A penalty function approach to constrained Pontryagin-based Nonlinear Model Predictive Control

A penalty function approach to constrained Pontryagin-based Nonlinear Model Predictive Control

Michele Pagone, Mattia Boggio, Carlo Novara, Anton Proskurnikov, Giuseppe C. Calafiore

***Abstract*— A Pontryagin-based approach to solve a class of constrained Nonlinear Model Predictive Control problems is proposed, which employs the method of penalty functions for dealing with the state constraints. Unlike the existing works in literature, the proposed method is able to cope with nonlinear input and state constraints without any significant modification of the optimization algorithm. Theoretical results are tested and confirmed by numerical simulations on the Lotka-Volterra prey/predator nonlinear system.**

I. INTRODUCTION

Over the last years, Model Predictive Control (MPC) has been accepted as a powerful control tool for a wide range of technological applications [1], [2], thanks to its capability to design control algorithms for multivariable systems under state, input, and output constraints. The resulting controllers also provide optimality of a predefined performance index.

The key point of the MPC design is the method for addressing optimal control problems (OCP) within a receding horizon strategy. To cope with nonlinear dynamics and constraints, as well as with non-convex performance indexes, Nonlinear MPC (NMPC) has been introduced (see, e.g. [3] and references therein). To find the global optimum in this situation is difficult, optimization algorithms are computationally intensive and, in general, the solution rarely admits an explicit closed-form representation [4], [5].

In this paper, we propose a solution that is based on Pontryagin's Minimum Principle (PMP)[6]: under some assumptions on the Hamiltonian function, we can obtain an explicit control law - a function of the state and the co-state - even if the system dynamics and/or constraints are nonlinear. The price paid for this is the necessity to solve a Two-Points Boundary Value Problem (TPBVP) in order to find the state and co-state functions. The first applications of the PMP to receding horizon control date back to works by [7], [8] and [9] who have also established important higher degree optimality conditions based on the theory of Lie algebras.

Although TPBVP problems usually cannot be solved analytically, a number of efficient numerical algorithms to solve OCP in real time have been proposed [10] such as, e.g., the stabilized continuation method [7] and its accelerated versions [11], the Newton-type algorithm [12] and the extended modal series method, approximating OCP with nonlinear constraints by standard LQR problems [13]. An efficient active set method of solving discrete-time PMP equations arising in MPC problems with input and terminal state constraints was developed in [14]. Continuous-time OCPs can be accurately approximated by discrete-time ones as demonstrated by the recent work [15].

Whereas initial and terminal state constraints can be accommodated by existing PMP-based MPC algorithms, direct application of PMP becomes problematic in the situation where the state vector is constrained at any time [3], [16]. In this situation, the differential equations of PMP are different for constrained and unconstrained trajectories: unconstrained and constrained pieces of the trajectory are 'tailored' by imposing additional interior tangency conditions at the junctions points [17], [18]. This substantially complicates the solution of TPBVP in real time except for the situations where the optimal solution structure is known a priori.

An alternative way to cope with state or mixed input-state constraints is based on the use of barrier functions that arise as penalty terms in the objective function. A general methodology to get rid of relaxing both state and input constraints by introducing penalty terms has been proposed in [19] under the assumption that the nonlinear system has a well-defined relative degree. A similar approach has been proposed for a special type of constraints in [20]. In this paper, we further elaborate the approach proposed in the example from [8], where the state constraint is replaced by an appropriate penalty term in the cost functional, without significantly modifying the algorithm of solving OCP compared to the unconstrained case. Unlike [19], [20], input constraints does not need to be relaxed and can be tackled by the standard PMP.

The penalty function method proposed in this paper is concerned with defining a methodology for accounting the state constraints within the TPBVP, without affecting the differential equation solution feasibility. This latter aspect was widely discussed by [21], which pointed out that, when employing the classical log-barrier function, some TPBVP numerical singularities can arise. A similar approach can be found in [22]: a Lagrangian-barrier function based method which adds the state constraints as a logarithmic term to the objective function. As remarked in [16], the penalty functions methods can be divided into two different classes: exterior and interior. We focus on the interior penalty methods since they are more likely to generate feasible solutions. This can be an interesting particularity in numerous nonlinear and non-convex applications: satisfaction of constraints is more important than optimality (see also [23] and the reference therein).

We propose a class of Gaussian-like penalty function.

The authors are with the Department of Electronics and Telecommunications, Corso Duca degli Abruzzi, 24, 10129 Torino, Italy. {michele.pagone, mattia.boggio, carlo.novara, anton.proskurnikov, giuseppe.calafiore}@polito.it

Thanks to this approach, the solution to the system of differential equations, in general, does not present numerical singularities in solution. An important advantage of the proposed penalty methodology relies in the relaxation on the constraints and penalty function assumptions, in particular, the penalty function has to be only $\mathscr{C}^1$-smooth unlike the approach from [19].

To sum up, the proposed NMPC framework shows the following advantages: i) Unlike numerical methods, where a discretization of state, input and constraints before optimization is required [24], the PMP-based solution does not need the input parametrization anymore, resulting in a better accuracy in tracking the reference; ii) the PMP-based NMPC seems to perform a more efficient trade-off between computational complexity and final reference tracking with respect to the direct methods, being suitable for on-line applications.

The paper is organized as follows. In Section II the NMPC scheme and its unconstrained Pontryagin-based solution are illustrated. The PMP-based solution of the constrained problem is shown in Section III. A simulated example is presented in Section IV. Finally, the conclusions are drawn in Section V.

II. NMPC Framework

Consider the following affine-in-the-input nonlinear system:

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

where $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$ are the state and the input, respectively. We assume that the state of system (1) is measured in real time, with a sampling time T_S. At each time $t = t_k$, a prediction $\hat{x}$ of the system state and output over the time interval $[t, t+T_p]$ is performed, where $T_p \geq T_S$ is the prediction horizon. The prediction is obtained by integrating (1). At each time $t = t_k$, we look for an input signal $u^*(t : t+T_P)$, minimizing a suitable cost function $J\big(u(t : t+T_p)\big)$ subject to possible constraints that may occur during the system's operations. Our goal is to track a reference signal $x_r \in \mathbb{R}^{n_x}$. The considered NMPC cost function $J\big(u(t : t+T_p)\big)$ in the Bolza form is

$$\begin{aligned} J = &\int_t^{t+T_p} \tilde{x}_p^T(\tau)\mathbf{Q}\tilde{x}_p(\tau)\ \mathrm{d}\tau\ + \\ &+\int_t^{t+T_p} u^T(\tau)\mathbf{R}u(\tau)\ \mathrm{d}\tau\ +\tilde{x}_p^T(t+T_p)\mathbf{P}\tilde{x}_p(t+T_p) \end{aligned} \tag{2}$$

where $\tilde{x}_p = x_r - \hat{x}$ is the predicted tracking error. Moreover, $\mathbf{Q} = \mathbf{Q}^T \geq 0$, $\mathbf{P} = \mathbf{P}^T \geq 0$, and $\mathbf{R} = \mathbf{R}^T > 0$ are diagonal matrices. Trivially, $\mathbf{Q}, \mathbf{P} \in \mathbb{R}^{n_x \times n_x}$ and $\mathbf{R} \in \mathbb{R}^{n_u \times n_u}$. Mathematically, at each time $t = t_k$, the following optimization problem is solved:

$$\begin{aligned} &u^*(t : t+T_p) = \arg\min_{u(\cdot)} J(u(t : t+T_p)) \\ &\text{subject to:} \\ &\dot{\hat{x}}(\tau) = f\big(\hat{x}(\tau)\big) + g\big(\hat{x}(\tau)\big)u(\tau), \quad \hat{x}(t) = x(t) \\ &\hat{x}(\tau) \in X_C,\ u(\tau) \in U_C, \quad \forall \tau \in [t : t+T_p], \\ &u(\tau) \in \mathscr{KC}\big([t, t+T_p]\big). \end{aligned} \tag{3}$$

X_C and U_C are sets describing possible constraints on the state and input, respectively and $\mathscr{KC}([t, t+T_p])$ is the space of piece-wise continuous functions. A receding control horizon strategy is employed: at a given time $t = t_k$, the input signal $u^*(t_k : t_k + T_p)$ is computed by solving (3). Then, only the first optimal input value $u(t) = u^*(t_k)$ is applied to the plant, keeping it constant for $t \in [t_k, t_{k+1}]$. The remainder of the solution is discarded. Then, the complete procedure is repeated at the next time steps $t = t_{k+1}, t_{k+2}, \ldots$

Assumption 1: Let $f \in \mathscr{C}^1(\mathbb{R}^{n_x} \to \mathbb{R}^{n_x})$ and $g \in \mathscr{C}^1(\mathbb{R}^{n_x} \to \mathbb{R}^{n_x})$.

Assumption 2: The admissible control set $U_C \subseteq \mathbb{R}^{n_u}$ is $U_C = \{u \in \mathbb{R}^{n_u} : u_{i_{min}} \leq u_i \leq u_{i_{max}}\}$, $i = 1, \ldots, n_u$.

Assumption 3: The state constraint set is $X_C = \{x \in \mathbb{R}^{n_x} : C(x) \leq 0\}$. Here, $C(x) \in \mathscr{C}^1(\mathbb{R}^{n_x} \to \mathbb{R})$ is, generally, a non-convex function.

Remark 1: The optimization problem (3) is numerically hard to tackle, since u is a continuous-time signal and thus the number of decision variables is infinite. The direct solution of the OCP requires a finite parametrization of the input signal u (see, e.g., [24]). For example, as illustrated in Section IV, a piece-wise constant parametrization can be assumed, with changes of value at the nodes $\tau_1, \ldots, \tau_N \in [t, t+T_p]$ with N the number of nodes. The choice of $N > 1$ can lead to satisfactory performances behaviors, but at cost of computational complexity increment. One can pick $N = 1$ (corresponding to a constant input for every $\tau \in [t, t+T_P]$) in order to reduce the computational complexity of the optimization algorithm. Nevertheless, this approach could not always guarantee an acceptable level of performance. This issue is mitigated when using the PMP approach presented in the manuscript which does not require any a-priori prarametrization of the control signal. This latter does not significantly effect the algorithm computational complexity.

A. Unconstrained Pontryagin-based NMPC Solution

We neglect for the moment possible constraints on the state and the input, focusing on the case where $X_C \equiv \mathbb{R}^{n_x}$ and $U_C \equiv \mathbb{R}^{n_u}$.

According to [6], a necessary condition for a trajectory $x(t)$ to be the extremal path and the corresponding control $u(t)$ to be the optimal input, is that the Hamiltonian scalar function $H(x(t), u(t), \lambda(t)) \in \mathscr{C}^k(\mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_x} \to \mathbb{R})$ attains its minimum value when $u = u^*$ and while satisfying the differential equations in (1), the time evolution of the Lagrangian multipliers $\lambda \in \mathbb{R}^{n_x}$) (or co-state variables), and a set of boundary conditions (B.C.). The Hamiltonian is defined as

$$H = \tilde{x}_p^T \mathbf{Q} \tilde{x}_p + u^T \mathbf{R} u + \lambda^T \big(f + gu\big). \tag{4}$$

The necessary conditions for optimality can be derived by analyzing the first-order variation of the augmented expression of (2). The rigorous mathematical formulation of the first-order variation can be found in [25]. Whereby, the Pontryagin formulation of the NMPC optimal control problem:

$$\begin{aligned}
&(x^*, u^*, \lambda^*) = \arg\min_{u(\cdot)} H \\
&\text{subject to:} \\
&\dot{x} = f + gu \\
&\dot{\lambda} = -\nabla_x H^T \\
&x_k - x(t_k) = 0 \\
&\lambda^T(t_k) = -\mu^T \\
&\lambda^T(t+T_p) = 2\mathbf{P}\tilde{x}_p(t+T_p).
\end{aligned} \tag{5}$$

From (5), we can note that the optimization problem is subject to both the state dynamics in (1), and the dynamic of the co-state variables λ, described by the so-called Euler-Lagrange differential equations. Both the state and co-state evolution must satisfy a set of boundary conditions to be imposed at the borders of the prediction horizon. The B.C. have to be satisfied by λ and x during the system evolution along the extremal path. At each time $t = t_k$, the state value cannot be chosen arbitrarily: the continuity between two successive sampling steps must be ensured, so that $x_k = x(t_k)$. In (5), at $t = t_k$, $\lambda(t_k) = -\mu$ where μ is the Lagrangian multipliers vector corresponding to the state continuity constraint.

The Euler-Lagrange equations - describing the λ time evolution - take the form of:

$$\dot{\lambda} = -\big(\lambda^T \nabla_x \big(f(x) + g(x)u\big) - 2\mathbf{Q}\tilde{x}_p\big)^T. \tag{6}$$

The optimal control law is obtained by minimizing the Hamiltonian with respect to u. From (4), we have

$$u^* = -\frac{1}{2}\mathbf{R}^{-1}\big(\lambda^T g(x)\big). \tag{7}$$

Accounting the PMP-based NMPC solution in (5), together with the optimal control law in (7), it is clear how the optimal control problem in (5) turns into a two-points boundary value problem. Indeed, the equations (1) together with (6) and the B.C. in (5) represents a TPBVP to be solved over the prediction horizon $[t, t+T_p]$. The TPBVP solution provides the λ and the x of the explicit control laws (7).

The TPBVP is formalized as follows:

$$\begin{aligned}
\dot{x} &= f + gu \\
\dot{\lambda} &= -\nabla_x H^T \\
x_k - x(t_k) &= 0 \\
\lambda^T(t_k + T_p) &= 2\mathbf{Q}\tilde{x}_p(t_k + T_p)
\end{aligned} \tag{8}$$

Remark 2: Observing the optimal control law (7), the input $u^*(\tau)$ depends on $\lambda(\tau)$ and $x(\tau)$, whose values change at each sampling step of the TPBVP over the prediction horizon. For this reason, the PMP-based NMPC solutions does not require an a-priori parametrization of the input signal. This is a very interesting results since the OCP algorithm achieves high performances without increasing the computational complexity independently from the input parametrization.

Remark 3: A preliminary proof on the finite-time stability of the closed loop is proposed by the authors in [26]. This study is mainly devoted in prooving the finite-time practical stability of the closed loop when the optimal control law is in feedback, by employing an innovative Lyapunov-like function based on the predicted values of the state.

III. Indirect Solution of the Constrained OCP

In general, the constrained case can be handled by means the indirect optimization problem only when the optimization is performed off-line, by augmenting the system with additional variables [25], [27]. Nevertheless, when dealing with an on-line optimization process, this aspect can be tough, since it is necessary to iterate the solution in order to identify the control arcs where the constraints are active and imposing additional B.C. at the junction points.

A. Input Constraints

We consider that the input is bounded linearly, such that $U_C = \{u(t) \in \mathbb{R}^{n_u} : \; u_{i_{min}} \leq u_i(t) \leq u_{i_{max}}, \; \forall t\}$. Consider the optimal control law (7), for the nonlinear system (1), the optimal control $u^* \in U_C$ is:

$$u^* = \text{sat}_{U_C}\left(-\frac{1}{2}\mathbf{R}^{-1}\big(\lambda^T g(x)\big)\right) \tag{9}$$

where the sat$(\cdot)$ represents the saturation operator and it applies element-wise to the input vector. In formulae, the i^{th} control component is:

$$u_i^* = \begin{cases} u_{i_{min}}, & \text{if } -\dfrac{\lambda_i g_i(x)}{2r_i} \leq u_{i_{min}} \\ u_{i_{max}}, & \text{if } -\dfrac{\lambda_i g_i(x)}{2r_i} \geq u_{i_{max}} \\ -\dfrac{\lambda_i g_i(x)}{2r_i}, & \text{otherwise} \end{cases} \tag{10}$$

where r_i is the i^{th} entry of the $\mathbf{R}$ diagonal.

Proposition 1: For the nonlinear system (1) with performance index (2), if $u \in U_C$, the constrained optimal command is given by (10).

Proof: From the optimal control equation we have $u^* = \arg\min_{u \in U_C} H$. For the problem at hand, since $\nabla_u \lambda^T f(x) = 0$, we can neglect the terms not depending on the control in the Hamiltonian. Then, picking only the control-depending terms of the Hamiltonian and recalling that $\mathbf{R}$ is a diagonal positive matrix:

$$u^* = \arg\min_{u \in U_C}\left[\sum_{i=1}^{n_u} r_i u_i^2 + \sum_{i=1}^{n_x} \lambda_i g_i(x) u_i\right]. \tag{11}$$

Since there are no coupled control terms, the optimal control equation can be solved by minimizing the Hamiltonian element-wise. This is straightforward, since, in this configuration, the Hamiltonian consists of an elliptic paraboloid whose main axes are parallel to the Cartesian axes. Consider the unconstrained case. Since the Hamiltonian is convex with respect to u we have that $H(u) \geq H(u^*) + \nabla_u H(u^*)^T(u - u^*)$, i.e. all the admissible values of the input are enclosed in one of the halfspaces $\mathscr{H}_{++}$ delimited by the hyperplane tangent at H in u^*. Denote, now, the constrained optimal input with u_c^*, we have that $u_c^* \in \mathscr{H}_{++}$ and $H(u^*) < H(u_c^*)$. Since the Hamiltonian is monotone with respect to the input, $H(u) \geq$

$H(u_c^*)+\nabla_u H(u_c^*)^T(u-u_c^*) \geq H(u^*)+\nabla_u H(u^*)^T(u-u^*)$, i.e. there are not any values of u which improve the Hamiltonian performance index. Hence (10) is an optimum for the input constrained problem. ■

B. Path Constraints

In order to incorporate the path constraints within the OCP, we define an augmented cost function $\tilde{J}$ such that, when the state approaches the boundary of the forbidden set, its value becomes significantly larger than J, $\lim_{C(x,t)\to 0}\tilde{J} \gg J$. Therefore, we augment the cost function by choosing a suitable penalty function $k(x)$ which prevents the states approach the boundary of the constrained set whilst its value is (almost) null when far from the boundaries. This is a well known methodology to deal with the path constraints [28].

Assumption 4: Assume the penalty function $k(x) \in \mathscr{C}^1(\mathbb{R}^{n_x} \to \mathbb{R}^{n_x})$.

The augmented cost index is given by

$$\tilde{J}(u(\tau)) = J(u(\tau)) + \int_t^{t+T_P} \sum_{i=1}^{n} k_i(x)\ \mathrm{d}\tau. \tag{12}$$

where n is the number of the state constrains. The augmented Hamiltonian is

$$\tilde{H}(x,u,\lambda) = H(x,u,\lambda) + \sum_{i=1}^{n} k_i(x). \tag{13}$$

With the slight modification of the NMPC performance index and the consequent Hamiltonian augmentation, the contribution of the penalty function will affect the Euler-Lagrange equations by adding the terms of $\nabla_x \sum_{i=1}^{n} k_i(x)$. In a more general form $\dot{\lambda} = -\nabla_x\big(H + \sum_{i=1}^{n} k_i(x)\big)$.

IV. SIMULATED EXAMPLES

Consider the predatory-prey Lotka-Volterra model, described by a couple of first-order nonlinear differential equations with an exogeneous input applied on both states:

$$\begin{cases} \dot{x}_1 = x_1(\alpha - \beta x_2) + x_1 u_1 \\ \dot{x}_2 = x_2(\gamma x_1 - \delta) + x_2 u_2 \end{cases} \tag{14}$$

where x_1 and x_2 are the prey and predator population respectively and u_1 and u_2 the corresponding input components. Let $\alpha = 0.25$, $\beta = 0.25$, $\gamma = 0.008$, and $\delta = 0.008$ be parameters describing the interaction between the two species [29]. The admissible input set is described by $U_C = \{u(t) \in \mathbb{R} :\ -u_{i_{max}} \leq u_i(t) \leq u_{i_{max}},\ \forall t\}$, where $u_{1_{max}} = 10$ and $u_{2_{max}} = 5$. Concerning the state constraints, a nonlinear function prevents the predator specie grows too abruptly with respect to the prey specie, then, avoiding the extinction of both species when the prey population goes to zero. Hence, $X_C = \{x(t) \in \mathbb{R}^2 : 5 - \big((x_1-100)^2 + (x_2-51.5)^2\big)^{1/2} \leq 0,\ \forall t\}$. Thus, the state constraints are handled employing a Gaussian-like penalty function $k(x) = a\exp\left(-bC(x)^2\right)$ with $C = 5 - \big((x_1-100)^2 + (x_2-51.5)^2\big)^{1/2}$, $a = 10^6$, and $b = 1$. a and b have been tuned through a trial and error procedure.

Remark 4: Note that, the penalty $k(x)$ reach the maximum value when $C(x) = 0$ and then it goes to zero when $C(x) > 0$ (i.e., the constraint is violated). This choice is a consequence of the use of a Gaussian-like function as penalty. It partially prevents possible numerical singularities in TPBVP solution but it does not guarantee a strict fulfillment of the constraint. This issue can be mitigated by a proper choice of the penalty parameters a and b, tuned by simulations.

TABLE I

NMPC PARAMETERS

T_S	T_p	**R**	**Q**	**P**
1	10	$500\cdot\mathbf{I}_{2\times 2}$	$diag(10,35)$	$diag(10,35)$

Hence, the augmented Hamiltonian is

$$\tilde{H} = \lambda_1\big(x_1(\alpha-\beta x_2)+x_1u_1\big) + \lambda_2\big(x_2(\gamma x_1-\delta)+x_2u_2\big) + \\ + \sum_i^{n_u} r_i u_i^2 + \sum_i^{n_x} q_i \tilde{x}_{p_i}^2 + k(x) \tag{15}$$

where q_i is the i^{th} entry of the **Q** diagonal and $\tilde{x}_{p_i}, i = 1,2$ is the predicted tracking error. Then, TPBVP is formalized as:

$$\begin{cases} \dot{x}_1 &= x_1(\alpha-\beta x_2)+x_1u_1 \\ \dot{x}_2 &= x_2(\gamma x_1-\delta)+x_2u_2 \\ \dot{\lambda}_1 &= -\alpha\lambda_2 + \beta\lambda_1\lambda_2 - \lambda_1 u_1 - \gamma\lambda_2 x_2 - 2q_1\tilde{x}_{p_1} - \dfrac{\partial k(x)}{\partial x_1} \\ \dot{\lambda}_2 &= \beta\lambda_1 x_1 - \gamma\lambda_2 x_1 + \delta\lambda_2 - \lambda_2 u_2 - 2q_2\tilde{x}_{p_2} - \dfrac{\partial k(x)}{\partial x_2} \\ x_i &= x(t_k) \\ \lambda(t_f) &= \big(2\mathbf{P}\tilde{x}_p(t_f)\big)^T \end{cases} \tag{16}$$

The solution of the TPBVP in (16) provides the λ for the explicit optimal control law:

$$u^* = -\frac{1}{2}\mathbf{R}^{-1}\left(\lambda^T x\right) \tag{17}$$

The NMPC parameters are listed in Table I. The desired state is a limit cycle. Indeed, $[x_{r_1}(t), x_{r_2}(t)]^T = [10\cos(t)+100, 10\sin(t)+50]^T$. The initial state is $x_0 = [40,40]^T$. This means that prey and predator populations are much larger than zero, that is the two species are both far from the risk of extinction. Throughout the simulations, T_S and T_p are dimensionless and they are meant as iteration steps.

In Figure 1, the phase-plane curve of the predator-prey populations is shown. In particular, it is highlighted how the NMPC approach is perfecty able to fulfill the input and state constraints, without affecting the tracking performance. Note that, by a proper tuning of the NMPC parameters, the state trajectory is 'forced' to reach the reference by avoiding the constraint from above. This choice is aimed to avoid a possible extinction of both species when the prey population goes to zero. Figure 2 displays the time evolution of populations x_1 and x_2, and the corresponding tracking errors e_1 and e_2. This latter have a very fast convergence to zero, proving the effectiveness of the optimization algorithm. Finally, in Figure 3, the command activity is reported.

We are now interested in comparing the behavior of the solutions when employing different optimization strategies in

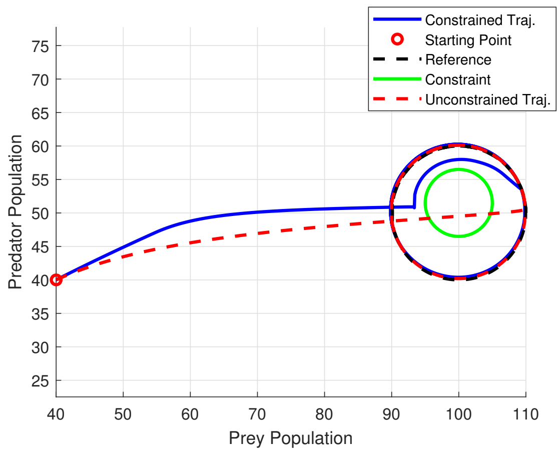

Fig. 1. Predator-prey populations phase-plane

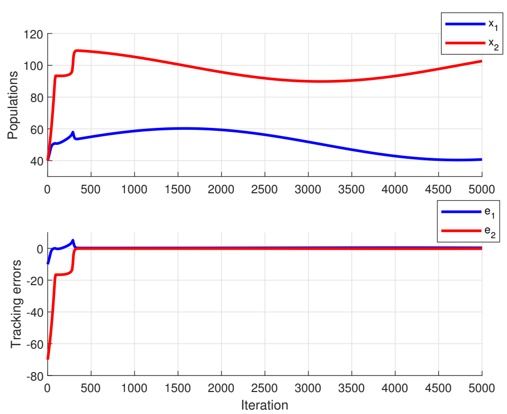

Fig. 2. Temporal evolution of populations and corresponding tracking errors

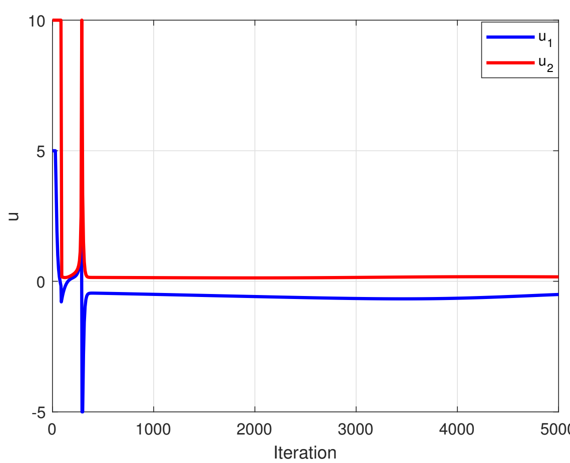

Fig. 3. Control components

an unconstrained scenario: the PMP-based and the Sequential Quadratic Programming (SQP) solutions. Concerning the SQP case, we further considered two different cases: i) constant input parametrization (NMPC-1), ii) piece-wise constant input parametrization with $N = 10$ (NMPC-10). In the latter case, the input is parametrized with the same sampling steps adopted in the PMP-based solution. Note that, the TPBVP has been solved by employing the Matlab function *bvp5c*. Figure 4 reports the results obtained in the unconstrained case, both for SQP and PMP. The resulting trajectories are slightly different. However, in all configurations, the NMPC is able to get a good tracking of the reference.

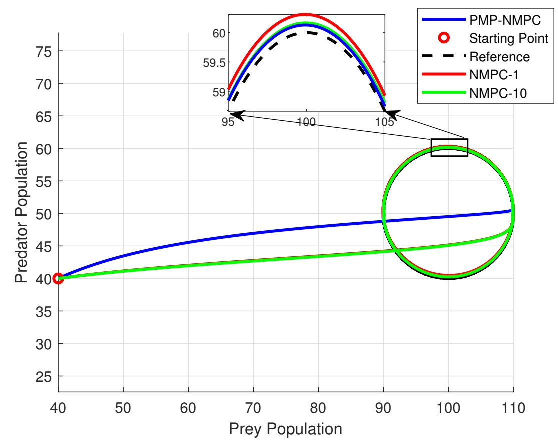

Fig. 4. PMP-NMPC and Numerical NMPCs phase-plane

From the computational burden point of view, Figure 5 presents a comparison between the solutions. If considering a similar input parametrization, the PMP-NMPC shows superior computational performances with respect to the SQP-NMPC-10. Moreover, also when considering the constant input parametrization, the PMP-NMPC owns slight better performances - together with a better reference tracking - with respect to the SQP solution.

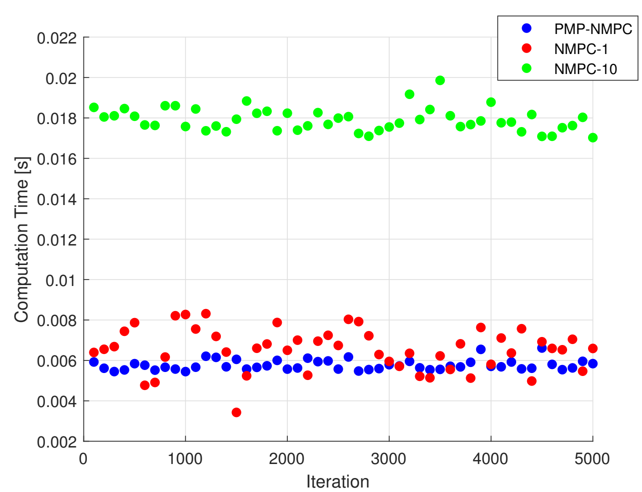

Fig. 5. Computational Cost analysis

Then, we highlight that the advantages of the proposed

PMP-based NMPC framework are: i) a better reference tracking than the NMPC-10 configuration, ii) a similar computational cost with respect to the NMPC-1 configuration.

V. CONCLUSIONS

We proposed an alternative approach for the Nonlinear Model Predictive Control optimization problem. We obtained a control law by developing an algorithm based on the Pontryagin Minimum Principle, turning the optimal control problem into a two-points boundary value problem. The resulting optimal input is function of the state and co-state variables, whose time evolution is described by the Euler-Lagrange differential equation. Hence, the optimal control law was obtained analytically by minimizing the Hamiltonian of the system. Moreover, we also coped with state constraints by exploiting a suitable penalty function within the cost function, without any modification of the optimization algorithm. The proposed methodology was then applied to the Lokta-Volterra nonlinear dynamics. The results highlighted the effectiveness of the control algorithm, showing excellent reference tracking and the compliance with the input and path constraints. Throughout the text, we assumed that all optimization problems are feasible (the solution exists and it is unique). Obtaining conditions of recursive feasibility is a topic of ongoing research.

REFERENCES

[1] J. Richalet, A. Rault, J.L. Testud, J. Papon, Model predictive heuristic control-application to industrial processes, Automatica, Vol. 14, pp. 413-428, 1978.

[2] S.J. Qin, T.A. Badgwell, An Overview of Nonlinear Model Predictive Control Applications, Progress in System and Control Theory, Vol. 26, pp. 3-32, 2000.

[3] M. Diehl, H.G. Bock, H. Diedam, P-B. Wieber, Fast Direct Multiple Shooting Algorithms for Optimal Robot Control, Fast Motions in Biomechanics and Robotics, Vol. 340, pp. 65-93, 2007.

[4] F. Allgöwer, R. Findeisen, Z.K. Nagy, Nonlinear Model Predictive Control: From Theory to Application, Journal of the Chinese Institute of Chemical Engineers, Vol. 35, No. 3, pp. 299-315, 2004.

[5] P.S. González Cisneros, H. Werner, Nonlinear model predictive control for models in quasi-linear parameter varying form, International Journal of Robust and Nonlinear Control, 30:3945-3959, 2020.

[6] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze, E.F. Mishchenko, The Mathematical Theory of Optimal Processes, Interscience Publisher, John Wiley and Sons, New York, 1962.

[7] T. Ohtsuka, H. Fujii, Receding horizon control of a space vehicle model using stabilized continuation method, Proceedings of the 3rd Workshop on Astrodynamics and Flight Mechanics, Institute of Space and Astronautical Science. Sagamihara, Japan, 1993.

[8] T. Ohtsuka, H. Fujii. Real-time Optimization Algorithm for Nonlinear Receding-horizon Control, Automatica, Vol. 33, No. 6, pp. 1147-1154, 1997.

[9] S. Palanki, C. Kravaris, H.Y. Wang, Synthesis of state feedback laws for end-point optimization in batch processes, Chemical Engineering Science, Vol. 48, No. 1, pp. 135-152, 1993.

[10] B. Srinivasan, S. Palanki, D. Bonvin, Dynamic optimization of batch processes I. Characterization of the nominal solution, Computers and Chemical Engineering, 27 (2003) 1-26.

[11] T. Ohtsuka, A continuation/GMRES method for fast computation of nonlinear receding horizon control, Automatica, 40 (2004) 563-574.

[12] H. Deng, T. Ohtsuka, A Parallel Newton-type method for nonlinear model predictive control, Automatica, 109 (2019) 108560.

[13] C.A. Devia, M.C. Roa, J. Colorado, D. Patino, Towards a Nonlinear Model Predictive Control using the Extended Modal Series Method, Proceedings of the European Control Conference, June 12-15, 2019.

[14] M. Cannon, W. Liao, B. Kouvaritakis, Efficient MPC Optimization using Potryagin's Minimum Principle, International Journal of Robust and Nonlinear Control, 18:831-844, 2008.

[15] A.L. Dontchev, I.V. Kolmanovsky, M.I. Krastanov, V.M. Veliov, P.T Voung, Approximating optimal finite horizon feedback by model predictive control, System & Control Letters, 139 (2020) 104666.

[16] P. Malisani, F. Chaplais, N. Petit, An interior penalty method for optimal control problems with state and input constraints of nonlinear systems, Optimal Control Applications and Methods, Vol. 37, Issue 1, pp 3-33, 2016.

[17] H. Pesch, A practical guide to the solution of real-life optimal control problems, Control and Cybernetics, Vol. 23, pp. 7-60, 1994.

[18] B. Bonnard, L. Faubourg, G. Launay, E. Trélat, Optimal control with state constraints and the space shuttle re-entry problem, Journal of Dynamical and Control Systems, Vol 9., pp. 155-199, 2003.

[19] K. Graichen, N. Petit, Incorporating a class of constraints into the dynamics of optimal control problems, Optimal Control Applications and Methods, Vol. 30, Issue 6, pp. 537-561, 2009.

[20] T. Kovaltchouk, F. Rongère, M. Primot, J. Aubry, H.B. Ahmed, B. Multon, Model Predictive Control of a Direct Wave Energy Converter Constrained by the Electrical Chain Using an Energetic Approach, Proceedings of the European Wave and Tidal Energy Conference, 2015.

[21] J. Hauser, A. Saccon, A Barrier Function Method for the Optimization of Trajectory Functionals with Constraints, Proceedings of the 45th IEEE Conference on Decision & Control, December 13-15, 2006.

[22] E. Suwartadi, S. Krogstad, B. Foss, A Lagrangian-Barrier Function for Adjoint State Constraints Optimization of Oil Reservoirs Water Flooding, Proceedings of the 49th Conference on Decision & Control, December 15-17, 2010.

[23] R.M. Murray, J. Hauser, A. Jadbabaie, M.B. Milam, N. Petit, W.B. Dunbar, R. Franz, Online control customization via optimization-based control, Software-Enabled Control, Information Technology for Dynamical Systems, T. Samad, G. Balas (eds). John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2003; 149–174.

[24] D. Boiroux, J.B. Jørgensen, Sequential ℓ_1 Quadratic Programming for Nonlinear Model Predictive Control, IFAC PapersOnLine, 52-1 (2019) 474-479.

[25] A.E. Bryson, Y. Ho, Applied optimal control: optimization, estimation and control, Taylor & Francis Inc, 1975.

[26] M. Pagone, M. Boggio, C. Novara, A. Proskurnikov, G.C. Calafiore, A barrier function approach to constrained Pontryagin-based Nonlinear Model Predictive Control, ArXiv, https://doi.org/10.48550/arXiv.2203.16377, 2022 (to be submitted).

[27] Z. Wang, Y. Li, Indirect method for inequality constrained optimal control problems, IFAC PapersOnLine, 50-1 (2017) 4070-4075.

[28] C. Wang, C. Ma, J. Zhou, A new class of exact penalty functions and penalty algorithms, Journal of Global Optimization, 58:51-73, 2014.

[29] Y. Zhang, X. Yan, B. Liao, Y. Zhang, Y. Ding, Z-type control of populations for Lotka-Volterra model with exponential convergence, Mathematical Biosciences, 272 (2016) 15-23.