Design and Experimental Validation of Tube-based MPC for Timed-constrained Robot Planning

Alexandros Nikou, Shahab Heshmati-alamdari and Dimos V. Dimarogonas

***Abstract*— This paper deals with the design and experimental validation of a state-of-the art tube-based Model Predictive Control (MPC) for achieving time-constrained tasks. Given the uncertain nonlinear dynamics of the robot as well as a high-level task written in Metric Interval Temporal Logic (MITL), the goal is to design a feedback control law that guarantees the satisfaction of the task. The workspace is divided into Regions of Interest (RoI) and contains also unsafe regions (obstacles) that the robot should not visit. The feedback control law consists of two terms: a control input which is the outcome of a Finite Horizon Optimal Control (FHOCP); and a state feedback law that guarantees that the nominal trajectories are bounded within a tube centered along the nominal trajectories. The aforementioned control law guarantees that the robot is safely navigated through the RoI within certain time bounds. The proposed framework can handle the rich expressiveness of MITL and is experimentally tested with a Nexus mobile robot in our lab facilities. The experimental results show that the proposed framework is promising for solving real-life robotic as well as industrial problems.**

I. Introduction

Over the last years the field of controlling systems under formal verification constraints has been gaining significant research attention due to important applications in robotics, autonomous driving and industrial automation. The timed logic that has primarily been used is Metric Interval Temporal Logic [1]–[4]. The control synthesis under Metric Interval Temporal Logic (MITL) consists of three parts:

1) first, the dynamics of the robot are abstracted into a Weighted Transition System (WTS) by providing feedback control laws that can drive the robot between states; the time duration that the robot needs to navigate between the states is modeled as a weight to the transition system.
2) a product between the WTS and an automaton which accepts all words that satisfy the given formula is computed.
3) once a run is found in the product automaton, it maps back into a sequence of feedback laws that satisfy the given formula.

In practical applications, regarding the first part, the feedback control laws need to be appropriately designed such that the following specifications are taken into consideration: (a) state and input constraints; (b) obstacle avoidance; and (c) robustness against potential external disturbances and uncertainties.

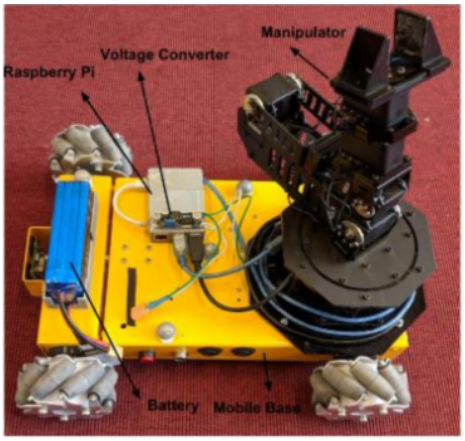

Fig. 1: A Nexus 10011 mobile robot with an attached 3 DoF manipulator.

A control technique that has been used for navigation of robotic agents with guaranteeing obstacle avoidance is the potential fields approach [5]. However, input constraints and potential external disturbances cannot be incorporated in the control design in a straightforward manner. In the same context, Prescribed Performance Controllers (PPC) that have been used for robotic navigation (see [6]) cannot handle input constraints as well as obstacle avoidance guarantees. Regarding real-time experiments, which is the main focus of the manuscript, both of the aforementioned methodologies require a significant amount of efforts in tuning the control gains. In particular, the potential fields approach requires computation of complicated formulas that consist of derivatives which might lead to numerical instabilities when applied to real platforms.

A feedback control law that has been recently proven to be efficient in incorporating the aforementioned specifications is the so-called tube-based Model Predictive Control (MPC) (see [7]–[10]). The idea of the tube-based MPC is based on the fact that the control law has two parts: a control law that is computed on-line and is the outcome of a FHOCP; and a state-feedback law designed offline and guarantees that the real system trajectories are always bounded in a tube whose volume depends on the bound of the disturbances as well as bounds of the derivatives of the dynamics. Experiments with MPC have been provided in [11]–[14] without incorporating any tube guarantees.

The contribution of this paper is to experimentally validate our recent theoretical results [2], [9], [10] in order

The authors are with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden. E-mail: {anikou, shaha, dimos}@kth.se. This work was supported by the H2020 ERC Grant BUCOPHSYS, the EU H2020 Co4Robots project, the Swedish Foundation for Strategic Research (SSF), the Swedish Research Council (VR) and the Knut och Alice Wallenberg Foundation (KAW).

to solve a time-constrained planning problem for a Nexus 10011 mobile robot (see Fig. 1). In particular, inside of the workspace there exist a set of Regions of Interest (RoI) some of which may be unsafe regions (obstacles). Given an MITL formula that the Nexus is desired to satisfy, the three steps control synthesis MITL framework is demonstrated in practice by application of a sequence of tube-based MPC laws that guarantees the satisfaction of the formula. The experimental results of the paper in hand verify the efficiency of the proposed framework which solves a general category of time-constrained robot navigation problems under state/input constraints, obstacle avoidance as well as uncertainties/disturbances. To the best of the authors' knowledge, this is the first time that such a general control synthesis framework in terms of dynamics, language expressiveness and state/input constraints is demonstrated. The experimental results show that the proposed framework is promising for solving real-life robotic as well as industrial problems.

II. Notation and Preliminaries

The n-fold Cartesian product of a set $\mathcal{S}$ and its cardinality are defined by $\mathcal{S}^n$ and $|\mathcal{S}|$, respectively; $\|y\|_2 := \sqrt{y^\top y}$ and $\|y\|_M := \sqrt{y^\top M y}$, $M \geq 0$ stand for the Euclidean and the weighted norm of a vector $y \in \mathbb{R}^n$, respectively; $\lambda_{\min}(M)$ stands for the minimum absolute value of the real part of the eigenvalues of $M \in \mathbb{R}^{n \times n}$; $0_{m \times n} \in \mathbb{R}^{m \times n}$ and $I_n \in \mathbb{R}^{n \times n}$ stand for the $m \times n$ matrix with all entries zeros and the identity matrix, respectively; The set $\mathcal{M}(\chi, r) := \{z \in \mathbb{R}^n : \|z - \chi\|_2 \leq r\}$ models a ball with center and radius $\chi \in \mathbb{R}^n$, $r > 0$, respectively. Given the sets $\mathcal{S}_1, \mathcal{S}_2 \subseteq \mathbb{R}^n$ and the matrix $M \in \mathbb{R}^{n \times m}$, the *matrix-set multiplication*, the *Minkowski addition* and the *Pontryagin difference* are respectively defined by: $M \circ \mathcal{S} := \{m : \exists s \in \mathcal{S}, m = Ms\}$, $\mathcal{S}_1 \oplus \mathcal{S}_2 := \{s_1 + s_2 : s_1 \in \mathcal{S}_1, s_2 \in \mathcal{S}_2\}$, and $\mathcal{S}_1 \ominus \mathcal{S}_2 := \{s_1 : s_1 + s_2 \in \mathcal{S}_1, \forall s_2 \in \mathcal{S}_2\}$.

Definition 1. [8] Consider a dynamical system $\dot{x} = f(x) + g(x)u + \delta$ where: $x \in \mathcal{X}$, $u \in \mathcal{U}$ and $\delta \in \Delta$. Consider a set $\mathcal{Q} \subseteq \mathcal{X}$. If there exists a feedback control law $u := \kappa(x) \in \mathcal{U}$, such that for all $x(0) \in \mathcal{Q}$ and for all $\delta(t) \in \Delta$ it holds that $x(t) \in \mathcal{Q}$ for all $t \geq 0$, along every solution $x(t)$, then $\mathcal{Q}$ is called a *Robust Control Invariant (RCI) set* for the system.

Definition 2. [2] A *Weighted Transition System* (WTS) is a tuple $(S, S_0, \mathrm{Act}, \longrightarrow, \mathfrak{t}, \Gamma, L)$ where S is a set of states; $S_0 \subseteq S$ is a set of initial states; Act is a set of actions; $\longrightarrow \subseteq S \times \mathrm{Act} \times S$ is a transition relation; $\mathfrak{t} : \longrightarrow \to \mathbb{Q}_+$ is a function that maps a weight to each transition; Γ is a set of atomic propositions; and $L : S \to 2^\Gamma$ stands for the labeling function.

Definition 3. [2] A *timed run* of a WTS is an infinite sequence $r^t = (r(0), \tau(0))(r(1), \tau(1)) \ldots$, such that $r(0) \in S_0$, and for all $l \geq 0$, it holds that $r(l) \in S$ and $(r(l), u(l), r(l+1)) \in \longrightarrow$ for a sequence of actions $u(0)u(1)u(2)\ldots$ with $u(l) \in \mathrm{Act}$, $\forall l \geq 0$. The *time stamps* $\tau(l)$, $l \geq 0$ are inductively defined as: 1) $\tau(0) = 0$; 2) $\tau(l+1) := \tau(l) + \mathfrak{t}(r(l), r(l+1))$, $\forall l \geq 0$.

The Metric Interval Temporal Logic (MITL) [1] over a set of atomic propositions Γ is defined by the grammar:

$$\varphi := \gamma \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \bigcirc_I \varphi \mid \Diamond_I \varphi \mid \Box_I \varphi \mid \varphi_1 \, \mathcal{U}_I \, \varphi_2,$$

where $\gamma \in \Gamma$, and $\bigcirc$, $\Diamond$, $\Box$ and $\mathcal{U}$ are the next, eventually, always and until temporal operator, respectively; $\neg$, $\wedge$ are the negation and conjunction operators, respectively; I stands for a non-empty timed interval. For the semantics of MITL see [1]. Technical details regarding timed verification can be found in [15].

III. Problem Formulation

A. System Model

Consider a robot operating in a bounded workspace $\mathcal{W} \subseteq \mathbb{R}^n$ with *uncertain dynamics:*

$$\dot{x} = f(x) + g(x)u + \delta, \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$, $g : \mathbb{R}^n \to \mathbb{R}^{n \times n}$ are known and continuously differentiable functions; $u \in \mathbb{R}^n$ stands for the control input; and $\delta \in \mathbb{R}^n$ models the external disturbances and uncertainties. Consider also input constraints as well as bounded disturbances:

$$u \in \mathcal{U} := \{u \in \mathbb{R}^n : \|u\|_2 \leq \widetilde{u}\},$$
$$\delta \in \Delta := \{\delta \in \mathbb{R}^n : \|\delta\|_2 \leq \widetilde{\delta}\},$$

where $\widetilde{u}$, $\widetilde{\delta} > 0$ are a priori known. Define the corresponding *nominal dynamics* for (1) by:

$$\dot{\hat{x}} = f(\hat{x}) + g(\hat{x})\hat{u},$$

which are the dynamics for the case of $\delta = 0$.

Assumption 1. Let the dynamics:

$$\dot{\hat{x}} = A\hat{x} + B\hat{u}.$$

be the Jacobian linearization of the nominal dynamics (1) around the equilibrium state $x = 0$. Then, we assume that the latter system is stabilizable.

Assumption 2. There exists a strictly positive constant $\underline{g}$ such that the following holds:

$$\lambda_{\min}\left[\tfrac{g(x) + g(x)^\top}{2}\right] \geq \underline{g} > 0, \quad \forall x \in \mathcal{W}.$$

Furthermore, it holds that $f(0) = 0$ and $g(0) = 0$.

In the given workspace, there exist $z \in \mathbb{N}$ Regions of Interest (RoI) labeled by $\mathcal{Z} := \{1, \ldots, z\}$. The RoI are modeled by the balls $\mathcal{R}_z := \mathcal{M}(y_z, p_z)$, $z \in \mathcal{Z}$, where y_z and $p_z > 0$ stands for the center and radius of RoI $\mathcal{R}_z$, respectively. At each time $t \geq 0$, the robot is occupying a ball $\mathcal{M}(x(t), \eta)$ that covers its volume, where $x(t)$ and $\eta > 0$ are its center and radius, respectively.

B. Objectives

The main goal of this paper is to design a feedback control law that steers the robot with dynamics as in (1) between RoI so that it obeys a high-level task given in MITL. Define the labeling function:

$$L : \bigcup_{z \in \mathcal{Z}} \mathcal{R}_z \to 2^{\Gamma}, \tag{2}$$

which maps each RoI with a subset of atomic propositions.

Definition 4. A trajectory $x(t)$ is *uniquely associated with a timed run* $r^t = (r(0), \tau(0))\ (r(1), \tau(1)) \ldots$, if:

1) $\tau(0) = 0$, i.e., the robot starts its motion at time $t = 0$;
2) $r(l) \in \bigcup_{z \in \mathcal{Z}} \mathcal{R}_z$, for every $l \in \mathbb{N}$;
3) $\mathcal{M}(x(\tau(0)), \eta) \subsetneq r(0)$, i.e., initially, the volume of the robot is entirely within the RoI $r(0)$;
4) $\mathcal{M}(x(\tau(l)), \eta) \subsetneq r(l)$, $\forall l \in \mathbb{N}$, i.e., the robot changes discrete state only when its entire volume is contained in the corresponding RoI;
5) $\tau(l+1) := \tau(l) + \mathfrak{t}(r(l), r(l+1))$, $\forall l \in \mathbb{N}$, where:
$$\mathfrak{t} : \left(\bigcup_{z \in \mathcal{Z}} \mathcal{R}_z \right) \times \left(\bigcup_{z \in \mathcal{Z}} \mathcal{R}_z \right) \to \mathbb{Q}_{+}, \tag{3}$$
is a function that models the time duration that the robot needs to be steered between regions $r(l)$ and $r(l+1)$.

Definition 5. A trajectory $x(t)$ *satisfies* a formula φ written in MITL over a set of atomic propositions Γ (written as $x(t) \models \varphi$, $\forall t \geq 0$), if and only if there exists a timed run r^t to which the trajectory is uniquely associated, according to Definition 4, which satisfies φ.

Problem 1. Consider a robot with dynamics as in (1), operating in a workspace $\mathcal{W} \subseteq \mathbb{R}^n$. In the workspace, there exist $z \in \mathbb{N}$ RoI modeled by the balls $\mathcal{M}(y_z, p_z)$, $\forall z \in \mathcal{Z}$. Then, given an MITL formula φ over a set of atomic propositions Γ and a labeling function as in (2), design a feedback control law $u = \kappa(x) \in \mathcal{U}$ which guarantees that:

$$x(t) \models \varphi, \forall t \geq 0,$$

according to Definition 5, while the robot remains in the workspace for all times.

IV. Problem Solution

A. Feedback Control Design

Consider a robot with dynamics (1) occupying a RoI $\mathcal{R}_i$, $i \in \mathcal{Z}$ at time $\mathfrak{t}_i \geq 0$. Denote by $x_j \in \mathcal{R}_j$, $j \in \mathcal{Z} \backslash \{i\}$ the center of a desired RoI towards which the robot is required to be navigated. Define the error vector $e := x - x_i \in \mathbb{R}^n$. Then, the *uncertain error dynamics* are given by:

$$\dot{e} = f(e + x_i) + g(e + x_i)u + \delta. \tag{4}$$

The corresponding *nominal error dynamics* are given by:

$$\dot{\hat{e}} = f(\hat{e} + x_i) + g(\hat{e} + x_i)\hat{u} \tag{5}$$

Define the set that captures the state constraints by:

$$\mathcal{X} := \{x \in \mathbb{R}^n : \mathcal{M}(x(t), \eta) \cap \mathcal{R}_{j'} = \emptyset, \ \ \forall j' \in \mathcal{Z} \backslash \{i, j\}, \\ \mathcal{M}(x(t), \eta) \subsetneq \mathcal{W}\}.$$

The first constraint denotes the fact that the robot should not intersect with any other RoI other than $\mathcal{R}_i$,$\mathcal{R}_j$; the second one, denotes the fact that the robot needs to remain in the workspace for all times.

Define by $q = e - \hat{e}$ the deviation between the real state of the system (4) and the nominal state of the system (5) with $q(0) = e(0) - \hat{e}(0) = 0$. The dynamics of the state q are given by:

$$\begin{aligned} \dot{q} &= f(e + x_i) - f(\hat{e} + x_i) \\ &\qquad + g(e + x_i)u - g(\hat{e} + x_i)\hat{u} + \delta \\ &= h(e, \hat{e}, u) + g(\hat{e} + x_i)(u - \hat{u}) + \delta. \end{aligned} \tag{6}$$

where the function h is defined by $h(e, \hat{e}, u) := f(e + x_i) - f(\hat{e} + x_i) + g(e + x_i)u - g(\hat{e} + x_i)u$. Note that the following holds:

$$\|h(e, \hat{e}, u)\|_2 \leq L \|q\|_2,$$

where $L := \max\{L_f, L_g\}$ and L_f, $L_g > 0$ are the Lipschitz constant of the functions f, g, respectively.

Lemma 1. *The feedback control law:*

$$u := \hat{u}(\hat{e}) - \sigma q, \tag{7}$$

where $\hat{u}$ is a nominal input to be computed afterwards through an on-line optimal control problem and the control gain is designed such that $\sigma = \frac{L}{\underline{g}} + \underline{\sigma}$, $\underline{\sigma} > 0$, renders the set:

$$\mathcal{Q} := \left\{ q \in \mathbb{R}^n : \|q\|_2 \leq \frac{\widetilde{\delta}}{\underline{\sigma}} \right\},$$

an RPI set for the system (6)*, according to Definition 1.*

Proof. The proof of this lemma follows similar arguments to [9], [10]. The time derivative of the function $\Lambda(q) = \frac{1}{2}\|q\|^2$ along the trajectories of the system (6) is:

$$\begin{aligned} \dot{\Lambda}(q) &= q^{\top} \dot{q} \\ &= q^{\top} h(e, \hat{e}, u) + q^{\top} g(e + x_i)(u - \hat{u}) + q^{\top} \delta \\ &\leq L \|q\|_2^2 - \sigma q^{\top} g(e + x_i) q + \widetilde{\delta} \|q\|_2 \\ &\leq \left[- \left(\sigma \underline{g} - L \right) \|q\|_2 + \widetilde{\delta} \right] \|q\|_2 \\ &\leq \left(-\underline{\sigma} \|q\|_2 + \widetilde{\delta} \right) \|q\|_2. \end{aligned}$$

Thus, $\dot{\Lambda} < 0$ when $\|q\|_2 > \frac{\widetilde{\delta}}{\underline{\sigma}}$. Since the fact that $q(0) = 0$, it holds that $\|q(t)\| \leq \frac{\widetilde{\delta}}{\underline{\sigma}}$, for every $t \geq 0$. □

Remark 1. The volume of the tube that is centered along the nominal trajectories $\hat{e}(t)$, $t \geq 0$ depends on the upper bound of the disturbance $\widetilde{\delta}$, and the parameters L, $\underline{g}$.

Hereafter we introduce the methodology under which the online control law $\hat{u}(\hat{e})$ is calculated. Denote by $h > 0$ and $N > h$ the sampling step and the finite prediction horizon.

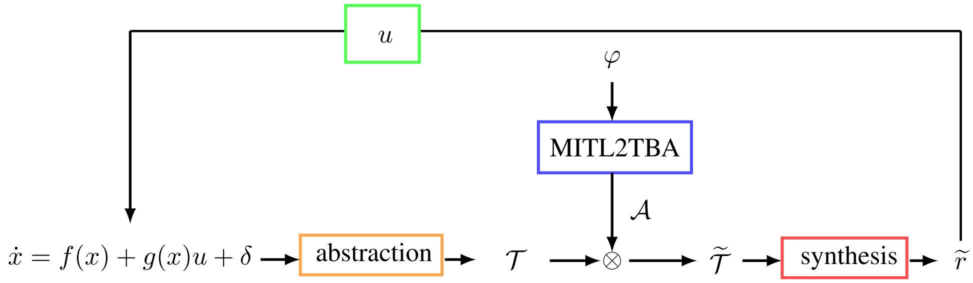

Fig. 2: A graphical representation of control design framework.

Consider a sequence of sampling times t_k, $k \in \mathbb{N}$. Then, at every sampling time t_k, $k \in \mathbb{N}$ the following FHOCP is solved:

$$\min_{\hat{u}(\cdot)} \left\{ \|\hat{e}(t_k + N)\|_P^2 + \int_{t_k}^{t_k+N} \Big[\|\hat{e}(s)\|_Q^2 + \|\hat{u}(s)\|_R^2 \Big] ds \right\} \quad \text{(8a)}$$

subject to:

$$\dot{\hat{e}}(s) = f(\hat{e}(s) + x_i) + g(\hat{e}(s) + x_i)\hat{u}(s), \quad \hat{e}(t_k) = e(t_k), \quad \text{(8b)}$$

$$\hat{e}(s) \in \mathcal{E}, \quad \hat{u}(s) \in \mathbb{U}, \quad s \in [t_k, t_k + T], \quad \text{(8c)}$$

$$\hat{e}(t_k + N) \in \mathcal{F}, \quad \text{(8d)}$$

where Q, P and R are positive definite gain matrices to be appropriately tuned. The state and input constraints sets are modified as:

$$\mathcal{E} := [\mathcal{X} \oplus (-x_i)] \ominus \mathcal{Q}, \quad \mathbb{U} := \mathcal{U} \ominus [-\sigma \circ \mathcal{Q}],$$

in order to guarantee that while the FHOCP is solved for the nominal system dynamics (5), the real trajectories satisfy the state and input constraints $\mathcal{X} \oplus (-x_i)$, $\mathcal{U}$, respectively. The set $\mathcal{F} := \{\hat{e} \in \mathcal{E} : \|\hat{e}\|_P \le \varepsilon\}$, $\varepsilon > 0$ is used in order to ensure the nominal stability of the system [16].

Theorem 1. *([9]) Assume that Assumptions 1 and 2 hold. Consider a time $\mathfrak{t}_i \ge 0$ that the robot occupies a RoI $\mathcal{R}_i$, $i \in \mathcal{Z}$ with center x_i. Let x_j, $j \neq i$ be the center of a desired RoI $\mathcal{R}_j$ that the robot is required to be navigated towards. Suppose also that the FHOCP* (8a)-(8d) *is feasible at time $\mathfrak{t}_i$. Then, there exist a time $\mathfrak{t}_{ij} > \mathfrak{t}_i$ such that:*

$$\|x_i(t) - x_j\| \le \frac{\varepsilon}{\sqrt{\lambda_{\max}(P)}} + \frac{\widetilde{\delta}}{\underline{\sigma}}, \quad \forall t \ge \mathfrak{t}_{ij}.$$

According to the previous theorem, the robot with dynamics as in (1), starting its motion at time $\mathfrak{t}_i$ at RoI $\mathcal{R}_i$, under the control law (7), will have been navigated to RoI $\mathcal{R}_j$ at time $\mathfrak{t}_{ij}$. For the computation of the time $\mathfrak{t}_{ij}$, Algorithm 1 is used. In particular, the time that the robot is reaching the terminal set is captured by $\mathfrak{t}_{ij}$. This time models the time the the robot will have been navigated from RoI $\mathcal{R}_i$ to $\mathcal{R}_j$, i.e, is consistent with (3).

B. Discrete Abstractions and Control Synthesis

Definition 6. The motion of the robot in the workspace is modeled through a WTS $\mathcal{T} = (S, S_0, \text{Act}, \longrightarrow, \mathfrak{t}, \Gamma, L)$ where:

Algorithm 1 Numerical computation of $\mathfrak{t}_{\text{ij}} := \mathfrak{t}(\mathcal{R}_{\text{i}}, \mathcal{R}_{\text{j}})$

```
1:  Input: $\mathfrak{t}_i$, $\hat{x}(t_k)$, $x_j$, $k = \{0, 1, 2, \dots\}$;
2:  Output: $\mathfrak{t}_{ij}$;
3:  $t_k \leftarrow \mathfrak{t}_i$;
4:  cond ← True
5:  while cond = True do
6:      measure $\hat{x}(t_k)$;
7:      if $\|\hat{x}(t_k) - x_j\|_2 \le \frac{\varepsilon}{\sqrt{\lambda_{\min}(P)}}$ then
8:          cond ← False;
9:          break;
10:         Go to "line 13"
11:     end if
12:     $t_k \leftarrow t_k + h$;
13: end while
14: $\mathfrak{t}_{ij} \leftarrow t_k$;
```

- $S = \bigcup_{z \in \mathcal{Z}} \mathcal{R}_z$ is the set of states;
- S_0 is the initial RoI that the robot starts its motion;
- Act is set of actions containing the union of all feedback control laws of the form (7) that are able to drive the robot between RoI;
- $\longrightarrow \subseteq S \times Act \times S$ is the transition relation. We say that $(\mathcal{R}_i, u', \mathcal{R}_j) \in \longrightarrow$, $i, j \in \mathcal{Z}$, $i \neq j$, if there exist a feedback control $u' \in$ Act that can drive the robot from RoI $\mathcal{R}_i$ to the RoI $\mathcal{R}_j$;
- and L, $\mathfrak{t}$ as given in (2) and (3), respectively.

By using the framework depicted in Fig. 2, a sequence of feedback laws as in (7) that guarantee the satisfaction of the given MITL formula φ is designed. In particular, the dynamics of the robot (1) are abstracted into a WTS $\mathcal{T}$ according to Definition 6; A product WTS $\widetilde{T}$ is constructed by computing the product between $\mathcal{T}$ and the Timed Büchi automaton $\mathcal{A}$, whose accepting runs are the ones that satisfy the formula φ. Then, by performing graph search in $\widetilde{T}$ a timed run $\widetilde{r}$ that maps into feedback control laws as in (7) can be computed. For the definitions of the product WTS and the Timed Büchi Automaton see [2].

V. Experimental Setup and Results

This section demonstrates the efficacy of the proposed framework via a real-time experiment employing a Nexus 10011 mobile robot (Fig. 1). The experiment was carried out at Smart Mobility Lab (SML) (see Fig. 3 and [17]). The robot dimensions are $400 \times 360 \times 100$ mm and consists of 4 aluminum mecanuum wheels which provide omni-directional capabilities through the 3 degrees of freedom (moving forward/backward, left/right and rotation). The rollers have a rotation of 45 degrees with reference to the the plane of the wheel. Each wheel is connected to $12V$ motors which they have optical encoders. The speed can be controlled by local PID controllers on an Arduino 328 Controller and Arduino IO expansion board.

SML provides a motion capture system (MoCap) with 12 Qualisys cameras spread across the lab. The MoCap provides the robot state vector, including pose, orientation as well as linear and angular velocities at frequency of 100Hz. The hardware used in experiments are connected using Gigabit Ethernet connections, USB connections, and wireless 5.8Ghz connections. The Raspberry Pi on the robot is connected with a TP-Link Router via wireless 5.8 Ghz connections. The host computer as well as the MoCap are connected with the TP-Link Router via Gigabit Ethernet connections. The host computer runs the node of the controller, which receives the measurement from MoCap, and calculates the control signal.

The software implementation of the proposed control strategy was conducted in C++ under the Robot Operating System (ROS) [18]. Moreover, the Nonlinear Model Predictive Controller employed in this work is designed using the NLopt Optimization library [19] and runs on a desktop with 8 cores, 3.40GHz Intel Core TM $i7-6700$ CPU and 32GB of RAM.

The workspace that the robot can operate in as well as a panoramic view of it are depicted in Fig. 3 and Fig. 4, respectively. It is captured by the set $\mathcal{W} = \{x \in \mathbb{R}^2 : \|x\|_\infty \leq 2.5\}$ and contains 9 RoI which are divided as follows:

- the RoI $\mathcal{R}_i$, $i \in \{1,2,3,4,5\}$ depicted with blue in Fig. 4; The RoI $\mathcal{R}_1$ and $\mathcal{R}_3$ map into the atomic propositions mission_1 and mission_2, respectively.
- the RoI $\mathcal{R}_i$, $i \in \{6,7,8,9\}$ depicted with red in Fig. 4 which stand for unsafe regions that the robot needs to avoid. They map into the atomic propositions obs_1, obs_2 and obs_3, respectively. Moreover, it holds that $L(\mathcal{R}_i) = \emptyset$, $\forall i \in \{2,4,5\}$.

The control input constraints are set to:

$$\mathcal{U} = \{u_i \in \mathbb{R}^3 : |u_i| \leq 0.2, \ i \in \{1,2,3\}\},$$

where u_1, u_2 stand for the linear velocities and u_3 stands for the angular velocity. The desired MITL formula over the set of atomic propositions:

$$\Gamma = \{\text{mission}_1, \text{mission}_2, \text{obs}_1, \text{obs}_2, \text{obs}_3\},$$

that the robot needs to satisfy is set to:

$$\varphi = \Box_{[0,\infty)}\{\neg\text{obs}_1 \wedge \neg\text{obs}_2 \wedge \neg\text{obs}_3 \wedge \neg\text{obs}_4\} \wedge \Diamond_{[30,50]}\{\text{mission}_2\} \wedge \Diamond_{[80,110]}\{\text{mission}_1\}, \quad (9)$$

The prediction horizon is chosen as $N = 1.2\,\text{sec}$. The gains are set to $Q = P = R = 0.5 I_3$.

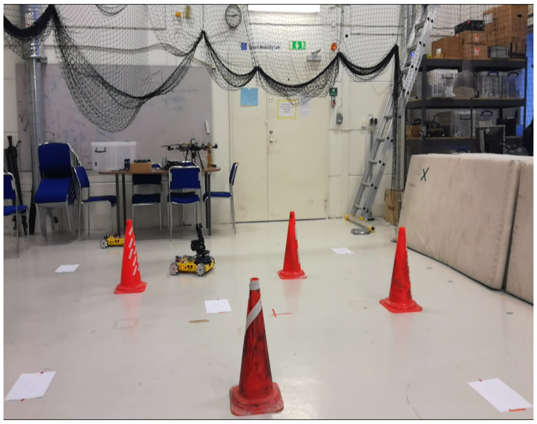

Fig. 3: The nexus robot performing the desired high-level task given in (9).

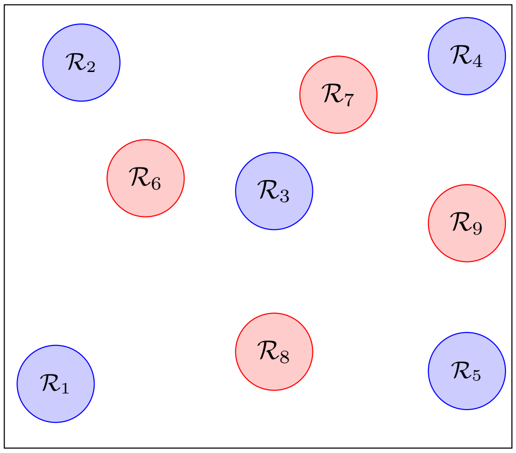

Fig. 4: A panoramic view of the workspace with the 9 RoI.

Video: A video demonstrating the experiment of this section can be found in the following link:

`https://youtu.be/FcB8Pp51Qpw`

By employing Algorithm 1 the transition times between the RoI that the robot is following are given as follows:

$$\mathfrak{t}(\mathcal{R}_1,\mathcal{R}_2) = 17.3, \quad \mathfrak{t}(\mathcal{R}_2,\mathcal{R}_3) = 20.1, \quad \mathfrak{t}(\mathcal{R}_3,\mathcal{R}_4) = 18.2,$$
$$\mathfrak{t}(\mathcal{R}_4,\mathcal{R}_5) = 18.5, \quad \mathfrak{t}(\mathcal{R}_5,\mathcal{R}_1) = 15.7.$$

The evolution of the states x, y as well as the angle of the robot are presented in Fig. 5, Fig. 6 and Fig. 7, respectively. The control input signals u_1, u_2 and u_3 are depicted in Fig. 8, Fig. 9 and Fig. 10, respectively. The transition times given above fulfill the constraints given by the MITL formula. Thus, it can be observed that the robot fulfills the MITL specification while all the imposed constraints from Problem 1 are satisfied.

VI. Conclusions

In this paper, we have experimentally validated recent theoretical results of robust nonlinear tube-based MPC along with timed-constrained high-level planning. In particular, given the uncertain dynamics of a robot and a timed specification written in MITL, we have provided a framework under which a sequence of control laws under which the robot

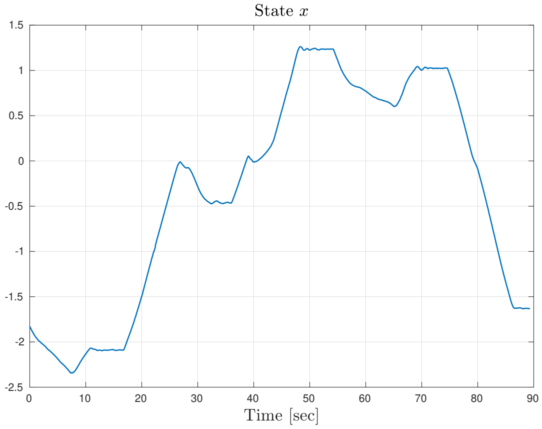

Fig. 5: The state x of the robot.

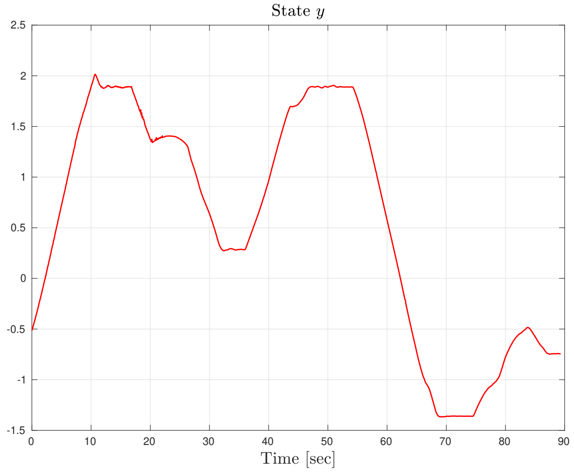

Fig. 6: The evolution of the state y of the robot over time.

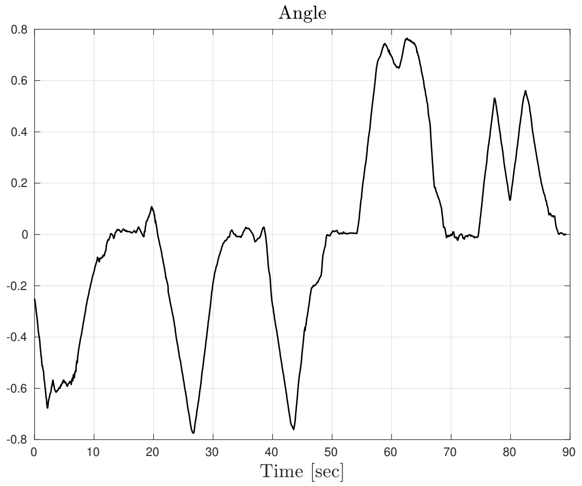

Fig. 7: The evolution of the angle of the robot over time.

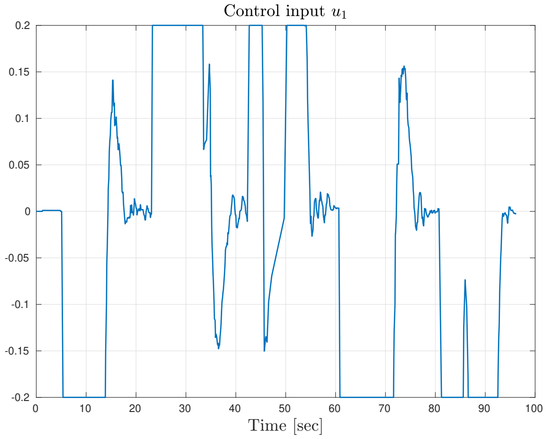

Fig. 8: The control input signal u_1 of the robot.

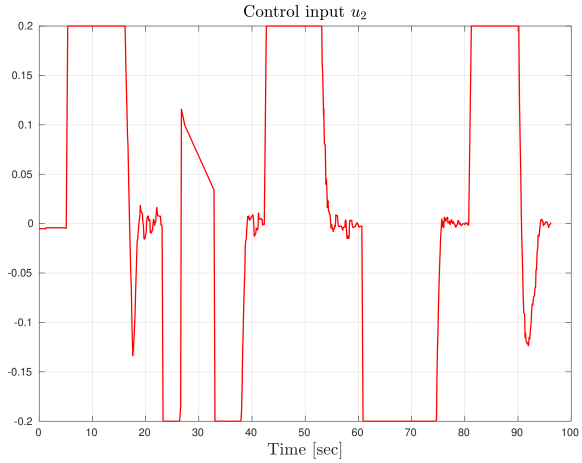

Fig. 9: The evolution of the control input signal u_2 of the robot.

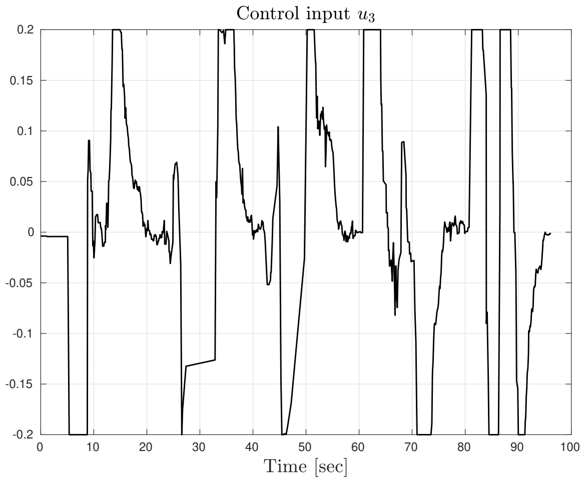

Fig. 10: The evolution of the control input signal u_3 of the robot.

satisfies the desired task. The experimental platform consists of a Nexus 10011 mobile robot with an attached manipulator. The preliminary experimental results of the paper in hand verifies the efficiency of the proposed framework that solves a general category of time-constrained robot navigation problems under state/input constraints, obstacle avoidance as well as uncertainties/disturbances.

References

[1] R. Alur, T. Feder, and T. Henzinger, "The Benefits of Relaxing Punctuality," *Journal of the Association for Computing Machinery (JACM)*, vol. 43, no. 1, pp. 116–146, 1994.

[2] A. Nikou, J. Tumova, and D. V. Dimarogonas, “Cooperative Task Planning of Multi-Agent Systems Under Timed Temporal Specifications,” *American Control Conference (ACC)*, pp. 13–19, 2016.

[3] A. Nikou, “Cooperative Planning Control and Formation Control of Multi-Agent Systems Multi-Agent Systems,” *Licentiate Thesis, KTH Royal Institute of Technology, Link: http://www.diva-portal.org/smash/get/diva2:1094985/FULLTEXT01.pdf*, 2017.

[4] A. Nikou, D. Boskos, J. Tumova, and D. V. Dimarogonas, “On the Timed Temporal Logic Planning of Coupled Multi-Agent Systems,” *Automatica*, vol. 97, pp. 339–345, 2018.

[5] D. Koditschek and E. Rimon, “Robot Navigation Functions on Manifolds with Boundary,” *Advances in Applied Mathematics*, vol. 11, no. 4, pp. 412 – 442, 1990.

[6] C. Bechlioulis and G. Rovithakis, “A Low-Complexity Global Approximation-Free Control Scheme with Prescribed Performance for Unknown Pure Feedback Systems,” *Automatica*, vol. 50, no. 4, pp. 1217 – 1226, 2014.

[7] W. Langson, I. Chryssochoos, S. Rakovic, and D. Mayne, “Robust Model Predictive Control Using Tubes,” *Automatica*, vol. 40, no. 1, pp. 125–133, 2004.

[8] S. Yu, C. Maier, H. Chen, and F. Allgöwer, “Tube MPC Scheme Based on Robust Control Invariant Set with Application to Lipschitz Nonlinear Systems,” *Systems and Control Letters*, vol. 62, no. 2, pp. 194–200, 2013.

[9] A. Nikou and D. V. Dimarogonas, “Robust Tube-based Model Predictive Control for Timed-constrained Robot Navigation,” *American Control Conference (ACC), https://arxiv.org/pdf/1809.09825.pdf*, Philadelphia, US, 2019.

[10] A. Nikou and D. V. Dimarogonas, “Decentralized Tube-based Model Predictive Control of Uncertain Nonlinear Multi-Agent Systems,” *International Journal of Robust and Nonlinear Control (IJRNC)*, pp. 1–20, https://doi.org/10.1002/rnc.4522, 2019.

[11] S. Heshmati-alamdari, G. Karras, P. Marantos, and K. Kyriakopoulos, “A Robust Model Predictive Control Approach for Autonomous Underwater Vehicles Operating in a Constrained Workspace,” *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–5, 2018.

[12] S. Heshmati-alamdari, G. Karras, and K. Kyriakopoulos, “A Distributed Predictive Control Approach for Cooperative Manipulation of Multiple Underwater Vehicle Manipulator Systems,” *2019 IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[13] S. Heshmati-alamdari, G. Karras, A. Eqtami, and K. Kyriakopoulos, “A Robust Self Triggered Image Based Visual Servoing Model Predictive Control Scheme for Small Autonomous Robots,” *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5492–5497, 2015.

[14] S. Heshmati-alamdari, A. Nikou, K. J. Kyriakopoulos, and D. V. Dimarogonas, “A Robust Force Control Approach for Underwater Vehicle Manipulator Systems,” *20th World Congress of the International Federation of Automatic Control (IFAC WC), Toulouse, France*, vol. 50, Issue 1, pp. 11197–11202, 2017.

[15] P. Bouyer, “From Qualitative to Quantitative Analysis of Timed Systems,” *Mémoire dhabilitation, Université Paris*, vol. 7, 2009.

[16] H. Chen and F. Allgöwer, “A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability,” *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.

[17] “Smart mobility lab (SML),” *https://www.kth.se/dcs/research/control-of-transport/smart-mobility-lab/smart-mobility-lab-1.441539*.

[18] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “ROS: An Open-Source Robot Operating System,” *International Conference on Robotics and Automation (ICRA) Workshop*, 2009.

[19] S. G. Johnson, “The NLopt Nonlinear-Optimization Package,” *http://ab-initio.mit.edu/nlopt*, 2009.