

# An integrated sensor and process fault diagnosis framework for marine vessels

## Master Thesis

## P. Pantela

Report Number: 2022.MME.8726





# An integrated sensor and process fault diagnosis framework for marine vessels

by

P. Pantela

## Master Thesis

In partial fulfilment of the requirements for the degree of

**Master of Science**

in Mechanical Engineering

at the Department Maritime and Transport Technology of Faculty Mechanical, Maritime and Materials  
Engineering of Delft University of Technology,  
to be defended publicly on Tuesday, December 20, 2022, at 10:00 AM.

Student Number:	5378745	
MSc Track:	Multi-Machine Engineering	
Report Number:	2022.MME.8726	
Supervisor:	Dr. Vasso Reppa, PhD candidate Nikos Kougiatsos	
Project duration:	April 26, 2022 – December 20, 2022	
Thesis committee:	Prof. Dr. X. Jiang, Prof. Dr. V. Reppa, PhD candidate N. Kougiatsos,	TU Delft TU Delft, supervisor TU Delft, supervisor
Date:	20th of December, 2022	

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

It may only be reproduced literally and as a whole. For commercial purposes only with written authorization of Delft University of Technology. Requests for consult are only taken into consideration under the condition that the applicant denies all legal rights on liabilities concerning the contents of the advice.



# Acknowledgements

Initially, I would like to thank and express my deepest gratitude to my parents, family members, and my friends for their endless support over the course of this graduation project, and also throughout the course of my master's studies in general. Furthermore, I would like to thank my supervisors for the guidance and close collaboration we had in all the phases of the graduation project. The information and knowledge that they shared were invaluable and they attributed to the timely delivery of this project. Finally, I want to thank everyone else that provided me with administrative or logistical support within TU Delft and the professors of the Maritime and Transport Technology department that provided me with the background knowledge to complete my studies.

*P. Pantela*  
*Delft, December 2022*



# Abstract

For maritime safety, condition monitoring and fault diagnosis on-board large sea-going vessels is a growing necessity. To this end, measurements from on-board sensors are needed, which are utilized by automated fault diagnosis methodologies. More specifically, such measurements are used to derive residuals that will determine whether a fault has occurred within the propulsion system. Furthermore, the sensors themselves are vulnerable to sensor faults that also need to be detected and isolated. Therefore, in this thesis, an integrated sensor and process fault diagnosis framework for the propulsion process of large marine vessels is presented.

In order to realize this integrated sensor and process fault diagnosis framework, certain steps are taken in the context of this thesis. Firstly, the literature regarding common ship propulsion faults and the state-of-art in fault diagnosis and condition monitoring for marine vessels are reviewed. The tools that will be used to carry out the framework are also reviewed. Then, the model of one of the most common and crucial ship propulsion components, the marine fuel engine is presented. This model was validated, using certain data from Factory Acceptance Tests (FAT). Moreover, a scheme is presented at a later stage to provide the methodology needed to carry out fault detection, isolation, and classification between sensor and process faults. This is achieved namely with the use of distributed monitoring agents, adaptive thresholds, and hardware redundancy. Simulation results regarding fault detection of both sensor and process faults are presented and some examples of multiple fault isolation are provided to showcase the effectiveness of the proposed method. Some performance indicators are also provided to demonstrate the efficiency of the scheme.

This thesis aims to provide a model-based, distributed fault diagnosis framework that will enhance online monitoring of marine propulsion systems to avoid future failures and accidents. Furthermore, the utilization of adaptive thresholds in the monitoring modules of the monitoring agents helps minimize false alarms and reduces the conservativeness in decision-making. Finally, a few conclusions are drawn regarding the performance of the proposed methodology, and suggestions for future research are also discussed.

**Keywords:** Condition monitoring, fault diagnosis, fault detection and isolation, prognostic health maintenance, fault modeling, sensor faults, marine process faults, marine propulsion process, non-linear interconnected systems, distributed fault diagnosis.



# Contents

<b>Acknowledgements</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Nomenclature</b>	<b>ix</b>
<b>List of Figures</b>	<b>xv</b>
<b>List of Tables</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Prologue . . . . .	1
1.2 Definitions and Motivation . . . . .	1
1.2.1 Motivation . . . . .	3
1.3 Development and Contribution. . . . .	3
1.3.1 Research Questions . . . . .	3
1.3.2 Scope . . . . .	3
1.3.3 Contribution. . . . .	4
1.3.4 Thesis Outline . . . . .	4
<b>2 Literature Review</b>	<b>5</b>
2.1 Marine Propulsion Process Faults Overview . . . . .	5
2.2 State of the Art of Monitoring and Fault Diagnosis . . . . .	8
2.2.1 Condition Monitoring . . . . .	8
2.2.2 Fault Diagnosis . . . . .	9
2.3 Fault Modeling and Fault Examples . . . . .	9
2.3.1 Fault Modeling . . . . .	10
2.3.2 Fault Examples . . . . .	10
2.4 Fault Tolerance and Redundancy . . . . .	11
2.4.1 Physical Redundancy . . . . .	11
2.4.2 Analytical Redundancy. . . . .	12
2.4.3 Redundancy in Fuel Engine Fault Diagnosis . . . . .	12
2.5 Multi-Agent Monitoring . . . . .	14
2.5.1 Distributed Fault Diagnosis . . . . .	14
2.6 Observer Based Monitoring . . . . .	15
2.6.1 Luenberger Observers . . . . .	16
2.6.2 Fault Detection with State Observers . . . . .	16
2.6.3 Fault Detection with Nonlinear Observers. . . . .	17
2.7 Distributed Multiple Fault Isolation. . . . .	17
2.7.1 Local Decision Logic for Fault Isolation . . . . .	18
2.7.2 Global Decision Logic for Fault Isolation . . . . .	18
2.8 Similar Works . . . . .	18
2.8.1 Revealed Research Gaps . . . . .	18
2.9 Discussion . . . . .	19
<b>3 Marine Propulsion Process Modeling</b>	<b>21</b>
3.1 General System and Sensor Equations . . . . .	22
3.2 Marine Fuel Engine State-Space Modeling . . . . .	23
3.2.1 Fuel Pump Model. . . . .	23
3.2.2 Engine Block Model . . . . .	23
3.2.3 Exhaust Receiver and Turbocharger Model. . . . .	25
3.2.4 Air Intake Model . . . . .	26

3.3	Modeled Process Faults . . . . .	27
3.4	Discussion . . . . .	28
<b>4</b>	<b>Propulsion Process Model Validation</b>	<b>29</b>
4.1	Fuel Pump Model Validation . . . . .	29
4.2	Diesel Engine Model Validation . . . . .	30
4.3	Exhaust Receiver and Turbocharger Model Validation . . . . .	30
4.4	Air Intake Model Validation . . . . .	32
4.5	Discussion . . . . .	32
<b>5</b>	<b>Fault Diagnosis Scheme</b>	<b>33</b>
5.1	Problem Formulation . . . . .	33
5.2	Assumptions Introduced . . . . .	34
5.3	Distributed Fault Detection and Isolation . . . . .	35
5.3.1	Residual Generation and Error Estimation . . . . .	36
5.3.2	Adaptive Thresholds Computation. . . . .	36
5.3.3	Multiple Fault Decision Logic . . . . .	37
5.4	Discussion . . . . .	39
<b>6</b>	<b>Fault Diagnosis Simulation Results</b>	<b>41</b>
6.1	Detection of Sensor Faults . . . . .	41
6.2	Detection of Process Faults . . . . .	48
6.3	Fault Signature Matrices . . . . .	51
6.4	Multiple Fault Isolation Examples . . . . .	55
6.4.1	Multiple Sensor Faults . . . . .	55
6.4.2	Multiple Process Faults . . . . .	55
6.4.3	Multiple Sensor and Process Faults in Different Systems . . . . .	56
6.4.4	Multiple Sensor and Process Faults in the Same System . . . . .	57
6.5	Key Performance Indicators . . . . .	58
6.5.1	Detection Time Standard Deviation . . . . .	58
6.5.2	Miss-Detection of Process Faults . . . . .	58
6.6	Discussion . . . . .	59
<b>7</b>	<b>Conclusions and Future Directions</b>	<b>61</b>
7.1	Conclusions. . . . .	61
7.2	Future Directions . . . . .	62
7.2.1	Limitations to Consider in Future Research. . . . .	62
7.3	Epilogue. . . . .	62
<b>A</b>	<b>Appendix</b>	<b>63</b>
A.1	Simulation Results for Multiple Faults . . . . .	63
<b>B</b>	<b>Appendix: Seiliger Cycle</b>	<b>77</b>
<b>C</b>	<b>Appendix: Academic Research Paper</b>	<b>81</b>
	<b>Bibliography</b>	<b>94</b>

# Nomenclature

## Symbols

Symbol	Definition [24]	Units
<i>Greek Symbols:</i>		
$\alpha$	Crank angle	[°]
$\alpha_z$	Zinner turbine area decrease factor	[-]
$\alpha_{eff}$	Effective angle of attack	[°]
$\beta$	Evolution mode of faults	[-]
$\gamma$	Nonlinear system dynamics	[-]
$\delta_f$	Fuel addition factor	[-]
$\varepsilon$	Parasitic heat exchanger effectiveness	[-]
$\varepsilon_c$	Geometric compression ratio	[-]
$\varepsilon_j$	Analytical redundancy relation $j$	[-]
$\varepsilon_y$	Residual	[-]
$\bar{\varepsilon}_y, \underline{\varepsilon}_y$	Adaptive thresholds	[-]
$\eta_x, \eta_z$	System disturbances	[-]
$\eta$	Efficiency	[-]
$\kappa$	Evolution rate of faults	[-]
$\kappa_a$	Specific heat ratio of air	[-]
$\kappa_g$	Specific heat ratio of exhaust gas	[-]
$\lambda$	Air excess ratio	[-]
$\lambda_h, \lambda_\gamma$	Lipschitz constants	[-]
$\lambda_{CR}$	Length ratio of the crank rod to the crankshaft radius	[-]
$\mu$	Mean detection time delay of monitoring module	[s]
$\xi$	Smooth vector field	[-]
$\Pi$	Pressure ratio of turbine or compressor	[-]
$\rho_d, \xi_d$	Constants used for state-based adaptive threshold	[-]
$\sigma$	Detection time standard deviation from the mean	[s]
$\tau_p$	Time constant representing pitch actuation delay	[s]
$\tau_\chi$	Fuel injection time delay	[s]
$\tau_{pd}$	Time delay for filling the exhaust receiver	[s]
$\phi$	Fault function	[-]
$\chi_g$	Ratio between the specific heats at constant pressure	[-]
$\chi$	interconnection variables between systems	[-]
$\Psi_{sc}$	Non-dimensional scavenge flow	[-]
<i>Latin Symbols:</i>		
$a, b, c$	Seiliger parameters for isochoric, isobaric and isothermal combustion	[-]
$a_\eta, b_\eta, c_\eta$	Polynomial coefficients of the turbocharger	[-]
$A_{eff}$	Effective area of the turbine	[m <sup>2</sup> ]
$a_{gb}, b_{gb}, c_{gb}$	Gearbox loss function parameters	[-]
$c_p$	Specific heat at constant pressure	[J/kgK]
$c_v$	Specific heat at constant volume	[J/kgK]
$C_Q, C_T$	Torque and thrust coefficient	[-]
$d$	Sensor noise vector	[-]
$D$	Binary decision of module or agent	[-]
$f$	Sensor or process fault equation	[-]

Symbol	Definition [24]	Units
$F$	Fault matrix	[-]
$g$	Standard gravity	[ $m/s^2$ ]
$h$	System interconnection dynamics	[-]
$h_L$	Lower heating value of fuel at ISO conditions	[J/kg]
$i_e$	number of cylinders in the engine	[-]
$k_e$	Number of revolutions per cycle	[-]
$L$	Observer gain	[-]
$m$	Mass	[kg]
$M_l$	Gearbox torque loss	[Nm]
$n_{fe}$	Engine speed	[rev/s]
$n_{bld}$	Polytropic expansion coefficient of blowdown	[-]
$n_{exp}$	Polytropic exponent for expansion	[-]
$P_{fe}$	Engine power	[W]
$p$	Pressure	[Pa]
$P_p$	Propeller pitch	[%]
$p_{max}$	Maximum cylinder pressure	[Pa]
$q$	Specific heat release	[J/kg]
$Q_{loss}^{grad}$	Torque loss gradient	[-]
$Q_{loss}$	Torque loss	[Nm]
$R_a, R_g$	Universal gas constants for air and exhaust gas respectively	[J/kgK]
$R_c$	Fault function increase rate	[Hz]
$R_v$	Ship resistance	[N]
$R_{CS}$	Crankshaft radius	[m]
$r_c$	Effective compression ratio	[-]
$r_{eo}$	Ratio of volume at Seiliger point 6 relative to 1	[-]
$r_{TTC}$	Driving temperature ratio of the turbocharger	[-]
$s_{sl}$	Slip ratio of the scavenge process with bypass air	[-]
$SFC$	Brake specific fuel consumption	[kg/J]
$T$	Temperature	[K]
$T_{bld}$	<i>Zinner</i> blowdown temperature	[K]
$T_{sl}$	Temperature of the air slip during scavenging	[K]
$T_f$	Time of fault occurrence	[s]
$T_D$	Fault detection time	[s]
$t$	Time	[s]
$t_1$	Fault initialization time	[s]
$t_2$	Fault termination time	[s]
$V_i$	Cylinder volume for Seiliger process $i$	[ $m^3$ ]
$w_i$	Specific indicated work	[J/kg]
$X_{ct}$	Portion of heat released at constant temperature	[-]
$X_{cv}$	Portion of heat released at constant volume	[-]
$x$	State variable	[kg or Pa]
$y$	Sensor output	[kg or Nm or Pa or K]
$z$	Algebraic variable	[Nm or Pa or K]

## Abbreviations

---

Abbreviation	Definition
CBM	Condition Based Maintenance
PHM	Prognostic Health Maintenance
CM	Condition Monitoring
RFID	Radio Frequency Identification
MEMS	Micro-Electro-Mechanical System
SCADA	Supervisory Control and Data Acquisition
PEID	Product Embeded Information Devices
O&M	Operations and Maintenance
DIaK	Data, Iformation and Knowledge
FD	Fault Diagnosis
MS&E	Marine Systems and Equipment
GTE	Gas Turbine Engine
CODLAG	Combined diesel and Gas
DC	Direct Current
FDI	Fault Detection and Isolation
FAT	Fault Acceptance Test
HMS	Health Monitoring System
ADC	Analog to Digital Converter
ML	Machine Learning
AI	Artificial Intelligence
FTC	Fault Tolerant Control
DAE	Differential and Algebraic Equation
ARR	Analytical Redundancy Relation
MVFP	Mean Value First Principle
CPP	Controllable Pitch Propeller
MOE	Measures of Effectiveness
SAT	Sea Acceptance Trials
KPI	Key Performance Indicator
ROV	Remotely Operated Vehicle

---



# List of Figures

1.1	Relationship diagram between CM, FD and health prognosis [74]. . . . .	2
2.1	FREMM frigates in open sea [12]. . . . .	9
2.2	Time-dependency of faults: <i>a) abrupt; b) incipient; c) intermittent</i> [34]. . . . .	10
2.3	Basic models of faults: <i>a) additive fault; b) multiplicative fault</i> [34]. . . . .	10
2.4	Examples of fault models with various time duration (intermittent, transient, permanent), time profiles (incipient, abrupt), and fault functions (offset, drift, precision degradation). In the case of the intermittent faults, only the abrupt evolution mode of occurrence and disappearance is presented [57]. . . . .	11
2.5	Basic scheme of a fault-tolerant system with parallel function modules as redundancy [35].	11
2.6	Residual space and control region in the two-dimensional case [70]. . . . .	13
2.7	Local controller and diagnoser [8]. . . . .	15
2.8	Block diagram showing the system model and the observer signal flow [21]. . . . .	15
3.1	Schematic representation of the propulsion systems of a marine vessel [44]. . . . .	21
3.2	Schematic representation of direct drive propulsion system for a marine vessel showing causal coupling between models [24]. . . . .	22
4.1	Specific fuel consumption vs engine speed. . . . .	29
4.2	Maximum combustion pressure at stage 4 vs engine speed. . . . .	30
4.3	Exhaust receiver temperature vs engine speed. . . . .	30
4.4	Exhaust temperature after turbocharger vs engine speed. . . . .	30
4.5	Charged air pressure vs engine speed. . . . .	32
5.1	Distributed FDI scheme application using an MVFP marine propulsion model [44]. . . .	35
5.2	Description of monitoring agent $\mathcal{M}^{(l)}$ , system $\Sigma^{(l)}$ , and sensors $\delta^{(l)}$ connected to it. . . .	35
6.1	No sensor noise and sensor fault initiated at 25 seconds. . . . .	42
6.2	Sensor noise and sensor fault initiated at 25 seconds. . . . .	42
6.3	No sensor noise and sensor fault initiated at 55 seconds. . . . .	43
6.4	Sensor noise and sensor fault initiated at 55 seconds. . . . .	43
6.5	No sensor noise and sensor fault initiated at 85 seconds. . . . .	43
6.6	Sensor noise and sensor fault initiated at 85 seconds. . . . .	43
6.7	No sensor noise and sensor fault initiated at 115 seconds. . . . .	44
6.8	Sensor noise and sensor fault initiated at 115 seconds. . . . .	44
6.9	No sensor noise and sensor fault initiated at 145 seconds. . . . .	44
6.10	Sensor noise and sensor fault initiated at 145 seconds. . . . .	44
6.11	No sensor noise and sensor fault initiated at 175 seconds. . . . .	45
6.12	Sensor noise and sensor fault initiated at 175 seconds. . . . .	45
6.13	No sensor noise and sensor fault initiated at 205 seconds. . . . .	45
6.14	Sensor noise and sensor fault initiated at 205 seconds. . . . .	45
6.15	No sensor noise and sensor fault initiated at 235 seconds. . . . .	46
6.16	Sensor noise and sensor fault initiated at 235 seconds. . . . .	46
6.17	No sensor noise and sensor fault initiated at 265 seconds. . . . .	46
6.18	Sensor noise and sensor fault initiated at 265 seconds. . . . .	46
6.19	No sensor noise and sensor fault initiated at 275 seconds. . . . .	47
6.20	Sensor noise and sensor fault initiated at 280 seconds. . . . .	47
6.21	No sensor noise in the presence of fuel injection valve process fault. . . . .	48
6.22	Fuel injection valve process fault in the presence of sensor noise. . . . .	48

6.23 No sensor noise in the presence of cylinder damage process fault. . . . .	48
6.24 Cylinder damage process fault in the presence of sensor noise. . . . .	48
6.25 No sensor noise in the presence of process fault regarding seal leakage inside the turbine. . . . .	49
6.26 Turbine seal leakage process fault in the presence of sensor noise. . . . .	49
6.27 No sensor noise in the presence of process fault regarding loss of lubrication of the turbine shaft. . . . .	49
6.28 Loss of lubrication of the turbine shaft process fault in the presence of sensor noise. . . . .	49
6.29 No sensor noise in the presence of process fault regarding seal leakage inside the compressor. . . . .	50
6.30 Compressor seal leakage process fault in the presence of sensor noise. . . . .	50
6.31 No sensor noise in the presence of process fault regarding loss of lubrication of the compressor shaft. . . . .	50
6.32 Loss of lubrication of the compressor shaft process fault in the presence of sensor noise. . . . .	50
6.33 No sensor noise in the presence of process fault regarding the blockage of the heat exchanger of the intercooler. . . . .	51
6.34 Blockage of the heat exchanger of the intercooler in the presence of sensor noise. . . . .	51
6.35 Detection time standard deviation of each monitoring module $\mathcal{M}^{(l,q)}$ . . . . .	58
A.1 Simulation results for the output of $\mathcal{M}^{(1,2)}$ for example 1. . . . .	63
A.2 Simulation results for the output of $\mathcal{M}^{(2,2)}$ for example 1. . . . .	63
A.3 Simulation results for the output of $\mathcal{M}^{(2,4)}$ for example 1. . . . .	64
A.4 Simulation results for the output of $\mathcal{M}^{(2,6)}$ for example 1. . . . .	64
A.5 Simulation results for the output of $\mathcal{M}^{(3,2)}$ for example 1. . . . .	64
A.6 Simulation results for the output of $\mathcal{M}^{(3,4)}$ for example 1. . . . .	65
A.7 Simulation results for the output of $\mathcal{M}^{(3,5)}$ for example 1. . . . .	65
A.8 Simulation results for the output of $\mathcal{M}^{(4,2)}$ for example 1. . . . .	65
A.9 Simulation results for the output of $\mathcal{M}^{(4,4)}$ for example 1. . . . .	66
A.10 Simulation results for the output of $\mathcal{M}^{(4,6)}$ for example 1. . . . .	66
A.11 Simulation results for the output of $\mathcal{M}^{(1,2)}$ for example 2. . . . .	66
A.12 Simulation results for the output of $\mathcal{M}^{(2,2)}$ for example 2. . . . .	67
A.13 Simulation results for the output of $\mathcal{M}^{(2,4)}$ for example 2. . . . .	67
A.14 Simulation results for the output of $\mathcal{M}^{(2,6)}$ for example 2. . . . .	67
A.15 Simulation results for the output of $\mathcal{M}^{(3,2)}$ for example 2. . . . .	68
A.16 Simulation results for the output of $\mathcal{M}^{(3,4)}$ for example 2. . . . .	68
A.17 Simulation results for the output of $\mathcal{M}^{(3,5)}$ for example 2. . . . .	68
A.18 Simulation results for the output of $\mathcal{M}^{(4,2)}$ for example 2. . . . .	69
A.19 Simulation results for the output of $\mathcal{M}^{(4,4)}$ for example 2. . . . .	69
A.20 Simulation results for the output of $\mathcal{M}^{(4,6)}$ for example 2. . . . .	69
A.21 Simulation results for the output of $\mathcal{M}^{(1,2)}$ for example 3. . . . .	70
A.22 Simulation results for the output of $\mathcal{M}^{(2,2)}$ for example 3. . . . .	70
A.23 Simulation results for the output of $\mathcal{M}^{(2,4)}$ for example 3. . . . .	70
A.24 Simulation results for the output of $\mathcal{M}^{(2,6)}$ for example 3. . . . .	71
A.25 Simulation results for the output of $\mathcal{M}^{(3,2)}$ for example 3. . . . .	71
A.26 Simulation results for the output of $\mathcal{M}^{(3,4)}$ for example 3. . . . .	71
A.27 Simulation results for the output of $\mathcal{M}^{(3,5)}$ for example 3. . . . .	72
A.28 Simulation results for the output of $\mathcal{M}^{(4,2)}$ for example 3. . . . .	72
A.29 Simulation results for the output of $\mathcal{M}^{(4,4)}$ for example 3. . . . .	72
A.30 Simulation results for the output of $\mathcal{M}^{(4,6)}$ for example 3. . . . .	73
A.31 Simulation results for the output of $\mathcal{M}^{(1,2)}$ for example 4. . . . .	73
A.32 Simulation results for the output of $\mathcal{M}^{(2,2)}$ for example 4. . . . .	73
A.33 Simulation results for the output of $\mathcal{M}^{(2,4)}$ for example 4. . . . .	74
A.34 Simulation results for the output of $\mathcal{M}^{(2,6)}$ for example 4. . . . .	74
A.35 Simulation results for the output of $\mathcal{M}^{(3,2)}$ for example 4. . . . .	74
A.36 Simulation results for the output of $\mathcal{M}^{(3,4)}$ for example 4. . . . .	75
A.37 Simulation results for the output of $\mathcal{M}^{(3,5)}$ for example 4. . . . .	75
A.38 Simulation results for the output of $\mathcal{M}^{(4,2)}$ for example 4. . . . .	75

---

A.39 Simulation results for the output of $\mathcal{M}^{(4,4)}$ for example 4. . . . .	76
A.40 Simulation results for the output of $\mathcal{M}^{(4,6)}$ for example 4. . . . .	76
B.1 The six-point Seiliger cycle [24]. . . . .	77
B.2 Schematic view of the geometry of cylinder, crank rod and crankshaft [24]. . . . .	78
B.3 Cylinder temperature vs engine speed. . . . .	79
B.4 Engine torque vs engine speed. . . . .	79
B.5 Combustion pressure at stage 6 vs engine speed. . . . .	79



# List of Tables

2.1	Marine propulsion process faults classification [45] [42] [67] [20] [73] [72] [31] [1] [19] [11] [52] [36] [18]. . . . .	6
2.2	Modeled root causes. . . . .	8
3.1	Affected state and algebraic variables from the process faults. . . . .	28
3.2	Characterization and modeling of permanent faults [57] [34]. . . . .	28
4.1	Fuel engine parameters [24]. . . . .	31
6.1	Design parameters of observers in monitoring modules $\mathcal{M}^{(1,1)}$ , $\mathcal{M}^{(1,2)}$ , $\mathcal{M}^{(3,1)}$ , $\mathcal{M}^{(3,2)}$ , $\mathcal{M}^{(4,1)}$ and $\mathcal{M}^{(4,2)}$ . . . . .	41
6.2	Sensor sets and index sets used in the local monitoring agents. . . . .	42
6.3	Sensors with their noise range and fault magnitudes used. . . . .	47
6.4	Design parameters that were used for the modeling of the process faults. . . . .	51
6.5	Process and sensor fault signature matrix of $\mathcal{M}^{(1)}$ (0,1). . . . .	52
6.6	Part of process and sensor fault signature matrix of $\mathcal{M}^{(2)}$ (*,0,1). . . . .	52
6.7	Part of process and sensor fault signature matrix of $\mathcal{M}^{(3)}$ (*,0,1). . . . .	53
6.8	Part of process and sensor fault signature matrix of $\mathcal{M}^{(4)}$ (*,0,1). . . . .	53
6.9	Part of the global process and sensor fault signature matrix (*,0,1). . . . .	54
6.10	Lowest fault magnitude values for the avoidance of miss-detection. . . . .	59
B.1	Seiliger cycle thermodynamic relations and ratios [62]. . . . .	78



# Introduction

In the next decade, waterborne trade demand is estimated to increase by 26%, while at the same time a significant shift towards decarbonization and digitalization will be made [26]. Marine vessels are essential for mapping oceans, conducting offshore operations, and shipping activities. Moreover, they are a prerequisite for humans to be able to harvest ocean-based resources [27]. As a result, there is a great motivation for classification societies and government associations worldwide, to monitor and maintain the active fleet. This can be done by adopting innovative and systematic fault diagnosis methodologies, that detect and isolate faults that occur during the operation of a marine vessel.

## 1.1. Prologue

**Maintenance** is defined as “a combination of all the technical and associated administrative activities required to keep equipment, installations, and other physical assets in the desired operating condition or to restore them to this condition” [55]. At first, maintenance was seen as a necessary evil. However, today it attributes to the operational functionality and life-cycle of different systems, like ships where it can limit their downtime to a minimum while reducing up to 30% of their operational expenses [46]. The traditional maintenance techniques applied on ships represented by corrective maintenance and preventive maintenance have moderately highlighted the disadvantages of “over repair” and “missing repair” that inevitably lead to the growth of a ship’s operation costs [74]. Conducting maintenance based on predetermined periods or running hours can frequently result in redundant maintenance and may cause wear-out, which in turn, leads to reduced equipment reliability and unwanted downtime [25]. For maritime safety purposes, implementations of new approaches based on the availability of large arrays of sensors are happening, improving the monitoring and performance of marine vessels, their machinery, structural elements, propulsion, and power systems. Such sensor data are produced in an unprecedented manner, on global ship fleets, with increased broadband connectivity to shore, and can contribute to the overall monitoring efforts [5].

## 1.2. Definitions and Motivation

The growing complexity of shipboard systems has increased the expectations and demands for ship efficiency. The impact of the data available on vessel operations favors a well-structured **Condition Based Maintenance** (CBM) regime. CBM is defined as “the maintenance policy carried out in response to a significant deterioration in a machine as indicated by a change in a monitored parameter of the machine condition” [46]. CBM may be often called **Prognostic Health Maintenance** (PHM) due to its ability to acknowledge the condition of equipment, plan and carry out maintenance decisions before a critical failure [28]. Indeed, recent studies agree that PHM is a positive alternative to traditional CBM and has therefore attained interest in both academia and the maritime world, even though PHM techniques require more accurate and robust data-driven algorithms than systems to date have utilized [14].

The term **monitoring** was used at first by the U.S. military to describe a type of aerial espionage that went over and beyond regular photography use and into the use of other parts of the electromagnetic spectrum besides the visible ones, such as the infrared and microwave spectrum [59]. The essence of CBM is **Condition Monitoring** (CM) which can be defined as “the process of monitoring a condition parameter, in order to identify a significant change that is indicative of a developing fault” [2]. Two main pillars hold PHM. The first is the prognosis, and the second is the diagnosis [28]. The outcome of an accurate PHM method provides the technique to maintain a component’s health condition attributes in making Operation and Maintenance (O&M) decisions. The PHM techniques for marine vessels and their equipment cover, in essence, every vital function carried out on a marine vessel, as shown below in fig. 1.1, by acquiring Data, Information, and Knowledge (DIAK) and utilizing them to reproduce results for O&M decision making [74]. The DIAK acquisition phase is achieved at the beginning of the monitoring process, where the online data acquired by sensors, can be stored and processed to reproduce several visualizations and plots that make DIAK transparent and easily accessible for the next phases of PHM. The results of CM attribute an early warning for the equipment of interest. According to possible abnormal behavior in the acquired data of CM, **Fault Diagnosis** (FD) will focus on the faults caused or possible future failure modes and their underlying reasons by demonstrating the relationship between the data and the fault condition [74]. The primary purpose of FD is to study methods that identify and characterize possible commencing faults in different parts of the system under study [10]. Afterward, the task is to establish a possible degradation model to estimate the remaining useful life using health prognosis, as can be seen at the bottom of fig. 1.1. This estimation focuses on the possible evolution of the fault [74]. Such faults may affect either the process on-board a vessel or the sensors monitoring it, or even both at the same time. A reliable FD method should be able to capture and distinguish all fault types, so that O&M decision-making may be conducted, to avoid equipment failure on-board. Therefore, this work focuses mostly on the FD block of fig. 1.1 and the way to develop a reliable and holistic FD method that will detect sensor and process faults of a marine propulsion process and isolate them.

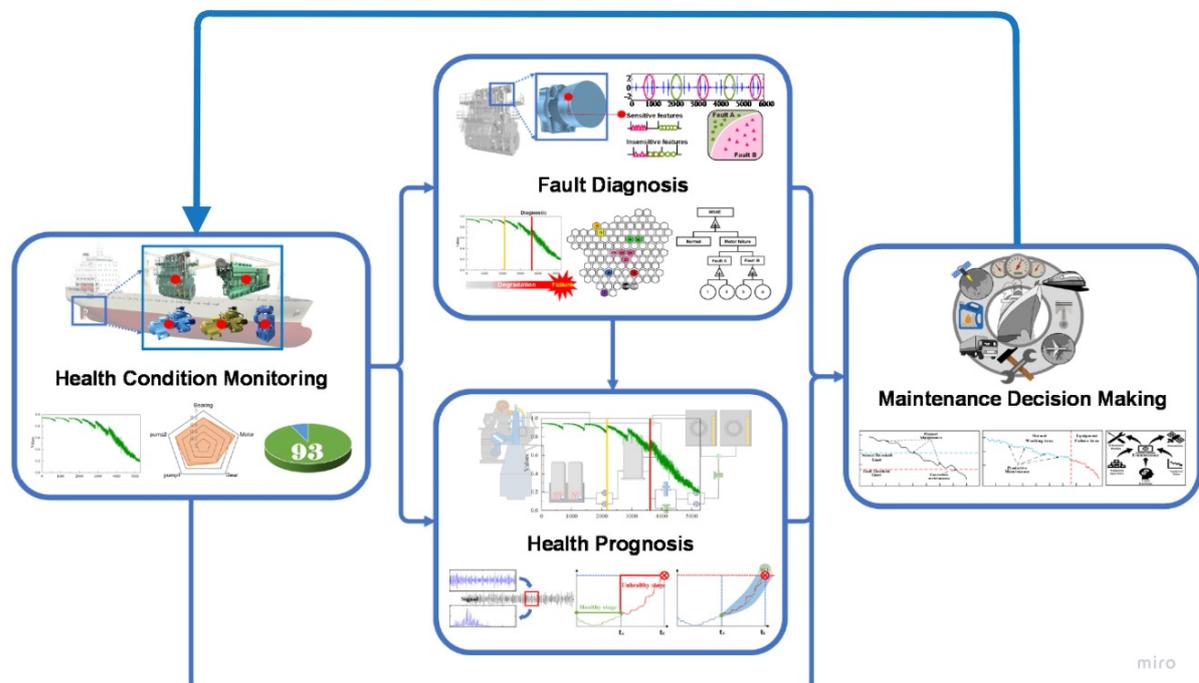


Figure 1.1: Relationship diagram between CM, FD and health prognosis [74].

### 1.2.1. Motivation

Problems that arise in the absence of FD on-board marine vessels, place the crew and passengers in danger and compromise the timely completion of a voyage. Therefore, the design of a model-based, distributed Fault Detection and Isolation (FDI) methodology, of multiple process and sensor faults, may allow the avoidance of implications and reassure the safety on-board. Additionally, the Mean Value First Principle (MVFP) model of a fuel engine part of the marine propulsion process, enables a better generalization ability of the results for different engines, since it allows for the reconfiguration of the used parameters and can be expanded to include more subsystems. This work focuses on systems described by nonlinear Differential Algebraic Equations (DAE). Due to the high complexity and the non-linear interconnected systems of the marine fuel engine, a distributed monitoring approach is proposed in this work [44]. More specifically, local monitoring agents are designed, that utilize information from local sensor sets and the sensors of neighboring agents to capture the occurrence of the process and sensor faults.

With regards to previous work and more details associated with FD methods, an extensive literature review is provided by the author in [53]. The reader can look into the aforementioned review and trace for information concerning marine vehicles and their functions, the sensors used for their monitoring, the state-of-the-art regarding maintenance and CM, as well as the model-based and model-free FD methods. At the end of the review, there is also a clear motivation that links it to this thesis, which comes to narrowing down the gaps in the field of FD for marine vessels.

## 1.3. Development and Contribution

The development of the FD framework shall be carried out by first presenting the models that describe the systems of the propulsion process of marine vessels which will be monitored in the simulation. Moreover, an FDI scheme shall be provided, with the logic of how to detect a fault and distinguish it between either a sensor or a process fault as well as how to isolate it. A further purpose of this work is to answer the research questions that will be mentioned hereafter.

### 1.3.1. Research Questions

This paper will answer a series of research questions regarding FD of a marine vessel's propulsion process. Such questions are the following:

1. What are the common faults regarding marine propulsion systems of large vessels, and what are the needed tools to carry out their modeling, detection, and isolation?
2. How to implement the physical models that describe the process of a marine propulsion system in a simulated environment and how to monitor them with simulated sensors?
3. How to validate the simulated models that describe the process of a marine propulsion system of a large vessel?
4. What would be a feasible model-based scheme to detect and isolate sensor and process faults for the integrated diagnosis of faults affecting marine propulsion systems?
5. How can the provided FD strategy be implemented in a simulated environment and what are the results and Key Performance Indicators (KPIs)?

### 1.3.2. Scope

The scope of this thesis evolves around developing a distributed FD integrated framework that will detect and isolate multiple sensor and process faults that affect the propulsion process of a marine vessel. Therefore, this work focuses on permanent, abrupt, and offset sensor faults that affect the sensors, as well as permanent, abrupt, or incipient, offset, or drift-like process faults that affect the propulsion process. To these means, the models that describe a turbocharged marine diesel engine, which represents a large part of the marine propulsion process, are simulated as well as the sensors to monitor it, and the aforementioned faults. Moreover, the introduction of monitoring agents and adaptive thresholds will be made that allow the detection and isolation of multiple faults, reduce conservativeness in decision-making, and eliminate false alarms.

### 1.3.3. Contribution

The main contribution of this thesis is a proposed framework as described earlier. This will be done by relying to a large extent on the work of [24] for the modeling of the marine propulsion process of a vessel but also on the work of [44] that conducted similar work for the model-based, distributed FDI of sensor faults on a marine fuel engine. This thesis will go one step further and introduce the process fault models in the marine propulsion process, but also redundant sensors to provide the distinction between the process and sensor faults. Therefore, a big part of the methodology for the FDI of multiple sensor faults is taken from [44] and is expanded to cover process faults, by also considering the theory needed for the modeling of such faults.

In an attempt to identify frequent faults in the marine propulsion process, this thesis reviews and mentions the faults that might appear in the systems that are used for the propulsion of vessels. Moreover, this work introduces the concept of utilizing redundant sensors and monitoring modules that enable the distinction of the sensor faults from the process faults. Additionally, these redundant sensors and modules provide better isolability of the detected faults, since their existence allows the formation of an increased number of unique signatures for the possible faults. The methodology that accommodates the redundant sensors and modules in order to create augmented fault signature matrices that enhance fault isolability is expressed extensively in this work. Furthermore, the modeling of process faults attributes to the perception of how abnormalities in system behavior may appear and how they may be detected and isolated. The modeling of such faults is not a trivial task and was to a large extent conducted based on the theory found in [34] and [57], but also on the author's knowledge about the behavior of similar mechanical systems. Lastly, a sensitivity analysis is conducted to showcase the performance regarding fault detection by the monitoring modules.

### 1.3.4. Thesis Outline

This thesis is structured as follows. In chapter 2 the answer to research question 1 will be given since the mechanical faults in the marine propulsion process and the root causes that lead to them are presented. In addition, an extensive literature review regarding the State-of-the-art of monitoring and diagnosis of marine vessels is provided. Furthermore, the relevant background is covered in chapter 2 regarding fault tolerance and redundancy, multi-agent monitoring, observer-based monitoring of complex systems such as marine vessels, and multiple fault isolation. In chapter 3 the answer to research question 2 is given since the marine propulsion process model is presented and explained, along with the various subsystems that compose it. These models are validated in chapter 4, where they are compared with some preexisting measurement data and research question 3 is answered. In chapter 5, the FD methodology in the case of multiple combined sensor and process faults is described, where the problem formulation is explained at first, and the distributed FDI is provided later in order to answer research question 4. Additionally, the concept of using redundant sensors and monitoring modules is presented in chapter 5. In chapter 6, simulation results are shown and the applicability and performance of the proposed methodology, as well as better isolability of the faults detected, are discussed therefore research question 5 is answered there. Finally, in chapter 7 some conclusions regarding whether the initial aim of the thesis was achieved and if the research questions were answered overall, and the room for improvement of this work by future research shall be discussed extensively.

# 2

## Literature Review

In this chapter, the first research question shall be answered, and some background knowledge regarding model-based FD will be provided. First, the various process faults which lead to mechanical failures that need to be avoided on marine vessels will be described. Then, the state-of-the-art of CM and FD for marine vessels will be given. Finally, model-based FD background knowledge will be discussed in an effort to prepare the ground for the chapters that will follow. At the end of this chapter, similar works are reviewed.

### 2.1. Marine Propulsion Process Faults Overview

*Seaborne trade* today is highly dependent on cargo shipping, as it reduces the cost of transport per tonnage more than any other transportation mode. It represents the value of freight being transferred across the distance of two seaports, and the advancements in modern shipping play a vital role in the further expansion of the sector. The introduction of the internal combustion engine led to a drastic shift in naval design and propulsion-related aspects but has also caused some new possible propulsion process implications due to the increased complexity of such engines, which have an increased number of moving parts [3]. Marine vessels have many different critical components and equipment that are essential, and their abnormal behavior or failure could lead to unwanted downtime of the whole vessel. Most of these components and equipment fall under the category marine engines and drive train systems, like Gas Turbine Engines (GTE), Combined Diesel and Gas (CODLAG), or diesel engines and their gearboxes or bearings and shafts that interconnect drive train systems. Other auxiliary systems that support the drive train and power systems like turbochargers, boilers, condensers, oil pumps, generator systems, and Direct Current (DC) machines are also under close watch by monitoring and FD methods. Furthermore, navigation and communication systems, like station-keeping devices of idle ships and weather observation devices on-board, are monitored with the help of FD methods. Finally, propulsion and steering systems, as well as hull structure monitoring, are also of fundamental importance to marine vessels since these systems are directly linked with opposing drag produced by large marine vessels through the water, and well-maintained propellers, rudders, and hulls are necessary.

#### **Propeller Faults**

Propeller faults will inevitably affect the operational efficiency and stability of the ship [29] and will cause severe propulsion problems. Propellers accumulate damage and physical surface roughness created by corrosion, cavitation, erosion, and impingement attack [43]. Meanwhile, the growth of bio-fouling or marine pollutants may also cause mechanical faults in blades [72] and blade surface roughness deterioration. In absolute terms, propeller roughness is less significant than hull roughness, but in terms of energy loss per unit area, propeller roughness is predominant [43]. Therefore, propellers must maintain a low roughness and good surface condition, monitored throughout their useful lifetime.

#### **Gearbox Faults**

Even if the gearbox is enclosed in the vessel's hull, harsh weather conditions when traveling through the ocean can lead to overload and strain of the gearbox. Therefore, gearbox condition must be monitored and maintained with PHM methods, the crew on-board must be trained to maintain it, and the gearbox status data must be frequently collected [61].

### Lube Oil Pump and Hydraulic Faults

Oil pumps are considered one of the most critical elements for seagoing vessels since only one oil pump unit is allocated on every ship. Hydraulic faults can consequently increase accidents or collisions due to the difficulty of a ship to achieve steering in its absence. In addition, once the lube oil pump fails, direct effects on the ship's operation will be witnessed [29].

### Diesel Engine Faults

Marine diesel engines are of paramount importance for seagoing vessels. The regular operation of the marine diesel engine is essential for the safety of voyages. However, traveling during harsh weather, marine diesel engines are overloaded beyond their limit, which sometimes leads to terrible marine accidents. Thus, it is imperative to monitor them and discover developing faults that could cause future failure [47]. By doing so, the scheduled completion of a voyage can be reassured.

Table 2.1: Marine propulsion process faults classification [45] [42] [67] [20] [73] [72] [31] [1] [19] [11] [52] [36] [18].

Propulsion Component	Criticality	Severity	Frequency	Possible Root Causes
<i>Lube Oil Pump</i> [1]	<i>High</i>	<i>High</i>	<i>Low</i>	<i>Damaged filter cartridge, flowmeter malfunction, pump driver malfunction, pump damaged coupling, damaged piping, human error.</i>
<i>Steering System</i> [31]	<i>High</i>	<i>High</i>	<i>Low</i>	<i>Steering gear fault, damaged bearings, hydraulic system malfunction, oil pump malfunction, electromagnetic directional valve malfunction, damaged rudder, remote control system failure, damaged combination valve.</i>
<i>Propeller</i> [72]	<i>High</i>	<i>High</i>	<i>Low</i>	<i>Marine pollutants and biofouling, corrosion and pitting, current flow velocity, cavitation, fatigue, damaged shaft.</i>
<i>Gearbox and Shaft</i> [67] [20] [73] [52]	<i>High</i>	<i>Moderate</i>	<i>Low</i>	<i>Corrosion of surfaces, fatigue, thermal stress, geometrical stress concentration, vibrations, shaft misalignment, overloading, internal material imperfections, damaged gears.</i>
<i>Cylinder Unit</i> [45]	<i>Moderate</i>	<i>Moderate</i>	<i>Moderate</i>	<i>Fuel injection fault system, fuel pipe leak, fuel pump fault, exhaust valve remain open, exhaust valve blocked, air inlet valve blocked.</i>

Propulsion Component	Criticality	Severity	Frequency	Possible Root Causes
<i>Turbine</i> [42][45] [18]	<i>Moderate</i>	<i>Moderate</i>	<i>Moderate</i>	<i>Labyrinth seal leakage, surging of turbo blockage, loss of lubrication, damaged bearings, exhaust duct contaminated, damaged nozzle rings, turbine blades fouled/damaged, turbo shaft damaged seal.</i>
<i>Intercooler</i> [42][45]	<i>Moderate</i>	<i>Moderate</i>	<i>Moderate</i>	<i>Compressor impeller fouled, air side impeller damaged, air side seal loss, damaged bearings, air filter fouled, cooler tube/air blockage, loss of lubrication.</i>
<i>Crankshaft</i> [19]	<i>Moderate</i>	<i>Moderate</i>	<i>Moderate</i>	<i>Bending fatigue, overloading, torsional fatigue, crankshaft misalignment, resonant vibrations, geometrical stress concentration, damaged bearings, insufficient lubrication.</i>
<i>Fuel Pump</i> [11] [36]	<i>Moderate</i>	<i>Moderate</i>	<i>Moderate</i>	<i>Early or delayed valve opening, damaged gears, insufficient lubrication, damaged bearings, high oil viscosity, inadequate pressure, valve seizing.</i>
<i>Diesel Engine</i> [45]	<i>High</i>	<i>High</i>	<i>Moderate</i>	<i>Turbocharger fault, cylinder damage, crankshaft fault, bedplate damage, camshaft fault, air cooler fault, exhaust gas receiver problem, chain drive damaged, blower fault.</i>

After an extensive study of the available literature regarding the faults occurring on marine propulsion components, table 2.1 was formed. In the table, the propulsion components are rated according to the criticality, severity, and frequency of an occurring fault. Criticality is high if only one component is allocated per vessel, while severity is high if the occurrence of a fault on a particular component causes severe issues that render the ship out of operation. Finally, the existence of a column representing the frequency, emphasizes how often a fault might occur. From table 2.1, one may notice that the most critical components in a marine propulsion system are the propeller, the lube oil pump unit, the steering system, the gearbox and shaft of the vessel, and the diesel engine. Each of these components has one instance onboard a vessel. Therefore, their maintenance is crucial. The components with high severity that are given in table 2.1 involve the oil pump, steering system, propeller, and diesel engine, while the rest of the propulsion process components are described by moderate severity. Most components show a moderate frequency of fault occurrence. This comes to show that the maintenance efforts attributed to CBM are sufficient and unwanted incidents during times of travel that place the crew, passengers, and cargo in danger are scarce. Hence, the remainder of the effort of early warning regarding the faults that have occurred can be covered by FD methods.

The root causes presented in table 2.1 with moderate frequency should be included in the propulsion process model in order to capture their contribution to the process. The frequency was determined after considering how often a fault might appear in a particular propulsion process, based on the reviewed sources mentioned in table 2.1. The sense for considering the faults with moderate frequency is that

Table 2.2: Modeled root causes.

Propulsion Process System	Modeled Root Causes	Units	Possible Effects
<i>Turbine</i>	<i>Turbo side seal loss</i>	<i>Pa</i>	Pressure drop inside the turbine.
	<i>Loss of lubrication turbo side</i>	<i>K</i>	Increase of the slip air temperature inside the turbine.
<i>Intercooler</i>	<i>Air side seal loss</i>	<i>Pa</i>	Pressure drop inside the compressor.
	<i>Cooler tube blockage</i>	<i>K</i>	Temperature increase of the air exiting the intercooler.
	<i>Loss of lubrication air side</i>	<i>K</i>	Temperature increase of the air inside the compressor.
<i>Cylinder Unit</i>	<i>Bore damaged</i>	<i>m<sup>3</sup></i>	Increase of the cylinder volume due to wear of its walls.
<i>Fuel Pump</i>	<i>Valve seizing</i>	<i>%</i>	Choking of the fuel injection.

they occur more often than the ones with low frequency, and the framework should give priority to detecting these faults. Therefore, root causes affecting the marine diesel engine and its subsystems were considered in this work. In many cases, root causes like the unpredictable sea currents that act on the propeller or the growth of marine and other pollutants are not straightforward to implement in the propulsion process model. Nevertheless, some root causes were modeled, as shown in table 2.2. The root causes that were modeled as process faults, were chosen in an attempt to introduce at least one process fault in every subsystem and one affecting each variable if possible, concerning the marine fuel engine and its subsystems. In this table one may view the modeled root cause, the system it affects, its unit of measurement, and the effects that each fault is modeled to impose on the systems. Details regarding these process faults shall be provided in chapter 3 along with the propulsion models.

## 2.2. State of the Art of Monitoring and Fault Diagnosis

As was mentioned in the introduction, PHM is dependent on CM to a large extent. Analysis of CM data can be developed, in an attempt to optimize maintenance and avoid failures or breakdowns [48]. Hereafter, the State-of-the-art of CM and FD for marine vehicles will be presented.

### 2.2.1. Condition Monitoring

Nowadays, CM technologies carry numerous recording and observation tools, that evaluate data and quantitative parameters. Such monitored data may be vibration, acoustic, optical, temperature, mechanical strain, electric current signals, and pressure measurements [46]. To date, efforts to establish effective maintenance operations have demanded much effort due to reduced data transparency during the component's usage period. Nevertheless, with the emerging technologies offered today, such as Radio Frequency Identification (RFID), an increased number of sensors, Micro-Electro-Mechanical Systems (MEMS), wireless networks, Supervisory Control and Data Acquisition (SCADA), and Product Embedded Information Devices (PEID), the maintenance and monitoring efforts of systems during their usage period, are expected to become more accessible and more redundant than ever before [55]. CM has the unique advantage, of addressing conditions that reduce the expected lifespan of equipment before they deteriorate into major failures. Some of the routine CM checks carried out on marine vessels include the usage of electromechanical equipment for engine performance measurements, the placing of temperature sensors in cylinder liners to monitor piston rings, blow-by gas temperature, and visual inspection of piston rings and liners [2]. Furthermore, the Italian navy adopted a Hull Monitoring System (HMS) on the new FREMM frigates to monitor the ship hull or auxiliary structures. The HMS monitors and logs in data such as hull rigid body motions, bending moment along the hull girder, local strains in structural elements, fatigue cycles experienced by the vessel's structural elements, pressures acting on the hull, sea conditions acting on the vessel and operative conditions of the ships [12].



Figure 2.1: FREMM frigates in open sea [12].

The utilization of wireless technology for CM has benefited industries such as electricity production as it reduced the cost of replacement due to failure and increased versatility. The industrial standards adopted for maritime monitoring applications operate in a different frequency range than the one of the typical Wi-Fi in order to overcome electromagnetic noise and reflections from metallic surfaces produced by machinery and equipment on-board and also provide a better signal range within areas overwhelmed by electromagnetic noise like engine rooms [48]. A CM system should be able to monitor the running machines and equipment during the existence of electromagnetic interference [32]. The benefit of adapting wireless CM networks, apart from cost reduction, is the ease of implementation since such a wireless network could be adopted on older vessels where CM capabilities do not exist, and the need for CM could be higher compared to the one on modern vessels [48].

### 2.2.2. Fault Diagnosis

There are two brought method categories for FD, named model-free methods and model-based ones. The former detects faults by comparing the results of measurements with ML predictions of mathematical simulation algorithms or Artificial Intelligence(AI)-based. For most model-based methods, frequency and time-domain signal processing technologies will be used to obtain “signatures” or residuals which can indicate an expected or faulty behavior [32]. FD evolves the detection of faults in a detailed manner, providing information such as the extent, location, and time of detected faults [35]. Fault detection provides a comparison of sensor data with the estimated operational performance, that is, the expected values of system parameters, to detect and report irregular operating conditions [14]. It exhibits a critical role in the demonstration of the association between the CM information, and the health condition [74]. Moreover, it is usually followed by Fault Tolerant Control (FTC) and, more specifically, by the creation of a reconfiguration unit which, based on the type of the faults detected, performs control reconfiguration in an attempt to achieve desired performances and restitution of the faulty system that is within the fault tolerance limits. Many strategies are adopted to perform the system reconfiguration, i.e. mechanical reconfiguration such as switching between redundant hardware or mechanical parameters variation can be utilized to avoid faults, and their consequences [10]. The detected abnormal behavior needs to be post-processed to develop a prescription as a guidance report to CBM. In the past, it was performed by experts or offline analysts but now tends to be implemented online and automatically by computer combined with ML and AI. The prescriptions provided to the users are accompanied by the name, time, and location of each defect, the status of the machine, and advice for maintenance [32].

## 2.3. Fault Modeling and Fault Examples

According to [34], “a fault is defined as an unpermitted deviation of at least one characteristic property of the system from the acceptable, usual standard condition.” As can be seen in fig. 2.2, the time dependency of faults can be classified as an abrupt fault (stepwise), incipient fault (drift-like), or intermittent fault. Concerning the process models, faults can be further distinguished, and this can be seen in fig. 2.3, where additive faults influence a variable  $Y$  by an addition of the fault  $f$ , and multiplicative faults by the product of another variable  $U$  with  $f$ . Additive faults appear, e.g., as offsets of sensors, whereas multiplicative faults are parameter changes within a process.

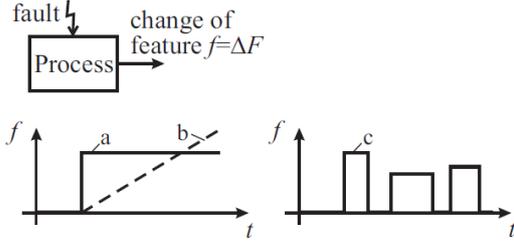


Figure 2.2: Time-dependency of faults: a) abrupt; b) incipient; c) intermittent [34].

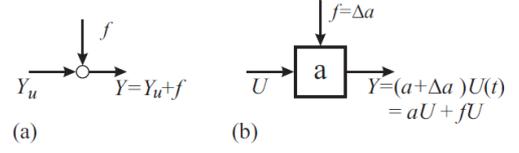


Figure 2.3: Basic models of faults: a) additive fault; b) multiplicative fault [34].

For the example of the additive fault the detectable change  $\Delta Y_a(t)$  of the variable is independent of any other signal as shown in (2.1), while in the example of the multiplicative fault, the detectable change of the output  $\Delta Y_m(t)$  is depended on the input signal  $U(t)$  as shown in (2.2). The size of the change  $\Delta Y_m(t)$  is also influenced on the size of  $U(t)$  [35].

$$\Delta Y_a(t) = f(t) \quad (2.1)$$

$$\Delta Y_m(t) = f(t)U(t) \quad (2.2)$$

### 2.3.1. Fault Modeling

According to [35], an intermittent fault  $f \in R$  can be modeled based on the time duration, in the following manner:

$$f(t) = \sum_{v=1}^V \beta_v \{t_{2v-1}, t_{2v}\} \phi_v \{t - t_{2v-1}\} \quad (2.3)$$

Where  $\beta_v \in R$  is the time profile and  $\phi_v \in R$  is the fault function. According to [57], the evolution mode  $\beta_v$ , can be modeled as follows:

$$\beta_v(t) = \begin{cases} 0, & t < 0 \\ 1 - e^{-\kappa_v t}, & t \geq 0 \end{cases} \quad (2.4)$$

Where  $\kappa_v > 0$  is the evolution rate. When  $\kappa_v \rightarrow \infty$ , the occurrence of a fault is characterized as abrupt. In equation (2.3), the term  $V$  expresses the number of time intervals of the fault presence. Special cases of intermittent faults are the transient ( $V = 1$ ) and permanent ( $V = 1, t_2 \rightarrow \infty$ ) faults. Particularly, transient faults can be modeled as follows [57]:

$$f(t) = \beta(t_1, t_2) \phi(t - t_1) \quad (2.5)$$

With the permanent fault described as:

$$f(t) = \beta(t_1, \infty) \phi(t - t_1) \quad (2.6)$$

### 2.3.2. Fault Examples

The time profiles and the time-varying fault function models that were shown shall be used to generate a fault profile for the modeled root causes of table 2.2. Some cases of fault functions like the ones presented in fig. 2.4 are the following [57]:

1. The offset faults, i.e.  $\phi_v(t) = \phi_v^0$ , where  $\phi_v^0$  expresses a constant magnitude.
2. The drift faults, i.e.  $\phi_v(t) = R_v t$ , where  $R_v$  is the constant slope.
3. Precision degradation, whose fault function  $\phi_v(t)$  is a random variate from a given distribution.

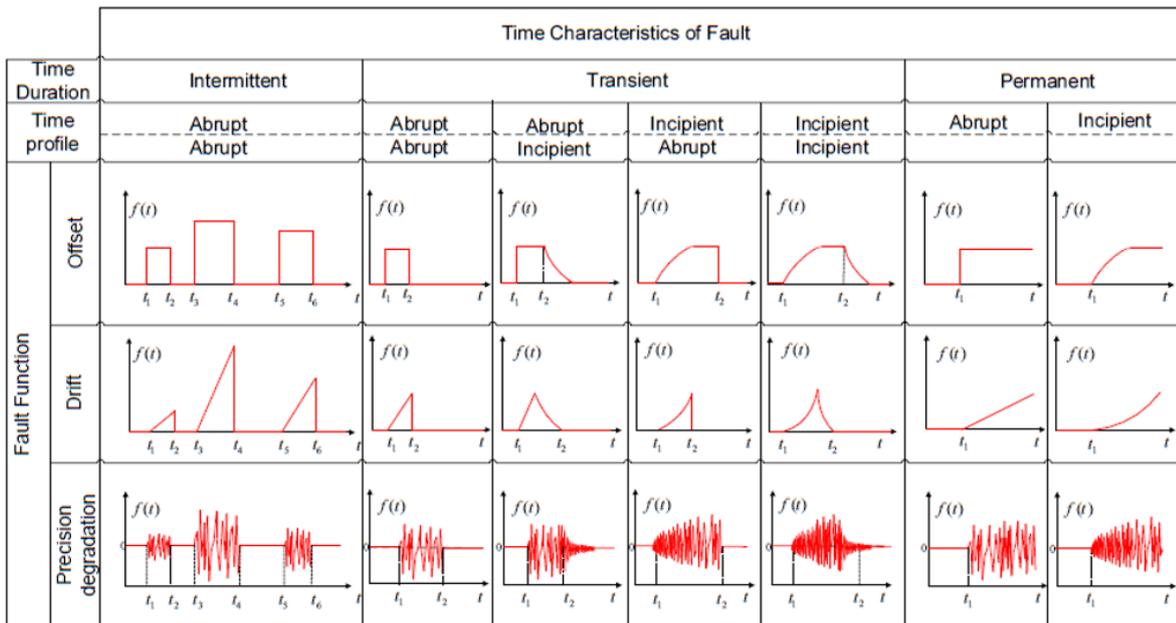


Figure 2.4: Examples of fault models with various time duration (intermittent, transient, permanent), time profiles (incipient, abrupt), and fault functions (offset, drift, precision degradation). In the case of the intermittent faults, only the abrupt evolution mode of occurrence and disappearance is presented [57].

## 2.4. Fault Tolerance and Redundancy

Fault tolerance is required for high-integrity systems. Faults are resolved so that they do not lead to system failures. After applying practices that attribute toward the impeccability of the components, the remaining effort to achieve this target is to introduce redundancy [35]. This logic can be viewed in fig. 2.5 below, where for every considered module, more reserve modules exist, in a parallel configuration, introducing redundancy to the system.

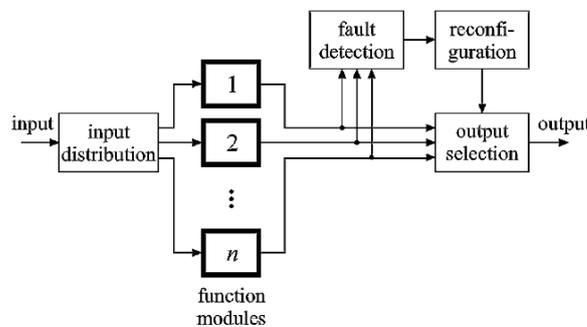


Figure 2.5: Basic scheme of a fault-tolerant system with parallel function modules as redundancy [35].

The modules that are shown in fig. 2.5 can be hardware or software components. For FD purposes, such function modules or sensors may also be installed on-board marine vehicles [35]. Different arrangements of fault-tolerant sensor fusion configurations exist providing redundancy to the data acquisition process. In general, the function modules presented by [6] are supervised with fault-detection capability for a navigation system of a surface vessel. When faults occur, faulty sensing devices are autonomously isolated, and faulty data is discarded.

### 2.4.1. Physical Redundancy

Physical redundancy is based solemnly on sensor data from sensors that measure a physical quantity. The first step involves the creation of quantifiable residuals from system measurements by comparing the outputs of two or more sensors responsible for the same physical quantity [71]. The residuals

are then checked against a threshold  $thr$ , which is presumed to be the same for positive and negative deflections and that the violation of the maximal or minimal threshold is triggered for each measurement  $m$  [35]:

$$res_i^* = \begin{cases} 0 & \text{if } |res_i(t)| \leq thr_i \\ 1 & \text{if } |res_i(t)| > thr_i \end{cases} \quad \forall i \in \{1, m\} \quad (2.7)$$

Where  $res_i^* = 1$  indicates that one of the thresholds was violated, the residual results can be filled in a fault signature matrix where faults can be isolated using distinguishable patterns for each residual vector. Two definitions need to be mentioned when dealing with redundancy relations [6]:

1. *Structural detectability*: A fault is structurally detectable  $c_j \in C_{det}$  if it has a nonzero boolean signature in the residual,  $C_{det} = \{c_j | \exists j : c_j \neq 0 \Rightarrow res_j \neq 0\}$ .
2. *Structural isolability*: A fault is structurally isolable if it has a unique signature in the residual vector, i.e. column  $m_i \in M$  is independent of all other columns in  $M$ ,  $c_j \in C_{iso}$  if  $\forall j \neq i : m_i \neq m_j$ .

Robustness refers to the property that residuals would not fire any false alarm as the result of unknown inputs acting on the system or as the result of uncertainties in the values of the system parameters. One means of designing robust residuals is the exact decoupling approach, in which the designed residuals are insensitive to unknown input and unknown or uncertain parameters. Faults which have different signatures are isolable from each other, while faults which share the same signature are non-isolable. The pattern of zero and non-zero residuals associated with a given fault is called its signature [7].

### 2.4.2. Analytical Redundancy

Analytical redundancy is deployed to diagnose faults and provide valid data, neglecting any faulty sensor and using sensor fusion to establish the best estimate for users. Analytical redundancy residuals are the deviation between measured values and model-based calculations [6]. The sensor measurements for one physical quantity are compared with a calculation based on the sensor measurements of another physical quantity.

When constraints determine the expected behavior of an entire system, any violation of one or more constraints would mean one or more faults exist in the system. Suppose the violation is isolated to a particular constraint. In that case, this will not inform on the physical reason behind the fault but will indicate that the specific device is no longer trustworthy [6].

### 2.4.3. Redundancy in Fuel Engine Fault Diagnosis

Marine engines can be modeled with nonlinear physical models that depend on thermodynamics. As a result, malfunctions and faults in marine engines have been studied longer than any other system on marine vessels. Such research can be viewed in the work of [68], where two different approaches are presented to detect increased blow-by and compression ratio failures. Similar work was conducted by [71], where according to them, FD in general, can be divided into two steps:

1. "Fault detection, during which it is determined whether a fault has occurred;
2. Fault isolation, during which it is determined what caused the fault."

By using residuals produced from system measurements according to physical redundancy relations, the detected faults can later be isolated based on thresholds, as was explained in subsection 2.4. Thus, a fault can be isolated if a residual value exists outside the bounded area predefined by the thresholds.

A residual is derived from the deviation between the actual and nominal measurement value, using the below calculation for each measurement  $m$  [71]:

$$res_j = \frac{act_j - nom_j}{nom_j} \quad \forall j \in \{1, m\} \quad (2.8)$$

At this point, a relevant discussion regarding measurement uncertainty and sensor accuracy can be made since the above equation strongly depends on the accuracy of the actual measurement value. As a matter of fact, in [70] it was emphasized that "measurement error which is so large that it exceeds

a tolerance range specified by a permitted error, regardless of its cause” like the residual formed in equation 2.8 can be considered as a fault.

Keeping in mind that every measurement consists of unavoidable uncertainty, which can affect residual generation, the actual measurement value will never agree precisely with the nominal value [71]. Therefore, the threshold values mentioned earlier are needed to compare each measurement's residual  $res_j$  as a fault term with its corresponding threshold  $thr_j$  for each measurement.

$$|res_j| \leq thr_j \quad \forall j \in \{1, m\} \quad (2.9)$$

While at the same time, as mentioned earlier, the threshold must include the measurement uncertainty of each sensor and the model inaccuracies caused during residual generation for each model [71], as can be seen below:

$$thr_j = u_j + u_{model,j} \quad \forall j \in \{1, m\} \quad (2.10)$$

In the works of [71], and [70], the physical behavior of engine test-beds has been depicted by models, for FD reasons, with the use of physical redundancy relations. Their work used quantitative, nonlinear models (static or dynamic) of test engines for fault detection. This was done using physical laws like mass balance, energy balance, and air ratios of the engine to derive the redundancy relations needed for the measurement residuals. With the use of thresholds as calculated in equation 2.10, the control region can be formed that the residuals compared to the thresholds must fall into. If for any reason a residual does not fall within the control region boundaries formed by the threshold, that will indicate a fault. Each fault must be isolated later, either as a sensor fault by comparing the measurement residuals of other sensors employed for the same task or as a system fault that all sensors responsible for that system can indicate. An example of a control region formed by thresholds in a residual space can be seen in fig. 2.6 below.

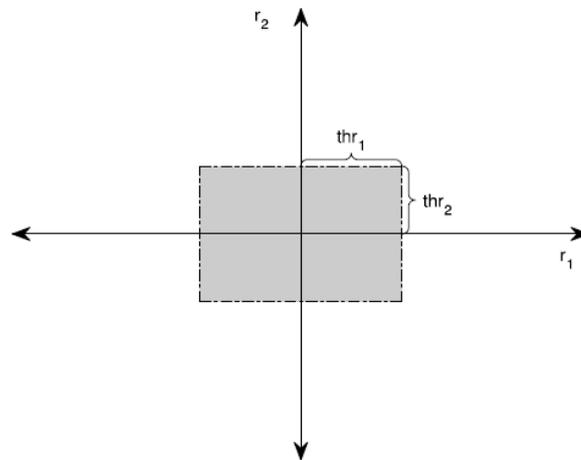


Figure 2.6: Residual space and control region in the two-dimensional case [70].

Specifically, in the work of [70], measurements were taken from a single-cylinder test engine concerning the rotational speed, torque, air mass flow, fuel mass flow, exhaust gasses concentration, intake pressure, intake and outlet temperatures, similar to the measurements that can be obtained on a vessels' marine fuel engine. These measurements were used to form control regions in residual spaces for each measurement type. Similarly, experiments conducted by [71] were later validated using statistical residual formation methods. In both works, the authors concluded that automatic calculation of thresholds enables highly sensitive fault diagnosis and robustness at the same time, detecting nearly 80% of faults when almost every detected fault can be isolated successfully.

## 2.5. Multi-Agent Monitoring

According to [51] when it is assumed that there is only one monitoring agent that has access to all network sensors and thus directly manages the physical network, this monitoring structure is referred to as a single-agent structure. This structure is ideal since, in principle, such a structure can present optimal performance, although this might increase computational costs. Suppose multiple monitoring agents exist in the structure, and some have authority over others, in the sense that they can impose authority or order other agents. In that case, this structure can be called multi-agent multi-layered. Such a structure is often present when one agent shall determine set points for other agents in groups.

In multi-layer FD, the communication in-between layers is commonly sporadic and event-driven, while the information transmitted to higher levels can be the decisions of the local agents, the time instances of fault detection of the local agents, or the calculated Analytical Redundancy Relations (ARRs) [56]. Multi-agent structures can be further discriminated as either decentralized or distributed, while the ideal single-agent structures can only be centralized. Today most CM techniques applied on marine vessels adopt the single-agent structure and a centralized monitoring architecture. These definitions can be seen hereafter [51]:

1. *Centralized monitoring architecture*: By definition, the centralized architecture places the authority of a sensor set to a single monitoring agent. This architecture can deliver the best performance achievable and it has been studied to an extensive degree in the literature. This performance comes at the expense of high computational time and effort.
2. *Decentralized monitoring architecture*: A decentralized architecture implies that one monitoring agent is dedicated to each interconnected subsystem and does not share information with other agents. An advantage of the decentralized multi-agent single-layer structures is that there is no communication between the agents, and therefore lower computational requirements are needed, at the expense of reduced performance.
3. *Distributed monitoring architecture*: An advantage of a distributed multi-agent structure is that increased performance can be obtained compared to a decentralized architecture, although at the price of increased computation time due to cooperation and communication.

### 2.5.1. Distributed Fault Diagnosis

For a long time, fault-tolerant distributed systems have been assessed in the software community to cope with hardware, software, and other fault types. Considering a system  $\Sigma$  equipped with a set  $I$  of  $m$  actuators, and a set  $J$  of  $p$  sensors, its behavior is described as follows [8]:

$$\dot{x}(t) = f(x(t), u(t), d(t), t) \quad (2.11)$$

$$y(t) = g(x(t), u(t), d(t), t) \quad (2.12)$$

Where  $x \in R^m$  is the control vector,  $y \in R^p$  is the measurement vector and  $d \in R^q$  is the disturbance vector. Assuming  $u_k$ ,  $y_k$  and  $x_k$ , where  $k = 1, \dots, s$ , to be partitions of  $u$ ,  $y$  and  $x$  into  $1 \leq s$  subvectors, and let:

$$\dot{x}_k(t) = f_k(x_k(t), \tilde{x}_k(t), u_k(t), \tilde{u}_k(t), d(t), t) \quad (2.13)$$

$$y_k = g_k(x(t), u(t), d(t), t) \quad (2.14)$$

( $k = 1, \dots, s$ ) be the occurring decomposition of equations (2.11) and (2.12), from the collection of the components of  $x$  except  $x_k$  by  $\tilde{x}_k$ . Each equation in (2.13), (2.14) can be explicated as expressing the behavior of a system  $\Sigma_k$  with  $u_k \in R^{m_k}$  the local control vector corresponding to a subset  $I_k$  of the actuators,  $y_k \in R^{p_k}$  the local measurement vector corresponding to a subset  $J_k$  of the sensors and  $x_k \in R^{n_k}$  the local state. In order to address distributed systems, the simple network architecture in which each subsystem  $\Sigma_k$  performs a part of the overall control and a part of the overall diagnosis is considered, as illustrated in fig. 2.7 [8].

The functions in the right part of (2.13) can take various forms, and a more detailed one is when  $f_k(x_k, \tilde{x}_k, u_k, \tilde{u}_k, d, t)$  is considered decomposable, namely, it becomes the summation of two functions, given as:

$$f_k(x_k, \tilde{x}_k, u_k, \tilde{u}_k, d, t) = f_k^{self}(x_k, u_k, d, t) + f_k^{coupled}(\tilde{x}_k, \tilde{u}_k, d, t) \quad (2.15)$$

Where  $f_k^{self}(x_k, u_k, d, t)$  describes the self-dynamics of subsystem  $\Sigma_k$  and  $f_k^{coupled}(\bar{x}_k, \bar{u}_k, d, t)$  describes the coupled dynamics with regard to the other systems denoting the influence of the other systems on system  $\Sigma_k$  [8].

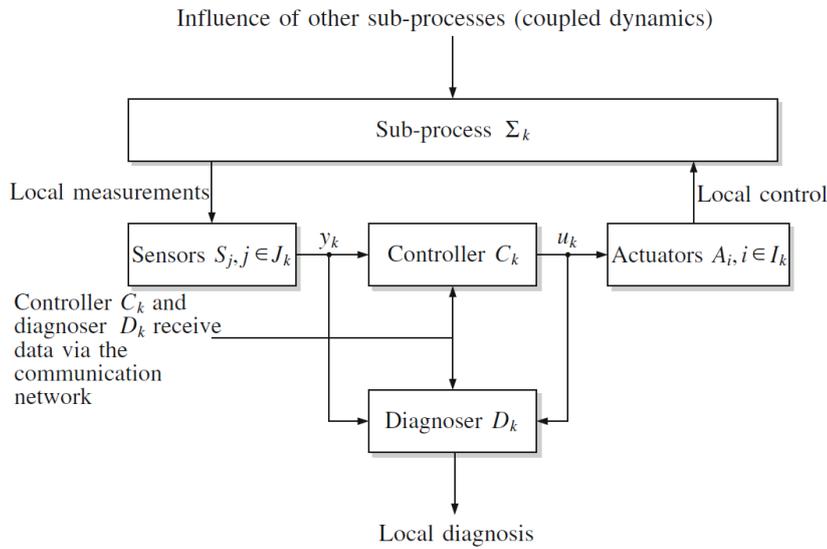


Figure 2.7: Local controller and diagnoser [8].

Regardless of the chosen design (which may be ARRs, observers, and identification-based designs), a centralized diagnoser assesses all the residuals using the data available through its linkage with the system sensors and controllers. On the other hand, in a distributed architecture, each system  $\Sigma_k, (k = 1, \dots, s)$  operates its local diagnoser, characterized by the residuals  $res_k$  it has been appointed and the decision procedure  $\delta_k$  on the residuals  $res_k$  [8].

## 2.6. Observer Based Monitoring

According to [22], the ability of a system to distinctively determine its state from the available sensor measurements is characterized by its observability. It may be a local property in nonlinear systems, with parts of the state space being more or less observable for a predetermined set of sensors and control inputs. Usually, both for linear and nonlinear systems, the issue of observability can be addressed locally by creating an appropriate observability matrix and checking its rank.

The issue of state observation refers to the design of an algorithm that enables one to recover actual state values from the observation of previous outputs [22]. Of course, for this to be meaningful, the system must first be observable. The most trivial state estimator is designed as a fixed gain observer where its ultimate goal is to reconstruct the unmeasured state vector  $\hat{x}(t)$  from the measurements  $u(t)$  and  $y(t)$  of a dynamical system, shown in the block diagram of fig. 2.8 below [21].

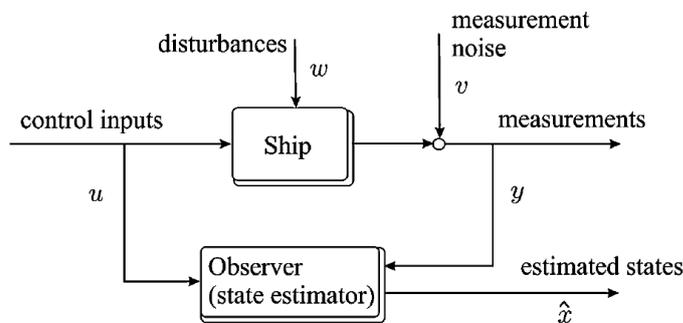


Figure 2.8: Block diagram showing the system model and the observer signal flow [21].

According to [21], more specifically, a system can be referred to as observable if, for any possible sequence of state and control vectors, the current state can be defined in finite time using only the outputs, meaning that from the outputs of the system it is possible to estimate the behavior of the entire system. On the other hand, if a system is not observable, the current values of some or all of its states cannot be defined using output sensors, implying that their value is unknown to the controller. Thus, it shall not fulfill the control requirements referred to in these outputs.

### 2.6.1. Luenberger Observers

At this point, the focus is turned to a theoretical, linear, time-invariant system. Such systems are not common in marine propulsion systems, but the following discussion will provide background knowledge of how to derive simple state observers, such as the Luenberger.

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (2.16)$$

$$y = Cx(t) \quad (2.17)$$

The disturbances in the above model can be included in the inputs vector  $u(t)$ , and there is no zero-mean white noise to consider. Considering the observer following the dynamics of equation (2.16) and (2.17), and under the assumptions just mentioned, the below equations may be derived [21]:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + \gamma(y(t), \hat{y}(t)) \quad (2.18)$$

$$\hat{y}(t) = C\hat{x}(t) \quad (2.19)$$

Where the term  $\gamma(y(t), \hat{y}(t))$  is an injection term to be constructed such that  $\hat{x}(t) \rightarrow x(t)$  as  $t \rightarrow \infty$ . The Luenberger observer can be derived by selecting the injection term  $\gamma(y(t), \hat{y}(t))$  as shown below:

$$\gamma(y(t), \hat{y}(t)) = K\epsilon(t), \quad \epsilon(t) = y(t) - \hat{y}(t) = C\tilde{x}(t) \quad (2.20)$$

Where  $\tilde{x}(t) = x(t) - \hat{x}(t)$  is the estimator error and  $K$  is a matrix of observer gains.  $\tilde{x}(t)$  implies that the difference between equation (2.18) and (2.19) can be written as follows:

$$\dot{\tilde{x}}(t) = A\tilde{x}(t) - \gamma(y(t), \hat{y}(t)) \quad (2.21)$$

A Luenberger observer can be used in marine vessels, amongst other component monitoring purposes, for navigation even if only compass measurements are available. Emphasis needs to be given to filtering, and the estimation of the yaw rate [21]. Such a Luenberger-like observer structure was used during the work of [40], where it was incorporated in the nonlinear FD model of a GTE. Besides the obvious task of defining the estimated states of the system, this observer was used to address residual generation for the model in the form of an estimation error. They concluded that FD with their proposed observers could be adopted in future works for GTE diagnosis.

### 2.6.2. Fault Detection with State Observers

Suppose the time-invariant system provided in equations (2.16) and (2.17) is now affected by immeasurable disturbances  $v(t)$  and  $n(t)$  and additive faults  $f_i(t)$  and  $f_m(t)$  as expressed [35]:

$$\dot{x}(t) = Ax(t) + Bu(t) + Vv(t) + Lf_i(t) \quad (2.22)$$

$$y(t) = Cx(t) + Nn(t) + Mf_m(t) \quad (2.23)$$

Where  $L$  and  $M$  are fault entry matrices. The introduction of these equations into the observer equation subsequently yields the following state estimation error:

$$\dot{\tilde{x}}(t) = [A - KC]\tilde{x}(t) + Vv(t) + Lf_i(t) - KNn(t) - KMf_m(t) \quad (2.24)$$

$$\epsilon = y(t) - \hat{y}(t) = C\tilde{x}(t) + Nn(t) + Mf_m(t) \quad (2.25)$$

After the asymptotic diminishing of initial state deviations  $[x(0) - \hat{x}(0)]$ , the state estimation error  $\tilde{x}(t)$  and the output error  $\epsilon(t)$  are influenced by the disturbances  $v(t)$  and  $n(t)$  and the faults  $f_i(t)$  and  $f_m(t)$ . It can be argued that the utilization of  $\tilde{x}$  as residuals if primary faults  $f_i(t)$  on the states are of interest, can be made. Nonetheless, the output error  $\epsilon(t) = res(t)$  is recommended to be

used instead as a residual. This residual is equal to zero if disturbances and faults are absent and it deviates from zero, if faults  $f_l(t)$  or  $f_m(t)$  make their occurrence and if disturbances  $n(t)$  and  $v(t)$  are nonzero. It is worth noting that the residuals are independent of the input signal  $u(t)$  as it may be noticed. Furthermore, for the acquisition of the input-output relation of the state observer (2.18) is Laplace transformed as shown [35]:

$$\begin{aligned} [sI - A + KC]\hat{x}(s) &= Bu(s) + Ky(s) \\ \Rightarrow \hat{x}(s) &= [sI - A + KC]^{-1}[Bu(s) + Ky(s)] \end{aligned} \quad (2.26)$$

When the residual error is introduced in (2.20) yields the following:

$$res(s) = \epsilon(s) = -C[sI - A + KC]^{-1}Bu(s) + [I - C[sI - A + KC]^{-1}K]y(s) \quad (2.27)$$

Applying the Laplace transform to the state observer equations with additive faults (2.24) and (2.25) and after excluding the disturbance terms leads to the following:

$$res(s) = \epsilon(s) = C[sI - (A - KC)]^{-1}[Lf_l(s) - KMf_m(s)] + Mf_m(s) \quad (2.28)$$

Additive faults  $f_l$  and  $f_m$  influence the output error  $\epsilon$  according to the observer dynamics  $[sI - (A - KC)]^{-1}$  while  $f_m$  influences  $\epsilon$ . Furthermore, if the output error  $\epsilon = res$  is utilized as the primary residual, its terms will vary since they are influenced by the faults as presented, e.g. in (2.28). For the determination of the single faults, enhanced residuals are needed, as in the instance of parity equations. The residuals can be created using fault-detection filters or with dedicated observers. It is worth noting that these techniques need multiple process outputs. Regarding the multiplicative faults, those shall arise as variations in the parameter matrices  $\delta A$ ,  $\delta B$  and  $\delta C$ , the process behavior then becomes [35]:

$$\dot{x}(t) = [A + \Delta A]x(t) + [B + \Delta B]u(t) \quad (2.29)$$

$$y(t) = [C + \Delta C]x(t) \quad (2.30)$$

The state and output error without disturbances is given as follows [35]:

$$\tilde{x}(t) = [A - KC]\tilde{x}(t) + [\Delta A - K\Delta C]x(t) + \Delta Bu(t) \quad (2.31)$$

$$e(t) = C\tilde{x}(t) + \Delta Cx(t) \quad (2.32)$$

Thus, the state and output error is dependent on the product of the parameter changes with the input signal  $u(t)$  and the state variables  $x(t)$  [35].

### 2.6.3. Fault Detection with Nonlinear Observers

A great amount of the work in nonlinear observability analysis has addressed continuous-time systems while fewer works focus on discrete-time systems. In any case, the focus here shall be turned on the ability to discern the initial state from other states given a set of known controls and measurements [50]. A general system of the following form is given by [4]:

$$\dot{x} = f(x, u); \quad y = h(x, u) \quad (2.33)$$

Here  $x$  is the state in  $R^{d_x}$ ,  $y$  the output in  $R^{d_y}$ ,  $f$  a continuously differentiable function,  $h$  a continuous function, and  $u : [0, \infty) \rightarrow U \subset R^{d_u}$  in a set  $\mathcal{U} \subset \mathcal{L}_{loc}^{\infty}([0, \infty))$  of considered inputs.  $X(x, t; s; u)$  is denoted the value at time  $s$  of the (unique) solution to system (2.33) with input  $u$ , initialized at  $x$  at time  $t$ , and  $Y(x, t; s; u)$  the respective output function at time  $s$ . A subset  $X_0$  of  $R^{d_x}$  including all the possible initial conditions for the system is considered and the following assumptions are introduced [4] that can be viewed in section 5.2.

## 2.7. Distributed Multiple Fault Isolation

Fault isolation can be carried out considering a multi-agent structure that monitors interconnected systems. In this sense, a multiple fault isolation decision logic with distributed architecture can be proposed, where a global agent is used to collect the decisions of the monitoring agents to isolate the combination of faults that have occurred and may have propagated in a network of interconnected systems [57]. In the distributed architecture, isolation occurs in two steps. The first concerns the local decision logic, while the second is the global decision logic [56].

### 2.7.1. Local Decision Logic for Fault Isolation

Consider a number of  $N$  interconnected systems  $\Sigma^{(l)}, l \in \{1, \dots, N\}$ . Each system is monitored by a number of sensors  $n_l$  and the sensor outputs are used by a local monitoring agent  $\mathcal{M}^{(l)}, l \in \{1, \dots, N\}$ . The agent  $\mathcal{M}^{(l)}$  uses a binary fault signature matrix  $F^{(l)}$  which consists of a number of rows equal to  $n_l$ , that are organized in a sensor set  $\mathcal{S}^{(l,q)}, q \in \{1, \dots, n_l\}$ , and  $N_{c_l}$  columns, with  $N_{c_l} = 2^{n_l} - 1$ . The  $q$ -th row corresponds to the  $q$ -th set of ARRs,  $\varepsilon^{(l,q)}, q \in \{1, \dots, n_l\}$ ; the  $i$ -th column, for all  $i \in \{1, \dots, N_{c_l}\}$  corresponds to the  $i$ -th combination of faults [57]. These faults might affect the sensors responsible to monitor system  $\Sigma^{(l)}$ .

Due to the exchange of sensor information through the interconnected systems, it is possible that agent  $\mathcal{M}^{(l)}$  has propagated multiple faults to neighboring agents. Such faults may provoke the violation of ARRs of neighboring agents. For this reason, the global decision logic is applied in order to isolate propagated sensor faults [57].

### 2.7.2. Global Decision Logic for Fault Isolation

The main goal of the global decision logic is to isolate sensor faults that have been propagated from neighboring agents through the measurement information that they exchange [56]. The global decision logic is based on the theoretical patterns of faults that have propagated, which prescribe the involvement of the sensor faults in the set of ARRs,  $\varepsilon^{(l)}, l \in \{1, \dots, N\}$ , expressed as [57]:

$$\varepsilon^{(l)} = \bigcup_{q \in \{1, \dots, n_l\}} \varepsilon^{(l,q)} \quad (2.34)$$

The theoretical patterns are the columns of the fault signature matrix, which consists of  $N$  rows and  $N_c = 2^p - 1$  columns, where  $p \leq \sum_{l=1}^N p_l$  where  $p_l$  is the number of propagated faults. The  $l$ -th row is associated with the set of ARRs,  $\varepsilon^{(l)}$  and the  $k$ -th column,  $k \in \{1, \dots, N_c\}$  with the  $k$ -th combination of faults [56].

## 2.8. Similar Works

When reviewing the accumulated literature available regarding model-based distributed FDI methods, the first impression is that there is a limited number of works concerning nonlinear systems with DAEs. Namely, these works belong to [75], [39] where the works are addressed for nonlinear dynamic systems in general. In the works of [60] and [65] regarding induction drives and automotive applications respectively, there is the use of DAEs but these methods do not consider distributed monitoring techniques. Other works that consider multiple faults distributed FDI for nonlinear systems with differential equations belong to [16], [9] and [41]. Some distributed FDI methods for sensor faults that find application in HVAC systems are [58] and [56].

Regarding marine vessels, a few model-based FDI methods for multiple process and sensor faults exist that focus on the navigation system of a vessel, namely [6] and [69] and they use linear models. As for the marine fuel engines, multiple process fault model-based FDI was conducted by [33], [40], [71], [70], and [68]. Considering the marine propulsion process of marine vessels, the methods that provide a multiple process fault model-based FDI namely [37] and [10] were found. It is important to mention that the methods aforementioned in this paragraph do not consider distributed monitoring agents.

### 2.8.1. Revealed Research Gaps

The revealed gaps concerning the studied FD methods for marine vessels are emphasized throughout the available literature regarding the topic. From the methods that could isolate faults and detect them, sensor faults are to a large extent ignored, as very few FD methods for marine vessels focus on sensor FDI. When taking marine engines as an example, it is apparent that the different types of marine engines that are operated today require different monitoring setups due to their wide range of sizes, combustion cycles, operating principles, and fuel mixtures, meaning that there is a lack of standardization amongst the FD methods for marine engines. However, the lack of standardization applies to FD methods for other parts of the propulsion process in general [53]. From the model-based methods available that utilize observers, the ones that consider nonlinear models are very limited [40]. Gaps exist in actuator fault localization and isolation [64] [70] which is also important for FTC purposes.

The FD methods that focus on drive train and propulsion systems, hull or structural elements, and navigation systems are often subordinated, as it was revealed by the literature. This could be due to the high criticality of marine engines as was mentioned earlier, which are brought to focus more often. Finally, a less obvious gap but one of great significance is the lack of deployment of the multi-agent structures and monitoring architectures. Today, mainly single-agent monitoring architectures are being used for CM purposes on marine vessels. This selectivity in single-agent structures for CM, at the very least, limits the prospect the rest of the multi-agent monitoring architectures provide, such as the distributed one, and rules out the advantages they can offer to the CM of marine vessels [53].

The aforementioned works make use of redundancy relations for the detection of faults, and they were considered for the development of the current work. As it will be shown later, a distributed model-based FDI methodology is proposed that managed to detect and isolate multiple process and sensor faults. This combination was not introduced by any of the aforementioned works.

## 2.9. Discussion

This far, the faults that might develop into critical failures on marine vessels were discussed. Moreover, state-of-the-art of monitoring and diagnosis for marine vessels was presented. Furthermore, the introduction of physical and analytical redundancy techniques was made. These techniques play a vital role in FD since physical redundancy allows the user to take the outputs of redundant sensors and compare them to check if they agree with each other, while analytical redundancy, makes use of model-based calculations that use sensor measurements to derive results, which are later compared with measured values to check if they agree. In both cases, any deviations from the known reference quantity considered a threshold, of either the redundant sensors or the model-based calculations that use sensor measurements should be indicated as a possible fault. Finally, in this chapter, a brief discussion was provided regarding distributed model-based FD and observer-based monitoring, as well as multiple fault isolation. In the next chapter, the marine propulsion process for a marine vessel that uses a diesel engine will be given, as well as the detailed equations that describe the associated models of the propulsion process for such a vessel and its process faults.



# 3

## Marine Propulsion Process Modeling

In this chapter, the answer to research question number 2 shall be provided. Air and exhaust gas flow dynamics are included in high-order MVFP models which demand an extensive set of parameters and detailed calibration [24]. The rationale of using a finite stage model in the MVFP simulation model is that it is capable to characterize the combustion and therefore heat input with a finite number of stages and its associated parameters and, further, is capable to calculate straightforwardly the net work output from all the stages and their parameters. Then, the in-cylinder process can be transformed into an engine cycle time-scale with reduced effort. Therefore, the acquisition of models and parameters for the Seiliger cycle is the key point to being able to apply a finite Seiliger-type process in an MVFP engine simulation environment [13]. In fig. 3.1 the various systems that constitute part of the marine propulsion process of a large vessel can be seen with their sensors. Redundant sensors are placed in all systems, in parallel configuration with the primary ones. For reasons explained earlier in subsection 2.1, like the higher frequency of process fault appearance, the focus shall be turned to the marine fuel engine of the propulsion system of a large vessel. The systems related to the marine fuel engine shall be modeled and simulated hereafter.

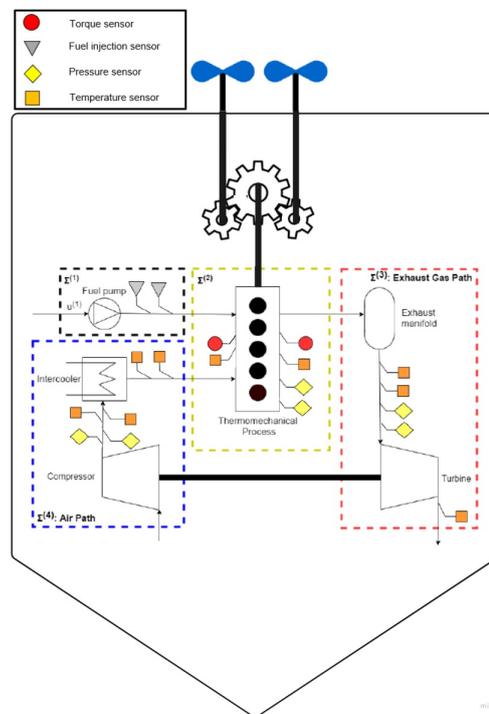


Figure 3.1: Schematic representation of the propulsion systems of a marine vessel [44].

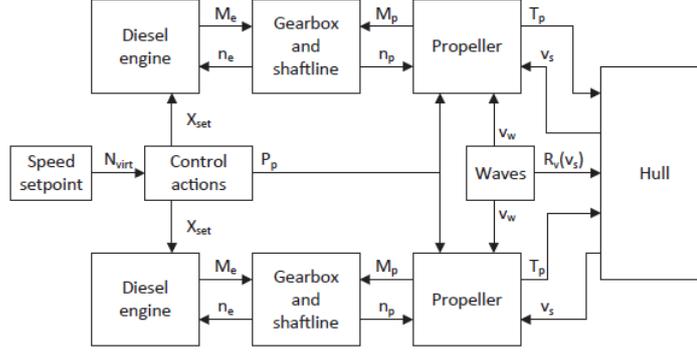


Figure 3.2: Schematic representation of direct drive propulsion system for a marine vessel showing causal coupling between models [24].

In fig. 3.2 the direction of the arrows depicts the causality of the coupled effort and flow variables, for example, engine torque  $M_e$  with engine speed  $n_e$ , propeller torque  $M_p$  with shaft speed  $n_p$  and propeller thrust  $T_p$  with ship speed  $V_s$ . Moreover, fuel injection set-point as an input and pitch ratio  $P_{pd}$  represent control variables and wave orbital speed  $V_w$  and ship resistance  $R_v$  denotes the disturbance caused by waves [24]. This work covers only a few of the models included in fig. 3.2, like the diesel engine and its associated subsystems such as the intercooler, the turbocharger, the fuel pump, and the cylinder unit as presented in fig. 3.1 with the dashed, colored lines. For more details regarding the modeling of the marine propulsion process, detailed information is provided in [24], which was the original source of the models provided hereafter.

### 3.1. General System and Sensor Equations

A general system is defined by [66]. With that work in mind, considering the heterogeneous dynamics and interconnections of the subsystems in marine fuel engines, the proposed fault diagnosis method is formed assuming a class of  $N$  nonlinear DAE-based interconnected systems  $\Sigma^{(I)}$ ,  $I = 1, \dots, N$ . Hereafter, the dependence of the signals on time (e.g.,  $x(t)$ ) will be dropped for notational brevity.

$$\Sigma^{(I)} : \begin{cases} \dot{x}^{(I)} = A^{(I)}x^{(I)} + \gamma^{(I)}(x^{(I)}, z^{(I)}, u^{(I)}) + h^{(I)}(x^{(I)}, z^{(I)}, \chi^{(I)}, u^{(I)}) + \eta_x^{(I)} + f_{x,c}^{(I)}(x^{(I)}, u^{(I)}) & (3.1) \\ 0 = \xi^{(I)}(x^{(I)}, z^{(I)}, \chi^{(I)}, u^{(I)}, f_{z,c}^{(I)}) + \eta_z^{(I)} & (3.2) \end{cases}$$

Where  $x^{(I)} \in R^{n_I-r_I}$  is the state variable vector,  $z^{(I)} \in R^{r_I}$  is the algebraic variable vector,  $\chi^{(I)} \in R^{k_I}$  are the interconnection variables from the neighbouring systems,  $u^{(I)} \in R^{l_I}$  is the control input vector,  $\gamma^{(I)} : R^{n_I-r_I} \times R^{l_I} \rightarrow R^{n_I-r_I}$  expresses the known nonlinear system dynamics,  $h^{(I)} : R^{n_I-r_I} \times R^{r_I} \times R^{k_I} \times R^{l_I} \rightarrow R^{n_I-r_I}$  denotes the known interconnection dynamics with the neighbouring subsystems,  $\eta^{(I)} \in R^{n_I-r_I}, \eta^{(I)} \in R^{r_I}$  define the system disturbances,  $\xi^{(I)} : R^{n_I} \times R^{k_I} \times R^{l_I} \rightarrow R^{n_I-r_I}$  is a smooth vector field. The term  $A^{(I)}x^{(I)}$  describes the linear part of the system's  $\Sigma^{(I)}$  dynamics, where  $A^{(I)} \in R^{(n_I-r_I) \times (n_I-r_I)}$  is the linearized part of the state equation [44]. Later, the model equations corresponding to (3.1) and (3.2), for each system, will be provided in more detail. The terms  $f_{x,c}^{(I)}$  and  $f_{z,c}^{(I)}$  in equations (3.1) and (3.2), define process faults that are affecting each state variable or algebraic variable respectively.

Each system incorporates a set of sensors  $\mathcal{S}^{(I)} = \cup_{j=1}^{n_I} \mathcal{S}^{(I)}\{j\}$  described as follows:

$$\mathcal{S}^{(I)} : \begin{cases} y_x^{(I)} = x^{(I)} + d_x^{(I)} + f_x^{(I)} \\ y_z^{(I)} = z^{(I)} + d_z^{(I)} + f_z^{(I)} \end{cases} \quad (3.3)$$

Where  $y_x^{(I)} \in R^{n_I-r_I}$  denotes the sensor values corresponding to state variables,  $y_z^{(I)} \in R^{r_I}$  defines the sensor values corresponding to algebraic variables,  $d_x^{(I)} \in R^{n_I-r_I}$ ,  $d_z^{(I)} \in R^{r_I}$  express the measurement noise vectors and  $f_x^{(I)} \in R^{n_I-r_I}$ ,  $f_z^{(I)} \in R^{r_I}$  denote sensor fault vectors [44].

## 3.2. Marine Fuel Engine State-Space Modeling

Over the 19th and 20th centuries, most engines developed from reciprocal steam engines and steam turbines into internal combustion engines, while in other cases, gas turbines [23]. At this point, the models associated with the marine fuel engine, which is part of the propulsion process, shall be presented.

### 3.2.1. Fuel Pump Model

The diesel fuel-injection system comprises an injection pump, delivery pipes and fuel injector nozzles, the governor, and a timing device [49]. Fuel injection time delay in this model is considered constant due to its small value, while a better estimate is achieved when using the engine speed as feedback in a control loop [24].

$$\Sigma^{(1)} : \dot{x}^{(1)} = -\frac{1}{\tau_X} x^{(1)} + \frac{x_{nom}^{(1)}}{\tau_X} f_u^{(1)} u^{(1)} \quad (3.4)$$

Where  $x^{(1)} \in R$  denotes the amount of fuel injected per cylinder per engine cycle in  $kg$ ,  $x_{nom}^{(1)} \in R$  is the same quantity under nominal engine conditions,  $u^{(1)} \in R$  is the fuel injection setting in %, and  $\tau_X = \frac{1}{4n_{fe}^{nom}}$  defines the fuel injection time delay in  $sec$ . The term  $f_u^{(1)}$  represents the actuator fault regarding the seizing of the fuel injection valve in %. The nominal fuel injection amount  $x_{nom}^{(1)} \in R$  is calculated below [44]:

$$x_{nom}^{(1)} = \frac{SFC^{nom} P_{fe}^{nom} k_e}{i_e n_{fe}^{nom}} \quad (3.5)$$

The term  $n_{fe}^{nom}$  represents the nominal rotational engine speed in  $rev/s$ ,  $SFC^{nom}$  is the nominal fuel consumption of the engine in  $kg/J$ , while  $P_{fe}^{nom}$  depicts the nominal power output of the no engine in  $W$ ,  $i_e$  describes the number of engine cylinders and  $k_e$  corresponds the number of crank revolutions per engine cycle ( $k_e = 1$  for a 2-stroke engine and  $k_e = 2$  for a 4-stroke engine). The output of the fuel injection sensor  $y^{(1)} \in R$  is described by [44]:

$$\mathcal{S}^{(1)} : y_x^{(1)} = x^{(1)} + d_x^{(1)} + f_x^{(1)} \quad (3.6)$$

### 3.2.2. Engine Block Model

The engine block system consists of three algebraic variables, namely the pressure  $z_1^{(2)}$  in  $Pa$  and the temperature  $z_2^{(2)}$  in  $K$  inside the engine's cylinders and the engine's shaft torque  $z_3^{(2)}$  in  $Nm$ . The mathematical expression of the system is the following [44]:

$$\begin{aligned} \Sigma^{(2)} : 0 &= \begin{bmatrix} z_1^{(2)} - \xi_{z1}^{(2)}(x^{(1)}, x^{(4)}, z_1^{(4)}, f_{z,V_1}^{(2)}) \\ z_2^{(2)} - \xi_{z2}^{(2)}(x^{(1)}, x^{(4)}, z_1^{(4)}, f_{z,V_1}^{(2)}) \\ z_3^{(2)} - \xi_{z3}^{(2)}(x^{(1)}, z_3^{(2)}, x^{(4)}, z_1^{(4)}, f_{z,V_1}^{(2)}) \end{bmatrix} \\ &= \xi^{(2)}(x^{(1)}, z^{(2)}, x^{(4)}, z_1^{(4)}, f_{z,V_1}^{(2)}) \end{aligned} \quad (3.7)$$

According to the Seiliger cycle in Appendix B and after some laborious manipulation of the equations given by [24] in table B.1,  $\xi_{z1}^{(2)}$ ,  $\xi_{z2}^{(2)}$ ,  $\xi_{z3}^{(2)}$  may be derived and are given below:

$$\xi_{z1}^{(2)} = x^{(4)} r_c^{k_a} \left( 1 + \frac{1}{c_{v,a}} \left( X_{cv} \frac{\eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z,V_1}^{(2)}) x^{(4)}} \right) \right) \frac{1}{z_1^{(4)} r_c^{k_a - 1}}$$

$$\exp \left( \frac{r_{eo} r_c}{1 + \frac{(1 - X_{cv} - X_{ct}) \frac{\eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z,V_1}) x^{(4)}}}{c_{p,a} \left( z_1^{(4)} r_c^{(\kappa_a - 1)} + \frac{X_{cv} \eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z,V_1}) x^{(4)} c_{v,a}} \right)}}} \right)^{-n_{exp}} \cdot \exp \left( - \frac{(n_{exp} - 1) \frac{X_{ct} \eta h_L r_c^{(1 - \kappa_a)} x^{(1)}}{(V_1 + f_{z,V_1}) x^{(4)}}}{\left( 1 + \frac{X_{cv} \eta h_L r_c^{(1 - \kappa_a)} x^{(1)}}{c_{v,a} (V_1 + f_{z,V_1}) x^{(4)}} + (1 - X_{cv} - X_{ct}) \frac{\eta h_L R_a r_c^{(1 - \kappa_a)} x^{(1)}}{c_{p,a} (V_1 + f_{z,V_1}) x^{(4)}} \right)} \right) \quad (3.8)$$

$$\xi_{z2}^{(2)} = z_1^{(4)} r_c^{(\kappa_a - 1)} \left( \frac{r_{eo} r_c}{1 + \frac{(1 - X_{cv} - X_{ct}) \frac{\eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z,V_1}) x^{(4)}}}{c_{p,a} \left( z_1^{(4)} r_c^{(\kappa_a - 1)} + \frac{X_{cv} \eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z,V_1}) x^{(4)} c_{v,a}} \right)}}} \right)^{1 - n_{exp}} \cdot \left( 1 + \frac{\eta h_L r_c^{(1 - \kappa_a)} x^{(1)} (c_{p,a} X_{cv} + c_{v,a} (1 - X_{cv} - X_{ct}))}{(V_1 + f_{z,V_1}) x^{(4)} c_{v,a} c_{p,a}} \right). \quad (3.9)$$

$$\exp \left( \frac{(n_{exp} - 1) \frac{X_{ct} \eta h_L r_c^{(1 - \kappa_a)} x^{(1)}}{(V_1 + f_{z,V_1}) x^{(4)}}}{\left( 1 + \frac{X_{cv} \eta h_L r_c^{(1 - \kappa_a)} x^{(1)}}{c_{v,a} (V_1 + f_{z,V_1}) x^{(4)}} + (1 - X_{cv} - X_{ct}) \frac{\eta h_L R_a r_c^{(1 - \kappa_a)} x^{(1)}}{c_{p,a} (V_1 + f_{z,V_1}) x^{(4)}} \right)} \right) \xi_{z3}^{(2)} = \frac{(V_1 + f_{z,V_1}) i_e x^{(4)}}{2\pi k_e} \cdot \left( \frac{r_c^{(\kappa_a - 1)} - 1}{\kappa_a - 1} + \frac{(1 - X_{cv} - X_{ct}) \eta h_L R_a x^{(1)}}{c_{p,a} (V_1 + f_{z,V_1}) x^{(4)}} - \frac{r_c^{(\kappa_a - 1)} + \frac{\eta h_L R_a x^{(1)} (c_{p,a} X_{cv} + c_{v,a} (1 - X_{cv} - X_{ct}))}{c_{v,a} c_{p,a} (V_1 + f_{z,V_1}) x^{(4)}}}{n_{exp} - 1} + \frac{X_{ct} \eta h_L R_a x^{(1)}}{(V_1 + f_{z,V_1}) x^{(4)}} + \frac{\xi_{z2}^{(2)}}{z_1^{(4)} (n_{exp} - 1)} \right) - Q_{loss}^{nom} \left( 1 + Q_{loss}^{grad} \frac{n_{fe}^{nom} - n_{fe}}{n_{fe}^{nom}} \right) \quad (3.10)$$

$$n_{fe} = \sqrt{\frac{2\pi}{c} z_3^{(2)}} \quad (3.11)$$

$$X_{cv} = X_{cv}^{nom} + X_{cv}^{grad} \cdot \left( \frac{n_{fe} - n_{fe}^{nom}}{n_{fe}^{nom}} \right) \quad (3.12)$$

$$X_{ct} = X_{ct}^{nom} \cdot \left( \frac{x^{(1)}}{x_{nom}^{(1)}} \right) \quad (3.13)$$

Where  $X_{cv}^{nom}$  denotes the nominal constant volume portion,  $X_{cv}^{grad}$  represents the gradient of the constant volume portion,  $X_{ct}^{nom}$  denotes the nominal constant temperature portion,  $\eta$  is the thermal efficiency incorporating both the combustion and heat release processes,  $h_L$  denotes the lower heating value of fuel at ISO conditions in  $J/kg$ .  $V_1$  depicts the cylinder volume at start of compression in  $m^3$  and  $R_a$  is the universal gas constant of air in  $J/kgK$ . The term  $f_{z,V_1}^{(2)}$  represents the additive process fault regarding bore damage. Finally,  $r_c$  is the effective compression ratio given in (B.6) later on.

The output values of this system's pressure, temperature, and torque sensors  $y^{(2)} \in R^3$  are described by [44]:

$$\mathcal{S}^{(2)} : y_z^{(2)} = z^{(2)} + d_z^{(2)} + f_z^{(2)} \quad (3.14)$$

### 3.2.3. Exhaust Receiver and Turbocharger Model

The process of blow down after the exhaust valve opens, gas ejecting during the exhaust stroke, and scavenging after the inlet opens can be described by *Zinner* blowdown [24]. The energy consumed to drive the turbine arrives from heat transfer released from the cylinder, which is the so-called exhaust-driven turbine [49]. This turbine system can be represented by one state variable and two algebraic variables, the temperature before  $z_1^{(3)}$  and after  $z_2^{(3)}$  the turbine in  $K$ . This system is described as follows [44]:

$$\Sigma^{(3)} : \begin{cases} \dot{x}^{(3)} = -\frac{1}{\tau_{pd}} x^{(3)} + \frac{1}{\tau_{pd}} f_{x,tur}^{(3)} + h^{(3)}\{x^{(3)}, z^{(3)}, \chi^{(3)}\} \\ 0 = \xi^{(3)}\{x^{(3)}, z^{(3)}, \chi^{(3)}, f_{z,Tsl}^{(3)}\} \end{cases} \quad (3.15)$$

Where  $\chi^{(3)} = [x^{(1)} \ z^{(2)} \ x^{(4)} \ z^{(4)}]^T$  denote the interconnection variables. The terms  $f_{x,tur}^{(3)}$  and  $f_{z,Tsl}^{(3)}$  represent the additive process faults regarding loss of pressure inside the turbine and loss of lubrication at the turbine shaft respectively.

The interconnection dynamics are defined below [44]:

$$h^{(3)}\{x^{(3)}, z^{(3)}, \chi^{(3)}\} = \frac{1}{\tau_{pd}} \cdot \sqrt{p_{ex}^2 + \frac{z_1^{(3)} R_g \left( \psi_1 \cdot \frac{x^{(4)} n_{fe}}{z_1^{(4)}} + x^{(1)} n_{fe} \frac{i_e}{k_e} \right)^2}{\alpha_z^2 A_{eff}^2}} \quad (3.16)$$

$$\psi_1 = \psi_1\{x^{(3)}, x^{(4)}, z^{(4)}, n_{fe}\} = \frac{i_e V_1 s_{sl}\{x^{(3)}, x^{(4)}, z^{(4)}, n_{fe}\}}{R_a k_e} \quad (3.17)$$

$$s_{sl}\{x^{(3)}, x^{(4)}, z^{(4)}, n_{fe}\} = s_{sl}^{nom} \cdot \frac{n_{fe}^{nom} m_1^{nom} x^{(4)} \Psi_{sc}\{x^{(3)}, x^{(4)}\}}{n_{fe} \left( \frac{x^{(4)} V_1}{R_a z_1^{(4)}} \right) p_1^{nom} \Psi_{sc}^{nom}} \quad (3.18)$$

$$\Psi_{sc}\{x^{(3)}, x^{(4)}\} = \sqrt{\frac{2\kappa_g}{\kappa_g - 1}} \cdot \sqrt{\left( \frac{x^{(3)}}{x^{(4)}} \right)^{(2/\kappa_g)} - \left( \frac{x^{(3)}}{x^{(4)}} \right)^{\frac{\kappa_g + 1}{\kappa_g}}} \quad (3.19)$$

The algebraic part of the subsystem is expressed as follows [44]:

$$\xi^{(3)}\{x^{(3)}, z^{(3)}, \chi^{(3)}\} = \left[ z_1^{(3)} - \frac{\psi_2 (T_{sl} + f_{z,Tsl}^{(3)}) + \tilde{\psi}_3 z_2^{(2)}}{z_2^{(3)} - \psi_4 z_1^{(3)}} \right] \quad (3.20)$$

$$\psi_2 = \psi_2\{x^{(3)}, x^{(4)}, z^{(4)}, z^{(2)}\} = \frac{c_{p,a} V_1 x^{(4)} s_{sl}\{x^{(3)}, x^{(4)}, z^{(4)}, z^{(2)}\}}{z_1^{(4)} R_a} \quad (3.21)$$

$$\psi_3 = \psi_3\{x^{(1)}, x^{(4)}, z^{(4)}\} = c_{p,g} \left( x^{(1)} + \frac{V_1 x^{(4)}}{R_a z_1^{(4)}} \right) \quad (3.22)$$

$$\tilde{\psi}_3 = \tilde{\psi}_3\{x^{(1)}, x^{(4)}, z^{(4)}, z^{(2)}\} = \left( \frac{1}{n_{bld}} + \tau_{pd} \frac{(n_{bld} - 1)h^{(3)}}{n_{bld} z_1^{(2)}} \right) \psi_3 \quad (3.23)$$

$$\psi_4 = \psi_4\{x^{(3)}, x^{(4)}\} = 1 + \eta_{turbo}\{x^{(4)}\} \cdot (\Pi_{turbo} - 1) \quad (3.24)$$

$$\Pi_{turbo} = \Pi_{turbo}\{x^{(3)}\} = \left( \frac{p_{ex}}{x^{(3)}} \right)^{\frac{\kappa_g - 1}{\kappa_g}} \quad (3.25)$$

$$\eta_{turbo}\{x^{(4)}\} = a_{turbo} + b_{turbo}x^{(4)} + c_{turbo}(x^{(4)})^2 \quad (3.26)$$

Where  $\tau_{pd}$  expresses the time delay for filling the exhaust receiver in *sec*,  $p_{ex}$  denotes the pressure after the turbocharger in *Pa* assumed equal to the atmospheric pressure,  $a_z$  is the *Zinner* turbine area decrease factor assumed 1 for a constant pressure turbocharger,  $A_{eff}$  is the turbine's effective area in  $m^2$ ,  $R_g$  is the universal gas constant of the exhaust gas in  $J/kgK$ ,  $n_{bld}$  denotes the polytropic expansion coefficient of blowdown,  $c_{p,g}$  represents the specific heat capacity at constant pressure for the exhaust gas in  $J/kgK$ ,  $s_{sl}$  defines the total slip ratio of the engine expressed in [24],  $T_{sl}$  is the temperature of the air slip during scavenging in *K* and  $a_{turbo}, b_{turbo}, c_{turbo}$  denote the polynomial coefficients of the isentropic turbine efficiency and  $\kappa_g$  expresses the specific heat ratio of the exhaust gas [44].

The output values of this system's pressure and temperature sensors  $y^{(3)} \in R^3$  are described by [44]:

$$\mathcal{S}^{(3)} : \begin{bmatrix} y_x^{(3)} \\ y_z^{(3)} \end{bmatrix} = \begin{bmatrix} x^{(3)} \\ z^{(3)} \end{bmatrix} + \begin{bmatrix} d_x^{(3)} \\ d_z^{(3)} \end{bmatrix} + \begin{bmatrix} f_x^{(3)} \\ f_z^{(3)} \end{bmatrix} \quad (3.27)$$

### 3.2.4. Air Intake Model

The flow receiver elements can be modeled with the utilization of the open thermodynamic system concept [30]. The air intake characteristics of the engine express the air excess ratio  $\lambda$ , which describes the amount of air that is not combusted after the end of each cycle. The scavenge efficiency of the engine can be set as unity due to the 4-stroke engine that the model considers with a significant air slip [62]. Thus, the air excess ratio can be defined as follows [24]:

$$\lambda = \frac{m_1(t)}{m_f(t)\sigma_f} \quad (3.28)$$

Where  $m_f(t) = x^{(1)}$  denotes the amount of fuel injected per cylinder per engine cycle and  $\sigma_f$  expresses the stoichiometric air-fuel ratio of the fuel. Furthermore, the entrapped mass at the start of compression in *kg* namely  $m_1$  is demarcated by the charge air pressure  $p_1$  and  $V_1$ , using the ideal gas law, as follows [24]:

$$m_1(t) = \frac{p_1(t)V_1}{R_a T_1} \quad (3.29)$$

The air intake system is expressed by one state variable, the charge air pressure after the compressor  $x^{(4)}$  in *Pa*, and two algebraic variables, namely the temperatures before  $z_1^{(4)}$  and after  $z_2^{(4)}$  the intercooler in *K* [44].

$$\Sigma^{(4)} : \begin{cases} \dot{x}^{(4)} = -\frac{1}{\tau_{TC}}x^{(4)} + \frac{1}{\tau_{TC}}f_{x,int}^{(4)} + h^{(4)}\{x^{(4)}, z^{(4)}, \chi^{(4)}\} \\ 0 = \xi^{(4)}\{x^{(4)}, z^{(4)}, \chi^{(4)}, f_{z,T_c}^{(4)}, f_{z,T_{inl}}^{(4)}\} \end{cases} \quad (3.30)$$

Where  $\chi^{(4)} = [x^{(1)} \ x^{(3)} \ z^{(3)}]^T$  denote the interconnection variables. The terms  $f_{x,int}^{(4)}$ ,  $f_{z,T_c}^{(4)}$  and  $f_{z,T_{inl}}^{(4)}$  are additive process faults concerning the loss of pressure inside the compressor, blockage of the heat exchanger of the intercooler and loss of lubrication at the shaft of the compressor respectively.

The interconnection dynamics are defined below [44]:

$$h^{(4)}\{x^{(4)}, z^{(4)}, \chi^{(4)}\} = \frac{p_{amb}}{\tau_{TC}} \cdot (1 + \chi_g \delta_f \eta_{TC}\{x^{(4)}\} r_{TC}\{z^{(3)}\} (1 - \Pi_{turbo}))^{\left(\frac{\kappa_a - 1}{\kappa_a}\right)} \quad (3.31)$$

$$\delta_f = \delta_f\{x^{(4)}, z^{(4)}, \chi^{(4)}\} = 1 + \frac{x^{(1)}}{(1 + s_{sl}) \frac{V_1 x^{(4)}}{R_a z_1^{(4)}}} \quad (3.32)$$

$$\eta_{TC}\{x^{(4)}\} = a_\eta + b_\eta x^{(4)} + c_\eta (x^{(4)})^2 \quad (3.33)$$

$$r_{TC}\{z^{(3)}\} = \frac{z_1^{(3)}}{T_{amb}} \quad (3.34)$$

$$\chi_g = \frac{c_{p,g}}{c_{p,a}} \quad (3.35)$$

The algebraic part of this system is expressed as follows [44]:

$$\xi^{(4)}\{x^{(4)}, z^{(4)}, \chi^{(4)}\} = \begin{bmatrix} z_1^{(4)} - \xi_{z1}^{(4)} \\ z_2^{(4)} - \xi_{z2}^{(4)}\{x^{(3)}, z^{(3)}\} \end{bmatrix} \quad (3.36)$$

$$\xi_{z1}^{(4)} = T_c - \epsilon_{inl} (T_{inl} + f_{z,T_{inl}} - T_c) \quad (3.37)$$

$$\xi_{z2}^{(4)} = T_{amb} + x^{(3)} \chi_g \eta_{turbo} (\delta_f + \eta_{com}) (z_2^{(3)} - z_1^{(3)}) + f_{z,T_c}^{(4)} \quad (3.38)$$

Where  $\tau_{TC}$  denotes the compressor time delay in *sec*,  $p_{amb}$  represents the ambient air pressure in *Pa*,  $T_{amb}$  is the ambient air temperature in *K*,  $a_\eta, b_\eta, c_\eta$  represent the polynomial coefficients of the turbocharger for estimating its efficiency,  $\eta_{com}$  denotes the mechanical efficiency of the compressor that can be considered constant,  $T_c$  is the charge air temperature after the intercooler in *K*,  $\epsilon_{inl}$  describes the parasitic effectiveness of the heat exchange between the inlet duct and the air, while  $T_{inl}$  depicts the temperature of the inlet duct that heats the inducted air in *K* [44].

The output values of this system's pressure and temperature sensors  $y^{(4)} \in R^3$  are [44]:

$$\mathcal{S}^{(4)} : \begin{bmatrix} y_x^{(4)} \\ y_z^{(4)} \end{bmatrix} = \begin{bmatrix} x^{(4)} \\ z^{(4)} \end{bmatrix} + \begin{bmatrix} d_x^{(4)} \\ d_z^{(4)} \end{bmatrix} + \begin{bmatrix} f_x^{(4)} \\ f_z^{(4)} \end{bmatrix} \quad (3.39)$$

### 3.3. Modeled Process Faults

At this point, the modeled marine propulsion process faults, that were added to the propulsion process models shall be shown. Considering the time dependency and the additive or multiplicative nature of the faults, table 3.1 can be created. The modeling of the process faults is in accordance with what was shown in subsection 2.3.1 and in fig. 2.4. The variables that are affected by each process fault can be seen in table 3.1 in the last column. Furthermore, in table 3.1 one may notice the system that is affected by each process fault and its units of measurement, and the symbol that was used in the system equations. The permanent faults that were introduced in the propulsion process, are either abrupt or incipient, offset or drift faults, and this characterization is attributed to their time profile and fault function.

Here it was assumed that the process faults will have a specific profile, according to the development of similar process faults from the available literature and the author's knowledge regarding the behavior of mechanical systems. In table 3.2, the last column indicates their fault equation that was introduced in the propulsion model systems for each process fault respectively.

Table 3.1: Affected state and algebraic variables from the process faults.

Propulsion Process System	Process Fault	Symbol	Units	Variable Affected
$\Sigma^{(3)}$	<i>Turbo side seal loss</i>	$f_{x,tur}^{(3)}$	Pa	$x^{(3)}$
	<i>Loss of lubrication turbo side</i>	$f_{z,Tsl}^{(3)}$	K	$z_1^{(3)}$
$\Sigma^{(4)}$	<i>Air side seal loss</i>	$f_{x,int}^{(4)}$	Pa	$x^{(4)}$
	<i>Cooler tube blockage</i>	$f_{z,Tc}^{(4)}$	K	$z_2^{(4)}$
	<i>Loss of lubrication air side</i>	$f_{z,Tinl}^{(4)}$	K	$z_1^{(4)}$
$\Sigma^{(2)}$	<i>Bore damaged</i>	$f_{z,V_1}^{(2)}$	$m^3$	$z_1^{(2)}, z_2^{(2)}, z_3^{(2)}$
$\Sigma^{(1)}$	<i>Valve seizing</i>	$f_u^{(1)}$	%	$x^{(1)}$

Table 3.2: Characterization and modeling of permanent faults [57] [34].

Fault	Fault Type	Time profile	Evolution $\kappa$	Function $\phi$	Fault Equation $\forall t \in \{t_1, \infty\}$
$f_{x,tur}^{(3)}$	<i>Additive</i>	<i>Incipient</i>	$0 < \kappa \ll \infty$	$R_{x,tur}^{(3)} x^{(3)}(t - t_1)$	$(1 - e^{-\kappa(t-t_1)})\phi$
$f_{z,Tsl}^{(3)}$	<i>Additive</i>	<i>Abrupt</i>	$\kappa \rightarrow \infty$	$R_{z,Tsl}^{(3)}(t - t_1)$	$(1 - e^{-\kappa(t-t_1)})\phi$
$f_{x,int}^{(4)}$	<i>Additive</i>	<i>Incipient</i>	$0 < \kappa \ll \infty$	$R_{x,int}^{(4)} x^{(4)}(t - t_1)$	$(1 - e^{-\kappa(t-t_1)})\phi$
$f_{z,Tc}^{(4)}$	<i>Additive</i>	<i>Incipient</i>	$0 < \kappa \ll \infty$	$R_{z,Tc}^{(4)}(t - t_1)$	$(1 - e^{-\kappa(t-t_1)})\phi$
$f_{z,Tinl}^{(4)}$	<i>Additive</i>	<i>Abrupt</i>	$\kappa \rightarrow \infty$	$R_{z,Tinl}^{(4)}(t - t_1)$	$(1 - e^{-\kappa(t-t_1)})\phi$
$f_{z,V_1}^{(2)}$	<i>Additive</i>	<i>Incipient</i>	$0 < \kappa \ll \infty$	$R_{z,V_1}^{(2)}(t - t_1)$	$(1 - e^{-\kappa(t-t_1)})\phi$
$f_u^{(1)}$	<i>Multiplicative</i>	<i>Incipient</i>	$0 < \kappa \ll \infty$	$\phi_u^{(1)}$	$1 - (1 - e^{-\kappa(t-t_1)})\phi$

### 3.4. Discussion

In this chapter, the models that constitute part of the propulsion process of a marine vessel were presented. A state space representation was provided for the systems and the corresponding algebraic variables for each system were given. These variables provide a way to calculate the values that will later be compared with sensor measurements to create residuals and examine if the error bounds are violated at any time instant, as it shall be described in chapter 5. The validation of the marine propulsion process models shall be provided in the next chapter.

# 4

## Propulsion Process Model Validation

So far, the analysis to obtain the conceptual model for the marine engine was described. At this point, the model quantitative validation that is conducted for each of the four systems shall be shown. The aim here is to witness how well the behavior of a propulsion plant can be predicted, based on the models and the comparison with Factory Acceptance Test (FAT) data that were found in [24]. The data concern only the specific fuel consumption, the maximum cylinder pressure, and the temperatures before and after the turbine and intercooler of the turbocharger, as well as compressor pressure. Therefore, the answer to research question number 3 shall be provided in this chapter.

It is important to mention here that some of the models introduced in the previous chapter could not be compared with FAT measurements since they were not available. For those models, only the model results are presented and their validation remains incomplete until FAT measurement data is acquired. Nevertheless, the model that could be compared with available FAT measurements can be viewed, and for most of them, there is a good match with the measurements, at engine speeds close to the nominal operating point of the engine. This is due to the absence of a controllable pitch propeller in the simulation, which limits the performance of the simulated models below the nominal output of the engine, and the propeller pitch should change accordingly to meet the engine power output. The physical model that the FAT measurements were taken from, includes a controllable pitch propeller that can vary its pitch according to the engine speed. If such a propeller model was present in the simulation, the match with the FAT measurements would have been closer.

### 4.1. Fuel Pump Model Validation

For plentitude, fig. 4.1 shows the specific fuel consumption over the complete operating envelope of the engine. When comparing this with the specific fuel consumption of a typical high-speed engine, as published in [63], the model results are close to 10% load of the engine, for engine speeds greater than 800rpm.

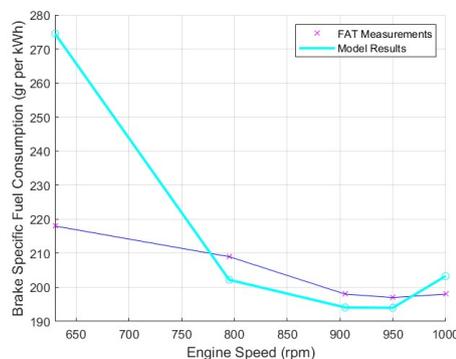


Figure 4.1: Specific fuel consumption vs engine speed.

## 4.2. Diesel Engine Model Validation

The diesel engine model is run at the FAT speed and power settings with the parameters from table 4.1. The results are shown in appendix B in figs. B.5, B.3, and B.4 for combustion pressure for Seiliger stage 6, cylinder temperature, and engine torque respectively.

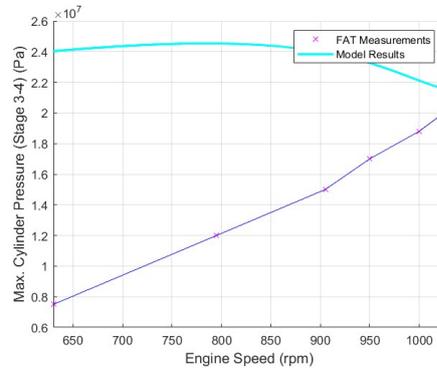


Figure 4.2: Maximum combustion pressure at stage 4 vs engine speed.

In fig. 4.2 the maximum cylinder pressure is compared with FAT measurements. Here, one may notice that the simulation results are accurate at an engine speed close to 1000rpm, which is the nominal speed of the marine fuel engine. This is because of the absence of the variable pitch propeller in the simulation, that the physical vessel has. The absence of such a propeller in the simulation limited the performance of these models since it was assumed that the propeller pitch is fixed and matches the nominal power output of the engine when it runs at the nominal engine speed.

## 4.3. Exhaust Receiver and Turbocharger Model Validation

Full validation of the model instructs measurements across all operating points of the engine. Therefore, a vast measurement campaign is advised for further model validation. Figure 4.3 shows the exhaust valve temperature and the exhaust receiver temperature, and therefore the entry temperature to the turbine. Figure 4.4 portrays the temperature after the exhaust gasses pass through the turbocharger's turbine. The trends in the figures 4.3 and 4.4 show greater deviation than the rest of the models, especially above 900rpm, at which point, according to [24], *"the cylinder bypass valve opens and provides extra cooling air to the exhaust receiver, which can also be seen as a discontinuity of the trend of the measurements, near that particular engine speed ."*

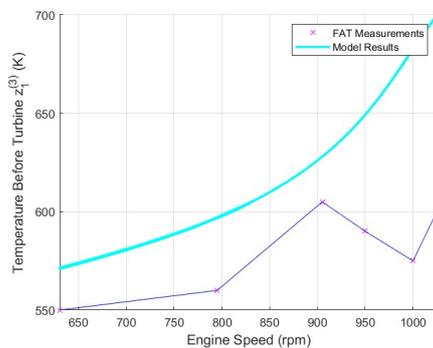


Figure 4.3: Exhaust receiver temperature vs engine speed.

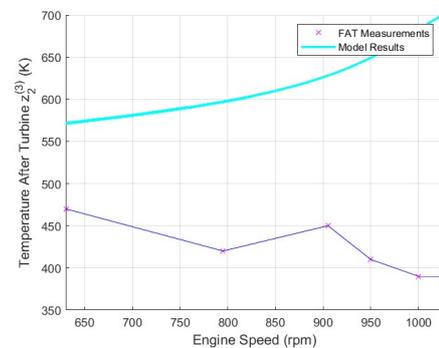


Figure 4.4: Exhaust temperature after turbocharger vs engine speed.

Table 4.1: Fuel engine parameters [24].

Diesel engine parameter description	Symbol	Value	Units
Nominal engine power	$P_{fe}^{nom}$	$5400 \times 10^3$	$W$
Nominal engine speed	$n_{fe}^{nom}$	16.71	$rev/s$
Nominal fuel injection mass	$x_1^{nom}$	0.003	$kg/rev$
Number of cylinders	$i_e$	12	-
Number of revs per cycle	$k_e$	2	-
Bore diameter	$D_B$	0.28	$m$
Stroke length	$L_S$	0.33	$m$
Crank rod length	$L_{CR}$	0.64063	$m$
Inlet closure angle	$\alpha_{IC}$	224	$^\circ$
Exhaust open angle	$\alpha_{EO}$	119	$^\circ$
Nominal spec. fuel consumption	$SFC^{nom}$	$5.5 \times 10^{-8}$	$kg/J$
Combined heat release and combustion efficiency	$\eta$	0.886	-
Geometric compression ratio	$\epsilon_c$	13.8	-
Cylinder volume at state 1	$V_1$	0.0199	$m^3$
Nominal pressure at state 1	$p_1^{nom}$	$4.1 \times 10^5$	$Pa$
Maximum cylinder pressure	$p_{max}^{nom}$	$188 \times 10^5$	$Pa$
Temperature after the intercooler	$T_c$	323	$K$
Temperature of the inlet duct	$T_{inl}$	423	$K$
Parasitic heat exchanger effectiveness	$\epsilon_{inl}$	0.05	-
Fuel injection time delay	$\tau_\chi$	0.015	$sec$
Turbocharger time constant	$\tau_{TC}$	51	$sec$
Exhaust receiver time constant	$\tau_{pd}$	0.01	$sec$
Gas constant of air	$R_a$	287	$J/kgK$
Gas constant of exhaust gas	$R_g$	271	$J/kgK$
Specific heat of air at constant volume	$c_{v,a}$	717.5	$J/kgK$
Specific heat of air at constant pressure	$c_{p,a}$	1005	$J/kgK$
Specific heat of exhaust gas at constant pressure	$c_{p,g}$	1100	$J/kgK$
Isentropic index of air	$\kappa_a$	1.4	-
Isentropic index of the exhaust gas	$\kappa_g$	1.353	-
Lower heating value of fuel	$h_L$	$42700 \times 10^3$	$J/kg$
Polytropic exponent for expansion	$n_{exp}$	1.38	-
Polytropic exponent for blowdown	$n_{bln}$	1.38	-
Constant volume portion gradient	$X_{cv}^{grad}$	-0.4164	-
Constant volume portion nominal	$X_{cv}^{nom}$	0.06017	-
Constant temperature portion nominal	$X_{ct}^{nom}$	0.4	-
Turbocharger factor	$a_\eta$	$-3.29 \times 10^{-12}$	-
Turbocharger factor	$b_\eta$	$-2.52 \times 10^{-6}$	-
Turbocharger factor	$c_\eta$	0.2143	-
Nominal slip ratio	$s_{sl}^{nom}$	0.1719	-
Ambient pressure	$P_{amb}$	$1 \times 10^5$	$Pa$
Ambient temperature	$T_{amb}$	318	$K$

#### 4.4. Air Intake Model Validation

Validation of the air intake model of the engine requires further measurement data. Nevertheless, in fig. 4.5 one can see the turbocharged air pressure with respect to engine speed. The charged air pressure deviates from the FAT charged air pressure data. The underlying reason for this great deviation could be the absence of a variable pitch propeller model inside the simulation since a convergence to the measurements appears only at a speed close to the nominal engine speed. Nevertheless, for the FDI purposes ahead, these models suffice and the remaining task this thesis is accountable for can be carried out.

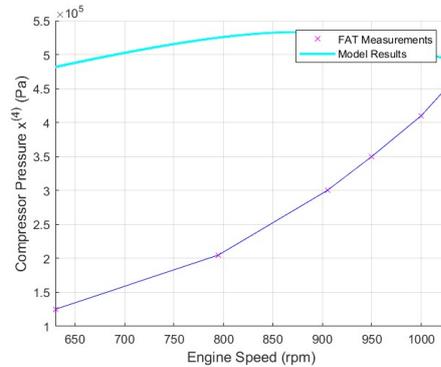


Figure 4.5: Charged air pressure vs engine speed.

#### 4.5. Discussion

In this chapter, the results of the model simulation were shown. It is true that most models that had FAT measurements to be compared with depict deviations far greater than those measurements. Nevertheless, for engine speeds (900-1000rpm) close to the nominal one, the models are closer to the FAT measurements and can be regarded as valid enough for the FDI purposes that shall be presented in chapter 5, to carry out the remaining goals of this project. Therefore, these simulated models shall form the basis to implement the model-based FDI scheme and reproduce some results regarding the FDI scheme that will be shown hereafter.

# 5

## Fault Diagnosis Scheme

In this chapter, the answer to research question 4 shall be provided. This will be done by first presenting the current problem at hand that the thesis tries to resolve. Thereafter, a feasible FDI methodology is proposed and explained thoroughly. The methodology is carried out with the use of monitoring modules, that exist in the monitoring agents of each system. The decision logic of the monitoring modules relies on analytical redundancy relations of residuals and adaptive thresholds, derived using observers and other error estimation techniques.

An FDI scheme should be robust concerning modeling errors and sensitive concerning incipient faults. Attention has been drawn over the years in the research society, regarding this trade-off between robustness and sensitivity [54]. The number of works that concern sensor fault diagnosis among distributed and decentralized FDI schemes for physically interconnected systems, even though, the detection and isolation of sensor faults have become of paramount importance lately, as a result of a large number of sensors and sensor networks, used for [56]:

1. *“Monitoring and controlling large-scale and complex systems;*
2. *Providing rich and redundant information for executing safety-critical tasks;*
3. *Offering information to the citizens and governmental agencies to resolve problems promptly in emergency situations.”*

### 5.1. Problem Formulation

In chapter 3, the system and sensor equations corresponding to (3.1), (3.2), and (3.3), respectively for each subsystem were provided. Each sensor fault vector is given by  $f^{(l)}(t) = [f_x^{(l)}(t) \ f_z^{(l)}(t)]^T = [f_1^{(l)}(t), \dots, f_{n_l}^{(l)}(t)]^T$ , where  $f_j^{(l)}(t) \forall j \in \{1, \dots, n_l\}$  denotes the change in the output caused by a fault in the  $j$ -th sensor. Permanent, abrupt, offset faults can be modeled as follows [57]:

$$f_j^{(l)} = \begin{cases} 0, & t < T_{f_j}^{(l)} \\ \phi_j^{(l)}, & t \geq T_{f_j}^{(l)} \end{cases} \quad \forall j \in \{1, \dots, n_l\} \quad (5.1)$$

Specifically, in the system equation (3.3) the additive sensor faults were introduced.  $T_{f_j}^{(l)}$  is the time instant of occurrence of the  $j$ -th fault,  $\phi_j^{(l)}$  is associated to a fault function. Note that multiple faults may occur simultaneously or sequentially, for example,  $T_{f_1}^{(l)} \leq T_{f_2}^{(l)} \leq \dots \leq T_{f_j}^{(l)}$  [56]. The fault vectors for the additive process faults there are given as  $f_c^{(l)}(t) = [f_{x,c}^{(l)}(t) \ f_{z,c}^{(l)}(t)]^T$  in system equation (3.1) and (3.2), while the multiplicative process fault was described by  $f_u^{(l)}(t)$  in (3.1). Such permanent incipient, drift-like faults can be modeled as follows:

$$f_c^{(l)} = \begin{cases} 0, & t < T_{f_c}^{(l)} \\ \left(1 - e^{-\kappa_c^{(l)}(t - T_{f_c}^{(l)})}\right) \phi_c^{(l)}(t - T_{f_c}^{(l)}), & t \geq T_{f_c}^{(l)} \end{cases} \quad (5.2)$$

$T_{fc}^{(l)}$  is the time instant of occurrence of the process fault that affects a specific component  $c$  of the propulsion process,  $\phi_c^{(l)}$  is associated to the fault function of each process fault, and  $\kappa_c^{(l)}$  is associated to a fault evolution mode as it was mentioned in section 2.3.

In this work, the inclusion of the process faults in the system equations has introduced the additional complexity of distinguishing the process faults from the sensor faults, after their detection. The discrimination between process and sensor faults can be achieved with the introduction of redundant sensors, as it was mentioned in section 2.4. In brief, if two or more sensors that are responsible to measure the same variable in a system, show a significant change in the residual that leads to the violation of the associated ARR, causing the detection of a fault, then this fault is assigned as a process fault. More details will be explained regarding the rationale for adding redundant sensors when the local decision logic is described.

The objective of this thesis is to design a methodology for the detection and isolation of multiple permanent, abrupt or incipient, offset or drift-like process faults such as the ones given in (5.2), but also permanent, abrupt, offset sensor faults given in (5.1) for nonlinear DAE interconnected systems defined by (3.1), (3.2) and (3.3), subject to the assumptions provided in section 5.2, but also to the ones given here:

## 5.2. Assumptions Introduced

For the development of this work, after reviewing the relevant literature regarding fault diagnosis and fault modeling, the following assumptions were considered necessary to be introduced:

- **Assumption 1:** Solutions to the system (2.33) initialized in  $X_0$  are well defined in positive time and belong to an open set  $\mathcal{X}$ . In other words, for all  $u$  in  $\mathcal{U}$ , for all  $x_0$  in  $X_0$  and for all  $s$  in  $[0, \infty)$ ,  $X(x_0, 0; s; u)$  is well defined and is in  $\mathcal{X}$ .

It is further assumed that the inputs and outputs are known in a causal way, the observer can use only their past or current values, i.e., at time  $t$ ,  $u|_{[0,t]}$  and  $y|_{[0,t]}$  only. The idea here is to transform system (2.33) into a Hurwitz form [4]:

$$\dot{\zeta} = A\zeta + By \quad (5.3)$$

- **Assumption 2:** Solutions to the system (2.33) initiated from  $\mathcal{X}$  do not deviate substantially from zero as time progresses [4], in the absence of faults.
- **Assumption 3:** In this work only permanent sensor and process faults shall be considered. The sensor faults are abrupt and offset, while the process faults can be either incipient or abrupt, offset or drift-like as was mentioned in section 2.3.1.

The reason *Assumption 3* exists is to turn the attention toward permanent faults since such faults influence the sensors or the systems they occur in, for the remainder of their useful lives, unlike intermittent faults that might make their appearance for a period of time followed by a disappearance period. Therefore, permanent faults were considered to have a longer-lasting impact. Intermittent or even transient faults could be introduced by future researchers to the current framework, to make it more inclusive and complete.

- **Assumption 4:** In this work, while the simulation results that are presented in chapter 6 were being created, it was assumed that no more than two process and sensor faults may affect the same system at the same time.

*Assumption 4* was introduced due to the fact that the simultaneous occurrence of many different process faults at the same time may lead to the simulation failure since no FTC was introduced and the fault tolerance of the simulated marine engine systems, in general, is low.

- **Assumption 5:** The system disturbance and the measurement noise of each sensor are unknown but uniformly bounded, meaning [44]:

$$|\eta_{xj}^{(l)}| \leq \bar{\eta}_{xj}^{(l)}, |\eta_{zj}^{(l)}| \leq \bar{\eta}_{zj}^{(l)}, |d_j^{(l)}| \leq \bar{d}_j^{(l)} \quad \forall j \in \{1, \dots, n_l\}$$

Where  $\bar{\eta}_{xj}^{(l)}$ ,  $\bar{\eta}_{zj}^{(l)}$  and  $\bar{d}_j^{(l)}$  are known.

- **Assumption 6:** The nonlinear vector fields  $\gamma^{(l)}$  and  $h^{(l)}$  are locally Lipschitz for  $x \in \mathcal{X}$ ,  $z \in \mathcal{Z}$  for all  $u \in \mathcal{U}$  and  $t \geq 0$  with Lipschitz constants  $\lambda_{\gamma_l}$  and  $\lambda_{h_l}$  respectively [44].

*Assumption 5* provides a bound commonly used for distinguishing between modeling uncertainties and faults and a representation of the existing knowledge for the sensor noise that is typically provided in a given range of operation by sensor manufacturers or introduced when a noise-free analog signal is converted into a digital one with a finite number of digits. These bounds can be acquired either analytically, by explicitly defining the sources of uncertainty and their corresponding bounds, or using offline identification methods. *Assumption 6* describes the class of nonlinear interconnected systems under consideration. Many nonlinearities in practical systems can be considered locally Lipschitz [56].

### 5.3. Distributed Fault Detection and Isolation

Allowing the exchange of sensor information between monitoring agents can improve sensor fault detectability, compared to a decentralized architecture with no communication between the monitoring agents, where often the effects of interconnections minister as bounded disturbances. On the other hand, the exchange of information may lead to fault propagation, which complicates the isolation of faults [56].

The first-level diagnosis consists of the local monitoring agents  $\mathcal{M}^{(l)}$ , shown in fig. 5.1, and is devised to detect and isolate multiple faults that may directly affect some sensors in the underlying set of sensors  $\mathcal{S}^{(l)}$  or process faults that affect the underlying system  $\Sigma^{(l)} \forall l \in \{1, \dots, N\}$  with  $N = 4$ . Neighboring monitoring agents can exchange data provided by the sensors that measure the interconnection state  $x^{(l)}$  and algebraic variables  $z^{(l)}$ . A fault in the sensor set  $\mathcal{S}^{(l,q)} \forall q \in \{1, \dots, n_l\}$  with  $n_1 = 2, n_2 = n_4 = 6, n_3 = 5$  can be propagated to neighboring modules due to the information exchange, as well as between the agent  $\mathcal{M}^{(l)}$  and the neighboring monitoring agents. Therefore, the global decision logic is needed by the global monitoring agent,  $\mathcal{G}$ , to attribute the propagating faults, to the sensors they originate from. In particular, the second-level diagnosis, denoted by  $\mathcal{G}$ , uses information from the monitoring agents  $\mathcal{M}^{(l)}$  for isolating sensor faults propagating in the top layer due to the communication of the monitoring agents  $\mathcal{M}^{(l)}$  [57].

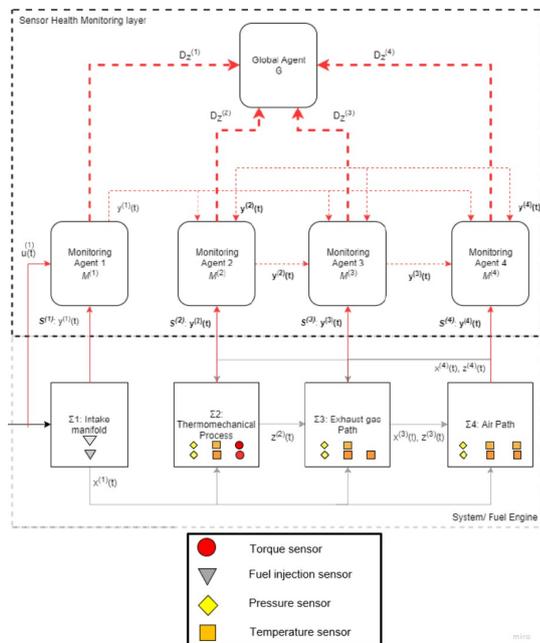


Figure 5.1: Distributed FDI scheme application using an MVFP marine propulsion model [44].

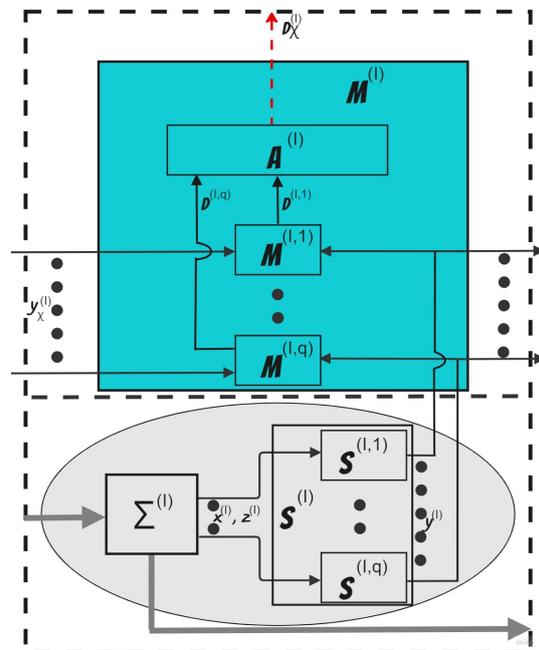


Figure 5.2: Description of monitoring agent  $\mathcal{M}^{(l)}$ , system  $\Sigma^{(l)}$ , and sensors  $\mathcal{S}^{(l)}$  connected to it.

The distributed scheme in fig. 5.1, is based on previous works [44], [57]. As shown in fig. 5.1, for each one of the interconnected systems,  $\Sigma^{(l)}$ , a monitoring agent  $\mathcal{M}^{(l)}$  is assigned. The monitoring agents have been created considering the DAE nature of the fuel engine and nonlinear algebraic residuals with their associated adaptive thresholds were proposed [44]. Additionally, in fig. 5.2 the illustration of the  $l$ -th monitoring agent, system  $\Sigma^{(l)}$ , and its sensors  $\mathcal{S}^{(l)}$  can be viewed. One may notice the content of the agent  $\mathcal{M}^{(l)}$  which is designed comprising  $n_l$  monitoring modules  $\mathcal{M}^{(l,q)}$ , and an aggregation module  $\mathcal{A}^{(l)}$ . The sensors that are connected to  $\Sigma^{(l)}$ , are organized in  $n_l$  sensor sets  $\mathcal{S}^{(l,q)}$ . The outputs of these sensor sets are fed to the monitoring modules that utilize them, along with the outputs of the interconnection sensors to form ARR. More details regarding the use of these ARRs by the modules and the need for the aggregation module  $\mathcal{A}^{(l)}$ , will be provided in the local decision logic later in this chapter.

### 5.3.1. Residual Generation and Error Estimation

This subsection deals with the design of the module  $\mathcal{M}^{(l,q)}$ ,  $q \in \{1, \dots, n_l\}$ . Considering *Assumptions 1-2*, the estimation model of  $\mathcal{M}^{(l,q)}$  is formed by selecting a Lipschitz nonlinear observer  $\mathcal{O}^{(l,q)}$ , shown in the following equation [56]:

$$\mathcal{O}^{(l,q)} : \dot{\hat{x}}^{(l,q)} = A^{(l)} \hat{x}^{(l,q)} + \gamma^{(l)} \left( \hat{x}^{(l,q)}, u^{(l)}, y_z^{(l,q)} \right) + h^{(l)} \left( \hat{x}^{(l,q)}, u^{(l)}, y_z^{(l,q)}, y_\chi^{(l,q)} \right) + L^{(l,q)} \left( y_x^{(l,q)} - \hat{x}^{(l,q)} \right) \quad (5.4)$$

Subtracting (3.1) from (5.4) yields the following [44]:

$$\dot{\epsilon}_x^{(l,q)} = A_L^{(l,q)} \epsilon_x^{(l,q)} + \tilde{\gamma}^{(l,q)} + \tilde{h}^{(l,q)} + \eta_x^{(l,q)} - L^{(l,q)} d_x^{(l,q)} \quad (5.5)$$

Where  $\hat{x}^{(l,q)} \in R^{n_l}$  is the estimation of  $x^{(l)}$  (based on the sensor measurements  $y^{(l,q)}$  with  $\hat{x}^{(l,q)}(0) \in \mathcal{X}^{(l)}$  that is an open set,  $L^{(l,q)} \in R^{(n_l-r_l) \times (n_l-r_l)}$  is the observer gain matrix is selected such that the matrix  $A_L^{(l,q)} = A^{(l)} - L^{(l,q)}$  is Hurwitz, and  $y_z^{(l,q)}$  is the transmitted sensor information. Furthermore,  $\epsilon_x^{(l,q)} = x^{(l)} - \hat{x}^{(l,q)}$  is the state estimation error,  $\tilde{\gamma}^{(l,q)} \triangleq \gamma^{(l)}(x^{(l)}, z^{(l)}, u^{(l)}) - \gamma^{(l)}(\hat{x}^{(l,q)}, y_z^{(l,q)}, u^{(l)})$  and  $\tilde{h}^{(l,q)} = h^{(l)}(x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)}) - h^{(l)}(\hat{x}^{(l,q)}, y_z^{(l,q)}, y_\chi^{(l,q)}, u^{(l)})$  [44]. Keep in mind that the given observer is based on the formulation of observers for Lipschitz nonlinear systems, which is adjusted appropriately for the nonlinear interconnected subsystems [56].

The residual vector is defined as follows [44]:

$$\epsilon_y^{(l,q)} = \begin{bmatrix} y_x^{(l,q)} - \hat{x}^{(l,q)} \\ -\xi^{(l)} \left( y_x^{(l,q)}, y_z^{(l,q)}, y_\chi^{(l,q)}, u^{(l)} \right) \end{bmatrix} = \begin{bmatrix} \epsilon_{y_x}^{(l,q)} \\ \epsilon_{y_z}^{(l,q)} \end{bmatrix} \in R^{n_l} \quad (5.6)$$

The residual  $\epsilon_{y_x}^{(l,q)}$  can also be defined as  $\epsilon_{y_x}^{(l,q)} = \epsilon_x^{(l,q)} + d_x^{(l,q)}$ . Adding (3.2) to the expression of  $\epsilon_{y_z}^{(l,q)}$  in (5.6) yields the following [44]:

$$\epsilon_{y_z}^{(l,q)} = \xi^{(l)} \left( x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)} \right) + \eta_z^{(l)} - \xi^{(l,q)} \left( y_x^{(l,q)}, y_z^{(l,q)}, y_\chi^{(l,q)}, u^{(l)} \right) \quad (5.7)$$

### 5.3.2. Adaptive Thresholds Computation

The thresholds are designed to bound the respective residuals  $\epsilon_{y_x}^{(l,q)}$  and  $\epsilon_{y_z}^{(l,q)}$  under healthy sensor conditions. The boundedness of  $\epsilon_{y_x}^{(l,q)}$  and  $\epsilon_{y_z}^{(l,q)}$  is exploited in order to formulate the adaptive thresholds [57].

$$\left| \epsilon_{y_{x_j}}^{(l,q)}(t) \right| \leq \bar{\epsilon}_{y_{x_j}}^{(l,q)}(t) \quad \forall j \in \{1, \dots, n_l - r_l\} \quad (5.8)$$

$$\epsilon_{y_{z_j}}^{(l,q)}(t) \in \left[ \underline{\epsilon}_{y_{z_j}}^{(l,q)}(t), \bar{\epsilon}_{y_{z_j}}^{(l,q)}(t) \right] \quad \forall j \in \{1, \dots, r_l\} \quad (5.9)$$

Considering *Assumptions 5-6* and after some mathematical manipulations of (5.5) the adaptive threshold can be derived as follows [56]:

$$\bar{\epsilon}_{y_{x_j}}^{(l,q)}(t) = E^{(l,q)}(t) + \rho^{(l,q)} \Lambda_l \int_0^t E^{(l,q)}(\tau) e^{-\xi^{(l,q)}(t-\tau)} d\tau + \bar{d}_{x_j}^{(l,q)} \quad (5.10)$$

Where

$$E^{(l,q)}(t) = \rho^{(l,q)} e^{-\xi^{(l,q)} t} \bar{x}^{(l,q)} + \frac{\rho_d^{(l,q)} \bar{d}_x^{(l,q)} + \rho^{(l,q)} \lambda_{h_l} \bar{d}_x^{(l,q)}}{\xi^{(l,q)}} \left(1 - e^{-\xi_d^{(l,q)} t}\right) \quad (5.11)$$

$$\Lambda_l = \lambda_{h_l} + \lambda_{\gamma_l} \quad (5.12)$$

Here  $\bar{d}_x^{(l,q)}$  is the noise bound such that  $|d_x^{(l,q)}| \leq \bar{d}_x^{(l,q)}$  [57]. Moreover,  $\rho^{(l,q)}$ ,  $\xi^{(l,q)}$ ,  $\rho_d^{(l,q)}$  and  $\xi_d^{(l,q)}$  are positive constants, such that the following equations hold:

$$\left| e^{A_L^{(l,q)} t} \right| \leq \rho^{(l,q)} e^{-\xi^{(l,q)} t} \quad (5.13)$$

$$\left| e^{A_L^{(l,q)} t} L^{(l,q)} \right| \leq \rho_d^{(l,q)} e^{-\xi_d^{(l,q)} t} \quad (5.14)$$

For the formation of the algebraic thresholds, inclusion functions can be utilized [38]. Given that  $[x^{(l)}] = y_x^{(l,q)} + [d_x^{(l,q)}]$  since  $[x^{(l)}] = [y_x^{(l,q)} - \bar{d}_x^{(l,q)}, y_x^{(l,q)} + \bar{d}_x^{(l,q)}] = y_x^{(l,q)} + [d_x^{(l,q)}] \forall j \in \{1, \dots, n_l - r_l\}$  also  $[z^{(l)}] = y_z^{(l,q)} + [d_z^{(l,q)}]$ ,  $[\chi^{(l)}] = y_\chi^{(l,q)} + [d_\chi^{(l,q)}]$  and  $[u^{(l)}] = [\underline{u}^{(l)}, \bar{u}^{(l)}]$ , the following can be inferred as follows [44]:

$$\xi_j^{(l)} \{x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)}\} \in [\xi_j^{(l,q)}, \bar{\xi}_j^{(l,q)}] \quad \forall j \in \{1, \dots, n_l - r_l\} \quad (5.15)$$

Where  $[\xi_j^{(l,q)}, \bar{\xi}_j^{(l,q)}] = \xi_j^{(l)} \{y_x^{(l,q)} + [d_x^{(l,q)}], y_z^{(l,q)} + [d_z^{(l,q)}], y_\chi^{(l,q)} + [d_\chi^{(l,q)}], [u^{(l)}]\}$ . Then according to (5.7) and Assumption 5, the following stands [44]:

$$\begin{cases} \underline{\varepsilon}_{y_{z_j}}^{(l,q)} = \xi_j^{(l,q)} - \xi_j^{(l)} \{x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)}\} \\ \bar{\varepsilon}_{y_{z_j}}^{(l,q)} = \bar{\xi}_j^{(l,q)} - \xi_j^{(l)} \{x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)}\} \end{cases} \quad \forall j \in \{1, \dots, n_l - r_l\} \quad (5.16)$$

### 5.3.3. Multiple Fault Decision Logic

This subsection provides the fault decision logic. As noted at the beginning of the section, but also in section 2.7, isolation occurs in two steps; the local decision logic and the global decision logic [56]. This shall be shown in detail hereafter.

#### Local Decision Logic

If faults occur in  $\mathcal{S}^{(l,q)}$  they are detected by  $\mathcal{M}^{(l,q)}$  with the use of ARR. The  $j$ -th state-based ARR can be expressed as [44]:

$$\varepsilon_j^{(l,q)} : \left| \varepsilon_{y_{x_j}}^{(l,q)}(t) \right| - \bar{\varepsilon}_{y_{x_j}}^{(l,q)}(t) \leq 0 \quad \forall j \in \{1, \dots, n_l - r_l\} \quad (5.17)$$

Regarding the monitoring modules that make use of the algebraic residual expression  $\varepsilon_{y_z}^{(l,q)}$  that was given in (5.6) as well as the threshold expression of (5.16), the way to create the  $j$ -th ARR is provided [44]:

$$\varepsilon_j^{(l,q)} : \varepsilon_{y_{z_j}}^{(l,q)}(t) \in [\underline{\varepsilon}_{y_{z_j}}^{(l,q)}(t), \bar{\varepsilon}_{y_{z_j}}^{(l,q)}(t)] \quad \forall j \in \{1, \dots, r_l\} \quad (5.18)$$

The set of ARRs that the module decides on the existence of local faults is expressed as  $\varepsilon^{(l,q)} = \cup_{j \in \mathcal{J}^{(l,q)}} \varepsilon_j^{(l,q)}$ , where  $\mathcal{J}^{(l,q)}$  is an index set. The fault detection time  $T_{D_j}^{(l,q)}$  is denoted as the first time instant that the violation of (5.17) or (5.18) for at least one  $j \in \mathcal{J}^{(l,q)}$  occurs, by the local module  $\mathcal{M}^{(l,q)}$ , which can be expressed as  $T_{D_j}^{(l,q)} = \min\{t : \left| \varepsilon_{y_{x_j}}^{(l,q)}(t) \right| - \bar{\varepsilon}_{y_{x_j}}^{(l,q)}(t) > 0\}$  or  $T_{D_j}^{(l,q)} = \min\{\varepsilon_{y_{z_j}}^{(l,q)}(t) \notin [\underline{\varepsilon}_{y_{z_j}}^{(l,q)}(t), \bar{\varepsilon}_{y_{z_j}}^{(l,q)}(t)]\}$  respectively. Until this violation occurs, the local sensor set  $\mathcal{S}^{(l,q)}$  or system  $\Sigma^{(l)}$  is considered as non-faulty, meaning that no fault exists or that faults exist but remain undetected. A binary decision is considered as the output of  $\mathcal{M}^{(l,q)}$  expressed as  $D^{(l,q)}$  for the case of permanent fault occurrence as follows [44]:

$$D^{(l,q)}(t) = \begin{cases} 0, & t < T_D^{(l,q)} \\ 1, & t \geq T_D^{(l,q)} \end{cases} \quad (5.19)$$

Where  $T_D^{(l,q)} = \min\{T_{D_j}^{(l,q)} : j \in \mathcal{J}^{(l,q)}\}$ . As was shown in fig. 5.2 the decision of each module is fed to the aggregation module  $\mathcal{A}^{(l)}$  of the monitoring agent  $\mathcal{M}^{(l)}$  to acquire a binary decision vector  $D^{(l)} = [D^{(l,1)}, \dots, D^{(l,q)}]$  and compare it with a binary fault signature matrix  $F^{(l)}$ , consisting of  $n_l$  rows and  $N_{C_l} + 2$  columns where  $N_{C_l} = 2^{n_l} - 1 + n_p^{(l)}$  where  $n_p^{(l)}$  represents the columns associated with the considered process faults for each system  $\Sigma^{(l)}$ . The design of this matrix will be provided in chapter 6 for the plenitude of the analysis. Keep in mind that when  $D^{(l)}(t) = 0_{n_l}$ , the diagnosis set  $\mathcal{D}_s^{(l)}$  is empty. Furthermore, if  $D^{(l,q)}(t) = f_q^{(l)} \forall q \in \{1, \dots, n_l\}$ , then the observed pattern  $\mathcal{D}^{(l)}(t)$  is characterised as consistent with the theoretical pattern  $F_i^{(l)}$  and the diagnosis set is expressed as  $\mathcal{D}_s^{(l)}(t) = \{F_{ci}^{(l)} : i \in \mathcal{J}_D^{(l)}(t)\}$  where  $\mathcal{J}_D^{(l)}(t)$  is the consistency index set expressed as  $\mathcal{J}_D^{(l)}(t) = \{i : F_i^{(l)} = D^{(l)}(t), i \in \{1, \dots, N_{C_l}\}\}$ . Moreover, the agent  $\mathcal{M}^{(l)}, I \in \{1, \dots, N\}$  also forms a decision on the propagation of sensor or process faults from the interconnected systems, aside from the local diagnosis set  $\mathcal{D}_s^{(l)}$ . This decision is made by the aggregation module  $\mathcal{A}^{(l)}$  of the agent, shown in fig. 5.2, and is expressed as  $D_\chi^{(l)}(t)$  [44]:

$$D_\chi^{(l)}(t) = \begin{cases} 0, & \text{if } f_\chi^{(l)} \notin \mathcal{D}_s^{(l)}(t) \text{ and } f_p^{(l)} \notin \mathcal{D}_s^{(l)}(t) \\ 1, & \text{otherwise.} \end{cases} \quad (5.20)$$

Where  $f_p^{(l)} \in R^{n_l^*}, n_l^* \leq n_l$  stands for the total number of faults that are propagated from the agent  $\mathcal{M}^{(l)}$ , to its neighboring agents, and  $f_\chi^{(l)}$  is associated with the faults propagated to the agent from the neighboring agents, due to the exchange of sensor information [44].

In the case a process fault occurs in a particular system  $\Sigma^{(l)}$ , it is again attributed to the local module  $\mathcal{M}^{(l,q)}$  of the particular system to detect it, and a task of the monitoring agent  $\mathcal{M}^{(l)}$  to distinguish it as a process fault instead of a sensor fault, according to its local decision logic. A prerequisite for this to happen is the use of redundant sensors. More specifically, if the outputs of two or more sensors that are responsible to measure the same variable show a significant change that causes the residuals to violate the associated ARRs, then this fault is declared as a process fault. The same logic applies if one process fault affects more than one variable in one system. In this case, the sensors responsible to measure these different variables may detect the fault after the violation of (5.17) or (5.18) and if it is not associated with fault propagation, then it is assigned as a process fault. It is worth mentioning that the process faults that affect a system, cannot be detected by the monitoring agents of neighboring systems. This happens because the output measurements the agents use, that originate from the interconnection sensors, for the formation of residuals according to (5.6) and (5.7), cancel out with the interconnection variables  $\chi^{(l)}$  that are used in the formation of  $h^{(l)}(x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)})$  and  $\xi(x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)})$ , of the equations that describe each system. Therefore, the process faults cannot propagate from one agent to another, unlike the sensor faults.

### Global Decision Logic

The global decision logic establishes the isolation of the faults propagated via the interconnections between the monitoring agents. It is therefore the responsibility of a global agent  $\mathcal{G}$  to gather the decisions on the propagation of faults from the  $N$  local agents  $D_\chi(t) = [D_\chi^{(1)}(t), \dots, D_\chi^{(N)}(t)]$  and compare them with the columns of a global fault signature matrix  $F^\chi$  comprising of  $N$  rows and  $N_C = 2^p - 1 + n_p$  columns ( $p \leq \sum_{I=1}^N \{p_I\}, p_I$  is the length of  $f_\chi^{(l)}$ ) [44].  $n_p$  represents the total number of the considered process faults that affect the propulsion process.

The star (\*) is placed in  $F^\chi$  instead of 1 in case the sensor fault is propagated to the agent  $\mathcal{M}^{(l)}$ , from the other agents  $\mathcal{M}^{(j)}, j \in \{1, \dots, N\}, j \neq I$  as it can not be assured it will be picked up by the agent  $\mathcal{M}^{(l)}$ . If  $\mathcal{D}_\chi(t)$  is consistent with the  $k$ -th column of  $F^\chi(F_k^\chi)$ , meaning that  $\mathcal{D}_\chi(t) = F_k^\chi$ , the diagnosis set of propagated faults is denoted as  $\mathcal{D}_s^\chi(t) = \{F_{ck}^\chi : k \in \mathcal{J}_\chi(t)\}$ , where  $\mathcal{J}_\chi(t)$  is an index set expressed as  $\mathcal{J}_\chi(t) = \{k : F_{ik}^\chi = D_\chi^{(l)}(t), k \in \{1, \dots, N_C\} \forall I \in \{1, \dots, N\}\}$ . The non-empty local diagnosis

set  $\mathcal{D}_s^X(t)$  of  $\mathcal{M}^{(l)}$ , is updated by the set  $\mathcal{D}_s^X(t)$  by excluding the occurrence of  $f_\chi^{(l)}$  and its combinations, if  $f_\chi^{(l)} \notin \mathcal{D}_s^X(t)$ . Thus, the global diagnosis set occurs as follows [44].

$$\mathcal{D}_s^G(t) = \mathcal{D}_s^X(t) \bigcap \mathcal{D}_s(t) \quad (5.21)$$

Where

$$\mathcal{D}_s(t) = \bigcup_{I=1}^N \mathcal{D}_s^{(I)}(t) \quad (5.22)$$

## 5.4. Discussion

In this chapter, the FDI scheme for process and sensor faults was explained in a generic form. Furthermore, the local decision logic was provided to explain how the local diagnosis sets are derived and the global decision logic that can isolate the faults that propagated from one system to another, was also shown. In the following chapter, the FD simulation results will be given, as well as some performance indicators for the FDI methodology that was adopted and presented earlier.



## Fault Diagnosis Simulation Results

In this chapter, the answer to the final research question shall be provided. As was explained in the previous chapter, the FD scheme was implemented in the simulation and the sensor and process faults were initiated at certain times during the simulation, in the absence and presence of sensor noise. The simulation was carried out for a duration of 300sec. It was assumed that the sensor measurements are affected by uniformly distributed noise that is 3% of the maximum amplitude of the noiseless measurement of each sensor. All the faults are permanent and the sensor faults are abrupt and offset, while process faults can be either incipient or abrupt, offset or drift-like according to their fault equation that was presented in table 3.2.

After the initiation of these faults, their detection and isolation were achieved, as can be seen hereafter, and the discrimination between sensor and process faults was established. These results shall be shown and explained for each sensor and process fault in the following sections. Thereafter, some examples will be described, where fault isolation is conducted. Later, some performance indicators will be shown that highlight some important performance criteria of the FD scheme that was adopted, such as standard deviation of detection time and miss-detection of process faults.

### 6.1. Detection of Sensor Faults

The design parameters for monitoring modules  $\mathcal{M}^{(1,1)}$ ,  $\mathcal{M}^{(1,2)}$ ,  $\mathcal{M}^{(3,1)}$ ,  $\mathcal{M}^{(3,2)}$ ,  $\mathcal{M}^{(4,1)}$  and  $\mathcal{M}^{(4,2)}$ , are presented in table 6.1. These parameters are used for the derivation of the state-based adaptive thresholds as they were provided in (5.10) in the previous chapter. It is important to mention here, that the choice of the observer gains in table 6.1 was done in such a way, as to satisfy the trade-off between the reduction of noise amplification as much as possible and the convergence of the state estimation to its final value in a relatively short time frame. The remaining parameters were chosen thereafter provided that they satisfy inequalities (5.13) and (5.14) of the previous chapter.

At this point, it is a good time to mention how the 19 sensors used to monitor the propulsion process are divided into the following sensor sets, which were decided according to the relevant theory in [57]. These 19 sets can be seen in table 6.2 along with their index sets. Each monitoring module  $\mathcal{M}^{(l,q)}$  was designed to use information from the respective sensors  $\mathcal{S}^{(l)}\{j\}$ . Table 6.2 also depicts the interconnection sensors chosen in the sensor sets. Modules  $\mathcal{M}^{(1,1)}$ ,  $\mathcal{M}^{(1,2)}$ ,  $\mathcal{M}^{(3,1)}$ ,  $\mathcal{M}^{(3,2)}$ ,  $\mathcal{M}^{(4,1)}$  and  $\mathcal{M}^{(4,2)}$  incorporate state-based ARRs of the form expressed in (5.17) in the previous chapter, while the rest of the modules make use of algebraic-based ARRs as given in (5.18).

Table 6.1: Design parameters of observers in monitoring modules  $\mathcal{M}^{(1,1)}$ ,  $\mathcal{M}^{(1,2)}$ ,  $\mathcal{M}^{(3,1)}$ ,  $\mathcal{M}^{(3,2)}$ ,  $\mathcal{M}^{(4,1)}$  and  $\mathcal{M}^{(4,2)}$ .

Module $\mathcal{M}^{(l,q)}$	Gain $L^{(l,q)}$	Constant $\rho^{(l,q)}$	Constant $\xi^{(l,q)}$	Constant $\rho_d^{(l,q)}$	Constant $\xi_d^{(l,q)}$
$\mathcal{M}^{(1,1)}$ , $\mathcal{M}^{(1,2)}$	33.3833	2	100	40	100
$\mathcal{M}^{(3,1)}$ , $\mathcal{M}^{(3,2)}$	1.25	0.02	101.25	2.67	101.27
$\mathcal{M}^{(4,1)}$ , $\mathcal{M}^{(4,2)}$	0.3764	$1.67 \times 10^{-7}$	0.2	0.02	0.4

Table 6.2: Sensor sets and index sets used in the local monitoring agents.

Agent $\mathcal{M}^{(l)}$	Sensor Set $\mathcal{S}^{(l,q)}$	Sensors $\mathcal{S}^{(l)}\{j\} \forall j \in n_l$	Interconnection Sensors
$\mathcal{M}^{(1)}$	$\mathcal{S}^{(1,1)}$	$\mathcal{S}^{(1)}\{1\}$	-
	$\mathcal{S}^{(1,2)}$	$\mathcal{S}^{(1)}\{2\}$	-
$\mathcal{M}^{(2)}$	$\mathcal{S}^{(2,1)}$	$\mathcal{S}^{(2)}\{1\}$	$\mathcal{S}^{(1)}\{1\}, \mathcal{S}^{(4)}\{1\}, \mathcal{S}^{(4)}\{3\}$
	$\mathcal{S}^{(2,2)}$	$\mathcal{S}^{(2)}\{2\}$	$\mathcal{S}^{(1)}\{2\}, \mathcal{S}^{(4)}\{2\}, \mathcal{S}^{(4)}\{4\}$
	$\mathcal{S}^{(2,3)}$	$\mathcal{S}^{(2)}\{3\}$	$\mathcal{S}^{(1)}\{1\}, \mathcal{S}^{(4)}\{1\}, \mathcal{S}^{(4)}\{3\}$
	$\mathcal{S}^{(2,4)}$	$\mathcal{S}^{(2)}\{4\}$	$\mathcal{S}^{(1)}\{2\}, \mathcal{S}^{(4)}\{2\}, \mathcal{S}^{(4)}\{4\}$
	$\mathcal{S}^{(2,5)}$	$\mathcal{S}^{(2)}\{3\}, \mathcal{S}^{(2)}\{5\}$	$\mathcal{S}^{(1)}\{1\}, \mathcal{S}^{(4)}\{1\}, \mathcal{S}^{(4)}\{3\}$
	$\mathcal{S}^{(2,6)}$	$\mathcal{S}^{(2)}\{4\}, \mathcal{S}^{(2)}\{6\}$	$\mathcal{S}^{(1)}\{2\}, \mathcal{S}^{(4)}\{2\}, \mathcal{S}^{(4)}\{4\}$
$\mathcal{M}^{(3)}$	$\mathcal{S}^{(3,1)}$	$\mathcal{S}^{(3)}\{1\}, \mathcal{S}^{(3)}\{3\}$	$\mathcal{S}^{(1)}\{1\}, \mathcal{S}^{(4)}\{1\}, \mathcal{S}^{(4)}\{3\}$
	$\mathcal{S}^{(3,2)}$	$\mathcal{S}^{(3)}\{2\}, \mathcal{S}^{(3)}\{4\}$	$\mathcal{S}^{(1)}\{2\}, \mathcal{S}^{(4)}\{2\}, \mathcal{S}^{(4)}\{4\}$
	$\mathcal{S}^{(3,3)}$	$\mathcal{S}^{(3)}\{1\}, \mathcal{S}^{(3)}\{3\}$	$\mathcal{S}^{(1)}\{1\}, \mathcal{S}^{(2)}\{1\}, \mathcal{S}^{(2)}\{3\}, \mathcal{S}^{(4)}\{1\}, \mathcal{S}^{(4)}\{3\}$
	$\mathcal{S}^{(3,4)}$	$\mathcal{S}^{(3)}\{2\}, \mathcal{S}^{(3)}\{4\}$	$\mathcal{S}^{(1)}\{2\}, \mathcal{S}^{(2)}\{2\}, \mathcal{S}^{(2)}\{4\}, \mathcal{S}^{(4)}\{2\}, \mathcal{S}^{(4)}\{4\}$
	$\mathcal{S}^{(3,5)}$	$\mathcal{S}^{(3)}\{2\}, \mathcal{S}^{(3)}\{4\}, \mathcal{S}^{(3)}\{5\}$	$\mathcal{S}^{(4)}\{2\}$
$\mathcal{M}^{(4)}$	$\mathcal{S}^{(4,1)}$	$\mathcal{S}^{(4)}\{1\}, \mathcal{S}^{(4)}\{3\}$	$\mathcal{S}^{(1)}\{1\}, \mathcal{S}^{(3)}\{1\}, \mathcal{S}^{(3)}\{3\}$
	$\mathcal{S}^{(4,2)}$	$\mathcal{S}^{(4)}\{2\}, \mathcal{S}^{(4)}\{4\}$	$\mathcal{S}^{(1)}\{2\}, \mathcal{S}^{(3)}\{2\}, \mathcal{S}^{(3)}\{4\}$
	$\mathcal{S}^{(4,3)}$	$\mathcal{S}^{(4)}\{3\}$	-
	$\mathcal{S}^{(4,4)}$	$\mathcal{S}^{(4)}\{4\}$	-
	$\mathcal{S}^{(4,5)}$	$\mathcal{S}^{(4)}\{1\}, \mathcal{S}^{(4)}\{3\}, \mathcal{S}^{(4)}\{5\}$	$\mathcal{S}^{(1)}\{1\}, \mathcal{S}^{(3)}\{1\}, \mathcal{S}^{(3)}\{3\}, \mathcal{S}^{(3)}\{5\}$
	$\mathcal{S}^{(4,6)}$	$\mathcal{S}^{(4)}\{2\}, \mathcal{S}^{(4)}\{4\}, \mathcal{S}^{(4)}\{6\}$	$\mathcal{S}^{(1)}\{2\}, \mathcal{S}^{(3)}\{2\}, \mathcal{S}^{(3)}\{4\}, \mathcal{S}^{(3)}\{5\}$

The sensor sets were formulated in such a way that the isolability of sensor and process faults is enhanced. This shall be seen in the fault signature matrices 6.5-6.8 that will be presented later in this chapter, where almost all of the columns in these matrices are unique. This is achieved thanks to the use of different combinations of sensor outputs that are utilized in the monitoring modules. For instance, in table 6.2 one may notice that every first, third, and fifth monitoring module of an agent uses outputs from sensors with the index  $j$  being an odd number, while every second, fourth and sixth monitoring module uses outputs from sensors with the index  $j$  being an even number. Furthermore, for the sensors presented in table 6.2 the ones that have index  $j$  being an even number, are the redundant sensors. For even more clarity the redundant sensors have a brown font color, while the primary ones have a blue color in the table.

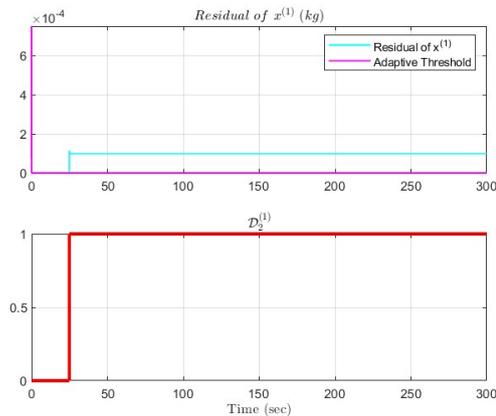


Figure 6.1: No sensor noise and sensor fault initiated at 25 seconds.

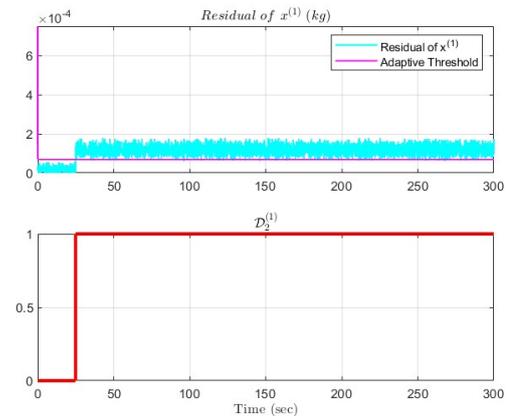


Figure 6.2: Sensor noise and sensor fault initiated at 25 seconds.

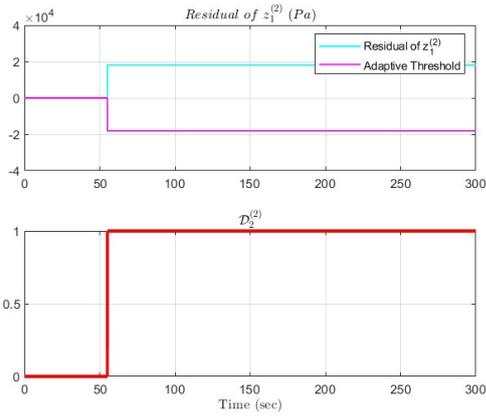


Figure 6.3: No sensor noise and sensor fault initiated at 55 seconds.

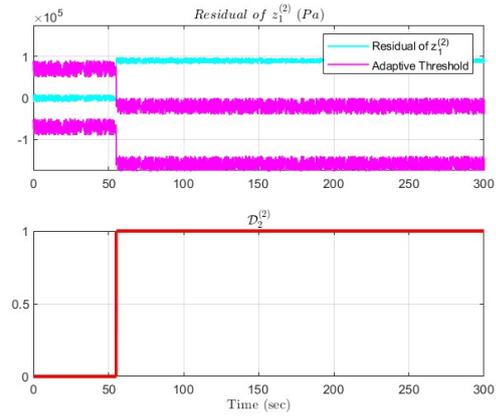


Figure 6.4: Sensor noise and sensor fault initiated at 55 seconds.

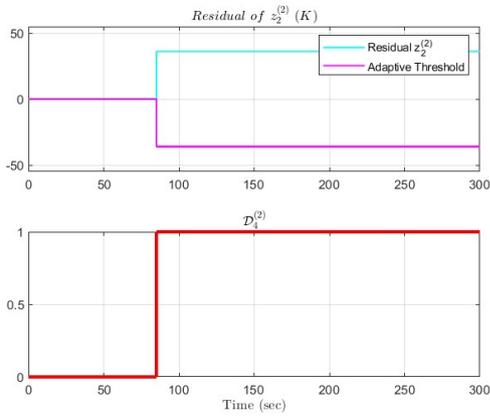


Figure 6.5: No sensor noise and sensor fault initiated at 85 seconds.

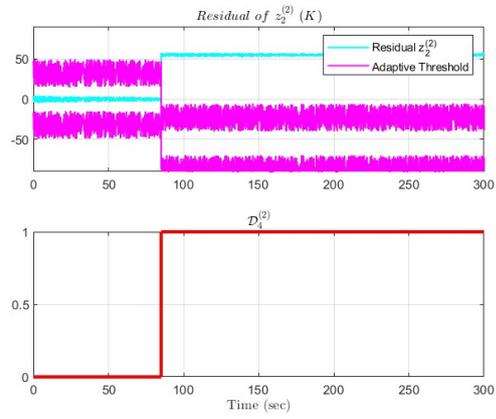


Figure 6.6: Sensor noise and sensor fault initiated at 85 seconds.

In figures 6.1 and 6.2 one can view the residual of  $x^{(1)}$  and fault detection time for the cases in absence and presence of sensor noise respectively. In the case where there is no sensor noise, the adaptive threshold approaches zero as time progresses and that's because the noise bound  $\bar{a}_x^{(l,q)}$  must be equal to zero since noise is absent. Consequently, the sensor fault amplitude can be set lower than 5% of the maximum measurement indicated by the sensor and it was detected almost immediately after it appears, as can be seen in fig. 6.1. In the case of sensor noise being present, the noise bound  $\bar{a}_x^{(l,q)}$  was set such that  $|a_x^{(l,q)}| \leq \bar{a}_x^{(l,q)}$ , to avoid any false alarms triggered by the detection of the sensor noise. The sensor fault amplitude was set equal to 6% of the maximum measurement indicated by the sensor, due to the noise, and it was detected almost immediately after it appears, as can be seen in fig. 6.2.

In figures 6.3 and 6.4 one may view the residual of  $z_1^{(2)}$  generated for the cases in the absence and presence of sensor noise respectively. The sensor fault amplitude in the case of sensor noise absence was set to 2% of the maximum measurement indicated by the sensor and it can be detected almost immediately after it appears, as can be seen in fig. 6.3. Regarding fig. 6.4 and its associated residual, the sensor fault amplitude was set to 5% of the maximum measurement indicated by the sensor and it was detected almost immediately after it appears, as can be seen in fig. 6.4 at the bottom of the figure.

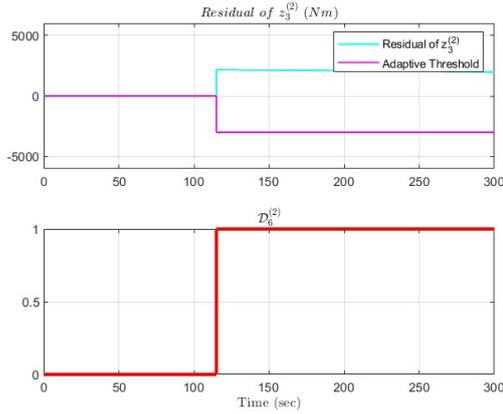


Figure 6.7: No sensor noise and sensor fault initiated at 115 seconds.

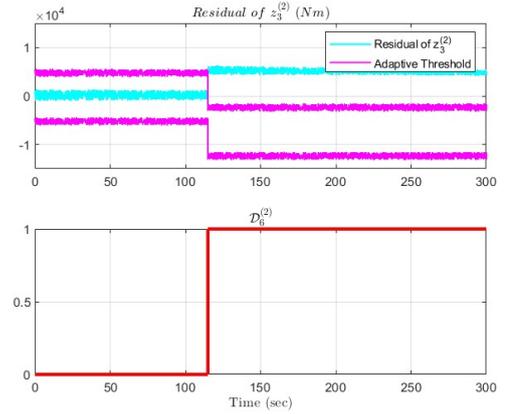


Figure 6.8: Sensor noise and sensor fault initiated at 115 seconds.

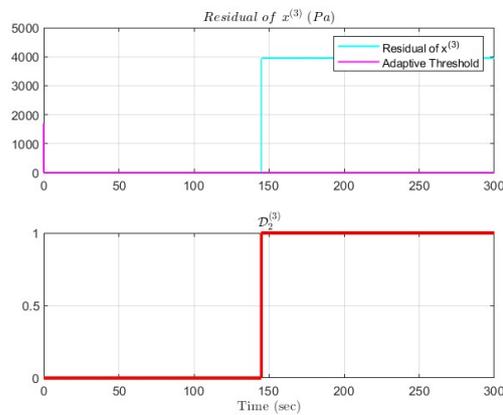


Figure 6.9: No sensor noise and sensor fault initiated at 145 seconds.

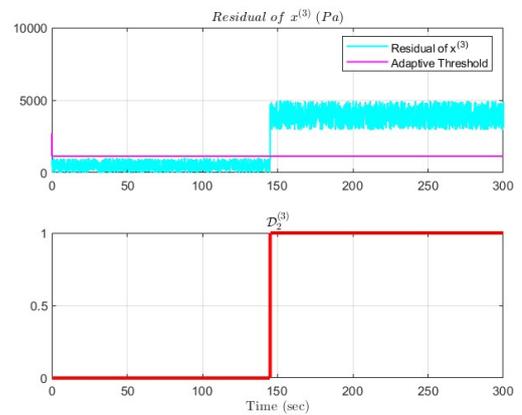


Figure 6.10: Sensor noise and sensor fault initiated at 145 seconds.

In figures 6.5 and 6.6 the residual of  $z_2^{(2)}$  generated for the cases in the absence and presence of sensor noise respectively can be viewed. The sensor fault amplitude in the case of sensor noise absence was set to 4% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.5. When looking at fig. 6.6, at first one can notice that the noise affects the thresholds as well as the residual due to the use of sensor outputs for the formation of thresholds. The sensor fault amplitude was set to 5.5% of the maximum measurement indicated by the sensor and was detected instantly after it appears.

In figures 6.7 and 6.8 one can see the residual of  $z_3^{(2)}$  generated for the cases in absence and presence of sensor noise respectively. The sensor fault amplitude in the case of sensor noise absence is set to 5% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.7. When looking at fig. 6.8 and its associated residual, the sensor fault amplitude was set to 10% of the maximum measurement indicated by the sensor and it was detected almost immediately after it appears, as can be seen in fig. 6.8 at the lower part of this figure.

In figures 6.9 and 6.10 one can view the residual of  $x^{(3)}$  and fault detection time for the cases in absence and presence of sensor noise respectively. In the case where there is no sensor noise, the adaptive threshold approaches zero as time progresses and that's because the noise bound  $\bar{d}_x^{(j,q)}$  must be equal to zero since sensor noise is absent. Consequently, the sensor fault amplitude can be set to 4% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.9.

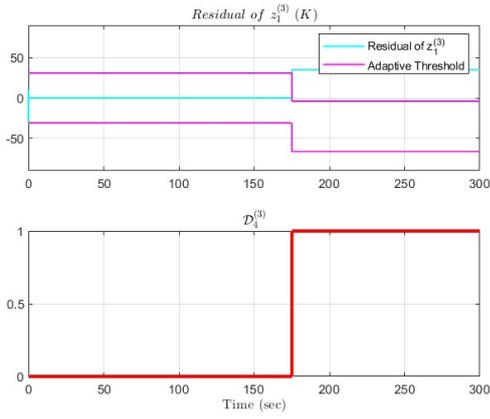


Figure 6.11: No sensor noise and sensor fault initiated at 175 seconds.

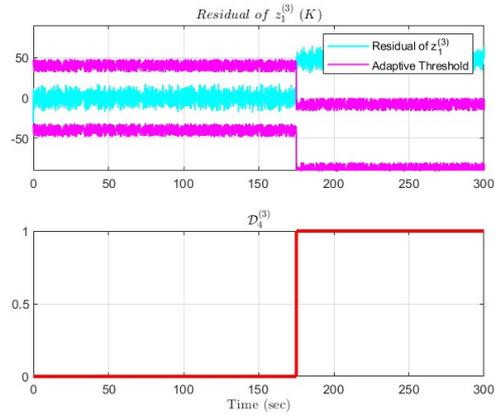


Figure 6.12: Sensor noise and sensor fault initiated at 175 seconds.

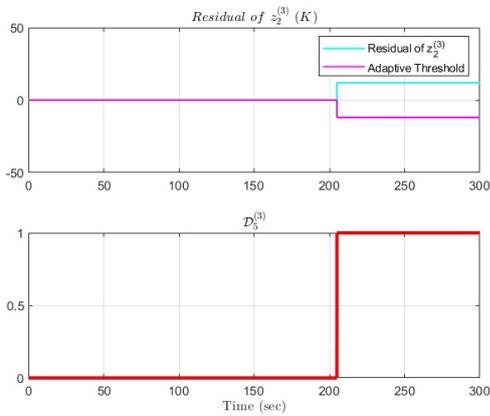


Figure 6.13: No sensor noise and sensor fault initiated at 205 seconds.

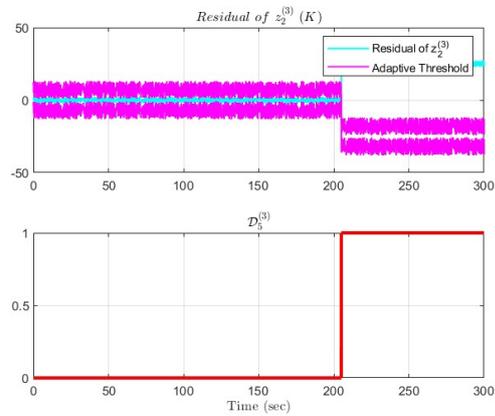


Figure 6.14: Sensor noise and sensor fault initiated at 205 seconds.

In the case of sensor noise being present, the noise bound  $\bar{d}_x^{(l,q)}$  was set such that  $|d_x^{(l,q)}| \leq \bar{d}_x^{(l,q)}$ , to avoid any false alarms triggered by the detection of the sensor noise. The sensor fault amplitude was set equal to 5% of the maximum measurement indicated by the sensor, due to the noise, and was detected almost immediately after it appears, as can be seen in fig. 6.10.

In figures 6.11 and 6.12 one can observe the residual of  $z_1^{(3)}$  generated for the cases in absence and presence of sensor noise respectively. The sensor fault amplitude in the case of sensor noise absence is set to 5% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.11. Regarding fig. 6.12, and the associated residual, the sensor fault amplitude was set to 7.5% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.12.

In figures 6.13 and 6.14 one can observe the residual of  $z_2^{(3)}$  generated for the cases in absence and presence of sensor noise respectively. The sensor fault amplitude in the case of sensor noise absence is set to 2.5% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.13. When looking at fig. 6.14 and the associated residual, the sensor fault amplitude was set to 5% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.14 at the bottom of the figure.

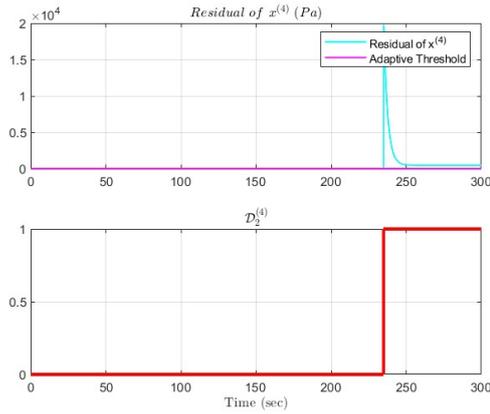


Figure 6.15: No sensor noise and sensor fault initiated at 235 seconds.

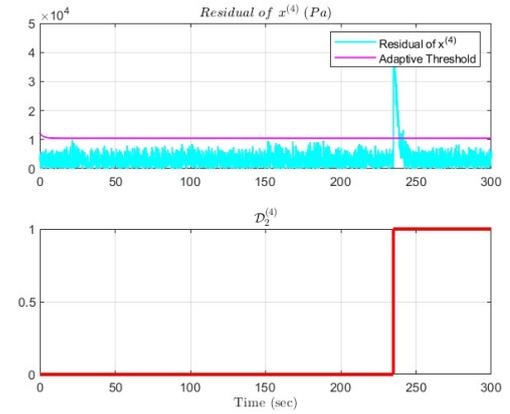


Figure 6.16: Sensor noise and sensor fault initiated at 235 seconds.

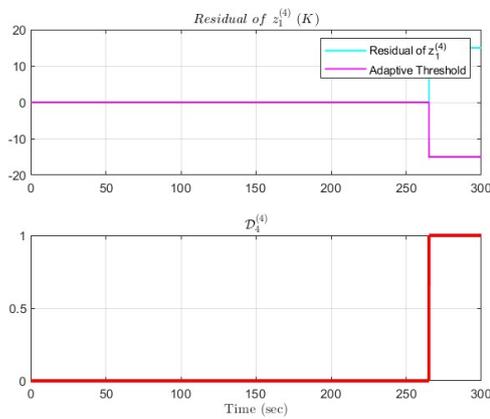


Figure 6.17: No sensor noise and sensor fault initiated at 265 seconds.

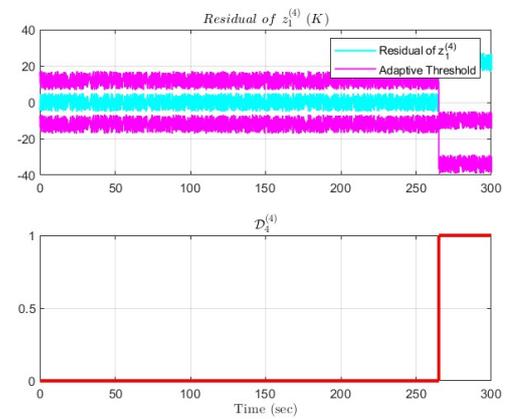


Figure 6.18: Sensor noise and sensor fault initiated at 265 seconds.

In figures 6.15 and 6.16 one can view the residual of  $x^{(4)}$  and fault detection time for the cases in absence and presence of sensor noise respectively. In the case where there is no sensor noise, the adaptive threshold approaches zero as time progresses and that's because the noise bound  $\bar{d}_x^{(l,q)}$  must be equal to zero since noise is absent. Consequently, the sensor fault amplitude can be set to 5% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.15. In the case of sensor noise being present, the noise bound  $\bar{d}_x^{(l,q)}$  was set such that  $|d_x^{(l,q)}| \leq \bar{d}_x^{(l,q)}$ , to avoid any false alarms triggered by the detection of the sensor noise. The sensor fault amplitude was set equal to 10% of the maximum measurement indicated by the sensor, due to the noise, and was detected almost immediately after it appears, as can be seen in fig. 6.16.

In figures 6.17 and 6.18 one can observe the residual of  $z_1^{(4)}$  generated for the cases in absence and presence of sensor noise respectively. The sensor fault amplitude in the case of sensor noise absence is set to 4.5% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.17 at the bottom. Regarding fig. 6.18 and the associated residual, the sensor fault amplitude was set to 7% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.18 at the bottom of the figure.

Table 6.3: Sensors with their noise range and fault magnitudes used.

Sensor $\mathcal{S}^{(l)}\{j\}$	Measurement	Noise Range	Sensor Fault Magnitude $\phi_j^{(l)}$	Units
$\mathcal{S}^{(1)}\{1\}$	Fuel Injection	$[-4.5x10^{-5}, 4.5x10^{-5}]$	$\phi_1^{(1)} = 1.8x10^{-4}$	$kg$
$\mathcal{S}^{(1)}\{2\}$	Fuel Injection	$[-4.5x10^{-5}, 4.5x10^{-5}]$	$\phi_2^{(1)} = 1.8x10^{-4}$	$kg$
$\mathcal{S}^{(2)}\{1\}$	Pressure	$[-2x10^4, 2x10^4]$	$\phi_1^{(2)} = 9x10^4$	$Pa$
$\mathcal{S}^{(2)}\{2\}$	Pressure	$[-2x10^4, 2x10^4]$	$\phi_2^{(2)} = 9x10^4$	$Pa$
$\mathcal{S}^{(2)}\{3\}$	Temperature	$[-17, 17]$	$\phi_3^{(2)} = 55$	$K$
$\mathcal{S}^{(2)}\{4\}$	Temperature	$[-17, 17]$	$\phi_4^{(2)} = 55$	$K$
$\mathcal{S}^{(2)}\{5\}$	Torque	$[-0.75x10^3, 0.75x10^3]$	$\phi_5^{(2)} = 7.125x10^3$	$Nm$
$\mathcal{S}^{(2)}\{6\}$	Torque	$[-0.75x10^3, 0.75x10^3]$	$\phi_6^{(2)} = 7.125x10^3$	$Nm$
$\mathcal{S}^{(3)}\{1\}$	Pressure	$[-0.99x10^3, 0.99x10^3]$	$\phi_1^{(3)} = 5x10^3$	$Pa$
$\mathcal{S}^{(3)}\{2\}$	Pressure	$[-0.99x10^3, 0.99x10^3]$	$\phi_2^{(3)} = 5x10^3$	$Pa$
$\mathcal{S}^{(3)}\{3\}$	Temperature	$[-8, 8]$	$\phi_3^{(3)} = 48$	$K$
$\mathcal{S}^{(3)}\{4\}$	Temperature	$[-8, 8]$	$\phi_4^{(3)} = 48$	$K$
$\mathcal{S}^{(3)}\{5\}$	Temperature	$[-6, 6]$	$\phi_5^{(3)} = 25$	$K$
$\mathcal{S}^{(4)}\{1\}$	Pressure	$[-5.5x10^3, 5.5x10^3]$	$\phi_1^{(4)} = 5x10^4$	$Pa$
$\mathcal{S}^{(4)}\{2\}$	Pressure	$[-5.5x10^3, 5.5x10^3]$	$\phi_2^{(4)} = 5x10^4$	$Pa$
$\mathcal{S}^{(4)}\{3\}$	Temperature	$[-5, 5]$	$\phi_3^{(4)} = 22$	$K$
$\mathcal{S}^{(4)}\{4\}$	Temperature	$[-5, 5]$	$\phi_4^{(4)} = 22$	$K$
$\mathcal{S}^{(4)}\{5\}$	Temperature	$[-5, 5]$	$\phi_5^{(4)} = 25$	$K$
$\mathcal{S}^{(4)}\{6\}$	Temperature	$[-5, 5]$	$\phi_6^{(4)} = 25$	$K$

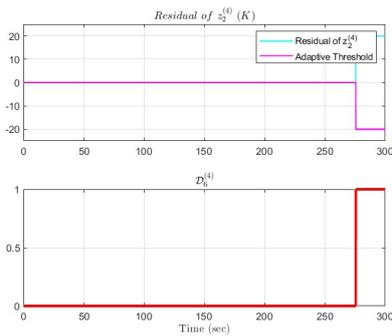


Figure 6.19: No sensor noise and sensor fault initiated at 275 seconds.

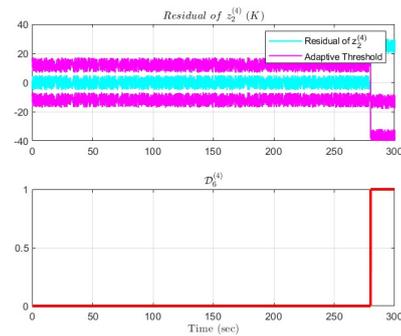


Figure 6.20: Sensor noise and sensor fault initiated at 280 seconds.

In figures 6.19 and 6.20 one can observe the residual of  $z_2^{(4)}$  generated for the cases in absence and presence of sensor noise respectively. The sensor fault amplitude in the case of sensor noise absence is set to 5% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.19. Regarding fig. 6.20 and the associated residual, the sensor fault amplitude was set to 8% of the maximum measurement indicated by the sensor and was detected almost immediately after it appears, as can be seen in fig. 6.20 at the bottom of the figure.

To provide the reader with a quantitative understanding of the aforementioned sensor fault amplitudes, which were given in a percentage form earlier, table 6.3 was created. The table shows the type of measurement each sensor takes. Furthermore, this table depicts the noise range and sensor fault magnitude for each sensor that was used. The noise range and fault magnitudes were determined according to the maximum measurement amplitude that each sensor provided. As it was mentioned earlier, the assumption was made that these noise ranges correspond to 3% of the maximum noiseless measurement of each sensor. As a result, the sensor fault magnitudes are more or less around 5% to 10% of the maximum sensor measurement of the sensor they are attributed. These magnitudes were set in this way to make sure that they violate the adaptive threshold of their sensors if they occurred.

## 6.2. Detection of Process Faults

For the process faults, the FDI logic that was presented in the previous chapter is sufficient to distinguish them from the sensor faults. The following figures illustrate the evolution of the process faults until the point that they violate the thresholds that were set to constrain the residuals, and their detection as a result of this violation. Most process faults were detected almost immediately, while a few are detected at a later stage of the simulation. The following figures cover the cases in which sensor noise is either absent or present, just to see how the fault magnitude, the thresholds, and the detection time is different in each of these two cases.

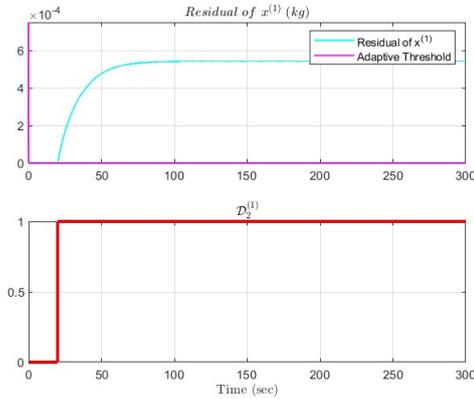


Figure 6.21: No sensor noise in the presence of fuel injection valve process fault.

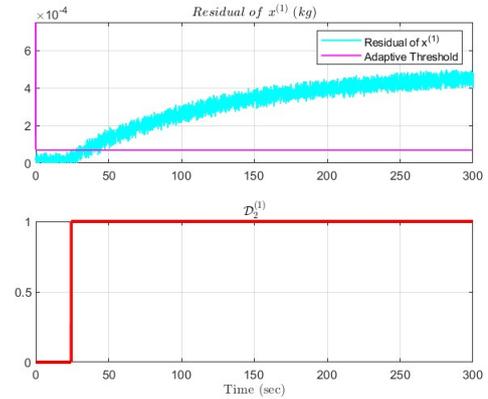


Figure 6.22: Fuel injection valve process fault in the presence of sensor noise.

In figures, 6.21 and 6.22 one may observe the effect of a simulated process fault regarding the process fault  $f_{fu}^{(1)}$  of the fuel injection valve of the fuel pump in  $\Sigma^{(1)}$ . In the presence of sensor noise as can be seen in fig. 6.22, the fault was detected almost 10 seconds later of its initiation, as a result of the higher adaptive threshold imposed due to sensor noise. Furthermore, the incipient nature of a seizing injection valve is observed, which captures the choking effect of the fuel that is fed into the engine cylinder. In both cases, the fault was detected successfully and false alarms were avoided.

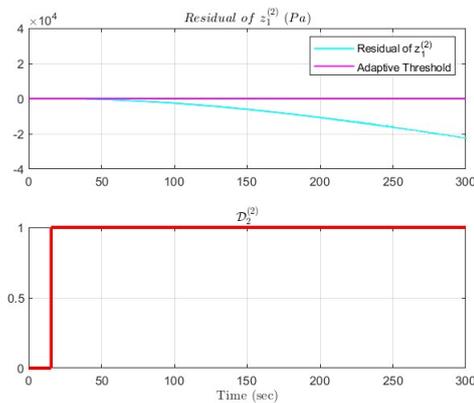


Figure 6.23: No sensor noise in the presence of cylinder damage process fault.

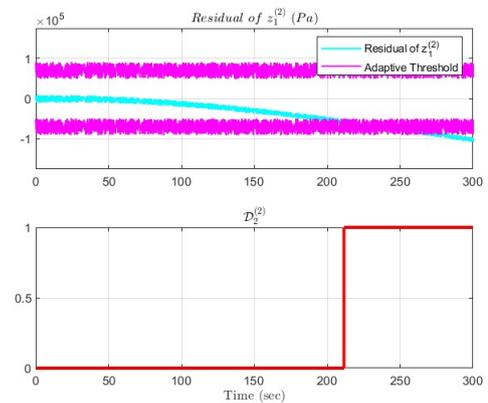


Figure 6.24: Cylinder damage process fault in the presence of sensor noise.

In figures 6.23 and 6.24, the simulation of the process fault  $f_{z,V_1}^{(2)}$  of the cylinder bore damage is presented in the absence and presence of sensor noise respectively. This process fault is incipient in nature, as it deteriorates as time progresses, and it affects all the sensors of  $\Sigma^{(2)}$ , some earlier than others. The fault was initiated at 15sec in both cases and it was detected at a far later stage as can be seen in the case where the sensor noise is present. More specifically, in fig. 6.23 it was detected almost immediately after its activation, due to the value of the thresholds being equal to zero, while in

fig. 6.24 the fault is detected at a far later stage due to the higher threshold values. Overall, in both cases, the fault was detected successfully.

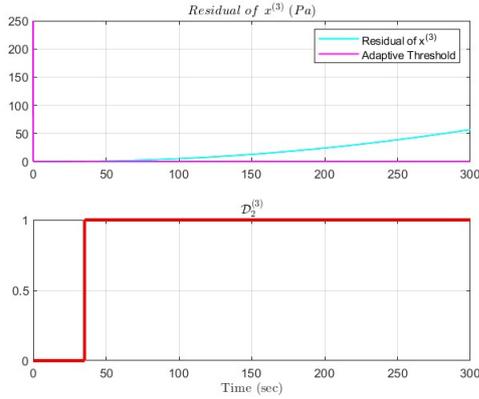


Figure 6.25: No sensor noise in the presence of process fault regarding seal leakage inside the turbine.

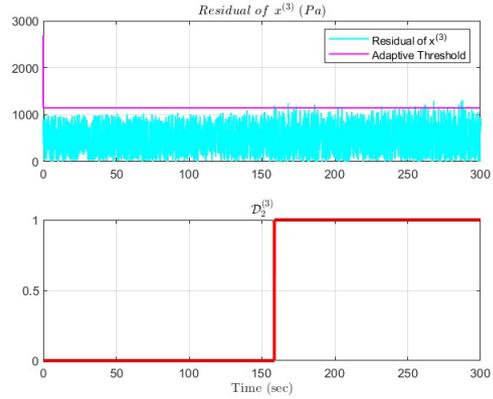


Figure 6.26: Turbine seal leakage process fault in the presence of sensor noise.

In figures 6.25 and 6.26 the simulated process fault  $f_{x,tur}^{(3)}$  of a cracked seal of the turbocharger turbine is shown. This process fault is again incipient and is a function of the turbine pressure, in the sense that, as the turbine pressure increases, the effect of this fault is also increasing exponentially. The fault is initiated at 15 seconds in both cases and it was detected almost immediately in fig. 6.25, while in the presence of sensor noise it takes longer to detect. In both cases, this process fault was detected successfully and in a relatively short time after it occurred.

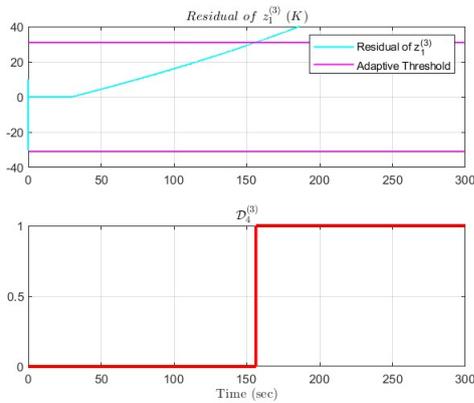


Figure 6.27: No sensor noise in the presence of process fault regarding loss of lubrication of the turbine shaft.

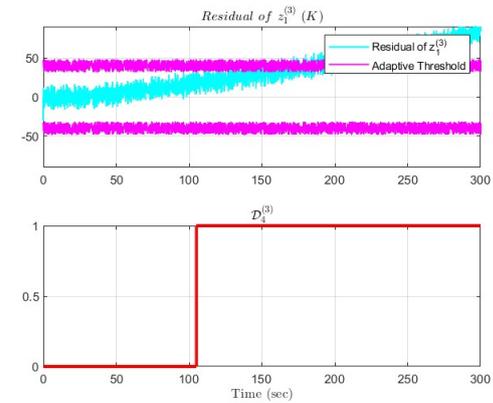


Figure 6.28: Loss of lubrication of the turbine shaft process fault in the presence of sensor noise.

In figures 6.27 and 6.28 the simulated process fault  $f_{z,T_{sl}}^{(3)}$  of insufficient lubrication of the turbocharger turbine impeller shaft is witnessed. This process fault causes the temperature of the air slip  $T_{sl}$  to rise as a result of the increased friction of the shaft with the bearings holding it. This fault is incipient and the temperature increases at a constant rate as time progresses due to the constant rotational speed of the turbine shaft. Both in the absence and presence of sensor noise, detection of the fault happens around the middle of the simulation time, as can be seen in both these figures, although the fault was initiated at 30 seconds in both cases. More specifically, in fig. 6.28 the fault is detected earlier than in fig. 6.27, due to the effect of the sensor noise on the thresholds that can be witnessed. In both cases, the fault was detected successfully, with no false alarms.

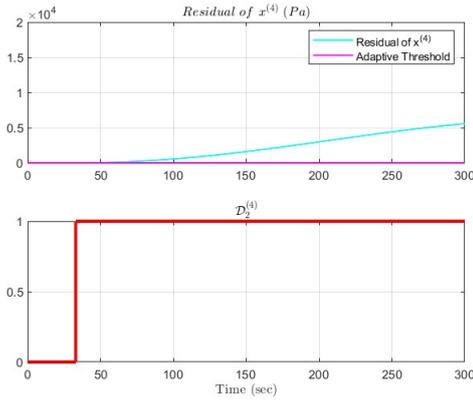


Figure 6.29: No sensor noise in the presence of process fault regarding seal leakage inside the compressor.

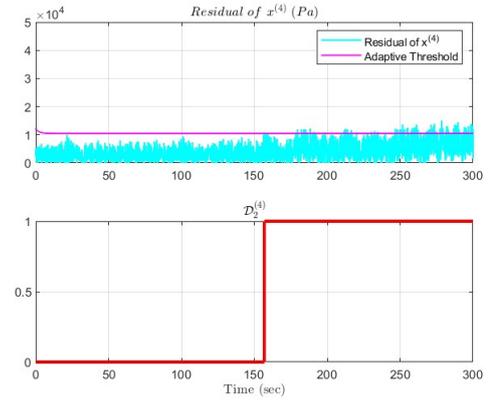


Figure 6.30: Compressor seal leakage process fault in the presence of sensor noise.

In figures 6.29 and 6.30 the simulated process fault  $f_{x,int}^{(4)}$  of a cracked seal of the turbocharger's compressor is presented. This fault is a function of the compressor pressure and increases exponentially as the pressure increases and time progresses. The fault is initiated at 30 seconds in both cases. In fig. 6.29, the absence of sensor noise allows the adaptive threshold to approach zero over time. This is the reason why the fault is detected a very short time after its activation. In fig. 6.30, the sensor noise presence leads to a higher adaptive threshold, to avoid false alarms, and as a result, the fault detection happens at a far later stage.

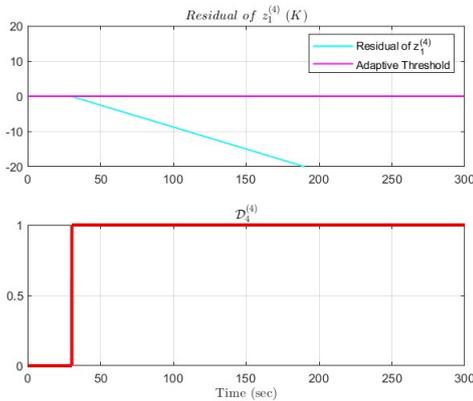


Figure 6.31: No sensor noise in the presence of process fault regarding loss of lubrication of the compressor shaft.

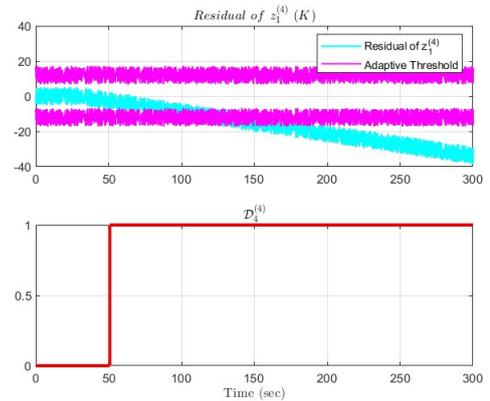


Figure 6.32: Loss of lubrication of the compressor shaft process fault in the presence of sensor noise.

Figures 6.31 and 6.32 illustrate the simulated process fault  $f_{z,T_{inl}}^{(4)}$  of insufficient lubrication of the air side impeller shaft of the turbocharger. The friction caused between the shaft and the bearings that hold it, causes a constant increase of the temperature  $T_{inl}$  of the air that is being compressed. This fault is incipient and the temperature increases at a constant rate as time progresses due to the constant rotational speed of the compressor's shaft. The fault was activated at 30 seconds and was detected shortly after that, both in the absence and presence of sensor noise, as can be seen in fig. 6.31 and 6.32 respectively. In both cases, the fault was detected successfully with no false alarms.

Figures 6.33 and 6.34 present the simulated process fault  $f_{z,T_c}^{(4)}$  of a blockage of the heat exchanger of the intercooler, causing the exponential temperature increase of the air exiting the intercooler  $T_c$ , due to insufficient cooling. This process fault was initiated at 30 seconds and it was detected about 15 seconds after its initialization in the presence of sensor noise. Overall, the fault was detected successfully in both cases, and no false alarms were triggered.

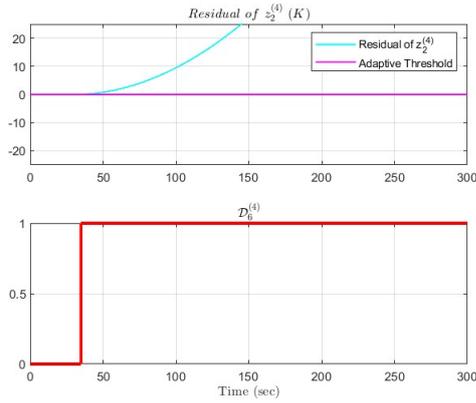


Figure 6.33: No sensor noise in the presence of process fault regarding the blockage of the heat exchanger of the inter-cooler.

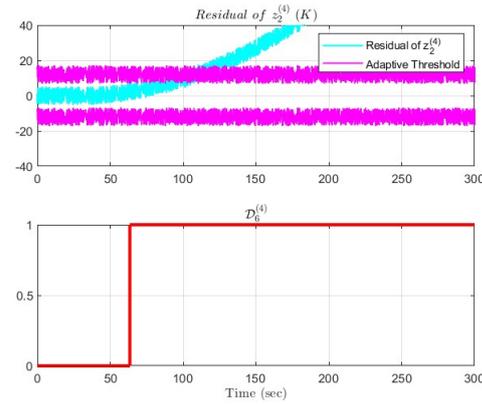


Figure 6.34: Blockage of the heat exchanger of the inter-cooler in the presence of sensor noise.

Table 6.4: Design parameters that were used for the modeling of the process faults.

Fault	Initialization Time $t_1$ (sec)	Evolution $\kappa$	Fault Function $\phi$	Units of $\phi$
$f_{x,tur}^{(3)}$	15	$\kappa_{x,tur}^{(3)} = 2 \times 10^{-5}$	$R_{x,tur}^{(3)} = 3.5 \times 10^{-4}$	Hz
$f_{z,T_{sl}}^{(3)}$	30	$\kappa_{z,T_{sl}}^{(3)} = 1 \times 10^3$	$R_{z,T_{sl}}^{(3)} = 1.25$	K/s
$f_{x,int}^{(4)}$	30	$\kappa_{x,int}^{(4)} = 1 \times 10^{-4}$	$R_{x,int}^{(4)} = 5 \times 10^{-2}$	Hz
$f_{z,T_c}^{(4)}$	30	$\kappa_{z,T_c}^{(4)} = 1 \times 10^{-3}$	$R_{z,T_c}^{(4)} = 2$	K/s
$f_{z,T_{inl}}^{(4)}$	30	$\kappa_{z,T_{inl}}^{(4)} = 1 \times 10^3$	$R_{z,T_{inl}}^{(4)} = 2.5$	K/s
$f_{z,V_1}^{(2)}$	15	$\kappa_{z,V_1}^{(2)} = 1 \times 10^{-5}$	$R_{z,V_1}^{(2)} = 3 \times 10^{-3}$	$m^3/s$
$f_u^{(1)}$	20	$\kappa_u^{(1)} = 1 \times 10^{-2}$	$\phi_u^{(1)} = 25$	%

The design parameters that were used to model the process faults that affect the marine propulsion process, in the presence of sensor noise, can be seen in table 6.4. These parameters were chosen empirically by witnessing the behavior of the faults and whether this behavior matched the desired characteristics of the fault. Later in this chapter, the setting of lower design parameters, which reproduce fault profiles that are on the limit of being detected by the monitoring modules, shall be shown. The parameters in table 6.4 can be used as a benchmark by future researchers that want to build more accurate process fault models or create new process faults with similar characteristics. Regarding the detection of process faults in the presence of sensor noise, in general, the faults are detected at a far later stage than the time of their occurrence. It can be argued that the time duration between fault occurrence and fault detection is large enough for a failure to occur within a system.

### 6.3. Fault Signature Matrices

The fault signature matrices of each monitoring agent  $\mathcal{M}^{(l)}$  and the one regarding the global monitoring agent  $\mathcal{G}$ , will be provided hereafter. In tables 6.5, 6.6, 6.7 and 6.8 the local process and sensor fault signature matrices for monitoring agents  $\mathcal{M}^{(1)}$ ,  $\mathcal{M}^{(2)}$ ,  $\mathcal{M}^{(3)}$  and  $\mathcal{M}^{(4)}$  are given in the presence of sensor noise. Table 6.9 represents the global decision matrix. These matrices are a result of the adopted multiple fault decision logic that was shown in subsection 5.3.3 of the previous chapter. More specifically they involve the isolation of the sensor and process faults according to the aforementioned logic. In the fault signature matrices provided below, only single fault signatures are shown, due to space limitations mainly.

When reviewing the fault matrices 6.5, 6.6, 6.7, and 6.8, the left part of the tables represents the local fault signatures for the sensor faults, the right part of the tables represents the fault signatures for the process fault affecting each respective system, and the rest of the columns indicate the propagated sensor fault signatures from the neighboring agents. As it was mentioned in the previous chapter,

Table 6.5: Process and sensor fault signature matrix of  $\mathcal{M}^{(1)}$  (0,1).

	$f_1^{(1)}$	$f_2^{(1)}$	$f_u^{(1)}$
$\varepsilon^{(1,1)}$	1	0	1
$\varepsilon^{(1,2)}$	0	1	1

Table 6.6: Part of process and sensor fault signature matrix of  $\mathcal{M}^{(2)}$  (\*,0,1).

	$f_1^{(2)}$	$f_2^{(2)}$	$f_3^{(2)}$	$f_4^{(2)}$	$f_5^{(2)}$	$f_6^{(2)}$	$f_1^{(1)}$	$f_2^{(1)}$	$f_1^{(4)}$	$f_2^{(4)}$	$f_3^{(4)}$	$f_4^{(4)}$	$f_{z,V_1}^{(2)}$
$\varepsilon^{(2,1)}$	1	0	0	0	0	0	1	0	1	0	0	0	1
$\varepsilon^{(2,2)}$	0	1	0	0	0	0	0	1	0	1	0	0	1
$\varepsilon^{(2,3)}$	0	0	1	0	0	0	1	0	1	0	1	0	1
$\varepsilon^{(2,4)}$	0	0	0	1	0	0	0	1	0	1	0	1	1
$\varepsilon^{(2,5)}$	0	0	*	0	1	0	*	0	*	0	*	0	*
$\varepsilon^{(2,6)}$	0	0	0	*	0	1	0	*	0	*	0	*	*

the process faults that affect a system, cannot be detected by the monitoring agents of neighboring systems. This happens due to the output measurements the agents use, originating for the interconnection sensors, cancel out with the interconnection variables  $\chi^{(l)}$  that are used for the formation of  $h^{(l)}(x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)})$  and  $\xi(x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)})$  in the system equations. Therefore the process faults can be isolated locally, as will be witnessed in the multiple fault isolation examples later. Regarding the rows of the tables, each row is associated with a local monitoring module  $\mathcal{M}^{(l,q)}$  that uses the ARRs expressed with  $\varepsilon^{(l,q)}$ . In these tables, a star (\*) is used instead of 1 to differentiate the sensitivity of the different ARRs in case of local and propagated sensor faults and in case of algebraic and state-based ARRs.

Table 6.5, represents the fault signature matrix for monitoring agent  $\mathcal{M}^{(1)}$ .  $\Sigma^{(1)}$  is dependent on the control input  $u^{(1)}$  and does not utilize sensor outputs from neighboring monitoring agents. Therefore, no faults from the other monitoring agents propagate to the agent of the first system. The last column of table 6.5 indicates that the process fault regarding fuel injection valve seizing is detected by both monitoring modules in of the agent  $\mathcal{M}^{(1)}$ . Table 6.6 represents the fault signature matrix for monitoring agent  $\mathcal{M}^{(2)}$ . Table 6.6 shows that almost all the ARRs in the table are affected by the propagation of faults from the other monitoring agents. In the last column of this table, one may notice that the process fault  $f_{z,V_1}^{(2)}$  regarding bore damage, is detected by all the modules of  $\mathcal{M}^{(2)}$ . Table 6.7 represents the fault signature matrix for monitoring agent  $\mathcal{M}^{(3)}$ . This matrix indicates that most ARRs of  $\mathcal{M}^{(3)}$  are affected by the propagation of faults from the other monitoring agents, as well as from the propagation of faults between the sensors of  $\mathcal{M}^{(3)}$ . In the last columns of this matrix, one may notice that the process faults  $f_{x,tur}^{(3)}$  and  $f_{z,sl}^{(3)}$  are detected by modules  $\mathcal{M}^{(3,1)}$ ,  $\mathcal{M}^{(3,2)}$  and  $\mathcal{M}^{(3,3)}$ ,  $\mathcal{M}^{(3,4)}$  respectively. Table 6.8 represents the fault signature matrix for monitoring agent  $\mathcal{M}^{(4)}$ . In table 6.8 no faults propagate within  $\mathcal{M}^{(4)}$ . In contrast, we see that sensor faults propagate from  $\mathcal{M}^{(3)}$ , to the first and second sensor of  $\mathcal{M}^{(4)}$ . In this matrix, the last three columns show the effect of the process faults  $f_{x,int}^{(4)}$ ,  $f_{z,int}^{(4)}$  and  $f_{z,T_c}^{(4)}$ . More specifically, the occurrence of  $f_{x,int}^{(4)}$  is detected by  $\mathcal{M}^{(4,1)}$ ,  $\mathcal{M}^{(4,2)}$ , the occurrence of  $f_{z,int}^{(4)}$  is detected by  $\mathcal{M}^{(4,3)}$ ,  $\mathcal{M}^{(4,4)}$  and the occurrence of  $f_{z,T_c}^{(4)}$  is detected by  $\mathcal{M}^{(4,5)}$ ,  $\mathcal{M}^{(4,6)}$ . Table 6.9 is the global fault signature matrix. More specifically, in this matrix one may notice that the sensors faults from  $\Sigma^{(1)}$  may propagate toward all the other monitoring agents. In contrast, the faults from  $\Sigma^{(2)}$  may propagate to the monitoring agent  $\mathcal{M}^{(3)}$ . Sensor faults occurring in  $\Sigma^{(3)}$  may propagate to monitoring agent  $\mathcal{M}^{(4)}$ , while sensor faults occurring  $\Sigma^{(4)}$  may propagate towards the monitoring agents of  $\Sigma^{(2)}$  and  $\Sigma^{(3)}$ , except the last two sensors of  $\Sigma^{(4)}$ , that any other monitoring agent does not utilize their outputs of these two sensors. The last seven columns of table 6.9 belong to the process faults that might appear in the propulsion process. For reasons explained earlier, one may notice that these faults do not propagate from one system to another. Therefore, their isolation is a task attributed to the local monitoring agents. This shall be witnessed in the multiple fault isolation examples that will follow in the next section.

Table 6.7: Part of process and sensor fault signature matrix of  $\mathcal{M}^{(3)}$  (\*,0,1).

	$f_1^{(3)}$	$f_2^{(3)}$	$f_3^{(3)}$	$f_4^{(3)}$	$f_5^{(3)}$	$f_1^{(1)}$	$f_2^{(1)}$	$f_1^{(2)}$	$f_2^{(2)}$	$f_3^{(2)}$	$f_4^{(2)}$	$f_1^{(4)}$	$f_2^{(4)}$	$f_3^{(4)}$	$f_4^{(4)}$	$f_{x,tur}^{(3)}$	$f_{z,Tsl}^{(3)}$
$\mathcal{E}^{(3,1)}$	1	0	*	0	0	1	0	0	0	0	0	*	0	*	0	1	0
$\mathcal{E}^{(3,2)}$	0	1	0	*	0	0	1	0	0	0	0	0	*	0	*	1	0
$\mathcal{E}^{(3,3)}$	*	0	1	0	0	*	0	*	0	*	0	*	0	*	0	0	1
$\mathcal{E}^{(3,4)}$	0	*	0	1	0	0	*	0	*	0	*	0	*	0	*	0	1
$\mathcal{E}^{(3,5)}$	*	0	1	1	1	0	0	0	0	0	0	*	*	0	0	0	0

Table 6.8: Part of process and sensor fault signature matrix of  $\mathcal{M}^{(4)}$  (\*,0,1).

	$f_1^{(4)}$	$f_2^{(4)}$	$f_3^{(4)}$	$f_4^{(4)}$	$f_5^{(4)}$	$f_6^{(4)}$	$f_1^{(1)}$	$f_2^{(1)}$	$f_1^{(3)}$	$f_2^{(3)}$	$f_3^{(3)}$	$f_4^{(3)}$	$f_5^{(3)}$	$f_{x,int}^{(4)}$	$f_{z,Tint}^{(4)}$	$f_{z,Tc}^{(4)}$
$\mathcal{E}^{(4,1)}$	1	0	*	0	0	0	*	0	1	0	*	0	0	1	0	0
$\mathcal{E}^{(4,2)}$	0	1	0	*	0	0	0	*	0	1	0	*	0	1	0	0
$\mathcal{E}^{(4,3)}$	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0
$\mathcal{E}^{(4,4)}$	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0
$\mathcal{E}^{(4,5)}$	*	0	*	0	1	0	*	0	*	0	*	0	*	0	0	1
$\mathcal{E}^{(4,6)}$	0	*	0	*	0	1	*	*	0	*	0	*	0	0	0	1



## 6.4. Multiple Fault Isolation Examples

At this point, some examples of successful fault isolation of multiple faults that occur in the same simulation run shall be shown. The first example concerns the initiation of two sensor faults in different systems, while the second example concerns the initiation of two process faults that affect different systems. A third example concerns the activation of one process and one sensor fault in different systems, while the last example concerns the initiation of one process and one sensor fault in the same system. The details regarding the results of these examples can be seen in appendix A in the figures of section A.1.

### 6.4.1. Multiple Sensor Faults

In this example, the scenario of multiple sensor faults that occur in different systems shall be described. Specifically, two sensor faults were initiated in  $\Sigma^{(3)}$  and  $\Sigma^{(4)}$  respectively. This example was considered to check if the aforementioned FDI methodology manages to detect and isolate multiple sensor faults in different systems. The first fault was initiated at 175sec at one of the two temperature sensors responsible for the measurement of  $z_1^{(3)}$ , and the second fault was initiated at 235sec at one of the two pressure sensors responsible for the measurement of  $x^{(4)}$ . Here it is expected to witness the effect of these two faults, firstly on the sensors, they occur in, and secondly if they propagate to other agents. It may be noticed that in table 6.7, the occurrence of sensor fault  $f_3^{(3)}$  or  $f_4^{(3)}$  in their respective columns, should propagate and affect the ARR  $\varepsilon^{(3,5)}$ , as can be seen in the last row of the table. These two sensor faults may also propagate to other agents, as is shown in the rest of the fault signature matrices, except the first one. Furthermore, the occurrence of the sensor fault  $f_1^{(4)}$  or  $f_2^{(4)}$ , may affect the ARRs that correspond to the last two rows of table 6.8. It is expected that they propagate to agent  $\mathcal{M}^{(2)}$  as can be seen in table 6.6 in the respective columns for these two faults, while they may also propagate to agent  $\mathcal{M}^{(3)}$ .

#### Example Evolution

To check if the aforementioned expectations are realized or not, a detailed description of the evolution of this example shall be discussed. For  $t < 175sec$ , the diagnosis set  $\mathcal{D}(t)$  is empty. For  $175sec \leq t < 235sec$  the agents  $\mathcal{M}^{(2)}$ ,  $\mathcal{M}^{(3)}$  and  $\mathcal{M}^{(4)}$  produce the decision vectors  $D^{(2)}(t) = [0\ 0\ 0\ 0\ 0\ 0]^T$ ,  $D^{(3)}(t) = [0\ 0\ 0\ 1\ 1]^T$  and  $D^{(4)}(t) = [0\ 0\ 0\ 0\ 0\ 0]^T$  respectively. Therefore, with the use of tables 6.6, 6.7 and 6.8, the resulting local diagnosis sets are  $\mathcal{D}_s^{(2)} = \{\}$ ,  $\mathcal{D}_s^{(3)} = \{f_4^{(3)}, f_5^{(3)}, \{f_4^{(3)}, f_5^{(3)}\}\}$ , and  $\mathcal{D}_s^{(4)} = \{\}$  respectively. The global decision vector is  $\mathcal{D}_\chi = [0\ 0\ 1\ 0]^T$  and if compared to table 6.9 yields the diagnosis set on fault propagation  $\mathcal{D}_s^\chi = \{f_1^{(3)}, f_2^{(3)}, f_3^{(3)}, f_4^{(3)}, f_5^{(3)}\}$  and any other combination of these faults. Therefore, the global diagnosis set lies in the intersection  $\mathcal{D}_s^G(t) = \mathcal{D}_s^\chi \cap \mathcal{D}_s = \{f_4^{(3)}, f_5^{(3)}, \{f_4^{(3)}, f_5^{(3)}\}\}$  since  $\mathcal{D}_s = \mathcal{D}_s^{(3)}$  because only  $\mathcal{D}_s^{(3)}$  is nonempty. For  $t > 235sec$  the output of monitoring agents  $\mathcal{M}^{(2)}$  and  $\mathcal{M}^{(4)}$  are  $D^{(2)}(t) = [0\ 1\ 0\ 1\ 0\ 0]^T$ ,  $D^{(3)}(t) = [0\ 0\ 0\ 1\ 1]^T$  and  $D^{(4)}(t) = [0\ 1\ 0\ 0\ 0\ 0]^T$  respectively while the output of  $\mathcal{M}^{(3)}$  and  $\mathcal{D}_s^{(3)}$  remains the same as can be seen in the figs. A.1-A.10. According to tables 6.6 and 6.8 the new diagnosis sets of  $\mathcal{M}^{(2)}$  and  $\mathcal{M}^{(4)}$  are  $\mathcal{D}_s^{(2)} = \{f_2^{(1)}, f_2^{(4)}, \{f_2^{(4)}, f_2^{(1)}\}\}$  and  $\mathcal{D}_s^{(4)} = \{f_2^{(4)}, f_2^{(1)}, f_2^{(3)}, f_4^{(3)}\}$  with any other combination of these faults. The unified local diagnosis sets yield  $\mathcal{D}_s(t) = \mathcal{D}_s^{(2)} \cup \mathcal{D}_s^{(3)} \cup \mathcal{D}_s^{(4)} = \{f_2^{(1)}, f_2^{(3)}, f_4^{(3)}, f_5^{(3)}, f_2^{(4)}\}$  and any other combination of these faults. Then, according to table 6.9 and the new global decision vector  $\mathcal{D}_\chi = [0\ 1\ 1\ 1]^T$ , the diagnosis set on fault propagation is  $\mathcal{D}_s^\chi = \{f_5^{(2)}, f_6^{(2)}, f_1^{(3)}, f_2^{(3)}, f_3^{(3)}, f_4^{(3)}, f_5^{(3)}, f_1^{(4)}, f_2^{(4)}, f_3^{(4)}, f_4^{(4)}, \{f_5^{(2)}, f_1^{(4)}\}, \{f_5^{(2)}, f_2^{(4)}\}, \{f_5^{(2)}, f_3^{(4)}\}, \{f_5^{(2)}, f_4^{(4)}\}, \{f_6^{(2)}, f_1^{(4)}\}, \{f_6^{(2)}, f_2^{(4)}\}, \{f_6^{(2)}, f_3^{(4)}\}, \{f_6^{(2)}, f_4^{(4)}\}, \{f_1^{(3)}, f_1^{(4)}\}, \{f_1^{(3)}, f_2^{(4)}\}, \{f_1^{(3)}, f_3^{(4)}\}, \{f_1^{(3)}, f_4^{(4)}\}, \{f_2^{(3)}, f_1^{(4)}\}, \{f_2^{(3)}, f_2^{(4)}\}, \{f_2^{(3)}, f_3^{(4)}\}, \{f_2^{(3)}, f_4^{(4)}\}, \{f_3^{(3)}, f_1^{(4)}\}, \{f_3^{(3)}, f_2^{(4)}\}, \{f_3^{(3)}, f_3^{(4)}\}, \{f_3^{(3)}, f_4^{(4)}\}, \{f_4^{(3)}, f_1^{(4)}\}, \{f_4^{(3)}, f_2^{(4)}\}, \{f_4^{(3)}, f_3^{(4)}\}, \{f_4^{(3)}, f_4^{(4)}\}, \{f_4^{(3)}, f_5^{(4)}\}\}$ . Thus the global diagnosis set is  $\mathcal{D}_s^G(t) = \mathcal{D}_s^\chi \cap \mathcal{D}_s = \{f_2^{(4)}, f_2^{(3)}, f_4^{(3)}, f_5^{(3)}\}$  and any other combination of these faults. The global diagnosis set  $\mathcal{D}_s^G$  and the local diagnosis sets remain unchanged until the end of the simulation and the simulation results can be viewed in figs. A.1-A.10.

### 6.4.2. Multiple Process Faults

In this example, the scenario of multiple process faults that occur in different systems shall be described. Specifically, two sensor faults were initiated in  $\Sigma^{(2)}$  and  $\Sigma^{(1)}$  respectively. This example was considered to check if the aforementioned FDI methodology manages to detect and isolate multiple process faults in different systems. More specifically, the first process fault concerns the internal damage of a cylinder

unit initiated at 5sec, while the second one concerns the blocking of the fuel injection valve initiated at 200sec. Here it is expected to witness that these process faults shall affect the systems that they occur in and that they will be detected and distinguished as such, thanks to the use of redundant sensors and modules. It is also expected that they will not propagate to other agents for reasons explained in the previous chapter.

### Example Evolution

To check if the aforementioned expectations are realized or not, a detailed description of the evolution of this example shall be discussed, but also one can review it in figs. A.11-A.20. For  $t < 146.1sec$ , the diagnosis set  $\mathcal{D}(t)$  is empty. For  $146.1sec \leq t < 201.6sec$ , monitoring agent  $\mathcal{M}^{(2)}$  produces the decision vector  $D^{(2)}(t) = [0 \ 0 \ 1 \ 1 \ 0 \ 0]^T$  while the rest of the monitoring agents maintain empty decision vectors. For  $201.6sec \leq t < 207.1sec$ , monitoring agent  $\mathcal{M}^{(2)}$  changes its decision vector once more to  $D^{(2)}(t) = [1 \ 1 \ 1 \ 1 \ 0 \ 0]^T$  while the rest of the monitoring agents maintain empty decision vectors. After consulting table 6.6, the resulting local diagnosis set is  $\mathcal{D}_s^{(2)} = \{f_{z,V_1}^{(2)}\}$ . The global decision vector is  $\mathcal{D}_\chi = [0 \ 1 \ 0 \ 0]^T$  and if compared to table 6.9 yields the diagnosis set on fault propagation  $\mathcal{D}_s^\chi = \{f_1^{(2)}, f_2^{(2)}, f_3^{(2)}, f_4^{(2)}, f_5^{(2)}, f_6^{(2)}, f_{z,V_1}^{(2)}\}$  and any other combination of these faults. Therefore, the global diagnosis set lies in the intersection  $\mathcal{D}_s^G(t) = \mathcal{D}_s^\chi \cap \mathcal{D}_s^{(2)} = \{f_{z,V_1}^{(2)}\}$ . Then, for  $t > 207.1sec$  the agents  $\mathcal{M}^{(1)}$ ,  $\mathcal{M}^{(2)}$ ,  $\mathcal{M}^{(3)}$  and  $\mathcal{M}^{(4)}$  produce the decision vectors  $D^{(1)}(t) = [1 \ 1]^T$ ,  $D^{(2)}(t) = [1 \ 1 \ 1 \ 1 \ 0 \ 0]^T$ ,  $D^{(3)}(t) = [0 \ 0 \ 0 \ 0 \ 0]^T$  and  $D^{(4)}(t) = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$  respectively. With the use of tables 6.5, 6.6, 6.7 and 6.8, the resulting local diagnosis sets are  $\mathcal{D}_s^{(1)} = \{f_u^{(1)}\}$ ,  $\mathcal{D}_s^{(2)} = \{f_{z,V_1}^{(2)}\}$ ,  $\mathcal{D}_s^{(3)} = \{\}$ , and  $\mathcal{D}_s^{(4)} = \{\}$  respectively. The unified local diagnosis sets yield  $\mathcal{D}_s(t) = \mathcal{D}_s^{(1)} \cup \mathcal{D}_s^{(2)} = \{f_u^{(1)}, f_{z,V_1}^{(2)}\}$ . The new global decision vector is  $\mathcal{D}_\chi = [1 \ 1 \ 0 \ 0]^T$  and if compared to table 6.9 yields the diagnosis set on fault propagation  $\mathcal{D}_s^\chi = \{f_1^{(1)}, f_2^{(1)}, f_u^{(1)}, f_1^{(2)}, f_2^{(2)}, f_3^{(2)}, f_4^{(2)}, f_5^{(2)}, f_6^{(2)}, f_{z,V_1}^{(2)}, \{f_1^{(1)}, f_1^{(2)}\}, \{f_1^{(1)}, f_2^{(2)}\}, \{f_1^{(1)}, f_3^{(2)}\}, \{f_1^{(1)}, f_4^{(2)}\}, \{f_1^{(1)}, f_5^{(2)}\}, \{f_1^{(1)}, f_6^{(2)}\}, \{f_1^{(1)}, f_{z,V_1}^{(2)}\}, \{f_2^{(1)}, f_1^{(2)}\}, \{f_2^{(1)}, f_2^{(2)}\}, \{f_2^{(1)}, f_3^{(2)}\}, \{f_2^{(1)}, f_4^{(2)}\}, \{f_2^{(1)}, f_5^{(2)}\}, \{f_2^{(1)}, f_6^{(2)}\}, \{f_2^{(1)}, f_{z,V_1}^{(2)}\}, \{f_u^{(1)}, f_1^{(2)}\}, \{f_u^{(1)}, f_2^{(2)}\}, \{f_u^{(1)}, f_3^{(2)}\}, \{f_u^{(1)}, f_4^{(2)}\}, \{f_u^{(1)}, f_5^{(2)}\}, \{f_u^{(1)}, f_6^{(2)}\}, \{f_u^{(1)}, f_{z,V_1}^{(2)}\}\}$ . Thus the global diagnosis set is  $\mathcal{D}_s^G(t) = \mathcal{D}_s^\chi \cap \mathcal{D}_s = \{f_u^{(1)}, f_{z,V_1}^{(2)}\}$ . It is important to mention here that the process faults can be isolated locally by the local agents, so the global agent is not contributing to the isolation of the two process faults in this case. The global diagnosis set  $\mathcal{D}_s^\chi$  and the local diagnosis sets remain unchanged until the end of the simulation and the simulation results can be viewed in figs. A.11-A.20.

### 6.4.3. Multiple Sensor and Process Faults in Different Systems

In this example, the scenario of multiple sensor and process faults that occur in different systems shall be described. Specifically, one sensor fault was initiated in  $\Sigma^{(2)}$  and a process fault in  $\Sigma^{(4)}$ . This example was considered to check if the aforementioned FDI methodology manages to detect and isolate multiple sensor and process faults in different systems. The sensor fault was initiated at 20sec at one of the two pressure sensors responsible for the measurement of  $z_1^{(2)}$ , and the process fault concerning the seal leakage inside the turbocharger's compressor was activated at 30sec. Here it is expected to witness firstly the effect of the sensor fault, secondly the effect of process fault on the system it occurred, and third possible fault propagation. It may be noticed that in table 6.6, the occurrence of sensor fault  $f_1^{(2)}$  or  $f_2^{(2)}$  in their respective columns, should only affect ARR $s$   $\varepsilon^{(2,1)}$  and  $\varepsilon^{(2,2)}$  respectively. These two sensor faults may also propagate to other agents, as is shown in the rest of the fault signature matrices, except the first one. Furthermore, the occurrence of the process fault  $f_{x,int}^{(4)}$  should affect the ARR $s$  that correspond to the rows  $\varepsilon^{(4,1)}$  and  $\varepsilon^{(4,2)}$  at the same time, as can be seen in table 6.8 regarding the particular fault column.

### Example Evolution

To check if the aforementioned expectations are realized or not, a detailed description of the evolution of this example shall be discussed, but also one can review it in figs. A.21-A.30. For  $t < 20sec$ , the diagnosis set  $\mathcal{D}(t)$  is empty. For  $20sec \leq t < 156.8sec$  the agents  $\mathcal{M}^{(2)}$ ,  $\mathcal{M}^{(3)}$  and  $\mathcal{M}^{(4)}$  produce the decision vectors  $D^{(2)}(t) = [0 \ 1 \ 0 \ 0 \ 0 \ 0]^T$ ,  $D^{(3)}(t) = [0 \ 0 \ 0 \ 0 \ 0]^T$  and  $D^{(4)}(t) = [0 \ 0 \ 0 \ 0 \ 0]^T$  respectively. Therefore, with the use of tables 6.6, 6.7 and 6.8, the resulting local diagnosis sets are  $\mathcal{D}_s^{(2)} = \{f_2^{(2)}\}$ ,  $\mathcal{D}_s^{(3)} = \{\}$ , and  $\mathcal{D}_s^{(4)} = \{\}$  respectively. The global decision vector is  $\mathcal{D}_\chi = [0 \ 1 \ 0 \ 0]^T$  and if compared to table 6.9 yields the diagnosis set on fault propagation  $\mathcal{D}_s^\chi = \{f_1^{(2)}, f_2^{(2)}, f_3^{(2)}, f_4^{(2)}, f_5^{(2)}, f_6^{(2)}, f_{z,V_1}^{(2)}\}$ . Therefore, the global diagnosis set lies in the intersection  $\mathcal{D}_s^G(t) = \mathcal{D}_s^\chi \cap \mathcal{D}_s^{(2)} = \{f_2^{(2)}\}$ . Thereafter, for

$t > 156.8sec$  the agents  $\mathcal{M}^{(2)}$ ,  $\mathcal{M}^{(3)}$  and  $\mathcal{M}^{(4)}$  produce the decision vectors  $D^{(2)}(t) = [0 \ 1 \ 0 \ 0 \ 0 \ 0]^T$ ,  $D^{(3)}(t) = [0 \ 0 \ 0 \ 0 \ 0]^T$  and  $D^{(4)}(t) = [1 \ 1 \ 0 \ 0 \ 0 \ 0]^T$  respectively. With the use of tables 6.6, 6.7 and 6.8, the resulting local diagnosis sets are  $\mathcal{D}_s^{(2)} = \{f_2^{(2)}\}$ ,  $\mathcal{D}_s^{(3)} = \{\}$ , and  $\mathcal{D}_s^{(4)} = \{f_{x,int}^{(4)}\}$  respectively. The local diagnosis sets can be unified  $\mathcal{D}_s(t) = \mathcal{D}_s^{(2)} \cup \mathcal{D}_s^{(4)} = \{f_{x,int}^{(4)}, f_2^{(2)}\}$ . The new global decision vector is  $\mathcal{D}_\chi = [0 \ 1 \ 0 \ 1]^T$  and if compared to Table 6.9 yields the diagnosis set on fault propagation  $\mathcal{D}_s^\chi = \{f_1^{(4)}, f_2^{(4)}, f_3^{(4)}, f_4^{(4)}, \{f_5^{(4)}, f_1^{(2)}\}, \{f_5^{(4)}, f_2^{(2)}\}, \{f_5^{(4)}, f_3^{(2)}\}, \{f_5^{(4)}, f_4^{(2)}\}, \{f_5^{(4)}, f_5^{(2)}\}, \{f_5^{(4)}, f_6^{(2)}\}, \{f_5^{(4)}, f_{z,V_1}^{(2)}\}, \{f_6^{(4)}, f_1^{(2)}\}, \{f_6^{(4)}, f_2^{(2)}\}, \{f_6^{(4)}, f_3^{(2)}\}, \{f_6^{(4)}, f_4^{(2)}\}, \{f_6^{(4)}, f_5^{(2)}\}, \{f_6^{(4)}, f_6^{(2)}\}, \{f_6^{(4)}, f_{z,V_1}^{(2)}\}, \{f_{x,int}^{(4)}, f_1^{(2)}\}, \{f_{x,int}^{(4)}, f_2^{(2)}\}, \{f_{x,int}^{(4)}, f_3^{(2)}\}, \{f_{x,int}^{(4)}, f_4^{(2)}\}, \{f_{x,int}^{(4)}, f_5^{(2)}\}, \{f_{x,int}^{(4)}, f_6^{(2)}\}, \{f_{x,int}^{(4)}, f_{z,V_1}^{(2)}\}, \{f_{z,T_{inl}}^{(4)}, f_1^{(2)}\}, \{f_{z,T_{inl}}^{(4)}, f_2^{(2)}\}, \{f_{z,T_{inl}}^{(4)}, f_3^{(2)}\}, \{f_{z,T_{inl}}^{(4)}, f_4^{(2)}\}, \{f_{z,T_{inl}}^{(4)}, f_5^{(2)}\}, \{f_{z,T_{inl}}^{(4)}, f_6^{(2)}\}, \{f_{z,T_{inl}}^{(4)}, f_{z,V_1}^{(2)}\}, \{f_{z,T_c}^{(4)}, f_1^{(2)}\}, \{f_{z,T_c}^{(4)}, f_2^{(2)}\}, \{f_{z,T_c}^{(4)}, f_3^{(2)}\}, \{f_{z,T_c}^{(4)}, f_4^{(2)}\}, \{f_{z,T_c}^{(4)}, f_5^{(2)}\}, \{f_{z,T_c}^{(4)}, f_6^{(2)}\}, \{f_{z,T_c}^{(4)}, f_{z,V_1}^{(2)}\}$ . The global diagnosis set is  $\mathcal{D}_s^\chi(t) = \mathcal{D}_s^\chi \cap \mathcal{D}_s = \{f_{x,int}^{(4)}, f_2^{(2)}\}$ . The global diagnosis set  $\mathcal{D}_s^\chi$  and the local diagnosis sets remain unchanged until the end of the simulation and the simulation results can be viewed in figs. A.21-A.30.

#### 6.4.4. Multiple Sensor and Process Faults in the Same System

In this example, the scenario of multiple sensor and process faults initiated in the same system will be described. Specifically, these faults occurred in system  $\Sigma^{(3)}$ . This example was considered to check if the aforementioned FDI methodology manages to detect and isolate multiple sensor and process faults in the same system. The sensor fault was initiated at 15sec at one of the two pressure sensors responsible for the measurement of  $x^{(3)}$ , and the process fault regarding the loss of lubrication of the turbine shaft  $f_{z,T_{sl}}^{(3)}$ , was initiated at 30sec. Here it is expected to witness firstly the effect of the sensor fault, secondly the effect of process fault on the system, and third possible fault propagation. It may be noticed that in table 6.7, the occurrence of sensor fault  $f_1^{(3)}$  or  $f_2^{(3)}$  in their respective columns, should affect ARRs  $\varepsilon^{(3,1)}$  and  $\varepsilon^{(3,2)}$  respectively and it may propagate to other modules of agent  $\mathcal{M}^{(3)}$  as can be seen in the rest of the rows. This sensor fault should also propagate to agent  $\mathcal{M}^{(4)}$  as can be seen in table 6.8. Furthermore, the occurrence of the process fault  $f_{z,T_{sl}}^{(3)}$  should affect the ARRs that correspond to the rows  $\varepsilon^{(3,3)}$  and  $\varepsilon^{(3,4)}$  at the same time, as shown in table 6.7 regarding the particular fault.

##### Example Evolution

To check if the aforementioned expectations are realized or not, a detailed description of the evolution of this example shall be discussed, but also one can review it in figs. A.31-A.40. For  $t < 15sec$ , the diagnosis set  $\mathcal{D}(t)$  is empty. For  $15sec \leq t < 104.6sec$  the agents  $\mathcal{M}^{(2)}$ ,  $\mathcal{M}^{(3)}$  and  $\mathcal{M}^{(4)}$  produce the decision vectors  $D^{(2)}(t) = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ ,  $D^{(3)}(t) = [0 \ 1 \ 0 \ 0 \ 0]^T$  and  $D^{(4)}(t) = [0 \ 1 \ 0 \ 0 \ 0 \ 0]^T$  respectively. Thus, with the use of tables 6.6, 6.7 and 6.8, the resulting local diagnosis set  $\mathcal{D}_s^{(2)} = \{\}$ ,  $\mathcal{D}_s^{(3)} = \{f_2^{(3)}, f_2^{(1)}, f_2^{(4)}, f_4^{(4)}, \{f_2^{(3)}, f_2^{(1)}\}, \{f_2^{(3)}, f_2^{(4)}\}, \{f_2^{(3)}, f_4^{(4)}\}, \{f_2^{(4)}, f_2^{(1)}\}, \{f_2^{(4)}, f_2^{(1)}\}, \{f_2^{(4)}, f_4^{(4)}\}$ , and  $\mathcal{D}_s^{(4)} = \{f_2^{(4)}, f_2^{(1)}, f_2^{(3)}, f_4^{(3)}, \{f_2^{(4)}, f_2^{(1)}\}, \{f_2^{(4)}, f_2^{(3)}\}, \{f_2^{(4)}, f_4^{(3)}\}, \{f_2^{(3)}, f_2^{(1)}\}, \{f_2^{(3)}, f_2^{(1)}\}, \{f_2^{(3)}, f_4^{(3)}\}$  respectively. Moreover, the global decision vector is  $\mathcal{D}_\chi = [0 \ 0 \ 1 \ 1]^T$  and if compared to table 6.9 yields the diagnosis set on fault propagation  $\mathcal{D}_s^\chi = \{f_1^{(3)}, f_2^{(3)}, f_3^{(3)}, f_4^{(3)}, f_5^{(3)}, f_1^{(4)}, f_2^{(4)}, f_3^{(4)}, f_4^{(4)}, f_{x,tur}^{(3)}, f_{z,T_{sl}}^{(3)}, f_{x,int}^{(4)}, f_{z,T_{inl}}^{(4)}, \{f_1^{(3)}, f_1^{(4)}\}, \{f_1^{(3)}, f_2^{(4)}\}, \{f_1^{(3)}, f_3^{(4)}\}, \{f_1^{(3)}, f_4^{(4)}\}, \{f_1^{(3)}, f_{x,int}^{(4)}\}, \{f_1^{(3)}, f_{z,T_{inl}}^{(4)}\}, \{f_2^{(3)}, f_1^{(4)}\}, \{f_2^{(3)}, f_2^{(4)}\}, \{f_2^{(3)}, f_3^{(4)}\}, \{f_2^{(3)}, f_4^{(4)}\}, \{f_2^{(3)}, f_{x,int}^{(4)}\}, \{f_2^{(3)}, f_{z,T_{inl}}^{(4)}\}, \{f_2^{(3)}, f_{x,int}^{(4)}\}, \{f_2^{(3)}, f_{z,T_{inl}}^{(4)}\}, \{f_3^{(3)}, f_1^{(4)}\}, \{f_3^{(3)}, f_2^{(4)}\}, \{f_3^{(3)}, f_3^{(4)}\}, \{f_3^{(3)}, f_4^{(4)}\}, \{f_3^{(3)}, f_{x,int}^{(4)}\}, \{f_3^{(3)}, f_{z,T_{inl}}^{(4)}\}, \{f_4^{(3)}, f_1^{(4)}\}, \{f_4^{(3)}, f_2^{(4)}\}, \{f_4^{(3)}, f_3^{(4)}\}, \{f_4^{(3)}, f_4^{(4)}\}, \{f_4^{(3)}, f_{x,int}^{(4)}\}, \{f_4^{(3)}, f_{z,T_{inl}}^{(4)}\}, \{f_5^{(3)}, f_1^{(4)}\}, \{f_5^{(3)}, f_2^{(4)}\}, \{f_5^{(3)}, f_3^{(4)}\}, \{f_5^{(3)}, f_4^{(4)}\}, \{f_5^{(3)}, f_{x,int}^{(4)}\}, \{f_5^{(3)}, f_{z,T_{inl}}^{(4)}\}, \{f_{x,tur}^{(3)}, f_1^{(4)}\}, \{f_{x,tur}^{(3)}, f_2^{(4)}\}, \{f_{x,tur}^{(3)}, f_3^{(4)}\}, \{f_{x,tur}^{(3)}, f_4^{(4)}\}, \{f_{x,tur}^{(3)}, f_{x,int}^{(4)}\}, \{f_{x,tur}^{(3)}, f_{z,T_{inl}}^{(4)}\}, \{f_{z,T_{sl}}^{(3)}, f_1^{(4)}\}, \{f_{z,T_{sl}}^{(3)}, f_2^{(4)}\}, \{f_{z,T_{sl}}^{(3)}, f_3^{(4)}\}, \{f_{z,T_{sl}}^{(3)}, f_4^{(4)}\}, \{f_{z,T_{sl}}^{(3)}, f_{x,int}^{(4)}\}, \{f_{z,T_{sl}}^{(3)}, f_{z,T_{inl}}^{(4)}\}$ . The unified local diagnosis set yield  $\mathcal{D}_s(t) = \mathcal{D}_s^{(3)} \cup \mathcal{D}_s^{(4)} = \{f_2^{(1)}, f_2^{(3)}, f_4^{(3)}, f_2^{(4)}, f_4^{(4)}\}$  and any other combination of these faults. Hence, the global diagnosis set lies in the intersection  $\mathcal{D}_s^\chi(t) = \mathcal{D}_s^\chi \cap \mathcal{D}_s = \{f_2^{(3)}, f_4^{(3)}, f_2^{(4)}, f_4^{(4)}\}$  and any other combination of these faults. Thereafter, for  $t > 104.6sec$  the agents  $\mathcal{M}^{(2)}$ ,  $\mathcal{M}^{(3)}$  and  $\mathcal{M}^{(4)}$  produce the decision vectors  $D^{(2)}(t) = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ ,  $D^{(3)}(t) = [0 \ 1 \ 1 \ 1 \ 0]^T$  and  $D^{(4)}(t) = [0 \ 1 \ 0 \ 0 \ 0 \ 0]^T$  respectively. With the use of tables 6.6, 6.7 and 6.8, the resulting local diagnosis sets  $\mathcal{D}_s^{(2)} = \{\}$ ,  $\mathcal{D}_s^{(3)} = \{f_{T_{sl}}^{(3)}\}$ , and  $\mathcal{D}_s^{(4)}$  remains the same. The unified local diagnosis sets yield  $\mathcal{D}_s(t) = \mathcal{D}_s^{(3)} \cup \mathcal{D}_s^{(4)} = \{f_{T_{sl}}^{(3)}, f_2^{(4)}, f_2^{(1)}, f_2^{(3)}, f_4^{(3)}, \{f_2^{(4)}, f_2^{(1)}\}, \{f_2^{(4)}, f_2^{(3)}\}, \{f_2^{(4)}, f_4^{(3)}\}, \{f_2^{(3)}, f_2^{(1)}\}, \{f_4^{(3)}, f_2^{(1)}\}, \{f_2^{(3)}, f_4^{(3)}\}, \{f_{T_{sl}}^{(3)}, f_2^{(4)}\}, \{f_{T_{sl}}^{(3)}, f_2^{(1)}\}, \{f_{T_{sl}}^{(3)}, f_2^{(3)}\}, \{f_{T_{sl}}^{(3)}, f_4^{(3)}\}$ . The global decision vector remains  $\mathcal{D}_\chi = [0 \ 0 \ 1 \ 1]^T$  and if compared to table 6.9 yields the diagnosis set on fault propagation remains the same. Hence, the

global diagnosis set is  $\mathcal{D}_s^G(t) = \mathcal{D}_s^\chi \cap \mathcal{D}_s = \{f_2^{(3)}, f_4^{(3)}, f_2^{(4)}, \{f_2^{(3)}, f_4^{(3)}\}, \{f_4^{(3)}, f_2^{(4)}\}, \{f_2^{(3)}, f_2^{(4)}\}, \{f_{T_{sl}}^{(3)}, f_2^{(4)}\}, \{f_{T_{sl}}^{(3)}, f_2^{(3)}\}, \{f_{T_{sl}}^{(3)}, f_4^{(3)}\}\}$ . Here, the process fault was detected locally by the local monitoring agent of the system. The global diagnosis set  $\mathcal{D}_s^\chi$  and the local diagnosis sets remain unchanged until the end of the simulation and the simulation results can be viewed in figs. A.31-A.40.

## 6.5. Key Performance Indicators

In this section, the performance of the FDI scheme that was implemented shall be discussed. This will be given in the form of KPIs, which were chosen to highlight some criteria such as detection time standard deviation and miss-detection of faults by the monitoring modules.

### 6.5.1. Detection Time Standard Deviation

The aim of the KPI regarding detection time standard deviation  $\sigma^{(l,q)}$  is to show that the fault detection time  $T_{D_k}^{(l,q)}$  is independent of the sensor noise evolution. The simulation was iterated 100 times and at each repetition, the sensor noise vector had a different evolution but the same amplitude. In fig. 6.35, one may notice that the detection time delay of the faults by the different monitoring modules has a specific variability from the average or mean detection time delay depicted as  $\mu^{(l,q)}$  of each module  $\mathcal{M}^{(l,q)}$ . This standard deviation from the mean detection time delay  $\mu^{(l,q)}$  of each module, describes the ability to detect a fault, regardless of the sensor noise evolution since the standard deviation is maintained below  $\pm 0.003 \text{sec}$  for permanent, abrupt, offset sensor faults that were detected by the monitoring modules. The formula used to calculate the standard deviation of the population of detection time delays in seconds, which was created for each module over the 100 iterations can be seen [17].

$$\sigma^{(l,q)} = \pm \sqrt{\frac{\sum_{k=1}^K (T_{D_k}^{(l,q)} - T_f^{(l,q)} - \mu^{(l,q)})^2}{K}} \quad (6.1)$$

Here  $K$  represents the number of iterations set to 100. The mean detection time delay  $\mu^{(l,q)}$  of each module is calculated below.  $T_f^{(l,q)}$  denotes the occurrence time of the fault that affects each sensor.

$$\mu^{(l,q)} = \frac{1}{K} \sum_{k=1}^K (T_{D_k}^{(l,q)} - T_f^{(l,q)}) = \frac{(T_{D_1}^{(l,q)} - T_f^{(l,q)}) + (T_{D_2}^{(l,q)} - T_f^{(l,q)}) + \dots + (T_{D_k}^{(l,q)} - T_f^{(l,q)})}{K} \quad (6.2)$$

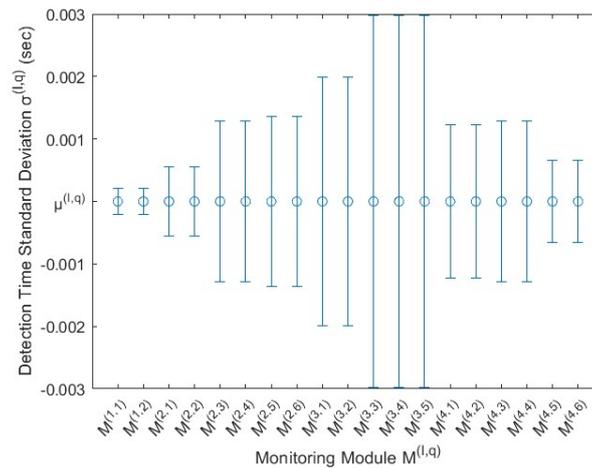


Figure 6.35: Detection time standard deviation of each monitoring module  $\mathcal{M}^{(l,q)}$ .

### 6.5.2. Miss-Detection of Process Faults

The miss-detection of process faults may depict the lowest fault magnitude value that a process fault can have and still be detected in the time frame of the 300sec of the simulation. Of course, this means

Table 6.10: Lowest fault magnitude values for the avoidance of miss-detection.

Fault	Fault Function $\phi$	Units
$f_{x,tur}^{(3)}$	$R_{x,tur}^{(3)} = 5x10^{-5}$	Hz
$f_{z,Tsl}^{(3)}$	$R_{z,Tsl}^{(3)} = 0.3$	K/s
$f_{x,int}^{(4)}$	$R_{x,int}^{(4)} = 78x10^{-4}$	Hz
$f_{z,Tc}^{(4)}$	$R_{z,Tc}^{(4)} = 3.75x10^{-2}$	K/s
$f_{z,T_{intl}}^{(4)}$	$R_{z,T_{intl}}^{(4)} = 0.165$	K/s
$f_{z,V_1}^{(2)}$	$R_{z,V_1}^{(2)} = 1.5x10^{-3}$	$m^3/s$
$f_u^{(1)}$	$\phi_u^{(1)} = 1.85x10^{-2}$	%

that a process fault will remain undetected for the largest part of the simulation. Furthermore, since the modeling of the process faults was conducted under some assumptions that may be questioned, now the lowest fault magnitude value will enable future researchers to have a reference value to compare their future process fault models, that they may develop taking into consideration sources that provide credible process fault modeling data. The lowest fault magnitude values for each process fault can be seen in table 6.10 in the fault function column. Any fault with a fault magnitude lower than the value indicated in this table will not be detected during the simulation time of 300sec and therefore is considered a miss-detection.

## 6.6. Discussion

In this chapter, the simulation results regarding the implementation of the FDI scheme that was provided earlier were presented. The aforementioned methodology was proven effective as it served its original purpose which was to detect and isolate both sensor and process faults. Furthermore, it showed good performance as was witnessed at the end of this chapter, where the KPI regarding detection time was given for all monitoring modules, and the minimum values for the fault magnitude of process faults were provided as a reference point for future researchers. In the next and final chapter, an evaluation of this work will be made, to discuss whether or not the initial research questions were covered and to what extent. Moreover, the room for improvement regarding this work, which can be carried out by future researchers, will be discussed. Finally, some conclusions will be made regarding the contribution of the present work.



# Conclusions and Future Directions

Overall, the adaptation of the FDI methodology presented in this work provides promising results that may indeed make a difference in the field of FD for marine vessels. The presented framework can become an early warning system for the avoidance of catastrophic failures on-board marine vessels, and a means of online monitoring that can be applied in multiple interconnected systems that are used for the propulsion of vessels.

## 7.1. Conclusions

The research questions that were initially formulated at the beginning of this work, were answered to a large extent. In the presented research work, the importance of FD on marine vessels was highlighted by witnessing faults that may occur within a part of the marine propulsion process. Some of the models that constitute a big part of this process, and more specifically the marine fuel engine, were provided and modeled in a simulation, as well as the sensors that monitor it, and the faults that affect either the sensors or the process itself. The validation of these models was attempted, although the results show a lot of deviation from preexisting FAT measurements. Furthermore, a theoretical model-based approach for the detection and isolation of process and sensor faults, with the adaptation of adaptive thresholds, monitoring agents, and redundant sensors, was provided. This methodology was adopted in the simulation and its results were presented eventually. These results included sensor and process fault detection, the theoretical fault signature matrices that were used to isolate the faults, and some KPIs to evaluate the performance of the adopted FDI scheme.

One of the benefits of the presented FDI methodology is its easiness of adaptation on-board vessels. This statement can be made since there is no need for data storage capabilities since this method runs online. Furthermore, no additional personnel is needed on-board to run it. Additionally, the incorporation of adaptive thresholds in the monitoring modules helps minimize false alarms and reduces the conservativeness in decision-making. The true potential that this framework may provide is the reduction of downtime of vessels, due to the early warning ability it provides to the crew, to take action and avoid failures. Finally, this discussion leads to the most important purpose that this framework may serve, which is non-other than the maritime safety of the crew on-board large marine vessels. This framework can be the basis for adopting maintenance decisions.

On the other hand, the adaptation of redundant sensors for the classification between sensor and process faults will lead to higher assumed costs for vessel owners. This is an inevitable evil, since the aforementioned framework, relies on these redundant sensors to make clear discrimination between sensor and process faults. Moreover, the existence of redundant sensors is not necessarily a loss of money, in the sense that if a sensor is declared faulty at any time, then a decision could be made by the crew to disregard the output of that sensor and only use the output of the redundant one that is used to measure the same physical quantity.

The contribution of this framework for the integrated process and sensor FDI is unique in many ways. First of all, it is one of the limited number of research works carried out in the field of model-based FDI for marine vessels, that manages to detect and isolate multiple process and sensor faults. The work that it can be compared to is namely [15] that concerns the FDI of actuator and sensor faults occurring

in the propulsion system of an underwater Remotely Operated Vehicle (ROV). Most FD methods for marine fuel engines today, are not model-based and comprise measurements from different engine parts. Most of these methods focus on the detection and isolation of actuator or other types of process faults. In fact, almost none of them detect and isolate both process and sensor faults as was also expressed in [53].

## 7.2. Future Directions

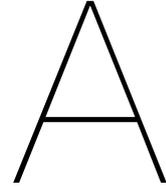
When reviewing the work presented so far, it is fair to say that the purpose of providing an integrated framework for the diagnosis of process and sensor faults for a marine propulsion process was achieved. With this being said, the room for improvement that needs to be narrowed down by conducting future research is undeniable and worth noting at this point. First of all, more systems related to the propulsion process of a marine vessel, such as its gearbox, shaft-line, propeller, and hull, should also be considered and this framework should be expanded to cover the complete propulsion process. Moreover, the future adaptation of this framework on a small fuel engine within a laboratory can be made to reproduce results that are based on the physical system measurements, in order to validate the simulation results of the FDI scheme and check its real-life adaptation feasibility, is also a target that needs to be carried out at some point in the future. Furthermore, if the framework is implemented on-board a marine vessel, the simulation should act as a “*digital twin*” of the physical system to compare the real-life indications of the framework, with the equivalent situation in the simulated environment. In this way, the simulation could provide more details regarding the alarms indicated by the framework, by importing the measurements of the physical system in the simulation. In this way, better maintenance decisions can be taken promptly and failures or accidents can be avoided. Finally, regarding the modeling of the process faults, one might argue that they could be further improved to represent more realistic characteristics during their evolution. This is a point of argument worth mentioning, since in the available literature, the characteristics of such faults are very limited, and in many cases non existing whatsoever. Therefore, a fault modeling campaign regarding process faults needs to be carried out to identify the parameters needed to simulate the characteristics of process faults, so that they can be introduced in the propulsion process.

### 7.2.1. Limitations to Consider in Future Research

After carrying out the presented research work and the simulations, a few limitations became apparent. Such limitations concern the sensor noise vectors that influence the outputs of the sensors. This noise also affects the residuals as well as the algebraic adaptive thresholds as it was presented in the previous chapter since the sensor outputs are used to derive them. Due to this influence, process faults were detected at a far later stage after their occurrence and this provides a finite time period for undetected faults to evolve into catastrophic failure. Furthermore, it can be argued that as the framework expands to include even more propulsion systems and sensors, the computational effort needed to carry out the simulation will increase substantially, since the dimensions of the fault signature matrices increase following a power law to determine all the fault combinations. This means that at some point in the future, the available computational power may become a limit, but that is something future researchers will have to confirm if they expand the current framework. Lastly, as was witnessed in chapter 5, for the algebraic adaptive thresholds the design parameters available for their formulation are very limited, unlike the state-based ones. This causes them to be prone to noise amplification.

## 7.3. Epilogue

Throughout the course of this thesis, it became obvious that the proposed model-based FD methodology can be feasibly implemented in a simulated environment. Overall, when conducting FD for any type of machinery or system in general, one must never neglect the importance of correct fault detection, determination of fault location, as well as isolation of the fault. This is the essence of FD, and any inaccuracy in doing so may lead to failures that can quickly cascade to other systems. Avoiding the failure of specific machinery is the end task and the product of any PHM effort, which can positively affect the efficiency and regular operation of the machinery or equipment of interest. Most importantly, it is the moral responsibility possessed by the vessel owners towards the crew on-board vessels to ensure that no failure will occur and no human life will be placed in danger as a consequence of lacking maintenance due to the absence of an adequate FD methodology.



# Appendix

## A.1. Simulation Results for Multiple Faults

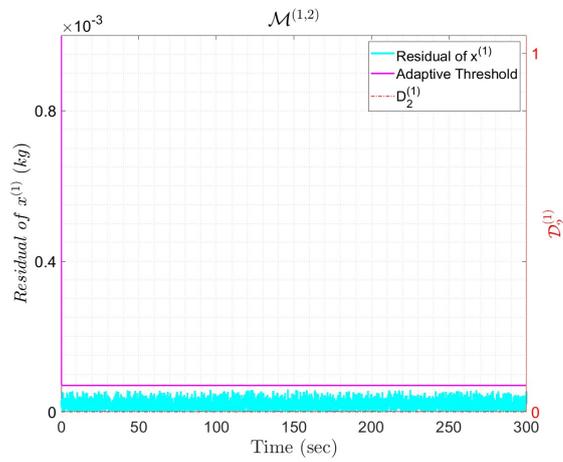


Figure A.1: Simulation results for the output of  $\mathcal{M}^{(1,2)}$  for example 1.

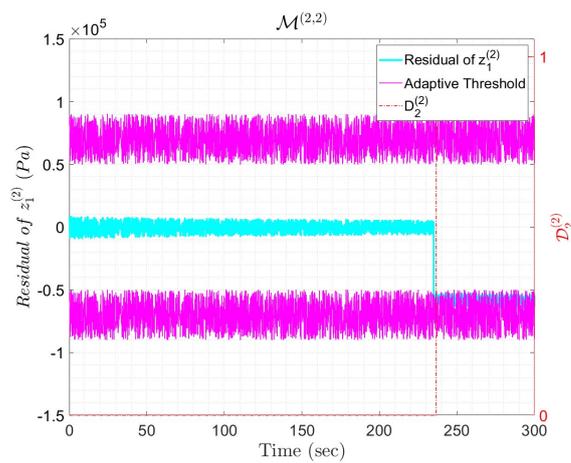
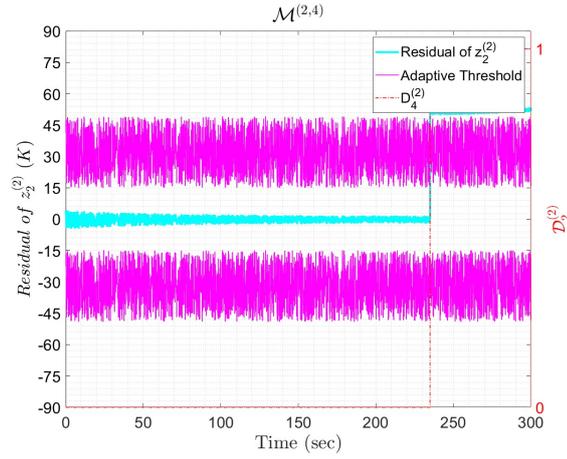
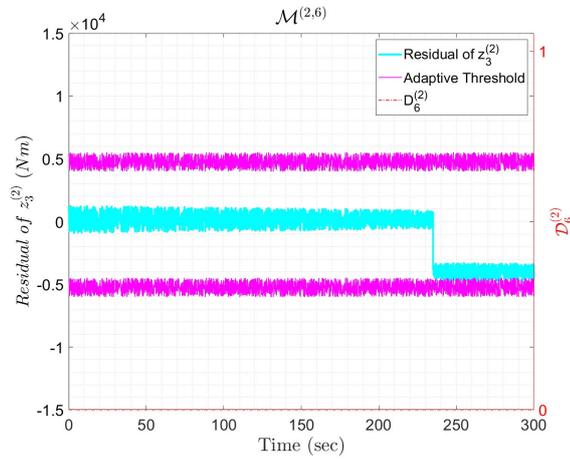
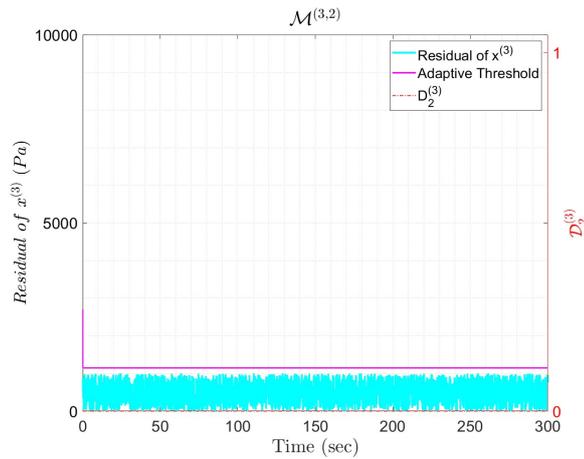


Figure A.2: Simulation results for the output of  $\mathcal{M}^{(2,2)}$  for example 1.

Figure A.3: Simulation results for the output of  $\mathcal{M}^{(2,4)}$  for example 1.Figure A.4: Simulation results for the output of  $\mathcal{M}^{(2,6)}$  for example 1.Figure A.5: Simulation results for the output of  $\mathcal{M}^{(3,2)}$  for example 1.

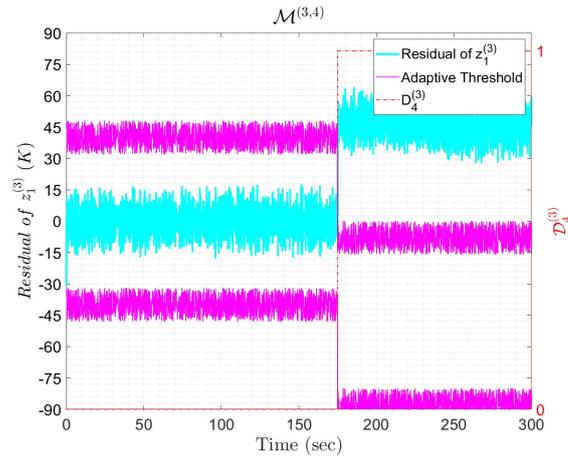


Figure A.6: Simulation results for the output of  $\mathcal{M}^{(3,4)}$  for example 1.

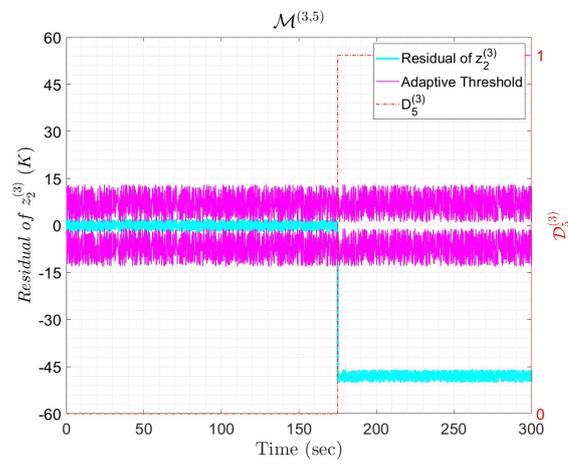


Figure A.7: Simulation results for the output of  $\mathcal{M}^{(3,5)}$  for example 1.

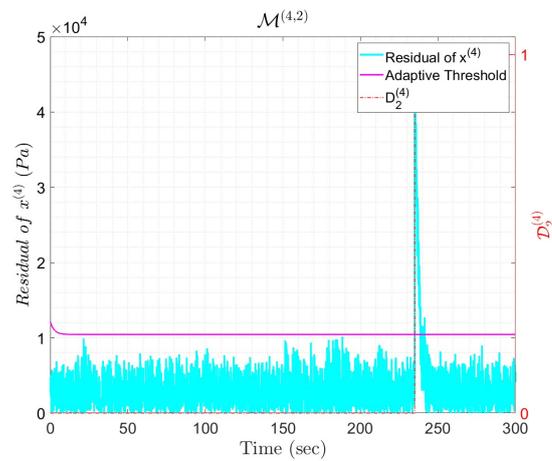
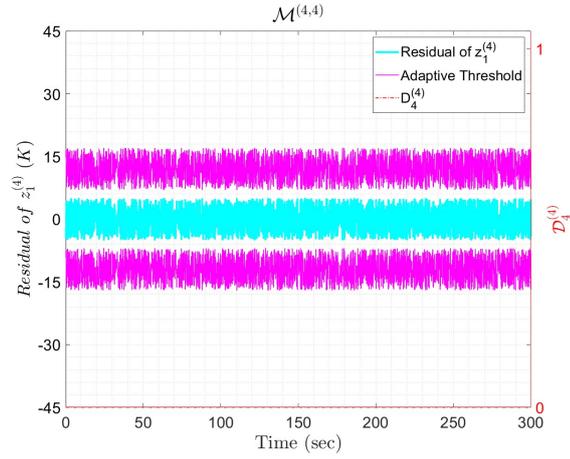
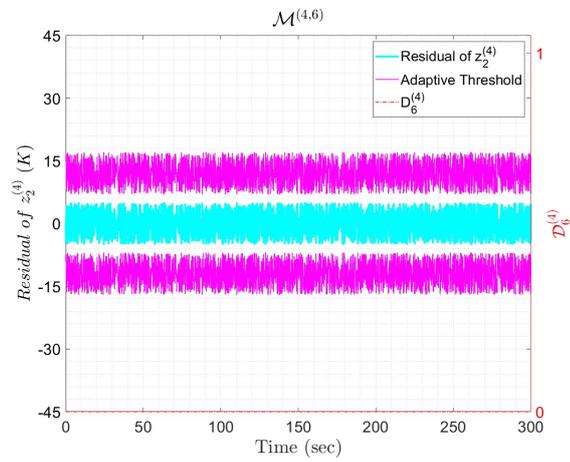
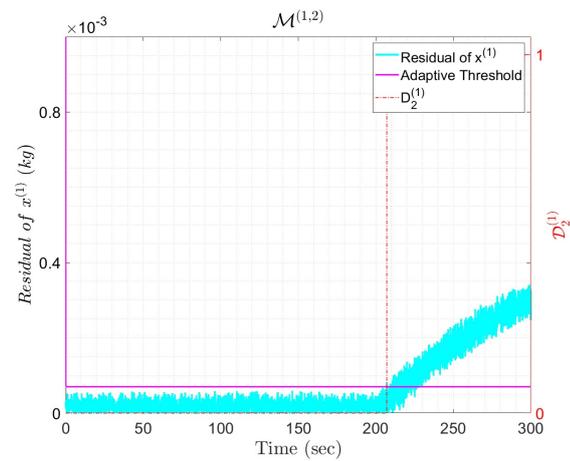


Figure A.8: Simulation results for the output of  $\mathcal{M}^{(4,2)}$  for example 1.

Figure A.9: Simulation results for the output of  $\mathcal{M}^{(4,4)}$  for example 1.Figure A.10: Simulation results for the output of  $\mathcal{M}^{(4,6)}$  for example 1.Figure A.11: Simulation results for the output of  $\mathcal{M}^{(1,2)}$  for example 2.

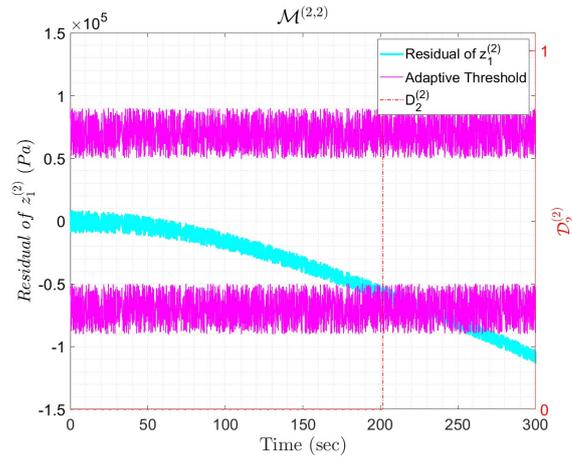


Figure A.12: Simulation results for the output of  $\mathcal{M}^{(2,2)}$  for example 2.

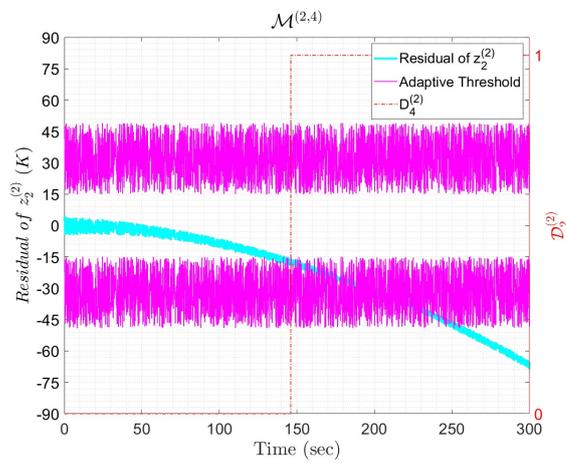


Figure A.13: Simulation results for the output of  $\mathcal{M}^{(2,4)}$  for example 2.

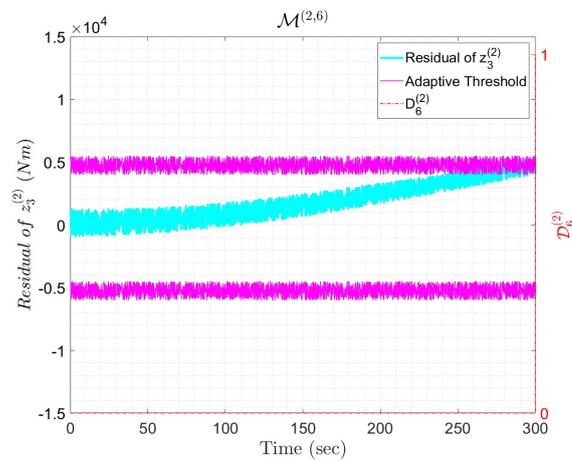
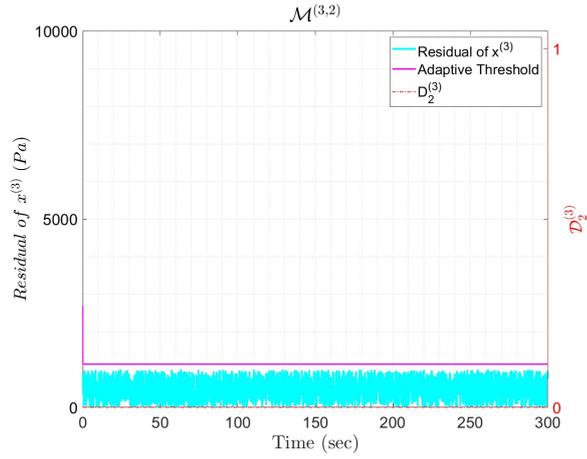
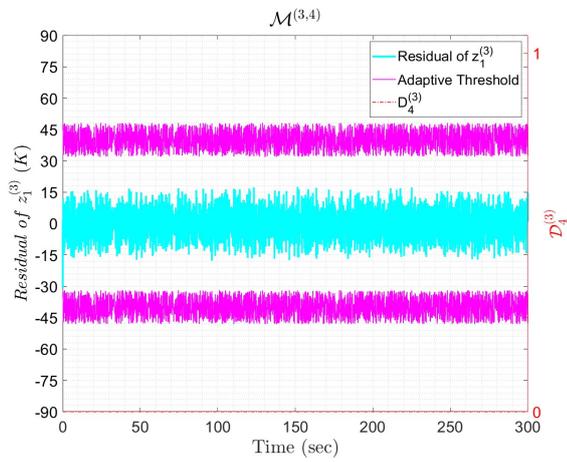
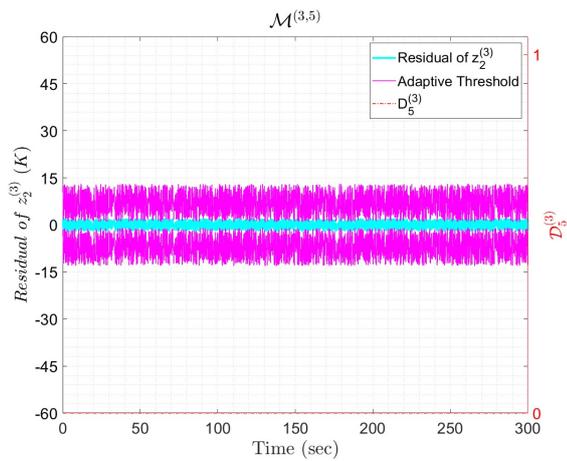


Figure A.14: Simulation results for the output of  $\mathcal{M}^{(2,6)}$  for example 2.

Figure A.15: Simulation results for the output of  $\mathcal{M}^{(3,2)}$  for example 2.Figure A.16: Simulation results for the output of  $\mathcal{M}^{(3,4)}$  for example 2.Figure A.17: Simulation results for the output of  $\mathcal{M}^{(3,5)}$  for example 2.

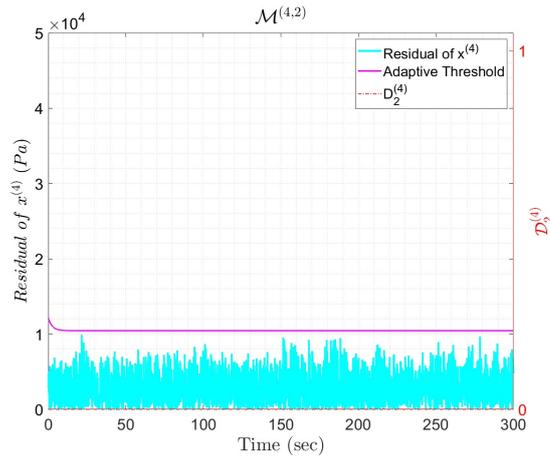


Figure A.18: Simulation results for the output of  $\mathcal{M}^{(4,2)}$  for example 2.

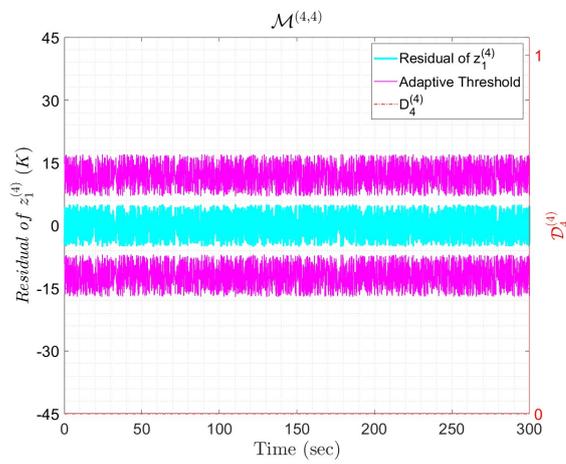


Figure A.19: Simulation results for the output of  $\mathcal{M}^{(4,4)}$  for example 2.

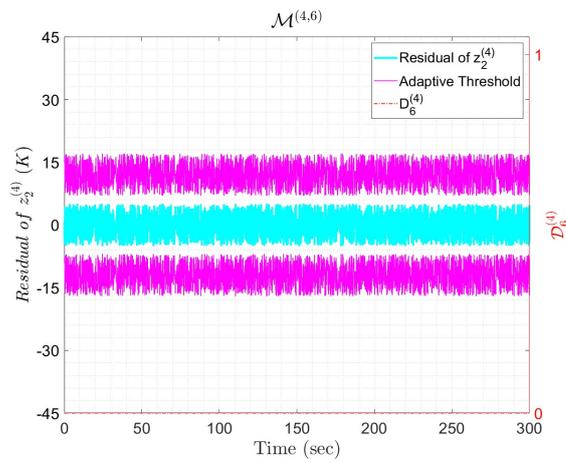
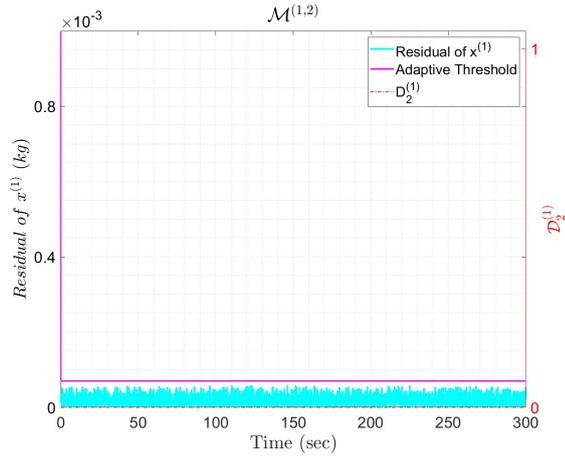
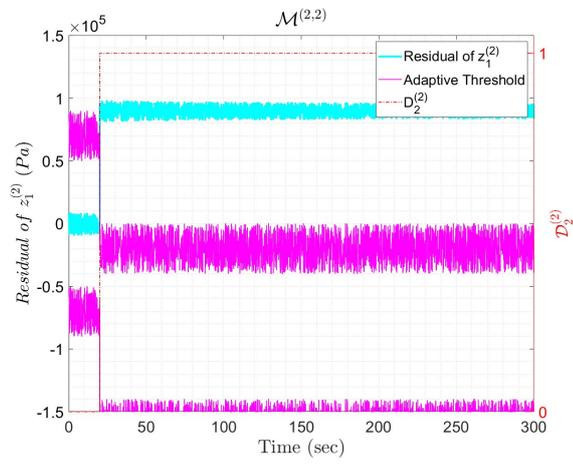
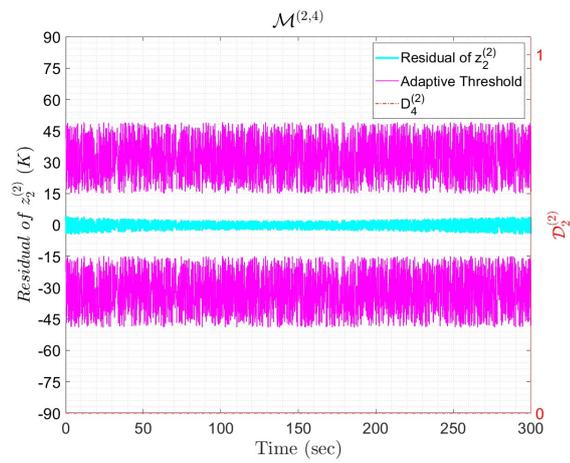


Figure A.20: Simulation results for the output of  $\mathcal{M}^{(4,6)}$  for example 2.

Figure A.21: Simulation results for the output of  $\mathcal{M}^{(1,2)}$  for example 3.Figure A.22: Simulation results for the output of  $\mathcal{M}^{(2,2)}$  for example 3.Figure A.23: Simulation results for the output of  $\mathcal{M}^{(2,4)}$  for example 3.

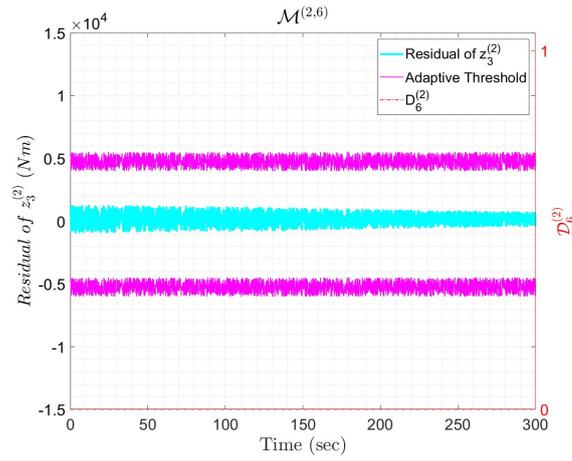


Figure A.24: Simulation results for the output of  $\mathcal{M}^{(2,6)}$  for example 3.

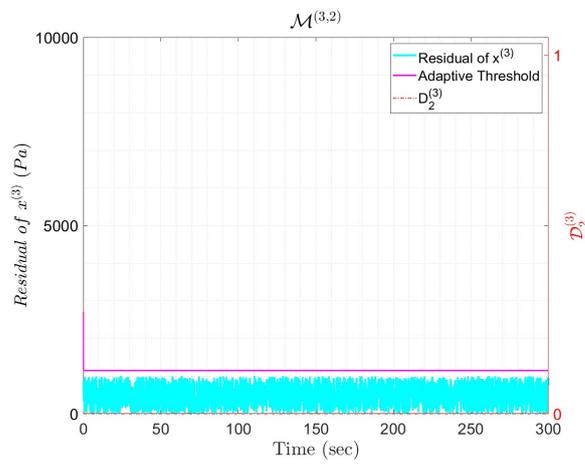


Figure A.25: Simulation results for the output of  $\mathcal{M}^{(3,2)}$  for example 3.

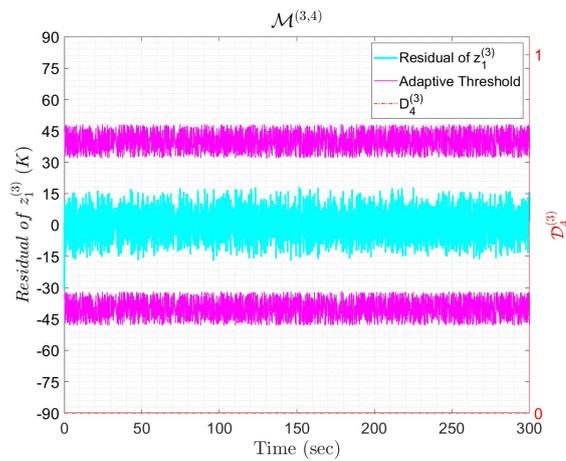
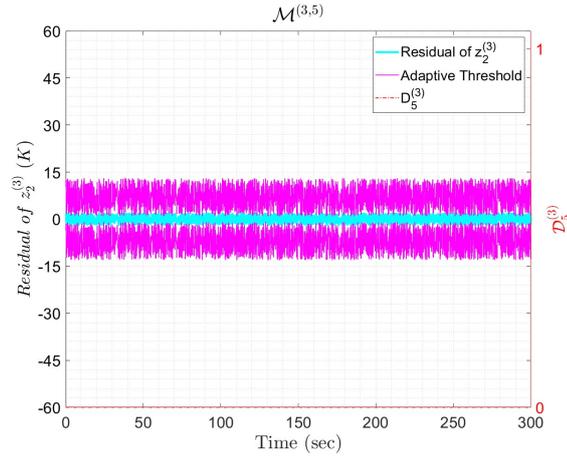
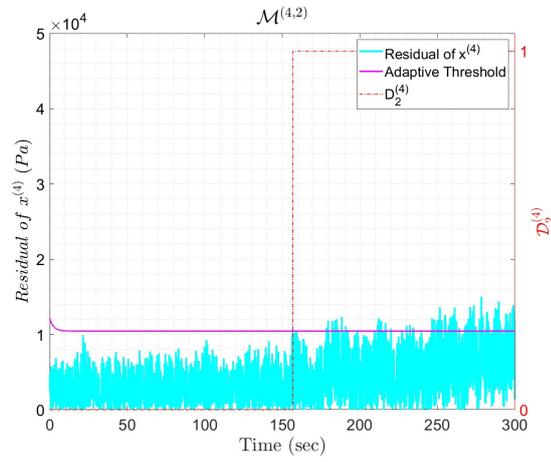
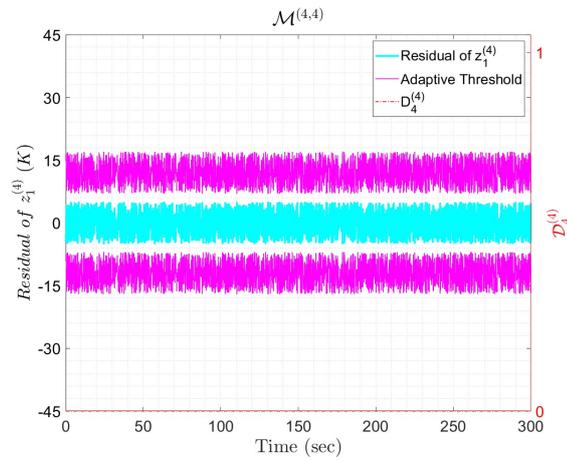


Figure A.26: Simulation results for the output of  $\mathcal{M}^{(3,4)}$  for example 3.

Figure A.27: Simulation results for the output of  $\mathcal{M}^{(3,5)}$  for example 3.Figure A.28: Simulation results for the output of  $\mathcal{M}^{(4,2)}$  for example 3.Figure A.29: Simulation results for the output of  $\mathcal{M}^{(4,4)}$  for example 3.

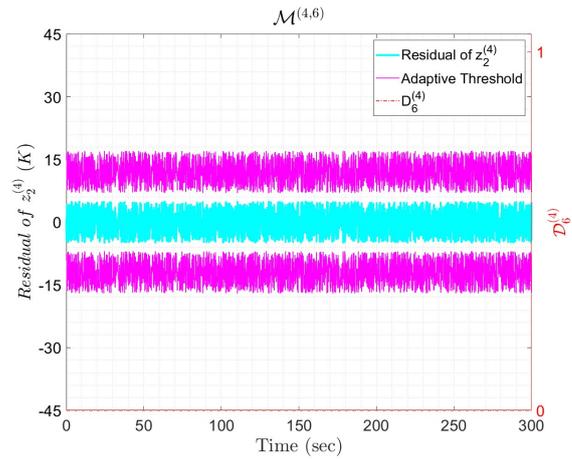


Figure A.30: Simulation results for the output of  $\mathcal{M}^{(4,6)}$  for example 3.

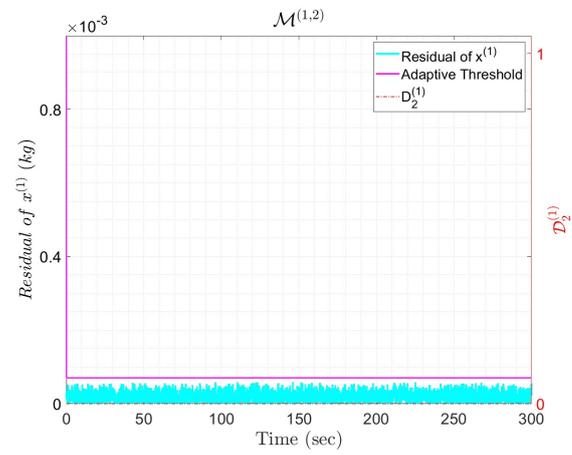


Figure A.31: Simulation results for the output of  $\mathcal{M}^{(1,2)}$  for example 4.

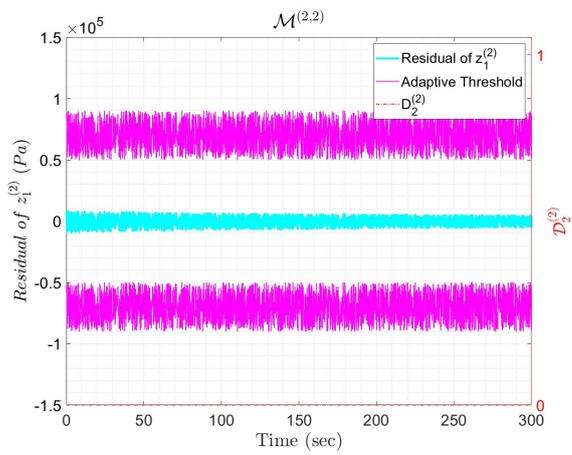
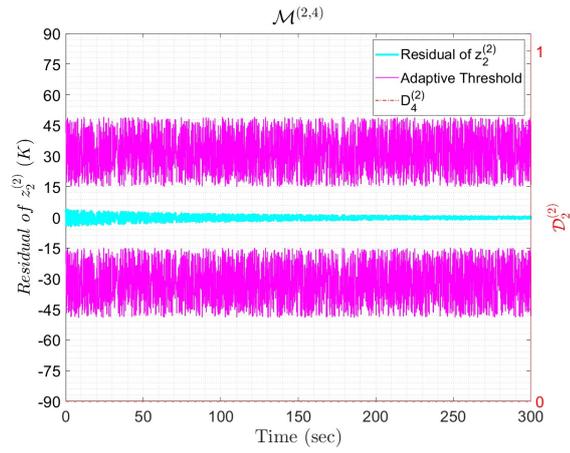
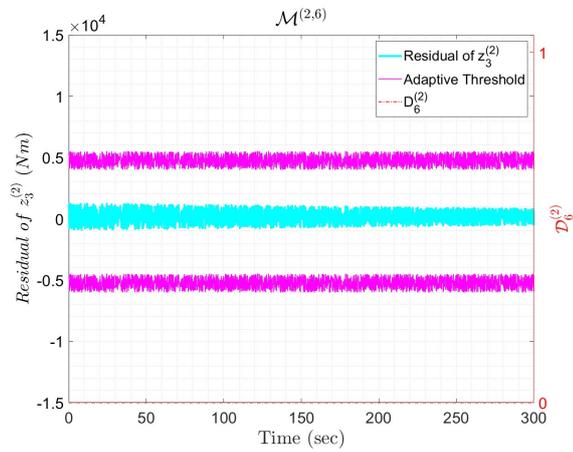
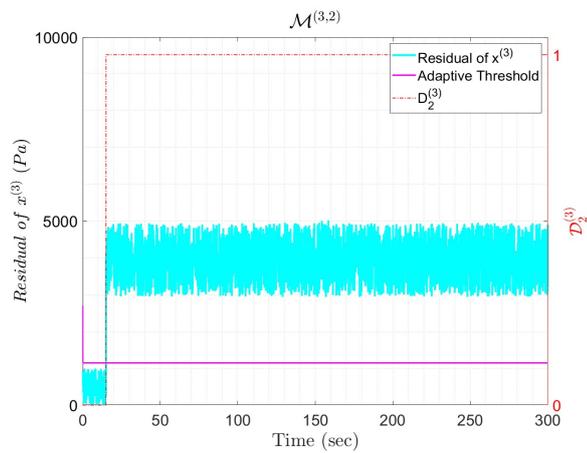


Figure A.32: Simulation results for the output of  $\mathcal{M}^{(2,2)}$  for example 4.

Figure A.33: Simulation results for the output of  $\mathcal{M}^{(2,4)}$  for example 4.Figure A.34: Simulation results for the output of  $\mathcal{M}^{(2,6)}$  for example 4.Figure A.35: Simulation results for the output of  $\mathcal{M}^{(3,2)}$  for example 4.

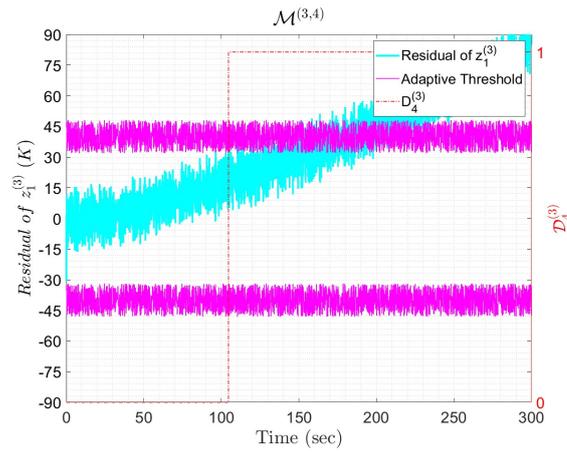


Figure A.36: Simulation results for the output of  $\mathcal{M}^{(3,4)}$  for example 4.

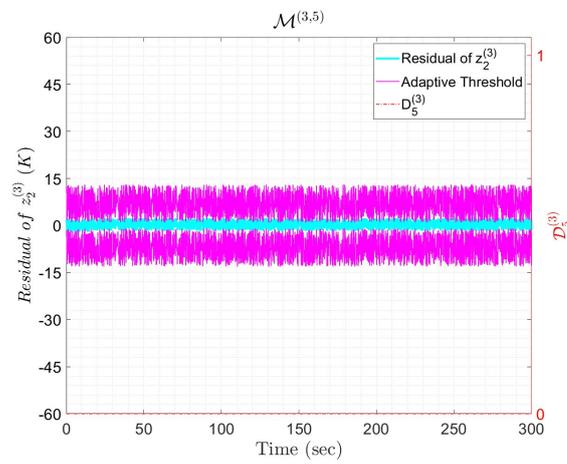


Figure A.37: Simulation results for the output of  $\mathcal{M}^{(3,5)}$  for example 4.

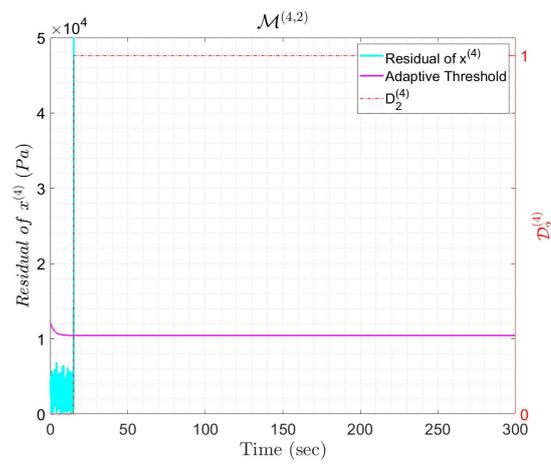


Figure A.38: Simulation results for the output of  $\mathcal{M}^{(4,2)}$  for example 4.

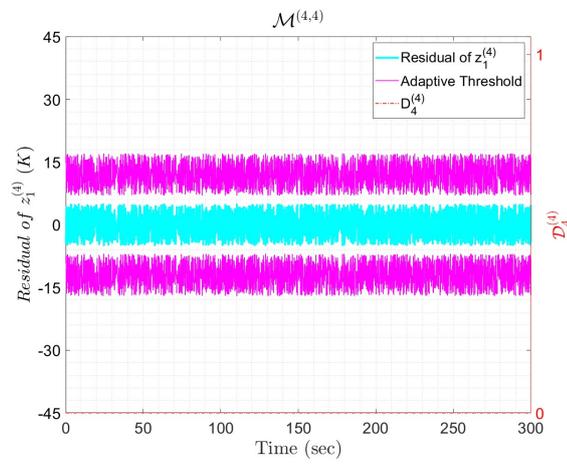


Figure A.39: Simulation results for the output of  $\mathcal{M}^{(4,4)}$  for example 4.

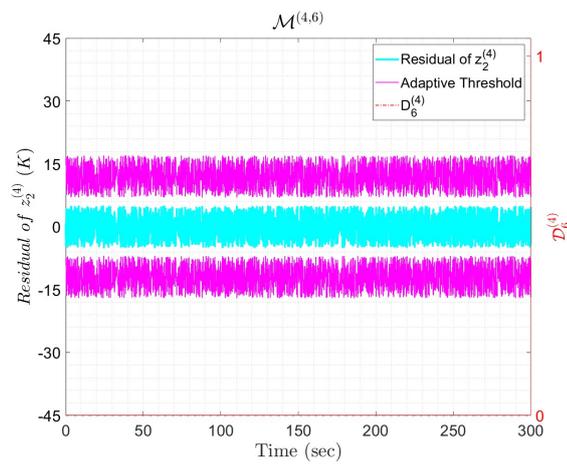


Figure A.40: Simulation results for the output of  $\mathcal{M}^{(4,6)}$  for example 4.

# B

## Appendix: Seiliger Cycle

Along with a gas exchange model, the Seiliger cycle is a suitable method to derive the in-cylinder process [13]. The cycle consists of a polytropic compression, a constant volume combustion, an isobaric combustion, an isothermal combustion, and a polytropic expansion. The work produced during the closed cylinder process can therefore be determined, and the exhaust gas properties at the end of each expansion can be specified. The gas is assumed to be ideal, and in a homogeneous composition [24].

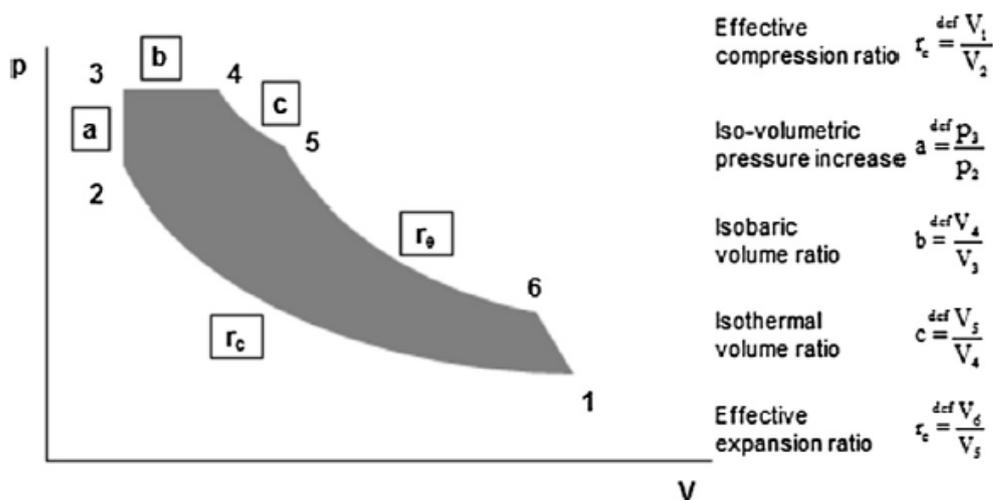


Figure B.1: The six-point Seiliger cycle [24].

The typical six-point Seiliger or dual cycle in a pressure (p) – volume (V) plot, where one may view the compression (1–2), the isochoric combustion (2–3), the isobaric combustion (3–4), the isothermal combustion (4–5) and the polytropic expansion (5–6) in fig. B.1 [24]. The associated equations are shown in table B.1 [62] where  $V_i$ ,  $p_i$  and  $T_i$  are the volume in  $m^3$ , pressure in  $Pa$  and temperature in  $K$  at state  $i$ ,  $w_{ij}$  and  $q_{ij}$  denote the specific work in  $kNm/kg$  and specific heat in  $kJ/kg$  produced during the process from state  $i$  to state  $j$ ,  $a$ ,  $b$  and  $c$  are the Seiliger parameters as expressed in [62],  $c_{p,a}$  and  $c_{v,a}$  represent the specific heat capacity at constant pressure and constant volume for air in  $J/kgK$  respectively,  $n_{exp}$  is the polytropic exponent for expansion, as polytropic expansion permits for jacket water cooling, and  $r_{eo}$  denotes the ratio of the volume at Seiliger point 6, when the exhaust valve opens, to point 1, when the inlet valve shuts, which is determined by the exhaust valve opening angle  $\alpha_{EO}$  [24]. The geometric characteristics of the cylinder can be viewed in fig. B.2 from which  $r_{eo}$  and other cylinder features can be derived.

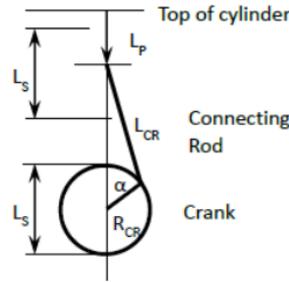


Figure B.2: Schematic view of the geometry of cylinder, crank rod and crankshaft [24].

$$L_{EO} = L_S \left[ \frac{1}{\epsilon_c - 1} + \frac{1}{2} (1 - \cos \alpha_{EO} + \frac{1}{\lambda_{CR}} (1 - r_{tg})) \right] \quad (\text{B.1})$$

$$L_{IC} = L_S \left[ \frac{1}{\epsilon_c - 1} + \frac{1}{2} (1 - \cos \alpha_{IC} + \frac{1}{\lambda_{CR}} (1 - r_{tg})) \right] \quad (\text{B.2})$$

$$\lambda_{CR} = \frac{L_S}{2L_{CR}} \quad (\text{B.3})$$

$$r_{tg} = \sqrt{1 - \lambda_{CR}^2 \sin^2 \alpha_{IC}} \quad (\text{B.4})$$

$$r_{eo} = \frac{L_{EO}}{L_{IC}} \quad (\text{B.5})$$

$$r_c = (\epsilon_c - 1)x_c + 1 \quad (\text{B.6})$$

$$x_c = \frac{L_{IC}}{L_{BDC}} \quad (\text{B.7})$$

$$L_{BDC} = \frac{\epsilon L_S}{\epsilon - 1} \quad (\text{B.8})$$

Where  $L_S$  denotes the stroke length in  $m$  and  $r_c$  is the effective compression ratio [24].  $L_{EO}$  denotes the cylinder space length when the exhaust valve opens. The term  $\epsilon_c$  symbolizes the geometric compression ratio, determined by the cylinder dimensions. The total specific work  $w_i$  in  $kJ/kg$  can then be derived from the work of the Seiliger stages in table B.1, as follows [24]:

$$w_i = w_{12} + w_{34} + w_{45} + w_{56} \quad (\text{B.9})$$

Table B.1: Seiliger cycle thermodynamic relations and ratios [62].

Process	Volume (V)	Pressure (p)	Temperature (T)	Specific Work (w)	Heat Release (q)
1-2	$\frac{V_1}{V_2} = r_c$	$\frac{p_2}{p_1} = r_c^{\kappa_a}$	$\frac{T_2}{T_1} = r_c^{(\kappa_a - 1)}$	$w_{12} = \frac{R_a(T_2 - T_1)}{\kappa_a - 1}$	-
2-3	$\frac{V_3}{V_2} = 1$	$\frac{p_3}{p_2} = a$	$\frac{T_3}{T_2} = a$	-	$q_{23} = c_{v,a}(T_3 - T_2)$
3-4	$\frac{V_4}{V_3} = b$	$\frac{p_4}{p_3} = 1$	$\frac{T_4}{T_3} = b$	$w_{34} = R_a(T_4 - T_3)$	$q_{34} = c_{p,a}(T_4 - T_3)$
4-5	$\frac{V_5}{V_4} = c$	$\frac{p_5}{p_4} = c$	$\frac{T_5}{T_4} = 1$	$w_{45} = R_a T_4 \ln c$	$q_{45} = R_a T_4 \ln c$
5-6	$\frac{V_6}{V_5} = \frac{r_{eo} r_c}{b \cdot c}$	$\frac{p_6}{p_5} = \left( \frac{r_{eo} r_c}{b \cdot c} \right)^{n_{exp}}$	$\frac{T_6}{T_5} = \left( \frac{r_{eo} r_c}{b \cdot c} \right)^{n_{exp} - 1}$	$w_{56} = \frac{R_a(T_6 - T_5)}{n_{exp} - 1}$	-

The diesel engine model is run at the FAT speed and power settings with the parameters from table 4.1. The simulated model results are presented below in figs. B.3, B.4, and B.5 for cylinder temperature, engine torque, and in-cylinder pressure for Seiliger cycle stage 6, respectively.

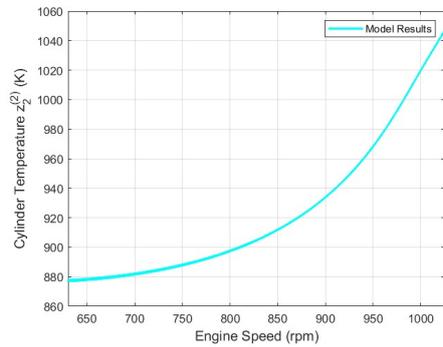


Figure B.3: Cylinder temperature vs engine speed.

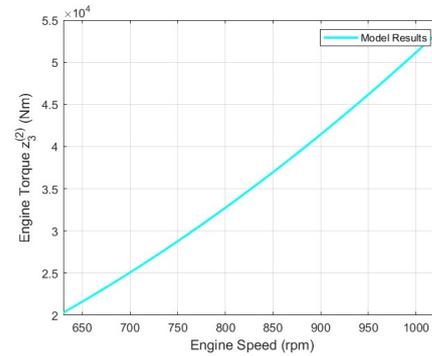


Figure B.4: Engine torque vs engine speed.

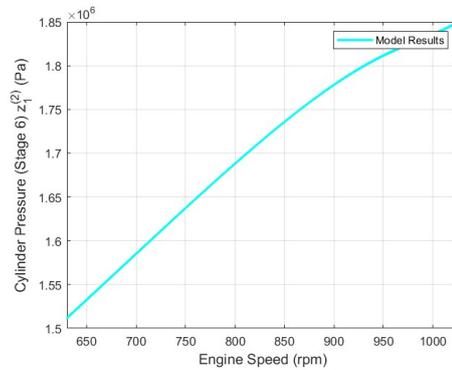
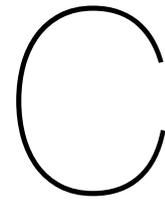


Figure B.5: Combustion pressure at stage 6 vs engine speed.





## Appendix: Academic Research Paper

In the pages that follow one may review the academic research paper that was written as part of this thesis. It was an effort carried out that sums up the work of the thesis in a concise manner.

# An integrated sensor and process fault diagnosis framework for marine vessels

**Patroclus K. Pantela, Vasso Reppa and Nikos Kougiatsos**

Department of Maritime and Transport Technology, Delft University of Technology, 2628CD, the Netherlands.

Email: [p.pantela@student.tudelft.nl](mailto:p.pantela@student.tudelft.nl), [v.reppa@tudelft.nl](mailto:v.reppa@tudelft.nl), [n.kougiatsos@tudelft.nl](mailto:n.kougiatsos@tudelft.nl)

**Abstract**—This research work tries to provide a model-based, distributed Fault Diagnosis (FD) framework that will eventually act as an early warning system for online monitoring of the marine propulsion process, in order to avoid future failures and accidents. Furthermore, the introduction of adaptive thresholds in the monitoring modules of the monitoring agents helps minimize false alarms and reduces the conservativeness in decision-making. Finally, this work comes to provide an integrated sensor and process FD framework that can be adopted on-board marine vessels in the future, and narrow down the gap, regarding FD for marine vessels which is closely linked with maritime safety and maintenance decision-making.

**Index Terms**—Fault detection and isolation, sensor and process faults, nonlinear interconnected systems, distributed fault diagnosis.

## I INTRODUCTION

In the next decade, waterborne trade demand is estimated to increase by 26%, while at the same time a significant shift towards decarbonization and digitalization will be made [1]. Marine vessels are essential for mapping oceans, conducting offshore operations, and shipping activities and they are a prerequisite for humans to be able to harvest ocean-based resources [2]. As a result, there is a great motivation for classification societies and government associations worldwide, to monitor and maintain the active fleet.

There are two brought method categories for FD, named model-free methods and model-based ones. The former detects faults by comparing the results of measurements with predictions of mathematical simulation algorithms. For most model-based methods, frequency and time-domain signal processing technologies are used to obtain “signatures” or residuals which can indicate an expected or faulty behavior [3]. FD evolves the detection of faults in a detailed manner, providing information such as the extent, location, and time of detected faults [4]. The prescriptions provided to the users are accompanied by the name, time, and location of each defect, the status of the machine, and advice for maintenance [3].

When reviewing the available literature, one may notice that from the methods that can isolate faults and detect them, sensor faults are to a large extent ignored, as very few FD methods focus on sensor fault isolation. When taking marine engines as an example, it is apparent that the different types of marine engines that are operated today require different monitoring setups due to their wide range of sizes, combustion cycles, operating principles, and fuel mixtures, meaning that there is a lack of standardization amongst the FD methods for marine engines. However, the lack of standardization applies to FD methods for other components of the propulsion process in general [5]. From the model-based methods available that utilize observers, the ones that consider nonlinear systems are very limited. Regarding marine vessels, a few model-based Fault Detection and Isolation (FDI) methods for multiple process and sensor faults exist that focus on the

navigation system of a vessel, namely [6] and [7] and they use linear models. As for the marine fuel engines, multiple process fault model-based FDI was conducted by [8], [9], [10], [11], and [12]. Considering the marine propulsion process of marine vessels, the methods that provide a multiple process fault model-based FDI namely [13] and [14] were found. It is important to mention that the aforementioned methods do not consider distributed monitoring agents. The only similar work that was found is namely [15] which concerns the FDI of actuator and sensor faults occurring in the propulsion system of an underwater Remotely Operated Vehicle (ROV).

The main objective of this work is to develop a model-based method that manages to detect and isolate both sensor and process faults in the propulsion process of a marine vessel. This will be proven feasible later on, by simulating part of the propulsion process model, the sensors that monitor it, and the faults that can be activated at any point and affect the process in the case of process faults, or the sensors in the case of sensor faults. The general scope of this work is to tackle the problems that arise in the absence of FD, with the design of a distributed FDI methodology, that establishes the isolation of multiple processes and sensor faults. Furthermore, in this work, the approach of using monitoring agents and adaptive thresholds that will be presented later allows fault detection and reduces conservativeness in decision-making, excluding false alarms. Additionally, the Mean Value First Principle (MVFP) model of the fuel engine, which is part of the marine propulsion process, enables a better generalization ability of the results for different engines, since it allows for the reconfiguration of the used parameters and can be expanded with more systems. This work focuses on systems described by nonlinear Differential Algebraic Equations (DAE) [16]. Since the system is characterized by high complexity and incorporates highly nonlinear interconnected subsystems, a distributed monitoring approach is proposed. Specifically, local monitoring agents are designed, that utilize information from sensor sets to capture the occurrence of the process and sensor faults.

The structure of this research paper is as follows. At first, the problem is described and the need for a model-based, distributed FDI methodology is expressed. The marine propulsion models are provided in a state-space form thereafter, as well as the sensor equations for each subsystem. The process models that were later simulated include some process faults and the sensor equations include sensor faults. Later, the FDI scheme to detect and isolate these process and sensor faults is presented. The simulation results regarding the implementation of the FDI scheme, in a simulation for the detection and isolation of both process and sensor faults, are shown and a multiple fault isolation example is provided. Finally, a conclusion and some remarks regarding this work are given.

## II PROBLEM FORMULATION

A general system is defined by [17]. With that work in mind, considering the heterogeneous dynamics and interconnections of the subsystems in marine fuel engines, the proposed fault diagnosis method is formed assuming

a class of  $N$  nonlinear DAE based interconnected systems  $\Sigma^{(I)}, I = 1, \dots, N$ . Hereafter, the dependence of the signals on time (e.g.,  $x(t)$ ) will be dropped for notational brevity.

$$\Sigma^{(I)} : \begin{cases} \dot{x}^{(I)} = A^{(I)}x^{(I)} + h^{(I)}(x^{(I)}, z^{(I)}, \chi^{(I)}, u^{(I)}) \\ + \gamma^{(I)}(x^{(I)}, z^{(I)}, u^{(I)}) + \eta_x^{(I)} + f_{x,c}^{(I)}(x^{(I)}, u^{(I)}) \\ 0 = \xi^{(I)}(x^{(I)}, z^{(I)}, \chi^{(I)}, u^{(I)}, f_{z,c}^{(I)}) + \eta_z^{(I)} \end{cases} \quad (1)$$

$$(2)$$

Where  $x^{(I)} \in R^{n_I-r_I}$  is the state variable vector,  $z^{(I)} \in R^{r_I}$  is the algebraic variable vector,  $\chi^{(I)} \in R^{k_I}$  are the interconnection variables from the neighbouring systems,  $u^{(I)} \in R^{l_I}$  is the control input vector,  $\gamma^{(I)} : R^{n_I-r_I} \times R^{l_I} \rightarrow R^{n_I-r_I}$  expresses the known nonlinear system dynamics,  $h^{(I)} : R^{n_I-r_I} \times R^{r_I} \times R^{k_I} \times R^{l_I} \rightarrow R^{n_I-r_I}$  denotes the known interconnection dynamics with the neighbouring subsystems,  $\eta^{(I)} \in R^{n_I-r_I}, \eta^{(I)} \in R^{r_I}$  define the system disturbances, and  $\xi^{(I)} : R^{n_I} \times R^{k_I} \times R^{l_I} \rightarrow R^{n_I-r_I}$  is a smooth vector field. The term  $A^{(I)}x^{(I)}$  describes the linear part of the system's  $\Sigma^{(I)}$  dynamics, where  $A^{(I)} \in R^{(n_I-r_I) \times (n_I-r_I)}$  is the linearized part of the state equation [16]. Later, the model equations corresponding to (1) and (2), for each system, will be provided in more detail. The terms  $f_{x,c}^{(I)}$  and  $f_{z,c}^{(I)}$  in equations (1) and (2) respectively, define process faults that are affecting each state variable or algebraic variable respectively.

Each system incorporates a set of sensors  $S^{(I)} = \cup_{j=1}^{n_I} S^{(I)}\{j\}$  described as follows:

$$S^{(I)} : \begin{cases} y_x^{(I)} = x^{(I)} + d_x^{(I)} + f_x^{(I)} \\ y_z^{(I)} = z^{(I)} + d_z^{(I)} + f_z^{(I)} \end{cases} \quad (3)$$

Where  $y_x^{(I)} \in R^{n_I-r_I}$  denotes the sensor values corresponding to state variables,  $y_z^{(I)} \in R^{r_I}$  defines the sensor values corresponding to algebraic variables,  $d_x^{(I)} \in R^{n_I-r_I}, d_z^{(I)} \in R^{r_I}$  express the measurement noise vectors and  $f_x^{(I)} \in R^{n_I-r_I}, f_z^{(I)} \in R^{r_I}$  denote sensor fault vectors [16]. Each sensor fault vector is given by  $f^{(I)}(t) = [f_x^{(I)}(t) \ f_z^{(I)}(t)]^T = [f_1^{(I)}(t), \dots, f_{n_I}^{(I)}(t)]^T$ , where  $f_j^{(I)}(t) \forall j \in \{1, \dots, n_I\}$  denotes the change in the output caused by a fault in the  $j$ -th sensor. Permanent, abrupt, offset faults can be modeled as follows [18]:

$$f_j^{(I)} = \begin{cases} 0, & t < T_{f_j}^{(I)} \\ \phi_j^{(I)}, & t \geq T_{f_j}^{(I)} \end{cases} \quad \forall j \in \{1, \dots, n_I\} \quad (4)$$

Specifically, in the system equation (1) and (2) the additive and multiplicative process faults were introduced.  $T_{f_j}^{(I)}$  is the time instant of occurrence of the  $j$ -th fault,  $\phi_j^{(I)}$  is associated to a fault function. Note that multiple faults may occur simultaneously or sequentially, for example,  $T_{f_1}^{(I)} \leq T_{f_2}^{(I)} \leq \dots \leq T_{f_j}^{(I)}$  [19]. The fault vectors for the additive process faults there are given as  $f_c^{(I)}(t) = [f_{x,c}^{(I)}(t) \ f_{z,c}^{(I)}(t)]^T$ , while the multiplicative process fault was described by  $f_u^{(I)}(t)$ . Such permanent incipient, drift-like faults can be modeled as follows:

$$f_c^{(I)} = \begin{cases} 0, & t < T_{f_c}^{(I)} \\ \left(1 - e^{-\kappa_c^{(I)}(t-T_{f_c}^{(I)})}\right) \phi_c^{(I)}(t-T_{f_c}^{(I)}), & t \geq T_{f_c}^{(I)} \end{cases} \quad (5)$$

Specifically, in the system equation (1) and (2) the additive and multiplicative process faults were introduced.  $T_{f_c}^{(I)}$  is the time instant of occurrence of the process fault that affects a specific component  $c$  of the propulsion process,  $\phi_c^{(I)}$  is associated to the fault function of each process fault, and  $\kappa_c^{(I)}$  is associated to a fault evolution mode.

In this work, the inclusion of the process faults in the system equations has introduced the additional complexity of distinguishing the process faults from the sensor faults, after their detection. The discrimination between

process and sensor faults can be achieved with the introduction of redundant sensors. In brief, if two or more sensors that are responsible to measure the same variable in a system, show a significant change in the residual that leads to the violation of the associated ARRs, causing the detection of a fault, then this fault is assigned as a process fault. More details will be explained regarding the rationale for adding redundant sensors when the local decision logic is described.

The objective of this thesis is to design a methodology for the detection and isolation of multiple permanent, abrupt or incipient, offset or drift-like process faults such as the ones given in (5), but also permanent, abrupt, offset sensor faults given in (4) for nonlinear DAE interconnected systems defined by (1), (2) and (3), subject to the following assumptions [16]:

- **Assumption 1:** The system disturbance and the measurement noise of each sensor are unknown but uniformly bounded, meaning:

$$|\eta_{x_j}^{(I)}| \leq \bar{\eta}_{x_j}^{(I)}, |\eta_{z_j}^{(I)}| \leq \bar{\eta}_{z_j}^{(I)}, |d_j^{(I)}| \leq \bar{d}_j^{(I)} \quad \forall j \in \{1, \dots, n_I\}$$

Where  $\bar{\eta}_{x_j}^{(I)}, \bar{\eta}_{z_j}^{(I)}$  and  $\bar{d}_j^{(I)}$  are known.

- **Assumption 2:** The nonlinear vector fields  $\gamma^{(I)}$  and  $h^{(I)}$  are locally Lipschitz for  $x \in \mathcal{X}, z \in \mathcal{Z}$  for all  $u \in \mathcal{U}$  and  $t \geq 0$  with Lipschitz constants  $\lambda_{\gamma I}$  and  $\lambda_{h I}$  respectively.

### III MARINE ENGINE MODELLING

The rationale of using a finite stage model in the MVFP simulation model is that it is capable to characterize the combustion and therefore heat input with a finite number of stages and its associated parameters and, further, is capable to calculate in a simple way the net work output from all the stages and their parameters. Then, the in-cylinder process can be transformed to engine cycle time scale effortlessly and efficiently. Therefore, the key point to be able to apply a finite Seiliger-type process definition in an MVFP engine simulation environment is to obtain models for the Seiliger parameters [20]. In Fig. 1 the various systems that constitute the marine propulsion process of a large vessel can be seen with their sensors.

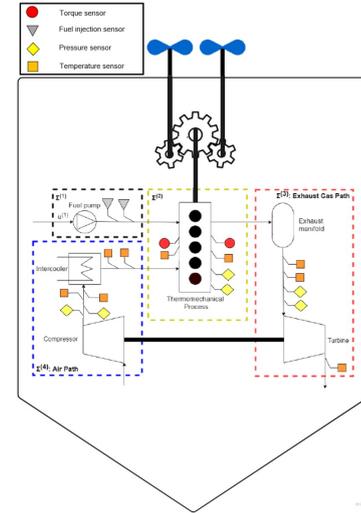


Fig. 1: Illustration of the marine propulsion systems [16].

#### A Fuel Pump Model

The diesel fuel-injection system comprises an injection pump, delivery pipes and fuel injector nozzles, the governor, and a timing device [21]. Fuel

injection time delay in this model is considered constant due to its small value, while a better estimate is achieved when using the engine speed as feedback in a control loop [22].

$$\Sigma^{(1)} : \dot{x}^{(1)} = -\frac{1}{\tau_X} x^{(1)} + \frac{x_{nom}^{(1)}}{\tau_X} f_u^{(1)} u^{(1)} \quad (6)$$

Where  $x^{(1)} \in R$  is the amount of fuel injected per cylinder per engine cycle in  $kg$ ,  $x_{nom}^{(1)} \in R$  is the same quantity under nominal engine conditions,  $u^{(1)} \in R$  is the fuel injection setting in %, and  $\tau_X = \frac{1}{4n_{fe}^{nom}}$  defines

the fuel injection time delay in  $sec$ . The term  $f_u^{(1)}$  represents the actuator fault regarding the seizing of the fuel injection valve in %. The nominal fuel injection amount  $x_{nom}^{(1)} \in R$  is [16]:

$$x_{nom}^{(1)} = \frac{SFC^{nom} P_{fe}^{nom} k_e}{i_e n_{fe}^{nom}} \quad (7)$$

The term  $n_{fe}^{nom}$  represents the nominal rotational engine speed in  $rev/s$ ,  $SFC^{nom}$  is the nominal fuel consumption of the engine in  $kg/J$ , while  $P_{fe}^{nom}$  depicts the nominal power output of the no engine in  $W$ ,  $i_e$  describes the number of engine cylinders and  $k_e$  corresponds the number of crank revolutions per engine cycle ( $k_e = 1$  for a 2-stroke engine and  $k_e = 2$  for a 4-stroke engine). The output of the fuel injection sensor  $y^{(1)} \in R$  is described by [16]:

$$S^{(1)} : y_x^{(1)} = x^{(1)} + d_x^{(1)} + f_x^{(1)} \quad (8)$$

### B Engine Block Model

The diesel engine system has three algebraic variables, namely the pressure  $z_1^{(2)}$  in  $Pa$  and the temperature  $z_2^{(2)}$  in  $K$  inside the engine's cylinders and the engine's shaft torque  $z_3^{(2)}$  in  $Nm$ . The expression of the system is [16]:

$$\Sigma^{(2)} : 0 = \begin{bmatrix} z_1^{(2)} - \xi_{z_1}^{(2)}(x^{(1)}, x^{(4)}, z_1^{(4)}, f_{z_1 V_1}^{(2)}) \\ z_2^{(2)} - \xi_{z_2}^{(2)}(x^{(1)}, x^{(4)}, z_1^{(4)}, f_{z_2 V_1}^{(2)}) \\ z_3^{(2)} - \xi_{z_3}^{(2)}(x^{(1)}, z_3^{(2)}, x^{(4)}, z_1^{(4)}, f_{z_3 V_1}^{(2)}) \end{bmatrix} \quad (9)$$

Where the functions  $\xi_{z_1}^{(2)}$ ,  $\xi_{z_2}^{(2)}$ ,  $\xi_{z_3}^{(2)}$  can be modelled using the Seilinger thermodynamic cycle.

$$\xi_{z_1}^{(2)} = x^{(4) r_c^{\kappa_a}} \left( 1 + \frac{\frac{1}{c_{v,a}} \left( X_{cv} \frac{\eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} \right)}{z_1^{(4)} r_c^{\kappa_a - 1}} \right)^{-n_{exp}}$$

$$\exp \left( \frac{r_{eo} r_c}{(1 - X_{cv} - X_{ct}) \frac{\eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} + \frac{c_{p,a} \left( z_1^{(4)} r_c^{\kappa_a - 1} + \frac{X_{cv} \eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)} c_{v,a}} \right)}{(n_{exp} - 1) \frac{X_{ct} \eta h_L R_c^{(1-\kappa_a)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}}} \right)$$

$$\exp \left( \frac{(n_{exp} - 1) \frac{X_{ct} \eta h_L R_c^{(1-\kappa_a)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}}}{\left( 1 + \frac{X_{cv} \eta h_L R_c^{(1-\kappa_a)} x^{(1)}}{c_{v,a} (V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} + (1 - X_{cv} - X_{ct}) \frac{\eta h_L R_a r_c^{(1-\kappa_a)} x^{(1)}}{c_{p,a} (V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} \right)} \right) \quad (10)$$

$$\xi_{z_2}^{(2)} = z_1^{(4) r_c^{\kappa_a - 1}} \left( \frac{r_{eo} r_c}{(1 - X_{cv} - X_{ct}) \frac{\eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} + \frac{c_{p,a} \left( z_1^{(4)} r_c^{\kappa_a - 1} + \frac{X_{cv} \eta h_L R_a z_1^{(4)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)} c_{v,a}} \right)}{(n_{exp} - 1) \frac{X_{ct} \eta h_L R_c^{(1-\kappa_a)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}}} \right)^{1-n_{exp}} \quad (11)$$

$$\exp \left( \frac{(n_{exp} - 1) \frac{X_{ct} \eta h_L R_c^{(1-\kappa_a)} x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}}}{\left( 1 + \frac{X_{cv} \eta h_L R_c^{(1-\kappa_a)} x^{(1)}}{c_{v,a} (V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} + (1 - X_{cv} - X_{ct}) \frac{\eta h_L R_a r_c^{(1-\kappa_a)} x^{(1)}}{c_{p,a} (V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} \right)} \right)$$

$$\xi_{z_3}^{(2)} = \frac{(V_1 + f_{z_1 V_1}^{(2)}) i_e x^{(4)}}{2\pi k_e} \cdot \left( \frac{r_c^{\kappa_a - 1} - 1}{\kappa_a - 1} + \frac{(1 - X_{cv} - X_{ct}) \eta h_L R_a x^{(1)}}{c_{p,a} (V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} - \frac{r_c^{\kappa_a - 1} + \frac{\eta h_L R_a x^{(1)} (c_{p,a} X_{cv} + c_{v,a} (1 - X_{cv} - X_{ct}))}{c_{v,a} c_{p,a} (V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}}}{n_{exp} - 1} + \frac{X_{ct} \eta h_L R_a x^{(1)}}{(V_1 + f_{z_1 V_1}^{(2)}) x^{(4)}} + \frac{\xi_{z_2}^{(2)}}{z_1^{(4)} (n_{exp} - 1)} - Q_{loss}^{nom} \left( 1 + Q_{loss}^{grad} \frac{n_{fe}^{nom} - n_{fe}}{n_{fe}^{nom}} \right) \right) \quad (12)$$

$$n_{fe} = \sqrt{\frac{2\pi}{c} z_3^{(2)}} \quad (13)$$

$$X_{cv} = X_{cv}^{nom} + X_{cv}^{grad} \cdot \left( \frac{n_{fe} - n_{fe}^{nom}}{n_{fe}^{nom}} \right) \quad (14)$$

$$X_{ct} = X_{ct}^{nom} \cdot \left( \frac{x^{(1)}}{x_{nom}^{(1)}} \right) \quad (15)$$

Where  $X_{cv}^{nom}$  denotes the nominal constant volume portion,  $X_{cv}^{grad}$  represents the gradient of the constant volume portion,  $X_{ct}^{nom}$  denotes the nominal constant temperature portion,  $\eta$  is the thermal efficiency incorporating both the combustion and heat release processes,  $h_L$  denotes the lower heating value of fuel at ISO conditions in  $J/kg$ .  $V_1$  depicts the cylinder volume at start of compression in  $m^3$  and  $R_a$  is the universal gas constant of air in  $J/kgK$ . The term  $f_{z_1 V_1}^{(2)}$  represents the additive process fault regarding bore damage. Finally,  $r_c$  is the effective compression ratio.

The output values of this system's pressure, temperature and torque sensors  $y^{(2)} \in R^3$  are described by [16]:

$$S^{(2)} : y_z^{(2)} = z^{(2)} + d_z^{(2)} + f_z^{(2)} \quad (16)$$

### C Exhaust Receiver and Turbocharger Model

This turbine system can be represented by one state variable and two algebraic variables, the temperature before  $z_1^{(3)}$  and after  $z_2^{(3)}$  the turbine in  $K$ . This system is described as follows [16]:

$$\Sigma^{(3)} : \begin{cases} \dot{x}^{(3)} = -\frac{1}{\tau_{p,d}} x^{(3)} + \frac{1}{\tau_{p,d}} f_{x,tur}^{(3)} + h^{(3)} \{x^{(3)}, z^{(3)}, \mathcal{X}^{(3)}\} \\ 0 = \xi^{(3)} \{x^{(3)}, z^{(3)}, \mathcal{X}^{(3)}, f_{z,T_d}^{(3)}\} \end{cases} \quad (17)$$

Where  $\mathcal{X}^{(3)} = [x^{(1)} \ z_1^{(2)} \ x^{(4)} \ z_1^{(4)}]^T$  denote the interconnection variables. The terms  $f_{x,tur}^{(3)}$  and  $f_{z,T_d}^{(3)}$  represent the additive process faults regarding loss

of pressure inside the turbine and loss of lubrication at the turbine shaft respectively.

The interconnection dynamics are defined below [16]:

$$h^{(3)}\{x^{(3)}, z^{(3)}, \mathcal{X}^{(3)}\} = \frac{1}{\tau_{pd}} \cdot \sqrt{p_{ex}^2 + \frac{z_1^{(3)} R_g \left( \psi_1 \cdot \frac{x^{(4)} n_{fe} + x^{(1)} n_{fe} \frac{i_e}{k_e} \right)^2}{\alpha_z^2 A_{eff}^2}} \quad (18)$$

$$\psi_1 = \psi_1\{x^{(3)}, x^{(4)}, z^{(4)}, n_{fe}\} = \frac{i_e V_1 s_{sl}\{x^{(3)}, x^{(4)}, z^{(4)}, n_{fe}\}}{R_a k_e} \quad (19)$$

$$s_{sl}\{x^{(3)}, x^{(4)}, z^{(4)}, n_{fe}\} = s_{sl}^{nom} \cdot \frac{n_{fe}^{nom} m_1^{nom} x^{(4)} \Psi_{sc}\{x^{(3)}, x^{(4)}\}}{n_{fe} \left( \frac{x^{(4)} V_1}{R_a z_1^{(4)}} \right) P_1^{nom} \Psi_{sc}^{nom}} \quad (20)$$

$$\Psi_{sc}\{x^{(3)}, x^{(4)}\} = \sqrt{\frac{2\kappa_g}{\kappa_g - 1}} \cdot \sqrt{\left( \frac{x^{(3)}}{x^{(4)}} \right)^{(2/\kappa_g)} - \left( \frac{x^{(3)}}{x^{(4)}} \right)^{\left( \frac{\kappa_g + 1}{\kappa_g} \right)}} \quad (21)$$

The algebraic part of the subsystem is as follows [16]:

$$\xi^{(3)}\{x^{(3)}, z^{(3)}, \mathcal{X}^{(3)}\} = \begin{bmatrix} z_1^{(3)} - \frac{\psi_2(T_{sl} + f_{z,T_{sl}}^{(3)}) + \psi_3 z_2^{(2)}}{z_2^{(3)} - \psi_4 z_1^{(3)}} \\ \psi_2 + \psi_3 \end{bmatrix} \quad (22)$$

$$\psi_2 = \psi_2\{x^{(3)}, x^{(4)}, z^{(4)}, z^{(2)}\} = \frac{c_{p,a} V_1 x^{(4)} s_{sl}\{x^{(3)}, x^{(4)}, z^{(4)}, z^{(2)}\}}{z_1^{(4)} R_a} \quad (23)$$

$$\psi_3 = \psi_3\{x^{(1)}, x^{(4)}, z^{(4)}\} = c_{p,g} \left( x^{(1)} + \frac{V_1 x^{(4)}}{R_a z_1^{(4)}} \right) \quad (24)$$

$$\bar{\psi}_3 = \bar{\psi}_3\{x^{(1)}, x^{(4)}, z^{(4)}, z^{(2)}\} = \left( \frac{1}{n_{bld}} + \tau_{pd} \frac{(n_{bld} - 1) h^{(3)}}{n_{bld} z_1^{(2)}} \right) \psi_3 \quad (25)$$

$$\psi_4 = \psi_4\{x^{(3)}, x^{(4)}\} = 1 + \eta_{turbo}\{x^{(4)}\} \cdot (\Pi_{turbo} - 1) \quad (26)$$

$$\Pi_{turbo} = \Pi_{turbo}\{x^{(3)}\} = \left( \frac{p_{ex}}{x^{(3)}} \right)^{\left( \frac{\kappa_g - 1}{\kappa_g} \right)} \quad (27)$$

$$\eta_{turbo}\{x^{(4)}\} = a_{turbo} + b_{turbo} x^{(4)} + c_{turbo} (x^{(4)})^2 \quad (28)$$

Where  $\tau_{pd}$  expresses the time delay for filling the exhaust receiver in *sec*,  $p_{ex}$  denotes the pressure after the turbocharger in *Pa* assumed equal to the atmospheric pressure,  $\alpha_z$  is the Zinner turbine area decrease factor assumed 1 for a constant pressure turbocharger,  $A_{eff}$  is the turbine's effective area in  $m^2$ ,  $R_g$  is the universal gas constant of the exhaust gas in  $J/kgK$ ,  $n_{bld}$  denotes the polytropic expansion coefficient of blowdown,  $c_{p,g}$  represents the specific heat capacity at constant pressure for the exhaust gas in  $J/kgK$ ,  $s_{sl}$  defines the total slip ratio of the engine expressed in [22],  $T_{sl}$  is the temperature of the air slip during scavenging in *K* and  $a_{turbo}$ ,  $b_{turbo}$ ,  $c_{turbo}$  denote the polynomial coefficients of the isentropic turbine efficiency and  $\kappa_g$  expresses the specific heat ratio of the exhaust gas [16].

The output values of this system's pressure and temperature sensors  $y^{(3)} \in R^3$  are described by [16]:

$$S^{(3)} : \begin{bmatrix} y_x^{(3)} \\ y_z^{(3)} \end{bmatrix} = \begin{bmatrix} x^{(3)} \\ z^{(3)} \end{bmatrix} + \begin{bmatrix} d_x^{(3)} \\ d_z^{(3)} \end{bmatrix} + \begin{bmatrix} f_x^{(3)} \\ f_z^{(3)} \end{bmatrix} \quad (29)$$

#### D Air Intake Model

The air intake system is expressed by one state variable, the charge air pressure after the compressor  $x^{(4)}$  in *Pa*, and two algebraic variables, namely the temperatures before  $z_1^{(4)}$  and after  $z_2^{(4)}$  the intercooler in *K* [16].

$$\Sigma^{(4)} : \begin{cases} \dot{x}^{(4)} = -\frac{1}{\tau_{TC}} x^{(4)} + \frac{1}{\tau_{TC}} f_{x,ini}^{(4)} + h^{(4)}\{x^{(4)}, z^{(4)}, \mathcal{X}^{(4)}\} \\ 0 = \xi^{(4)}\{x^{(4)}, z^{(4)}, \mathcal{X}^{(4)}, f_{z,T_c}^{(4)}, f_{z,T_{inl}}^{(4)}\} \end{cases} \quad (30)$$

Where  $\mathcal{X}^{(4)} = [x^{(1)} \ x^{(3)} \ z^{(3)}]^T$  denote the interconnection variables. The terms  $f_{x,ini}^{(4)}$ ,  $f_{z,T_c}^{(4)}$  and  $f_{z,T_{inl}}^{(4)}$  are additive process faults concerning the loss

of pressure inside the compressor, blockage of the heat-exchanger of the intercooler and loss of lubrication at the shaft of the compressor respectively.

The interconnection dynamics are defined below [16]:

$$h^{(4)}\{x^{(4)}, z^{(4)}, \mathcal{X}^{(4)}\} = \frac{p_{amb}}{\tau_{TC}} \cdot$$

$$\left( 1 + \chi_g \delta_f \eta_{TC}\{x^{(4)}\} r_{TC}\{z^{(3)}\} (1 - \Pi_{turbo}) \right) \left( \frac{\kappa_a - 1}{\kappa_a} \right) \quad (31)$$

$$\delta_f = \delta_f\{x^{(4)}, z^{(4)}, \mathcal{X}^{(4)}\} = 1 + \frac{x^{(1)}}{(1 + s_{sl}) \frac{V_1 x^{(4)}}{R_a z_1^{(4)}}} \quad (32)$$

$$\eta_{TC}\{x^{(4)}\} = a_\eta + b_\eta x^{(4)} + c_\eta (x^{(4)})^2 \quad (33)$$

$$r_{TC}\{z^{(3)}\} = \frac{z_1^{(3)}}{T_{amb}} \quad (34)$$

$$\chi_g = \frac{c_{p,g}}{c_{p,a}} \quad (35)$$

The algebraic part of this system is expressed as follows [16]:

$$\xi^{(4)}\{x^{(4)}, z^{(4)}, \mathcal{X}^{(4)}\} = \begin{bmatrix} z_1^{(4)} - \xi_{z_1}^{(4)} \\ z_2^{(4)} - \xi_{z_2}^{(4)} \{x^{(3)}, z^{(3)}\} \end{bmatrix} \quad (36)$$

$$\xi_{z_1}^{(4)} = T_c - \varepsilon_{inl} \left( T_{inl} + f_{z,T_{inl}}^{(4)} - T_c \right) \quad (37)$$

$$\xi_{z_2}^{(4)} = T_{amb} + x^{(3)} \chi_g \eta_{turbo} (\delta_f + \eta_{com}) \left( z_2^{(3)} - z_1^{(3)} \right) + f_{z,T_c}^{(4)} \quad (38)$$

Where  $\tau_{TC}$  denotes the compressor time delay in *sec*,  $p_{amb}$  represents the ambient air pressure in *Pa*,  $T_{amb}$  is the ambient air temperature in *K*,  $a_\eta$ ,  $b_\eta$ ,  $c_\eta$  represent the polynomial coefficients of the turbocharger for estimating its efficiency,  $\eta_{com}$  denotes the mechanical efficiency of the compressor that can be considered constant,  $T_c$  is the charge air temperature after the intercooler in *K*,  $\varepsilon_{inl}$  describes the parasitic effectiveness of the heat exchange between the inlet duct and the air, while  $T_{inl}$  depicts the temperature of the inlet duct that heats the inducted air in *K* [16].

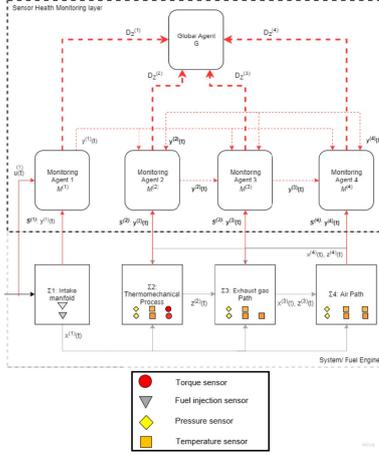
The output values of this system's pressure and temperature sensors  $y^{(4)} \in R^3$  are described by [16]:

$$S^{(4)} : \begin{bmatrix} y_x^{(4)} \\ y_z^{(4)} \end{bmatrix} = \begin{bmatrix} x^{(4)} \\ z^{(4)} \end{bmatrix} + \begin{bmatrix} d_x^{(4)} \\ d_z^{(4)} \end{bmatrix} + \begin{bmatrix} f_x^{(4)} \\ f_z^{(4)} \end{bmatrix} \quad (39)$$

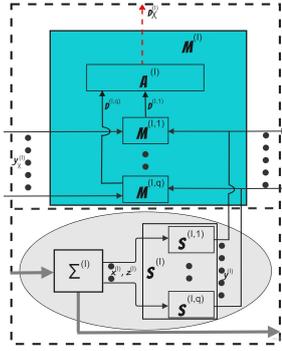
## IV DISTRIBUTED FAULT DETECTION AND ISOLATION

A successful FDI scheme should be robust concerning modeling errors and sensitive concerning incipient faults. This trade-off between robustness and sensitivity has drawn attention over the years in the research society [23].

*Remark 1:* The distributed scheme in Fig. 2, is based on previous works [16], [18]. As shown in Fig. 2, for each one of the interconnected systems,  $\Sigma^{(I)} \forall I \in \{1, \dots, N\}$  with  $N = 4$ , a monitoring agent  $\mathcal{M}^{(I)}$  is designed comprising  $n_I$  monitoring modules  $\mathcal{M}^{(I,q)} \forall q \in \{1, \dots, n_I\}$  with  $n_1 = 2, n_2 = n_4 = 6, n_3 = 5$ . The monitoring agents have been created considering the DAE nature of the fuel engine and nonlinear algebraic residuals with their associated adaptive thresholds were proposed [16]. Additionally, in fig. 3 the illustration of the  $I$ -th monitoring agent, system  $\Sigma^{(I)}$ , and its sensors  $\mathcal{S}^{(I)}$  can be viewed. One may notice the content of the agent  $\mathcal{M}^{(I)}$  which is designed comprising  $n_I$  monitoring modules  $\mathcal{M}^{(I,q)}$ , and an aggregation module  $\mathcal{A}^{(I)}$ . The sensors that are connected to  $\Sigma^{(I)}$ , are organized in  $n_I$  sensor sets  $\mathcal{S}^{(I,q)}$ . The outputs of these sensor sets are fed to the monitoring modules that utilize them, along with the outputs of the interconnection sensors to form ARR. More details regarding the use of these ARR by the modules and the need for the aggregation module  $\mathcal{A}^{(I)}$ , will be provided in the local decision logic later



**Fig. 2:** Distributed FDI scheme application using a MVFP marine propulsion model [16].



**Fig. 3:** Description of monitoring agent  $\mathcal{M}^{(l)}$ , system  $\Sigma^{(l)}$ , and sensors  $S^{(l)}$  connected to it..

### A Residual Generation and Error Estimation

This subsection deals with the design of the module  $\mathcal{M}^{(l,q)}$ ,  $q \in \{1, \dots, n_l\}$ . The estimation model of  $\mathcal{M}^{(l,q)}$  is formed by selecting a Lipschitz nonlinear observer  $\mathcal{O}^{(l,q)}$ , shown in the following equation [19]:

$$\begin{aligned} \mathcal{O}^{(l,q)} : \dot{\hat{x}}^{(l,q)} &= A^{(l)} \hat{x}^{(l,q)} + \gamma^{(l)} \left( \hat{x}^{(l,q)}, u^{(l)}, y_z^{(l,q)} \right) + \\ & h^{(l)} \left( \hat{x}^{(l,q)}, u^{(l)}, y_z^{(l,q)}, y_{\chi}^{(l,q)} \right) + L^{(l,q)} \left( y_x^{(l,q)} - \hat{x}^{(l,q)} \right) \end{aligned} \quad (40)$$

Subtracting (1) from (40) yields the following [16]:

$$\dot{\varepsilon}_x^{(l,q)} = A_L^{(l,q)} \varepsilon_x^{(l,q)} + \tilde{\gamma}^{(l,q)} + \tilde{h}^{(l,q)} + \eta_x^{(l,q)} - L^{(l,q)} d_x^{(l,q)} \quad (41)$$

Where  $\hat{x}^{(l,q)} \in R^{n_l}$  is the estimation of  $x^{(l)}$  (based on the sensor measurements  $y^{(l,q)}$  with  $\hat{x}^{(l,q)}(0) \in \mathcal{X}^{(l)}$  that is an open set,  $L^{(l,q)} \in R^{(n_l-r_l) \times (n_l-r_l)}$  is the observer gain matrix is selected such that the matrix  $A_L^{(l,q)} = A^{(l)} - L^{(l,q)}$  is Hurwitz, and  $y_z^{(l,q)}$  is the transmitted sensor information. Furthermore,  $\varepsilon_x^{(l,q)} = x^{(l)} - \hat{x}^{(l,q)}$  is the state estimation error,  $\tilde{\gamma}^{(l,q)} \triangleq \gamma^{(l)}(x^{(l)}, z^{(l)}, u^{(l)}) - \gamma^{(l)}(\hat{x}^{(l,q)}, y_z^{(l,q)}, u^{(l)})$  and  $\tilde{h}^{(l,q)} = h^{(l)}(x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)}) - h^{(l)}(\hat{x}^{(l,q)}, y_z^{(l,q)}, y_{\chi}^{(l,q)}, u^{(l)})$  [16]. Keep in mind that the given observer is based on the formulation of observers for Lipschitz nonlinear systems, which is adjusted appropriately for the nonlinear interconnected subsystems [19].

The residual vector is defined as follows [16]:

$$\varepsilon_y^{(l,q)} = \begin{bmatrix} y_x^{(l,q)} - \hat{x}^{(l,q)} \\ -\xi^{(l)} \left( y_x^{(l,q)}, y_z^{(l,q)}, y_{\chi}^{(l,q)}, u^{(l)} \right) \end{bmatrix} = \begin{bmatrix} \varepsilon_{y_x}^{(l,q)} \\ \varepsilon_{y_z}^{(l,q)} \end{bmatrix} \in R^{n_l} \quad (42)$$

The residual  $\varepsilon_{y_x}^{(l,q)}$  can also be defined as  $\varepsilon_{y_x}^{(l,q)} = \varepsilon_x^{(l,q)} + d_x^{(l,q)}$ . Adding (2) to the expression of  $\varepsilon_{y_z}^{(l,q)}$  in (42) yields the following [16]:

$$\varepsilon_{y_z}^{(l,q)} = \xi^{(l)} \left( x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)} \right) + \eta_z^{(l)} - \xi^{(l,q)} \left( y_x^{(l,q)}, y_z^{(l,q)}, y_{\chi}^{(l,q)}, u^{(l)} \right) \quad (43)$$

### B Adaptive Thresholds Computation

The thresholds are designed to bound the respective residuals  $\varepsilon_{y_x}^{(l,q)}$  and  $\varepsilon_{y_z}^{(l,q)}$  under healthy sensor conditions. The boundedness of  $\varepsilon_{y_x}^{(l,q)}$  and  $\varepsilon_{y_z}^{(l,q)}$  is exploited in order to formulate the adaptive thresholds [18].

$$\left| \varepsilon_{y_{x_j}}^{(l,q)}(t) \right| \leq \bar{\varepsilon}_{y_{x_j}}^{(l,q)}(t) \quad \forall j \in \{1, \dots, n_l - r_l\} \quad (44)$$

$$\varepsilon_{y_{z_j}}^{(l,q)}(t) \in \left[ \underline{\varepsilon}_{y_{z_j}}^{(l,q)}(t), \bar{\varepsilon}_{y_{z_j}}^{(l,q)}(t) \right] \quad \forall j \in \{1, \dots, r_l\} \quad (45)$$

Considering *Assumptions 1-2* and after some mathematical manipulations of (41) the adaptive threshold can be derived as follows [19]:

$$\bar{\varepsilon}_{y_{x_j}}^{(l,q)}(t) = E^{(l,q)}(t) + \rho^{(l,q)} \Lambda_l \int_0^t E^{(l,q)}(\tau) e^{-\xi^{(l,q)}(t-\tau)} d\tau + \bar{d}_{x_j}^{(l,q)} \quad (46)$$

Where

$$E^{(l,q)}(t) = \rho^{(l,q)} e^{-\xi^{(l,q)} t} \bar{x}^{(l,q)} + \frac{\rho_d^{(l,q)} \bar{d}_x^{(l,q)} + \rho^{(l,q)} \lambda_{h_1} \bar{d}_{\chi}^{(l,q)}}{\xi^{(l,q)}} \left( 1 - e^{-\xi_d^{(l,q)} t} \right) \quad (47)$$

$$\Lambda_l = \lambda_{h_1} + \lambda_{\eta} \quad (48)$$

Here  $\bar{d}_x^{(l,q)}$  is the noise bound such that  $|d_x^{(l,q)}| \leq \bar{d}_x^{(l,q)}$  [18]. Moreover,  $\rho^{(l,q)}$ ,  $\xi^{(l,q)}$ ,  $\rho_d^{(l,q)}$  and  $\xi_d^{(l,q)}$  are positive constants, such that the following equations hold:

$$\left| e^{A_L^{(l,q)} t} \right| \leq \rho^{(l,q)} e^{-\xi^{(l,q)} t} \quad (49)$$

$$\left| e^{A_L^{(l,q)} t} L^{(l,q)} \right| \leq \rho_d^{(l,q)} e^{-\xi_d^{(l,q)} t} \quad (50)$$

For the formation of the algebraic thresholds, inclusion functions can be utilized [24]. Given that  $[x^{(l)}] = y_x^{(l,q)} + [d_x^{(l,q)}]$  since  $[x^{(l)}] = [y_x^{(l,q)} - \bar{d}_x^{(l,q)}, y_x^{(l,q)} + \bar{d}_x^{(l,q)}] = y_x^{(l,q)} + [d_x^{(l,q)}] \forall j \in \{1, \dots, n_l - r_l\}$  also  $[z^{(l)}] = y_z^{(l,q)} + [d_z^{(l,q)}]$ ,  $[\chi^{(l)}] = y_{\chi}^{(l,q)} + [d_{\chi}^{(l,q)}]$  and  $[u^{(l)}] = [\underline{u}^{(l)}, \bar{u}^{(l)}]$ , the following can be inferred as follows [16]:

$$\xi_j^{(l)} \{x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)}\} \in [\underline{\xi}_j^{(l,q)}, \bar{\xi}_j^{(l,q)}] \quad \forall j \in \{1, \dots, n_l - r_l\} \quad (51)$$

Where  $[\underline{\xi}_j^{(l,q)}, \bar{\xi}_j^{(l,q)}] = \xi_j^{(l)} \{y_x^{(l,q)} + [d_x^{(l,q)}], y_z^{(l,q)} + [d_z^{(l,q)}], y_{\chi}^{(l,q)} + [d_{\chi}^{(l,q)}], [u^{(l)}]\}$ . Then according to (43) and *Assumption 3*, the following stands [16]:

$$\begin{cases} \varepsilon_{y_{z_j}}^{(l,q)} = \underline{\xi}_j^{(l,q)} - \xi_j^{(l)} \{x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)}\} \\ \varepsilon_{y_{z_j}}^{(l,q)} = \bar{\xi}_j^{(l,q)} - \xi_j^{(l)} \{x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)}\} \end{cases} \quad \forall j \in \{1, \dots, n_l - r_l\} \quad (52)$$

### C Multiple Fault Decision Logic

This subsection provides the fault decision logic. As noted at the beginning of the section, isolation occurs in two steps; the local decision logic and the global decision logic [19]. This shall be shown in detail hereafter.

#### Local Decision Logic

If faults occur in  $\mathcal{S}^{(l,q)}$  they are detected by  $\mathcal{M}^{(l,q)}$  with the use of ARR. The  $j$ -th state-based ARR can be expressed as [16]:

$$\varepsilon_j^{(l,q)} : \left| \varepsilon_{y_{x_j}}^{(l,q)}(t) \right| - \bar{\varepsilon}_{y_{x_j}}^{(l,q)}(t) \leq 0 \quad \forall j \in \{1, \dots, n_l - r_l\} \quad (53)$$

Regarding the monitoring modules that make use of the algebraic residual expression  $\varepsilon_{y_z}^{(l,q)}$  that was given in (42) as well as the threshold expression of (52), the way to create the  $j$ -th ARR is provided [16]:

$$\varepsilon_j^{(l,q)} : \varepsilon_{y_{z_j}}^{(l,q)}(t) \in [\underline{\varepsilon}_{y_{z_j}}^{(l,q)}(t), \bar{\varepsilon}_{y_{z_j}}^{(l,q)}(t)] \quad \forall j \in \{1, \dots, r_l\} \quad (54)$$

The set of ARRs that the module decides on the existence of local faults is expressed as  $\varepsilon^{(l,q)} = \cup_{j \in \mathcal{J}^{(l,q)}} \varepsilon_j^{(l,q)}$ , where  $\mathcal{J}^{(l,q)}$  is an index set. The fault detection time  $T_{D_j}^{(l,q)}$  is denoted as the first time instant that the violation of (53) or (54) for at least one  $j \in \mathcal{J}^{(l,q)}$  occurs, by the local module  $\mathcal{M}^{(l,q)}$ , which can be expressed as  $T_{D_j}^{(l,q)} = \min\{t : \left| \varepsilon_{y_{x_j}}^{(l,q)}(t) \right| - \bar{\varepsilon}_{y_{x_j}}^{(l,q)}(t) > 0\}$  or  $T_{D_j}^{(l,q)} = \min\{\varepsilon_{y_{z_j}}^{(l,q)}(t) \notin [\underline{\varepsilon}_{y_{z_j}}^{(l,q)}(t), \bar{\varepsilon}_{y_{z_j}}^{(l,q)}(t)]\}$  respectively. Until this violation occurs, the local sensor set  $\mathcal{S}^{(l,q)}$  or system  $\Sigma^{(l)}$  is considered as non-faulty, meaning that no fault exists or that faults exist but remain undetected. A binary decision is considered as the output of  $\mathcal{M}^{(l,q)}$  expressed as  $D^{(l,q)}$  for the case of permanent fault occurrence as follows [16]:

$$D^{(l,q)}(t) = \begin{cases} 0, & t < T_{D_j}^{(l,q)} \\ 1, & t \geq T_{D_j}^{(l,q)} \end{cases} \quad (55)$$

Where  $T_D^{(l,q)} = \min\{T_{D_j}^{(l,q)} : j \in \mathcal{J}^{(l,q)}\}$ . As was shown in fig. 3 the decision of each module is fed to the aggregation module  $\mathcal{A}^{(l)}$  of the monitoring agent  $\mathcal{M}^{(l)}$  to acquire a binary decision vector  $D^{(l)} = [D^{(l,1)}, \dots, D^{(l,q)}]$  and compare it with a binary fault signature matrix  $F^{(l)}$ , consisting of  $n_l$  rows and  $N_{C_l} + 2$  columns where  $N_{C_l} = 2^{n_l} - 1 + n_p^{(l)}$  where  $n_p^{(l)}$  represents the columns associated with the considered process faults for each system  $\Sigma^{(l)}$ . The design of this matrix will be provided in the simulation results for the plenitude of the analysis. Keep in mind that when  $D^{(l)}(t) = 0_{n_l}$ , the diagnosis set  $\mathcal{D}_s^{(l)}$  is empty. Furthermore, if  $D^{(l,q)}(t) = f_q^{(l)} \forall q \in \{1, \dots, n_l\}$ , then the observed pattern  $\mathcal{D}^{(l)}(t)$  is characterised as consistent with the theoretical pattern  $F_i^{(l)}$  and the diagnosis set is expressed as  $\mathcal{D}_s^{(l)}(t) = \{F_{ci}^{(l)} : i \in \mathcal{D}_D^{(l)}(t)\}$  where  $\mathcal{D}_D^{(l)}(t)$  is the consistency index set expressed as  $\mathcal{D}_D^{(l)}(t) = \{i : F_{ci}^{(l)} = D^{(l)}(t), i \in \{1, \dots, N_{C_l}\}\}$ . Moreover, the agent  $\mathcal{M}^{(l)}$ ,  $I \in \{1, \dots, N\}$  also forms a decision on the propagation of sensor or process faults from the interconnected systems, aside from the local diagnosis set  $\mathcal{D}_s^{(l)}$ . This decision is made by the aggregation module  $\mathcal{A}^{(l)}$  of the agent, shown in Fig. 3, and is expressed as  $D_\chi^{(l)}(t)$  [16]:

$$D_\chi^{(l)}(t) = \begin{cases} 0, & \text{if } f_\chi^{(l)} \notin \mathcal{D}_s^{(l)}(t) \text{ and } f_p^{(l)} \notin \mathcal{D}_s^{(l)}(t) \\ 1, & \text{otherwise.} \end{cases} \quad (56)$$

Where  $f_p^{(l)} \in R^{n_p^*}$ ,  $n_p^* \leq n_l$  stands for the total number of faults that are propagated from the agent  $\mathcal{M}^{(l)}$ , to its neighboring agents, and  $f_\chi^{(l)}$  is associated with the faults propagated to the agent from the neighboring agents, due to the exchange of sensor information [16].

In the case a process fault occurs in a particular system  $\Sigma^{(l)}$ , it is again attributed to the local module  $\mathcal{M}^{(l,q)}$  of the particular system to detect it, and a task of the monitoring agent  $\mathcal{M}^{(l)}$  to distinguish it as a process fault instead of a sensor fault, according to its local decision logic. A prerequisite for this to happen is the use of redundant sensors. More specifically, if the outputs of

two or more sensors that are responsible to measure the same variable show a significant change that causes the residuals to violate the associated ARRs, then this fault is declared as a process fault. In this case that one process fault affects more than one variable in one system, the sensors responsible to measure these variables may detect the fault after the violation of (53) or (54) and if it is not associated with fault propagation, then it is assigned as a process fault. It is worth mentioning that the process faults that affect a system, cannot be detected by the monitoring agents of neighboring systems. This happens because the output measurements the agents use, that originate from the interconnection sensors, for the formation of residuals according to (42) and (43), cancel out with the interconnection variables  $\chi^{(l)}$  that are used in the formation of  $h^{(l)}(x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)})$  and  $\xi(x^{(l)}, z^{(l)}, \chi^{(l)}, u^{(l)})$ , of the equations that describe each system. Therefore, the process faults cannot propagate from one agent to another, unlike the sensor faults.

#### Global Decision Logic

The global decision logic establishes the isolation of the faults propagated via the interconnections between the monitoring agents. It is therefore the responsibility of a global agent  $\mathcal{G}$  to gather the decisions on the propagation of faults from the  $N$  local agents  $D_\chi(t) = [D_\chi^{(1)}(t), \dots, D_\chi^{(N)}(t)]$  and compare them with the columns of a global fault signature matrix  $F^\chi$  comprising of  $N$  rows and  $N_C = 2^p - 1 + n_p$  columns ( $p \leq \sum_{l=1}^N \{p_l\}$ ,  $p_l$  is the length of  $f_\chi^{(l)}$ ) [16].  $n_p$  represents the total number of the considered process faults that affect the propulsion process.

The star (\*) is placed in  $F^\chi$  instead of 1 in case the sensor fault is propagated to the agent  $\mathcal{M}^{(l)}$ , from the other agents  $\mathcal{M}^{(j)}$ ,  $J \in \{1, \dots, N\}$ ,  $J \neq I$  as it can not be assured it will be picked up by the agent  $\mathcal{M}^{(l)}$  [16]. If  $\mathcal{D}_\chi(t)$  is consistent with the  $k$ -th column of  $F^\chi(F_k^\chi)$ , meaning that  $\mathcal{D}_\chi(t) = F_k^\chi$ , the diagnosis set of propagated faults is denoted as  $\mathcal{D}_s^\chi(t) = \{F_{ck}^\chi : k \in \mathcal{S}_\chi(t)\}$ , where  $\mathcal{S}_\chi(t)$  is an index set expressed as  $\mathcal{S}_\chi(t) = \{k : F_{ck}^\chi = D_\chi^{(l)}(t), k \in \{1, \dots, N_C\} \forall I \in \{1, \dots, N\}\}$ . The non-empty local diagnosis set  $\mathcal{D}_s^\chi(t)$  of  $\mathcal{M}^{(l)}$ , is updated by the set  $\mathcal{D}_s^\chi(t)$  by excluding the occurrence of  $f_\chi^{(l)}$  and its combinations, if  $f_\chi^{(l)} \notin \mathcal{D}_s^\chi(t)$ . Thus, the global diagnosis set occurs as follows [16].

$$\mathcal{D}_s^{\mathcal{G}}(t) = \mathcal{D}_s^\chi(t) \cap \mathcal{D}_s(t) \quad (57)$$

Where

$$\mathcal{D}_s(t) = \bigcup_{I=1}^N \mathcal{D}_s^{(I)}(t) \quad (58)$$

## V SIMULATION RESULTS

In this section, the simulation results regarding the implementation of the aforementioned distributed sensor and process FDI scheme are presented. The simulation was run with the parameters provided in [22], and it was carried out for a duration of 300sec. It was assumed that the sensor measurements are affected by uniformly distributed noise that is 3% of the maximum amplitude of the noiseless measurement of each sensor. The faults are permanent and the sensor faults are abrupt and offset, while process faults can be either incipient or abrupt, offset or drift-like according to their fault equation. After the initiation of these faults, fault detection and isolation were achieved, and the discrimination between sensor and process faults was established. The 19 sensors of the system are distributed to the sensor sets that can be seen in Table 1, which were formed in such a way to enhance isolability of sensor and process faults, since almost all of the columns in the fault signature matrices are unique. This is achieved thanks to the use of different combinations of sensor outputs that are utilized in the monitoring modules. For instance, in table 1 one may notice that every first, third, and fifth monitoring module of an agent uses outputs from sensors with the index  $j$  being an odd number, while every second, fourth and sixth monitoring module uses outputs from sensors with the index  $j$  being an even number.

Therefore, the number of sensor combinations used is equal to the number of sensor sets. The design parameters for monitoring modules  $\mathcal{M}^{(1,1)}$ ,  $\mathcal{M}^{(1,2)}$ ,  $\mathcal{M}^{(3,1)}$ ,  $\mathcal{M}^{(3,2)}$ ,  $\mathcal{M}^{(4,1)}$  and  $\mathcal{M}^{(4,2)}$ , are presented in table 2. These parameters are used for the derivation of the adaptive thresholds as they were provided in (46) in the previous chapter.

**TABLE 1: SENSOR SETS AND INDEX SETS USED.**

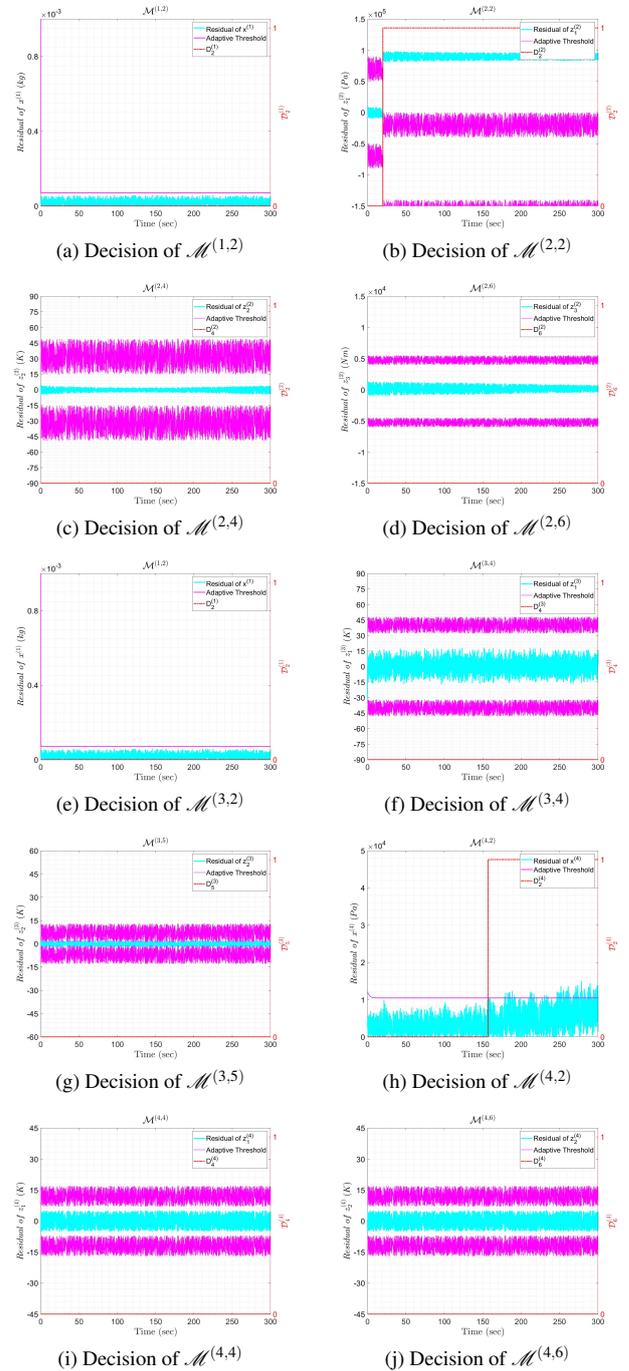
$\mathcal{M}^{(l)}$	$S^{(l,q)}$	Sensors $S^{(l)} \{j\} \forall j \in n_l$
$\mathcal{M}^{(1)}$	$s^{(1,1)}$	$s^{(1)} \{1\}$
	$s^{(1,2)}$	$s^{(1)} \{2\}$
$\mathcal{M}^{(2)}$	$s^{(2,1)}$	$s^{(2)} \{1\}$
	$s^{(2,2)}$	$s^{(2)} \{2\}$
	$s^{(2,3)}$	$s^{(2)} \{3\}$
	$s^{(2,4)}$	$s^{(2)} \{4\}$
	$s^{(2,5)}$	$s^{(2)} \{3\}, s^{(2)} \{5\}$
	$s^{(2,6)}$	$s^{(2)} \{4\}, s^{(2)} \{6\}$
$\mathcal{M}^{(3)}$	$s^{(3,1)}$	$s^{(3)} \{1\}, s^{(3)} \{3\}$
	$s^{(3,2)}$	$s^{(3)} \{2\}, s^{(3)} \{4\}$
	$s^{(3,3)}$	$s^{(3)} \{1\}, s^{(3)} \{3\}$
	$s^{(3,4)}$	$s^{(3)} \{2\}, s^{(3)} \{4\}$
	$s^{(3,5)}$	$s^{(3)} \{2\}, s^{(3)} \{4\}, s^{(3)} \{5\}$
$\mathcal{M}^{(4)}$	$s^{(4,1)}$	$s^{(4)} \{1\}, s^{(4)} \{3\}$
	$s^{(4,2)}$	$s^{(4)} \{2\}, s^{(4)} \{4\}$
	$s^{(4,3)}$	$s^{(4)} \{3\}$
	$s^{(4,4)}$	$s^{(4)} \{4\}$
	$s^{(4,5)}$	$s^{(4)} \{1\}, s^{(4)} \{3\}, s^{(4)} \{5\}$
	$s^{(4,6)}$	$s^{(4)} \{2\}, s^{(4)} \{4\}, s^{(4)} \{6\}$

Table 3 represents part of the fault signature matrix for monitoring agent  $\mathcal{M}^{(2)}$ . Table 3 shows that almost none of the ARRs in the table are affected by the propagation of faults from the other monitoring agents. Table 4 represents part of the fault signature matrix for monitoring agent  $\mathcal{M}^{(4)}$ . In table 4 no faults propagate within the monitoring modules of  $\mathcal{M}^{(4)}$ . Here only part of the tables was provided due to space limitations.

Table 5 is the partial global decision matrix. Specifically, one may notice that the faults from  $\Sigma^{(2)}$  may propagate to the monitoring agent  $\mathcal{M}^{(3)}$ , while the faults in the pressure sensors of  $\Sigma^{(4)}$  may propagate towards the monitoring agents of the second and third subsystems. A sensor fault was initiated at 20sec at one of the two pressure sensors responsible for the measurement of  $z_1^{(2)}$ , and the process fault concerning the seal leakage inside

**TABLE 2: DESIGN PARAMETERS IN MONITORING MODULES.**

Module $\mathcal{M}^{(l,q)}$	Gain $L^{(l,q)}$	Constant $\rho^{(l,q)}$	Constant $\xi^{(l,q)}$	Constant $\rho_d^{(l,q)}$	Constant $\xi_d^{(l,q)}$
$\mathcal{M}^{(1,1)}, \mathcal{M}^{(1,2)}$	33.3833	2	100	40	100
$\mathcal{M}^{(3,1)}, \mathcal{M}^{(3,2)}$	1.25	0.02	101.25	2.67	101.27
$\mathcal{M}^{(4,1)}, \mathcal{M}^{(4,2)}$	0.3764	$1.67 \times 10^{-7}$	0.2	0.02	0.4


**Fig. 4: Simulation results of FDI scheme for the given scenario.**



- [13] R. Izadi-Zamanabadi and M. Blanke. A ship propulsion system as a benchmark for fault-tolerant control. *Control Engineering Practice*, 7(2):227–239, 1999.
- [14] C. Bonivento, A. Paoli, and L. Marconi. Fault-tolerant control of the ship propulsion system benchmark. *Control engineering practice*, 11(5):483–492, 2003.
- [15] T. Falkenberg, R.T. Gregersen, and M. Blanke. Navigation system fault diagnosis for underwater vehicle. *IFAC Proceedings Volumes*, 47(3):9654–9660, 2014.
- [16] N. Kougiatsos, R.R. Negenborn, and V. Reppa. Distributed model-based sensor fault diagnosis of marine fuel engines. *IFAC-PapersOnLine*, 55(6):347–353, 2022.
- [17] A.T. Vemuri, M.M. Polycarpou, and A.R. Ciric. Fault diagnosis of differential-algebraic systems. *IEEE Transactions on Systems, man, and Cybernetics-Part A: Systems and Humans*, 31(2):143–152, 2001.
- [18] V. Reppa, M.M. Polycarpou, G.C. Panayiotou, et al. Sensor fault diagnosis. 2016.
- [19] V. Reppa, M.M. Polycarpou, and C.G. Panayiotou. Distributed sensor fault diagnosis for a network of interconnected cyberphysical systems. *IEEE Transactions on Control of Network Systems*, 2(1):11–23, 2014.
- [20] Y. Ding, D. Stapersma, and H. Grimmelius. Using parametrized finite combustion stage models to characterize combustion in diesel engines. *Energy & fuels*, 26(12):7099–7106, 2012.
- [21] S.A. Miedema and Z. Lu. The dynamic behavior of a diesel engine. In *Proc. WEDA XXII Technical Conference & 34th Texas A&M Dredging Seminar, June*, pages 12–15, 2002.
- [22] R.D. Geertsma, R.R. Negenborn, K. Visser, M.A. Loonstijn, and J.J. Hopman. Pitch control for ships with diesel mechanical and hybrid propulsion: Modelling, validation and performance quantification. *Applied energy*, 206:1609–1631, 2017.
- [23] M.M. Polycarpou and A.J. Helmicki. Automated fault detection and accommodation: a learning systems approach. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(11):1447–1458, 1995.
- [24] L. Jaulin and E. Walter. Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29(4):1053–1064, 1993.

# Bibliography

- [1] A. Ait Allal et al. "Ship Main Engine Lubricating Oil System's Reliability Analysis by Using Bayesian Network Approach". In: *International Journal of Engineering Research in Africa*. Vol. 48. Trans Tech Publ. 2020, pp. 108–125.
- [2] M.P. Asuquo et al. "Condition monitoring of marine and offshore machinery using evidential reasoning techniques". In: *Journal of Marine Engineering & Technology* 20.2 (2021), pp. 93–124.
- [3] J.F. Bash. "New Ship Technology and Design". In: *Marine Technology Society Journal* 42.1 (2008), pp. 21–25.
- [4] P. Bernard and V. Andrieu. "Luenberger observers for nonautonomous nonlinear systems". In: *IEEE Transactions on Automatic Control* 64.1 (2018), pp. 270–281.
- [5] BigInsight. *Sensor systems — BigInsight*. 2022. URL: <https://www.biginsight.no/personalised-marketing-2> (visited on 02/08/2022).
- [6] M. Blanke. "Fault-tolerant sensor fusion for marine navigation". In: *7th IFAC Conference on Manoeuvring and Control of Marine Craft*. 2006.
- [7] M. Blanke et al. *Diagnosis and fault-tolerant control*. Vol. 2. Springer, 2006.
- [8] M. Blanke et al. "Distributed fault diagnosis and fault-tolerant control". In: *Diagnosis and Fault-Tolerant Control*. Springer, 2016, pp. 467–518.
- [9] F. Boem, R.M.G. Ferrari, and T. Parisini. "Distributed fault detection and isolation of continuous-time non-linear systems". In: *European Journal of Control* 17.5-6 (2011), pp. 603–620.
- [10] C. Bonivento, A. Paoli, and L. Marconi. "Fault-tolerant control of the ship propulsion system benchmark". In: *Control engineering practice* 11.5 (2003), pp. 483–492.
- [11] K. Cicek et al. "Risk-based preventive maintenance planning using Failure Mode and Effect Analysis (FMEA) for marine engine systems". In: *2010 Second International Conference on Engineering System Management and Applications*. IEEE. 2010, pp. 1–6.
- [12] G. Cusano and S. La Marca. "Evaluation and Forecasting of Elapsed Fatigue Life of Ship Structures by Analyzing Data from Full Scale Ship Structural Monitoring". In: *Journal of Shipping and Ocean Engineering* 5 (2015), pp. 59–74.
- [13] Y. Ding, D. Stapersma, and H. Grimmelius. "Using parametrized finite combustion stage models to characterize combustion in diesel engines". In: *Energy & fuels* 26.12 (2012), pp. 7099–7106.
- [14] A.L. Ellefsen et al. "A comprehensive survey of prognostics and health management based on deep learning for autonomous ships". In: *IEEE Transactions on Reliability* 68.2 (2019), pp. 720–740.
- [15] T. Falkenberg, R.T. Gregersen, and M. Blanke. "Navigation system fault diagnosis for underwater vehicle". In: *IFAC Proceedings Volumes* 47.3 (2014), pp. 9654–9660.
- [16] R.M.G. Ferrari, T. Parisini, and M.M. Polycarpou. "Distributed fault detection and isolation of large-scale discrete-time nonlinear systems: An adaptive approximation approach". In: *IEEE Transactions on Automatic Control* 57.2 (2011), pp. 275–290.
- [17] R.S. Figliola and D.E. Beasley. *Theory and design for mechanical measurements*. 2015.
- [18] J. Filipczyk. "Causes of automotive turbocharger faults". In: *Transport Problems* 8 (2013).
- [19] M. Fonte and M. De Freitas. "Marine main engine crankshaft failure analysis: a case study". In: *Engineering Failure Analysis* 16.6 (2009), pp. 1940–1947.
- [20] M. Fonte, L. Reis, and M. Freitas. "Failure analysis of a gear wheel of a marine azimuth thruster". In: *Engineering Failure Analysis* 18.7 (2011), pp. 1884–1888.

- [21] T. Fossen. *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.
- [22] T. Fossen, K.Y. Pettersen, and H. Nijmeijer. *Sensing and control for autonomous vehicles*. Springer, 2017.
- [23] R.D. Geertsma et al. "Design and control of hybrid power and propulsion systems for smart ships: A review of developments". In: *Applied Energy* 194 (2017), pp. 30–54.
- [24] R.D. Geertsma et al. "Pitch control for ships with diesel mechanical and hybrid propulsion: Modelling, validation and performance quantification". In: *Applied energy* 206 (2017), pp. 1609–1631.
- [25] DNV GL. *Condition Based Maintenance (CBM)*. 2022. URL: <https://www.dnv.com/services/condition-based-maintenance-cbm--12179> (visited on 02/08/2022).
- [26] DNV GL. *Impact on maritime - DNV*. 2022. URL: <https://www.dnv.com/to2030/impact/impact-on-maritime.html> (visited on 02/08/2022).
- [27] DNV GL. *Mapping our oceans with autonomous vehicles - DNV*. 2022. URL: <https://www.dnv.com/to2030/technology/mapping-our-oceans-with-autonomous-vehicles.html> (visited on 02/08/2022).
- [28] DNV GL. *Prognostic maintenance: seeing into the future - DNV*. 2022. URL: <https://www.dnv.com/to2030/technology/prognostic-maintenance-seeing-into-the-future.html> (visited on 02/08/2022).
- [29] Q. Gong et al. "Research on the Maintenance and Common Failures of the Marine Machinery and Equipment of the Scientific Investigation Ship". In: *Journal of Physics: Conference Series*. Vol. 1802. 2. IOP Publishing. 2021, p. 022071.
- [30] C. Guan et al. "Computational investigation of a large containership propulsion engine operation at slow steaming conditions". In: *Applied Energy* 130 (2014), pp. 370–383.
- [31] X. Han et al. "Analysis and Countermeasures of Stuck Failure of SR Steering Gear of a Ship". In: *E3S Web of Conferences*. Vol. 165. EDP Sciences. 2020, p. 06001.
- [32] Y. Han and Y.H. Song. "Condition monitoring techniques for electrical equipment-a literature survey". In: *IEEE Transactions on Power delivery* 18.1 (2003), pp. 4–13.
- [33] L. Huang et al. "Development of a bond graph based model library for turbocharged diesel engines". In: *Energy* 148 (2018), pp. 728–743.
- [34] R. Isermann. *Fault-diagnosis applications: model-based condition monitoring: actuators, drives, machinery, plants, sensors, and fault-tolerant systems*. Springer Science & Business Media, 2011.
- [35] R. Isermann. *Fault-diagnosis systems: an introduction from fault detection to fault tolerance*. Springer Science & Business Media, 2006.
- [36] R. Islam et al. "Reliability assessment of a main propulsion engine fuel oil system-what are the failure-prone components?" In: *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation* 13.2 (2019).
- [37] R. Izadi-Zamanabadi and M. Blanke. "A ship propulsion system as a benchmark for fault-tolerant control". In: *Control Engineering Practice* 7.2 (1999), pp. 227–239.
- [38] L. Jaulin and E. Walter. "Set inversion via interval analysis for nonlinear bounded-error estimation". In: *Automatica* 29.4 (1993), pp. 1053–1064.
- [39] D. Jung. "Structural Methods for Distributed Fault Diagnosis of Large-Scale Systems". In: *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE. 2020, pp. 2690–2695.
- [40] H. Kazemi and A. Yazdizadeh. "Fault detection and isolation of gas turbine engine using inversion-based and optimal state observers". In: *European Journal of Control* 56 (2020), pp. 206–217.
- [41] A.S. Khan et al. "Distributed fault detection and isolation in second order networked systems in a cyber-physical environment". In: *ISA transactions* 103 (2020), pp. 131–142.
- [42] V. Knežević et al. "Fault tree analysis and failure diagnosis of marine diesel engine turbocharger system". In: *Journal of Marine Science and Engineering* 8.12 (2020), p. 1004.

- [43] Ž. Koboević, D. Bebić, and Ž. Kurtela. "New approach to monitoring hull condition of ships as objective for selecting optimal docking period". In: *Ships and Offshore structures* 14.1 (2019), pp. 95–103.
- [44] N. Kougiatsos, R.R. Negenborn, and V. Reppa. "Distributed model-based sensor fault diagnosis of marine fuel engines". In: *IFAC-PapersOnLine* 55.6 (2022), pp. 347–353.
- [45] R. Laskowski. "Fault Tree Analysis as a tool for modelling the marine main engine reliability structure". In: *Zeszyty Naukowe Akademii Morskiej w Szczecinie* 41 (113) (2015), pp. 71–77.
- [46] I. Lazakis, Y. Raptodimos, and T. Varelas. "Predicting ship machinery system condition through analytical reliability tools and artificial neural networks". In: *Ocean Engineering* 152 (2018), pp. 404–415.
- [47] Z. Li et al. "Condition monitoring and fault diagnosis for marine diesel engines using information fusion techniques". In: *Elektronika ir Elektrotechnika* 123.7 (2012), pp. 109–112.
- [48] A.L. Michala et al. "Wireless condition monitoring for ship applications". In: *Smart Ship Technology* (2016), pp. 59–66.
- [49] S.A. Miedema and Z. Lu. "The dynamic behavior of a diesel engine". In: *Proc. WEDA XXII Technical Conference & 34th Texas A&M Dredging Seminar, June. 2002*, pp. 12–15.
- [50] K.A. Morgansen and N. Brace. "Observability-Based Sensor Sampling". In: *Sensing and Control for Autonomous Vehicles*. Springer, 2017, pp. 229–248.
- [51] R.R. Negenborn, H. Hellendoorn, and Z. Lukszo. *Intelligent infrastructures*. Vol. 2. Springer, 2010.
- [52] D.E. Onwuegbuchunam et al. "Propulsion Shaft and Gearbox Failure in Marine Vessels: A Duration Model Analysis". In: *Journal of Transportation Technologies* 10.04 (2020), p. 291.
- [53] P.Pantela. "Fault Diagnosis for Marine Vehicles". In: *TU Delft repository* (2022), pp. 1–42.
- [54] M.M. Polycarpou and A.J. Helmicki. "Automated fault detection and accommodation: a learning systems approach". In: *IEEE Transactions on Systems, Man, and Cybernetics* 25.11 (1995), pp. 1447–1458.
- [55] Y. Raptodimos et al. "Ship sensors data collection and analysis for condition monitoring of ship structures and machinery systems". In: *Smart Ship Technology* (2016).
- [56] V. Reppa, M.M. Polycarpou, and C.G. Panayiotou. "Distributed sensor fault diagnosis for a network of interconnected cyberphysical systems". In: *IEEE Transactions on Control of Network Systems* 2.1 (2014), pp. 11–23.
- [57] V. Reppa, M.M. Polycarpou, G.C. Panayiotou, et al. "Sensor fault diagnosis". In: (2016).
- [58] V. Reppa et al. "Distributed detection and isolation of sensor faults in HVAC systems". In: *21st Mediterranean Conference on Control and Automation*. IEEE. 2013, pp. 401–406.
- [59] H. Ripley. "Remote Sensing—State of the Art". In: *Marine Technology Society Journal* 42.1 (2008), pp. 15–20.
- [60] Y. Rkhisssi-Kammoun et al. "Two current sensor fault detection and isolation schemes for induction motor drives using algebraic estimation approach". In: *Mathematics and Computers in Simulation* 157 (2019), pp. 39–62.
- [61] S. Schirmacher, L. Overmeyer, and S. Lorsch. "Wireless condition monitoring of a marine gearbox". In: *Ship Technology Research* 63.1 (2016), pp. 38–49.
- [62] D. Stapersma. "Diesel engines—a fundamental approach to performance analysis, turbocharging, combustion, emissions and heat transfer". In: *Emissions and Heat Transfer, Part II: Diesel Engines B—Combustion, Emissions and Heat Transfer* (2010).
- [63] O.F.A. van Straaten and M.J. de Boer. "Optimum propulsion engine configuration from fuel economic point of view". In: *Proc 11th INEC, Edinburgh, UK* (2012).
- [64] Y. Sun et al. "Thruster fault diagnosis method based on Gaussian particle filter for autonomous underwater vehicles". In: *International Journal of Naval Architecture and Ocean Engineering* 8.3 (2016), pp. 243–251.

- [65] C. Svärd. "Methods for automated design of fault detection and isolation systems". PhD thesis. PhD thesis dissertations, Linköping Studies in Science and Technology ..., 2012.
- [66] A.T. Vemuri, M.M. Polycarpou, and A.R. Ciric. "Fault diagnosis of differential-algebraic systems". In: *IEEE Transactions on Systems, man, and Cybernetics-Part A: Systems and Humans* 31.2 (2001), pp. 143–152.
- [67] G. Vizentin, G. Vukelić, and M. Srok. "Common failures of ship propulsion shafts". In: *Pomorstvo* 31.2 (2017), pp. 85–90.
- [68] D. Watzenig, M.S. Sommer, and G. Steiner. "Engine state monitoring and fault diagnosis of large marine diesel engines". In: *e & i Elektrotechnik und Informationstechnik* 126.5 (2009), pp. 173–179.
- [69] X. Wen et al. "Fault detection and diagnosis in the INS/GPS navigation system". In: *2014 World Automation Congress (WAC)*. IEEE. 2014, pp. 27–32.
- [70] M. Wohlthan et al. "A multi-stage geometric approach for sensor fault isolation on engine test beds". In: *Measurement* 168 (2021), p. 108313.
- [71] M. Wohlthan et al. *Methodology for Automated Fault Diagnosis at Engine Test Beds*. Tech. rep. SAE Technical Paper, 2017.
- [72] T. Xie et al. "A review of current issues of marine current turbine blade fault detection". In: *Ocean Engineering* 218 (2020), p. 108194.
- [73] X. Xu and Z. Yu. "Failure analysis of a marine intermediate shaft". In: *Engineering Failure Analysis* 126 (2021), p. 105464.
- [74] P. Zhang et al. "Marine Systems and Equipment Prognostics and Health Management: A Systematic Review from Health Condition Monitoring to Maintenance Strategy". In: *Machines* 10.2 (2022), p. 72.
- [75] Q. Zhang, M. Basseville, and A. Benveniste. "Fault detection and isolation in nonlinear dynamic systems: A combined input–output and local approach". In: *Automatica* 34.11 (1998), pp. 1359–1373.