TuneMPC - A Tool for Economic Tuning of Tracking (N)MPC Problems

Jochem De Schutter[1], Mario Zanon[2], Moritz Diehl[1,3]

Abstract— **Economic nonlinear model predictive control (NMPC) is a variant of NMPC that directly optimizes an economic performance index instead of a tracking error. Although economic NMPC can achieve excellent closed-loop performance, the associated computational effort as well as the difficulty of guaranteeing stability in practice are its main drawbacks. Motivated by these difficulties, a formal procedure was developed that tunes a tracking (non)linear MPC scheme so that it is first-order equivalent to economic NMPC. This paper introduces TuneMPC, a new open-source software framework that closes the gap between the underlying theory and practical application of this tuning procedure. For user-provided system dynamics, constraints and economic objective, TuneMPC enables automated computation of optimal steady states and periodic trajectories, and returns the corresponding tuned stage cost matrices. To demonstrate the potential of the tool, we apply the technique to the challenging example of an autonomous tethered aircraft flying periodic orbits for airborne wind energy harvesting.**

I. INTRODUCTION

Nonlinear model predictive control (NMPC) is an advanced control technique that owes its growing popularity to its ability to explicitly handle constrained nonlinear systems with multiple in- and outputs [15]. In tracking NMPC, a quadratic cost function that penalizes tracking error is minimized, in order to steer the system back from its initial state to the given reference. This reference can be the result of an offline optimization procedure with respect to an economic performance index. Economic NMPC (ENMPC), on the other hand, directly optimizes the economic performance index online, leading to superior closed-loop performance in many applications [4].

Despite recent advances in ENMPC stability theory [14], establishing nominal stability is in practice not straightforward. Stability proofs for ENMPC with and without terminal constraints typically rely on a strict dissipativity assumption [2, 11] that can only be verified in special cases. Other approaches suffer from similar drawbacks, such as ENMPC using Lyaponuv constraints [12], which relies on the non-trivial construction of a suitable Lyapunov function for the nonlinear system. On the contrary, nominal stability of tracking NMPC is arguably easier to enforce [10, 15].

On the computational side, tailored algorithms that allow the efficient numerical solution of tracking problems in real-time cannot be directly applied to economic NMPC without losing good convergence properties or even stability guarantees. A notable example is the Real-Time Iteration (RTI) scheme using the Gauss-Newton Hessian approximation [7].

With the aim of overcoming these drawbacks of economic MPC, a procedure was proposed in [21] to compute positive definite stage cost matrices for a tracking (N)MPC scheme, so that it delivers a feedback law that is first-order equivalent to that of economic NMPC. In [22], this analysis was extended to the case of optimal periodic operation.

The resulting economically tuned tracking NMPC scheme combines the benefits of economic and tracking NMPC: good closed-loop performance in the neighborhood of the optimal operating point and nominal stability in the presence of large state perturbations. An additional practical advantage is that with this technique, controller tuning relies on heuristics less than standard tracking NMPC (ideally not at all) and can be done in a more systematic and automated way. The potential of the tuning technique has been demonstrated in simulations considering, e.g., a CSTR example [20]; an evaporation process [21] and energy-optimal autonomous driving [13].

This paper introduces TuneMPC, a Python toolbox for economic tuning of tracking (N)MPC problems around an economically optimal steady-state or periodic trajectory. The main goal of the software is to close the gap between the theory established in [21, 22] and its application in practice by facilitating the:

- necessary modifications to the optimal control problem;
- computation of the optimal (periodic) operating point;
- computation of positive-definite stage cost matrices that quadratically approximate the economic cost;
- rapid MPC prototyping for embedded application.

These steps can be performed in a couple of lines of code, without needing a deep understanding of the underlying procedure nor having to deal with implementation details. The toolbox is freely available [1] and open-source under the GNU LGPLv3, which allows use in proprietary software. It is written in Python with a lean interface based on the symbolic framework CasADi [3], which is also openly available. Future software updates will provide a user-friendly MATLAB interface as well.

The remainder of the text is structured as follows. Section II discusses the theoretical background of the tuning technique. Section III then presents the software itself and outlines via code examples the necessary steps to obtain the economically tuned matrices of a tracking NMPC scheme. Section IV showcases the potential of the software through the example of optimal periodic operation of an autonomous tethered aircraft for airborne wind energy.

This research was supported by an industrial project with the company Kiteswarms Ltd., by the German Federal Ministry for Economic Affairs and Energy (BMWi) via eco4wind (0324125B) and DyConPV (0324166B), and by DFG via Research Unit FOR 2401.

[1]Department of Microsystems Engineering, University of Freiburg, Germany. jochem.de.schutter@imtek.de

[2]IMT School for Advanced Studies Lucca, Italy.

[3]Department of Mathematics, University of Freiburg, Germany.

II. PRELIMINARIES

We consider nonlinear discrete-time systems of the form

$$x_{k+1} = f_k(x_k, u_k) \ , \tag{1}$$

that should be operated so that the cost $\sum_{k=0}^{\infty} l_k(x_k, u_k)$ is minimized, while satisfying the constraints $h_k(x, u) \geq 0$. We assume that all functions are periodically time-varying with period p known a priori. We discern explicitly between linear and nonlinear constraints:

$$h_k(x, u) := \left[h_{k,\mathrm{lin}}(x, u)^\top, h_{k,\mathrm{nl}}(x, u)^\top\right]^\top \ . \tag{2}$$

Let $\mathbb{I}_a^b$ denote the integer set $\{a, \ldots, b\}$. Then, the optimal $p-$periodic orbit is the solution of the following offline periodic optimal control problem (POCP):

$$\min_{x,\, u} \quad \sum_{k=0}^{p-1} l_k(x_k, u_k) \tag{3a}$$

$$\text{s.t.} \quad x_{k+1} = f_k(x_k, u_k), \qquad k \in \mathbb{I}_0^{p-2}, \tag{3b}$$

$$x_0 = f_{p-1}(x_{p-1}, u_{p-1}), \tag{3c}$$

$$h_k(x_k, u_k) \geq 0, \qquad k \in \mathbb{I}_0^{p-1} \tag{3d}$$

Let p be the period yielding the lowest cost, which we assume to be unique, and x_k^p, u_k^p denote the corresponding optimal solution. The steady-state case is obtained for $p = 1$.

We assume that the stage cost function $l_k(x, u)$ is of *economic* type, i.e. $\forall\ k \in \mathbb{I}_0^{p-1}$, $\nexists$ a class-$\mathcal{K}$ function α s.t.

$$l_k(x, u) \geq \alpha(\|x - x_k^\mathrm{p}\|), \ \forall (x, u) \in \mathbb{Z}_k \ , \tag{4}$$

with $\mathbb{Z}_k := \{(x, u) \mid h_k(x, u) \geq 0\}$. In the converse case, the stage cost is defined to be of *tracking* type.

At each sampling instant i, economic NMPC solves the following finite-horizon OCP:

$$w^* := \ \arg\min_{w} \quad \sum_{k=0}^{N-1} l_{[k]}(x_k, u_k) \tag{5a}$$

$$\text{s.t.} \quad x_0 - \hat{x}_0 = 0, \tag{5b}$$

$$x_{k+1} = f_{[k]}(x_k, u_k), \ k \in \mathbb{I}_0^{N-1}, \tag{5c}$$

$$h_{[k]}(x_k, u_k) \geq 0, \qquad k \in \mathbb{I}_0^{N-1}, \tag{5d}$$

$$x_N - x_{[N]}^\mathrm{p} = 0, \tag{5e}$$

with $w := \left[x_0^\top, u_0^\top, x_1^\top, u_1^\top, \ldots, u_{N-1}^\top, x_N^\top\right]^\top$. The notation $[k] := (i + k) \mod p$ allows reference to periodic functions and quantities using the NMPC time index k. The parameter $\hat{x}_0$ denotes the current state estimate and economic NMPC applies the feedback law $\pi_i^\mathrm{e}(\hat{x}_0) := u_0^*$ until the next state measurement becomes available. For simplicity, we employ a terminal point constraint. However, any other stabilizing terminal condition can be used.

A. First-Order Equivalent MPC Formulations

As mentioned in the introduction, the two main difficulties related to economic NMPC are the difficulty in providing stability guarantees and the increased computational effort. In order to mitigate these drawbacks, a "tuning" procedure was proposed in [21, 22] that computes positive definite stage cost matrices for a (non)linear tracking MPC scheme so that it is first-order equivalent to the economic NMPC scheme (5), i.e., the tracking feedback policy $\pi_i^\mathrm{t}(\cdot)$ satisfies

$$\|\pi_i^\mathrm{t}(\hat{x}_0) - \pi_i^\mathrm{e}(\hat{x}_0)\| = \mathcal{O}(\|\hat{x}_0 - x_{[i]}^\mathrm{p}\|^2) \ . \tag{6}$$

In the special case of linear dynamics and indefinite quadratic cost, the equivalence is exact, i.e., $\pi_i^{\mathrm{t,lq}}(\hat{x}_0) = \pi_i^{\mathrm{e,lq}}(\hat{x}_0)$.

We further distinguish between two types of equivalence. For *type A*, the set of $\hat{x}_0$ for which the equivalence holds is independent of the active set of the MPC problem. For *type B*, the equivalence is only guaranteed to hold for initial states for which the optimal active set of the MPC problem includes all constraints which are active for the optimal orbit. For more details and a formal discussion we refer to [21, 22].

B. Technical Problem Modifications

In order to guarantee positive-definiteness of the stage cost matrices of the first-order equivalent tracking NMPC problem, the contribution of the nonlinear equality and inequality constraints to the Lagrangian Hessian blocks must be zero at the optimal orbit [21, section 7]. To achieve this, the following two modifications are made to Problem (3).

First, the nonlinear inequality constraints are reformulated using slack variables s, so that all inequalities $\hat{h}_k$ become linear and the nonlinear equalities g_k are added:

$$\hat{h}_k(x_k, u_k, s_k) := \left[h_{k,\mathrm{lin}}(x_k, u_k)^\top, s_k^\top\right]^\top \geq 0 \ , \tag{7}$$

$$g_k(x_k, u_k, s_k) := h_{k,\mathrm{nl}}(x_k, u_k) - s_k = 0 \ . \tag{8}$$

The reformulation of POCP (3) then reads as

$$\min_{x,\, u,\, s} \quad \sum_{k=0}^{p-1} l_k(x_k, u_k) + \|g_k(x_k, u_k, s_k)\|^2 \tag{9a}$$

$$\text{s.t.} \quad (3b), (3c), \tag{9b}$$

$$g_k(x_k, u_k, s_k) = 0, \quad k \in \mathbb{I}_0^{p-1}, \tag{9c}$$

$$\hat{h}_k(x_k, u_k, s_k) \geq 0, \quad k \in \mathbb{I}_0^{p-1} \ . \tag{9d}$$

By construction, the optimal state and control sequences x_k^p and u_k^p of Problem (3) are also optimal for Problem (9) with $s_k^\mathrm{p} = h_{k,\mathrm{nl}}(x_k^\mathrm{p}, u_k^\mathrm{p})$.

In order to define the second modification, we first define the optimal multipliers λ_k^p, μ_k^p and ν_k^p related to (9b), (9c), and (9d) respectively. We then define a periodic series of *rotated* time-varying economic cost functions for $k \in \mathbb{I}_0^{p-1}$:

$$\begin{aligned} L_k(x_k, u_k, s_k) &:= l_k(x_k, u_k) + \mu_k^{\mathrm{p}\top} g_k(x_k, u_k, s_k) \\ &\quad + \lambda_k^{\mathrm{p}\top} x_k - \lambda_{k+1}^{\mathrm{p}\top} f_k(x_k, u_k) + \|g_k(x_k, u_k, s_k)\|^2, \end{aligned} \tag{10}$$

which we use to define the following rotated POCP:

$$\min_{x,\, u,\, s} \quad \sum_{k=0}^{p-1} L_k(x_k, u_k, s_k), \quad \text{s.t. (9b), (9c), (9d)} \ . \tag{11}$$

The primal solution $\bar{x}_k^\mathrm{p}, \bar{u}_k^\mathrm{p}, \bar{s}_k^\mathrm{p}$ of Problem (11) is identical to that of Problem (9), but the optimal multipliers are $\bar{\lambda}_k^\mathrm{p} = 0$, $\bar{\mu}_k^\mathrm{p} = 0$ and $\bar{\nu}_k^\mathrm{p} = \nu_k^\mathrm{p}$, as proven in [21, 22].

Problem (11) has the desired property that at the optimal operating point neither the nonlinear equality constraints (zero multipliers) nor the (linear) inequality constraints contribute to the Lagrangian Hessian blocks, i.e.,

$$\nabla^2_{w_k}\mathcal{L}_k(\bar{w}^{\mathrm{p}}_k,\bar{\lambda}^{\mathrm{p}}_k,\bar{\mu}^{\mathrm{p}}_k,\bar{\nu}^{\mathrm{p}}_k)=\nabla^2_{w_k}L_k(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k,\bar{s}^{\mathrm{p}}_k)\ , \tag{12}$$

with $w_k := \left[x_k^\top, u_k^\top, s_k^\top\right]^\top$. Note that it is not possible to rotate the cost using the (active) nonlinear inequality constraints without changing the primal solution of Problem (9) [21, Lemma 6], hence the need for the slack reformulation.

The reformulated rotated economic NMPC scheme that stabilizes the system around the optimal solution of (11) is:

$$\begin{aligned}
\min_{x,u,s}\quad & \sum_{k=0}^{N-1} L_{[k]}(x_k,u_k,s_k) && \text{(13a)}\\
\text{s.t.}\quad & \text{(5b), (5c), (5e)}, && \text{(13b)}\\
& g_{[k]}(x_k,u_k,s_k)=0,\quad k\in\mathbb{I}_0^{N-1}, && \text{(13c)}\\
& \hat{h}_{[k]}(x_k,u_k,s_k)\geq 0,\quad k\in\mathbb{I}_0^{N-1}. && \text{(13d)}
\end{aligned}$$

Since this rotated scheme has the same primal solution as (5), the feedback policies coincide. Note that we do not solve Problem (13) online, but we use it to construct a tracking NMPC scheme that is first-order equivalent to Problem (5).

C. Indefinite Linear-Quadratic Approximation

We define

$$\begin{aligned}
H_k &:= \nabla^2_{w_k}\mathcal{L}_k(\bar{w}^{\mathrm{p}}_k,\bar{\lambda}^{\mathrm{p}}_k,\bar{\mu}^{\mathrm{p}}_k,\bar{\nu}^{\mathrm{p}}_k), & q_k &:= \nabla_{w_k}L_k(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k,\bar{s}^{\mathrm{p}}_k),\\
A_k &:= \nabla_{x_k}f_k(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k)^\top, & B_k &:= \nabla_{u_k}f_k(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k)^\top,\\
G_k &:= \nabla_{w_k}g_k(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k,\bar{s}^{\mathrm{p}}_k)^\top, & r_k &:= g_k(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k,\bar{s}^{\mathrm{p}}_k),\\
C_k &:= \nabla_{w_k}\hat{h}_k(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k,\bar{s}^{\mathrm{p}}_k)^\top, & e_k &:= \hat{h}_k(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k,\bar{s}^{\mathrm{p}}_k),
\end{aligned}$$

with all expressions evaluated at the primal-dual solution of Problem (11), and with every block of the Lagrangian Hessian given by (12). Remark that $H_k \succ 0$ does not hold in general, so that the following linear-quadratic (LQ) problem is possibly indefinite:

$$\begin{aligned}
\min_{x,u,s}\quad & \sum_{k=0}^{N-1}\frac{1}{2}\Delta w_k^\top H_{[k]}\Delta w_k + q_{[k]}^\top\Delta w_k && \text{(14a)}\\
\text{s.t.}\quad & x_0-\hat{x}_0=0, && \text{(14b)}\\
& \Delta x_{k+1}=A_{[k]}\Delta x_k+B_{[k]}\Delta u_k,\ k\in\mathbb{I}_0^{N-1}, && \text{(14c)}\\
& G_{[k]}\Delta w_k=0, \qquad k\in\mathbb{I}_0^{N-1}, && \text{(14d)}\\
& C_{[k]}\Delta w_k+e_{[k]}\geq 0 \qquad k\in\mathbb{I}_0^{N-1}, && \text{(14e)}\\
& \Delta x_N = 0\ , && \text{(14f)}
\end{aligned}$$

where we define $\Delta w_k := w_k - \bar{w}^{\mathrm{p}}_{[k]}$. The feedback law delivered by the LQ scheme (14) is type A first-order equivalent to the feedback law delivered by the economic NMPC scheme (13) [21, Theorem 4].

D. Hessian Convexification

We now present the Hessian convexification procedure that can be used to construct the tracking LQ problem

$$\begin{aligned}
\min_{x,u,s}\quad & \sum_{k=0}^{N-1}\frac{1}{2}\Delta w_k^\top \tilde{H}_{[k]}\Delta w_k + q_{[k]}^\top\Delta w_k && \text{(15a)}\\
\text{s.t.}\quad & \text{(14b)} - \text{(14f)}, && \text{(15b)}
\end{aligned}$$

that is (type A or B) equivalent to the indefinite LQ problem (14). For $k\in\mathbb{I}_0^{p-1}$ we define the matrix sequences

$$\tilde{H}_k := H_k + \delta H_k\ , \tag{16}$$

$$\delta H_k := \tfrac{1}{\alpha}\left(\mathcal{H}_k([\delta P]^p)+\eta_{\mathrm{F}}C_{k,\mathbb{A}_k}^\top F_k C_{k,\mathbb{A}_k}+\eta_{\mathrm{T}}T_k\right),\tag{17}$$

$$\mathcal{H}_k([\delta P]^p) := \begin{bmatrix} A_k^\top\delta P_{k+1}A_k-\delta P_k & A_k^\top\delta P_{k+1}B_k\\ B_k^\top\delta P_{k+1}A_k & B_k^\top\delta P_{k+1}B_k\end{bmatrix},\tag{18}$$

with $[\delta P]^p$ a periodic series of symmetric matrices. Matrices $C_{k,\mathbb{A}_k}$ denote the Jacobian of the active inequality constraints, where $\mathbb{A}_k$ the set of strictly active inequality constraints at time k of the optimal solution of (11):

$$\mathbb{A}_k := \{i \mid \hat{h}_i(\bar{x}^{\mathrm{p}}_k,\bar{u}^{\mathrm{p}}_k,\bar{s}^{\mathrm{p}}_k)=0,\ \bar{\nu}^{\mathrm{p}}_{k,i}>0\}\ . \tag{19}$$

These definitions allow us to formulate the following convex semidefinite program (SDP):

$$\begin{aligned}
\min_{\delta P,F,T,\alpha,\beta}\quad & \beta+\rho\sum_{k=0}^{p-1}(\|F_k\|+\|T_k\|) && \text{(20a)}\\
\text{s.t.}\quad & \beta I\succeq\alpha\tilde{H}_k\succeq I,\quad k\in\mathbb{I}_0^{p-1}, && \text{(20b)}\\
& \alpha>0\ . && \text{(20c)}
\end{aligned}$$

The proposed convexification procedure computes the variables α, $[\delta P]^p$, $[F]^p$ and $[T]^p$ by solving SDP (20) in several steps, each corresponding to a different pair of values of parameters $\eta_{\mathrm{F}},\eta_{\mathrm{T}}\in\{0,1\}$.

Step 1: Solve SDP (20) with $\eta_{\mathrm{F}}=\eta_{\mathrm{T}}=0$. Here, the maximum condition number β of the convexified Hessians is minimized while constraining them to be positive definite. If a solution exists, Problem (15) delivers a feedback law that is type A equivalent to the feedback law delivered by Problem (14). Note that Problem (15) is of tracking type since by construction $\tilde{H}_k \succ 0$, and at the optimal orbit, the gradient directions q_k are blocked by active inequality constraints. If the problem is infeasible, proceed to step 2.

Step 2: Solve SDP (20) with $\eta_{\mathrm{F}}=1$ and $\eta_{\mathrm{T}}=0$. Here, we allow additional curvature to be added specifically in the direction of the strictly active inequality constraints. If a solution exists, the feedback law implemented by Problem (15) is type B equivalent to the feedback law of the indefinite LQ problem. The user-defined tuning parameter $\rho\geq 0$ weighs off Hessian matrix conditioning and the amount of "Hessian distortion" in the direction of the active constraints.

If the problem remains infeasible, it was proven in [21, Corollary 8] that strict dissipativity does not hold for the indefinite LQ problem (14), which therefore does not deliver a stabilizing feedback law. Also in this case, the system is not optimally operated at solution $\bar{x}^{\mathrm{p}}_k$, $\bar{u}^{\mathrm{p}}_k$. This means that other operating modes, e.g., a periodic orbit for a different value of p, can outperform the current operating point. In case the current operating point is preferred, it is still possible to enforce the tracking LQ MPC scheme to be stabilizing while approximating the economic cost as closely as possible:

Step 3: Solve SDP (20) with $\eta_{\mathrm{F}}=1$ and $\eta_{\mathrm{T}}=1$. Here the solver additionally finds the matrices $[T]^p$ that render the Hessian matrices positive definite, while regularizing the reduced Hessians as little as possible. In this case, the convexified Hessians $\tilde{H}_k$ can be used as a tuning heuristic to approximate the indefinite quadratic cost, without any equivalence guarantees.

E. First-Order Equivalent Tracking NMPC

The tracking NMPC equivalence is obtained as a direct generalization from the tracking LQ MPC case. Let us define the tracking NMPC problem as:

$$\min_{x,u,s} \quad \sum_{k=0}^{N-1} \frac{1}{2}\Delta w_k^\top \tilde{H}_{[k]} \Delta w_k + q_{[k]}^\top \Delta w_k \tag{21a}$$

$$\text{s.t.} \quad (13b), (13c), (13d)\ . \tag{21b}$$

If system (1) is optimally operated at the optimal orbit $\bar{x}_k^{\mathrm{p}}, \bar{u}_k^{\mathrm{p}}$, then NMPC scheme (21) is type A or type B first-order equivalent to the economic NMPC scheme (5), depending on the value of η_{F} [22, Theorem 3]. Note that Problem (21) is guaranteed to be asymptotically stable [10, 15], while for the economic scheme (5) only local stability is guaranteed [21].

III. TUNEMPC SOFTWARE

The goal of TuneMPC is to facilitate automated application of the tuning procedure outlined in the previous section. It provides an intuitive and user-friendly interface, with access to all software tuning parameters and intermediate results. The tool is written in Python 3 and builds on the following open-source software packages: the symbolic framework for nonlinear optimization CasADi [3], the nonlinear program (NLP) solver IPOPT [18], the QP solver qpOASES [8], the Python interface to conic optimization solvers PICOS [16] and the framework for fast embedded optimal control `acados` [17].

A. Input Parameters

There are three mandatory inputs to the tool, namely the dynamics $f_k(x,u)$ and cost functions $l_k(x,u)$, in the form of a list of CasADi `Function` objects `f` and `l` and the desired period p as parameter `p`. Note that it is straightforward to discretize continuous-time systems given in ODE/DAE form, using the built-in CasADi numerical `integrator` class.

Additionally, it is possible to specify constraint functions $h_k(x,u)$ in the original form (2) as `h`. Finally, since in general Problem (9) is a non-convex problem, the user can provide an initial guess of the primal and dual variables, given as `w0` and `lam0`.

B. Computation of Optimal Steady-state

After specifying all required and optional input parameters, one can formulate and solve the slacked Problem (9) with the following lines of code:

```
import tunempc
tuner = tunempc.Tuner(f, l, h, p)
tuner.solve_ocp(w0, lam0)
```

We automatically detect the nonlinear elements in the constraints $h_k(x,u)$, create the functions $\hat{h}_k(x,u)$ and $g_k(x,u)$ and formulate POCP (9) using CasADi symbolics.

In order to retrieve the exact active sets $[\mathbb{A}_k]^p$ for the Hessian convexification procedure, we solve the problem with an active-set based NLP solver. The default solver uses sequential quadratic programming (SQP) with QP solver qpOASES. However, the user can specify any active-set based NLP solver available through CasADi. Since the built-in active-set SQP algorithms in CasADi have limited reliability, by default we first solve the POCP with the interior-point NLP solver IPOPT and use its optimal solution as initial guess for the SQP solver.

In case $p > 1$, the solution of Problem (9) is not unique, due to the inherent phase invariance of periodic problems. Hence, the reduced Hessian contains a zero eigenvalue, which renders the corresponding LQ problem not strictly dissipative. This problem is tackled by re-solving the POCP while adding to the cost (9a) the term $\gamma\|x_0 - \bar{x}_0^{\mathrm{p}}\|^2$, $\gamma > 0$.

C. Hessian Convexification Procedure

Based on the solution of POCP (9), it is possible to retrieve the necessary sensitivity information of its rotated equivalent (11). SDP (20) is then formulated and solved following the procedure explained in Section II-D. The convexified Hessians $\tilde{H}_k$ are constructed as:

```
Hc = tuner.convexify(rho = 1.0, force = False)
```

Parameter `rho` denotes the tuning parameter ρ appearing in the cost function of the SDP. The `force` flag allows one to force convexification using step 3 in the procedure in case the system is not optimally operated at the optimal orbit.

D. Rapid MPC Prototyping

In order to facilitate rapid evaluation of the equivalence relation or closed-loop performance in practice, TuneMPC also constructs and solves the MPC problems (5) and (21):

```
EMPC = tuner.create_mpc('economic', N)
TUNEMPC = tuner.create_mpc('tuned', N)
u0 = TUNEMPC.step(x0)
```

The software takes care of all problem modifications, correctly initializes the primal and dual solution and deals with periodic reference and initial guess shifting.

The user can also code-generate highly efficient embeddable solvers for the aforementioned MPC schemes, using the software framework `acados`:

```
TUNEMPC.generate()
u0 = TUNEMPC.step_acados(x0)
```

The generated code can be directly compiled on the embedded platform, or it can be accessed through the TuneMPC interface for verification purposes, as showcased here.

IV. NUMERICAL EXAMPLE

In order to illustrate the potential of the tuning software, we simulate an airborne wind energy (AWE) system and provide the code in the tool repository. Airborne wind energy is a novel wind energy technology that reaches higher altitudes and the corresponding stronger and steadier winds, that cannot be reached by conventional wind turbines [6]. It does so by relying on one or more tethered aircraft that fly fast crosswind periodic orbits. In the drag-mode type AWE system considered here, on-board turbines convert aerodynamic power to electricity, which is transferred to a ground station through the tether.

A. System model

We consider here a single-aircraft AWE system and adopt the optimization-friendly model proposed in [19]. The aircraft is modeled in non-minimal coordinates as a point-mass, with states and controls

$$x := \left[q, \dot{q}, C_{\mathrm{L}}, \phi, \kappa\right]^{\top} \in \mathbb{R}^9 \quad \text{and} \quad u := \left[\dot{C}_{\mathrm{L}}, \dot{\phi}, \dot{\kappa}\right]^{\top} \in \mathbb{R}^3,$$

where $q := \left[q_{\mathrm{x}}, q_{\mathrm{y}}, q_{\mathrm{z}}\right]^{\top}$ and $\dot{q}$ are the aircraft position and velocity respectively. The lift coefficient C_{L} and roll angle ϕ model the control over the magnitude and direction of the aerodynamic forces. Coefficient κ determines the drag force applied by the on-board turbines to extract power.

The dynamics are an implicit index-1 DAE [19, Eq. 4] and contain two invariants which are stabilized using Baumgarte stabilization [9]. The discrete system dynamics $f(x, u)$ are obtained by numerical integration with sampling time T_{s}.

The economic cost function is the average aerodynamic power converted by the on-board turbines:

$$l(x, u) := \frac{1}{T_{\mathrm{s}}} \int_0^{T_{\mathrm{s}}} -\kappa(t)\|v(t)\|^3 + u(t)^{\top} R u(t) \; \mathrm{d}t \;, \quad (22)$$

with v the apparent wind speed of the aircraft [19]. The control penalization mitigates actuator fatigue.

The system inequality constraints $h(x, u)$ include upper and lower bounds $\zeta \in [\zeta_{\min}, \zeta_{\max}]$, $\zeta := \left[C_{\mathrm{L}}, \dot{C}_{\mathrm{L}}, \phi, \dot{\phi}, \dot{\kappa}\right]^{\top}$; bounds on the tether force $T(x) \in [0, T_{\max}]$, which must be positive but not exceed the tether yield strength; and limits on the aircraft acceleration $\|\ddot{q}(x)\|^2 \leq \ddot{q}_{\max} = 12\bar{g}$, with $\bar{g}$ the gravitational acceleration, to preserve the hardware integrity. The bound values $\zeta_{\min}, \zeta_{\max}, T_{\max}$ and functions $T(x)$ and $\ddot{q}(x)$ are defined in [19, Eq. 8-9].

B. Optimal periodic orbit

POCP (9) is formulated with $p = 40$. The initial guess w_0 is provided using the open-source AWE optimization toolbox AWEbox [5], which optimizes both state and control trajectories, as well as the orbit duration and system parameters such as tether length l_{t} and diameter d_{t}. For a wing span of $b = 70$ m, the optimal orbit duration is $T_{\mathrm{p}} = 7.86$ s, yielding sampling time $T_{\mathrm{s}} = T_{\mathrm{p}}/p = 197$ ms. The active sets $\mathbb{A}_k$ include the constraints related to $T_{\max}$ and $C_{\mathrm{L,max}}$, for all $k \in \mathbb{I}_0^{p-1}$, and the one related to $\ddot{q}_{\max}$ for $k \in \mathbb{I}_4^{31}$.

The convexified Hessians $\tilde{H}_k$ are computed with $\eta_{\mathrm{F}} = 1$ and $\eta_{\mathrm{T}} = 0$, yielding type B equivalence of the tuned tracking NMPC scheme with economic NMPC.

C. Open-loop scenario

The following open-loop scenario is considered. The aircraft is initialized at $\hat{x}_0 = \bar{x}_0^{\mathrm{p}}$ with an altitude deviation Δz, so that $\hat{q}_{\mathrm{z},0} = \bar{q}_{\mathrm{z},0}^{\mathrm{p}} + \Delta z$. The initial condition is adjusted as

$$\begin{aligned} \hat{q}_{\mathrm{x},0} &:= \sqrt{-\hat{q}_{\mathrm{z},0}^2 - \bar{q}_{\mathrm{y},0}^{\mathrm{p}\;2} + l_{\mathrm{t}}^2}, \\ \hat{\dot{q}}_{\mathrm{z},0} &:= -\left(\hat{q}_{\mathrm{x},0}\bar{\dot{q}}_{\mathrm{x},0}^{\mathrm{p}} + \bar{q}_{\mathrm{y},0}^{\mathrm{p}}\bar{\dot{q}}_{\mathrm{y},0}^{\mathrm{p}}\right)/\hat{q}_{\mathrm{z},0}, \end{aligned}$$

so that the system consistency conditions [19, Eq. 5] are met.

The open-loop response to this state deviation is then compared for different NMPC controllers. To simplify the discussion of the results, all NMPC formulations use a terminal point constraint and prediction horizon $N = 20$. For this horizon length, the closed-loop response is extremely close to the open-loop prediction.

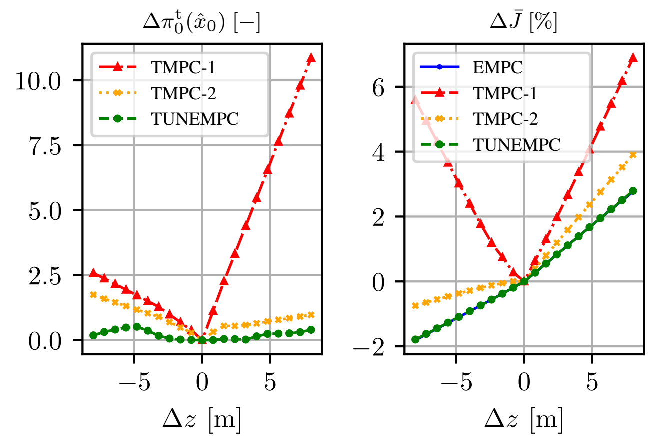

Fig. 1: Open-loop tracking feedback policy deviation with respect to economic NMPC: $\Delta\pi_0^{\mathrm{t}}(\hat{x}_0) := \|\pi_0^{\mathrm{t}}(\hat{x}_0) - \pi_0^{\mathrm{e}}(\hat{x}_0)\|$ (left) and open-loop relative transient cost $\Delta\bar{J}$ (right).

We compare baseline economic NMPC (EMPC) (5); tuned tracking NMPC (TUNEMPC) (21); and a standard tracking scheme (TMPC-1), where the tuned Hessians in the cost (21a) are replaced by $H_k^{\mathrm{t},1} := \mathrm{blkdiag}(I_{12\times 12}, 0_{3\times 3})$. The final zeros on the diagonal correspond to the slack variables used to reformulate the nonlinear inequalities. We also consider a tracking scheme (TMPC-2) that is manually tuned for economic performance. The stage cost matrices $H_k^{\mathrm{t},2} := \mathrm{diag}([0.1 \cdot 1_{1\times 3}, 1_{1\times 3}, 1000, 1, 100, 1_{1\times 3}, 0_{1\times 3}])$. From experience it is known that higher penalties on the tracking error of C_{L}, κ and $\dot{q}$ and a lower penalty on position tracking error leads to more power-optimal feedback policies. The performance of the MPC schemes can be measured with the relative transient cost:

$$\Delta\bar{J} := \frac{\sum_{k=0}^{N} l(x_k, u_k) - l(\bar{x}_k^{\mathrm{p}}, \bar{u}_k^{\mathrm{p}})}{\sum_{k=0}^{N} l(\bar{x}_k^{\mathrm{p}}, \bar{u}_k^{\mathrm{p}})} 100 \;. \quad (23)$$

The left graph in Fig. 1 illustrates the first-order equivalence relation (6) between TUNEMPC and EMPC. Since the TUNEMPC feedback policy is a linear approximation of the EMPC feedback policy around the optimal orbit, it holds that $\nabla_{\hat{x}_0}\Delta\pi_0^{\mathrm{t}}(0) = 0$, with $\Delta\pi_0^{\mathrm{t}}(\hat{x}_0) := \|\pi_0^{\mathrm{t}}(\hat{x}_0) - \pi_0^{\mathrm{e}}(\hat{x}_0)\|$. As expected, the TMPC-2 feedback policy approximates that of EMPC better than that of TMPC-1. Nevertheless, neither of these TMPC schemes display the same first-order equivalence as TUNEMPC.

The right graph in Fig. 1 shows that the transient cost of TMPC-1 increases quickly for a growing initial perturbation. TMPC-2 performs considerably better though there is no guarantee that this holds for all possible state perturbation directions. The transient cost of TUNEMPC almost coincides

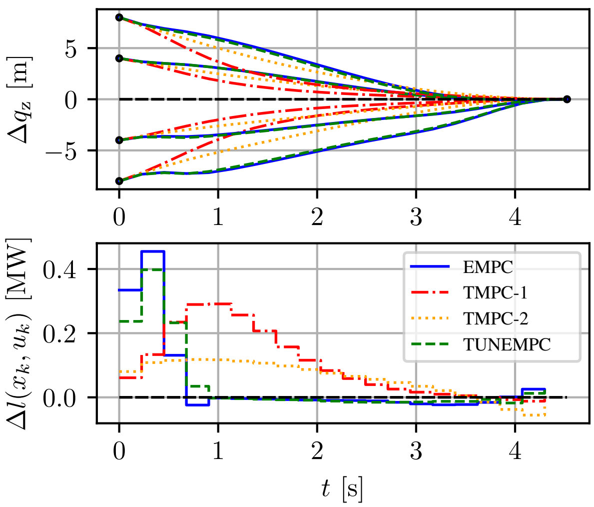

Fig. 2: Vertical trajectory deviation $\Delta q_z := q_z - \bar{q}_z^p$ for different Δz (top), stage cost deviation $\Delta l(x_k, u_k) := l(x_k, u_k) - l(\bar{x}_k^p, \bar{u}_k^p)$ for $\Delta z = 8$ m (bottom).

with the EMPC transient cost over the entire disturbance range. Starting from an initial altitude perturbation of several meters, the predicted power output over half a power cycle is increased by an order of magnitude of percentage points relative to the TMPC schemes.

The top graph in Fig 2 shows how TMPC-1 and to a lesser degree TMPC-2 return to the optimal trajectory considerably faster than the EMPC and TUNEMPC. The latter schemes, however, exploit the information on how the power output depends on the aircraft state: while in the beginning of the prediction horizon the stage cost is higher, this allows obtaining a lower cost in later stages - as shown in the bottom graph in Fig 2 - hence outperforming both TMPC schemes.

The experiments were carried out using the efficient code-generated `acados`-solvers on an Intel Core i7 2.5 Ghz, 16GB RAM. The average time per SQP-iteration for EMPC was 213 ms, whereas for the tracking schemes it was 13 ms. Note that in this example, EMPC is not real-time feasible, whereas the average TUNEMPC timing is well below the proposed sampling time.

V. CONCLUSION

In this paper we presented TuneMPC, a novel open-source software package for economic tuning of linear and nonlinear tracking MPC problems. The tool automatically computes optimal steady states or periodic trajectories for constrained nonlinear systems with associated economic objective. It then returns corresponding positive-definite stage cost matrices for a tracking (N)MPC problem that delivers a feedback law that is first-order equivalent to economic NMPC.

We demonstrated the equivalence relation in simulations for a challenging periodically operated system. We showed that in the considered example significant performance gains with respect to standard tracking MPC can be expected in a large neighborhood around the optimal trajectory.

Future work will further investigate possible improvements in the computational efficiency and problem formulation; the use of the convexified cost to construct a positive definite Hessian approximation for efficient EMPC; further investigating the trade-off between performance and computational times, in particular for AWE systems.

REFERENCES

[1] TuneMPC. `https://github.com/jdeschut/tunempc`, 2020.

[2] R. Amrit, J. Rawlings, and D. Angeli. Economic optimization using model predictive control with a terminal cost. *Annual Reviews in Control*, 35(2):178–186, 2011.

[3] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl. CasADi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 2018.

[4] D. Angeli, R. Amrit, and J. Rawlings. On Average Performance and Stability of Economic Model Predictive Control. *IEEE Transactions on Automatic Control*, 57:1615–1626, 2012.

[5] J. De Schutter, T. Bronnenmeyer, R. Leuthold, and M.Diehl. `awebox`: Modelling and optimal control of single- and multiple-kite systems for airborne wind energy. `https://github.com/awebox`, 2019.

[6] M. Diehl. Airborne wind energy: Basic concepts and physical foundations. In *Airborne Wind Energy*, pages 3–22. Springer Berlin Heidelberg, 2013.

[7] M. Diehl, R. Findeisen, F. Allgöwer, H. G. Bock, and J. P. Schlöder. Nominal stability of the real-time iteration scheme for nonlinear model predictive control. *IEE Proc.-Control Theory Appl.*, 152(3):296–308, 2005.

[8] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl. qpOASES: a parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, 6(4):327–363, 2014.

[9] S. Gros and M. Zanon. Numerical optimal control with periodicity constraints in the presence of invariants. *IEEE Transactions on Automatic Control*, 63(9):2818–2832, 2018.

[10] L. Grüne and J. Pannek. *Nonlinear Model Predictive Control*. Springer-Verlag, second edition edition, 2017.

[11] L. Grüne and M. Stieler. Asymptotic stability and transient optimality of economic mpc without terminal conditions. *Journal of Process Control*, 24(8), 2014.

[12] M. Heidarinejad, J. Liu, and P. Christofides. Economic model predictive control of nonlinear process systems using lyapunov techniques. *AIChE Journal*, 58(3):855–870, 2012.

[13] R. Hult, M. Zanon, S. Gros, and P. Falcone. Energy-optimal coordination of autonomous vehicles at intersections. In *Proceedings of the European Control Conference (ECC)*, 2018.

[14] M. Müller and F. Allgöwer. Economic and distributed model predictive control: Recent developments in optimization-based control. *SICE Journal of Control, Measurement, and System Integration*, 10(2):39–52, 2017.

[15] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl. *Model Predictive Control: Theory, Computation, and Design*. Nob Hill, 2nd edition edition, 2017.

[16] G. Sagnol. Picos: A python interface for conic optimization solvers. `https://picos-api.gitlab.io/picos/`, 2020.

[17] R. Verschueren, G. Frison, D. Kouzoupis, N. van Duijkeren, A. Zanelli, B. Novoselnik, J. Frey, T. Albin, R. Quirynen, and M. Diehl. acados: a modular open-source framework for fast embedded optimal control. *arXiv preprint*, 2019.

[18] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.

[19] M. Zanon, S. Gros, J. Andersson, and M. Diehl. Airborne wind energy based on dual airfoils. *IEEE Transactions on Control Systems Technology*, 21:1215–1222, July 2013.

[20] M. Zanon, S. Gros, and M. Diehl. Indefinite linear MPC and approximated economic MPC for nonlinear systems. *Journal of Process Control*, 24:1273–1281, 2014.

[21] M. Zanon, S. Gros, and M. Diehl. A tracking mpc formulation that is locally equivalent to economic mpc. *Journal of Process Control*, 45:30–42, 2016.

[22] M. Zanon, S. Gros, and M. Diehl. A periodic tracking mpc that is locally equivalent to periodic economic mpc. In *Proceedings of the IFAC World Congress*, 2017.