \Sigma^\omega$, and let $x(t)$ and $\sigma(t)$ be t^{th} element of $\boldsymbol{x}$ and $\boldsymbol{\sigma}$. The STL semantics is given by:

1) $(\boldsymbol{x}, t) \models p^\pi \Leftrightarrow \pi(x(t)) \geq 0$ (i.e., $\sigma(t) = p^\pi = \top$),
2) $(\boldsymbol{x}, t) \models \neg\varphi \Leftrightarrow (\boldsymbol{x}, t) \not\models \varphi$,
3) $(\boldsymbol{x}, t) \models \varphi_1 \vee \varphi_2 \Leftrightarrow (\boldsymbol{x}, t) \models \varphi_1$ or $(\boldsymbol{x}, t) \models \varphi_2$,
4) $(\boldsymbol{x}, t) \models \varphi_1 \wedge \varphi_2 \Leftrightarrow (\boldsymbol{x}, t) \models \varphi_1$ and $(\boldsymbol{x}, t) \models \varphi_2$,
5) $(\boldsymbol{x}, t) \models \varphi_1\mathcal{U}_{[t_1,t_2]}\varphi_2 \Leftrightarrow \exists t' \in [t+t_1, t+t_2] : (\boldsymbol{x}, t') \models \varphi_2$ and $\forall t'' \in [t, t'] : (\boldsymbol{x}, t'') \models \varphi_1$,
6) $(\boldsymbol{x}, t) \models \Diamond_{[t_1,t_2]}\varphi \Leftrightarrow \exists t' \in [t + t_1, t + t_2], (\boldsymbol{x}, t') \models \varphi$,
7) $(\boldsymbol{x}, t) \models \Box_{[t_1,t_2]}\varphi \Leftrightarrow \forall t' \in [t + t_1, t + t_2], (\boldsymbol{x}, t') \models \varphi$.

We write $\boldsymbol{x} \models \varphi$ if $(\boldsymbol{x}, 0) \models \varphi$.

Next, leveraging [10], we present the mixed-integer encoding of STL formulas for the satisfaction of STL semantics, i.e., $(\boldsymbol{x}, t) \models \varphi$. First, we equivalently express $(\boldsymbol{x}, t) \models p^\pi$, i.e.,

$$P_{p^\pi} = 1 \Leftrightarrow \pi(x) \geq 0,$$

using the following mixed-integer encoding:

$$\begin{aligned} &\pi(x) + s^+ \geq 0, \quad \pi(x) < s^-, \\ &\text{SOS-1} : \{P_{p^\pi}\mathbb{1}, s^+\}, \quad \text{SOS-1} : \{\big(1 - P_{p^\pi}\big)\mathbb{1}, s^-\}, \end{aligned} \quad (2)$$

where s^+ and s^- are unconstrained real-valued slack variables. Then, we present the encodings of the following operators of the STL semantics, where p^π, p^ϕ and p^{π_i} are predicates, and P_φ^t is the truth value of formula φ at time t.

Negation: The formula $\varphi = \neg p^\pi$ can be modeled as:

$$P_\varphi^t = \left(1 - P_{p^\pi}^t\right). \quad (3)$$

Disjunction: The formula $\varphi = \bigvee_{i=1}^k p^{\pi_i}$ can be modeled as:

$$P_\varphi^t \leq \Sigma_{i=1}^k P_{p^{\pi_i}}^t; \quad P_\varphi^t \geq P_{p^{\pi_i}}^t, i \in \mathbb{Z}_+^k. \quad (4)$$

Conjunction: The formula $\varphi = \bigwedge_{i=1}^k p^{\pi_i}$ can be modeled as:

$$P_\varphi^t \geq \Sigma_{i=1}^k P_{p^{\pi_i}}^t - (k - 1); \quad P_\varphi^t \leq P_{p^{\pi_i}}^t, i \in \mathbb{Z}_+^k. \quad (5)$$

Next: The formula $\varphi = \bigcirc p^\pi$ can be modeled as:

$$P_\varphi^t = P_{p^\pi}^{t+1}. \quad (6)$$

Until: The formula $\varphi = p^\pi \mathcal{U}_{[t_1,t_2]} p^\phi$ can be modeled as:

$$\begin{aligned} &\alpha_{tj} \geq P_{p^\phi}^j + \Sigma_{\tau=t}^{j-1} P_{p^\pi}^\tau - (j - t), j \in \mathbb{Z}_{t+t_1}^{t+t_2}; \\ &\alpha_{tj} \leq P_{p^\phi}^j, \ \alpha_{tj} \leq P_{p^\pi}^\tau, j \in \mathbb{Z}_{t+t_1}^{t+t_2}, \tau \in \mathbb{Z}_t^{j-1}; \\ &P_\varphi^t \leq \Sigma_{j=t+t_1}^{t+t_2} \alpha_{tj}, P_\varphi^t \geq \alpha_{tj}, j \in \mathbb{Z}_{t+t_1}^{t+t_2}. \end{aligned} \quad (7)$$

Eventually: The formula $\varphi = \Diamond_{[t_1,t_2]} p^\pi$ can be modeled as:

$$P_\varphi^t \leq \Sigma_{\tau=t+t_1}^{t+t_2} P_{p^\pi}^\tau; \quad P_\varphi^t \geq P_{p^\pi}^\tau, \tau \in \mathbb{Z}_{t+t_1}^{t+t_2}. \quad (8)$$

Always: The formula $\varphi = \Box_{[t_1,t_2]} p^\pi$ can be modeled as:

$$P_\varphi^t \geq \Sigma_{\tau=t+t_1}^{t+t_2} P_{p^\pi}^\tau - (t_2 - t_1); \quad P_\varphi^t \leq P_{p^\pi}^\tau, \tau \in \mathbb{Z}_{t+t_1}^{t+t_2}. \quad (9)$$

Further, to obtain algorithms with a finite number of variables, we assume that the formulas φ are *time-bounded*, i.e., of the (unbounded global/safety) form: $\varphi = \phi_b \wedge \Box\phi_g$ where ϕ_b and ϕ_g are bounded negation-free formulas with bounds b^{ϕ_b} and b^{ϕ_g} (cf. [10, Definition 4]).

III. Problem Formulation

A. Modeling Framework

We consider a collaborative setting where an agent must coordinate with other agents to achieve tasks without "verbally"/explicitly communicating its intent to its teammates. Communication is "non-verbally"/indirectly achieved by planning and executing intent-expressive trajectories that allow its stationary teammates to accurately infer the intent. We consider affine system models with state and output equations:

$$x(k + 1) = Ax(k) + Bu(k) + Ww(k) + f, \quad (10)$$

$$z(k) = Cx(k) + D_u u(k) + D_v v(k) + g, \quad (11)$$

with states $x \in R^n$, inputs $u \in R^m$, outputs $z \in R^{n_z}$, process noise $w \in R^{m_w}$ and measurement noise $v \in R^{m_v}$.

The initial condition, denoted by $x_0 = x(0)$, as well as the noise signals $w(k)$ and $v(k)$ for all time steps k are uncertain but bounded by polyhedral sets:

$$y \in \mathcal{Y} = \{y \in \mathbb{R}^a \mid P_b y \leq p_b\}, \quad (12)$$

for $y \in \{x(0), w(k), v(k), \forall k\}$ with $\mathcal{Y} \in \{\mathcal{X}_0, \mathcal{W}, \mathcal{V}\}$, $a \in \{n, m_w, m_v\}$ and $b \in \{0, w, v\}$, respectively.

The system input u is constrained to a polyhedral set:

$$u \in \mathcal{U} \triangleq \{u \in \mathbb{R}^m \mid P_u u \leq p_u\}, \tag{13}$$

and the state $x(k)$ at all times k must be controlled to *robustly* remain within a collision-free space under worst-case uncertainties $\mathcal{X}_0$, $\mathcal{W}$ and $\mathcal{V}$, i.e., $x(k) \in \mathcal{X}_s \triangleq \mathcal{X} \setminus (\bigcup_j \mathcal{O}_j)$, where $\mathcal{X}$ and $\mathcal{O}_j, \forall j \in \mathbb{Z}^+_{N_{obs}}$ are polyhedral:

$$\mathcal{X} \triangleq \{x \in \mathbb{R}^n \mid P_x x \leq p_x\}, \mathcal{O}_j \triangleq \{x \in \mathbb{R}^n \mid P^j_\otimes x \leq p^j_\otimes\}. \tag{14}$$

Finally, the *intent* i of an agent is modeled as a task specified by a time-bounded STL, φ^i, whose predicate functions are all linear inequalities. For instance, an intent i may be to eventually reach and stay in a goal region, i.e., $\varphi^i = \Diamond_{[0,T_f]}\Box P^i_g x \leq p^i_g$. Then, each intent model is a tuple $\mathcal{G}^i = \{A, B, W, C, D_u, D_v, f, g, \varphi^i\}$.

B. Problem Statement

This letter will address two complementary problems, where the observed agent solves an intent-expressive trajectory planning problem to find an output feedback control policy to produce trajectories that implicitly/"non-verbally" encodes/conveys the intended goal, while the observer agents solves an intent estimation problem to infer/decode the intent at run time from (only) observed output trajectories.

Problem 1 (Intent-Expressive Trajectory Planning): For a given set of affine models $\{\mathcal{G}^i\}_{i=1}^{N_g}$ with state, input and uncertainty sets, $\mathcal{X}$, $\mathcal{U}$ and $\{\mathcal{X}_0, \mathcal{W}, \mathcal{V}\}$, respectively, find a set of N_g output-feedback control policies/laws at each time step k, $\{\pi_i(\{z(\kappa), \kappa\}_{\kappa=0}^k)\}_{i=1}^{N_g} \in \mathcal{U}$, such that the resulting closed-loop output feedback policies/laws

1) yield distinct/disjoint output trajectories corresponding to each intent that differ by a threshold ϵ in at least one time instance within the first T steps,[1]
2) while staying within $\mathcal{X}_s$ for all $k \in \mathbb{Z}^0_{T_f}$,
3) and attaining the intents φ^i by time step T_f,

despite worst-case realizations of $x(0) \in \mathcal{X}_0$, $w(k) \in \mathcal{W}$ for all $k \in \mathbb{Z}^0_{T_f}$ and $v(k) \in \mathcal{V}$ for all $k \in \mathbb{Z}^0_{T-1}$.

Problem 2 (Intent Estimation): Given the current time $t \in \mathbb{Z}^0_T$ and the measured/observed noisy output trajectory $z_t = \{z(k)\}_{k=0}^t$, determine which intent models are compatible with the observations and eliminate incompatible intent models until only one compatible intent is left (that by construction, is the intended intent model).

The requirements laid out in Problem 1 can be encoded using time-bounded STLs. In this case, we refer to specifications 1 & 2 as the *separation* and *safety* conditions, respectively, while specification 3 is the *intent* statement. In the next sections, we describe how this encoding can be done.

IV. Main Approach

In this section, we present a formulation for designing closed-loop intent-expressive trajectory planning and intent estimation algorithms. Our approach for designing the intent-expressive trajectory planner with output feedback is informed by tube-based predictive control, e.g., [12], which provides a framework for ensuring that our uncertainty sets do not grow exponentially but remains invariant.

[1]Our goal is to enable the intent models to be inferred from observations from a (much) shorter time horizon $T < T_f$ before the agent/robot attains its intent in T_f time steps (i.e., intent-expressive motion planning).

A. Tube-Based Output Feedback and Invariant Sets

We begin by presenting a tube-based output feedback design. This design, as in [4], involves the designs of: (i) a state estimator for each $i \in \mathbb{Z}^+_{N_g}$:

$$\hat{x}_i(k+1) = (A - LC)\hat{x}_i(k) + Bu_i(k) + Lz_i(k) + f + Lg, \tag{15}$$

where $z(k)$ is the measured output at run time and L is any observer gain such that $(A-LC)$ is Schur stable, (ii) a nominal system for each $i \in \mathbb{Z}^+_{N_g}$:

$$\bar{x}_i(k+1) = A\bar{x}_i(k) + B\bar{u}_i(k) + f, \tag{16}$$

as well as (iii) an output feedback law for each $i \in \mathbb{Z}^+_{N_g}$:

$$u_i(k) = \bar{u}_i(k) - K\big(\hat{x}_i(k) - \bar{x}_i(k)\big), \tag{17}$$

with any feedback gain K such that $(A - BK)$ is Schur stable. Note that the feedback term $-K(\hat{x}_i(k) - \bar{x}_i(k))$ is the low-level controller in a tube-based framework, while the nominal state $\bar{x}_i(k)$ satisfying the dynamics in (16) and the nominal input $\bar{u}_i(k)$ are designed by the intent-expressive trajectory planner in Section IV-B.

Next, we define the estimation error $\tilde{x}_i \triangleq x_i - \hat{x}_i$, and the tracking error $s_i \triangleq x_i - \bar{x}_i$, whose dynamics are given by

$$\tilde{x}_i(k+1) = (A - LC)\tilde{x}_i(k) + Ww(k) - LD_v v(k), \tag{18}$$

$$s_i(k+1) = (A - BK)s_i(k) + BK\tilde{x}_i(k) + Ww(k), \tag{19}$$

and review the notion of robust positive invariance below that we will leverage to ensure that the estimation and tracking error sets are small and not growing with time.

Definition 3 (RPI Set): Given a system $x(k+1) = Ax(k) + w(k)$, $\Omega \subset \mathbb{R}^n$ is a robust positively invariant (RPI) set if for all $x(k) \in \Omega$, and $w(k) \in \mathcal{W}$, $x(k+1)$ satisfies $x(k+1) \in \Omega$.

The minimal RPI (mRPI) set is the smallest RPI set that exists and can be found using the methods outlined in [13]. Using this, we propose to compute the mRPI sets $\mathcal{E}$ and $\mathcal{S}$ for the estimation and tracking errors, $\tilde{x}$ and s, respectively,

$$\tilde{x}_i(k) \in \mathcal{E} = \{\tilde{x} \in \mathbb{R}^n : P_{\tilde{x}}\tilde{x} \leq p_{\tilde{x}}\}, \tag{20}$$

$$s_i(k) \in \mathcal{S} = \{s \in \mathbb{R}^n : P_s s \leq p_s\}. \tag{21}$$

such that we can guarantee that the actual state $x(k)$ remains close to the estimated and nominal states, as follows:

$$x_i(k) = \hat{x}_i(k) + \tilde{x}_i(k) \in \{\hat{x}(k)\} \oplus \mathcal{E},$$
$$x_i(k) = \bar{x}_i(k) + s_i(k) \in \{\bar{x}(k)\} \oplus \mathcal{S}.$$

Then, the output equation in (11) can be rewritten as

$$z_i(k) = C\bar{x}_i(k) + Cs_i(k) + D_u\bar{u}_i(k) + Vv(k) + g, \tag{22}$$

and similarly, from (17), the output feedback law is

$$u_i(k) = \bar{u}_i(k) - K\big(\hat{x}_i(k) - \bar{x}_i(k)\big) = \bar{u}_i(k) + K(\tilde{x}_i(k) - s_i(k)). \tag{23}$$

Thus, the input set $\mathcal{U}$ in (13) can be rewritten to obtain

$$\bar{u} \in \bar{\mathcal{U}} = \mathcal{U} \ominus K(\mathcal{E} \ominus \mathcal{S}) \triangleq \{Q_u\bar{u} \leq q_u\}, \tag{24}$$

while the state constraints $x_i(k) \in \mathcal{X}_s = \mathcal{X} \setminus \bigcup_j \mathcal{O}_j$ can be rewritten with the sets $\mathcal{X}$ and $\mathcal{O}_j$ given by

$$\mathcal{X} : P_x(\bar{x}_i(k) + s_i(k)) \leq p_x, \ \ \mathcal{O}_j : P^j_\otimes(\bar{x}_i(k) + s_i(k)) \leq p^j_\otimes. \tag{25}$$

Moreover, we define $P_\otimes \triangleq \mathrm{vec}_{j=1}^{N_{obs}}\{P^j_\otimes\}$ and $p_\otimes \triangleq \mathrm{vec}_{j=1}^{N_{obs}}\{p^j_\otimes\} - \epsilon\mathbb{1}$ such that the state x_i must stay outside of $\mathcal{O} = \bigcup_j \mathcal{O}_j \triangleq \{P_\otimes x_i < p_\otimes\}$. Further, the constraints for $\mathcal{X}$ in (25) can be equivalently written as constraints on the nominal states $\bar{x}_i(k)$ and $\bar{x}_i(T_f)$, by considering the worst-case $s_i \in \mathcal{S}$ in a procedure commonly known as constraint tightening [12], as follows:

$$\bar{x}_i(k) \in \bar{\mathcal{X}} = \mathcal{X} \ominus \mathcal{S} \triangleq \{x \in \mathbb{R}^n \mid Q_x x \leq q_x\}. \quad (26)$$

Finally, we derive the tightened constraint on $\bar{x}_i(0) = \bar{x}_{i,0}$ that ensures that $\mathcal{X}_0$ is contained in the tracking error set centered at $\bar{x}_{i,0}$, i.e., $\mathcal{X}_0 \subseteq \{\bar{x}_{i,0}\} \oplus \mathcal{S}$ and consequently, $-\bar{x}_{i,0} \in \mathcal{S} \ominus \mathcal{X}_0$ that can be rewritten in terms of $\bar{x}_{i,0}$ as

$$\bar{x}_{i,0} \in \bar{\mathcal{X}}_0 = -(\mathcal{S} \ominus \mathcal{X}_0) \triangleq \{\bar{x}_0 \in \mathbb{R}^n \mid Q_0 \bar{x}_0 \leq q_0\}. \quad (27)$$

B. Output Feedback Intent-Expressive Trajectory Planner

Having introduced the output feedback control framework in Section IV-A above, our nominal system model can now be defined as $\bar{\mathcal{G}}^i = \{A, B, C, D_u, D_v, f, g, \varphi^i\}$. And subsequently, we can now restate Problem 1 to reflect the nominal system variables $\{\bar{x}, \bar{u}\}$ and uncertainty sets $\{\mathcal{E}, \mathcal{S}\}$.

Problem 3 (Intent-Expressive Trajectory Planning): For a given set of affine models $\{\bar{\mathcal{G}}^i\}_{i=1}^{N_g}$ with state, input and uncertainty sets, $\bar{\mathcal{X}}$, $\mathcal{U}$ and $\{\bar{\mathcal{X}}_0, \mathcal{E}, \mathcal{S}, \mathcal{V}\}$, respectively, find a set of N_g (causal) output-feedback control policies/laws at each time step k, $\{\pi_i(\{z(\kappa), \kappa\}_{\kappa=0}^{k})\}_{i=1}^{N_g} \in \mathcal{U}$, to minimize a given cost function $\sum_{i=1}^{N_g} J(\mu_i(\{z(\kappa), \kappa\}_{\kappa=0}^{T_f}, \mathcal{X}_{g,i}))$, such that the resulting closed-loop output feedback policies/laws

1) yield distinct/disjoint output trajectories corresponding to each intent that differ by a threshold ϵ in at least one time instance within the first T steps,
2) while staying within $\mathcal{X}_s$ for all $k \in \mathbb{Z}_{T_f}^0$,
3) and attaining the intents φ^i by time step T_f,

despite worst-case realizations of $\bar{x}(0) \in \bar{\mathcal{X}}_0$, $s(k) \in \mathcal{S}$ for all $k \in \mathbb{Z}_{T_f+1}^0$, and $\tilde{x}(k) \in \mathcal{E}$, $v(k) \in \mathcal{V}$ for all $k \in \mathbb{Z}_{T-1}^0$.

Our approach to solve the above problem is to formulate the three requirements above (referred to as *separation*, *safety* and *intent*, respectively) using STL and then to find the equivalent mixed-integer linear formulation via the following lemmas, before integrating them into a single mixed-integer linear program that can optimally solve Problem 3 using any off-the-shelf solvers, e.g., Gurobi [11].

Lemma 1 (Separation as an STL): Given a pair $q = (i, j)$, $i \neq j$ of intent models, the separation condition, i.e., the requirement that the output trajectories for both intents must differ by a threshold $\epsilon > 0$ in at least one time instance within the first T steps, can be encoded as a time-bounded STL:

$$\Psi^q \triangleq \Diamond_{[0,T-1]} \neg \big(\exists \omega_{i,T}, \omega_{j,T} \in \Omega {:} \|z_i - z_j\|_\infty \leq \epsilon\big), \quad (28)$$

where $\omega_{i,T} \triangleq \big[s_{i,T}^\intercal \ \tilde{x}_{i,T}^\intercal \ v_{i,T}^\intercal\big]^\intercal$ and similarly for j, and $\Omega \triangleq \mathcal{S}^T \times \mathcal{E}^T \times \mathcal{V}^T$, with the time-concatenated vector notations in the Appendix. Moreover, the corresponding mixed-integer linear formulation for Ψ^q is given by:

$$\begin{gathered}
\begin{array}{l} \forall k \in \mathbb{Z}_{T-1}^0 \\ \forall \ell \in \mathbb{Z}_{2n_z}^+ \end{array} : \begin{bmatrix} (\tilde{q}_k^q)^\intercal \\ (\tilde{P}_k^q)^\intercal \\ -(\tilde{P}_k^q)^\intercal \\ -\mathbb{I} \end{bmatrix} \Pi_{k,i}^q \leq \begin{bmatrix} -\Big(\Gamma_{u,k}^{q,\ell} \bar{u}_k^q + \Gamma_{x,k}^{q,\ell} \bar{x}_0^q + \epsilon\Big) \\ -\Big(\Gamma_{\omega,k}^{q,\ell}\Big)^\intercal \\ \Big(\Gamma_{\omega,k}^{q,\ell}\Big)^\intercal \\ 0 \end{bmatrix} + s_i^q \mathbb{1}, \\
b_{q,k}^\Psi \in \{0,1\}^{2n_z}, \ \tilde{s}_q^\Psi \in \mathbb{R}^T, \ a_q^\Psi \in \{0,1\}^T, \\
\textstyle\sum_{i=1}^{2n_z} b_{q,k}^\Psi(i) + \tilde{s}_q^\Psi(k) \geq 1, \ \sum_{k=0}^{T-1} a_q^\Psi(k) \geq 1, \qquad (29) \\
\text{SOS-1} : \{b_{q,k}^\Psi, s^q\}, \ \text{SOS-1} : \{a_q^\Psi, \tilde{s}_q^\Psi\},
\end{gathered}$$

with the vector and matrix definitions in the Appendix and the superscript ℓ denotes the ℓ-th row of given matrix.

Proof: It is straightforward to encode separation as the STL in (28) using the negation of the existence of uncertainties $\omega_{i,T}$ and $\omega_{j,T}$ that make the output trajectories indistinguishable up to a precision ϵ. This is equivalent to

$$\begin{aligned}
\Psi^q &= \Diamond_{[0,T-1]} \neg \begin{pmatrix} \exists \omega_k^q \in \{\tilde{P}_k^q \omega_k^q \leq \tilde{q}_k^q\}: \\ R_k^q \omega_k^q \leq S_k^q \bar{u}_k^q + Y_k^q \bar{x}_0^q + \epsilon \mathbb{1} \end{pmatrix} \\
&= \bigvee_{i=1}^{2n_z} \begin{pmatrix} \forall \omega_k^q \in \{\tilde{P}_k^q \omega_k^q \leq \tilde{q}_k^q\}: \\ \Gamma_{\omega,k}^{q,i} \omega_k^q \geq \Gamma_{u,k}^{q,i} \bar{u}_k^q + \Gamma_{x,k}^{q,i} \bar{x}_0^q + \epsilon \end{pmatrix} \\
&= \bigvee_{i=1}^{2n_z} \begin{pmatrix} (\tilde{q}_k^q)^\intercal \Pi_{k,i}^q \leq -\Big(\Gamma_{u,k}^{q,i} \bar{u}_k^q + \Gamma_{x,k}^{q,i} \bar{x}_0^q + \epsilon\Big) \\ (\tilde{P}_k^q)^\intercal \Pi_{k,i}^q = -\Big(\Gamma_{\omega,k}^{q,i}\Big)^\intercal \\ \Pi_{k,i}^q \qquad \geq 0 \end{pmatrix},
\end{aligned}$$

where the first equality uses the vector and matrix definitions in the Appendix, the second equality applies the encoding of $\Diamond$ using a disjunction and enforces the negation by flipping the inequality sign with a small ϵ (to account for strict inequality), while the final equality is obtained as the robust counterpart of the second using robust optimization [14]. Finally, the mixed-integer linear formulation in (29) is obtained by applying [15, Lemma 2] for encoding disjunctions. ■

Lemma 2 (Safety as an STL): For a given intent model i, the safety condition, defined as always avoiding collision with $\mathcal{O}$, can be specified as a time-bounded STL:

$$\Phi^i = \Box_{[0,T_f]} \neg \Big(\exists s_{i,T_f} \in \mathcal{S}^{T_f}, k \in \mathbb{Z}_{T_f}^0 : x_i(k) \in \mathcal{O}\Big), \quad (30)$$

with $\mathcal{O}$ defined below (25). Moreover, the corresponding mixed-integer linear formulation for Φ^i is given by:

$$\begin{gathered}
\begin{array}{l} \forall k \in \mathbb{Z}_{T_f}^0 \\ \forall j \in \mathbb{Z}_{N_{obs}}^+ \end{array} : \begin{bmatrix} \bar{P}_{s,k}^\intercal \\ \bar{P}_{s,k}^\intercal \\ -\bar{P}_{s,k}^\intercal \\ -\mathbb{I} \end{bmatrix} \Xi_k^i \leq \begin{bmatrix} -\Big(\Lambda_{u,k}^i \bar{u}_{i,k} + \Lambda_{x,k}^i \bar{x}_{i,0} - \Lambda_k^i - \eta_k^i\Big) \\ -\Big(\Lambda_{s,k}^i\Big)^\intercal \\ \Big(\Lambda_{s,k}^i\Big)^\intercal \\ 0 \end{bmatrix}, \\
b_{i,k}^\Phi \in \{0,1\}^{N_{obs}}, \ \tilde{s}_i^\Phi \in \mathbb{R}^{Tf+1}, \ a_i^\Phi \in \{0,1\}^{T_f+1}, \\
\textstyle\sum_{j=1}^{N_{obs}} b_{i,k}^\Phi(j) + \tilde{s}_i^\Phi(k) \geq 1, \ \sum_{k=0}^{T_f} a_i^\Phi(k) = T_f + 1, \quad (31) \\
\text{SOS-1} : \{b_{i,k}^\Phi(j), \eta_k^i(j)\}, \ \text{SOS-1} : \{a_i^\Phi(k), \tilde{s}_i^\Phi(k)\},
\end{gathered}$$

with the vector and matrix definitions in the Appendix.

Proof: Safety is enforced by negating the existence of any uncertainty, s_{T_f}, that drives the system into the unsafe region, which can be encoded as the STL in (30). Consequently,

$$\begin{aligned}
\Phi^i &= \Box_{[0,T_f]} \neg \begin{pmatrix} \exists s_{i,T_f} \in \{P_{s,T_f} s_{i,T_f} \leq p_{s,T_f}\}, \\ k \in \mathbb{Z}_{T_f}^0 \end{pmatrix} : P_\otimes x_i(k) \leq q_\otimes \Bigg) \\
&= \bigwedge_{k=0}^{T_f} \Big(\forall s_{i,k} \in \{\bar{P}_{s,k} s_{i,k} \leq \bar{p}_{s,k}\} : P_\otimes x_i(k) < q_\otimes + \eta^i(k)\Big) \\
&= \bigwedge_{k=0}^{T_f} \begin{pmatrix} \forall s_{i,k} \in \{\bar{P}_{s,k} s_{i,k} \leq \bar{p}_{s,k}\}: \\ \Lambda_{s,k}^i s_{i,k} \geq -\Lambda_{u,k}^i \bar{u}_{i,k} - \Lambda_{x,k}^i \bar{x}_{i,0} + \lambda_k^i + \eta_k^i \end{pmatrix},
\end{aligned}$$

where for the second equality, we enforce the negation by flipping the inequality sign and by introducing a slack variable/vector η, that must have at least one zero element to reflect that any one constraint violation would satisfy the negation, as well as enforce the $\Box$ operator with a conjunction. The third equality comes from expressing the inequalities in terms of $s_{i,k}$, $\bar{u}_{i,k}$ and $\bar{x}_{i,0}$ instead of $x(k)$. Finally, we obtain (31) by finding the robust counterpart of the former using robust optimization [14]. ■

Further, the mixed-integer linear encoding of the intent constraint given as STL specifications φ^i can be obtained as described in Section II-A. Finally, the intent-expressive trajectory planning algorithm is obtained with the intent constraint, the separation and safety constraints in Lemmas 1–2, and the input and state constraints in (33)–(34).

Complexity Analysis: For brevity, we mainly focus on the number of integer variables in the resulting optimization formulation for separation, since this is the main distinction from [4]. Specifically, our STL approach has $Q(2n_z+1)T$, while the previous bilevel optimization method in [4] has $(2Q((c_s+c_v)+n_z)T$, where c_s, c_v denote the number of inequalities of the uncertainty sets $\mathcal{S}, \mathcal{V}$. The decrease of $(2Q(c_s+c_v)-1)T$ integer variables is likely the reason for the observed speedup in solving times and for the increased tractability for problems with longer planning horizons.

C. Intent Estimation

Next, we present the intent estimation algorithm for the observer agents to decode the intended goal state. For brevity, we mainly recap the feasibility check algorithm needed for this purpose, and our readers are referred to [4, Sec. III-C] for more details on the complete algorithm.

Proposition 1 (Feasibility Check): [4, Proposition 1] Given a measured output trajectory $z_t = \{z(k)\}_{k=0}^{t}$ at current time $t \in \mathbb{Z}_{T-1}^{0}$, the i-th candidate intent model with φ^i and $u_t = \{u(k)\}_{k=0}^{t}$ is compatible/consistent with z_t if there exist $s_t = \{s(k)\}_{k=0}^{t}$, $\tilde{x}_t = \{\tilde{x}(k)\}_{k=0}^{t}$ and $v_t = \{v(k)\}_{k=0}^{t}$ such that:

$$\bar{x}_{i,t} = \bar{A}_t\bar{x}_{i,0} + \bar{B}_t\bar{u}_{i,t} + \bar{f}_t, \tag{32a}$$

$$u_t = \bar{u}_{i,t} + \bar{K}_t(\tilde{x}_{i,t} - s_{i,t}) \tag{32b}$$

$$z_t = \bar{E}_t\bar{x}_{i,0} + \bar{F}_{u,t}\bar{u}_{i,t} + \bar{F}_{s,t}s_t + \bar{F}_{\tilde{x},t}\tilde{x}_t + \bar{F}_{v,t}v_t + \tilde{g}_t, \tag{32c}$$

$$s(k) \in \mathcal{S}, \tilde{x}(k) \in \mathcal{E}, v(k) \in \mathcal{V}, \bar{u}(k) \in \bar{\mathcal{U}} \quad \forall k \in \mathbb{Z}_t^0, \tag{32d}$$

where the concatenated matrices are defined in the Appendix.

V. Illustrative Example

As a demonstration of our proposed method, we consider a similar example to [3], [4] (for the sake of comparison):

$$x(k+1) = x(k) + \mathrm{v}_x(k)\delta t, \tag{a}$$

$$\mathrm{v}_x(k+1) = \left(1 - \frac{C_x}{M}\delta t\right)\mathrm{v}_x(k) + \frac{\delta t}{M}u_x(k) + w_x(k)\delta t, \tag{b}$$

$$y(k+1) = y(k) + \mathrm{v}_y(k)\delta t, \tag{c}$$

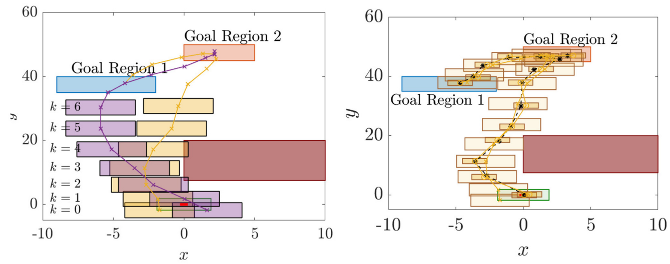
$$\mathrm{v}_y(k+1) = \left(1 - \frac{C_y}{M}\delta t\right)\mathrm{v}_y(k) + \frac{\delta t}{M}u_y(k) + w_y(k)\delta t, \tag{d}$$

where x and y (in m) are the positions of the agent/rover, v_x and v_y (in m/s) are the velocities in the x and y directions, respectively. Similarly, u_x, u_y and w_x, w_y (in N) are the acceleration input and process noise signals in the x and y directions, respectively. We also assume that x and y are the only observed states for intent estimation over a time horizon of length $T = 6$ with signals bounded such that:

$$u_x(k), u_y(k) \in [-8000, 8000]\,N, \quad w_x(k), w_y(k) \in [-50, 50]\,N,$$
$$\mathrm{v}_x(k), \mathrm{v}_y(k) \in [-1, 1]\,m.$$

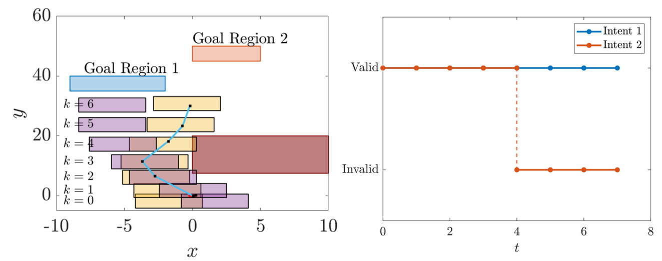
The initial state set is $\mathcal{X}_0 = \{x_0\} = \{[0.25\ 0.1\ 0.25\ 0.1]^\top\}$, while the obstacle $\mathcal{O}$ and state constraints $\mathcal{X}$ are given by

$$\mathcal{O} = \left\{\begin{bmatrix} x \\ y \end{bmatrix} : \begin{matrix} 0 \le x \le 10, \\ 7.5 \le y \le 20 \end{matrix}\right\} \quad \mathcal{X} = \left\{\begin{bmatrix} x \\ y \\ \mathrm{v}_x \\ \mathrm{v}_y \end{bmatrix} : \begin{matrix} -100 \le x \le 100, \\ -100 \le y \le 100, \\ -45 \le \mathrm{v}_x \le 45 \\ -10 \le \mathrm{v}_y \le 45 \end{matrix}\right\},$$



(a) Output reachable sets (magenta and yellow boxes with borders) showing separation of output trajectories within $k = 5$ time steps. (b) Actual state trajectory (black line with $*$ markers) for agent satisfying intent φ^1 (visit goal region 2 and then 1).

Fig. 1. Intent-expressive motion planning results (cf. Section IV-B). Nominal state trajectories are marked with $\times$ and estimated state trajectories with $\circ$, with the corresponding tracking and estimation error sets (larger and smaller boxes without borders in Fig. 1(b)).



(a) Observed (partial) output trajectory for intent estimation. (b) Output trajectory in Fig. (2a) is only compatible with φ^1.

Fig. 2. Intent estimation results (cf. Section IV-C).

and the intent is specified using an STL formula $\varphi^i \triangleq \Diamond_{[0,T_1]}(x_i \in \mathcal{X}_{g,i} \wedge \Diamond_{[1,T_f-T_1]}[x_i \in \mathcal{X}_{g,j}])$ with $j = \{1,2\} \setminus i$,

$$\mathcal{X}_{g,1} = \left\{\begin{bmatrix} x \\ y \\ \mathrm{v}_x \\ \mathrm{v}_y \end{bmatrix} : \begin{matrix} -5 \le x \le 2, \\ 35 \le y \le 40 \\ -9 \le \mathrm{v}_x \le 9 \\ -9 \le \mathrm{v}_y \le 9 \end{matrix}\right\}, \mathcal{X}_{g,2} = \left\{\begin{bmatrix} x \\ y \\ \mathrm{v}_x \\ \mathrm{v}_y \end{bmatrix} : \begin{matrix} 0 \le x \le 5, \\ 50 \le y \le 45 \\ -9 \le \mathrm{v}_x \le 9 \\ -9 \le \mathrm{v}_y \le 9 \end{matrix}\right\}.$$

Other parameters: $\epsilon = \varepsilon = 0.001$, $\lambda = 10^{15}$, $T_f = 15$, $T_1 = 10$, $M = 500\,kg$, $\delta t = 0.75\,s$, $b = 0.02\,\frac{N}{m}$, $C_x = 50\,\frac{Ns}{m}$ and $C_y = 70\,\frac{Ns}{m}$. In Fig. 1, the output reachable sets become disjoint after $k = 5$ (indicating intent-expressiveness), while Fig. 2 shows that the intent ϕ^1 is identified within 4 time steps.

In a second case, the intents are STL formulas $\varphi^i = \Diamond_{[0,T_f]}\Box\,(x \in \mathcal{X}_{g,i})$, $i \in \{1,2\}$ with $T_f = 15$, such that the approach in [4] can also be applied for comparison. The CPU time using Gurobi [11] for the proposed STL formulation for intent-expressive trajectory planning is 33.6 s, whereas the bilevel approach in [4] did not terminate within 24 hours. This speedup is likely due to the reduction of about 6.5 times in the number of integer variables with our new approach.

VI. Conclusion

In this letter, we presented an STL formulation for set-based intent-expressive trajectory planning and intent estimation with (noisy) output feedback for multi-agent teams. The intent-expressive trajectory planning algorithm encoded intent information by making the output reachable sets for satisfying all STL task specifications disjoint, while the intent estimation algorithm decoded the intent by ruling out all intent models that are incompatible with run-time noisy observations. Our proposed approach has lower computational complexity and was found in simulations to result in much faster computation than a previous bilevel program formulation. Future work include extensions to continuous-time trajectories, nonlinear dynamics and environments with occlusion.

Appendix

A. Time-Concatenated Models and Constraints

We make use of time concatenated expressions to improve legibility of our models and constraints. Given a time horizon τ, we concatenate the states, outputs, inputs and noises:

$$\bar{x}_{i,\tau} \triangleq \mathrm{vec}_{k=0}^{\tau}\{\bar{x}_i(k)\},\ \ *_{i,\tau} \triangleq \mathrm{vec}_{k=0}^{\tau-1}\{*_i(k)\},$$

for all $* \in \{z, \bar{u}, s, \tilde{x}, v\}$. Then, (16)–(23) can be written as:

$$\begin{aligned}
\bar{x}_{i,\tau} &= \bar{A}_\tau \bar{x}_{i,0} + \bar{B}_\tau \bar{u}_{i,\tau} + \bar{f}_\tau,\\
z_{i,\tau} &= \bar{C}_\tau\left(\bar{x}_{i,\tau} + s_{i,\tau}\right) + \bar{D}_{u,\tau} u_{i,\tau} + \bar{D}_{v,\tau} v_{i,\tau} + \bar{g}_\tau\\
&= \bar{E}_\tau \bar{x}_{i,0} + \bar{F}_{u,\tau}\bar{u}_{i,\tau} + \bar{F}_{s,\tau} s_{i,\tau} + \bar{F}_{\tilde{x},\tau}\tilde{x}_{i,\tau} + \bar{F}_{v,\tau} v_{i,\tau} + \tilde{g}_\tau,
\end{aligned}$$

with

$$\bar{A}_\tau \triangleq \begin{bmatrix} \mathbb{I} \\ A \\ \vdots \\ A^\tau \end{bmatrix},\ \Theta_\tau \triangleq \begin{bmatrix} \mathbb{I} & \mathbf{0} & \cdots & \mathbf{0} \\ A & \mathbb{I} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ A^{\tau-1} & A^{\tau-2} & \cdots & \mathbb{I} \end{bmatrix},$$

$$\begin{aligned}
&\bar{B}_\tau \triangleq \begin{bmatrix} \mathbf{0} \\ \Theta_\tau \mathrm{diag}_\tau B \end{bmatrix},\ \bar{f}_\tau \triangleq \begin{bmatrix} \mathbf{0} \\ \Theta_\tau \mathrm{vec}_\tau f \end{bmatrix},\\
&\bar{C}_\tau \triangleq \left[\mathrm{diag}_\tau C\ \mathbf{0}\right],\ \bar{D}_{u,\tau} \triangleq \mathrm{diag}_\tau D_u,\ \bar{D}_{v,\tau} \triangleq \mathrm{diag}_\tau D_v,\\
&\bar{g}_\tau \triangleq \mathrm{vec}_\tau g,\ \bar{E}_\tau \triangleq \bar{C}_\tau \bar{A}_\tau,\ \bar{F}_{u,\tau} \triangleq \bar{D}_{u,\tau} + \bar{C}_\tau \bar{B}_\tau,\\
&\bar{F}_{s,\tau} \triangleq \mathrm{diag}_\tau(C - D_u K),\ \bar{F}_{\tilde{x},\tau} \triangleq \mathrm{diag}_\tau(D_u K),\\
&\bar{F}_{v,\tau} \triangleq \bar{D}_{v,\tau},\ \ \tilde{g}_\tau \triangleq \bar{C}_\tau \bar{f}_\tau + \bar{g}_\tau,\ \bar{K}_\tau \triangleq \mathrm{diag}_\tau K.
\end{aligned}$$

The corresponding constraints can be written as:

$$\bar{P}_{y,\tau} y_{i,\tau} \leq \bar{p}_{y,\tau},\ \bar{P}_{y,\tau} = \mathrm{diag}_\tau\{P_y\},\ \bar{p}_{y,\tau} = \mathrm{vec}_\tau\{p_y\},$$

for $y \in \{s, \tilde{x}, v\}$. We also define an auxiliary variable ω to represent the vector of all uncertain variables:

$$\omega_{i,\tau} = \begin{bmatrix} s_{i,\tau} \\ \tilde{x}_{i,\tau} \\ v_{i,\tau} \end{bmatrix},\ Q_{\omega,\tau} = \begin{bmatrix} \bar{P}_{s,\tau} & 0 & 0 \\ 0 & \bar{P}_{\tilde{x},\tau} & 0 \\ 0 & 0 & \bar{P}_{v,\tau} \end{bmatrix},\ q_{\omega,\tau} = \begin{bmatrix} \bar{p}_{s,\tau} \\ \bar{p}_{\tilde{x},\tau} \\ \bar{p}_{v,\tau} \end{bmatrix}.$$

Moreover, we concatenate the input constraint matrices in (24) across the time horizon $\tau \in \{T, T_f\}$ such that

$$\bar{Q}_u \bar{u}_{i,T_f} \leq \bar{q}_u,\ \text{with}\ \bar{Q}_u \triangleq \mathrm{diag}_\tau\{Q_u\},\ \bar{q}_u \triangleq \mathrm{vec}_\tau\{q_u\}. \quad (33)$$

By the same notion, we define

$$\bar{P}^i_{x,\tau} \triangleq \mathrm{diag}_{\tau+1}\{Q_x\},\ \bar{p}^i_{x,\tau} \triangleq \mathrm{vec}_{\tau+1}\{q_x\}.$$

We can then combine the initial state and state constraints from (26)–(27) as:

$$\bar{H}^i_{x,\tau}\bar{x}_{i,0} \leq \bar{h}^i_{x,\tau} - \begin{bmatrix} \mathbf{0} \\ \bar{P}^i_{x,\tau} \end{bmatrix} \bar{B}_\tau \bar{u}_{i,\tau}, \quad (34)$$

with

$$\bar{H}^i_{x,T} \triangleq \begin{bmatrix} [Q_0 \quad \mathbf{0}] \\ \bar{P}^i_{x,\tau} \end{bmatrix} \bar{A}_\tau,\ \bar{h}^i_{x,\tau} \triangleq \begin{bmatrix} q_0 \\ \bar{p}^i_{x,\tau} - \bar{P}^i_{x,\tau}\bar{f}_\tau \end{bmatrix}.$$

Further, we also concatenate the obstacle constraints over time to obtain

$$\begin{aligned}
&\bar{P}_{\otimes,\tau} \triangleq \mathrm{diag}_{\tau+1}\{P_\otimes\},\ \bar{p}_{\otimes,\tau} \triangleq \mathrm{vec}_{\tau+1}\{p_\otimes\},\\
&\Lambda_{x,\tau} \triangleq \bar{P}_{\otimes,\tau}\bar{A}_\tau,\ \ \Lambda_\tau \triangleq \bar{p}_{\otimes,\tau} - \bar{P}_{\otimes,\tau}\bar{f}_\tau,\\
&\Lambda_{u,\tau} \triangleq \bar{P}_{\otimes,\tau}\bar{B}_\tau,\ \ \ \Lambda_{s,\tau} \triangleq \bar{P}_{\otimes,\tau},\ \ \bar{\eta}_\tau \triangleq \mathrm{vec}_{k=0}^{\tau}\eta(k).
\end{aligned} \quad (35)$$

B. Pair-Concatenated Constraints

To ensure the outputs of the intent models are distinct from each other within a time horizon of $T \leq T_f$, we further introduce the model pair, which consists of two different models of $\mathcal{G}_i$. Considering N_g discrete-time affine models, i.e., $\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_{N_g}$, there are $Q = \binom{N_g}{2}$ model pairs and let $q \in \mathbb{Z}^+_Q$ denote the pair of models $(\mathcal{G}_i, \mathcal{G}_{i'})$. Next, we concatenate $\bar{x}^i_0$, $\bar{u}_{i,T}$ and $z_{i,T}$ for each model pair $q = (i, i')$ using $*^q \triangleq \mathrm{vec}_{\ell=\{i,i'\}}\{*_\ell\}$ for all $* \in \{\bar{x}_0, \bar{u}_T, z_T\}$. Then, the pairwise uncertainty constraints can be obtained as

$$\tilde{P}^q_T \bar{\omega}^q_T \leq \tilde{p}^q_T, \quad (36)$$

with $\tilde{P}^q_T \triangleq \mathrm{diag}_{\ell=\{i,i'\}}\{Q^\ell_{\omega,T}\}$, $\tilde{p}^q_T \triangleq \mathrm{vec}_{\ell=\{i,i'\}}\{q^\ell_{\omega,T}\}$.

Then, the difference between pairwise outputs $\|z^i_T - z^{i'}_T\|_\infty \leq \delta^q \mathbb{1}$ can be rewritten as

$$R^q \bar{\omega}^q_T \leq \delta^q \mathbb{1} + S^q \bar{u}^q_T + Y^q \bar{x}^q_T, \quad (37)$$

with

$$R^q \triangleq \begin{bmatrix} \begin{bmatrix} \bar{F}_{s,T} & \bar{F}_{\tilde{x},T} & \bar{F}_{v,T} \\ -\bar{F}_{s,T} & -\bar{F}_{\tilde{x},T} & -\bar{F}_{v,T} \end{bmatrix} \begin{bmatrix} -\bar{F}_{s,T} & -\bar{F}_{\tilde{x},T} & -\bar{F}_{v,T} \\ \bar{F}_{s,T} & \bar{F}_{\tilde{x},T} & \bar{F}_{v,T} \end{bmatrix} \end{bmatrix},$$

$$S^q \triangleq \begin{bmatrix} -\bar{F}_{u,T} & \bar{F}_{u,T} \\ \bar{F}_{u,T} & -\bar{F}_{u,T} \end{bmatrix},\ Y^q \triangleq \begin{bmatrix} -\bar{E}_T & \bar{E}_T \\ \bar{E}_T & -\bar{E}_T \end{bmatrix}.$$

Finally, we define the following for (29):

$$M_k \triangleq \begin{bmatrix} 0_{n_z \times p(k-1)} & \mathbb{1}_p & 0_{n_z \times p(2T-k)} \\ 0_{n_z \times p(T+k-1)} & \mathbb{1}_p & 0_{n_z \times p(T-k)} \end{bmatrix},$$

$$\Gamma^q_{x,k} \triangleq M_k Y^q,\ \Gamma^q_{u,k} \triangleq M_k S^q,\ \Gamma^q_{\omega,k} \triangleq M_k R^q.$$

References

[1] C. Lichtenthäler and A. Kirsch. "Legibility of robot behavior: A literature review." Apr. 2016. [Online]. Available: https://hal.science/hal-01306977

[2] A. D. Dragan, K. C. Lee, and S. S. Srinivasa, "Legibility and predictability of robot motion," in *Proc. 8th ACM/IEEE Int. Conf. Human–Robot Interact. (HRI)*, 2013, pp. 301–308.

[3] E. Gah, R. Niu, B. Geisel, and S. Z. Yong, "Set-based intent-expressive trajectory planning and intent estimation," *IEEE Control Syst. Lett.*, vol. 7, pp. 151–156, 2022.

[4] E. Gah and S. Z. Yong, "Closed loop intent-expressive trajectory planning and intent estimation," in *Proc. Amer. Control Conf.*, 2024, p. 4. [Online]. Available: https://tinyurl.com/yzs6wnze

[5] Y. Ding, F. Harirchi, S. Z. Yong, E. Jacobsen, and N. Ozay, "Optimal input design for affine model discrimination with applications in intention-aware vehicles," in *Proc. ACM/IEEE Int. Conf. Cyber Phys. Syst.*, 2018, p. 15.

[6] R. Nikoukhah and S. Campbell, "Auxiliary signal design for active failure detection in uncertain linear systems with a priori information," *Automatica*, vol. 42, no. 2, pp. 219–228, Feb. 2006.

[7] A. Donzé, "On signal temporal logic," in *Proc. Int. Conf. Runtime Verification*, 2013, pp. 382–383.

[8] R. S. Smith and J. C. Doyle, "Model invalidation: A connection between robust control and identification," in *Proc. IEEE Amer. Control Conf.*, 1989, pp. 1435–1440.

[9] F. Harirchi, S. Z. Yong, and N. Ozay, "Guaranteed fault detection and isolation for switched affine models," in *Proc. IEEE Conf. Decis. Control (CDC)*, 2017, pp. 5161–5167.

[10] R. Niu, S. M. Hassaan, L. Yang, Z. Jin, and S. Z. Yong, "Model discrimination of switched nonlinear systems with temporal logic-constrained switching," *IEEE Control Syst. Lett.*, vol. 6, pp. 151–156, 2021.

[11] Gurobi Optimization, Inc. "Gurobi Optimizer reference manual." 2015. [Online]. Available: http://www.gurobi.com

[12] D. Limón, I. Alvarado, T. Alamo, and E. F. Camacho, "Robust tube-based MPC for tracking of constrained linear systems with additive disturbances," *J. Process Control*, vol. 20, no. 3, pp. 248–260, 2010.

[13] S. V. Rakovic, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne, "Invariant approximations of the minimal robust positively invariant set," *IEEE Trans. Autom. Control*, vol. 50, no. 3, pp. 406–410, Mar. 2005.

[14] D. Bertsimas, D. Brown, and C. Caramanis, "Theory and applications of robust optimization," *SIAM Rev.*, vol. 53, no. 3, pp. 464–501, 2011.

[15] Q. Shen, R. Niu, and S. Z. Yong, "Tractable model discrimination for safety–critical systems with disjunctive and coupled constraints," *Nonlinear Anal. Hybrid Syst.*, vol. 46, no. 2, 2022, Art. no. 101217.