Real-Time Tube MPC Applied to a 10-State Quadrotor Model

Haimin Hu[1], Xuhui Feng[1], Rien Quirynen[2], Mario Eduardo Villanueva[1,*], and Boris Houska[1]

***Abstract*— This paper discusses a real-time implementation of tube model predictive controllers for nonlinear input-affine systems. This is achieved by combining recent theoretical and practical advances on the construction of forward invariant tubes with state-of-the-art algorithms for nonlinear MPC, such as the real-time iteration scheme. The focus of the paper is on presenting these ideas in a tutorial style, using a 10-state quadcopter model as an example. The controller is implemented using the automatic code generation capabilities of `ACADO Toolkit`. Numerical experiments show that the tube MPC scheme can achieve run-times in the lower millisecond range.**

I. Introduction

Certainty-equivalent model predictive control (MPC) is a modern optimization-based control strategy used in many industrial processes [1], [2], [3]. Being a feedback control scheme, certainty-equivalent MPC exhibits a certain degree of robustness due to its inherent ability to reject disturbances, which is often sufficient for less safety critical applications [4]. However, since MPC predicts the state trajectory without accounting for uncertainty, constraints may become violated when large disturbances occur.

Unlike certainty-equivalent MPC, robust MPC is capable of providing a feasibility guarantee for the controlled system, at least within a prescribed uncertainty range [5]. However, a rigorous formulation of robust MPC, calls for the solution of feedback optimal control problems, which are hard to solve in general. In recent decades, certain formulations leading to tractable approximations of the robust MPC problem have appeared (see [6], [7], [8]). Among these, tube MPC [9], [10] has emerged as a promising methodology to tackle the synthesis of robust MPC controllers. This methodology adopts a set-oriented perspective, inspired by set and viability theory [11], [12]. The main idea of tube MPC is to replace predicted state trajectories by robust forward invariant tubes (RFITs) [13], which are set-valued functions including every possible state trajectory under a given feedback law—independently of the uncertainty realization.

One of the biggest concerns when implementing MPC controllers, is the computational time needed to obtain the feedback law. For certainty-equivalent MPC, this issue has been addressed extensively (e.g. [14], [15], [16], [17], [18]). In particular, combining nonlinear MPC techniques with optimized auto-generated code, has been used to synthesize MPC controllers with CPU times in the milli- and microsecond range [19], [20]. Despite the number of real-time algorithms for certainty-equivalent MPC, the list of similar algorithms for robust MPC is limited to linear systems [20].

This paper presents a first attempt towards the real-time implementation of tube MPC for input-affine nonlinear systems. Tube MPC formulations typically involve a parameterization of the feedback law which is then used to construct an RFIT [9]. Here, we use a recently proposed tube MPC formulation based on min-max differential inequalities, which parameterizes the tube instead of the feedback law. The implementation of this tube MPC controller is based on an ellipsoidal parameterization of the tube's cross-sections. This leads to a standard optimal control problem formulation, albeit with a larger number of states and controls. This optimal control problem is then solved in a receding horizon manner using an auto-generated real-time iteration (RTI) scheme [19], which for a problem of a moderate size with mild nonlinearities is able to deliver a computational performance in the millisecond range. Another feature of the ellipsoidal tube MPC formulation is the availability of a nonlinear feedback control law associated to the tube. This feedback law is used to control the actual process at even higher sampling rates. Thus, in contrast to certainty-equivalent MPC, robust MPC is a hierarchical control scheme, which updates the feedback law online.

The remainder of the paper is organized as follows, Section II introduces the mathematical formulation of tube MPC. Section II-B presents the quadcopter model used throughout, to illustrate the theory. Section III introduces a tractable reformulation of ellipsoidal tube MPC as well as its real-time implementation. Section IV presents the performance results for the controller applied to a 10-state quadcopter model. Finally, Section V concludes the paper.

Notation: Besides standard mathematical notation, we denote by L_2^n the set of n-dimensional Lebesgue integrable functions, and $W_{1,2}^n$ denotes the Sobolev space of weakly differentiable functions with square-integrable derivatives. The set of compact and convex compact sets in $\mathbb{R}^n$ are denoted respectively by $\mathbb{K}^n$ and $\mathbb{K}_{\mathrm{C}}^n$. The set of $n \times n$ symmetric positive semidefinite and definite matrices are denoted by $\mathbb{S}_+^n$ and $\mathbb{S}_{++}^n$, respectively. An ellipsoid with center $q \in \mathbb{R}^n$ and shape matrix $Q \in \mathbb{S}_+^n$ is given by

$$\mathcal{E}(q,Q) = \left\{ q + Q^{\frac{1}{2}} v \;\middle|\; v^{\mathsf{T}} v \leq 1 \right\},$$

where $Q^{\frac{1}{2}}$ is the positive semidefinite square root of Q. The ith row of a matrix A is denoted by A_i. For matrices and vectors, the symbols $\geq, \leq$, are understood componentwise.

*Corresponding Author.

[1]School of Information Science and Technology (SIST), ShanghaiTech University, 393 Middle Huaxia Road, Pudong, Shanghai, 201210, China. [huhm,fengxh,meduardov,borish]@shanghaitech.edu.cn

[2]Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA, United States. quirynen@merl.com

II. Problem Formulation

A. Tube MPC: Mathematical Formulation

We consider uncertain control systems of the form

$$\forall t \in \mathbb{R}: \quad \dot{x}(t) \;=\; f(x(t), w(t)) + Gu(t). \tag{1}$$

Here, $x : \mathbb{R} \to \mathbb{R}^{n_x}$ denotes the state trajectory, which is subject to polytopic hard constraints,

$$\forall t \in \mathbb{R}: \; x(t) \in \mathbb{X} := \{\, \xi \in \mathbb{R}^{n_x} \,|\, C^\mathsf{T} \xi \leq c \,\} \subseteq \mathbb{R}^{n_x}, \tag{2}$$

with $C \in \mathbb{R}^{n_x \times n_c}$ and $c \in \mathbb{R}^{n_c}$. The control $u : \mathbb{R} \to \mathbb{R}^{n_u}$ and exogenous inputs $w : \mathbb{R} \to \mathbb{R}^{n_w}$ are assumed to be bounded,

$$\forall t \in \mathbb{R}: \quad u(t) \in \mathbb{U} \in \mathbb{K}_\mathrm{C}^{n_u} \quad \text{and} \quad w(t) \in \mathbb{W} \in \mathbb{K}_\mathrm{C}^{n_w}.$$

The reachable set of the closed-loop system at time t for a given feedback law $\mu : \mathbb{R} \times \mathbb{R}^{n_x} \to \mathbb{U}$ and a given initial state $x(0) = x_0 \in \mathbb{R}^{n_x}$ is given by

$$X(t, x_0, \mu) = \left\{ \xi_t \;\middle|\; \begin{array}{l} \exists x \in W_{1,2}^{n_x}, \exists w \in L_2^{n_w} : \forall \tau \in [0,t], \\ \dot{x}(\tau) = f(x(\tau), w(\tau)) + G\mu(\tau, x(\tau)), \\ x(0) = x_0, \; x(t) = \xi_t, \; w(\tau) \in \mathbb{W} \end{array} \right\}.$$

Tube MPC calls for the receding horizon solution of,

$$\begin{aligned} \inf_{\mu : \mathbb{R} \times \mathbb{R}^{n_x} \to \mathbb{U}} \quad & \int_0^T \mathcal{L}(X(t, \hat{x}_0, \mu)) \, \mathrm{d}t \\ \text{s.t.} \quad & X(t, \hat{x}_0, \mu) \subseteq \mathbb{X}, \; \forall t \in [0,T], \end{aligned} \tag{3}$$

on the current time horizon $[0,T]$. Here, $\mathcal{L} : K_\mathrm{C}^{n_x} \to \mathbb{R}$ denotes a stage cost and $\hat{x}_0$ the current state measurement. The optimal input $u(0) = \mu(0, \hat{x}_0)$, is then sent to the process until a new measurement becomes available.

Problem (3) is intractable, in all but the simplest of cases. Fortunately, if we relax the problem and choose a suitable set-parameterization, we can find tractable approximations of tube MPC. Here, we use the concept of robust forward invariant tubes (RFITs). We say that a set-valued function $\overline{X} : [0,T] \to \mathbb{K}^{n_x}$ is an RFIT for (1) on $[0,T]$, if there exists a feedback $\mu : [0,T] \times \mathbb{R}^{n_x} \to \mathbb{U}$, such that

$$\overline{X}(t_2) \supseteq \bigcup_{x_1 \in \overline{X}(t_1)} X(t_2 - t_1, x_1, \mu)$$

for all $t_1, t_2 \in [0,T]$ with $t_2 \geq t_1$. Thus, (3) is equivalent to

$$\inf_{\overline{X} \in \mathcal{X}} \int_0^T \mathcal{L}(\overline{X}(t)) \, \mathrm{d}t \quad \text{s.t.} \begin{cases} \overline{X}(t) \subseteq \mathbb{X}, \; \forall t \in [0,T] \\ \overline{X}(0) = \{\hat{x}_0\} \end{cases}, \tag{4}$$

where $\mathcal{X}$ denotes the set of all RFITs of (1) on $[0,T]$. RFITs can be selected according to different objectives $\mathcal{L}$. An example is the generalized rotational inertia of $\overline{X}(t)$ with respect to a reference x_ref with weighting matrix $P \in \mathbb{S}_+^{n_x}$,

$$\mathcal{L}(\overline{X}(t)) = \frac{\displaystyle\int_{\overline{X}(t)} (\xi - x_\mathrm{ref})^\mathsf{T} P (\xi - x_\mathrm{ref}) \mathrm{d}\xi}{\displaystyle\int_{\overline{X}(t)} \mathrm{d}\xi}.$$

At this point we have only traded the difficult task of optimizing over feedback laws by an optimization problem over the space of RFITs. Fortunately, as demonstrated in [10], restricting the search to RFITs with convex images, can—at least in some cases—lead to tractable formulations.

B. Tutorial: Stabilization of a 10D quadrotor model

We consider a simplified 10D scaled quadrotor model [21]

$$\dot{x}(t) = \underbrace{\begin{pmatrix} x_4(t) + w_1(t) \\ x_5(t) + w_2(t) \\ x_6(t) + w_3(t) \\ g\tan(x_7(t)) \\ g\tan(x_8(t)) \\ -g \\ -d_1 x_7(t) + x_9(t) \\ -d_1 x_8(t) + x_{10}(t) \\ -d_0 x_3(t) \\ -d_0 x_7(t) \end{pmatrix}}_{f(x(t), w(t))} + \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & k_T \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ n_0 & 0 & 0 \\ 0 & n_0 & 0 \end{pmatrix}}_{G} u(t). \tag{5}$$

Here, (x_1, x_2, x_3) can be interpreted as the position of the quadrotor; (x_4, x_5, x_6) its velocities; x_7 and x_8 denote pitch and roll; x_9 and x_{10} the pitch and roll rates. The disturbances are denoted by w_1, w_2, and w_3, which represent wind in the three axes. The control inputs u_1, u_2, and u_3 represent adjustable pitch angle, roll angle, and vertical thrust. In addition to the above variables, g denotes the gravitational acceleration, k_T a thrust coefficient, n_0 the roll/pitch control gain, and d_0, d_1 are attitude coefficients.

The state constraint set, a wall in the x_1 dimension, is $\mathbb{X} := \{x_1 | x_1 \leq \bar{x}_1\}$ with given $\bar{x}_1 \in \mathbb{R}$. The control constraints are given by $\mathbb{U} := \{v \in \mathbb{R}^{n_u} \mid \underline{u} \leq v \leq \overline{u}\}$ with given $\underline{u}, \overline{u} \in \mathbb{R}^{n_u}$. The disturbance signal is bounded by $\mathbb{W} := \mathcal{E}(q_w, Q_w)$, with $q_w = (0,0,0)^\mathsf{T}$ and $Q_w = \mathrm{diag}(0.01, 0.09, 0.25)$. The control objective is to track a hovering position x_ref.

TABLE I
Parameter values for the example.

Parameter	Symbol	Value
coefficient	d_0, d_1	$10, 8$
pitch/roll gain	n_0	10
position of the wall	$\bar{x}_1$	4
thrust coefficient	k_T	0.91
gravitational constant	g	9.8
hovering state	x_ref	$(3,3,10,0_{1\times 7})^\mathsf{T}$
initial state	x_0	$(0,2,0_{1\times 8})^\mathsf{T}$
weighting matrix	P	$\mathrm{diag}(1,1,1,0_{1\times 7})$
lower bound of control	$\underline{u}$	$(-\frac{\pi}{9}, -\frac{\pi}{9}, 0)^\mathsf{T}$
upper bound of control	$\overline{u}$	$(\frac{\pi}{9}, \frac{\pi}{9}, 2g)^\mathsf{T}$

III. Real-time Tube MPC

A. Ellipsoidal Tube MPC

A practical implementation of tube MPC is based on the parameterization of RFITs. The focus here is on ellipsoidal RFITs $\overline{X} : t \mapsto \mathcal{E}(q_x(t), Q_x(t))$, parameterized by a central path $q_x : \mathbb{R} \to \mathbb{R}^{n_x}$ and a time-varying shape matrix $Q_x : \mathbb{R} \to \mathbb{S}_+^{n_x}$. Sufficient conditions for functions q_x and Q_x to parameterize an ellipsoidal RFIT are given in Theorem 1.

Theorem 1: Consider (1) and assume the uncertainty and control bounds are given respectively by $\mathbb{W} := \mathcal{E}(q_w, Q_w)$ and $\mathbb{U} := \{v \in \mathbb{R}^{n_u} \mid \underline{u} \leq v \leq \overline{u}\}$, with given $(q_w, Q_w) \in \mathbb{R}^{n_w} \times \mathbb{S}_+^{n_w}$. If the functions q_x and Q_x satisfy

$$\begin{aligned} \dot{q}_x(t) &= f(q_x(t), q_w) + G u_x(t) && \forall t \in [0,T] \\ \dot{Q}_x(t) &= \Phi(q_x(t), Q_x(t), \lambda(t), \kappa(t), K(t)), && \forall t \in [0,T] \,, \end{aligned} \tag{6}$$

for some Lebesgue integrable functions $\lambda, \kappa : [0,T] \to \mathbb{R}_{++}$, $u_x : [0,T] \to \mathbb{U}$ and $K : [0,T] \to \mathbb{R}^{n_u \times n_x}$ satisfying

$$\begin{aligned} u_{x,i}(t) + \sqrt{K_i(t) Q_x(t) K_i(t)^\mathsf{T}} &\leq \overline{u}_i \,, \quad i \in \{1, \ldots, n_u\} \,, \\ u_{x,i}(t) - \sqrt{K_i(t) Q_x(t) K_i(t)^\mathsf{T}} &\geq \underline{u}_i \,, \quad i \in \{1, \ldots, n_u\} \,, \end{aligned} \tag{7}$$

then $\overline{X} : t \mapsto \mathcal{E}(q_x(t), Q_x(t))$ is an RFIT for (1) on $[0,T]$.

Proof: See [10, Theorem 5] for a proof. ∎

In Theorem 1, we have introduced the shorthands

$$\begin{aligned} A(t) = \frac{\partial f}{\partial x}(q_x(t), q_w) \,, \quad B(t) = \frac{\partial f}{\partial w}(q_x(t), q_w) \,, \quad \text{and} \\ \Phi(q_x(t), Q_x(t), \lambda(t), \kappa(t), K(t)) = A(t) Q_x(t) + Q_x(t) A(t)^\mathsf{T} \\ + \lambda(t)^{-1} Q_x(t) + \lambda(t) B(t) Q_w B(t)^\mathsf{T} \\ + \kappa(t)^{-1} Q_x(t) + \kappa(t) \Omega_n(q_x(t), Q_x(t)) \\ - G K(t) Q_x(t) - Q_x(t) K(t)^\mathsf{T} G^\mathsf{T} . \end{aligned}$$

The nonlinearity bounder $\Omega_n : \mathbb{R}^{n_x} \times \mathbb{S}_+^{n_x} \to \mathbb{S}_+^{n_x}$, must satisfy

$$\begin{aligned} f(\xi, \omega) - f(q_x(t), q_w) - A(t)(\xi - q_x(t)) \\ - B(t)(\omega - q_w) \in \mathcal{E}(0, \Omega_n(q_x(t), Q_x(t))) \,, \end{aligned} \tag{8}$$

for each $(\xi, \omega) \in \mathcal{E}(q_x(t), Q_x(t)) \times \mathcal{E}(q_w, Q_w)$.

From a tube MPC perspective, an ellipsoidal RFIT can be constructed by solving a nonlinear OCP over q_x, Q_x, u_x, λ, κ, and K, with initial condition

$$q_x(0) = \hat{x}_0 \quad \text{and} \quad Q_x(0) = \varepsilon I_{n_x} \in \mathbb{R}^{n_x \times n_x} \,, \tag{9}$$

for some small $\varepsilon > 0$.

The final ingredient for ellipsoidal tube MPC is to formulate the constraints $\overline{X}(t) \subseteq \mathbb{X}$, for the ellipsoidal tube parameterization. In our setting, the constraint $\forall t \in [0,T] : \mathcal{E}(q_x(t), Q_x(t)) \subseteq \mathbb{X}$ is satisfied if and only if

$$\begin{aligned} \forall t \in [0,T] \,, \, \forall i \in \{1, \ldots, n_c\} : \\ C_i^\mathsf{T} q_x(t) + \sqrt{C_i^\mathsf{T} Q_x(t) C_i} \leq c_i \,. \end{aligned} \tag{10}$$

Now, an ellipsoidal tube MPC controller requires solving

$$\inf_{\substack{q_x, Q_x, u_x, \\ \lambda, \kappa, K}} \int_0^T \mathcal{L}\left(\mathcal{E}(q_x(\tau), Q_x(\tau))\right) \mathrm{d}\tau$$

$$\text{s.t.} \quad \begin{cases} \text{ODES (6) with initial conditions (9)} \\ \text{State constraints (10) and inequalities (7)} \\ \lambda(t), \, \kappa(t) > 0, \; Q_x(t) \in \mathbb{S}_+^{n_x} \;\; \forall t \in [0,T] \\ \underline{u} \leq u_x(t) \leq \overline{u}, \qquad\qquad \forall t \in [0,T] \,, \end{cases} \tag{11}$$

on a receding time horizon. The generalized rotational inertia for the ellipsoidal cross-sections of the tube is then given by

$$\begin{aligned} \mathcal{L}(\mathcal{E}(q_x(t), Q_x(t))) = (q_x(t) - x_{\text{ref}})^\mathsf{T} P (q_x(t) - x_{\text{ref}}) \\ + \frac{\mathrm{Tr}(P Q_x(t))}{n_x + 2} \,. \end{aligned} \tag{12}$$

B. Tutorial: Setting up the Optimal Control Problem

This section discusses details regarding the construction of a tractable implementation of ellipsoidal tube MPC. OCP (11) requires n_x states for the central path and $({n_x}^2 + n_x)/2$ states for the shape matrix (due to symmetry) adding to 65 states in total. Likewise, the number of controls grows to $2 + n_u + n_u \times n_x = 35$ (λ, κ, u_x, and K). Defining the constraints for problem (11) is straightforward apart from (6), as a nonlinearity bounder needs to be constructed. This construction can be done automatically [10] or analytically using (8) (see [22]). Lemma 1 provides an explicit nonlinearity bounder for the 10D quadrotor model (5).

Lemma 1: The function $\Omega_n : \mathbb{R}^{10} \times \mathbb{S}_+^{10} \to \mathbb{S}_+^{10}$ with

$$\Omega_{n_{(i,i)}}(q, Q) = \begin{cases} \omega\left(q_j, \sqrt{Q_{(j,j)}}\right)^2 & \text{if } (i,j) = (4,7), (5,8) \\ 0 & \text{otherwise} \end{cases}$$

for each $i, j \in \{1, \ldots, 10\}$ with

$$\omega(q_j, \Delta_j) = \frac{g}{\cos^2(q_j)} \frac{\tan(\Delta_j) - \Delta_j + \Delta_j \tan(|q_j|) \tan(\Delta_j)}{1 - \tan(|q_j|) \tan(\Delta_j)}$$

is a nonlinearity bounder, in the sense of (8), for Eq. (5).

Proof: The nonlinearities in (5) have the form $f_i(\xi, \omega) = g \tan(\xi_j)$ for $(i,j) = (4,7), (5,8)$. Thus, the nonlinearity will have zeros everywhere except in the fourth and fifth diagonal elements. Setting $\delta_j = \xi_j - q_j$, the error of the first order expansion of f_i at q_j is given by

$$n_i(q_j, \delta_j) = g \left| \tan(q_j + \delta_j) - \tan(q_j) - \frac{\delta_j}{\cos^2(q_j)} \right| .$$

The addition theorem,

$$\tan(q_j + \delta_j) = \frac{\tan(q_j) + \tan(\delta_j)}{1 - \tan(q_j) \tan(\delta_j)} \,,$$

yields

$$\begin{aligned} n_i(q_j, \delta_j) &= \frac{g}{\cos^2(q_j)} \left| \frac{\tan(\delta_j) - \delta_j + \delta_j \tan(q_j) \tan(\delta_j)}{1 - \tan(q_j) \tan(\delta_j)} \right| \\ &\leq \underbrace{\frac{g}{\cos^2(q_j)} \frac{\tan(\Delta_j) - \Delta_j + \Delta_j \tan(|q_j|) \tan(\Delta_j)}{1 - \tan(|q_j|) \tan(\Delta_j)}}_{=: \omega(q_j, \Delta_j)} , \end{aligned}$$

which holds for all $\delta_j \in [-\Delta_j, \Delta_j]$. Setting $\Delta_j = \sqrt{Q_{(j,j)}}$ in the above inequality leads to a proof of this lemma. ∎

Once Ω_n has been defined, the next step is to work out explicitly the remaining constraints in (11).

C. Practical Aspects of Ellipsoidal Tube MPC

In this section we will discuss the implementation of ellipsoidal tube MPC in real-time. Just as in certainty-equivalent MPC, it is important to ensure recursive feasibility, that is, the ability to find at every time instant and for every initial condition a feasible state trajectory. This

issue was already addressed in [10], where it was suggested to construct an ellipsoidal forward invariant set $\mathcal{E}(x_{\text{ref}}, Q_{\text{ref}})$ around the reference trajectory and imposing the constraints $\mathcal{E}(q_x(t), Q_x(t)) \subseteq \mathcal{E}(x_{\text{ref}}, Q_{\text{ref}})$, for each $t \in [0,T]$, in (11). In principle, such a set can be constructed by solving

$$\inf_{\substack{Q_{\text{ref}}, u_{\text{ref}}, \lambda_{\text{ref}} \\ \kappa_{\text{ref}}, K_{\text{ref}}}} \quad \text{Tr}(Q_{\text{ref}})$$

$$\text{s.t.} \quad \begin{cases} 0 \succeq \Phi(x_{\text{ref}}, Q_{\text{ref}}, \lambda_{\text{ref}}, \kappa_{\text{ref}}, K_{\text{ref}}), & Q_{\text{ref}} \in \mathbb{S}^{n_x}_{++} \\ \underline{u} \leq u_{\text{ref}} \leq \overline{u}, \quad \lambda_{\text{ref}}, \kappa_{\text{ref}} > 0, & \underline{K} \leq K_{\text{ref}} \leq \overline{K} \\ \overline{u}_i \geq u_{\text{ref}_i} + \sqrt{K_{\text{ref}_i} Q_{\text{ref}} K_{\text{ref}_i}{}^{\mathsf{T}}}, & \forall i \in \{1, \ldots, n_u\} \\ \underline{u}_i \leq u_{\text{ref}_i} - \sqrt{K_{\text{ref}_i} Q_{\text{ref}} K_{\text{ref}_i}{}^{\mathsf{T}}}, & \forall i \in \{1, \ldots, n_u\} \end{cases}.$$

Closed-loop stability is another important aspect that must be taken into account when implementing MPC controllers. Ensuring stability for finite-horizon nonlinear MPC schemes, requires, among other conditions, a tracking (quadratic) stage cost [23], [1], [24]. Unfortunately, the trace term in the chosen stage cost prevents the application of nonlinear tracking MPC stability analysis techniques to tube MPC schemes. Instead, a rigorous stability analysis of the tube MPC scheme calls for the use of analysis techniques from the field of economic MPC [25], [26], which are, however, beyond the scope of this paper.

D. Initialization Procedure for Ellipsoidal Tube MPC

The proposed ellipsoidal tube MPC scheme is based on the solution of a standard nonconvex nonlinear optimal control problem. However, it is more expensive than its certainty-equivalent counterpart. The number of states increases by $(n_x + n_x^2)/2$, while the number of controls increases to $2 + n_u + n_u n_x$. Unlike the states, these controls have no physical interpretation. This makes finding good initial guesses, needed for gradient-based solvers, a challenging task. Heuristics for computing λ, κ, and K, have been proposed in [27]:

$$\lambda(t) = \frac{\sqrt{\text{Tr}\left(Q_x(t)\right)}}{\sqrt{\text{Tr}\left(B(t) Q_w B(t)^{\mathsf{T}}\right)}},$$

$$\kappa(t) = \frac{\sqrt{\text{Tr}\left(Q_x(t)\right)}}{\sqrt{\text{Tr}\left(\Omega_n(q_x(t), Q_x(t))\right)}}, \quad \text{and} \quad K(t) = \alpha(t)\widetilde{K}(t).$$

Here, the feedback gain is $\widetilde{K}(t) = G^{\mathsf{T}} P(t)$, with $P(t) \in \mathbb{S}^{n_x}_{+}$ being the solution of a perturbed Riccati algebraic equation

$$\widehat{A}(t)P(t) + P(t)\widehat{A}(t)^{\mathsf{T}} - P(t)GG^{\mathsf{T}}P(t) + I = 0$$

with $\widehat{A}(t) = \left(A(t) + \left(\frac{1}{\kappa(t)} + \frac{1}{\lambda(t)}\right) I\right)$. The factor

$$\alpha(t) = \min_i \left(\frac{\overline{u}_i - u_{x,i}(t)}{\sqrt{K_i(t) Q_x(t) K_i(t)^{\mathsf{T}}}}, \; \frac{u_{x,i}(t) - \underline{u}_i}{\sqrt{K_i(t) Q_x(t) K_i(t)^{\mathsf{T}}}} \right)$$

is used to scale the feedback to satisfy the control bounds. Now, a valid, albeit suboptimal, RFIT can be computed by optimizing only over the function u_x. This suboptimal RFIT is used as an initial guess for (11).

E. A Hierarchical Controller for Real-Time Tube MPC

Figure 1 presents a two-layer hierarchical robust controller design. In the lower-frequency layer, the tube MPC problem (11) is solved once a measurement arrives. The tube states and controls are then sent to a feedback block, where an explicit feedback law is evaluated. This feedback law is used as a lower-level controller, to deliver a control signal at a much faster sampling rate. At the next sampling time, the feedback law is updated in the higher-level controller.

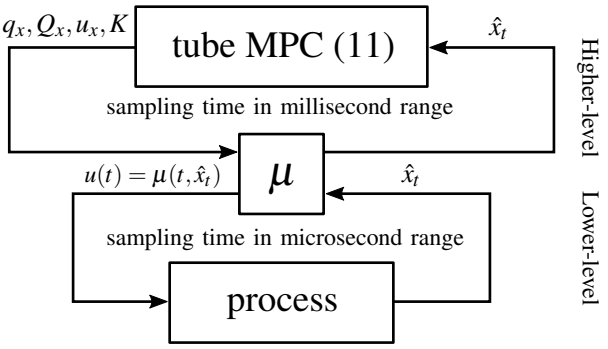

Fig. 1. A two-layer control loop for ellipsoidal tube MPC. The ellipsoidal tube MPC is used to update an explicit feedback law μ. This feedback law is robust and can be used to provide a control signal at a microsecond range.

The tube MPC formulation presented in this paper also provides a natural way to implement a feedback law in real-time. Recall that every RFIT is induced by at least one feedback law which, if known explicitly, could be used to provide feedback at a much higher sampling rate. Using the natural boundary feedback law of the ellipsoidal RFIT [10]

$$\mu(\tau, x(\tau)) = \begin{cases} u_x(\tau) & , \text{ if } x(\tau) = q_x(\tau) \\ u_x(\tau) - \dfrac{(K(\tau) Q_x(\tau) K(\tau)^{\mathsf{T}}) G^{\mathsf{T}} d(x(\tau))}{\left\| Q_x(\tau)^{\frac{1}{2}} K(\tau)^{\mathsf{T}} G^{\mathsf{T}} d(x(\tau)) \right\|_2}, & \text{otherwise} \end{cases}$$

with

$$d(x(\tau)) = \frac{Q_x(\tau)^{\dagger}(x(\tau) - q_x(\tau))}{\|Q_x(\tau)^{\dagger}(x(\tau) - q_x(\tau))\|_2},$$

where $(\cdot)^{\dagger}$ denotes the Moore-Penrose pseudoinverse.

One of the biggest concerns when implementing MPC controllers is that of the computational time needed to solve the nonlinear OCP at each sampling time. Nowadays tailored implementations for real-time nonlinear MPC can be auto-generated. For example, a tool for auto-generating RTI schemes can be found in [19]. This RTI scheme uses multiple-shooting to discretize the OCP and exploits the fact that the sequence of optimization problems are neighboring. This is exploited for the initialization of subsequent problems via an initial value embedding strategy [28], which allows to perform only one iteration per optimization problem. One further advantage of this scheme is that each sampling time is divided into a short feedback phase and a longer phase where the next feedback is prepared as much as possible without knowledge of the next state. The computational time for this

RTI scheme can further be reduced by the use of automatic code generation tools [19].

IV. Numerical Results

In this section we illustrate the performance of ellipsoidal tube MPC by applying it to the quadrotor stabilization problem from Section II-B. All numerical experiments presented in this section were implemented in `ACADO Toolkit` [29]. In particular, we used the automatic code generation module through its `MATLAB` interface to export an RTI scheme. The exported solver used multiple shooting with a 4th order Runge-Kutta integrator (RK4). The QP subproblems were solved using an online active set strategy implemented in `qpOASES` [30]. The code was compiled using the `GNU Compiler Collection` (`https://gcc.gnu.org/`) version 4.8. All the results were obtained on an Ubuntu 14.04 LTS operating system with a 1.7 GHz processor and 8GB of RAM (1600 MHz).

A. Closed-Loop Performance: Certainty-Equivalent MPC

Figure 2 shows the closed-loop performance of the certainty-equivalent MPC controller with and without uncertainties. The controller uses a prediction horizon of $T = 1.2\,\mathrm{s}$, with 6 shooting intervals and 18 Runge-Kutta integration steps in total. The nominal closed-loop scenario was obtained using the center q_w of the ellipsoidal uncertainty bounds. For the disturbed trajectories, the dynamic system is simulated under uniformly distributed process noise. As expected, the certainty-equivalent MPC controller under nominal conditions (blue), manages to drive the quadrotor to the desired position while steering it away from the wall. On the other hand, when subject to disturbances the system violates the constraint (red trajectories). The average computational time per RTI was $159\,\mu\mathrm{s}$.

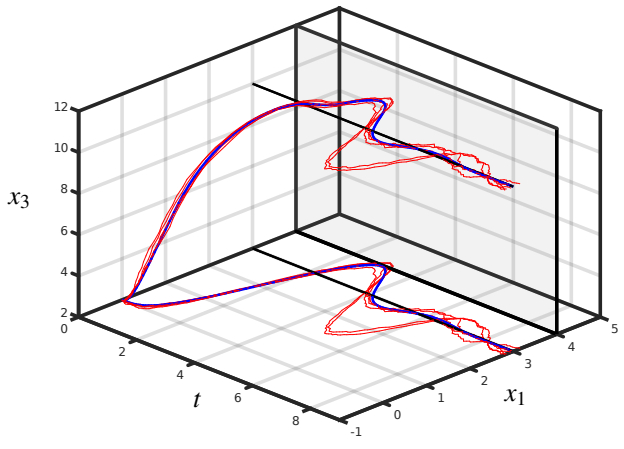

Fig. 2. Closed-loop performance of certainty-equivalent MPC. The nominal and perturbed trajectories are shown in blue and red respectively. Constraint violations (around $t = 3\mathrm{s}$) can be observed projected in the (x_1, t)-plane.

B. Closed-Loop Performance: Ellipsoidal Tube MPC

The ellipsoidal tube MPC controller was set up using the procedure described in Section III. In order to provide a good initial guess to the solver, a reduced ellipsoidal tube optimal control problem was solved. The reduced problem was formulated as (11) setting λ, κ, and K per [27] (cf. Section III-D), and only optimizing over the control input u_x, which was initialized using the control input for the nominal certainty-equivalent controller. Figure 3 shows a projection of the suboptimal ellipsoidal RFIT onto the (t, x_1, x_3)-space.

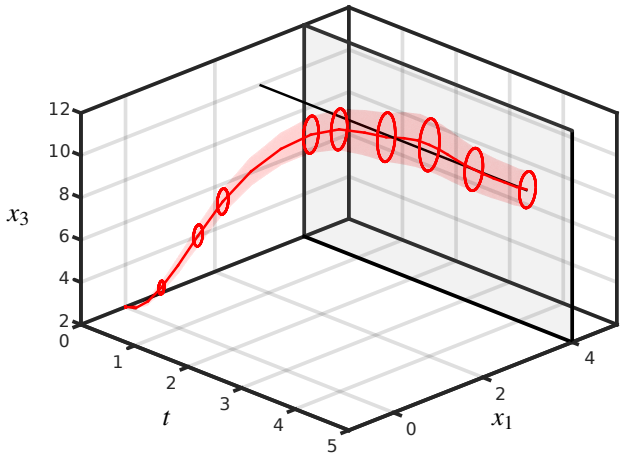

Fig. 3. Suboptimal ellipsoidal RFIT obtained from the initialization procedure. The central path and tube cross-sections are depicted in red.

After obtaining the suboptimal ellipsoidal RFIT, the tube MPC controller was implemented in closed-loop. The controller was set up using a prediction horizon of $T = 0.6\,\mathrm{s}$, divided into 3 shooting intervals and discretized using 9 integrator steps. Figure 4 shows a projection of the tubes for the closed-loop controller on the (t, x_1, x_3)-space, together with the perturbed trajectories. The controller is able to steer the system towards the reference position, while avoiding constraint violations.

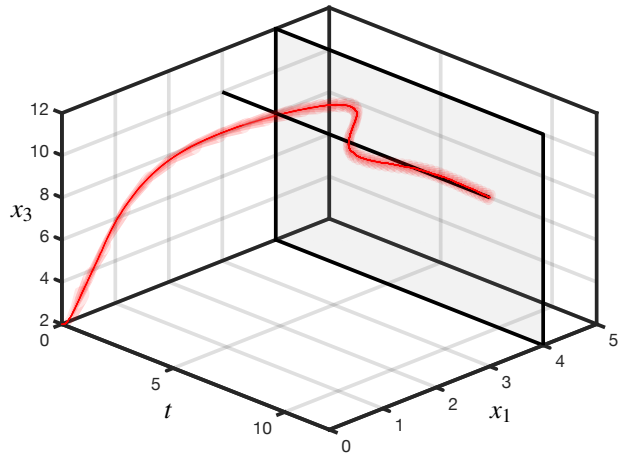

Fig. 4. Closed-loop performance of ellipsoidal tube MPC. The closed-loop trajectory is shown as a red line and the shaded red area.

The average computational time for the tube MPC controller was $81.6\,\mathrm{ms}$ per RTI. Given this time, the faster loop was only implemented with a sampling time of $20\,\mu\mathrm{s}$. Ta-

ble II shows a breakdown for the average computational time per RTI into its most time-consuming phases. Unsurprisingly, most of the time is spent on the integration and sensitivity generation.

TABLE II

RUN-TIME PERFORMANCE OF THE AUTO-GENERATED SCHEME FOR TUBE MPC.

Higher-level	CPU Time	%
Integration & sensitivities	77.3 ms	95%
Condensing	3.6 ms	4%
QP solution (with qpOASES)	0.6 ms	1%
One complete RTI	81.6 ms	100%
Lower-level	CPU Time	%
Explicit feedback evaluation μ	20 µs	100%

V. CONCLUSIONS

This paper has presented a real-time implementation of tube MPC for a 10-state quadcopter model. This tube MPC scheme uses RFITs with ellipsoidal cross-sections. The parameterization, allows to compute the tube by solving a standard optimal control problem, albeit of an increased size compared to the original certainty-equivalent MPC formulation. In particular, the ellipsoidal tube MPC problem requires $(n_x^2+n_x)/2$ extra states, $n_x n_u$ controls, $2n_x n_u$ extra nonlinear constraints mixing states and controls as well as 2 bound constraints. The ellipsoidal tube MPC problem is solved using an auto-generated RTI scheme. The controller is then implemented as a hierarchical controller, where the tube MPC problem is used to update an explicit feedback law associated to the ellipsoidal tube, which is in turn used to control the process at a higher sampling rate. The computational experiments, based on tracking a hovering position for a simplified 10-state nonlinear quadcopter model show that a tube MPC controller can be implemented on a millisecond (microsecond) range for the MPC loop.

ACKNOWLEDGMENTS

This research was supported by National Science Foundation China (NSFC), Nr. 61473185, as well as ShanghaiTech University, Grant-Nr. F-0203-14-012. In addition, we thank Dr. Mo Chen for providing the 10-state quadrotor model.

REFERENCES

[1] J. Q. Rawlings and D. Q. Mayne, *Viability theory*. Systems & Control: Foundations & Applications. Birkhäuser Boston, Inc., Boston, MA., 1991.

[2] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: theory and practicea survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.

[3] F. Allgöwer and A. Zheng, *Nonlinear model predictive control*. Birkhäuser, 2012, vol. 26.

[4] G. Pannocchia, J. B. Rawlings, and S. J. Wright, "Conditions under which suboptimal nonlinear mpc is inherently robust," *Systems & Control Letters*, vol. 60, no. 9, pp. 747–755, 2011.

[5] A. Bemporad and M. Morari, "Robust model predictive control: A survey," *Robustness in identification and control*, pp. 207–226, 1999.

[6] M. V. Kothare, V. Balakrishnan, and M. Morari, "Robust constrained model predictive control using linear matrix inequalities," *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996.

[7] D. Q. Mayne, S. V. Raković, R. Findeisen, and F. Allgöwer, "Robust output feedback model predictive control of constrained linear systems: Time varying case," *Automatica*, vol. 45, no. 9, pp. 2082–2087, 2009.

[8] S. Yu, M. Reble, H. Chen, and F. Allgöwer, "Inherent robustness properties of quasi-infinite horizon nonlinear model predictive control," *Automatica*, vol. 50, no. 9, pp. 2269–2280, 2014.

[9] S. V. Rakovic, B. Kouvaritakis, M. Cannon, C. Panos, and R. Findeisen, "Parameterized tube model predictive control," *IEEE Transactions on Automatic Control*, vol. 57, no. 11, pp. 2746–2761, 2012.

[10] M. E. Villanueva, R. Quirynen, M. Diehl, B. Chachuat, and B. Houska, "Robust mpc via min–max differential inequalities," *Automatica*, vol. 77, pp. 311–321, 2017.

[11] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[12] J. P. Aubin, *Viability theory*. Systems & Control: Foundations & Applications. Birkhäuser Boston, Inc., Boston, MA., 1991.

[13] W. Langson, I. Chryssochoos, S. V. Raković, and D. Q. Mayne, "Robust model predictive control using tubes," *Automatica*, vol. 40, no. 1, pp. 125–133, 2004.

[14] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer, "Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations," *Journal of Process Control*, vol. 12, no. 4, pp. 577–585, 2002.

[15] R. Milman and E. J. Davison, "A fast mpc algorithm using nonfeasible active set methods," *Journal of optimization theory and applications*, vol. 139, no. 3, pp. 591–616, 2008.

[16] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 2, pp. 267–278, 2010.

[17] V. M. Zavala and L. T. Biegler, "The advanced-step nmpc controller: Optimality, stability and robustness," *Automatica*, vol. 45, no. 1, pp. 86–93, 2009.

[18] T. Ohtsuka, "A continuation/gmres method for fast computation of nonlinear receding horizon control," *Automatica*, vol. 40, no. 4, pp. 563–574, 2004.

[19] B. Houska, H. J. Ferreau, and M. Diehl, "An auto-generated real-time iteration algorithm for nonlinear mpc in the microsecond range," *Automatica*, vol. 47, no. 10, pp. 2279–2285, 2011.

[20] M. N. Zeilinger, D. M. Raimondo, A. Domahidi, M. Morari, and C. N. Jones, "On real-time robust model predictive control," *Automatica*, vol. 50, no. 3, pp. 683–694, 2014.

[21] P. Bouffard, "On-board model predictive control of a quadrotor helicopter: Design, implementation, and experiments," Department of Computer Science, University of California Berkeley, Tech. Rep., 2012.

[22] B. Houska, F. Logist, J. Van Impe, and M. Diehl, "Robust optimization of nonlinear dynamic systems with application to a jacketed tubular reactor," *Journal of Process Control*, vol. 22, no. 6, pp. 1152–1160, 2012.

[23] 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.

[24] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[25] R. Amrit, J. B. Rawlings, and D. Angeli, "Economic optimization using model predictive control with a terminal cost," *Annual Reviews in Control*, vol. 35, no. 2, pp. 178–186, 2011.

[26] L. Grüne, "Economic receding horizon control without terminal constraints," *Automatica*, vol. 49, no. 3, pp. 725–734, 2013.

[27] M. E. Villanueva, J. C. Li, X. Feng, B. Chachuat, and B. Houska, "Computing ellipsoidal robust forward invariant tubes for nonlinear mpc," in *In Proceedings of the 20th IFAC World Congress, Toulouse, France*, 2017, pp. 7436–7441.

[28] M. Diehl, "Real-time optimization for large scale nonlinear processes," Ph.D. dissertation, Universität Heidelberg, 2001.

[29] B. Houska, H. J. Ferreau, and M. Diehl, "Acado toolkitan open-source framework for automatic control and dynamic optimization," *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, 2011.

[30] H. Ferreau, C. Kirches, A. Potschka, H. Bock, and M. Diehl, "qpOASES: A parametric active-set algorithm for quadratic programming," *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.