2013 European Control Conference (ECC)
July 17-19, 2013, Zürich, Switzerland.

Trajectory Tracking Nonlinear Model Predictive Control for Autonomous Surface Craft

Bruno J. Guerreiro, Carlos Silvestre, Rita Cunha, and António Pascoal

Abstract— **This paper presents a solution to the problem of trajectory-tracking control for autonomous surface craft (ASC) in the presence of ocean currents. The proposed solution is rooted in nonlinear model predictive control (NMPC) techniques and addresses explicitly state and input constraints. Whereas state saturation constraints are added to the underlying optimization cost functional as penalties, input saturation constraints are made intrinsic to the nonlinear model used in the optimization problem, thus reducing the computational burden of the resulting NMPC algorithm. Simulation and experimental results show that the NMPC strategy adopted yields good performance in the presence of constant currents and validate the real-time implementation of the proposed techniques.**

Fig. 1. Delfim ASC system sea trials.

I. Introduction

This paper addresses the problem of trajectory-tracking control of an autonomous surface craft (ASC) under the effect of constant ocean currents, taking explicitly into account the physical limitations of the vehicle. The increasing demand by marine scientists for adequate technological tools to sample the ocean at appropriate temporal and spatial scales motivates the use of ASCs capable of automatically acquiring and transmitting large data sets to one or more support units installed on shore. In the future, this practical setup will enable scientists to control the execution of sea missions from the security and comfort of their laboratories. trajectory-tracking controllers are traditionally based on a two-step design methodology: a fast inner loop that stabilizes the vehicle's attitude and, using a time-scale separation criterion, a slower outer loop that relies on the kinematic equations of the vehicle and converts the tracking errors into inner loop commands. An integrated approach to the design of inner-outer loop control structures for autonomous vehicles moving in 3D space was proposed in [1] and [2]. The methodology adopted relied on linearization techniques for linear controller design about trimming trajectories, together with gain scheduling techniques to switch among the linear controllers. The interested reader is referred to [3] and [4] for a discussion of topics related to this circle of ideas, and [5] for the application of similar techniques to the control of the DELFIMx ASC. It is also worth pointing out that some authors use nonlinear or adaptive control techniques to tackle the ASC control problem [6], [7].

The design methodology proposed is formulated in the scope of nonlinear model-based predictive control (NMPC) (see [8] and [9]), in line with the methodology developed in [10] for an autonomous rotorcraft. The control law adopted here is obtained by solving on-line, at each sampling instant, a finite horizon open-loop optimal control problem and using the actual state of the vehicle as the initial state. The resulting optimization problem is solved numerically using the quasi-Newton method to compute search directions and resorting to the Wolfe conditions in a line search algorithm to solve a step size optimization subproblem [11]. Throughout this work a nonlinear dynamic model of an ASC derived from first physics principles is used. Based on the nonlinear model derived, a trajectory-tracking error-space model is proposed that, when linearized about trimming trajectories, yields a time-invariant system. Furthermore, the intrinsic physical limitations of the actuators are incorporated in the design model using smooth saturation functions. This improved model allows for the optimization algorithm to generate valid control actions, even without using constraints in the cost functional. The key contributions of this paper include: i) the use of a conveniently defined trajectory-tracking error-space that is rooted in a physically sound nonlinear vehicle model, ii) the incorporation of the intrinsic physical limitations of the vehicle into the design model; iii) the use of simple and well established optimization techniques to solve the trajectory-tracking NMPC problem for a full nonlinear model of an ASC under constant disturbances, yielding a controller structure that lends itself to real-time implementation; and iv) the simulation and experimental evaluation of the methodology, providing insightful information about the performance of this strategy and its real-time implementation using the DELFIM catamaran ASC, shown in Fig. 1.

The paper is organized as follows. Section II presents a

This work was partially supported by project FCT [PEst-OE/EEI/LA0009/2011] and by the EU Project TRIDENT (Contract No. 248497). The work of Bruno Guerreiro was supported by the PhD Student Grant SFRH/BD/21781/2005 from the Portuguese FCT POCTI programme.

The authors are with the Institute for Systems and Robotics, Instituto Superior Técnico, at the Technical University of Lisbon, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal. C. Silvestre is also with the Faculty of Science and Technology, University of Macau, Taipa, Macau.
{bguerreiro,cjs,rita,antonio}@isr.ist.utl.pt

summary of the ASC dynamic model, Section III formulates the NMPC problem by describing the control problem, the constraints, and the optimization algorithms. Simulation and experimental results from sea trials are presented in Section IV, whereas Section V contains the main conclusions and discusses issues that warrant further research work.

II. Catamaran Model

This section describes the dynamic model of an ASC, which has two hulls, two propellers driven by electrical motors, and a submerged torpedo-shaped sensor container, attached to the vehicle by a central wing-shaped structure (see [12] and [13] for an in-depth presentation of this model and a description of the catamaran surface craft). Adopting standard notation in the field, let $\{I\}$ denote an inertial coordinate frame and $\{B\}$ a body fixed coordinate frame attached to the vehicle's center of mass. Further consider the position ${}^{I}\mathbf{p}_{B} = [x\ y]^T$ of the origin of $\{B\}$ with respect to $\{I\}$, the linear velocity $\mathbf{v} = [u\ v]^T$ of frame $\{B\}$ relative to $\{I\}$ and expressed in $\{B\}$, the heading angle ψ that describes the orientation of frame $\{B\}$ with respect to $\{I\}$, and the angular velocity r of frame $\{B\}$ relative to $\{I\}$, expressed in $\{B\}$. In what follows, let the generalized variables for the horizontal plane motion be given by $\boldsymbol{\nu} = \begin{bmatrix} u & v & r \end{bmatrix}^T$, $\boldsymbol{\eta} = \begin{bmatrix} x & y & \psi \end{bmatrix}^T$, and $\boldsymbol{\tau} = \begin{bmatrix} X & Y & N \end{bmatrix}^T$, which denote the generalized velocity, position and force vectors, respectively. The actuation vector is given by $\mathbf{n} = [n_c\ \ n_d]^T$, where n_c and n_d denote the common and differential modes of the propellers' speed of rotation, respectively. With this notation, the generalized equations of motion for the vehicle kinematics and dynamics are defined by

$$\begin{aligned} \dot{\boldsymbol{\eta}} &= \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu}\ , \\ \mathbf{M}\,\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\,\boldsymbol{\nu} &= \boldsymbol{\tau}(\dot{\boldsymbol{\nu}}, \boldsymbol{\nu}, \mathbf{n})\ , \end{aligned}$$

where $\mathbf{J}(\boldsymbol{\eta})$ is the rotation matrix from $\{B\}$ to $\{I\}$, $\mathbf{M}$ is the rigid body inertia matrix, and $\mathbf{C}$ is the matrix of Coriolis and centripetal terms.

A model that captures the effect of constant currents on the ASC dynamics can be obtained by rewriting the above equations in terms of the vehicle's velocity relative to the fluid. It is assumed that the generalized velocity results from the sum of two components $\boldsymbol{\nu} = \boldsymbol{\nu}_r + \mathbf{J}(\boldsymbol{\eta})^{-1}\,\boldsymbol{\nu}_f$, where $\boldsymbol{\nu}_r$ is the vehicle's generalized velocity with respect to the fluid expressed in the body frame $\{B\}$ and $\boldsymbol{\nu}_f$ is the fluid generalized velocity described in the inertial frame $\{I\}$. To obtain the new dynamic equations depending on $\boldsymbol{\nu}_r$ instead of $\boldsymbol{\nu}$, note that the generalized force $\boldsymbol{\tau}$ can be decomposed as $\boldsymbol{\tau}(\dot{\boldsymbol{\nu}}_r, \boldsymbol{\nu}_r, \mathbf{n}) = \mathbf{M}_2\,\dot{\boldsymbol{\nu}}_r + \boldsymbol{\tau}_1(\boldsymbol{\nu}_r, \mathbf{n})$, where $\mathbf{M}_2$ is a constant parameter matrix. Considering that $\boldsymbol{\tau}_2(\boldsymbol{\nu}_r, \mathbf{n}) := -\mathbf{C}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \boldsymbol{\tau}_1(\boldsymbol{\nu}_r, \mathbf{n})$ and $\mathbf{M}_3 := \mathbf{M} - \mathbf{M}_2$ is a full rank matrix, it is a matter of algebraic manipulation to show that the kinematic and dynamic equations of motion are given by

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu}_r + \boldsymbol{\nu}_f\ , \tag{1}$$

$$\dot{\boldsymbol{\nu}}_r = \mathbf{M}_3^{-1}\,\boldsymbol{\tau}_2(\boldsymbol{\nu}_r, \mathbf{n})\ . \tag{2}$$

A. Generalized Error Dynamics

This section presents a generalized error-space to describe the vehicle's motion about trimming trajectories in the absence of currents. Consider the equations of motion presented in (1) and (2) without the effect of constant currents (that is $\boldsymbol{\nu}_f = \mathbf{0}$), yielding $\boldsymbol{\nu}_r = \boldsymbol{\nu}$, and let $\boldsymbol{\nu}_c$, $\boldsymbol{\eta}_c$, and $\mathbf{n}_c$ denote the trimming values of the state and input vectors. At trimming, the generalized velocity satisfies $\dot{\boldsymbol{\nu}}_c = \mathbf{0}$, implying that $\dot{\mathbf{n}}_c = \mathbf{0}$. It can be shown that the trimming trajectories are straight lines and circles described by the vehicle at constant speed. These trajectories can be fully described by the parameter vector $\boldsymbol{\xi} = [V_c\ r_c]^T$ where $V_c = \|\mathbf{v}_c\|_2 = \sqrt{u^2 + v^2}$. Therefore, $\boldsymbol{\xi}$ fully parameterizes the set of achievable trimming trajectories.

The generalized error vector between the vehicle state and the desired trajectory is defined as

$$\mathbf{x}_e = \begin{bmatrix} \boldsymbol{\nu}_e \\ \boldsymbol{\eta}_e \\ \mathbf{x}_i \end{bmatrix} = \begin{bmatrix} \boldsymbol{\nu} - \boldsymbol{\nu}_c \\ \mathbf{J}^{-1}(\boldsymbol{\eta})\,(\boldsymbol{\eta} - \boldsymbol{\eta}_c) \\ \int_0^t \boldsymbol{\Pi}\,\boldsymbol{\eta}_e dt \end{bmatrix}\ ,$$

where $\boldsymbol{\Pi}$ denotes the projection matrix $\boldsymbol{\Pi} = \begin{bmatrix} \mathbf{I}_{2\times 2} & \mathbf{0}_{2\times 1} \end{bmatrix}$. Note that designing the controller to drive the error vector component $\mathbf{x}_i$ to zero will provide integral action. In the new coordinates, the error dynamics take the form

$$\left\{ \begin{array}{lcl} \dot{\boldsymbol{\nu}}_e &=& \dot{\boldsymbol{\nu}} \\ \dot{\boldsymbol{\eta}}_e &=& \boldsymbol{\nu} - \mathbf{J}^{-1}(\boldsymbol{\eta}_e)\,\boldsymbol{\nu}_c - \mathbf{Q}(\boldsymbol{\nu})\,\boldsymbol{\eta}_e \\ \dot{\mathbf{x}}_i &=& \boldsymbol{\Pi}\boldsymbol{\eta}_e \end{array} \right.\ , \tag{3}$$

where $\mathbf{Q}(\boldsymbol{\nu}_r) = \mathbf{S}([0\ 0\ r]^T)$ and $\mathbf{S}(\mathbf{a})$ stands for the skew-symmetric matrix that verifies $\mathbf{S}(\mathbf{a})\,\mathbf{b} = \mathbf{a} \times \mathbf{b}$. Using (3), it is straightforward to show that the linearization of the error dynamics about $\boldsymbol{\nu}_e = 0$ and $\mathbf{n}_e := \mathbf{n} - \mathbf{n}_c = \mathbf{0}$ is time invariant. The ASC error model described in (3) can also be rewritten as

$$\dot{\mathbf{x}}_e = \mathbf{f}_e\,(\mathbf{x}_e, \mathbf{n}_e)\ .$$

In most practical mission scenarios involving ASCs, the only available velocity measurements are those of the velocity of the vehicle relative to the fluid, provided by a Doppler. Therefore, the implementation of this error-space uses the velocity relative to the fluid $\boldsymbol{\nu}_r$, that is, $\boldsymbol{\nu}_e \simeq \boldsymbol{\nu}_r - \boldsymbol{\nu}_c$.

B. Intrinsic Input Saturation

As only the error states and inputs are used within the optimization problem, the definition of input constraints on the actual vehicle inputs could be achieved either by including additional constraints or by using variable constraints for the error inputs at each sampling instant along the prediction horizon. Alternatively, even complex physical constraints can be easily incorporated in the nonlinear design model of the vehicle, as decribed below.

Let the new inputs $\bar{\mathbf{n}} = [\bar{n}_c\ \ \bar{n}_d]^T$ be defined as smoothly saturated functions of the regular inputs $\mathbf{n} = [n_c\ \ n_d]^T$, so that the dynamic equation is now given by $\dot{\boldsymbol{\nu}}_r = \mathbf{M}_3^{-1}\,\boldsymbol{\tau}_2(\boldsymbol{\nu}_r, \bar{\mathbf{n}}(\mathbf{n}))$. The saturation functions are derived from the basic function $\bar{a}(a) = \frac{a}{1+|a|}$, applying translations and scaling both to the function and its derivative, such that

inside the bounds $\bar{a} = a$ and outside the bounds $\bar{a}$ tends smoothly to the maximum value a_{max} or the minimum value a_{min}. For the type of vehicle considered in this work, it is necessary to impose a minimum value for the common mode input $\mathbf{n}_c$, and also bounds for the differential input, which depend on the current value of the common mode input. The saturation function of the common mode input is defined as

$$\bar{n}_c(n_c) = \begin{cases} n_c & , \ \underline{\epsilon}_c \leq n_c \leq \overline{\epsilon}_c \\ \overline{\epsilon}_c + \frac{n_c - \overline{\epsilon}_c}{1 + \frac{n_c - \overline{\epsilon}_c}{\epsilon}} & , \ n_c > \overline{\epsilon}_c \\ \underline{\epsilon}_c + \frac{n_c - \underline{\epsilon}_c}{1 - \frac{n_c - \underline{\epsilon}_c}{\epsilon}} & , \ n_c < \underline{\epsilon}_c \end{cases} ,$$

with $\overline{\epsilon}_c = n_{c_{max}} - \epsilon$ and $\underline{\epsilon}_c = n_{c_{min}} + \epsilon$, where $0 < \epsilon < 1$ is a constant (typically 0.01), that defines the length of the smooth transition. The saturation of the differential input is given by the function $\bar{n}_d(\bar{n}_c, n_d)$, obtained using the same approach presented above. In brief, considering that $\mathbf{n} \in \mathcal{N} \subset \mathbb{R}^{n_n}$, the procedure described above defines the new saturated input vector as $\bar{\mathbf{n}} \in \mathbb{R}^{n_n}$, simplifying the optimization problem formulation.

C. Discretization and Delay Modeling

In what follows, the control problem is formulated as a discrete-time open-loop optimal control problem. For this reason, the equations of motion of the vehicle are described as difference equations. Considering the notation $\mathbf{a}_k := \mathbf{a}(k\,T_s)$, for some time-dependent vector $\mathbf{a}(t)$ and sample time T_s, the difference system equations are obtained using the forward Euler discretization, yielding

$$\mathbf{x}_{e_{k+1}} \approx \mathbf{x}_{e_k} + T_s\, \mathbf{f}_e(\mathbf{x}_{e_k}, \mathbf{n}_{e_k})) = \mathbf{f}_d(\mathbf{x}_{e_k}, \mathbf{n}_{e_k}) \ .$$

To model the delay between the instant the state variables $\mathbf{x}_{e_k}$ are measured and the instant a new control action $\mathbf{n}_{e_{k+1}}$ is made available, the model is augmented with an extra delay state. Considering the new state vector $\mathbf{x}_k := [\mathbf{x}_{e_k}^T \ \mathbf{x}_{n_k}^T]^T$, the input vector $\mathbf{u}_k := \mathbf{n}_{e_k}$, and $\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) := [\ \mathbf{f}_d(\mathbf{x}_{e_k}, \mathbf{x}_{n_k})^T \quad \mathbf{n}_{e_k}^T \ ]^T$, the model takes the form

$$\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right) \ . \tag{4}$$

III. Model Predictive Control Problem

In this section the NMPC problem is formulated as a discrete-time open-loop optimal control problem with finite horizon, subject to the discrete nonlinear model equations as well as state and input saturation constraints. From (4), the vehicle dynamics can be modeled as a discrete-time state-space equation with state $\mathbf{x}_k \in \mathcal{X}$ and input $\mathbf{u}_k \in \mathcal{U}$, where $\mathcal{X} \subset \mathbb{R}^{n_x}$ and $\mathcal{U} \subset \mathbb{R}^{n_u}$ denote the sets of admissible state and control vectors, respectively. At each instant of time, the NMPC algorithm uses the nonlinear model of the vehicle and the current state to predict the evolution of the system within a predefined time horizon. For simplicity, each instant k is considered to be the initial instant of the horizon prediction, so that in the rest of this section $\mathbf{x}_{k+i}$ and $\mathbf{u}_{k+i}$ are denoted as $\mathbf{x}_i$ and $\mathbf{u}_i$, respectively. Let N be the prediction horizon of the control problem, $\mathbf{U} = \{\mathbf{u}_0, \ldots, \mathbf{u}_{N-1}\}$ the sequence of control inputs, and $\mathbf{X} = \{\mathbf{x}_0, \ldots, \mathbf{x}_N\}$ the sequence of state vectors generated by that control sequence. The saturation constraints for the state and input sequences are defined by the conditions $\mathbf{X} \in \mathfrak{X}_N$ and $\mathbf{U} \in \mathfrak{U}_N$, where $\mathfrak{X}_N = \{\mathbf{X} : \mathbf{x}_i \in \mathcal{X}, \forall_{i=0,\ldots,N}\}$ and $\mathfrak{U}_N = \{\mathbf{U} : \mathbf{u}_i \in \mathcal{U}, \forall_{i=0,\ldots,N-1}\}$. Using (4) and denoting the model function as $\mathbf{f}_i := \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i)$, the model constraint can be written as

$$F_M(\mathbf{X}, \mathbf{U}) = \left[(\mathbf{f}_0 - \mathbf{x}_1)^T \quad \cdots \quad (\mathbf{f}_{N-1} - \mathbf{x}_N)^T\right]^T = \mathbf{0} \ .$$

Given these constraints, the NMPC problem can be defined as the nonlinear optimization problem

$$\mathbf{U}^* = \arg\min_{\mathbf{U}} J \tag{5}$$

$$s.t. \quad \mathbf{X} \in \mathfrak{X}_N \ , \ \mathbf{U} \in \mathfrak{U}_N \tag{6}$$

$$F_M(\mathbf{X}, \mathbf{U}) = \mathbf{0} \tag{7}$$

where $J = F_N + \sum_{i=0}^{N-1} L_i$, $F_i = \frac{1}{2}\,\mathbf{x}_i^T\,\mathbf{P}\,\mathbf{x}_i$, $L_i = \frac{1}{2}\left[\mathbf{x}_i^T\,\mathbf{Q}\,\mathbf{x}_i + \mathbf{u}_i^T\,\mathbf{R}\,\mathbf{u}_i\right]$, whereas $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$ are symmetric positive definite matrices. In brief, the NMPC objective is to find, at each instant k, the optimal control sequence $\mathbf{U}^*$ with horizon N, such that the resulting state sequence $\mathbf{X}^*$ together with $\mathbf{U}^*$ minimize the cost functional J without violating the state and input constraints imposed by (6). Following a by now standard approach, the constrained optimization problem presented above can be solved by reformulating it as an unconstrained optimization problem and using gradient methods to estimate the optimal solution.

A. State and Input Saturation Constraint

The saturation constraints defined in (6) are included in the optimization problem to complement the intrinsic input constraints described in Section II-B and enable the definition of mission specific bounds for both state and input vectors. These constraints can be incorporated in the cost functional as a penalty function $F_R\left(\mathbf{x}, \mathbf{u}\right)$, which is zero-valued for $\mathbf{x} \in \mathcal{X}$ and $\mathbf{u} \in \mathcal{U}$ and behaves as a quadratic function outside these sets. Defining the feasibility sets for state and input vectors as $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^{n_x} : \underline{x}^{(j)} \leq x^{(j)} \leq \overline{x}^{(j)} \ \forall_{j=1,\ldots,n_x}\}$ and $\mathcal{U} = \{\mathbf{u} \in \mathbb{R}^{n_u} : \underline{u}^{(l)} \leq u^{(l)} \leq \overline{u}^{(l)} \ \forall_{l=1,\ldots,n_u}\}$, respectively, the penalty function is defined as

$$F_S(\mathbf{x}, \mathbf{u}) = \sum_{j=1}^{n_x} f_S(x^{(j)}) + \sum_{l=1}^{n_u} f_S(u^{(l)}) \ ,$$

where $f_S(a) := \frac{1}{2}\,h^2(|a - a_{\text{center}}| - a_{\text{range}})\,w_a$, $a_{\text{center}} := (\overline{a} + \underline{a})/2$, $a_{\text{range}} := \overline{a} - a_{\text{center}}$, w_a is a positive scalar weight, whereas $h(a) = a$ if $a > 0$, and $h(a) = 0$ otherwise.

B. Unconstrained Optimization Problem

Adding the saturation constraints to the optimization cost functional, the new problem can be written as

$$\mathbf{U}^* = \arg\min_{\mathbf{U}} \bar{J} \tag{8}$$

$$s.t. \quad F_M(\mathbf{X}, \mathbf{U}) = \mathbf{0} \tag{9}$$

where $\bar{J} = \bar{F}_N + \sum_{i=0}^{N-1} \bar{L}_i$,, $\bar{F}_i = F_i + F_R(\mathbf{x}_i, \mathbf{0})$ and $\bar{L}_i = L_i + F_R(\mathbf{x}_i, \mathbf{u}_i)$. The elimination method using Lagrange multipliers is used to solve the model constraint (9). Introducing the Lagrange multiplier sequence $\mathbf{\Lambda} = \{\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_N\}$ and the Hamiltonian $H_i = H\left(\mathbf{x}_i, \mathbf{u}_i\right) = \bar{L}_i + \boldsymbol{\lambda}_{i+1}^T\,\mathbf{f}_i$, after

some algebraic manipulations, the cost functional $\bar{J}$ can be rewritten as

$$\bar{J} = \bar{F}_N - \boldsymbol{\lambda}_N^T \mathbf{x}_N + \sum_{i=1}^{N-1} \left[H_i - \boldsymbol{\lambda}_i^T \mathbf{x}_i \right] + H_0.$$

For a fixed initial state $\mathbf{x}_0$, the first order conditions of optimality yield

$$\frac{\partial \bar{J}}{\partial \mathbf{x}_i} = \frac{\partial H_i}{\partial \mathbf{x}_i} - \boldsymbol{\lambda}_i = \mathbf{0}\,, \quad \forall_{i=1,\ldots,N-1}\,, \tag{10}$$

$$\frac{\partial \bar{J}}{\partial \mathbf{x}_N} = \frac{\partial \bar{F}_N}{\partial \mathbf{x}_N} - \boldsymbol{\lambda}_N = \mathbf{0}\,, \tag{11}$$

$$\frac{\partial \bar{J}}{\partial \mathbf{u}_i} = \frac{\partial H_i}{\partial \mathbf{u}_i} = \mathbf{0}\,, \quad \forall_{i=0,\ldots,N-1}\,, \tag{12}$$

where $\frac{\partial H_i}{\partial \mathbf{u}_i} = \frac{\partial \bar{L}_i}{\partial \mathbf{u}_i} + \frac{\partial \mathbf{f}_i}{\partial \mathbf{u}_i}\boldsymbol{\lambda}_{i+1}$ and $\frac{\partial H_i}{\partial \mathbf{x}_i} = \frac{\partial \bar{L}_i}{\partial \mathbf{x}_i} + \frac{\partial \mathbf{f}_i}{\partial \mathbf{x}_i}\boldsymbol{\lambda}_{i+1}$. Because in the cost functional the Lagrange multipliers sequence is multiplied by zero value terms, $\boldsymbol{\lambda}_{i+1}(\mathbf{f}_i - \mathbf{x}_{i+1})$, they can be arbitrarily chosen. In particular, by choosing $\boldsymbol{\lambda}_N = \frac{\partial \bar{F}_N}{\partial \mathbf{x}_N}$ and $\boldsymbol{\lambda}_i = \frac{\partial H_i}{\partial \mathbf{x}_i}$, for all $i = N-1, \ldots, 1$, the first order conditions of optimality reduce to (12). Thus, an iterative algorithm based on the first order gradient method can be readily applied to estimate $\mathbf{U}^*$, whereby at each optimization iteration j, the control sequence is updated according to

$$\mathbf{U}^{(j+1)} = \mathbf{U}^{(j)} + s\,\boldsymbol{\Delta}^{(j)}\,, \tag{13}$$

where s denotes the step size and $\boldsymbol{\Delta}^{(j)}$ the search direction. The optimization algorithm can be summarized as follows.

Algorithm 1: Minimization algorithm for the NMPC unconstrained problem.

1) Initialize $\mathbf{X}^{(0)}$, $\mathbf{U}^{(0)}$ and $j = 0$;
2) Compute $\{\boldsymbol{\lambda}_i\}$, $i = N, \ldots, 1$;
3) Compute $\left\{\frac{\partial H_i}{\partial \mathbf{u}_i}\right\}$, $i = 0, \ldots, N-1$;
4) Compute the search direction $\boldsymbol{\Delta}^{(j)}$;
5) Compute the step size s using Wolfe conditions;
6) Compute $\mathbf{U}^{(j+1)}$ using (13) and $\mathbf{X}^{(j+1)} = \{\mathbf{x}_i\}$ using $\mathbf{x}_{i+1} = \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i)$, for $i = 0, \ldots, N-1$;
7) If $\|\nabla \bar{J}^{(j)}|_{(j)}\| \geq \varepsilon$: repeat from (2); else, apply $\hat{\mathbf{u}}_0$ to system and set $\mathbf{U}^{(0)} = \{\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_{N-1}\}$.

The search direction is obtained using the quasi-Newton method, $\boldsymbol{\Delta}^{(j)} = -\mathbf{D}^{(j)} \nabla \mathbf{H}^{(j)}|_{\mathbf{U}^{(j)}}$, where $\nabla \mathbf{H}^{(j)}|_{\mathbf{U}^{(j)}}$ is the sequence of vectorized Hamiltonian derivatives $\text{vec}\left(\frac{\partial H_i^{(j)}}{\partial \mathbf{u}_i}\right)$, for all $i = 0, \ldots, N-1$, and $\mathbf{D}^{(j)}$ is an estimate of the inverse matrix of the second-order derivative of the Hamiltonian sequence, as detailed in [11].

The line search optimization subproblem is numerically solved using the Wolfe rule. This approach guarantees a decrease of the cost functional, as the well known Armijo rule does, and ensures reasonable progress by ruling out unacceptably short steps [11]. Consider the step size optimization subproblem defined by

$$s^* = \arg\min_{s \geq 0} \phi(s)\,,$$

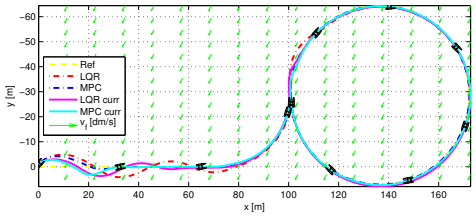
where $\phi(s) = \bar{J}^{(j+1)} = \bar{J}\left(\mathbf{X}^{(j+1)}, \mathbf{U}^{(j+1)}\right)$, with $\mathbf{U}^{(j+1)}$ as in (13) and its derivative is given by $\phi'(s) = \frac{d\,\bar{J}^{(j+1)}}{ds}$. The Wolfe algorithm finds an acceptable step size, which is an estimate of the optimal step size.

A formal stability analysis of the proposed NMPC methodology is beyond the scope of this paper, noting that significant work on related approaches is available in the literature (see [8] and references therein). From the literature, it can be concluded that the stability of the proposed algorithm relies on the choice of the horizon N, the parameter matrices $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$, as well as the terminal cost function, $\bar{F}(\mathbf{x}_N)$. In summary, considering a locally stabilizing terminal control law, $\boldsymbol{\kappa}_f(\mathbf{x})$, and respective positively invariant terminal set $\mathcal{X}_f$, it must be guaranteed that the system converges to the terminal set, which, by definition, ensures convergence to the equilibrium point if the terminal control law is used, through an appropriate definition of $\bar{F}(\mathbf{x}_N)$. Although necessary for the formal convergence analysis, most NMPC techniques do not require the explicit use of the terminal set or the terminal control law for the computation of the NMPC control law.

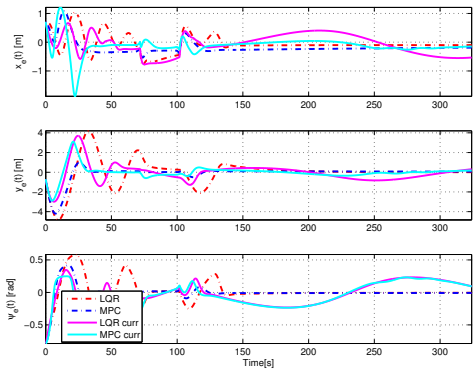
IV. Simulation and Experimental Results

In this section, the performance of the nonlinear NMPC controller introduced above is first evaluated in simulation by drawing a comparison with the results obtained with a basic LQR gain switching methodology (see [14] for a similar approach using $\mathcal{H}_2$ synthesis). In the results presented hereafter, the ASC nonlinear model described in Section II is parameterized for the DELFIMx Catamaran and used both in the NMPC control algorithm and plant simulation. The simulations where carried out in an Intel Pentium Centrino processor at 1.7 GHz, using Matlab/Simulink with C mex-functions. The reference trajectory, defined in the inertial frame, was selected to illustrate the behavior of the control algorithms in extreme conditions, which include discontinuities in the reference velocities and non zero initial errors, and is composed of three different sections: i) a straight line ($\|\mathbf{v}_c\| = 1$ m/s and $r_c = 0$ rad/s), to be tracked between $\boldsymbol{\eta}_c = \mathbf{0}_{3\times 1}$ and $\boldsymbol{\eta}_c = [72\ 0\ 0]'$, with initial conditions $\boldsymbol{\nu}_0 = [0.6\ 0\ 0]'$ and $\boldsymbol{\eta}_0 = [0\ \ -1\ \ -\pi/4]'$; ii) one fourth of a circle turning to port side ($\|\mathbf{v}_c\| = 1.5$ m/s and $r_c = -3$ deg/sec); and iii) a complete circle turning to starboard ($\|\mathbf{v}_c\| = 1$ m/s and $r_c = 1.6$ deg/sec). The sample time is $T_s = 0.2$ s and the horizon is $N = 30$ sample times, or equivalently, 6 seconds. The precision of the solution is determined by the algorithm stop conditions, e. g., $|\bar{J}_k^{(j)} - \bar{J}_k^{(j-1)}| < 10^{-2}$.

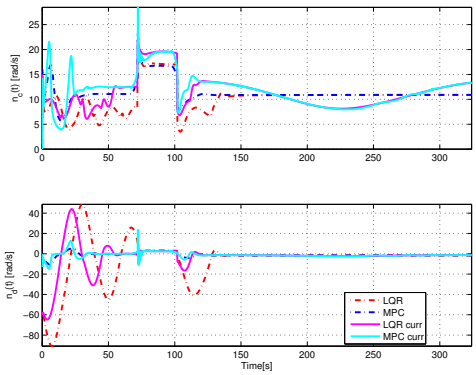
Two different scenarios were simulated in order to highlight the major differences between the LQR and NMPC controllers: 1) trajectory-tracking without current; and 2) trajectory-tracking with a constant current ($\boldsymbol{\nu}_f = [-0.1\ \ 0.2\ \ 0]'$ m/s). The simulation results of these two scenarios are presented in Fig. 2, showing the trajectories described by the ASC using the NMPC and LQR controllers, as well as the time evolution of the position error and the actuation. It can be seen that both the LQR and the NMPC control methodologies achieve the tracking objective, with or without constant current. However, the LQR method presents larger excursions in actuation both at the initial stage



(a) Trajectory



(b) Generalized position error



(c) Actuation

Fig. 2. Simulation results

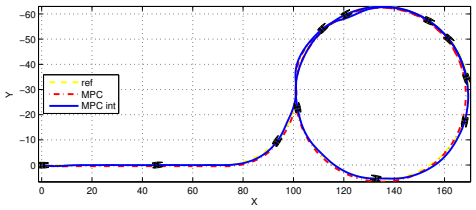
and during the transitions between sections, which generally translate into larger position errors. Moreover, the control effort demanded by the LQR method is far greater than that of the NMPC, and it even violates the conditions for valid operation (by having $|n_d| > n_c$). The major limitation of the NMPC method is the computation time needed to determine the next control action, which must be smaller than the sampling time $T_s = 0.2$ s. For the specified simulations, this threshold was never exceeded and the average and maximum CPU times obtained in the absence of currents were 0.016 s and 0.14 s, respectively, whereas in the presence of a constant current these values increased slightly to 0.023 s and 0.16 s.

The real-time implementation of the trajectory-tracking NMPC methodology for ASCs presented in this paper is also experimentally validated in this section. The vehicle used for these sea trials is the ASC DELFIM catamaran, shown in Fig. 1, which was designed, built, and instrumented by IST/ISR. The vehicle is equipped with a MEMSENSE nanoIMU and a global positioning system (GPS) unit working in differential mode, from which the position, attitude, and velocities of the vehicle in $\{I\}$ can be computed using well known Kalman filtering techniques [15]. In order to enable real-time implementation of the NMPC strategy, the controller is implemented in a dedicated onboard computer featuring a Intel® Core™ 2 Duo T9550 processor, with 4GB of memory and Ubuntu 12.04 operative system with the Robotics Operative System (ROS) software framework. The ASC Delfim was tested in the Lisbon Oceanarium lake, in Portugal, using a trajectory very similar to the one used in the simulation results and a horizon of $N = 50$ sampling periods. This is a harsh trajectory, with abrupt changes in velocity and direction, and is used with the objective of testing the robustness of the proposed algorithms.

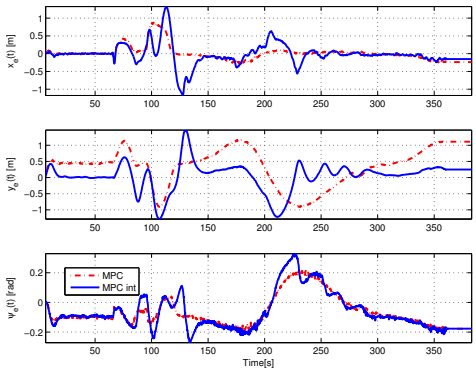
The main goals of presenting these experimental results are twofold: the validation of the proposed NMPC strategy for real-time control of ASCs and the evaluation of performance of different configurations of the error-space model. In particular, two different error-space vectors are considered: 1) the complete error-space vector, as defined in (II-A), which includes position integral action; and 2) the error-space vector without integral action, defined as $\mathbf{x}_e = \begin{bmatrix}\boldsymbol{\nu}_e^T & \boldsymbol{\eta}_e^T\end{bmatrix}^T$. The results from the described sea-trials are shown in Fig. 3, which provides the time evolution of the generalized position errors, actuation, and the computation time. It can be seen that both algorithms were able to successfully control the catamaran along the desired trajectory. The analysis of these results shows that the position tracking error is always lower than 1.21 m, when using integral action, and below 0.85 m without integral action. Nonetheless, the average error is lower when using integral action, 0.26 m, than when no integral states are considered, 0.45 m. This is the result of demanding more from the NMPC controller, naturally yielding lower steady state errors, even in the presence of disturbances or unmodeled dynamics. Fig. 3(d) shows the CPU time that the algorithms used in each sampling period to compute the next actuation values. As the CPU time is limited above by the sampling time, the implemented algorithms are forced to abort the optimization procedure whenever the elapsed time values go beyond $1.25\,T_s$, maintaining the previous control value, and saving the optimization terminal conditions conditions to enable a better initialization value for the next sampling period. It can be seen that the algorithm that does not use integral action is much faster, with a maximum CPU time of 92 ms. Conversely, the integral action gets the NMPC algorithm close to the CPU limit, as the maximum CPU time is reached during some transitions between trimming trajectories. However, this does not compromise the operation of the vehicle.

V. Conclusions

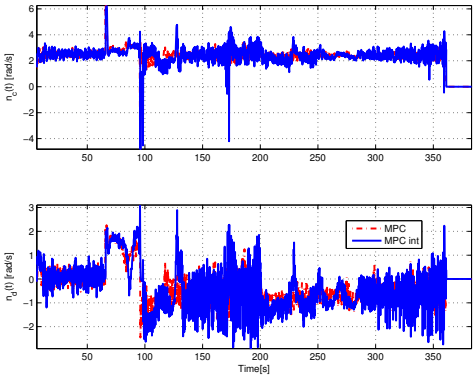
This paper presented a NMPC strategy for motion control of ASCs under the effect of constant currents. A nonlinear model of an ASC catamaran is used to define an error-space dynamic model, which is then used by the NMPC algorithm to find the adequate control action. In contrast to the standard approach in NMPC literature, the actuation constraints were



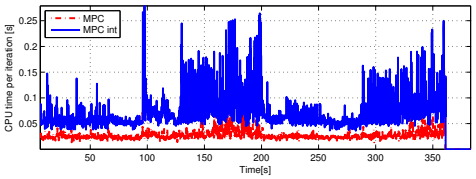
(a) 2-D Trajectory



(b) Generalized position error, $\boldsymbol{\eta}_e$



(c) Actuation $\mathbf{n}$



(d) CPU time

Fig. 3. Trajectory-tracking NMPC sea trials data.

incorporated into the model so that, while working with an error dynamics model, every control action provided by the NMPC algorithm is always valid without affecting the overall computational time. The simulation and experimental results validate the real-time implementation of the proposed NMPC strategy and show that the presented solution can effectively steer the vehicle along a demanding reference trajectory and in the presence of constant currents.

For some vehicles, the use of trajectory-tracking may impose performance bounds on the controlled system. As some of these bounds are not present when using path-following, and bearing in mind mission scenarios with no time-critical requirements, further modifications shall include the formulation of a path-following error-space. Additionally, it is important that the model of the vehicle include wave disturbances in order to test the control algorithm performance under more realistic scenarios.

ACKNOWLEDGEMENTS

The authors express their gratitude to the DSOR Lab Delfim team, in particular to L. Sebastião, A. Oliveira, B. Cardeira, M. Rufino, and P. Batista, having developed the DELFIM prototype ASC and for helping with the complex logistics necessary for testing such a platform, which has proven instrumental in the experimental validation of the techniques presented in this paper.

REFERENCES

[1] I. Kaminer, A. Pascoal, E. Hallberg, and C. Silvestre, "Trajectory tracking for autonomous vehicles: An integrated approach to guidance and control," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 1, pp. 29–38, January 1998.

[2] C. Silvestre, A. Pascoal, and I. Kaminer, "On the design of gain-scheduled trajectory tracking contollers," *International Journal of Robust and Nonlinear Control*, vol. 12, no. 9, pp. 797–839, July 2002.

[3] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, no. 10, pp. 1401–1425, October 2000, survey Paper.

[4] N. Paulino, C. Silvestre, and R. Cunha, "Affine parameter-dependent preview control for rotorcraft terrain following," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 29, no. 6, pp. 1350–1359, 2006.

[5] P. Gomes, C. Silvestre, A. Pascoal, and R. Cunha, "A coastline following preview controller for the delfimx vehicle," in *16th International Offshore and Polar Engineering Conference*, Lisbon, Portugal, July 2007.

[6] P. Encarnação, A. Pascoal, and M. Arcak, "Path following for autonomous marine craft," in *5th IFAC Conference on Marine Craft Maneuvering and Control*, Aalborg, Denmark, August 2000, pp. 117–122.

[7] K. D. Do, Z. P. Jiang, and J. Pan, "Robust adaptive path following of underactuated ships," *Automatica*, vol. 40, no. 6, pp. 929–944, June 2004.

[8] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 790–814, 2000, survey Paper.

[9] G. Sutton and R. Bitmead, "Computational implementation of nonlinear model predictive control to nonlinear submarine," in *Nonlinear Model Predictive Control*, ser. Progress in Systems and Control Theory, F. Allgöwer and A. Zheng, Eds. Basel-Boston-Berlin: Birkhäuser Verlag, 2000, vol. 26, pp. 461–471.

[10] B. J. Guerreiro, C. Silvestre, and R. Cunha, "Terrain Avoidance Nonlinear Model Predictive Control for Autonomous Rotorcraft," *Journal of Intelligent & Robotic Systems*, vol. 69, no. 1, pp. 69 –85, 2012.

[11] J. Nocedal and S. Wright, *Numerical Optimization*, ser. Springer Series in Operation Reasearch. Springer, 1999.

[12] M. Prado, "Modeling and control of an autonomous oceanographic vehicle," Master's thesis, Instituto Superior Técnico, Lisbon, 2002.

[13] T. I. Fossen, *Guidance and Control of Ocean Vehicles*. New York, USA: Wiley, 1994.

[14] B. J. Guerreiro, C. Silvestre, R. Cunha, and D. Antunes, "Trajectory tracking H2 controller for autonomous helicopters: an application to industrial chimney inspection," in *17th IFAC Symposium on Automatic Control in Aerospace*, June 2007.

[15] J. F. Vasconcelos, B. Cardeira, C. Silvestre, P. Oliveira, and P. Batista, "Discrete-Time Complementary Filters for Attitude and Position Estimation: Design, Analysis and Experimental Validation," *IEEE Transactions on Control Systems Technology*, vol. 19, pp. 181 – 198, January 2011.