2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

Energy–Optimal Planning and Shrinking Horizon MPC for Vessel Docking in River Current Fields

Hannes Homburger[1], Stefan Wirtensohn[1], Moritz Diehl[2] and Johannes Reuter[1]

***Abstract*— The problem of controlling autonomous surface vessels in an energy–optimal way is important for the electrification of maritime systems and is currently being investigated by many researchers. In this paper, we use numerical optimal control to plan an energy–optimal docking trajectory in river currents and show that it can save energy compared to other widespread planning approaches. An optimal control problem including a detailed vessel model is defined, transcribed into a nonlinear optimization problem via direct multiple shooting, and solved using a homotopy procedure. The optimal solution is compared to a geometrical path planning approach with path–velocity decomposition. The results of this comparison show that prescribing a path with fixed vessel orientation leads to very suboptimal results. Further, we demonstrate how shrinking horizon MPC can control the vessel in an energy–optimal way even under severe disturbances, by replanning the energy–optimal trajectories in real–time. We believe that energy–optimal MPC could become a key technology for the electrification of maritime systems.**

I. Introduction

Numerical optimal control (NOC) is a decision making technique for high–dimensional constrained nonlinear systems. Recently, NOC has been used in the field of maritime systems for both, control in the form of model predictive control (MPC) [1] and planning in the form of trajectory optimization [2]. Recently, different methods like feedback linearization [3], flatness–based optimal control [4] or methods based on graph search [5] have been used to optimize trajectories in maritime systems. Further, direct approaches to optimal trajectory planning were presented e.g. in the form of direct collocation [6] and pseudospectral optimization [7]. In this paper, we employ a direct multiple shooting approach [8] to calculate the energy–optimal docking trajectory of an autonomous surface vessel (ASV) time efficiently. This method could be a key feature in energy–efficient operations of autonomous water taxis [9] or autonomous ferries [10] that have to dock several times a day. Because these transportation systems usually act in flowing waters, we consider current effects in the dynamic model and the optimization problem. We investigate the question: How to define an optimal control problem (OCP) for an energy–optimal docking task considering the river current and how to employ direct multiple shooting to solve this OCP time efficiently? Further, we compare the method to an energy–optimal path–velocity decomposition and present a new method combining both. The paper is structured as follows: In Section II, the OCP for the docking task is stated and defined in detail. In Section III, we present the transcription of the infinite–dimensional continuous time OCP to a finite–dimensional nonlinear program (NLP) via direct multiple shooting. In Section IV, the trajectory planning method is extended to shrinking horizon MPC. The presented method is applied to compute the energy optimal docking trajectory of the ASV *Solgenia* in Section V. Finally, Section VI concludes the results of the paper by describing the potential and limitations of the approach.

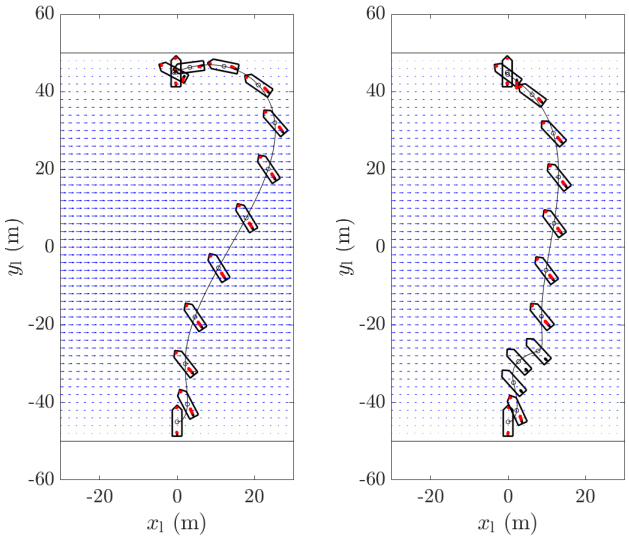

Fig. 1. Solution trajectories of the energy–optimal maneuver with maximal current $v_{\text{river}} = 1.5$ m/s (left) and $v_{\text{river}} = 1.1$ m/s with disturbance in form of failed actuators for $t \in [20, 40]$ s (right). The current field is visualized in blue. The vessel pose (black) and the actuator force vectors (red) are plotted at the time instances $t = k\Delta T$ with $k = 0, ..., 12$ and $\Delta T = 10$ s.

II. Definition of the Optimal Control Problem

Following standard notation [1], we state a continuous time OCP with fixed initial and terminal state as

$$\underset{x(\cdot),u(\cdot)}{\text{minimize}} \int_0^T L(x(t),u(t))dt + E(x(T)) \tag{1a}$$

$$\text{subject to} \quad x(0) - x_0 = 0, \tag{1b}$$

$$x(T) - x_T = 0, \tag{1c}$$

$$\dot{x}(t) - f(x(t),u(t)) = 0, \ t \in [0,T], \tag{1d}$$

$$h(x(t),u(t)) \leq 0, \ t \in [0,T], \tag{1e}$$

where $T > 0$ denotes the terminal time, $x : [0,T] \to \mathbb{R}^{n_x}$ denotes the state trajectory, $u : [0,T] \to \mathbb{R}^{n_u}$ denotes the input trajectory, $L : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ denotes the running cost,

[1] Institute of System Dynamics, HTWG Konstanz - University of Applied Sciences, 78462 Konstanz, Germany {hhomburg, stwirten, jreuter} @htwg-konstanz.de
[2] Department of Microsystems Engineering (IMTEK) and Department of Mathematics, University of Freiburg, 79110 Freiburg, Germany moritz.diehl@imtek.uni-freiburg.de

$E : \mathbb{R}^{n_x} \to \mathbb{R}$ denotes the terminal cost, $x_0, x_T \in \mathbb{R}^{n_x}$ denote the desired initial and terminal state, $\dot{x} : [0,T] \to \mathbb{R}^{n_x}$ denotes the time derivative of the state trajectory, the system dynamics is denoted by $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$, the path constraints are denoted by $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_h}$, and $n_x, n_u, n_h \in \mathbb{N}$ denote the state dimension, the control dimension, and the number of scalar inequality constraints. In the following, the different parts of the OCP are defined to dock a vessel energy–optimal.

A. Cost Function

The cost function (1a) is given in standard Bolza form. To find the trajectory that needs the least electric energy, the cost function must represent the electric energy consumption, and therefore the running costs must model the power consumption of the vessel. In the literature, a variety of different approaches to modeling the energy consumption of a vessel can be found. Selected examples of the costs are given by a quadratic function of the actuator states [4], by a lookup table [5], or by linear damping [7]. The parameters of all these models can be identified using real–world data on the vessel's electric power consumption. To obtain this model, a database

$$\mathbb{D} = \begin{bmatrix} (x_1, u_1, P_1) \\ (x_2, u_2, P_2) \\ \vdots \\ (x_K, u_K, P_K) \end{bmatrix} \tag{2}$$

with $K \in \mathbb{N}$ entries is recorded while driving the vessel in different scenarios. The database contains measurements of the system state x_k, the system input u_k, and the corresponding power consumption P_k, where the subscript k denotes the measurement at time $t = k\Delta t$ with $\Delta t > 0$ and $k = 1, \ldots, K$. The optimal model parameters denoted by $\theta^* \in \mathbb{R}^{n_\theta}$ can be identified by solving the least–squares problem

$$\theta^* = \underset{\theta \in \mathbb{R}^{n_\theta}}{\operatorname{argmin}} \sum_{k=1}^{K} \left(P_{\text{model}}(x_k, u_k, \theta) - P_k \right)^2 \tag{3}$$

numerically. Note that the parameters of both, physics–based gray box models and black box models e.g. neural networks can be identified using this strategy. In Figure 2, an exemplary gray box model characteristic is plotted over full–scale data.

B. Initial and Terminal Constraints

The definitions of the initial and terminal conditions of the trajectory are given in the form of equality constraints in (1b) and (1c). The initial state x_0 determines the state at the start of the investigated trajectory, i.e., the current state of the vessel. The terminal state x_T specifies the state that shall be reached at the end of the trajectory, e.g., at a docking position.

C. System State, System Input, and System Dynamics

According to [11], by neglecting the pitch, roll, and heave motion due to calm water conditions, the system state of a three–degree–of–freedom (3–DOF) model of a vessel can be decomposed into three parts $x = \left(\eta^\top, \nu_\mathrm{r}^\top, a^\top\right)^\top$. The first part

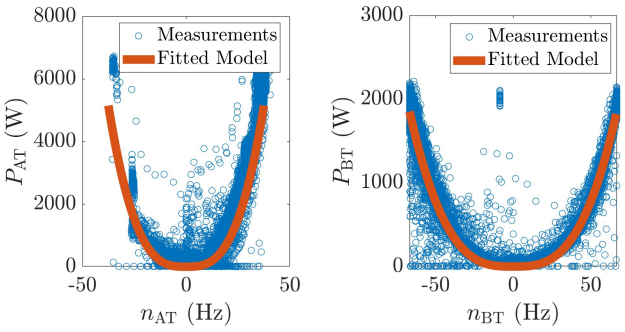

Fig. 2. Exemplary propeller characteristic curves of a bow thruster (BT) and an azimuth thruster (AT) plotted with measurement data on the propellers' electrical power consumption recorded in real–world experiments.

is the pose in a local east–north–up (ENU) frame. The pose is given by $\eta = (x_\mathrm{l}, y_\mathrm{l}, \psi)^\top \in \mathbb{R}^3$, where the vessel's position is defined by x_l and y_l, and the vessel's yaw angle w.r.t. the x_l–axis is denoted by ψ. The second part of the system state is given by the velocity of the vessel relative to the surrounding water in a body–fixed frame. The relative body–fixed velocity is given by $\nu_\mathrm{r} = (u_\mathrm{r}, v_\mathrm{r}, r_\mathrm{r})^\top$, where u_r denotes the relative velocity in the surge direction, v_r denotes the relative velocity in the sway direction, and r_r denotes the velocity of the yaw angle. The third part of the system state is the actuator state denoted by $a \in \mathbb{R}^{n_a}$. Note that n_a depends on the individual actuator configuration of the vessel. A visualization of the different frames is given in Figure 3. The system input is given by the time derivative of the actuator state collected in $u \in \mathbb{R}^{n_u}$ with $n_u = n_a$. The dynamics of the actuators is fast compared to the dynamics of the vessel and is controlled by lower level controllers, whose dynamics is neglected. However, the rate of change will be limited. The actuator configuration differs between vessels. However, we assume that a function $\tau_\mathrm{a} : \mathbb{R}^{n_a} \times \mathbb{R}^3 \to \mathbb{T} \subseteq \mathbb{R}^3, a \times \nu_r \mapsto \tau_\mathrm{a}$ exists, where the generalized applied force and torque vector is given by $\tau_\mathrm{a} = (X_\mathrm{a}, Y_\mathrm{a}, N_\mathrm{a})^\top$. This vector contains the applied force in the surge direction denoted by X_a, the applied force in the sway direction denoted by Y_a, and the applied torque in the yaw direction denoted by N_a. According to [11], the dynamics of a vessel $f = \left(\dot{\eta}^\top, \dot{\nu}_\mathrm{r}^\top, \dot{a}^\top\right)^\top$ is given by

$$\dot{\eta} = J(\psi)\nu_r + \dot{\eta}_\mathrm{c}(\eta), \tag{4a}$$

$$\begin{aligned} M\dot{\nu}_\mathrm{r} = {} & \tau_\mathrm{a}(a, \nu_r) + \tau_\mathrm{d} - M_\mathrm{RB}\dot{\nu}_\mathrm{c} \\ & - C_\mathrm{RB}(\nu_\mathrm{r} + \nu_\mathrm{c})(\nu_\mathrm{r} + \nu_\mathrm{c}) - N(\nu_\mathrm{r})\nu_\mathrm{r}, \end{aligned} \tag{4b}$$

$$\dot{a} = u, \tag{4c}$$

where the temporal dependencies are omitted for legibility, $J(\psi)$ denotes the rotation matrix due to ψ, $\dot{\eta}_\mathrm{c}$ denotes a current field, $M = M_\mathrm{A} + M_\mathrm{RB}$ denotes the invertible combined mass matrix given by the the sum of the added mass matrix denoted by M_A and the rigid body mass matrix denoted by M_RB, C_RB denotes the Coriolis matrix, the nonlinear hydrodynamic damping effects are collected in N, τ_d denotes a generalized torque and force vector of disturbances and unmodelled effects, $\nu_\mathrm{c} = J^{-1}(\psi)\dot{\eta}_\mathrm{c}(\eta)$ denotes the current in the body–fixed frame, and $\dot{\nu}_\mathrm{c}$ denotes the total time derivative

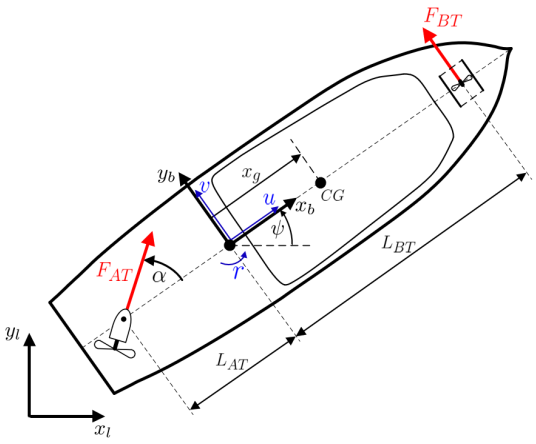

Fig. 3. Schematic drawing of the research vessel *Solgenia* with visualization of the local frame, the body–fixed frame, and the actuator configuration.

of ν_c. For more detailed information about the dynamic model of a vessel, the reader is referred to [11], [12], and [13].

D. Path Constraints

To define the path constraints (1e), we define the following terms for the docking process and translate them into mathematical expressions:

1) Stay on the river $\leftrightarrow \underline{y} \leq y_l(t) \leq \overline{y}$
2) Limit the relative velocity $\leftrightarrow \underline{\nu} \leq \nu_r(t) \leq \overline{\nu}$
3) Limit the actuator states $\leftrightarrow \underline{a} \leq a(t) \leq \overline{a}$
4) Limit the actuator dynamics $\leftrightarrow \underline{u} \leq u(t) \leq \overline{u}$

Based on these expressions the path constraints (1e) can be formulated as

$$h(x(t),u(t)) = \begin{pmatrix} \underline{y} - y_l(t) \\ y_l(t) - \overline{y} \\ \underline{\nu} - \nu_r(t) \\ \nu_r(t) - \overline{\nu} \\ \underline{a} - a(t) \\ a(t) - \overline{a} \\ \underline{u} - u(t) \\ u(t) - \overline{u} \end{pmatrix} \leq 0, \quad t \in [0,T], \tag{5}$$

where the underlined values denote the minimal values, the overlined values denote the maximum values, and every scalar inequality constraint is evaluated separately.

III. Methods

In the first part of this section, the OCP is transformed into an NLP using a direct multiple shooting approach. In the second part, a homotopy procedure is described to solve the NLP.

A. Direct Multiple Shooting Discretization

By employing direct multiple shooting [8], the original infinite–dimensional optimal control problem (1a)–(1e) is transformed into a finite–dimensional NLP by discretization. Therefore, we assume piecewise constant controls

$$u(t) = q_k, \quad t \in [t_k, t_{k+1}]$$

on each of $k = 0,\ldots,N-1$ shooting intervals with $q_k \in \mathbb{R}^{n_u}$, $t_0 = 0$, and $t_N = T$. The discrete time system dynamics is given by

$$s_{k+1} = F_k(s_k, q_k), \quad k = 0,\ldots,N-1$$

where $s_k \in \mathbb{R}^{n_x}$ with $k = 0,\ldots,N$ denote the states at the boundaries of the intervals. The corresponding discrete time costs can be denoted as $l_k(s_k,q_k)$. Note, that both, the discrete time dynamics F_k and the discrete time costs l_k contain numerical approximations of the integrals given in the OCP. Here we use $n \in \mathbb{N}$ steps of an explicit Runge–Kutta method of order four (RK4). The NLP arising from the OCP is stated as

$$\underset{s_0,\ldots,s_N,q_0,\ldots,q_{N-1}}{\text{minimize}} \quad \sum_{k=0}^{N-1} l_k(s_k,q_k) + E(s_N) \tag{6a}$$

$$\text{subject to} \quad s_0 - x_0 = 0, \tag{6b}$$

$$s_N - x_T = 0, \tag{6c}$$

$$s_{k+1} - F_k(s_k,q_k) = 0, \quad k = 0,\ldots,N-1, \tag{6d}$$

$$h_k(s_k,q_k) \leq 0, \quad k = 0,\ldots,N-1, \tag{6e}$$

that can be solved numerically starting from an initial guess for the state trajectory and the control trajectory. Because the NLP is in general nonlinear and non–convex, the performance of numerical optimization methods is strongly dependent on this initial guess. In the next subsection, a homotopy procedure is defined, which is a reliable heuristic for generating an initial guess to solve the NLP numerically.

B. Homotopy Procedure

According to [14], a homotopy parameter $\phi \in [0,1]^{n_\phi}$ is introduced. Based on this parameter, a generalization of (6a)-(6e) is defined as

$$\mathscr{H}(\phi) := \underset{w}{\text{minimize}} \quad \Phi_H(w,\phi) \tag{7a}$$

$$\text{subject to} \quad G_H(w,\phi) = 0, \tag{7b}$$

$$H_H(w,\phi) \leq 0, \tag{7c}$$

in such a way that $\mathscr{H}(0_{n_\phi \times 1})$ is equal to (6a)–(6e) and $\mathscr{H}(1_{n_\phi \times 1})$ is a problem much easier to solve. After initially solving $\mathscr{H}(1_{n_\phi \times 1})$, the homotopy parameter is reduced sequentially and the initial guess is warmstarted with the solution of the previously solved problem. This transforms the problem and its solution along a so-called homotopy

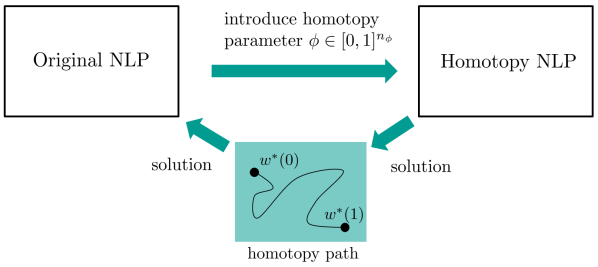

Fig. 4. Visualization of the homotopy procedure to solve an NLP.

Algorithm 1 Interior–Point–based Homotopy [14]

Input: w_0: Initial guess for $\mathscr{H}(1_{n_\phi \times 1})$;
$\gamma \in \mathbb{N}$: Number of steps on the homotopy path;
Output: w_f: Solution

$\phi \leftarrow 1_{n_\phi \times 1}$;
$w^{(0)} \leftarrow$ NLPSOLVER$(\mathscr{H}(\phi), w_0)$;
for $i \in \{1,2,...,n_\phi\}$ **do**
  $w^{(i)} \leftarrow w^{(i-1)}$;
  for $j \in \{1,2,...,\gamma\}$ **do**
    $\phi_i \leftarrow \phi_i - \frac{1}{\gamma}$;
    $w^{(i)} \leftarrow$ NLPSOLVER$(\mathscr{H}(\phi), w^{(i)})$;
  end for
end for
return $w_\mathrm{f} \leftarrow w^{(n_\phi)}$;

path. According to [15], it can be shown that, if the homotopy problem satisfies the linear independence constraint qualification (LICQ) and second–order sufficient conditions (SOSC) for all ϕ, there exists a unique and piecewise smooth homotopy path $w^*(\phi)$ between the optimal solutions $w^*(1_{n_\phi \times 1})$ and $w^*(0_{n_\phi \times 1})$. In Algorithm 1, a proceeding to numerically cover the homotopy path by γn_ϕ steps is described. The homotopy procedure is visualized in Figure 4. Note that introducing a multidimensional homotopy parameter with $n_\phi > 1$ creates the possibility to incorporate different nonlinearities in sequence. With this approach, even highly nonlinear and non–convex problems can be solved. For more detailed information about the homotopy procedure, the reader is referred to [14] and [15].

IV. Extension to Shrinking Horizon MPC

Due to unforeseen disturbances and unmodeled effects, a state–feedback controller is needed to perform a docking maneuver. E.g. in [11] and [12] various methods to track a predefined trajectory with an ASV are presented. In contrast, a shrinking horizon MPC scheme [1] is considered, which is a direct extension of the trajectory planning method presented in the last section. This method reacts to disturbances in an optimal way and reaches the desired docking position x_T at time $t = T$. The time–variant feedback law of the shrinking horizon MPC is implicitly given by the first element $q_i^*(x_i, i)$ of the solution $w^*(x_i, i) = (q_i^*, ..., q_{N-1}^*, s_i^*, ..., s_N^*)$ defined for $i = 0, 1, ..., N-1$ and given by

$$w^*(x_i,i) = \underset{q_i,\dots,q_{N-1},s_i,\dots,s_N}{\operatorname{argmin}} \sum_{k=i}^{N-1} l_k(s_k,q_k) + E(s_N) \tag{8a}$$

$$\text{subject to} \quad s_i - x_i = 0, \tag{8b}$$

$$s_N - x_T = 0, \tag{8c}$$

$$s_{k+1} - F_k(s_k,q_k) = 0, \quad k = i,\dots,N-1, \tag{8d}$$

$$h_k(s_k,q_k) \leq 0, \quad k = i,\dots,N-1. \tag{8e}$$

In the following, the NLP (8a)–(8e) is referred as $\mathscr{P}_\mathrm{MPC}(x_i, i, w_i)$, where the initial guess for numerical optimization is denoted by w_i. The NLP has to be solved in real–time. A key to reaching real–time capability using this control method is warm starting with the initial guess of the solution. According to [1], the initial guesses w_i with $i > 0$ can be warm started by a part of the previous solution w_{i-1}^*. The control method is described in Algorithm 2. Note that the first initial guess w_0 for Algorithm 2 can be computed using the interior–point–based homotopy given in Algorithm 1. In the following section, an application example is presented.

V. Application Example

In this section, the described methods are applied to dock the research vessel *Solgenia* in simulation. In the investigated scenario, the docking process must be terminated in a given time and shall require the lowest possible electrical energy. Note that this is a realistic scenario for autonomous water taxis [9] or autonomous ferries [10]. For reasons of simplicity, the fully actuated case, but no collision avoidance is considered. Data from real–world experiments is used to model the propeller characteristic curve. This characteristic is needed to calculate the required energy for the maneuvers. Examples of the physical implementation of controllers on the research vessel *Solgenia* including sensor fusion are given in previous work [12], [13].

A. Setup

First, the costs (1a), the initial constraint (1b), the terminal constraint (1c), the vessel dynamics (1d), and the path constraints (1e) are specified in the given application scenario.

1) Costs: The research vessel *Solgenia* has two propellers. One bow thruster (BT) with a fixed direction is located at the bow of the vessel and one turnable azimuth thruster (AT) with orientation α is located at the stern of the vessel. So the control vector contains $n_u = 3$ dimensions and is given by $u = (n_\mathrm{AT}, \alpha, n_\mathrm{BT})^\top$. This propulsion configuration is shown in Figure 3. According to [16], the torque of a propeller is proportional to its squared speed n. The mechanical power is given by a product of the torque and the propeller speed. These physical foundations motivate the use of a cubic model in the form of

$$P_\mathrm{model,BT} = \beta_\mathrm{BT} |n_\mathrm{BT}^3|, \quad P_\mathrm{model,AT} = \beta_\mathrm{AT} |n_\mathrm{AT}^3|,$$

where the scalar parameters β_BT and β_AT are dependent on the diameter and the shape of the propeller. A database $\mathbb{D}$ (2) with $K = 68359$ entries recorded in real–world experiments is used to specify the least squares problem (3). The power required to change the orientation of AT is negligible.

Algorithm 2 Shrinking horizon MPC

Input: w_0: Initial guess for state and input trajectory;
for $i \in 0,...,N-1$ **do**
  $x_i \leftarrow$ GetStateEstimate();
  $w^*(x_i, i) \leftarrow$ NLPSOLVER$(\mathscr{P}_\mathrm{MPC}(x_i, i, w_i))$;
  SendToActuators(q_i^*);
  $w_{i+1} \leftarrow (q_{i+1}^*, ..., q_{N-1}^*, s_{i+1}^*, ..., s_N^*)$;
end for
Notification: Docking process finished;

TABLE I
PARAMETERS SPECIFYING THE APPLICATION SCENARIO – VALUES WITHOUT DECLARED UNITS ARE GIVEN IN SI UNITS

Parameter	Value	Parameter	Value
β_{AT}	97.6 mW/Hz^3	Δt	1
β_{BT}	6.25 mW/Hz^3	T	120
v_{river}	1.1	N	120
R	50	Q_{H}	diag(1,10,10)
x_0	$(0,-45,\pi/2,0_{1\times6})^\top$	γ	10
x_T	$(0,45,\pi/2,0_{1\times6})^\top$	n_u	3
$\underline{y}$	-50	$\overline{y}$	50
$\underline{v}$	$(0,-0.5,-1/9\pi)^\top$	$\overline{v}$	$(1.5,0.5,1/9\pi)^\top$
$\underline{a}$	$(-37.5,-\pi,-66.7)^\top$	$\overline{a}$	$(37.5,\pi,66.7)^\top$
$\underline{u}$	$(-12.5,-\pi/4,-33.3)^\top$	$\overline{u}$	$(12.5,\pi/4,33.3)^\top$
ξ	$-(0_{1\times5},560,400,933,560,400,933,70,38,50,70,38,50)^\top$		

Therefore, the running costs are given by

$$L(x(t),u(t)) = \beta_{\mathrm{BT}}|n_{\mathrm{BT}}^3| + \beta_{\mathrm{AT}}|n_{\mathrm{AT}}^3| \tag{9}$$

as the sum of the AT's power and BT's power. While the identified model parameters are listed in Table I, the corresponding model characteristic is visualized in Figure 2. Due to the terminal constraint $x(T)$ is fixed and $E(x(T)) = 0$ is chosen. By choosing this model, a constant efficiency to transform electrical energy into mechanical energy is assumed. More detailed hydrodynamic effects on the propeller power models are neglected due to the low velocities of the vessel during the docking maneuver [16].

2) Dynamics: The general 3–DOF dynamics of the vessel is described in Section II *C*. While for different vessels and scenarios most parts of the dynamics model only change quantitatively via the parameters, the current field $\dot{\eta}_{\mathrm{c}}(\eta)$ used in (4a) and the actuator model $\tau_{\mathrm{a}}(a,\nu_{\mathrm{r}})$ used in (4b) have to be adapted qualitatively. To model complex river streams partial differential equations (PDE) can be stated and solved numerically [17]. However, the definition of reasonable assumptions for those models is a hard task. Instead, we want to fit a model to real–world measurements. We measured the current v_{river} in the middle of the river in a real–world experiment and define the parabolic model

$$\dot{\eta}_{\mathrm{c}}(\eta) = \begin{pmatrix} v_{\mathrm{river}}\left(1-\frac{\zeta(\eta)^2}{R^2}\right) \\ 0 \\ 0 \end{pmatrix},$$

where $\zeta(\eta)$ denotes the distance to the middle of the river, and the width of the river is given by $2R$. Using this model, no current in the y_{I}–direction and no rotation is assumed. Further, we assume $\ddot{\eta}_{\mathrm{c}} = 0$ due to low velocities in the docking process. For a detailed description of the research vessel *Solgenia*, including the dynamic propulsion model, the reader is referred to [13].

3) Conditions and Path Constraints: The initial constraint (1b) determines the initial state where the docking process is starting. The terminal constraint (1c) specifies the target state as the docking position with zero velocity. All specific parameters are listed in Table I.

4) Application of Direct Multiple Shooting: For the discretization a constant time period $\Delta t = T/N$ for each discretization interval (III-A) is defined. For the numerical approximations of the dynamics the RK4 method with $n = 2$ immediate steps is used. The path constraints are evaluated at the time interval boundaries.

5) Details of the Homotopy Procedure: To define the homotopy NLP $\mathscr{H}(\phi)$ for the application, we define a three–dimensional homotopy parameter $\phi = (\phi_1,\phi_2,\phi_3)^\top$ with $\phi_1,\phi_2,\phi_3 \in [0,1]$. The first homotopy parameter ϕ_1 is used to generalize the dynamics (4b) in the form

$$M\dot{\nu}_{r,\mathrm{H}} = -C_{\mathrm{RB}}(\nu)\nu - N(\nu_{\mathrm{r}})\nu_{\mathrm{r}} + (1-\phi_1)\tau_{\mathrm{a}}(a,\nu_r) + \phi_1 Tu + \tau_{\mathrm{d}},$$

where $\nu = \nu_{\mathrm{r}} + \nu_{\mathrm{c}}$, $T = \mathrm{diag}(1\ \mathrm{N/Hz}, 1\ \mathrm{N/rad}, 1\ \mathrm{Nm/Hz})$ is the unity matrix to transform the units, and ϕ_1 determines the influence of the nonlinear actuator model. For $\phi_1 = 0$ the original dynamics (4b) is given. For $\phi_1 = 1$ the values of u replace the generalized applied force and torque vector τ_a. The second homotopy parameter ϕ_2 is used to generalize the costs (9) in the form

$$L_{\mathrm{H}}(x,u) = (1-\phi_2)\left(\beta_{\mathrm{BT}}|n_{\mathrm{BT}}^3| + \beta_{\mathrm{AT}}|n_{\mathrm{AT}}^3|\right) + \phi_2 a^\top Q_{\mathrm{H}} a,$$

where ϕ_2 determines if the original costs are used ($\phi_2 = 0$) or for $\phi_2 = 1$ a quadratic cost function with positive definite matrix $Q_{\mathrm{H}} \in \mathbb{R}^{n_a \times n_a}$ is used. Finally, the third homotopy parameter ϕ_3 is employed to generalize the inequality constraints (5) resulting in

$$h_{\mathrm{H}}(x,u) = h(x,u) + \phi_3 \xi,$$

where $\xi \in \mathbb{R}^{n_h}$ is a parameter chosen in an appropriate way to relax the inequality constraints. All parameters used to specify the application scenario are listed in Table I. The algorithm is implemented via the `MATLAB` interface of `CasADi` [18] and the NLPs are solved with `IPOPT` [19].

B. Results

Using the described homotopy procedure, `IPOPT` finds a feasible solution to the given problem in various tested scenarios. It is important to note that if a zero–initialization is used, no feasible solution can be found without the homotopy procedure. Therefore, no direct comparison with and without the usage of the homotopy procedure is reasonable. The optimization was performed with four different settings $v_{\mathrm{river}} = \{0,\ 0.5,\ 1.1,\ 1.5\}$ m/s. The exemplary actuator trajectories

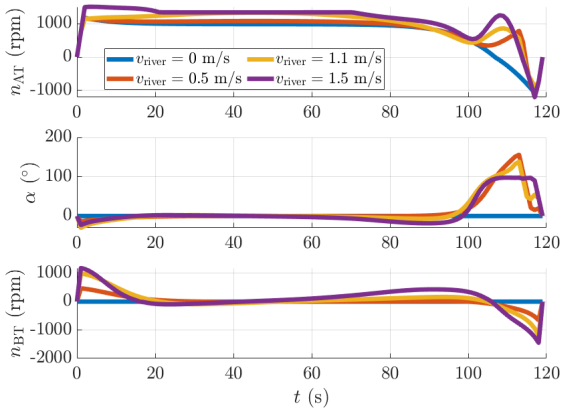

Fig. 5. Optimal actuator trajectories for the different current settings.

TABLE II

ENERGY CONSUMPTION OF DIFFERENT SCENARIOS AND METHODS

v_{river}	Method A	Method B	Method C	Method C (dist.)
0 m/s	43.27 kJ	43.27 kJ	43.27 kJ	64.43 kJ
0.5 m/s	177.7 kJ	56.44 kJ	51.35 kJ	80.01 kJ
1.1 m/s	infeasible	90.22 kJ	80.03 kJ	133.3 kJ
1.5 m/s	infeasible	infeasible	107.12 kJ	infeasible

are shown in Figure 5 . While for $v_{\text{river}} = 0$, the energy–optimal trajectory is a straight line, in the other cases the energy–optimal trajectories have nontrivial shapes. Examples of different scenarios are shown in Figure 1 . The failure of the actuators visualized in the right plot of Figure 1 is modeled by

$$\tau_{\text{d}}(t) = \begin{cases} -\tau_{\text{a}}(a(t), v_r(t)), & \text{for } t \in [20, 40] \text{ s} \\ (0 \text{ N}, 0 \text{ N}, 0 \text{ Nm})^\top, & \text{else} \end{cases} . \quad (10)$$

The question arises of how much energy is saved using the described approach. To answer this question we compare three methods with $\tau_{\text{d}}(t) = 0$:

1) *Method A*: Path–velocity decomposition defined as a straight line with orientation tangential to the path.
2) *Method B*: Path–velocity decomposition defined as a straight line with optimized orientation.
3) *Method C*: The paper's method.

In the experiments labeled *Method C* (dist.), we apply *Method C* to the disturbance scenario given in (10). *Method A* and *Method B* are not applicable in this case. *Method A* was first presented in [20] and is recently used e.g. see [5]. *Method B* is a generalization of *Method A* that requires high–dimensional nonlinear optimization. The required energy amounts are listed in Table II . The solutions coincide in case of no current. In all considered scenarios, *Method C* requires the least energy. In the case of non–zero current, *Method A* requires a lot of energy or is not applicable in cases with high currents. It is important to note that *Method A*'s performance can be improved by optimizing the path for example using heuristics or human experts [5]. In contrast, *Method B* requires only slightly more energy than *Method C* even under current conditions, and might be regarded as a compromise between optimality and planning certainty of the geometric path. The *Method C* (dist.) experiments show the ability of the shrinking horizon MPC scheme to reach the docking position in time even in the presence of unforeseen or misestimated disturbances along the way.

VI. CONCLUSION AND FUTURE WORK

This paper shows the application of direct multiple shooting with a homotopy procedure to an energy–optimal docking task for the research vessel *Solgenia*. A database recorded in real–world experiments was used within a gray box model of the propellers' power consumption. The numerical experiments show the significance of optimizing the vessel's orientation even if the path of the center of mass is fixed. In still water conditions all methods compared in Section V require the same amount of energy. In the presence of current fields, the least energy is required if the path is optimized as well. Experiments with temporarily failed actuators show the ability of shrinking horizon MPC to reject disturbances in an energy–optimal way. In future work, it would be interesting to improve the model of the propeller characteristic and the river current using recent data within learning algorithms. Further, we aim to investigate and evaluate the real–time capability of the presented methods in real–world experiments including collision avoidance scenarios.

REFERENCES

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

[2] A. V. Rao, "Trajectory Optimization: A Survey," in *Lecture Notes in Control and Information Sciences*, 455, 2014. DOI: 10.1007/978-3-319-05371-4

[3] R. Damerius, J. R. Marx, T. Jeinsch, "Fast Trajectory Generation on a Path using Feedback Linearization," in *Proceedings of the IFAC World Congress*, 2023.

[4] M. Lutz, T. Meurer, "Optimal Trajectory Planning and Model Predictive Control of Underactuated Marine Surface Vessels using a Flatness-Based Approach," in *Proceedings of the American Control Conference*, 2021, pp. 4667–4673.

[5] M. Kosch, P. Koschorrek, and D. Abel, "A reference trajectory-based approach to safe and efficient trajectory planning for an overactuated river ferry," *IFAC-PapersOnLine*, vol. 55, no. 31, pp. 31–36, 2022.

[6] A. B. Martinsen, A. M. Lekkas, S. Gros, "Autonomous docking using direct optimal control," *IFAC-PapersOnLine*, vol. 52, no. 21, pp. 97–102, 2019.

[7] G. Bitar, M. Breivik, A. M. Lekkas, "Energy-optimized path planning for autonomous ferries," *IFAC-PapersOnLine*, vol. 51, no. 29, pp. 389–394, 2018.

[8] H. G. Bock and K. J. Plitt, "A multiple shooting algorithm for direct solution of optimal control problems," in *Proceedings of the IFAC World Congress*, 1984, pp. 242–247.

[9] W. Wang et al., D. Fernández-Gutiérrez, R. Doornbusch, J. Jordan, T. Shan, P. Leoni, N. Hagemann, J. Schiphorst, F. Duarte, C. Ratti, D. Rus, "Roboat III: An autonomous surface vessel for urban transportation," *Journal of Field Robotics*, 2023. DOI: 10.1002/rob.22237

[10] E. F. Brekke et al., "milliAmpere: An Autonomous Ferry Prototype", *Journal of Physics: Conference Series,* 2022, 2311 012029. DOI: 10.1088/1742-6596/2311/1/012029

[11] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, John Wiley & Sons, Ltd, 2011.

[12] S. Wirtensohn, O. Hamburger, H. Homburger, L. M. Kinyo, J. Reuter, "Comparison of Advanced Control Strategies for Automated Docking," *IFAC-PapersOnLine*, vol. 54, no. 16, pp. 295–300, 2021.

[13] H. Homburger, S. Wirtensohn, M. Diehl and J. Reuter, "Feature-Based MPPI Control with Applications to Maritime Systems," *Machines*, 10(10):900. 2022. DOI: 10.3390/machines10100900

[14] J. de Schutter, R. Leuthold, T. Bronnenmeyer, E. Malz, S. Gros, M. Diehl, "AWEbox: An Optimal Control Framework for Single- and Multi-Aircraft Airborne Wind Energy Systems," *Energies*, 16, 1900, 2023. DOI: 10.3390/en16041900

[15] P. Deuflhard, *Newton Methods for Nonlinear Problems: Affine Invariance and Adaptive Algorithms*, Springer: Berlin/Heidelberg, Germany, vol. 35, 2011.

[16] J. Carlton, *Marine propellers and propulsion*, Elsevier Ltd., Burlington, USA, second edition, 2007.

[17] V. Churuksavea, A. Starchenko, "Mathematical modeling of a river stream based on a shallow water approach," in *Procedia Computer Science*, 66, pp. 200-209, 2015. DOI: 10.1016/j.procs.2015.11.024

[18] 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*, vol. 11, no. 1, pp. 1–36, 2019.

[19] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[20] K. Kant, S. W. Zucker, "Toward efficient trajectory planning: the path-velocity decomposition," *International Journal of Robotics Research,* 1986, vol. 5 (3), pp. 72-89.