Active Thruster Fault Diagnosis for an Overactuated Autonomous Surface Vessel

Tsolakis, Anastasios; Ferranti, Laura; Reppa, Vasso

DOI
10.1016/j.ifacol.2024.07.191

Publication date
2024

Document Version
Final published version

Published in
IFAC-PapersOnline

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 58-4 (2024) 43–48

Active Thruster Fault Diagnosis for an Overactuated Autonomous Surface Vessel

Anastasios Tsolakis * **Laura Ferranti** * **Vasso Reppa** *

* *Faculty of Mechanical Engineering, 2628 CD Delft, The Netherlands (e-mail: {a.tsolakis, l.ferranti, v.reppa}@tudelft.nl)*

Abstract: As Autonomous Surface Vessels (ASVs) become increasingly prevalent in marine applications, ensuring their safe operation, in the presence of faults, is crucial to human safety. This paper presents a scheme that encompasses the detection and isolation of actuator faults within ASVs to ensure uninterrupted and safe operation. The method primarily addresses the loss of thruster effectiveness as a specific actuator fault. For fault detection, the proposed method leverages residuals generated by nonlinear observers, coupled with adaptive thresholds, enhancing fault detection accuracy. The active fault isolation strategy employs actuator redundancy to insulate specific system states from faults by dynamically reconfiguring the actuation configuration in response to detected faults. Comprehensive simulation results demonstrate the effectiveness of this methodology across diverse marine traffic scenarios where the ASV needs to perform a collision avoidance maneuver.

Keywords: Autonomous Surface Vessels, Active Fault Diagnosis, Actuator Faults

1. INTRODUCTION

In recent years, there has been a strong interest in developing autonomous solutions for marine systems, spanning various applications. These include for example autonomous surface vessels in the transportation of passengers and goods, unmanned surface vessels for environmental monitoring and bathymetric mapping, and autonomous underwater vehicles employed in tasks such as exploration and inspection of underwater structures. While these constitute promising, cost-effective solutions that could enhance efficiency, there are still concerns regarding the safe operation and reliability of these systems especially in environments shared with other human-operated vehicles.

As autonomous vehicles heavily rely on components such as sensors, actuators, computation units, and various other systems, a major concern revolves around the potential consequences of component faults or complete failures during operation. Recent research has predominantly focused on Fault-Tolerant Control (FTC) for these systems, aiming to maintain system functionality or ensure safety despite the occurrence of faults or failures. This paper focuses on Fault Diagnosis (FD), a critical component of FTC that aims to enhance the system's health understanding. Our work highlights the importance of accurately identifying and localizing faults, thereby improving the safety and reliability of autonomous marine vessels.

⋆ The research leading to these results has received funding from the Dutch Science Foundation NWO-TTW within the Veni project No. 18165 (HARMONIA) and the European Union's Horizon 2020 research and innovation program under grant agreement No. 101096923 (SEAMLESS Project). This publication reflects only the authors' view, exempting the European Union and the granting authority from any liability.

Fault Diagnosis encompasses both Fault Detection and Isolation (FDI), enabling the system to precisely pinpoint and identify the nature and location of the fault. For instance, in Corradini et al. (2010), an actuator fault-tolerant control scheme designed for an underwater Remotely Operated Vehicle (ROV) integrates detection, isolation, and accommodation modules. This work relies on residual generation modules for detection and exploits the specific actuator configuration for isolation through the sliding surface of a designed sliding mode controller (Cocquempot et al. (1998)). The same ROV was studied in Freddi et al. (2013) where the authors focus on the problem of detection only, based on a nonlinear Thau observer for residual generation and on a sequential change detection algorithm for residual evaluation. In Baldini et al. (2022a) an active FD method is proposed for the same system so that actuator faults can be discerned from other disturbances by applying an auxiliary sinusoidal input system that is designed to propagate into the control system when a fault occurs while having minimal impact on the system dynamics. In Baldini et al. (2022b) a bank of observers is used for FDI in cascade with a nonlinear disturbance observer for fault estimation under the assumption that only a single fault may occur. Fault detection was studied for an underactuated surface vessel in Park and Yoo (2016) where a robust fault detection observer and a time-varying detection criterion are presented to detect the actuator faults distinguished from uncertainties and external disturbances. An FTC strategy for linear systems is proposed in Cristofaro and Johansen (2014) with active FD that relies on the control redundancy of an overactuated ASV by constraining the inputs in prescribed configurations for Fault Detection, Isolation, and Reconfiguration (FDIR). However, this work relied on the linearization of vessel dynamics, assuming that the vessel's rotation is negligible

10.1016/j.ifacol.2024.07.191

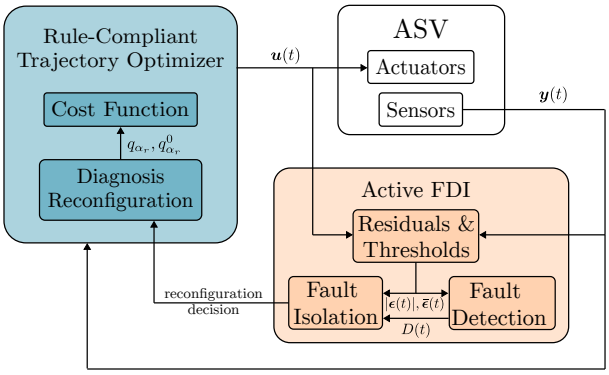

Fig. 1. Schematic method overview (light orange block). Fault detection and isolation are realized given the input $\boldsymbol{u}(t)$ and measurement $\boldsymbol{y}(t)$.

with respect to translation motion, which might not hold in collision avoidance maneuvers.

This paper introduces an active FDI scheme designed to work in tandem with a rule-compliant trajectory optimization algorithm for ASVs proposed in Tsolakis et al. (2024). Our primary contribution is the development of a planning-integrated active FD algorithm capable of detecting and isolating actuator faults, enhancing overall safety by proactively accounting for actuator faults. For fault detection, unlike existing methods that use fixed or heuristic thresholds, we derive adaptive thresholds that are dynamically adjusted based on the system's nonlinearities improving fault detection accuracy while also accounting for bounded noise and disturbances. For fault isolation, our method leverages control redundancy alongside the explicit description of model dynamics and input constraints in the Model Predictive Control (MPC) formulation of Tsolakis et al. (2024), facilitating the isolation process without the need for auxiliary modules for control allocation and input saturation in contrast to previous works.

This paper is organized as follows: Section 2 describes the formulated FDI problem for a 3 Degrees of Freedom (DoF) ASV under environmental disturbances, measurement noise, and specific actuator faults. Section 3 details the fault diagnosis method, which involves a cascaded detection and isolation procedure. Finally, Section 4 presents simulation results and Section 5 concludes the paper with some additional remarks on future work.

2. PROBLEM FORMULATION

We consider the ASV dynamics as a 3-DoF system in planar motion. We assume that the ASV is already equipped with a set of sensors and actuators as well as the trajectory optimizer developed in Tsolakis et al. (2024) for rule-compliant collision avoidance (highlighted in blue in Figure 1). In this work, we focus on developing the "FDI" block, highlighted in orange in Figure 1, that utilizes the system's input and output data.

The vessel dynamics are described by the maneuvering model in Fossen (2011). The ASV's configuration is described by its position $\boldsymbol{p} = (x, y)^\top$, orientation ψ, longitudinal and lateral velocities u, v, and yaw rate r. Note that the velocities are expressed in the body reference frame of the vessel. We then denote as $\boldsymbol{z} = (x, y, \psi, u, v, r)^\top \in \mathcal{Z} \subset$

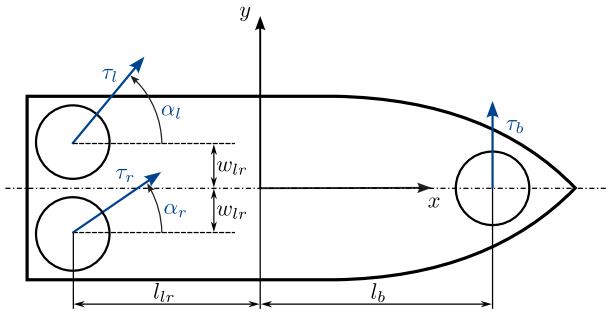

Fig. 2. Schematic representation of the actuators' configuration with two azimuth thrusters at the stern and one bow thruster.

$\mathbb{R}^6$ the system's state and as $\boldsymbol{u} = (\tau_l, \tau_r, \tau_b, \alpha_l, \alpha_r)^\top \in \mathcal{U} \subset \mathbb{R}^5$ the control input of an overactuated ASV with two azimuth thrusters at its beam and one bow thruster. Specifically, we denote as τ_l, τ_r, and α_l, α_r the thrusts and azimuths of the left and right azimuth thruster respectively, and as τ_b the thrust produced by the bow thruster of the ASV. Environmental disturbance forces from the wind and waves are denoted as $\boldsymbol{\tau_d} \in \mathcal{D} \subset \mathbb{R}^3$. The evolution of the system's state is expressed by the following continuous, nonlinear system:

$$\dot{\boldsymbol{z}} = \underbrace{\begin{bmatrix} \boldsymbol{0}_{3\times3} & \boldsymbol{R}(\boldsymbol{z}) \\ \boldsymbol{0}_{3\times3} & -\boldsymbol{M}^{-1}(\boldsymbol{C}(\boldsymbol{z}) + \boldsymbol{D}(\boldsymbol{z})) \end{bmatrix} \boldsymbol{z}}_{\boldsymbol{f}(\boldsymbol{z})} + \underbrace{\overbrace{\begin{bmatrix} \boldsymbol{0}_{3\times3} \\ \boldsymbol{M}^{-1} \end{bmatrix}}^{\tilde{\boldsymbol{M}}} \boldsymbol{\tau}(\boldsymbol{u})}_{\boldsymbol{g}(\boldsymbol{u})} + \underbrace{\overbrace{\begin{bmatrix} \boldsymbol{0}_{3\times3} \\ \boldsymbol{M}^{-1} \end{bmatrix}}^{\tilde{\boldsymbol{M}}} \boldsymbol{\tau_d}}_{\boldsymbol{d}} \tag{1a}$$

with:

$$\boldsymbol{M} = \boldsymbol{M_{RB}} + \boldsymbol{M_A}, \tag{1b}$$

$$\boldsymbol{C}(\boldsymbol{z}) = \boldsymbol{C_{RB}}(\boldsymbol{z}) + \boldsymbol{C_A}(\boldsymbol{z}), \tag{1c}$$

$$\boldsymbol{D}(\boldsymbol{z}) = \boldsymbol{D_L} + \boldsymbol{D_{NL}}(\boldsymbol{z}), \tag{1d}$$

$$\boldsymbol{\tau}(\boldsymbol{u}) = \begin{pmatrix} \eta_l \tau_l \cos\alpha_l + \eta_r \tau_r \cos\alpha_r \\ \eta_l \tau_l \sin\alpha_l + \eta_r \tau_r \sin\alpha_r + \eta_b \tau_b \\ w_{lr}(\eta_r \tau_r \cos\alpha_r - \eta_l \tau_l \cos\alpha_l) - \\ \quad l_{lr}(\eta_l \tau_l \sin\alpha_l - \eta_r \tau_r \cos\alpha_r) + l_b \eta_b \tau_b \end{pmatrix} \tag{1e}$$

where $\boldsymbol{R}(\boldsymbol{z})$ is the rotation matrix, $\boldsymbol{M_{RB}}$ the rigid-body mass matrix, $\boldsymbol{C_{RB}}(\boldsymbol{z})$ the rigid-body Coriolis and centripetal matrix, $\boldsymbol{M_A}$ the added-mass matrix, $\boldsymbol{C_A}(\boldsymbol{z})$ the added Coriolis and centripetal matrix, $\boldsymbol{D_L}$, $\boldsymbol{D_{NL}}(\boldsymbol{z})$, the linear and nonlinear damping matrices, $\boldsymbol{\tau}$ the generalized force vector acting on the vessel, and w_{lr}, l_{lr}, l_b are length parameters that describe the configuration of the thrusters. The added-mass and Coriolis matrices are introduced due to hydrodynamic forces when we consider the additional forces resulting from the fluid acting on the vessel. The thrust force from the actuators in healthy conditions is denoted as $\boldsymbol{\tau}(\boldsymbol{u})$ with $\{\eta_l, \eta_r, \eta_b\}$ fault parameters described at the end of this section. Actuator limitations are considered as well. The actuators' configuration is illustrated in Figure 2.

We model disturbances based on Du et al. (2021) where the prevailing disturbance force is due to the wind, and then wave and current disturbance forces are due to the

wind forces. We assume that this disturbance is unknown but with a known upper bound denoted as $\bar{\boldsymbol{\tau}}_{\boldsymbol{d}}$. We, therefore, model disturbance as a truncated Gaussian random variable with mean $\boldsymbol{\mu_d}$ and variance $\boldsymbol{\Sigma_d}$:

$$\boldsymbol{\tau_d} \sim \mathcal{N}(\boldsymbol{\mu_d}, \boldsymbol{\Sigma_d}) \quad \text{for} \quad \boldsymbol{\mu_d} - 2\boldsymbol{\Sigma_d} \leq \boldsymbol{\tau_d} \leq \boldsymbol{\mu_d} - 2\boldsymbol{\Sigma_d} \tag{2}$$

We assume that we have access to a full-state measurement that is corrupted by a noise signal $\boldsymbol{n}$ that is unknown but bounded with the bound denoted as $\bar{\boldsymbol{n}}$:

$$\boldsymbol{y} = \boldsymbol{z} + \boldsymbol{n} \tag{3}$$

We model noise measurement as a zero-mean truncated Gaussian random variable with variance $\boldsymbol{\Sigma_n}$:

$$\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma_n}) \quad \text{for} \quad -2\boldsymbol{\Sigma_n} \leq \boldsymbol{n} \leq -2\boldsymbol{\Sigma_n} \tag{4}$$

Lastly, for fault modeling, we consider actuator faults and more specifically thruster loss of effectiveness (LoE), which is widely considered as a relevant actuator fault Cristofaro and Johansen (2014); Chen et al. (2016); Park and Yoo (2016); Baldini et al. (2022b). The fault parameters in (1e) are then given as:

$$\eta_j = \begin{cases} 1, & t < t_{f_i} \\ 0 < \eta_j < 1, & t \geq t_{f_i} \end{cases} \tag{5}$$

For healthy conditions, we have $\eta_j = 1$ while $\eta_j = 0$ means complete failure. Time instant t_{f_i} denotes the time a fault occurs. We further assume that the fault happens abruptly after fault time t_{f_i} and that only single faults occur since in practice it is infrequent that two or more actuator faults occur simultaneously (Baldini et al. (2022b); Wang (2020)). We also assume that there are no sensor faults affecting the system.

The goal of this work is to develop a scheme that can detect and isolate parameters η_j, $i \in \{l, r, b\}$ when these deviate from healthy conditions (i.e., when $\eta_j \neq 1$) under disturbances (2) and measurement noise (4).

3. ACTIVE THRUSTER FAULT DIAGNOSIS

3.1 Residuals & Thresholds

The system equations (1) and (3) can be re-written in compact form as:

$$\dot{\boldsymbol{z}} = \boldsymbol{f}(\boldsymbol{z}) + \boldsymbol{g}(\boldsymbol{u}) + \boldsymbol{d} \tag{6a}$$

$$\boldsymbol{y} = \boldsymbol{z} + \boldsymbol{n} \tag{6b}$$

We design a nonlinear observer to generate residuals and the respective adaptive thresholds. The nonlinear observer can be expressed as:

$$\dot{\hat{\boldsymbol{z}}} = \boldsymbol{f}(\hat{\boldsymbol{z}}) + \boldsymbol{g}^H(\boldsymbol{u}) + \boldsymbol{\Lambda}(\boldsymbol{y} - \hat{\boldsymbol{y}}) \tag{7a}$$

$$\hat{\boldsymbol{y}} = \hat{\boldsymbol{z}} \tag{7b}$$

where $\hat{\boldsymbol{z}}$ is the state estimate vector and $\boldsymbol{\Lambda}$ is the observer gain which is a positive definite diagonal matrix and $\boldsymbol{g}^H(\boldsymbol{u})$ denotes the input map in healthy conditions i.e., when there are no actuator faults and $\eta_j = 1$, $\forall j \in \{l, r, b\}$. The residual is expressed as:

$$\boldsymbol{\epsilon} = \boldsymbol{y} - \hat{\boldsymbol{y}} \tag{8}$$

Substituting (6b) to (8) and using the triangle inequality we get:

$$|\underbrace{\boldsymbol{y} - \hat{\boldsymbol{y}}}_{\text{residual}}| \leq |\tilde{\boldsymbol{z}}| + \bar{\boldsymbol{n}} \tag{9}$$

with $\tilde{\boldsymbol{z}} = \boldsymbol{z} - \hat{\boldsymbol{z}}$ the state error. Note that inequalities between matrices are to be interpreted element-wise where $|\cdot|$ denotes the matrix modulus function, i.e., the element-wise absolute value as in Johansson et al. (2006). In the following, we derive the expressions on the two sides of (9) for detection and isolation.

For the adaptive threshold given in the right-hand side of (9), we have the known noise bound $\bar{\boldsymbol{n}}$. A bound on the state error $\tilde{\boldsymbol{z}}$, however, is more involved to derive. Following the same approach of Reppa et al. (2016), we first derive the state error dynamics by subtracting (7a) from (6a):

$$\dot{\tilde{\boldsymbol{z}}} = \boldsymbol{f}(\boldsymbol{z}) - \boldsymbol{f}(\hat{\boldsymbol{z}}) + \boldsymbol{g}(\boldsymbol{u}) - \boldsymbol{g}^H(\boldsymbol{u}) - \boldsymbol{\Lambda}\tilde{\boldsymbol{z}} - \boldsymbol{\Lambda}\boldsymbol{n} + \boldsymbol{d} \tag{10}$$

If we further assume healthy conditions, (10) takes the form:

$$\dot{\tilde{\boldsymbol{z}}} = \boldsymbol{f}(\boldsymbol{z}) - \boldsymbol{f}(\hat{\boldsymbol{z}}) + \boldsymbol{\Lambda}\tilde{\boldsymbol{z}} - \boldsymbol{\Lambda}\boldsymbol{n} + \boldsymbol{d} \tag{11}$$

After rearranging terms and integrating both sides of the equation we get:

$$\tilde{\boldsymbol{z}} = e^{-\boldsymbol{\Lambda}t}\tilde{\boldsymbol{z}}(0) + \int_0^t e^{\boldsymbol{\Lambda}(\tau - t)} \left[\boldsymbol{f}(\boldsymbol{z}) - \boldsymbol{f}(\hat{\boldsymbol{z}}) - \Lambda\boldsymbol{n} + \boldsymbol{d}\right] d\tau \tag{12}$$

This equation cannot be evaluated as $\boldsymbol{n}$ and $\boldsymbol{d}$ are unknown. Nevertheless, we can look for a proper bound of $\tilde{\boldsymbol{z}}$ based on the boundness assumptions for $\boldsymbol{n}$ and $\boldsymbol{d}$. We then have:

$$|\tilde{\boldsymbol{z}}| = \left| e^{-\boldsymbol{\Lambda}t}\tilde{\boldsymbol{z}}(0) + \int_0^t e^{\boldsymbol{\Lambda}(\tau - t)} \left[\boldsymbol{f}(\boldsymbol{z}) - \boldsymbol{f}(\hat{\boldsymbol{z}}) - \Lambda\boldsymbol{n} + \boldsymbol{d}\right] d\tau \right| \tag{13}$$

which by leveraging the triangle inequality becomes:

$$|\tilde{\boldsymbol{z}}| \leq \underbrace{\left|e^{-\boldsymbol{\Lambda}t}\right| |\tilde{\boldsymbol{z}}(0)|}_{\boldsymbol{\alpha}} + \underbrace{\int_0^t \left|e^{\boldsymbol{\Lambda}(\tau - t)}\right| \underbrace{\left(|\boldsymbol{f}(\boldsymbol{z}) - \boldsymbol{f}(\hat{\boldsymbol{z}})| + \Lambda\bar{\boldsymbol{n}} + \bar{\boldsymbol{d}}\right)}_{\boldsymbol{\xi}} d\tau}_{\boldsymbol{\beta}} \tag{14}$$

The homogeneous term $\boldsymbol{\alpha}$ depends on the initial state error for which we have $|\tilde{\boldsymbol{z}}(0)| \leq |\boldsymbol{y}(0)| + \bar{\boldsymbol{n}} + \bar{\hat{\boldsymbol{x}}}$ where $\bar{\hat{\boldsymbol{x}}}$ is a known bound in the initial state estimate. This term will die out because of the exponential term in a negative power. Term $\boldsymbol{\beta}$ can be computed by numerical integration of $\dot{\boldsymbol{\beta}} = -\boldsymbol{\Lambda}\boldsymbol{\beta} + \boldsymbol{\xi}$ with zero initial conditions. While we can have the state estimate $\hat{\boldsymbol{z}}$ to evaluate $\boldsymbol{f}(\hat{\boldsymbol{z}})$ in (14), the state $\boldsymbol{z}$ is unknown and thus the term $\boldsymbol{f}(\boldsymbol{z})$ is unknown as well. Nevertheless, we can substitute $\boldsymbol{z} = \boldsymbol{y} - \boldsymbol{n}$ and then expand the expression $\boldsymbol{f}(\boldsymbol{y} - \boldsymbol{n})$ by leveraging again the triangular inequality to get an upper bound for the right-hand side of (14). Thus, a bound for the right-hand side of (8) can be computed and will be denoted as $\bar{\boldsymbol{\epsilon}}$ with:

$$|\tilde{\boldsymbol{z}}| + \bar{\boldsymbol{n}} \leq \underbrace{\bar{\boldsymbol{\epsilon}}}_{\text{threshold}} \tag{15}$$

Thus, we have the following inequality that holds in healthy conditions where both terms can be evaluated:

$$|\boldsymbol{\epsilon}(\boldsymbol{y}, \hat{\boldsymbol{y}})| \leq \bar{\boldsymbol{\epsilon}}(\boldsymbol{y}, \hat{\boldsymbol{y}}, \bar{\boldsymbol{n}}, \bar{\boldsymbol{d}}) \tag{16}$$

The adaptive threshold varies with respect to the estimate of the system while it takes into account the corruption of this signal from worst-case disturbance and noise signals.

3.2 Fault Diagnosis

The presence of actuator faults is detected by the following set of analytical redundancy relations (ARRs):

	η_l	η_r	η_b
$\mathcal{E}_u$	1	1	0
$\mathcal{E}_v$	1	1	1
$\mathcal{E}_r$	1	1	1

Table 1. Actuator FSM $\boldsymbol{F}$

$$\mathcal{E}_i : |\epsilon_i(t)| - \bar{\epsilon}_i(t) \leq 0, i \in \{x, y, \psi, u, v, r\} \tag{17}$$

where $|\epsilon_i(t)|$ and $\bar{\epsilon}_i(t)$ are the elements of $|\boldsymbol{\epsilon}(t)|$ and $\bar{\boldsymbol{\epsilon}}(t)$ in (16) respectively. Violation of one of these ARRs at any time instance means that the real system is behaving significantly differently with respect to the healthy system model used in the nonlinear observer. Since this discrepancy is not due to measurement noise or disturbances as they have already been accounted for, we can then conclude that a fault has occurred. The first time instant that (17) is invalid for at least one $i \in \{x, y, \psi, u, v, r\}$ signifies the time instant of fault detection defined as:

$$t_{D_i} = \min\{t : |\epsilon_i(t)| - \bar{\epsilon}_i(t) > 0\} \tag{18}$$

Until this instant, we assume that either no faults have occurred or there are faults that have not been detected yet. The binary decision for a detected fault is defined as:

$$D(t) = \begin{cases} 0, & t < t_D \\ 1, & t \geq t_D \end{cases} \tag{19}$$

with $t_D = \min\{t_{D_i} : i \in \{x, y, \psi, u, v, r\}\}$. Thus a fault is detected at any time when $D(t) = 1$.

For isolation, we investigate how each one of the LoE faults η_j, $j \in \{l, r, b\}$ affects the system. First, we need to create a binary Fault Signature Matrix (FSM) as a reference and then a binary decision vector that through comparison with the FSM will pinpoint the exact location of the fault. For the FSM, we need to find the effect of each actuator fault on the residuals, that is, how the discrepancy due to the occurring faults denoted as $\tilde{\boldsymbol{g}}(\boldsymbol{u}) = \boldsymbol{g}(\boldsymbol{u}) - \boldsymbol{g}^H(\boldsymbol{u})$ affects the error dynamics (10). We compute the Jacobian of $\tilde{\boldsymbol{g}}(\boldsymbol{u})$ with respect to the vector of faults $\boldsymbol{\eta} = (\eta_l, \eta_r, \eta_b)^\top$ as:

$$\boldsymbol{\nabla}_{\boldsymbol{\eta}} \tilde{\boldsymbol{g}}(\boldsymbol{u}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \tilde{\boldsymbol{g}}_{\boldsymbol{ul}}(\tau_l, \alpha_l) & \tilde{\boldsymbol{g}}_{\boldsymbol{ur}}(\tau_r, \alpha_r) & 0 \\ \tilde{\boldsymbol{g}}_{\boldsymbol{vl}}(\tau_l, \alpha_l) & \tilde{\boldsymbol{g}}_{\boldsymbol{vr}}(\tau_r, \alpha_r) & \tilde{\boldsymbol{g}}_{\boldsymbol{vb}}(\tau_b, \alpha_b) \\ \tilde{\boldsymbol{g}}_{\boldsymbol{rl}}(\tau_l, \alpha_l) & \tilde{\boldsymbol{g}}_{\boldsymbol{rr}}(\tau_r, \alpha_r) & \tilde{\boldsymbol{g}}_{\boldsymbol{rb}}(\tau_b, \alpha_b) \end{bmatrix} \tag{20}$$

The Jacobian matrix, commonly used in sensitivity analysis, captures the rate of change of the system's output (ARRs in our case) concerning small changes in the actuator faults. Each row of this matrix corresponds to a specific ARR $\mathcal{E}_i$, $i \in \{x, y, \psi, u, v, r\}$, and each column corresponds to a particular actuator fault η_j, $j \in \{l, r, b\}$, offering insights into the impact of each fault on the ARRs. After computing the Jacobian matrix (20), it is observed that the first three ARRs ($\mathcal{E}_i$, $i \in \{x, y, \psi\}$) do not exhibit sensitivity to the actuator faults (η_j, $j \in \{l, r, b\}$). Consequently, these states are omitted from the FSM, as their inclusion would not contribute valuable information regarding fault isolation. The FSM matrix, denoted as $\boldsymbol{F}$, is constructed based on the relevant ARRs ($\mathcal{E}_i$ where $i \in \{u, v, r\}$), resulting in a focused representation that captures the impact of actuator faults on the observable system dynamics. The matrix is shown in Table 1.

	η_l	η_r	η_b
$\mathcal{E}_u$	1	1	0
$\mathcal{E}_v$	0	1	1
$\mathcal{E}_r$	1	1	1

(a) FSM $\boldsymbol{F_l^R}$

	η_l	η_r	η_b
$\mathcal{E}_u$	1	1	0
$\mathcal{E}_v$	1	0	1
$\mathcal{E}_r$	1	1	1

(b) FSM $\boldsymbol{F_r^R}$

Table 2. FSMs after reconfiguration

From Table 1, we can deduce that due to the geometrical symmetry of the actuators, they affect the dynamics in the same way. Geometrical symmetry in this context refers to the similar spatial arrangement or characteristics of the actuators. Specifically, the actuator faults η_l and η_r exhibit identical impacts on the ARRs of the system and thus, distinguishing between these faults becomes challenging. By utilizing overactuation, the key idea in this work is to actively change the input vector $\boldsymbol{u}$ so that the effect of some actuators is nullified in specific ARRs and thus, the faults can be isolable. Furthermore, this needs to be realized while keeping the ASV controllable and able to perform the collision avoidance maneuver needed. Indeed, if we examine matrix $\boldsymbol{F}$ in Table 1 we see that the two columns that correspond to faults η_l and η_r are identical. However, by placing a zero in either column at the row that corresponds to ARR, $\mathcal{E}_v$, all three columns become linearly independent, as shown in Table 2. This can be realized by setting the corresponding term of (20) to zero, meaning that we want to make that component invariant of the corresponding control action. Solving either $\tilde{\boldsymbol{g}}_{\boldsymbol{vl}}(\tau_l, \alpha_l) = 0$ for α_l or $\tilde{\boldsymbol{g}}_{\boldsymbol{vr}}(\tau_r, \alpha_r) = 0$ for α_r, we get expressions for the azimuths of the form:

$$\alpha_l = \alpha_l^0(\tilde{\boldsymbol{M}}, l_{lr}, w_{lr}), \quad \alpha_r = \alpha_r^0(\tilde{\boldsymbol{M}}, l_{lr}, w_{lr}), \tag{21}$$

and since these parameters are constant, α_l and α_r can be set to the constant values α_l^0 and α_r^0 respectively so that they do not affect the sway dynamics and the corresponding ARR, that is, either $\tilde{\boldsymbol{g}}_{\boldsymbol{vl}}(\tau_l, \alpha_l^0) = 0$ or $\tilde{\boldsymbol{g}}_{\boldsymbol{vr}}(\tau_r, \alpha_r^0) = 0$. Choosing for example to nullify the effect of the right azimuth thruster will result in a different FSM matrix denoted as $\boldsymbol{F_r^R}$ and shown in Table 2b. Note that fulfillment of either equation in (21) does not nullify the rest of the terms in (20). To reconfigure the actuators for isolation purposes, we leverage the MPC controller in Tsolakis et al. (2024) (see Fig. 1). The MPC recursively optimizes a multi-objective cost that includes a penalty on the control inputs accounting at the same time for collision avoidance and system constraints. The part of the cost function of our MPC controller (relevant for the reconfiguration here) regarding the right azimuth input takes the form:

$$J_{\alpha_r}(\boldsymbol{u}_k) = q_{\alpha_r} \alpha_r^2 + q_{\alpha_r^0}(\alpha_r - \alpha_r^0)^2 \tag{22}$$

where q_{α_r} and $q_{\alpha_r^0}$ are tuning penalty weights and their value depends on whether we are in normal conditions or the reconfiguration mode is activated. More specifically, a higher value of $q_{\alpha_r^0}$ *forces* the optimizer to stir $\alpha_r \to \alpha_r^0$. If the actuator is faulty, then, the real system will not be able to follow the commanded action. If it is healthy, we are able to isolate the fault on the other actuator. To complete the isolation, we need to derive the binary decision vector to be compared with the updated FSM $\boldsymbol{F^r}$. The binary decision vector is obtained as:

$$\boldsymbol{D} = (D_u, D_v, D_r)^\top \tag{23}$$

with:

$$D_i(t) = \begin{cases} 0, & t < t_{D_i} \\ 1, & t \geq t_{D_i} \end{cases} \tag{24}$$

with $t_{D_i} : i \in \{u, v, r\}$ the time that the i^{th} ARR is violated for the first time. The active FDI logic is summarized in Algorithm 1.

Algorithm 1 Proposed Fault Diagnosis Logic

Input: $\boldsymbol{y}(t)$, $\boldsymbol{u}(t)$
Output: Fault ID: $\{r, l, b\}$
$q_{\alpha_r} \leftarrow 1$
$q_{\alpha_r^0} \leftarrow 0$
for $t = 1, 2, \dots$ **do**
  Compute $|\boldsymbol{\epsilon}(\boldsymbol{y}(t), \hat{\boldsymbol{y}}(t))|$ from Eq. (8)
  Compute $\bar{\boldsymbol{\epsilon}}(\boldsymbol{y}, \hat{\boldsymbol{y}}, \bar{\boldsymbol{n}}, \bar{\boldsymbol{d}})$ from Eq. (15)
  Compute $D(t)$ from Eq. (19)
  if $D(t) = 1$ **then**
    Compute $\boldsymbol{D}(t)$ from Eq. (23)
    if $\boldsymbol{D}(t) = \boldsymbol{F}(:, 3)$ **then**
      Fault ID: b ($\eta_b \neq 1$)
    else
      $q_{\alpha_r} \leftarrow 0$
      $q_{\alpha_r^0} \leftarrow 10^6$
      And thus:
      $\alpha_r \leftarrow \alpha_r^0(\tilde{\boldsymbol{M}}, b, l)$
      $\boldsymbol{F} \leftarrow \boldsymbol{F^r}$
      Compute $\boldsymbol{D}(t)$ from Eq. (23)
      if $\boldsymbol{D}(t) = \boldsymbol{F}(:, 2)$ **then**
        Fault ID: r ($\eta_r \neq 1$)
      else if $\boldsymbol{D}(t) = \boldsymbol{F}(:, 1)$ **then**
        Fault ID: l ($\eta_l \neq 1$)
      else
        Fault is not yet isolated
      end if
    end if
  end if
end for

4. SIMULATION RESULTS

This section presents simulation results to validate the efficacy of our algorithm in a simple traffic scenario. Our framework is implemented in ROS: the controller and FDI module in C++ and the simulator of the ASV and OV in Python. The algorithm runs in an Ubuntu machine with an Intel i7 CPU@1.8GHz and 16GB of RAM.

In this simple traffic scenario, the ASV is obliged by the traffic rules to avoid collision by turning to its starboard (right side) and passing behind the OV. While the ASV is performing the collision avoidance maneuver, at time $t_F = 7s$ we inject a permanent fault to the right thruster, $\eta_r = 0.1$. In Figure 3 we can see the instances of the ASV at the time the fault occurs $t_F = 7$ sec, the fault is detected $t_D = 7.86s$, and lastly the fault is isolated $t_I = 14.15s$. Notice that the time between fault occurrence t_F and isolation t_I is relatively small, i.e., the ASV has not traversed a large distance, highlighting the aptness of the diagnosis procedure. Figures 4, 5, and 6 show the norm of the residual coupled with its adaptive threshold and the violation decision for each one of the three ARRs related to the velocity states that contribute to the diagnosis procedure. The first violation is noticed in Figure 4 at $t_D = 7.86s$. Reconfiguration starts subsequently to nullify the effect of the faulty thruster on $\mathcal{E}_v$ as seen in Figure 5. After a few seconds, at time $t_I = 14.15s$ seconds, a

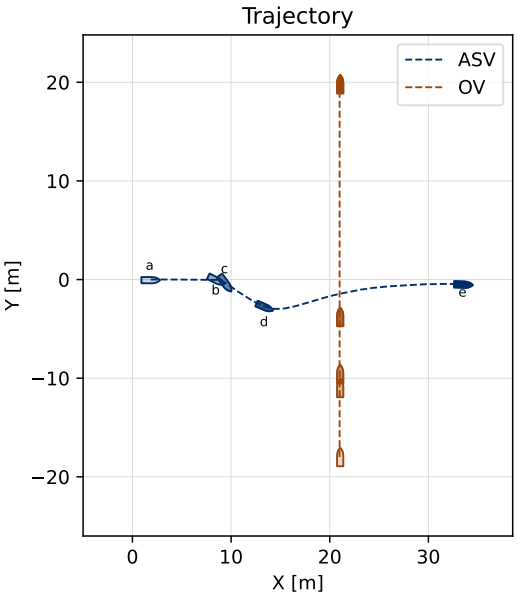

Fig. 3. Plotted trajectories with instances of the vessels at a) initial time, b) time of fault occurrence, c) time of detection, d) time of isolation, and e) final time.

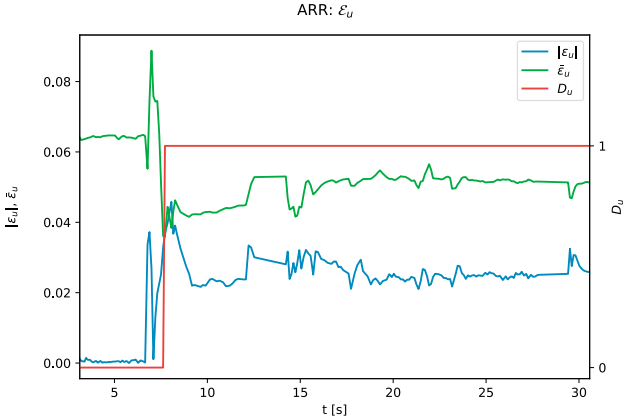

Fig. 4. The residual norm, threshold, and decision for the ARR corresponding to the surge velocity state u.

violation of $\mathcal{E}_r$ indicates that the fault has occurred in the right thruster, and thus, isolation is completed. Lastly, Figure 7 shows the control input in blue solid lines along with the faulty input signal $\tau_r^F = \eta_r \tau_r$ that is applied on the ASV plotted in a light blue dashed line right after the fault has occurred. In the same figure, the reconfigured control input α_r is plotted in red to show the constant value it has been imposed for isolation purposes.

5. CONCLUSIONS & FUTURE WORK

This work proposed a thruster fault diagnosis algorithm for an ASV, comprised of a cascaded interconnection of a detection and an isolation module. The residuals generated by nonlinear observers are coupled with adaptive thresholds that accommodate noise and disturbance bounds to enhance robustness against false alarms. For isolation, we rely on the system's redundancy in actuation and the capability to set actuation constraints in our MPC controller

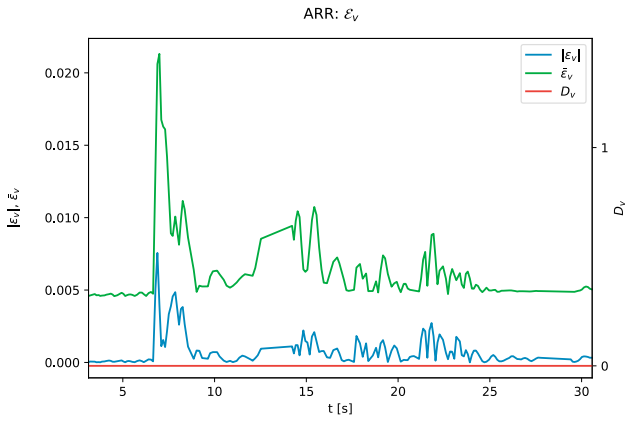

Fig. 5. The residual norm, threshold, and decision for the ARR corresponding to the sway velocity state v.

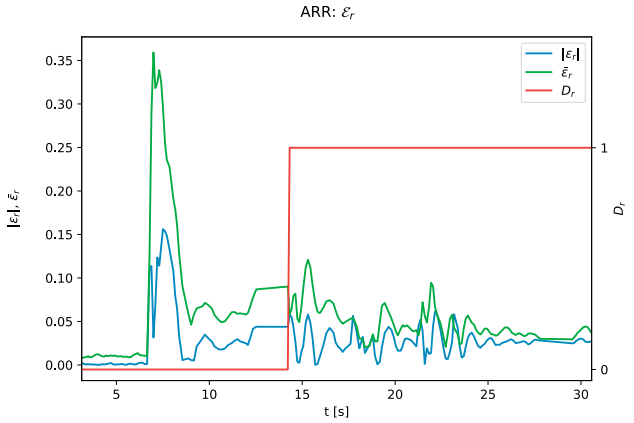

Fig. 6. The residual norm, threshold, and decision for the ARR corresponding to the yaw velocity state r.

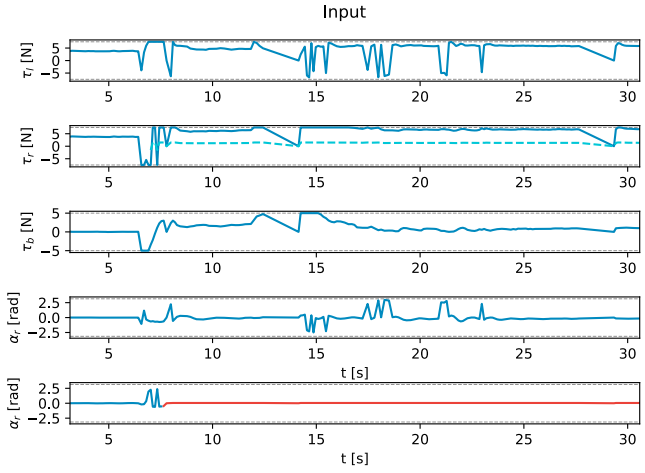

Fig. 7. Input signals with the actual input signal τ_r (light blue dashed line) after the fault has occurred and the input signal α_r (red line) after reconfiguration.

so that we can isolate thruster faults that otherwise would be indistinctive due to the system's symmetry. Simulation results demonstrate the effectiveness of this methodology. For future work, we aim to study the accuracy of the method in more fault scenarios and investigate its sensitivity to minor faults which are harder to detect in the presence of other uncertainties. We also aim to extend our method with fault identification and reconfiguration to influence the planning process, contributing to safer and more adaptive collision avoidance maneuvers.

REFERENCES

Baldini, A., Felicetti, R., Freddi, A., Longhi, S., and Monteriù, A. (2022a). Actuator fault tolerant control via active fault diagnosis for a remotely operated vehicle. *IFAC-PapersOnLine*, 55(6), 310–316.

Baldini, A., Felicetti, R., Freddi, A., Longhi, S., and Monteriù, A. (2022b). Fault tolerant control for remotely operated vehicles with thruster faults using nonlinear disturbance observers. *IFAC-PapersOnLine*, 55(31), 275–280.

Chen, M., Jiang, B., and Cui, R. (2016). Actuator fault-tolerant control of ocean surface vessels with input saturation. *International Journal of Robust and Nonlinear Control*, 26(3), 542–564.

Cocquempot, V., Izadi-Zamanabadi, R., Staroswiecki, M., and Blanke, M. (1998). Residual generation for the ship benchmark using structural approach. In *UKACC International Conference on Control'98 (Conf. Publ. No. 455)*, 1480–1485. IET.

Corradini, M.L., Monteriu, A., and Orlando, G. (2010). An actuator failure tolerant control scheme for an underwater remotely operated vehicle. *IEEE Transactions on Control Systems Technology*, 19(5), 1036–1046.

Cristofaro, A. and Johansen, T.A. (2014). Fault tolerant control allocation using unknown input observers. *Automatica*, 50(7), 1891–1897.

Du, Z., Negenborn, R.R., and Reppa, V. (2021). Cooperative multi-agent control for autonomous ship towing under environmental disturbances. *IEEE/CAA Journal of Automatica Sinica*, 8(8), 1365–1379.

Fossen, T.I. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons.

Freddi, A., Longhi, S., and Monteriu, A. (2013). Actuator fault detection system for a remotely operated vehicle. *IFAC Proceedings Volumes*, 46(33), 356–361.

Johansson, A., Bask, M., and Norlander, T. (2006). Dynamic threshold generators for robust fault detection in linear systems with parameter uncertainty. *Automatica*, 42(7), 1095–1106.

Park, B.S. and Yoo, S.J. (2016). Fault detection and accommodation of saturated actuators for underactuated surface vessels in the presence of nonlinear uncertainties. *Nonlinear Dynamics*, 85, 1067–1077.

Reppa, V., Polycarpou, M.M., Panayiotou, C.G., et al. (2016). Sensor fault diagnosis. *Foundations and Trends® in Systems and Control*, 3(1-2), 1–248.

Tsolakis, A., Negenborn, R.R., Reppa, V., and Ferranti, L. (2024). Model predictive trajectory optimization and control for autonomous surface vessels considering traffic rules. *IEEE Transactions on Intelligent Transportation Systems*.

Wang, X. (2020). Active fault tolerant control for unmanned underwater vehicle with actuator fault and guaranteed transient performance. *IEEE Transactions on Intelligent Vehicles*, 6(3), 470–479.