Linear Parameter Varying Model Predictive Control of an Autonomous Underwater Vehicle

MEHMET AVINC, CHINGIZ HAJIYEV
Faculty of Aeronautics and Astronautics,
Istanbul Technical University,
Maslak, 34469, Istanbul,
TURKEY

Abstract: - This paper presents a study on the application of Linear Parameter Varying Model Predictive Control (LPV-MPC) to the control of an Autonomous Underwater Vehicle (AUV). The study focuses on the development of an LPV-MPC-based control system that enables the AUV to follow given angular rate commands. The proposed control algorithm uses the mathematical model of the AUV. The paper also explains how to implement the proposed control algorithm and presents its results in a simulation environment.

Key-Words: - underwater vehicle, nonlinear dynamics, motion control, model predictive control, navigation, AUV.

Received: November 9, 2024. Revised: February 15, 2025. Accepted: April 9, 2025. Published: May 9, 2025.

1 Introduction

Autonomous Underwater Vehicles (AUVs) have been used frequently in search and rescue, exploration, civil and military areas in recent years. These vehicles require highly good guidance, navigation, and control algorithms because they must do all duties successfully without human aid, [1].

To adapt autonomous underwater vehicles to complex and changing underwater conditions, it is necessary to develop reliable control algorithms. In recent years, Model Predictive Control (MPC) and other optimum control methods have been used in the control of AUVs, [2].

Using system dynamics, MPC has been used to provide navigation and hovering for autonomous underwater vehicles, [3]. A study [4] evaluated a group of MPC methods and found that they effectively meet the real-time control requirements of AUV dynamics. In [5], a double closed-loop control method was presented that divides the process into two stages. This method involves the outer-loop controller sending a desired speed command to the inner-loop speed controller. The inner-loop controller then determines the necessary control inputs to accurately track the trajectory in the closed-loop system.

An effective way to solve this problem is nonlinear predictive control (NMPC) methods such as Linear Parameter-Varying Model Predictive Control (LPV-MPC). In [6], a control framework based on LPV-MPC for docking maneuvering of an autonomous underwater vehicle was presented. The LPV-MPC technique has been used in a variety of technical domains, such as automotive and aviation systems. In [7], drone dynamics are modeled using the LPV format.

This modeling approach enables the application of basic MPC methods that are suitable for linear systems. In [8], LPV theory is used to model vehicle dynamics, and an LPV-MPC model is implemented which can be calculated online with lower computational cost. An offline optimal trajectory planner is also utilized to solve the optimal time problem, determining the best route within the constraints of the environment. In [9], a new controller is introduced for nonlinear missile autopilots based on model predictive control with constraints. The nonlinear model is reformulated into a state-dependent linear structure, serving as the internal model for prediction, and the constrained solution is determined by solving a quadratic programming problem online at each sampling instance. Furthermore, different controllers in [10], [11], [12] were developed for AUVs.

The work [13] on robust adaptive MPC for systems with state-dependent disturbances is highly relevant, as it provides a framework for dealing with the uncertainties and disturbances AUVs face in underwater environments, ensuring constraint fulfillment and system stability. Simplifying complex AUV dynamics is also essential, the study [14] proposes a predictive control method that approximates higher-order systems with time-delay models, helping to manage the computational complexity of real-time AUV control.

Compared to traditional MPC methods, LPV-MPC has several advantages. It takes into account how AUVs behave differently under different circumstances, such as speed changes or environmental disturbances. Since AUV has significant nonlinearities, this study proposes a linear parameter-varying model predictive control to regulate it. By adding a linear parameter-varying model, LPV-MPC extends the capabilities of conventional MPC.

2 Nonlinear AUV Model

The REMUS autonomous underwater vehicle [15] is used as the research platform for this study. Figure 1 illustrates the Earth-fixed and body-fixed coordinate systems utilized for the REMUS AUV. Here, ϕ, θ, and ψ represent the roll, pitch, and yaw angles, respectively, and are defined as Euler angles. Additionally, (*p*, *q*, *r*) refer to the body angular rate vector, while the linear velocity of the AUV is denoted by (*u*, *v*, *w*). As shown in Figure 1, the forces and moments acting on the AUV are represented by the vector $[X, Y, Z, K, M, N]^{\mathrm{T}}$. Using these definitions, the following vectors can be established to represent the equations of motion.

$$\boldsymbol{n}_1 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \boldsymbol{n}_2 = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \boldsymbol{v}_1 = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \boldsymbol{v}_2 = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (1)$$

where, $\boldsymbol{\eta}_1$ is the position vector consisting of North, East, and Down components. $\boldsymbol{\eta}_2$ is roll, pitch, and yaw angle vector, $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are linear and angular velocities respectively.

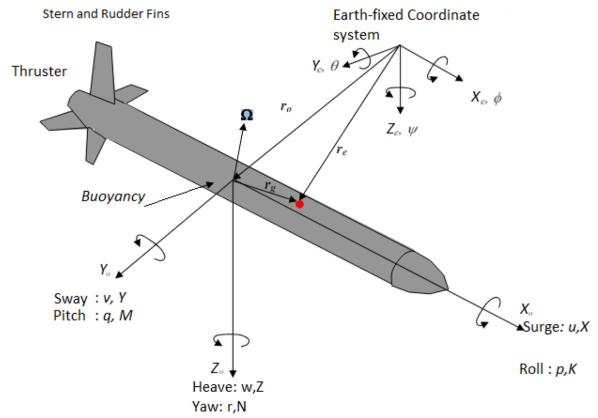

Fig. 1: Earth-fixed and body-fixed frames

2.1 AUV Kinematics

The transformation of translational velocities between body-fixed and Earth-fixed coordinate frames are given as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \boldsymbol{J}_1(\boldsymbol{\eta}_2) \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2)$$

where

$$J_1(\boldsymbol{\eta}_2) = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi s\theta c\phi \\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & -c\psi s\phi + s\psi s\theta c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \quad (3)$$

where *c* and *s* correspond cosine and sine respectively. Furthermore, the rotational velocities between the body-fixed and Earth-fixed coordinate frames can be expressed as follows:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \boldsymbol{J}_2(\boldsymbol{\eta}_2) \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4)$$

where

$$\boldsymbol{J}_2(\boldsymbol{\eta}_2) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \dfrac{\sin\phi}{\cos\theta} & \dfrac{\cos\phi}{\cos\theta} \end{bmatrix} \quad (5)$$

2.2 AUV Rigid Body Dynamics

The equations of motion of the AUV model were taken from, [1]. The position vector of the center of gravity and buoyancy defined as follows:

$$\boldsymbol{r}_G = \begin{bmatrix} x_g \\ y_g \\ z_g \end{bmatrix}, \boldsymbol{r}_B = \begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} \quad (6)$$

Hence equations of motion can be written as follows, [16]:

$$\begin{aligned} M\dot{\boldsymbol{v}} &= f(\boldsymbol{\eta}, \boldsymbol{v}) + g(\boldsymbol{v})\boldsymbol{U} \\ \dot{\boldsymbol{\eta}} &= J(\boldsymbol{\eta})\boldsymbol{v} \end{aligned} \quad (7)$$

where

$$M = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 & mz_g & -my_g \\ 0 & (m - Y_{\dot{v}}) & 0 & -mz_g & 0 & (mx_g - Y_{\dot{r}}) \\ 0 & 0 & (m - Z_{\dot{w}}) & my_g & -(m + Z_{\dot{q}}) & 0 \\ 0 & -mz_g & my_g & (I_x - K_{\dot{p}}) & -I_{xy} & -I_{xz} \\ mz_g & 0 & -(mx_g + Z_{\dot{q}}) & -I_{xy} & (I_y - M_{\dot{q}}) & -I_{yz} \\ -my_g & (mx_g - Y_{\dot{r}}) & 0 & -I_{xy} & -I_{yz} & (I_z - N_{\dot{r}}) \end{bmatrix} \quad (8)$$

$$\begin{aligned} f_1(\eta, v) = -m\big(wq - vr - x_g(r^2 + q^2) + y_g qp + z_g rp\big) + X_{HS} \\ + X_{u|u|}u|u| + Z_w wq + Z_q q^2 - Y_{\dot{v}} vr - Y_r r^2 \end{aligned}$$

$$
\begin{aligned}
f_2(\eta, v) = &-m\big(ur - pw + x_g pq - y_g(r^2+p^2) + z_g rq\big) + Y_{HS} \\
&+ Y_{v|p|}v|v| + Y_{r|r|}r|r| - Z_w wp - Z_q qp + Y_{ur}ur \\
&+ Y_{uv}uv \\
f_3(\eta, v) = &-m\left(vp - qu + x_g rp + y_g rq - z_g(q^2+p^2)\right) + Z_{HS} \\
&+ Z_{w|w|}w|w| + Z_{q|q|}q|q| + Y_{\dot{v}}vp + Y_{\dot{r}}rp \\
&+ Z_{uq}uq + Z_{uw}uw \\
f_4(\eta, v) = &-I_{xy}pr + I_{xz}pq - I_{yz}(r^2-q^2) - \big(I_z - I_y\big)rq - \\
&m\left(y_g(vp-uq) - z_g(ur-wp)\right) + K_{HS} + K_{p|p|}p|p| + \\
&(Z_w - Y_v)wv + \big(Z_q + Y_r\big)vq - \big(Y_r + Z_q\big)rw + \\
&\big(N_r - M_q\big)rq
\end{aligned}
\tag{9}
$$

$$
\begin{aligned}
f_5(\eta, v) = &\, I_{xy}qr - I_{yz}pq + I_{xz}(r^2-p^2) - (I_x - I_z)pr \\
&- m\big(z_g(wq-vr) - x_g(vp-uq)\big) + M_{HS} \\
&+ M_{w|w|}w|w| + M_{q|q|}q|q| + M_{uq}uq - Y_{\dot{r}}vp \\
&+ M_{uw}uw + \big(K_{\dot{p}} - N_{\dot{r}}\big)rp \\
f_6(\eta, v) = &\, I_{xz}qr + I_{yz}rp + I_{xy}(p^2+q^2) + \big(I_x - I_y\big)pq \\
&- m\big(x_g(ur-wp) - y_g(wq-vr)\big) + N_{HS} \\
&+ N_{v|p|}v|v| + N_{r|r|}r|r| + N_{uv}uv + Y_{\dot{r}}ur + Z_{\dot{q}}wp \\
&+ \big(M_{\dot{q}} - K_{\dot{p}}\big)pq
\end{aligned}
$$

$$
g(v) = \begin{bmatrix} 0 & 0 & 1 \\ Y_{uu\delta_r}u^2 & 0 & 0 \\ 0 & Z_{uu\delta_s}u^2 & 0 \\ 0 & 0 & -R \\ 0 & M_{uu\delta_s}u^2 & 0 \\ N_{uu\delta_r}u^2 & 0 & 0 \end{bmatrix} \tag{10}
$$

$$
U = \begin{bmatrix} \delta_r \\ \delta_s \\ X_{prop} \end{bmatrix}
$$

where, m is mass and I_{xx}, I_{yy}, I_{zz} are moment of inertia values and I_{xy}, I_{xz}, I_{yz} are products of inertia. The terms on the right-hand side of the equations represent external forces, including hydrodynamic forces and moments, as well as gravitational and buoyancy forces and moments. These forces are dependent on the AUV's states. δ_r, δ_s and X_{prop} are rudder, stern deflections and propeller thrust force which are control input of AUV. Also, J is augmented transformation matrix and $\boldsymbol{\eta}, \boldsymbol{v}$ are augmented state vectors and can be written as follows:

$$
\begin{aligned}
J(\boldsymbol{\eta}) &= \begin{bmatrix} J_1(\boldsymbol{\eta}_2) & 0_{3\times 3} \\ 0_{3\times 3} & J_2(\boldsymbol{\eta}_2) \end{bmatrix} \\
\boldsymbol{v} &= [\boldsymbol{v}_1 \quad \boldsymbol{v}_2]^T \\
\boldsymbol{\eta} &= [\boldsymbol{\eta}_1 \quad \boldsymbol{\eta}_2]^T
\end{aligned}
\tag{11}
$$

All coefficients that include the subscript of the state variables (u, v, w, p, q, r), such as $X_{u|u|}, M_{uq}$, represent the hydrodynamic force and moment coefficients. The values of hydrodynamic forces and moment were taken from, [16].

3 Problem Formulation

3.1 State-Space Representation

In LPV-MPC, the control problem is defined as an optimization problem where cost is based on error between reference and actual states, subject to system dynamics and control action. The derive formulation, first of all, Eqs. (7-10) must be converted into a general linear form as follows:

$$
\begin{aligned}
\dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} \\
\boldsymbol{y} &= C\boldsymbol{x}
\end{aligned}
\tag{12}
$$

It is quite difficult to convert nonlinear model in linear form defined above and it must be done cleverly and carefully. In order to convert the nonlinear model into a linear model, the following manipulation was proceeded.

$$
A = \begin{bmatrix} [M^{-1}A_1]_{6\text{x}6} & 0_{6\text{x}3} \\ [0_{3\text{x}3}\, J_2(\boldsymbol{\eta_2})]_{3\text{x}6} & 0_{3\text{x}3} \end{bmatrix}_{9\text{x}9} \tag{13}
$$

$$
B = M^{-1}B_1 \tag{14}
$$

where

$$
A_1 = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{15}
$$

$$
A_{11} = \begin{bmatrix} \frac{X_{HS}}{u} + X_{uu}|u| & (-Y_{\dot{v}} + m)r & (Z_{\dot{w}} - m)q \\ (Y_{ur} - m)r + Y_{uv}v + \frac{Y_{HS}}{u} & Y_{vv}|v| & (-Z_{\dot{w}} + m)p \\ \frac{K_{HS}}{u} + mz_g r + my_g q & (Y_{\dot{v}} - m)p & Z_{ww}|w| \end{bmatrix}
$$

$$
A_{12} = \begin{bmatrix} -\big(my_g q + m_z qr\big) & \big(Z_{\dot{q}} + mx_g\big)q & \big(-Y_{\dot{r}} + mx_g\big)r \\ my_g p & \big(-Z_{\dot{q}} - mx_g\big)p - mz_g r & Y_{rr}|r| + my_g r \\ mz_g p & Z_{qq}|q| + mz_q q & -my_g q + \big(Y_{\dot{r}} - mx_g\big)p \end{bmatrix}
$$

$$
A_{21} = \begin{bmatrix} \frac{K_{HS}}{u} + mz_g r + my_g q & \big(Z_{\dot{q}} + Y_{\dot{r}}\big) - my_g p & (Z_{\dot{w}} - Y_{\dot{v}})v - mz_g p \\ \frac{M_{HS}}{u} + \big(M_{uq} - mx_g\big)q + M_{uw}w & \big(-Y_{\dot{r}} + mx_g\big)p + mz_g r & M_{ww}|w| - mz_g q \\ \frac{N_{HS}}{u} + \big(N_{ur} - mx_g\big)r + N_{uv}v & N_{vv}|v| - my_g r & \big(Z_{\dot{q}} + mx_g\big)p + my_g q \end{bmatrix}
$$

$$
A_{22} = \begin{bmatrix} K_{pp}|p| & \big(N_{\dot{r}} - M_{\dot{q}}\big)r & -\big(Z_{\dot{q}} + Y_{\dot{r}}\big)w + \big(I_{yy} - I_{zz}\big)q \\ (I_{zz} - I_{xx})r & M_{qq}|q| & \big(K_{\dot{p}} - N_{\dot{r}}\big)p \\ \big(M_{\dot{q}} - K_{\dot{p}}\big)q & \big(I_{xx} - I_{yy}\big)p & N_{rr}|r| \end{bmatrix}
$$

$$
B_1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & Y_{uu\delta r}u^2 & 0 \\ Z_{uu\delta_s}u^2 & 0 & 0 \\ 0 & 0 & -R \\ M_{uu\delta_s}u^2 & 0 & 0 \\ 0 & N_{uu\delta r}u^2 & 0 \end{bmatrix}
$$

Also state vector $\boldsymbol{x}$, and control input vector $\boldsymbol{u}$ is defined as follows:

$$
\begin{aligned}
\boldsymbol{x} &= [u\ v\ w\ p\ q\ r\ \phi\ \theta\ \psi]^T \\
\boldsymbol{u} &= [\delta_s\ \delta_r\ X_{prop}]
\end{aligned}
\tag{16}
$$

Since LPV-MPC uses a discretized model, these equations can be discretized by using the forward-Euler approximation:

$$\boldsymbol{x}_{k+1} = A_{d_k}\boldsymbol{x}_k + B_{d_k}\boldsymbol{u}_k \qquad (17)$$
$$\boldsymbol{y}_k = C_{d_k}\boldsymbol{x}_k$$

where

$$A_{d_k} = I_{9\text{x}9} + A.T_s, \qquad B_{d_k} = B.T_s \;\; C_{d_k} = C \quad (18)$$

Here, T_s is the sample time of the controller, and C determined based on the states that are desired to be controlled. As observed, the components of matrix A vary with the states of the AUV, which change over time. This is why it is referred to as linear parameter-varying. It should also be noted that some components are divided by the forward body velocity, u. While one could choose a different variable for these terms, such as v, the same differential equations outlined in equations 7 to 10 must still hold. The primary reason for using u in the state-space formulation is that the forward velocity cannot be zero or close to zero, except at the start when the AUV has no velocity. In contrast, other state variables can reach zero, which would cause problems during optimization and introduce infinity terms in matrix A. Although zero forward velocity at the start can also pose an issue, this can be managed by either applying a simple PID controller or waiting a brief period for the forward velocity to increase before activating the LPV-MPC controller.

3.2 LPV-MPC Formulation

The general form of cost function defined for LPV-MPC is defined as sum of stage costs over a finite horizon period as follows:

$$J = \frac{1}{2}\boldsymbol{e}_{k+N}^T S\boldsymbol{e}_{k+N} + \frac{1}{2}\sum_{i=0}^{N-1}[\boldsymbol{e}_{k+i}^T Q\boldsymbol{e}_{k+i} + \boldsymbol{u}_{k+i}^T R\boldsymbol{u}_{k+i}] \qquad (19)$$

Here, N represents the prediction horizon, indicating how many future time steps the states will be forecasted. S,Q, and R are positive-definite weighting matrices corresponding to the terminal cost, running cost, and input cost, respectively. Additionally, $\boldsymbol{e}$ denotes the error between the desired and current states, as defined below:

$$\boldsymbol{e}_k = \boldsymbol{r}_k - C\boldsymbol{x}_k \qquad (20)$$

Here $\boldsymbol{r}_k$ is desired or reference states and time step k. This cost function does not ensure that the error will reach zero, as it depends not only on the error but also on the control inputs. As a result, even at steady-state, the cost function may be minimized while the error remains non-zero. To address this issue, a change of control input can be used in cost and the state vector can be augmented. Consequently, the minimization problem can be described below.

$$minimize \quad \frac{1}{2}\boldsymbol{e}_{k+N}^T S\boldsymbol{e}_{k+N} + \frac{1}{2}\sum_{i=0}^{N-1}[\boldsymbol{e}_{k+i}^T Q\boldsymbol{e}_{k+i} + \Delta\boldsymbol{u}_{k+i}^T R\Delta\boldsymbol{u}_{k+i}] \quad (21)$$

$s.t$

$$\tilde{\boldsymbol{x}}_{k+1} = \tilde{A}_k\tilde{\boldsymbol{x}}_k + \tilde{B}_k\Delta\boldsymbol{u}_k$$
$$\Delta\boldsymbol{u}_{k_{min}} \leq \Delta\boldsymbol{u}_k \leq \Delta\boldsymbol{u}_{k_{max}}$$

where

$$\tilde{A}_k = \begin{bmatrix} A_{d_k} & B_{d_k} \\ 0 & I \end{bmatrix}, \tilde{B}_k = \begin{bmatrix} B_{d_k} \\ I \end{bmatrix}$$
$$\tilde{C} = \begin{bmatrix} C_{d_k} & 0 \end{bmatrix} \qquad (22)$$

$$\tilde{\boldsymbol{x}}_k = \begin{bmatrix} \boldsymbol{x}_k & \boldsymbol{u}_{k-1} \end{bmatrix}^{\boldsymbol{T}}$$
$$\boldsymbol{e}_k = \boldsymbol{r}_k - \tilde{C}\tilde{\boldsymbol{x}}_k$$

4 Problem Solution

With the problem defined, the minimization problem can be solved analytically in the unconstrained case, while the constrained case can be formulated into its final form, suitable for solving using quadratic programming methods. The primary objective is to determine the change in control inputs over the entire finite time horizon that minimizes the cost function outlined in Eq. (21). Mathematically, we need to find following vector:

$$[\Delta\boldsymbol{u}_k \quad \Delta\boldsymbol{u}_{k+1} \quad \Delta\boldsymbol{u}_{k+2} \; . \; . \; . \Delta\boldsymbol{u}_{k+N-1} \;]^{\boldsymbol{T}} \qquad (23)$$

At time step k, the change in control inputs is determined to minimize the cost over the N step horizon. However, instead of applying this result across the entire horizon and recalculating after N steps, the process is repeated at each time step. Only the first control input adjustment is applied to the system, and the minimization is performed again at the next time step. In summary, this approach can be described as follows:

$$\boldsymbol{u}_k = [1\;0\;0\;...\;0]\begin{bmatrix} \Delta\boldsymbol{u}_k \\ \Delta\boldsymbol{u}_{k+1} \\ \Delta\boldsymbol{u}_{k+2} \\ \vdots \\ \Delta\boldsymbol{u}_{k+N-1} \end{bmatrix} + \boldsymbol{u}_{\boldsymbol{k-1}} \qquad (24)$$

In order to solve this optimization problem, the error define in Eq. (22) can be written in the cost function and cost can be expanded as follows:

$$J = \frac{1}{2}\begin{pmatrix} \boldsymbol{r}_{k+N}^{T} S\, \boldsymbol{r}_{k+N} - \boldsymbol{2r}_{k+N}^{T} S\tilde{C}\tilde{\boldsymbol{x}}_{k+N} \\ +\tilde{\boldsymbol{x}}_{k+N}^{\boldsymbol{T}}\tilde{C}^{\boldsymbol{T}} S\tilde{C}\tilde{\boldsymbol{x}}_{k+N} \end{pmatrix} +$$
$$\frac{1}{2}\sum_{i=0}^{N-1}\begin{bmatrix} \boldsymbol{r}_{k+i}^{T} Q\, \boldsymbol{r}_{k+i} - \boldsymbol{2r}_{k+i}^{T} Q\tilde{C}\tilde{\boldsymbol{x}}_{k+i} + \\ \tilde{\boldsymbol{x}}_{k+i}^{\boldsymbol{T}}\tilde{C}^{\boldsymbol{T}} Q\tilde{C}\tilde{\boldsymbol{x}}_{k+i} + \Delta\boldsymbol{u}_{k+i}^{T} R\Delta\boldsymbol{u}_{k+i} \end{bmatrix} \tag{25}$$

Since constant terms do no effect the minimization problem, these terms can be eliminated and adjusted cost function can be rewritten as follows in matrix form:

$$J' = \frac{1}{2}\boldsymbol{X}_{\boldsymbol{G}}^{\boldsymbol{T}}\bar{M}\boldsymbol{X}_{\boldsymbol{G}} - \Gamma_{\boldsymbol{G}}^{\boldsymbol{T}}\bar{N}\boldsymbol{X}_{\boldsymbol{G}} + \frac{1}{2}\Delta\boldsymbol{U}_{\boldsymbol{G}}^{\boldsymbol{T}}\bar{R}\Delta\boldsymbol{U}_{\boldsymbol{G}} \tag{26}$$

where

$$\bar{M} = \begin{bmatrix} \tilde{C}^{\boldsymbol{T}}Q\tilde{C} & 0 & 0 & 0 \\ 0 & \tilde{C}^{\boldsymbol{T}}Q\tilde{C} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \tilde{C}^{\boldsymbol{T}}S\tilde{C} \end{bmatrix}$$
$$\bar{N} = \begin{bmatrix} Q\tilde{C} & 0 & 0 & 0 \\ 0 & Q\tilde{C} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & S\tilde{C} \end{bmatrix}$$
$$\bar{R} = \begin{bmatrix} Q\tilde{C} & 0 & 0 & 0 \\ 0 & Q\tilde{C} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & S\tilde{C} \end{bmatrix}$$

Additionally, $\boldsymbol{X}_{\boldsymbol{G}}$, $\boldsymbol{\Gamma}_{\boldsymbol{G}}$, and $\Delta\boldsymbol{U}_{\boldsymbol{G}}$ are defined as the global state vector, global reference vector, and global control change vector, respectively, with the following expressions:

$$\boldsymbol{X}_{\boldsymbol{G}} = \begin{bmatrix} \tilde{\boldsymbol{x}}_{\boldsymbol{k+1}} \\ \tilde{\boldsymbol{x}}_{\boldsymbol{k+2}} \\ \vdots \\ \tilde{\boldsymbol{x}}_{\boldsymbol{k+N}} \end{bmatrix},\ \ \Gamma_{\boldsymbol{G}} = \begin{bmatrix} \boldsymbol{r}_{\boldsymbol{k+1}} \\ \boldsymbol{r}_{k+2} \\ \vdots \\ \boldsymbol{r}_{\boldsymbol{k+N}} \end{bmatrix}, \Delta\boldsymbol{U}_{\boldsymbol{G}} = \begin{bmatrix} \Delta\boldsymbol{u}_{\boldsymbol{k}} \\ \Delta\boldsymbol{u}_{\boldsymbol{k+1}} \\ \vdots \\ \Delta\boldsymbol{u}_{\boldsymbol{k+N-1}} \end{bmatrix}$$

To find the global control change vector $\Delta\boldsymbol{U}_{\boldsymbol{G}}$ that minimizes the cost function in equation 26, we need to eliminate $\boldsymbol{X}_{\boldsymbol{G}}$. This can be done by using the system's discrete model provided in equation 21. In essence, since the future states of the system cannot be known with certainty, they are predicted using a mathematical model. This is the fundamental concept of model predictive control. The more accurate the model, the better the state predictions will be, resulting in more reliable control inputs and greater efficiency. It should also be noted that a traditional model predictive control approach could be used, where matrices A and B are constants obtained by linearizing the nonlinear model around a trim point. However, because the AUV's dynamics are highly nonlinear, the system's behavior might significantly deviate from the linearized model, leading the MPC to generate control inputs that are ineffective in controlling the system. That's why the LPV-MPC structure where A and B are updated to predict future states is used.

All in all, $\boldsymbol{X}_{\boldsymbol{G}}$ can be predicted through the N horizon period as follows:

$$\begin{gathered} \tilde{\boldsymbol{x}}_{k+1} = \tilde{A}_k\tilde{\boldsymbol{x}}_k + \tilde{B}_k\Delta\boldsymbol{u}_k \\ \tilde{\boldsymbol{x}}_{k+2} = \tilde{A}_{k+1}\tilde{\boldsymbol{x}}_{\boldsymbol{k+1}} + \tilde{B}_{k+1}\Delta\boldsymbol{u}_{k+1} = \\ \tilde{A}_{k+1}\tilde{A}_k\tilde{\boldsymbol{x}}_k + \tilde{A}_{k+1}\tilde{B}_k\Delta\boldsymbol{u}_{\boldsymbol{k}} + \tilde{B}_{k+1}\Delta\boldsymbol{u}_{k+1} \\ \vdots \end{gathered} \tag{27}$$
$$\begin{aligned} &\tilde{\boldsymbol{x}}_{k+N} \\ &= \tilde{A}_{k+N-1}\tilde{A}_{k+N-2}\dots\tilde{A}_{k+1}\tilde{A}_k\tilde{\boldsymbol{x}}_{\boldsymbol{k}} \\ &+ \tilde{A}_{k+N-1}\tilde{A}_{k+N-2}\dots\tilde{A}_{k+1}\tilde{B}_k\Delta\boldsymbol{u}_k \\ &+ \tilde{A}_{k+N-1}\tilde{A}_{k+N-2}\dots\tilde{A}_{k+2}\tilde{B}_{k+1}\Delta\boldsymbol{u}_{k+1} + \dots \\ &+ \tilde{B}_{k+N-1}\Delta\boldsymbol{u}_{k+N-1} \end{aligned}$$

In matrix form:

$$\boldsymbol{X}_{\boldsymbol{G}} = \hat{A}\tilde{\boldsymbol{x}}_k + \hat{B}\Delta\boldsymbol{U}_{\boldsymbol{G}} \tag{28}$$

where

$$\hat{A} = \begin{bmatrix} \tilde{A}_k \\ \tilde{A}_k\tilde{A}_{k+1} \\ \tilde{A}_k\tilde{A}_{k+1}\tilde{A}_{k+2} \\ \vdots \\ \tilde{A}_{k+N-1}\tilde{A}_{k+N-2}\dots\tilde{A}_{k+1}\tilde{A}_k \end{bmatrix}$$
$$\hat{B} = \begin{bmatrix} \tilde{B}_k & 0 & 0 & 0 & 0 \\ \tilde{A}_{k+1}\tilde{B}_k & \tilde{B}_{k+1} & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ \tilde{A}_{k+N-1}\tilde{A}_{k+N-2}\dots\tilde{A}_{k+1}\tilde{B}_k & \tilde{A}_{k+N-1}\tilde{A}_{k+N-2}\dots\tilde{A}_{k+2}\tilde{B}_{k+1} & \cdots & \cdots & \tilde{B}_{k+N-1} \end{bmatrix}$$

It must be emphasized that in order to predict $\tilde{A}$ and $\tilde{B}$, control input change that found one step previous can be used, and initially, $\Delta\boldsymbol{U}_{\boldsymbol{G_0}}$ will be assumed zero.

Finally, equation 28 can be used in cost function and following expression can be obtained.

$$J' = \frac{1}{2}\left(\hat{A}\tilde{\boldsymbol{x}}_k + \hat{B}\Delta\boldsymbol{U}_{\boldsymbol{G}}\right)^T\bar{M}\left(\hat{A}\tilde{\boldsymbol{x}}_k + \hat{B}\Delta\boldsymbol{U}_{\boldsymbol{G}}\right) -$$
$$\Gamma_G^T\bar{N}\left(\hat{A}\tilde{\boldsymbol{x}}_k + \hat{B}\Delta\boldsymbol{U}_{\boldsymbol{G}}\right) + \frac{1}{2}\Delta\boldsymbol{U}_{\boldsymbol{G}}^{\boldsymbol{T}}\bar{R}\Delta\boldsymbol{U}_{\boldsymbol{G}} \tag{29}$$

By eliminating constant terms, final cost can be expressed as follows:

$$J'' = \frac{1}{2}\Delta \boldsymbol{U}_{\boldsymbol{G}}^{T}\bar{\bar{H}}\Delta \boldsymbol{U}_{\boldsymbol{G}} + [\tilde{\boldsymbol{x}}_k^T \quad \Gamma_G^T]\bar{\bar{F}}^T \Delta \boldsymbol{U}_{\boldsymbol{G}} \tag{30}$$

where

$$\bar{\bar{H}} = (\hat{B}^T \bar{M} \hat{B} + \bar{R}), \qquad \bar{\bar{F}}^T = \begin{bmatrix} \hat{A}^T \bar{M} \hat{B} \\ -\bar{N}\hat{B} \end{bmatrix}$$

This optimization problem can be solved analytically if there are no any constraints for $\Delta \boldsymbol{U}_{\boldsymbol{G}}$ as follows:

$$\frac{\partial J''}{\partial \boldsymbol{\Delta U}_{\boldsymbol{G}}} = \bar{\bar{H}}\Delta \boldsymbol{U}_{\boldsymbol{G}} + \bar{\bar{F}}\begin{bmatrix} \tilde{\boldsymbol{x}}_k \\ \Gamma_G \end{bmatrix} = 0$$

$$\rightarrow \Delta \boldsymbol{U}_{\boldsymbol{G}} = -\bar{\bar{H}}^{-1}\bar{\bar{F}}\begin{bmatrix} \tilde{\boldsymbol{x}}_k \\ \Gamma_G \end{bmatrix} \tag{31}$$

If there are constraints for $\Delta \boldsymbol{U}_{\boldsymbol{G}}$ and states, the optimization problem can be written as follows:

$$minimize \quad J'' = \frac{1}{2}\Delta \boldsymbol{U}_G^T \bar{\bar{H}}\Delta \boldsymbol{U}_G + [\tilde{\boldsymbol{x}}_k^T \quad \Gamma_G^T]\bar{\bar{F}}^T \Delta \boldsymbol{U}_G$$

$s.t.$

$$G\Delta \boldsymbol{U}_G \leq \boldsymbol{h} \tag{32}$$

The G matrix and $\boldsymbol{h}$ vector represent constraints on both the control inputs and states, linking the control inputs to the system dynamics. To solve this quadratic optimization problem, one can use MATLAB's *quadprog()* function or the *solve_qp()* function from Python's qpsolvers library. Once $\Delta \boldsymbol{U}_G$ is determined, the first term $\Delta \boldsymbol{u}_k$ will be used to compute the control input applied to the plant. The process will be repeated at each time step.

5 Simulation Results

In order to show the effectiveness of the controller, simulations are conducted. Table 1 shows all values of all constants used.

Table 1. Parameters for LPV-MPC

Parameter	Value	Unit	Description
T_s	0.01	s	Time step
Hz	10	-	Horizon period
Q	$\begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0.2 \end{bmatrix}$	-	Running cost weight matrix
S	$\begin{bmatrix} 5 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	-	Terminal cost weight matrix
R	$\begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{bmatrix}$	-	Control input change cost weight matrix

Also, change of control inputs are limited as follows:

$$|\Delta u_s| \leq 1.6^o, |\Delta u_r| \leq 1.6^o, |\Delta u_T| \leq 0.48 \ N$$

For inner dynamics, *u (forward velocity)*, *q (pitch rate)* and *r (yaw rate)* are controlled, therefore following $\tilde{C}$ matrix is used.

$$\tilde{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{33}$$

Figure 2, Figure 3 and Figure 4 show the simulation results for LPV-MPC and classical PID controller.

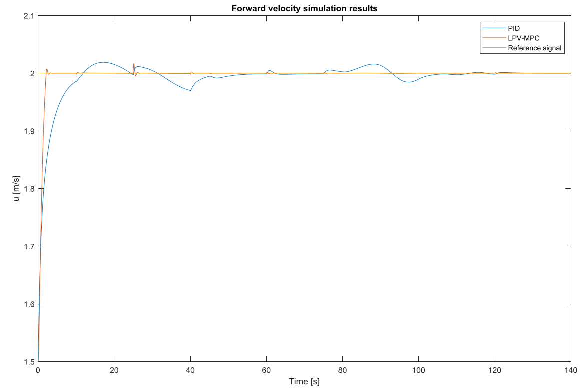

Fig. 2: LPV-MPC versus PID for forward velocity

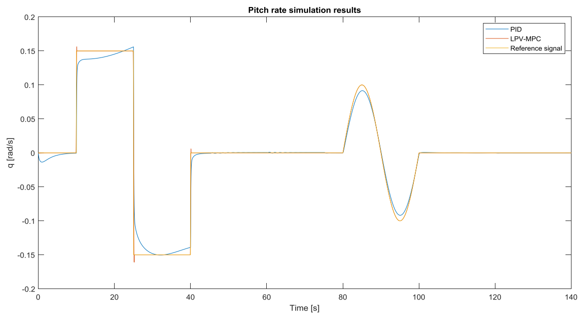

Fig. 3: LPV-MPC versus PID for pitch rate

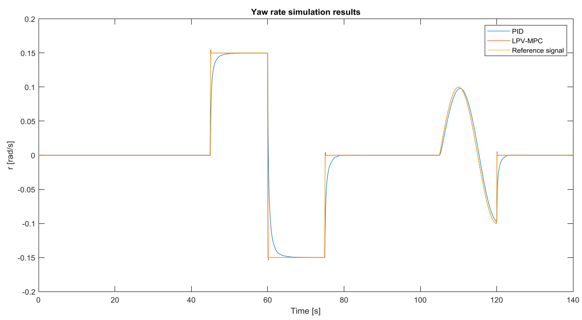

Fig. 4: LPV-MPC versus PID for yaw rate

Furthermore, tracking errors for both controllers are presented in Figure 5. In addition, the control action over time is demonstrated in Figure 6. The simulation results demonstrate that LPV-MPC surpasses PID control in both the accuracy of state tracking and the efficiency of the control actions employed. Linear Parameter-Varying Model Predictive Control provides several advantages over conventional control methods. It eliminates the need for trim, linearization, and linear control analysis methods, thereby simplifying the control design process.

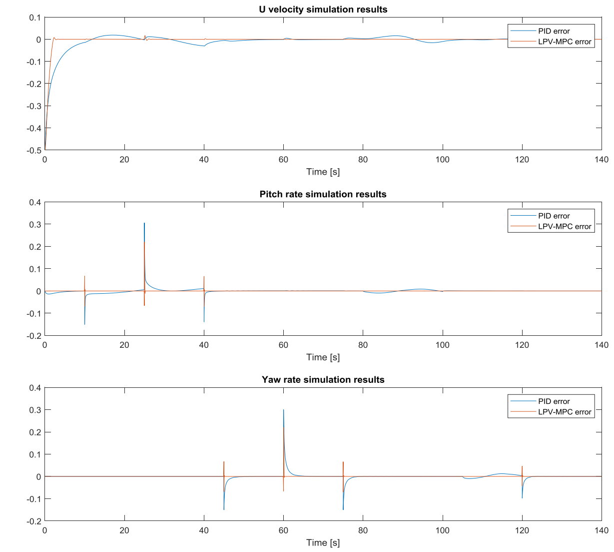

Fig. 5: Controller errors

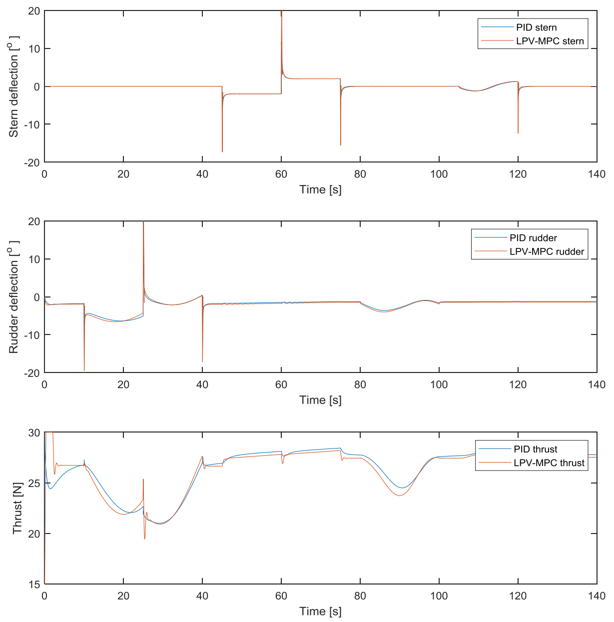

Fig. 6: Control actions

6 Conclusions

The objective of this study is to provide a predictive control method for autonomous underwater vehicles using a linear parameter-varying model. A comparison with traditional PID control methods is performed. The LPV-MPC architecture was specifically designed to address the challenges associated with the highly nonlinear dynamics of AUVs. It eliminates the need for gain scheduling and model linearization, allowing real-time adaptation to changing operating conditions.

According to the simulation results, LPV-MPC performs significantly better than PID control. On the other hand, PID control cannot remain accurate when circumstances change. However, LPV-MPC performs well in various situations, resulting in improved control efficiency and stability.

References:

[1] Prestero, T. *Verification of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle*. MS Thesis. Massachusetts Institute of Technology, 2001. doi: 10.1575/1912/3040.

[2] Rawlings, J. B., Mayne, D. Q., & Diehl, M. *Model Predictive Control*, Nob Hill Publishing, 2017. ISBN: 9780975937730.

[3] Steenson, L. V., Turnock, S. R., Phillips, A. B., Harris, C., Furlong, M. E., Rogers, E., Wang, L., Bodles, K., & Evans, D. W. Model predictive control of a hybrid autonomous underwater vehicle with experimental verification. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, vol. 228, no. 2, pp. 166–179. https://doi.org/10.1177/1475090213506185.

[4] Gomes, R., & Pereira, F. L. Model Predictive Control for Autonomous Underwater Vehicles. *Procedia Computer Science*, vol. 150, 2019, pp. 19–27. https://doi.org/10.1016/j.procs.2019.02.006.

[5] Yan, Z., Gong, P., Zhang, W., & Wu, W. Model predictive control of autonomous underwater vehicles for trajectory tracking with external disturbances. *Ocean Engineering*, vol. 217, 2020, 107884, pp. 1-10. https://doi.org/10.1016/j.oceaneng.2020.107884.

[6] Uchihori, H., Cavanini, L., Tasaki, M., Majecki, P., Yashiro, Y., Grimble, M. J., Yamamoto, I., van der Molen, G. M., Morinaga, A., & Eguchi, K. Linear

Parameter-Varying Model Predictive Control of AUV for Docking Scenarios. *Applied Sciences*, vol. 11, no. 10, 2021, 4368, pp. 1-22. https://doi.org/10.3390/app11104368.

[7] Misin, M., & Puig, V. LPV MPC Control of an Autonomous Aerial Vehicle. *2020 28th Mediterranean Conference on Control and Automation* (MED), Saint-Raphael, France, 2020, pp. 109-114. https://doi.org/10.1109/med48518.2020.9183041.

[8] Alcala, E., Puig, V., Quevedo, J., & Rosolia, U. Autonomous racing using Linear Parameter Varying-Model Predictive Control (LPV-MPC). *Control Engineering Practice*, vol. 95, 2020, 104270, pp. 1-8. https://doi.org/10.1016/j.conengprac.2019.104270.

[9] Tang, W., Rui, Z., Gao, H., Jiang, H. Constrained missile autopilot design based on model predictive control. *WSEAS Transactions on Systems and Control,* vol. 12, 2017, pp. 123-129. E-ISSN: 2224-2856.

[10] Pshikhopov, V., Medvedev, M. & Gurenko, B.V. Control System Design for Autonomous Underwater Vehicle. *2013 Latin American Robotics Symposium and Competition (LARS/LARC),* Arequipa, Peru, pp. 77-82, 2013. doi: 10.1109/LARS.2013.61.

[11] Garcia-Valdovinos, L., Salgado-Jimenez, T., Bandala-Sanchez, M., Nava-Balanzar, L., Hernandez-Alvarado, R., and Cruz-Ledesma, J. A. Modelling, Design and Robust Control of a Remotely Operated Underwater Vehicle. *International Journal of Advanced Robotic Systems*, vol. 11, no. 1, 2014, pp. 1-16. doi: doi.org/10.5772/56810.

[12] Wang, X., Wang, Q., Su, Y., Ouyang, Y., Sun, C. Adaptive Sensor-Fault Tolerant Control of Unmanned Underwater Vehicles with Input Saturation. IEE/CAA *Journal of Automatica Sinica*, vol. 11, No. 4, pp. 907-918, 2024. doi: 10.1109/JAS.2023.123837.

[13] Kim, T-H. Robust Predictive Control Combined with an Adaptive Mechanism for Constrained Uncertain Systems Subject to Disturbances. *WSEAS Transactions on Systems and Control*, vol. 9, pp. 574-589, 2014**.** E-ISSN: 2224-2856.

[14] Kubalcik, M., Bobal, V. Predictive Control of Higher Order Systems Approximated by Lower Order Time-Delay Models. *WSEAS Transactions on Systems.* vol. 11, pp. 607-616, 2012. E-ISSN: 2224-2678.

[15] Von Alt, C.J. and Grassle, J.F. (1992). LEO-15: An unmanned long-term environmental observatory. *In Proceedings MTS/IEEE Oceans*. Newport, RI, USA, vol.2, pp. 849-854. doi: 10.1109/OCEANS.1992.607695.

[16] Josserand. T. M. *Optimally-Robust Nonlinear Control of a Class of Robotic Underwater Vehicles.* PhD Thesis. The University of Texas at Austin, 2006, [Online]. http://hdl.handle.net/2152/3452 (Accessed Date: October 12, 2024).

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

- Mehmet Avinc has implemented proposed controller, investigation, executed simulation and developed classical controller, writing.
- Chingiz Hajiyev carried out the conceptualization, methodology, validation, review & editing.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself
No funding was received for conducting this study.

Conflict of Interest
The authors have no conflicts of interest to declare.