2016 American Control Conference (ACC)
Boston Marriott Copley Place
July 6-8, 2016. Boston, MA, USA

Path-Following Control of an AUV using Multi-Objective Model Predictive Control

Chao Shen[1], *Student Member, IEEE*, Yang Shi[1], *Senior Member, IEEE*, Brad Buckham[1]

***Abstract*— The path-following (PF) problem of an autonomous underwater vehicle (AUV) is studied, in which the speed profile of the vehicle is taken into consideration as a secondary task. A multi-objective model predictive control (MO-MPC) framework is developed attempting to accommodate the prioritized tasks in PF. To solve the MO-MPC problem, we adopt the weighted sum method with the introduction of a logistic function that automatically selects the appropriate weights for each objective function. Pontryagin minimum principle (PMP) is subsequently applied for the implementation. Simulations using identified hydrodynamic coefficients of the Saab SeaEye Falcon open-frame ROV/AUV are carried out, which validates the effectiveness of the proposed MO-MPC path-following control.**

I. Introduction

Dramatic progress in underwater robotics provides advanced tools for various oceanic missions. The autonomous underwater vehicle (AUV) is a representative that exhibits a broader scope of potential applications while less operational cost comparing to those having human involved. At the same time, the high level of autonomy justifies a complicated control design procedure due to the nonlinear, coupled hydrodynamics of the AUV [1], [2].

Being one of the basic motion control problems, path-following (PF) control has been extensively studied. The work reported in [3] presents a powerful PF control structure for land robots. The control structure, however, only depends on the kinematic model of the robot, while the nonlinear, coupled dynamics has to be considered for AUVs, thus the technique cannot be directly borrowed. In [4], the authors use the PF control structure and design the dynamics controller from the kinematics controller via backstepping technique. [5] considers more practical scenarios extending the PF control design for AUVs subject to ocean currents. The AUV PF controllers in [4], [5] are designed based on the choice of target point to be the closest point on the path relative to the vehicle, hence inherit the big limitation that there exist singularities for certain initial positions of the vehicle, located exactly at the center of path curvature. To solve this problem, [6] proposes a new PF control strategy for AUVs, in which the target point is modeled as a virtual moving point having its own dynamics. The new strategy essentially creates an extra degree of freedom (DOF) that can be utilized to achieve other specifications such as a desired surge velocity. In [7], the authors follow this strategy and develop a robust PF controller by involving an adaption scheme to deal with the parameter uncertainty of the AUV. Nevertheless, the aforementioned PF control designs lack the capability of handling the constraints on state or input, which invokes recent studies of model predictive control (MPC) for PF problems [8]–[10].

MPC is a well-known time domain control strategy, calculating the control input by solving finite horizon open-loop optimal control problems in a receding horizon fashion [11]. Because of its optimization essential, MPC provides a rather flexible framework to digest complicated system dynamics, and to incorporate intractable constraints. These salient features attract many theorists and practice experts to study MPC for all kinds of control problems [12]–[16]. For the PF problem, there exist several pioneer work in the literature. [17] studies the PF problem for marine vessels, in which the MPC is used to optimize the lookahead distance in the line-of-sight (LOS) guidance law. Similar results can be found in [18], where, differently, the linear MPC structure is exploited for the LOS-based PF control rather than just an “add-on” in the controller design. But the LOS guidance law can only be used for PF of straight lines (way-point routes). For PF of curves, the authors of [9] formulate the PF control problem with an augmented system consisting of the vehicle dynamics and the path dynamics, and sufficient stabilizing conditions are derived. An extended work in terms of output MPC version is presented in [19]. The PF problems addressed in [9] and [19] only require a nonzero forward motion of the path dynamics but not the possible consideration of the speed assignment. Therefore, the authors of [20] incorporate the speed assignment into the MPC PF formulation for underactuated vehicles. [8] takes another approach that transforms the PF problem into a regulation problem where the initial state of the path dynamics is also viewed as a decision variable. MPC-based PF applications to industrial robots and tower cranes can be found in [21] and [10], respectively.

Recently, multi-objective model predictive control (MO-MPC) becomes a new focus in system control community [22], [23]. MO-MPC enlarges the capacity of MPC that can deal with vector-valued cost function in the optimal control problems, which exhibits a more attractive and more powerful framework. In this paper, we propose a novel MO-MPC framework for the PF problem of a fully actuated AUV, in which the secondary task of speed assignment can be explicitly incorporated. The target path is defined in the output space of the vehicle. As proposed in [6], we

*This work was supported by the Natural Science and Engineering Research Council of Canada and the Canadian Foundation of Innovation.

[1]Chao Shen, Yang Shi and Brad Buckham are with the Department of Mechanical Engineering, University of Victoria, PO Box 3055, STN CSC, Victoria, BC, V8W 3P6, Canada shenchao@uvic.ca; yshi@uvic.ca; bbuckham@uvic.ca

Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on July 04,2024 at 03:23:37 UTC from IEEE Xplore. Restrictions apply.

view the path as the trajectory of a moving point controlled by its own dynamics. Then, an explicit parametrization of the zero-path-error manifold is found in the state space, which facilitates the construction of reference state for the AUV. All the PF requirements are formulated into the MO-MPC framework where two distinct optimal control problems are identified for the augmented system. Regarding the prioritized requirements, the weighted sum method is adopted with an introduced logistic function that chooses the appropriate weight according to the system state. The Pontryagin minimum principle is applied so that expensive operations of successive linearizations can be circumvented.

The main contributions of this paper are three-fold:

- An MO-MPC framework is developed for the AUV PF control, in which the secondary task of speed assignment can be explicitly considered.
- The weighted sum method is proposed with a novel logistic function to deal with the different priorities of the requirements in PF problem.
- With several properly defined barrier functions, the Pontryagin minimum principle is applied for the implementation of the MO-MPC PF control.

The paper is organized as follows. Section II introduces the mathematical model that is used for the PF control. In Section III, the detailed MO-MPC PF control is depicted. In Section IV, we present the simulation results. And the closed-loop stability issue is briefly discussed in Section V. Section VI concludes the paper.

Notations: column operation $[\rho_1^{\mathrm{T}}, ...\rho_n^{\mathrm{T}}]^{\mathrm{T}}$ is denoted by $col(\rho_1, ..., \rho_n)$; diagonal operation is abbreviated as $\mathrm{diag}(\cdot)$; and weighted norm $\sqrt{\rho^T A \rho}$ is denoted by $\|\rho\|_A$.

II. AUV System Modeling

The dynamic equations are derived via Newtonian mechanics [1]

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{C}(\mathbf{v})\mathbf{v} + \mathbf{D}(\mathbf{v})\mathbf{v} + \mathbf{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} \tag{1}$$

where $\mathbf{v}$ denotes the velocities and $\boldsymbol{\eta}$ is the position and orientation. $\mathbf{M}$, $\mathbf{C}(\mathbf{v})$ and $\mathbf{D}(\mathbf{v})$ are system matrices. $\mathbf{g}(\boldsymbol{\eta})$ denotes the restoring forces and $\boldsymbol{\tau}$ is the propulsion forces.

Three mild assumptions are introduced: (i) the AUV is with three planes of symmetry; (ii) the mass distribution is homogeneous; (iii) the pitch and roll motions are neglectable. Then the detailed dynamic equations of motion are simplified

$$\begin{aligned}
\dot{u} &= \frac{M_y}{M_x} vr - \frac{X_u}{M_x} u - \frac{D_u}{M_x} u|u| + \frac{F_u}{M_x} \\
\dot{v} &= -\frac{M_x}{M_y} ur - \frac{Y_v}{M_y} v - \frac{D_v}{M_y} v|v| + \frac{F_v}{M_y} \\
\dot{r} &= \frac{M_x - M_y}{M_\psi} uv - \frac{N_r}{M_\psi} r - \frac{D_r}{M_\psi} r|r| + \frac{F_r}{M_\psi}
\end{aligned} \tag{2}$$

where $\mathbf{v} = [u, v, r]^{\mathrm{T}}$ denotes the surge, sway and yaw velocities; $\boldsymbol{\tau} = [F_u, F_v, F_r]^{\mathrm{T}}$ represents thrust forces and moments; M_x, M_y and M_ψ are inertial coefficients, while X_u, Y_v, N_r and D_u, D_v, D_r are hydrodynamic coefficients.

We also have the kinematic equations:

$$\begin{aligned}
\dot{x} &= u\cos\psi - v\sin\psi \\
\dot{y} &= u\sin\psi + v\cos\psi \\
\dot{\psi} &= r
\end{aligned} \tag{3}$$

where $\boldsymbol{\eta} = [x, y, \psi]^{\mathrm{T}}$ denotes the global position and orientation. Define the system state $\mathbf{x} = col(\boldsymbol{\eta}, \mathbf{v})$ and the control input $\mathbf{u} = \boldsymbol{\tau}$, and simplify the expression of (3) by $\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\mathbf{v}$. Then we have the control system dynamics

$$\dot{\mathbf{x}} = \left[\begin{array}{c} \mathbf{J}(\boldsymbol{\eta})\mathbf{v} \\ \mathbf{M}^{-1}(\mathbf{u} - \mathbf{C}\mathbf{v} - \mathbf{D}\mathbf{v} - \mathbf{g}) \end{array} \right] \triangleq f(\mathbf{x}, \mathbf{u}) \tag{4}$$

Since we aim to design an MPC-based PF controller the built model (4) has to be discretized when implementing the MPC algorithm.

III. Model Predictive Path-Following Control

A. The Path-Following Problem

In the following, we consider the PF problem. Given a mapping which describes a geometric path

$$\mathcal{P} = \{\bar{p} \in \mathbb{R}^d \mid \bar{p} = p(s), s \in [s_0, s_1]\} \tag{5}$$

in a d-dimensional output space. The scalar variable s is referred to the path parameter taking values on the domain of $[s_0, s_1]$. The domain should be closed but may be unbounded, i.e., $s_1 = +\infty$. And the mapping p is often assumed sufficiently smooth, i.e., continuously differentiable with enough order of derivatives. Then, the path $\mathcal{P}$ can be interpreted as the output of the path dynamics governed by

$$\dot{s} = g(s, v_s), \ \bar{p} = p(s) \tag{6}$$

where v_s is the imaginary path control input.

Alternatively, the desired path $\mathcal{P}$ might also be defined in the state space by the mapping

$$s \mapsto \bar{p}_{\mathbf{x}}(s) \in \mathbb{R}^n, \ s \in [s_0, s_1] \subset \mathbb{R} \tag{7}$$

Usually the components of $\bar{p}_{\mathbf{x}}(s)$ are coupled according to the dynamics of the controlled system, hence cannot be freely chosen. More precisely, every output path (5) implicitly defines a so-called zero-path-error manifold [24]

$$s \mapsto p_{\mathbf{x}}(s, \dot{s}, \ddot{s}, ...) \in \mathbb{R}^n, \ s \in [s_0, s_1] \subset \mathbb{R} \tag{8}$$

in the state space. Since most of existing stability results on MPC consider to penalize the full state through the cost function [11], we particularly take advantage of (8) when formulating the MPC problem.

The PF problem is to determine the control inputs $\mathbf{u}(t)$ such that the following requirements are met:

- Path convergence. The system states converge to the zero-path-error manifold:

$$\lim_{t\to\infty} \|\mathbf{x}(t) - p_{\mathbf{x}}(s(t), \dot{s}(t), ...)\| = 0 \tag{9}$$

- Forward motion. The system states follow the path in the direction of $\dot{s} > 0$.
- Speed assignment. When moving along the path, the desired speed profile is pursued: $\dot{s} \to \dot{s}_r$.
- Constraint satisfaction. The constraints on system state $\mathbf{x}$ and control input $\mathbf{u}$ are always satisfied.

Note that, the third requirement "speed assignment" does not have to be always considered in PF, c.f., [9], [10]. For AUV applications, since there do exist an extra DOF

Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on July 04,2024 at 03:23:37 UTC from IEEE Xplore. Restrictions apply.

that can be utilized, a desired speed profile is preferred. However, distinct from trajectory tracking problems [13], the "path convergence" should be identified as the primary task and of the most importance, and "speed assignment" is the secondary task which can be sacrificed for the primary one.

B. Zero-Path-Error Manifold

In general, finding an explicit parametrization of the zero-path-error manifold (8) could be difficult. However, for the AUV system, each state has its physical meanings, which provides a guideline for the parametrization. Consider a desired path defined in the output space

$$\mathcal{P} = \{\bar{p} \in \mathbb{R}^2 \mid \bar{p} = [p_x(s), p_y(s)]^{\mathrm{T}},\ s \in [s_0, s_1]\} \tag{10}$$

Regarding the kinematic equations (3), we additionally choose the AUV surge velocity u to always be the tangential velocity of the path, then the zero-path-error manifold can be explicitly parameterized in the following way

$$p_{\mathbf{x}}(s,\dot{s}) = [p_x(s), p_y(s), p_\psi(s), p_u(s,\dot{s}), p_v(s), p_r(s,\dot{s})]^{\mathrm{T}}$$

with

$$p_\psi(s) = \mathrm{atan2}(\frac{\partial p_y}{\partial s}, \frac{\partial p_x}{\partial s}) \tag{11a}$$

$$p_u(s,\dot{s}) = \sqrt{(\frac{\partial p_x}{\partial s})^2 + (\frac{\partial p_y}{\partial s})^2}\dot{s} \tag{11b}$$

$$p_v(s) = 0 \tag{11c}$$

$$p_r(s,\dot{s}) = \frac{\frac{\partial p_x}{\partial s}\frac{\partial^2 p_y}{\partial s^2} - \frac{\partial p_y}{\partial s}\frac{\partial^2 p_x}{\partial s^2}}{(\frac{\partial p_x}{\partial s})^2 + (\frac{\partial p_y}{\partial s})^2}\dot{s} \tag{11d}$$

In fact, (11) does not cover the whole manifold implicitly defined by the path (10) since we have additionally confined the surge velocity of the AUV. Rather, (11) defines a subset of the zero-path-error manifold which is easy to be parameterized by the path parameter. However, this is important because the output path (10) only defines the desired position of the vehicle and leaves the yaw angle a DOF, while to fit into the MPC framework, the reference states have to be uniquely determined.

Notice that, in the manifold (11), only the first order derivative of the path parameter is included to calculate the reference state for the AUV. Therefore, we could choose the path dynamics to be

$$\dot{s} = g(s, v_s) = v_s \tag{12}$$

Actually, this choice also potentially facilitates the implementation of MPC algorithm. When we augment the AUV control system (4) with the path dynamics (12), the "forward motion" requirement will be formulated as inequality constraints on the input rather than on the state, for which case indirect MPC algorithms based on Pontryagin minimum principle can also be used [25], [26].

C. Multi-objective MPC Formulation

The AUV PF problem mainly contain two aspects, namely, path convergence and speed assignment, and each has its own focus. The path convergence requires the AUV to converge to $\mathcal{P}$ as fast as possible, while the speed assignment requires the path parameter to move in a preferred pace. Since $\dot{s}_r$ is known, stringent fulfilment of the speed assignment results in the determined reference states $p_{\mathbf{x}}(s_r, \dot{s}_r)$. This substantially degrades the PF problem to a trajectory tracking problem, which is not always compatible with the fast convergence requirement.

To solve the conflicting requirements, the MO-MPC framework is investigated for the PF problem. Consider the following multi-objective optimization problem:

$$\min_{U}.\quad J(U,\xi) \tag{13a}$$

$$\begin{aligned} &\text{subject to} \\ &\xi_{k+1} = h_d(\xi_k, \omega_k),\ \xi_0 = \xi \\ &\xi_k \in \Xi,\ k = 1,2,...,N \\ &\omega_k \in \Omega, k = 0,1,...,N-1 \\ &\xi_N \in \Pi \end{aligned} \tag{13b}$$

with

$$\xi = \begin{bmatrix} \mathbf{x} \\ s \end{bmatrix}, \omega = \begin{bmatrix} \mathbf{u} \\ v_s \end{bmatrix}, h_d(\xi,\omega) = \begin{bmatrix} f_d(\mathbf{x},\mathbf{u}) \\ g_d(s, v_s) \end{bmatrix} \tag{14}$$

Here, (14) describes the discretized augmented system; $J(U,\xi) = [J_1(U,\xi), J_2(U,\xi)]^{\mathrm{T}}$ is a vector-valued objective function constructed for the two conflicting PF requirements; $U = col(\omega_0, \omega_1, ..., \omega_{N-1})$ is the sequence of control inputs to be optimized; ξ_k denotes the k-step predicted state from initial condition $\xi_0 = \xi$; Ξ, Ω and Π represent the constraints on state, input and terminal state, respectively. And each objective function has the following form:

$$J_i(U,\xi) = \sum_{k=0}^{N-1} L_i(\xi_k, \omega_k) + E_i(\xi_N) \tag{15}$$

where

$$L_1(\xi,\omega) = \|\xi - \xi_p\|_{Q_1}^2 + \|\omega\|_{R_1}^2,\ E_1(\xi) = \|\xi - \xi_p\|_{P_1}^2 \tag{16a}$$

$$L_2(\xi,\omega) = \|\xi - \xi_t\|_{Q_2}^2 + \|\omega\|_{R_2}^2,\ E_2(\xi) = \|\xi - \xi_t\|_{P_2}^2 \tag{16b}$$

Here, $\xi_p = col(p_{\mathbf{x}}(s), s)$ can be viewed as the reference for the "path convergence" while $\xi_t = col(p_{\mathbf{x}}(s_r), s_r)$ is the reference for "speed assignment" with s_r generated by $s_{r,k+1} = g_d(s_{r,k}, v_{sr,k})$ from $s_{r,0} = s_0$; Q_i, R_i and P_i are weighting matrices, positive definite.

Since $J_i(U,\xi)$ are in general conflicting with each other, it is not possible to optimize both of them simultaneously. Instead, we can take advantage of the notion of Pareto optimality [27].

Definition 1: Let $\bar{U}$ be a feasible point for (13). Then, $\bar{U}$ is said to be Pareto optimal if there is no other feasible point U such that $J_i(U,\xi) \leq J_i(\bar{U},\xi)$ for all i, and $J_i(U,\xi) < J_i(\bar{U},\xi)$ for at least one i.

By definition, we shall notice that there usually exist more than one Pareto optimal points. More specifically, a point dominated by another point is not Pareto optimal, and the remaining points constitute the so-called Pareto frontier. The complete solution of (13) is to determine the Pareto frontier using evolutionary methods [27] or classical methods [28].

Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on July 04,2024 at 03:23:37 UTC from IEEE Xplore. Restrictions apply.

However, for the PF problem, it is unnecessary to calculate the whole frontier. What we need is just to find one preferred point on the Pareto frontier. In this regard, the weighted sum method [28] is adopted.

D. Weighted Sum Method with a Logistic Function

The weighted sum method scalarizes the vector-valued objective function by assigning a weight to each objective. Instead of solving (13), the weighted sum method solves the following single objective problem:

$$\begin{aligned} \min_{U}. \quad & a^{\mathrm{T}} J(U, \xi) \\ \text{s.t.} \quad & (13\text{b}) \end{aligned} \tag{17}$$

where $a = [\alpha, 1-\alpha]^{\mathrm{T}}$ with $0 < \alpha < 1$ is the weight vector.

Here list several comments for the weighted sum method. First, there is a well-known theorem [28] which proves the solution U^* of (17) to be always Pareto optimal for (13) without any further assumptions. This essentially provides a solid theoretical support for use of the weighted sum method. Second, by scalarizing the original problem, we can choose from a wide class of optimization algorithms, namely, the descent methods [29], to efficiently solve the PF problem. Besides, recent developments of fast MPC algorithms are exclusively designed for single objective optimizations [25]. Hence the use of the weighted sum method serves as a link between MO-MPC and a vast majority of results for single objective MPC. Third, through the use of the weighted sum method we can incorporate some priori knowledge into the MO-MPC formulation by selecting preferred weights for each objective.

We propose a method to choose the appropriate weight α at each sampling instant. Consider a logistic function:

$$\alpha(Er) = \frac{1}{1 + e^{-\beta Er}} \tag{18}$$

where $Er = \|\boldsymbol{\eta} - \boldsymbol{\eta}_p\|_A$ with $\boldsymbol{\eta}_p = [p_x(s), p_y(s), p_\psi(s)]^{\mathrm{T}}$, serves as an indicator of path convergence; $\beta > 0$ controls the change rate of the function; A is a weighting matrix, positive definite. The logistic function (18) is smooth, monotonic having the range of $[0.5, 1)$ corresponding to its domain $[0, +\infty)$. Since the path convergence is more important than the speed assignment, (18) seems an appropriate choice.

The MPC-based AUV path-following control procedure can be briefly described as follows: (i) at sampling instant t, evaluate the value of (18) using $Er(t)$, then solve (17) with current state of the augmented system $\xi = \xi(t)$; (ii) let $U^* = col(\omega_0^*, ..., \omega_{N-1}^*)$ denote the solution, then implement the first element ω_0^* to the augmented system for one sampling period; (iii) at $t+\Delta t$, new state measurement $\xi(t+\Delta t)$ is fed back, then evaluate (18) and solve (17) substituting t with $t + \Delta t$. Repeat from (ii).

IV. Simulation Results

In this section, we present the simulation results of an AUV to follow a sinusoidal path $p_x(s) = s$ and $p_y(s) = \sin(s)$ with $s \geq 0$ in the local level plane. And for the speed assignment, instead of directly setting a preferred path velocity v_{sr}, we set a desired surge velocity $u_r = 1$ [m/sec] for the AUV. Then, according to (11b) and (12), we have the explicit expression

$$v_{sr} = ((\frac{\partial p_x}{\partial s})^2 + (\frac{\partial p_y}{\partial s})^2)^{-\frac{1}{2}} \tag{19}$$

to calculate ξ_t at each sampling instant.

A. Implementing with Pontryagin Minimum Principle

Since we select the path dynamics to be a single integrator (12), the "forward motion" requirement can be formulated as constraints on the input of the augmented system. Together with AUV thrust limits, we have the input constraint set Ω polyhedral:

$$\Omega = \{\omega \in \mathbb{R}^4 |\ \underline{\omega}(j) \leq \omega(j) \leq \bar{\omega}(j),\ j = 1, ..., 4\} \tag{20}$$

where $\underline{\omega}(j)$ and $\bar{\omega}(j)$ represent the lower and upper bounds of the jth component. And also there exists no constraint on the system state, i.e., $\Xi = \mathbb{R}^4$. If we choose free end-point conditions, i.e., $\Pi = \mathbb{R}^4$, the MO-MPC problem (13) can be solved by applying Pontryagin minimum principle (PMP) with the help of barrier functions. Define the Hamiltonian by

$$H(\xi, \lambda, \omega) = \ell(\xi, \omega) + \lambda^{\mathrm{T}} h_d(\xi, \omega) - \gamma(b_1(\omega) + b_2(\omega))$$

with

$$\ell(\xi, \omega) = \alpha L_1(\xi, \omega) + (1-\alpha) L_2(\xi, \omega)$$

$$b_1(\omega) = \sum_{j=1}^{4} \log(\omega(j) - \underline{\omega}(j)),\ b_2(\omega) = \sum_{j=1}^{4} \log(\bar{\omega}(j) - \omega(j))$$

Here, λ is called the costate, and $b_1(\omega)$, $b_2(\omega)$ are barriers for the inequality (20); α is the weight (18) and γ is a small positive number. Define $E(\xi) = \alpha E_1(\xi) + (1-\alpha) E_2(\xi)$, then PMP claims that for a local optimal control $\{\omega_i^*\}_{i=0}^{N-1}$, there exist $\{\lambda_i^*\}_{i=0}^{N}$ satisfying the following conditions:

$$\xi_{k+1}^* = h_d(\xi_k^*, \omega_k^*) \tag{21a}$$

$$\lambda_i^* = \lambda_{i+1}^* + H_\xi^{\mathrm{T}}(\xi_i^*, \lambda_{i+1}^*, \omega_i^*) \Delta t \tag{21b}$$

$$\lambda_N^* = E_\xi^{\mathrm{T}}(\xi_N^*) \tag{21c}$$

$$\xi_0^* = \xi \tag{21d}$$

$$H_\omega(\xi_i^*, \lambda_{i+1}^*, \omega_i^*) = 0 \tag{21e}$$

which sufficiently solves the Karush-Kuhn-Tucker (KKT) conditions by imposing N boundary conditions (21b-21c). Observe that given the initial state ξ and a control input $U = col(\omega_0, ..., \omega_{N-1})$, the states $\{\xi_i^*\}_{i=0}^{N}$ and costates $\{\lambda_i^*\}_{i=0}^{N}$ can be determined via recurrence relations (21a-21b). Therefore, at each sampling instant, we only need to solve equations (21e), which avoids expensive numerical operations of successive linearizations in solving the KKT conditions [29].

B. Parameter Selection

The AUV parameters are extracted from the identified model of the Saab SeaEye Falcon open-frame ROV/AUV

Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on July 04,2024 at 03:23:37 UTC from IEEE Xplore. Restrictions apply.

based on [30]. The mass $m = 116$ [kg], moment of inertia $I_z = 13.1$ [kg m^2]; added mass $X_{\dot{u}} = -167.6$, $Y_{\dot{v}} = -477.2$ and $N_{\dot{r}} = -15.9$ with $M_x \triangleq m - X_{\dot{u}}$, $M_y \triangleq m - Y_{\dot{v}}$ and $M_\psi \triangleq I_z - N_{\dot{r}}$; hydrodynamic coefficients $X_u = 26.9$, $Y_v = 35.8$, $N_r = 3.5$, $D_u = 241.3$, $D_v = 503.8$ and $D_r = 76.9$; and thrust limits $F_{u,\max} = 2000$ [N], $F_{v,\max} = 2000$ [N] and $F_{r,\max} = 900$ [Nm]. Initial conditions are given $\mathbf{x}(0) = [0.5, 0, 0, 0, 0, 0]^{\mathrm{T}}$ and $s(0) = 0$.

The MPC parameters are selected such that the sampling period $\Delta t = 0.1$[sec], prediction horizon $N = 10$, $\gamma = 10^{-4}$, $A = I$ and other weighting matrices are chosen as follows,

$$
\begin{aligned}
Q_1 &= \operatorname{diag}([10^5, 10^5, 10^2, 0.1, 0.1, 0.1, 0.1]) \\
Q_2 &= \operatorname{diag}([1, 1, 1, 10^3, 0.1, 0.1, 0.1]) \\
P_1 &= \operatorname{diag}([100, 100, 10, 10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}]) \\
P_2 &= \operatorname{diag}([0.1, 0.1, 0.1, 100, 10^{-3}, 10^{-3}, 10^{-3}]) \\
R_1 &= R_2 = \operatorname{diag}([10^{-3}, 10^{-3}, 10^{-3}, 10^{-3}])
\end{aligned}
$$

C. Path-Following Performance

The AUV path-following results with different β values are shown in Fig. 1. It can be observed that (i) for all of the three cases, the AUV trajectory successfully converge to the desired sinusoidal path, which validates the effectiveness of the proposed MO-MPC PF method; (ii) for each case,

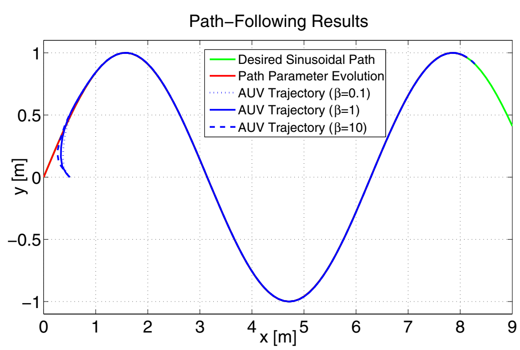

Fig. 1. *AUV path-following results.*

at the beginning, the AUV moves in the direction that is almost perpendicular to the path in order to get the fastest convergence, which is a desirable property; (iii) it seems that the larger β value results in faster path convergence because the β value accounts for the sensitivity (slope) of the logistic function. However, β potentially influences the stability issue, which will be discussed in the next section.

Fig. 2 records the surge velocity of the AUV during the simulation (for the case of $\beta = 1$). As we can see, the surge velocity keeps very well at the desired speed after the sacrifice for path convergence in the beginning. And in Fig. 3, the AUV thrust forces and moments as well as the imaginary path control input are plotted. As expected, they are all within the corresponding ranges of permitted values, which validate the usefulness of barrier functions in the PMP implementation.

Fig. 4 records the values of weights α. Generally speaking, α drops rapidly to the expected value 0.5. Since there are

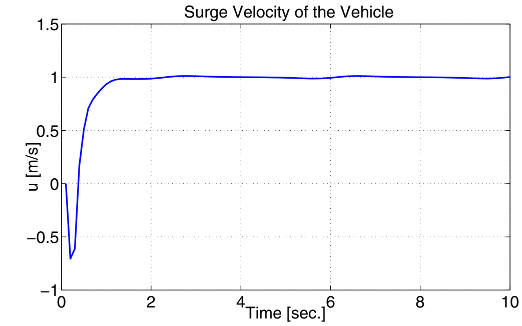

Fig. 2. *Surge velocity of the AUV.*

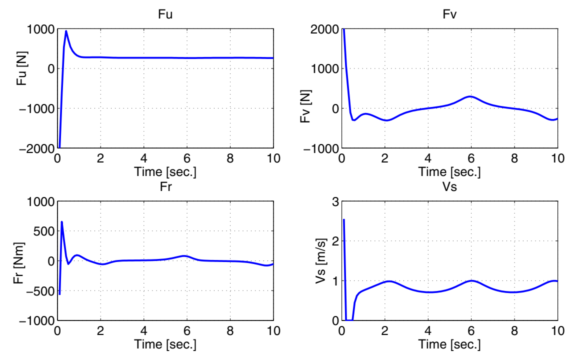

Fig. 3. *Control inputs of the augmented system.*

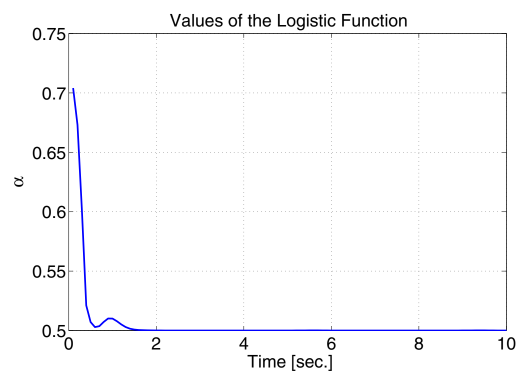

Fig. 4. *The values of weight α.*

some overshoot in the beginning (observed from Fig. 1), the α value fluctuates accordingly.

V. Discussion on Stability

Being the most commonly used method in multi-objective optimizations, the weighted sum method based MO-MPC, however, does not have many rigorous theoretical result-s on the closed-loop stability. [22] provides an available framework to guarantee the closed-loop stability. Before solving the original multi-objective optimizations (13), at each sampling instant, a subproblem that determines the appropriate weight α has to be solved:

$$\alpha^*(\xi, \alpha_d, J_\alpha) = \arg\min_{\alpha} \quad f_\alpha(\alpha - \alpha_d) \tag{22a}$$

Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on July 04,2024 at 03:23:37 UTC from IEEE Xplore. Restrictions apply.

$$\text{s.t.} \quad \begin{aligned} & V^*(\xi,\alpha) \le J_\alpha \\ & 0 \le \alpha \le 1 \end{aligned} \tag{22b}$$

where $V^*(\xi,\alpha) = a^{\mathrm{T}} J(U^*(\xi,\alpha),\xi)$ is the value function associated with problem (17); α_d is the target weight calculated by (18); f_a is a convex function that measures of the distance between α and α_d; J_α is the value of $a^{\mathrm{T}} J$ depends on the previous time optimal solution (see [22] for details). Providing the satisfaction of the standard MPC stabilizing conditions (c.f., [11], [22], [24]), the subproblem (22) guarantees the non-increasing property of the value function of (17) hence the closed-loop stability.

However, there exists an implementation issue on solving the subproblem (22). Since the constraints (22b) are essentially dependent on the optimal value of α, the computational cost of solving the subproblem is usually unaffordable (see [31]). In this paper, we mainly focus on proposing a possible framework for the AUV PF problem, thus the insertion of (22) is omitted. An interesting issue subsequently arises. Due to the lack of insertion (22), the desired weight $\alpha_d = \alpha(Er)$ should be intuitively changing relatively slow, which is corresponding to a small β. However, as mentioned in section IV-C, a relatively large β is preferred for fast path convergence. There must be a tradeoff in choosing the value of β, which is challenging yet worth being further investigated in the future.

VI. Conclusions

In this paper, the PF problem of an AUV was investigated. In view of the prioritized PF requirements, an MO-MPC framework has been proposed along with the weighted sum method. A zero-path-error manifold was found for the formulation of the MO-MPC problem and a logistic function was proposed for the appropriate choice of weights associated with the weighted sum method. In addition, for the particular choice of path dynamics, the Pontryagin minimum principle was applied instead of solving KKT conditions. The closed-loop stability was briefly discussed and can be in principle covered by existing results on MO-MPC. Simulation results based on the identified model of the Falcon revealed the effectiveness of the MO-MPC PF control method.

References

[1] T. Fossen, *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*. Trondheim, Norway: Marine Cybernetics, 2002.

[2] Z. Peng, D. Wang, Y. Shi, H. Wang, and W. Wang, "Containment control of networked autonomous underwater vehicles with model uncertainty and ocean disturbances guided by multiple leaders," *Information Sciences*, vol. 316, pp. 163–179, 2015.

[3] A. Micaelli and C. Samson, "Trajectory tracking for a unicycle-type and two steering wheels mobile robots," INRIA, Sophia-Antipolis, France, Tech. Rep. 2097, 1993.

[4] P. Encarnacao and A. Pascoal, "3d path-following for autonomous underwater vehicles," in *Proceedings of 39th IEEE Conference on Desicion and Control*, Sydney, Australia, 2000.

[5] P. Encarnacao, A. Pascoal, and M. Arcak, "Path-following for marine vehicles in the presence of unknown currents," in *Proceedings of 6th IFAC Symposium on Robot Control*, Vienna, Austria, 2000.

[6] L. Lapierre and D. Soetanto, "Nonlinear path-following control of an AUV," *Ocean Engineering*, vol. 34, 2007.

[7] L. Lapierre and B. Jouvencel, "Robust nonlinear path-following control of an AUV," *IEEE Journal of Oceanic Engineering*, vol. 33, 2008.

[8] S. Yu, X. Li, H. Chen, and F. Allgower, "Nonlinear model predictive control for path following problems," *International Journal of Robust and Nonlinear Control*, vol. 25, 2015.

[9] T. Faulwasser, B. Kern, and R. Findeisen, "Model predictive path-following for constrained nonlinear systems," in *Proceedings of 48th IEEE Conference on Decision and Control*, Shanghai, China, 2009.

[10] M. Bock and A. Kugi, "Real-time nonlinear model predictive path-following control of a laboratory tower crane," *IEEE Transactions on Control Systems Technology*, vol. 22, pp. 1461–1474, 2014.

[11] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–841, 2000.

[12] S. Qin and T. Badgwell, "An overview of nonlinear model predictive control applications," in *Nonlinear Model Predictive Control*. Birkhauser, 2000, pp. 369–392.

[13] Y. Shi, J. Huang, and B. Yu, "Robust tracking control of networked control systems: Application to a networked dc motor," *IEEE Transactions on Industrial Electronics*, vol. 60, pp. 5864–5874, 2013.

[14] H. Li and Y. Shi, "Robust distributed model predictive control of constrained continuous-time nonlinear systems: A robustness constraint approach," *IEEE Transactions on Automatic Control*, vol. 59, pp. 1673–1678, 2014.

[15] H. Li, Y. Shi, and W. Yan, "Distributed receding horizon control of constrained nonlinear vehicle formations with guaranteed γ-gain stability," *Automatica*, vol. 68, pp. 148–154, 2016.

[16] H. Li and W. Yan, "Receding horizon control based consensus scheme in general linear multi-agent systems," *Automatica*, vol. 56, pp. 12–18, 2015.

[17] A. Pavlov, H. Nordahl, and M. Breivik, "MPC-based optimal path following for underactuated vessels," in *Proceedings of 8th IFAC International Conference on Manoeuving and Control of Marine Craft*, Guaruja, Brazil, 2009.

[18] S. Oh and J. Sun, "Path following of underactuated marine vessels using line-of-sight based model predictive control," *Ocean Engineering*, vol. 37, 2010.

[19] T. Faulwasser and R. Findeisen, "Constrained output path-following for nonlinear systems using predictive control," in *Proceedings of 8th IFAC Symposium on Nonlinear Control Systems*, Bologna, Italy, 2010.

[20] A. Alessandretti, P. Aguiar, and C. Jones, "Trajectory-tracking and path-following controllers for constrained underactuated vehicles using model predictive control," in *Proceedings of European Control Conference 2013*, Zurich, Switzerland, 2013.

[21] T. Faulwasser, T. Weber, P. Zometa, and R. Findeisen, "Implementation of nonlinear model predictive path-following control for an industrial robot," *arXiv:1506.09084 [cs.SY]*, 2015.

[22] A. Bemporad and D. Munoz de la Pena, "Multiobjective model predictive control," *Automatica*, vol. 45, pp. 2823–2830, 2009.

[23] V. Zavala and A. Flores-Tlacuahuac, "Stability of multiobjective predictive control: a utopia-tracking approach," *Automatica*, vol. 48, pp. 2627–2632, 2012.

[24] T. Faulwasser, "Optimization-based solutions to constrained trajectory-tracking and path-following problems," Ph.D. dissertation, Otto-von-Guericke-Universitat, Magdeburg, Germany, 2012.

[25] D. Diehl, H. Ferreau, and N. Haverbeke, "Efficient numerical methods for nonlinear MPC and moving horizon estimation," *Nonlinear Model Predictive Control*, vol. 384, pp. 391–417, 2009.

[26] K. Graichen, "A fixed-point iteration scheme for real-time model predictive control," *Automatica*, vol. 48, pp. 1300–1305, 2012.

[27] K. Deb, *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons, NY, 2001.

[28] K. Miettinen, *Nonlinear multiobjective optimization*. Kluwer Academic Publishers, 1999.

[29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[30] A. Proctor, "Semi-automous guidance and control of a Saab SeaEye Falcon ROV," Ph.D. dissertation, University of Victoria, 2014.

[31] B. Colson, P. Marcotte, and G. Savard, "An overview of bilevel optimization," *Annals of Operations Research*, vol. 153, pp. 235–256, 2007.

Authorized licensed use limited to: Huazhong University of Science and Technology. Downloaded on July 04,2024 at 03:23:37 UTC from IEEE Xplore. Restrictions apply.