

Exploring the role of system operation modes in failure analysis in the context of first generation cyber-physical systems

Ruiz Arenas, Santiago

DOI

[10.4233/uuid:c6fa5814-99c7-4492-95cc-beed45286c71](https://doi.org/10.4233/uuid:c6fa5814-99c7-4492-95cc-beed45286c71)

Publication date

2018

Document Version

Final published version

Citation (APA)

Ruiz Arenas, S. (2018). *Exploring the role of system operation modes in failure analysis in the context of first generation cyber-physical systems*. [Dissertation (TU Delft), Delft University of Technology]. <https://doi.org/10.4233/uuid:c6fa5814-99c7-4492-95cc-beed45286c71>

Important note


To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

The background features a complex, abstract pattern of light blue lines resembling a circuit board or data paths. These lines are set against a dark blue gradient. Several nodes are highlighted with glowing white circles, and other nodes are represented by smaller, solid light blue circles. The overall aesthetic is clean, modern, and technological.

Exploring the role of
system operation modes in
failure analysis in the context of
first generation cyber-physical systems

Santiago Ruiz-Arenas

Exploring the role of system operation modes in failure analysis in the context of first generation cyber-physical systems

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. T.H.J.J. van der Hagen;
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op
Woensdag 16 mei 2018 om 10:00 uur

door

Santiago RUIZ-ARENAS
Master of Science in Engineering
Universidad EAFIT, Medellín, Colombia
geboren te Medellín, Colombia

Dit proefschrift is goedgekeurd door de promotor:
Prof. dr. I. Horváth

Composition of the doctoral committee:

Rector Magnificus	chairman
Prof. dr. I. Horváth	promotor, Technische Universiteit Delft
Prof. dr. R. Mejia-Gutierrez	promotor, Universidad EAFIT, Colombia
Dr. Z. Rusák	copromotor, Technische Universiteit Delft

Independent members:

Prof. dr. K. Zeman	Johannes Kepler Universität, Austria
Prof. dr.ir. B. Tekinerdogan	Wageningen University
Prof. dr. F.M. Brazier	Technische Universiteit Delft
Prof. dr. C.C.L. Wang	Technische Universiteit Delft
Prof. ir. Jos Oberdorf	Technische Universiteit Delft, reserve member



Exploring the role of system operation modes in failure analysis in the context of first generation cyber-physical systems

Keywords: Cyber-physical systems, failure analysis, system reliability, failure diagnosis, failure forecasting, preventive maintenance, complex systems, system dependability

Printed by: Ipskamp Printing

This research was funded by Colciencias. Universidad EAFIT (Colombia) made important contributions to the work described in this dissertation

ISBN 978-94-6186-916-6

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

Copyright © 2018 by S. Ruiz-Arenas

sruizare@gmail.com

Acknowledgements

I would like to express my sincere gratitude to my supervisors Prof. Dr. Imre Horváth, Prof. Dr. Ricardo Mejía-Gutiérrez and Dr. Zoltan Rusák for their support, patience and valuable guidance. Their feedback, encouragement, trust and inspirational ideas were key for the development of this project.

Special thanks to my fellow researchers at the Cyber-Physical Systems Research Group of Industrial Design Engineering at TU Delft who contributed to the development of this thesis. The long discussions we had on the coffee corner, as well as their advice in the daily issues made it easier this difficult path of doing a PhD in a foreign country.

I would also like to thank to my Colombian fellows and friends at the Grupo de Investigación en Ingeniería de Diseño (GRID) at Universidad EAFIT with whom I shared the funniest lunch sessions ever. I am particularly grateful with Lis for her support in the technical and methodological issues, as well as for introducing me to Delft's culture and offering me her home until I found a place on my own.

Finally, I am writing the most inner words to express my finest love to my family, specially to my mother, sister, aunt Margoth and her husband Oscar who supported me every single day of this extensive journey. Thanks for believing in me and never letting my down. Mami, gracias por estar ahí siempre y por soportar conmigo los momentos más difíciles de este doctorado.

Table of contents

Chapter 1	Introduction	1
1.1	Background of the research	1
1.2	Current trends in system engineering	3
1.3	Introducing the paradigm of cyber-physical systems	4
1.4	The landscape of cyber-physical systems	6
1.4.1	Evolution of the paradigm of CPSs	6
1.4.2	Effects of the progression on the theory and practice of failure diagnosis and avoidance	8
1.5	Description of the concept of system operation modes	9
1.6	Brief overview of the current failure analysis approaches	10
1.6.1	Fundamentals of failure analytics and maintenance	10
1.6.2	Information engineering for failure analysis	11
1.6.3	Supporting decision making in failure analysis	13
1.6.4	Overview of indicators used as decision enablers	14
1.7	The phenomenon of changing SOMs as compensatory action	16
1.8	Description of the research problem	17
1.9	Research methodology	18
1.10	Structure of the thesis	20
1.11	Forerunning publications	21
1.12	References	22
Chapter 2	State of the art review	27
2.1	Aggregation of knowledge concerning the state of the art	27
2.1.1	General objective of this study	27
2.1.2	The research approach	28
2.1.3	The reasoning model	29
2.1.4	Overview of the challenges of aggregating knowledge and investigating cyber-physical systems	30
2.2	Fundamentals of failure analytics	33
2.2.1	Consideration of the specificities of the CPS hardware in failure analytics	33
2.2.2	Consideration of the specificities of the CPS software in failure analytics	34
2.2.3	Consideration of the specificities of the CPS cyberware in failure analytics	35
2.3	Fundamentals of signal-based failure analytics	36
2.3.1	A concise overview of the types of signals	36
2.3.2	Supporting decision making based on signal analysis	37

2.4	Model-based failure analytics	39
2.4.1	Fundamentals of model-based failure analytics	39
2.4.2	Quantitative model-based failure analytics	40
2.4.3	Qualitative model-based failure analytics	43
2.5	Data-driven failure analytics	48
2.5.1	Fundamentals of data-driven failure analytics	48
2.5.2	Artificial neural network-based failure analytics	49
2.5.3	Statistical pattern based failure analytics	50
2.5	Implications of the findings	53
2.6	Conclusions	57
2.7	References	58

Chapter 3	A testbed system for empirical study of the influences of changing SOMs	71
3.1	Introduction	71
3.1.1	General objective	71
3.1.2	On the phenomenon of self-tuning	72
3.1.3	Implications of self-tuning for failure analysis	73
3.2	Implications of self-tuning induced change of SOMs	75
3.2.1	Implications of SOMs on failure analytics	75
3.2.2	Investigation of system operation modes of cyber-physical systems	77
3.3	Requirements for the implementation of a physical testbed system	79
3.3.1	Functional Requirements for Traditional systems: The greenhouse testbed perspective	79
3.3.2	Functional Requirements for Cyber-Physical Systems	82
3.4	Architecture of the testbed system	86
3.4.1	Initial considerations	86
3.4.2	Details on the testbed system architecture	87
3.4.3	Description of the placement of the components	92
3.5	Greenhouse information analysis	97
3.5.1	Operation flow of the testbed system	97
3.5.2	Signal and information processing	101
3.6	Discussion	101
3.7	Conclusions	105
3.8	References	106

Chapter 4	Investigation of the role of SOM in a signal based failure analysis	109
4.1	Introduction to research cycle 3	109
4.1.1	Objectives	109
4.1.2	Research approach	109
4.2	Failure diagnosis with unsegmented signals	110
4.2.1	Description of the experiment	110
4.2.2	Method for analyzing output data	113
4.2.3	Failure analysis based on unsegmented signals	116

4.3	Method for SOM based signal segmentation for failure analysis	117
4.3.1	Elements of the theory underpinning the conducted exploration	117
4.3.2	Description of the proposed Failure Indicator (FI)	118
4.3.3	Proposed procedure for signal segmentation	119
4.4 A	pilot study of the SOMs-oriented failure analysis methodology	123
4.4.1	Derivation of failure indicator matrix	124
4.4.2	Analysis of the discriminant power of the failure indicators	130
4.4.3	Sensitivity of SOM based failure diagnosis to variations of operating conditions	131
4.4.4	Findings of the Pilot Study	136
4.5 A	real-life study of the SOMs-oriented failure analysis methodology	138
4.5.1	Description of the experiment	138
4.5.2	Deriving failure indicator matrix	139
4.5.3	Analysis of the discriminant power of the obtained failure indicators	149
4.6	Discussion of the results and their implications	150
4.1	Conclusions	151
4.1	References	152
Chapter 5	The role of SOM in a behavior-based analysis of failures	153
5.1	Introduction	153
5.1.1	Objectives of the research cycle 4	153
5.1.2	Research hypothesis and assumptions	154
5.2	Approach for investigating trends of the SOMs behavior	155
5.2.1	Investigation of the consistency and discriminant power of the trends	156
5.2.2	Evaluation of the changes in SOM frequency and duration as indicators for failure analysis	157
5.2.3	Exploring potential use of the changing SOM frequency and SOM duration for failure forecasting	157
5.3	A pilot study for failure forming analysis considering SOMs behavior	159
5.3.1	Introduction	159
5.3.2	Analysis of the uniqueness and discriminant power of the observed trends	160
5.3.3	Evaluation of the SOMs' frequency and duration as indicators of the failure forming process	169
5.4	A pilot study for failure forecasting considering SOMs behavior	172
5.4.1	Failure forecasting in the pilot-study of a simulated kettle model	172
5.4.2	Conclusions about the pilot implementation	175
5.5	A real-life case study for failure forecasting based on SOMs frequency and duration	176
5.5.1	Evolution of frequency and duration of SOMs	176
5.5.2	Combined analysis of SOMs' frequency and duration	182
5.6	Discussion	187
5.7	Conclusions	188

5.8	References	189
Chapter 6	Investigation of the role of SOMs in broader context of maintenance	191
6.1	Introduction	191
6.1.1	Research objectives	191
6.1.2	Introduction of the relevant terms	192
6.2	Analysis of existing maintenance principles	193
6.2.1	Overview of the maintenance principles used in the context of OG-CPS	193
6.2.2	Projecting the maintenance principles of OG-CPS to 1G-CPSs	197
6.2.3	Operationalization of relevant maintenance principles for CPSs	201
6.3	About maintenance principles for CPSs	201
6.3.1	Opportunities emerging from the SOMs concept	201
6.3.2	Some hints on specific maintenance principles for CPSs	206
6.4	Conclusions	207
6.5	References	208
Chapter 7	Conclusions, propositions, reflections and further research	211
7.1	Main results of the research	211
7.1.1	Moving towards failure management dedicated to CPSs	211
7.1.2	Major findings concerning the state of the art of computational failure analysis	212
7.1.3	Major findings related to the influential factors of the investigated phenomenon	214
7.1.4	Major findings related to the analysis of SOM effect over failure manifestations	216
7.1.5	Major findings concerning the analysis of SOMs in the failure forming process	217
7.1.6	Major findings concerning the role of SOMs in possible maintenance principles	219
7.2	Propositions	221
7.3	Reflections on the completed research	223
7.3.1	Reflections about the process	223
7.3.2	Reflections about the experiments and the results	224
7.4	Further research	225
7.4.1	Short term challenges	226
7.4.2	Long term challenges	227
Summary	229
Sammenvatting	233

List of figures

Figure 1.1.	The overall organization and the methodological framing of the research.....	19
Figure 2.1.	Reasoning model for knowledge aggregation.....	30
Figure 3.1.	Requirements for traditional systems and augmented requirements for cyber physical systems in the case of 1G-CPS	79
Figure 3.2.	Architecture of the proposed testbed.....	88
Figure 3.3.	User interface	92
Figure 3.4.	Introducing the testbed	93
Figure 3.5.	Placement of the plant bed units	93
Figure 3.6.	Placement of the top of the household	94
Figure 3.7.	Placement of the low part of the household	95
Figure 3.8.	Placement of the tank	96
Figure 3.9.	Flowchart of the plant-bed units	98
Figure 3.10.	Flowchart of the greenhouse unit.....	99
Figure 3.11.	Flowchart of the reasoning unit.....	100
Figure 4.1.	Diagram of the kettle model.....	111
Figure 4.2.	Use conditions for the Outflow valve	112
Figure 4.3.	Example of a colored R matrix, where rows correspond to the sensor signals and columns to system Operation modes.....	123
Figure 4.4.	Failure indicators corresponding to the kettle model, evaluated with the same scenarios for the reference case and the failed ones.....	124
Figure 4.5.	Explanation of the composition of the kettle's model failure indicator.....	125
Figure 4.6.	Failure indicators corresponding to the kettle model.....	131
Figure 4.7.	Similarity level between the failure indicators derived with the scenarios, and the ones derived with different scenarios	132
Figure 4.8.	Failure indicators for the greenhouse testbed.....	140
Figure 4.9.	Comparison of soil humidity for the irrigation valve obstruction failure mode	145
Figure 4.10.	Fragment of soil moisture that presents signal segments corresponding to ζ_{41} for both, failure-free and failed operation.....	146
Figure 4.11.	Comparison of the water temperature signal between the failure-free operation and F_2	148
Figure 5.1.	Approach for exploring the role of SOMs' frequency and duration in failure diagnosis and forecasting.....	155
Figure 5.2.	Extrapolated trend	158

Figure 5.3.	Illustrative example of the time-series based forecasting application	159
Figure 5.4.	Failure progress	161
Figure 5.5.	Filtered trends corresponding to F_3	162
Figure 5.6.	Comparison of Fq distribution based on failure modes	164
Figure 5.7.	Comparison of Fq distribution based on failure modes	165
Figure 5.8.	Comparison of D distribution based on failure modes.....	166
Figure 5.9.	Comparison of D distribution based on failure modes.....	167
Figure 5.10.	Obtained FI matrices for Fq and D indicators.....	168
Figure 5.11.	Evolution of failure prediction for the analyzed failure modes	171
Figure 5.12.	Failure forecasting for the analyzed failure modes	174
Figure 5.13.	Evolution of the greenhouse ‘Tank leak’ failure mode.....	177
Figure 5.14.	Comparison of the variation presented by SOM frequency for the tank leak and the failure-free case	179
Figure 5.15.	Comparison of the variation presented by SOM duration for the tank leak and the failure-free case	180
Figure 5.16.	Filtered trends corresponding to tank leak	181
Figure 5.17.	Failure progression processes	183
Figure 5.18.	Evolution of failure prediction for failure-leak	185
Figure 5.19.	Failure forecasting for tank leak.....	186
Figure 6.1.	Clarification of the main terms.....	192
Figure 6.2.	Separation of concerns with regards to maintenance strategies.....	194
Figure 6.3.	Taxonomy of maintenance principles	196
Figure 6.4.	Doctrine of integral maintenance for CPSs.....	198
Figure 6.5.	Roadmap towards maintenance principles for CPSs.....	200

List of tables

Table 2.1.	Summary of the key elements considered in failure analysis for all the analyzed techniques	55
Table 3.1.	Description of the components of the greenhouse unit	89
Table 3.2.	Description of the components of the Plant bed unit	91
Table 3.3.	Description of the actuator signals of the testbed	101
Table 3.4.	Description of the sensor signals of the testbed	102
Table 3.5.	Implementation of the functional requirement.....	104
Table 4.1.	Control settings of the kettle model	111
Table.4.2.	Results from significance test corresponding to the whole length signal segment analysis	115
Table 4.3.	Effect size of the statistical test conducted to the whole length signal	116
Table 4.4.	Occurring SOM in the kettle model	125
Table 4.5.	Effect size for F_1	126
Table 4.6.	Effect size for F_2	127
Table 4.7.	Effect size for F_3	128
Table 4.8.	Effect size for F_4	129
Table 4.9.	Similarity level between failure indicators.....	130
Table 4.10.	Average effect size for F_1 with randomly selected scenarios	133
Table 4.11.	Average effect size for F_2 with randomly selected scenarios	134
Table 4.12.	Average effect size for F_3 with randomly selected scenarios	135
Table 4.13.	Average effect size for F_4 with randomly selected scenarios	136
Table 4.14.	Similarity level between failure indicators for randomly selected scenarios	137
Table 4.15.	Occurring system operation modes	139
Table 4.16.	Average effect size of F_1 for the testbed case	142
Table 4.17.	Average effect size of F_2 for the testbed case	143
Table 4.18.	Average effect size of F_3 for the testbed case	144
Table 4.19.	Similarity level of the failure indicators derived for the testbed.....	149
Table 5.1.	Results of the statistical test for SOM duration	168
Table 5.2.	Results of the statistical test for SOM frequency.....	168
Table 5.3.	Confusion matrix of classification obtained by analyzing the measured Fq and D parameters	169
Table 5.4.	Comparison of the predicted and forecasted failure diagnosis	175
Table 5.5.	Results of the statistical test for SOM frequency and SOM duration	177
Table 5.6.	Variance per SOM in the greenhouse's case.....	178
Table 5.7.	Variance per SOM in the kettle's case	182
Table 5.8.	Confusion matrix of the greenhouse's classification model	183

Nomenclature

AFD	Active Fault Detection
ANN	Artificial Neural Networks
CBM	Condition-Based Maintenance
CCA	Controlled Core Area
CDN	Cross Domain Networking
CM	Corrective Maintenance
COM	Component Operation Mode
CPA	Cyber-Physical Augmentation
CPC	Cyber-Physical Computing
CPGH	Cyber Physical Green-House
CPSs	Cyber-Physical Systems
D	Duration of SOM
DIK	Data Information and Knowledge
DIR	Design Inclusive Research
DOM	Design Out Maintenance
EFA	Extended Field of Application
EKF	Extended Kalman Filter
FI	Failure Indicator
FIOM	Failure Induced Operation Mode
F _q	Frequency of occurrence of SOM
FTA	Fault Tree Analysis
GUI	User Interface
IAE	Integral of Absolute Error
LDA	Linear Discriminant Analysis
OBM	Opportunistic-Based Maintenance
PAR	Photosynthetic Active Radiation
PCA	Principal Component Analysis
PDR	Practice-Driven Research
PM	Preventive Maintenance
Qstate	Qualitative States
QTA	Qualitative Trend Analysis
RCPS	Requirements for Cyber-Physical Systems
RDC	Research in Design Context
RMS	Root Mean Square
RTS	Requirements for Traditional Systems
SDG	Signed Disgraphs
SOM	System Operation Mode
SoS	System of Systems
SPC	Statistical Pattern Classifier
ST	Statistical Test
STF	Short Time Fourier

SVM	Support Vector Machine
TBM	Time-Based Maintenance
TFDA	Time Frequency Domain Analysis
TTF	Time To Failure
WVD	Wigner-Ville Distribution
0G-CPS	Zero Generation Cyber-Physical System
1G-CPS	First Generation Cyber-Physical System
2G-CPS	Second Generation Cyber-Physical System
3G-CPS	Third Generation Cyber-Physical System
4G-CPS	Fourth Generation Cyber-Physical System

List of symbols

Symbols used in the description of SOM based segmentation process:

S_S	Sensor signals
S_A	Actuator signals
a_m	Signal features
F_r	Failure modes
$\zeta_{S_{A_j}}$	Component operation mode
ζ_d	System operation mode
\emptyset	Symptoms
Sg	‘Signal segment’
FI	‘Failure indicator’
\mathcal{O}	System regular operation
∂	Reference behavior
A^∂	Dataset from ‘reference (failure-free) operation’
A^O	Dataset from ‘observed (failed) operation’
$A_{\zeta_d}^\partial$	‘System’s reference behavior’
$A_{\zeta_d}^O$	‘System’s observed behavior’
$OM_{v \times n}$	‘Operation Mode Matrix’
FI^∂	‘Reference indicator’
FI^O	Observed indicator
r	Pearson’s correlation coefficient
Fq	Frequency of occurrence of SOMs
D	Duration of the occurrence of SOMs
ΔFq	Variation of the frequency of SOMs
ΔD	Variation of the SOM’s duration
w	Failure progress step
P	Predictor vector
F	Class vector
P_w	Predictor vector for each step w
\hat{P}^{w+h}	Predictor vector to arrange forecasted data
c	Time to failure
Fq_{ζ_d}	Frequency of a particular SOM
D_{ζ_d}	Duration of a particular SOM
F_f	Failure-free
$\Delta Fq_{\zeta_d}^{F_f}$	Frequency variation for SOM in failure free operation
$\Delta Fq_{\zeta_d}^{F_r}$	Frequency variation for SOM in a particular failure mode operation
$\Delta D_{\zeta_d}^{F_f}$	Duration variation for SOM in failure free operation
$\Delta D_{\zeta_d}^{F_r}$	Duration variation for SOM in a particular failure mode operation

Sensor signals from pilot study:

- S_{S_1} 'Water temperature' signal
- S_{S_2} 'Water tank level' signal
- S_{S_3} 'Heating power' signal

Actuator signals from pilot study:

- S_{A_1} 'Inflow valve'
- S_{A_2} 'Outflow valve'
- S_{A_3} 'Heater'
- S_{A_4} 'Additive injection valve'

Signal Features from pilot study:

- a_1 'Derivative' feature
- a_2 'Standard deviation' feature
- a_3 'Mean' feature
- a_4 'Area' feature
- a_5 'Median' feature

Failure Modes from pilot study:

- F_1 'Tank leak'
- F_2 'Inflow valve obstruction'
- F_3 'Loss of heating power'
- F_4 'Outflow valve obstruction'

Sensor signals from real-life study:

- S_{S_1} 'White light level plant bed 1'
- S_{S_2} 'Lighting power consumption plant bed 1'
- S_{S_3} 'Soil moisture plant bed 1'
- S_{S_4} 'Soil temperature plant bed 1'
- S_{S_5} 'PAR lighting plant bed 1'
- S_{S_6} 'UV light level plant bed 1'
- S_{S_7} 'PH level plant bed 1'
- S_{S_8} 'Water level'
- S_{S_9} 'Water temperature'
- $S_{S_{10}}$ 'Greenhouse temperature'
- $S_{S_{11}}$ 'Relative humidity'
- $S_{S_{12}}$ 'CO₂ level'
- $S_{S_{13}}$ 'RPM fan-in'
- $S_{S_{14}}$ 'RPM fan-out'

$S_{S_{15}}$	‘White light level plant bed 2’
$S_{S_{16}}$	‘Lighting power consumption plant bed 2’
$S_{S_{17}}$	‘Soil moisture plant bed 2’
$S_{S_{18}}$	‘Soil temperature plant bed 2’
$S_{S_{19}}$	‘PAR lighting plant bed 2’
$S_{S_{20}}$	‘UV light level plant bed 2’
$S_{S_{21}}$	‘PH level plant bed 2’

Actuator signals from real-life study:

S_{A_1}	‘Electro valve plant bed 1’, corresponds to irrigation valve in ‘plant-bed 1’
S_{A_2}	‘Electro valve water reservoir’, corresponds to the inlet tank valve.
S_{A_3}	‘Heater’, corresponding to a water resistance for the heater
S_{A_4}	‘Fan-in’, corresponds the inlet fan of the central unit.
S_{A_5}	‘Fan-out’, corresponds the outlet fan of the central unit.
S_{A_6}	‘Electro valve plant bed 2’, corresponds to irrigation valve in ‘plant-bed 2’

Failure Modes from real-life study:

F_1	‘Tank leak’
F_2	‘Irrigation pipe blocked’
F_3	‘Irregular fan operation’

Signal Features from real-life study:

a_1	‘Derivative’ feature
a_2	‘Standard deviation’ feature
a_3	‘Median’ feature
a_4	‘Area’ feature

Chapter 1

Introduction

1.1 Background of the research

System failures have a strong impact on the performance of industrial systems as well as on the efficiency of the operational and servicing processes. According to the literature, system failures causes losses up to 20 billion US dollars per year in the petroleum and chemistry industry, and some 27 billion US dollars in the UK [1]. Due to faults and failures, systems work below their normal production capacities or qualities, with frequent and increased downtimes, and with a reduced trust and dependability. Having recognized these, multiple failure detection and diagnosis techniques have been developed in order to maintain continuous operation of industrial systems in a cost-effective way. The first approaches of failure detection and diagnosis techniques directly involved the system operators and system experts. They used to analyze system characteristics such as components wearing, sound and smell in order to determine abnormal events that could be associated with failures. But it was just until the early 70's when failure analysis technologies started to gain relevance with the advent of microcomputers and analog controllers [2]. They facilitated a shift from failure detection based on manual parameter measurement by limit checking to sophisticated methods of real time failure diagnosis and classification.

System paradigms, system technologies and system implementations have evolved considerably since the onset of the first failure detection and diagnosis techniques. These incipient types of techniques were developed for time-invariant engineered systems, which presented a predictable linear behavior. These deterministic systems used to be mono-functional and their operation largely depended on human intervention. However, the advent of low-cost sensors, high-capacity and sophisticated computing devices, powerful wireless networks, abundant internet bandwidth, and improvement in energy consumption and energy storage brought out important opportunities in terms of new services and system capabilities [3]. This has highlighted the need for scientifically based solutions in achieving continuous and reliable system operations. System engineers started to incorporate new functionalities and advanced computing technologies. They implemented sophisticated feedback control and reduced the human involvement during system operation. The integration of sensors, controllers and actuators provided new systems with growing, but still limited, decision-making capabilities. In like manner, the development of artificial intelligence-based algorithms, along with the integration of data transmission and distributed processing, enabled the evolution of the learning capabilities of ordinary, complicated and complex systems. There is a shift in the roles of humans in terms of interoperation with these kinds of systems. There is a move from the

execution/operation controller role through the (remote) supervisory controller role to strategic controller/planner role. On the other hand, physical processes have become highly dependent on systems performance.

The main change is that system maintenance has been forwarded from time-based maintenance to condition-based maintenance. This shift was enabled by a direct monitoring of system parameters with the aim of making judgement on system states and status. Failure analysis techniques rapidly evolved to assure continuous system operation. Researchers developed advanced methods to analyze system signals in both the time domain and the frequency domain with the aim to generate hints for detecting and diagnosing faults. Attention was given to real-time failure analysis and various approaches of failure management have been developed and brought into practical applications. Some of the strategies proposed for managing failure occurrences were based on the concept of redundant systems, and principles such as reasoning with variations of the operation intensity, switching to safe operation state, or turn the system off in critical situations were considered [4]. In addition, the developments in the field of robust control also added a lot to maintaining disruption-free or limited-disruption system operations and to preserving the stability of systems by manipulating system actuators, regardless of the presence of faults.

It can be recognized by studying the related literature and professional achievements that failure analysis techniques have evolved hand in hand with complex systems. Notwithstanding, in the current time, we are experiencing an important turning point in the evolution of engineered systems. A new family of systems, known as cyber-physical systems (CPSs), has emerged [5]. These systems are moving towards the implementation of data-enabled run-time decided upon, smart operation. The currently developed systems not only fully integrate physical devices with computational resources, but feature many novel paradigmatic features such as self-healing, building awareness, unsupervised learning, situation dependent strategizing, and context-based adaptation. CPSs realize a high level of interaction (actually interoperation) with their surrounding environments, and are capable to manage operational deviations and uncertainties should they be subjected to variable use and operating conditions. The move towards smart cyber-physical systems introduced remarkable changes in the normal operation of these systems. The latest cyber-physical systems are equipped with sophisticated control solutions. In fact, these enablers make them capable to introduce alteration or even adaptation in their routine operation in accordance with the internal and external changes. Furthermore, their self-tuning capability makes them able to regulate themselves according to emerging working contexts. From the perspective of the research presented in this thesis, a mentioning-worth consequence of this advancement is that the latest CPS control mechanisms are able to compensate efficiently for early-phase faults and slight failures on their own, but only up to a given ceiling, and not beyond. This compensatory operation is realized through subsequent interventions by the control sub-system.

The means used in the interventions are controlled settings of particular system operation modes (SOMs). By initiating various compensatory SOMs, first generation CPSs are able to change their settings quasi-autonomously, so that they can present a satisfactory performance in spite of the variations in the context conditions. This has been termed as the self-tuning behavior of CPSs. When the system self-tunes itself, it makes interventions in order to maintain the targeted overall behavior or servicing by operationalizing corrective operations or actions). These operations compensate for the unwanted

operational changes - in harmony with the alterations and deviations detected in the operational state and/or in the working context, respectively. Though the capability of self-tuning is unquestionably useful, it hampers the use of the currently existing failure analysis and failure management techniques. The non-linear and unpredictable system behavior, as well as the setting of different SOMs in the process of self-tuning, poses challenges from the perspective of failure analysis and failure management. The reason of this is that the existing failure analysis techniques are sensitive to non-controllable conditions such as the environmental and operational changes and the changes in terms of the system characteristics during operation [6].

1.2 Current trends in system engineering

The present day system area is full of terms such as embedded software systems, Internet of things, ubiquitous computing, Industry 4.0, cognitive robotics, cooperating agent systems, industrial internet, cyber-physical systems, intelligent application systems, and so forth. It was already mentioned that these existing and emerging systems display increasing complexity and heterogeneity. They moved from systems operating in the linear and deterministic realm to the realm of systems exhibiting dynamic and stochastic behaviors. This was enabled by: (i) miniaturization and embedding of physical system components, (ii) incorporation of advanced feedback control and other sophisticated control algorithms, (iii) proliferation of networking and transmission technologies, and (iv) the onset of public/industrial Internet, just to mention a few.

Feedback control contributed to the implementation of systems, which are more robust and stable when used in uncertain environments [7]. This type of systems measure the observable outputs of processes and compare them with the desired output at determining the necessary response actions [8]. By manipulating the system actuators they achieve a compensatory effect that is required for tolerating the operational deviations and malfunctions of the concerned system, while keeping its stability [9]. An apparently strong trend in the last two decades is to intellectualize engineering systems. In the practice, it means to incorporate features in systems that: (i) provide them with some form of intelligence, (ii) equip them with some ‘decision making’ capability, and (iii) enable higher-level autonomy in their operation. Despite these efforts, the majority of engineered systems still operate according to pre-determined working scenarios [10]. Only the near-future systems are expected to have more sophisticated self-regulation and self-adaptation capabilities.

Nowadays, around 98% of microprocessors are embedded in the physical components of engineered systems and a large part of them are connected directly to the outside world [11]. Software has become the most important integrator element of engineering systems featuring high functional and structural complexity, as it is embedded in every physical components [12]. The use of Internet services and data transmission technologies allowed opening systems boundaries and enabled the development of distributed architectures, which perform real-time communication and collaboration. Distributed systems incorporate several control loops, which are made closed by communication networks [13]. Locating the sensors and the actuator in distant units facilitates the fulfilment of the objectives (mission) of the system level, but poses challenges from the viewpoints of control, failure recognition, and corrective maintenance. This is especially important issue

in the case of controlling complex distributed infrastructures, and remote operation of system of systems.

Internet development brought out important opportunities too. It did not only allow the storage of data in the cloud. It also allows conducting remote processing and provided access to repositories of information and databases that can be used to support system operation. Here is where artificial intelligence based algorithms play a crucial role on systems operation. AI constitutes the first approach for providing intelligence on systems. It aims to mimic human mental processes, so that, it can be developed “machines that are able to think in a human like manner” [14]. Learning and pattern recognition capabilities are widely implemented nowadays through sophisticated algorithms such as neural networks, genetic algorithms, among others. It contributes to self-regulation capabilities as it provides means for autonomously recognizing different context of operation, and conducting decision making about variations on system settings.

1.3 Introducing the paradigm of cyber-physical systems

The paradigm of cyber-physical systems (CPSs) appeared in the field of systems research and engineering hardly more than a decade ago. However, cyber-physical computing, and the systems relying on it, rapidly penetrates into innumerable industrial, commercial, social and personal application domains. The acronym CPSs describes a family of systems that tightly connect the physical world with the information (cyber) world and obtain control information directly from real life processes, very often in real time in run time [15]. There are many definitions for CPSs at this moment, which intend to capture somewhat different aspects and essential features. For instance:

“CPSs are physical, chemical, biological and engineered systems whose operations are coordinated, controlled and monitored by a computing and communication core.” [3]

“CPSs involve digital computational, communication and control components, which closely interact with physical sensing and actuation components to enable better interaction with physical processes and environments.” [16]

“cyber-physical systems (CPSs) are confluences of knowledge and technologies of computing and informing, and knowledge and technologies of physical artefacts and engineered systems towards situated intelligent operation and servicing as actors in human and social contexts.” [17]

Dealing with cognitive engineering of CPSs, the Cyber-Physical Systems Design research group of the Faculty of Industrial Design Engineering at the Delft University of Technology interpreted the CPSs as:

“Smart anticipating multi-actor systems, which (i) bring analogue and digital hardware, control and application software, and data and knowledge inclusive cyberware into synergy, (ii) achieve deep diffusion into real life physical processes and objects, (iii) are enabled by cyber-physical computing, (iv) implement multiple and recurrent sensing-reasoning-learning-adapting cycles, (v) may have applications in industrial, commercial, social, and human contexts, (vi) create values by resource

and service provisioning, and (vii) represent a kind of model of future intelligent and autonomous systems.” [18]

It follows from the above definitions that CPSs blend physical technologies, software (middleware) technologies, and cyber technologies in a synergistic way [19]. Physical technologies include analog and digital hardware components which are located, operated and/or controlled in the physical world [20]. Advanced software technologies enable the development of computational algorithms and applications that capture, analyze and process data coming from the physical world. Cyber technologies focus on data, information, knowledge and media engineering and processing, and facilitate the development of data, information and knowledge (DIK) models, DIK structures, digital repositories, ontologies, and knowledge basis for reasoning. Synergic technologies combine functionalities and implementations originally belonging to one of the above domains.

CPSs constitute an incipient approach to future intelligent and autonomous systems. This system engineering paradigm offers a kind of a borderless interoperation between physical and cyber elements where the design of the computational aspects of physical, software and cyberware components is becoming a holistic and integrated activity [21]. This integration leads to a new pool of services that includes the autonomous and optimum control of complex infrastructures (such as nuclear plants, traffic systems, air control systems, among others), the monitoring of physical processes, and the provisioning of critical services in geographically distributed environments. Toward this end, the cyber, software and hardware parts of systems should achieve a high level synergy [22]. However, problem of synergistic operation of hardware, software and cyber elements is yet not completely resolved in these systems. It needs novel system engineering principles, which enable the implementation of compositional system features, as well as new operational (working) and architecting principles that go beyond component-based design and model-based system development and control.

CPSs are often connected in a hierarchical manner, as systems of systems, where one system monitors, coordinates, controls and integrates the operation of other systems [3]. For this reason, they can be considered as multi-dimensional complex systems [23]. The National Academy of Science and Engineering of Germany (Acatech) describes CPSs by an onion-like structure, which is composed of three main layers: (i) controlled core area, (ii) extended field of application, and (iii) cross domain networking [11]. The controlled core area is composed of embedded systems equipped with sensors, actuators and control capabilities. They enable interaction between the system and the environment. These components are task orientated and provide local control based on their set points and the feedback they get from sensors. The extended field of application allows the system and its components to cooperate in specific usage situations. In this context, the data coming from the controlled core area is used to determine response actions that contribute to the fulfillment of system level objectives, among others such as optimization of the performance of the system, reallocation of components, resource assignment. Finally, the cross-domain networking dynamically enables collaboration with external systems belonging to different domains.

One of the main characteristics that distinguish CPSs from traditional complex systems is their capability of functional and structural adaptation and (non-biological) evolution. They enable CPSs to change their system operation mode and structure with regards to provide an optimal behavior in different working conditions. It leads to multiple emergent

behaviors that deviates from predefined acting ways and which are determined in the running time [10]. Information obtained from the physical and cyber worlds is used as basis for determining the most optimal settings, according to the variable environment, operational and use conditions. Achieving self-adaptation and self-evolution is still a challenge. CPSs currently available are capable to conduct self-regulation and self-tuning. The implementation of feedback control in local and distributed way enables to conduct advanced process control, diagnosis and supervision, optimization, and planning and scheduling [24]. However, they are still closed systems with non-adaptive control.

1.4 The landscape of cyber-physical systems

1.4.1 Evolution of the paradigm of CPSs

Having recognized rapid change of paradigm of CPSs that happens in line with the overall trend of intellectualization of engineering systems, as well as the various possible implementation of CPSs, Horváth *et alias* introduced the concept of generations of CPSs [10]. They provided a reasoning framework that sorts CPSs into five classes (evolutionary stages) that range from the incipient conventional implementation up to the most sophisticatedly intellectualized one. According to their reasoning, there are five generations of CPSs, which are differentiated based on the levels of self-intelligence and self-organization. Zero generation CPSs are systems that utilize some partial implementations of cyber-physical computing and/or reflect a subset of the paradigmatic features of CPSs. The first generation CPSs are characterized by specific self-regulation and self-tuning capabilities. Second generation CPSs are able to build up self-awareness, implement reasoning and learning, and perform self-adaptation. Third generation CPSs are equipped with the capabilities of self-cognizance (building awareness and understanding simultaneously) and (non-biological) self-evolution. As the highest-level implementation, fourth generation CPSs are supposed to achieve self-consciousness and to implement self-reproduction on a system of systems (SoS) level.

Zero generation CPSs include linear and time-invariant systems, which are regulated by feedback-based control sub-systems, but whose set points are either predetermined or adjusted by the users through external controls. Representative examples of these are closed embedded systems, which are not capable to manage run-time variations. Moreover, they are not supposed to change their functionality or architecture, neither to optimize their behavior. The first generation of CPSs is characterized by self-regulation and self-tuning capabilities. In this type of systems the tight interaction between the physical and cyber elements provides the conditions required for the planned regular operation, as they embed intelligence in the physical world [25].

First generation CPSs are equipped with advanced feedback control systems that enable keeping system stability and provide reliable operation. However, they have only limited adaptation capabilities that allow them to modify the system set points as a respond to varying working conditions. In the case of these systems, the phenomenon of tuning the operation by shifting system operation modes (SOMs) can be observed. By purposefully changing SOMs, first generation CPSs can enable multiple system settings so that they can ‘adapt’ themselves to different use context and operational conditions in an optimal way. For instance, if it is necessary to operate over an extended period of time, they shall

achieve higher energy efficiency in operation [26]. The goal of energy efficiency is realized by changing the system settings and modifying the operation modes of the actuators run time according to the actual operational conditions. Typically, first generation CPSs are: (i) linear, (ii) closed, (iii) distributed and networked, (iv) sensing and reasoning enabled, (v) embedded and feedback controlled, and (vi) collaborative systems. They make the first step toward the implementation of intellectualized self-managing systems, which are able to show an anticipating (proactive) and context-aware behavior. Based on the sensed operation data they can make decision about shifting or switching from one SOM to another one in run time. Nevertheless, they cannot change their functionality and architecture.

Second and third generations of CPSs are usually non-linear, complex, open and decentralized, heterogeneous and multi-scale, increasingly intellectualized, partially autonomous, self-learning and context-aware systems. Humans are typically involved in the operation loop as supervisory controllers. The second generation of CPSs is characterized by self-awareness and self-adaptation capabilities, as paradigmatic features. Self-awareness is related to the capability of “constructing a secondary conceptual representation of itself” [27]. It allows not only building awareness of the surrounding environment, but also understanding the momentarily role of a particular system in a system of systems context [10]. This capability enables the system to learn from its own operation and experience so that it can optimize its performance and adapt itself to the context of operation [28]. Self-adaptation makes it possible to introduce operational and structural changes in the system, in order to adjust and respond as needed. The rearrangement of the system components and the modification of the system settings also entail the change of the system operation modes.

Third generation CPSs will be equipped with smart reasoning (software) components, which: (i) implement logical/semantic inferencing, (ii) learn in various contexts, (iii) adapt their structure to working situations, and (iv) evolve over a longer period of time of operation [29]. These systems will provide not only a tight connection between the physical and cyber elements, but also a high-level computational synergy of the knowledge-intensive components. Furthermore, they are controlled by operation strategy-planning non-conventional control technologies [30]. These technologies will allow them to develop their own operational strategy and to achieve a high level of automation and independence in comparison with the traditional complex systems. They will feature the capabilities of self-cognizance and self-evolution. Self-cognizance refers to the capability of developing multiple models of the surrounding world with the aim to determine new system configurations. These configurations will be enabled by a modular system composition that allows the system to evolve in order to meet the context and use requirements. This generation of CPSs will also be characterized by the emergence of unexpected system behaviors that is caused by the addition or subtraction of system components.

Finally, the fourth generation of CPSs will display organization without any predefined organizing principle and change their functionality, structure and behavior by self-learning, self-adaption, or self-evolving. Some of these systems will ought to operate in quasi real time applications and to provide a precisely timed behavior [31]. In addition, they are expected to achieve a truly synergetic interoperation between the physical and the cyber worlds and machine consciousness-based autonomy [19]. This generation is seen currently as an ultimate level of implementation of CPSs, featuring even a non-

genetically-based self-reproduction. This obviously raises the need for a run-time system resource management that is in its infancy nowadays. This level of operation, under strategic and supervisory control of humans, is already mentioned in the literature in the context of cybermatics systems. Research is facing a long road to provide the proper theoretical and methodological fundamentals, and to make the needed enabling technologies available.

1.4.2 Effects of the progression on the theory and practice of failure diagnosis and avoidance

All of the afore-mentioned generations of CPSs are deemed to operate according to run-time defined and adapted performance and behavioral objectives, and under dynamically changing operating conditions or even unforeseen circumstances. Our research project restricts itself to the analysis of the influence of changing SOMs on the recognition of emerging and progressing failures only in the case of the first generation of CPSs. This research phenomena itself has provided sufficient theoretical and experimental research opportunities in the framework of this promotion research. The obtained insights and results will be utilized as a starting point of the inquiry in the case of second generation CPSs, and will facilitate not only the investigations of failure mechanisms, but also failure diagnostics and prevention. We believe that based on an extensive study of the phenomenon of shifting SOMs the various system adaptation, system evolution, and system reproduction issues can be effectively addressed. The gained insights are supposed to provide mechanisms for systems that allow them to migrate from one particular working condition to another one, while keeping an optimal system performance. In the context of this promotion research, the study of the phenomenon of shifting SOMs was considered not only as a factor influencing failure recognition and forecasting, but also as a basis of developing efficient computational algorithms and tools for these purposes.

From a failure analysis and failure management perspective, the implementation of first generation and second generation of CPSs implies important challenges though. First and second generations of CPSs entail a sophisticated implementation of feedback control. This makes the systems dynamically controllable. However, it also mask failure effect on the output system signals [32]. It prevents timely failure detection and diagnosis, and the timely execution of corrective actions that avoid failure evolution. The complexity that information and communication systems entail, the improperly tools used, and the limited skills to deal with uncertain situations makes it urgent to develop new scientific principles and methodologies to create the CPSs upon which our lives will depend [33]. This introduces challenges from the point of view of dependability, maintenance and repair cyber-physical systems [34]. In real-time systems unforeseen changes, alterations on systems, and abnormal events will lead to use online measurement results to make decisions and adjust system's operations in real time [35]. However, those decisions require evaluating multiple aspects and data coming from a high number of components, (such as energy consumption, business objectives, time restrictions, deadlines, volume of work, among others). It may cause important delays on the decision process. A late decision could cause catastrophic problems, as well as important loss of money. Forecasting capabilities are desired in CPSs, so that system can anticipate to critical situations, and take decisions that enable preventing or ameliorating its negative effects.

We claim that a proper failure analysis in the context of cyber-physical systems require tackling the three main factors above discussed, namely: (i) dynamic system operation, (ii) masked failure effect due to control, and (iii) handling large amount of data in parallel. Dynamic system operation hampers the use of analytical models for failure detection, as well as the implementation of experience-based methods. Feedback control affects the implementation of data-driven failure analysis when evaluating output system signals. Big data handling affects the timely decision making of the system. We consider the use of input and output signals through a data-driven approach that may contribute to overcoming the aforementioned limitations. Considering the fact that forecasting is a desirable property in CPSs, we claim that implementation of failure forecasting in CPSs (i) helps anticipate failures, (ii) enables a timely decision-making, and (iii) overcome the unfavorable effects of big data handling.

1.5 Description of the concept of system operation modes

The behavior of systems can be observed and explained by inspecting the operation of the actuators and transformers incorporated in the systems, and the signals produced by the components and the system as a whole. It is widely known that system signals can provide information about both uninterrupted and interrupted system performance. It is also a fact that the combined effects of system actuators determine the entire system behavior. According to our viewpoint, the states of the system actuators and effectors in conjunction determine the system's operation mode. In turn, SOMs govern how the system responds to external and internal events. It allows the system to adapt its behavior to assure the desired system performance under different use and operational scenarios. The natural variations in the surrounding environment, as well as the frequent changes in the use conditions require that the system should present multiple operational behaviors. Every operational behavior is enabled by a particular combination of system settings. A system operation mode (SOM) describes a system's behavior at time t based on the actual system settings. It enables self-regulation, self-tuning and self-adaptation capabilities, as it provides the means for the system to modify its behavior through SOM transitions.

SOMs can be considered as a subset of the state concept, as they describe the situation of the system at a particular time t [36]. A system state is defined by a set of variables that in conjunction provides relevant information for characterizing a system behavior. The set of possible states a system can take are determined by the state space of the system [37]. The approach we will consider for analyzing SOMs is based on the input variables of the system. It means that system behavior at time t can be described through the joint operation of the system's actuators. SOM state space will be determined then, by the potential combinations of component operation modes determined by system actuators.

For the sake of a formal treatment, SOM has been defined as a singular combination of operation modes (COM) of all components of the system in a particular time t . COMs are regarded as the component state at a time t . The actuators can obviously be in multiple various states. As the most basic ones, we have considered the active and inactive states. For instance, the states of an outflow valve in charge of irrigation in a greenhouse can be symbolically represented as $\mathbb{E}_{S_{A_j}} = \{\text{ValveClose}, \text{ValveOpen}\}$, where \mathbb{E} denotes the set

of COMs of a particular component, and S_{A_j} indicates the signal coming from the actuator j (which is the outflow valve in the above example). Consideration of the COMs of all system actuators at a time t determines the particular SOM at a time t , so that:

$$\varsigma_d = \{\zeta_{S_{A_1}}(t), \zeta_{S_{A_2}}(t), \zeta_{S_{A_3}}(t), \zeta_{S_{A_4}}(t)\}$$

where: ς_d denotes the system's operation mode d , and $\zeta_{S_{A_{1-4}}}(t)$ denotes the component operation modes of (four) actuators, $j = 1$ to 4. SOMs are not necessarily associated with a single task. They may be related to several tasks concurrently.

The above reasoning clarifies the potential role of the concept of system operation mode in the context of self-tuning cyber-physical systems. It lends itself to a conceptual means needed for capturing the dynamic and adaptive system behavior in failure analytics. The actuators orientated thinking makes it possible to considering the entering and exiting of system components into operation, which normally leads to the occurrence of new SOMs. In like manner, transitions between SOMs makes it possible to consider the constraints defined by the control settings too, which play a role in the variation of the system operation modes and that influence the manifestation of system dynamics. In this promotion research, we will focus only on SOMs, which includes only discrete two-state (binary) COMs, i.e. active and inactive states, as a first approach to analyzing the effects of SOMs on failure analytics.

1.6 Brief overview of the current failure analysis approaches

1.6.1 Fundamentals of failure analytics and maintenance

Before going into the analysis of the currently existing failure analysis techniques, it is important to explain some crucial notion and terms that are widely used in the literature, but often times with a slightly or largely different meaning. The most pertinent terms are: (i) fault, (ii) error, and (iii) failure. A failure is an event that occurs when the service to be delivered deviates from correct service to incorrect service [38]. It is caused by errors which are the part of the system state that can lead to failures [39]. Faults are the hypothesized cause of errors [40]. A failure occurs when a fault-triggered error, is propagated and causes the service delivered to deviate from correct service [41]. When the system affected by the failure provide multiple services and functions, the failure of one or more of these services may leave the system in a degraded mode that can still provide some of the services it was designed to deliver [40]. This partial failure can evolve and start affecting the rest of the system's functions depending on its criticality and location.

Most of the failure analysis methods aim to detect and to manage faults. The reason is that faults are the first manifestations of failures. Several coinciding faults (more precisely, their effects in conjunction) constitute the characteristic set of symptoms of a particular failure mode. It is important to distinguish between failure analysis and system maintenance. The objective of failure analysis is to provide the means for understanding the occurring failure modes, specific manifestations of failures, the effects of failures, and

the root cause of failures. The objective of system maintenance is to assure continuous system operation by reducing the down and death times of a system or s system of systems. Maintenance relies on information generated by failure analysis in order to accomplish the mission of the system over its life cycle. Our research interest was in the failure analysis field, more specifically in failure detection and diagnosis, and did not extend to the state and issues of system maintenance.

There is no universal method for failure detection and diagnosis. Every single system requires development/configuration of dedicated failure diagnosis techniques that suit the characteristic of the system and its components. A priori knowledge of (i) system architecture, (ii) suitability of features for failure diagnosis, (iii) possible detection thresholds, (iv) and existing failure modes and their related symptoms is a prerequisite for the development of a dedicated failure analysis technique [42]. Failure diagnosis process is a set of sequential transformations through which system measurements lead to a decision about the occurring failure mode [43]. Data gathered in the measurement space is mapped into a feature space. It allows extracting system parameters to discriminate failed behavior from failure-free behavior and other failure modes. The measured features are then transformed in a decision space through discriminant or threshold functions. They determine if the observed features correspond to failure-free operation or to a particular failure-mode. Finally, the obtained results are interpreted in the decision space, where the decision about the occurring failure mode is delivered. A generic failure analysis can be intuitively conducted by answering the following questions:

- What system parameter should be observed?
- How is a failure manifested in the observed parameter?
- What reference should be used for judging the observed behavior, and
- What decision enabler is to be implemented for determining failure occurrence?

Answers to the above questions give insights into the key factors of failure analysis: (i) information engineering for failure analysis, and (ii) supporting decision making concerning failure analysis. Information engineering for failure analysis is composed by the failure information carriers and data features. The former are means through which failures are manifested. The last ones are system attributes that conveys relevant information about failures and that can be used for failure detection and diagnosis. The support of decision making about failures is mainly composed by references and decision enablers. References, determine the values or system characteristics from which it can be determined there is a failure. Decision enablers are failure indicators that are measured to determine if data features approach the reference value. A general overview of the above-mentioned key factors is presented as follows.

1.6.2 Information engineering for failure analysis

Failure information carriers constitute the parameters or system characteristics to be evaluated for determining failure occurrence. Traditionally, experts analyzed system degradation based on visual inspection, where wearing of system components was evaluated. Observable wearing signs, as well as acoustic signals were typically used as basic failure information carriers. However, the proliferation of sensor, processing, and wireless technologies, enabled the implementation of e-maintenance and condition based monitoring [44]. Currently, most machines depend on sensor-driven systems that provide

alerts and measure the most relevant system parameters [28]. They enable planning corrective maintenance actions, as well as fostering cost-effective maintenance [45].

Most of the currently existing failure analysis methods are based on the measurement of system parameters. System signals are widely implemented for determining failure occurrence, as well as their failure mode. These are considered not just in data-driven techniques, but also in model-based analysis and even in qualitative methods. Nowadays, the visual and acoustic information carriers are still used for failure detection, but through automated algorithms [46], [47], [48]. Data sets composed by sensed system parameters can be used for limit checking, or can be converted into data features in order to conduct failure detection and failure diagnosis through more sophisticated algorithms. For instance, system operational data distributed over time can be considered in the form of time series (i.e. as time-dependent signals), and can be used to determine the stage of degradation of the system. Processing sensed system signals is very effective at dealing with open-loop control, where the deviation of the output signal from the set control values is evaluated [49].

System operational signals are widely used for condition monitoring and failure diagnosis. Modern systems are highly instrumented - allowing sensing and measuring multiple system parameters. It enables not just determining system performance, but also, forecasting system behavior. However, managing large amount of data is still challenge in the era of Big Data. Data features are data attributes that convey relevant information about system performance. They allow retaining relevant information about failures, while discarding meaningless information [50] through a transformation process. It enables their use, in exchange of the raw system signals, contributing to data reduction, performance improvement, and data understanding [51]. Considering there are digital signal processing data features, and qualitative features we will use the term data features, as it comprises both of them.

In the data-driven domain, data features are the input to the failure detection or classification techniques. They are extracted from the raw data, as pre-processing, and delivered to the classification model. In this context, data features can be either categorical, binary, or continuous [51]. As for data-driven domain, model-based analysis also implements data features. In the case of qualitative models, data features are mostly semantic and categorical, so that they can describe a parameter status or a system state in a qualitative way. Moving from raw data to features is still a challenge. It is a trial-error process that depends on the experts' knowledge in every single domain [52]. Literature differentiates between two types of approaches for feature definition: (i) feature selection, and (ii) feature extraction/construction [53]. The former aims to select a set of characteristic descriptors from the original measurement space [43]. The latter implies the development of a new set of features from already derived features [53]. It allows, reducing data dimensions, standardization, signal filtering, discretization, non-linear expansion, among others [51]. These are desirable operations, as raw sensed signals convey noise, comes from multiple different parameters whose measurement units are non-comparable among them; and the overwhelming amount of data derived from intensive data sensing slow data processing.

There are multiple types of digital sensing data features, which can be considered for failure analysis. These can be divided into the time-domain features, frequency-domain features, and time-frequency features. The signal features we are presenting below are taken from [54]. Among others, time domain signal features include: (i) effective value,

(ii) standard deviation, (iii) skewness, (iv) kurtosis, and (v) root mean square (RMS). Some of them are of a statistical nature, while others are working with characteristic attributes (e.g. peak, slope etc.). The frequency domain signal features include: (i) power in dominant band, and (ii) power in other bands, such as, 62-125 Hz, 125-250 Hz, 250-500 Hz, etc. The time-frequency signal features include: (i) logarithmic energy, (ii) skewness, (iii) kurtosis, (iv) effective value, (v) threshold crossing rate, and (vi) pulse width. Another group of data features used in failure analysis is formed by those approaches that apply qualitative representations. For instance, data trends and direction of changes are widely used in qualitative trend analysis, as well as signed digraphs. Although qualitative signal features are the most suitable for understanding system performance, they are unreliable and inaccurate, and often difficult to capture.

An important characteristic of data features is that they belong to a specific domain of knowledge. Every single system requires determining the most suitable feature set depending on the system characteristics and failure manifestations. The reason is, that not all failures manifest in the same way. While some failures affect certain signal characteristics, for instance, the variance, other may affect RMS, or kurtosis. It makes it required a systematic exploration of system data when affected by a particular failure mode. It allows identifying the most representative data features. Thorough testing should also be conducted (for instance, cross correlation) in order to evaluate the selected features. Nevertheless, the lack of data proceeding from several failure episodes is required in order to succeed in data feature selection.

1.6.3 Supporting decision making in failure analysis

Providing information for decision making about the emergence and progression of failures is a crucial element failure analysis. Failures are deviations from a normal operation. A reference plays an important role in determining failure occurrence. It provides information for discerning if the analyzed data are within an acceptable region, or if they correspond to abnormal behavior. Moreover, they provide means not just for detecting failures, but also for diagnosing the occurring failure mode. These references can come from prior knowledge of the studied process, or can be the result of computing an analytical model. The most basic reference is a signal's threshold. It is widely used in limit checking approach, and has proved to be effective in linear time-invariant systems. However, models, state sequences, hyperplanes and curves are also used with this purpose. Models are analytical description of system operation. They use input system data for estimating system outputs. These outputs are used as reference for comparison with the observed system outputs in order to determine abnormal events. State sequence is based on the analysis of state transitions. It is underpinned on the assumption that failure-free behavior presents a particular sequence of transitions that characterize it. Deviations from such transitions can be interpreted as abnormal events. There are various concepts and means, such as: (i) signal threshold, (ii) mapping functions, and (iii) hyperplanes that support processing of system signals.

Signal threshold is operationalized through the limit checking strategy. It investigates if certain signals are within the pre-defined upper and lower limits [55], which are determined by prior knowledge of the expected system processes. Signal threshold is the most basic reference that can be used for failure analysis. It specifies acceptable values of a specific signal as well as its limits. Many faults are manifested as a deviation of the

observed data from the defined threshold [56]. Limit checking is widely used in: rule-based approaches (where experts determine the value of a threshold based on their knowledge and experiences), and in data distribution-based approaches [57]. In this last category, the probability value (p-value) is used as reference. It is derived by estimating the likeliness if a particular data sample belongs to a specific data distribution. In the literature, typically $p\text{-value} = 0.05$ is used for this purpose.

The use of signal threshold goes together with some essential drawbacks, in particular in the case of non-linear systems, which have unpredictable behavior. Their investigation requires the specification of a wide range of threshold values. A wide range of threshold values, in turn, makes the fault detection unreliable [58]. To overcome these limitations, adaptable thresholding methods have been proposed. These adaptable thresholds are mostly based on uncertainty models that consider statistical parameters such as mean and variance to estimate threshold change [59] [60]. These methods are capable to set the range of acceptable performance of a component in a system in various application contexts. However, they are focused on the isolated problem of component faults and do not consider the dynamic interrelationship between the various signals of the system [61]. This last issue is very relevant in the context of CPSs, as this type of systems present multiple system operation modes in response to internal operation and external use conditions. The multiple system operation modes imply a combined operation of system actuators that concurrently and variously influence the system signals.

The other means are mapping functions and hyperplanes. Data driven techniques allow determining functions that map a set of examples into a set of classes based on data density [62]. These functions can be regarded as boundaries and limits that determine the class of the observed data. If only two features are considered, the problem is limited to partitioning the feature space into two regions where the obtained function defines the decision boundary [63]. Linear discriminant analysis (LDA), logistic regression and artificial neural networks are underpinned on this principle. They enable deriving the function that determines the two regions, so that, data examples can be further classified into such regions. However, this curve becomes a hyperplane as the number of features increase. In that case, the decision boundary is not a curve, is a surface that delimits data in a \mathbb{R}^n feature space. It provides more discriminant power, as the number of features that characterize a failure increases. However, it also affects the processing time, and it difficult the graphical evaluation of the data.

1.6.4 Overview of indicators used as decision enablers

Several types of indicators have been defined and used for failure detection and diagnosis. We consider such indicators as decision enablers, as their computation enables determining not just failure occurrence, but also, failure detection. The most widely reported indicators in literature are: (i) residuals, (ii) statistical probability, and (iii) data distance. Although there are other decision enablers such as fuzzy rules, and qualitative deviation from references, in this section we will only analyze the most common ones, namely (i) residuals, (ii) statistical probability, and (iii) data distance.

Residual-based analysis checks the consistency between known variables, inputs and measured outputs based on comparing measured data to a system's model or measurements of redundant hardware components [64]. The literature distinguishes hardware and software redundancy. Hardware redundancy involves at least two identical

sensor or actuators as signal and data sources for characterizing the system performance from the perspective of fault diagnosis [65]. On the other hand, as discussed by Isermann, software redundancy involves residual estimation based on analytical models [66]. In this case residual indexes are used to compare the observed system's output with the one predicted by an analytical model and to identify discrepancies [67].

In the most general interpretation, residuals determine how large the difference between the observed variable and the reference one is. Given the obtained residual, judgement is made concerning the potential occurrence of a failure. However, generation of a residual index goes together with limitations since it mostly relies on the reliability of the analytical model of a complex system. Simplifications applied in the analytical models may cause a mismatch between the output of the model and the observed system behavior [68]. Furthermore, residual indicator is limited to specific system's functions that are widely known and predictable.

Statistical probability is widely used in data driven approaches methods and it constitutes one of the most relevant indicators, as it is domain independent. It means that, the same threshold can be used for evaluating all system signals indistinctively of the analyzed parameter. Statistical probability enables comparison whenever it is required for evaluating the effect and impact of a failure. It calculates and expresses the probability distributions of different data features - among other in such measures as (i) data mean, (ii) standard deviation, (iii) amplitude, and (iv) frequency) - in order to determine the likelihood of failures. Conventional numerical approaches such as linear regression and statistical tests make use of the abovementioned indicators. However, they lack the capabilities of noise rejection [69]. Other classification methods, which are more reliable at dealing with noise and data variance, also make use of statistical probability in the process of failure diagnosis. This is true in the case of Bayes classifiers, which estimates the probability distribution of a training data set that represent symptoms, given its class [70].

Data distance is also used as indicator for failure diagnosis. They determine the distance of any new observation with respect to the available classified failure clusters. It identifies the outer race fault as novelty [71]. This indicator facilitates unsupervised classification by recognizing hidden relationships in unlabeled data [72]. K-means based fault detection is one of the diagnosis methods that are underpinned by such an indicator. K-means clustering aims to partition a dataset into k clusters, based on their distance with respect to the center of the available clusters [73]. However, some classification methods such as support vector machines (SVM) are also based on the data distance indicator. A model generated by an SVM is a hyperplane that separates a dataset composed by examples from two different classes which are spatially separated [74]. Should be used as a failure indicator, distance index usually causes problems in managing overlapping classes [2]. These indicators can be used in combination in order to complement each other. For instance, in order to implement a residual-based failure diagnosis effectively it is required either to determine a fix threshold that enables deciding on whether the observed distance corresponds to a failure-free system behavior, or not, or enables the evaluation of its statistical probability. Although the afore-analyzed indicators do not constitute any drawback by themselves, the methods used for their estimation limit their applicability.

The above presented factors constitute the main steps to be conducted in most of the failure analysis methods. The right selection of failure information carriers, data features, references and decision enablers strongly depend on experts' knowledge. They are who

determine the way in which failures are manifested, based on their previous experience. They also determine the means required for failure detection and diagnosis. Output signals are by far the most widely used failure information carriers. Most of the data-driven techniques are fed by data features extracted from system signals. Residual is one of the most relevant decision enablers too. It can be used as optimization criteria in order to derive the data-driven models that are to be used for failure diagnosis. This combination of output signals and residual indicator has shown to be very effective in open-loop systems [49]. However, it is not the case of feedback control.

1.7 The phenomenon of changing SOMs as compensatory action

The adaptation capabilities required from CPSs are enabled by transitions between different system operation modes. They determine the settings required for assuring an optimal system operation. System signals convey information about current system and environmental status, providing self-awareness capabilities. Cyber technologies allow processing and analyzing system signals to realize self-tuning. It allows reacting to the observed operational and use conditions scenarios via actuators. Actuators execute the actions determined by the control system, making it possible the operationalization of self-tuning and self-adaptation. First generations of CPSs conduct self-tuning thanks to the intensive implementation of feedback control. In these types of controls “an output from an actor is fed back to affect an input of the same actor” [36]. The difference between the observed output signal and the defined set points determine the actuator status.

The role of SOMs in first generations of CPSs is to keep system stability according to operational and use conditions. Such stability is directly related to the mission and performance objectives of the system, as the smart augmentation that CPS implies go beyond of meeting functional requirements. It aims to obtain an optimal system operation. This stability favors a certain predictability level on system operation, while it is on a specific SOM. It is desirable, as currently existing failure analysis techniques can be used during these steady state operations. Self-tuning capabilities provided by the joint action of close-loop controllers allows tolerating malfunctions, while keeping the desired performance and stability [9]. It assures a continuous system operation despite the presence of faults. However, it also masks the effects of faults on system outputs hampering fault diagnosis [75]. This situation is particularly critical for developing failures, which cannot be predicted during their first forming stages.

Transitions between SOMs present important challenges in terms of failure analysis. They bring out uncertainty and short periods of instability on the system. It hampers the use of the currently available failure analysis methods, which have demonstrated to be sensitive to uncertainty, and changes to machine characteristics during operation [6]. This situation is more troublesome in the case of CPSs, whose operation may imply frequent operational variations that lead to frequent SOM transitions. While the failure manifestations masking effect caused by close-loop controllers may lead to false negative results, the effect of frequent SOM transitions may lead to false positive failure diagnosis. It causes unreliable failure detection and failure management, putting under risk system mission. According to our best knowledge, the role of shifting SOMs in failure analysis has not been studied yet, especially not in the context of first generation of cyber-physical systems. It is a very

critical topic, as many CPSs are mission critical-systems, which means that their correct functioning is critical to: (i) the success of a mission, (ii) provisioning an important supply, or (iii) safeguarding security and well-being [76] [77]. Activation of SOMs with the purpose of compensation for the effect of emerging failures is definitely the backbone of operationalization of first generation CPSs. For this reason, studying the implications of shifting SOMs on failure analytics is seen as an important research topic. It is paramount for developing new theories or other kind of knowledge by which future methods of failure analysis for CPSs can be underpinned.

1.8 Description of the research problem

Considering the aforementioned facts, the phenomenon to be studied in this promotion research has been formulated as follows:

There is a knowledge gap and investigations are needed to explore the roles played by system-initiated compensatory actions and operational mode changes in the emergence and proliferation of technical failures in the case of self-regulatory and self-tuning cyber-physical systems.

The conducted research is exploratory in nature and aims at providing new fundamental and descriptive knowledge about the implications and effects of SOMs on failure manifestations. At the same time, attention was given to both justification of the logical properness (coherence and consistence) of the developed theories. In addition, both internal validation (concerning the factors that might have caused biases) and external validation (concerning the applicability of the results in the specific context of a cyber-physical greenhouse testbed and beyond) actions were completed. The obtained results and the conducted approach can be reused as a basis for a follow up research with resembling objectives and contexts. The conducted research is multidisciplinary as it is based on the knowledge and methods of three disciplinary domains: (i) cyber-physical systems (self-management and failure recovery), (ii) failure analysis (means and approaches), and (iii) system engineering (operation modes and dependability).

Considering the above-mentioned factors, the addressed generic research question has been worded as follows:

What is the role of system operation modes in the failure analysis for first generation cyber-physical systems?

We decided to focus on cyber-physical systems due to their envisaged future role and rapid proliferation not only in various industrial sectors, but also in human and social application context. We had to take into consideration the fact that only the first generation of CPSs has manifestations in the above fields, whereas as the second generation of CPSs are still to come (though are just around the corner). With a view to a proper orientation and scoping of the research project, we argued that addressing the knowledge gap related to failure analysis of CPSs equipped with self-regulation and self-tuning capabilities will suffice. We trusted that a deep-going research would be able to provide insights necessary for the development of principles, methods and tools for a sophisticated failure management and prevention. We had to consider the research challenges that originate from the fact that the currently available methods and tools have been develop for the sake of ordinary and complicated linear systems, rather for self-

tuning CPSs, and therefore they present many limitations when applied in this new emerging domain of complex systems. In this research, we focused primarily on mechanical and sensor failures. A practical approach has been conducted that made use of simulations and an instrumented testbed as means for experimentation.

The main research objective addressed by this promotion research is to study the effect of SOM on failure manifestations on CPSs, and exploring its potential implementation for failure diagnosis and failure forecasting in the first generation of cyber physical systems. The overall objective was decomposed into five main research questions:

- What failure analysis processes are known currently and how do they deal with the phenomenon of SOMs?
- How the self-tuning and self-regulation properties influence the phenomenon?
- How the SOM influence failure manifestations on system signals?
- How the SOM affect the failure forming process, and how can it be used for forecasting?
- What are the challenges and opportunities that SOM bring out in the context of maintenance of CPSs?

We argue that answering the afore-mentioned research questions contributes to derive valuable and fundamental knowledge that underpin further research about SOM operationalization.

1.9 Research methodology

The overall structuring and the methodological framing of the completed research cycles are shown in Figure 1.1. In order to provide methodological support, the PhD research was broken down into five research cycles. Each of these cycles has a specific objective and methodological framing that seek to provide a systematic process that enables to get the knowledge required for answering the research questions. The considered methodological framing approaches are: (i) research in design context (RDC), (ii) design inclusive research (DIR), and (iii) practice driven research. The methodological conduct of the defined research cycles have been done according the principles presented in publication [78]. Below, we provide a concise explanation of the overall structuring and the methodological framing of the individual research cycles.

The first research cycle aims to explore the currently existing failure detection and diagnosis techniques, with regards to determine to what extent the SOM concept is implemented. For this purpose, failure information carriers, data features, decision enablers and references used by each of the studied methods are analyzed. We aim to aggregate knowledge about the limitations and opportunities the currently existing methods imply considering SOMs. For this purpose, research in design context methodology is implemented. A literature review is conducted for knowledge aggregation, and the obtained information is analyzed through a critical analysis, with a view to SOMs.

The second research cycle, aims to analyze the influential factors on the phenomenon of failure analysis. Research in design context is implemented here. This research cycle is divided in two main approaches, a theoretical approach and a practical approach. The theoretical approach aims to determine the implications of SOMs on the self-regulation

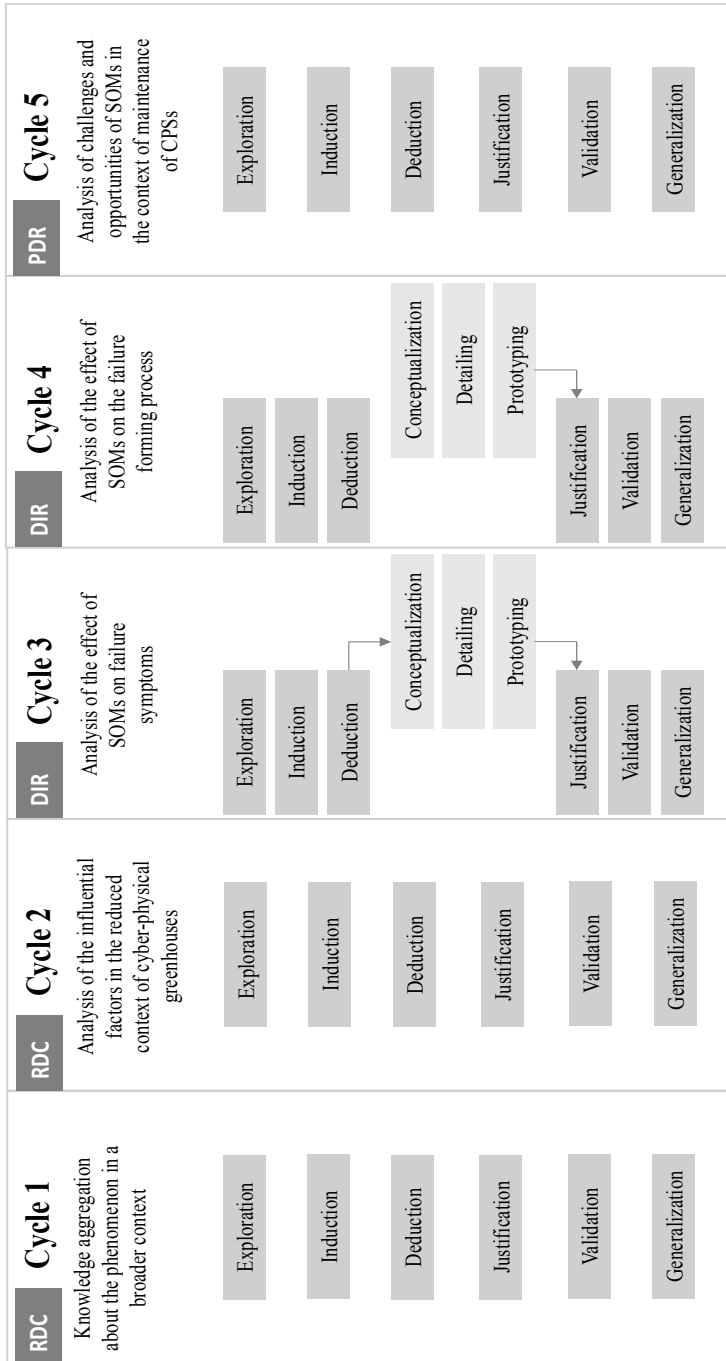


Figure 1.1. The overall organization and methodological framing of the research

and self-tuning properties of CPSs. The practical approach aims to determine the technical requirements for the instrumentation of the testbed, considering our experimentation purposes. In this research cycle, we narrowed down the context of the research to a particular family of CPSs, so that the findings contextualized by the case of a specific CPS, which was a testbed system of cyber-physical greenhouse.

The third research cycle aims to analyze the effect of SOM on failure symptoms, in the context of first generation of CPSs. The rationale behind this investigation is that segmenting system signals, based on SOMs, will remove the control effect over signals revealing failure effects. Design inclusive research is the methodology that guides this research cycle. For this purpose, a computational model, and the instrumented testbed will be used as basis for deriving knowledge. Failures were injected in both the computational model and the testbed, and a systematic exploration of the data was conducted. A failure indicator concept was introduced and elaborated in order to provide means for the exploration of the effects of SOMs on failure symptoms. Its application is evaluated based on the data coming from the simulation and the testbed.

The fourth research cycle aims at analyzing the effects of changing SOMs on the failure forming process, with a view to provide insights for failure forecasting. The findings obtained during the third research cycle will be used as basis for evaluating the failure forming process through SOMs. Design inclusive research is the methodology used for this research cycle. Here, failure evolution is analyzed, based on experimentation conducted through the computational model and the testbed. A forecasting concept is proposed as means for exploring the potential of SOM for failure prognosis. A demonstrative implementation of the concept will be conducted through data coming from the simulated computational model and the testbed. The obtained results will contribute to generate hypothesis about the potential implementation of SOM for failure diagnosis and failure forecasting.

The objective of the fifth research cycle is to evaluate the challenges and opportunities of using the concept of using SOMs-sensitive failure analytics in the context of preventive maintenance of 1G-CPSs. For this purpose, a practice-driven research is conducted in this cycle. Currently existing maintenance principles are analyzed and contrasted with the findings regarding the use of system operation modes-related indicators in failure recognition and forecasting. Based on the obtained results, a critical analysis of the challenges and opportunities was conducted and some propositions were formulated. It is expected that the obtained conclusions can be used as a basis for further research.

1.10 Structure of the thesis

This thesis book is composed by seven chapters and a summary section that is attached at the end of this manuscript. Throughout this book, the exploratory and the confirmatory activities are presented along with the findings. The research contents have been distributed over the Chapters as follows: In the current Chapter, the reader received information about the background of the research and the addressed research problem. The main objectives of the research were discussed together with the methodological framing of the completed research cycles. Chapter 2 analyses the state of the art of failure analysis methods with the intent of gaining insights in the road paving works and understanding to what extent the SOM concept has been explored previously. The existing

failure analysis methods were studied and compared, while a special attention was given to failure information carriers, data features, references, and decision enablers. The opportunities provided by the current methods for the introduction of the SOM concept were also studied, as well as the limitations posed by them. Chapter 3 analyses the self-regulation and self-tuning properties of CPSs and concludes about the necessity of investigating the role in and the influence of SOMs on failure diagnosis and forecasting. Based on a literature review, this Chapter also presents an overview of the technical requirements concerning the instrumentation of a greenhouse testbed system and the planned experimentation process.

Chapter 4 analyzes the effects of SOMs on failure manifestations by injecting failures in a computational model and then in the instrumented testbed. Chapter 5 studies the effects of SOMs on the failure forming process, proposes a forecasting approach, and presents a demonstrative application and the findings. Both the computational model and the testbed were used for experimentation. Chapter 6 places the findings in the context of the maintenance principles currently available for complex systems. It evaluates the relevance and appropriateness of the SOMs-based failure diagnosis and forecasting in preventive maintenance of CPSs. The objective of the work presented in this Chapter was to analyze the exploitation opportunities based on critical system reasoning. Finally, Chapter 7 discusses the whole PhD research work, concludes about the novelty and the scientific/professional value of the outcomes, and presents the list of the scientific proposition derived from the whole project. The thesis is completed with a list of propositions, which have been specified according to the Guide of the Doctoral Regulations.

1.11 Forerunning publications

1. S. Ruiz-Arenas, I. Horváth, R. Mejia-Gutierrez, E.Z Opiyo, “Towards the maintenance principles of Cyber-Physical Systems,” *Journal of Mechanical Engineering – SV*, pp. 815 – 831, 2014.
2. S. Ruiz-Arenas, I. Horváth, E. Opiyo and R. Mejia-Gutierrez, “Dual aspect model for failure forecasting in Cyber-physical systems,” in *Proceedings of the 11th TMCE 2014 Symposium*, Budapest, Hungary, vol. 1, pp. 1473-1486, 2014.
3. S. Ruiz-Arenas, Z. Rusak, S.R-Colina, R.Mejia-Gutierrez and I. Horváth, “Testbed for validating failure diagnosis and preventive maintenance methods by a low-end cyber-physical system,” in *Proceedings of the 11th TMCE 2016 Symposium*, Aix-en-Provence, France, vol. 1, pp. 1-11, 2016.
4. S. Ruiz-Arenas, Z. Rusák, I. Horváth and R. Mejia-Gutierrez, “Systematic exploration of signal-based indicators for failure diagnosis in the context of cyber-physical systems,” (accepted for publication at *Journal of Frontiers of Information Technology & Electronic Engineering*), 2017.
5. S. Ruiz-Arenas, Z. Rusák, I. Horváth and R. Mejia-Gutierrez, “A new approach to failure forecasting in the case of self-tuning cyber-physical systems,” in preparation for *journal publication*), 2018.

1.12 References

- [1] Y. Shu, L. Ming, F. Cheng, Z. Zhang, and J. Zhao, “Abnormal situation management: Challenges and opportunities in the big data era,” *Comput. Chem. Eng.*, vol. 91, no. Supplement C, pp. 104–113, Aug. 2016.
- [2] R. Isermann, “Fault diagnosis with classification methods,” in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, R. Isermann, Ed. Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 295–310.
- [3] R. (Raj) Rajkumar, I. Lee, L. Sha, and J. Stankovic, “Cyber-physical systems: the next computing revolution,” in *47th DAC.*, New York, NY, USA, 2010, pp. 731–736.
- [4] R. Isermann, “Fault-tolerant design,” in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, R. Isermann, Ed. Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 347–354.
- [5] B. J. Krämer, “Evolution of Cyber-Physical Systems: A Brief Review,” in *Applied Cyber-Physical Systems*, Suh S., Tanik U., Carbone J., Eroglu A, Eds. New York, NY, USA: Springer, 2014, pp. 1–3.
- [6] A. Bahga and V. K. Madiseti, “Analyzing Massive Machine Maintenance Data in a Computing Cloud,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 10, pp. 1831–1843, Oct. 2012.
- [7] Y. Liu, Y. Peng, B. Wang, S. Yao, and Z. Liu, “Review on cyber-physical systems,” *IEEECAA J. Autom. Sin.*, vol. 4, no. 1, pp. 27–40, Jan. 2017.
- [8] M.S Mahmoud, Y. Xia, “Nominal Control Design,” In *Applied Control Systems Design*, London, England: Springer-Verlag London, 2012, pp. 215-307
- [9] Y. Zhang and J. Jiang, “Bibliographical review on reconfigurable fault-tolerant control systems,” *Annu. Rev. Control*, vol. 32, no. 2, pp. 229–252, Dec. 2008.
- [10] I. Horváth, Z. Rusák, and Y. Li, “Order beyond chaos: Introducing the notion of generation to characterize the continuously evolving implementations of cyber-physical systems,” in *Proceedings of the ASME 2017*, 2017.
- [11] Acatech - National Academy of Science and Engineering, “Cyber-Physical Systems — Merging the Physical and Virtual Worlds,” in *Cyber-physical systems. Driving force for innovation in mobility, health, energy and production*, Berlin, Germany: Springer-Verlag Berlin Heidelberg, 2011, pp. 15 - 21.
- [12] J. Shi, J. Wan, H. Yan, and H. Suo, “A survey of Cyber-Physical Systems,” in *2011 WCSP*, Nanjing, pp. 1–6.
- [13] A. Çela, M. Ben Gaid, X.-G. Li, and S.-I. Niculescu, “Introduction to Distributed Control and Embedded Systems (DCES),” in *Optimal Design of Distributed Control and Embedded Systems*. Cham, Switzerland: Springer, Cham, 2014, pp. 3–8.
- [14] S. G. Tzafestas, “Artificial Intelligence,” in *Roboethics*, Switzerland: Springer, Cham, 2016, pp. 25–33.
- [15] E. A. Lee, “The Past, Present and Future of Cyber-Physical Systems: A Focus on Models,” *Sensors*, vol. 15, no. 3, pp. 4837–4869, Feb. 2015.
- [16] E. A. Lee, “CPS foundations,” in *Proc 47th DAC.*, New York, NY, USA, 2010, pp. 737–742.
- [17] I. Horváth, Z. Rusák, Y. Hou, and L. Ji, “On some theoretical issues of interaction with socialized and personalized cyber-physical systems.,” in *Proc. ADCP 2014*, 2014, pp. 1995–2000.

- [18] I. Horváth, “Humans. Technologies. Systems. Contexts. Together. Do we understand the design challenges?” presented at the Scientific Committee of Engineering, Hungarian Academy of Sciences, 2017.
- [19] R. Baheti and H. Gill, “Cyber physical systems,” in *The Impact of Control Technology*, T. Samad and A.M. Annaswamy, Eds. Boston, USA: IEEE Control Systems Society, 2011, pp. 1-6 [Online]. Available: Available at www.ieeecss.org
- [20] S. Kumara, L. Cui, and J. Zhang, “Sensors, networks and internet of things: research challenges in health care,” in *Proc.8th International Workshop on Information Integration on the Web: in conjunction with WWW 2011*, New York, NY, USA, 2011, p. 2:1–2:4.
- [21] G. Karsai and J. Sztipanovits, “Model-Integrated Development of Cyber-Physical Systems,” in *Software Technologies for Embedded and Ubiquitous Systems*, U. Brinkschulte, T. Givargis, S. Russo Eds., vol. 5287, U. Brinkschulte, T. Givargis, and S. Russo, eds. Berlin, Germany: Springer Berlin Heidelberg, 2008, pp. 46–54.
- [22] L. Hu, N. Xie, Z. Kuang, and K. Zhao, “Review of Cyber-Physical System Architecture,” in *2012 IEEE ISORCW*, pp. 25–30.
- [23] P. Li, Q. Xu, and N. Wang, “Evolution Course and Analysis of Internet of Things,” in *Proc. IEA 2012*, pp. 221–228.
- [24] M. S. Mahmoud and P. Y. Xia, “Introduction,” in *Applied control system design*, London, England: Springer- Verlag London, 2012, pp. 1–10.
- [25] M. Lora, R. Muradore, D. Quaglia, and F. Fummi, “Simulation alternatives for the verification of networked cyber-physical systems,” *Microprocess. Microsyst.*, vol. 39, no. 8, pp. 843–853, Nov. 2015.
- [26] A. C. S. Beck, C. A. L. Lisbôa, L. Carro, G. L. Nazar, M. M. Pereira, and R. R. Ferreira, “Adaptability: The Key for Future Embedded Systems,” in *Adaptable Embedded Systems*, A. C. S. Beck, C. A. L. Lisbôa, and L. Carro, Eds. New York, NY, USA: Springer New York, 2013, pp. 1–12.
- [27] P. R. Lewis, A. Chandra, and K. Glette, “Self-awareness and Self-expression: Inspiration from Psychology,” in *Self-aware Computing Systems*, Cham, Switzerland: Springer, Cham, 2016, pp. 9–21.
- [28] J. Lee and H. Wang, “New Technologies for Maintenance,” in *Complex System Maintenance Handbook*, London, England: Springer London, 2008, pp. 49–78.
- [29] P. Leitão, A. W. Colombo, and S. Karnouskos, “Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges,” *Comput. Ind.*, vol. 81, pp.11-35, Sep.2016.
- [30] C. D. Galvan et al., “Review. Advantages and disadvantages of control theories applied in greenhouse climate control systems,” *Span. J. Agric. Res.*, no. 4, pp. 926–938, 2012.
- [31] R. M. Santos, J. Santos, and J. D. Orozco, “Power saving and fault-tolerance in real-time critical embedded systems,” *J. Syst. Archit.*, vol. 55, no. 2, pp. 90–101, Feb. 2009.
- [32] R. Isermann, “Supervision, fault-detection and fault-diagnosis methods – a short introduction,” in *Combustion Engine Diagnosis: Model-based Condition Monitoring of Gasoline and Diesel Engines and their Components*, Berlin, Germany: Springer Vieweg, 2017, pp. 25–47.
- [33] L. Miclea and T. Sanislav, “About Dependability in Cyber-Physical Systems,” *2011 9th EWDTS.*, Sevastopol, Ukraine, 2011, pp. 17-21.

- [34] Horváth and Gerritsen, "Cyber-Physical Systems: Concepts, technologies and implementation principles," presented at the TMCE 2012, Karlsruhe, Germany, 2012, pp. 19–36.
- [35] L. Rodríguez-Mazahua, C.-A. Rodríguez-Enríquez, J. L. Sánchez-Cervantes, J. Cervantes, J. L. García-Alcaraz, and G. Alor-Hernández, "A general perspective of Big Data: applications, tools, challenges and trends," *J. Supercomput.*, vol. 72, no. 8, pp. 3073–3113, 2016.
- [36] E. A. Lee and S. A. Seshia, "Discrete Dynamics" in *Introduction to embedded systems, A cyber-physical systems approach*. Berkeley, CA, USA: MIT Press, 2011, pp. 41–71.
- [37] M. Kunze and M. Weske, "State Spaces," in *Behavioural Models: From Modelling Finite Automata to Analysing Business Processes*, M. Kunze and M. Weske, Eds. Cham, Switzerland: Springer International Publishing, 2016, pp. 163–185.
- [38] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," in *Dependable Secure Comput.* IEEE Trans. On, vol. 1, no. 1, pp. 11–33, Mar. 2004.
- [39] J.-C. Laprie, "Dependability — Its Attributes, Impairments and Means," in *Predictably Dependable Computing Systems*, B. Randell, J.-C. Laprie, H. Kopetz, and B. Littlewood, Eds. Berlin, Germany: Springer Berlin Heidelberg, 1995, pp. 3–18.
- [40] A. Avizienis, J.-C. Laprie, and B. Randell, "Dependability and Its Threats: A Taxonomy," in *Building the Information Society*, R. Jacquart, Ed. Springer US, 2004, pp. 91–120.
- [41] A. Bondavalli, A. Ceccarelli, F. Brancati, D. Santoro, and M. Vadursi, "Differential analysis of Operating System indicators for anomaly detection in dependable systems: An experimental study," *Measurement*, vol. 80, pp. 229–240, Feb. 2016.
- [42] A. Shui, W. Chen, P. Zhang, S. Hu, and X. Huang, "Review of fault diagnosis in control systems," in 2009 CCDC, pp. 5324–5329.
- [43] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis Part I," *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 293–311, Mar. 2003.
- [44] A.D Márquez, "The E-maintenance Revolution," in *The Maintenance Management Framework: Models and Methods for Complex Systems Maintenance*, London, England: Springer London, 2007, pp. 305–327.
- [45] J. Kautto, C. Roux, I. Elias, and A. Numminen, "Remote Condition Based Maintenance in Modern Life Cycle Services," in *Proc. WCEAM 2015*, K. T. Koskinen, H. Kortelainen, J. Aaltonen, T. Uusitalo, K. Komonen, J. Mathew, and J. Laitinen, Eds. Cham, Switzerland: Springer International Publishing, 2016, pp. 347–356.
- [46] J. A. Tsanakas, L. Ha, and C. Buerhop, "Faults and infrared thermographic diagnosis in operating c-Si photovoltaic modules: A review of research and future challenges," *Renew. Sustain. Energy Rev.*, vol. 62, pp. 695–709, Sep. 2016.
- [47] Z. Shen-pei, L. Xi, Q. Bing-chen, and H. Hui, "Research on Insulator Fault Diagnosis and Remote Monitoring System Based on Infrared Images," *Procedia Comput. Sci.*, vol. 109, pp. 1194–1199, Jan. 2017.
- [48] B. V. Hecke, J. Yoon, and D. He, "Low speed bearing fault diagnosis using acoustic emission sensors," *Appl. Acoust.*, vol. 105, pp. 35–44, 2016.

- [49] R. Isermann, "Fault detection of control loops," in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. R. Isermann, Ed. Berlin, Germany: Springer-Verlag, 2006, pp. 253–265
- [50] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin, "A review of process fault detection and diagnosis Part III," *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 327–346, Mar. 2003.
- [51] I. Guyon and A. Elisseeff, "An Introduction to Feature Extraction," in *Feature Extraction: Foundations and Applications*, I. Guyon, M. Nikraves, S. Gunn, and L. A. Zadeh, Eds. Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 1–25.
- [52] H. Zhao, A. P. Sinha, and W. Ge, "Effects of feature construction on classification performance: An empirical study in bank failure prediction," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 2633–2644, Mar. 2009.
- [53] H. Motoda and H. Liu, "Feature selection, extraction and construction," *Commun. IICM Inst. Inf. Comput. Mach. Taiwan*, vol. 5, pp. 67–72, 2002.
- [54] K. Jemielniak, T. Urbański, J. Kossakowska, and S. Bombiński, "Tool condition monitoring based on numerous signal features," *Int. J. Adv. Manuf. Technol.*, vol. 59, no. 1, pp. 73–81, Mar. 2012.
- [55] R. Fujimaki, T. Yairi, and K. Machida, "Adaptive limit-checking for spacecraft using relevance vector autoregressive model," presented at the 8th iSAIRAS, Munich, Germany, 2005.
- [56] D. M. Johnson, "A review of fault management techniques used in safety-critical avionic systems," *Prog. Aerosp. Sci.*, vol. 32, no. 5, pp. 415–431, 1996.
- [57] P. He, G. Liu, C. Tan, and Y. Lu, "Nonlinear fault detection threshold optimization method for RAIM algorithm using a heuristic approach," *GPS Solut.*, pp. 1–13, Nov. 2015.
- [58] C. Lee, D. Lee, J. Koo, and J. Chung, "Proactive Fault Detection Schema for Enterprise Information System Using Statistical Process Control," in *Human Interface and the Management of Information. Designing Information Environments*, Berlin, Germany: Springer Berlin Heidelberg, 2009, pp. 113–122.
- [59] J. Qi, X. Zhao, Z. Jiang, and J. Han, "An Adaptive Threshold Neural-Network Scheme for Rotorcraft UAV Sensor Failure Diagnosis," in *Proc. 4th ISNN 2007*, Nanjing, China, pp. 589–596.
- [60] K. Patan, "Decision Making in Fault Detection," in *Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes*, vol. 377, Berlin, Germany: Springer Berlin Heidelberg, 2008, pp. 123–140.
- [61] E. Sobhani-Tehrani and K. Khorasani, "Fault Detection and Diagnosis," in *Fault Diagnosis of Nonlinear Systems Using a Hybrid Approach*, Boston, MA, USA: Springer, 2009, pp. 21–49.
- [62] Ó. M. Mozos, "Supervised Learning," in *Semantic Labeling of Places with Mobile Robots*, Berlin, Germany: Springer Berlin Heidelberg, 2010, pp. 7–13.
- [63] G. Dougherty, "Supervised Learning," in *Pattern Recognition and Classification: An Introduction*, New York, NY, USA: Springer New York, 2013, pp. 75–98.
- [64] S. Tornil-Sin, C. Ocampo-Martinez, V. Puig, and T. Escobet, "Robust Fault Diagnosis of Nonlinear Systems Using Interval Constraint Satisfaction and Analytical Redundancy Relations," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 44, no. 1, pp. 18–29, Jan. 2014.
- [65] W. Hwang, K. Han, and K. Huh, "Fault detection and diagnosis of the electromechanical brake based on observer and parity space," *Int. J. Automot. Technol.*, vol. 13, no. 5, pp. 845–851, Aug. 2012.

- [66] R. Isermann, "Supervision and fault management of processes — tasks and terminology," in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 13–30.
- [67] S. X. Ding, Ed., "Introduction," in *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*, Berlin, Germany: Springer Berlin Heidelberg, 2008, pp. 3–11.
- [68] V. Puig, T. Escobet, R. Sarrate, and J. Quevedo, "Fault Diagnosis and Fault Tolerant Control in Critical Infrastructure Systems," in *Intelligent Monitoring, Control, and Security of Critical Infrastructure Systems*, Berlin, Germany: Springer Berlin Heidelberg, 2015, pp. 263–299.
- [69] H. M. Elattar, H. K. Elminir, and A. M. Riad, "Prognostics: a literature review," *Complex Intell. Syst.*, vol. 2, no. 2, pp. 125–154, Jun. 2016.
- [70] M. Ramoni and P. Sebastiani, "Robust Bayes classifiers," *Artif. Intell.*, vol. 125, no. 1, pp. 209–226, Jan. 2001.
- [71] J. Zhang, Q. Yan, Y. Zhang, and Z. Huang, "Novel Fault Class Detection Based on Novelty Detection Methods," in *ICIC 2006 Kunming, China*, pp. 982–987.
- [72] A. Abdolsamadi, P. Wang, and P. Tamilselvan, "A Generic Fusion Platform of Failure Diagnostics for Resilient Engineering System Design," in *ASME 2015., vol. 2A: 41st Design Automation Conference*, pp. 1-10.
- [73] J. MacQueen, "Some methods for classification and analysis of multivariate observations." in *Proc. of the 5th Berkeley Symp. on Mathematical Statistics and Probability*, 1967, pp. 281–297.
- [74] C. Cortes and V. Vapnik, "Support-Vector Networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.
- [75] X.-J. Li and G.-H. Yang, "Adaptive fault detection and isolation approach for actuator stuck faults in closed-loop systems," *Int. J. Control Autom. Syst.*, vol. 10, no. 4, pp. 830–834, Aug. 2012.
- [76] T. S. Perraju and G. Uma, "Mission Critical Intelligent Systems," in *Intelligent Knowledge-Based Systems*, C. T. Leondes, Ed. Boston, MA, USA: Springer, Boston, MA, 2005, pp. 1184–1210.
- [77] M.-O. Stehr et al., "Fractionated Software for Networked Cyber-Physical Systems: Research Directions and Long-Term Vision," in *Formal Modeling: Actors, Open Systems, Biological Systems*, G. Agha, O. Danvy, and J. Meseguer, Eds. Berlin, Germany: Springer Berlin Heidelberg, 2011, pp. 110–143.
- [78] I. Horváth, "Comparison of three methodological approaches of design research," presented at *ICED'07*, Paris, France, 2007.

Chapter 2

State of the art review

2.1 Aggregation of knowledge concerning the state of the art

2.1.1 General objective of this study

With the proliferation of cyber-physical systems and various other complex adaptive systems failure analysis and forecasting has become an important research topic. Methods and techniques of failure analysis and forecasting currently existing aim to prevent failures and maintain continuous fault free operation of systems. However, transitions between multiple system operation modes (self-regulation and self-tuning capabilities) that CPSs imply, leads to emergent and dynamic system behavior. This situation questions the applicability of existing failure diagnosis and forecasting techniques for preventing failures in a cost-effective way. The objective of this chapter is to aggregate knowledge about the currently available failure analysis techniques, and to analyze the extents and manners of considering shifting SOMs in these techniques. Towards this end, the various subfields of failure analytics and forecasting, and the technical aspects of their implementation and application are discussed in this Chapter.

The overall (guiding) research question for the study presented in this Chapter has been formulated as follows:

What sorts of failure analysis techniques and processes have been developed and how do they deal with the phenomenon of changing system operation modes?

In order to address the objective of the knowledge aggregation systematically and purposefully, the above general research question was decomposed into four specific research questions. These subordinate research questions are:

What system parameters are used by the current failure detection techniques?

How are failure manifestations observed and recognized based on these parameters?

What references are used or should be used for judging on the emergence or existence of failures?

What specific decision enablers have been, or should be, implemented for determining the occurrence of failures?

Our argumentation was that answering the above-presented questions could lead us to a proper understanding not only the essence of the available methods, but also the potentials for considering the changing system operation modes (SOMs) in failure analysis and forecasting. The collected literature was carefully filtered in order to provide a robust basis for inferring about to what extent and in which manner SOMs are considered in current failure analysis processes. Our first observation was that the term ‘system operation mode’ was not used in the literature in the same way as we have interpreted it in the context of dynamic failure analysis and forecasting. This was a significant observation since we would have liked to figure out if and how the concept of SOMs was used in the recently proposed approaches and how it was handled technically by existing techniques.

2.1.2 The research approach

In order to aggregate the required knowledge, a literature review and a keyword based web search was simultaneously conducted. The objective of the literature review was to find theoretical information from scientific articles, books and reports. Towards this end, we investigated that part of the potential knowledge sources, which could be supposed to provide explanation about the existing methods of failure analysis, more specifically, about those articles, which describe how these methods have been implemented. Among others, keywords such as: ‘fault detection’, ‘fault diagnosis’, ‘forecasting’, ‘maintenance of complex systems’, ‘system dependability’, ‘failure management’, ‘system availability’, ‘system reliability’, ‘maintenance’, and ‘signal-based analysis’ were considered in the study. The analyzed journal articles and conference papers belonged to various fields such as mechanical engineering, computer science, electronics, robotics, software engineering, and industrial engineering. Among others, Springer link, Science direct, IEEE, Emerald, and Google Scholar were consulted as the major scientific source databases.

The practical side of failure analysis and forecasting has been covered by collecting investigating information through query-based web searches. In this part of the study, we aggregated information from complementary sources such as: (i) corporative web pages, (ii) commercial information portals, (iii) experts’ blogs, and (iv) professional videos. Thematic research videos were also used and open web-based courses were attended with the aim of collecting information about the above-discussed questions. The specific objective of the completed information aggregation was to explore the current situation concerning the implementations of the various methods and techniques, and to analyze their real-life applications and implications.

Although our bibliography includes articles from different periods, we tried to concentrate our search to articles and papers back from 2005. The main reason was that this point in time was when the paradigm of cyber-physical systems popped up. Nevertheless, we also collected papers from the earlier period, in particular those theoretical articles and books, which presented fundamental information about the proposed methods and tools and whose focus was on non-linear, complicated, adaptive and complex systems. However, it was observed that, on the one hand, the approaches from the early time did not match to the specificities of CPSs, and, on the other hand,

the research and application of some particular techniques decreased after the early period. In the process, preference was given to the most relevant and the most cited publications.

Our literature study started through the evaluation of already existing reviews and fundamental books. It provided an overall perspective about the main subfields involved in failure analysis and their relevance. Afterwards, we explored some of the literature cited in such documents in order to deepen on the mentioned topics. We supported our knowledge aggregation process with videos and complementary material available in the web. Once we got sufficient knowledge about the studied topics, we proceeded to analyze practical application cases, by investigating conference papers, journal articles with emphasis in industrial application and corporate webpages. The obtained information was consolidated by filtering and was compared with other information from equivalent sources with the aim of data triangulation.

2.1.3 The reasoning model

Our analysis of existing reviews revealed that most of the currently analyzed failure analysis techniques can be clustered into: (i) model-based methods and (i) data-driven approaches [1] [2] [3] [4]. However, a significant number of articles have been published on signal-based failure analysis. This raised our attention to this important topic and we included it as another specific domain of failure analytics in our study. While the existing methods belonging to the cluster of data-driven approaches consider historical measurements regarding the whole system for failure analysis, signal-based analysis approaches aim at generating insights about failures by exploring time-domain, frequency-domain and time-frequency domain related information. Model-based approaches, in turn, design and implement analytical system/process models as reference and compare their outputs with observations in real-life.

Considering the above-mentioned situation, the reasoning model that guided our literature review was based on the three main families of approaches, namely on: (i) model-based failure analysis, (ii) data-driven failure analysis, and (iii) signal-based failure analysis (Figure 2.1). These approaches reflect important differences in terms of: (i) the observed system parameters, (ii) the assumed manifestation of failures, (iii) the data used for generating the features, and/or (iv) the used references and decision enablers. The study of these approaches is of paramount importance since these can provide the fundamentals for approaches to dynamic failure analysis of CPSs. It can be foreseen that consideration of adaptability and evolution of CPSs may imply the need for largely different approaches. On the other hand, the traditional families of approaches mentioned above, in particular those, which are based on analytical representation of system behavior, pose strong limitations in this context.

We have studied the three afore-mentioned families of approaches in details. A technical detail of importance is that the total number of publications and web sources were not in balance for the three families of approaches. For each family of approaches, we have considered and analyzed: (i) the information carriers, (ii) the data features, (iii) the references (kinds and values), and (iv) the decision enablers. The issue of self-managed manipulation of system operation states was analyzed in each of the three families of approaches. Our intension was to explore if the existing failure analysis and forecasting methods have considered the phenomenon of self-managed manipulation of

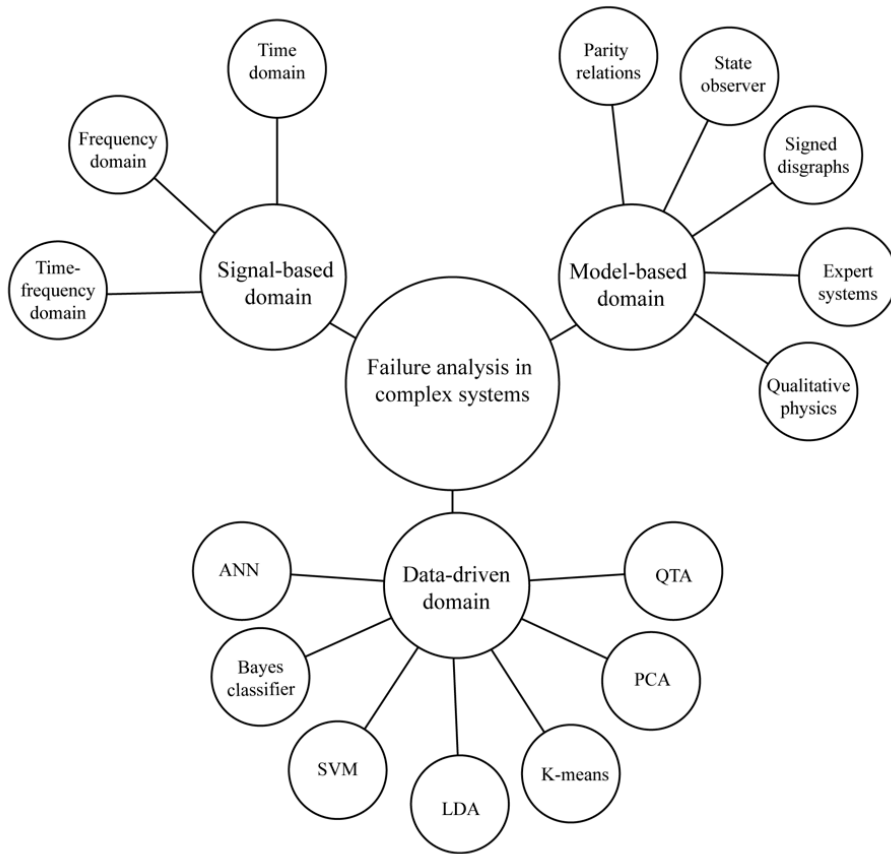


Figure 2.1. Reasoning model for knowledge aggregation

system operation states, and if so, which way they have operationalized it. In the end, these pieces of information were used to reason about which methods and tools could potentially be used for failure analysis in CPSs. The aggregated information was also used to determine which characteristics were relevant for backing up the exploitation of the SOM concept and, more specifically, for supporting the compensatory use of system operational modes in failure analysis and forecasting.

2.1.4 Overview of the challenges of aggregating knowledge and investigating cyber-physical systems

The challenges of knowledge aggregation can be traced back to two main sources. One is the complexity of the knowledge domain, which can be explained by the fact that failure analysis and forecasting is a long time existing and investigated phenomenon in the context of ordinary and complex engineered (technical) systems. However, the overwhelming majority of publications are focusing on traditional engineering system,

and only the recent ones focus on issues and solutions associated with failure analysis and forecasting for zeroth and first generation of cyber-physical systems and various other complex adaptive systems. The other challenge is the inherent heterogeneity of cyber-physical systems. These systems include analog and digital hardware constituents, control and application software constituents, and knowledge and data cyberware constituents. They need different approaches and the current literature reflects even different cultures.

Due to the premature and emergent nature of cyber-physical systems development, implementation and application, an integral handling of these constituents in the perspective of dedicated failure analysis and forecasting approaches is still in its infancy. The total number of the collected scientific and professional publications reflects this fact. The issue of heterogeneity is intertwined with the issues associated with real-time processing and massive data handling that are implemented by these systems. This further stretched the range of challenges that we had to take at completing our literature study. For the sake of completeness, we must also mention as an issue the variety of technologies currently available to enable detecting and diagnosing failures in and to assure continuous system operation in a cost-effective manner of traditional and complex adaptive systems. In order to deliver more information about the observed knowledge gap and the multiplicity of the issues related to cyber-physical systems, we provide further details about their main paradigmatic features below:

Increased system complexity:

CPSs typically include large number of interconnected actor nodes, each of which integrated a large number of components, as well as a large number of operational and structural relationships among the nodes and components, and with the embedding environment. The operation of CPSs is characterized by: (i) a high number of feedback controls, (ii) a relatively high level of automation, and (iii) behavioral dynamism and/or non-linearity. Like traditional (linear) complicated systems, CPSs also work on non-dedicated networks [5]. Furthermore, CPSs are frequently interconnected in a hierarchical manner, as systems of systems, where one system monitors, coordinates, controls and integrates the operation of other systems [6]. For this reason, they can be considered not only as multi-node, but also as multi-dimensional complex systems [7]. The behavior of the individual components is highly influenced by their interaction with other components and subsystems, so that they are more susceptible to cascade failures. This fact means that ‘time to failure’ decreases as the size and complexity of a distributed system increases [8]. However, the application of current methods do not suffice to manage these system characteristics, as most of them have been developed for linear systems and to specific domains and working conditions [9].

Dynamic and emergent behavior:

A large part of cyber-physical systems shows dynamic (context and time dependent), even emergent (incidental situation dependent), behavior. The reason, among others, is their tight connection and interrelation with the external environment that makes them very susceptible to non-controllable situations and conditions. These characteristics entail that specific attention is to be given to the various application domains and cases, and to the environment-related specific information. On the other hand, most of the methods available for analyzing reliability and failure data typically address main stream data and present errors due to incomplete domain information and non-

controllable conditions such as environmental and operational influences [10]. They do not fulfil the expectation of providing the means for managing faults if system characteristics vary during operation. One of the reasons can be that the actual reliability indicators of the system behavior may differ from the predicted values [10]. Consequently, there is an inherent challenge for failure management in the context of CPSs, if they show dynamic and emergent behaviors.

Real-time operation:

Another crucial characteristics of a sub-family of CPSs are real-time operation, communication, monitoring and decision-making [11]. These enable the accomplishment of the system mission and delivering services in the shortest possible time or in near-zero time. However, due to external environmental effects or internal work allocation conditions, the concerned systems may have to manage complicated tasks execution scenarios, communication latencies and drops that may delay recognition of operational errors and malfunctioning. They also need to make decisions on the necessary corrective or preventive interventions. In the end, these can result in a delayed or late fault detection, which may in turn lead to a permanent or temporal interruption of the system operation. This situation may further worsen by the tasks of big data handling and processing that many CPSs also perform, as these can also delay data transmission and data processing.

Fault-tolerant control:

Many CPSs are mission critical or service guarantying systems. For these CPSs, fault-tolerant control is one of the most important requirements. It aims at assuring the desirable performance of the system and its continuous operation, despite the presence of faults [12]. Typically, feedback-based control systems are adopted for this purpose, as they are able to maintain system stability under the presence of disturbance by manipulating system actuators. However, the control actions executed based on output signals (i.e. based on the sensed system variables) mask the effect of failures and, by doing so, they prevent early detection and diagnosis of failures [13].

As a conclusion, we can claim that the chosen research phenomenon and the related research question have not been comprehensively studied and reported upon in the literature, though it plays an important role from the perspective of a near future advancement. Consequently, there is a knowledge gap concerning the potential role of the concept of system operation modes in the context of failure management for self-tuning cyber-physical systems. Obviously, this also offers the opportunity for reconceptualization of the investigation methodology and computational techniques of analyzing and forecasting failures in CPSs having self-regulation and self-tuning capabilities. In order to explore this phenomenon and to derive meaningful insights by a state of the art study of this topic, we will first investigate the existing failure analysis methods relying on the three kernel approaches (knowledge domains) introduced by the above reasoning model. The information obtained in this Chapter was used to underpin the research activities conducted in the rest of our project.

2.2 Fundamentals of failure analytics

2.2.1 Consideration of the specificities of the CPS hardware in failure analytics

Hardware components represent the physical dimension of CPSs. They have an influential role in failure analysis. While traditional systems typically consist of analog hardware components, cyber-physical systems also have many digital hardware components. Since they represent two different manifestations, they imply the need for different attention in failure management. Failure management of analogue physical components is realized through a combination of failure avoidance, fault tolerance and maintenance strategies, which are associated with these components from the beginning of the design stage. As failure avoidance is concerned, hardware manufacturers analyze the possible sources of failures and weaknesses of components with the objective to improve their robustness and reliability. Towards this end, the design characteristics, the used materials, and the applied manufacturing processes are concurrently analyzed in order to reduce the chance of failures during the operation [14]. The fault-free lifetime of the components and the entire system is estimated based on rational inquiries and empirical experimentation. For these, both simulation of real life conditions and failure injection were considered.

While the principle of fault avoidance is to reduce the chance of failures ever since the component design phase, preventive maintenance aims at avoiding failures during system operation. The principles of preventive maintenance facilitate the reduction the failure rate of system components so that failure costs and machine down time could be minimized [15]. For this reason, it implemented revision, exchange of components and repair actions based on either the monitoring of system condition or a fix maintenance schedule [16]. This schedule was determined either by the component's life length declared by the manufacturer, or by experience, due to historic records.

Fault avoidance and system maintenance are complemented with fault tolerance, which is a self-defending capability of systems. The most important fault tolerance method is hardware redundancy [17]. It aimed to guarantee system operation by multiplying hardware components, as well as providing reference for fault detection. Cross checks, consistency checks, and voting mechanisms were conducted for detecting faults and errors in system components [18]. Initially, the principle of hardware redundancy was applied, among others, in the case of low-level components such as sensors and gates. However, it was later on extended to other components such as processor units, which improved the reliability of systems [19]. Nowadays, hardware redundancy is still widely applied in powertrains [20], aircraft industry [21] [22], nuclear plants [23], and similar critical systems and infrastructures.

Notwithstanding the implementation of the above-mentioned failure management strategies, hardware is still one of the main sources of failures. They are subjected to wearing and corrosion, which are caused by external factors. It causes these types of components present progressive degradation, which makes them to operate under their normal capacity. This situation is very critic in highly complex systems, as higher number of components, increase the chance of system failure [8]. Recently, the emergence of embedded digital hardware and software components has worsened this

situation. It caused a tight interrelation between both domains, which led, in turn, to new failure types and failure sources. This interrelation has also made it difficult to determine if the failure is originated on hardware or software [24], hampering the decision making about repair and corrective maintenance actions.

We are facing a shift of paradigm in failure analysis. The prominent role of hardware in failure analysis is fading away. Software has gained importance, along with the cyber-ware. It has caused that researchers attention moves from a purely hardware-based failure analysis to a holistic system perspective that integrates hardware, software and cyber-ware. The intensive sensor instrumentation, along with the availability of advanced processing capabilities and sophisticated algorithms has enabled proactive failure management. It has detracted from traditional hardware oriented activities to software based solutions that has extended the fault tolerance options [24].

2.2.2 Consideration of the specificities of the CPS software in failure analytics

As it was already stated before, software components gain importance as hardware components are augmented with smart capabilities. Software has a dual role, namely it executes the system control and makes logical decisions. The control software processes the sensed data and determines the necessary states of actuators and effectors. However, the operation of the control software not only manifests in the virtual and cyber spaces, but it is also associated with the operation of digital and analogue hardware in the physical space. Should a control software malfunction or crash, this may have a large impact on the operation of the hardware [25]. If it happens in a critical real-life situation, then it may entail other threads and may even endanger human life, environmental resources, or industrial assets [26]. It is extensively discussed in the literature that software failures may occur due to: (i) design errors, (ii) digital hardware problems, (iii) data or signal inadequacy, (iv) malicious attacks, and (v) software aging. Malfunctioning may also be the consequence of: (i) data corruption, (ii) lack of resources, and (iii) software bugs. The aggregation or accumulation of these may lead to decrease in software performance and ultimately to break down [27]. These seemingly commonsensical pieces of knowledge had an important message in the context of our research. These imply that failure management and preventive maintenance of CPSs assume conducting actions not only for fault avoidance, but concurrently for software maintenance too.

Fault avoidance is conducted in the case of software in a similar way than in the case of hardware. In order to be able to avoid them, it requires foreseeing software risks and threats from the design stages [28], [29]. Software prototype investigations, as well as usability tests are conducted in order to continuously improve software operation. For this purpose, automatic detection and report of bugs is performed through bug tracking systems. These systems convey information about the observed fault, as well as the conditions in which it occurred [30] in order to facilitate corrective design actions.

The current practice is that, after the release of a software system for public use, various software maintenance actions are done in order to guarantee a high-level functional performance and usability. Actually, software maintenance has become a standard and systematized practice over the years. Based on the work of Christa et al., software maintenance can be divided into four main categories: (i) corrective maintenance, (ii)

adaptive maintenance, (iii) perfective maintenance, and (iv) preventive maintenance [31]. In the case of software constituents, the purpose of corrective maintenance and preventive maintenance is the same as for hardware components. They imply checking bugs and conducting corrections on the software codes either to prevent a foreseeing failure or to repair the system once it has occurred. Adaptive maintenance aim at conducting software modification in response to changes in the environment, or hardware upgrade and perfective maintenance aims at improving software features continuously to provide proper behavior [31]. During the operational stages of a software system, bug tracking is carried out in order to monitor the performance of the software and to facilitate failure prevention.

Not only software may cause failures in hardware - hardware may also lead to failures in software. For instance, Lyer and Velardi analyzed software errors and found that 35% of them were related to physical components in one way, or another [32]. It shows a strong interdependence between the two manifestation and technological domains. Although it is undeniable that software failure management approaches have gained importance during the last years, we are nowadays facing a turning point in which authors and experts are claiming development of new principles. They expect that these principles should rule the development of integrated approaches that can avoid and manage failures in cyber-physical systems [25].

2.2.3 Consideration of the specificities of the CPS cyberware in failure analytics

The physical and computational interaction between hardware and software is enabled by information flows. Control and application software constituents of CPSs process both preprogrammed information and run-time acquired information concerning the physical world. From an architectural perspective, the sensors and actuators of the systems create a closed information loop by processing the signals coming from the monitoring units to execute control actions. In this context, hardware and software strongly depend on the availability as well as on the quality of information. Typically, an intensive monitoring of the state and condition of the sensors and other information processing units is performed in order to avoid information insufficiency caused problems of system operation, as well as the incurred costs. As evidenced by the literature, system data and signals are intensively used of failure information carriers, by comparing them with an existing models, knowledge or data features [33]. Nowadays, however, this is challenged by the overwhelming amount of data that should be transmitted and processed, in order to assure a proper system performance.

Big data is characterized by large data samples, high dimensionality, and heterogeneity. Big data technologies provide new opportunities for failure management, but there are no low hanging fruits. The amount of information is increasing faster than the improvement of information processing methods [34]. Transmission, storage and processing of massive amount of data present problems, which can affect the performance and resilience of CPSs. If this happen with an adverse tendency, then it can in turn affect the quality of the obtained data, as well as the timely execution of real-time processes. Data sensing is subjected to noise and disturbance caused by the environment [35]. Having multiple active data sources lead to noise accumulation, which makes the use of data filters necessary. However, the filters should be carefully

selected and set in order to avoid discarding relevant information [36]. It also causes delay in the data processing, which along with high dimensionality of data leads to high computational costs and algorithmic instability [35]. Data transmission also plays a critical role at transferring huge amount of data. It may cause delays in timed operation of CPSs, and loss of information that will affect the execution of real time failure detection.

Information explicitly conveyed by data and signals is by far the most important element for failure analysis. On the other hand, this is also the most sensitive enabler. Problems in data collection, representation, transmission, storage and processing may not just affect the performance of failure detection and failure diagnosis, but may even contribute to another cause of system failure. The lack of data samples during processing, or having low-quality sensed data, can lead to missing emerging failures or conversely to false failure alarms and can even hinder detection of real failures.

One of our take away can be formulated as follows: Traditionally, hardware, software and information have been analyzed in isolation in the context of failure management. However, this does not give opportunity for consideration of the increased system complexity and the tight relations between the cyber and the physical system components. These limitations require us to analyze hardware, software and information concurrently. Old approaches focusing only on hardware or software cannot identify failures that are caused by the interrelation of the components from the two domains. In this context, the information conveyed by system signals requires attention as failure information carrier. There is however a difficulty in this respect, which is caused by the huge amount of data that should be sensed, transmitted and processed properly. In the end, signal-based failure analytics should assure continuous system operation in a cost-effective way.

2.3 Fundamentals of signal-based failure analytics

2.3.1 A concise overview of the types of signals

As it was already stated, system signals in general convey information that can be used for analyzing operation states and conditions of systems. Signals are observations in the physical world over the time. They are collected through sensors or any other measuring instruments [37]. Usually, signal analysis techniques extract meaningful information based on analysis of signal features and transform it in a recognizable form [38]. Features can be both global and local characteristics of signals. One of the major challenges for computational signal analysis is that there are very different types of signal features, as well as signals. Depending on their statistical properties, signals can be sorted into stationary and non-stationary categories [39]. Stationary signals are those whose properties, such as amplitude, frequency and phase, do not change in time [40]. Evidently, the opposite applies to non-stationary signal. The behavior of non-stationary signals, whose properties present variation throughout time, cannot be analyzed without considering the change of the time parameter.

In view of the way of variation, signals can be either deterministic or random. Both transient signals and continuous signals belong to this category. Transient signals are characterized by short duration. They are also known as finite-energy signals. On the contrary, constant signals are those that present ‘infinite energy’ [41]. Their distinguishing characteristic is that they do not change over time, and thus, their derivative is zero. Deterministic signals are those whose frequency, amplitude and initial phase do not show ‘predictable regular’ alteration. It makes them highly predictable for the reason that their composing sinusoids suffice for determining any value of the signal [39]. On the contrary, random signals lack predictability as they do not present a clear pattern of alteration [42]. White noise is a good example of a random signal.

Signal-based analysis represents signals as combinations of simpler signals [37]. In this context, the notion of periodicity is paramount. A signal is considered to be periodic when it repeats itself over a finite time period [37]. This characteristic allows modeling signals as a composition of sinusoids that have a time-varying nature [43]. Sinusoids are described by:

$$g(t) := A \cdot \sin(2\pi(\omega t - \varphi)) \quad (2.1)$$

where: A is the amplitude, ω is the frequency, and φ is the phase shift. Although signals can be modeled as sums of sinusoids, sensed signals usually present noise and useless components that should be filtered [39].

2.3.2 Supporting decision making based on signal analysis

By digital signal processing, we can extract features that convey information about failures. In order to profit from this opportunity, signal-based failure analysis applies various digital processing techniques to recognize and identify features that can be taken into consideration as symptoms of failures. Such features can be extracted in: (i) the time domain, (ii) the frequency domain, or (iii) the time-frequency domain. Time domain-based investigation regards the measured signals (variables) as a function of time. It focuses on geometric factors such as amplitude and peaks, or on statistical parameters. Typical time domain-based methods are cross-correlation analysis and statistical features-based analysis [39]. The former can be used to evaluate the similarity between two signals. For instance, it can be employed to compare a failure-free reference signal with an observed real-life operation signal to detect failures. The latter makes use of the statistical properties of signals for the purpose of failure detection.

Certain sort of distortions or smaller disturbances of signals are difficult to detect in the time domain. Due to this limitation, frequency domain analysis (FDA) is applied as an alternative feature extraction method. FDA transforms the captured signal from the time domain to the frequency domain. The most popular technique is Fourier analysis, which reconstructs any arbitrary function as a sum of sinusoids in a finite interval [44]. This technique is also useful for providing a compact representation of signals that can be used for various purposes, for instance, for signal classification. While representation of signals in time domain involves the evolution of the signal amplitude over time, representation in frequency-domain shows how quickly such changes take place [45]. The general Fourier transform is described by the following expression:

$$F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \quad (2.2)$$

where: $F(\omega)$ is the Fourier transform of $f(t)$, and ω is the signal frequency. In this section, we cannot go deeper into the mathematics and applications of Fourier transformation. Should the reader be interested in collecting more knowledge on these subjects, we advise to check publication [46].

The need for time-frequency domain analysis (TFDA) typically arise due to the lack of time information in the frequency domain analysis [47]. Information about time may be required in some cases as the actual signal frequency composition may change with time. Therefore, TFDA analysis simultaneously captures both frequency and time information carried by the processed signal. Widely used methods for performing TFDA analysis are: (i) short time Fourier (STF) analysis, (ii) Wigner-Ville distribution (WVD) analysis, and (iii) wavelets analysis. As a fact of the matter, this last method is the most popular one. Like other signal-based analysis approaches, it represents signals as a composition of a set of basic functions. By appropriate selection of such functions minimization of the complexity of the representation can be achieved [37]. The basic functions are wavelets, which are derived from a mother wavelet that has the form:

$$\psi^*(t, a, \tau) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-\tau}{a}\right) \quad (2.3)$$

where: a is dilatation factor, and τ is the signal translation. Based on it, the continuous-time wavelet transform can be described as:

$$CWT(a, \tau) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} y(t) \psi\left(\frac{t-\tau}{a}\right) dt \int_{-\infty}^{\infty} y(t) \psi\left(\frac{t-\tau}{a}\right) d\tau \quad (2.4)$$

There are many technical details of the wavelets analysis and using wavelets transform methods. For more information about in the context of failure analysis we suggest to check publication [48].

The implementation of signal-based analysis relies on the assumption that if failures are emerging, then they typically imply some sort of disturbances and anomalies. Signal-based analysis is underpinned by the development of articulated signal models, which enable the observation of deviations from a reference signal. These deviations can be observed in the signal attributes, which, as discussed above, can be extracted in the time domain, the frequency domain, or the time-frequency domain. There have been many signal-based approaches proposed for failure detection in the literature. For instance, Guo, H. et al. used wavelets in the analysis of information coming from vehicles to determine whether there are failures or not [49]. Murphey, Y.L. et al. analyzed recorded behavior signals of a testbed system to determine whether the signal presents normal or fault behavior. They applied expert knowledge through fuzzy logic to detect and identify failures [50].

Daigle, M.J. et al. performed event-oriented signal analysis to find failures based on the deviations of a predicted signal (generated by a model) from an observed one [51]. Abul Masrur, M. et al. used voltage and current signals as input for a machine learning algorithm that was able to detect and isolate faults in electric driver inverters [52]. Barakat, M. et al., as well as Xian G.-M. and Zeng B.-Q., used wavelets for extraction of features of signals and used the features for failure identification [53] [54]. Yen, G.G. and Meesad, P. used neuro-fuzzy signal analysis for health monitoring of machinery

[55]. These publications indicate that signal-based prediction is a very useful technique for condition-based maintenance of complex systems. The reason is that signals typically have features that may serve as indicators of the emergence and manifestation of failures. However, these methods rely on prior knowledge concerning how the signal characteristics are influenced by the manifestation of specific failures.

The so-called ‘signal models’ provide useful representations of the behavior of signals. They enable extracting signal features that convey deviation information for computational failure detection and diagnosis. However, the analysis of signals, no matter if it happens in the time-domain, frequency-domain or time-frequency domain, can be affected by SOM transitions made by the system itself. The SOM transitions usually do not show a fixed sequence (i.e. transitions do not regularly occur in the same order). The reason of this is that the changes of system operation modes are reactions of the system to variations appearing in the working conditions. Therefore, they can alter the properties of the concerned signals. On the other hand, since the effect of SOM transitions can be interpreted as a symptom of some failure, it may lead to incorrect failure diagnosis. This situation is in particular critical when investigation of the signals is made in the frequency domain. Normally, frequency domain-based methods cannot consider time information associated with the occurrence of SOM transitions. Consequently, this approach does not make possible to determine the particular time instant in which the deviations are observed, and thus, to determine if such deviations are indeed the result of a SOM transition, or not.

SOM transitions also imply transient behaviors. These behaviors may be observed during the time lapse it takes to the control mechanism to stabilize the system after disturbances or changes in the settings. If there are frequent SOM transitions, then there will also be a chance for more transient behaviors. In the end, it will lead to variations in the signal parameters, and thus, as mentioned above, to false failure alarms. This situation can be particularly apparent when the frequency of the SOM transitions of a system is unpredictable. In other words, certain systems may behave rather stable (their controllers initiate only few SOM transitions), but some others can be very dynamic (their controllers initiate very frequent SOM transitions). These issues will be addressed in detail later on in the thesis.

2.4 Model-based failure analytics

2.4.1 Fundamentals of model-based failure analytics

The most common approaches of failure detection and diagnosis are based on the use of system models. Modeling replaces the physical and virtual elements of the system with qualitative or quantitative digital models that can describe the complete operational processes of a system [3]. A comprehensive modeling requires both data and knowledge about (i) the physical processes, (ii) the computational processes, and (iii) the system dynamics. Model-based failure analysis compares the measured outputs of the observed process with the outputs of the model. In many cases, a residual index is used to compare the observed system’s output with the output predicted by the used analytical model [56]. The comparison allows identifying discrepancies. As a next step, the observed discrepancies are interpreted as fault indicators.

Model-based fault detection and diagnosis has reached a high level of maturity. It is widely applied, among others, in vehicle control systems, transport systems, robots, and manufacturing processes [56]. Taking into consideration the kind of the used models, Venkatasubramanian et al. proposed to differentiate qualitative and quantitative model-based approaches [3]. Below we briefly analyze the most popular quantitative approaches: (i) parity relations, (ii) observers, and (iii) extended Kalman filter (EKF). We also provide a concise survey of the various qualitative approaches such as: (a) signed digraphs, (b) fault trees, (c) qualitative physics, and (d) expert systems.

2.4.2 Quantitative model-based failure analytics

The major quantitative model-based failure analytics approaches are based on: (i) investigation of parity relations, or (ii) using state observer(s).

Parity relations-based approach:

The aim of the investigation of parity relations is to generate residuals by comparing the outputs of an analytical model with the outputs of a real life process [1]. The computing of residuals can be done based on directional or structural properties, which allows the recognition and differentiation of different faults and noises [57]. In principle, all of the various parity relation-based methods can discriminate different types of faults. However, the efficiency of the residual estimation process may differ depending on the type of representation used for modeling. Most of the models reported in the literature are described by transfer functions and state space models. They assume linearity.

The transfer functions associate the output signals to the input signals. They are described by various mathematical means. It makes possible to observe an output y_1 during the computation process, whenever an input signal u_1 is fed into the system [58]. However, it implies that the entire input-output history is required for the description of the system operation. The sequence of inputs is needed to compute future output values [59]. The mathematical representation of a transfer function is in the form:

$$G_p(s) = \frac{y(s)}{u(s)} \quad (2.5)$$

where: $y(s)$ are the outputs, and $u(s)$ are the inputs of the process.

State-space models use the concept of state for process representation. The state is characterized a range of values conveyed by one or multiple system signals in a time instant t , to represent the temporal mode of a system. The description of the state not only integrates all past information, but also the initial conditions for the outputs and their derivatives [59], so that only $u(t)$ and the state $x(t)$ are required to compute $y(t)$, for $t \geq t_o$. The mathematical description of a process through state space representation is:

$$\dot{x} = \mathbf{A}x(t) + \mathbf{B}u(t) \quad (2.6)$$

$$y = \mathbf{C}x(t) + \mathbf{D}u(t) \quad (2.7)$$

where: x , y and u are the state, output and input vectors, and \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} are matrices composed by constant elements. Equation 2.6 is the state equation, while Equation 2.7 is the output equation. In both equations, faults are represented in either an additive

manner or a multiplicative manner as factors that alter the balance of the system. That is:

$$\begin{aligned}\dot{x} &= \mathbf{A}x(t) + \mathbf{B}u(t) + \mathbf{F}p(t) & (2.8) \\ y &= \mathbf{C}x(t) + \mathbf{D}u(t) + \mathbf{F}'p(t) + q(t) & (2.9)\end{aligned}$$

where: $p(t)$ are the actuator faults, and $q(t)$ are the sensor faults in the case of additive faults in state-space representation. In a similar manner, faults in the input-output system are described as:

$$\mathbf{H}(s)y(t) = \mathbf{G}(s)u(t) + \mathbf{H}(s)q(t) + \mathbf{E}(s)p(t) \quad (2.10)$$

where: \mathbf{H} and \mathbf{G} are polynomial matrices, and $p(t)$ and $q(t)$ represent the actuator faults and the sensor faults, respectively. Due to space limitation, we cannot go deeper into the calculation of the residual estimation for the state-space models and transfer function ones. Should the reader need more information about this, we advise to consult publications [60] and [3].

It follows from the above description that the analytically computed outputs serve as reference for determining if the observed behavior corresponds to a failure-free operation. The residual vector is estimated by comparing the set of outputs corresponding to the observed behavior with the ones coming from the analytical model. Distinguishing between $p(t)$ and $q(t)$ allows us to determine the fault location, while the direction of the residual vector serves as a signature for different types of failure modes. In the case of state-space models, system signals are used to estimate those system states, which can be considered as features conveying the discriminant characteristics for fault diagnosis.

The parity equation-based method is widely used in those cases where it is possible and is feasible to describe the processes and operation of the system by mathematical equations. This method also requires that the parameters defining the system model are observable, i.e. it is possible to measure them directly in the system through involvement of sensors. Some special adaptations of state-space models have been used for fault detection and isolation in cases in which observability is not guaranteed. This is supported by the state observer approach. However, in the case of CPSs, the mathematical description of the whole system is either rather complicated or simply impractical. CPSs are typically composed of a very large number of components. Some of them may be added or removed during run-time, in particular, as we move ahead toward the second generation of CPSs. This change may lead to the appearance of new SOMs (i.e. to emergent system behaviors). The result of this is that parity equations fail to describe variable system behavior sufficiently.

State observer-based approach:

The concept of state-observer arose from the need to estimate some state variables that cannot be accessed easily. State observers make use of measured input and output signals in order to reconstruct the unmeasurable state variables [61]. When applied in faults detection and diagnosis, the set of state observers should be sensitive to some specific subset of faults, while they should be insensitive to faults belonging to a different subset [3]. These models can differentiate between various types of failures, and can mitigate the disturbance effects. The state observer is grounded in state-space models, based on which the input failures, the output failures, and the external

disturbances can be estimated. In this approach, an analytical model running parallel with the real system is used and inferred, and the system states are estimated in real-time. Fault detection and diagnosis is performed by comparing the analytically computed values of the state variables with the observed ones through a residual index [61].

In the state observer models, input and output signals of the system are the failure information carriers, i.e. these are monitored in the system. System signals are also used to estimate unobservable system parameters. Based on them, the observed system state is computationally reconstructed. In like manner, an analytical system state is also computed based on the signals coming from the analytical model. The system states (derived from the observed system and from the analytical model) are used as the data features. The analytically computed state is used as reference for determining if the system is under the effect of one or more failures. The residual between the observed and the estimated states serves as the basis of the decisions concerning the occurrence of fault.

The classical state observer methods were proposed by Luenberger [62]. These methods are applied in many contexts, for instance in electrical railways [63], aircrafts [64], and batteries [65]. However, the most widely applied state observer method is the Kalman filter. While the classic state observer approaches are deterministic, the Kalman filter is stochastic. It applies probability density functions for estimating process and measurement noises with the aim to determine the effect of random variations in the system. Initially, Kalman filters were applicable only to linear systems. However, the Kalman filter can be extended to non-linear systems by computing the local linear approximations of state equations at each sample time [66]. This enabler is called the extended Kalman filter (EKF).

Several applications of EKF to fault detection are reported in the literature. Chen and Liu estimated sensor faults and actuator faults related to an altitude control system of a satellite [67]. Yan et al. implemented EKF with support vector machines (SVM) for online fault detection in chillers [68]. In this application case, EKF was used to reduce the signal noise and to separate the faulty and normal data samples for the SVM classifier. Singelton et al. used EKF for estimating the remaining useful life of bearings by estimating the trend of degradation [69]. Hoseini et al. proposed an online method for detecting inter-turn short-circuit fault in switched reluctance motors through the implementation of EKF [70, p.]. Online use of EKF makes it possible to detect failures faster, since these computations are done mostly in real-time.

Even though multiple applications of EKF are reported upon, it is still difficult to estimate the process noise covariance in nonlinear systems as it assumes a prior knowledge about noise statistic. Therefore, it is not an optimal estimator [71]. To overcome this situation, adaptive Kalman filters have been developed [72]. However, the use of the Kalman filters and the various state observer methods is still a complex challenge due to the required sophisticated modeling [73]. The main drawback of the analytical model-based approach is that the computation of the residual index strongly depends on the reliability of the analytical model of the investigated complex system. To reduce complexity, simplifications are used. Simplifications of the analytical models may cause a mismatch between the output of the model and the observed system behavior [74]. This mismatch may be increased by the effect of noise and disturbances that cannot be modeled, causing the residual to become nonzero during the failure-free

operation of the system [75]. This problem limits the applicability of model-based failure indicators to well-defined operation scenarios that are not subjected to uncertainties, unlike non-linear systems [76] [77]. Furthermore, the residual indicator is limited to specific system functions that are sufficiently known and predictable.

The state observers approach introduced the concept of states in failure diagnosis and forecasting. Nevertheless, the concept of states may go beyond the information conveyed by single signals, if multiple signals are taken into account. This approach also allows considering multiple information sources for determining system states. Putting everything together, it provides a better understanding of system dynamics. However, both the parity equations and the state observers strongly depend on the information carried by the sensed system signals.

2.4.3 Qualitative model-based failure analytics

One of the distinguishing characteristic of qualitative models is that they are not dependent on obtainable system signals. They mostly depend on the understanding of the physics and chemistry of the represented processes [78]. These methods are useful to explore cause-effect relationships, as well as to determine failure effects based on system architecture. Another characteristic of these methods is that they are derived based on a prior expert knowledge about system dynamics. There are many methods included in the family of qualitative model-based failure analytics methods. In this Sub-section, we will explore: (i) signed digraphs, (ii) expert systems, (iii) fault trees, and (iv) qualitative physics as the basis of reasoning.

Signed digraphs:

Signed digraphs (SDGs) capture cause-effect information about system processes. This set of information enables qualitative simulation. They facilitate the prediction of system's response to the occurring failures by identifying variations with respect to regular system behavior [79]. Traditional digraphs describe processes based on a graphical representation, composed of nodes and edges (or arcs). Nodes are process variables or events, while edges describe the relationship between the nodes [78]. Signed digraphs are directed graphs with the capability to represent the variable influences of processes on each other. For this reason, the relationships between variables (nodes) are discretized and characterized (labelled) by positive or negative signs. To describe the types of influence, signs “+” and “-” are used. The sign “+” indicates that two connected nodes present the same trends of change, e.g. if one variable increases, the connected variable increases too. The sign “-” indicates that the two nodes present changes in the opposite directions, e.g. if one variable increases, the connected variable decreases [71].

In the context of fault detection, SDG nodes (i.e. system variables) are compared with a set of reference values. If no difference is found between the observed values and the reference values, the node is labelled with “0”. This indicates a normal behavior. However, if there are discrepancies, the concerned node can be labelled either with “+”, if the behavior is above normal, or with “-”, if the behavior is below normal [81]. The signed arcs depict the proportionality of the effect of one variable on the other one (i.e. how much one variable increases or decreases in relation to the increase or decrease of

the variable connected to it). This enables a qualitative analysis of the system dynamics and fault propagation.

Signed digraphs have been used in particular for evaluating the root cause and propagation of faults. However, SDGs suffer from multiple limitations when used alone. In a real process, variables present noise and variations of work states [82]. Recently, the specification of SDGs has been extended by involving fuzzy logic theory and probability statistics in order to tackle these limitations. Lü and Wang implemented conditional probabilities to discriminate different fault types, considering that multiple fault causes may present the same qualitative symptoms [83]. Peng et al. implemented a probabilistic SDG for reducing the amount of false alarms in industrial plants [84]. Tarifa and Scenna presented a fault diagnostic system to supervise a desalination plant, which combines SDG and fuzzy theory [85]. In this particular case, SDG was used to forecast the possible evolution of the plant and fuzzy logic was used to determine the type of the occurring fault based on the results of forecasting.

The failure information carriers of SDG are the system parameters and the experts' knowledge about their behavior under regular and faulty operation. SDG does not require transforming the data into features. A knowledge base, which includes information about the regular values of system variables, is used as reference for diagnosis. In the various implementations of SDGs, probabilistic distribution or fuzzy logic-based rules were also used as decision enablers for determining failure occurrence [86], [87], [88], [89]. Once an abnormal behavior of a system parameter is observed, emergence and propagation of a failure can be predicted. The SDG approach can be implemented relatively simply. This approach is able to: (i) provide insight in gradual failure forming, (ii) monitor the variation in system dynamics caused by failures, and (iii) support decision-making. However, it is constrained by the need for expert knowledge and its complexity that increases as the system complexity increases.

The concept of system states is also utilized by this method. However, the status of the system is not estimated based on the status of its components. The estimation is made based on the relationship between the system variables, which are represented by the nodes of the digraph. For instance, the state of a reservoir is not estimated based on the current state of the outflow valve (whether it is open or closed), but on the outflow rate of the water contained in the vessel. It is also the case for parity equations and state observers when represented through state-space models.

Expert systems:

Expert systems aim to mimic the reasoning of human experts when dealing with a knowledge intensive problem [90]. This type of methods describes system operation based on the expert knowledge, which is usually represented by if-then rules. The concept of expert systems supports decision-making based on qualitative knowledge. Expert systems are mostly used when conducting real time data collection and data processing is difficult [91]. Applications of expert systems to fault analysis involve the diagnosis of the occurring failure mode, and the analysis of the root-cause of failure.

Žarković and Stojković implemented fuzzy expert systems for fault detection and classification in power transformers [92]. Tang and Wang, [93], and Xu et al., [94], implemented expert systems using artificial neural networks in order to conduct fault diagnosis. Nabende and Wanyama implemented expert systems with Bayesian networks for diagnosing heavy-duty diesel engine faults [95]. One of the main problems of the

offered reasoning method is that it does not have any understanding of the physics behind system operation [78]. Moreover, it cannot be updated automatically and is limited by the quality of the captured human knowledge, the formalization and representation of this knowledge, and the reasoning capabilities offered by the inference engine. Due to this, expert systems are usually realized by using hybrid reasoning. Shao et al. developed an expert-based fault diagnosis system by combining failure mode effect analysis and fault tree for flight control software [96].

Fault tree analysis:

Fault tree analysis (FTA) is often seen as a particular variation of expert systems. Its objective is to implement deductive logic to determine the underlying causes of failure [97]. For this purpose, it generates a diagram that enables the graphical representation of multiple scenarios of events that can lead to failure occurrence. FTA decomposes the failure potential of the system into failures of its subsystems, and components down to basic events [98]. FTA diagram starts with a top event, which is interpreted as a system failure. Then, it is read backwards from the top event down to the leaf events, which represent root causes of failures [99]. There can be several events that together lead to failure occurrence. In certain cases, concurrent appearance of certain events results in failure with high probability. At the same time, we can foresee multiple sequences of events that can in isolation trigger the same failure mode. Logical gates, such as “and” and “or” are integrated into the FTA in order to explain the different ways in which the top event is triggered [100].

FTA can be implemented in both a qualitative and a quantitative form. Qualitative analysis reduces the faults tree to minimum cut sets [99]. A cut set is a set of primal events that, in case of simultaneous occurrence, trigger the top event. Finding the minimum cut sets implies identifying the minimum number of events that triggers the top event - in order to determine its causes. Quantitative analysis estimates the likelihood of occurrence of the top event based on the rate of occurrence and fault duration of all basic events [101]. Although this method enables understanding failures and does not require major resources for its operationalization, it is limited to static systems. It implies that, in its classical formulation, it cannot be implemented in systems that present multiple states [99].

Various extended versions of FTA have been developed in order to overcome such limitations. The dynamic failure tree is one of them [102]. It includes additional gates to the model in order to consider the dynamic behavior of the system. These gates aim to capture dynamic system behavior, among others, such as spares, sequence dependent events, dynamic redundancy, and priorities [103]. There are multiple applications of fault trees discussed in the literature, either dynamic or static. Wang et al. implemented FTA for evaluating the instantaneous risk of the actual configuration of a nuclear plant [104]. Liu et al. combined FTA with quantitative analysis to investigate high-speed railway accidents [105]. Dongiovanni and Lesmantas realized a hybrid implementation of FTA and Bayesian networks for analysis of failure mechanisms of power plants [106]. Kabir et al. combined dynamic FTA with fuzzy reasoning theory to facilitate the analysis of failures in complex systems [107].

Due to its easy implementation and interpretation, FTA is one of the most common methods used in practice. However, it has some important drawbacks as well. State space of complex systems can have an exponential size hampering the development of

the dynamic FTA [98]. Moreover, this method depends on the expert knowledge. It requires deep understanding of failure manifestations, cause and effect, as well as awareness about the multiple failure modes that can affect the system. It prevents the detection of unknown failure types. Although multiple iterations can be conducted for updating the model as new failures are discovered, these cannot be conducted automatically. A proper update of the existing FTA needs the expert, who should be aware of all aspects of operation and failures.

There are multiple possible failure information carriers for expert systems and FTA. They range from observable wearing of components up to parameter deviations in system signals. The experts determine the relationship between the observed symptoms and the top event. The experts' knowledge is modelled through a FTA. Although this method is mostly used as input for re-design, it can also be used for failure detection. The derived FTA is used as a reference for either finding the cause of the occurred failure, or predicting the forming failure, based on the observed events. The decision enabler is the estimated probability of occurrence of the analyzed failure mode, or the minimum cut set.

Due to their versatility, expert-based systems can also be implemented in the context of CPSs. However, considering (i) the high complexity of these systems (due to the high number of components and their interrelations), (ii) the emergent behavior they can present, and (iii) their capability of self-evolution and self-reproduction, the applicability of expert system-based approach of failure analytics would be rather limited. The high number of components would imply the need for a deep understanding from the experts - not just about the expected components behavior, but also about the effect of their interrelations. Moreover, since there can be components that are added or removed during run time, the modelled knowledge of components interrelationships would be obsolete or would not apply any longer. The self-evolution and self-reproduction capabilities would require updating the knowledge-based models very often. This is unpractical if we consider the effort it would imply and the high frequency of changes that evolutionary processes produce. Knowledge-based models can address permanent system functions. They could work for describing the system behavior corresponding to the most common system operation modes. They could also describe some of their transitions. However, in the case of dynamically changed or emergent SOMs, or the most infrequent ones, could not be foreseen and modeled through expert systems. A potential implementation of knowledge-based models would require determining one or multiple different models per SOM.

Qualitative physics:

High fidelity quantitative models of complex systems and processes are typically defined by equations that describe the physical system, control model, and input/output parameters. These models are typically computationally expensive [108]. To overcome these limitations, qualitative physics was introduced, which describe physical processes based on qualitative reasoning, as humans do. This method describes physical processes as if they were machines, i.e. composed by multiple and constitutive components that in conjunction defines the entire behavior of a complex system [109]. Every component is described by qualitative states and a set of confluences enabling the estimation of all the possible process/system behaviors [110].

Confluences are qualitative differential equations, which are satisfied by particular assignments to parameters. Unlike quantitative modeling that assigns real numbers to parameters, qualitative modeling applies ordinal, categorical, and fuzzy parameter representations [109]. Ordinal representation divides the range of values that can be assigned to a particular parameter into regions. Categorical representations implement symbolic values, such as {low, medium, high}, for the parameter assignment. However, they do not imply a fix order. Fuzzy representation assigns fuzzy sets to the system parameters. The relationship between the qualitative variables is determined by the differential equations that describe the analyzed process.

In qualitative physics, component and system states are described as a set of qualitative values taken by every parameter of the system in a time t . These states are called qualitative states (Qstate). Usually, a device presents multiple operating regions. Every operating region is described by a particular set of confluences. Qstate represents system behavior in an operating region. These Qstates are triggered by preconditions [111], which are a set of events that determine when the operating region is active. Every single component is composed by a set of Qstates, which describe the qualitative status of each component. The joint Qstates of the components active at a time t determine the Qstate of the whole process at the same time instant.

QSIM is one of the most widely implemented methods in qualitative physics-based failure analysis. It determines the constraints of the physical processes, along with the initial states, for predicting their future states [110]. Lu et al. used QSIM for diagnosing faults in centrifugal compressors [112]. Platzner et al. proposed a computer architecture for embedded computer platforms [113]. Zhang and Ren developed a hybrid model that joins QSIM with parity space methods in order to enable fault diagnosis in a satellite control system [114]. Applying QSIM, Junior and Martin developed a simulation of a fluid flow system, which allows the analysis of failure effects on this type of systems. Although QSIM has been widely applied, it presents the same limitations as the rest of qualitative methods. The lack of numerical values leads to inaccurate results. The use of categorical values instead of real numbers in the system equations may cause loss of valuable information, which can lead to false negative results.

In general, qualitative physics is applied to failure diagnosis in order to: (i) analyze fault effects by injecting or simulating faults based on a qualitative model of the system and determining its effect on system dynamics, and (ii) to detect failures in an operating system by determining deviations in terms of the system dynamics of the qualitative model. The system dynamics is described by the sequence of states of the process. Failure information carriers of qualitative physics are system parameters, which can be measured through sensors. These measurements are used to determine the operation range and to transform the quantitative measurements to qualitative representations (categorical, symbolic or fuzzy, among others). They constitute the data features. The sequence of states is evaluated and compared with the one corresponding to the reference model. Reference models are usually defined to represent failure-free operation and known failure modes. Decision about the occurring failure is conducted based on the observed sequence of states and its deviation with respect to the reference models. Moreover, constraints violation, due to failure, can also be identified by comparing the sequence of states of the model and the observed process.

Although qualitative physics methods provide means for analyzing system dynamics, and states sequences, their application in failure manifestation has quite some limitations. Qualitative parameters used in the differential equations can lead to spurious and inaccurate results [109] and to both false positive and false negative results about failure detection. Nevertheless, the concept of qualitative states in combination with quantitative measurements contributes to an easy understanding of system behavior and system constraints. It is easier to determine the effect over the system of increasing a particular parameter, than determining the exact value it will take.

The concept of Qstate is very similar to SOMs. Both of them describe a set of potential system behaviors that are determined by the combined effect of the system's components. Qstate, as well as SOMs, imply a categorization of the component states. However, while the operationalization of Qstates is based on analytical models that describe system behavior, SOMs do not rely on a mathematical description of the system. They depend on the control settings of the system and the sensed parameters. It represents an important limitation for Qstates, as the analysis of the system is limited to the scenarios that can be described by the analytical model. As for the rest of model-based approaches, it cannot manage emergent system behaviors, and a full description of the system would require multiple complex differential equations.

2.5 Data-driven failure analytics

2.5.1 Fundamentals of data-driven failure analytics

As it was already argued above, model-based fault detection is limited in terms of the capability of representing process or system operation in an analytical way. Modelling of non-linear and time varying systems require simplifications and expensive computation processes. Moreover, the effect of environmental and external disturbances on the system cannot be predicted with these analytical models. To cope with these challenges, data driven methods have been introduced.

In general, data-driven methods use data generated by the concerned system to identify outliers that may indicate a failure [115]. Data-driven approaches rely on historical data for determining the occurrence and type of failures. Unlike model-based approaches, they do not require a prior model that determines the relationship between the input and the output variables. They derive a model based on historical data by identifying patterns in the input data. Data-driven techniques operate with large amount of data [4]. It makes them suitable for failure detection and diagnosis in large and complex systems [116], as it allows learning the characteristic pattern of every failure mode.

Venkatasubramanian et alia discussed the application of: (i) artificial neural networks (ANN), (ii) principal component analysis (PCA), and (iii) statistical pattern classifiers to data driven failure diagnosis and forecasting [117]. We will analyze these three methods to learn about (i) the failure information carriers, (ii) the used data features, (iii) applied references, and (iv) the decision enablers. Our interest is in how they are implemented in the above data driven failure analysis methods. We will also analyze the how the concept of component and system states are used in the above-mentioned methods.

2.5.2 Artificial neural network-based failure analytics

Artificial neural network (ANN) is an artificial intelligence method that aims to imitate the structure and behavior of the human brain [118]. This method enables conducting classification tasks, approximation of functions, prediction, and grouping based on input data. ANN are weighted directed graphs, where its nodes compute the combination of weighted signals in order to estimate its output [119]. They have learning capabilities that can be either supervised or unsupervised. Supervised learning implies delivering a training dataset to the network with input data whose output is known a priori. Whenever there is a difference between the observed output and the desired one during training, the weights are re-estimated to reduce the difference [120]. On the contrary, unsupervised learning evaluates data without requiring the desired outputs. Towards this end, the ANN evolves to capture the data density characteristics, i.e. to identify patterns based on the observed data [121].

Artificial neural networks can be used for classification of faults based on the evaluation of residual signals [122] [75] [123] [124]. In this case, different datasets corresponding to the failure-free operation and different failures modes can be delivered as input, along with their corresponding classes. In the context of failure analysis, classes are the different failure modes to which datasets belongs [125]. They constitute the output of the network too. During the training stage, the ANN iteratively estimates the signal weights and compares the obtained outputs with the real ones. Weights are systematically changed whenever the observed output does not coincide with the real one until ANN can discriminate the failure modes in the delivered datasets [126]. This approach provides high flexibility because no previous knowledge about the relationship between the inputs and outputs is required due to the lack of analytical models that describes system behavior [74]. ANN can be considered as a “black box”, for which only the inputs and the outputs are known in advance.

To emphasize the significance of neural analogy-based approaches, we exemplify [127] [128]. To emphasize the significance of neural analogy-based approaches, we exemplify some ANN applications in the context of failure analysis. Kumar and Sing implement ANN for fault diagnosis in digital circuits [127]. Chine et al. used ANN for identifying faulty operative conditions in photovoltaic systems [128]. Jahromi et al. implemented fuzzy neural networks in order to provide a condition monitoring technique that is tolerant to drifts in process dynamics [129]. Tayarani-Bathaie et al. proposed a fault detection and isolation scheme to diagnose component faults that may occur in a gas turbine engine [130]. Wu and Kuo implemented discrete wavelet and ANN for fault diagnosis of automotive generators [131]. In this last case, discrete wavelets were used for feature extraction, while ANN was used for fault classification. Concerning the state concept, state-space neural models are not very common [132].

Failure information carriers, in ANN, are system signals. System signals require extracting features in order to implement ANN. These features include wide range of parameters, such classical statistic, or frequency-domain parameters. Feature selection depends on the manifestation of the failure mode. It is one of the most critical steps in the development of ANN, as there is no single feature that can be used indistinctively for all failure modes. Its selection requires prior knowledge of experts and extensive data mining or data analytics processes. Automatic extraction of features is still a challenge [133]. In the case of supervised learning, references are provided as input for

the development of the ANN model. These are known as classes, and they provide a sufficient fidelity representation of the failure modes that are described by the data values. In unsupervised approaches, correlation and data density is used for determining clusters. The decision enabler for both approaches is the ANN model derived during the training process (composed by signals and weights), which determines the class of the delivered input data.

Although ANN is one of the most popular methods for failure analysis, and it is leading its current state of the art, it presents some drawbacks. ANN is typically implemented as a black box, within which the relationships between the symptoms and failure modes remain hidden for the user. As a result, an ANN-based approach is not able to provide insight into failure manifestations and failure forming processes. Moreover, application of ANN has two important drawback in the context of failure detection and diagnosis: (i) its effectiveness decreases when it is applied to diagnosis with multiple faults, and (ii) it easily traps into local minimum [134], preventing the residual to reach its real minimum.

2.5.3 Statistical pattern based failure analytics

Statistical patterns classifiers (SPC) are widely used with fault diagnosis purposes. It aims to determine the future system state based on a probabilistic setting that enables dealing with random variations of the data taking into account noise and disturbances [117]. SPC allows classifying data based on their data density. Classification methods relate reference symptoms of every known failure mode with the observed ones in order to conduct failure diagnosis [135, p.]. There are multiple statistics-based approaches (or classification-based methods) such as Bayes-classifiers, support vector machine, linear discriminant analysis (LDA), k-means, which enjoy a wide acceptance and range of applications. All the data-driven methods make use of system signals as failure information carriers. Their data features can range from statistical parameters in the time-domain, up to time-frequency domain features.

Bayes-classifiers:

Bayes-classifiers have been widely used for fault detection and diagnosis [136] [137] [138] [139]. It estimates the probability distributions of different attributes (among others such as data mean, standard deviation, amplitude, and frequency) that represent symptoms, given the class from a training data set [140]. It aims to minimize the average probability error, in order to provide reliable classification results [141]. The analyzed dataset should present a Gaussian probability density distribution, as well as its class specific densities [142]. This is a significant drawback as few datasets can actually meet this particular requirement. Bayes-classifiers use probability as decision enabler. It determines how likely is a particular dataset belongs to a specific class. A probability threshold is chosen for making a decision on the occurring failure mode.

Support vector machine:

A support vector machine (SVM) uses classification distance as criteria for determining the occurrence of a failure mode [143] [144] [145] [146] [147]. On its classical version, this method determines a hyper-plane that optimally divides data corresponding to two different classes through train data sets [148]. Although this method has a high predictive accuracy, it is very sensitive to the parameter selection [149]. It is also

sensitive to overlapping groups, as observations near one another are treated alike [150]. Although the classical implementation is limited to a two-class classification problem, multiclass classification can be used by implementing a one-versus-all strategy. In SVM, the obtained hyper-plane plays the role of decision enabler and the reference for determining the occurring failure mode.

Linear discriminant analysis:

Linear discriminant analysis (LDA) implements a projection hyperplane that maximizes the distance between the projected means of the different classes considered [151]. Failures are classified on a projected hyperplane based on data density. There have been multiple implementations of LDA developed for failure diagnosis purposes [152] [153] [154] [155]. The limitation of this method is that it requires multivariate normality of the explanatory variables and equal covariance matrices [156].

In the case of LDA, the decision enabler for failure diagnosis is the distance between the projected means of the failure modes, as well as their variance. The criterion - or the reference used for determining the class (i.e. failure mode) of the observed measurement - is provided by a threshold value. This threshold is a central point between the means of the different classes. The above-considered methods are based on supervised learning. Supervised learning methods are typically considered as classification techniques, whereas the unsupervised learning methods are regarded as pattern recognition methods. They can recognize hidden relationship in unlabeled data [157]. In this section, we will only analyze K-means. However, there are multiple statistic-based methods into this category.

K-means:

K-means identifies data patterns based on distance of data points. This method does not require prior specification of the classes in the training phase. It enables classification not only of known failures, but also of emerging ones [158]. K-means determine the distance of any new observation with respect to the center of each of the available classified failure clusters, and classifies it. This capability is strongly desired for the investigation of non-linear systems. However, these methods are not as powerful and reliable as the supervised ones, as they have troubles for dealing with overlapping data. Applications in the fault analysis context can be found in [159] [160] [161]. This method is more useful for data mining for the simple reason that the identified patterns on the analyzed data can be used as a basis for further classification processes. K-means considers the Euclidean distance from data points to the centroid of the different classes considered (i.e. failure modes) as decision enabler and as reference.

Principal component analysis:

Principal component analysis (PCA) is a statistical method that aims to reduce the dimensionality of data, while retaining its variation as much as possible [162]. It enables projecting multiple dimensions into one or more dimensions that explains in greater extent data variation. It ease data analysis, as it enables focusing in a smaller set of uncorrelated variables [117]. Although this method per se is not capable of diagnosing failures, it is widely used as a preprocessing step before of implementing ANN or any SPC method. Han et alias integrated PCA, genetic algorithms and ANN for the development of a condition monitoring and fault diagnosis system for induction motors [163]. In their case of application, PCA and genetic algorithms were used to reduce

feature dimensionality. Aminian & Aminian proposed a neural network based fault diagnosis system where PCA is used to derive a set of features for training the ANN [164].

Thukaram et al. combined PCA, SVM and ANN for locating faults in radial distribution system [165]. In this case, PCA was used for preprocessing, and SVM and ANN are used as classifiers. Although PCA has shown to be very useful for reducing the dimensions of the datasets delivered to classifiers, it also presents important drawbacks. As PCA is a linear combination of all the variables initially considered, the obtained results are difficult to interpret [166]. The delivered data do not describe the sensed variables any more. They describe a new variable that is derived by the projection of the original variables in the direction that explains most of the data variation. Moreover, if the analyzed data is not linearly correlated or it is not scaled, PCA do not suffice for determining the direction that explains a greater extent data variation.

The failure information carriers are multiple system signals. Data features are extracted either from signals in the time domain or from signals in the frequency domain. These features are used as input for PCA. Principal component analysis delivers an array of transformed data, as well as the percentage of influence of every dimension on data variability. Those dimensions are selected and used as input for the classification method that in conjunction explain most of the data variability. The decision enabler, as well as the reference, depends on the selected method.

Qualitative trend analysis:

Qualitative trend analysis (QTA) uses of signal trends for predicting future states and diagnosing faults [79]. These trends are extracted from signal segments, called episodes, that present a unique set of signs for their first and second derivative [167], i.e. these are time segments where the sign does not change between its starting time and its ending time. Although this method has some quantitative elements, it is considered as a qualitative method, as the measured signals are represented by a sequence of shapes (i.e. trend directions), which are denoted by increase (“+”), decrease (“-”) and constant (“0”) behaviors [168].

Episodes can be described through their start time, end time, and primitives. Primitives can be defined as a tuple of signs for the first and second signal derivatives [169]. There are seven types of primitives, which can be used to represent any type of segment. These are: A(0, 0), B(+, +), C(+, 0), D(+, -), E(-, +), F(-, 0), G(-, -) [168]. The objective of this simplification is to reduce the measured data to a finite set of qualitative states that can be interpreted and assessed by operators easily [170]. For example, an increasing segment may represent acceleration of a vehicle, while a constant segment may represent permanent cruise speed. A sequence of primitives composes a trend.

Trend sequences and their likelihood are associated with fault scenarios. These can be used for characterizing faults, and thus, for their diagnosis. Experts do their reasoning about the obtained signal trends by elaborating on systems of low-level complexity. They can extract and interpret the observed trends easily and conclude about the observed trend behavior. However, increase of system complexity requires automation of trend analysis. Automated and reliable trend extraction is still a challenge, though [171]. Currently, it is conducted by fitting polynomials on the observed data. However, this method is sensitive to the presence of signal noise. To address these issues, research in QTA methods aims to (i) ease trend extraction [172], (ii) provide ways to deal with

discontinuities in the input data [173], (iii) handle consecutive inflection points [174], and (iv) provide a similarity measure that enables to classify faults based on qualitative sequences [170].

In the case of QTA, data features convey the extracted signal trends, which are represented through sequences of primitives. A database composed by different trend sequences that describe failure-free operation and the failed ones (one per known failure mode) are used as reference. Finally, the likelihood of occurrence of every single sequence of primitives constitutes the decision enablers when automating the reasoning process. This last criterion applies for systems whose failure reasoning is conducted automatically. However, the experts' knowledge can also be used as decision making in less complex systems.

2.6 Implications of the findings

The above Sub-sections provided an overview of the state of the art concerning to failure analysis. The most relevant approaches, along with their most representative methods, were analyzed by exploring their failure information carriers, data features, references and failure decision enablers. Our literature review revealed that signal-based methods can be used equally well for two purposes, namely (i) extracting signal features, and (ii) comparing signals. From the failure analysis perspective, the extraction of signal features is required for all the data-driven techniques concerning discard useless data and keeping these that convey information about the analyzed failure mode. The comparison of signals also enables failure detection and failure diagnosis. However, it presents some limitations in the context of CPSs. The frequent transitions between system operation modes enabled by the self-tuning capability of CPSs, and the disturbances and transient behavior it causes may lead to false failure alarms. Moreover, the analysis of signals in the frequency-domain does not consider the time variable, which is where SOM transitions occur. It hampers the differentiation of a failure effect and a SOM transition effect. If we extensively analyze the literature, we can find plenty of articles and papers that present signal-based failure analysis. However, considering the above-mentioned situation, we consider its use for pre-processing data can be more profitable in the context of CPSs.

The analysis of model-based methods revealed that these are one of the most widely used methods nowadays. They provide an analytically derived reference output that can be compared with the observed one in order to determine deviations that can be adjudged to failures. However, it presupposes a predictable and stable system behavior that is not subjected to emergent or multiple behaviors. We consider that, the implementation of context-based models (one per SOM) could be an option to overcome the above-described limitation. However, the self-regulation and self-tuning capabilities that CPSs imply can lead to very frequent SOM transitions. Moreover, high end CPSs can evolve and change its configuration due to its learning capabilities. Although self-evolvable systems are not available yet, it is the future of CPSs and any new failure management principle should be created with a view to their further development. An evolutionary system would cause that the analytical models available would become obsolete as the system evolves. It would require updating the existing models as the system changes its characteristics. It would be impractical, considering how fast these

changes can occur. Moreover, if we consider that model-based approaches do not have learning capabilities.

If we compare the applicability of the quantitative model-based approaches with that of the qualitative model-based ones, the qualitative models have more potential to be applied in CPSs. On the one hand, the need for mathematical equations at representing physical processes in highly complex systems limits. It implies expensive computation processes and conducting simplifications and assumptions that may affect the diagnosis results. On the other hand, qualitative models have demonstrated to be very powerful, as these are more flexible. It enables tackling better the dynamic operation of CPSs. Modelling expert knowledge, and updating expert-based models is still a challenge, though. Its application is limited by the knowledge available in experts. As for quantitative models, their implementation would require having multiple expert's models, one per SOM, as well as models that explain SOM transitions. This situation can be problematic in the case of a system that is subjected to emergent behavior and that is under constant evolution. In a futuristic scenario, the modeling of expert knowledge would take longer than the evolution of the system. It would hamper its application in high-level CPSs. The model-based failure analysis can still be used in CPSs. It can be implemented in subsystems that present steady state behavior. It can also be used in systems, which are only moderately, or not at all, affected by the surrounding environment and/or by the changing interrelations with other subsystems. Methods such as state observers have proved to be very useful for determining failures in complex systems. However, their operation is mainly focused on specific system tasks.

Unlike model-based approaches, data-driven approaches are more flexible for dealing with self-tuning and uncertain situations. They do not depend on analytical models that describe the interrelation of components. Their learning capabilities enable discovering such relationships. It represents an enormous potential for CPSs, as it would enable tackling emergent behavior and system evolution. From the three domains analyzed in our reasoning model, the data-driven approaches are the most suitable for failure management in CPSs. However, the success of data-driven approaches strongly depends in a proper selection of data features. Signal-based analysis can analyze system data in the time and frequency domains. Nevertheless, the selection of the features is subjected to a trial and error process, which hampers its automation. It would make the recognition of new failure modes or the identification of the symptoms in the case of new SOMs difficult.

Enabled by self-tuning capabilities, the transitions between SOMs can also be an issue for data-driven techniques. As for signal-based analysis, the frequent transitions between system operation modes and the disturbances and transient behavior it implies can have a negative effect on failure detection and diagnosis. Data-driven techniques require the selection of data features, which are the main input for the analyzed classifiers. However, SOM transitions are not observable when a system signal is transformed into a data feature or (more data features). It can lead to misclassification if the system presents significant variations concerning the sequence of SOMs. Signal segmentation can be an option to enable the implementation of signal-based and data-driven approaches in cyber-physical systems. It would assure that the analyzed segment corresponds to a particular SOM, and thus, it would be possible to discriminate between a failure effect and a SOM transition effect. However, the effectiveness of this approach

should still be evaluated. The main limitations of data-driven techniques concern the system evolution that is a paradigmatic system feature of high generation CPSs. Due to the stability-plasticity dilemma, the re-learning processes bring about serious challenges. It causes the model to forget already learned classes when it learns new clusters and patterns of relationships [175]. Moreover, unsupervised methods are not as powerful as supervised ones, and may lead to unreliable results.

The analysis of the key factors considered in the failure analysis process, i.e. failure information carriers, data features, references and decision enablers, revealed that most

Table 2.1 Summary of the key elements considered in failure analysis for all the analyzed techniques

Technique	Failure information carrier	Data features	Reference	Decision enabler
Parity relations	system signals (input/output)	system states	Analytically computed outputs	Residual
Sate observer				
Signed digraphs	Measured variables	NA	Knowledge base	Probabilistic distribution and/or fuzzy rules
Expert systems	Observable characteristics Image, Sounds Smell, etc. Measured variables	NA	Experts model output	Probabilistic distribution
Qualitative physics	Measured variables and actuator status	Categorical variables Symbolic variables Fuzzy sets	Sequence of states derived through the model	Deviation from the estimated sequence of states
ANN	system signals	Statistical parameters Frequency-domain features Time-frequency-domain features	Entered classes (training dataset)	Output of the ANN model
Bayes classifier			Probability threshold	Probability distribution
SVM			Hyper-plane	Hyper-plane
LDA			Central point between the means of the different classes	Distance of the projected mean and variance
K-means			Euclidean distance	Euclidean distance
QTA		Trends	Trends sequences	Probability distribution

of the existing methods analyze system signals (through a constant monitoring), or measure system variables (by taking data samples sporadically) for determining failure occurrence (Table 2.1). Nowadays, sensors can measure certain characteristics, which used to be evaluated by the experts, such as visuals, sounds, postures, smells and physical changes. It allows treating them as signals, and thus it enables the use of data-driven techniques and the automation of the failure diagnosis process in CPSs. Although the use of output signals, i.e. sensed system variables, is very common, our literature review also revealed that few of them consider the control signals as criterion for failure diagnosis. It is the case of state observer, and qualitative physics.

The data obtained about system actuators, along with other system variables, can be used to determine the time instants in which SOM transitions occur. It is very useful in practice, as it can be used as reference for activating a different system model (if a model-based failure management approach is used), and for varying the reference thresholds according to the working conditions. During the processing of the data, information about the actuator status can contribute to discriminate between deviations caused by transient behavior or variations on the system settings, and deviations caused by failure effect. It would contribute to a more reliable implementation of signal-based and data-driven techniques, as long as the analyzed data keep record of the time variable.

It has already been stated that feature selection is a critical issue, particularly for data-driven methods. All data-driven approaches include this step. However, feature selection mostly relies on expert's knowledge. One of the main difficulties it entails is that feature selection also depends on the type of the analyzed system and the failure mode to be diagnosed. It implies that certain failure modes manifest better in a particular feature, while some others do in others. Whenever there is a new failure mode, a data mining process should be completed in order to determine its main symptoms in the system signals. This process is even more difficult for failure diagnosis, as the characteristic pattern the failure causes on the signals should be identified with the objective of enabling failure classification. System operation modes can also affect feature selection. Considering system behaves different during different SOMs, it is reasonable to think that certain SOMs can favor failure manifestation of certain failure modes, while some others can mask it. Further research in this topic is required.

About reference and decision enablers, both issues are closely related. Probability-based decision enablers can be used for multiple applications. They are estimated based on the observed data, enabling to determine failure occurrence. Probability-based decision is easily interpretable, as they do not require determining a threshold (i.e. reference) according to the observed process/system. It means that the same threshold can be used indistinctively, no matter of the analyzed feature. It also favors comparison. Other references and decision enablers, such as residuals, require experts to determine what the acceptable difference between an estimated quantity and the observed one is. This process requires a prior experience on the process performance - to be able to determine from which reference it is possible to suspect the existence of a failure. This situation is even worst in the case of qualitative methods, where the deviation between the estimated state and the observed one cannot be measured. Variable thresholds are also an issue. In those cases, in which failures manifest different, due to the working conditions, it might be required to determine a set of thresholds or a context-based

strategy. This strategy would determine the most suitable threshold for evaluating system condition in the actual working context.

Although probability-based decision enablers seem to be the most suitable options for failure analysis in CPSs, they have some drawbacks. They are underpinned by various assumptions about the distribution of data, and they can be very sensitive to the sample size. It may cause false positive results that could lead to wrong failure diagnosis. Moreover, the decision making when the estimated probability is neither high, nor low, is not possible. It makes it required to wait until the failure develops further, so that, maintenance actions can be conducted. However, it could endanger the cost-effective operation of the system. To our best knowledge, the influence of SOMs on the failure detection and failure diagnosis has not been explored yet in literature. We have already stated that SOMs imply multiple different system behavior. It is reasonable to think that it also implies different failure manifestations. It may have an effect in the selection of the failure information carriers, data features, references and decision enablers. A properly understanding of the effect of SOMs on failure manifestations is then required.

Based on the afore-conducted analysis, data-driven techniques are the most suitable methods for failure analysis in the context of CPSs. The intensive sensing processes that CPSs typically implement a reason typically implement provide a reason for thinking so. Due to this, it is reasonable to use system signals as an instrument for studying the effect of SOMs on failure manifestations. However, to to achieve a proper understanding, it is required to be able to discriminate between the effects of SOM transitions, the effects of failures, and the effects of external disturbances. Signal segmentation based on SOM seems to be a suitable option to discriminate between the control actions and the failure effects. Nevertheless, it must be noted that the effects of the environment, as well as the variations in the use and the operating conditions should also be evaluated.

2.7 Conclusions

The latest challenge is supporting dependability, maintenance and repair of mission critical cyber-physical systems of self-regulatory and self-tuning capabilities [176]. Due to (i) the natural complexity and growing intellectualization of these systems, as well as to (ii) the conventional nature of the development tools used and (iii) the limited skills available to deal with uncertain situations, there is an urgent need to develop new scientific principles and methodologies for maintaining high-level dependence of CPSs upon which our lives will very much depend already in the near future [177]. New methodologies and tools are needed not only to support designing for dependability in design, but also to provide means for controlling dependability in run-time. As we argue below, shifting system operation modes is a concept that has not been explored in deep yet in the context of failure analysis. That is the reason why we have chosen this phenomenon for an extensive study.

The objective of this chapter was to analyze to what extent system operation modes have been explored in the context of failure analysis. A review of the most relevant fault analysis methods was conducted, in which signal-based, model-based and data-driven based techniques were analyzed. It was found that most of the methods make use of system signals and system parameters as failure information carriers for detecting and

diagnosing failures. It was also found that only few methods consider control actions and actuators status as input for failure diagnosis. Although many of the methods found in literature make use of analytical models as basis for determining failure occurrence, the analysis of the main characteristics of CPSs indicates that their implementation in these types of systems is impractical, due to the self-tuning, self-adaptation, and self-evolution capabilities.

In like manner, our literature review revealed that, data-driven techniques are the most suitable for failure management in CPSs, considering these types of systems have to deal with emergent behaviors. The data-driven learning capabilities provide the opportunity and the means for identifying and learning new system behaviors, as well as new failure modes. Nevertheless, these types of systems strongly depend on the selection of data features, which is still a trial error process. This situation can also have a negative effect when considering SOMs. Transforming system signals in data features neglect the time dimension. It causes that transient system behavior caused by SOM transitions, as well as the variations on the system behavior that SOMs imply can be confounded with failure effect. It can lead to false failure alarms, or false positive results. It is also the case with signal-based analysis, particularly in the frequency-domain.

There was no evidence found in literature that any extensive investigations are conducted with regards to the role of SOMs in failure analysis in the context of CPSs. For this reason, we see our initiative as a pioneering work. Our investigations have been conducted with the objective to provide a robust and flexible basis for failure recognition and forecasting, and for developing principles for preventive failure management for systems that present self-regulation and self-tuning capabilities. For this purpose, it was required: (i) implementing data-driven methods, which are provided with learning capabilities, (ii) using system output signals as failure information carriers, and (iii) differentiating between deviations caused by SOM effect, deviations caused by the environment, and deviations caused by failure effect. This last condition is defined with the aim of properly understanding the effect of SOM transitions on failure manifestations.

2.8 References

- [1] Z. Gao, C. Cecati, and S. X. Ding, "A Survey of Fault Diagnosis and Fault-Tolerant Techniques; Part I: Fault Diagnosis With Model-Based and Signal-Based Approaches," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757–3767, Jun. 2015.
- [2] Y. Shu, L. Ming, F. Cheng, Z. Zhang, and J. Zhao, "Abnormal situation management: Challenges and opportunities in the big data era," *Comput. Chem. Eng.*, vol. 91, no. Supplement C, pp. 104–113, Aug. 2016.
- [3] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis Part I," *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 293–311, Mar. 2003.
- [4] D. Wang, M. Yu, C. B. Low, and S. Arogeti, "Health Monitoring of Engineering Systems," in *Model-based Health Monitoring of Hybrid Systems*, New York, NY, USA: Springer, 2013, pp. 1–29

- [5] C. Talcott, "Cyber-Physical Systems and Events," in *Software-Intensive Systems and New Computing Paradigms*, vol. 5380, M. Wirsing, J.-P. Banâtre, M. Hölzl, and A. Rauschmayer, Eds. Berlin, Germany: Springer Berlin Heidelberg, 2008, pp. 101–115.
- [6] R. (Raj) Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: the next computing revolution," in *47th DAC.*, New York, NY, USA, 2010, pp. 731–736.
- [7] P. Li, Q. Xu, and N. Wang, "Evolution Course and Analysis of Internet of Things," in *Proc. (IEA) 2012*, Z. Zhong, Ed. Springer London, 2013, pp. 221–228.
- [8] S. Bansal, S. Sharma, and I. Trivedi, "A Detailed Review of Fault-Tolerance Techniques in Distributed System," *IJIDCS*, vol. 1, no. 1, Oct. 2017.
- [9] L. Li, "Introduction," in *Fault Detection and Fault-Tolerant Control for Nonlinear Systems*, Duisburg, Germany: Springer Vieweg, Wiesbaden, 2016, pp. 1–9.
- [10] A. Bahga and V. K. Madiseti, "Analyzing Massive Machine Maintenance Data in a Computing Cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 10, pp. 1831–1843, Oct. 2012.
- [11] T. Dillon, V. Potdar, J. Singh, and A. Talevski, "Cyber-physical systems: Providing Quality of Service (QoS) in a heterogeneous systems-of-systems environment," in *IEEE DEST 2011.*, pp. 330–335.
- [12] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annu. Rev. Control*, vol. 32, no. 2, pp. 229–252, Dec. 2008.
- [13] R. Isermann, "Fault detection of control loops," in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. R. Isermann, Ed. Berlin, Germany: Springer-Verlag, 2006, pp. 253–265.
- [14] V. Castano and I. Schagaev, "Dealing with Faults: Redundancy," in *Resilient computer system design*, 1st ed, Switzerland: Springer Cham, 2015, pp. 39–78.
- [15] R. Ahmad and S. Kamaruddin, "An overview of time-based and condition-based maintenance in industrial application," *Comput. Ind. Eng.*, vol. 63, no. 1, pp. 135–149, Aug. 2012.
- [16] S. Ruiz-Arenas, I. Horváth, R. Mejía-Gutiérrez, and E. Z. Opiyo, "Towards the Maintenance Principles of Cyber-Physical Systems," *Stroj. Vestn. – J. Mech. Eng.*, vol. 60, no. 12, pp. 815–831, Dec. 2014.
- [17] I. Eusgeld, B. Fechner, F. Salfner, M. Walter, P. Limbourg, and L. Zhang, "Hardware Reliability," in *Dependability Metrics*, F.C Eusgeld, F. Freiling, R. Reussner Eds. Berlin, Germany: Springer-Verlag Berlin Heidelberg, 2008, pp. 59–103.
- [18] A. Zolghadri, "Advanced model-based FDIR techniques for aerospace systems: Today challenges and opportunities," in *Prog. Aerosp. Sci.*, vol. 53, pp. 18–29, Aug. 2012.
- [19] E. Dubrova, "Hardware Redundancy," in *Fault-Tolerant Design*, E. Dubrova, Ed. New York, NY, USA: Springer New York. 2013, pp. 55–86.
- [20] B. Tabbache, A. Kheloui, M. E. H. Benbouzid, A. Mamoune, and D. Diallo, "Research on fault analysis and fault-tolerant control of EV/HEV powertrain," in *ICGE 2014.*, 2014, pp. 284–289.
- [21] P. Goupil, J. Boada-Bauxell, A. Marcos, E. Cortet, M. Kerr, and H. Costa, "AIRBUS efforts towards advanced real-time Fault Diagnosis and Fault Tolerant Control," *IFAC Proc. Vol.*, vol. 47, no. 3, pp. 3471–3476, Jan. 2014.

- [22] M. Saied, B. Lussier, I. Fantoni, C. Francis, H. Shraim, and G. Sanahuja, "Fault diagnosis and fault-tolerant control strategy for rotor failure in an octorotor," in *ICRA 2015.*, pp. 5266–5271.
- [23] J. Ma and J. Jiang, "Applications of fault detection and diagnosis methods in nuclear power plants: A review," *Prog. Nucl. Energy*, vol. 53, no. 3, pp. 255–266, Apr. 2011.
- [24] X. Teng, H. Pham, and D. R. Jeske, "Reliability Modeling of Hardware and Software Interactions, and Its Applications," in *IEEE Trans. Reliab.*, vol. 55, no. 4, pp. 571–577, Dec. 2006.
- [25] S. Lazarova-Molnar, N. Mohamed, and H. R. Shaker, "Reliability modeling of cyber-physical systems: A holistic overview and challenges," in *2017 MSCPE*, Pittsburgh, PA, USA, 2017, pp. 1–6.
- [26] N. G. Leveson, "Software Safety: Why, What, and How," in *ACM Computing Surveys (CSUR)*, vol. 18, no. 2, pp. 125–163, Jun. 1986.
- [27] M. Grottko, R. Matias, and K. S. Trivedi, "The fundamentals of software aging," in *ISSRE Workshop.*, 2008, pp. 1–6.
- [28] S. Savarino, R. Fitz, L. Fesq, G. Whitman, "Fault Management Architectures and the Challenges of Providing Software Assurance," presented at the Space Symp., 2015, pp. 1–10.
- [29] B. W. Boehm, "Software risk management: principles and practices," in *IEEE Software*, vol. 8, no. 1, pp. 32–41, Jan. 1991.
- [30] J. Xuan, Y. Hu, and H. Jiang, "Debt-Prone Bugs: Technical Debt in Software Maintenance," in *IJACT*, 2017, pp. 1–9.
- [31] S. Christa, V. Madhusudhan, V. Suma, and J. J. Rao, "Software Maintenance: From the Perspective of Effort and Cost Requirement," in *Proc. ICDECT 2016.*, Vol. 2, 2017, pp. 759–768.
- [32] R. K. Iyer and P. Velardi, "Hardware-Related Software Errors: Measurement and Analysis," *IEEE Trans. Softw. Eng.*, vol. SE-11, no. 2, pp. 223–231, Feb. 1985.
- [33] X. Dai and Z. Gao, "From Model, Signal to Knowledge: A Data-Driven Perspective of Fault Detection and Diagnosis," in *IEEE Trans. Ind. Inform.*, vol. 9, no. 4, pp. 2226–2238, Nov. 2013.
- [34] C. L. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on Big Data," *Inf. Sci.*, vol. 275, no. Supplement C, pp. 314–347, Aug. 2014.
- [35] J. Fan, F. Han, and H. Liu, "Challenges of Big Data analysis," *Natl. Sci. Rev.*, vol. 1, no. 2, pp. 293–314, Jun. 2014.
- [36] H. V. Jagadish *et al.*, "Big Data and Its Technical Challenges," *Commun ACM*, vol. 57, no. 7, pp. 86–94, Jul. 2014.
- [37] P. Albertos and I. Mareels, "Signal Analysis," in *Feedback and Control for Everyone*, Berlin, Germany: Springer Berlin Heidelberg, 2010, pp. 87–112.
- [38] Zhiyue Lin, "An introduction to time-frequency signal analysis," *Sens. Rev.*, vol. 17, no. 1, pp. 46–53, Mar. 1997. Accessed on: Dec, 14, 2017, doi: 10.1108/02602289710163364, Available: <http://www.emeraldinsight.com/doi/full/10.1108/02602289710163364>
- [39] Y. Lei, "Signal processing and feature extraction," in *Intelligent fault diagnosis and remaining useful life prediction of rotating machinery*, Elsevier Inc, 2017, pp. 17–66.

- [40] W. A. Woyczyński, “Stationary Signals,” in *A First Course in Statistics for Signal Analysis*, Boston, MA, USA: Birkhäuser Boston, 2011, pp. 105–125.
- [41] Y. Shmaliy, “Introduction,” in *Continuous-Time Signals*. Dordrecht, The Netherlands: Springer, 2006, pp. 1–46.
- [42] P. J. Fish, “Random Signals,” in *Electronic Noise and Low Noise Design*, Palgrave, London: Macmillan New Electronics Series, 1993, pp. 6–30.
- [43] A. K. Ziarani and A. Konrad, “An adaptive noise reduction technique,” in *MWSCAS 2001*, Dayton, OH, USA, 2001, pp. 251–254.
- [44] M. S. Alencar and V. Da Rocha, “Signal Analysis,” in *Communication Systems*, Springer US, 2005, pp. 1–40.
- [45] J.-C. Montaño, J.-C. Bravo, and M.-D. Borrás, “Joint Time–Frequency Analysis of the Electrical Signal,” in *Power Quality*, A. Moreno-Muñoz, Ed. London, England: Springer London, 2007, pp. 41–72.
- [46] M. Müller, “Fourier Analysis of Signals,” in *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*, M. Müller, Ed. Cham, Germany: Springer International Publishing, 2015, pp. 39–114.
- [47] P. M. Ramos, R. C. Martins, S. Rapuano, and P. Daponte, “Frequency and Time—Frequency Domain Analysis Tools in Measurement,” in *Data Modeling for Metrology and Testing in Measurement Science*, F. Pavese and A. B. Forbes, Eds. Birkhäuser Boston, 2009, pp. 1–27.
- [48] R. Isermann, “Fault detection with signal models,” in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, R. Isermann, Ed. Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 111–146.
- [49] H. Guo, J. A. Crossman, Y. L. Murphey, and M. Coleman, “Automotive signal diagnostics using wavelets and machine learning,” *IEEE Trans. Veh. Technol.*, vol. 49, no. 5, pp. 1650–1662, Sep. 2000.
- [50] Y. L. Murphey, J. A. Crossman, Z. H. Chen, and J. Cardillo, “Automotive fault diagnosis - Part II: A distributed agent diagnostic system,” *IEEE Trans. Veh. Technol.*, vol. 52, no. 4, pp. 1076–1098, Jul. 2003.
- [51] M. J. Daigle, I. Roychoudhury, and A. Bregon, “Qualitative event-based diagnosis applied to a spacecraft electrical power distribution system,” *Control Eng. Pract.*, vol. 38, pp. 75–91, May. 2015.
- [52] M. Abul Masrur, Z. Chen, and Y. Murphey, “Intelligent diagnosis of open and short circuit faults in electric drive inverters for real-time applications,” *IET Power Electron.*, vol. 3, no. 2, pp. 279–291, Mar. 2010.
- [53] M. Barakat, F. Druaux, D. Lefebvre, M. Khalil, and O. Mustapha, “Self adaptive growing neural network classifier for faults detection and diagnosis,” *Neurocomputing*, vol. 74, no. 18, pp. 3865–3876, Nov. 2011.
- [54] G.-M. Xian and B.-Q. Zeng, “An intelligent fault diagnosis method based on wavelet packer analysis and hybrid support vector machines,” *Expert Syst. Appl.*, vol. 36, no. 10, pp. 12131–12136, Dec. 2009.
- [55] G. G. Yen and P. Meesad, “An effective neuro-fuzzy paradigm for machinery condition health monitoring,” *2011 IEEE CCA*, vol. 2, pp. 1567–1572.
- [56] S. X. Ding, “Introduction,” in *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms and Tools*, S. X. Ding, Ed. London, England: Springer London, 2013, pp. 3–11.
- [57] J. Gertler, “Fault detection and isolation using parity relations,” *Control Eng. Pract.*, vol. 5, no. 5, pp. 653–661, May. 1997.

- [58] R. D. Lorenzo, "Transfer Functions," in *Trading Systems: Theory and Immediate Practice*, Milano, Italy: Springer-Verlag Mailand, 2013, pp. 111–114.
- [59] K. M. Hantos, J. Bokor, and G. Szederkényi, Eds., "State-space Models," in *Analysis and Control of Nonlinear Process Systems*, London, England: Springer London, 2004, pp. 23–37.
- [60] R. Isermann, "Fault detection with parity equations," in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, R. Isermann, Ed. Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 197–229.
- [61] R. Isermann, "Fault detection with state observers and state estimation," in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, R. Isermann, Ed. Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 231–252.
- [62] D. Luenberger, "An introduction to observers," *IEEE Trans. Autom. Control*, vol. 16, no. 6, pp. 596–602, Dec. 1971.
- [63] A. B. Youssef, S. K. E. Khil, and I. Slama-Belkhodja, "State Observer-Based Sensor Fault Detection and Isolation, and Fault Tolerant Control of a Single-Phase PWM Rectifier for Electric Railway Traction," *IEEE Trans. Power Electron.*, vol. 28, no. 12, pp. 5842–5853, Dec. 2013.
- [64] H. Lee, S. Steven, A. Patterson, and N. Hovakimvan, "Aircraft actuator fault detection and isolation using piecewise constant fault estimation scheme," presented at AIAA Guidance, Navigation, and Control Conf., San Diego, California, 2016, pp. 1-9.
- [65] W. Chen, W. T. Chen, M. Saif, M. F. Li, and H. Wu, "Simultaneous Fault Isolation and Estimation of Lithium-Ion Batteries via Synthesized Design of Luenberger and Learning Observers," in *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 1, pp. 290–298, Jan. 2014.
- [66] J. H. Lee and N. L. Ricker, "Extended Kalman Filter Based Nonlinear Model Predictive Control," *Ind. Eng. Chem. Res.*, vol. 33, no. 6, pp. 1530–1541, Jun. 1994.
- [67] X. Chen and M. Liu, "A two-stage extended kalman filter method for fault estimation of satellite attitude control systems," *J. Frankl. Inst.*, vol. 354, no. 2, pp. 872–886, Jan. 2017.
- [68] K. Yan, Z. Ji, and W. Shen, "Online fault detection methods for chillers combining extended kalman filter and recursive one-class SVM," *Neurocomputing*, vol. 228, pp. 205–212, Mar. 2017.
- [69] R. K. Singleton, E. G. Strangas, and S. Aviyente, "Extended Kalman Filtering for Remaining-Useful-Life Estimation of Bearings," *IEEE Trans. Ind. Electron.*, vol. 62, no. 3, pp. 1781–1790, Mar. 2015.
- [70] S. R. K. Hoseini, E. Farjah, T. Ghanbari, and H. Givi, "Extended Kalman filter-based method for inter-turn fault detection of the switched reluctance motors," *IET Electr. Power Appl.*, vol. 10, no. 8, p. 714–722(8), Sep. 2016.
- [71] Z. Liu and H. He, "Sensor fault detection and isolation for a lithium-ion battery pack in electric vehicles using adaptive extended Kalman filter," *Clean Effic. Afford. Energy Sustain. Future*, vol. 185, pp. 2033–2044, Jan. 2017.
- [72] A. S. Willsky, J. J. Deyst Jr., and B. S. Crawford, "Adaptive Filtering and Self-test Methods for Failure Detection and Compensation," in *Jt Autom Control Conf, 15th Proc.*, 1974, pp. 637–645.

- [73] S. Yin, S. X. Ding, X. Xie, and H. Luo, "A Review on Basic Data-Driven Approaches for Industrial Process Monitoring," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6418–6428, Jan. 2014.
- [74] V. Puig, T. Escobet, R. Sarrate, and J. Quevedo, "Fault Diagnosis and Fault Tolerant Control in Critical Infrastructure Systems," in *Intelligent Monitoring, Control, and Security of Critical Infrastructure Systems*, Polycarpou M. (eds), Berlin, Germany: Springer Berlin Heidelberg, 2015, pp. 263–299.
- [75] G.-C. Luh, C.-Y. Wu, and W.-C. Cheng, "Artificial Immune Regulation (AIR) for Model-Based Fault Diagnosis," in *ICARIS 2004, Catania, Sicily, Italy*, pp. 28–41.
- [76] O. Zweigle, B. Keil, M. Wittlinger, K. Häussermann, and P. Levi, "Recognizing Hardware Faults on Mobile Robots Using Situation Analysis Techniques," in *Intelligent Autonomous Systems 12*. Berlin, Germany: Springer Berlin Heidelberg, 2013, pp. 397–409.
- [77] C. D. Bocaniala and V. Palade, "Computational Intelligence Methodologies in Fault Diagnosis: Review and State of the Art," in *Computational Intelligence in Fault Diagnosis*, Bocaniala C.D. ed. London, England: Springer London, 2006, pp. 1–36.
- [78] V. Venkatasubramanian, R. Rengaswamy, and S. N. Kavuri, "A review of process fault detection and diagnosis Part II," *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 313–326, Mar. 2003.
- [79] A. Shui, W. Chen, P. Zhang, S. Hu, and X. Huang, "Review of fault diagnosis in control systems," in *2009 Chinese Control and Decision Conference*, 2009, pp. 5324–5329.
- [80] H. WANG, B. CHEN, X. HE, T. QIU, and L. ZHANG, "A Signed Digraphs Based Method for Detecting Inherently Unsafe Factors of Chemical Process at Conceptual Design Stage," *Chin. J. Chem. Eng.*, vol. 16, no. 1, pp. 52–56, Feb. 2008.
- [81] B. Ould Bouamama, G. Biswas, R. Loureiro, and R. Merzouki, "Graphical methods for diagnosis of dynamic systems: Review," *Annu. Rev. Control*, vol. 38, no. 2, pp. 199–219, Jan. 2014.
- [82] Y. J. Liu, Q. H. Meng, M. Zeng, and S. G. Ma, "Fault diagnosis method based on probability extended SDG and fault index," in *2016 WCICA*, Guilin, China, 2016, pp. 2868–2873.
- [83] N. Lü and X. Wang, "A Probabilistic SDG Approach to Fault Diagnosis of Industrial Systems," in *ICIC 2007, Qingdao, China*, pp. 517–527.
- [84] D. Peng, X. Gu, Y. Xu, and Q. Zhu, "Integrating probabilistic signed digraph and reliability analysis for alarm signal optimization in chemical plant," *J. Loss Prev. Process Ind.*, vol. 33, pp. 279–288, Jan. 2015.
- [85] E. E. Tarifa and N. J. Scenna, "Fault diagnosis for MSF dynamic states using a SDG and fuzzy logic," *Desalination*, vol. 166, pp. 93–101, Aug. 2004.
- [86] J. H. Park and P. H. Seong, "Nuclear power plant pressurizer fault diagnosis using fuzzy Signed-Digraph and spurious faults elimination methods," *Ann. Nucl. Energy*, vol. 21, no. 6, pp. 357–369, Jun. 1994.
- [87] X. Z. Wang, S. A. Yang, E. Veloso, M. L. Lu, and C. McGreavy, "Qualitative process modelling - a fuzzy signed directed graph method," *Comput. Chem. Eng.*, vol. 19, no. Supplement 1, pp. 735–740, Jun. 1995.
- [88] X. Z. Wang, S. A. Yang, S. H. Yang, and C. McGreavy, "The application of fuzzy qualitative simulation in safety and operability assessment of process

- plants,” *Eur. Symp. Comput. Aided Process Eng.-6*, vol. 20, no. Supplement 1, pp. S671–S676, Jan. 1996.
- [89] Y. C. Huang and X. Z. Wang, “Application of fuzzy causal networks to waste water treatment plants,” *Chem. Eng. Sci.*, vol. 54, no. 13, pp. 2731–2738, Jul. 1999.
- [90] C. Grosan and A. Abraham, “Rule-Based Expert Systems,” in *Intelligent Systems: A Modern Approach*, vol. 17, Berlin, Germany: Springer Berlin Heidelberg, 2011, pp. 149–185.
- [91] W. Rebizant, J. Szafran, and A. Wiszniewski, “Expert Systems,” in *Digital Signal Processing in Power System Protection and Control*, London, England: Springer London, 2011, pp. 293–301.
- [92] M. Žarković and Z. Stojković, “Analysis of artificial intelligence expert systems for power transformer condition monitoring and diagnostics,” *Electr. Power Syst. Res.*, vol. 149, pp. 125–136, Aug. 2017.
- [93] J.-Z. Tang and Q.-F. Wang, “Online fault diagnosis and prevention expert system for dredgers,” *Expert Syst. Appl.*, vol. 34, no. 1, pp. 511–521, Jan. 2008.
- [94] F. Xu, H. Li, X. Yang, and B. Li, “Fault Diagnosis System of Electrical System Based on Classified Hierarchical Model,” in *AMEME 2016.*, pp. 1–6.
- [95] P. Nabende and T. Wanyama, “An Expert System for Diagnosing Heavy-Duty Diesel Engine faults,” in *Advances in Computer and Information Sciences and Engineering*, T. Sobh, Ed. Dordrecht: Springer Netherlands, 2008, pp. 384–389.
- [96] Y. Shao, B. Liu, G. Li, and R. Yan, “A Fault Diagnosis Expert System for Flight Control Software Based on SFMEA and SFTA,” in *2017 IEEE QRS-C.*, Prague, Czech Republic, 2017, pp. 626–627.
- [97] D. Kritzinger, “Fault tree analysis,” in *Aircraft System Safety*, D. Kritzinger, Ed. Cambridge, England: Woodhead Publishing, 2017, pp. 59–99.
- [98] O. Bäckström, Y. Butkova, H. Hermanns, J. Krčál, and P. Krčál, “Effective Static and Dynamic Fault Tree Analysis,” in *SAFECOMP 2016.*, Trondheim, Norway, pp. 266–280.
- [99] S. Kabir, “An overview of fault tree analysis and its application in model based dependability analysis,” *Expert Syst. Appl.*, vol. 77, pp. 114–135, Jul. 2017.
- [100] J. Hao, L. Zhang, and L. Wei, “Reliability Analysis Based on Improved Dynamic Fault Tree,” in *Engineering Asset Management 2011: WCEAM 2011*, pp. 283–299.
- [101] W. S. Lee, D. L. Grosh, F. A. Tillman, and C. H. Lie, “Fault Tree Analysis, Methods, and Applications #2013; A Review,” *IEEE Trans. Reliab.*, vol. R-34, no. 3, pp. 194–203, Aug. 1985.
- [102] M. A. Boyd, “Dynamic Fault Tree Models: Techniques for Analysis of Advanced Fault Tolerant Computer Systems,” Ph.D. dissertation, Duke University, Durham, NC, 1992.
- [103] K. D. Rao, V. Gopika, V. V. S. S. Rao, H. S. Kushwaha, A. K. Verma, and A. Srividya, “Dynamic fault tree analysis using Monte Carlo simulation in probabilistic safety assessment,” *Reliab. Eng. Syst. Saf.*, vol. 94, no. 4, pp. 872–883, Apr. 2009.
- [104] J. Wang *et al.*, “Fault-tree-based instantaneous risk computing core in nuclear power plant risk monitor,” *Ann. Nucl. Energy*, vol. 95, pp. 35–41, Sep. 2016.
- [105] P. Liu, L. Yang, Z. Gao, S. Li, and Y. Gao, “Fault tree analysis combined with quantitative analysis for high-speed railway accidents,” *Saf. Sci.*, vol. 79, pp. 344–357, Nov. 2015.

- [106] D. N. Dongiovanni and T. Iesmantas, "Failure rate modeling using fault tree analysis and Bayesian network: DEMO pulsed operation turbine study case," *Fusion Eng. Des.*, vol. 109, pp. 613–617, Nov. 2016.
- [107] S. Kabir, M. Walker, Y. Papadopoulos, E. Rüde, and P. Securius, "Fuzzy temporal fault tree analysis of dynamic systems," *Int. J. Approx. Reason.*, vol. 77, pp. 20–37, Oct. 2016.
- [108] K. D. Forbus, "Qualitative Physics: Past, Present and Future," in *Exploring Artificial Intelligence: Survey Talks from the National Conferences on Artificial Intelligence*, H. E. Shrobe Ed. San Mateo, CA, USA: American Association for artificial intelligence, 1988, ch.7, pp. 239–290.
- [109] R. Janowsky, "An introduction to QSIM and qualitative simulation," *Artif. Intell. Eng.*, vol. 2, no. 2, pp. 65–71, Apr. 1987.
- [110] B. Kuipers, "Qualitative simulation," *Artif. Intell.*, vol. 29, no. 3, pp. 289–338, Sep. 1986.
- [111] A. G. Cohn, "Qualitative reasoning," in *Advanced Topics in Artificial Intelligence: 2nd Advanced Course, ACAI 1987.*, R. T. Nossum, Ed, 1988, pp. 60–95.
- [112] Y. Lu, F. Wang, M. Jia, and Y. Qi, "Centrifugal compressor fault diagnosis based on qualitative simulation and thermal parameters," *Mech. Syst. Signal Process.*, vol. 81, pp. 259–273, Dec. 2016.
- [113] M. Platzner, B. Rinner, and R. Weiss, "Toward embedded qualitative simulation: a specialized computer architecture for QSim," *IEEE Intell. Syst. Their Appl.*, vol. 15, no. 2, pp. 62–68, Mar. 2000.
- [114] X. G. Zhang and Z. Ren, "A new modeling method of satellite fault diagnosis based on the qualitative/quantitative hybrid model," *Aerosp Control Appl*, vol. 35, no. 5, pp. 38–42, May. 2009.
- [115] E. Khalastchi, M. Kalech, G. A. Kaminka, and R. Lin, "Online data-driven anomaly detection in autonomous robots," *Knowl. Inf. Syst.*, pp. 657–688, May. 2014.
- [116] A. Alzghoul, B. Backe, M. Löfstrand, A. Byström, and B. Liljedahl, "Comparing a knowledge-based and a data-driven method in querying data streams for system fault detection: A hydraulic drive system application," *Comput. Ind.*, vol. 65, no. 8, pp. 1126–1135, Oct. 2014.
- [117] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin, "A review of process fault detection and diagnosis Part III," *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 327–346, Mar. 2003.
- [118] F. Neukart, "An outline of artificial neural networks," in *Reverse Engineering the Mind: Consciously Acting Machines and Accelerated Evolution*, Wiesbaden, Germany: Springer Fachmedien Wiesbaden, 2017, pp. 39–101.
- [119] A. Brabazon, M. O'Neill, and S. McGarraghy, "Neural Networks for Supervised Learning," in *Natural Computing Algorithms*, Berlin, Germany: Springer Berlin Heidelberg, 2015, pp. 221–259.
- [120] D. K. Chaturvedi, "Artificial Neural Network and Supervised Learning," in *Soft Computing: Techniques and its Applications in Electrical Engineering*, vol. 103, Berlin, Germany: Springer Berlin Heidelberg, 2008, pp. 23–50.
- [121] P. Melin and O. Castillo, "Unsupervised Learning Neural Networks," in *Hybrid Intelligent Systems for Pattern Recognition Using Soft Computing: An Evolutionary Approach for Neural Networks and Fuzzy Systems*, vol. 172, Berlin, Germany: Springer Berlin Heidelberg, 2005, pp. 85–107.

- [122] K. Patan, "Decision Making in Fault Detection," in *Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes*, vol. 377, Berlin, Germany: Springer Berlin Heidelberg, 2008, pp. 123–140.
- [123] J. Zarei, M. A. Tajeddini, and H. R. Karimi, "Vibration analysis for bearing fault detection and classification using an intelligent filter," *Mechatronics*, vol. 24, no. 2, pp. 151–157, Mar. 2014.
- [124] W. Li, A. Monti, and F. Ponci, "Fault Detection and Classification in Medium Voltage DC Shipboard Power Systems With Wavelets and Artificial Neural Networks," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 11, pp. 2651–2665, Nov. 2014.
- [125] T. Sorsa, H. N. Koivo, and H. Koivisto, "Neural networks in process fault diagnosis," *IEEE Trans. Syst. Man Cybern.*, vol. 21, no. 4, pp. 815–825, Jul. 1991.
- [126] A. A. Mazroua, M. M. A. Salama, and R. Bartnikas, "PD pattern recognition with neural networks using the multilayer perceptron technique," *IEEE Trans. Electr. Insul.*, vol. 28, no. 6, pp. 1082–1089, Dec. 1993.
- [127] A. Kumar and A. P. Singh, "Transistor level fault diagnosis in digital circuits using artificial neural network," *Measurement*, vol. 82, pp. 384–390, Mar. 2016.
- [128] W. Chine, A. Mellit, V. Lughi, A. Malek, G. Sulligoi, and A. Massi Pavan, "A novel fault diagnosis technique for photovoltaic systems based on artificial neural networks," *Renew. Energy*, vol. 90, pp. 501–512, May. 2016.
- [129] A. Torabi Jahromi, M. J. Er, X. Li, and B. S. Lim, "Sequential fuzzy clustering based dynamic fuzzy neural network for fault diagnosis and prognosis," *Neurocomputing*, vol. 196, pp. 31–41, Jul. 2016.
- [130] S. S. Tayarani-Bathaie, Z. N. S. Vanini, and K. Khorasani, "Dynamic neural network-based fault diagnosis of gas turbine engines," *Neurocomputing*, vol. 125, pp. 153–165, Feb. 2014.
- [131] J.-D. Wu and J.-M. Kuo, "An automotive generator fault diagnosis system using discrete wavelet transform and artificial neural network," *Expert Syst. Appl.*, vol. 36, no. 6, pp. 9776–9783, Aug. 2009.
- [132] P. Witzczak, K. Patan, M. Witzczak, and M. Mrugalski, "A neural network approach to simultaneous state and actuator fault estimation under unknown input decoupling," *Neurocomputing*, vol. 250, pp. 65–75, Aug. 2017.
- [133] X. Guo, L. Chen, and C. Shen, "Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis," *Measurement*, vol. 93, pp. 490–502, Nov. 2016.
- [134] W. Wu, M. Liu, Q. Liu, and W. Shen, "A quantum multi-agent based neural network model for failure prediction," *J. Syst. Sci. Syst. Eng.*, vol. 25, no. 2, pp. 210–228, Jun. 2016.
- [135] N. Ye, B. Zhao, and G. Salvendy, "Neural-networks-aided fault diagnosis in supervisory control of advanced manufacturing systems," *Int. J. Adv. Manuf. Technol.*, vol. 8, no. 4, pp. 200–209, Jul. 1993.
- [136] Z.-J. Zhou, C.-H. Hu, D.-L. Xu, J.-B. Yang, and D.-H. Zhou, "Bayesian reasoning approach based recursive algorithm for online updating belief rule based expert system of pipeline leak detection," *Expert Syst. Appl.*, vol. 38, no. 4, pp. 3937–3943, Apr. 2011.
- [137] C.S. Hood and C. Ji, "Proactive network-fault detection [telecommunications]," *IEEE Trans. Reliab.*, vol. 46, no. 3, pp. 333–341, Sep. 1997.

- [138] B. Krishnamachari and S. Iyengar, "Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks," *IEEE Trans. Comput.*, vol. 53, no. 3, pp. 241–250, Mar. 2004.
- [139] N. Mehranbod, M. Soroush, and C. Panjapornpon, "A method of sensor fault detection and identification," *J. Process Control*, vol. 15, no. 3, pp. 321–339, Apr. 2005.
- [140] M. Ramoni and P. Sebastiani, "Robust Bayes classifiers," *Artif. Intell.*, vol. 125, no. 1, pp. 209–226, Jan. 2001.
- [141] M. N. Murty and V. S. Devi, "Bayes Classifier," in *Pattern Recognition: An Algorithmic Approach*, London, England: Springer London, 2011, pp. 86–102.
- [142] R. Isermann, "Fault diagnosis with classification methods," in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*, R. Isermann, Ed. Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 295–310.
- [143] G. Yin, Y.-T. Zhang, Z.-N. Li, G.-Q. Ren, and H.-B. Fan, "Online fault diagnosis method based on Incremental Support Vector Data Description and Extreme Learning Machine with incremental output structure," *Neurocomputing*, vol. 128, pp. 224–231, Mar. 2014.
- [144] A. Swetapadma and A. Yadav, "Directional relaying using support vector machine for double circuit transmission lines including cross-country and inter-circuit faults," *Int. J. Electr. Power Energy Syst.*, vol. 81, pp. 254–264, Oct. 2016.
- [145] J. Hang, J. Zhang, and M. Cheng, "Application of multi-class fuzzy support vector machine classifier for fault diagnosis of wind turbine," *Themed Sect. Fuzzy Syst.*, vol. 297, pp. 128–140, Aug. 2016.
- [146] Y. Liu, L. Ye, P. Zheng, X. Shi, B. Hu, and J. Liang, "Multiscale classification and its application to process monitoring," *J. Zhejiang Univ. Sci. C*, vol. 11, no. 6, pp. 425–434, May. 2010.
- [147] V. Muralidharan, V. Sugumaran, and V. Indira, "Fault diagnosis of monoblock centrifugal pump using SVM," *Eng. Sci. Technol. Int. J.*, vol. 17, no. 3, pp. 152–157, Sep. 2014.
- [148] B. Kishore, M. R. S. Satyanarayana, and K. Sujatha, "Efficient fault detection using support vector machine based hybrid expert system," *Int. J. Syst. Assur. Eng. Manag.*, vol. 7, no. 1, pp. 34–40, Dec. 2016.
- [149] M. Stockman, R. S. E. Ramli, M. Awad, and R. Jabr, "An asymmetrical and quadratic Support Vector Regression loss function for Beirut short term load forecast," in *2012 IEEE SMC.*, pp. 651–656.
- [150] R. A. Berk, "Support Vector Machines," in *Statistical Learning from a Regression Perspective*, New York, NY, USA: Springer New York, 2008, pp. 1–28.
- [151] P. Xanthopoulos, P. M. Pardalos, and T. B. Trafalis, "Linear Discriminant Analysis," in *Robust Data Mining*, New York, NY, USA: Springer New York, 2013, pp. 27–33.
- [152] Y. Cai and M. Y. Chow, "Exploratory analysis of massive data for distribution fault diagnosis in smart grids," in *IEEE PES 2009 General Meeting*, pp. 1–6.
- [153] X. Jin, M. Zhao, T. W. S. Chow, and M. Pecht, "Motor Bearing Fault Diagnosis Using Trace Ratio Linear Discriminant Analysis," *IEEE Trans. Ind. Electron.*, vol. 61, no. 5, pp. 2441–2451, May. 2014.

- [154] A. M. D. Younus and B.-S. Yang, "Intelligent fault diagnosis of rotating machinery using infrared thermal image," *Expert Syst. Appl.*, vol. 39, no. 2, pp. 2082–2091, Feb. 2012.
- [155] C. P. Mbo'o and K. Hameyer, "Fault Diagnosis of Bearing Damage by Means of the Linear Discriminant Analysis of Stator Current Features From the Frequency Selection," *IEEE Trans. Ind. Appl.*, vol. 52, no. 5, pp. 3861–3868, Sep. 2016.
- [156] H. Yan and Y. Dai, "The Comparison of Five Discriminant Methods," in *ICMSS 2011*, pp. 1–4.
- [157] A. Abdolsamadi, P. Wang, and P. Tamilselvan, "A Generic Fusion Platform of Failure Diagnostics for Resilient Engineering System Design," in *ASME 2015*, vol. 2A: 41st Design Automation Conference, pp. 1-10.
- [158] W. Zhang, "Fault Diagnosis Method Based on Artificial Immune System," in *Failure Characteristics Analysis and Fault Diagnosis for Liquid Rocket Engines*, Berlin, Germany: Springer Berlin Heidelberg, 2016, pp. 193–217.
- [159] A. Glowacz and Z. Glowacz, "Diagnosis of stator faults of the single-phase induction motor using acoustic signals," *Appl. Acoust.*, vol. 117, pp. 20–27, Feb. 2017.
- [160] C. T. Yiakopoulos, K. C. Gryllias, and I. A. Antoniadis, "Rolling element bearing fault detection in industrial environments based on a K-means clustering approach," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2888–2911, Mar. 2011.
- [161] E. G. Strangas, S. Aviyente, and S. S. H. Zaidi, "Time-Frequency Analysis for Efficient Fault Diagnosis and Failure Prognosis for Interior Permanent-Magnet AC Motors," *IEEE Trans. Ind. Electron.*, vol. 55, no. 12, pp. 4191–4199, Dec. 2008.
- [162] I. Jolliffe, "Introduction," in *Principal Component Analysis*, 2nd Ed. New York, NY, USA: Springer New York, 2002, ch. 1, pp. 1–9.
- [163] T. Han, B.-S. Yang, and J. M. Lee, "A new condition monitoring and fault diagnosis system of induction motors using artificial intelligence algorithms," in *2005 IEEE IEMDC*, pp. 1967–1974.
- [164] F. Aminian and M. Aminian, "Fault Diagnosis of Nonlinear Analog Circuits Using Neural Networks with Wavelet and Fourier Transforms as Preprocessors," *J. Electron. Test.*, vol. 17, no. 6, pp. 471–481, Dec. 2001.
- [165] D. Thukaram, H. P. Khincha, and H. P. Vijaynarasimha, "Artificial neural network and support vector Machine approach for locating faults in radial distribution systems," *IEEE Trans. Power Deliv.*, vol. 20, no. 2, pp. 710–721, Apr. 2005.
- [166] H. Zou, T. Hastie, and R. Tibshirani, "Sparse Principal Component Analysis," *J. Comput. Graph. Stat.*, vol. 15, no. 2, pp. 265–286, Jun. 2006.
- [167] K. Villez and R. Rengaswamy, "A generative approach to qualitative trend analysis for batch process fault diagnosis," in *ECC 2013*, pp. 1958–1963.
- [168] M. R. Maurya, R. Rengaswamy, and V. Venkatasubramanian, "Fault Diagnosis by Qualitative Trend Analysis of the Principal Components," *Chem. Eng. Res. Des.*, vol. 83, no. 9, pp. 1122–1132, Sep. 2005.
- [169] K. Villez, "Qualitative trend analysis for process monitoring and supervision based on likelihood optimization: state-of-the-art and current limitations," *IFAC Proc. Vol.*, vol. 47, no. 3, pp. 7140–7145, Jan. 2014.
- [170] F. I. Gamero, J. Meléndez, and J. Colomer, "Process diagnosis based on qualitative trend similarities using a sequence matching algorithm," *J. Process Control*, vol. 24, no. 9, pp. 1412–1424, Sep. 2014.

- [171] B. Zhou and H. Ye, "A study of polynomial fit-based methods for qualitative trend analysis," *J. Process Control*, vol. 37, pp. 21–33, Jan. 2016.
- [172] B. Zhou, H. Ye, H. Zhang, and M. Li, "A new qualitative trend analysis algorithm based on global polynomial fit," *AIChE J.*, vol. 63, no. 8, pp. 3374–3383, Aug. 2017.
- [173] K. Villez and J. Habermacher, "Shape anomaly detection for process monitoring of a sequencing batch reactor," *Comput. Chem. Eng.*, vol. 91, pp. 365–379, Aug. 2016.
- [174] K. Villez, C. Rosén, F. Anctil, C. Duchesne, and P. A. Vanrolleghem, "Qualitative Representation of Trends (QRT): Extended method for identification of consecutive inflection points," *Comput. Chem. Eng.*, vol. 48, pp. 187–199, Jan. 2013.
- [175] H. Fernando and B. Surgenor, "An unsupervised artificial neural network versus a rule-based approach for fault detection and identification in an automated assembly machine," *Robot. Comput.-Integr. Manuf.*, vol. 43, pp. 79–88, 2017.
- [176] Horváth and Gerritsen, "Cyber-Physical Systems: Concepts, technologies and implementation principles," presented at the TMCE 2012, Karlsruhe, Germany, 2012, pp. 19–36.
- [177] L. Miclea and T. Sanislav, "About Dependability in Cyber-Physical Systems," in *2011 9th EWDTs.*, Sevastopol, Ukraine, 2011, pp. 17-21.

Chapter 3

A testbed system for empirical study of the influences of changing SOMs

3.1 Introduction

3.1.1 General objective

Growing self-intelligence and self-organization are the main characteristics of future cyber-Physical Systems (CPSs). In this sense, there is a significant difference between both, the operational behavior and architectural manifestation of traditional complicated/complex engineering systems and future smart CPSs. While the current trend makes it evident that progress is going on in this direction, the basis of the above-mentioned characteristics is not yet worked out completely in research and thus are not implemented in the systems used in practice. The matter of fact is that we witness the proliferation and articulation of the implementation of the first generation CPSs (1G-CPSs) that are able to achieve self-regulation in terms of control and self-tuning in terms of operational behavior and architectural manifestation capabilities [1]. Self-regulation and self-tuning enable CPSs to maintain their operations according to the set objectives under the effects of various influential factors, such as internal failures or external environmental disturbances. They adapt themselves by tuning their control parameters to the moderately changing internal and external circumstances until this regulatory capacity is not exhausted.

The above compensatory capabilities influence, not only the manifestation of failures in 1G-CPSs, but also the opportunities and proper techniques of failure analysis and prevention. Due to the compensatory behavior, the use of traditional failure analysis and prevention techniques has become questionable. Indicated by some current strands of research in the literature, further studies are needed in the context of self-regulatory and self-tuning 1G-CPSs and even more in the context of self-aware and self-adaptive second generation CPSs (2G-CPSs). Not considering the latter, this chapter has three main objectives: (i) to obtain a deeper understanding of self-regulation and self-tuning capabilities of CPSs, (ii) to determine the means for analyzing the implications of self-tuning

in failure manifestations, and (iii) To conduct experimental investigations on a practical implementation of the testbed system. The investigation related to the first objective are supposed to cast light on how the control process of CPSs is changing if they can increasingly be adjusted for operation in different varying conditions. This investigation will also provide a deeper understanding of the implications of such system behavior on the recognition and management of failures. The investigation related to the second objective was done to find the most suitable means for exploring the consequences of self-tuning in failure manifestations. Finally, the investigation concerning the third objective seeks to identify the most important technical (functional, architectural and control) requirements for the development of a testbed system that allows replicating self-tuning characteristics of CPS as response to induced failures in a particular application case.

3.1.2 On the phenomenon of self-tuning

The concepts of robust control and adaptive control are well known in the related literature. They are complementing synthesis procedures generally considered for dynamic system and environment interaction. The main objectives of adaptive control are that the controlled system (i) remains stable under varying operational conditions, (ii) has satisfactorily transient response and (iii) does not use too much input energy. Adaptive control assumes that the “control system is uncertain and that the character of the control process may change in time” [2]. Control processes with changing dynamics are often used in various manufacturing processes in which machine tools may be in different operating states. However, these are deterministic, time-invariant, finite-dimensional linear systems. Due to self-intelligence and self-organization, operation of CPSs can be more complicated. They enable reasoning about system status, the surrounding environment, the logistic processes, the work processing and the system interactions, in order to provide the means for decision making and adaptation [3]. Even the limited changes in the operational and architectural parameters of a 1G-CPS that the current technologies enable, reduce the determinism significantly and making decisions that go beyond pre-defined actions is not possible.

While in the case of systems with adaptive control the set point is usually fixed, in the case of self-tuning systems the system may change the set point in a given range of variation. Self-tuning is the enabling mechanism for systems to adjust their operations to varying conditions without functional and structural reconfiguration. It makes possible for a system to respond to disturbances and failures and/or maintaining an optimal system operation under varying conditions. Therefore, self-tuning differs from self-adaptation in the sense that it does not imply new functionalities or system architecture. It is just a limited form of adaptation (which is considered a typical feature of 2G-CPSs). The capability of self-tuning makes it possible for a system to change only certain

predefined parameters (restricted adjustment) in order to assure the needed system behavior [1]. The task of the control mechanism of a self-tuning system is to compute the values of the control variables in an event orientated manner [4]. The control is enabled by the self-regulation capabilities (restricted computational intelligence) of the system.

Self-tuning is an important feature to achieve system stability. A system is considered stable when “for all bounded inputs, all conceivable signals are bounded” [5]. In a stable system, the controlled parameters converge to a specific value, despite the variations caused by external sources. Systems equipped with self-tuning capabilities manipulate system actuators so that they minimize the difference between the reference value and the observed one. However, achieving system stability depends on the fulfilling of controllability and observability properties. Controllability measures the ability of an actuator configuration to control system states [6]. This property determines to what extent the available systems actuators manage to keep system parameters in the expected values, in a finite time interval [7]. Observability, in turn, is the ability of a sensor configuration to provide the information required to estimate the states of the system [6], i.e. to evaluate the controlled parameters. While controllability is the basis for conducting self-tuning, observability is the basis for conducting self-regulation, since it provides the means for evaluating how far the observed measurements are from their target value.

The implementation of self-tuning in 1G-CPSs conveys certain implications in terms of system operation. The system does not present a regular and predictable behavior any more. In the case of Zero Generation CPSs (0G-CPSs), the behavior of the system is the outcome of the preprogrammed operations and any interaction caused change is considered as a bias. This kind of systems, such as embedded systems or manipulative robotic systems, execute certain time-scheduled or event-triggered actions without determining to what extent they meet the objectives of the task. In other words, they do not check if the controlled variables reached their reference values. These types of systems are “isolated” from their surrounding environment, or present only a limited interaction with it. Consequently, the environment does not influence or alter the pre-defined system settings. On the other hand, self-tuning systems continuously modify the system settings. It implies variation not only in terms of the system set points, but among others, also in the operational intensity of actuators and the timing of activation and deactivation of the actuators. It makes them more suitable for working in critical environments and it provides a somewhat higher-level, but still rather restricted autonomy.

3.1.3 Implications of self-tuning for failure analysis

Self-tuning not only affects system performance, but it also influences how failures are manifested and can be detected. Systems featuring the above

presented self-tuning characteristics tend to foster fault tolerance. What it means is that the change of the settings of the active system components (actuators and effectors) made by a system itself prevents to losing stability whenever a failure alters the value of a particular parameter. It allows the system to operate normally, in spite of the presence of a failure, or even multiple interacting failures and to deliver the regular services without noticing any failure symptoms. This situation, on the one hand, is favorable for assuring a continuous system operation and keeping a desired optimal performance but, on the other hand, it has a negative influence on the detectability and prevention of faults and failures. In other words, self-tuning favors an early ‘virtual’ failure management, but its compensatory potential may be exhausted soon. Typically, it can operationalize only some basic response actions whenever changes in the system operations (system parameters) are detected, in order to avoid malfunctioning or sudden collapse of the system. Self-tuning moves the system to a safe state by: (i) modifying the intensity of system operation, (ii) turning the system off, or (iii) activating redundant components. It allows avoiding catastrophic accidents due to system malfunction, as well as the cascade effect over other system components.

Despite the advantages that self-tuning offers for failure analysis, it also brings along important challenges. It is reasonable to think that the constant variation on system settings and system dynamics can lead to different wearing patterns and fatigue on system components. In 0G-CPSs the invariant system operation allows making a reliable estimation of the components life and their time to failure. On the other hand, the continuous changes on system settings based on the owned self-tuning capabilities can lead to a situation where system components operate on the limits of their capacity and to an increased frequency of activation of components. It is widely known that frequent component activation/deactivation reduces the life of components and increases wear and fatigue [8]. In addition to leading to early failures, this also invalidates the currently existing failure prediction models and the methods of estimating the components life.

Thus, any variation on system settings, can also avoid failure detection. The compensatory actions carried out by the feedback control, manage to stabilize the system, but at the same time, it can also affect failure manifestations, by hiding symptoms [9]. This situation may be critical, as failure can continue to progress up, until reaching a critical state, where controllability is no longer possible. It makes the system unable to achieve stability, leading to system collapse. Another implication of systems self-tuning occurrence may be the intermittent failure manifestations. Although certain failure symptoms can be noticed, the fitful failure signs may prevent the process of tracing the failure evolution, making difficult to forecast the time to failure and identifying the occurring failure.

Finally, self-tuning may also lead to false failure alarms, the onset of new failure modes, or modifying priorities in terms of failure management and maintenance actions. Consequently, any variation on working conditions may lead to confuse control actions and external disturbances with failure symptoms, affecting the reliability of failure analysis. Likewise, constant variations on the operation of system's components can lead to the emergence of new failure modes, due to new ways of interaction among those components. Consequently, it leads to re-evaluate the criticality of the potential failure and to update the maintenance strategies. This situation makes it necessary to analyze the effect of implementing self-tuning in failure analysis, so that, we can ensure a proper system operation and to avoid catastrophic consequences.

3.2 Implications of self-tuning induced change of SOMs

The System Operation Mode (SOM) concept is strongly linked to self-tuning. A SOM is a particular set of variables settings that determines the way in which the system will perform; in a time, t . SOMs compile all components' status which, in conjunction, induce a particular system performance. A change on the working conditions may lead control units to execute self-tuning, by activating, deactivating, increasing or decreasing the intensity of system actuators, triggering SOM transitions. It leads us to consider SOMs as the consequence of self-tuning operationalization, or a manifestation of it.

Due to the afore-mentioned reasons, self-tuning can be studied through SOMs. The advantage of considering SOMs is that they:

- Can be evaluated from the aspect of the actual status of system components.
- Allow the analysis of system performance at system level.
- Enable inferring the relationship between subsystems.

Nevertheless, their analysis requires to monitor control signals (or sensing the actuators' status) and system signals (output signals). In the context of 1G-CPSs, control signals are determined by feedback control, based on the comparison of the sensed signals and a pre-defined reference.

3.2.1 Implications of SOMs on failure analytics

Our main purpose in this explorative research is to understand the role of SOMs in failure manifestations. It implies to observe failure symptoms in multiple SOMs and to analyze the effect that SOM transitions exert over symptoms. The literature review presented in the chapter 2 revealed that most of the studied failure analysis methods make use of output signals as failure information

carriers. The faults occurred in open-loop systems, can be recognized based on deviations of the observed system parameters. The existing failure analysis techniques work properly in open loop systems. However, failure manifestations, environmental disturbances, transient system behavior (due to SOM transitions) and the variations that every SOM imply can be masked in signals. Moreover, closed-loop systems (feedback control systems) also mask the effect of faults on signal features, reducing the reliability and accuracy of fault detection and diagnosis methods [10]. This situation emphasizes the need for discriminating among deviations caused by (i) SOM transitions during regular operations, (ii) environmental disturbances, and (iii) failures.

Several approaches for failure analysis use, input and output, signals to evaluate failure detection, as a means for avoiding the masking effect. The comparison between the input signals and the output signals manages to establish if the observed behavior is due to a failure or a self-tuning action. There are several techniques that are effective for determining failure occurrence. For example, one of the approaches conducted for fault detection in closed loops is the ‘Detection of oscillations’. It assumes that fault occurrence will modify the system stability, avoiding or delaying the convergence of the controlled signal. Multiple techniques are based on it, such as ‘cross-correlation analysis’ of input and output signals [9]. Considering that control actions have an influence in the process outputs, this method analyzes their cross-correlation for determining fault occurrence. One of the most common application of this method is the of valve stiction detection [11] [12] [13]. The estimation of the Integral of Absolute Error (IAE) is also another technique for detecting oscillations. This index uses the number of zero crossing of the oscillation signals for determining the limits for computing the oscillation area [9]. A critical threshold about the admissible IAE is used as reference for identifying failure occurrence. IAE index is widely used for analyzing control performance [14] [15]. Some other relevant indicators such as the control deviation, along with the manipulation effort are also reported for analyzing faults in control loops [9]. In this last case, the effort invested in the effectuating of the changes in the output signal is compared to the observed control deviation.

Nevertheless, as these techniques are conducted at component level, they provide information about the component behavior, but it is not possible to study failure effect over other system components. It hampers the analysis of SOMs and their transitions. Moreover, these analyses do not provide information about the type of failure and the effect of external disturbances can be confounded with failure effect.

Lately, Active Fault Detection (AFD) has also been implemented for failure analysis. This method does not just consider the input signals. In order to ease fault diagnosis it also injects auxiliary inputs or modifies existing ones [16]. The injected auxiliary inputs excite specified potential faults allowing fault discrimination. This diagnosis is conducted by evaluating the signature from the

injected inputs in the output vector or residual vector [17], allowing to keep the fault diagnosis part unchanged, after the execution of control actions [18]. Although this method enables determining the occurring failure mode, it is conducted at component level. It implies that in a context of systems of systems, complex failures that are manifested in parallel in several system components or subsystems can be misclassified. Moreover, the analysis of failure propagation and failure progression over the whole system is complex and can lead to wrong conclusions. This same situation occurs with state observers and parity equations (see section 2.4.2), which also implement input and output signals for failure diagnosis.

The failure analysis to be performed in this project, needs to be conducted at system level. It should enable the investigation of failure emergence, their forming process and propagation in order to understand, to what extent SOMs may have an influence in the way in which SOMs are manifested. It should also allow understanding on how failures differ between them and how their discriminating power can be strengthened or diminished due to SOMs transitions. Nevertheless, the above-analyzed techniques are focused only on the component level and their limited scope prevents us to accomplish our objectives.

We consider that the analysis of the whole system signals in parallel (either they are controlled or not) can provide relevant information that allows evaluating failures at system level. The concept of SOMs already implies the use of control signals for the investigation of failure manifestations. We assume that segmenting signals, based on SOM transitions, improves the discrimination of the failure effect on system operation over regular system operation. System signals typically have different characteristic features in each SOM and in transitions between SOMs. These transitions are typically triggered by control actions, external disturbances or failure effects, converting SOMs as primary information carriers of system operation and system behavior. For these reasons, we hypothesize that signal segmentation based on SOM transitions can improve the accuracy and reliability of failure diagnosis and forecasting methods. We claim that this approach will allow understanding the influence of SOMs on signals, when the system is in failure-free operation or in a particular failure mode.

3.2.2 Investigation of system operation modes of cyber-physical systems

Many of the approaches found in literature for failure analysis are based on simulations. We consider simulation as a good mean for providing a controlled environment for experimentation. It is a cost-effective method, which allows conducting early verification of research assumptions without endangering human beings or the environment. However, complex systems such as CPSs

require realistic and practical experimentation with real systems. Their tight interrelation with the surrounding environment, as well as the self-tuning capabilities they are provided with, requires experimentation and testing of CPSs in a real operation context.

It is widely known that it is difficult to find sufficiently controlled data sets about the operation of an existing system under a specific failure mode. On the other hand, injecting failures in existing system may jeopardize system integrity and stability. For these reasons, the accomplishment of the present research project requires an environment in which the self-tuning capabilities of CPSs can be safely evaluated. For this purpose, the development of a self-tuning system as a testbed that really simulates a subset of the full-scale operation of CPSs is needed.

This chapter discusses the development of a testbed that is used for exploring the role of SOMs in failure analysis and forecasting. The testbed has been conceptualized as a conventional greenhouse with Cyber-Physical augmentation. It is implemented as first generation of CPSs (1G-CPS), which can be equipped with self-tuning control and monitoring functionalities. For this purpose, it should provide the means for reproducing different failure modes under controlled conditions.

The target system implements a Cyber-Physical Green-House (CPGH) as testbed. It includes sensors and actuators in order to provide a controlled environment for plants growing. There were three main aspects considered in the testbed design:

- Understanding the functional requirements for a testbed CPGH in order to be able to design the basic functionality of this 1G-CPS.
- Identifying the functional requirements that the cyber-physical augmentation (CPA) should fulfill.
- Determining what type of instrumentation is required for CPA.

These aspects contribute to determine the design features to be fulfilled by our testbed, in order to mimic 1G-CPS operation. The method conducted for eliciting system specifications was literature review.

Having this in mind, we evaluated failure-induced signal deviations by using controlled failure injection in a simulation model, and in a testbed. We studied how SOM causes changes in generic statistical properties of signals (e.g. variation, maximum, minimum, peaks, trends, etc.). As presented earlier, self-regulating and self-tuning systems may compensate the effect of failures by tuning the system operation. This compensation not only improves the resilience of the system against failure, but it may also mask the effect of failures on the information carriers (such as signals and SOMs).

3.3 Requirements for the implementation of a physical testbed system

Cyber-Physical Systems (CPSs) augment the performance of physical processes by providing smart capabilities. The implementation of such augmentation requires a deep understanding of the controlled processes in order to determine the requirements of the system. This section aims to explore the main characteristics of both: (i) the processes related to greenhouse operation and (ii) the functions and services to be provided by a first-generation cyber-physical system (1G-CPS).

As seen in Figure 3.1, traditional systems meet a set of requirements, denoted RTS (Requirements for Traditional Systems). Likewise, the augmented characteristics of a 1G-CPS, should meet a set of Requirements for Cyber-Physical Systems (RCPS). The following subsections will explore those requirements to derive the list of functions to be fulfilled by our testbed.

3.3.1 Functional requirements for traditional systems: The greenhouse testbed perspective

Theoretically, greenhouses are controlled environments that seek to maintain

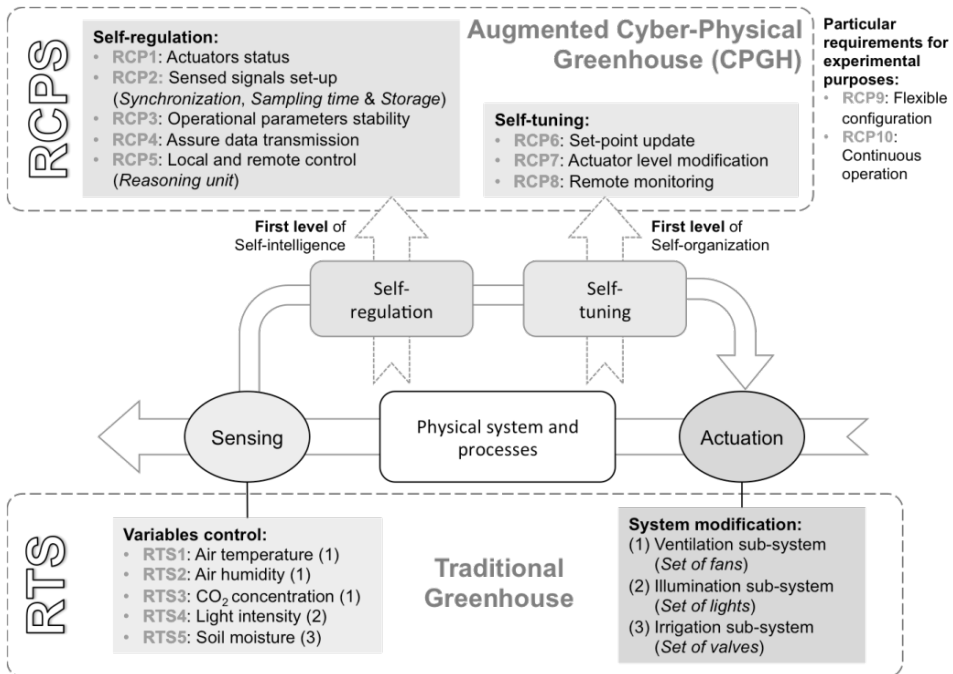


Figure 3.1. Requirements for traditional systems and augmented requirements for cyber-physical systems in the case of 1G-CPS

optimal climate conditions for crops [19]. They implement a non-linear multivariable process that is influenced by several biological processes [20]. Our cyber-physical testbed is a scaled greenhouse system that emulates real conditions of a fully operational greenhouse system. Greenhouse operation requires controlling several interrelated variables (such as temperature, soil humidity, among others), which are sensitive to environmental changes. Together with a set of tasks to be accomplished by the system (e.g. irrigation, maintaining temperature and humidity), these changes necessitate self-tuning capabilities, which are supposed to provide an optimal environment for plant growth despite varying climate conditions. Due to that, a first-generation cyber-physical greenhouse (1G-CPGH) testbed is a suitable mean for conducting practical experimentation. The testbed implements a set of plants, influenced by parameters such as light, air humidity, water, air temperature and soil humidity, which should be controlled and kept on the desired levels [8]. These parameters, along with CO₂ concentration, are non-linear and interdependent [20], requiring the continuous adjustment of greenhouse variables.

The main objective of greenhouse is to provide the required conditions for photosynthesis, transpiration, respiration and cell division in plants [8]. We analyzed literature with the goal of identifying means for facilitating these biological processes, so that we could derive technical requirements from the perspective of greenhouses. Consequently, the most relevant ‘Requirements from Traditional Systems’ (RTS) comes from the “sensing” component of a physical system, and their related “actuation” components: From the “sensing” component, variables to be controlled need to be identified. The related main RTSs are listed below:

RTS1 The system should control the air temperature

Air temperature has an important effect on plant growth [21], as it affects photosynthesis and respiration [22]. Extreme temperatures (either low or high) negatively affect the growth and development of plants. Regular plant processes occur between 0 °C-40 °C [23]. However, every single type of crop has a different optimal temperature. For example, for tomato, 30 °C is the optimal temperature. Temperature is usually regulated, either by heating up or cooling down the greenhouse air. The most common cause of temperature increase is radiation, for which cooling is required [24]. There are different types of systems reported in literature, which are used to cool down the air within the greenhouse, for instance: (i) natural ventilation, (ii) evaporative cooling, (iii) shading, (iv) fan-pad system and (v) spray cooling system [25] [26] [24]. Whenever heating up of the greenhouse air is required, a heat generator, such as radiators, are typically used [8].

RTS2 The system should control the relative air humidity

Air humidity is another critical variable that influences plant growth. It is mainly determined by air temperature, plant transpiration and water

evaporation from the soil [24]. It is strongly related to air temperature. Increase of air temperature causes decrease of relative humidity and vice versa [27]. Plant transpiration increases as air humidity decreases [28]. It causes, in turn, an increase on water uptake. Due to its strong dependence on temperature, humidity control is usually achieved by adjustment of the temperature. Keeping humidity below the dew point allows avoiding condensation [29], as it may lead to diseases and pests in the plant.

RTS3 The system should maintain the CO₂ concentration in a predefined range

CO₂ concentration is an important parameter that influences photosynthesis, which provides the energy required for cell division [8]. Regulation of the CO₂ concentration and achieving the appropriate CO₂ levels is required. This parameter is controlled through: (i) ventilation, for evacuating the excess of CO₂ and (ii) enrichment, for increasing CO₂ concentration (by liquid CO₂ or combustion of gas natural). CO₂ enrichment is not very common in mild weathers, though. Due to that, a combined control of ventilation and CO₂ enrichment are usually implemented [30]. It contributes to maximize plant growth. A CO₂ control system aims to keep CO₂ levels as in the outside level, or above. In the outside air, the CO₂ concentration is around 340 ppm. However, plants grow faster with values are between 340 and 1000 ppm.

RTS4 The system should control the light intensity (by letting natural light to pass)

Light is directly related with photosynthesis, thus radiation of light is essential for plant production [31]. Red light contributes to increase CO₂ absorption and, therefore, stimulates plant growth [8]. Under this consideration, greenhouses are built and oriented so as to maximize light penetration [32]. The implementation of transparent cover that allows natural light to enter into the system is, thus, required. However, artificial light systems, that improve the light levels in the greenhouse, can also be installed. Light between 400-700 nm is advised to foster photosynthesis.

RTS5 The system should control soil moisture by providing the water supply required for plants

Water is one of the most important supplies for plants. It intervenes in all crops growing processes, i.e. respiration, transpiration, photosynthesis and cell division. It provides the nutrients required for cell division and cell elongation [8]. Therefore, water irrigation systems are one of the vital systems of the greenhouse. They do not only provide the required water, but they are also used to distribute fertilizers. Currently, irrigation processes are conducted based on a fix schedule, which does not take into account the actual needs of the crops. The measurement of soil moisture

can be used to control the irrigation and to avoid water stress in plants. This stress is caused because of limited water supply or due to intense water transpiration [33]. High levels of soil humidity can also cause saturation. Depending on the soil type, soil moisture should be kept around 2% to 10% for sandy soil and between 40% and 60% for clay soil.

From the “Actuation” component, the set of subsystems, that are able to perform system modifications, will contribute to the “control” process of the aforementioned requirements (air temperature, air humidity, CO₂ concentration, light intensity and soil moisture). Therefore, ventilation sub-system (composed by the set of fans), illumination sub-system (composed by the set of lights) and irrigation sub-system (composed by the set of valves) are the main sub-systems to be integrated in the regular greenhouse. In these types of systems, air temperature, air humidity and CO₂ concentration are controlled through the ventilation sub-system; light intensity is influenced by the illumination sub-system, and the irrigation sub-system influences the soil moisture.

3.3.2 Functional requirements for cyber-physical systems

We already analyzed the main requirements in a traditional greenhouse. We also listed the main variables that are monitored and controlled in these types of environments, as well as the sub-systems that influence their control. However, in order to analyze the role of SOM in failure manifestation in 1G-CPS, it is important to provide the greenhouse testbed with self-regulation and self-tuning capabilities. This may offer an environment that enable to reproduce CPS behavior. Our experimentation process required the ability:

- Induce several failure modes in the system.
- Monitor system signals and actuator status.
- Evaluate the magnitude of failures and their progress over the time.

Failure injection (i) is required in order to reproduce and replicate failures in the system, in a systematic and controlled way, as it is widely known that getting statistically significant number of failures, without intervention, makes the analysis impractical [34]. Monitoring actuator status allows determining the time instants in which SOM transitions occur to enable SOM-based signal segmentation. Monitoring system signals allows evaluating failure effect on every signal segment; and the analysis of failures throughout the time seeks to provide insights about failure manifestations and their forming process.

In the following paragraphs, we will analyze the requirements (RCPS) that should be fulfilled by a system in order to be considered a 1G-CPS. We will also analyze a set of particular requirements, for experimental purposes. From the aspect of “self-regulation”, the RCPS are listed below:

RCPS1 The testbed system should be able to monitor actuator status

Actuator status refers to the component operation mode in which a particular actuator present in a time t . It can be represented in a numerical or categorical way. Numerical indicates the intensity of operation of the analyzed actuator, based on the sensed parameters that describe it. For instance, the speed of rotation of the fan is $v = 1000 \text{ RPM}$. Categorical reports semantically describe actuator operation based on bounded regions. For instance, the use of $\{\text{low, medium, high}\}$, to describe fan speed. Into the categorical report we can consider binary reports. Binary reports inform if the analyzed actuator is either active or inactive, no matter its intensity. E.g. the use of $\{\text{On, off}\}$ to describe the fan status. The implementation of sensors for indicating actuator status is required in order to track SOM transitions. In this explorative research, we will focus on binary reports. It allows evaluating the effect of turning on or turning off system actuators in the whole operation of the system. It also provides a basis for a further study that allows evaluating variations on actuators intensity.

RCPS2 The (sensed) system set-up must be defined.

Our main interest for conducting experimentation is to explore failure manifestations through signals. Signals are generated by sensors that measure the parameter values of physical phenomena [35]. It was already determined that the main variables to be measured in traditional greenhouses are CO_2 concentration, air humidity, air temperature, light intensity and soil moisture. It was also stated that all actuator status should be monitored throughout time in our testbed. However, some properties from sensed signals needs to be standardized in the testbed implementation, in order to enable better self-regulation. Those properties are:

RCPS2.1 The (sensed) system signals should be synchronized by the system:

SOMs-based signal segmentation requires the synchronization of the actuator signals and sensor signals in time. It allows the analysis of multiple signals, which were concurrently measured during the same SOM.

RCPS2.2 The (sensed) system signals should be sampled at specific moments in time:

A frequent signal sampling provides more detail about events that occur with a short duration. Considering we are interested in segmenting signals based on SOM transitions, the data should capture the moment of transition. Small sampling frequency reduces the certainty of capturing the moment of SOM transition. Moreover, SOMs with durations smaller than the sampling frequency are not detected. Similar situation occurs

with failure symptoms, particularly with transient faults. Transient faults are those which occur for a short time period and which can reappear later [36]. If transient faults occur between sampling times, they will not be detected. This incompleteness of data about system operation may negatively affect our study, as it would bias the observation of failure manifestations. We consider a sampling frequency of one second suffices for our analysis, as changes on greenhouse variables occur gradually (it does not present sudden changes where a smaller sampling time is needed). A lower sampling time would lead to unnecessary amount of data that would slow its processing.

RCPS2.3 The (sensed) system signals should be stored and they need to be accessible:

System signals (including both, sensor signals and actuator signals) are the main input for our analysis. They convey information about failure manifestations and SOM occurrence from which we will derive knowledge about the studied phenomenon. Due to that, a permanent storage of the signals is required, as well as an easy access to the information. Arranging data into a single table representation and storing them in a standard format (such as ‘.csv’ or ‘.txt’) is also needed. It allows its use in advanced software tools such as Matlab, R, Python, among others. Data arrays should include the date and time records in which the signals were sensed.

RCPS3 The testbed system should keep the values of the operational parameters stable even under uncertainty

Controllers enable system stability. They make it possible for the system to conduct self-tuning and to meet the defined set points through setting the operational parameters. CPSs must adopt feedback control system [37]. They provide robustness and stable control even if the inputs present noise and disturbances [38]. Feedback control compare the actual output of the system (i.e. the measured parameter) with its corresponding set-point and uses the difference as a mean of control [39]. A stable system operation is paramount for the success of our experiments, as it avoids signal fluctuations caused by non-managed disturbances that can be confounded with failures.

RCPS4 The testbed system should assure data transmission between the local and remote processing units

Central processing unit further processes data sent through the sensor network using standard data transmission technologies. Our testbed is located in the campus of Universidad EAFIT in an open area that is around 12 meters of distance from the central data processing unit. The transmission devices implemented should provide reliable communication

between these systems, so that, there is no loss of data samples. From that perspective, the (a) communication data range (kbits/s), (b) coverage (Km), (c) transmission power (mW), (d) application area, (e) data transmission protocol and (f) network topology should be designed.

RCPS5 The testbed system should be able to combine local and remote control

As it was already explained in section 1.3, CPSs are systems of systems. Smart-sensors enable local processing through embedded microcontrollers [40] in the controlled core area (see 1.3). However, these types of components have low memory and limited capacity [41] (such as CPU speed) which makes it impossible to run advanced algorithms. For this reason, high-level systems with high processing capabilities are implemented in the extended field of application. These systems conduct complex decision making by considering data coming from the controlled core area in order to execute coordination activities and data filtering. In our testbed, it is required to include a processing unit that is connected to the local controllers that manage greenhouse processes. Nevertheless, such units are highly susceptible to damage caused by greenhouse environment [41]. Remote unit location is needed, as it provides a cost-effective solution for avoiding the damages that its placement at the greenhouse would imply. It also prevents interruptions during experimentation, ease the manipulation of system variables and the access to the sensed data.

From the “self-tuning” aspect, the RCPS are listed below:

RCPS6 The system allows updating variables set-points, according to the operating conditions

One of the enablers of self-tuning is the possibility of modifying system settings in run-time, as a response to any particular condition that can be established based on performance needs. It allows the system to present different behaviors for different operating scenarios, so that it can conduct an optimal resource management. These set-points respond to a pre-defined (or learned) set of rules that determine the most suitable settings, according to the working conditions.

RCPS7 The system modifies the actuators’ status autonomously

The modification of system status aims to keep system variables on the desired levels, so that we can assure a stable and optimal system operation. Whenever a controlled variable exceeds its ideal limits, system actuators can manage to return it to the pre-defined levels by modifying the actuator status, i.e. by activating/deactivating system actuator or by modifying its operational intensity. The modification of the actuator status requires the actuator to be embedded in a feedback-based control system, so that it can

be manipulated autonomously, based on the measurements of the sensed variables.

RCPS8 The testbed system should enable remote monitoring

The proposed system should be provided with a user interface that allows performing manipulation of the system's set points, as well as the experimentation setup. It allows the user to monitor the system operation in real-time and to respond to actions triggered from a remote location. This capability is desired to avoid interruptions during experimentation, whenever it is required adjusting system parameter, or checking system condition.

Finally, requirements related to our experimental purposes are listed below:

RCPS9 The testbed system should have a flexible configuration that allows conducting large variations of experiments

Experimentation requires controlled adjustments to sensors and crops setups in the greenhouse testbed. Switching from one experiment to another may require system manipulation. Re-allocation of components or the intervention of certain system actuators might be needed. It makes it required a flexible system configuration that allows performing such changes easily.

RCPS10 The testbed system should assure continuous system operation during the experiments

This requirement is very important for our testbed. The experiments to be conducted in the testbed may last long time. The use of batteries in wireless operation may limit system autonomy, preventing the execution of tests for extended periods. For this reason, continuous energy supply is critical for assuring the implementation of the required experiments.

We have already stated the main requirements for the experimental analysis. In the following sections, we will determine the design characteristics of the system, based on the above-presented requirements.

3.4 Architecture of the testbed system

3.4.1 Initial considerations

In Section 1.3, we have introduced the main characteristics of CPSs, which are influential for our work. It was discussed that CPSs are composed by three main layers namely [42]: Controlled Core Area (CCA), Extended Field of Application (EFA), and Cross Domain Networking (CDN). These three layers describe a hierarchical system where, controlled core area is composed by

embedded systems that accomplish simple control tasks, the extended field of application coordinates the embedded systems and takes high-level decisions (such as system performance optimization), and cross domain networking, enables the interaction with external systems. In Chapter 1, we stated that the focus of the conducted research is on 1G-CPSs that are characterized by their self-regulation and self-tuning properties. We narrowed down the context of the research to a greenhouse application in Sub-chapter 3.3. The testbed we are proposing corresponds to a first-generation cyber-physical greenhouse (CPGH). This testbed includes a complete implementation of controlled core area and extended field of application and a partial implementation of the cross-domain networking. In the cross-domain networking, only system performance monitoring was enabled, but the interaction between the system and external systems was not implemented.

In the context of 1G-CPSs, the interaction between the CCA and EFA layers provides the necessary conditions for the testbed operation. In the case of our CPGH, CCA allows sensing the system's parameters (light, air humidity, water, air temperature and soil humidity), conducting decision based on monitoring the states of conventional controls (on/off, PID, artificial intelligence-based) [41] and performing response actions by the activation and deactivation of actuators (lights, fan, irrigation valves, etc.). EFA, in turn, collects data coming from the CCA, store the sensed signals, and allows the monitoring of system operation. EFA is provided with power computation capabilities that enable pre-processing signals (for signal filtering) and executing algorithms that allow signal segmentation, pattern recognition, failure diagnosis and failure forecasting. CCA and EFA provide the basis for the design of the proposed first generation CPGH testbed. In this section, we will describe the configuration of both layers, so that, regular operation can be performed. The following subsection will explain in detail the main subsystems and components of the physical testbed. Figure 3.2, graphically illustrates the considered architecture.

3.4.2 Details on the testbed system architecture

The performance of the regular system functions (irrigation, lighting, temperature and humidity control and CO₂ control) of the proposed testbed is possible through the implementation of three main control units: a greenhouse unit and two plant-bed units. They constitute the CCA. A reasoning unit is also considered. It constitutes the EFA layer. The greenhouse unit and the plant-bed units are located inside the greenhouse. Each of the system's units is composed by elements belonging to the physical and cyber dimensions. Such a unit's composition is deeply explained in the following paragraphs.

Greenhouse unit

The greenhouse unit has two main tasks: i) controlling the environmental conditions inside the greenhouse and ii) ensuring water supply, as well as

mixing water with the additives required by plants. For these purposes, the physical layer is instrumented with one CO₂ sensor, one air temperature sensor, one humidity sensor, one water temperature sensor and one water level sensor. These sensors allow measuring CO₂, temperature, air humidity and water level (in the rain-water reservoir), respectively. Water temperature and water level were added to the set of variables identified in Section 3.3.1, as a water reservoir was included in the design. The water reservoir aims to enable the use of fertilizers and a better water supply. The sensors should provide accurate information for activating and deactivating the electronic valves of the water irrigation subsystem and the two fans that control air temperature, air humidity and CO₂ concentration.

All the considered sensors are connected to a controller that make use of the sensed data for determining control actions and a XBee module for transmitting the sensed data. It is important to note that the control actions performed by this unit are locally conducted. For this purpose, the sensor nodes and transmission units are integrated through the implementation of Arduino. Arduino are modular embedded systems, which supports prototyping of electronic systems.

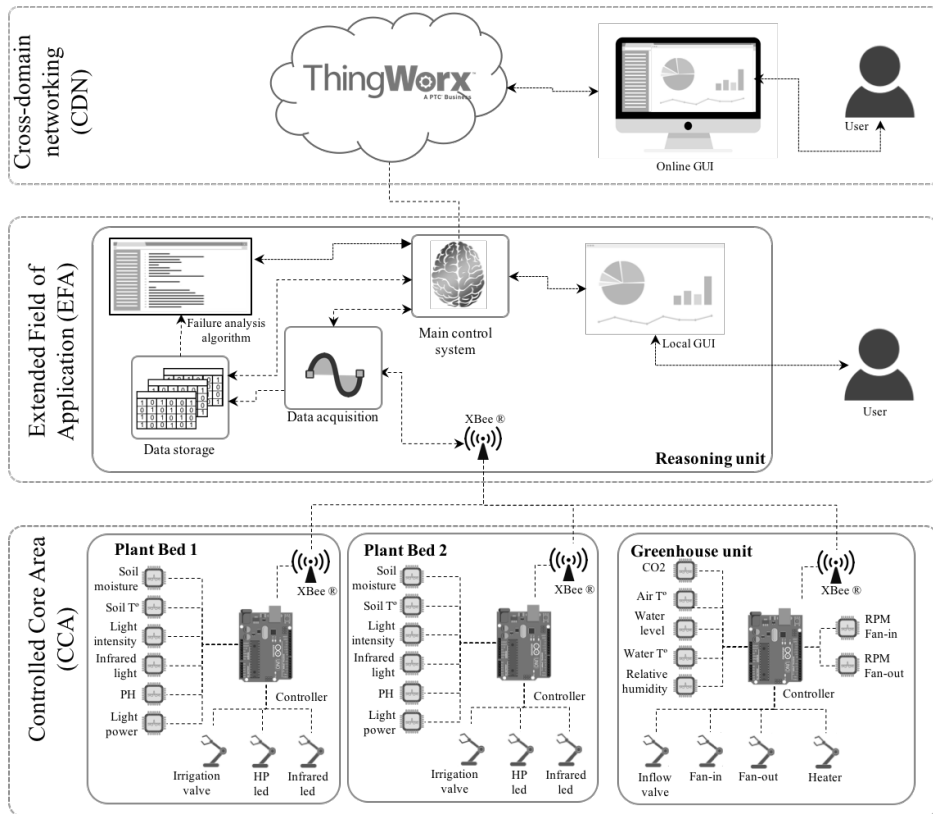


Figure 3.2. Architecture of the proposed testbed

This characteristic is widely desirable in our context, as we are proposing the development of a testbed for conducting experimentation in a first generation of CPSs. Considering the presence of noise and disturbances during sensing the signals, a digital filter in the controller is implemented, in order to smooth data before delivering it to the processing unit. For this purpose, every sensor collects 40 samples per second, which are used as basis for the digital filter. The measured samples are averaged to get one single sample per second, per signal. An overview of the components included in this unit is shown in Table 3.1.

Plant-bed units

The plant-bed units have two functions: (i) dosing water with fertilizers for plant irrigation and (ii) controlling the amount of light required by the plants. There are two plant-bed units installed in the greenhouse testbed, which are controlled independently. Each of these units are instrumented with one soil moisture sensor, one soil temperature sensor, two light sensors (one for white light and

Table 3.1. Description of the components of the greenhouse unit

Element	Type	Specifications	Purpose
Controller	Micro-controller	CPU: 16 MHz RAM: 8kB	Controls operation in this unit
xBee module	ZigBee enabler	RF Data Rate: 10Kbps Indoor Range: 370m Transmit power: 250mW	Transmits/receives data to/from reasoning units
Electro valves	Actuator	On/Off	Injects additives and water into reserve tank
Fan-in	Actuator	On/Off	injects air into the system
Fan-out	Actuator	On/Off	Removes the hot air of the system
CO2 Sensor	Sensor	Analog	Reads CO2 level inside greenhouse
Temperature sensor	Sensor	Analog	Reads temperature inside greenhouse
Heater	Actuator	On/Off	Increases temperature inside greenhouse
Water level sensor	Sensor	Analog	Measures the amount of water present in reserve tank
Humidity sensor	Sensor	Analog	Reads air humidity

another one for Photosynthetic Active Radiation (PAR) light) and one PH sensor. Soil temperature and PH sensor were considered in order to provide complementary information for analyzing failure effect on the system. These two signals do not have control implication though. Soil moisture sensor provides the information required for activating and deactivating one electro valve that allows plant irrigation and the light sensors allow controlling two lamps per plant bed: white light (for maintenance purposes) and PAR (for fostering crop growth). As for the greenhouse unit, all sensors are connected to a controller (Arduino) and a XBee module. It allows conducting local control and transmitting data to the reasoning unit. They also implement a digital filter for noise reduction on the signal. A summary of the components included in this unit is presented in Table 3.2, while the proposed system is shown in Figure 3.2.

The reasoning unit

The reasoning unit performs four primary functions: (i) data acquisition, (ii) data storage and (iii) main control function and (iv) data analytics. Data acquisition imports system signals from the greenhouse and plant-bed units and synchronize them at the reasoning unit. Signals are sampled and transmitted on demand. The reasoning unit request data from the greenhouse and plant-bed units and synchronize them. The obtained data is put together and arranged so that, it can be either locally stored or displayed in the user interface (GUI). All the functions of the reasoning unit are supported on the main control function. Main control function, is in charge of managing system resources, executing the algorithms related to this unit (algorithms for data acquisition and arrangement, signals pre-processing and failure analysis) and to connect the EFA with the CDN through internet. The connection between the main control function and the GUI is duplex. It allows graphically delivering system performance data and overriding the automatic control of the testbed in a remote way. A deeper description of the processes conducted at this unit is presented in section 3.5.

Our failure analysis algorithm is augmenting the control unit. It takes the stored system signals, which are available in a “.txt” format and processes them in Matlab. Matlab was selected, as it provides a set of pre-defined libraries that enables data analysis. Up to now, the failure analysis algorithm is built with explorative purposes, so that it does not influence the system control. A deeper explanation of the failure analysis algorithm is presented in Section 4.3.3. Nevertheless, this analysis constitutes the core of this research. All the infrastructure described here was built in order to provide the inputs for running the failure analysis.

Remote access: ‘Internet of Things’ module

Internet enables the partial implementation of the cross-domain networking. It provides controlled access to other systems, applications and resources that are available online. CPSs can interact with other CPSs through Internet

Table 3.2. Description of the components of the Plant bed unit

Element	Type	Specifications	Purpose
Controller	Micro-controller	CPU: 16 MHz RAM: 2kB	Controls operations in unit
XBee module	ZigBee enabler	RF Data Rate: 10Kbps Indoor Range: 610m Transmission power: 250mW	Transmits/ receives data to/from reasoning unit
Electro valve	Actuator	On/Off	Doses water for irrigation
Soil moisture sensor	Sensor	Analog	Reads soil moisture
PH sensor	Sensor	Analog	Reads pH level on soil
Light sensor	Sensor	Analog	Reads regular light levels in each Plant bed
UV light sensor	Sensor	Analog	Reads UV light level in each Plant bed
PAR light sensor	Sensor	Analog	Reads PAR levels in each plant bed
High-power led	Actuator	Analog	Provides light to each plant bed
PAR light	Actuator	Analog	Provides red light to the plant bed
Light power consumption	Sensor	Analog	Reports the power consumed for the activation of the High-power leds

complementing their operation by using information coming from interrelated systems, in order to support their decision-making. CDN layer was included in the design of our testbed. For this purpose, we used Thingworx IoT® platform developed by PTC. This implementation was only exclusively used as a user interface that enables collecting data from different sources, remote monitoring

and modifying system set points. A screen capture of the visualization of system parameters performance is presented in Figure 3.3.

3.4.3 Description of the placement of the components

In this project, we will focus on the analysis of system data which is originated at the control core area layer. Due to that, plant-bed units and greenhouse unit are discussed in detail. The aspects of reasoning and transmission will be briefly discussed in this chapter. In terms of hardware, the proposed testbed can be divided in two major parts: (i) the household and (ii) the water reservoir. We make this distinction, as they are located separately. The categorization presented in 3.4.2 was based on the control units (functional categorization), here, we will just describe the physical location of the components. The hardware that composes the reasoning is also described in this section.

The household aims to provide the artificial environment required for the plant growth; and the water reservoir is in charge of storing, heating up the water and dissolving the additives used as fertilizers. The household and the water reservoir are placed next to each other. Both of them were located into a bigger greenhouse with the aim to protect the electronic components from the rain. The testbed was placed in the campus of Universidad EAFIT in Medellin, Colombia. This city does not present major weather variations throughout the year, due to its proximity to the Equator. It is located 1500 m over the sea level, its temperature ranges between 16 and 35°C, it has a rainfall of 1685 mm per year and an average solar radiation of 4350 Wh/m². Medellin is characterized by strong tropical storms that could damage our testbed. The instrumented testbed is presented in Figure 3.4.

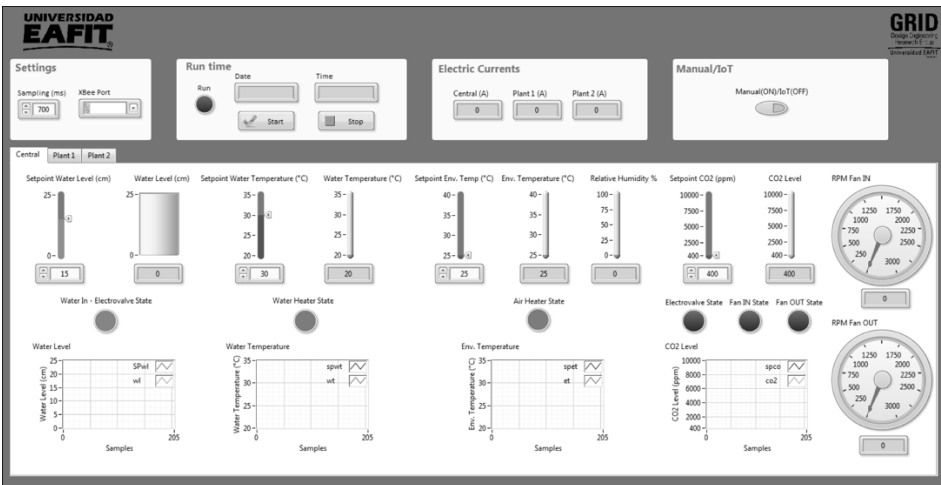


Figure 3.3. User interface



Figure 3.4. Introducing the testbed

Household description

The household of the testbed is made by a tubular structure which houses the plant beds (with their corresponding sensors and actuators) and the controllers.

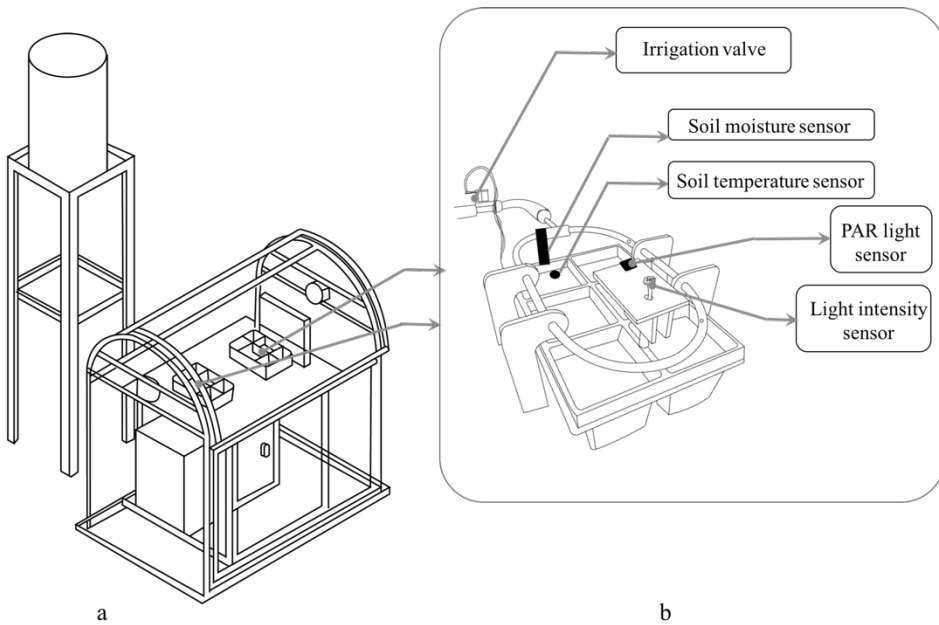


Figure 3.5. Placement of the plant bed units. (a) Testbed system, (b) Description of the elements

The plant beds are located next to each other. However, each of them is equipped with an independent irrigation and light control. Irrigation needs are determined by a capacitive soil moisture sensor (SEN0193), which was placed in the first cube of each plant bed, as it is shown in Figure 3.5. Although this sensor was only installed in one of the 6 cubes that compose every plant bed, its measurements were used as reference for irrigating all plant cubes. The soil temperature was also measured in the same cube through an analog temperature sensor (LM35). Two solenoid valves (EV12V-NC-1/2) were installed at the gate of the water reservoir (one per plant bed). Each solenoid is controlled by its corresponding plant bed control, so that, irrigation can be performed independently.

Light requirements were determined based on a light dependent resistor (LDR10K12mm) and an PAR sensor (LI-190R) which are installed in the center of every plant bed, as it is shown in Figure 3.5. Measurements of the light sensor were used as reference signal for the activation of six led lights. These lights are located at the left lamp structure presented in Figure 3.6 and they aim to ease the visibility for the researchers during the night time. Likewise, measurements from the PAR light sensor activates six PAR led lamps (SMD 5050 led grow light) located at the right side of the same lamp structure. These emitters are used for facilitating plant growth. Every plant bed is equipped with its own set of lamps, which is controlled independently.

Two analog PH sensors (SEN0169) were also installed in the household. These sensors aim to determine the PH levels of the water in every plant bed. These were located at the bottom of the structure in a water collecting box (see Figure 3.7). A tube-shaped channel was installed (one per plant bed), to address the

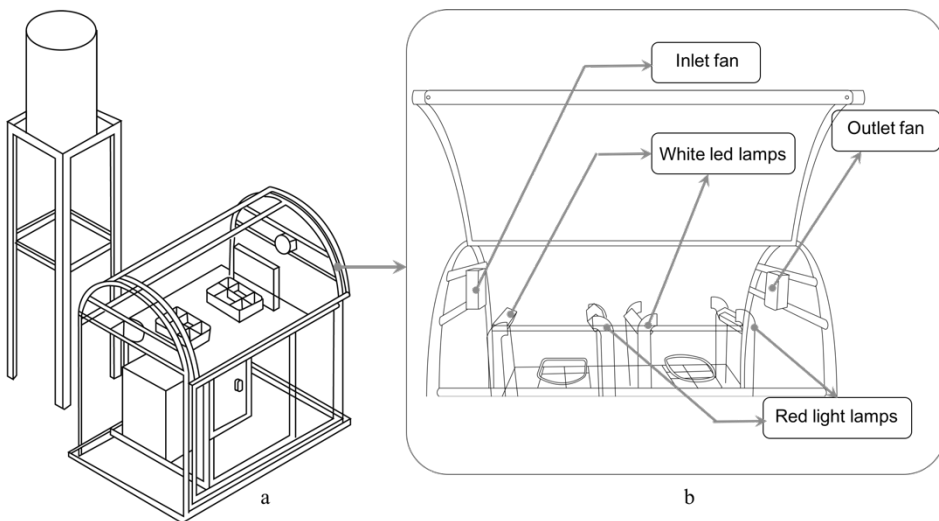


Figure 3.6. Placement of the top of the household. (a) Testbed system, (b) Description of the elements

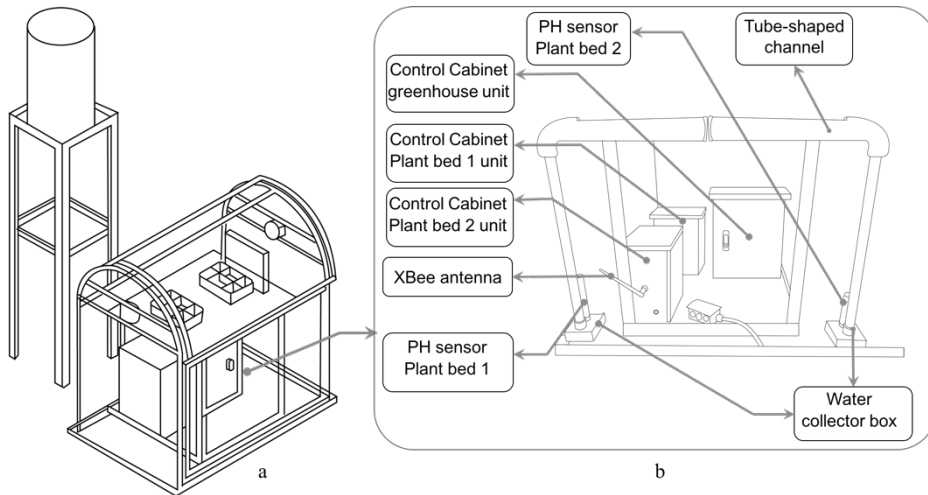


Figure 3.7. Placement of the low part of the household. (a) Testbed system, (b) Description of the elements

residual water (from irrigation) to the water collecting box. A slit releases the excess of water in the box. PH measurements are not used with control purposes.

The household also contains an air temperature and air humidity sensors that are integrated into the same module (SEN0137). These sensors are located between the two plant beds, just in the middle of the household, along with an analog CO₂ gas sensor (SEN0159) that measures the CO₂ concentration in the cabin. Controller determines the activation of either the air inlet fan (Fan-in) or the air outlet fan (Fan-out), based on the CO₂ measurements and air temperature. Both fans are located at the opposite side walls of the household and they operate in shifts, i.e. they are not supposed to work in parallel. The specific fans installed are DC axial fans (F602).

The three control units included in the testbed are located inside the household. These were placed in the bottom of the structure, under the plant beds, so that, they were isolated from the water used for irrigation. Three metal cabinets, one per system unit, contains the controllers corresponding to the plant-bed units and the greenhouse unit. It also includes their corresponding XBee modules (XBee PRO S3B). However, their antenna was placed out of the box in order to avoid communication problems. Arduino Uno was selected for the plant-bed units and Arduino Mega was selected for the Greenhouse unit

Water reservoir description

The water reservoir included in our testbed was placed over a structure made of iron that aims to generate a height difference between the household and the reservoir (see Figure 3.8). It enables the water flow for irrigation, without the

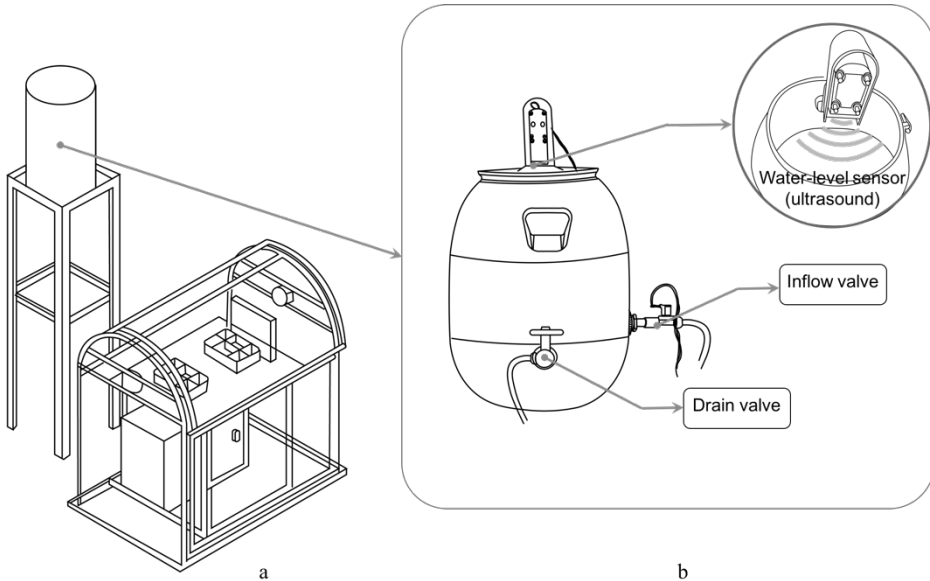


Figure 3.8. Placement of the tank. (a) Testbed system, (b) Description of the elements

need of a pump. This reservoir was equipped with one ultrasonic sensor (RU100-CP40-LIUX) that allows measuring the water level in the tank. This sensor was located on the top of the tank, placed in a box made of transparent acrylic. A solenoid valve (EV12V-NC-1/2), was also installed in one of the walls of the tank, in order to fill the tank. This valve is activated based on the measurements of the water level sensor. A second valve was also installed in the back of the reservoir with the purpose of draining and for measuring tank leak. However, this last valve is activated manually by the user.

For water heating, one thermocouple type K was installed in the tank. It facilitates the measurement of the water temperature in the reservoir. It was connected to a Maxthermo (MC-5738) unit, which allowed controlling water temperature through the manipulation of a heating resistor of 110V. This heater was located in the bottom of the tank, in order to assure it is always immersed in water. The tank is a plastic basket.

Reasoning unit description

The reasoning unit consists of two main components, a XBee module (XBee PRO S3B) that works as Master and a laptop (MacBook Pro13, 8Gb, 2.9 GHz Intel Core i7) that structures, stores and processes the obtained data. Labview was used for receiving data, synchronizing it and structuring it into a single table representation. It provides the means for reprogramming the control of the system and for graphically delivering the data used for monitoring the testbed

performance. The reasoning unit is located at a 12 meters distance from the testbed. Matlab was also used to pre-processing the obtained data and running the failure analysis algorithm presented in section 4.3.3.

3.5 Greenhouse information analysis

3.5.1 Operation flow of the testbed system

In this PhD research, the reasoning unit was used for visualization purposes, data management, changing system set-points and for the execution of the failure analysis algorithms. As it was already stated before, greenhouse signals are sampled on demand. It is the reasoning unit, which sends a sampling request to the greenhouse and plant-bed sensing units. All the considered system units report both the sensed greenhouse parameters and the actuator status. This process is conducted with frequency of the data sampling, i.e. 1 sample/second. In our experiment, the failure analysis algorithm was executed offline on the collected data samples. Offline execution of the failure analysis and forecasting facilitates our explorative research by enabling systematic analysis and comparison of alternative data processing, pattern recognition and forecasting methods on the same data. In the next paragraphs, we will explain the operation flow of the three units that compose our testbed.

Every plant-bed unit monitors the soil moisture, soil temperature, light intensity, PAR light, PH of the irrigation water and the light power consumption signals in parallel. Not all signals are used for control, though. Only soil moisture, light intensity and PAR light are controlled. Whenever the soil moisture presents measurements below 90%, the irrigation valve is opened until it reaches 90%. A dead zone of 3% was considered in order to avoid erratic control. The status of the irrigation valve is also monitored every second, taking as basis the control signals. Sensed data corresponding to light intensity is also evaluated in every plant-bed unit. There, led lights increase their intensity when the sensors report measurements below 500 lm. It is also the case of the PAR light which, uses a set point of 675nm as reference. As for the irrigation valve, the intensity of the white led and IR led are reported every second.

The non-controlled signals in the plant-bed units, namely, soil temperature, PH and light power consumption are sampled for complement the failure analysis by external conditions influencing the experiments. Nevertheless, none actuator is manipulated with regards to keep them in a pre-defined value. All signals are reported at the same time and are sent to the reasoning unit via XBee, so that, these can be arranged and processed. Figure 3.9 presents the operation flow of the plant-bed units. It is important to remark that the system is composed by two plant-beds which are independent one from each other.

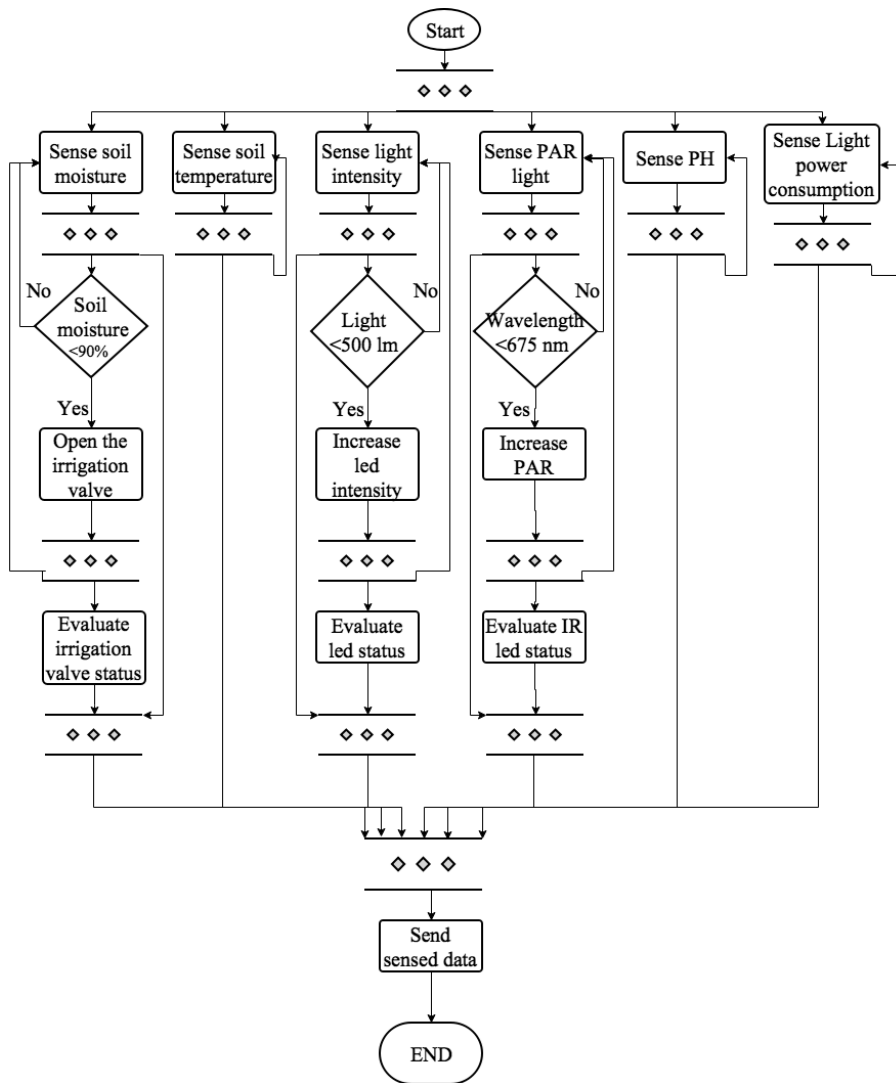


Figure 3.9. Flowchart of the plant-bed units

The greenhouse unit senses in parallel the CO_2 concentration, air temperature, water level, water temperature, air humidity and rotation speed of fan-in and fan-out. As for the plant-bed units, not all the sensed signals are controlled. Only CO_2 concentration, air temperature, water temperature and water level are controlled. CO_2 and air temperature are controlled together.

Whenever CO_2 levels are lower than 600 ppm or the air temperature is above 30°C the fan-in is activated and the fan-out is turned off. Otherwise, the fan-out is activated and the fan-in is turned off. In like manner, the water level in the water reservoir has a set point of 20 cm. If the water level is below this

reference the inflow valve is activated, up to reach the set point. A dead zone of 2 cm was considered to avoid erratic control. Water temperature is also controlled. It has a set-point of 25°C, so that, the heater is activated if water temperature is below such temperature. A tolerance of 1°C was considered for this variable.

All the sensed signals (those that are controlled and those that are only sensed) are sent together to the reasoning unit, along with the reports of the manipulated actuators (fans and inflow valve). Figure 3.10 presents the operation flows of the greenhouse unit. All the control processes implemented feedback control in the greenhouse unit and the plant-bed units. It enables self-regulation and self-tuning of the greenhouse testbed.

The reasoning unit processes the signals retrieved from the testbed in X steps. Signals coming from the greenhouse and plant-bed units are requested by the reasoning unit. There, these are synchronized and stored in a structured array, in which actuator signals (S_A) and sensor signals (S_S) are discriminated. The rows

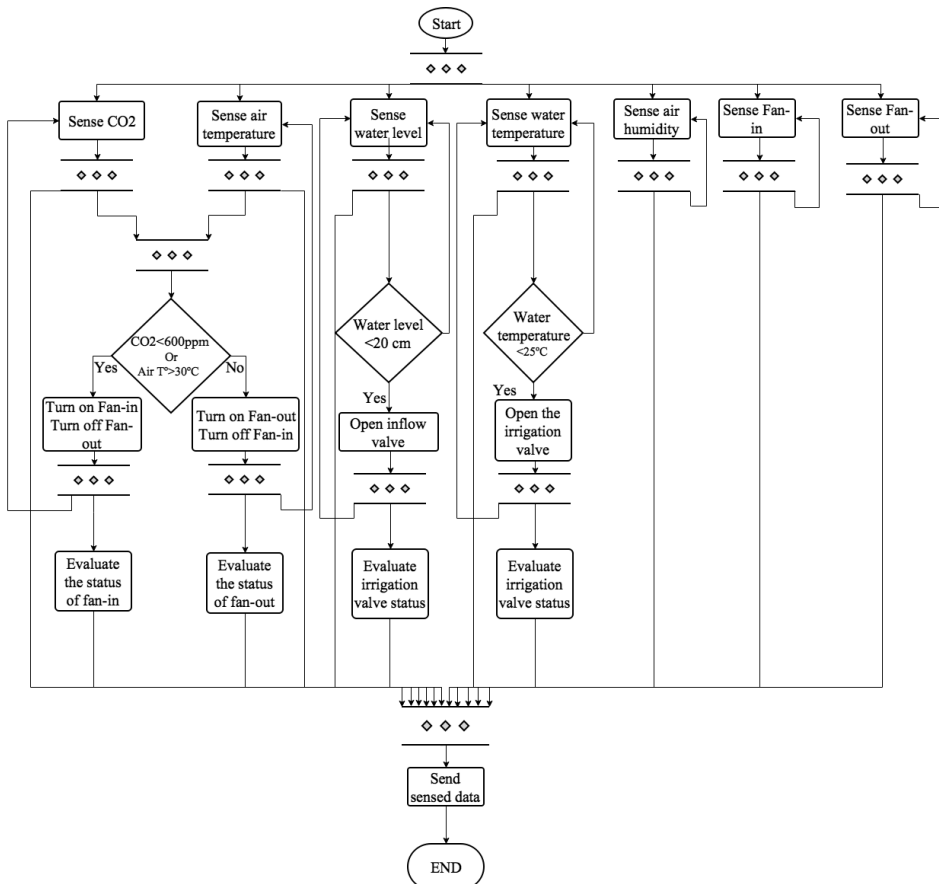


Figure 3.10. Flowchart of the greenhouse unit

of the array represent the different data samples measured during the experiments and the columns correspond to all the sensed signals (including both S_A and S_S). Once the data has been arranged in a single table, two main tasks are conducted in parallel: data is locally stored in a '.txt' file and data is graphically delivered via GUI (as seen in Figure 3.11). Considering our failure analysis will be performed offline, it can only be conducted once the actual experiment has been completed. Although the different controllers implemented a digital filter, in order to clean the signals, signals pre-processing is also conducted in the reasoning unit, before the performance of the failure analysis process. Pre-processing is described deeper in section 3.5.2. This process is

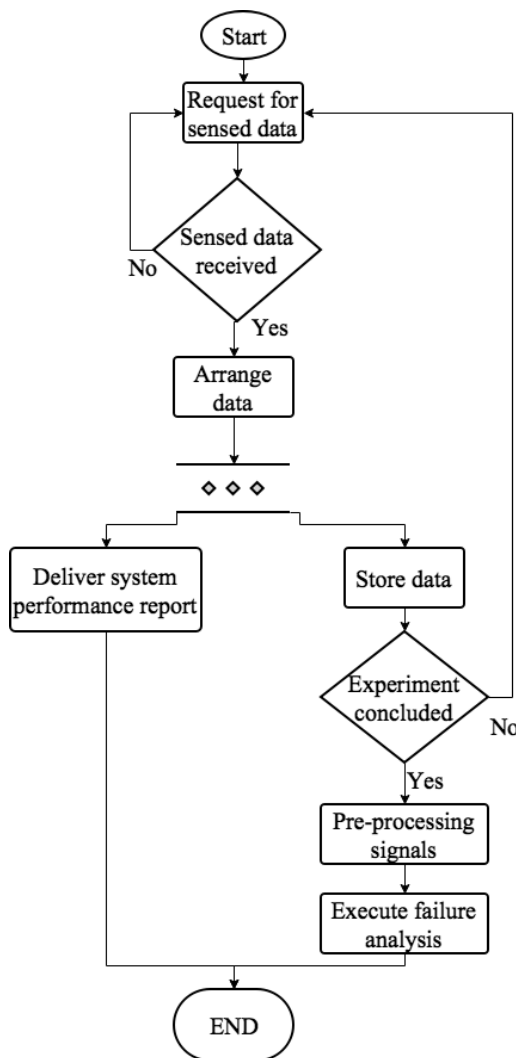


Figure 3.11. Flowchart of the reasoning unit

executed in Matlab. For this purpose, Matlab imports the data stored in the ‘.txt’ file and saves the cleaned data in a ‘.mat’ file, so that it can be reused further. Finally, the failure analysis algorithm is executed by using the ‘.mat’ previously stored. Figure 3.11 presents the operation flow of the reasoning unit.

3.5.2 Signal and information processing

We separated processing of actuator signals S_A from sensor signals S_S . S_A contains information about the operation modes of components and the entire systems and their changes in the time. S_S are the sensed system parameters, such as water level, soil moisture, and air temperature. The Table 3.3 and Table 3.4 present the actuator signals and sensor signals, respectively, considered in the experiment. Actuator signals are binary signals with possible values of 0 or 1, where 0 represents the inactive state and 1 is the active state. For example, the <ResistanceOff> component operation mode corresponding to S_{A_3} is reported as 0, while the <ResistanceOn> component operation mode corresponding to the same component is reported as 1. Sensor signals require sophisticated signal processing techniques. A digital filter was implemented in order to soften the sensor signals and remove disturbances. The filter averages the 40 data samples that are sensed every second to deliver 1 sample per second to the external processing unit. The data was also cleaned from noise caused by communication problems between the coordinator node and the processing unit. This filtering

Table 3.3. Description of the actuator signals of the testbed

System component	Variable	Description	Domain/Set-point
Electro valve Plant bed 1	S_{A_1}	Irrigation valve of Plant bed 1	$\mathbb{E}_{S_{A_1}} = \{\text{ValveClose}, \text{ValveOpen}\}$
Electro valve water reservoir	S_{A_2}	Inlet tank valve	$\mathbb{E}_{S_{A_2}} = \{\text{ValveClose}, \text{ValveOpen}\}$
Heater	S_{A_3}	Water resistance for the heater	$\mathbb{E}_{S_{A_3}} = \{\text{ResistanceOff}, \text{ResistanceOn}\}$
Fan-in	S_{A_4}	Fan-in of the central unit	$\mathbb{E}_{S_{A_4}} = \{\text{Fan-inOff}, \text{Fan-inOn}\}$
Fan-out	S_{A_5}	Fan-out of the central unit	$\mathbb{E}_{S_{A_5}} = \{\text{Fan-inOff}, \text{Fan-inOn}\}$
electro valve Plant bed 2	S_{A_6}	Irrigation valve of Plant bed 2	$\mathbb{E}_{S_{A_6}} = \{\text{ValveClose}, \text{ValveOpen}\}$

process was conducted by (i) analyzing of the signal derivatives, (ii) identifying those measurements ($S_{S_j}(t)$) in which $S_{S_j}(t+1) - S_{S_j}(t) < SD_{S_{S_j}}$, (where $SD_{S_{S_j}}$ denotes the standard deviation of S_{S_j}) and (iii) replacing them with the previous value of ($S_{S_j}(t-1)$).

3.6 Discussion

According to the functional requirements presented in Section 3.3, it can be seen in Table 3.5 the way in which every RTS and RCPS where implemented in the

Table 3.4. Description of the sensor signals of the testbed

System component	Variable	Description
LDR sensor	S_{S1}	White light level on Plant bed 1
Sensor	S_{S2}	Lighting power consumption Plant bed 1
Soil moisture sensor	S_{S3}	Soil humidity Plant bed 1
Soil temperature sensor	S_{S4}	Soil temperature Plant bed 1
PAR light sensor	S_{S5}	PAR light Plant bed 1
Sun light sensor	S_{S6}	UV light level Plant bed 1
PH sensor	S_{S7}	PH level Plant bed 1
Water level sensor	S_{S8}	Water level in tank
Water temperature sensor	S_{S9}	Water temperature in tank
Thermohygrometer	S_{S10}	Greenhouse temperature
Thermohygrometer	S_{S11}	Relative humidity into the greenhouse
CO2 sensor	S_{S12}	CO2 level inot the greenhouse
RPM sensor	S_{S13}	RPM Fan-in
RPM sensor	S_{S14}	RPM Fan-out
LDR sensor	S_{S15}	White light level on Plant bed 2
Sensor	S_{S16}	Lighting power consumption Plant bed 2
Soil moisture sensor	S_{S17}	Soil humidity Plant bed 2
Soil temperature sensor	S_{S18}	Soil temperature Plant bed 2
PAR light sensor	S_{S19}	PAR light Plant bed 2
Sun light sensor	S_{S20}	UV light level Plant bed 2
PH sensor	S_{S21}	PH level Plant bed 2

testbed of the greenhouse, enabling the physical prototype to emulate the behavior needed for the experimental phase of the research. In the following paragraphs, we analyzed the most relevant aspects deeper. The designed testbed allows monitoring the main system variables, as well as SOM transitions. It favors the implementation of a signal-segmentation based analysis that takes as reference the occurring system operation modes.

According to RCPS2.2, it can be detailed that the system samples the signals with 40 samples per second and filters the noise and external disturbances by a digital filter. The filter signals are resampled to a frequency of one sample per second. Digital filters such as moving average are very useful for reducing the

noise and disturbances caused by external factors. They ease the interpretation of the system signals and their analysis, by removing unnecessary data. However, they can also remove relevant information that can be useful for failure analysis, such as transient failure symptoms. Moreover, the smoothing effect caused by the digital filter can mask certain failure manifestations, preventing failure detection. This situation does not seem to be critical in the case of greenhouses, though. Greenhouses are systems where variations of system parameters occur gradually. They do not respond to sudden changes that can last a short period of time and thus, a smaller sampling time is not needed. A lower sampling time would lead to unnecessary amount of data that would make it slower its processing.

According to RCPS4, it can be said that the implementation of wireless communication is very useful for facilitating the transport and relocation of the testbed. It contributes to meet the requirement of generating a testbed that provides a flexible reconfiguration (RCPS9). However, it can also lead to communication drops that can prevent the observation of SOM transitions. This situation requires special attention during the execution of the experiments. In order to overcome these types of problems, the availability of large datasets corresponding to the same failure mode is required. For this purpose, every single experiment will be repeated multiple times in order to have a reliable dataset that can capture the essence of failure manifestations despite the loss of certain data samples.

The defined testbed architecture is required to enable a higher level of decision-making at EFA. The designed testbed includes all required infrastructure for doing so. Nevertheless, it was not used in our experiments. It can be conceived that, based on our findings, further experiments can explore the consequences of the interaction between the three architecture layers (CCA, EFA and CDN) in failure manifestations. We are aware that in exploratory research it is better to build new knowledge through a gradual and systematic process. It allows having a deeper understanding of the influential factors of the studied phenomenon and evaluating the different variables involved, as well as their impact.

The current research will only focus on the study of SOM transitions in failures. We will also only consider those SOM transitions that are caused by the activation and deactivation of system actuators. Although the interaction between CCA, EFA and CDN, as well as the variations on the intensity of operation of system actuators can also alter failure manifestations, we will not study these topics in the current research. The present project is the first of a set of projects that aim to provide insights about failures in CPSs. The above-mentioned aspects (interaction between CCA, EFA and CDN and variations on the intensity of operation of system actuators) will be explored in the further projects, based on the results reported in this document.

Table 3.5. Implementation of the functional requirement

		Requirement	Accomp	Implementation
RTS: Variables control	RTS1	The system should control the air temperature	✓	This requirement is met, thanks to the installed fans
	RTS2	The system should control the relative air humidity	×	Although the RH is monitored, there is not any control action related
	RTS3	The system should maintain the CO2 concentration in a predefined the range	✓	This requirement is met, thanks to the installed fans
	RTS4	The system should control the light intensity (by letting natural light to pass)	✓	The system is equipped with artificial lights whose intensity is adjusted
	RTS5	The system should control soil moisture	✓	The irrigation system is autonomously controlled
RCPS: Self-regulation	RCPS1	The testbed system should be able to monitor actuator status	✓	Actuator status is monitored based on the report of the control actions
	RCPS2.1	The (sensed) system signals should be synchronized by the system	✓	The reasoning unit synchronizes the signals, through Labview
	RCPS2.2	The (sensed) system signals should be sampled at specific moments in time	✓	All sensed signals are sampled every second
	RCPS2.3	The (sensed) system signals should be stored and they need to be accessible	✓	System signals are stored at the reasoning unit in a '.txt' format
	RCPS3	The testbed system should keep the values of the operational parameters stable even under uncertainty	Partially met	Although system ventilation determines the activation and deactivation of the fans, it does not regulate their intensity.
	RCPS4	The testbed system should assure data transmission between the local and remote processing units	✓	The implementation of Xbee units enables duplex data transmission
	RCPS5	The testbed system should be able to combine local and remote control	Partially met	The main system tasks are locally controlled and their setpoints can be overridden through ThingWorx
RCPS: Self-tuning	RCPS6	The system updates variables set-points, according to the operating conditions	✓	ThingWorx implements algorithms for modifying system setpoints.
	RCPS7	The system modifies the actuators' status autonomously	✓	ThingWorx implements algorithms for updating actuators status.
	RCPS8	The testbed system should enable remote monitoring	✓	ThingWorx allows monitoring system parameters and modifying system setpoints via web
Req. For Experiments	RCPS9	The testbed system should have a flexible configuration	×	The system can be transported, but cables hamper its easy re-configuration
	RCPS10	The testbed system should assure continuous system operation	✓	Wire-based energy supply assures the continuous system operation

This dissertation is focused on the data analytics domain. The testbed instrumentation as well as its architecture is here described with the aim to contextualize the reader about the controlled process from which signals were obtained. It provides insights about the type of system from which the

knowledge is derived and thus, about the types of systems where this knowledge can be generalized.

3.7 Conclusions

In this chapter, we analyzed the main characteristics of 1G-CPSs and their implications on failure analysis. This analysis highlighted important challenges in terms of failure manifestations. Variations on system operation modes can be confounded with failure symptoms. They can also mask failure effects, preventing the system to detect failures timely. The analysis of failures through a SOM-based signal segmentation approach seems to be a good option for exploring the role of SOMs in failure manifestations. We developed an artificial environment that mimics the operation of 1G-CPSs. The tight relationship between the system and the environment and the SOM transitions enabled by their self-regulation and self-tuning capabilities can be better studied in real-life environments.

For the development of the testbed we selected a particular application of 1G-CPS. We aimed to derive generalizable knowledge from it, through an inductive approach. A cyber-physical greenhouse was chosen, as it is highly affected by its surrounding environment, its operation implies multiple behaviors, it requires monitoring and controlling multiple variables and failures in these types of systems do not jeopardize human life. The development of the testbed implied studying the main characteristics of greenhouses. Based on it and 1G-CPS characteristics, we derived a set of requirements for the development of our testbed.

The proposed testbed allows monitoring both, system signals, as well as actuator signals in real time, with the aim to determine the time instants in which SOM transitions occur. This particular characteristic, along with the synchronized sensing processes in the system, constitute two of the key factors for enabling SOM-based signal segmentation. The implementation of feedback control in all the implemented processes provides the means for evaluating the self-regulation and self-tuning capabilities that characterize 1G-CPSs. The implementation of an external reasoning unit with higher computing power supports the implementation of complex algorithms for provisioning smart services in the greenhouse.

Considering the literature review conducted in Chapter 2, the analysis performed in this chapter and the testbed features achieved, we will focus this project on data analysis. The analysis will be emphasized on the SOM transitions enabled by feedback controllers at the controlled core area. This analysis will only consider those transitions that are caused by the activation and deactivation of system actuators, i.e. transitions caused by variations on the

intensity of actuators operation will not be studied. In our explorative research, all investigations of failure manifestations will be conducted offline.

3.8 References

- [1] I. Horváth, Z. Rusák, and Y. Li, “Order beyond chaos: Introducing the notion of generation to characterize the continuously evolving implementations of cyber-physical systems,” in *Proc. ASME 2017.*, vol. 2, pp. 1-14.
- [2] J. H. Van Schuppen, “Jump linear systems in automatic control: M. Mariton,” *Automatica*, vol. 28, no. 6, pp. 1284–1285, Nov. 1992.
- [3] M. Broy, “Engineering Cyber-Physical Systems: Challenges and Foundations,” in *Complex Systems Design & Management: Proc. CSD&M 2012*, M. Aiguier, Y. Caseau, D. Krob, and A. Rauzy, Eds., pp. 1–13.
- [4] P. Albertos and I. Mareels, “Control Design,” in *Feedback and Control for Everyone*, P. Albertos and I. Mareels, Eds. Berlin, Germany: Springer Berlin Heidelberg, 2010, pp. 249–274.
- [5] P. Albertos and I. Mareels, “Stability, Sensitivity and Robustness,” in *Feedback and Control for Everyone*, Berlin, Germany: Springer Berlin Heidelberg, 2010, pp. 151–178.
- [6] A. Preumont, “Controllability and Observability,” in *Vibration Control of Active Structures: An Introduction*, 3rd ed. A. Preumont, Ed. Dordrecht, Netherlands: Springer Netherlands, 2011, pp. 275–297.
- [7] P. J. Antsaklis and A. N. Michel, “Controllability and Observability: Fundamental Results,” in *A Linear Systems Primer*, P. J. Antsaklis and A. N. Michel, Eds. Boston, MA, USA: Birkhäuser Boston, 2007, pp. 195–236.
- [8] G. J. Timmerman and P. Kamp, *Computerised environmental control in greenhouses. A step by step approach*. Ptc+.
- [9] R. Isermann, “Fault detection of control loops,” in *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. R. Isermann, Ed. Berlin, Germany: Springer-Verlag, 2006, pp. 253–265
- [10] R. Isermann, “Supervision, fault-detection and fault-diagnosis methods – a short introduction,” in *Combustion Engine Diagnosis: Model-based Condition Monitoring of Gasoline and Diesel Engines and their Components*, Berlin, Germany: Springer Vieweg, 2017, pp. 25–47.
- [11] A. Horch, “A simple method for detection of stiction in control valves,” *Control Eng. Pract.*, vol. 7, no. 10, pp. 1221–1231, Oct.1999.
- [12] Q. P. He, J. Wang, M. Pottmann, and S. J. Qin, “A Curve Fitting Method for Detecting Valve Stiction in Oscillating Control Loops,” *Ind. Eng. Chem. Res.*, vol. 46, no. 13, pp. 4549–4560, May. 2007.
- [13] N. Stanfelj, T. E. Marlin, and J. F. MacGregor, “Monitoring and diagnosing process control performance: the single-loop case,” *Ind. Eng. Chem. Res.*, vol. 32, pp. 301–314, Feb. 1993.
- [14] C. Liu, R. R. Negenborn, H. Zheng, and X. Chu, “A state-compensation extended state observer for model predictive control,” *Eur. J. Control*, vol. 36, pp. 1–9, Jul. 2017.
- [15] T. Hägglund, “A control-loop performance monitor,” *Control Eng. Pract.*, vol. 3, no. 11, pp. 1543–1551, Nov. 1995.

- [16] G. R. Marseglia and D. M. Raimondo, "Active fault diagnosis: A multi-parametric approach," *Automatica*, vol. 79, pp. 223–230, May. 2017.
- [17] H. Niemann and N. K. Poulsen, "Active Fault Diagnosis in Sampled-data Systems," *IFAC-Pap.*, vol. 48, no. 21, pp. 883–888, 2015.
- [18] H. Niemann, "A Setup for Active Fault Diagnosis," *IEEE Trans. Autom. Control*, vol. 51, no. 9, pp. 1572–1578, Sep. 2006.
- [19] R. Garro, L. Ordinez, and O. Alimenti, "Design Patterns for Cyber-Physical Systems: The Case of a Robotic Greenhouse," presented at SBESC, Florianopolis, Santa Catarina, Brazil, 2011, pp. 15–20.
- [20] H. Mirinejad, S. H. Sadati, S. Hasanzadeh, A.M. Shahri, M. Ghasemian "Design and Simulation of an Automated System for Greenhouse using LABVIEW," *American-Eurasian J. agric. & Environ*, vol. 3, no. 2, pp. 279 - 284, 2008.
- [21] J. L. Hatfield and J. H. Prueger, "Temperature extremes: Effect on plant growth and development," *USDA Res. Programs Extreme Events*, vol. 10, no. Part A, pp. 4–10, Dec. 2015.
- [22] Y. Murata and J. Iyama, "Studies on the Photosynthesis of Forage Crops : II. Influence of air-temperature upon the photosynthesis of some forage and grain crops," *Jpn. J. Crop Sci.*, vol. 31, no. 4, pp. 315–322, 1963.
- [23] F. W. Went, "The Effect of Temperature on Plant Growth," *Annu. Rev. Plant Physiol.*, vol. 4, no. 1, pp. 347–362, Jun. 1953.
- [24] Y. Edan, S. Han, and N. Kondo, "Automation in Agriculture," in *Springer Handbook of Automation*, P. S. Y. Nof, Ed. Berlin, Germany: Springer Berlin Heidelberg, 2009, pp. 1095–1128.
- [25] K. S. Kumar, K. N. Tiwari, and M. K. Jha, "Design and technology for greenhouse cooling in tropical and subtropical regions: A review," *Energy Build.*, vol. 41, no. 12, pp. 1269–1275, Dec. 2009.
- [26] C. von Zabeltitz, "Cooling," in *Integrated Greenhouse Systems for Mild Climates*, Berlin, Germany: Springer Berlin Heidelberg, 2011, pp. 251–283.
- [27] Z. Qiu, X. Wang, and Q. Gao, "Temperature and Humidity Precise Control of Greenhouse Based on Nonlinear Model," in *IEMI2012 Proc.*, E. Qi, J. Shen, and R. Dou, Eds. 2013, pp. 805–814.
- [28] H. R. Gisleröd, A. R. Selmer-Olsen, and L. M. Mortensen, "The effect of air humidity on nutrient uptake of some greenhouse plants," *Plant Soil*, vol. 102, no. 2, pp. 193–196, Sep. 1987.
- [29] A. Perales, A. Perdignes, J. L. García, J. I. Montero, and A. Antón, "El control de la condensación en invernaderos," *Tecnología de Producción*, pp. 14-19. Apr. 2003.
- [30] C. von Zabeltitz, "CO2 Enrichment," in *Integrated Greenhouse Systems for Mild Climates*, Berlin, Germany: Springer Berlin Heidelberg, 2011, pp. 341–342.
- [31] N. Gruda and J. Tanny, "Protected Crops," in *Horticulture: Plants for People and Places, Volume 1*, G. R. Dixon and D. E. Aldous, Eds. Netherlands: Springer, 2014, pp. 327–405.
- [32] M. J. Berlinger, W. R. Jarvis, T. J. Jewett, and S. Lebiush-Mordechi, "Managing the Greenhouse, Crop and Crop Environment," in *Integrated Pest and Disease Management in Greenhouse Crops*, R. Albajes, M. L. Gullino, J. C. van Lenteren, and Y. Elad, Eds. Netherlands: Springer, 1999, pp. 97–123.

- [33] S. Y. . Lisar, R. Motafakkerasad, M. M. Hossain, and I. M. Rahman, “Water stress in plants: causes, effects and responses,” in *Water Stress*, Ismail Md. Mofizur Rahman., InTech, 2012, pp. 3–14.
- [34] Y. Yu and B. W. Johnson, “Fault Injection Techniques: A Perspective on the State of Research,” in *Fault Injection Techniques and Tools for Embedded Systems Reliability Evaluation*, Boston, MA, USA: Springer, 2003, pp. 7–39.
- [35] W. A. Woyczyński, “Description of Signals,” in *A First Course in Statistics for Signal Analysis*, Boston, MA, USA: Birkhäuser Boston, 2011, pp. 1–19.
- [36] D. Ilić, E. Troubitsyna, L. Laibinis, and C. Snook, “Formal Development of Mechanisms for Tolerating Transient Faults,” in *Rigorous Development of Complex Fault-Tolerant Systems*, M. Butler, C. B. Jones, A. Romanovsky, and E. Troubitsyna, Eds. Berlin, Germany: Springer Berlin Heidelberg, 2006, pp. 189–209.
- [37] Y. Liu, Y. Peng, B. Wang, S. Yao, and Z. Liu, “Review on cyber-physical systems,” *IEEECAA J. Autom. Sin.*, vol. 4, no. 1, pp. 27–40, Jan. 2017.
- [38] F. Hu *et al.*, “Robust Cyber-Physical Systems: Concept, models, and implementation,” *Future Gener. Comput. Syst.*, vol. 56, pp. 449–475, Mar. 2016.
- [39] M. S. Mahmoud and Y. Xia, “Nominal Control Design,” in *Applied Control Systems Design*, M. S. Mahmoud and Y. Xia, Eds. London, England: Springer London, 2012, pp. 215–307.
- [40] A. Flammini, P. Ferrari, D. Marioli, E. Sisinni, and A. Taroni, “Wired and wireless sensor networks for industrial applications,” *Microelectron. J.*, vol. 40, no. 9, pp. 1322–1336, Sep. 2009.
- [41] C. D. Galvan *et al.*, “Review. Advantages and disadvantages of control theories applied in greenhouse climate control systems,” *Span. J. Agric. Res.*, no. 4, pp. 926–938, 2012.
- [42] Acatech - National Academy of Science and Engineering, “Cyber-Physical Systems — Merging the Physical and Virtual Worlds,” in *Cyber-physical systems. Driving force for innovation in mobility, health, energy and production*, Berlin, Germany: Springer-Verlag Berlin Heidelberg, pp. 15 - 21.

Chapter 4

Investigation of the role of SOM in a signal based failure analysis

4.1 Introduction to research cycle 3

4.1.1 Objectives

The self-regulation and self-tuning capabilities of first generation CPSs present new research challenges for failure diagnosis and forecasting. It was found in our literature review that the control system of first generation CPSs allows system stability by compensating the effect of failure via control actions. It enables the system to operate even under failure conditions. The self-regulating and self-tuning capabilities, however, masks failure manifestations when failures are diagnosed based on the analysis of unsegmented signals. The system controller compensates the effect of failures by manipulation of system actuators that introduces variations in the system operation, that is called “system operation mode” (SOM). SOMs determine the operative status of the system. It is reasonable to think that changes of SOMs will cause changes to failure manifestations.

This chapter presents an explorative analysis of the effect of SOMs on failure symptoms in the context of first generation of CPSs. We report on our investigation on how segmentation of sensor and actuator signals based on system operation modes could improve failure diagnosis. We argue that, while in certain SOMs a failure cannot be observed due to the ‘masking’ effect of system controller, in others, the combination of activated/deactivated actuators can distinctively show the presence of a particular failure. We consider that getting knowledge about the effect of SOMs on failure manifestations could be further operationalized for failure diagnosis and forecasting in first generation of CPSs. The main objectives of this research cycle are: (i) to determine if SOMs accentuate failure manifestations, and (ii) to assess the distinctive power of SOMs in failure diagnosis. The expected result of this chapter is descriptive knowledge that explains to what extent SOMs favor the observation of failure symptoms.

4.1.2 Research approach

Our explorative analysis is conducted based on signal segmentation, where the segmentation criteria are the SOMs. This segmentation is equivalent to extracting the effect of control actions from the system parameters. Statistical features of segmented signals corresponding to a particular SOM under failed mode are compared with signal

segments from the same SOM in a failure-free dataset. Significant difference between the failure-free and failed datasets constitutes a failure symptom. The simulated case of a water kettle is used in order to evaluate our first assumptions in a controlled way. It enables discarding the effect of environmental disturbances on the observed signals. This chapter also presents the results of our experimentation in the instrumented cyber-physical greenhouse testbed.

The chapter is structured as follows: section 4.2 presents an explorative analysis of failure diagnosis based on unsegmented signals. For this purpose, the model of a water kettle is introduced. Section 4.3 presents the key concepts considered for the exploration of the effect of SOM on failure symptoms; section 4.4 describes the technical procedure conducted for operationalizing the explorative method; section 4.5 presents the results concerning to the pilot case of the simulated kettle; section 4.6 presents the results obtained by conducting practical experimentation in the instrumented cyber-physical greenhouse testbed. Finally, Section 4.7 presents the discussion about the observed results and the most relevant conclusions, obtained throughout the whole explorative analysis, are presented in section 4.8.

4.2 Failure diagnosis with unsegmented signals

In this section, we aim to analyze the development of failure symptoms based on unsegmented signals. For this purpose, we introduce the computational model of a kettle, that is used in our experimentation. The results are used as further reference for determining to what extent the SOM-based segmentation improves insight into failure manifestations.

4.2.1 Description of the experiment

The model of the water kettle aims to simulate the behavior of a system whose main function is to heat up water and to inject additives, such as fertilizers, to the water. In Chapter 3 we differentiated between sensor signals (S_S) and actuator signals (S_A). Consequently, the kettle system consists of a set of measured parameters:

- the ‘Water temperature’ (S_{S_1}).
- the ‘Water tank level’ (S_{S_2}).
- the ‘Heating power’ (S_{S_3}).

And a set of actuators:

- an ‘Inflow valve’ (S_{A_1}), which is automatically opened or closed depending exclusively on the ‘Water tank level’ (S_{S_2}).
- an ‘Outflow valve’ (S_{A_2}), which is manually manipulated by the user.
- a ‘Heater’ (S_{A_3}), which is automatically switched on and off depending on ‘Water temperature’ (S_{S_1})

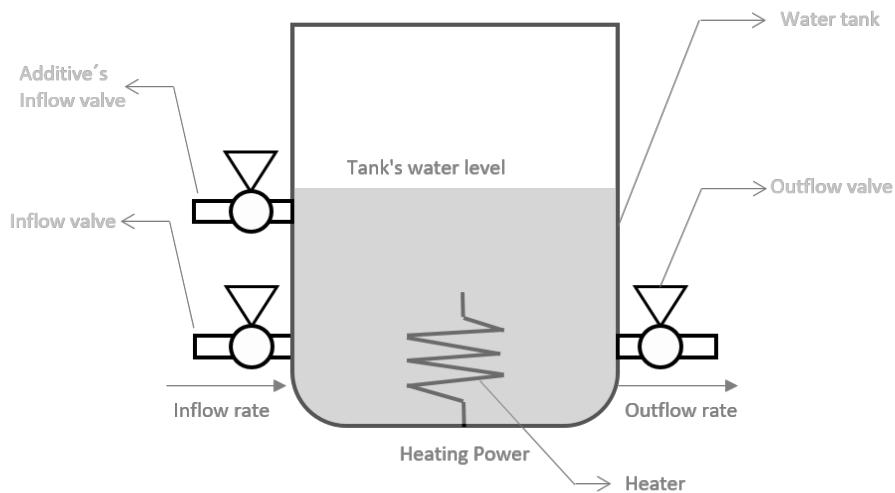


Figure 4.1. Diagram of the kettle model

- an 'Additive injection valve' (S_{A_4}), which is automatically opened or closed. Its activation is based on 'Water tank level' (S_{S_2}) and 'Water temperature' (S_{S_1}), in order to provide a homogeneous mixture.

All sensor and actuator signals are recorded by the system for further processing in failure diagnosis. A diagram of the kettle is presented in Figure 4.1 and the control settings are presented in Table 4.1. Control settings of the kettle model, where 1 means 'open' for valves and 'ON' for the heater. Likewise, 0 means 'closed' for valves and 'OFF' for the heater.

Four progressive failures were injected into the system considering a simulation of 1000 steps:

- **Tank leak:** was injected by adding an additional outflow to the mass conservation

Table 4.1. Control settings of the kettle model

Actuator	Control settings
Inflow valve	if $S_{S_2} < 2L$, then $S_{A_1} = 1$ if $S_{S_2} \geq 3,78L$, then, $S_{A_1} = 0$
Outflow valve	Manually opened/closed
Additive injection valve	if $S_{S_2} \geq 3,78L$, then, $S_{A_4} = 0$ if $S_{S_2} < 3L$, then $S_{A_4} = 1$
Heater	if $S_{S_1} < 50^{\circ}C$, then, $S_{A_3} = 1$ if $S_{S_1} > 55^{\circ}C$, then, $S_{A_3} = 0$

equation. It was gradually increased from value of $1.0344e-7$ L/s to 0.004 L/s.

- **Inflow valve obstruction:** the inflow rate was gradually reduced from 0.1261 L/s to 0.063 L/s by gradually decreasing this variable in each step of the simulation.
- **Loss of heating power:** the heating power was reduced from 4000 Watts, which is the regular value to 0 Watts in the last step of the failure progression.
- **Outflow valve obstruction:** the outflow rate of water was also gradually reduced from 0.063 L/s up to 0 L/s in order to simulate a complete obstruction of the outflow valve.

All failures modes were separated injected into several simulation scenarios. We denote the failure modes as F_r , where: F_1 denotes ‘Tank leak’, F_2 denotes ‘Inflow valve obstruction’, F_3 denotes ‘Loss of heating power’, and F_4 denotes ‘Outflow valve obstruction’.

A failure free version of the kettle’s model was simulated with a sampling of 5000 data points. Considering that the user manually operates the outflow valve (S_{A_2}), six different use conditions were included, as it is shown in Figure 4.2. Each condition varies in the opening times of S_{A_2} , its duration and frequency. Operating conditions were also considered through controlled variations of ambient temperature (T_{amb}) and ‘Water tank level’ (S_{S_2}) with an initial value of volume (V_o). T_{amb} was systematically sampled between 20°C and 30°C and V_o was sampled between 1 and 2 L. The combinations between operating and use conditions led to different scenarios that allow simulating the effect of external (non-controllable) conditions over system performance. A single combination composed by a particular use and a particular operating condition is considered as one scenario. A total of 15 kettle models that correspond to 15 different combinations of use and operating conditions (i.e. scenarios) were analyzed. The same combination of use conditions and operating conditions considered for failure-free operation (i.e. the same 15 scenarios used for the failure-free models) were used for the failed models.

For the analyzes presented in this chapter, we considered data corresponding to the following failure levels:

- a leak rate of 0.0002 L/s.
- a reduction of 0.0126 L/s in the inflow rate.

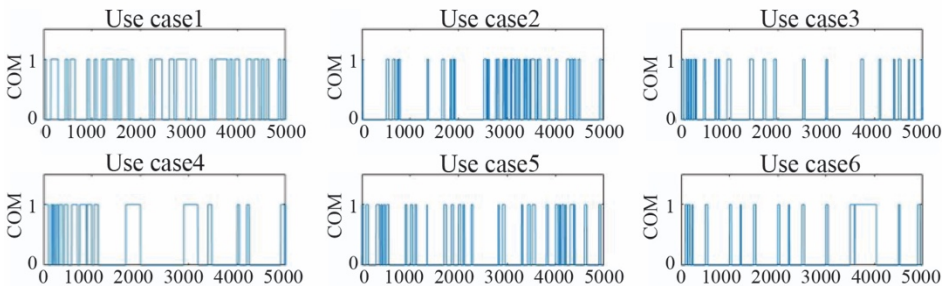


Figure 4.2. Use conditions for the Outflow valve

- a reduction of 400 Watts in the heating power.
- a reduction of 0.0126 L/s in the outflow of the outflow valve.

4.2.2 Method for analyzing output data

1. Data analysis

To explore failure symptoms in the collected data sets, various statistical tests were done based on the steps presented below:

- Signals of all sensors and actuators were recorded both for failure-free and failed operation of the water kettle and they were used as information carriers in the failure diagnosis process. Signals corresponding to all the aforementioned scenarios were collected and stored in a database.
- The obtained signals were transformed to statistical data features for all recorded scenarios: derivative, standard deviation, mean, area, and median. The statistical data features were determined as follows:

- Derivative:

$$a_1 = S'_s(t) = \left(\frac{dS_s}{dt} \right) \quad (4.1)$$

- Standard deviation:

$$a_2 = \sigma = \left(\sqrt{\left(\sum_{i=1}^N (S_{S_i} - \bar{S}_S)^2 \right) / N - 1} \right) \quad (4.2)$$

- Mean:

$$a_3 = \bar{S}_S = \frac{1}{n} (\sum_{i=1}^n S_{S_i}) \quad (4.3)$$

- Area:

$$a_4 = A_{S_S} = \left(\int_{t_e}^{t_s} S_S \right) \quad (4.4)$$

- Median:

$$a_5 = \tilde{S}_S \quad (4.5)$$

- The statistical data features of failure-free operation were compared to features corresponding to failed operation in a pair-wise way. For instance, the feature 'mean', measured from a failure-free experiment was compared with the feature 'mean' from tank-leak failure signal. Both, the failure-free and the failed features, corresponded to the same use and operative conditions (scenarios). The Kruskal-wallis test was selected as it is a non-parametric test which does not assume normal distribution.
- The calculated test significance (p) was determined and used as a decision enabler of failure symptoms.

- A $p = 0.05$ (Fisher's criterion) was considered as the threshold for determining failure symptoms. When the P_{value} (p) was below the defined threshold there was no evidence for discarding that the observed difference is caused by failure.

2. Data structuring

The characterization of sensor signals S_S through a set of features, requires a transformation process. This process can be denoted as a function: $a_m(S_{S_i})$. Where, a_m represents a signal feature from the set of m features (explained in 4.2.2), where:

- $a_{m=1}$ is the 'Derivative' feature.
- $a_{m=2}$ is the 'Standard deviation' feature.
- $a_{m=3}$ is the 'Mean' feature.
- $a_{m=4}$ is 'Area' feature.
- $a_{m=5}$ is 'Median' feature.

Signal features obtained from different experiments corresponding to the same failure mode were arranged into a set of vectors V , as follows:

$$V_{i_r}^m = [a_m(S_{S_i}^{r,sc=1}), a_m(S_{S_i}^{r,sc=2}), \dots, a_m(S_{S_i}^{r,sc=15})] \quad (4.6)$$

where: a_m corresponds to the signal feature m , r denotes the analyzed Failure Mode F_r , i corresponds to one of the three analyzed system signals, and $sc = [1, \dots, 15]$ denotes the considered scenarios.

Consequently, every vector $V_{i_r}^m$ of F_r , for feature a_m and signal S_{S_i} was compared with a vector $V_{i_{Free}}^m$ corresponding to the failure free cases for feature a_m and signal S_{S_i} . The latter vector is defined as:

$$V_{i_{Free}}^m = [a_m(S_{S_i}^{Free,sc=1}), a_m(S_{S_i}^{Free,sc=2}), \dots, a_m(S_{S_i}^{Free,sc=15})] \quad (4.7)$$

3. Considerations for the statistical test (ST):

A sample dataset of failure-free behavior was compared with a sample of dataset derived from system operation during failure mode (F_r). For the sake of clearness, let's denote:

- A^∂ : Datasets corresponding to 'Reference (failure-free) operation'
- A^O : Datasets corresponding to 'Observed (failed) operation'.

Considering $ST(A^\partial, A^O)$, our null hypothesis H_0 states that both samples A^∂ and A^O have the same distribution Θ , that is:

$$H_0: \mu_\partial \in \Theta \wedge \mu_O \in \Theta$$

where μ_∂ is the mean of the sample A^∂ and μ_O is the mean of the observed data A^O . Conversely, our alternative hypothesis H_1 states that both samples A^∂ and A^O belongs to different distributions:

$$H_1: \mu_\partial \in \Theta \wedge \mu_0 \notin \Theta$$

Therefore,

$$\Phi(A^\partial, A^0) = \begin{cases} 1 & \text{if } ST(A^\partial, A^0) \in \Omega \\ 0 & \text{if } ST(A^\partial, A^0) \notin \Omega \end{cases} \quad (4.8)$$

where Ω is the rejection region. When $\Phi(A^\partial, A^0) = 1$, it means that H_0 is rejected, i.e. there is low likelihood that the observed deviation occurred by effect of chance. Conversely, when $\Phi(A^\partial, A^0) = 0$, means that H_0 is accepted and, thus, we cannot discard that observed difference is caused by effect of chance. In order to determine the result of Φ , the statistical probability indicator (p -value) is used. The p -value determines the probability that $ST(A^\partial, A^0) \in \Omega$, due to the effect of failure. In our case, we have considered a p -value of 0.05 as the threshold that determines the significance of the results, so that:

Table.4.2. Results from significance test corresponding to the whole length signal segment analysis

	Signal	Derivative	Standard deviation	Mean	Area	Median
F1	S_{S_1}	0.76	0.82	0.71	1.00	1.00
	S_{S_2}	0.82	0.88	0.36	0.55	0.13
	S_{S_3}	0.82	0.94	0.71	0.20	1.00

	Signal	Derivative	Standard deviation	Mean	Area	Median
F2	S_{S_1}	1.00	0.94	1.00	0.71	0.76
	S_{S_2}	0.71	0.94	0.60	0.50	0.13
	S_{S_3}	0.94	0.38	0.97	0.66	0.66

	Signal	Derivative	Standard deviation	Mean	Area	Median
F3	S_{S_1}	0.20	0.20	0.20	0.17	0.08
	S_{S_2}	0.43	0.43	0.38	0.57	0.62
	S_{S_3}	0.88	0.00	0.94	0.83	0.94

	Signal	Derivative	Standard deviation	Mean	Area	Median
F4	S_{S_1}	0.20	0.29	0.20	0.20	0.55
	S_{S_2}	0.76	0.36	0.71	0.94	0.88
	S_{S_3}	0.26	1.00	0.33	0.69	1.00

$$\begin{cases} ST(A^\partial, A^o) \in \Omega \text{ if } p \leq 0.05 \\ ST(A^\partial, A^o) \notin \Omega \text{ if } p > 0.05 \end{cases} \quad (4.9)$$

The result of this process is reported in a vector called Failure Indicator FI , where $FI_i = 1$ if $p_{S_{S_i}} \leq 0.05$, and $FI_i = 0$ if $p_{S_{S_i}} > 0.05$.

4.2.3 Failure analysis based on unsegmented signals

Table.4.2 presents the results corresponding to the significance analysis conducted through Kruskal Wallis. We can observe, that the only Failure Mode (F_r) that presented failure symptoms (i.e. $p \leq 0.05$), was F_3 (Loss of heating power). It occurred in the combination of the ‘Heating power’ (S_{S_3}) signal and at the ‘Standard deviation’ feature (a_2). The rest of the features presented no symptoms in any of the analyzed signals.

In order to complement the statistical test results, we also evaluated the effect size of the failure by analyzing the Pearson’s correlation coefficient with a Wilcoxon signed-rank

Table 4.3. *Effect size (r) of the statistical test conducted to the whole length signal*

	Signal	Derivative	Standard deviation	Mean	Area	Median
F1	S_{S_1}	-0.15	-0.08	0.10	0.13	0.13
	S_{S_2}	-0.07	-0.14	0.06	-0.17	0.27
	S_{S_3}	0.13	0.20	0.00	-0.54	-0.08
F2	S_{S_1}	0.08	-0.06	0.03	0.23	-0.08
	S_{S_2}	0.07	0.01	0.24	0.03	0.01
	S_{S_3}	-0.03	0.01	-0.08	-0.35	0.11
F3	S_{S_1}	0.39	-0.39	0.44	0.39	-0.34
	S_{S_2}	0.35	-0.26	0.11	0.08	-0.26
	S_{S_3}	-0.07	0.84	0.00	-0.21	0.11
F4	S_{S_1}	-0.13	0.13	-0.03	-0.13	0.13
	S_{S_2}	-0.24	0.18	-0.34	-0.18	0.17
	S_{S_3}	0.13	0.07	-0.08	-0.28	-0.10

test. We aimed to determine how significant was the effect of the injected failure on the observed data by quantifying the difference between the failure-free data and the failed one. This statistical technique delivers an index r that is constrained between 0 and 1. Considering we are working with the Pearson's correlation coefficient, 0 means there is no effect and 1 means that there is a clear effect. It is widely accepted in literature that a value of $r=.1$ denotes a small effect, $r=.3$ denotes a medium effect, and $r=.5$ denotes a large effect. This index can also present negative values, which indicates the observed difference is below the mean difference between both groups.

Table 4.3 presents the results of the effect size analysis. It can be observed that the only Failure Mode (F_r) that presented a large effect was 'Loss of heating power' (F_3), in the combination of 'Heating power' (S_{S_3}) signal and at the 'Standard deviation' (a_2) feature. It coincides with the results corresponding to the significance test where $p \leq 0.05$ for the same signal, at the same feature, under the effect of the same failure mode (F_r). This example demonstrates that manifestations from 'Tank leak' (F_1), 'Inflow valve obstruction' (F_2) and 'Outflow valve obstruction' (F_4) cannot be perceived at the failure progress level analyzed. Only 'Loss of heating power' (F_3) presented clear failure symptoms based on the 'Heating power' (S_{S_3}) signal.

So far, we cannot derive any conclusions about the control effect over the lack of symptoms in the output signal. We can only make inferences about the effect of the size of the failure on the studied signals. This particular analysis aimed to determine to what extent failure symptoms manifested in the unsegmented signals at the studied failure progress level. In the forthcoming sections, we will analyze the influence of control actions on the output signals through SOM-based signal segmentation. These results will be used as reference for evaluating the benefits of SOMs in failure analysis.

4.3 Method for SOM based signal segmentation for failure analysis

4.3.1 Elements of the theory underpinning the conducted exploration

The method to analyze the effect of SOMs on failure manifestations on output signals constitutes our explorative approach for deriving descriptive knowledge about the role of SOMs on failure symptoms. The research methodology follows the guidelines of Design Inclusive Research [1]. This method is based on the estimation of a statistical probability indicator that enables to determine the likelihood that the observed data of a failure-free dataset.

In chapter 1 we introduced the concept of system operation modes. We stated that it is based on the input signals of the system, for which the concept of Component Operation Mode (COM) was also presented. From now on, the set of all potential COMs (ζ) of S_{A_j} will be expressed as $\mathbb{E}_{S_{A_j}} = \{\zeta_1, \dots, \zeta_u\}$. Likewise, the COM that a given S_{A_j} presents at time t , will be denoted as $\zeta_{S_{A_j}}(t)$. All the above-presented definitions are necessary for technically introducing the concept of 'System Level Failure Indicator'. We will denote

as ζ_d , any particular SOM at time t , where $\zeta(t) = \{\zeta_d(t) \mid d = 1, \dots, l\}$. Considering that SOM denotes a unique combination of COM at time t , ζ_d can be expressed as $\zeta_d = \{\zeta_{S_{A_1}}(t), \zeta_{S_{A_2}}(t), \dots, \zeta_{S_{A_k}}(t)\}$. For instance, let's assume that our water tank is irrigating and heating up the water in the reservoir at the same time t . Considering the aforementioned notation, the System Operation Mode (ζ_d) at time t can be represented as $\zeta_d = \{\zeta_{S_{A_1}}(t), \zeta_{S_{A_2}}(t), \zeta_{S_{A_3}}(t), \zeta_{S_{A_4}}(t)\}$, where, for example, it may represent a situation such as $\zeta_{S_{A_1}}(t) = \text{valveOpen}$, $\zeta_{S_{A_2}}(t) = \text{valveClose}$, $\zeta_{S_{A_3}}(t) = \text{valveClose}$, and $\zeta_{S_{A_4}}(t) = \text{HeaterOn}$.

Some other important concepts should also be introduced, for defining a system level failure indicator concept. These are: (i) 'Failure Mode' (F_r) and (ii) 'Signal Segment' (Sg). 'Failure Mode' is a particular type of failure that can occur in a system. Every 'Failure Mode' presents certain characteristic symptoms (\emptyset) that can be used for failure diagnosis. We will consider as a symptom any deviation of a signal from its expected behavior, which can be explained by the effect of a failure.

A 'Signal segment' (Sg), is a part of a signal that was measured during a particular time interval. It includes all measured/sampled values of a Signal (S_{S_j}) between a start point in time (t_s) and an end point in time (t_e). The start point and the end point are determined by any variations on any COM. Consequently, Sg represents a system variable during a specific SOM (ζ_d). The same SOM (ζ_d) can occur several times during the operation of the system. Therefore, each individual segment, representing a SOM, is the set $Sg_h = [S_S(t_s), \dots, S_S(t_e)]$ where $S_S(t)$ is the measured value of a particular sensor signal (S_S) in a specific time t . The implication of this definition is that the SOM time, $t_s - 1$ and $t_e - 1$ is different than the SOM of Sg_h .

4.3.2 Description of the proposed Failure Indicator (FI)

We define 'Failure Indicator' (FI) as a $m \times n$ matrix, whose rows are system Sensor Signals (S_S) and whose columns are the system Actuator Signals (S_A). In the FI matrix, each item ($FI_{i,j}$) is composed by the presence or absence of a Symptom (\emptyset) in a Segment (Sg_h) from signal (S_{S_j}) corresponding to a particular SOM (ζ_d). That is:

$$FI = \begin{bmatrix} \emptyset_{1,1} & \dots & \emptyset_{1,n} \\ \vdots & \ddots & \vdots \\ \emptyset_{m,1} & \dots & \emptyset_{m,n} \end{bmatrix} \quad (4.10)$$

Where:

$$\emptyset(S_{S_j}, \zeta_d) = \begin{cases} \emptyset = 1, & \text{if there is a symptom} \\ \emptyset = 0, & \text{if there is a lack of symptom} \end{cases}$$

In order to build a failure indicator, it can be inferred that failures are derived as the deviation between the signal features obtained from the System Regular Operation (\emptyset) (i.e. without failure) and the irregular operation of the system (i.e. under fault or failure conditions). The term, Reference Behavior (∂) is introduced for describing the characteristic behavior of, either a particular Failure Mode (F_r), or a failure-free system. In every SOM (ζ_d), data corresponding to the observed Signal (S_{S_j}), are compared with

data of the system's reference behavior ∂ , corresponding to a given Signal (S_{S_j}), during ζ_d . A detailed description of the process to be followed for building a FI is described in the following sections.

4.3.3 Proposed procedure for signal segmentation

1. Building system database:

Digitalized signals of the system are represented in a vector S where signals from actuators S_A and sensors S_S are arranged in the following form:

$$S = [S_{A_1}, \dots, S_{A_n}, S_{S_1}, \dots, S_{S_p}],$$

Where n denotes the number of actuators and p is the number of sensors. All signals have the same sample time, t , so each measure from sensors, at that specific moment, is stored as follows:

$$S(t) = [S_{A_1}(t), \dots, S_{A_n}(t), S_{S_1}(t), \dots, S_{S_p}(t)]$$

Consequently, the entire system database DT is composed of all stored $S(t)$ vectors, measured in a progressively augmentation in time t , so that:

$$DT = \begin{bmatrix} S(t = 1) \\ S(t = 2) \\ \vdots \\ S(t = k) \end{bmatrix} \quad (4.11)$$

2. Identification and representation of operation modes of a system:

The SOMs are the result of a unique combination of the COMs. The progression in time t enables SOM identification by analyzing if any $S_{A_j}(t) \neq S_{A_j}(t + 1)$ implying a change in the SOM (by a change in the COMs). For this reason, a collection of COM is gathered in an 'Operation Mode Matrix' (OM) with v rows, each one with the vector $S_A^{\zeta_l}$ (representing the vector S_A -COM- in the SOM ζ_l) and n columns (disaggregating each Actuator's Signal S_{A_j} present in the system). The $OM_{v \times n}$ matrix is, then, denoted as:

$$OM_{v \times n} = \begin{bmatrix} S_{A_1}^{\zeta_1} & \dots & S_{A_n}^{\zeta_1} \\ \vdots & \ddots & \vdots \\ S_{A_1}^{\zeta_l} & \dots & S_{A_n}^{\zeta_l} \end{bmatrix} = \begin{bmatrix} S_A^{\zeta_1} \\ \vdots \\ S_A^{\zeta_l} \end{bmatrix} \quad (4.12)$$

The time t' in which $S_{A_j}(t') \neq S_{A_j}(t' + 1)$ is collected, in order to determine the moment in which the SOM changes. Those times are collected in a $T_{v \times 2}$ matrix (see Eq. 4.13) where the starting time of a ζ_d is denoted as t_s , and its SOM ending time, is denoted by t_e . Therefore, t_{s_1} denotes the time in which ζ_1 started and t_{e_1} is the time in which ζ_1 ends. Consequently, t_{e_1} is the time where the system switches to the next SOM ζ_2 . The starting time of ζ_2 , is then $t_{s_2} = t_{e_1} + 1$.

$$T = \begin{bmatrix} t_{s_1} & t_{e_1} \\ \vdots & \vdots \\ t_{s_l} & t_{e_l} \end{bmatrix} \quad (4.13)$$

A column-matrix is also created, as seen in Eq 4.14, in order to gather all the observed system operation modes ζ_d in OM matrix, where, v denotes the total amount of ζ that may occur in the system.

$$SOM = \begin{bmatrix} \zeta_1 \\ \vdots \\ \zeta_v \end{bmatrix} \quad (4.14)$$

This matrix is used as reference for conducting signal segmentation.

3. Segmentation of signals:

Each row of T (See Eq. 4.13) is used as reference for signal segmentation. Data represented in DT (See Eq. (4.11)) is extracted and ‘Signal segments’ (Sg) are stored in vectors denoted as Sg_h , where $h = 1, \dots, w$, being w the total amount of signal segments. Time instants t_s and t_e that are stored in the row h (from Sg_h) within matrix T (See Eq. 4.13), are used as reference to extract all data corresponding to S_s from DT that is in the time interval between t_{s_h} and t_{e_h} , so that:

$$Sg_h = \{S_s(t_{s_h}), \dots, S_s(t_{e_h})\} \quad (4.15)$$

Therefore, there is a data set DS that contains all signal segments, so that:

$$DS = \{Sg_1, \dots, Sg_h, \dots, Sg_w\} \quad (4.16)$$

The segmented data are stored in an array called *Segments* that organizes each Sg_h depending on the SOM that was occurring in the period of time when signal segment Sg_h was sensed. For example, if segments Sg_7 , Sg_{15} and Sg_{30} were obtained in different periods of time, in which the system was on ζ_3 , we can denote this phenomenon as $Segment.\zeta_3 = \{Sg_7, Sg_{15}, Sg_{30}\}$. Likewise, if segments Sg_5 , Sg_{40} , Sg_{58} and Sg_{72} were sensed while the system was on the System Operation Mode ζ_1 , we denote this as $Segment.\zeta_1 = \{Sg_5, Sg_{40}, Sg_{58}, Sg_{72}\}$. The difference between DS and *Segment* is that the last one classifies signal segments based on the SOM that was occurring when the data was obtained. DS only integrates all signal segments into a single data set.

4. Characterization of signal segments:

A signal feature, $a(Segment.\zeta_g)$, provides a simple representation that is capable to capture the trend and characteristic features of signal segments. Depending on the type of signal, the feature may be determined by parameters relevant for time-domain or frequency-domain signal analysis. Considering that $a(Segment.\zeta_g)$, there is a vector A that groups the features of all segments Sg_h stored in $Segment.\zeta$, so that:

$$A_{\zeta_d} = \begin{bmatrix} a(\text{Segment. } \zeta_1) \\ \vdots \\ a(\text{Segment. } \zeta_z) \end{bmatrix} \quad (17)$$

Where z is the total amount of segments corresponding to ζ_d . In a similar way, as explained in section 4.2.2, a set of signal features can be used to characterize signal segments. The selection of a depends on the particular characteristics of each signal.

5. Evaluation of deviation of signals:

The evaluation of the deviation of signals aims to identify significant anomalies in the operation of the system (which may be considered as a failure) by the application of a statistical test (ST). For this test, let's use a sample $A_{\zeta_d}^\partial$ that comes from the system's Reference Behavior (∂) during the system's operation mode ζ_d . Then, the sample $A_{\zeta_d}^\partial$ is compared with the system's observed behavior $A_{\zeta_d}^o$, during the same ζ_d . The observed behavior can, or cannot, be under the effect of a particular failure. The aim of ST is determining whether $A_{\zeta_d}^o$ present any considerable deviation from the Reference Behavior (∂), due to the effect of a failure. For instance, considering $ST(A_{\zeta_d}^\partial, A_{\zeta_d}^o)$, our null hypothesis H_0 states that both samples $A_{\zeta_d}^\partial$ and $A_{\zeta_d}^o$ have the same distribution Θ , that is:

$$H_0: \mu_\partial \in \Theta \wedge \mu_o \in \Theta$$

Where μ_∂ is the mean of the sample $A_{\zeta_d}^\partial$ and μ_o is the mean of the observed data $A_{\zeta_d}^o$. Conversely, our alternative hypothesis H_1 states that both samples $A_{\zeta_d}^\partial$ and $A_{\zeta_d}^o$ belongs to different distributions:

$$H_1: \mu_\partial \in \Theta \wedge \mu_o \notin \Theta$$

Therefore,

$$\Phi(A_{\zeta_d}^\partial, A_{\zeta_d}^o) = \begin{cases} 1 & \text{if } ST(A_{\zeta_d}^\partial, A_{\zeta_d}^o) \in \Omega \\ 0 & \text{if } ST(A_{\zeta_d}^\partial, A_{\zeta_d}^o) \notin \Omega \end{cases} \quad (4.18)$$

Where Ω is the rejection region. When $\Phi(A_{\zeta_d}^\partial, A_{\zeta_d}^o) = 1$, it means that H_0 is rejected and, therefore, as described in Equation 4.8 (section 4.3.2), $\emptyset(S_{S_j}, \zeta_D) = 1$, while $\Phi(A_{\zeta_d}^\partial, A_{\zeta_d}^o) = 0$ means that H_0 is accepted and, thus, $\emptyset(S_{S_j}, \zeta_D) = 0$. In order to determine the result of Φ and, thus, the result of $\emptyset(S_{S_j}, \zeta_D)$, the use of p -value is suggested for statistical hypothesis testing. The p -value determines the probability that $ST(A_{\zeta_d}^\partial, A_{\zeta_d}^o) \in \Omega$, due to the effect of failure. In our case we have considered a p -value of 0.05 as the threshold that determines the significance of the results, so that:

$$\begin{cases} ST(A_{\zeta_d}^\partial, A_{\zeta_d}^o) \in \Omega & \text{if } p \leq 0.05 \\ ST(A_{\zeta_d}^\partial, A_{\zeta_d}^o) \notin \Omega & \text{if } p > 0.05 \end{cases} \quad (4.19)$$

Considering that this process is conducted for all signal segments, there is a p -value for each element $p_{i,j}$, of the FI matrix (See Eq.4.10), so that $\emptyset(S_{S_j}, \zeta_D)$ can be determined. The result of this process is the matrix FI, where $FI_{i,j} = 1$ if $p_{i,j} < 0.05$, and $FI_{i,j} = 0$ if

$p_{i,j} \leq 0.05$. Therefore, FI will be a matrix composed by zeros and ones. To evaluate the deviations of signals, different statistical tests can be implemented depending on the type of data distributions.

6. Deriving failure indicators:

The analysis of failure manifestations should be conducted based on the frequency of symptoms occurrence in multiple sets of experiments. It aims to reduce sensitivity of the failure indicator to noise and disturbance caused by external factors such as the time of operation or weather conditions. For this reason, failure indicators of multiple experiments, in which the system is subjected to the same ‘Failure mode’ (F_r), are combined into a single matrix, that will be denoted as the ‘Reference Indicator’ (FI^∂). The experiments are conducted during different use scenarios, which are defined by various initial conditions representing the operation context, or system operations influenced by the surrounding environment and user’s interaction. A resultant FI matrix, so called FI^∂ , is formed, so as:

$$\left\{ \begin{array}{ll} \text{if } \left(\sum_{Sc_1}^{Sc_\psi} FI_{i,j} \right) < 0.4 * \psi, & \text{then, } FI_{i,j}^\partial = 0 \\ \text{if } 0.4 * \psi < \left(\sum_{Sc_1}^{Sc_\psi} FI_{i,j} \right) < 0.95 * \psi, & \text{then, } FI_{i,j}^\partial = 0.5 \\ \text{if } \left(\sum_{Sc_1}^{Sc_\psi} FI_{i,j} \right) > 0.95 * \psi, & \text{then, } FI_{i,j}^\partial = 1 \end{array} \right. \quad (4.20)$$

The index ∂ denotes the ‘Reference Behavior’ that the FI matrix represents. Consequently, the FI^∂ corresponds to the ‘Reference Indicator’ and, likewise, the FI^O will denote the ‘Observed Indicator’ corresponding to the current failure performance. Sc is a data set corresponding to a specific scenario, and ψ is the total amount of cases analyzed. If the $FI_{i,j}$ considered for deriving $FI_{i,j}^\partial$ present failure symptoms, i.e. $\emptyset(S_S, \zeta_D) = 1$, with a frequency higher than 95%, then $FI_{i,j}^\partial$ will be colored red. Yellow cells are those cells, where symptoms occurred with a frequency lower than 95%, but higher than 40% of the total of the experiments considered. Conversely, if 40% of the $FI_{i,j}$ considered for deriving $FI_{i,j}^\partial$ do not present any symptom, i.e. $\emptyset(S_S, \zeta_D) = 0$, then, it will be colored green. In order to visually clarify the concept, Figure 4.3 depicts the way FI matrix will look like, by coloring $FI_{i,j}$ cells. There, rows correspond to the Sensor signals (S_S) and columns to System Operation Modes (ζ_d).

4.4 A pilot study of the SOMs-oriented failure analysis methodology

An evaluation of the proposed failure indicator concept was conducted through a pilot study. This explorative experiment was performed with the previously presented kettle model (See section 4.2.1), in order to reduce disturbances caused by external factors, and to guarantee the observed deviations were caused by failure symptoms. Our main goal is to investigate the role of the SOMs in failure diagnosis through signal segmentation, under idealistic scenarios in which the environmental and external disturbances can be controlled. This analysis is composed by the following steps:

1. Derivation of the failure indicator matrix: Failure indicators are derived in the analysis of significance and size effect step. For this purpose, the process presented in section 4.3.3 is followed. Failure indicator is derived by comparing every failure-free scenario with the failed version of the same scenario through a pair-wise comparison. It enables discarding the observed deviations that are caused by variations in use and operative conditions.

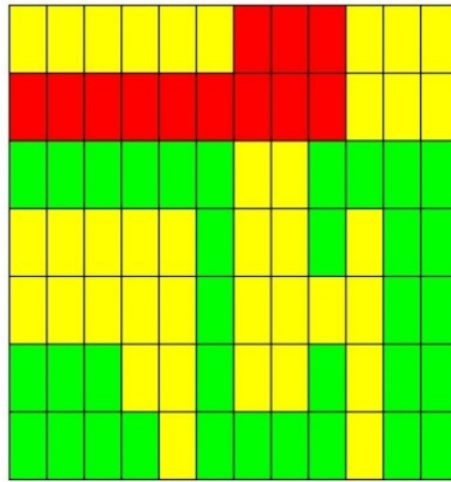


Figure 4.3. Example of a colored R matrix, where rows correspond to the Sensor signals (S_S) and columns to System Operation Modes (ζ_a)

2. Analysis of the discriminant power of the failure indicator: The analysis of the discriminant power aims to determine to what extent the failure indicator matrix of a particular Failure Mode (F_r) is similar to the one presented by another Failure Mode (F_r'). For this purpose, the percentage of common elements of FI matrix between failure indicators is evaluated through

$$Sl = \left(\frac{C}{T} \right) \times 100 \quad (4.21)$$

where C are the number of common elements of the (F_r) and (F_r') matrices. (i.e. number of symptoms and lack of symptoms in common) and T is total number of positions of the matrix.

3. Analysis of sensitivity to variations in use and operative conditions: aims to explore to what extent variations in use and operative conditions can affect the derived failure indicators. For this purpose, every failure-free scenario was compared with a

randomly selected failed scenario in order to derive a new set of failure indicators (FI). Derived matrices are compared with the ones presented on step 1, by evaluating the percentage of elements in common between them.

As this first case uses the simulation model of the kettle, the implementation of this analysis in a system subject to real-life conditions was conducted in the instrumented testbed. This case is presented in section 4.5.

4.4.1 Derivation of failure indicator matrix

In this step, we derived fifteen FI matrices per Failure Mode (F_r), with the aim to determine their failure indicators FI^θ . We are considering five signal features in our analysis (those proposed in section 4.2.2): (i) Derivative (a_1), (ii) Standard deviation (a_2), (iii) Mean (a_3), (iv) Area (a_4), (v) Median (a_5).

Every signal feature derived in a failure indicator FI^θ per Failure Mode (F_r). The obtained matrices FI^θ corresponds to:

- FI^{F_1} for ‘Tank leak’ (F_1) corresponding to a leak rate of 0.0002 L/s.
- FI^{F_2} for ‘Inflow valve obstruction’ (F_2) with a reduction of 0.0126 L/s in the inflow rate.
- FI^{F_3} for ‘Loss of heating power’ (F_3) corresponding to a reduction level of 400 Watts
- FI^{F_4} for ‘Outflow valve obstruction’ (F_4) with a reduction of 0.0126 L/s

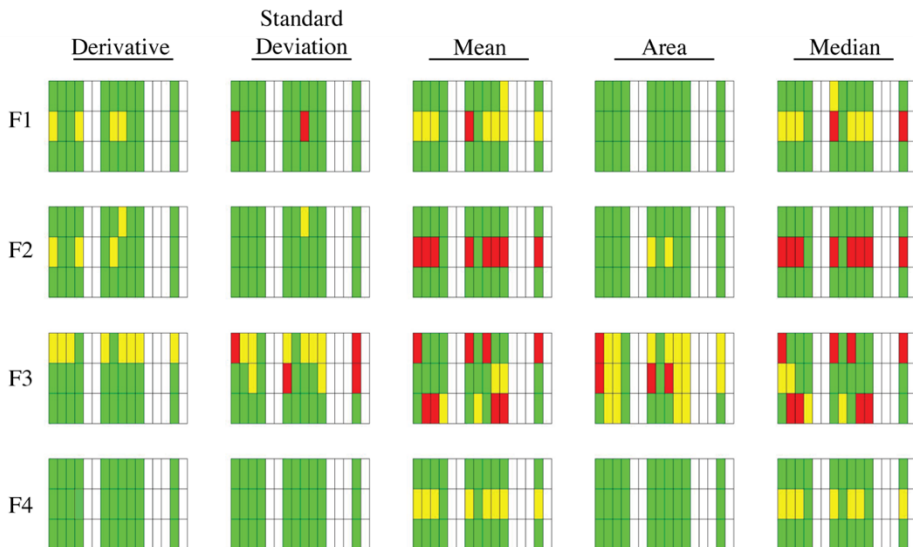


Figure 4.4. Failure indicators corresponding to the kettle model, evaluated with the same scenarios for the reference case and the failed ones

Table 4.4 Occurring SOM in the kettle model

SOM	S_{A_1}	S_{A_2}	S_{A_3}	S_{A_4}
1	OFF	OFF	OFF	OFF
2	OFF	OFF	ON	OFF
3	OFF	ON	ON	OFF
4	ON	ON	ON	OFF
7	OFF	ON	OFF	OFF
8	ON	OFF	ON	OFF
9	OFF	OFF	OFF	ON
10	OFF	OFF	ON	ON
11	OFF	ON	ON	ON
15	OFF	ON	OFF	ON

These matrices are illustrated in Figure 4.4.

Theoretically our kettle model can have 16 System Operation Modes (SOM). However, only 10 of them were actually activated due to control settings triggered by the analyzed scenarios. Table 4.4 presents the SOMs that occurred during our experiments. This will be used as input to each analysis.

In order to clarify the concept, let's take a particular example of one Failure Indicator. We took, for example, the case of Tank Leak (F_1) and its corresponding FI^{F_1} considering a_1 ('Derivative' signal feature) as reference, in order to present the meaning of every row (Sensor signals S_s) and every column (SOMs) of the failure indicators. As it can be seen in Figure 4.5, every cell is colored according to Eq. 4.20. The SOMs that

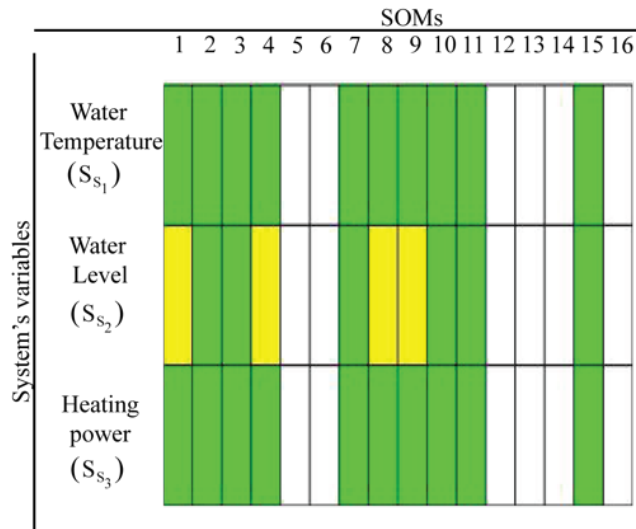


Figure 4.5. Explanation of the composition of the kettle's model failure indicator

Table 4.5. Effect size for F_1

		Derivative									
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 15
F1	S_{S_1}	0.11	0.02	0.03	0.01	0.12	0.00	0.07	0.02	0.07	0.09
	S_{S_2}	0.28	0.01	0.06	0.17	0.15	0.00	0.34	0.04	0.04	0.11
	S_{S_3}	0.00	0.03	0.05	0.04	0.00	0.00	0.00	0.01	0.03	0.00

		Standard deviation										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.07	0.03	0.07	0.02	0.06	0.00	0.12	0.02	0.03	0.13	
	S_{S_2}	0.36	0.01	0.02	0.76	0.11	0.00	0.33	0.00	0.04	0.08	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Mean										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.11	0.08	0.06	0.04	0.07	0.00	0.09	0.09	0.06	0.16	
	S_{S_2}	0.25	0.26	0.27	0.20	0.31	0.00	0.26	0.32	0.30	0.34	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Area										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.07	0.02	0.05	0.04	0.08	0.00	0.09	0.02	0.04	0.07	
	S_{S_2}	0.09	0.01	0.05	0.24	0.13	0.00	0.10	0.04	0.06	0.13	
	S_{S_3}	0.00	0.00	0.07	0.05	0.00	0.00	0.00	0.01	0.04	0.00	

		Median										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.11	0.08	0.08	0.03	0.10	0.00	0.08	0.07	0.15	0.08	
	S_{S_2}	0.27	0.25	0.27	0.13	0.35	0.00	0.23	0.24	0.24	0.29	
	S_{S_3}	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	

did not occur are represented by white cells in the resultant matrix for FI^{θ} .

The complete set of failure indicators is presented in Figure 4.4. We can observe that, in the case of Tank Leak (F_1), most of the cells that report statistical significance, with respect to the failure-free case, are those that correspond to Water Temperature (S_{S_1}) and Water Tank Level (S_{S_2}). This signal sources are also dominating as failure symptoms of the Inflow Valve Obstruction (F_2), and Outflow Valve Obstruction (F_4), where most of the symptoms occurred with a frequency lower than 80% in the analyzed experiments. For this reason, most of the cells are yellow colored. Only the ‘Water tank level’ (S_{S_2}) signal reported a significant difference with a frequency of symptom occurrence over 80% in the analyzed experiments, during F_1 and F_2 . It resulted in the red colored cells that can be observed, particularly, on the ‘Mean’ (a_3) and ‘Median’

(a_5) features corresponding to ‘Inflow valve obstruction’ (F_2) failure mode. ‘Loss of Heating Power’ (F_3) was the only failure mode that reported statistical significance in the ‘Heating power’ (S_{S_3}) signal.

In order to verify the importance of the observed symptoms, we need to calculate the effect size, which allows quantifying the difference between the two analyzed signal segments (i.e. the reference one and the failure-free). In our case, it allowed us to determine how large were the symptoms we reported in our failure indicator. We used Pearson’s correlation coefficient (r) as effect size index. It was computed with a Wilcoxon signed-rank test. Considering that we conducted 15 experiments to obtain our failure indicator, we estimated the effect size for every single experiment. Then, we computed the average of the absolute value of the effect size from all experiments.

Table 4.6. Effect size for F_2

		Derivative										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.09	0.01	0.05	0.03	0.13	0.00	0.14	0.01	0.04	0.14	
	S_{S_2}	0.11	0.01	0.09	0.33	0.10	0.00	0.10	0.02	0.05	0.10	
	S_{S_3}	0.00	0.02	0.06	0.06	0.00	0.00	0.00	0.03	0.01	0.00	

		Standard deviation										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.14	0.02	0.07	0.11	0.09	0.00	0.14	0.01	0.09	0.15	
	S_{S_2}	0.00	0.02	0.07	0.64	0.06	0.00	0.01	0.00	0.07	0.11	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Mean										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.07	0.07	0.07	0.13	0.07	0.00	0.10	0.09	0.17	0.10	
	S_{S_2}	0.49	0.54	0.32	0.31	0.50	0.00	0.48	0.52	0.33	0.50	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Area										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.11	0.01	0.07	0.02	0.07	0.00	0.10	0.00	0.06	0.08	
	S_{S_2}	0.12	0.03	0.09	0.33	0.09	0.00	0.15	0.04	0.07	0.15	
	S_{S_3}	0.00	0.01	0.06	0.04	0.00	0.00	0.00	0.01	0.05	0.00	

		Median										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.15	0.08	0.10	0.14	0.11	0.00	0.14	0.09	0.12	0.12	
	S_{S_2}	0.50	0.52	0.36	0.32	0.44	0.00	0.52	0.53	0.43	0.49	
	S_{S_3}	0.00	0.00	0.00	0.24	0.00	0.00	0.00	0.00	0.00	0.00	

Table 4.7. Effect size for F_3

		Derivative										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.24	0.17	0.17	0.19	0.27	0.00	0.25	0.18	0.17	0.27	
	S_{S_2}	0.10	0.04	0.07	0.11	0.07	0.00	0.09	0.08	0.07	0.06	
	S_{S_3}	0.00	0.04	0.05	0.16	0.00	0.00	0.00	0.12	0.07	0.00	

		Standard deviation										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.21	0.19	0.19	0.10	0.27	0.00	0.27	0.17	0.21	0.24	
	S_{S_2}	0.01	0.02	0.18	0.02	0.27	0.00	0.00	0.01	0.20	0.31	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Mean										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.35	0.10	0.11	0.23	0.29	0.00	0.29	0.08	0.14	0.33	
	S_{S_2}	0.08	0.13	0.16	0.15	0.14	0.00	0.13	0.17	0.20	0.13	
	S_{S_3}	0.00	0.99	0.99	0.68	0.00	0.00	0.00	0.99	0.99	0.00	

		Area										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.32	0.18	0.16	0.15	0.32	0.00	0.34	0.17	0.20	0.27	
	S_{S_2}	0.30	0.19	0.16	0.02	0.27	0.00	0.33	0.19	0.19	0.29	
	S_{S_3}	0.00	0.16	0.16	0.44	0.00	0.00	0.00	0.20	0.19	0.00	

		Median										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.28	0.07	0.10	0.22	0.34	0.00	0.34	0.10	0.10	0.29	
	S_{S_2}	0.13	0.15	0.17	0.15	0.17	0.00	0.15	0.15	0.16	0.09	
	S_{S_3}	0.00	0.99	0.99	0.83	0.00	0.00	0.00	0.99	0.99	0.00	

The average effect size corresponding to F_1 , F_2 , F_3 , and F_4 are presented in the Table 4.5, Table 4.6, Table 4.7 and Table 4.8 respectively. In these tables, we highlighted bold the If we compare the symptoms presented by the failure indicator with their average effect size, only 21.2% of the total of symptoms present a medium effect (i.e. $r > .3$), while just the 18% of the symptoms present a large effect (i.e. $r > .5$) (see Table 4.5, Table 4.6, Table 4.7 and Table 4.8). From all symptoms that presented a large effect, 55.5% correspond to ‘Loss of heating power’ (F_3), particularly to the ‘Heating power’ signal (S_{S_3}). The remaining 44.4% of the symptoms correspond to obstruction in the inflow valve (F_2). Similarly, 29.6% of the symptoms of the tank leak (F_1), 33.3% of the ‘Inflow valve obstruction’ (F_2) and 37.03% of the ‘Loss of heating power’ (F_3) had a medium effect. From the analyzed failure modes, ‘Loss of heating power’ (F_3)

presented stronger failure symptoms and ‘Outflow valve obstruction’ (F_4) presented weaker symptoms. It is important to recall that the red cells presented in Figure 4.4 only represent the frequency on which statistical significance was reported per cell observed symptoms from Figure 4.4. We interpret it as the frequency of occurrence of symptoms, per signal, per SOM.

The preponderance of small effects can be adjudged to the level of the injected failures (see section 4.2.1). Anywise, the SOM-based signal segmentation analysis managed to present failure symptoms for all the analyzed failure modes, while the whole length signal analysis only presented symptoms for F_3 (see section 4.2.1). We can infer from it, that SOM-based signal segmentation really accentuates failure manifestations and, thus, it seems to be more effective for detecting failure symptoms than the analysis of the whole length signal.

Table 4.8. *Effect size for F_4*

		Derivative										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.09	0.03	0.07	0.24	0.10	0.00	0.13	0.01	0.05	0.09	
	S_{S_2}	0.11	0.03	0.06	0.15	0.08	0.00	0.12	0.00	0.07	0.09	
	S_{S_3}	0.00	0.05	0.07	0.81	0.00	0.00	0.00	0.02	0.05	0.00	

		Standard deviation										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.08	0.02	0.06	0.25	0.10	0.00	0.11	0.01	0.09	0.11	
	S_{S_2}	0.01	0.02	0.05	0.73	0.18	0.00	0.00	0.02	0.10	0.12	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Mean										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.09	0.07	0.10	0.23	0.09	0.00	0.15	0.05	0.12	0.10	
	S_{S_2}	0.26	0.15	0.10	0.27	0.17	0.00	0.18	0.23	0.15	0.19	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Area										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.08	0.00	0.07	0.34	0.10	0.00	0.10	0.01	0.06	0.09	
	S_{S_2}	0.11	0.01	0.07	0.31	0.14	0.00	0.12	0.02	0.06	0.11	
	S_{S_3}	0.00	0.00	0.07	0.08	0.00	0.00	0.00	0.01	0.03	0.00	

		Median										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.10	0.11	0.09	0.26	0.16	0.00	0.10	0.08	0.14	0.12	
	S_{S_2}	0.22	0.25	0.13	0.42	0.23	0.00	0.25	0.26	0.14	0.20	
	S_{S_3}	0.00	0.00	0.00	0.36	0.00	0.00	0.00	0.00	0.00	0.00	

4.4.2 Analysis of the discriminant power of the failure indicators

This analysis aims to determine to what extent the derived indicators from a particular failure mode are similar to the ones presented by another failure mode. For this purpose, the percentage of cells with the same colors for two different failures but corresponding to the same data feature were estimated (see 4.21). The results are presented in Table 4.9. An overall analysis of these results, led us to conclude that the failure mode of ‘Loss of heating power’ (F_3) presents the highest discriminant power. This failure mode presented the lowest levels of cells in common in all the analyzed data features. The rationale behind this result is this failure mode contained symptoms with higher frequency (see Figure 4.4) and with higher effect size (see Table 4.7). On the contrary, ‘Tank leak’ (F_1) was the failure mode that presented the highest levels of similarity in four out of five of the data features considering. This means it this is the failure mode with the lowest discriminant power.

The ‘Area’ (a_4) signal feature presented the lowest levels of similarity between all the analyzed signal features by reaching a minimum of 58.3% of cells in common (for F_3 versus F_1 ; F_3 versus F_2 ; and, F_3 versus F_4). The highest level of similarity was observed by ‘Standard deviation’ (a_2) signal feature reaching 97.9% similarity of the cells for (F_2) and (F_4).

Symptoms shown in Figure 4.4 occurred with a medium frequency (as indicated by the yellow color) meaning that the discriminant power obtained is weak for most of the failure modes, except for F_3 . It causes that the observed combination of symptoms hardly contributes to failure differentiation.

The present analysis considered the same use and operative conditions for comparing datasets of failure-free and failed operation modes. These conditions guarantee that any deviation of the reference and the failed signals can only be caused by the injected failure, as there are no external disturbances. These results

Table 4.9. Similarity level between failure indicators

		Derivative			
		F1	F2	F3	F4
F1		100.0	95.8	75.0	91.7
F2		95.8	100.0	79.2	91.7
F3		75.0	79.2	100.0	83.3
F4		91.7	91.7	83.3	100.0

		Standard deviation			
		F1	F2	F3	F4
F1		100.0	91.7	68.8	93.8
F2		91.7	100.0	77.1	97.9
F3		68.8	77.1	100.0	75.0
F4		93.8	97.9	75.0	100.0

		Mean			
		F1	F2	F3	F4
F1		100.0	83.3	62.5	95.8
F2		83.3	100.0	60.4	83.3
F3		62.5	60.4	100.0	64.6
F4		95.8	83.3	64.6	100.0

		Area			
		F1	F2	F3	F4
F1		100.0	95.8	58.3	100.0
F2		95.8	100.0	58.3	95.8
F3		58.3	58.3	100.0	58.3
F4		100.0	95.8	58.3	100.0

		Median			
		F1	F2	F3	F4
F1		100.0	85.4	64.6	91.7
F2		85.4	100.0	60.4	83.3
F3		64.6	60.4	100.0	66.7
F4		91.7	83.3	66.7	100.0

demonstrate that SOM-based signal segmentation enables to observe more symptoms than the analysis of unsegmented signals. Nevertheless, some of these symptoms are still weak, as most of them present a low effect size and a weak discriminant power. In the next section, we will evaluate the sensitivity of SOM based failure manifestations when subjected to changes of use and operative conditions.

4.4.3 Sensitivity of SOM based failure diagnosis to variations of operating conditions

Our previous analysis considered the same use and operative conditions for the datasets with failure-free and failed operations. To evaluate the sensitivity of SOM based failure diagnosis to variations of operating conditions, we selected scenarios with varying external conditions. In this analysis, scenarios were randomly selected from the 15 sets previously defined. We aim to evaluate to what extent our SOM based failure analysis is affected by dissimilar operative and use conditions. Fifteen FI matrices, per failure mode F_r , at the same failure level, were generated in order to derive their reference indicators FI^θ . We considered the same signal features as in section 4.2.2: ‘Derivative’ (a_1), ‘Standard deviation’ (a_2), ‘Mean’ (a_3), ‘Area’ (a_4) and ‘Median’ (a_5). The matrices FI^{F_1} for ‘Tank leak’, FI^{F_2} for ‘Inflow valve obstruction’, FI^{F_3} for ‘Loss of heating power’ and FI^{F_4} for ‘Outflow valve obstruction’ are presented in Figure 4.6.

The failure indicators present similar patterns to the ones observed in Figure 4.4. These patterns are the result of the injected failures showing a consistent manifestation. As in Figure 4.4, most cells that report statistical significance, with respect to the failure-free case, correspond to ‘Water temperature’ (S_{S_1}) and ‘Water tank level’ (S_{S_2}), in all the

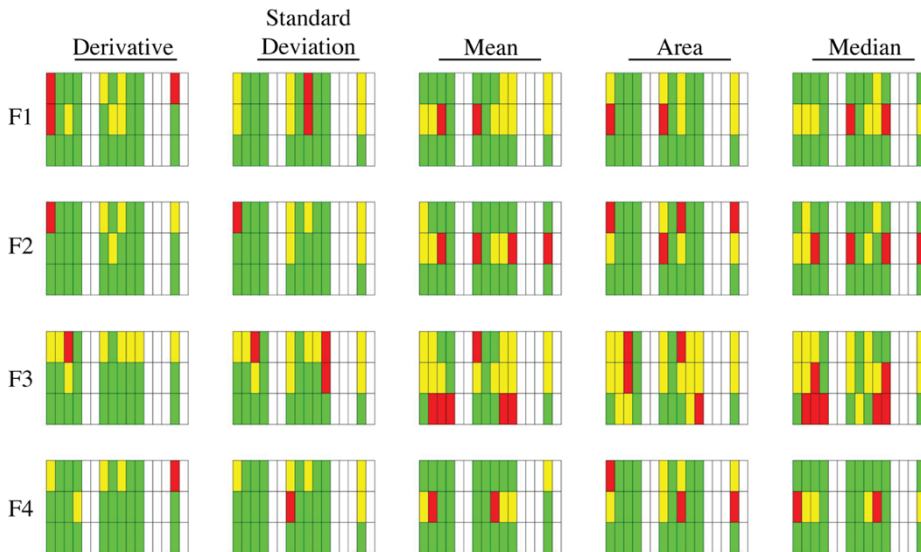


Figure 4.6. Failure indicators corresponding to the kettle model

analyzed failure modes. In Figure 4.6, as in Figure 4.4, most of the symptoms occurred in less than 80% of the total of the experiments. Hence, most cells are yellow colored (according to Eq. 4.20). However, unlike the indicators presented in Figure 4.4, we can observe a greater number of symptoms whose frequency of occurrence is higher than 80% (i.e. symptoms that were observed in more than 80% of the experiments conducted). That's the reason why the number of red cells is greater in Figure 4.6 than in Figure 4.4. Figure 4.7 presents the similarity levels (measured in %) of the obtained failure indicators. It compares the failure indicators derived from scenarios with the same use and operative conditions with the failure indicators derived from randomly selected scenarios with different use and operative conditions. We evaluated their level of similarity by computing the percentage of common cells of failure indicators. For example, failure indicator corresponding to F_1 and 'Derivative' feature (a_1) obtained by comparing same scenarios from failure-free and failed cases, was compared with the failure indicator corresponding to F_1 and 'Derivative' feature (a_1) obtained by randomly selected scenarios.

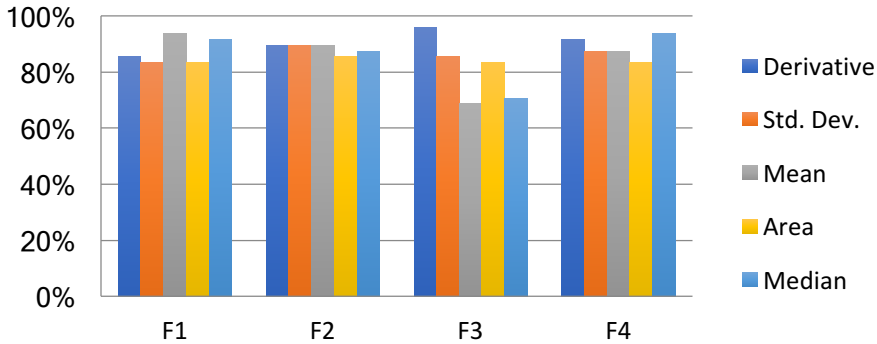


Figure 4.7. Similarity level (%) between the failure indicators derived with the scenarios, and the ones derived with different scenarios

Based on Figure 4.7, we can observe that, there is a high level of similarity between the indicators presented in the Figure 4.4 and those presented in the Figure 4.6. 17 out of 20 of the derived indicators present a similarity level over the 80%. However, some combinations of failure modes and data features such as the 'Mean' (a_3), or the 'Median' (a_5) of 'Loss of heating power' (F_3) reached values up to 68.75% and 70.83%, respectively. In general terms, an increase of the number of cells that present statistical significance is observed when varying the use and operative conditions. This result implies that false failure alarms can occur in the SOM-based signal segments due to variations caused by external factors. Nevertheless, the failure signatures can still be observed, (i.e. the pattern of symptoms observed in FI caused by a particular failure mode), despite the presence of false failure symptoms.

The analysis of the averaged effect size reveals that 44.38% of the observed symptoms correspond to a medium effect, while just the 6.63% correspond to a large effect. Results of the effect size are presented in Table 4.10, Table 4.11, Table 4.12 and Table 4.13. We highlighted bold the cells that presented statistical significance in Figure 4.6.

These results show an increase of 23.13% for the number of symptoms corresponding to medium effect (in proportion to the total of symptoms observed) and a reduction of 7.54% for the large effect symptoms with respect to the indicators presented in Figure 4.4. The variations caused by changes of use and operative conditions mostly presented a medium size effect. It implies that the SOM-based signal segmentation is sensitive to these types of changes. The increase of the frequency of occurrence of symptoms observed in Figure 4.6, particularly of yellow cells, demonstrates an increase of occurrence of false positive cases. The variations of use and operative conditions affected both the magnitude and frequency of occurrence of deviations of SOM-based signal segments with respect to the failure-free case.

The analysis of the discriminant power of failure indicators was also conducted for randomly selected scenarios. Results are presented in Table 4.14. This analysis revealed an increased level of similarity for all data features compared to the analysis of the same

Table 4.10. Average effect size for F_1 with randomly selected scenarios

		Derivative										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.30	0.03	0.12	0.31	0.32	0.00	0.31	0.05	0.09	0.30	
	S_{S_2}	0.34	0.01	0.12	0.39	0.09	0.00	0.34	0.02	0.10	0.11	
	S_{S_3}	0.00	0.01	0.06	0.21	0.00	0.00	0.00	0.02	0.04	0.00	

		Standard deviation										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.35	0.05	0.10	0.08	0.30	0.00	0.32	0.00	0.10	0.30	
	S_{S_2}	0.44	0.00	0.10	0.32	0.33	0.00	0.39	0.01	0.10	0.29	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Mean										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.12	0.13	0.12	0.27	0.16	0.00	0.16	0.14	0.16	0.13	
	S_{S_2}	0.21	0.21	0.24	0.16	0.29	0.00	0.20	0.23	0.25	0.27	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Area										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.31	0.03	0.09	0.31	0.35	0.00	0.34	0.02	0.09	0.40	
	S_{S_2}	0.34	0.02	0.08	0.31	0.31	0.00	0.36	0.03	0.13	0.36	
	S_{S_3}	0.00	0.02	0.09	0.29	0.00	0.00	0.00	0.02	0.14	0.00	

		Median										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F1	S_{S_1}	0.17	0.15	0.14	0.16	0.16	0.00	0.08	0.17	0.19	0.12	
	S_{S_2}	0.25	0.19	0.25	0.16	0.26	0.00	0.26	0.25	0.27	0.24	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

Table 4.11. Average effect size for F_2 with randomly selected scenarios

		Derivative										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.29	0.03	0.08	0.05	0.31	0.00	0.33	0.03	0.05	0.35	
	S_{S_2}	0.12	0.05	0.10	0.34	0.08	0.00	0.12	0.00	0.08	0.14	
	S_{S_3}	0.00	0.02	0.04	0.05	0.00	0.00	0.00	0.02	0.03	0.00	

		Standard deviation										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.26	0.01	0.08	0.19	0.32	0.00	0.37	0.03	0.09	0.29	
	S_{S_2}	0.01	0.01	0.07	0.42	0.30	0.00	0.00	0.02	0.12	0.37	
	S_{S_3}	0.00	0.00	0.00	0.28	0.00	0.00	0.00	0.00	0.00	0.00	

		Mean										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.18	0.19	0.16	0.06	0.16	0.00	0.19	0.16	0.15	0.19	
	S_{S_2}	0.33	0.30	0.37	0.32	0.47	0.00	0.36	0.27	0.40	0.50	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Area										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.30	0.02	0.08	0.03	0.37	0.00	0.33	0.00	0.09	0.31	
	S_{S_2}	0.34	0.03	0.07	0.29	0.37	0.00	0.33	0.04	0.03	0.37	
	S_{S_3}	0.00	0.03	0.08	0.03	0.00	0.00	0.00	0.00	0.07	0.00	

		Median										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F2	S_{S_1}	0.11	0.16	0.17	0.11	0.08	0.00	0.14	0.17	0.15	0.18	
	S_{S_2}	0.28	0.26	0.31	0.32	0.50	0.00	0.34	0.27	0.40	0.53	
	S_{S_3}	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	

scenarios (See Table 4.9). This can be seen in the increased value of the minimum similarity values reached by each feature:

- In the ‘Derivative’ feature (a_1) from 75.0% to 81.3% (similarity between F_1 and F_3).
- In the ‘Standard deviation’ feature (a_2), from 68.8% to 81.3% (similarity between F_1 and F_3).
- In the ‘Mean’ feature (a_3), from 60.4% to 70.8% (similarity between F_2 and F_3).
- In the ‘Area’ feature (a_4), from 58.3% to 66.7% (similarity between F_3 and F_4).
- In the ‘Median’ feature (a_5) from 60.4% to 68.8% (similarity between F_3 and F_2).

A slight reduction on the maximum levels of similarity was also observed though. This can be seen in the maximum similarity values reached by each feature:

In the ‘Derivative’ feature (a_1) from 95.8% to 91.7% (similarity between F_1 and F_2).

- In the ‘Standard deviation’ feature (a_2), from 97.9% (F_2 and F_4) to 91.7% (F_1 versus F_2 and F_1 versus F_4)
- In the ‘Mean’ feature (a_3), from 95.8% to 85.4% (similarity between F_1 and F_4).
- In the ‘Area’ feature (a_4), from 95.8% to 91.7% (similarity between F_1 and F_2).
- In the ‘Median’ feature (a_5) maximum similarity stayed the same.

Although variations of use and operative conditions negatively influenced the similarity level of the failure indicators, it did not affect considerably their discriminant power. In this study, it was shown that variations of use and operative conditions have an impact on the frequency and magnitude of signal deviations of the SOM-based signal segments.

Table 4.12. Average effect size for F_3 with randomly selected scenarios

		Derivative										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.30	0.19	0.27	0.29	0.38	0.00	0.31	0.17	0.23	0.27	
	S_{S_2}	0.08	0.08	0.14	0.02	0.12	0.00	0.09	0.07	0.19	0.07	
	S_{S_3}	0.00	0.07	0.08	0.19	0.00	0.00	0.00	0.06	0.03	0.00	
		Standard deviation										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.29	0.19	0.29	0.14	0.28	0.00	0.31	0.18	0.26	0.30	
	S_{S_2}	0.00	0.00	0.26	0.31	0.32	0.00	0.01	0.01	0.27	0.31	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
		Mean										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.20	0.17	0.20	0.22	0.32	0.00	0.27	0.22	0.20	0.27	
	S_{S_2}	0.35	0.37	0.35	0.22	0.19	0.00	0.36	0.34	0.34	0.24	
	S_{S_3}	0.00	0.99	0.99	0.94	0.00	0.00	0.00	0.99	0.99	0.00	
		Area										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.28	0.19	0.25	0.21	0.29	0.00	0.35	0.17	0.25	0.32	
	S_{S_2}	0.33	0.17	0.25	0.33	0.29	0.00	0.32	0.18	0.27	0.27	
	S_{S_3}	0.00	0.17	0.27	0.35	0.00	0.00	0.00	0.16	0.24	0.00	
		Median										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F3	S_{S_1}	0.29	0.15	0.20	0.19	0.27	0.00	0.28	0.21	0.18	0.28	
	S_{S_2}	0.39	0.41	0.30	0.22	0.21	0.00	0.40	0.40	0.28	0.23	
	S_{S_3}	0.00	0.99	0.99	0.94	0.00	0.00	0.00	0.99	0.99	0.00	

Table 4.13. Average effect size for F_4 with randomly selected scenarios

		Derivative										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.26	0.02	0.06	0.08	0.28	0.00	0.25	0.01	0.08	0.26	
	S_{S_2}	0.08	0.00	0.12	0.35	0.15	0.00	0.11	0.01	0.09	0.19	
	S_{S_3}	0.00	0.05	0.02	0.80	0.00	0.00	0.00	0.05	0.06	0.00	

		Standard deviation										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.28	0.01	0.11	0.29	0.27	0.00	0.28	0.03	0.07	0.27	
	S_{S_2}	0.00	0.02	0.12	0.30	0.35	0.00	0.02	0.01	0.10	0.28	
	S_{S_3}	0.00	0.00	0.00	0.17	0.00	0.00	0.00	0.00	0.00	0.00	

		Mean										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.14	0.20	0.17	0.05	0.10	0.00	0.14	0.19	0.09	0.12	
	S_{S_2}	0.44	0.41	0.14	0.29	0.12	0.00	0.45	0.41	0.17	0.16	
	S_{S_3}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	

		Area										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.33	0.03	0.12	0.28	0.38	0.00	0.37	0.06	0.08	0.36	
	S_{S_2}	0.31	0.02	0.09	0.21	0.34	0.00	0.32	0.00	0.08	0.30	
	S_{S_3}	0.00	0.06	0.06	0.05	0.00	0.00	0.00	0.02	0.10	0.00	

		Median										
		Signal	SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
F4	S_{S_1}	0.13	0.17	0.15	0.10	0.15	0.00	0.13	0.16	0.13	0.10	
	S_{S_2}	0.47	0.40	0.16	0.22	0.19	0.00	0.39	0.44	0.15	0.17	
	S_{S_3}	0.00	0.00	0.00	0.27	0.00	0.00	0.00	0.00	0.00	0.00	

It implies that these deviations can be misinterpreted as failure symptoms affecting failure diagnosis.

4.4.4 Findings of the pilot study

The pilot study presented in this section aimed to evaluate the effect of segmenting signals based on SOMs for conducting failure analysis. It was found that signal segmentation has a positive effect on failure detection. It enabled the observation of symptoms corresponding to all the injected failures, while the analysis of the whole length signal only revealed symptoms for one failure mode only, i.e. the loss of heating power. It was also found that failure modes present a pattern of symptoms that can be represented in a failure indicator matrix, *FI*. This representation could be used for failure characterization. Nevertheless, the study revealed that the failure indicator matrix

has a weak discriminant power, since in all the analyzed cases prevailed symptoms that presented medium frequency of occurrence (yellow cells). The analysis of variations of use and operative conditions revealed that failure analysis through SOM-based signal segmentation is sensitive to external disturbances. The use and operative conditions increased the frequency of symptoms for randomly selected scenarios. It implies, that differences on the use and operative conditions between the compared indicators increase the false failure alarms, affecting the reliability of the results.

The studied signal features revealed failures in the case of SOM-based signal segmentation analysis. All of them presented different patterns of failure symptoms in their corresponding *FI* matrix. However, the mean and median features had similar patterns with an average similarity of 92% in the investigated cases (see Figure 4.4 and Figure 4.6). Both features are used interchangeably, but median feature is more robust against outliers. In our study, the benefits of the median feature were not observable. Moreover, the use of both features is redundant and it increased the computational efforts.

So far, we have analyzed an idealistic case of a kettle that was subjected to variations of use and environmental conditions. In this study, the ambient temperature as external factor was systematically manipulated. Nevertheless, some other factors, such as fluctuations on the environmental temperatures, variations on light radiation, among others can also influence the way in which failures are observed. Moreover, technical issues concerning system instrumentation, such as the quality of the data transmission, sensor saturation, and electromagnetic fields can also have an effect on the measured signals, and thus, on failure analysis results.

Table 4.14. Similarity level between failure indicators for randomly selected scenarios

		Derivative			
		F1	F2	F3	F4
F1		100.0	91.7	81.3	87.5
F2		91.7	100.0	85.4	91.7
F3		81.3	85.4	100.0	85.4
F4		87.5	91.7	85.4	100.0

		Standard deviation			
		F1	F2	F3	F4
F1		100.0	91.7	81.3	91.7
F2		91.7	100.0	85.4	95.8
F3		81.3	85.4	100.0	85.4
F4		91.7	95.8	85.4	100.0

		Mean			
		F1	F2	F3	F4
F1		100.0	87.5	79.2	85.4
F2		87.5	100.0	70.8	83.3
F3		79.2	70.8	100.0	68.8
F4		85.4	83.3	68.8	100.0

		Area			
		F1	F2	F3	F4
F1		100.0	91.7	68.8	89.6
F2		91.7	100.0	68.8	89.6
F3		68.8	68.8	100.0	66.7
F4		89.6	89.6	66.7	100.0

		Median			
		F1	F2	F3	F4
F1		100.0	91.7	68.8	89.6
F2		91.7	100.0	68.8	83.3
F3		68.8	68.8	100.0	64.6
F4		89.6	83.3	64.6	100.0

The pilot study was conducted on a relatively simple system that included a few system actuators and sensors, and a limited number of system operation modes. It is important to analyze the effects of a higher number of actuators and sensed signals, as it can have significant implications on the potential application of SOMs in failure diagnosis and forecasting. These factors were not investigated in this pilot study. Considering the afore-mentioned issues, the next section presents a real-life implementation of failure diagnosis in the instrumented testbed, with the aim to determine the implications of disturbances caused by external and environmental factors. In this real-life case study, we aim to investigate not only the way in which failure manifest due to uncontrollable factors, but also the effects of increasing system complexity.

4.5 A real-life study of the SOMs-oriented failure analysis methodology

4.5.1 Description of the experiment

To analyze failure manifestations in a realistic environment, we used the instrumented testbed introduced in section 3.4 as a means for studying failures. The testbed has 21 sensor signals (S_S) and six actuator signals (S_A). The goal of this experiment was to evaluate the three objectives of this research cycle stated in section 4.1 by making use of data collected in the instrumented testbed. For this purpose, we injected three different failure modes into the system:

- ‘Tank leak’ (F_1): To inject the failure of a leak in a controlled manner, a drain valve was installed on one of the walls of the tank. It was placed close to its bottom and below the inlet and the outlet valves. This installation enabled the manipulation of the outflow of leaking water. For this purpose, 23 different opening levels of the valve were evaluated in order to be able to replicate the failure with different intensities. Each opening level was recorded in a separate data set.
- ‘Irrigation pipe blocked’ (F_2): The irrigation hole located next to the ‘Soil moisture sensor’ ($S_{S_{17}}$), corresponding to plant bed 2, was obstructed by teflon tape. The manipulated variable was the flowrate of irrigation (‘Electro valve plant bed 2’ S_{A_6}).
- ‘Irregular fan operation’ (F_3): A resistance that reduces the electrical current that feeds the fan was installed, in order to modify the regular speed of rotation of the inlet air fan. The manipulated variable was the RPM of the fan (‘Fan In’ S_{A_4}).

A set of data obtained during the regular operation of the system, in different experiments, and in different periods of the day, were collected and stored as reference (A^0) in a single dataset. For all the experiments, we obtained the corresponding ‘Failure Indicator’ (FI) of every Failure Mode (F_r), in order to understand the characteristics of failure manifestations. We evaluated the coherence of the obtained indicators by analyzing 9400 independent data samples with 5 different failure injection processes for each Failure Mode (F_r) during 5 days.

External factors, such as sunlight, ambient temperature and inflowing water temperature can have a strong influence on the system operation and affect the obtained results. Therefore, in our testbed setup we monitored these external factors and considered them in our analysis process. As for the pilot study presented in section 4.4, data from ‘Observed (failed) operation’ (A^O) were compared with data of ‘Reference (failure-free) operation’ (A^θ).

4.5.2 Deriving failure indicator matrix

In this experiment, we also used Kruskal-Wallis test to evaluate the deviation of signals for deriving the failure indicator. The different datasets corresponding to reference data were merged together into a single dataset A^θ . One dataset per failure mode ($A_{F_r}^O$) were generated as follows:

- $A_{F_1}^O$ which is the dataset corresponding to ‘Tank leak’ (F_1);
- $A_{F_2}^O$ which is the dataset corresponding to ‘Irrigation pipe blocked’ (F_2)
- $A_{F_3}^O$ which is the dataset corresponding to ‘Irregular fan operation’ (F_3).

We randomly selected the data samples from $A_{F_r}^O$ and A^θ that we used to obtain failure indicator by their pair-wise comparisons. A total of 500 FI matrices were generated per F_r for deriving its corresponding FI^θ by following the process described in section 4.3.3. This process was conducted for every signal feature considered.

Four signal features (a_m) considered for the real-life study: (I) ‘Derivative’ (a_1), (ii) ‘Standard deviation’ (a_2), (iii) ‘Median’ (a_3), (iv) ‘Area’ (a_4). In the pilot study, we found that the mean and median features had similar results. In this experiment, we only included median feature because it performs better than mean in presence of outliers and can be used for not normal distributions. Median is the statistic parameter that supplies the mean in robust statistics. Theoretically, our testbed can have 64 System Operation Modes (SOM), however, only 10 of them were actually activated during our experiment. The activated SOMs are presented in Table 4.15.

The obtained matrices FI^θ : FI^{F_1} for ‘Tank leak’, FI^{F_2} for ‘Irrigation pipe blocked’ and, FI^{F_3} for ‘Irregular fan operation’ can be seen in Figure 4.8. The SOMs that did not occur, are represented through white cells in the resultant matrices FI^θ . Unlike the pilot

Table 4.15. Occurring system operation modes

SOM	S_{A_1}	S_{A_2}	S_{A_3}	S_{A_4}	S_{A_5}	S_{A_6}
9	Off	Off	Off	On	Off	Off
11	Off	On	Off	On	Off	Off
12	Off	On	On	On	Off	Off
13	Off	Off	On	On	Off	Off
15	Off	On	On	On	Off	Off
33	Off	Off	Off	Off	Off	On
41	Off	Off	Off	On	Off	On
43	Off	On	Off	On	Off	On
45	Off	Off	On	On	Off	On
47	Off	On	On	On	Off	On

study (See section 4.4), in this experiment, we found that there are also some SOMs that were triggered by the emerging failures, pushing the system into an “abnormal” operation mode (i.e. a combination of component operation modes not typical under regular circumstances). These will be called Failure Induced Operation Mode (FIOM) and cells corresponding to FIOM were highlighted in light blue in Figure 4.8.

The results show that only a few cells presented statistical significance in the case of ‘Tank leak’ (F_1) and ‘Derivative’ feature (a_1). The rest of data features, however, had statistical significance of the deviations in most of the signals, particularly in SOMs ζ_9 , ζ_{11} , ζ_{13} and ζ_{41} . ‘Median’ (a_3) and ‘Area’ (a_4) had more red cells than the other signal features. A similar pattern is observed in ‘Irrigation pipe blocked’ (F_2) and ‘Irregular fan operation’ (F_3) failure modes. There, ‘Derivative’ feature (a_1) hardly presented statistical significance with medium and high frequency, while ‘Median’ (a_3) and ‘Area’ (a_4) features presented a large number of cells with significant difference between the failure-free signal segments and the failed ones.

We estimated the average effect size of the obtained indicators in order to determine the effect size of the injected failures. We found that 19.15% of cells that reported statistical significance correspond to medium effect size, while 65.13% correspond to high effect size. Results are presented in Table 4.16, Table 4.17 and Table 4.18. We highlighted bold the cells that presented statistical significance in Figure 4.8. From all cells that presented a large effect, 38.52% of these cells corresponds to ‘Tank leak’ (F_1), 27.35% occurred on ‘Irrigation pipe blocked’ (F_2) while 34.11% were reported on ‘Irregular fan operation’ (F_3). Likewise, 19.15% of the observed symptoms presented medium effect size. These symptoms were distributed as follows: 40% occurred in F_1 , 21% corresponds to F_2 and 39% occurred on F_3 . Medium and large effect size symptoms in conjunction counts for 84.28% of the observed symptoms throughout all analyzed indicators. It implies a preponderance of large size symptoms.

Even though most cells, that report statistical significance, present medium to large

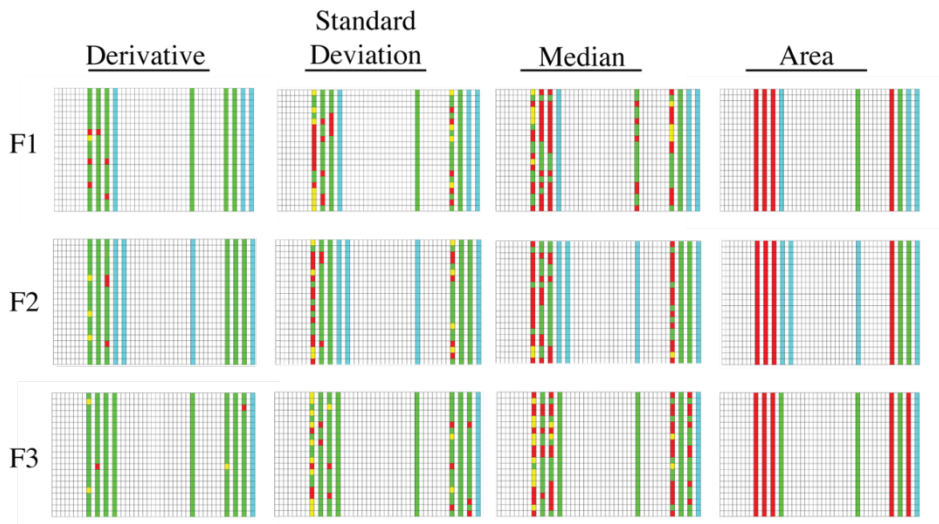


Figure 4.8. Failure indicators for the greenhouse testbed

effect size, the previously described results cannot be only adjudged to the injected failure modes. For instance, it is not likely that ‘Tank leak’ (F_1) causes a significant difference in the infrared light in plant bed 2 signal (‘Sunlight Sensor’ $S_{S_{19}}$), or in the rotation speed of the inlet fan (‘RPM sensor’ $S_{S_{13}}$). The high number of cells that present statistical significance led us to infer that the effect of environmental factors, as well as operative issues (such as sensor saturation) have an important impact on the analyzed signal segments. A deeper analysis of the signal segments provided insights into the reasons of the unexpected presence of symptoms in failure indicators. We found three main critical factors that can provide explanation to the observed failure indicators: (I) loss of information due to signal segmentation, (II) the environmental and external disturbances and (III) the compensation effect of the control.

1. Loss of information due to signal segmentation:

SOM-based signal segmentation assumes that failure symptoms are stronger in certain SOMs. However, we found that it sensitive to the duration of SOMs and to failures effecting system operation with a delay. For instance, Figure 4.9(a) shows the signal of ‘Soil moisture sensor’ from plant bed 2, ($S_{S_{17}}$), for failure-free operation and during the failed operation with blocked irrigation pipe, (F_2). Figure 4.9(b) presents the boxplots of these data sets. Based on these box plots it is reasonable to think that there is an effect of F_2 on this sensor signal. Nevertheless, the failure diagnosis by SOM based signal segmentation did not present any failure symptom in any of the SOMs where the irrigation valve of plant bed 2 (‘Electro valve plant bed 2’ S_{A_6}), was active (i.e. in the SOMs ζ_{33} , ζ_{41} , ζ_{43} , ζ_{45} and ζ_{47} (see *Table 4.15*).

We studied the case where only the inlet fan (‘Fan-in’ S_{A_4}) and the irrigation valve of plant bed 2 (‘Electro valve plant bed 2’ S_{A_6}) were active (i. e. ζ_{41} in *Table 4.15*) in order to determine the causes of the lack of symptom. Figure 4.10 presents the soil humidity signal (‘Soil moisture sensor’ from plant bed 2 $S_{S_{17}}$) corresponding to both the failed and the failure-free cases. We chose one signal fragment that effectively illustrates the point we explain below. The yellow curve corresponds to the failure-free cases and the green one corresponds to the failed cases. We highlighted in blue the signal segments from the failure-free operation corresponding to the SOM ζ_{41} . Similarly, we highlighted in red the signal segments from F_2 corresponding to the SOM ζ_{41} . Figure 4.10 shows that SOM ζ_{41} is located at the low peaks of soil humidity signals. However, variation caused by F_2 is manifested on the top peaks of the signals (as the failure-free cases reach higher measurements than the failed ones) that are not captured during the occurrence of the SOM ζ_{41} .

Signal segmentation prevented the failure to be observed in the SOM ζ_{41} . It caused an opposite effect to the one desired. Signal segmentation pruned the signal up to the point that even the failure manifestation was lost. This failure is still manifested in SOM ζ_9 and SOM ζ_{11} , thought. It was also explored that SOM ζ_{41} of the failure free cases only occurred six times in the signal fragment presented in Figure 4.10, while it occurred 16 times for the failed cases. This finding shows that failure analysis with SOM based segmentation can be sensitive to failures effecting system operation with a delay. It causes failure to manifest not in the expected SOM, which in turn may lead to misinterpretations. Variations on the frequency and duration of SOM suggests that failures influence the control actions and behavior of the system.

Table 4.16. Average effect size of F_1 for the testbed case

SOM							SOM						
Signal	9	11	13	33	41	43	Signal	9	11	13	33	41	43
S_{S_1}	0.08	0.20	0.11	0.00	0.17	0.00	S_{S_1}	0.19	0.14	0.23	0.00	0.23	0.00
S_{S_2}	0.07	0.09	0.23	0.00	0.05	0.00	S_{S_2}	0.00	0.00	0.00	0.00	0.00	0.00
S_{S_3}	0.11	0.06	0.19	0.00	0.08	0.00	S_{S_3}	0.13	0.12	0.04	0.00	0.08	0.00
S_{S_4}	0.11	0.02	0.04	0.00	0.05	0.00	S_{S_4}	0.17	0.14	0.38	0.00	0.27	0.00
S_{S_5}	0.10	0.07	0.25	0.00	0.16	0.00	S_{S_5}	0.11	0.09	0.47	0.00	0.04	0.00
S_{S_6}	0.20	0.13	0.04	0.00	0.08	0.00	S_{S_6}	0.12	0.37	0.77	0.00	0.30	0.00
S_{S_7}	0.18	0.05	0.41	0.00	0.17	0.00	S_{S_7}	0.29	0.01	0.46	0.00	0.22	0.00
S_{S_8}	0.49	0.31	0.21	0.00	0.06	0.00	S_{S_8}	0.33	0.23	0.62	0.00	0.16	0.00
S_{S_9}	0.22	0.00	0.19	0.00	0.04	0.00	S_{S_9}	0.12	0.28	0.25	0.00	0.26	0.00
$S_{S_{10}}$	0.13	0.09	0.11	0.00	0.04	0.00	$S_{S_{10}}$	0.42	0.00	0.22	0.00	0.03	0.00
$S_{S_{11}}$	0.14	0.11	0.09	0.00	0.10	0.00	$S_{S_{11}}$	0.28	0.00	0.06	0.00	0.00	0.00
$S_{S_{12}}$	0.08	0.09	0.18	0.00	0.04	0.00	$S_{S_{12}}$	0.53	0.00	0.10	0.00	0.00	0.00
$S_{S_{13}}$	0.41	0.06	0.53	0.00	0.07	0.00	$S_{S_{13}}$	0.42	0.02	0.37	0.00	0.08	0.00
$S_{S_{14}}$	0.09	0.07	0.16	0.00	0.09	0.00	$S_{S_{14}}$	0.63	0.00	0.00	0.00	0.00	0.00
$S_{S_{15}}$	0.07	0.08	0.09	0.00	0.10	0.00	$S_{S_{15}}$	0.14	0.33	0.03	0.00	0.35	0.00
$S_{S_{16}}$	0.08	0.09	0.23	0.00	0.09	0.00	$S_{S_{16}}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{S_{17}}$	0.26	0.10	0.08	0.00	0.04	0.00	$S_{S_{17}}$	0.42	0.10	0.23	0.00	0.22	0.00
$S_{S_{18}}$	0.06	0.01	0.11	0.00	0.11	0.00	$S_{S_{18}}$	0.14	0.18	0.43	0.00	0.31	0.00
$S_{S_{19}}$	0.08	0.02	0.45	0.00	0.05	0.00	$S_{S_{19}}$	0.19	0.31	0.21	0.00	0.08	0.00
$S_{S_{20}}$	0.13	0.16	0.13	0.00	0.07	0.00	$S_{S_{20}}$	0.26	0.37	0.16	0.00	0.35	0.00
$S_{S_{21}}$	0.08	0.11	0.16	0.00	0.07	0.00	$S_{S_{21}}$	0.20	0.23	0.19	0.00	0.00	0.00
Median							Area						
SOM							SOM						
Signal	9	11	13	33	41	43	Signal	9	11	13	33	41	43
S_{S_1}	0.05	0.59	0.85	0.00	0.46	0.00	S_{S_1}	0.82	0.78	0.84	0.00	0.76	0.00
S_{S_2}	0.00	0.00	0.00	0.00	0.00	0.00	S_{S_2}	0.83	0.81	0.84	0.00	0.77	0.00
S_{S_3}	0.40	0.86	0.49	0.00	0.31	0.00	S_{S_3}	0.83	0.82	0.84	0.00	0.76	0.00
S_{S_4}	0.18	0.32	0.85	0.00	0.68	0.00	S_{S_4}	0.83	0.80	0.84	0.00	0.76	0.00
S_{S_5}	0.20	0.38	0.86	0.00	0.39	0.00	S_{S_5}	0.83	0.80	0.84	0.00	0.77	0.00
S_{S_6}	0.08	0.39	0.88	0.00	0.28	0.00	S_{S_6}	0.83	0.81	0.84	0.00	0.76	0.00
S_{S_7}	0.07	0.50	0.88	0.00	0.18	0.00	S_{S_7}	0.84	0.79	0.84	0.00	0.78	0.00
S_{S_8}	0.42	0.69	0.58	0.00	0.18	0.00	S_{S_8}	0.84	0.82	0.84	0.00	0.76	0.00
S_{S_9}	0.74	0.67	0.84	0.00	0.26	0.00	S_{S_9}	0.83	0.79	0.84	0.00	0.73	0.00
$S_{S_{10}}$	0.14	0.06	0.85	0.00	0.52	0.00	$S_{S_{10}}$	0.83	0.79	0.84	0.00	0.79	0.00
$S_{S_{11}}$	0.14	0.04	0.89	0.00	0.58	0.00	$S_{S_{11}}$	0.84	0.81	0.84	0.00	0.75	0.00
$S_{S_{12}}$	0.37	0.00	0.10	0.00	0.00	0.00	$S_{S_{12}}$	0.83	0.81	0.84	0.00	0.74	0.00
$S_{S_{13}}$	0.17	0.10	0.36	0.00	0.18	0.00	$S_{S_{13}}$	0.84	0.78	0.84	0.00	0.78	0.00
$S_{S_{14}}$	0.45	0.00	0.00	0.00	0.00	0.00	$S_{S_{14}}$	0.84	0.80	0.84	0.00	0.78	0.00
$S_{S_{15}}$	0.03	0.46	0.84	0.00	0.13	0.00	$S_{S_{15}}$	0.84	0.79	0.84	0.00	0.78	0.00
$S_{S_{16}}$	0.00	0.00	0.00	0.00	0.00	0.00	$S_{S_{16}}$	0.83	0.81	0.84	0.00	0.76	0.00
$S_{S_{17}}$	0.46	0.81	0.49	0.00	0.26	0.00	$S_{S_{17}}$	0.84	0.79	0.84	0.00	0.75	0.00
$S_{S_{18}}$	0.73	0.12	0.56	0.00	0.17	0.00	$S_{S_{18}}$	0.83	0.80	0.84	0.00	0.73	0.00
$S_{S_{19}}$	0.14	0.58	0.85	0.00	0.35	0.00	$S_{S_{19}}$	0.82	0.79	0.84	0.00	0.78	0.00
$S_{S_{20}}$	0.10	0.55	0.86	0.00	0.45	0.00	$S_{S_{20}}$	0.83	0.80	0.84	0.00	0.76	0.00
$S_{S_{21}}$	0.74	0.76	0.64	0.00	0.07	0.00	$S_{S_{21}}$	0.83	0.81	0.84	0.00	0.74	0.00

Table 4.17. Average effect size of F_2 for the testbed case

Derivative							Standard deviation						
SOM							SOM						
Signal	9	11	13	41	43	45	Signal	9	11	13	41	43	45
S_{S_1}	0.10	0.05	0.00	0.06	0.00	0.00	S_{S_1}	0.23	0.16	0.00	0.17	0.00	0.00
S_{S_2}	0.09	0.08	0.00	0.07	0.00	0.00	S_{S_2}	0.00	0.00	0.00	0.00	0.00	0.00
S_{S_3}	0.06	0.18	0.00	0.11	0.00	0.00	S_{S_3}	0.09	0.38	0.00	0.59	0.00	0.00
S_{S_4}	0.07	0.00	0.00	0.05	0.00	0.00	S_{S_4}	0.41	0.43	0.00	0.54	0.00	0.00
S_{S_5}	0.07	0.09	0.00	0.04	0.00	0.00	S_{S_5}	0.13	0.02	0.00	0.06	0.00	0.00
S_{S_6}	0.06	0.00	0.00	0.05	0.00	0.00	S_{S_6}	0.16	0.18	0.00	0.29	0.00	0.00
S_{S_7}	0.21	0.01	0.00	0.06	0.00	0.00	S_{S_7}	0.44	0.15	0.00	0.19	0.00	0.00
S_{S_8}	0.11	0.04	0.00	0.12	0.00	0.00	S_{S_8}	0.11	0.06	0.00	0.17	0.00	0.00
S_{S_9}	0.09	0.00	0.00	0.11	0.00	0.00	S_{S_9}	0.37	0.11	0.00	0.05	0.00	0.00
$S_{S_{10}}$	0.06	0.16	0.00	0.05	0.00	0.00	$S_{S_{10}}$	0.38	0.00	0.00	0.00	0.00	0.00
$S_{S_{11}}$	0.15	0.08	0.00	0.08	0.00	0.00	$S_{S_{11}}$	0.15	0.00	0.00	0.14	0.00	0.00
$S_{S_{12}}$	0.07	0.21	0.00	0.06	0.00	0.00	$S_{S_{12}}$	0.70	0.00	0.00	0.00	0.00	0.00
$S_{S_{13}}$	0.25	0.16	0.00	0.12	0.00	0.00	$S_{S_{13}}$	0.11	0.28	0.00	0.10	0.00	0.00
$S_{S_{14}}$	0.14	0.20	0.00	0.10	0.00	0.00	$S_{S_{14}}$	0.52	0.00	0.00	0.00	0.00	0.00
$S_{S_{15}}$	0.06	0.12	0.00	0.05	0.00	0.00	$S_{S_{15}}$	0.08	0.09	0.00	0.15	0.00	0.00
$S_{S_{16}}$	0.08	0.08	0.00	0.07	0.00	0.00	$S_{S_{16}}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{S_{17}}$	0.22	0.00	0.00	0.07	0.00	0.00	$S_{S_{17}}$	0.64	0.29	0.00	0.20	0.00	0.00
$S_{S_{18}}$	0.10	0.04	0.00	0.17	0.00	0.00	$S_{S_{18}}$	0.36	0.38	0.00	0.55	0.00	0.00
$S_{S_{19}}$	0.08	0.24	0.00	0.06	0.00	0.00	$S_{S_{19}}$	0.28	0.30	0.00	0.23	0.00	0.00
$S_{S_{20}}$	0.12	0.11	0.00	0.07	0.00	0.00	$S_{S_{20}}$	0.34	0.08	0.00	0.68	0.00	0.00
$S_{S_{21}}$	0.10	0.00	0.00	0.10	0.00	0.00	$S_{S_{21}}$	0.15	0.00	0.00	0.00	0.00	0.00
Median							Area						
SOM							SOM						
Signal	9	11	13	41	43	45	Signal	9	11	13	41	43	45
S_{S_1}	0.41	0.02	0.00	0.47	0.00	0.00	S_{S_1}	0.84	0.81	0.00	0.86	0.00	0.00
S_{S_2}	0.00	0.00	0.00	0.00	0.00	0.00	S_{S_2}	0.85	0.82	0.00	0.87	0.00	0.00
S_{S_3}	0.86	0.87	0.00	0.90	0.00	0.00	S_{S_3}	0.84	0.83	0.00	0.86	0.00	0.00
S_{S_4}	0.43	0.23	0.00	0.50	0.00	0.00	S_{S_4}	0.84	0.81	0.00	0.88	0.00	0.00
S_{S_5}	0.22	0.03	0.00	0.35	0.00	0.00	S_{S_5}	0.84	0.81	0.00	0.86	0.00	0.00
S_{S_6}	0.23	0.06	0.00	0.48	0.00	0.00	S_{S_6}	0.84	0.82	0.00	0.86	0.00	0.00
S_{S_7}	0.74	0.47	0.00	0.76	0.00	0.00	S_{S_7}	0.84	0.81	0.00	0.86	0.00	0.00
S_{S_8}	0.13	0.25	0.00	0.18	0.00	0.00	S_{S_8}	0.84	0.82	0.00	0.86	0.00	0.00
S_{S_9}	0.66	0.55	0.00	0.69	0.00	0.00	S_{S_9}	0.84	0.82	0.00	0.86	0.00	0.00
$S_{S_{10}}$	0.70	0.50	0.00	0.68	0.00	0.00	$S_{S_{10}}$	0.84	0.81	0.00	0.86	0.00	0.00
$S_{S_{11}}$	0.65	0.48	0.00	0.71	0.00	0.00	$S_{S_{11}}$	0.84	0.81	0.00	0.86	0.00	0.00
$S_{S_{12}}$	0.66	0.00	0.00	0.00	0.00	0.00	$S_{S_{12}}$	0.84	0.81	0.00	0.87	0.00	0.00
$S_{S_{13}}$	0.43	0.00	0.00	0.40	0.00	0.00	$S_{S_{13}}$	0.85	0.80	0.00	0.86	0.00	0.00
$S_{S_{14}}$	0.82	0.00	0.00	0.00	0.00	0.00	$S_{S_{14}}$	0.84	0.81	0.00	0.87	0.00	0.00
$S_{S_{15}}$	0.16	0.19	0.00	0.39	0.00	0.00	$S_{S_{15}}$	0.84	0.81	0.00	0.86	0.00	0.00
$S_{S_{16}}$	0.00	0.00	0.00	0.00	0.00	0.00	$S_{S_{16}}$	0.84	0.82	0.00	0.87	0.00	0.00
$S_{S_{17}}$	0.81	0.84	0.00	0.12	0.00	0.00	$S_{S_{17}}$	0.86	0.83	0.00	0.87	0.00	0.00
$S_{S_{18}}$	0.73	0.33	0.00	0.79	0.00	0.00	$S_{S_{18}}$	0.84	0.81	0.00	0.87	0.00	0.00
$S_{S_{19}}$	0.20	0.02	0.00	0.22	0.00	0.00	$S_{S_{19}}$	0.84	0.81	0.00	0.86	0.00	0.00
$S_{S_{20}}$	0.03	0.02	0.00	0.43	0.00	0.00	$S_{S_{20}}$	0.84	0.81	0.00	0.86	0.00	0.00
$S_{S_{21}}$	0.86	0.87	0.00	0.89	0.00	0.00	$S_{S_{21}}$	0.84	0.82	0.00	0.86	0.00	0.00

Table 4.18. Average effect size of F_3 for the testbed case

Derivative									Standard deviation								
SOM									SOM								
Signal	9	11	13	15	33	41	43	45	Signal	9	11	13	15	33	41	43	45
S_{S_1}	0.11	0.00	0.05	0.00	0.00	0.12	0.00	0.00	S_{S_1}	0.18	0.00	0.23	0.00	0.00	0.05	0.00	0.13
S_{S_2}	0.29	0.00	0.06	0.00	0.00	0.06	0.00	0.10	S_{S_2}	0.21	0.00	0.08	0.00	0.00	0.05	0.00	0.06
S_{S_3}	0.05	0.00	0.17	0.00	0.00	0.07	0.00	0.49	S_{S_3}	0.18	0.00	0.19	0.00	0.00	0.12	0.00	0.04
S_{S_4}	0.06	0.00	0.09	0.00	0.00	0.06	0.00	0.01	S_{S_4}	0.19	0.00	0.08	0.00	0.00	0.00	0.00	0.34
S_{S_5}	0.06	0.00	0.08	0.00	0.00	0.08	0.00	0.00	S_{S_5}	0.15	0.00	0.06	0.00	0.00	0.06	0.00	0.35
S_{S_6}	0.12	0.00	0.06	0.00	0.00	0.08	0.00	0.11	S_{S_6}	0.26	0.00	0.07	0.00	0.00	0.33	0.00	0.38
S_{S_7}	0.06	0.00	0.11	0.00	0.00	0.05	0.00	0.09	S_{S_7}	0.29	0.00	0.10	0.00	0.00	0.00	0.00	0.13
S_{S_8}	0.05	0.00	0.13	0.00	0.00	0.12	0.00	0.19	S_{S_8}	0.06	0.00	0.09	0.00	0.00	0.13	0.00	0.11
S_{S_9}	0.08	0.00	0.12	0.00	0.00	0.09	0.00	0.23	S_{S_9}	0.28	0.00	0.07	0.00	0.00	0.17	0.00	0.13
$S_{S_{10}}$	0.04	0.00	0.05	0.00	0.00	0.08	0.00	0.02	$S_{S_{10}}$	0.05	0.00	0.05	0.00	0.00	0.15	0.00	0.18
$S_{S_{11}}$	0.13	0.00	0.04	0.00	0.00	0.07	0.00	0.07	$S_{S_{11}}$	0.14	0.00	0.04	0.00	0.00	0.04	0.00	0.19
$S_{S_{12}}$	0.04	0.00	0.08	0.00	0.00	0.07	0.00	0.03	$S_{S_{12}}$	0.19	0.00	0.07	0.00	0.00	0.00	0.00	0.27
$S_{S_{13}}$	0.11	0.00	0.09	0.00	0.00	0.19	0.00	0.01	$S_{S_{13}}$	0.77	0.00	0.61	0.00	0.00	0.33	0.00	0.34
$S_{S_{14}}$	0.14	0.00	0.07	0.00	0.00	0.15	0.00	0.29	$S_{S_{14}}$	0.21	0.00	0.06	0.00	0.00	0.00	0.00	0.00
$S_{S_{15}}$	0.23	0.00	0.04	0.00	0.00	0.06	0.00	0.17	$S_{S_{15}}$	0.10	0.00	0.06	0.00	0.00	0.05	0.00	0.16
$S_{S_{16}}$	0.15	0.00	0.13	0.00	0.00	0.05	0.00	0.01	$S_{S_{16}}$	0.08	0.00	0.05	0.00	0.00	0.07	0.00	0.07
$S_{S_{17}}$	0.25	0.00	0.05	0.00	0.00	0.06	0.00	0.26	$S_{S_{17}}$	0.36	0.00	0.09	0.00	0.00	0.34	0.00	0.20
$S_{S_{18}}$	0.13	0.00	0.20	0.00	0.00	0.05	0.00	0.15	$S_{S_{18}}$	0.71	0.00	0.47	0.00	0.00	0.00	0.00	0.18
$S_{S_{19}}$	0.05	0.00	0.04	0.00	0.00	0.11	0.00	0.07	$S_{S_{19}}$	0.08	0.00	0.07	0.00	0.00	0.11	0.00	0.41
$S_{S_{20}}$	0.08	0.00	0.05	0.00	0.00	0.05	0.00	0.06	$S_{S_{20}}$	0.27	0.00	0.04	0.00	0.00	0.39	0.00	0.00
$S_{S_{21}}$	0.08	0.00	0.06	0.00	0.00	0.12	0.00	0.03	$S_{S_{21}}$	0.26	0.00	0.06	0.00	0.00	0.11	0.00	0.41
Median									Area								
SOM									SOM								
Signal	9	11	13	15	33	41	43	45	Signal	9	11	13	15	33	41	43	45
S_{S_1}	0.41	0.00	0.40	0.00	0.00	0.34	0.00	0.43	S_{S_1}	0.84	0.00	0.82	0.00	0.00	0.78	0.00	0.77
S_{S_2}	0.27	0.00	0.05	0.00	0.00	0.14	0.00	0.25	S_{S_2}	0.84	0.00	0.80	0.00	0.00	0.81	0.00	0.81
S_{S_3}	0.86	0.00	0.82	0.00	0.00	0.88	0.00	0.87	S_{S_3}	0.84	0.00	0.82	0.00	0.00	0.79	0.00	0.78
S_{S_4}	0.88	0.00	0.83	0.00	0.00	0.98	0.00	0.93	S_{S_4}	0.84	0.00	0.82	0.00	0.00	0.81	0.00	0.78
S_{S_5}	0.16	0.00	0.10	0.00	0.00	0.07	0.00	0.28	S_{S_5}	0.84	0.00	0.82	0.00	0.00	0.79	0.00	0.77
S_{S_6}	0.23	0.00	0.24	0.00	0.00	0.49	0.00	0.71	S_{S_6}	0.84	0.00	0.81	0.00	0.00	0.80	0.00	0.79
S_{S_7}	0.33	0.00	0.30	0.00	0.00	0.46	0.00	0.31	S_{S_7}	0.85	0.00	0.82	0.00	0.00	0.79	0.00	0.80
S_{S_8}	0.19	0.00	0.22	0.00	0.00	0.23	0.00	0.05	S_{S_8}	0.85	0.00	0.82	0.00	0.00	0.80	0.00	0.77
S_{S_9}	0.19	0.00	0.11	0.00	0.00	0.36	0.00	0.24	S_{S_9}	0.84	0.00	0.81	0.00	0.00	0.78	0.00	0.77
$S_{S_{10}}$	0.86	0.00	0.87	0.00	0.00	0.90	0.00	0.87	$S_{S_{10}}$	0.84	0.00	0.81	0.00	0.00	0.80	0.00	0.78
$S_{S_{11}}$	0.86	0.00	0.86	0.00	0.00	0.88	0.00	0.86	$S_{S_{11}}$	0.84	0.00	0.81	0.00	0.00	0.81	0.00	0.81
$S_{S_{12}}$	0.23	0.00	0.04	0.00	0.00	0.00	0.00	0.27	$S_{S_{12}}$	0.85	0.00	0.82	0.00	0.00	0.80	0.00	0.80
$S_{S_{13}}$	0.09	0.00	0.14	0.00	0.00	0.28	0.00	0.19	$S_{S_{13}}$	0.84	0.00	0.82	0.00	0.00	0.81	0.00	0.78
$S_{S_{14}}$	0.21	0.00	0.04	0.00	0.00	0.00	0.00	0.00	$S_{S_{14}}$	0.85	0.00	0.82	0.00	0.00	0.80	0.00	0.78
$S_{S_{15}}$	0.27	0.00	0.20	0.00	0.00	0.18	0.00	0.18	$S_{S_{15}}$	0.84	0.00	0.82	0.00	0.00	0.78	0.00	0.81
$S_{S_{16}}$	0.20	0.00	0.45	0.00	0.00	0.41	0.00	0.64	$S_{S_{16}}$	0.84	0.00	0.81	0.00	0.00	0.83	0.00	0.82
$S_{S_{17}}$	0.47	0.00	0.33	0.00	0.00	0.40	0.00	0.18	$S_{S_{17}}$	0.85	0.00	0.83	0.00	0.00	0.79	0.00	0.79
$S_{S_{18}}$	0.72	0.00	0.47	0.00	0.00	0.84	0.00	0.37	$S_{S_{18}}$	0.85	0.00	0.82	0.00	0.00	0.79	0.00	0.79
$S_{S_{19}}$	0.32	0.00	0.51	0.00	0.00	0.42	0.00	0.59	$S_{S_{19}}$	0.85	0.00	0.82	0.00	0.00	0.80	0.00	0.78
$S_{S_{20}}$	0.13	0.00	0.33	0.00	0.00	0.06	0.00	0.35	$S_{S_{20}}$	0.85	0.00	0.82	0.00	0.00	0.81	0.00	0.75
$S_{S_{21}}$	0.40	0.00	0.13	0.00	0.00	0.42	0.00	0.22	$S_{S_{21}}$	0.85	0.00	0.82	0.00	0.00	0.79	0.00	0.79

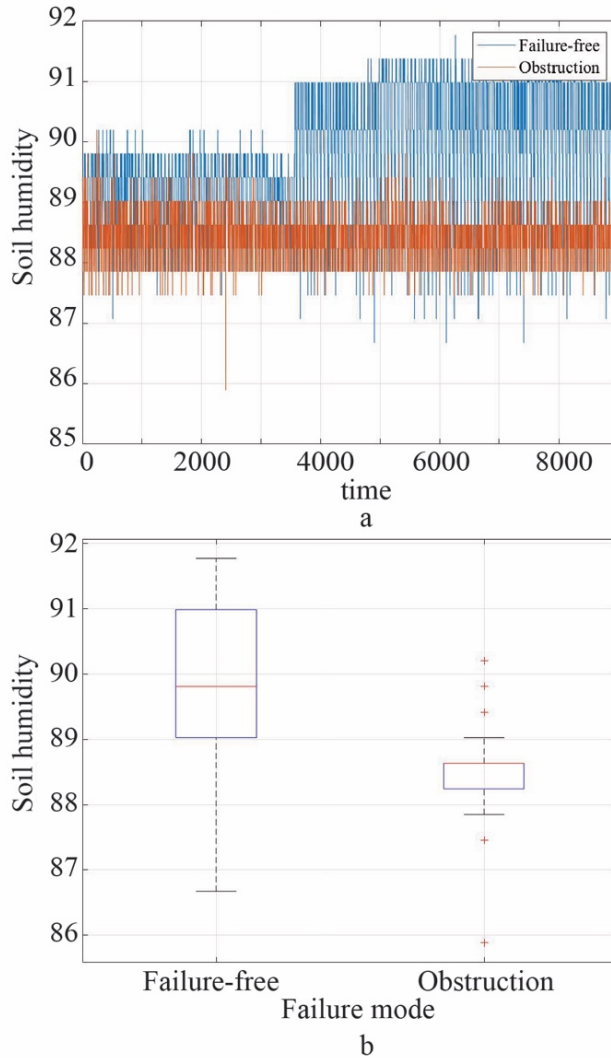


Figure 4.9. Comparison of soil humidity for the irrigation valve obstruction failure mode. (a) Analysis of signals, (b) Boxplot

2. The environmental and external disturbances:

The obtained failure indicators present a high number of symptoms even in signal that do not seem to be directly related to the injected failure. For instance, inlet fan's rotation speed ('Fan-in' S_{A_4}) presents significant difference during the occurrence of 'Tank leak' (F_1) in most of the occurring SOMs and in most of the analyzed signal features (as seen in Figure 4.8). However, we did not find any insight that enabled to determine if the observed difference was caused by failure effect or environmental disturbances. It is not likely that 'Tank leak' failure mode (F_1), affects the rotation speed of the inlet fan ('Fan-in' S_{A_4}). In the case of 'Irrigation pipe blocked' (F_2), there are some other system

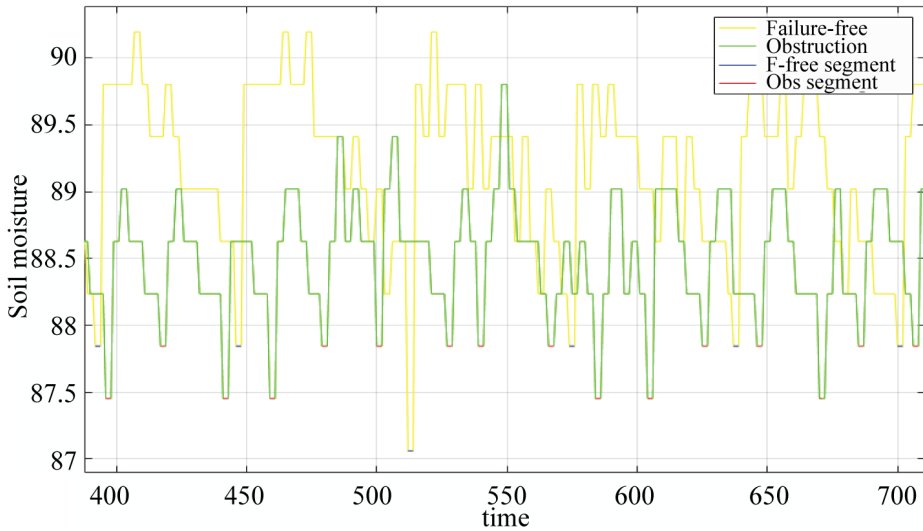


Figure 4.10. Fragment of soil moisture that presents signal segments corresponding to ζ_{41} for both, failure-free and failed operation

signals that also reported significant difference through segmentation, such as: soil temperature from plant bed 1 ('Soil temperature sensor' S_{S_3}), PH of the plant bed 1 ('PH sensor' S_{S_7}), PH of plant bed 2 ('PH sensor' $S_{S_{21}}$) and the rotation speed of the inlet fan ('Fan-in' S_{A_4}). However, we did not find evidence that allows concluding the observed differences are caused by the obstruction of the irrigation pipe. Considering the current failure was injected into plant bed 2, it is not likely the failure is manifested in signals corresponding to plant bed 1. Variations caused by unknown factors, such as natural greenhouse processes may have affected the afore-mentioned variables leading to false failure symptoms.

3. The compensation effect of the control:

The analysis of failures conducted through the testbed explored new effect of failures, namely: Failure Induced Operation Modes (FIOM). FIOM is defined as changes of system operation modes caused by the injected failures. We found that FIOM affects the size of data samples of specific SOMs. This characteristic of FIOM can invalidate the results of the statistical test as the frequency of occurrence of specific SOM can approach to zero. For instance, in the analysis of failure mode of 'Irrigation pipe blocked' (F_2), 'Water temperature sensor' (S_{S_9}) presented symptoms in SOMs ζ_9 , ζ_{11} and ζ_{41} . These symptoms were not present in failure free operations, as water heating occurs at the tank before irrigation and there is no control rule that directly relates water heating ('Water temperature sensor' S_{S_9}) to soil moisture levels ('Soil moisture sensor' of plant bed 2, $S_{S_{17}}$). The analysis of the whole length signal of 'Water temperature sensor' (S_{S_9}), did not present major differences between the failure-free operation and the failed one. Images presented in the Figure 4.11(a) and Figure 4.11(b), demonstrates a high level of similarity on the data distribution from both datasets. However, we observed a significant frequency variation of the irrigation valve's openings/closings of

plant bed 2 ('Electro valve plant bed 2' S_{A_6}). While the merged dataset that integrates several experiments corresponding to failure-free operation only reported 450 opening times, the dataset of failed operations reached up to 1259. Similarly, the inlet tank valve ('Electro valve water reservoir' S_{A_2}) reported 72 opening times during the failure free operation, while it reported 123 with the dataset of failed operation. The analysis of the 'Heater' (S_{A_3}) also presents a difference in the total time the heater was 'On' throughout the total dataset. While the heater was active only for 12 seconds in the failure-free operation, it was activated for 55 seconds in the operation with the failure.

Variations of system dynamics was observed due to the failure of blocked irrigation pipe (F_2). This failure increased the frequency of irrigation and as a direct consequence the refilling of the water tank. It also caused a slight reduction of the water temperature (shown by (S_{S_9})) that resulted in the observed symptoms. Although this situation did not affect negatively failure detection on the afore-mentioned SOMs, ζ_{13} , ζ_{43} and ζ_{45} did not present any symptoms. SOM ζ_{13} is the system operation mode where only the 'Heater' (S_{A_3}) and the inlet fan ('Fan-in' S_{A_4}) are active. We observed that this failure mode only occurred 2 times during the failure-free operation, while it occurred 53 times in the scenarios of failure modes. This negatively affects the results of failure indicator (see Figure 4.8) for this particular SOM not only for this signal, but for all system signals due to the lack of data samples available for the failure-free operations. SOM ζ_{43} (when only S_{A_2} , S_{A_4} and S_{A_6} are active) and SOM ζ_{45} (when the S_{A_3} , S_{A_4} and S_{A_6} are active) were also activated only once during the failure-free operation.

The reduction in the SOM frequency of occurrence and SOM duration are not the only factors that may affect the results of the statistical tests. In general terms, the size of the analyzed data samples is a critical factor in statistical tests. Table 4.16 shows that the effect size of SOM ζ_{33} and SOM ζ_{43} is zero for all the analyzed signals in 'Tank leak' (F_1). In 'Irrigation pipe blocked' (F_2), the SOMs ζ_{13} , ζ_{43} and ζ_{45} also reported effect size of zero (see Table 4.17). Likewise, in 'Irregular fan operation' (F_3), the SOMs ζ_{11} , ζ_{15} , ζ_{33} and ζ_{43} reported effect size of zero for all the analyzed variables (see Table 4.18). This result was caused by the insufficient number of data samples available for the evaluation of the statistical test. For instance, SOMs ζ_{33} and ζ_{43} only occurred in once and twice out of five experiments conducted for F_1 , respectively, and they did not last longer than 5 seconds. A similar situation was observed with the above-mentioned SOMs in their corresponding failure modes.

Another factor that should be considered is the duration time of the SOMs. The duration time of system operation modes is determined by system dynamics. Control actions enables SOM transitions based on the use and operative conditions. The frequent activation and deactivation of system components causes the system to often switch from one SOM to another. It leads to short SOM duration, which results in short signal segments. Short signal segments that do not reach more than one signal measurement per segment, hampers the estimation of signal features (such as derivative and area that require at least two data measurements per segment). Whenever the duration of the SOM does not allow measuring data more than once, the derivative is zero. (e.g. if the sensor sampling is every second and the SOM only lasts for one second). This results in large number of measurements with zero values that do not characterize the real behavior of the analyzed feature. If the SOMs lasted less than two samples in both the failure-free and the failed cases, the statistical test will deliver a p -value > 0.05 .

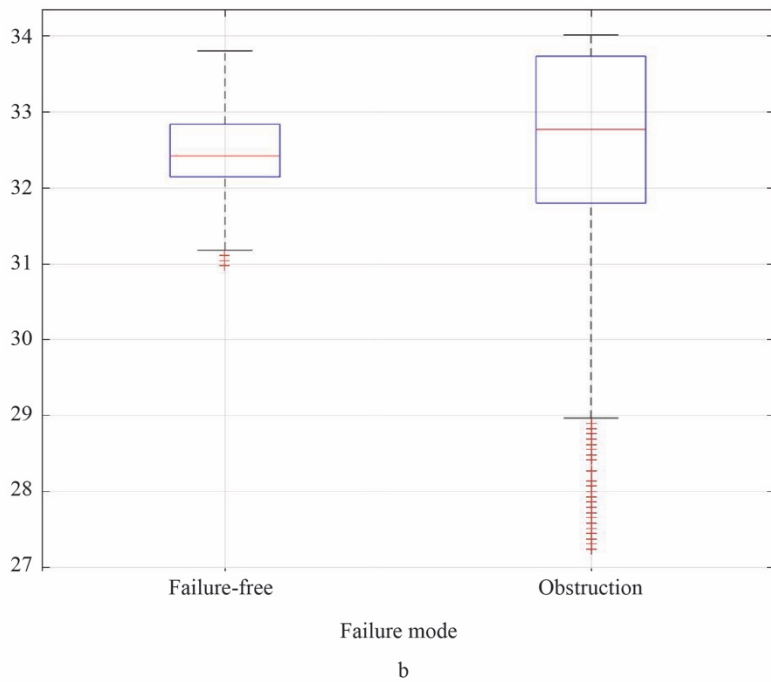
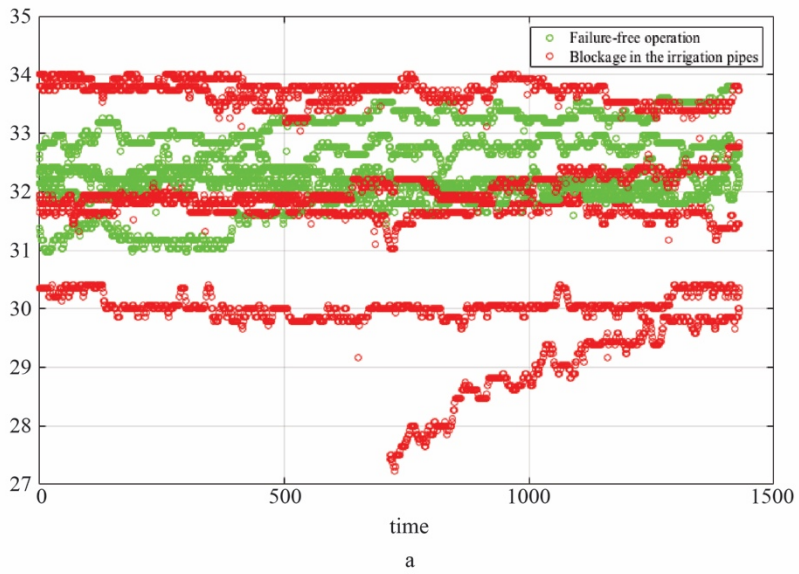


Figure 4.11. Comparison of the water temperature signal between the failure-free operation and F_2 . A.) Scatter plot that includes all the conducted experiments. B.) Boxplot

4.5.3 Analysis of the discriminant power of the obtained failure indicators

In this section, we analyzed the similarity level of the derived failure indicators, in order to estimate their discriminant power. For this purpose, we followed the same procedure presented in section 4.4.3. Results are presented in Table 4.19 where we can observe that the combination of indicators that presented the lowest levels of similarity are ‘Tank leak’ (F_1) versus ‘Irrigation pipe blocked’ (F_2) in all the analyzed signal features (except ‘Area’ (a_4) feature). Accordingly, ‘Derivative’ feature (a_1) reached a minimum of 92.5% of cells in common between both failure modes; ‘Standard deviation’ (a_2) presented a minimum of 90.68% of similarity level; and ‘Median’ (a_3) presented 90.88% of cells in common. The lowest level of similarity in ‘Area’ (a_4) was reached by the comparison between ‘Irrigation pipe blocked’ (F_2) and ‘Irregular fan operation’ (F_3) with 91.49% of cells in common. In all aforementioned cases, ‘Irrigation pipe blocked’ (F_2) was involved, which means that this is the failure mode with the highest discriminant power.

The combination of failure indicators that presented the highest levels of similarity was ‘Tank leak’ (F_1) versus ‘Irregular fan operation’ (F_3). This occurred in all the analyzed signal features alike: ‘Derivative’ (a_1) reached a maximum of 94.73%, ‘Standard deviation’ (a_2) reported 93.11%, ‘Median’ (a_3) reported 90.88% and ‘Area’ (a_4) reached a maximum of 95.74% of cells in common. We can observe that the ‘Median’ (a_3) the feature presented the highest discriminant power with a maximum value of 90.88% and with minimum value 89.06%.

In general terms, the discriminant power presented by the obtained failure indicators is weak, as most of them reached similarity levels over 90%. Even though, there are plenty of symptoms that occurred with high frequency (i.e. red cells), most of these symptoms can be observed in several failure modes. This situation can be caused by lack of response to real failure manifestations, and false response to external disturbances. The most determinant factor in the discriminant power analysis was the Failure Induced Operation Modes

Table 4.19. Similarity level of the failure indicators derived for the testbed

		Derivative		
		F1	F2	F3
F1		100.00	92.50	94.73
F2		92.50	100.00	92.71
F3		94.73	92.71	100.00

		Standard deviation		
		F1	F2	F3
F1		100.00	90.98	93.11
F2		90.98	100.00	90.68
F3		93.11	90.68	100.00

		Median		
		F1	F2	F3
F1		100.00	89.06	90.88
F2		89.06	100.00	89.67
F3		90.88	89.67	100.00

		Area		
		F1	F2	F3
F1		100.00	93.62	95.74
F2		93.62	100.00	91.49
F3		95.74	91.49	100.00

(FIOM). For instance, FIOM cells represented the 84% of cells that were different between F_1 and F_2 in the ‘Derivative’ feature (a_1). They distinguished 68.47% of cells for the same failure modes by the ‘Standard deviation’ feature (a_2); 53% of the cells by the ‘Median’ feature (a_3) and 100% of the cells by the ‘Area’ feature (a_4). This example demonstrates that deviations of signal segments was not the most relevant factor for failure discrimination. The variation of the system dynamics had greater influence in determining the difference between the analyzed failure modes.

4.6 Discussion of the results and their implications

Failure analysis by SOM-based signal segmentation provides meaningful information about failures as opposed to failure analysis based on unsegmented signals. The conducted analysis revealed the appearance of failure symptoms that were not foreseen, as they are the manifestation of combined actions of the control system. In the real-life study (See section 4.5), the analysis of the effect of ‘Irrigation pipe blocked’ (F_2) and the analysis of ‘Tank leak’ (F_1) through SOM-based signal segmentation, revealed the effect of both failures over the water temperature. These symptoms could not be observed by the study of the unsegmented signals. This situation can be explained by the compensatory effect that the control system applies to the systems’ parameters. Control principles aim to guarantee system stability by keeping system variables on the defined set points. For this purpose, they manipulate system actuators pushing them to conditions beyond their regular operation. This facilitates operation with fault tolerance but it also hinders observability of failure manifestations on signal measurements.

SOMs also convey information about system dynamics. The pilot study presented in section 4.4 found that the role of SOMs in signal-based failure analysis was to isolate the effect of failures on system signals from the variations caused by self-tuning actions (activation and deactivation of actuators). However, the analysis of the real-life case showed us a wider potential. Failures cause changes on system dynamics modifying regular operation. This was shown by the variations of the frequency and duration time of the SOMs. At the same time, this variation may considerably reduce size of the datasets of certain SOMs and may invalidate the statistical tests of failure classifications. However, these changes of SOM frequency and duration are also failure symptoms and constitutes failure manifestations.

The compensatory effect of the control system causes changes on the activation of actuators. These activations are limited by control constraints that determine the joint operation of system actuators. Analyzing failures only based on the actuator status would imply a component-level analysis, which would not provide information about the interactions of system components and its effect over the entire system behavior. However, SOM-based-analysis enables understanding variations on system dynamics at system level. It constitutes a valuable source of information about system operation due to failures. Moreover, it represents a meaningful opportunity for studying failure progression and system degradation. The gradual evolution of SOM-frequency and SOM-duration can provide means for understanding the failure forming process and even to conduct failure forecasting.

The SOM-based signal segmentation presented important drawbacks thought. The real-life application demonstrated that it is very sensitive to environmental disturbances, causing false positive results. The ambient temperature caused false failure symptoms on ‘Irregular fan operation’ (F_3) failure mode. It was not possible to find any direct relationship between the analyzed failure and the observed symptoms. However, significant differences between the ambient temperature of the considered failure-free cases, and the failed ones, were observed. All studied failure modes presented symptoms on certain system signals that could not be explained by the induced failures. It required effort from the researchers to characterize the effect of failures, which in fact questions the reliability of the obtained indicators. The proposed failure analysis method is also sensitive to the amount of measurements per signal segments. Signal features such as ‘Derivative’ cannot be applied with less than two data measurements. It leads to erratic symptoms affecting the reliability of the results.

Failure symptoms that manifest with delay cannot be observed during the SOMs where symptoms are developed. This phenomenon was demonstrated by the case of ‘Soil moisture sensor’ (S_{S_3}) in the failure mode F2, ‘Irrigation pipe blocked’. This failure was expected to be observed whenever the irrigation valve was active, due to the reduction on water supply on the plant bed. However, due to the short duration lapse of the SOM, and to the slow absorption process of the water, this failure manifestation was not observed on the SOM in which the irrigation valve was active, but in a further one. It causes confusion and it misleads attention to other causes, affecting the failure interpretation.

Although the SOM-based signal segmentation manages to reduce the disturbances caused by the activation and deactivation of actuators (which can be misinterpreted as failures), it does not reduce the effect of environmental disturbances. These external disturbances make the failure manifestations difficult to understand and it hinders the analysis of failures based on signal segmentation. Nevertheless, the obtained results suggest that the real value of the SOM is its capability to capture changes of system dynamics rather than to represent failure manifestation by signal deviations. It represents a turning point in our research, as we will move from studying failure manifestations on system signals to analyze failures through the analysis of variations on system dynamics. The following chapter will be focused on studying the SOM frequency and duration and its applicability in failure analysis and forecasting. We will study to what extent these two variables can be used to characterize failures and their evolution.

4.7 Conclusions

This chapter aimed to analyze how SOMs influence failure manifestations. For this purpose, we developed a new approach to failure analysis based on SOM-based signal segmentation. A computational model of a kettle, as well as the instrumented testbed were used as means for experimentation. Failure injection was performed to reproduce typical failure modes of systems and system signals were recorded in large number of datasets. Datasets were analyzed through a failure indicator *FI* approach that evaluated the deviation of signal segmented by system operation modes.

The results revealed that SOM-based signal segmentation allows observing failure symptoms that cannot be detected when analyzing the unsegmented signals. However, it also presents some limitations. Signal segmentation leads to loss of information, particularly in the cases in which SOMs duration is very short, and in the cases in which failure effect is gradual, i.e. when failure effects influences system operation with a delay. This approach is sensitive to environmental disturbances and external factors. External disturbances cause signal deviations that can be misinterpreted as failure symptoms making the interpretation of the observed results difficult. The compensation effect of control causes significant variations to the frequency of occurrence of SOMs and their duration. It leads to the occurrence of failure induced operation modes (FIOM) that are SOMs triggered by the control of the system to compensate the effect of failures. This phenomenon hinders the execution of the statistical test, as it reduces sample size and the statistical power either for the dataset of failure-free or failure mode operation.

Changes of SOM frequency and SOM duration provides a new opportunity for failure analysis, though. They represent variations in system dynamics as compensation to the effect of failure. The analysis of the discriminant power of FI revealed that instead of signal deviations, FIOMs were the most determinant factor for differentiating failures. This suggest that changes on the frequency and duration of SOMs can be considered as failure indicators, and thus, they can be used not just for failure diagnosis, but also, for studying the failure forming process. Variations on SOM frequency and SOM duration as failure manifestations requires a deeper study thought. The upcoming chapter will be focused on investigating deeper this phenomenon.

4.8 References

- [1] I. Horváth, “Comparison of three methodological approaches of design research,” presented at the International Conference on Engineering Design, ICED’07, Paris - France, 2007.

Chapter 5

The role of SOM in a behavior-based analysis of failures

5.1 Introduction

5.1.1 Objectives of the research cycle 4

As it was already stated earlier, this research focuses on exploring the role of System Operation Modes (SOMs) in failure diagnosis and forecasting in first generation Cyber-Physical Systems (CPS) with self-tuning capabilities. These types of systems are instrumented with multiple sensors and actuators, which are controlled by setting the system into a finite set of SOMs. The experiments reported in chapter 4 explored some new insights about not only the failures manifestation, but also the reactive control of the system to maintain its stability in failure modes. We observed that failures induced unexpected SOM, which were not occurring under failure-free operation. We have found that some SOMs that used to occur during failure-free operation were not activated due to the effect of the failure. It led us to the conclusion that failures cause changes to SOM frequency and duration affecting the overall system operation (see Subsection 4.5.2).

In this chapter, we aim to explore the potential role of SOM frequency and duration in characterizing failure-forming process. We investigated how they can be used for defining failure indicators for failure diagnosis and forecasting. Therefore, the objectives of the study reported in this chapter are, firstly, to explore how SOM frequency and duration reflect failures as they evolve due to wearing and system degradation. Secondly, we aim to investigate how SOMs' duration and frequency characterizes the failure forming process for different types of failures. Thirdly, we aim to identify the potential role of SOMs' frequency and duration in failure forecasting, in which the type and expected time of occurrence of failures are predicted based on the analyzed trends.

These roles of SOM will be studied through two demonstrative examples of first generation Cyber Physical Systems. First, we study the SOM behavior of a water kettle model in Matlab/Simulink with the goal of analyzing the role of SOMs' frequency and duration in a controlled simulation environment. In our second study, we will investigate how a real-life system, such as the testbed of a greenhouse (detailed in Chapter 3), behaves under failure forming. With this case, we will be able to explore if SOMs' frequency and duration can play a role in failure diagnosis and forecasting.

5.1.2 Research hypothesis and assumptions

SOM's frequency of occurrence (Fq) refers to the number of times a SOM occurs during a time period. SOM's duration (D) is the average time that a SOM is active. Control system intends to maintain the stability of the system by activating SOMs in various sequences and lengths. Our experiments with the kettle model and the greenhouse testbed proved there are variations on SOMs due to failure. The reason of this phenomenon is that the control system activates specific Component Operation Modes (COMs) in order to compensate the effect of the failure and to maintain the operation of the system defined by the control model in the form of set-points.

The hypothesis that leads this research cycle can then be written in the following way: Failure forming process and system degradation can be recognized based on variations of the SOMs behavior, which is reflected by changes in its frequency of occurrence and its average duration. This hypothesis is supported by the fact that failure forming process forces the system to compensate the increasing deviation between the set-points and the observed signals. It is accomplished by activating the systems components to work beyond their normal operation, to maintain the stability of the system [6]. Considering that system's aging or degradation increases as time progresses, changes in SOMs' frequency and duration increase too, forming downward or upward trends in such parameters that do not occur during failure-free operation. These changes in the frequency and duration of SOMs keep occurring until system actuators are no longer able to maintain the stability of the system. Once the system's actuators can no longer compensate the effect of degradation, the system will be out of its stable control domain, affecting the controllability property.

Considering the aforementioned factors, our primary assumptions are:

- The pattern composed by failure induced trends of the frequency and duration of SOMs is consistent every time the system is affected by the same 'Failure mode' (F_r),
- Every 'Failure mode' (F_r) presents a characteristic pattern of trends on the SOMs' frequency (Fq) and duration (D) indicators which differentiates it from other failure modes,
- The joint analysis of SOMs' frequency (Fq) and duration (D) enables the diagnosis and forecasting of failures.

Based on the derived failure indicators in chapter 4, we claim that, a particular failure mode tends to affect the same set of variables every time it occurs. Due to that, the system manipulates the same set of actuators as compensation, leading to similar variations on the frequency and duration of SOMs. Consequently, a different failure will affect a different set of variables, and thus, the compensation actions (variations on the frequency and duration of SOMs) will be different. These assumptions over the consistent behavior of a system under a failure mode are the necessary conditions for failure classification of 1st generation CPSs. This chapter will be focused on the validation of these assumptions.

5.2 Approach for investigating trends of the SOMs behavior

In this chapter, we will analyze to what extent SOMs' frequency (Fq) and duration (D) can be used for characterizing failures and studying their evolution. The approach we followed consists of a set of subsequent studies (see Figure 5.1):

1. Analysis of the consistency and discriminant power of the observed trends of SOMs' frequency (Fq) and duration (D).
2. Evaluation of SOMs' frequency (Fq) and duration (D) as indicators of the failure progress.
3. Analysis of the potential use of the changing frequency (Fq) and duration (D) of SOMs in failure forecasting.

These aspects are analyzed based on data collected in the use simulation of the water kettle model and long terms usage of the greenhouse testbed (see Chapter 4). All the above-mentioned steps are to be conducted for both the computer model of the kettle

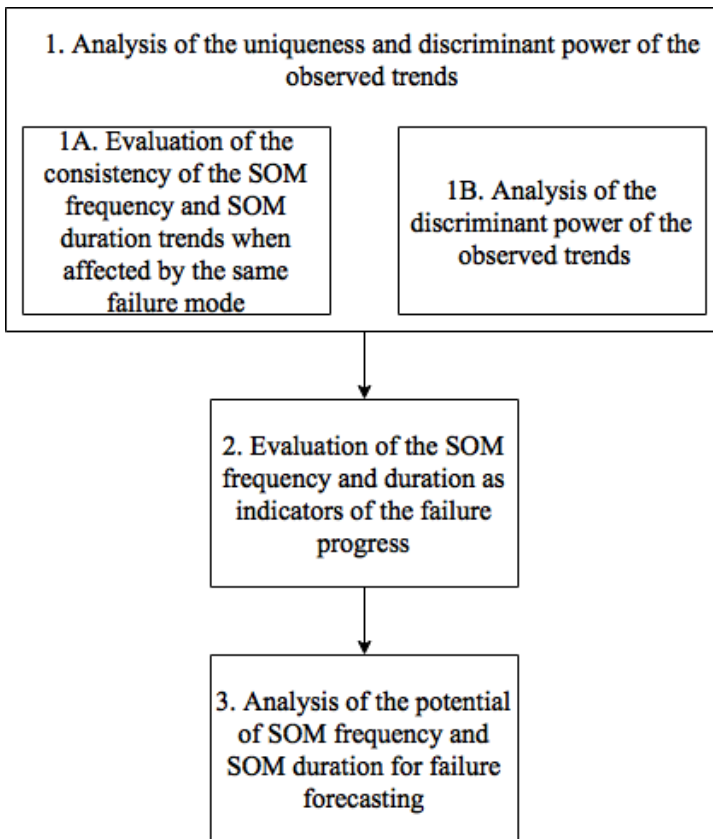


Figure 5.1. Approach for exploring the role of SOMs' frequency (Fq) and duration (D) in failure diagnosis and forecasting

and the greenhouse testbed. The kettle model was used as basis for exploring the role of the frequency and duration of SOMs and for conceptualizing a demonstrative approach for conducting failure forecasting. Nevertheless, it responded to an idealistic implementation where external disturbances were not considered, and where few variables and SOMs were involved. The testbed implementation aims to analyze the role of the changes of frequency and duration of SOMs in a more sensitive environment. For this purpose, we will conduct the same exploratory analysis performed with the kettle model, as well as to implement the forecasting approach conceptualized through the kettle.

5.2.1 Investigation of the consistency and discriminant power of the trends

This study aims to explore two aspects of the discriminant power of SOMs' frequency (Fq) and duration (D):

- The consistency of the trends when produced by the same 'Failure mode' (F_r).
- The discriminant power for distinguishing different 'Failure modes' (F_r).

If the observed trends corresponding to different datasets from the same 'Failure mode', F_r , have strong similarity, it confirms that F_r is manifested through consistent behavior. If trends of SOM frequency and duration caused by 'Failure mode' (F_r) differ from trends of all other failures, we can infer that they have sufficient discriminant power. Step (i) and step (ii) aims to determine to what extent the trends of SOM frequency and duration changes can be used for failure diagnosis. For this purpose, the trends for frequency and duration of each SOM are filtered and put next to each other in a single representation. To prepare these diagrams, a Savitzky-Golay filter is used to isolate the trends and to filter the effect of external factors out such as use and operation conditions. We compared the trends of SOM frequency and duration in order to determine their similarity in a qualitative manner.

This study is complemented by an analysis of the variation of the SOM frequency (ΔFq) and SOM duration (ΔD), as the failure progresses (i.e. how much the SOM frequency of occurrence (Fq) and the averaged SOM duration (D) changes as the failure progresses). As trends are characterized best by non-parametric normal distribution we have chosen to apply Kruskal-Wallis for comparing their similarity. This analysis was conducted for ΔFq and ΔD separately, thus, two null hypotheses (H_0) were formulated:

- *The observed ΔFq , collected from different experiments and different failure mode, belongs to the same distribution.* That is, they are equal.
- *The observed ΔD , collected from different experiments and different failure mode, belongs to the same distribution.*

The alternative hypothesis (H_1) are: The observed ΔFq , collected from different experiment, and different failure mode, belongs to different distributions. That is, they are different; and, the observed ΔD , collected from different experiment, and different failure mode, belongs to different distributions. If the observed variations of both failure modes belong to the same distribution, H_0 cannot be rejected. If so, it can be inferred that ΔFq and ΔD have low discriminant power. On the contrary, if, H_0 is rejected, we

have not arguments to affirm that ΔFq and ΔD have a low discriminant power, suggesting that these can be used for failure discrimination. P_value is used for falsifying the hypotheses. The threshold considered is $p = 0.05$, where the rejection is determined by $p \leq 0.05$.

5.2.2 Evaluation of the changes in SOM frequency and duration as indicators for failure analysis.

In this step, we aim to explore to what extent Fq and D trends can be used for monitoring the failure progress. Fq and D measurements of the failure forming trends represented by multiple ‘failure progress steps’ (w) are evaluated by linear discriminant analysis. Gradual failures progress over time and their effects on the behavior of the system can become more critical. In this promotion research, we will use the term failure progress step, w to refer to the sampling used for evaluating the failure evolution. SOMs’ frequency (Fq) and duration (D) are evaluated in every w , in order to determine how fast the failure is progressing. Every w collects data corresponding to Fq and D during a fixed time length L . The same time length L is used in all w alike. It enables to determine the regular values of Fq and D in a specific time-lapse providing a reference for evaluating their variation as the failure progresses.

The analysis of failure progress is conducted through failure diagnosis. Failure diagnosis makes use of a classification model that facilitates discrimination of Fq and D data based on the occurring failure mode. To derive a classification model, data of Fq and D , measured when the system performance is no longer acceptable (during the critical threshold of failures), are collected in a predictor vector P , so that, $P = [Fq|D]$. A class vector F composed by all known ‘Failure modes’ (F_r) and the Failure-free (F_f) system’s state is also generated, so that, $F = [F_1, F_2, \dots, F_h]$ where h is the total number of known classes. P vectors generated based on different ‘Failure modes’ (F_r) at their corresponding thresholds are required, in order to train a failure classification model. Once the training process is completed and a classification model is derived, a predictor vector for each step, P_w , is generated in order to determine (by the use of the already derived classification model) whether the observed data belongs to the Failure-free (F_f) set or it indicates a forming failure. Vector P_w is then denoted as:

$$P_w = [Fq_{1_w}, Fq_{2_w}, \dots, Fq_{l_w} | D_{1_w}, D_{2_w}, \dots, D_{l_w}] \quad (5.1)$$

Where l is the total number of SOMs. We will denote as $w = 0$, the step in which failure is detected for the first time throughout its failure forming process. It does not mean failure was started at $w = 0$. It means that, this is the step w in which data corresponding to a dataset from failed operation is no longer classified as Failure-free (F_f). I.e. this is the failure progress step w in which a particular failure becomes detectable.

5.2.3 Exploring potential use of the changing SOM frequency and SOM duration for failure forecasting

This study aims to determine if it is possible to detect and classify failures before step

$w = 0$ by extrapolating the observed Fq and D trends. It intends to determine if these indicators can be used for failure prognosis, where not only the type of failure is predicted, but also the time of occurrence.

Using historical data of Fq and D , we will implement time-series based forecasting models. These models will use forecasted trends for estimating the forming failure type, as well as its Time To Failure (TTF) in a future time interval $[t + 1, \dots, t + b]$. Figure 5.2 graphically illustrates this principle. Based on this, the extrapolation of Fq and D of each SOM (ζ_d), can be denoted by:

$$Fq_{d_{w+b}} = f(Fq_{d_w}, Fq_{d_{w-1}}, \dots, Fq_{d_{w-s}}) \quad (5.2)$$

$$D_{d_{w+b}} = f(D_{d_w}, D_{d_{w+1}}, \dots, D_{d_{w-s}}) \quad (5.3)$$

Where s is the total number of failure steps considered in the estimation of Fq_{w+b} and D_{w+b} . The term $w + b$ denotes the forecasting horizon (b). The forecasting horizon b determines the length of time period to be extrapolated based on the available historical information. A combined analysis of Fq and D for each SOM (ζ_d) will be determined as illustrated in Figure 5.3. Forecasted data is arranged into a predictor vector \hat{P}^{w+h} , that is defined as:

$$\hat{P}^{w+h} = [Fq_{1_{w+h}}, Fq_{2_{w+h}}, \dots, Fq_{l_{w+h}} | D_{1_{w+h}}, D_{2_{w+h}}, \dots, D_{l_{w+h}}] \quad (5.4)$$

Where l denotes the total amount of SOMs (ζ_d), and $w + h$ represents one of the forecasted time periods t' included in the forecasting horizon. If w is the current time, and b is the forecasting horizon, any value between w and $w + b$ is represented by $w + h$. Values $w + h$ are used for feeding the classification model, so that, we can determine $w = 0$. All forecasted observations from $w + 1$ to $w + b$ are delivered to the classification model (see Subsection 5.2.2) in order to determine the forming 'Failure mode' and time to failure. The time instant $t + c$ denotes the first forecasted failure progression $w + h$ that is no longer classified as Failure-free (F_f). It is considered as the Time To Failure, so it will be mathematically noted as c . In this dissertation, Time To Failure (TTF) is interpreted as the time that remains before failure reaches its critical

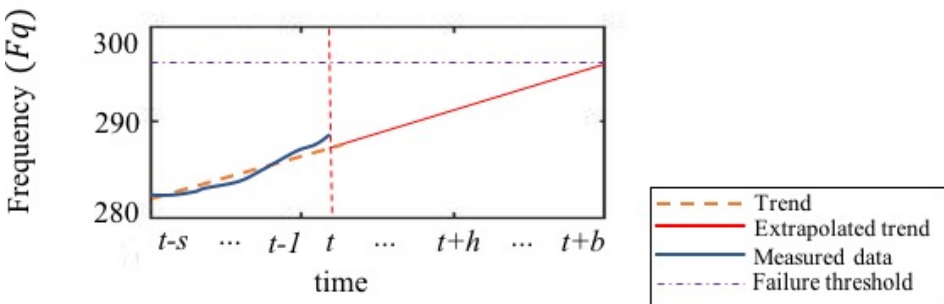


Figure 5.2. Extrapolated trend

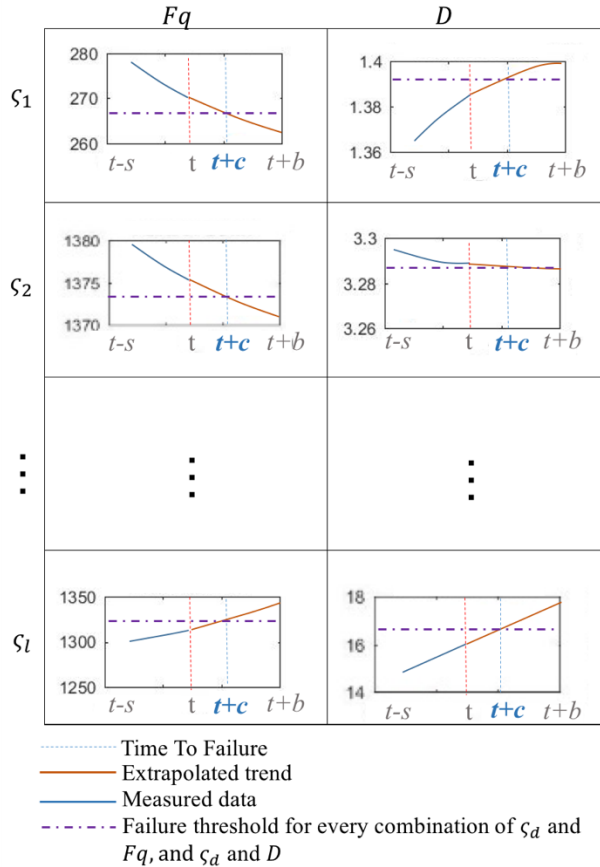


Figure 5.3. Illustrative example of the time-series based forecasting application

threshold. In our case, TTF is represented in ‘failure progress steps’ (w), as every w is equivalent L time instants.

5.3 A pilot study for failure forming analysis considering SOMs behavior

5.3.1 Introduction

The objective of this pilot study is to explore the role of SOMs’ frequency (Fq) and duration (D), as potential source of information for failure diagnosis and forecasting. This pilot study is applied to the model of a water kettle (the one previously described in Chapter 4, Subsection 4.2.1) with self-tuning control capabilities. Using a simulation model for our study, enables us to have better control over the system variables and exclude external disturbances from the data collected. This will enable us to explore the

pure nature of SOMs' frequency (Fq) and duration (D) under different 'Failure modes' (F_r). In Section 5.5, we will extend our study to a real-life application, where the sensitivity of the SOMs' frequency (Fq) and duration (D) for external disturbances will be investigated.

In this pilot study, the same four 'Failure modes' (F_r) described in Chapter 4 (see Subsection 4.2.1) were injected into the model of the water kettle. The failures and their corresponding failure progressions are:

- 'Tank leak' (F_1) with an outflow rate gradually increased from $1.0344e-7$ L/s to 0.004 L/s.
- 'Inflow valve obstruction' (F_2) inflow rate reduced from 0.1261 L/s to 0.063 L/s.
- 'Loss of heating power' (F_3): heating power reduced from 4000 Watts to 0 Watts.
- 'Outflow valve obstruction' (F_4): outflow rate reduced from 0.063 L/s up to 0 L/s.

Failures were manipulated according to a piecewise mathematical function defined as follows. In the first 150 steps, $w = 0 \dots 150$, a Failure-free (F_f) operation was simulated. At step $w = 151$, the failures were introduced according to the characteristics illustrated by Figure 5.4. For a more detailed description about the induced failures, please see Subsection 4.2.1.

Fifteen different experiments for each failure mode were considered in our study, in order to determine the effect of variations under known external conditions. Each of them was composed by a random selection of the scenarios with variations of the use and operative conditions as presented in Subsection 4.2.1. The failure progression process of every experiment also presents minor variations (see Figure 5.4). Scenarios used in this experiment are particular combination of use conditions, operative conditions, and variations on failure progress. Results concerning each of the steps described in Section 5.2 are presented in the following paragraphs.

5.3.2 Analysis of the uniqueness and discriminant power of the observed trends

Following the approach proposed in Section 5.2, Figure A.1 to Figure A.10 in the Appendix A offer a graphical representation of the trend of SOMs' frequency (Fq) and duration (D) as a result of the injected failures. Used as reference, these figures also include the trends of SOMs' frequency (Fq) and duration (D) in Failure-free (F_f) modes. All curves were plotted in the same scale to facilitate their visual comparison. Each plot overlaps the curves of Fq (in the case of Figure A.1 to Figure A.5) or D (in the case of Figure A.6 to Figure A.10) corresponding to the fifteen simulation-based experiments.

The regular behavior of the system characterized by the curves Fq and D corresponding to the Failure-free (F_f) case, presents a steady characteristic (see Figure A.1 and Figure A.6). Although there are minor fluctuations caused by variations of the use and operative conditions, there were no long-term trends observed in any of the SOMs. In the case, when failures were injected into the simulation, these trends became visible.

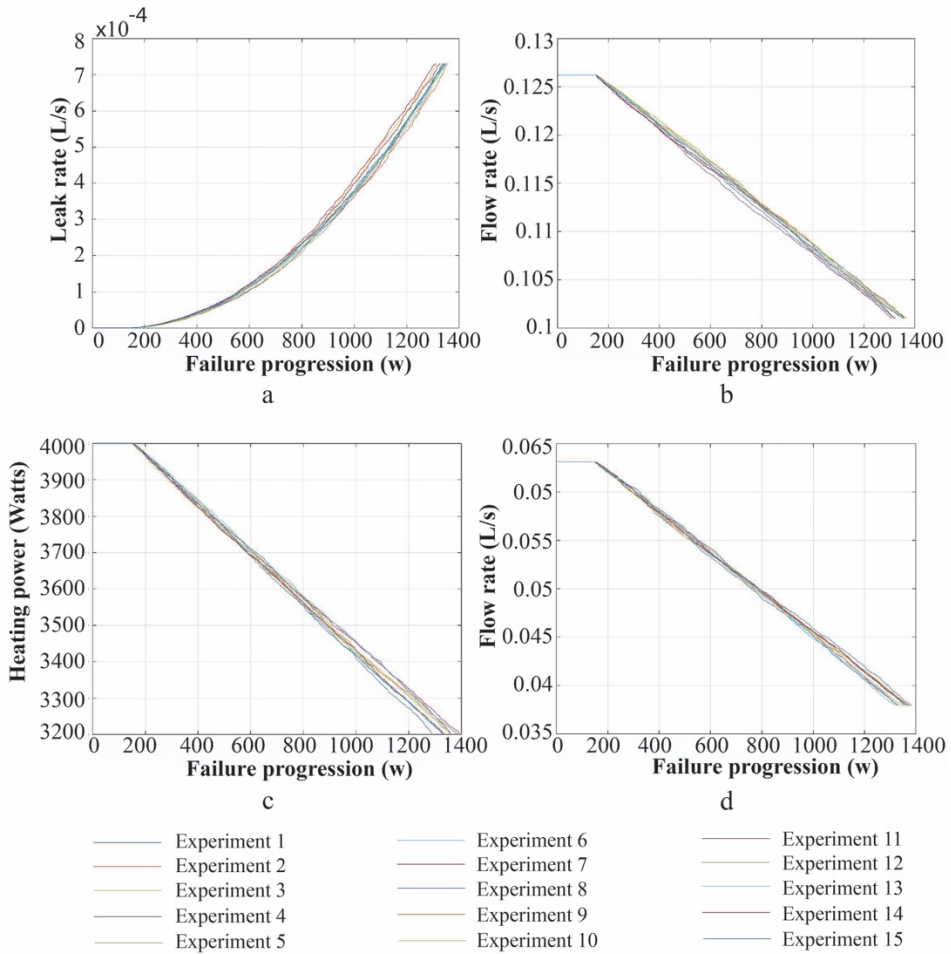


Figure 5.4. Failure progress (a) tank leak, (b) inflow valve obstruction, (c) loss of heating power, (d) outflow valve obstruction

There were some specific SOMs that presented an increasing or decreasing trends that can only be explained by the forming failure. For instance, the ‘Loss of heating power’ (F_3) generates long term trends on Fq_{ζ_1} , Fq_{ζ_2} , Fq_{ζ_7} , $Fq_{\zeta_{11}}$, $Fq_{\zeta_{15}}$ and D_{ζ_2} (see Figure 5.5), where Fq_{ζ_d} and D_{ζ_d} , are the frequency and duration of a particular SOM (ζ_d).

The figures obtained also suggest the pattern composed by the observed trends during the same failure mode is unique for each failure type, as none of them present the same combination of trends. For example, ‘Loss of heating power’ (F_3) only $Fq_{\zeta_{15}}$ presents a trend similar to the one presented by $Fq_{\zeta_{15}}$ in ‘Outflow valve obstruction’ (F_4). It evidences that in the analyzed failures, ‘Loss of heating power’ presents a characteristic pattern of trends in Fq and D , that differ from the ones presented by the rest of the analyzed failure modes.

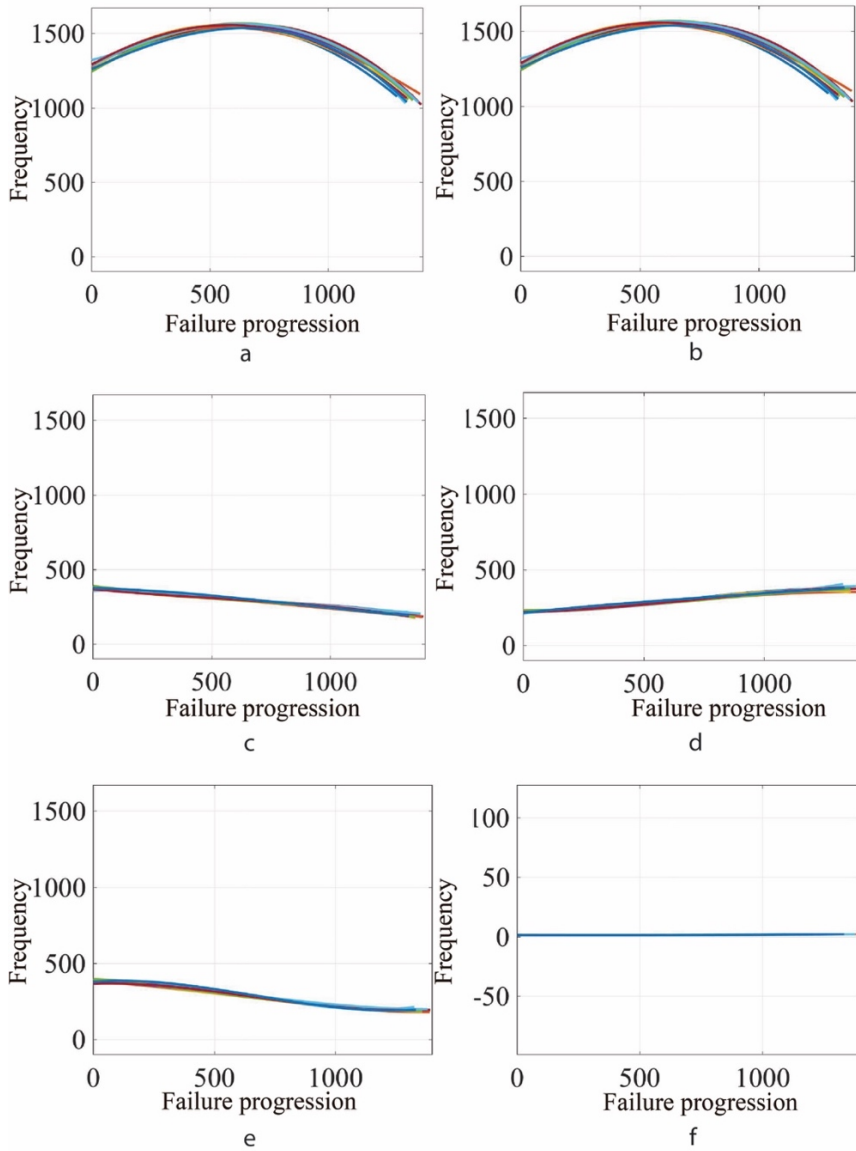


Figure 5.5. Filtered trends corresponding to F_3 . (a) Fq_{ζ_1} , (b) Fq_{ζ_2} , (c) Fq_{ζ_7} , (d) $Fq_{\zeta_{11}}$, (e) $Fq_{\zeta_{15}}$, (f) D_{ζ_2} .

All analyzed failure modes present at least one characteristic uptrend or downtrend, either in Fq or in D . The ‘Loss of heating power’ (F_3) is the failure mode that presents the highest number of SOMs with trends. Considering that the system is not capable to keep ‘Water temperature’ (S_{ζ_1}) on the stated levels, it should increase the time periods in which the ‘Heater’ (S_{A_3}) is on, and, as a consequence, frequency and duration of other SOMs will decrease, as it is case of ζ_1 , ζ_7 , ζ_9 and ζ_{15} . On the contrary, the ‘Inflow valve

obstruction' (F_2) presents less amount of SOMs with long-term trends, as we only observed trends in D_{ζ_8} (where the 'Inflow valve' (S_{A_1}) and the 'Heater' (S_{A_3}) are active at the same time). This trend is triggered by the increased duration of the inflow valve's opening, that is explained by the reduction on the inflow rate caused by the failure. The controller keeps the 'Inflow valve' (S_{A_1}) longer active in order to reach the set-point of the water level. F_2 is only manifested on D_{ζ_8} as its effect on other SOMs is compensated by the activation of the 'Additive injection valve' (S_{A_4}), which also injects water to the tank, even earlier than 'inflow valve' S_{A_1} .

The trends of Fq and D present a consistent pattern for all the simulation-based experiments in each particular failure mode. Despite minor differences manifested in local peaks and variations in the measured values, the long-term trends are consistent in all the fifteen cases. The cases in which SOMs do not present any upward or downward trends are also consistent between the scenarios. The boxplots presented in Figure 5.6, Figure 5.7, Figure 5.8, and Figure 5.9 supports the above presented conclusions. They represent the variations of Fq and D of the analyzed scenarios in every SOM, as the failure progressed, so that:

$$\Delta Fq_{\zeta_d} = \left(\frac{Fq_{\zeta_d, w=1} - Fq_{\zeta_d, w=1290}}{1290} \right) \quad (5.5)$$

The failure progress step $w = 1290$ was chosen as it corresponds to the last measurement available of the experiment 2, as it can be seen in Figure 5.4.

The boxplots presented from Figure 5.6 to Figure 5.9 prove that experiments present a low variance among them, on all analyzed ΔFq and ΔD . It means that, most of the variations of Fq and D , corresponding to the same failure mode, were similar between the tested scenarios. The patterns created based on the variation of the frequency and duration of all SOMs, from the same failure mode, presented significant differences with respect to the patterns corresponding to the other failure modes. Table 5.2 and Table 5.1 present the results of the statistical analysis conducted for Fq and D respectively. It is a pair-wise comparison of the variation observed on Fq and D (see Eq 5.5) parameters of the analyzed failure modes. Kruskal Wallis adjusted with a Bonferroni correction was used. Note, that the null hypotheses (H_0) presented in Subsection 5.2.1 states that (i) "*the observed ΔFq , collected from different experiments, and different failure mode belongs to the same distribution*"; and, (ii) "*the observed ΔD , collected from different experiments, and different failure mode belongs to the same distribution*". The null hypotheses are rejected when $p \leq 0.05$ and it cannot be rejected whenever the $p \geq 0.05$.

In order to ease the comparison of the results, we implemented a visual representation of failure indicator (similarly to the one proposed in Chapter 4). However, in this case, our FI matrix is a $2 \times \zeta_d$ matrix where the first row corresponds to the Fq indicator and the second row is the D indicator. Columns are the number of SOM, as it is the case of the FI matrices analyzed in chapter 4 (for the current case, the FI matrix is a 2×10). The pair-wise comparisons that involve the Failure-free (F_f) cases in Table 5.2 and Table

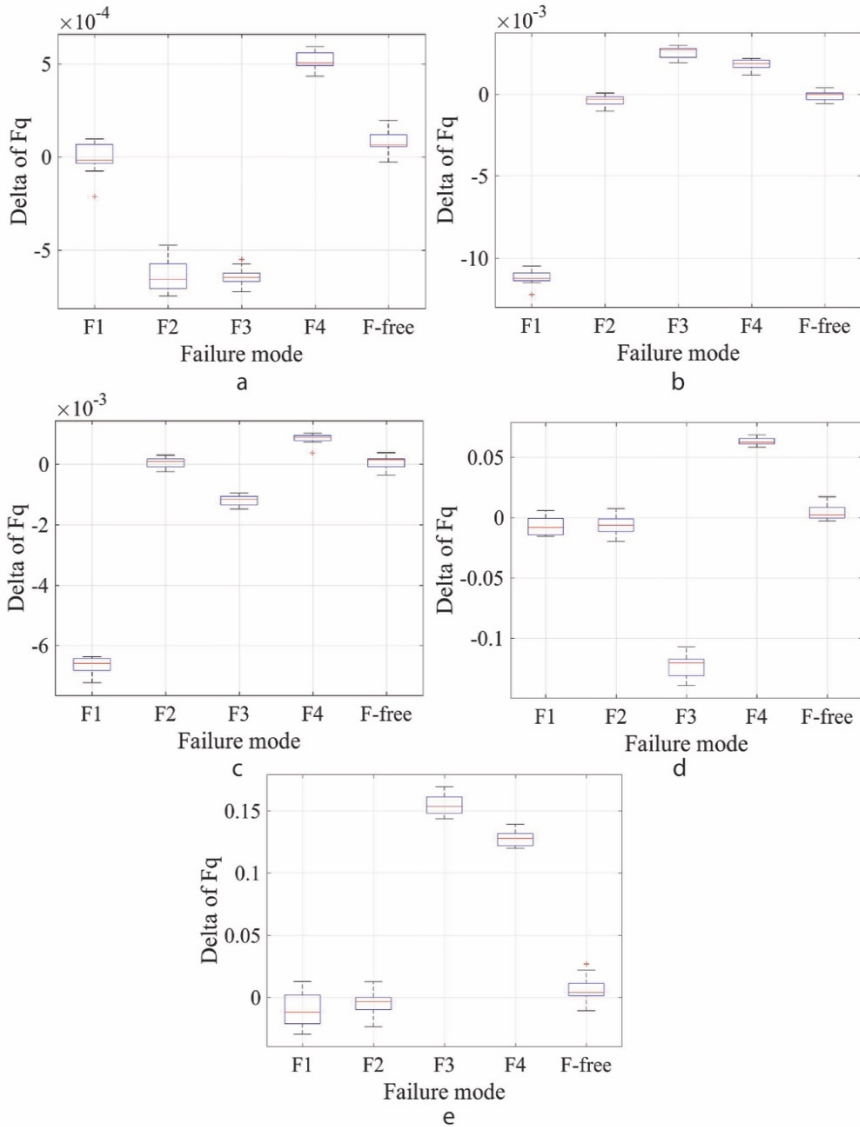


Figure 5.6. Comparison of F_q distribution based on failure modes. (a) c_8 , (b) c_9 , (c) c_{10} , (d) c_{11} , (e) c_{15}

5.1, are graphically represented in a FI matrix. When the comparison of F_q indicator of a particular SOM (c_d) between the Failure-free (F_f) case and one of the

analyzed ‘Failure modes’ F_r , the p -value (p) is analyzed and, if:

- $p \leq 0.05$, the cell FI_{1,c_d} is changed to red color. This indicates that there is a significant difference between the F_q corresponding to the Failure-free (F_f) case and the analyzed F_r .
- $p > 0.05$, the cell FI_{1,c_d} is colored green.

On the other hand, for the comparison of D indicator of a particular SOM (ζ_d), the p -value is analyzed and, if:

- $p \leq 0.05$, the cell FI_{2,ζ_d} is changed to red.
- $p > 0.05$, the cell FI_{2,ζ_d} is colored green

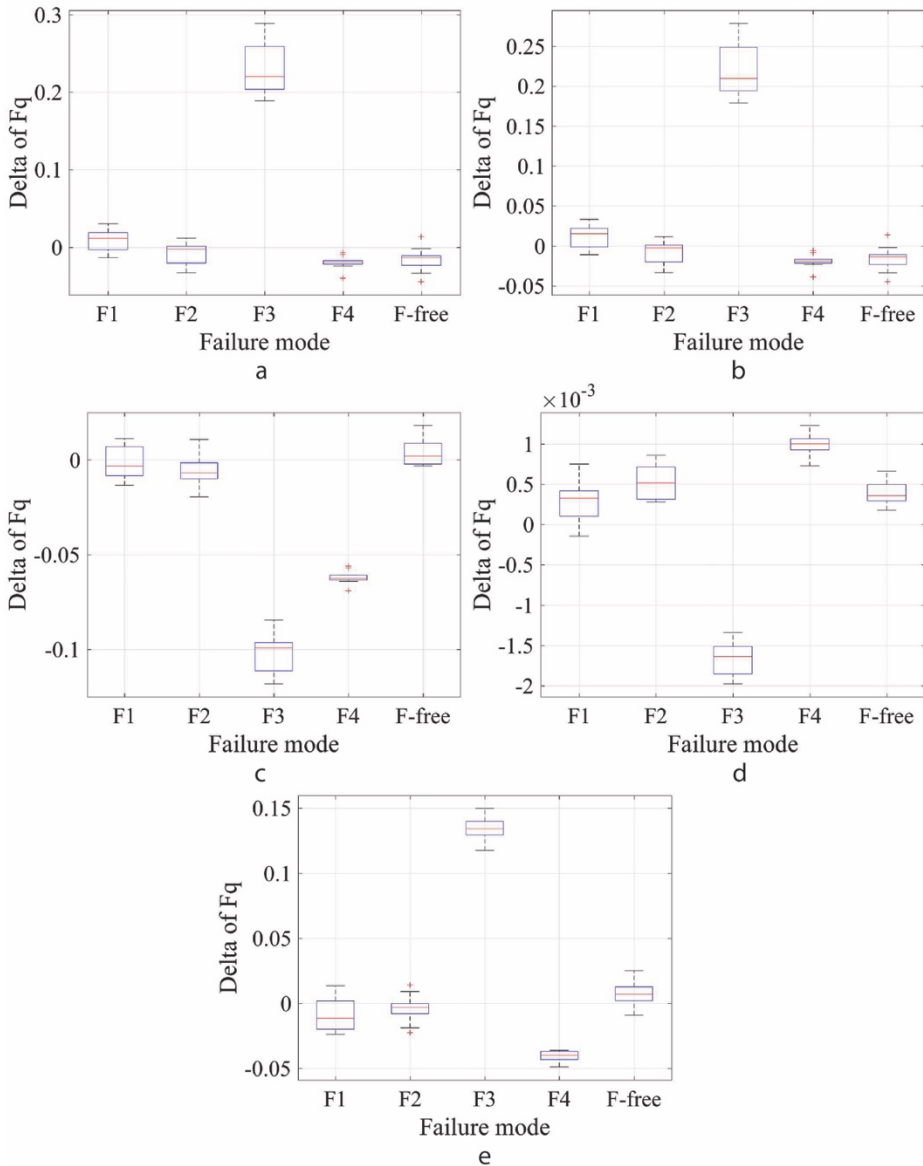


Figure 5.7. Comparison of Fq distribution based on failure modes. (a) ζ_1 , (b) ζ_2 , (c) ζ_3 , (d) ζ_4 , (e) ζ_7

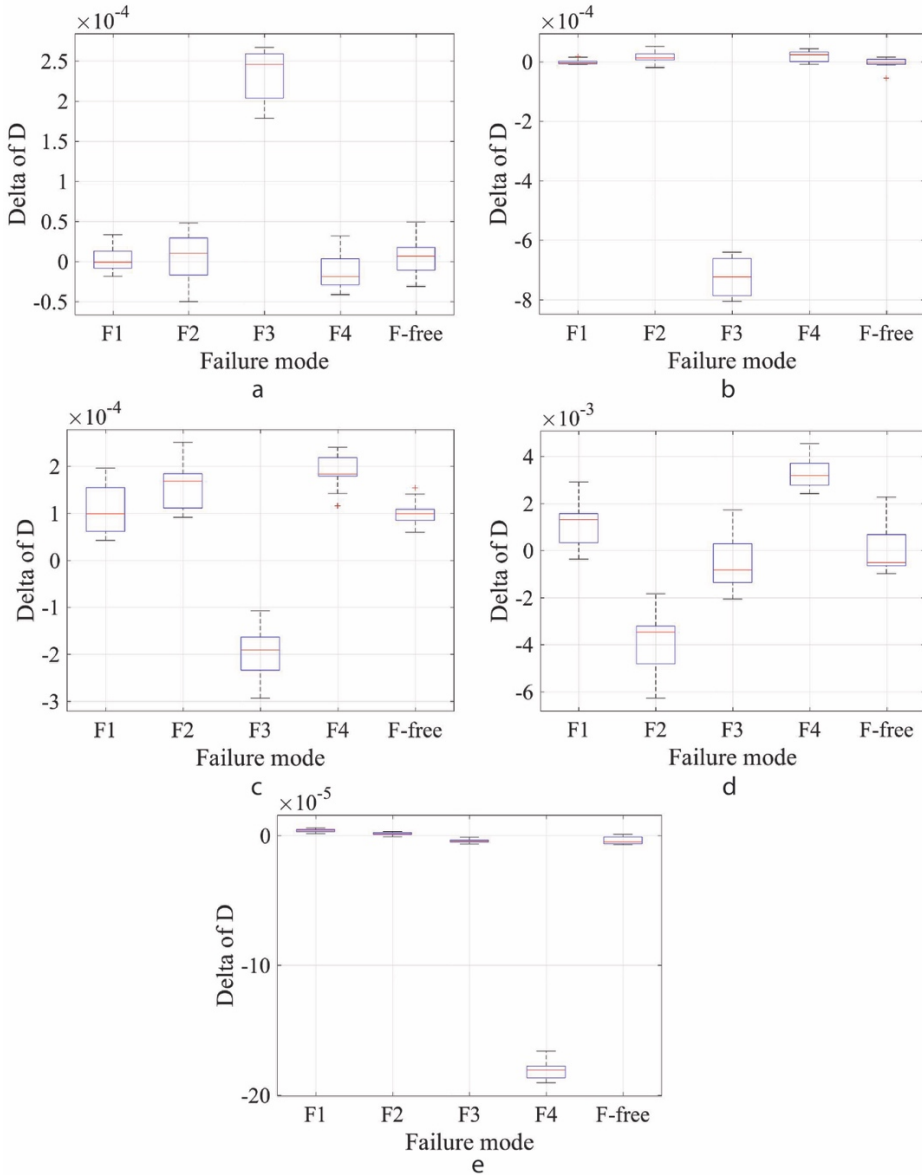


Figure 5.8. Comparison of D distribution based on failure modes. (a) ζ_1 , (b) ζ_2 , (c) ζ_3 , (d) ζ_4 , (e) ζ_7

A quick view on the obtained matrices (see Figure 5.10) is enough to conclude that every ‘Failure mode’ (F_r) has a characteristic pattern that differs from the rest of the failure modes. It implies that the combined analysis of the evolution of Fq and D have sufficient discriminant power for the case studied in this experiment, and that can be used for failure diagnosis.

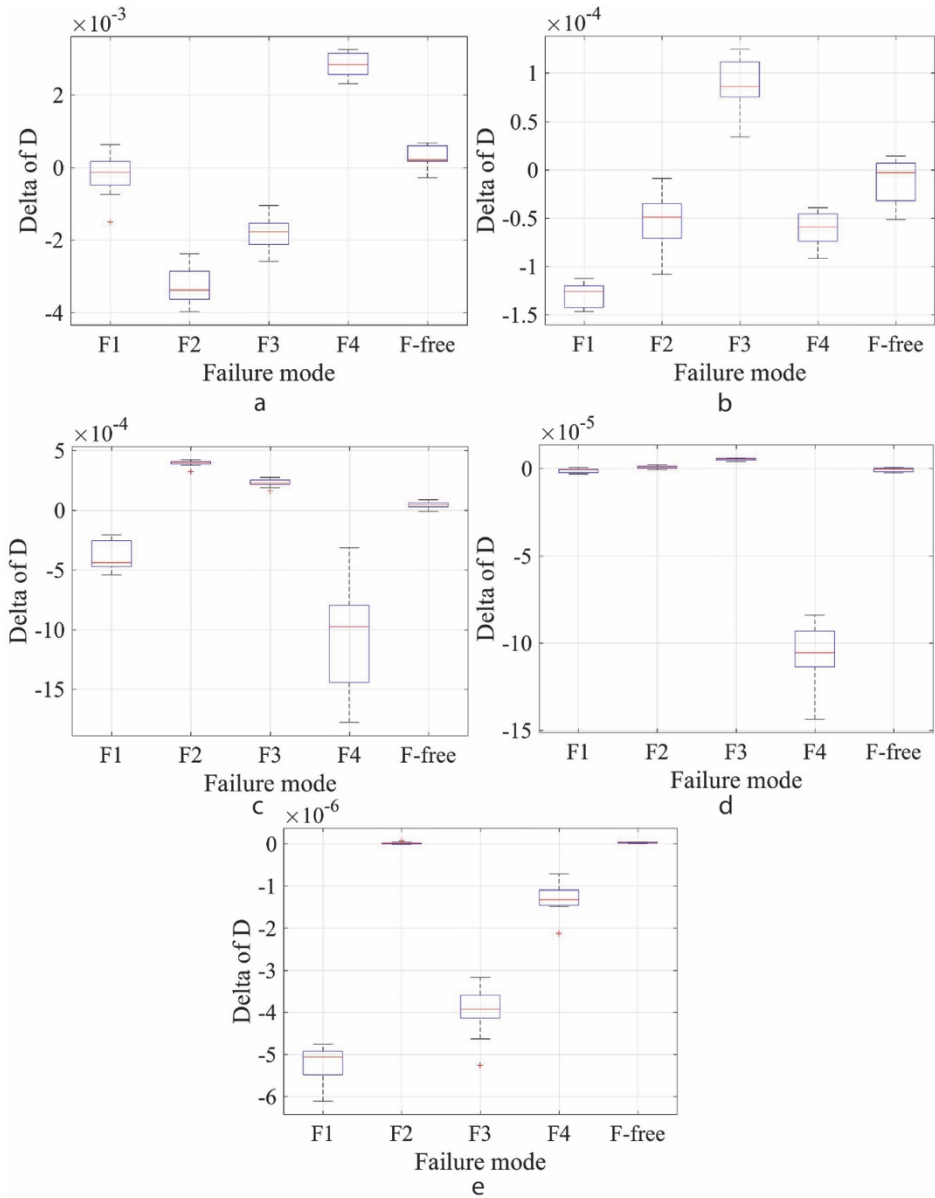


Figure 5.9. Comparison of D distribution based on failure modes. (a) ζ_8 , (b) ζ_9 , (c) ζ_{10} , (d) ζ_{11} , (e) ζ_{15}

Table 5.1. Results of the statistical test for SOM duration

Compared Failure modes	Duration P-value									
	SOM1	SOM2	SOM3	SOM4	SOM7	SOM8	SOM9	SOM10	SOM11	SOM15
F1-F2	1.00	0.20	0.42	0.00	0.91	0.00	0.01	0.00	0.19	0.00
F1-F3	0.00	0.01	0.00	0.17	0.00	0.27	0.00	0.00	0.00	1.00
F1-F4	1.00	0.18	0.01	0.16	0.00	0.01	0.07	0.86	0.04	0.00
F1-Ffree	1.00	1.00	1.00	1.00	0.00	1.00	0.00	0.69	1.00	0.00
F2-F3	0.00	0.00	0.00	0.07	0.05	0.69	0.00	0.59	0.19	0.00
F2-F4	1.00	1.00	1.00	0.00	0.00	0.00	1.00	0.00	0.00	0.14
F2-Ffree	1.00	0.77	0.16	0.00	0.03	0.00	0.47	0.00	0.34	1.00
F3-F4	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.50
F3-Ffree	0.00	0.00	0.01	1.00	1.00	0.00	0.17	0.35	0.00	0.00
F4-Ffree	1.00	0.70	0.00	0.00	0.05	0.31	0.15	0.00	0.02	0.02

Table 5.2. Results of the statistical test for SOM frequency

Compared Failure modes	Frequency P-value									
	SOM1	SOM2	SOM3	SOM4	SOM7	SOM8	SOM9	SOM10	SOM11	SOM15
F1-F2	0.41	0.25	1.00	1.00	1.00	0.01	0.20	0.00	1.00	1.00
F1-F3	0.11	0.15	0.00	0.02	0.00	0.02	0.00	0.59	0.03	0.00
F1-F4	0.00	0.00	0.00	0.00	0.05	0.01	0.00	0.00	0.00	0.00
F1-Ffree	0.03	0.01	1.00	1.00	0.82	1.00	0.00	0.00	0.21	0.38
F2-F3	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.06	0.02	0.00
F2-F4	1.00	1.00	0.07	0.02	0.01	0.00	0.01	0.05	0.00	0.00
F2-Ffree	1.00	1.00	0.53	1.00	1.00	0.00	1.00	1.00	0.32	1.00
F3-F4	0.00	0.00	0.68	0.00	0.00	0.00	1.00	0.00	0.00	0.69
F3-Ffree	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.02	0.00	0.00
F4-Ffree	1.00	1.00	0.00	0.00	0.00	0.27	0.26	0.11	0.21	0.10

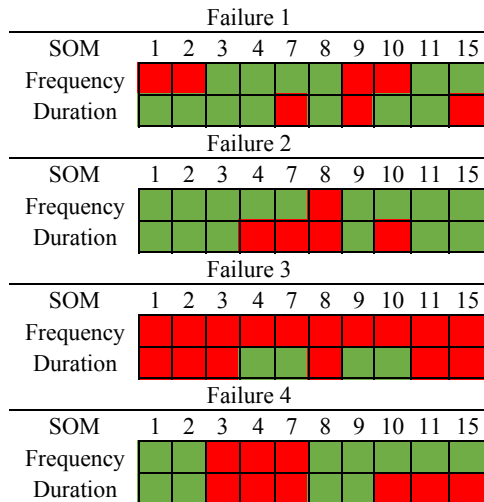


Figure 5.10. Obtained FI matrices for Fq and D indicators

5.3.3 Evaluation of the SOMs' frequency and duration as indicators of the failure forming process.

The results presented in the last section showed that the patterns formed by SOMs' frequency (Fq) and duration (D) can be used to characterize failures. They present a consistent behavior when analyzing multiple cases subjected to the same failure mode and they differ from the patterns presented by other failure types. These results demonstrated the discriminant power of the SOMs' frequency (Fq) and duration (D). Under this finding, we proposed to use these indicators as input for failure identification. In this section, we will investigate the discriminant power of Fq and D in the complete failure forming process. For this purpose, we used Linear Discriminant Analysis (LDA) for obtaining a failure classification model. This model was derived from data measured at the critical threshold of every 'Failure modes' (F_r), in which the critical thresholds are defined as:

- F_1 's threshold: leak rate of 0.0002 l/s in the case of the tank leak, which occurs at step $w \approx 760$.
- F_2 's threshold: reduction of 0.0126 l/s in the inflow rate due to the inflow valve obstruction, which takes place at step $w \approx 754$, when the inflow rate reduces to 0.113 l/s. 10% reduction on the normal performance.
- F_3 's threshold: reduction of 400 Watts in the heating power, that occurs at step $w \approx 752$, when the heating power reaches 3600 Watts. 10% reduction on the normal performance.
- F_4 's threshold: reduction of 0.0126 l/s in the outflow rate caused by an obstruction in the outflow valve, which is manifested at step $w \approx 690$, when the value of outflow rate is 0.052 l/s. A reduction of 20% on the normal performance.

These thresholds do not necessarily imply that the system is instable, but they indicate the time instant in which the desired performance of the system cannot be met. Data of the regular system operation was also gathered for training the classification model.

We used a dataset composed by 250 samples (50 per failure mode) for training and testing the classification model. 50 subsets of the samples were randomly generated, where 70% of data corresponding to each failure mode and the Failure-free (F_f) case was used for training purposes. The remaining 30% was used for testing. The best model was selected based on the success classification rate in every considered class, so that, none of the failure modes would present a low classification rate. The confusion

Table 5.3. Confusion matrix of classification obtained by analyzing the measured Fq and D parameters.

Failure mode	F_1	F_2	F_3	F_4	Failure-free
F_1	12	1	0	0	2
F_2	0	10	0	0	5
F_3	0	1	12	0	2
F_4	0	0	0	15	0
Failure-free	2	0	0	0	13

matrix of the test results are presented in Table 5.3. Rows represent the actual classes while columns represent predicted classes.

Based on the results summarized in Table 5.3, we infer that the classification model has limitations in distinguishing between the Failure-free (F_f) cases and failure mode F_2 (Inflow valve obstruction), in an early stage of failure forming. The reason of this is that F_2 is only observable in the failure indicators when the ‘Inflow valve’ (S_{A1}) is opened. However, for this model only 20% of the total number of SOMs has an open state for ‘Inflow valve’ (S_{A1}). This observation implies that the failure indicators of SOMs’ frequency (Fq) and duration (D) are sensitive for the number of SOMs in which the failures are manifested. The activation of the ‘Additive injection valve’ (S_{A4}) also had an effect on correct classification of F_2 . The control settings cause that this valve refills the tank earlier than the ‘Inflow valve’ (S_{A1}) itself. It prevents observing significant changes of the frequency of opening of the ‘Inflow valve’ (S_{A1}), as well as in its duration time. Despite that these results are not optimal for the early failure diagnosis, we will proceed to conduct a classification process in every failure progression step in order to explore in which stage the failure can be reliably identified by the proposed classification process.

We took the 15 scenarios presented in Subsection 5.3.1 in order to evaluate in which step w of the failure progression process, the failure was detected and properly classified. The results of our investigation are graphically presented in a bar-plot in Figure 5.11. Every single bar represents the distribution of the 100% of the analyzed experiments (15 scenarios) in step w . A code of colors indicates the predicted failure modes, as it is shown in each subfigure’s legend. The distribution of the analyzed experiments is presented in the y-axis, while the x-axis represents both, the failure progression (w) and the failure size. Failure size is represented in percentage and it was determined by considering the progression step $w = 1290$ as the failure size=100%.

Let’s consider an example with the F_1 cases (Figure 5.11(a)). If we take in the abscissa, the progression $w = 500$ (Failure size=11%), it can be seen that, at this point, this bar is blue up to 30% (y-axis) and then, it becomes red (wine) between 30% to 100%. It means that 30% of the cases, at that point, were predicted as failure 2 (F_2), when they actually are F_1 . At this same point, the remaining 70% of the cases were predicted as Failure-free (F_f). In general terms, what this figure is indicating is that the analyzed data is classified as Failure-free (F_f) while progressions are running and, around progression $w = 460$ (Failure size=8.43%), it can be seen that “something” is happening. Initially the system classifies as F_2 but then, around step $w = 510$ (Failure size=11.14%), F_1 starts to appear. As seen in Figure 5.11(a) this happens only in F_1 plot. The other plots, from all analyzed failures, shift progressively from the Failure-free (F_f) class to their actual failure mode.

In all cases, they present a satisfactory classification rate after progression $w = 550$ where the predicted ‘Failure modes’ (F_r) coincide with the actual failure in more than 70% of the cases (see Figure 5.11). Most cases present a clear transition from the Failure-free (F_f) case to the correct ‘Failure mode’ (F_r) without presenting false negatives, except for the case of ‘Water leak’ (F_1), in which there were some misclassifications in the range of $w = 460$ and $w = 540$. The highest rate of misclassification reached a 70% of the predictions at $w = 510$. The reason of these false

negatives can be explained by the similarity between Fq and D indicators of F_2 , and that of the Failure-free (F_f) cases. This is perfectly normal in diagnosis (even from other disciplines of science) as failure symptoms are weak at the incipient failure stages, hindering their discrimination from other failure modes. Failure symptoms strengthens as the failures progress increasing their discriminative power.

The use of SOMs' frequency (Fq) and duration (D) enabled to predict the 'Tank leak' (F_1) in more than 70% of the cases from $w \approx 548$ (13.3% of failure evolution considering $w = 1290$ as the Failure size=100%), which is 212 steps w before of its critical threshold ($w \approx 760$). In the case of the 'Inflow valve obstruction' (F_2) the failure is detected from step $w \approx 334$ (2.87% deviation from the regular flow rate and Failure size=16%), which is around 420 steps before of its critical threshold (at $w \approx$

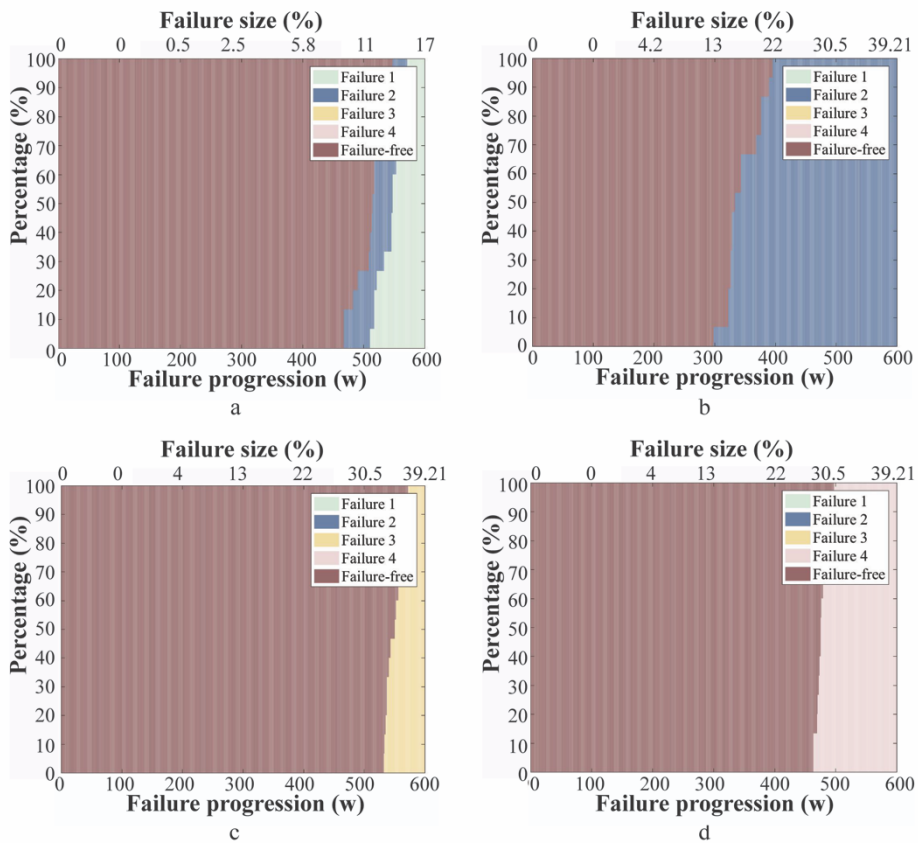


Figure 5.11. Evolution of failure prediction for the analyzed failure modes. (a) evolution of the diagnosis process of F_1 (water leak); (b) evolution of the diagnosis concerning to data corresponding to class F_2 (inflow valve obstruction); (c) evolution of the diagnosis of data belonging to F_3 (loss of heating power); (d) data belonging to the class F_4 (obstruction of the outflow valve).

754). The ‘Loss of heating power’ (F_3) is detected properly from $w \approx 551$ (6.71% deviation from the regular heating power and Failure size=35.29%), 201 steps before of its critical threshold ($w \approx 752$); and, the ‘Outflow valve obstruction’ (F_4) is detected from $w \approx 476$ (10.8% deviation from the regular flow rate and Failure size=28.47%), which is 214 steps before of its critical threshold ($w \approx 690$).

The overall results concerning the analysis SOMs’ frequency (Fq) and duration (D) throughout the failure forming process are satisfactory. The use of these failure indicators for failure prediction demonstrated that classification may be timely and reliable with regards to determining the forming failure mode. In all analyzed cases, failure effects are observable several steps before of achieving their critical threshold, which is the point where data used for deriving the classification model was taken. It demonstrates that, under the ideal conditions in a simulated scenario the Fq and D indicators are suitable and effective for detecting failure at an early stage.

5.4 A pilot study for failure forecasting considering SOMs behavior

In this section, we will explore the potential use of SOMs’ frequency (Fq) and duration (D) for failure forecasting. This investigation also uses data generated based on the simulated model of the kettle. It was explored in the previous section, that Fq and D indicators have unique trends in some SOMs as system degrades. Results of failure prediction reported in Subsection 5.3.3 lead us to infer that feeding the already derived classification model with extrapolated or forecasted data of Fq and D may contribute to predict failures earlier.

5.4.1 Failure forecasting in the pilot-study of a simulated kettle model

Our study was conducted with the same 15 experiments of the same Failure Modes (F_r) used in Subsection 5.3.3. We considered ‘dynamic windows’ with a length of $s = 100$, in order to derive the forecasting models, i.e. datasets composed of 100 samples of historical and consecutive Fq or D measurements. These datasets move one sample forward every time a new and more recent observation is added to the dataset, but it also removes the oldest observation in order to keep the size of dataset constant. Every time a new observation is added, a new forecasting model is generated and a set of extrapolated (or forecasted) data is derived with a sample size equal to the horizon size. The forecasting horizon ($r = 100$) means that every time that a forecasting model was derived, Fq and D has been extrapolated for 100 steps (w) ahead based on historical data from the past. We used this relatively large horizon length to explore how reliably the time to failure can be forecasted based on data from early stage of failure forming. Figure 5.12. presents both the evolution of the time to failure (TTF) and the forecasted ‘Failure mode’ (F_r).

Figures on the left (Figure 5.12(a), Figure 5.12(c), Figure 5.12(e), Figure 5.12(g)) are arranged as follows: the x-axis represents the failure progression steps w , while the y-axis is the estimated time to failure. Note that values on the y-axis are in the range of 0

and 100. The limits of x-axis vary depending on the analyzed failure mode. The reason of these variations is that there were no failure predictions on the forecasting horizon before of $w = 369$ (Failure size=4.64%) in the case of F_1 , $w = 190$ in F_2 (Failure size=3.62%), $w = 436$ in F_3 (Failure size=25%) and $w = 368$ in F_4 (Failure size=19%). The results of different simulation scenarios are combined into a single diagram, where each individual curve represents one of the 15 experiments. The vertical lines on the right side of the plots, are the failure's critical threshold for every experiment. The color of the threshold line is the same as the color of the curve corresponding to its respective experiment.

Figure 5.12(b), Figure 5.12(d), Figure 5.12(f) and Figure 5.12(h) are bar-plot diagrams of the failure progression process. However, figures presented in this section report the *forecasted failure mode* instead of the predicted one. Note that the difference between prediction and forecasting (or prognosis) is that former makes use of measured data, while the latter uses extrapolated data (that is obtained based in historical records) for determining failure occurrence.

Figure 5.12 shows that for all studied 'Failure modes' (F_r) the Time To Failure (TTF) decreases as the failure progresses. This is a gradual process where the first Time To Failure (TTF) is predicted 99 steps w ahead for all analyzed 'Failure modes' (F_r). That is, when the presence of failure is detected for the first time. For instance, Figure 5.12(a) shows that the first failure is forecasted at $w = 369$. In this particular case, TTF is forecasted to appear 99 steps ahead, at step $w = 468$. A TTF = 1 means that the failure is expected to occur in the next step, i.e. if w is the current step on which the forecasting model is derived, it is expected the system to fail at the next step, $w + 1$.

The diagrams on the left and right side of Figure 5.12 provide information about the TTF and the forecasted 'Failure mode' (F_r), respectively. Failure diagnosis based on the figures on the right correctly moved from failure-free (F_f) diagnosis to the right 'Failure mode' (F_r). Only 'Tank leak' (F_1), was misclassified as 'Inflow valve obstruction' at first between step $w = 369$ and step $w = 425$. After step $w=425$ (failure size=6.87%) this failure was also correctly forecasted.

We averaged the 'failure progress' steps w where the critical threshold of every 'Failure mode' (F_r) occurred. We also averaged the predicted TTF and the forecasted TTF to be able to compare the accuracy of these models. The results are presented in the Table 5.4. Forecasting based on SOMs' frequency (Fq) and duration (D) can diagnose failure significantly earlier in all of the analyzed cases than failure prediction. According to Table 5.4, 'Failure mode' (F_r), failure was forecasted in the case of each with a number of steps before, through extrapolated data, than by using the measured data. However, the average forecasted TTF of each 'Failure mode' (F_r) predicts the occurrence of failure earlier than the actual failure.

In all analyzed cases, the observed trends of the SOMs' frequency (Fq) and duration (D) indicators provide relevant information that enables failure forecasting. The observed results demonstrate that Fq and D can be used to forecast the type of forming failure from early stages. However, the obtained TTF results were somewhat inaccurate. These results reveal that obtaining TTF estimations is preferable to avoid failure occurrence, rather than not having information at all about forthcoming system failures. However, a premature TTF may cause the replacements of components that can still be used longer.

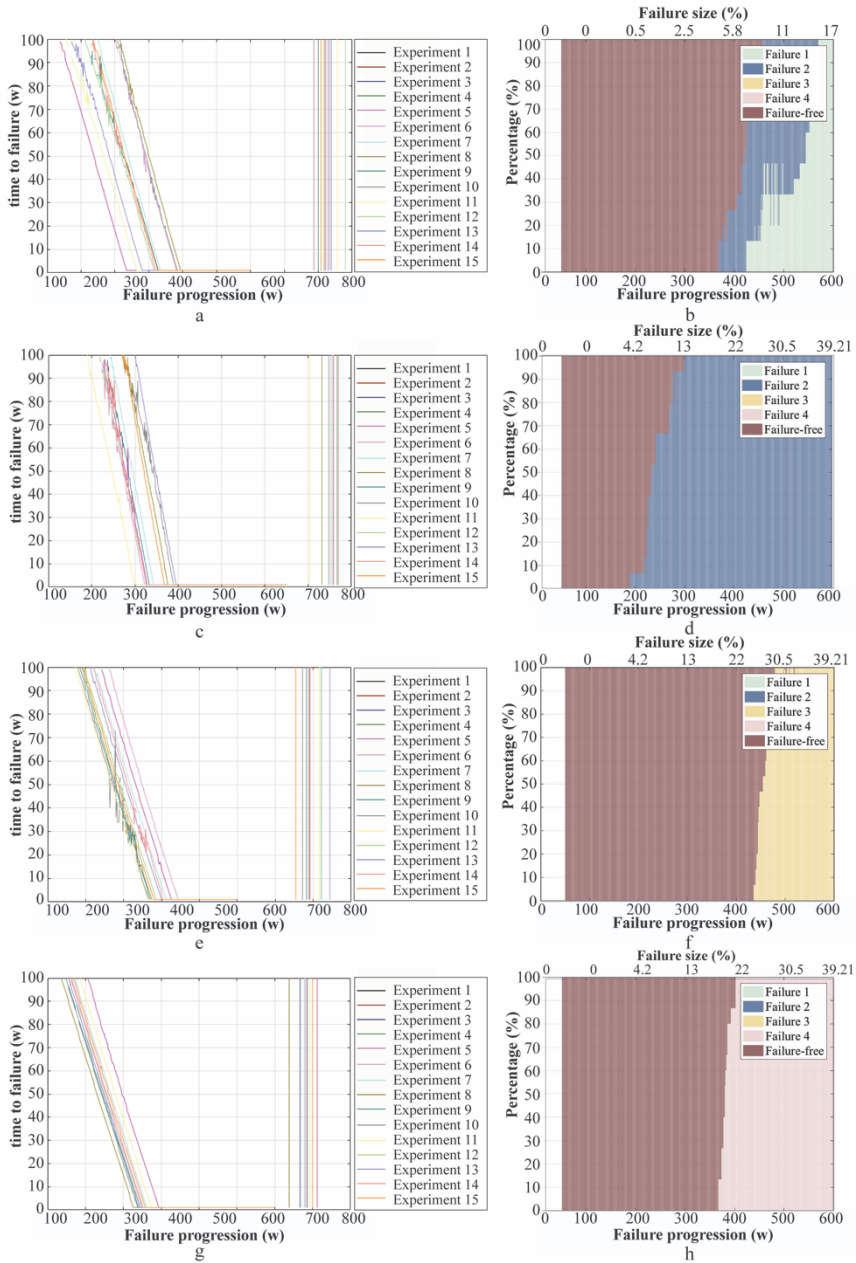


Figure 5.12. Failure forecasting for the analyzed failure modes. (a) TTF of class of F_1 (water leak); (b) forecasted failure mode of class F_1 ; (c) TTF of class of F_2 (obstruction of the inflow valve); (d) forecasted failure mode of class F_2 ; (e) TTF of class of F_3 (Loss of heating power); (f) forecasted failure mode of class F_3 ; (g) TTF of class of F_4 (obstruction of the outflow valve); (h) forecasted failure mode of class F_4

Table 5.4. Comparison of the predicted and forecasted failure diagnosis

	<u>Critical Threshold</u>	<u>Predicted</u>	<u>Forecasting</u>
F_1	760	548	504
F_2	754	334	243
F_3	752	551	456
F_4	690	476	383

These results may affect the trade-off between safe operation and optimized maintenance, but, still, it is important to have this information for decision-making processes.

5.4.2 Conclusions about the pilot implementation

The results of the analysis of SOMs' frequency (Fq) and duration (D) through the use of simulation-based experiments of a kettle model demonstrated that:

- The frequency and duration of System Operation Mode (SOM) conveys consistent information about failure modes. They are manifested through trends of Fq or D , which offers sufficient discriminant power for distinguishing different failure types.
- The combined analysis of SOMs' frequency (Fq) and duration (D) enables failure prediction in the entire failure forming process. The consistent behavior of the observed trends on the system operation modes enables analysis of the failure progress and evaluation of its severity before of reaching its critical threshold.
- The extrapolation of the SOMs' frequency (Fq) and duration (D) trends enables failure forecasting. The implemented approach proved to be able to diagnose failures, even earlier than by using the measured data (non-extrapolated data). It demonstrates a potential application of SOM that is failure analysis and forecasting for evaluating system degradation and failure progression. However, the estimated Time To Failure (TTF) is did not provide accurate results.

These conclusions were derived from a computational model under idealistic conditions. As for the pilot study conducted in Section 4.4, it responds to a limited number of system variables, and system actuators, which limits, in turn, the amount of SOMs. The success of the results may be affected as the system becomes more complex. Moreover, the lack of disturbances caused by the surrounding environment can provide explanation to the positive results obtained so far. However, in a real-life implementation, external factors can have a significant effect on SOM transitions. We argue that a changing environment modify the system's compensatory actions that could be misinterpreted as failure symptoms. It would detract reliability to any type of implementation of SOM frequency and duration as failure indicators.

The implemented failures so far, were induced by intervening the equations that describe the behavior of the kettle model. However, failure manifestations can be stressed or attenuated by external conditions too. These unforeseeable and uncontrollable factors can lead to inconsistent trends of the frequency and duration of SOMs, when subjected to the same failure mode. It would hinder their use as failure

indicators, as it would decrease their discriminant power. It would lead to wrong failure diagnosis and failure forecasting.

The results obtained through the implementation of this controlled test confirm our initial assumptions. This pilot-study worked as an early verification of such assumptions about SOMs behavior. However, the evaluation of the consistency of the observed trends of the SOM frequency and duration, their discriminant power, and their potential for failure forecasting should be also investigated in a real-life context. The following sections presents a study that investigates the role of SOM frequency and duration in failure diagnosis and forecasting with a real-life application, that is a greenhouse testbed.

5.5 A real-life case study for failure forecasting based on SOMs frequency and duration

The analysis conducted in Section 5.3 showed that System Operation Modes (SOMs) reflect failure effects through variations in their frequency of occurrence (Fq) and duration time (D). These variations provided characteristic values of Fq and D per 'Failure Mode' (F_r), if analyzed in a combinatorial way, they enable failure diagnosis, prediction and forecasting. The variations of both indicators presented strong trends in some SOMs that reflect failure evolution. It was found that long-term trends, approach to Fq and D values that characterize the occurring 'Failure Modes' (F_r). It enables forecasting. An important question arises from the obtained results, though. What is the effect of real-life conditions on the observed trends?

Changes in external conditions, such as environmental conditions, may have an impact on control actions. For example, in warmer days water temperature will be higher and, thus, the frequency of activation of the heater will be lower. This section aims to evaluate, through a real-life case study to what extent trends of the same 'Failure Mode' (F_r) under different environmental conditions (such as different days and under varying weather conditions) produce comparable results.

5.5.1 Evolution of frequency and duration of SOMs

For this purpose, the real-life case study will be based on the previously presented greenhouse testbed. For this analysis, we injected an incremental 'Tank leak' (F_1), through 27 progression levels that started with a leak rate of 0.000085 L/s and ended with a leak rate of 0.038 L/s at its critical threshold. Figure 5.13 depicts the progress of failure. We conducted 15 different experiments in which the same failure progression process was injected. These experiments were conducted in different days, which presented variations in the ambient temperature, light intensity, air humidity and water temperature.

Data of the different experiments was analyzed by collecting the frequency of occurrence and duration time of every SOM, for every failure size. We plotted the evolution of every SOMs' frequency (Fq) and duration (D) indicator as failure progresses for the failure of 'Tank leak' (F_1) and a Failure-free (F_f) case. Data collected

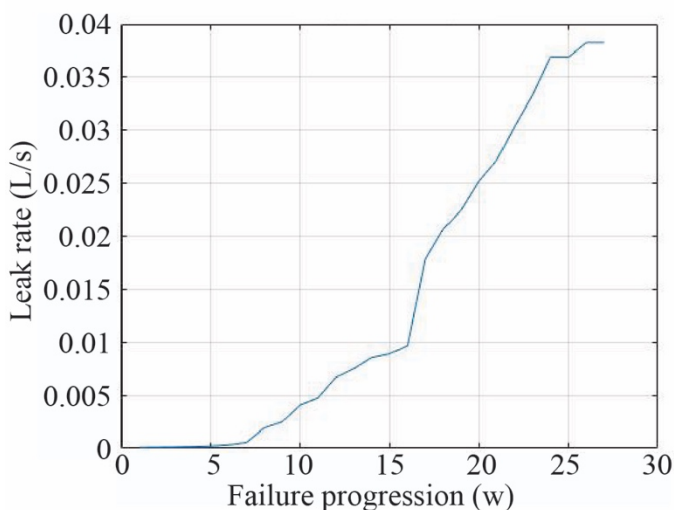


Figure 5.13. Evolution of the greenhouse ‘Tank leak’ failure mode (F_1)

from different experiments were overlapped into the F_1 category plot and data of Failure-free (F_f) operations of different experiments was combined into the Failure-free (F_f) category plots in Figure B.1 and Figure B.3 in Appendix B. Savitzky-Golay filter was used to remove disturbances that can hamper the visualization of the long-term trends. The images presented in Appendix B considered a single step per failure size.

The variation of Fq and D was analyzed for the ‘Tank leak’ (F_1) case and the Failure-free (F_f) one using Kruskal-Wallis statistical method adjusted with a Bonferroni correction. This statistical test provided a numerical value (p) that enables evaluating how significant was the difference between the failed and the Failure-free (F_f) cases. A pairwise comparison of $\Delta Fq_{\zeta_d}^{F_f}$ with $\Delta Fq_{\zeta_d}^{F_r}$ and $\Delta D_{\zeta_d}^{F_f}$ with $\Delta D_{\zeta_d}^{F_r}$, was conducted, where the SOM is ζ_d , F_f denotes the Failure-free case and F_r denotes the Failure Mode (which for this case is ‘Tank leak’ (F_1)). These results are presented in Table 5.5 and in a set of Boxplot plots shown in Figure 5.14 and Figure 5.15. The Boxplot plots allow graphically determining how similar are the trends of Fq and D , observed in different experiments subjected to tank-leak, and how they differ from the trends observed during failure-free operation.

The analysis of the Figure B.1 to Figure B.4 in Appendix B revealed that Failure-free (F_f) cases do not present any particular trend as effect of the regular system behavior for none of the 15 experiments conducted. However, clear trends were observed in the case

Table 5.5. Results of the statistical test for SOM frequency and SOM duration

	P-value							
	SOM 9	SOM 11	SOM 13	SOM 15	SOM 41	SOM 43	SOM 45	SOM 47
Frequency	0.065	0.017	3.07E-06	3.05E-06	6.26E-05	0.15	3.07E-06	6.12E-07
Duration	3.07E-06	0.98	0.95	3.73E-06	5.25E-05	0.78	0.00097	6.12E-07

of ‘Tank leak’ (F_1), more specifically in the cases of operation modes of ζ_{11} , ζ_{13} , ζ_{15} , ζ_{41} , ζ_{45} and ζ_{47} for Fq indicator and ζ_9 and ζ_{13} for D (see Figure 5.16). As it was shown in Table 4.16 in Subsection 4.5.2, these SOMs are defined by the following combinations of component operation modes:

- ζ_9 : only the ‘Fan-in’ (S_{A4}) is on.
- ζ_{11} : both, the ‘Fan-in’ (S_{A4}) and the ‘Electro valve water reservoir’ (S_{A2}) are opened.
- ζ_{13} : the ‘Heater’ (S_{A3}), along with the ‘Fan-in’ (S_{A4}) are active.
- ζ_{15} : the ‘Electro valve water reservoir’ (S_{A2}), the ‘Fan-in’ (S_{A4}) and the ‘Heater’ (S_{A3}) are on.
- ζ_{41} : the ‘Fan-in’ (S_{A4}) and the ‘Electro valve Plant Bed 2’ (S_{A6}) are active.
- ζ_{45} : the ‘Electro valve Plant Bed 2’ (S_{A6}), the ‘Fan-in’ (S_{A4}) and the ‘Heater’ (S_{A3}) are on.

The presence of upward or downward trends in operation under failure and the lack of trends in the case of the failure-free case were consistent with the results obtained in the pilot study presented in Section 5.3. Moreover, the observed trends in the ‘Tank leak’ (F_1) experiments present trajectories that are consistent with each other. It implies that the external factors caused by real-life environmental conditions do not have a significant effect on the observed trends.

The boxplots shown in Figure 5.14 and Figure 5.15 also supports the afore-mentioned claims. Significant differences between the SOMs’ frequency (Fq) and duration (D) indicators corresponding to the failed cases and the failure-free ones are actually observed. This result is also coherent with the observations in the pilot study in which the failed cases could be distinguished from the failure-free ones through Fq and D . The higher complexity of the greenhouse testbed did not significantly affect failure detection accuracy of SOM frequency and duration. Nevertheless, there is a higher variance in data of ‘Tank leak’ (F_1) than it was in the case of the simulated model of the kettle (see Table 5.6 and Table 5.7).

The analysis of variation of Fq and D in every single SOM provides valuable insights into the role of SOMs on failure progression. It was already seen (See Subsection 5.5.1) that a univariate analysis of these failure indicators is able to distinguish the failed cases from the failure-free ones. It was also observed that ‘Tank leak’ (F_1) failure mode presents clear trends on most of its SOMs reflecting system degradation. The univariate analysis showed that although there is a higher variance of the data, the environment does not hamper failure detection. However, we have not explored the effect of variance

Table 5.6. Variance per SOM in the greenhouse’s case

	Tank leak's variance - Greenhouse case							
	SOM 9	SOM 11	SOM 13	SOM 15	SOM 41	SOM 43	SOM 45	SOM 47
Frequency	0.021	0.002	0.013	8.90E-05	0.0087	9.76E-05	0.0004	0
Duration	0.086	9.87E-05	0.055	0.0008	2.80E-06	0.0004	0.0016	0

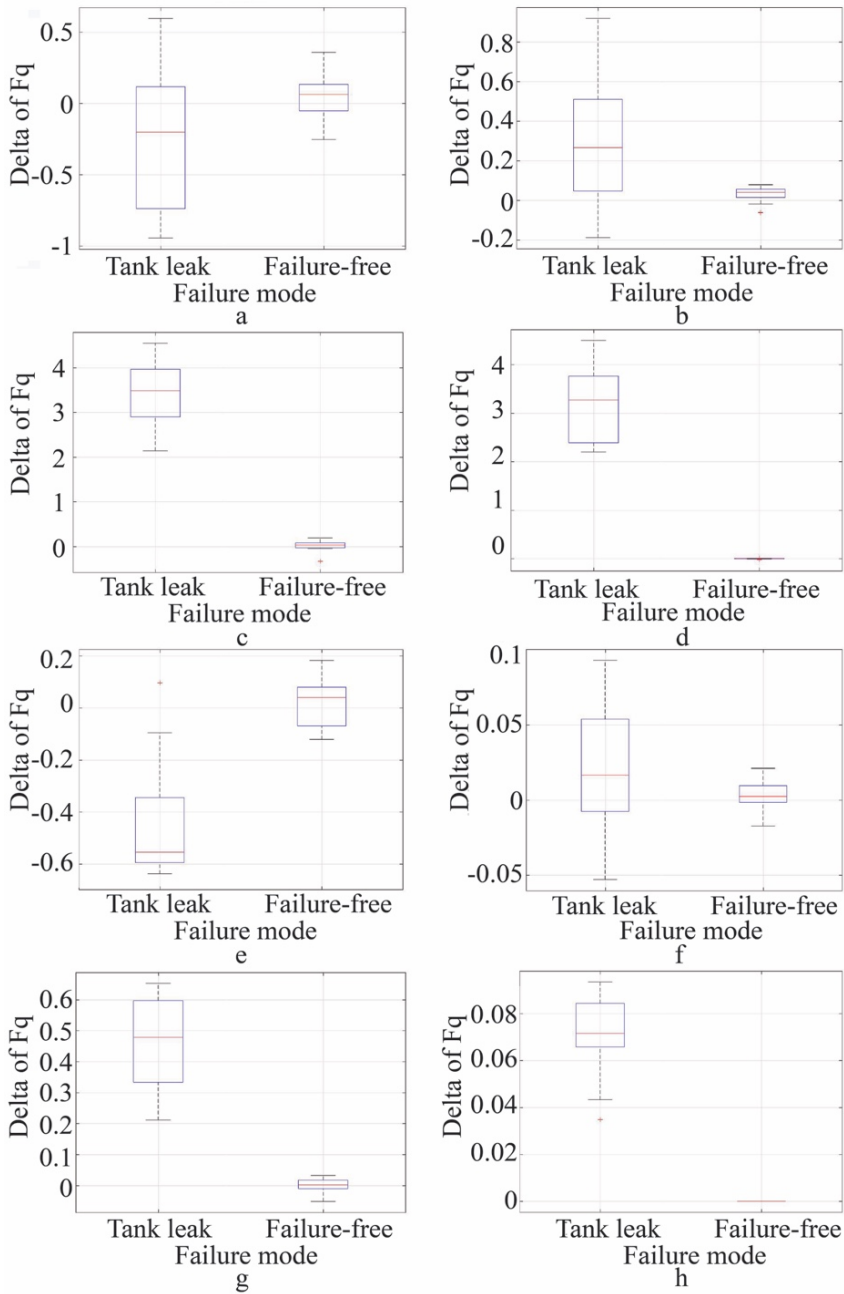


Figure 5.14. Comparison of the variation presented by SOM frequency (ΔFq) for the tank leak and the failure-free case. (a) ζ_9 , (b) ζ_{11} , (c) ζ_{13} , (d) ζ_{15} , (e) ζ_{41} , (f) ζ_{43} , (g) ζ_{45} , (h) ζ_{47} .

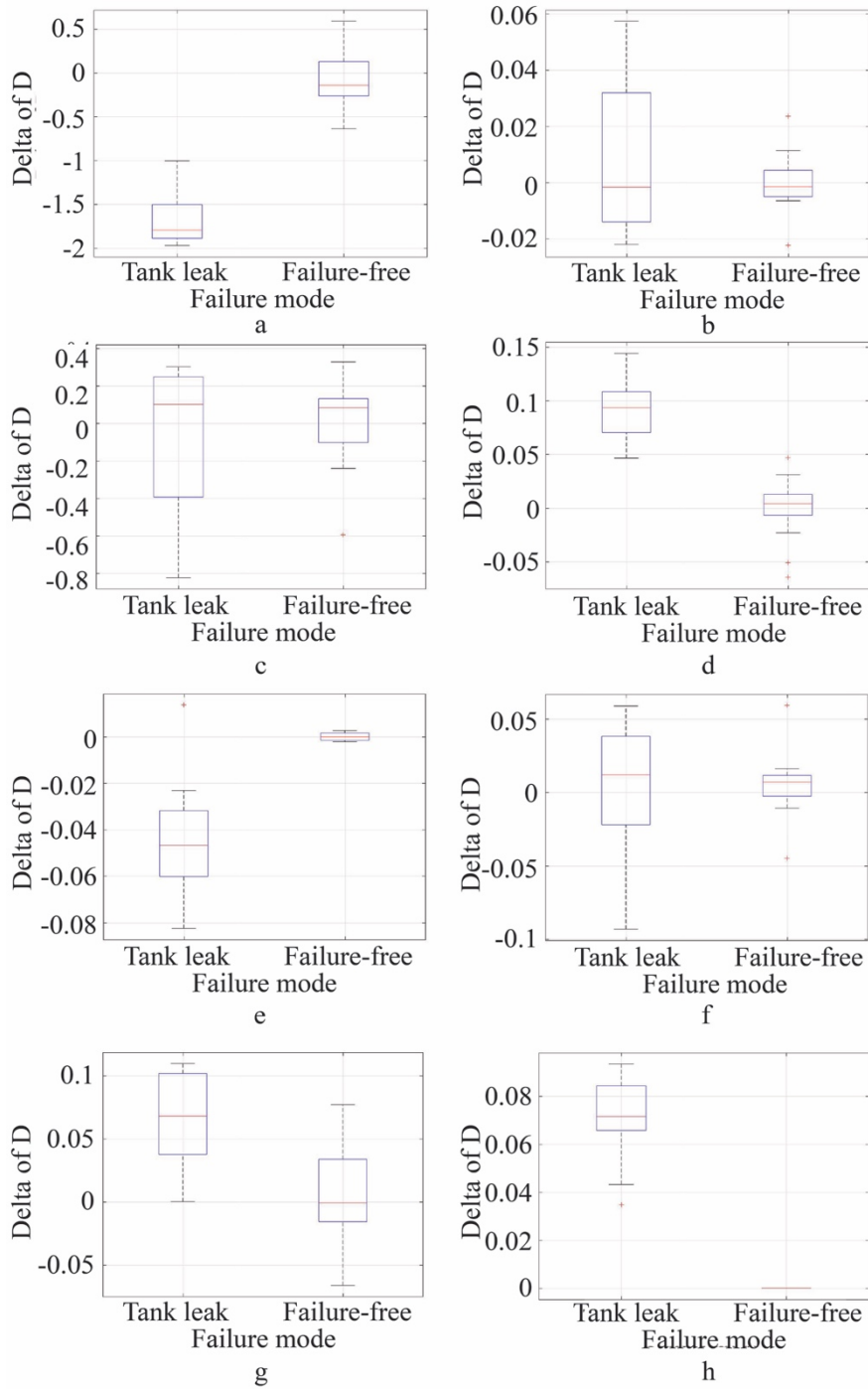


Figure 5.15. Comparison of the variation presented by SOM duration (ΔD) for the tank leak and the failure-free case

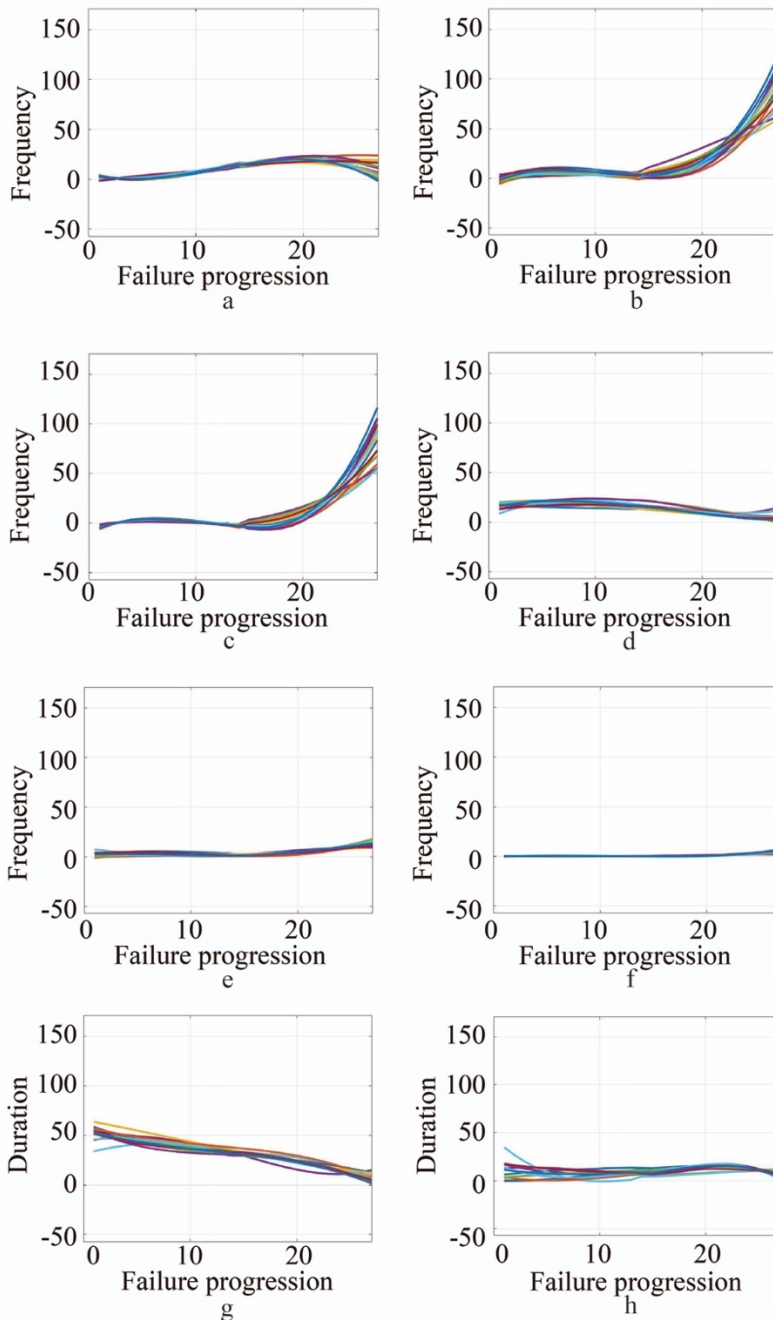


Figure 5.16. Filtered trends corresponding to tank leak. (a) Frequency of ζ_{11} , (b) Frequency of ζ_{13} , (c) Frequency of ζ_{15} , (d) Frequency of ζ_{41} , (e) Frequency of ζ_{45} , (f) Frequency of ζ_{47} , (g) Duration of ζ_9 , and (h) Duration of ζ_{13}

Table 5.7. Variance per SOM in the kettle's case

		F1's variance - Kettle model case									
		SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
Frequency		0.0002	0.0002	6.43E-05	6.83E-08	0.0001	6.46E-09	2.33E-07	8.00E-08	5.23E-05	0.0001
Duration		2.03E-10	8.46E-11	2.83E-09	9.96E-07	1.62E-12	3.05E-07	1.64E-10	1.35E-08	1.56E-12	1.94E-13

		F2's variance - Kettle model case									
		SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
Frequency		0.0002	0.0002	5.45E-05	4.95E-08	0.0001	8.86E-09	9.64E-08	2.91E-08	5.13E-05	0.0001
Duration		1.03E-09	3.75E-10	3.02E-09	1.66E-06	1.80E-12	2.93E-07	1.13E-09	6.05E-10	6.83E-13	5.45E-16

		F3's variance - Kettle model case									
		SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
Frequency		0.001	0.001	9.59E-05	4.99E-08	9.20E-05	1.88E-09	1.30E-07	3.05E-08	8.84E-05	7.90E-05
Duration		1.06E-09	3.60E-09	3.59E-09	1.34E-06	2.26E-12	2.07E-07	8.13E-10	9.30E-10	3.36E-13	3.76E-13

		F4's variance - Kettle model case									
		SOM 1	SOM 2	SOM 3	SOM 4	SOM 7	SOM 8	SOM 9	SOM 10	SOM 11	SOM 15
Frequency		9.19E-05	9.03E-05	1.18E-05	1.99E-08	1.47E-05	2.39E-09	1.00E-07	4.27E-08	8.83E-06	4.13E-05
Duration		5.30E-10	3.35E-10	1.19E-09	4.27E-07	4.78E-11	8.98E-08	3.11E-10	1.91E-07	3.24E-10	1.81E-13

on the joint analysis of Fq and D for all SOMs yet. This multivariate analysis is relevant, as it provides the means for conducting failure discrimination, so that, we can conduct failure diagnosis too. The next section aims to explore the joint effect of all SOMs' frequency (Fq) and duration (D) indicators on the real-life case.

5.5.2 Combined analysis of SOMs' frequency and duration

This analysis starts by building the classification model with Linear Discriminant Analysis (LDA) based on the collected data at the critical threshold of the 'Tank leak' (F_1) (i.e. when the leak rate reaches 0.038 L/s). A dataset consisting of 70 samples was used for training and testing purposes. It included:

- 20 data samples corresponding to SOM frequency (Fq) and SOM duration (D) gathered during the failure-free (F_f) operation of the testbed;
- 21 data samples of Fq and D measured when the system presented to a 'Tank leak' (F_1);
- 20 data samples corresponding to 'Irrigation pipe blocked' (F_2) representing a complete blockage of one of the irrigation pipes at plant bed 2;
- 9 data samples corresponding to 'Irregular fan operation' (F_3).

50 random datasets were generated by partitioning the total dataset. From each dataset 70% of data were used for training the failure classifier. The remaining 30% were used for testing. The selection of the best model was performed by maximizing the accuracy

Table 5.8. Confusion matrix of the greenhouse's classification model

Failure mode	F_1	F_2	F_3	Failure-free
F_1	6	0	0	0
F_2	0	6	0	0
F_3	0	0	2	0
Failure-free	0	0	0	6

of the classification (successful classification rate) for each class, so that, none of the failure modes would present a low classification accuracy. Samples used for the analysis presented in Section 4.5 were included for deriving the classification model. The results obtained through the test set are summarized in Table 5.8, which is the confusion matrix of the results for 'Tank leak' (F_1), 'Irrigation pipe blocked' (F_2) and 'Irregular fan operation' (F_3).

The confusion matrix of the selected model does not present any false-negative nor false-positive result for any of the analyzed failure modes. All data samples considered during the test were successfully classified despite the low amount of data available for deriving the classification model. Once the model was derived we computed Fq and D from the datasets used in Section 4.5, (15 data samples per failure progression level w were considered). New experiments were needed with regards to collect data corresponding to failure-free operation (F_f), and to the 27th failure progression level w

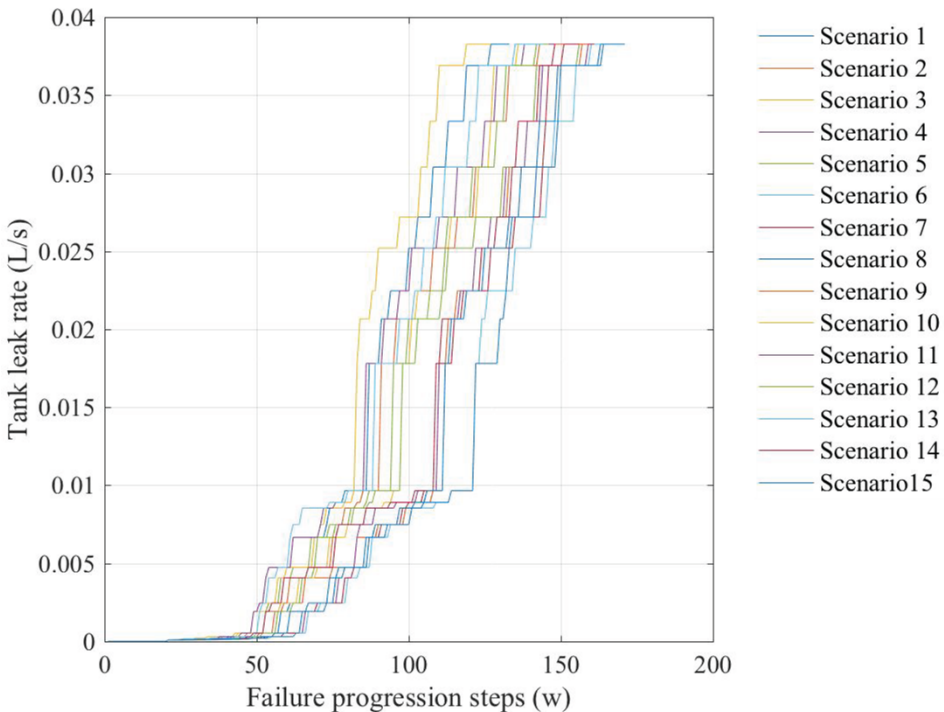


Figure 5.17. Failure progression processes

from tank-leak. Note that unlike the pilot-study in which the failure progression of all considered ‘Failure Modes’ (F_r) were analyzed, in this section we will only focus on the prediction and prognosis of the ‘Tank leak’ (F_1).

Our failure analysis considered 15 different scenarios of failure evolution. For this purpose, data belonging to the dataset of every w step was randomly selected and sequentially arranged in 15 different cases (see Figure 5.17). Unlike the univariate analysis where the failure size at w was different than the failure size at $w - 1$ and $w + 1$, the current analysis can present the same failure size during several consecutive w steps. The first 20 failure progression steps correspond to Failure-free (F_f) system operation that were taken from the dataset used during the univariate analysis.

The first analysis aimed to determine to what extent the observed SOMs’ frequency (Fq) and duration (D) indicators managed to predict the occurring ‘Failure Mode’ (F_r). For this purpose, the Fq and D of every failure size were used as input for the aforementioned classification model. The classification results are presented in Figure 5.18. It can be observed that ‘Tank leak’ (F_1) is recognized and properly diagnosed from step $w = 67$, when the first scenarios reach a leak rate of 0.0089 L/s (Failure size=2.34%). However, it is only at the step $w = 92$ when all the analyzed scenarios reach this leak rate. From that progression step, failure diagnosis is consistent and remains stable until reaching the critical threshold of the failure.

It can be observed, that a number of cases are misclassified as ‘Irregular fan operation’ (F_3) between $w = 24$ (Failure size=0.23%) and $w = 76$ (Failure size=15.75%), while real operation was tank leak. There are even some minor cases at $w = 1$, and $w = 2$ where 7% of the analyzed samples were misclassified as F_3 , while it was actually failure-free (F_f). Between $w = 48$ (Failure size=0.84%) and $w = 59$ (Failure size=6%) the number of misclassified scenarios as F_3 reaches its maximum at 67% of the analyzed cases. This phenomenon is similar to the one described in Subsection 5.3.3 where it was classified as F_3 at first, but as the failure progressed the correct diagnosis of F_1 appeared. This is interesting to analyze as it kept occurring during around 43 consecutive failure progression steps. There can be two main hypothetical answers to the observed misclassification: (i) the low number of data samples corresponding to F_3 considered during the training of the classification model, or (ii) the lack of a characteristic manifestation of F_3 through Fq and D indicators.

Another answer may be that the inlet fan (S_{A_4} = ‘Fan-in’), seeks to control the ambient temperature and CO₂ concentration levels into the greenhouse. However, there are no further control actions depending on the sensed fan speed ($S_{S_{13}}$ = ‘RPM Sensor’), which is related to the component where this failure mode (F_3) is manifested. The ‘Irregular fan operation’ (F_3) is observable through variations of the intensity of the actuator (S_{A_4}). However, it cannot be observed in the SOMs’ frequency (Fq) and duration (D) indicators due to the lack of control actions that depend on this parameter. The lack of transition between system operation modes limits the distinctive power of the classification model, which is not able to distinguish between the Failure-free (F_f) case and F_3 .

The misclassification problems cannot be linked to external factors, as they have a consistent behavior across the analyzed scenarios. However, the current concept of Fq and D of SOM has limitations when control actions are increasing or decreasing the

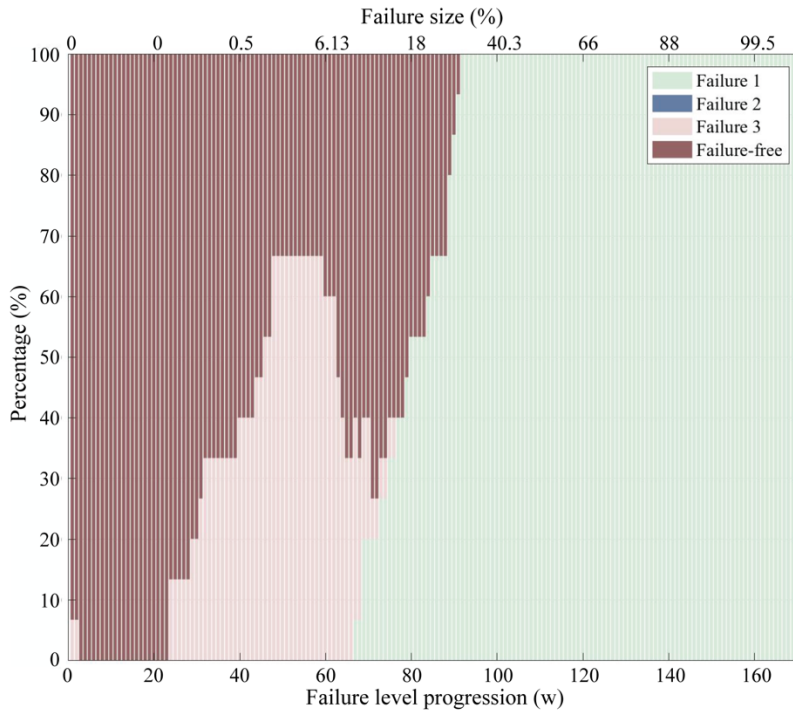


Figure 5.18. Evolution of failure prediction for failure-leak (F_1)

intensity of the actuators operation (i.e. when the actuator is analogous). Intensity of actuator's operation is a factor that should be considered in the definition of System Operation Modes. It describes a component operation mode that keeps the stability property by modifying the power of operation of the actuator, instead of just activating or deactivating it. Although, this new dimension is a very interesting aspect, it was out of the scope of the present explorative research.

Failure forecasting through Fq and D was also investigated. For this purpose, the classification model previously presented in this section was taken up. However, it was fed with extrapolated data. In order to derive the forecasting model that delivers the extrapolated data, the 15 scenarios presented in Figure 5.17 were included in this analysis. Moving windows with $s=20$ steps and a forecasting horizon of $b = 20$ were also considered. Results obtained are presented in Figure 5.19. This figure presents the evolution of the Time-To-Failure (TTF) estimation (Figure 5.19(a)) and the forecasted failure mode (Figure 5.19(b)).

The graph of Figure 5.19(b) shows a similar pattern to the one observed at Figure 5.18. However, the use of forecasted Fq and D indicators as input for the classification model enables knowing the occurring failure mode earlier (around progressions before). Two sets of curves can be identified in the TTF plot at Figure 5.19(a). The first set is between steps $w = 27$ (Failure size = 0.27%) and $w = 48$ (Failure size = 0.84%), and the second one is between steps $w = 60$ (Failure size = 6.13%) and $w = 98$ (Failure size=39%).

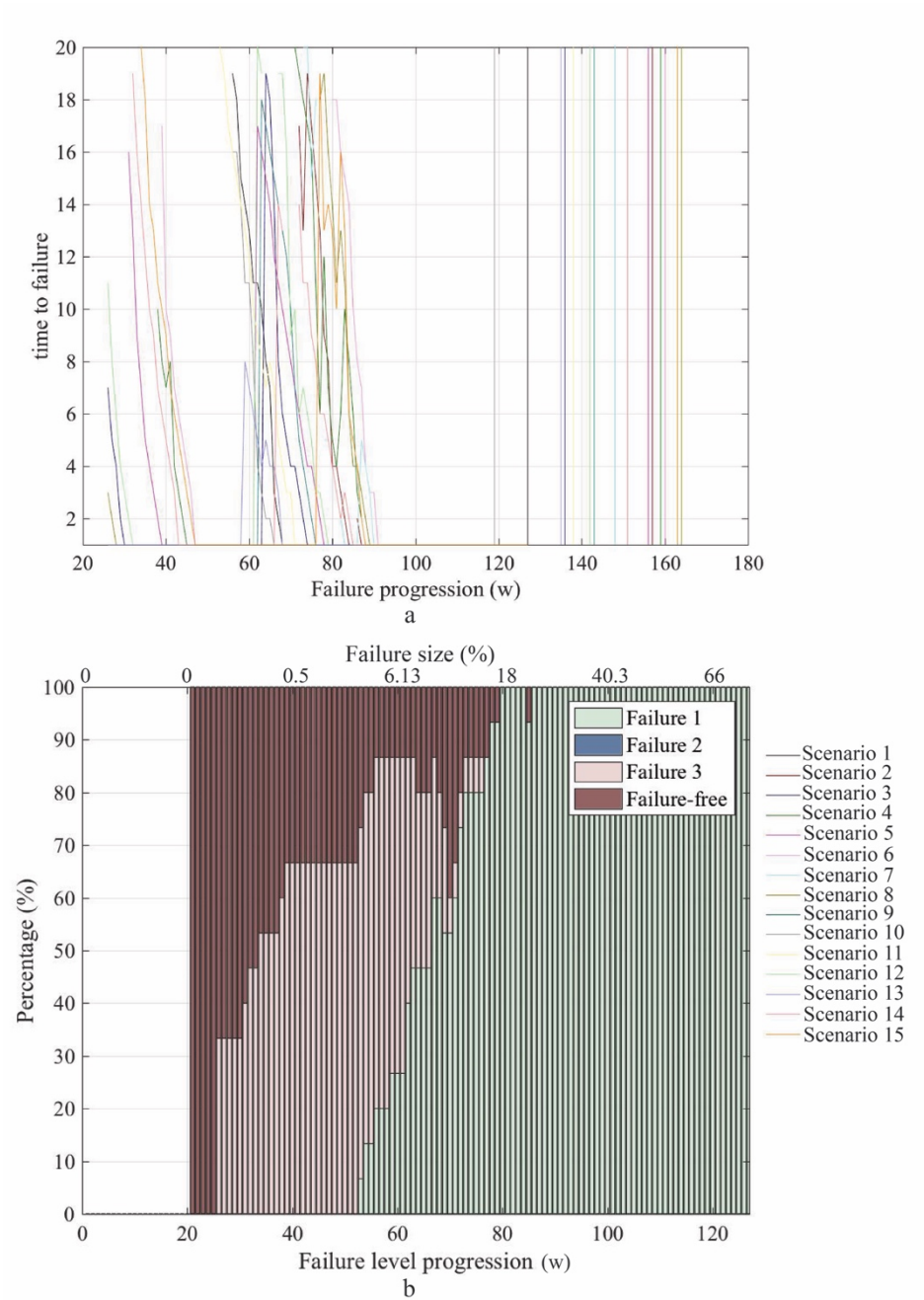


Figure 5.19. Failure forecasting for tank leak; (a) Time to failure of; (b) forecasting of the failure mode

Although these scenarios were misclassified as F_3 in the first set, their prediction gradually moved to F_1 in the second set, as the failure progressed (as shown in Figure 5.19(b)). The reason is, the estimated TTF changes gradually, as the actual step w approaches the critical threshold (represented by the vertical lines that can be observed in Figure 5.19(a)).

In addition to the above-mentioned problems, the predicted TTF is shorter than the actual time to failure. Our forecasting analysis delivered an average TTF = 84 for Time To Failures estimated for the steps between $w = 73$ and $w = 98$. However, the actual average failure threshold is at step $w = 146$ as failures from the analyzed scenarios occurred between $w = 127$ and $w = 171$. This situation was also observed in the pilot-study and, thus, the conclusions derived in Subsection 5.4.2 can be extended to the current experiment. The results obtained during the real-life case study (greenhouse testbed) did not present major differences with respect to the ones observed in the pilot-study. The environmental disturbances during the real-life system operation do not affect Fq and D failure indicators. Likewise, the increased system complexity, which is reflected in the increased number of actuators and control actions, do not affect neither the consistency nor the discriminant power of the Fq and D trends.

5.6 Discussion

This exploratory analysis provided insights into the role of SOMs in the failure forming process. It was found that system degradation and failure evolution is manifested through variations of the frequency and duration of SOMs, as the effect of the compensatory actions carried out by self-tuning systems. SOM frequency and duration allows monitoring failure progress based on data derived from control signals. This last characteristic is relevant, as it enables considering variations of system behavior and environmental factors that can affect the lifetime of the system. Moreover, this performance analysis is conducted at system level, facilitating exploration of failure symptoms both on the levels of single components and interaction of components.

Frequency and duration of SOMs are failure indicators. They allow studying the failure forming process. In our experiments, the trends of SOM frequency and duration presented strong similarity for same failure modes unique trends for different failure modes. A combined analysis of the trends of SOM frequency and duration showed that they have high discriminant power making them potential means for failure diagnosis purposes.

Our failure forecasting approach showed it is possible to conduct failure prognosis by extrapolating the trends of the SOM frequency and duration. It allowed estimating the time to failure (TTF) of the system as well as determining the forming failure mode. Nevertheless, the TTF was somewhat inaccurate. Further studies are required around this subject, in order to provide a more accurate estimation of TTF using compensatory methods. Although it is always required knowing in advance the time remaining before failure occurs, inaccurate estimations may affect negatively the cost-effective implementation of maintenance strategies. Nevertheless, it does not detract from the role of the frequency and duration of SOMs as indicators, they manage to reflect failure evolution satisfactorily.

Results concerning the discriminant power of the indicators, as well as their potentials for forecasting were consistent in both the computer model of the kettle and in the case of greenhouse testbed. They suggest that SOM frequencies and durations are not sensitive to disturbances caused by external factors such as the environment conditions. Nevertheless, it can prevent the detection of failures that are manifested through transient faults. Transient faults manifest intermittently, i.e. they are not manifested through long-term trends, and therefore they can be mistaken as noise effect. This particular aspect requires further study as our analysis is based on the historic manifestations of failures.

In this chapter, we have address failure evolution from its earliest stage. From the results we concluded that the role of SOMs in failure analysis is to reflect the failure forming process through the changing frequency and duration of SOMs. This is the main contribution of this promotion research. However, some aspects require further study:

- (i) Firstly, the implementation of the frequency and duration of SOMs for failure diagnosis and forecasting should be developed. In the present work, we proposed a basic diagnosis and forecasting concept, which aimed to demonstrate the potential application of SOMs as failure indicator.
- (ii) Secondly, the analysis of variations of actuators intensity is also required. So far, we have analyzed variations on frequency and duration of SOMs as means of self-tuning. However, variations on actuators' operation intensity can also provide relevant information about the failure forming process. Moreover, it can affect the SOM frequency and duration, since manipulating actuators operation intensity can compensate the effect of failure. Component operation modes should be defined based on actuators' operation intensity.

Thirdly, more failure forming patterns should be evaluated to determine their effect over the frequency and duration of SOMs. Due to their heterogeneous nature, failures can present multiple variations on their failure progress that can affect the analyzed long-term trends. For instance, failures with non-linear progression trends may influence the changes of frequency and duration of SOMs. We argue they cause sudden upward or downward trends that could affect the estimations of time to failure, as well as the prediction of the forming failure mode. The above-mentioned factors can be addressed in further investigation. Nevertheless, the present work is a step towards the understanding of the effect of self-tuning systems on failure analysis.

5.7 Conclusions

In this chapter, we explored the role of SOMs on failure manifestations. We studied the consistency and discriminant power of the trends of SOM frequency and duration during several failure modes. For this purpose, first a computational model of a water kettle was used in a pilot-study. It enabled simulating multiple failures and evaluating our main assumptions in a simple system, as well as it enabled avoiding the effect of external disturbances. Likewise, a real-life study was performed with a greenhouse testbed, used to determine to what extent the results observed during the pilot-study could be reproduced under real-life circumstances and in a more complex system.

We found that SOM's frequency and duration in the behavior-based failure analysis can reflect the system degradation process. It allows anticipating the occurrence of the critical failure threshold. SOM's frequency of occurrence (Fq) and their duration time (D) capture the failure progress through soft and stable long-term trends, when they are analyzed by univariate statistical methods. These trends, approach characteristic Fq and D values of the studied failure at its critical threshold, which enables extrapolation of the trends. Combined analysis of Fq and D for all SOMs facilitates characterization of failure modes and estimation of Time To Failure (TTF).

Our study supports the potential role of SOMs' frequency (Fq) and duration (D) in failure forecasting, that is, these failure indicators reflect the failure forming process and system degradation enabling their traceability. However, our study only considered failures that are manifested in 'on/off' components. The variation of the operation intensity of system actuators should be part of the definition of component operation modes, and as a result they can define new types of SOMs. Further studies on the influence of variations of actuator intensity is required.

Methods of SOMs' frequency (Fq) and duration (D), in failure prediction and forecasting, should be developed and further validated with industry cases (e.g. green houses, building operations, automotive applications). The current study provided insights about practical application of SOMs' frequency (Fq) and duration (D) to real-life testbed systems, however, technical aspects such as infrastructure requirements and manifestation of failures in real life has not been addressed in our research.

5.8 References

- [1] P. Albertos and I. Mareels, "Stability, Sensitivity and Robustness," in *Feedback and Control for Everyone*, Springer Berlin Heidelberg, 2010, pp. 151–178.
- [2] M. S. Mahmoud and P. Y. Xia, "Introduction," in *Applied control system design*, 2012, pp. 1–10.
- [3] K. Michels, F. Klawonn, R. Kruse, and Nürnberger, "Fundamentals of Control Theory - Springer," in *Fuzzy control Fundamentals, Stability and Design of Fuzzy Controllers*, 2006.
- [4] M. S. Mahmoud and Y. Xia, "Introduction," in *Applied Control Systems Design*, M. S. Mahmoud and Y. Xia, Eds. London: Springer London, 2012, pp. 1–10.
- [5] B. A. Francis and W. M. Wonham, "The internal model principle of control theory," *Automatica*, vol. 12, no. 5, pp. 457–465, Sep. 1976.
- [6] Y. Zhang and J. Jiang, "Bibliographical review on reconfigurable fault-tolerant control systems," *Annu. Rev. Control*, vol. 32, no. 2, pp. 229–252, Dec. 2008.
- [7] A. Preumont, "Controllability and Observability," in *Vibration Control of Active Structures: An Introduction Third Edition*, A. Preumont, Ed. Dordrecht: Springer Netherlands, 2011, pp. 275–297.

- [8] P. J. Antsaklis and A. N. Michel, “Controllability and Observability: Fundamental Results,” in *A Linear Systems Primer*, P. J. Antsaklis and A. N. Michel, Eds. Boston, MA: Birkhäuser Boston, 2007, pp. 195–236.
- [9] F. Hu *et al.*, “Robust Cyber–Physical Systems: Concept, models, and implementation,” *Future Gener. Comput. Syst.*, vol. 56, pp. 449–475, Mar. 2016.

Chapter 6

Investigation of the role of SOMs in broader context of maintenance

6.1 Introduction

6.1.1 Research objectives

It has been discussed in Chapter 1 that we are moving from zero generation CPSs (0G-CPS) to first generation CPSs (1G-CPS). Systems belonging to the latter generation are able to maintain or even optimize their behavior based on their self-regulatory and self-tuning capabilities. The self-tuning actions performed by a 1G-CPS reduce the chance of early observation of emerging failures and preventing their proliferation. This issue has been hardly addressed in the recent literature. This promotion research brought in and elaborated on the concept of system operation modes (SOM) as a mechanism of operationalizing self-tuning in CPSs. It also placed it in the context of signal feature-based failure recognition and forecasting. Chapters 4 and 5 revealed that two specific characteristics of SOMs, namely the frequency of changing SOMs and the durations of SOMs, are important factors from the perspective of failure analytics. It has been found that these operational characteristics can be used as failure indicators in failure diagnosis and forecasting. Based on the empirical investigations it can be argued that their active regulation can reduce the number and criticality of failures in 1G-CPSs. However, failures cannot completely be avoided. The influence of the external environment, user manipulation, external attacks, and components wearing may not only affect system performance, but can also lead to sudden, transient or progressive failures.

In the two previous Chapters, the change of the frequency of SOMs and the duration SOMs were used as behavior indicators, or in other words, as failure indicators, on an operative level. This operative level study mainly concentrated on the role and influence of these behavior indicators on failure diagnosis and failure forecasting. However, in order to assure a reliable system operation, systems should be subjects of efficient maintenance strategies that properly integrate multiple maintenance principles in a cost-effective way. Having this in mind, the role and influence of the above behavior indicators will be discussed from a strategic level in the rest of this Chapter. The overall objective is to analyze how the new knowledge obtained concerning the role of SOMs in failure diagnosis and forecasting can be reused or adapted in the context of developing principles for preventive maintenance of 1G-CPSs. Towards this end, we first analyzed the documented maintenance principles of 0G-CPSs. This particular analysis was based on a set of seminal publications. Using critical reasoning, we considered the possible

influences of a SOMs-based approach on these principles of preventive maintenance. As an outcome, we obtained additional knowledge that can facilitate the elaboration of preventive maintenance principles tailored to 1G-CPSs. The content reported in this chapter was in large part published in [1], as it was one of the main outputs of this research project. Below - prior to the discussion of the maintenance principles - we will first briefly explain some concepts that play an important role in the context of this chapter.

6.1.2 Introduction of the relevant terms

Maintenance is a set of important multi-faceted activities that are carried out to keep the operation of a system in an optimal state. It involves activities such as inspection, adjustment, replacement, repair, overhaul and renewal. Maintenance (i) increases the useful life and reliability of systems, (ii) reduces size, scale and number of repairs, as well as (iii) decreases the need for emergency repairs. In general terms, it allows lowering the overall costs of operation while increasing safety and security. Contrary to its importance, the terminology used in the literature related to system maintenance does not seem to be uniform. Terms such as maintenance ‘strategy’, ‘principles’ and ‘policy’ are used with various interpretations and meanings, often even interchangeably or confusingly in the literature. At the same time, there are other basic terms whose definition is often taken for granted. These terms are: ‘principle’, ‘method’, ‘rule’ and

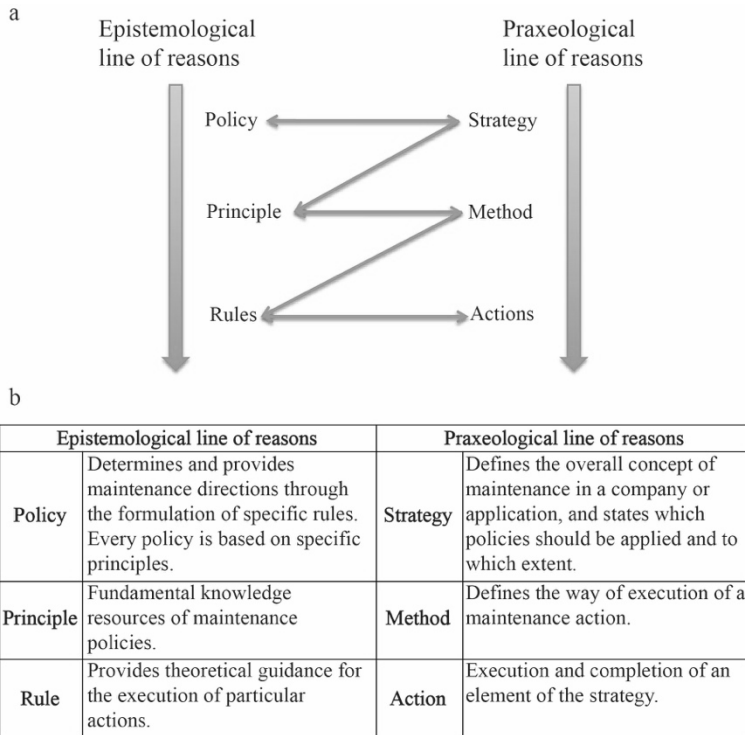


Figure 6.1. Clarification of the main terms: (a) relationships of the terms, and (b) interpretation of the terms

‘action’. For this reason, we elaborate on the used terminology.

In general, a ‘policy’ is defined as a collection of rules that, depending on the most essential state variables, “specifies what to do exactly in a particular situation” [2]. From a managerial point of view, ‘strategy’ is described as (i) the definition of long-term goals, objectives and courses of action for a company and (ii) the allocation of resources for the achievement of such objectives [3]. The definitions in the Oxford Dictionary are used as reference in this section [4]. Therefore, the term ‘principle’ is interpreted as “a fundamental source or basis of something”. A ‘method’ is a “particular procedure for accomplishing or approaching something” in a systematic way. A ‘rule’ is interpreted as a set of explicitly understood regulations. Finally, an ‘action’ is considered as a logically separable procedural element of doing something. The application of these terms in the maintenance context is shown in Figure 6.1. Figure 6.1(a) graphically shows the relationships between the above defined major terms and separates them according to whether they are of epistemological (knowing) or praxiological (executional) flavor. Figure 6.1(b) summarizes the above interpretation of the two groups.

Usually, the maintenance of 0G-CPSs, likewise traditional engineered systems, is conducted according to a specific maintenance strategy. It determines a set of objectives related to system reliability, so that a trade-off is achieved between the resources invested in system maintenance and the continuous system operation. The formulation of a maintenance strategy requires the selection and implementation of certain maintenance principles. These principles are operationalized through a set of methods that determine not only the actions to be conducted with regards to assure system operation, but also the order and time of its execution, as well as the responsible for the execution. Maintenance strategies are to be customized in order to meet the requirements of every particular system. They cannot be generalized. However, they are associated with maintenance principles, which are general (largely system independent) postulates that can be analyzed without considering the specific features of a particular system. Considering this fact, we will focus on the analysis of maintenance principles in this chapter, in particular on the implications that SOMs convey in the context of preventive maintenance principles for 1G-CPSs.

6.2 Analysis of existing maintenance principles

6.2.1 Overview of the maintenance principles used in the context of 0G-CPS

Maintenance seeks to ensure a permanent availability of a system through the application of its basic principles. As a consequence, it is assumed that these principles should be applicable to any system - including first generation CPSs. However, this claim is not obvious when 1G-CPSs are concerned. The reason is the increased complexity, heterogeneity and non-linearity that these systems are characterized by. The above characteristics influence not only the self-tuning capabilities of the systems, but also the strategies and principles that should be applied in their preventive maintenance.

It means that these systems should be considered differently from the perspective of preventive maintenance.

Zero generation CPSs are representatives of hybrid systems that include hardware, software and cyberware constituents. Their operation and performance are strongly influenced by the interactions among the various components. This fact has significance in the context of 0G-CPS, since the maintenance approaches of traditional engineered systems are based on a reductionist approach (which manages the hardware and software constituents separately), in which maintenance principles applied to software sub-systems (so-called ‘cyber’ components) do not consider the hardware sub-systems (‘physical’ components), and vice versa (Figure 6.2). This causes that failure symptoms originating in the interaction of the two types of components are not considered at all. Moreover, maintenance methods in the physical domain are mostly oriented to components, and they do not consider the higher level of system analysis.

In 0G-CPSs, the maintenance principles addressed to physical components intend to maintain system availability in a cost-effective way. These are based on the assumption that every component of the system has a limited life cycle and, thus, may be subject to wear or breakdown. Such maintenance principles are operationalized from two main approaches: Preventive Maintenance (PM) and Corrective Maintenance (CM). From one hand, PM principles aim to avoid failures before they occur, while, on the other hand CM principles allow system operation up to failure occurrence, in case of failure consequences are not critical or do not have an effect during a certain operation period [5]. PM may be conducted according to the principles of

- **Fault prevention (P_1):** seeks to avoid the occurrence of faults through preventive actions [6], such as spare change, revisions and repairs, previously to failure occurrence.
- **Fault removal (P_2):** seeks to reduce the number of faults and their severity [7], by considering the most critical aspects of the system, so that, failures can be avoided since the design of the system features.
- **Fault forecasting (P_3):** allows predicting failures and their impact, based on the fault records [8]. It entails estimating the incidence and consequences of faults, based on the present number of faults.

There are two main approaches of operationalization of P_1 and P_3 . These are Time-Based Maintenance (TBM) and Condition-Based Maintenance (CBM). TBM entails scheduling maintenance actions for preventing failures [9]. On the one hand, knowledge management techniques should be applied in the case of P_1 in order to determine a schedule for conducting revisions, spare changes, and

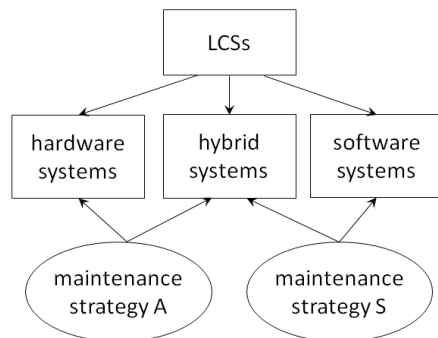


Figure 6.2 Separation of concerns with regards to maintenance strategies

repairs. On the other hand, the implementation of P_3 raises the need to determining the average life of components, based on experimentation or previous experience. Average components' life-cycle is usually estimated by manufacturers and sometimes stated in data sheets, previous to sales.

CBM is based on the completion of inspecting activities by which maintenance actions will be initiated and completed. It implies monitoring system performance, so that, preventive actions, or fault forecasting, can be conducted based on measurements taken in run time. This principle can be applied to components, which do not exhibit failure predictability or random fails. However, scheduled maintenance activities may be applied to those components, while they present evident signs of wearing [5].

Contrary to PM, CM is based on failure occurrence or breakdown [10]. It considers repair and conduct part replacement once a failure had occurred [11]. It can be implemented by following the principles of:

- **Fault detection and isolation (P_4):** aims to detect and determine whether a fault occurred in a particular system, by trying to autonomously detect these faults and to diagnose the affected component [12].
- **Fault tolerance (P_5):** aims to assure the continuity of system operation, despite the presence of faults, errors or attacks [13].
- **Fault reporting (P_6):** is based on alerting the user or operator in case of fault to allow corrective actions to be taken [14].
- **Opportunistic-based maintenance (OBM) (P_7):** suggests completing general inspection of all of the components when any of them fails. It has been reported that combining with PM activities can lead to savings in terms of system cost [15]. Therefore, the principle of OBM states that there is an opportunity to conduct general maintenance when a maintenance intervention is required for other components [16].
- **Design Out Maintenance (DOM) (P_8)** aims to use redesign to avoid the causes of failure. This principle is usually applied when breakdowns frequently occur [17].

The application of these principle depends on how likely the components exhibit wearing characteristics and how randomly the components fails.

In the context of software maintenance (and knowledge intensive information systems) the main assumption is that no physical processes will be exhibited. However, these types of systems are also subjected to aging. Software aging is the outcome of a decrease in software performance that is usually caused by the accumulation of operational errors [18]. As mentioned in section 2.2.2, there are four strategies used in software maintenance, namely, (i) corrective maintenance, (ii) adaptive maintenance, (iii) perfective software maintenance, and (iv) preventive maintenance.

In the case of hardware components, preventative maintenance and corrective maintenance are conducted with the same objective. However, in the case of a software system, preventative maintenance involves actions such as changing software components without affecting the external behavior of the code [19]. On the contrary, corrective actions imply reproducing the observed failure (through failure injection) in order to investigate its causes, and to conduct code modification for preventing a future

occurrence. Code modification entails the need for modifying the associated documentation [20]. There are two other important principles, which aim at avoiding failures that are caused by evolution of a system or its environment. Operationalization of these principles is strongly dependent on expert knowledge and intervention. These principles are:

- **Adaptive maintenance (P_9)**: aims at software modification in order to adapt a system to technology upgrades [19]
- **Perfective maintenance (P_{10})**: involves software improvements in order to adapt a system to new use requirements [20].

In summary, our analysis explored that:

- from a physical perspective, maintenance is in general conducted to avoid the emergence of system failures, or to reduce failure occurrence based on repairs, parts replacement, and revision activities, and
- from the software (and cyber) perspective, the principles of preventive and corrective maintenance are also applied, but software sub-system upgrade is also considered as a possibility for assuring a proper software performance.

It can be seen from the above overview that there is a wide variety of the maintenance approaches. The majority of them can be applied to 0G-CPSs, no matter if they comprise both physical and software sub-systems. In order to make the spectrum of maintenance approaches more transparent, we propose taxonomy, which arranges the approaches according to the involved maintenance principles. This taxonomy is shown in Figure 6.3. As much as dependability and maintenance of 1G-CPSs is concerned, the most relevant approach is deemed to be some sort of combination, or even merging of relevant maintenance principles. However, the increase on system complexity that 1G-CPSs imply brings out important challenges, such as higher level of unpredictability. It can hamper the operationalization of the already existing maintenance principles, or affecting their cost-effective implementation. This situation makes it necessary to investigate how maintenance principles should be adapted to meet the functional requirements of 1G-CPSs.

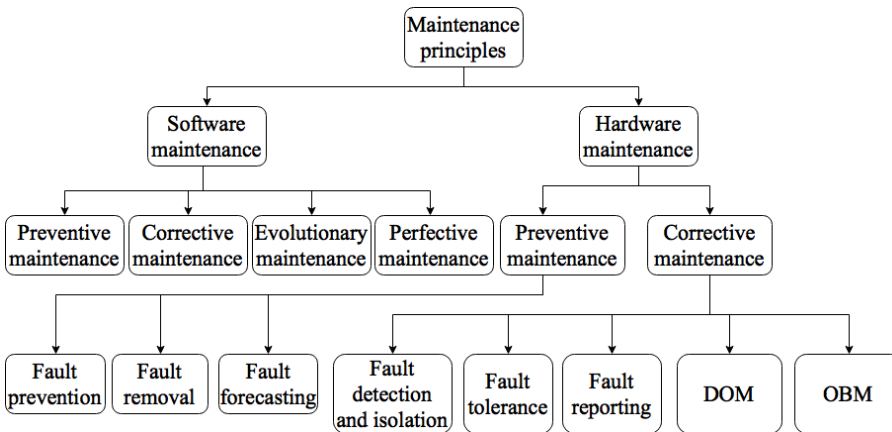


Figure 6.3 Taxonomy of maintenance principles

6.2.2 Projecting the maintenance principles of 0G-CPS to 1G-CPSs

In this section, we will analyze which maintenance principles relevant to 0G-CPSs can be applied to 1G-CPSs, considering similarities and differences between these two types of systems. We will study such principles with a view to facilitate the implementation of a holistic maintenance doctrine that covers hardware, middleware and software constituents in an integrated way [21], so that, new influential factors emerging from the interaction of the constituent elements are considered for failure reduction (see Figure 6.4). This doctrine involves not just new input information, but analyzing system condition from a system level perspective, making it required the development of new indicators and reasoning strategies. It results on the development of new maintenance procedures too. For this purpose, in this section we will mention the challenges that every maintenance principle entails, considering the main characteristics of 1G-CPSs, and in section 6.3.1 we will evaluate to what extent our findings contribute to facilitate the implementation of such principles, according to this holistic doctrine.

- ***Avoid failures in the system by preventing the occurrence of faults (P₁)***

This principle is paramount for assuring a reliable system operation. The best possible scenario, when thinking about CPS operation, is to avoid failure occurrence as long as possible. Failure prevention in 0G-CPSs is mostly conducted through TBM. However, the frequent SOM transitions in 1G-CPSs is still a challenge. System behavior is no longer predictable. Variations in system behavior (manifested through SOM transitions) affect the estimated lifecycle of the physical components and, thus, the frequency of maintenance may be different for each of them. It causes variations on wearing patterns that hamper the implementation of maintenance schedules. Prevention from the design stage brings important challenges too. 1G-CPSs are dynamic and highly complex systems and the currently used testing approaches cannot cover all aspects of the operation of CPSs, as they are mostly conducted at component level. Therefore, this principle cannot be transferred to 1G-CPSs without adaptation. Adaptation means either considering new information types and/or information sources, the development of new tools and methods, the definition of new models, or even a change of paradigm concerning to the way in which the studied principles are currently implemented.

- ***Reduce the amounts of faults and their severity from the design stages (P₂)***

This principle is applied during the system design, with the objective to avoid functional and structural failures. There are several design methods that can contribute to properly define the main system features, reducing the likelihood of failure during system operation. Due to that, we consider that this principle can be transferred to 1G-CPSs. Nevertheless, knowledge about the most critical system aspects should be known in advance, in order to use it to define design requirements. These critical aspects from 1G-CPSs might not coincide with those from 0G-CPSs, due to the multiple and different SOMs that 1G-CPSs can present. It requires testing of system performance under all SOM possible, to discard design aspects that can

lead to failures. It may happen that certain SOMs can make vulnerable some system components too. They may lead to early depletion of its components by requiring them to operate on the edge of their limits. Notwithstanding this aspect, this particular principle does not require adaptation. It is the information used as input that should be coherent with 1G-CPSs.

- **Forecasting failures on the system (P₃)**

The main objective of applying this principle is to forecast faults and failures and systematically avoid them. It was mainly implemented through TBM in 0G-CPSs, by considering components life estimations conducted by manufacturers. However, the dynamic behavior of 1G-CPSs caused by SOM transitions affects the wearing patterns of system components. They do not present a stable operation any longer. They are subjected to variations on their frequency of use and intensity, making it unreliable the use of existing estimations of components life. Moreover, SOM transitions triggered by feedback control may prevent to observe wearing symptoms in output signals. They start to be identified when controllability is no longer possible. It causes that currently existing forecasting models cannot be directly transferred from 0G-CPSs to 1G-CPSs. They require adaptation, with regards to tackle the challenges that dynamic systems such as 1G-CPSs imply.

- **Detect and isolate faults (P₄)**

0G-CPSs are equipped with capabilities to detect fault events, but they cannot reason about the consequences of emergent faults. This principle must be transferred to 1G-CPSs. However, it should have learning capabilities, so that, it can learn emergent and unknown failure modes. Moreover, the implementation of self-tuning capabilities affects failure manifestation, leading to false failure alarms, misdiagnosis or false positive results, due to the masking effect exerted by feedback control. The intensive sensor implementation in 1G-CPSs reveals important opportunities to conduct automated fault detection and isolation. However, the aforementioned aspects should be resolved with regards to properly implement this maintenance principle.

Such principle is considered critical, as it determines the corrective actions required for restoring system operation. Due to that, P₄ requires important adaptation.

- **Assure continuity of system operation despite the presence of faults (P₅)**

This principle has been mostly implemented through redundancy in both, hardware and software components.

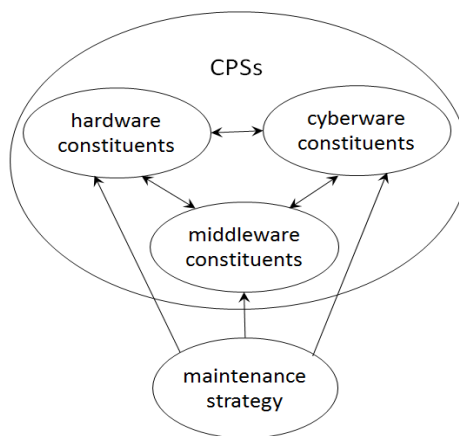


Figure 6.4 Doctrine of integral maintenance for CPSs

Nevertheless, the self-tuning capabilities of 1G-CPSs expose new opportunities in terms of fault tolerance. The first and more evident is the capability of compensating failure effects through the manipulation of system actuators, with the aim to keep system parameters on the desired levels. Some other characteristics such as distributed control allows conducting an adaptive resource management. It enables a redistribution of system tasks whenever the fulfillment of a particular function is affected by system occurrence. Even if compensatory actions manage to keep system operation, fault tolerance usually implies that failures cannot be perceived, avoiding failure detection and diagnosis. This situation is critical because not being aware about failure presence, can lead failure to reach its critical threshold, hampering system controllability. Due to that, this principle requires adaptation in order to be implemented in 1G-CPSs.

- ***Alerting the operator in case of fault (P₆)***

This principle can be transferred to CPSs as its implementation only entails the application of information technologies in physical devices, as 0G-CPSs and 1G-CPSs have this feature. The differences between system features of both systems do not affect their implementation. Nevertheless, not considering the variations that every single SOM implies on system behavior can lead to false failure alarms, or to letting pass failure symptoms. Nevertheless, this is mainly a problem of P₅

- ***Conduct general maintenance once any of the system components fail (P₇)***

This principle implies that, in a general stop once a failure has occurred, all components condition may be checked, taking the opportunity that the system is stopped. Although downtimes should be avoided, failures cannot be avoided at all. We can reduce their chance of occurrence and their severity. But failures will still occur. This principle can be transferred to 1G-CPSs. However, the maintenance strategy should not be supported in this principle. A preventive strategy should prevail. Due to that, failures should be avoided as long as possible. But when a particular failure occurs (i.e. whenever it could not be forecasted, prevented, nor tolerated) and the system is inevitably stopped, it is suggested to check the other non-failed components and conduct changes if it is required.

- ***Redesign to avoid the cause of failures (P₈)***

This principle is not related to any particular system feature of 0G-CPSs or 1G-CPSs. The application of this principle entails re-designing components and features of the system if they prone to be the source of recurrent failures [9]. Redesigning may be needed or be advantageous because recurrent failures can largely affect the overall availability of 1G-CPSs and can increase the costs of operation. In the case of 1G-CPSs, further consideration is needed if the redesign should focus on hardware or software as well as the information contents or any combination of them. Moreover, the system operation in the different SOMs should be analyzed. The system may operate properly in certain SOMs, but they may present poor performance in some other SOMs, making it required a redesign.

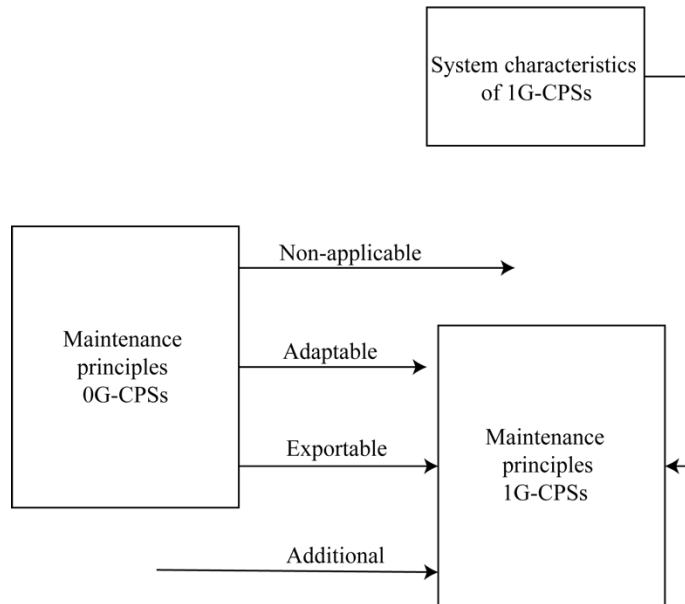


Figure 6.5. Roadmap towards maintenance principles for CPSs

- **Keep system operation despite hardware upgrade, or evolutionary external conditions (P₉)**

This principle is very important, considering that system upgrades may cause incompatibilities among system components. Failures caused by such incompatibilities may hamper the activation of certain SOMs (Considering SOMs are the combination of active components), limiting system operation. Considering the impact it may have, this principle should be extended to all the CPS dimensions, namely, hardware, software and middleware constituents. Due to that, this maintenance principle requires adaptation.

- **Incorporate new user requirements (P₁₀)**

Like P₉, this maintenance principle should also be extended to hardware and middleware constituents, by eliciting and implementing user requirements concerning the operation, and manipulation, of the physical components, as well as the technical specifications related to the use of information and service delivery. Nevertheless, its implementation in 1G-CPSs is not straightforward. Updating system implies important financial investment and downtimes, as the system or subsystems should be stopped in order to install new system components (either hardware or software). Hardware or software upgrade, due to new user requirements can result in a new SOMs or the end-up of an already occurring SOM. This particular principle does not seem to affect system reliability though, it does not endanger system operation, but the perception of the user concerning system performance. Due to that, we will not consider it yet for 1G-CPSs. This principle gains importance after the second generation CPSs (2G-CPS).

6.2.3 Operationalization of relevant maintenance principles for CPSs

Based on the analysis presented in the previous section, we can categorize the studied principles into four main categories, according to their potential implementation in 1G-CPSs, (Figure 6.5), that are: (i) non-applicable principles (P_x), (ii) adaptable principles (P_a), (iii) exportable principles (P_e), (iv) additional new principles (P_n). One maintenance principle, namely P_{10} , belongs to the category of non-applicable principles, that is, $P_x = (P_{10})$. This principle can be considered as a design issue, more than a maintenance concern and thus, we will not analyze it further in this dissertation. Nevertheless, it is important to remark, that this particular principle will gain importance as CPSs evolve to a higher generation with self-evolutionary capabilities. The other principles seem to be applicable but in different ways. For example, there are principles that can be applied without any modifications, so-called ‘Exportable Principles (P_e)’, such as $P_e = (P_2, P_6, P_7, P_8)$. They can be used without modifications, but the way in they are applied depends on variations on the input information or the specific CPS characteristics.

The remaining group of principles can be applied only after a purposeful adaptation, such as $P_a = (P_1, P_3, P_4, P_5, P_9)$. We observed that certain system features of 1G-CPSs would require additional (not yet specified) maintenance principles because they cannot be addressed by the principles known to be applicable to 0G-CPSs. During the next section (see section 6.3.1) we will analyze both, the exportable and adaptable set of principles, with regards to determine the role of SOMs on their implementation in 1G-CPSs.

6.3 About maintenance principles for CPSs

6.3.1 Opportunities emerging from the SOMs concept

SOMs can play an important role in the maintenance of CPSs. The operationalization of self-tuning requires a system equipped with sensing, reasoning and actuation technologies, enabling multiple SOMs. Every time the system should face a different working condition it can change its own settings to provide the most suitable behavior, through SOM transitions.

From the failure analysis perspective, our findings brought new opportunities to facilitate the implementation of the analyzed maintenance principles. One of them is that, by definition, SOM concept implies a system level analysis, since it is the combination of control actions what actually determines the current SOM (see section 1.5). It allows considering failure symptoms coming from components, but also those that emerge from the interaction among them. Considering that SOMs analysis is based on system monitoring, the decision-making process can be supported on data measured in run-time. It allows replacing the obsolete pre-defined life estimations, with continuous and updated time to failure estimation, leading to cost-effective maintenance decisions. Finally, the consideration of control signals (input signals), as well as sensed

system signals (output signals) enables avoiding the failure masking effect exerted by feedback control.

Failure diagnosis and forecasting enables the decision-making process about maintenance actions. These actions can range from scheduling part replacements, up to re-distribute system operations, in order to perform preventive maintenance actions. This section will investigate what role SOMs can play in implanting the aforementioned principles for preventive maintenance of CPSs.

1. The exportable principles:

The set of Exportable Principles (P_e) includes those principles that can be used in the maintenance of 1G-CPSs without requiring any adaptation. However, even if these principles do not need reinterpretation or redefinition, the way of operationalizing them in the case of 1G-CPSs may be different from the way they were applied in the case of 0G-CPSs. For example, principle P_2 (Reduce the amounts of faults and their severity) can be directly applied (without adaptation) to 1G-CPSs as it implies design actions. The most critical step is the availability of relevant information about potential system failures, so that a proper list of design specifications can be delivered. The elicitation of technical specifications should consider the different SOMs the system can present. It will allow properly sizing system components, so that, they do not have to operate beyond their limits to fulfill the working conditions of certain SOMs.

The reason why principle P_6 (Alerting the operator in case of fault) can be applied to 1G-CPSs, without adaptation, is that the system functionality and the used technologies can support alerting the operator in the case of a fault. This may include:

- Diagnosis report generation.
- Ubiquitous communication of failure information.
- Decision-making while continuous system operation.
- Propose maintenance/repair actions.
- Identification of parts to be replaced.
- Resources determination.
- Tool demands.
- Capabilities and activities planning.

We argue that SOM-based failure analysis can be used for providing information about the occurring or expected failure mode. Changes in SOMs frequency and SOMs duration, have demonstrated to be a good failure indicator (see Chapter 5) and, consequently, they can be used as input to provide information about failures to the user. Besides, the analysis of SOM transitions can also provide relevant information about system performance, which is not necessarily related to failure occurrence. It can contribute to keep an optimal system operation and to the accomplishment of system goals.

The support for the decision-making about response actions and resource management still require further research. So far, we only focused the research objectives on the availability of information about failures and system condition. Nevertheless, integrated maintenance advisory systems should also assist decision-making processes. To achieve this, it is not only required a suitable identification of the forming failure mode, but also

to know which are the most suitable actions for reacting to failures occurrence or prevention. Moreover, a more specific alert report system should also define what information should be delivered to which stakeholder, including failure description, location, surrounding/context and criticality.

The process of applying principle P_7 (Conduct general maintenance once any of the system components fail) to 1G-CPSs is almost the same as to 0G-CPSs. It involves the analysis of the criticality level of component failures, as well as the urgency of response and repair actions. The objective of this analysis is to determine the major risk level of components with high probability to fail, in order to take decisions about the (re)design strategy. It also contributes to determine which maintenance actions require human intervention.

Considering the implementation of SOM-based analysis, it implies a data-driven approach. This favors the availability of historical information that enables determining the most critical components, as well as identifying failure causes. Based on our explorative analysis, SOM Frequency and SOM Duration enable the analysis of the failure forming process. Moreover, the sequence of SOM transitions can also contribute to analyze failure causes and their influencing factors. It allows understanding how the compensatory actions evolved as effect of failure, so that, we can figure out which were the first variables affected by the occurring failure mode. Based on it, it can also be determined if applying structural redundancy, more resilient components, functional re-configuration or more robust system architecture, can be a better solution considering related costs and extra efforts [23].

Principle P_8 (Redesign to avoid the cause of failures) does not require any further adaptation process. Its use in 1G-CPSs should only respond to those cases in which failure could not be avoided and the system is fully stopped. It is a good opportunity to check the other components state and it is possible to change those that present evident wearing signs. Nevertheless, components condition should be constantly monitored to avoid failure occurrence (we cannot wait until a failure occurs for acting). The components' condition can be inferred from the compensation actions manifested through variations on the SOM Frequency and SOM Duration. Due to the aforementioned issues, a partial implementation of P_8 should be conducted in 1G-CPSs, i.e. not intentional but incidental implementation.

2. The adaptable principles

As mentioned before, the set of Adaptable Principles (P_a) contains those principles that should be (and can be) adapted. Their adaptation needs further considerations of the system features. In the following paragraphs the changes that should be made in case of these principles are considered. For instance, principle P_1 (Avoid failures in the system by preventing the occurrence of faults) requires adaptation in order to provide optimal results for 1G-CPSs. We should differentiate between the preventive actions that are conducted during the design and manufacturing stages and those actions conducted during system operation.

Concerning components design and manufacture, the adaptation should consider new different types of tests, which takes into account the effects of unexpected external and internal events. Currently, there are limitations in terms of what can be tested through functional or performance simulations and runtime tests. They should be able to deal

with unique faults and failures of 1G-CPSs, which do not occur in 0G-CPSs. It appears to be necessary to design new protocols for behavioral and performance tests, to determine how they should be conducted, which values are expected (as key performance indicators) and how to aggregate these in distributed and decentralized systems. These new protocols should consider that designed components are subject to different SOMs and transitions among them. Consequently, they evaluate component or system performance, during the various possible SOMs and their corresponding transitions.

During system operation, failure prevention must be based on system condition monitoring, through CBM. TBM is no longer trustable, due to the varying operative conditions that a 1G-CPSs should face, making CBM to be preferred over TBM. Its implementation is possible due to the availability of technologies, such as sensing, monitoring, information processing, fault diagnosis and failure prognosis algorithms [24]. Even if SOM Frequency and SOM Duration have shown to be good failure indicators, the processes of failure diagnosis and forecasting, that uses these indicators for determining the forming failure mode, are still required. Moreover, a maintenance advisory system that determines the most suitable response action, as well as the most optimal maintenance time¹ is needed, to replace the traditional maintenance schedules.

Principle P_3 (Predicting failures on the system) places the emphasis on the run-time prediction of possible failures and black outs of CPSs. This principle assumes prognosis and/or probabilistic models that can prognosticate system operation, based on evaluation of subsequent system states. The predictive models currently applied for 0G-CPSs are not transferable to 1G-CPSs due to the dynamic nature and operation conditions of 1G-CPS. Relevant predictive models should be able to capture the internal dynamics of the system, and the dynamics of the embedding environment. Towards an effective application of this principle, forecasting mechanisms are needed. They should be able to forecast future faults and failures of 1G-CPSs based on real-time operation or historical information. Based on our explorative study, we consider that the main contribution of the SOM concept is in failure forecasting.

The flexibility that SOM-based system description provides enables evaluating the failure forming process, despite variations on system operation. It was already seen (as described in section 5.3.2 and 5.5.1) that the study of system degradation based on SOM Frequency and SOM duration, presents clear trends that allows investigating system degradation. This enables the implementation of already existing time-series based forecasting methods. It allows preventing progressive failures. However, the analysis of sudden failures is still an issue. Electronic devices are prone to fail without presenting prior symptoms endangering the safe system operation and affecting system reliability. Even though, we did not analyze these failure types in this research, its prognosis should still be studied in order to provide a solution that contributes to improve system performance.

In the context of 1G-CPSs, the objective of principle P_4 (detect and isolate faults) is to achieve it by a collaborative strategy that involves maintenance experts, system instrumentation and the implementation of a failure diagnosis technique. These techniques require an information platform that can be generated by continuous

¹ Based on a trade-off between system condition and the accomplishment of the system objectives

monitoring of the system, reflective real-time modification of detection algorithms, introducing changes in the system arrangement and planning response actions. Once failures are known, they can be weighted based on the probability of occurrence, as well as on their criticality from the point of view of system operation. Based on our exploratory study, we consider that behavior indicators (SOM Frequency and SOM Duration) can provide the information required for evaluating failures and determining the response actions. However, the development of failure diagnosis tools that properly implement SOM Frequency and SOM Duration as failure indicators is required. Moreover, transitions between SOMs also provide relevant information, as it allows tracing failure evolution and, thus, determining the most suitable time instant for conducting repairs and changes of system components.

Application of principle P_5 (Assure continuity of system operation despite the presence of faults) requires minor adaptations. For example, some feedback control systems (i.e. Fault tolerant control) keep system stability by compensating the failures effect on system parameters. However, as it was already discussed before in section 6.2.2, this compensation may mask the failure effect and reduces reliability of failure diagnosis. Some already existing failure diagnosis methods make use of control signals (input signals) with regards to overcome such problems (see section 3.2.2). However, these methods are conducted at component level. Our results demonstrated that monitoring the compensation tasks that are manifested through changes on the SOM Frequency and SOM Duration could provide relevant information in failure diagnosis, while keeping fault tolerance. Unlike the already existing methods, the proposed analysis is performed at system-level, considering symptoms that arise from components interactions. Due to that, we consider that the adaptation required for a suitable implementation of P_5 in 1G-CPSs implies the development of failure analysis techniques that integrates the changing SOM Frequency and SOM Duration as failure indicators.

Finally, the application of P_9 (Keep system operation despite hardware upgrade, or evolutionary external conditions) in 1G-CPSs conveys important challenges. This principle should be extended to hardware and information, as well. It implies updating system components and database information without affecting system operation. Nevertheless, this process implies human intervention, due to hardware and software upgrades can't be conducted without user instructions. Moreover, the cost of continuous system upgrade cannot be neglected.

Components redundancy contributes to upgrade software and hardware components without requiring long downtimes. However, it is not always possible to implement redundant devices for all system components. Another option to avoid complicated processes during components replacement is to force the system to work in a particular SOM that does not involve the component to be changed. Nevertheless, this partial system operation is not always possible, particularly in mission critical systems. Further research should be conducted with regards to ease system upgrade. This situation is very important, as technology progresses faster day by day. Finding solution to the adaptation process of this principle implies a step ahead towards the development of self-evolutionary systems.

6.3.2 Some hints on specific maintenance principles for CPSs

Which new principles are needed for a particular family of CPSs? It is obvious that due to the complex functionality and self-tuning capabilities of 1G-CPSs, they need additional maintenance principles. Their intense interaction with the natural and engineered environment and penetration into the social and cognitive domains of stakeholders need further investigations, because of the increasing exposure of the environment and people. The main system features that raise the need for novel maintenance principles are: (i) System vulnerability (ii) applications in dynamic and harsh environments and (iii) growing level of automation.

Many 1G-CPSs are mission critical systems. They control critical infrastructure on which human life and natural resources depend. It makes them a target of attack and thus, they should be protected. If anyone manages to access a CPS control, they will be able to manipulate not only software resources, but also hardware. It will also provide access to interconnected systems, leading to tragic results. The development of new principles that aim, not just to avoid attacks, but also to tackle them properly (in case they cannot be avoided) are required. For this purpose, mechanisms for detecting attack attempts are required. These mechanisms should block the access to third parties and reporting the threat to interconnected systems. Besides of it, self-tuning capabilities can be exploited, so that, the system is put into a safe SOM in case an attack is detected.

The operation of CPSs in unpredictable and harsh environments, such as chemical reagents and humidity also imply the need for new maintenance principles. These and similar operating conditions invalidate the traditional forecasting models, as these conditions will most likely affect the hardware component's lifecycle and increases the chances of malfunctioning. Based on our explorative analysis, the study of the evolution of the SOM Frequency and SOM Duration provide the means for forecasting time to failure. Unlike the traditional methods, the SOM-based forecasting process, conduct prognosis based on monitored data coming from the running system. It enables considering the effect of use and operative conditions on the forecasting process. It fits with the current research trending topics, which are engaged with finding theories and technological solutions for inherently fault-tolerant dynamic architectures, as well as non-model-based zero-delay monitoring and proactive detection solutions. However, the wearing effect of the environment over electronic devices, as well as incomplete datasets caused by system malfunctions can affect forecasting results. Problems in sensing or control components will hamper the report of actuators status, as well as the availability of the sensed system parameters. It will hinder the execution of failure diagnosis and forecasting models, due to the lack of data samples, or it will lead to unreliable failure diagnosis results.

CPSs systems are reaching a high level of autonomy. This is enabled by the increasing smartness, which is a result of the development of sophisticated reasoning/infering techniques, and the implementation of software agents that supports evaluating system performance, and failure diagnosis in runtime. Additionally, maintenance automation has not only positive technical outcomes, but also reduces the required human efforts, intervention, costs and safety, as well as it improves servicing capabilities [25]. It seems to be necessary to include maintenance oriented abstractions in model-based design of

CPSs and to be able to detect near failure states in operation. Our explorative analysis suggests that failure diagnosis and forecasting (through SOMs), could provide relevant information for decision-making about maintenance. However, the automated operationalization of the maintenance actions is still a challenge.

Finally, there is a need to develop self-maintenance principles for various families of CPSs. As discussed earlier (See section 6.2.2), part of the current maintenance principles can be considered in the case of systems with self-detection (self-diagnosis) and failure prevention capabilities. State sensors built into physical components, smart materials and emergent behavior analyzers, are already used in current CPSs. These principles should fully cover the maintenance process in such a way that human involvement is considerably reduced. The next step would be the combination of currently used self-diagnosis and failure detection methods, with new techniques for monitoring, replacing and repairing components during system operation. J. Lee, et al. argued that “self-maintenance techniques should be aware of the changing operating regimes to dynamically select prognostic models to ensure accurate prediction” [26]. We consider that SOMs can be used for enhancing self-awareness. The operationalization of self-maintenance through response actions via automated actuators can be conceived. However, it requires further development in two main domains: i) Reasoning mechanisms that select (in runtime) the most suitable maintenance actions to be executed, according to multi-variable analysis. ii) The execution of autonomous maintenance actions requires a proper infrastructure, hardware and facilities that should be developed and properly installed to that particular purpose. Further research should be conducted in these two subjects.

6.4 Conclusions

In this chapter, we have reviewed the maintenance principles currently applied in 0G-CPS with the intention of determining if they are relevant to maintenance of CPSs. The role of the SOM concept for the implementation of such principles has also been explored. Due to the proliferation of 1G-CPSs and their applications, including in mission critical areas, there has been a growing need to analyze how maintenance of these systems should be conducted and to identify maintenance principles that can be successfully applied to them. 1G-CPSs are complicated complex systems, which nevertheless have some similarities with 0G-CPSs. For instance, both integrate information technologies into physical devices, are geographically distributed, have multiple energy sources, functional units and intense interactions with human stakeholders and the embedded environment. In contrast, 1G-CPSs feature a multitude of functional connections among the components, can optimize their operation and are developed to operate in dynamic or harsh environments. There is also a large dissimilarity between their system features. These inspired us to analyze to what extent generic maintenance principles of 0G-CPSs could be transferred to 1G-CPSs.

Our analysis revealed that self-tuning capabilities makes unpredictable system behavior, hindering the application of the current maintenance principles. Changes on system dynamics caused by the execution of self-tuning actions lead to changes on the wearing patterns of system components and mask failure effect, due to the compensation effect exerted by the system control to keep system stability. We also found that the changing frequency and duration of SOMs can contribute to ease the implementation and

adaptation of the currently existing maintenance principles to 1G-CPSs. It allows analyzing system performance from a system level perspective, letting to consider the symptoms that arise from components interaction. The use of run-time system data enables conducting failure diagnosis and estimating the system time to failure. It makes possible the implementation of optimal maintenance, by providing support to the maintenance decision-making supported by ‘on-the-go’ system operation information. It also allows tracing the failure cause, by providing information about failure evolution, based on the observed variations of the compensation tasks. Nevertheless, further research is needed to fully validate these propositions.

We also consider that CPSs can be vulnerable to attacks and harsh environments. It makes it necessary to develop new maintenance methods and tools to face these unexpected situations. The autonomous operation of maintenance actions is also required. It is the next step towards more reliable CPSs, so that, we can optimize maintenance processes. Nevertheless, further research is needed, mainly by studying real world environments, in order to reveal those new maintenance principles that are needed and how they can be operationalized.

6.5 References

- [1] S. Ruiz-Arenas, I. Horváth, R. Mejia-Gutierrez, and E.Z Opiyo, “Towards the maintenance principles of Cyber-Physical Systems,” *Journal of Mechanical Engineering – SV*, pp. 815 – 831, 2014.
- [2] G. Waeyenbergh and L. Pintelon, “Maintenance concept development: A case study,” *Int. J. Prod. Econ.*, vol. 89, no. 3, pp. 395–405, Jun. 2004.
- [3] M. Moesgaard, M. Froholdt, and F. “The strategy landscape,” *Return on Strategy: How to Achieve it!*, New York, NY,USA: Routledge, 2010.
- [4] A. Stevenson, *Oxford dictionary of English*, 3rd ed. Oxford: Oxford University Press, 2010.
- [5] A. Kelly, " Appendix 1: Maintenance terminology," In *Strategic Maintenance Planning*, Oxford, England: Elsevier Ltd, 2006, pp. 267-268.
- [6] S. Xu, “On dependability of computing systems,” *J. Comput. Sci. Technol.*, vol. 14, no. 2, pp. 116–128, Mar. 1999.
- [7] A. Avizienis, J.-C. Laprie, and B. Randell, “Dependability and Its Threats: A Taxonomy,” in *Building the Information Society*, R. Jacquart, Ed. Boston, MA, USA: Springer US, 2004, pp. 91–120.
- [8] A. Jhumka, “Dependability in Service-Oriented Computing,” in *Agent-Based Service-Oriented Computing*, N. Griffiths and K.-M. Chao, Eds. London, England: Springer London, 2010, pp. 141–160.
- [9] A. Starr, B. Al-Najjar, K. Holmberg, E. Jantunen, J. Bellew, and A. Albarbar, “Maintenance Today and Future Trends,” in *E-maintenance*, K. Holmberg, A. Adgar, A. Arnaiz, E. Jantunen, J. Mascolo, and S. Mekid, Eds. London, England: Springer London, 2010, pp. 5–37.
- [10] L. M. Pintelon and L. F. Gelders, “Maintenance management decision making,” *Eur. J. Oper. Res.*, vol. 58, no. 3, pp. 301–317, May. 1992.
- [11] M. A. Burhanuddin, S. M. Halawani, and A. R. Ahmad, “An Efficient Failure-Based Maintenance Decision Support System for Smalland Medium Industries,” *An Efficient Failure-Based Maintenance Decision Support System for Smalland*

- Medium Industries, Efficient Decision Support Systems - Practice and Challenges From Current to Future*, Sep. 2011 Accessed on: 12, 14, 2017. doi: 10.5772/18187, Available from: <https://www.intechopen.com/books/efficient-decision-support-systems-practice-and-challenges-from-current-to-future/an-efficient-failure-based-maintenance-decision-support-system-for-small-and-medium-industries>
- [12] E. Dubrova, "Fundamentals of Dependability," in *Fault-Tolerant Design*, New York, NY, USA: Springer New York, 2013, pp. 5–20.
- [13] J. Kontoleon, "Status and Trends in the Performance Assessment of Fault Tolerant Systems," in *Handbook of Performability Engineering*, P. K. B. Misra, Eds. London, England: Springer, 2008, pp. 1087–1106.
- [14] K. P. Birman and B. B. Glade, "Reliability through consistency," *IEEE Softw.*, vol. 12, no. 3, pp. 29–41, May. 1995.
- [15] X. Zhou, L. Xi, and J. Lee, "Opportunistic preventive maintenance scheduling for a multi-unit series system based on dynamic programming," *Int. J. Prod. Econ.*, vol. 118, no. 2, pp. 361–366, Apr. 2009.
- [16] L. Pintelon and P. N. Muchiri, "Safety and Maintenance," in *Handbook of Maintenance Management and Engineering*, M. Ben-Daya, S. O. Duffuaa, A. Raouf, J. Knezevic, and D. Ait-Kadi, Eds. London, England: Springer, 2009, pp. 613–648.
- [17] A. W. Labib and M. N. Yuniarto, "Maintenance Strategies for Changeable Manufacturing," in *Changeable and Reconfigurable Manufacturing Systems*, H. A. ElMaraghy, Ed. London, England: Springer, 2009, pp. 337–351.
- [18] M. Grottko, R. Matias, and K. S. Trivedi, "The fundamentals of software aging," in *2008 ISSRE Wksp*, 2008, pp. 1–6.
- [19] S. Christa, V. Madhusudhan, V. Suma, and J. J. Rao, "Software Maintenance: From the Perspective of Effort and Cost Requirement," in *Proc. ICDECT 2016*, pp. 759–768.
- [20] K. H. Bennett and V. T. Rajlich, "Software Maintenance and Evolution: A Roadmap," in *Proc. of the Conf. ICSE00*, 2000, pp. 73–87.
- [21] Horváth and Gerritsen, "Cyber-Physical Systems: Concepts, technologies and implementation principles," presented at the TMCE 2012, Karlsruhe, Germany, 2012, pp. 19–36.
- [22] I. Horváth, Z. Rusák, and Y. Li, "Order beyond chaos: Introducing the notion of generation to characterize the continuously evolving implementations of cyber-physical systems," in *Proc ASME 2017*, p.p. 1-10.
- [23] M. Colnaric, D. Verber, and W. a. Halang, "Real-time Characteristics and Safety of Embedded Systems," in *Distributed Embedded Control Systems*, London, England: Springer London, 2008, pp. 3–28.
- [24] C. Chen, D. Brown, C. Sconyers, B. Zhang, G. Vachtsevanos, and M. E. Orchard, "An integrated architecture for fault diagnosis and failure prognosis of complex engineering systems," *Expert Syst. Appl.*, vol. 39, no. 10, pp. 9031–9040, Aug. 2012.
- [25] A. Hanemann, M. Sailer, and D. Schmitz, "Assured Service Quality by Improved Fault Management," in *Proc. of the 2nd ICSOC*, New York, NY, USA, 2004, pp. 183–192.
- [26] J. Lee, M. Ghaffari, and S. Elmeligy, "Self-maintenance and engineering immune systems: Towards smarter machines and manufacturing systems," *Annu. Rev. Control*, vol. 35, no. 1, pp. 111–122, Apr. 2011.

Chapter 7

Conclusions, propositions, reflections and further research

7.1 Main results of the research

7.1.1 Moving towards failure management dedicated to CPSs

The literature study made it evidential that the paradigm of cyber-physical systems (CPSs) is rapidly developing. This is evidenced not only by the growing number of academic studies, but also by: (i) the growing number and versatility of systems implemented for practical applications, (ii) the increase of synergy achieved with regards to the enabling technologies, and (iii) the continuing efforts for providing novel functionalities and exploring further application needs. According to the ‘classical’ definition, CPSs are physical, chemical, biological and engineered systems whose operations are coordinated, controlled and monitored by a digital computing, communication and control core. At least as important is the fact that CPSs closely interact with and deeply penetrate into physical processes and environments, and that they perform smart behavior and act as adaptive actors in human and social contexts. CPSs are based on cyber-physical computing (CPC), which intends to overwrite the von Neumann theory-based computing (i.e. making calculations in a predefined way) and pursues dynamic computation based on information obtained run-time in recurrent cycles of sensing, reasoning, and actuating. CPC supports capturing emergent behavior of complex technical systems.

Cyber-physical systems are a new family of systems that fully integrate software and hardware components to provide new services and functionalities that were not possible without their synergistic operation. The main characteristics of CPSs are (i) functional, structural and interoperation (aggregative) complexity, (ii) functional and structural heterogeneity, (iii) quasi- or truly non-linear operation, and (iv) gradually extending smart behavior that allows them to adapt more and more autonomously to varying operation conditions. The paradigm of cyber-physical systems evolves from systems that need other systems (namely controlling subsystems) to regulate their system-level operation to systems that can manage, command, direct or regulate themselves and other systems. In order to place the various implements of CPSs in a conceptual framework, the concept of system generations was introduced. This identified four families of CPSs that represent different stages of paradigmatic evolution. We decided to focus our

research exclusively only on first generation CPSs. They are equipped with self-tuning capabilities, which enable them to optimize their behavior or protect themselves again unfavorable system states and operational conditions. Typically, 0G-CPSs work under the regime of control subsystems that measure the value of controlled variables of the system and apply the manipulated variables to the system to correct or limit the deviation of the measured value . No change in the system operation modes is supposed. In the case of 1G-CPSs, self-tuning is implemented by setting the system operation modes (SOMs) according to the objective of operation and the appropriate response to operation conditions. At the same time, first generation CPSs are equipped with only limited capabilities in terms of self-awareness, self-adaptation, or self-reconfiguration.

First generation CPSs are used for controlling critical infrastructures by performing reliable and autonomous system behavior based on changing SOMs. This promotion research aimed to investigate the effect of variations of SOMs on system behavior and the utilization of this information in failure analysis and forecasting. The knowledge generated and the insights gained in this promotion research enable to overcome the limitation of current failure analysis methods (that is the lack of ability to coping with variable system operation/architecture and non-linearity). Our main objective was to provide explanatory knowledge about the potential role of SOMs in failure analysis and forecasting. To systematize the conduct of research, five research cycles were defined, with the focus on: (i) exploring the currently existing failure detection and diagnosis techniques, (ii) analyzing the influential factors of the phenomenon of failure analysis (iii) analyzing the effect of SOM on failure symptoms in 1G-CPSs, (iv) analyzing the effect of SOM on the failure forming process, and (v) evaluating the challenges and opportunities that SOM implementation implies in the context of maintenance of 1G-CPSs. In the following paragraphs, we will present the main findings and their implications derived from our exploratory research.

7.1.2 Major findings concerning the state of the art of computational failure analysis

In the first research cycle, we surveyed the currently available failure analysis methods. Our conclusion was that were not developed according to the need of cyber-physical systems, therefore, they could not be applied without adaptation. At the same time, the strong need for novel (dedicated) methods was also recognized. We were interested in knowing to what extent the SOM concept has been considered and how has been operationalized in existing failure analysis and forecasting methods. Our investigations extended to the utilized (i) failure information carriers, (ii) data features, (iii) reference quantities, and (iv) failure decision enablers. The systematic analysis of the scientific literature was based on keywords such as fault analysis, diagnosis, prognosis, system reliability. The cited academic publications and web-based repositories (of technical videos and explanatory contents) served as sources for constructing a robust knowledge platform for the follow up studies.

Our literature review yielded that there were no articles that would have deeply explored the effect of system operation modes on failure analysis, in the context of 1G-CPSs. Although many approaches were focused on the analysis of system states, these approaches tend to present the following gaps:

Systems cannot deal with emergent behaviors and unknown failures

The shortcoming of these approaches is that systems cannot deal with emergent behaviors and unknown failures, which in turn affects system reliability. The tight connection with their surrounding environment makes 1G-CPSs more susceptible to the occurrence of unexpected failures, or emergent operative actions. It hampers the use of model-based failure analysis techniques, which are not capable to recognize failure modes whose symptoms are not known a priori. Likewise, data-driven techniques tend to misclassify data corresponding to unknown failure modes, by adjudging the observed data patterns to an already known failure. This situation triggers wrong failure diagnosis, hindering the appropriate selection of maintenance actions.

Existing failure analysis techniques strongly depend on prior knowledge

The study revealed that most of the already existing methods strongly depend on a priori expert knowledge. These methods strongly depend on the availability of knowledge about: (i) which system parameters to observe, (ii) which signal features to use, (iv) failure symptoms and manifestations, (v) the root-cause of failures and (vi) the way in which failures evolve. It hinders managing unknown failures, and their prevention. This situation is critical in 1G-CPSs, as failures can manifest differently in every SOM. Notwithstanding the importance of this aspect, we did not find in literature any study that aims to provide insights in this topic.

Differentiation of the failure symptoms from external disturbances is still an unsolved issue for complex systems

Our literature review also revealed that currently available failure analysis methods mainly use sensor signals (output signals) as failure information carriers. However, the sensor signals are influenced by the control regime, the failure modes and the external disturbances. This situation can lead to false failure alarms, or to prevent failure symptoms to be noticed, causing the failures to reach dangerous levels that can compromise the safe system operation. This problem is especially critical in 1G-CPSs, due to the frequent SOM transitions they present.

Systems equipped with feedback control mask failure symptoms

Signals-based failure analysis also presents another important drawback. Systems equipped with feedback control, such as 1G-CPSs, implement compensatory actions in order to maintain the stability of the system. This feature masks the effect of failures and hinders failure detection. We found that several failure analysis techniques use control signals in order to overcome this problem. However, these are mostly working on component level or require the implementation of analytical models. The former prevents studying the symptoms that arise from the components or subsystems interaction. The last one presupposes a deterministic system behavior and a simplification of the processes executed in the physical dimension.

Failure analysis is mostly conducted at component-level/

The conducted literature review demonstrated that most of the existing techniques are conducted at component level. Model-based techniques represent components operation based on their inputs and outputs. Nevertheless, the interrelation with other components or the surrounding environment is limited to representing the output of a system

component as the input of another one. This prevents the consideration of failure symptoms that are manifested in the interaction of system components. This situation is critical in 1G–CPSs, as these are systems of systems where the synergistic interaction between its forming components is paramount. The joint operation of the system constituents leads to new system operation modes where failures can manifest different.

Our literature review also revealed that the learning capabilities of data-driven techniques provide opportunity for an operational situation dependent failure analysis in the context of 1G–CPSs. Moreover, they allow conducting failure analysis with run-time data. This ability in turn facilitates the discovery of relationships between the analyzed variables without making a pre-defined model necessary. Based on this general-purpose failure management methods can be developed, which can be used for diagnosing different types of failure modes at component and system level. These methods allow coping with the complications that the dynamic behavior of CPSs conveys for failure analysis. Nevertheless, some important issues such as proper selection of signal features and the difficulties of interpretation of the obtained results are still problematic with respect to deriving knowledge about the role of SOMs in failure manifestations.

7.1.3 Major findings related to the influential factors of the investigated phenomenon

In the second research cycle, we studied the influential factors of failure analysis using theoretical and practical research approaches. In the theoretical approach, the concept of SOM was studied in order to analyze its implications in a first generation CPS. From this analysis, we determined that the most critical factors for succeeding in our analysis are:

- providing means for understanding the failure forming process,
- identifying the system parameters where failures are manifested,
- determining the time instants in which SOM transitions occur (so that control regime effect can be discriminated from failure effect),
- tracing the root-cause of failures.

An important achievement of our investigations is being able to capture the relationship between the effect of SOM transitions on system signals and the effect of failure and environmental disturbances. Towards this end, the following factors were highlighted as relevant in order to provide means for deriving the required knowledge:

Monitoring SOMs transitions combined with signal segmentation creates a robust basis for the differentiation of failure manifestations from regular self-adjusting operation of the control regime and for recognizing failures.

Recognition of the deviations is facilitated by segmentation of signals based on the operation mode of the system. Our failure analysis could be based on the assumption that statistical variance of signal characteristics in a specific operation mode is smaller than in the overall system operation. Based on it, we can infer that failures symptoms are manifested in systems signals differently depending on system's operation modes

and that both, symptom occurrence and the lack of symptom can be used as indicator for determining the type of failure.

Practical experimentation should be conducted with regards to derive knowledge about failure manifestations

1G-CPSs are characterized by their tight relationship with the surrounding environment. However, the effect of external disturbances, as well as the interaction system-environment cannot be accurately modelled nor predicted. Deriving knowledge about failure manifestations in CPSs should consider both factors, as these strongly affect the way in which failure symptoms are noticed. Moreover, these are also sources of malfunctioning or variations on system performance. This situation highlighted the need of using practical experimentation for fulfilling the purposes of our research. The selection of a particular family of CPSs was then required in order to deduct generalizable knowledge on the manifestation of the studied phenomenon from a concrete application. Our theoretical analysis led us to select a greenhouse testbed as a suitable CPS application, as it: (i) is strongly influenced by external factors, (ii) conducts multiple tasks in parallel (such as irrigation, ventilation, among others), (iii) represents a harsh environment for electronic components, and (iv) do not endanger human life or environmental resources in a failure mode.

1G-CPSs was mimicked in the greenhouse testbed system through the following characteristics:

- Full automation of the main greenhouse tasks, namely: lighting, irrigation, water reservoir filling, and ventilation.
- Multiple feedback-based control systems, particularly for regulating plant irrigation, adjusting water level in the water reservoir, and controlling lighting.
- The implementation of local and remote controls.

Self-regulation and self-tuning were enabled by the incorporation of multiple independent feedback-based controllers that allowed compensating disturbances and failure effect on the system. The implementation of sensors that evaluate the most relevant system parameters, and actuators that keeps them on the pre-defined levels made it possible for the system to present multiple system operation modes, and the execution of SOM transitions, which is one of the main characteristics of 1G-CPSs.

The testbed system was used to provide local control for performing the main greenhouse tasks. However, it was also connected to a reasoning unit that provided the means for the execution of more complex processes, such as failure analysis. The combination of the afore-mentioned characteristics enabled simulating the operation of real first generation CPSs, so that we could provide the means for experimentation.

Failure injection is required in order to reproduce failure effects in a controlled manner

Determining the most representative manifestations from every failure mode requires a large amount of data. This data should be collected in different failure events, so that we can reduce the bias that the operating conditions of every single event imply. However, failures do not occur very often, making it impractical waiting until failure occurrence for the execution of our explorative analysis. This situation led us to select failure

induction as a suitable mean for failure analysis. It enabled reproducing the same failure mode multiple times so that we could collect the amount of data required for our purposes, and repeating the experiment as many times as it is required.

In conclusion, research cycle 2 enabled getting insights by a literature review and reflecting on the most suitable way of deriving the required knowledge. There, a SOM-based signal segmentation approach was proposed for studying failure manifestations and the design and construction of a testbed that provides the data required for experimentation was conducted.

7.1.4 Major findings related to the analysis of SOM effect over failure manifestations

In the third research cycle an experimental investigation was conducted in order to analyze to what extent SOMs influence failure symptoms. A simulation model of a kettle, and the previously instrumented greenhouse testbed were used as means for experimentation. Different failure modes were injected in both systems. The rationale behind this investigation was that segmenting system signals based on SOMs would strengthen failure symptoms and contribute to discriminate between control actions and failure effect on signals. To implement signal segmentation based failure detection, the concept of failure indicator was introduced. This concept represents deviations of statistical features for each system signal in each system operation mode. The results revealed:

Signal segmentation improves failure detection as it strengthens symptoms

The analysis conducted through the kettle model demonstrated that failure indicators revealed symptoms that were not recognized based on the analysis of non-segmented signals. The implementation of SOM-based signal segmentation increased the statistical difference between the failure-free dataset and the failed one, enabling the observation of failure symptoms that were not noticed by analyzing the whole signal length.

The proposed failure indicator concept is sensitive to disturbances caused by external factors.

Our analysis revealed that investigation of failures when the system is subjected to variations in terms of use and operating conditions increases the number of cells that present statistical difference in the failure indicator. It demonstrates that, although SOM-based signal segmentation allows discriminating between the effect of control action and failure effects, it does not manage to discriminate the effect of disturbances caused by external factors. This characteristic is critical in first generation CPSs as their performance and operation are strongly influenced by their embedding environment. It results in increased number of false failure alarms, and it hampers the interpretation of failure indicators. Our analysis also revealed that signal-segmentation based failure analysis was sensitive to the number of measurements.

Failures lead to variations on the frequency and duration of SOMs as the system tries to compensate failure effect.

We also found that the frequency and duration of SOMs of a system in a failure mode has significantly changed compared to failure-free operation. The change of the

frequency decreased the observable sample size of the dataset in specific operation modes and reduced the statistical power of failure diagnosis method used by the failure indicator concept. The experiment also explored that the sequence of transition among the system operation modes was influenced by the failure mode. Those failure modes that did not occur in the regular sequence of operation, but were triggered by the failure manifestations were called as failure induced operation modes (FIOM). We found that FIOM hinders the application of SOM-based signal segmentation analysis, as it prevents comparison between the reference and the observed behavior. This conclusion seemed to be logical, especially in those cases in which failure manifests stronger in such a SOM, as it implied a loss of valuable information.

Failure induced operation modes are the most determinant factors when analyzing the discriminant power of the failure indicators

Our explorative analysis also investigated the discriminant power of the obtained indicators. We aimed to determine to what extent the observed symptoms could be used for characterizing specific failure modes. The results of our study demonstrated that there were differences between the failure indicators corresponding to the analyzed failure modes. However, such differences were weak, and were mostly caused by FIOMs. Despite the negative implications it had for SOM- segmentation based failure analysis, it casted light on a new opportunity too. We explored that variations of the frequency and duration of SOMs could be a potential failure indicator not only in failure diagnosis, but also in failure forecasting. The results suggested that variations in the frequency and duration of SOMs convey more relevant information for failure recognition and discrimination, than signal deviations. Nevertheless, it would require a deep-going and comprehensive analysis that could validate our first assumptions about the analytic potential of the changing frequency and duration of SOMs.

As a conclusion, research cycle 2 demonstrated that SOM-based signal segmentation strengthens failure symptoms. However, it is highly sensitive to external disturbances hampering failure detection and diagnosis. Nevertheless, the analysis conducted led us to an important finding. Variations in the frequency and duration of SOMs seem to be an important failure indicator that can contribute to differentiating between different failure modes, as well as to analyzing the failure forming process. In the next research cycle, we studied various failure manifestations based on the variations of such indicators.

7.1.5 Major findings concerning the analysis of SOMs in the failure forming process

In the fourth research cycle, we investigated the role of SOMs in the forecasting context. For this purpose, the variations of the frequency and duration of SOMs were considered as enablers for failure prognosis. The upward and downward trends of the frequency and duration of SOMs were used as failure indicators and for evaluating their uniqueness and discriminant power for various failure modes. This concept was tested both in a simulation and in the greenhouse testbed. The main findings obtained in this research cycle were:

Frequency and duration of SOMs can be considered as failure indicators

We analyzed the way in which frequency and duration of SOMs evolved as failure progresses. Clear upward and downward trends were formed in the frequency and duration of certain system operation modes, as effect of failure evolution. Our experiments led us to observe that datasets corresponding to the same failure mode presented similar trends, while the pattern composed by all SOMs trends differed between different failure modes. This finding enabled to conclude that the changing frequency and duration of SOMs are indicators of the failure forming process, and that, when analyzed for all SOMs in conjunction, can be used for discriminating data corresponding to different failure modes.

Frequency and duration of SOMs can be used as predictors for failure classification

To demonstrate the applicability of the changing frequency and duration of SOMs as failure indicator, we applied linear-discriminant-based classifier for failure diagnosis on the dataset collected from the testbed. Results were satisfactory as most of the analyzed cases were classified correctly with high rates of success. These results confirmed that the change of frequency and duration of SOMs has a significant distinctive power to be used for failure diagnosis. Nevertheless, these results may need further validation as the present research has an explorative nature.

The changing frequency and duration of SOMs enable failure forecasting

Our explorative analysis also revealed that failure indicators are able to capture trends of type of forming failures and can be used as information source for predicting time to failure. A demonstrative implementation was used to explore the potential application of the failure indicator concept in failure forecasting. In our study all failures were properly classified during the first stages of failure forming. Although the estimated time to failure was not accurately predicted (as it indicated shorter time to failure), failure indicators managed to report the correct type of failure before its occurrence. The inaccurate prediction of time to failure does not necessary mean that these failure indicators cannot be used for prediction of time to failure. As the time to failure was consistently shorter for each case in our study, their predicting accuracy could be improved by a compensation function taking into account the error of prediction.

As a conclusion, the fourth research cycle demonstrated that the changing frequency and duration of SOMs are failure indicators, when analyzed in conjunction for all SOMs. These indicators can be used as predictors for failure diagnosis and failure forecasting, enabling the estimation of the time to failure and determining the forming failure mode. Nevertheless, the concept of failure indicators needs validation in the highly complex systems as well as systems with non-linear behavior. Our study has only covered self-tuning systems with that present a low number of possible failure modes. Complete validation should explore how the number of signal sources, system operation mode, and failure modes influences the distinctive power and consistency of the failure indicator concept. This validation should also test the evolution of the indicators in failures that present different progress patterns.

7.1.6 Major findings concerning the role of SOMs in possible maintenance principles

In the fifth research cycle, we analyzed the applicability of existing maintenance principles to first generation cyber-physical systems. These principles were collected and reviewed in a literature study, which served as a knowledge base for critical reasoning on the potential embedding of SOM based failure diagnosis and forecasting in preventive maintenance of 1G- CPSs. Our main findings can be summarized as follows:

The passage from zero generation CPSs to first generation CPSs requires moving from time-based to condition-based maintenance strategy

Our analysis yielded that moving from a time-based maintenance strategy, to a monitoring-based one is necessary. This process makes it required stop taking maintenance decisions based on the components' life estimations conducted by manufacturers and start using run-time data as basis for decision making. It implies the use of data-driven techniques that are provided with learning capabilities. We claim that, the tedious work of feature selection in data-driven failure analysis can be replaced by the application of the frequency and duration of SOMs as failure indicator and failure predictor.

The implementation of SOM concept enables the definition of maintenance principles focused on the system level

Our analysis also highlighted that SOMs support analysis of failures on system level rather than on component level. The existing maintenance principles are typically component oriented. They prevent the detection of symptoms that arise from the interaction between system components and they do not support the analysis of failure propagation in the system. The concept of SOM is composed by the joint analysis of all system components states. Whenever a component switches to another state, it triggers a SOM transition. This allows evaluating how changes on system components impact the whole system behavior, based on the way they alter system dynamics (i.e. the regular sequence of SOM transitions). This knowledge contributes to the development of maintenance principles that consider the interaction between system components. It also contributes to take maintenance decisions for preventing failure occurrence and propagation.

System operation modes support corrective maintenance actions by allowing tracing the root cause of failures, whenever these cannot be prevented.

Using the change of frequency and duration of SOMs facilitates tracing of the root cause and the impact of failure. This observation implies that the change of SOM frequency and duration can not only fulfill the role of failure indicator for failure analysis and forecasting, but it can also be used for root cause and failure impact analysis. The sequence of SOM transitions, as well as the variations on their frequency and duration allows determining how actuators behave during the failure forming process. It provides hints for determining how failure was propagated, its origin, and the way in which the system managed to compensate the first failure manifestations. This information is very relevant, as it provides insights for avoiding and facing a further occurrence of the same failure.

The automation of maintenance actions is still an issue to be solved

Our analysis yielded that one of the research and technology development challenges remains is the automation of maintenance actions. The development of maintenance advisory systems that determine the preventive or corrective actions based on failure diagnosis is still required. Knowing the estimated time to failure, computational approaches utilizing multi-criteria optimization can contribute to determine the most optimal time for changing parts, conducting repair, and performing cleaning taking into account the impact of potential failure, the possible failure propagation, and the optimal operation of cyber-physical systems. Maintenance advisory systems should determine the best maintenance strategy based on the selection and implementation of the afore-analyzed maintenance principles (see section 6.4). Future research should explore if and how specific maintenance actions can be conducted by the system itself without the involvement of human operators. Research in resilient and self-healing cyber physical systems started to address these issues.

New maintenance principles that protects the system from external attacks are required

New principles related to the operation of the system in harsh environments, and for avoiding and managing external attacks are also required. Many first generation CPSs are mission critical systems. They control critical infrastructure on which human life, economic assets or natural environments depend, making them a target for external attacks. SOM transitions can be used as means for facing such threats. Once these are detected the system can self-tune by switching to a safe SOM that blocks sensible information, limits the actuators operation, or interrupt the internet connection. Nevertheless, further investigation should be conducted about this topic.

New maintenance principles that assure system performance in harsh environments are required.

Increased level of autonomy of first generation CPSs is used for controlling harsh environments that can be harmful for humans. Nevertheless, these can also be harmful for system components. Frequency and duration of SOMs can contribute to estimate the remaining life of components working under such conditions. However, the wearing effect of the environment over electronic devices can lead to sudden failures that cannot be predicted. Likewise, incomplete datasets caused by system malfunctions can prevent the implementation of SOM-based forecasting, due to the lack of historical samples. This situation makes it required the development of new maintenance principles that allows reducing system breakdowns in harsh environments, or which contributes to conduct reliable estimations of their life.

The SOM concept presented in this work plays an important role in failure analysis and forecasting. Its main value lies in its capability of tracing the failure forming process, based on the variations of SOM transitions. SOM transitions convey not only information about the system operation, but they also contain information about forming failures, which can be used for failure diagnosis and forecasting and for understanding the root-cause of failures. Our main contribution is the understanding of the role of frequency and duration of SOMs as failure indicator. Our explorative research suggests that both indicators can be used as data features for failure diagnosis and failure forecasting. We claim that these failure indicators have the potential to support the

implementation of failure forecasting for first generation CPSs. Forecasting failure types and time to failure can facilitate the implementation of preventive maintenance in CPSs.

7.2 Propositions

In line with the main objectives of this PhD project, and the observed results, the following propositions were formulated. They represent the main findings and conclusions of my PhD research project.

System operation modes-based signal segmentation enhances the information content of failure indicators and provides revealing failure symptoms.

Failures cause deviations on certain systems parameters with respect to the failure-free operation. However, some of these deviations are not observable and detectable during system operation. Some failures can only be observed on specific parameters of the system in particular system operation modes, while others propagate through the entire system. For example, a leak can hardly be identified by the water level sensor when the tank is being filled or when water is taken out from the tank. However, leak symptoms are more likely to be observed when none of the valves (the inlet or outlet valves) are opened. Our assumption was that statistical variance of signal characteristics in a specific operation mode is smaller than in the overall system operation. Our experiments demonstrated that SOM-based segmentation reveals symptoms that cannot be observed in unsegmented signals. It implies an improvement on failure detectability in systems that present multiple system operation modes. A further implementation of this finding allows overcoming the masking effect that control regime exert over failure symptoms in 1G-CPSs contributing to increase the reliability of these types of systems.

Changes in the frequency and duration of system operation modes caused by a self-tuning system have a discriminative power in terms of failure classification.

The control objective of feedback control systems is to make the controlled parameter converge to a predefined set-point in order to meet the stability property. Whenever there are uncertainties or failures that deviates the system from the set-point value, the system tries to compensate them by manipulating system's actuators. It leads to variation in the frequency and duration of SOMs that tends to manifest itself in similar manner and form every time this failure mode occurs. It provides discriminant power to the frequency and duration of SOMs, enabling their use for failure classification. This information is valuable for supporting the decision-making process related to the selection of maintenance actions.

The SOMs-based approach makes failures observable through system operation mode transitions, rather than through signal deviations.

Feedback-based control systems enable the system to keep operating despite faulty conditions, as long as the system remains in the controllable zone. This behavior, however, hides failure symptoms and hinders failure detection. This situation makes unreliable those failure diagnosis techniques that are underpinned on the analysis of output signals only. Nevertheless, the increase on control activity exerted to compensate failure effect is an undeniable failure indicator. It was manifested in our experiments through variations on the SOM transitions that led to changes on the frequency and

duration of SOMs. Our results revealed that the changing frequency and duration of SOMs provides more reliable results than analyzing signals deviations, allowing overcoming the limitations that traditional methods convey. It implies a change of paradigm in failure analysis, as it entails replacing the role of system signals as failure information carriers by the SOM transitions.

Changes in the frequency and duration of system operation modes show a strong correlation with the trends of system degradation.

In our experiments, we found that system degradation is represented by a unique combination of trends of SOM frequency and duration for each failure mode. These trends typically occur in SOMs, in which actuators are directly or indirectly impacted by the failure mode as a result of the feedback control. The extrapolation of trends demonstrated to be a suitable mean for predicting the forming failure time and to estimate the expected time to failure. It represents an important potential for failure forecasting in IG-CPSs, as it allows planning maintenance actions in advance, avoiding the failure to reach critical stages, and enabling a decision-making process based on the analysis of factors that influence a cost-effective maintenance.

The frequency and duration of SOMs represent data features that can be applied to multiple failure modes indistinctively.

Currently, data-driven based failure analysis requires determining the most suitable data feature per failure mode. This process strongly depends on the experts' experience, or on trial-error tests that can be time-consuming and do not guarantee finding a useful feature. Frequency and duration of SOMs can be utilized as data features, indistinctively of the occurring failure mode. The reason is, these are based on the analysis of the variations of system dynamics (i.e. SOM transitions) caused by failure occurrence, instead of on particular signal characteristics that strongly depend on the nature of the sensed parameter. It contributes to reduce the time invested on deriving classification models and to reduce the dependence on human experts.

Unlike most of the traditional failure analysis approaches, system operation modes-based exploration and forecasting allows dealing with and understanding failures at system level.

Traditional failure analysis methods are typically component oriented. They implement specialized models that only focus on analysis of signal features on component level. These methods provide support for diagnosing status of specific components, but they do not provide insight for analyzing symptoms that arise from the interaction between components. Unlike traditional methods, SOM-based failure analysis approaches CPSs as systems of systems. SOMs are time dependent variables of combination of component operation modes that capture information about the parallel states of system components and their time variant transitions. Changes on the state of any component trigger SOM transitions. It allows evaluating the influence of components failures on the whole system performance, enabling the evaluation of cascade effect of failures, and the understanding of the failure evolution.

SOMs facilitate the diagnosis, forecasting and root cause analysis of failures.

Changes of the frequency and duration of SOMs reflect failure occurrence. Combined analysis of these failure indicators for all SOMs facilitates diagnosis of failure modes

that are effecting system operation. Trends of the changes of SOM frequency and duration provide information about the time to failure that can be used for failure forecasting and predicting the forming failure mode. Analysis of SOM transitions provides information about the root cause of failures utilizing information generated as compensation of failure effect by the feedback control of the system. This situation makes of SOM-based failure analysis superior to the existing methods that only provide hints either for detecting currently occurring failures, evaluating failure occurrence in retrospective, or estimating a further failure appearance.

7.3 Reflections on the completed research

This PhD project explored the role of system operation modes for its potential application in failure diagnosis and forecasting in first generation cyber-physical systems. Case studies presented in this dissertation were relying on datasets collected from a Simulink model of a kettle and a testbed of a greenhouse. They were used as research instruments for studying failure indicators concepts based on SOM based signal segmentation and change of SOM frequency and duration in controlled experiments. Results obtained demonstrated that SOMs reflect the failure forming process through the long-term trends observed on the frequency and duration of SOMs, bringing out new opportunities for failure forecasting. In this section, we will reflect on the process conducted and the results obtained, with regards to evaluate their influence on our findings.

7.3.1 Reflections about the process

In the first research cycle, the conducted research was methodologically framed as research in design context. The context of inquiry was learning the state of the knowledge and practice in failure management. In this cycle, we evaluated the limitations and opportunities that the available failure analysis methods imply, also with a view to the consideration of SOMs. For this purpose, we conducted a state of the art that was mostly based on the analysis of scientific literature, industrial reports and professional videos that described the existing methods. This analysis revealed there are plenty of methods concentrated on failure analysis, which can be categorized as signal-based, model-based or data-driven techniques. There are also hybrid approaches that combine methods from the mentioned categories to overcome their limitations.

The investigation of literature gave us a sight on the existing techniques and their operationalization. Nevertheless, it does not suffice to provide a deep understanding of their limitations and implications. Only experimentation provides the criteria required for judging the existing techniques. However, the limited time and limited resources that a PhD project imply prevented the execution of practical tests during the exploration phase that would allow evaluating some of the existing methods. The analysis of documented reviews is useful for knowing the limitations and implications that existing methods entail. However, we found these are mostly focused on the most popular approaches and do not manage to cover the different variations enabled by the hybrid techniques. It forced us to underpin our argumentation about the less common techniques on the scarce information available on literature.

The second research cycle was conducted based on research in design context. In this cycle, we aimed at analyzing the influential factors on the phenomenon of failure analysis, by determining the implications of SOMs on the self-regulation and self-tuning properties of CPSs. This process was conducted based on the review of scientific articles. Nevertheless, few papers that analyze in detail the implications of self-regulation and self-tuning were found. Existing literature approaches this subject from a theoretical perspective, but there is a lack of practical implementations that discusses the embodiment of both concepts in terms of design features. This situation hindered the definition of functions and technical requirements for the instrumentation of the testbed, and forced us to modify the settings of the system several times during the pilot tests, delaying the execution of the experiments.

The third research cycle implemented design inclusive research with regards to analyze the effect of SOMs on failure symptoms. The complementary approach of simulation and practical experimentation conducted resulted to be very useful. The simulated kettle enabled controlled conditions that allowed getting important insights that could not be obtained through the testbed, due to the effect of external disturbances. The testbed, in turn, enabled experimentation with the studied phenomenon through a real system whose operation involved real disturbances and non-controllable factors. Nevertheless, more diversity regarding the type of systems considered could have provided more validity to our experiments. The kettle model and the implemented testbed presented important similarities between them in terms of system features (such as components and type of control) that could have led to certain bias on the obtained conclusions.

Research cycle 4 presented a similar situation than research cycle 3. There, failure evolution was analyzed, based on experimentation conducted through the computational model and the testbed. As for research cycle 3, the complementary approach facilitated the analysis of factors that could not be explored in the practical experimentation with the testbed. For instance, only the evolution of the tank leak was evaluated in the testbed, due to the time-consuming process that injecting progressive failures implied. Nevertheless, the conducted simulations allowed overcoming this situation by evaluating the progress of multiple failure modes in the kettle model. The obtained conclusions provided relevant hints for an explorative research as this PhD project. However, such conclusions cannot be generalized to systems that present different characteristics than the ones described by the kettle model and the greenhouse.

Finally, research cycle 5 conducted practice-driven research with the aim to evaluate the challenges and opportunities of using SOMs in the context of preventive maintenance. Nevertheless, we found there is a lack of maintenance principles specifically oriented to CPSs. It forced us to analyze to what extent maintenance principles belonging to zero generation CPSs can be transferred to 1G-CPSs, and what is the role of system operation modes in this process. This analysis is based on a critical reasoning that triangulates the characteristics of the existing maintenance principles, the characteristics of 1G-CPSs and the characteristics of SOMs. Nevertheless, the real implementation of such principles in 1G-CPSs can differ to certain extent from the foreseeing results.

7.3.2 Reflections about the experiments and the results

In the experiments, multiple variations of failure evolution were tested. The trend direction remained unchanged, but the trend slope presented changes. It did not affect

the prognosis and diagnosis results, though. We consider that the availability of a good data sample per failure mode contributed to the positive results obtained. It included different representative values of SOM frequency and duration. However, in the real system operation failure data is scarce and meeting the minimum required for training the classification algorithm can be an important issue. The way in which we conducted demonstrative application is highly sensitive to the size and quality of training dataset. We claim that this situation is critical in a real-life application as there are failures that do not occur very often. Data coming from equivalent systems, failure simulation, and failure injection could be used for providing the missing data. Retraining processes should also be conducted every time a new failure is reported.

One of the limitations of our study is the limited amount of data features considered in the SOM-based signal segmentation analysis. As our study focused on time dependent evolution of failures, we selected statistical data features primarily applied in the time-domain analysis of signals. We considered the most frequently used statistical features in our study, i.e.: derivative, median, mean, area, standard deviation, skewness, and kurtosis, but only the first five features were used in our experiment. Skewness and kurtosis were not used as they did not show any significant variation in any of tested cases. However, features belonging to the frequency and time-frequency domain can also present failure symptoms per signal segment. Further studies should evaluate data features corresponding to frequency and time-frequency domains, their role in representing failure symptoms, and their sensitivity to disturbances caused by internal and external factors.

The actuators implemented in the kettle model and the testbed only had on/off operation modes. The control actions were determined by the activation or deactivation of the actuator components. However, some actuators in different applications of CPSs are analogous. They are typically controlled by setting their activation level to a set of predefined finite states by the system control. These component activation levels can be used for defining discrete set of component operation modes similar to the approach we have presented in this thesis. The number of component operation modes would increase the possible system operation modes giving more insight into system behavior, and providing potentially more reliable failure diagnosis and forecasting using failure indicators, but also increasing the computational complexity. Besides the large set of finite states of components, the complexity of CPSs is also determining factor of computational complexity. We recommend tackling these research challenges by using the approach of system of systems. Methods like hierarchical system decomposition and parallel computing failure indicators for each sub-system can reduce the computational complexity and increase computation performance. The obtained results could be reported to a master system which manages the control of the failure diagnosis and forecasting processes.

7.4 Further research

In this section, we will discuss the research challenges that were associated with the work and results reported in this dissertation. Tackling these problems would lead to results that complement our research and would provide additional knowledge for a complete implementation of the SOM based failure diagnosis and forecasting. The

analysis presented in this section will consider short-term, and long-term challenges. Short-term challenges are all those aspects required for completing the conducted research. Long-term challenges, in turn, comprises complementary research that is triggered by the findings obtained in this PhD project. In the short-term, two main challenges are identified, namely: (i) validate the implementation of SOM based failure diagnosis and forecasting methods in complex systems, and (ii) to analyze data features in the frequency-domain. In the long term, we consider important to explore the sensitivity of SOM based failure diagnosis and forecasting to multi-state actuators, as this last aspect was not covered in the present research.

7.4.1 Short term challenges

The research reported in this thesis has an explorative nature. It provided descriptive knowledge about the role of system operation mode in failure diagnosis and forecasting, but it did not validate its implementation. The insights obtained on the role of SOM frequency and SOM duration in defining failure indicators should still be validated in different domains of CPS applications as well as with second and third generation CPSs (e.g. self-driving car). Moreover, sensitivity of SOM failure indicator concepts to the number of system components and number of component operation modes should also be evaluated in order to determine their limits. These validation processes should be exhaustive and should consider the following recommendations.

1. minimum number of data samples for each operation mode
2. representation of external factors in data set
3. treatment for missing, incomplete, noisy data
4. data set with failure induced operation modes
5. Guidelines for preparing reference data for non-failed cases

A minimum number of data samples for each SOM is required in order to capture their representative behavior during failed and failure-free operation. It allows avoiding the bias caused by the particular conditions in which data was collected, enabling failure symptoms recognition despite external disturbances. Considering that a further implementation in a real system is inevitably subjected to noise, data collected with validation purposes should come from multiple real-life applications where data is influenced by external factors. It allows evaluating in what types of application domains SOM based failure analysis manage to forecast progressive failures.

The quality of the obtained data is critical for providing reliable results. The validation of SOM based failure analysis should assure a proper dataset despite communication drops or corrupted sensor measurements. For this purpose, a suitable experiment setup and the implementation of data filters is needed in order to obtain reliable results. An equivalent number of data samples for all the induced failure modes (and failure-free data) is also critical. It contributes to prevent overfitting, and it increases the discriminant power of the SOM based failure analysis, as it counts with more information for capturing the representative characteristics of every failure mode.

All the aforementioned recommendations are paramount for the collection of the reference data (data used for deriving classification and forecasting models).

Considering reference data determines the accuracy of the derived models, a suitable training dataset is required. For this reason, it is recommended identifying the different working scenarios of the system (determined by external conditions, user manipulation patterns, among others) and executing multiple experiment in all the identified scenarios. It contributes to determine the way in which SOMs are affected by multiple and different external conditions and evaluating the properness of the frequency and duration of SOMs in changing operative contexts.

Besides of the validation of the SOM based failure diagnosis, it is also required analyzing our experimental data in the frequency-domain. So far, we studied the most relevant data features of the time-domain statistical signal analysis in Chapter 4. These features managed to unveil failure symptoms when analyzed through SOM-based signal segmentation. However, it is recommended to expand this research to data features of frequency and the time-frequency domain, as these are widely used in literature with failure detection purposes. Future research should investigate to what extent those features support identification of failure symptoms in SOM-based segmented signals. This process will also require a validation process, that enables determining their sensitivity to external and internal disturbances, as well as their effectivity in multiple CPS application domains.

Validation of our explorative results, as well as exploration of data features from the frequency and time-frequency domains is thus required. We claim that it will contribute to form a robust and reliable knowledge that could enable its further implementation in commercial systems.

7.4.2 Long term challenges

So far, our experiments only analyzed failures by considering two states actuators. Further research should investigate the sensitivity of SOM based failure diagnosis and forecasting to multi-state actuators. In these types of components, the controller keeps stability by increasing or decreasing the power of the actuator, enabling more than two potential component operation modes. We claim that this situation affects the way in which failures are manifested on the frequency and duration of SOMs, leading to new types of failure manifestations. Considering the above-mentioned situation, this future research should answer the following research question:

How failure manifestations on the frequency and duration of SOMs is affected by control actions that are operationalized through variations on the operative intensity of system actuators?

This future project should analyze sensed system data, in order to identify data patterns that can be associated to the injected failures in every SOM. This analysis should fulfill the recommendations previously presented on 7.4.1, so that, a suitable reference dataset can be obtained. This research should be followed by a validation process (as the one proposed on 7.4.1) in order to determine the limitations on the application of the derived knowledge.

Another long-term purpose is the implementation of the validated knowledge about SOM frequency and duration in commercial systems. The research reported in this manuscript aimed to explore the fundamental issues applying system operation modes in failure diagnosis and forecasting. Demonstrative cases illustrated the potentials and

limitations of SOM. Operationalization of these concepts, however, should be fully developed before they can be applied in the industrial practice. It raises the need of designing and developing a monitoring system that implements the frequency and duration of SOMs as indicators for failure forecasting. However, this design task requires the definition of important aspects such as: (i) the development of optimal algorithms for data filtering, classification and forecasting, (ii) the definition of the facilities and infrastructure required for the operationalization of the SOM-based failure forecasting concept, (iii) determining policies and protocols related to data storage and information delivery, among others. The design and development of such system can only be conducted once all the validation processes are completed.

Summary

Due to faults and failures, systems work below their normal production capacities or qualities, with frequent and increased downtimes, and with a reduced trust and dependability. These situations often cause significant economic losses and delays. Nowadays, first generation cyber-physical systems are often used in production and supply processes. Supported by an entry level system intelligence, these systems are typically equipped with compensatory control capabilities that make it difficult to recognize failures at their very moment of appearance or in an early stage of proliferation. This has been recognized as a new challenge in the case of mission critical systems since it cannot be addressed by the traditional failure analysis techniques.

The growing self-intelligence and self-adaptation capabilities that characterize CPSs leads to systems that efficiently compensate for early-phase faults and slight failures on their own, by purposefully shifting system operation modes (SOMs). This system behavior makes a satisfactory performance possible in spite of variations in the internal states and contextual conditions, making first generation CPSs (1G-CPSs) self-tuning systems. Implementation of a suitable failure diagnosis and forecasting in 1G-CPSs requires a proper understanding of the phenomenon of shifting SOMs. The transitions self-enabled by the system bring out uncertainty and short periods of instability of the system, hampering the use of the currently available failure analysis methods, which have demonstrated to be sensitive to uncertainty and variations in system behavior.

This promotion research aimed at studying the role of shifting SOMs in failure analysis of 1G-CPSs, as a technically feasible first manifestation of CPSs. Our previous studies revealed that this topic had not been studied yet in the context of forecasting failure analysis, especially not in the context of first generation CPSs. This situation raised the need for generating a body of knowledge that supports the development of new failure analysis and failure management solutions for these new types of systems. Considering the above-mentioned situation, the phenomenon to be studied in this promotion research was formulated as follows:

There is a knowledge gap that requires investigation of the roles played by system-initiated compensatory actions and operational mode changes in the emergence and proliferation of technical failures in the case of self-regulatory and self-tuning cyber-physical systems.

The promotion research included an explorative research part, for the reason that practical experimentation in a real-life context was deemed necessary for deriving reliable insights. For this purpose, we developed a cyber-physical greenhouse testbed that mimicked the operation of 1G-CPSs, as these types of application systems are strongly influenced by external variables such as light intensity, ambient temperature, and relative humidity. Controlled failure injection was chosen as the most suitable way

of analyzing failures of the testbed system, as it was difficult to find proper data sets about the operation of an existing system under a specific failure mode.

In order to study the effect of SOMs over the failure manifestations on system signals, we proposed the concept of ‘system-level failure indicator’ (FI). This concept allowed analyzing the statistical deviation of system signals during the different SOMs based on two main principles: (i) failures are manifested in system’s signals through deviations from the regular system’s behavior, and (ii) the signals generated vary in their parameter values depending on systems operation modes. The proposed FI was operationalized by segmenting system signals every time there was a transition from one particular SOM to another one. Such segments were represented and analyzed through statistical data features (derivative, standard deviation, mean, area and median), and compared with a set of reference segments corresponding to the same SOM and failure-free operation. The Kruskal-Wallis technique was used in order to determine the statistical difference between the abovementioned datasets. Whenever there was a statistically significant difference, it was interpreted as a potential failure symptom.

The proposed signal segmentation was introduced based on the assumption that statistical variance of signal characteristics in a specific operation mode was smaller than in the overall system operation. We argued that segmentation strengthened failure symptoms and allowed overcoming the masking effect that system control exerted over failure symptoms. The main assumption behind this failure indicator concept was that failures symptoms were manifested in systems signals differently depending on system’s operation modes and that both the symptom occurrence and the lack of symptom could be used as indicator for determining the type of failure.

For our analysis, we considered and introduced failures both in a computational model of a kettle (as a simple application case) and in the instrumented testbed (as a complicated application case). The kettle model was used to evaluate the proposed concept by providing controlled experimental conditions. In contrast, the instrumented testbed was used to provide real life conditions by considering the variance caused by external factors (such as environmental factors and use conditions). The injected failures were: (i) tank leak, (ii) obstruction in system valves, (iii) irregular operation of the inlet fan, and (iv) loss of heating power. These failures were selected as they involved both SOMs and system signals.

The obtained results revealed that signal segmentation had a positive effect on failure detection. It enabled the observation of symptoms that could not be observed at analyzing the unsegmented signals. However, the derived indicators had a weak discriminant power, making it difficult to distinguish different failure modes based on the observed pattern of symptoms. The derived (proposed) failure indicators also proved to be sensitive to external disturbances (hindering the discrimination between failure effect and environmental effects over the system signals), and presented troubles for dealing with short lasting SOMs. These limitations hindered the use of traditional failure indicators for failure diagnosis. However, the proposed FI also revealed an important finding, namely: some SOMs were triggered by the emerging failures, pushing the system into an “abnormal” operation mode (i.e. a combination of component operation modes not typical under regular circumstances). It caused that certain SOMs, which were supposed to occur during failure-free operation, stopped occurring due to failure occurrence, while some others, which did not occur under normal system operation,

started occurring. These new SOMs were named Failure Induced Operation Modes (FIOM) and their importance relied on the fact that they manifested a change on the regular regime of operation of the system due to failure. This change was observed through variations in both the frequency and the duration of SOMs that served as a means for compensating failure effects. This finding implied an important turning point in our research, as it envisioned an opportunity for analyzing the failure forming process and for conducting failure forecasting.

A deeper analysis was conducted in order to analyze if the frequency and the duration of SOMs could be used for characterizing the failure-forming process and for discriminating failures. For this purpose, we systematically increased the failure intensity (failure level) during our experimentation, with the intention to simulate a hypothesized failure forming and proliferation process. Time series composed by historical measurements of Fq and D , collected as the failure progressed during a time interval, were used as the basis for extrapolating their observed trends. Extrapolated data were used as predictors for training a Linear Discriminant Analysis (LDA) model to determine the forming failure mode and the time to failure (TTF). The kettle model and the instrumented testbed were used as means for experimentation. Progressive failures were injected and Fq and D data were obtained and analyzed. The results of the repeated experiments revealed that system degradation and failure evolution are manifested through variations on the frequency and the duration of SOMs, as effect of the compensatory actions carried out by self-tuning systems. It allowed investigating the failure forming process in run-time, as well as facilitating the decision-making process about maintenance actions. It also revealed that every failure mode presented a particular pattern composed by the trends of the frequency and the duration of all SOMs that could be used for failure characterization. Results concerning failure classification (by using Fq and D as predictors) were satisfactory, so as we managed to predict the occurring failures properly. This led us to conclude that the changing frequency and duration of SOMs provide a high discriminant power. It makes their use possible for the purpose of failure diagnosis.

Failure forecasting based on Fq and D was also extensively studied and evaluated. For this purpose, historical values of frequency and duration of SOMs were collected and used as input for an exponential smoothing algorithm (ETS). It enabled projecting the observed trends of Fq and D into the future, delivering their forecasted values. Such values were used, in turn, as input for the previously derived LDA model, enabling determining the occurring failure mode from earlier failure forming stages. Likewise, the time to failure (TTF) was also estimated by identifying the very first time, in the forecasting horizon, in which the forecasted values were not classified as failure-free.

The results obtained concerning the applied failure forecasting approach indicated that the change of the frequency and the duration of SOMs was an effective indicator of the failure forming process, enabling failure forecasting in 1G-CPSs. These indicators could also be used as data features for failure analysis, decreasing dependency on expert's knowledge, and allowing its application for multiple failure modes indistinctively. Concerning the possible extension of the obtained findings to the context of preventive maintenance, we found that the analysis of the changing frequency and duration of SOMs offered new opportunities for the implementation of novel maintenance principles in the context of 1G-CPSs. Elaboration on the SOMs enabled: (i) analyzing failures at system level, (ii) overcoming the failure masking effect exerted by system

control, (iii) analyzing the failure forming process, and (iv) decision making based on data measured in run-time. This was important since there are some other critical operational characteristics of 1G-CPS that are not covered by the currently existing maintenance principles. In this context the reader should think of: (i) vulnerability to external attacks, (ii) intensive operation in harsh environments, and (iii) autonomous initialization and execution of maintenance actions. The development of such sophisticated maintenance principles, as well as the development of computational means for the operationalization of these principles, is the next step towards a fully-fledged preventive maintenance and more reliable CPSs.

Further research is required in order to validate the results presented in this report with different domains of CPS applications. It is also recommended to evaluate the sensitivity of SOM-based failure diagnosis and forecasting to multi-state actuators. It is important to emphasize that this PhD project only focused on two-state actuators. Consequently, there is still a need to evaluate the effect of more extensive compensatory actions (e.g. adaptation, evolution) that imply changing the configuration and operational intensity of system actuators. The sensitivity of SOM failure indicator concepts to the number of system components and number of component operation modes should also be evaluated in order to determine their limits.

Samenvatting

Door storingen en faalvormen kan de productiecapaciteit of -kwaliteit van systemen achteruitgaan, waardoor vertragingen en extra kosten ontstaan, en men er niet meer op het systeem kan vertrouwen. Tegenwoordig worden in productie- en toeleveringsprocessen vaak cyberfysische systemen (CFS'en) van de eerste generatie (1G) ingezet. Die zijn vaak uitgerust met basale systeemintelligentie in de vorm van compenserende regelmechanismen die nauwelijks aanknopingspunten bieden voor het herkennen van storingen als ze ontstaan, of als de oorzaken zich net beginnen uit te breiden. Voor missie-kritische systemen wordt het realiseren van een dergelijk herkenningsvermogen gezien als een nieuwe uitdaging waarvoor traditionele faalanalysetechnieken geen oplossing bieden.

Het toenemende intelligentie- en adaptatievermogen dat kenmerkend is voor CFS'en leidt tot systemen die storingen voortijdig efficiënt en zelfstandig kunnen compenseren door doelgericht veranderingen van bedrijfsmodus te interpreteren. Door zulk systeemgedrag kunnen ze voldoende blijven presteren ondanks variaties in systeemtoestand en contextuele omstandigheden, waardoor 1G-CFS'en als zelfaanpassend kunnen worden beschouwd. Implementatie van een geschikte manier van faaldiagnose en -voorspelling in 1G-CFS'en vereist een goed begrip van het fenomeen 'veranderende bedrijfsmodus'. De overgangen die het systeem zichzelf toestaat kunnen meerduidigheid en korte perioden van systeeminstabiliteit veroorzaken, waardoor de huidige faalanalysemethoden, waarvan bekend is dat ze gevoelig zijn voor meerduidigheid en variaties in systeemgedrag, tekortschieten.

Dit promotieonderzoek richtte zich op het bestuderen van de rol van veranderingen in bedrijfsmodus bij faalanalyse van 1G-CFS'en als een technisch haalbare eerste manifestatie van CFS'en. Uit ons voorafgaande onderzoek bleek dat in de context van voorspellende faalanalyse dit onderwerp, zeker voor 1G-CFS'en, nog niet was bestudeerd. Er moest dus kennis worden gegenereerd die de ontwikkeling van nieuwe faalanalyse- en faalmanagementoplossingen bij zulke systemen kan ondersteunen. Daarom werd het te voor dit promotieonderzoek bestuderen fenomeen geformuleerd als:

Er is een kennishiaat dat onderzoek nodig maakt naar de rollen van systeemgeïnitieerde compenserende acties en bedrijfsmodusveranderingen bij het zich voordoende en uitbreiden van technische storingen in zichzelf regulerende en aanpassende cyberfysische systemen.

Omdat voor het afleiden van degelijk inzicht praktische experimentatie in een levensechte context noodzakelijk werd geacht, behelste het promotieonderzoek een verkennend onderzoeksdeel. Daartoe ontwikkelden we als proef een cyberfysische kas om de werking van een 1G-CFS na te bootsen – een type toepassing waarbij sprake is van sterke invloed van externe variabelen zoals lichtintensiteit, omgevingstemperatuur en relatieve luchtvochtigheid. Omdat het moeilijk was om geschikte datasets te vinden m.b.t. de

werking van een bestaand systeem bij een gegeven faalvorm, werd als geschiktste vertrekpunt voor faalanalyse van de proefkas gekozen voor gecontroleerde foutinjectie.

Om het effect van bedrijfsmodi op het tot uiting komen van storingen in systeemsignalen te onderzoeken, voerden we het begrip ‘faalindicator op systeemniveau’ in. Gebruikmakend van dit begrip konden we statistische afwijkingen in systeemsignalen analyseren uitgaand van twee basisprincipes: (i) aankomende storingen uiten zich in de signalen als afwijkingen van het reguliere systeemgedrag en (ii) de gegenereerde signalen variëren in parameterwaarde afhankelijk van de bedrijfsmodi. De voorgestelde faalindicator werd verwezenlijkt door systeemsignalen in de tijd te segmenteren gebaseerd op overgangen tussen verschillende bedrijfsmodi. De resulterende segmenten zijn beschreven d.m.v. hun statistische datakenmerken (afgeleide, standaarddeviatie, gemiddelde, oppervlakte en mediaan) en als zodanig geanalyseerd en vergeleken met een reeks referentiesegmenten die overeenkwamen met dezelfde bedrijfsmodus tijdens storingsvrije werking. Om statistisch het verschil met de referentiedata te bepalen is de Kruskal-Wallistechniek gebruikt. Elk statistisch significante verschil is geïnterpreteerd als potentieel symptoom van een faalvorm.

De voorgestelde signaalsegmentatie komt voort uit de aanname dat statistische verschillen in signaalkenmerken voor een specifieke bedrijfsmodus altijd kleiner moeten zijn dan voor de algehele werking van het systeem. We beredeneerden dat segmentatie de faalsyndromen uitvergroot, en zo het effect dat de systeemregeling syndromen maskeert tenietdoet. De hoofdaanname hierbij was dat, afhankelijk van de bedrijfsmodi, faalsyndromen zich verschillend manifesteren in systeemsignalen, en dat zowel het wel als niet optreden van symptomen gebruikt kan worden als indicator om het type faalvorm te bepalen.

Voor onze analyse gebruikten we een zowel een computermodel van een waterkoker (als eenvoudige toepassing) als de al genoemde geïnstrumenteerde plantenkas (als gecompliceerde toepassing). Met het waterkokermodel is het voorgestelde concept geëvalueerd onder gecontroleerde experimentele condities. Daartegenover stond de geïnstrumenteerde proefkas waarin onder levensechte condities de variaties veroorzaakt door externe oorzaken zoals omgevingsfactoren en gebruiksomstandigheden in beschouwing zijn genomen. De volgende fouten werden hierin geïnjecteerd: (i) lekkende tank, (ii) verstopping van kleppen, (iii) fluctuerende werking van de inlaatventilator, (iv) uitval van verwarmingsvermogen. Deze fouten waren gekozen omdat ze zowel bedrijfsmodi als systeemsignalen beïnvloedden.

Uit de resultaten bleek dat de foutdetectie dankzij signaalsegmentatie verbeterde. Symptomen konden worden waargenomen die niet naar voren kwamen bij analyse van de ongesegmenteerde signalen. De afgeleide indicatoren toonden echter weinig discriminerend vermogen, waardoor het moeilijk werd om uit de waargenomen patronen in de symptomen de verschillende faalvormen af te leiden. Ook bleken de voorgestelde afgeleide faalindicatoren gevoelig voor externe verstoringen, wat het onderscheid tussen het faaleffect en omgevingseffecten vertroebelde, en ze bleken weinig effectief bij kortdurende bedrijfsmodi. Dit bemoeilijkte het gebruik voor diagnose. De voorgestelde faalindicatoren leidden ook tot de belangrijke constatering dat sommige bedrijfsmodi geactiveerd worden door plotseling optredende storingen, waardoor het systeem in een “abnormale bedrijfsmodus” terechtkwam, d.w.z. een combinatie van bedrijfsmodi van

componenten die onder normale omstandigheden niet voortkwamen. Daardoor traden na zulke storingen bepaalde bedrijfsmodi die te verwachten zijn bij storingsvrije werking niet meer op, en weer andere die normaal niet optreden, juist wel. Deze nieuwe bedrijfsmodi hebben we aangeduid als “bedrijfsmodi opgewekt door storingen”. Deze zijn van belang omdat ze blijken te geven van een verandering in het normale regelschema als gevolg van een storing. Deze verandering is waargenomen in de vorm van variaties in zowel de frequentie als de tijdsduur van specifiek die bedrijfsmodi die dienen om gevolgen van fouten te compenseren. Deze constatering was een belangrijk keerpunt in het onderzoek, omdat hierdoor de gelegenheid naar voren kwam om het ontstaansproces van storingen te analyseren en op basis daarvan storingen te voorspellen.

Verdere analyse moest duidelijk maken of de frequentie Fq en duur D van bedrijfsmodi gebruikt konden worden om het faalvormingsproces te karakteriseren en storingen te kunnen onderscheiden. Daartoe hebben we in de experimenten systematisch de intensiteit van de storingen verhoogd (storingsniveau) teneinde veronderstelde processen van faalvorming en -ontwikkeling te simuleren. Tijdreeksen samengesteld uit historische meetwaarden van Fq en D tijdens voortschrijdende faalontwikkeling over een tijdsinterval werden gebruikt om trends te extrapoleren. De geëxtrapoleerde data werd gebruikt als predictors om een lineair discriminantanalysemodel (LDA) te trainen dat in het waterkokermodel en de geïnstrumenteerde proefkas de ontstaande faalvorm en de *time to failure* (TTF) moest voorspellen. Voortschrijdende fouten werden geïnjecteerd om data voor Fq en D te verkrijgen en analyseren. Volgens de uitkomsten worden systeemdegradatie en faalontwikkeling zichtbaar doordat de frequentie en de duur van bedrijfsmodi variëren, als gevolg van compenserende activiteit in zelfaanpassende systemen. Daardoor konden we het faalvormingsproces in *runtime* onderzoeken en ook het beslisproces m.b.t. onderhoudsactiviteiten faciliteren. Ook bleek dat het overzicht met het verloop van frequentie en duur van alle bruikbare bedrijfsmodi kan worden gebruikt als een vingerafdruk waarmee elke specifieke faalvorm met voldoende significantie kan worden gekarakteriseerd. Dit leidde tot de conclusie dat de veranderingen in frequentie en duur van bedrijfsmodi een hoog onderscheidend vermogen hebben en dus gebruikt kunnen worden voor faaldiagnose.

Ook het voorspellen van falen op basis van Fd en D is uitgebreid bestudeerd en geëvalueerd. Daartoe werden historische waarden van de frequentie en duur van optredende bedrijfsmodi als input gebruikt voor een *exponential smoothing*-algoritme. Zo konden de waargenomen trends in Fd en D vooruit worden geprojecteerd om toekomstige waarden te voorspellen. Deze waarden werden op hun beurt weer ingevoerd in het reeds afgeleide LDA-model waardoor de optredende faalvorm op basis van voorafgaande ontstaansstadia kon worden bepaald. Evenzo kon de TTF worden afgeschat door binnen de prognosehorizon het eerste tijdstip te bepalen dat niet als storingsvrij geïdentificeerd is.

De met onze aanpak verkregen faalvoorspellingsresultaten lieten zien dat verandering in frequentie en duur van bedrijfsmodi effectieve indicatoren bieden voor het ontstaansproces van faalvormen en dat deze faalvoorspelling in 1G-CFS'en mogelijk maken. De indicatoren kunnen ook worden gebruikt als datakenmerken voor faalanalyse en zonder onderscheid worden toegepast voor verschillende faalvormen. Hierdoor is men minder afhankelijk van menselijke expertise. Wat betreft het mogelijke doortrekken van de bevindingen naar ondersteuning van preventief onderhoud hebben we vastgesteld dat analyse van veranderingen in frequentie en duur van bedrijfsmodi kansen biedt om tot

nieuwe onderhoudsprincipes voor 1G-CFS'en te komen. Beschouwing van de bedrijfsmodi maakte het mogelijk om (i) storingen op systeemniveau te analyseren, (ii) het maskeereffect veroorzaakt door de systeemregeling weg te nemen, (iii) het ontstaansproces van storingen te analyseren en (iv) besluitvorming gebaseerd op *runtime* data te ondersteunen. Dit is belangrijk omdat in bestaande richtlijnen voor onderhoud bepaalde kritieke bedrijfskarakteristieken van 1G-CFS'en buiten beschouwing blijven. Hierbij kan worden gedacht aan (i) kwetsbaarheid voor aanvallen van buiten, (ii) intensief bedrijf in barre omstandigheden, (iii) autonoom initiatief tot, en uitvoering van, onderhoudsactiviteiten. Ontwikkeling van zulke geavanceerde onderhoudsprincipes alsook van computerondersteuning voor operationalisering ervan vormt de volgende stap op weg naar volwaardig preventief onderhoud en naar betrouwbaardere CFS'en

Vervolgonderzoek is nodig om de hier gepresenteerde resultaten te valideren in andere CFS-toepassingsgebieden. Ook wordt aanbevolen om de gevoeligheid van bedrijfsmodi-gebaseerde storingsdiagnose en -voorspelling te evalueren voor actuatoren die meer dan twee toestanden kunnen aannemen. Ook is er nog steeds behoefte aan evaluatie van het effect van geavanceerdere vormen van compenserende systeemactiviteit die leiden tot veranderingen in de configuratie en bedrijfsintensiteit van systeemactuatoren, zoals bijvoorbeeld adaptatie of evolutie. Om het geldigheidsbereik van onze aanpak om bedrijfsmodi als storingsindicator te gebruiken vast te stellen, moet ook de gevoeligheid ervan voor het aantal systeemcomponenten en het aantal bedrijfsmodi per component worden geëvalueerd.

Appendix A

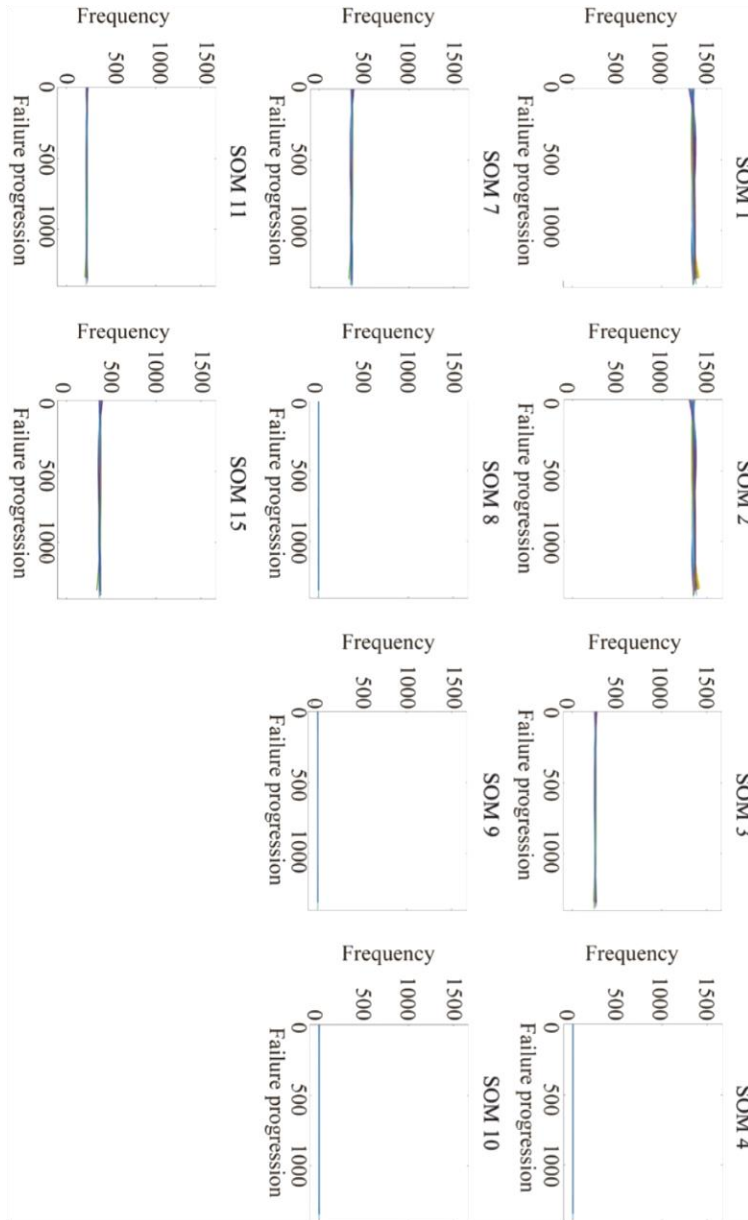


Figure A.1. Filtered trend of F_q for Failure-free scenarios of the simulated kettle

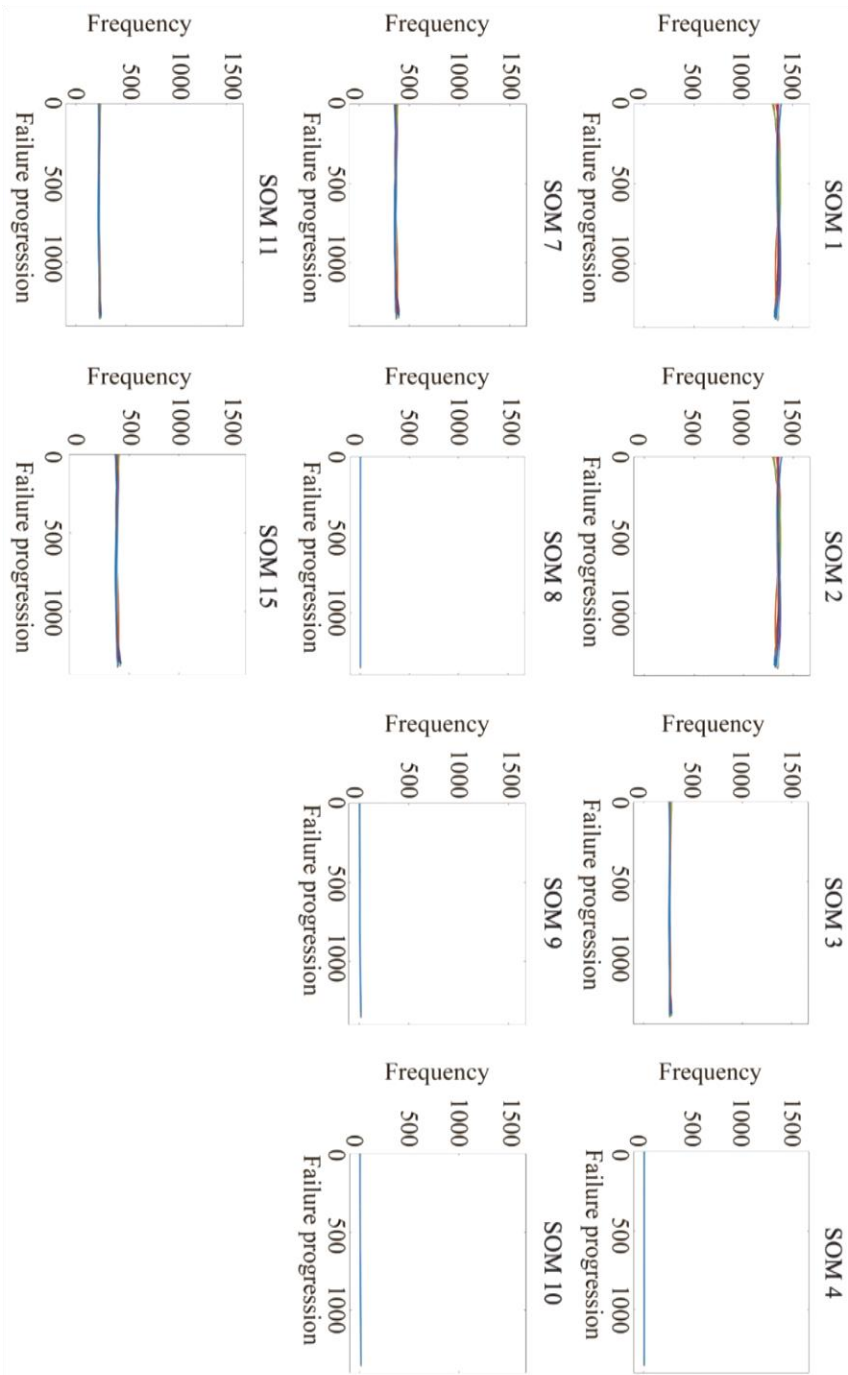


Figure A.2. Filtered trend of F_q for F_1 of the simulated kettle

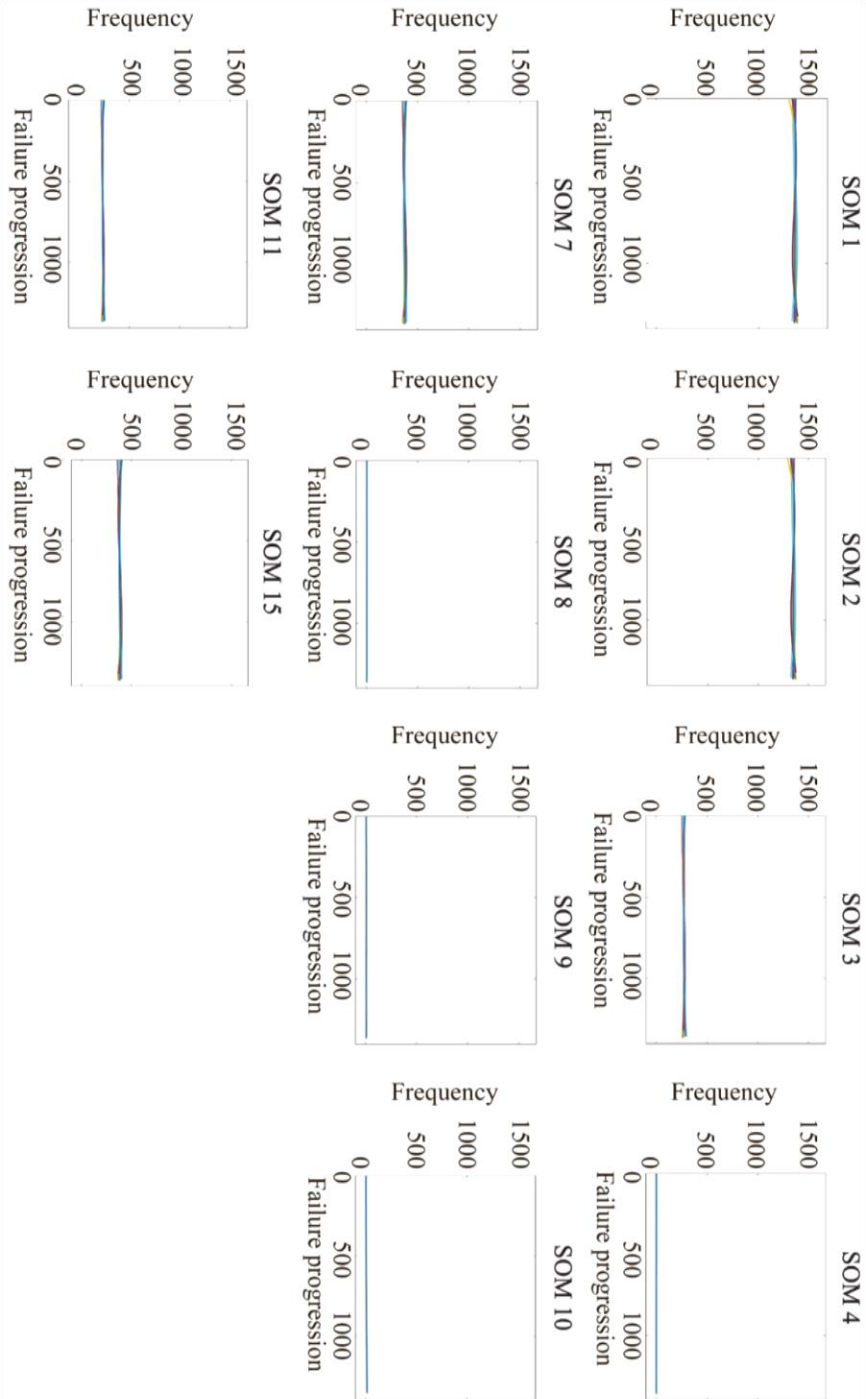


Figure A.3. Filtered trend of F_q for F_2 of the simulated kettle

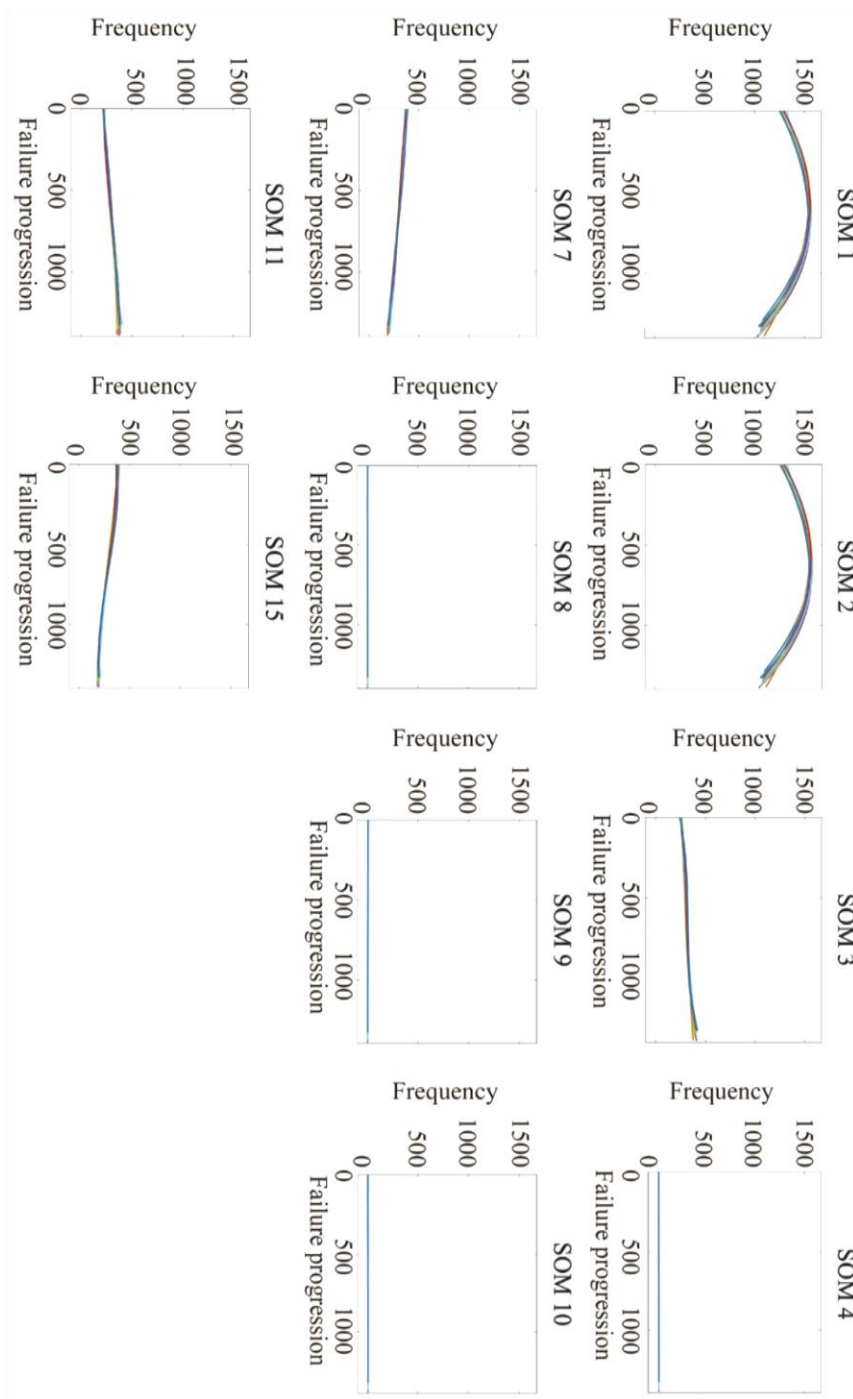


Figure A.4. Filtered trend of Fq for F3 of the simulated kettle

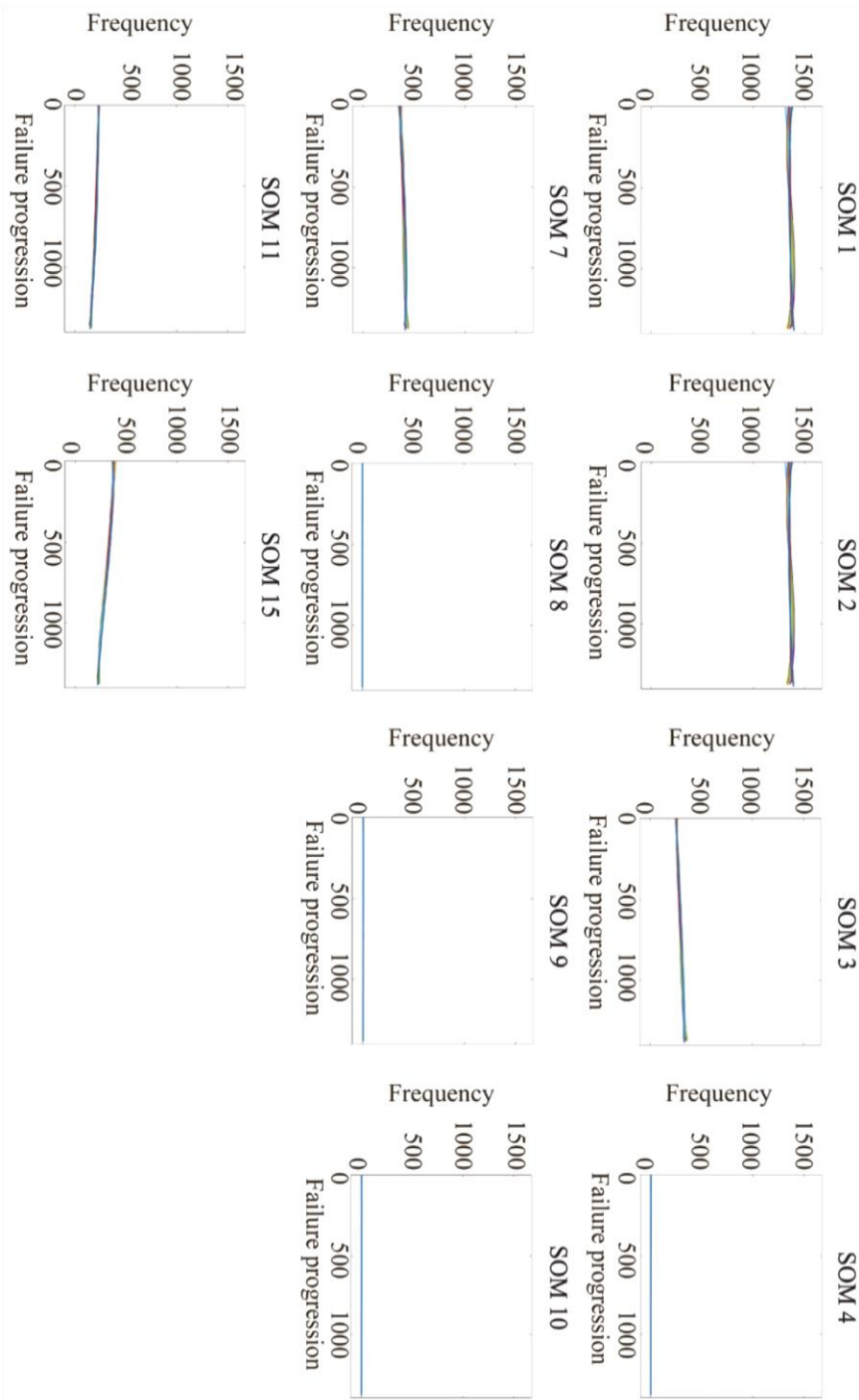


Figure A.5. Filtered trend of F_q for F_4 of the simulated kettle

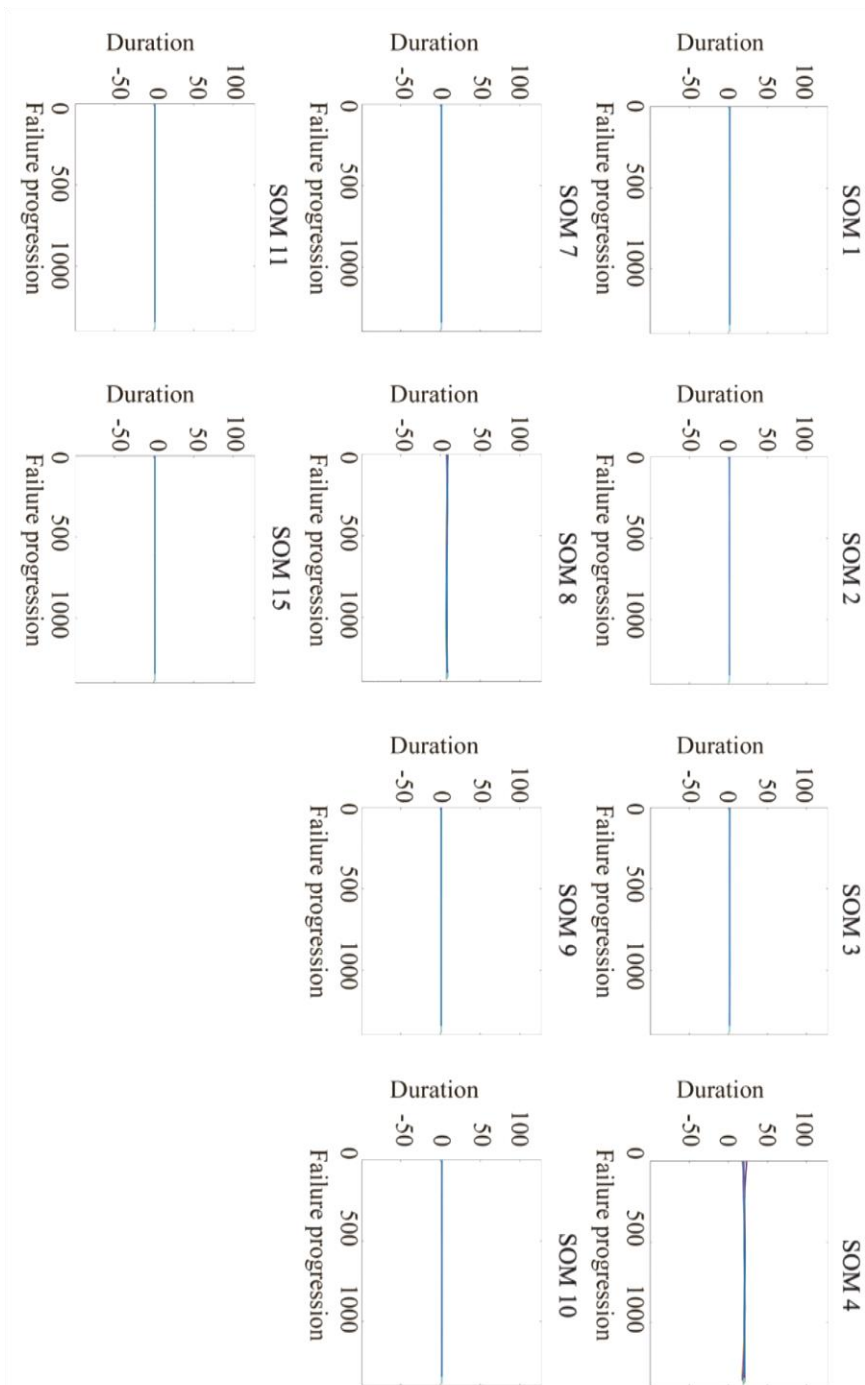


Figure A.6. Filtered trend of D for failure free scenarios of the simulated kettle

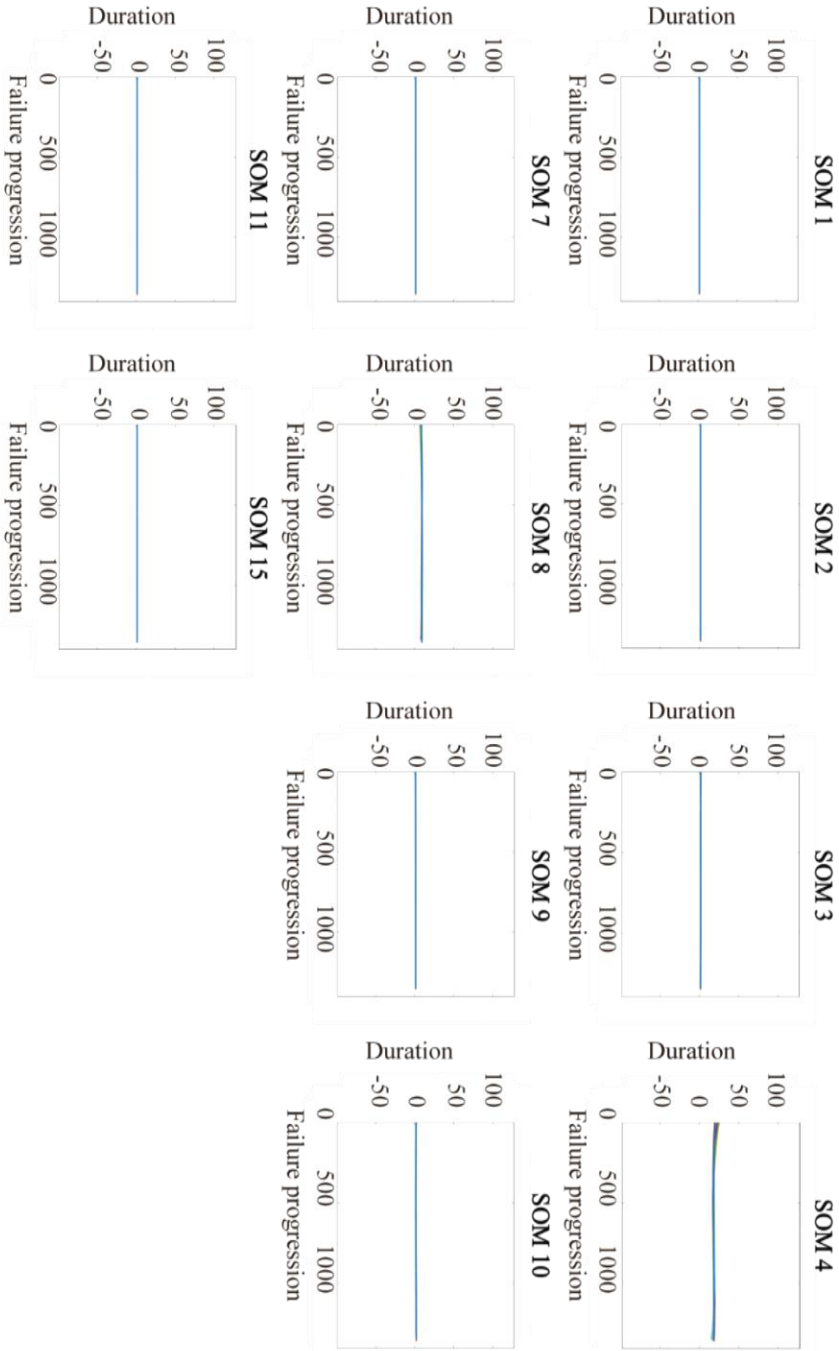


Figure A.7. Filtered trend of D for $F1$ of the kettle model

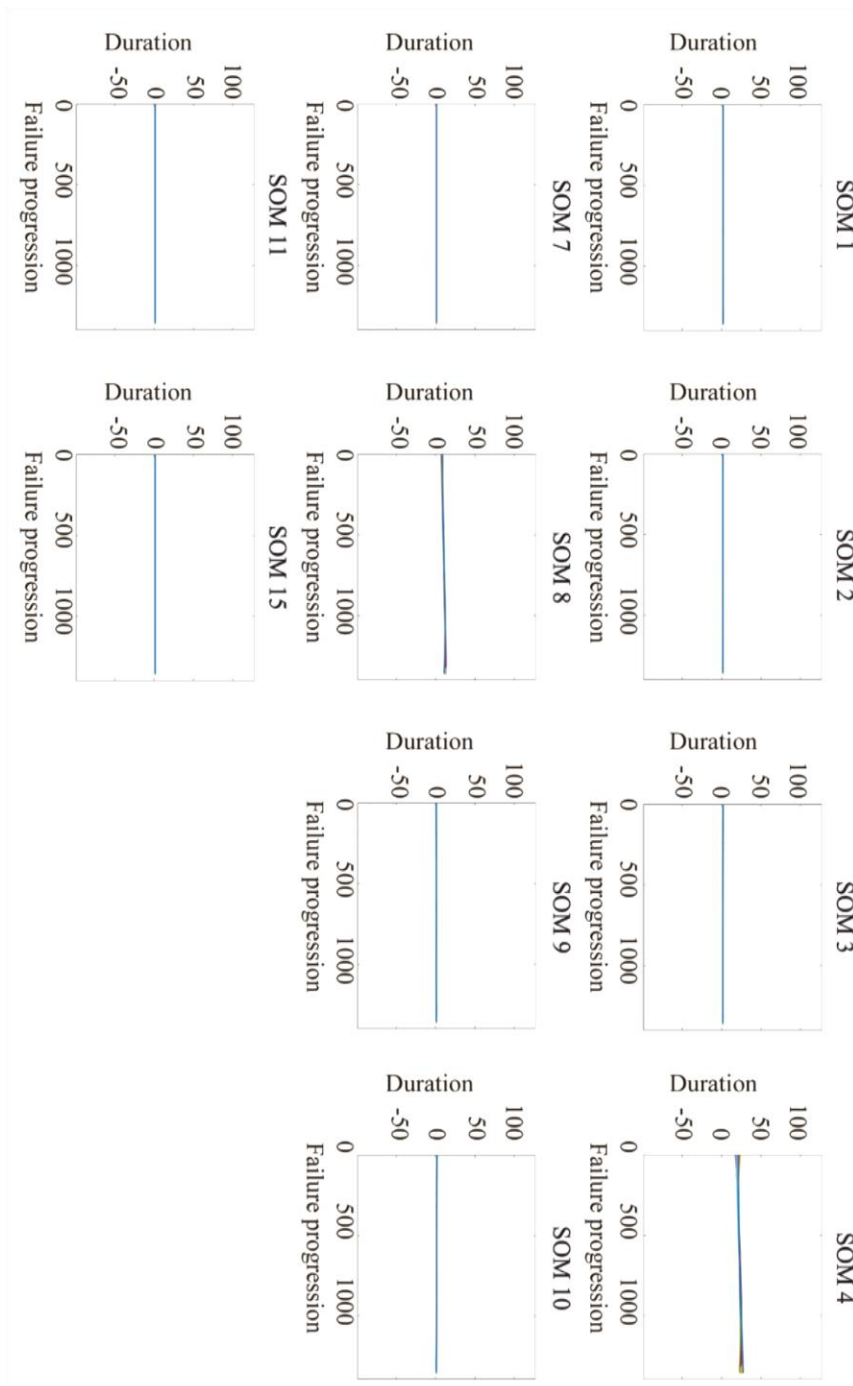


Figure A.8. Filtered trend of D for $F2$ of the kettle model

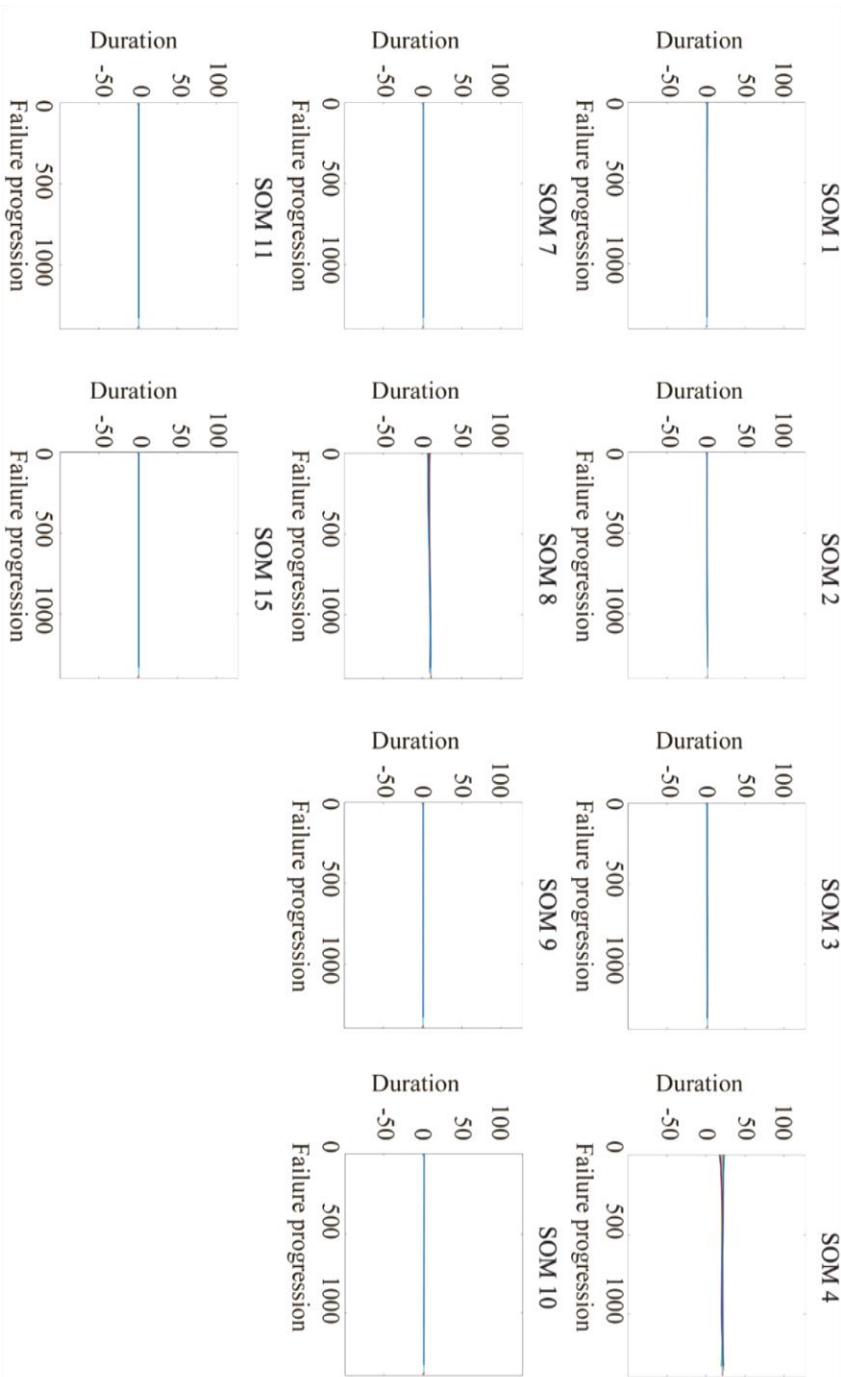


Figure A.9. Filtered trend of D for $F3$ of the kettle model

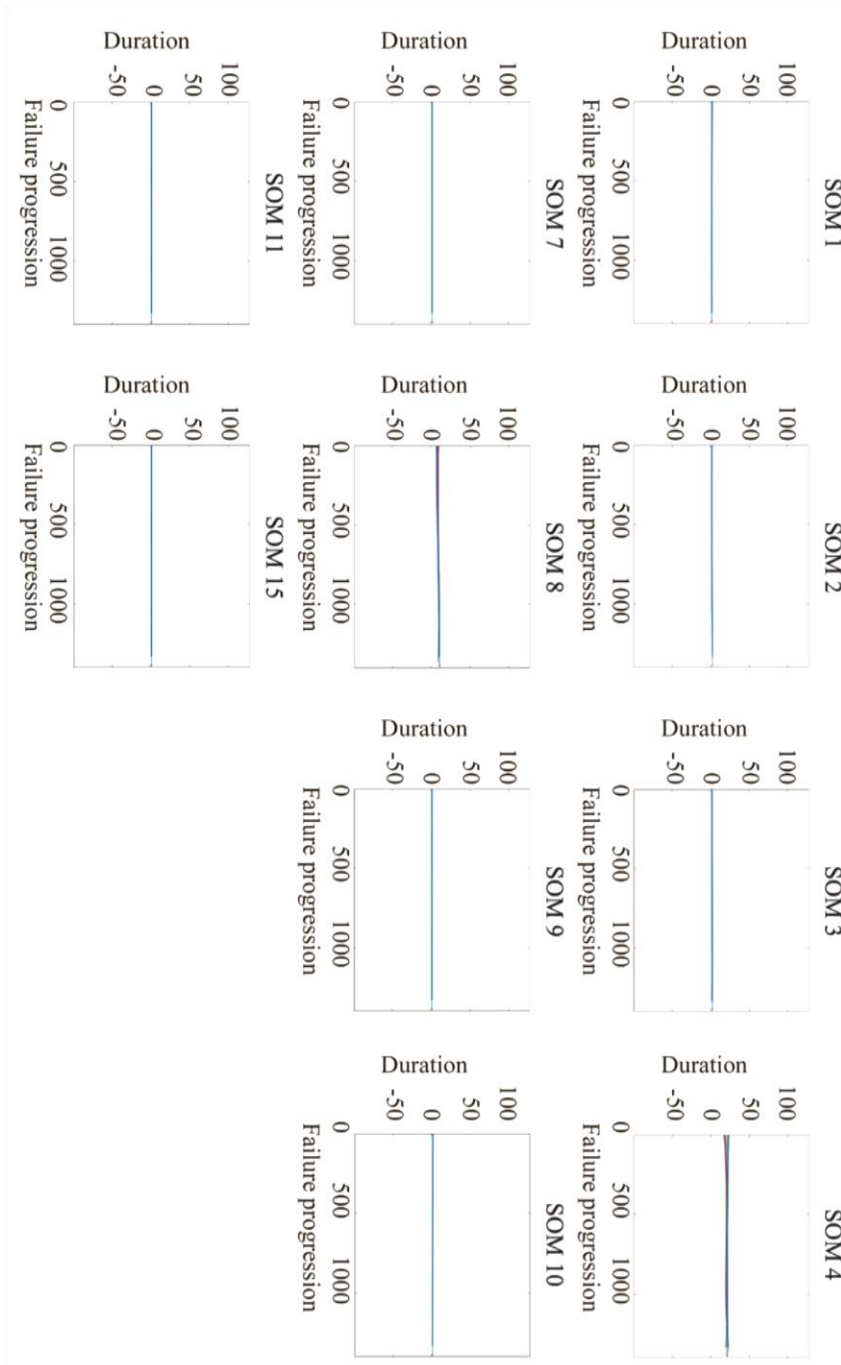


Figure A.10. Filtered trend of D for $F4$ of the kettle model

Appendix B

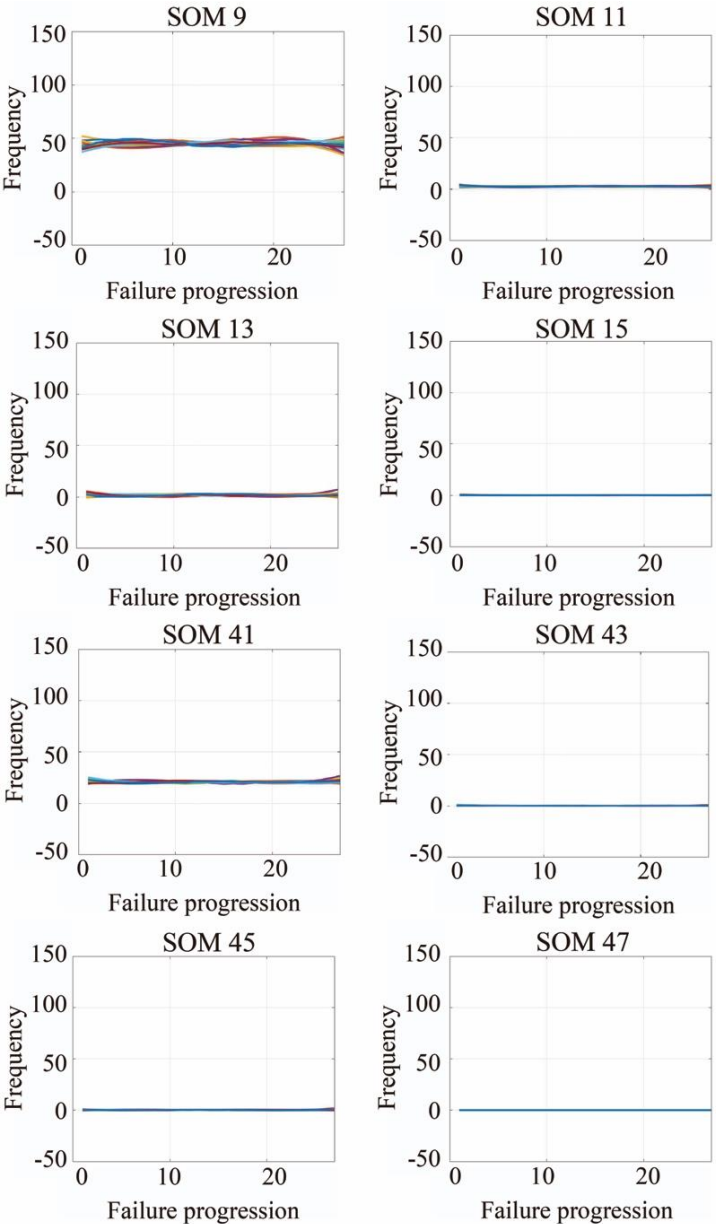


Figure B.1. Filtered trend of F_q for failure free scenarios of the testbed

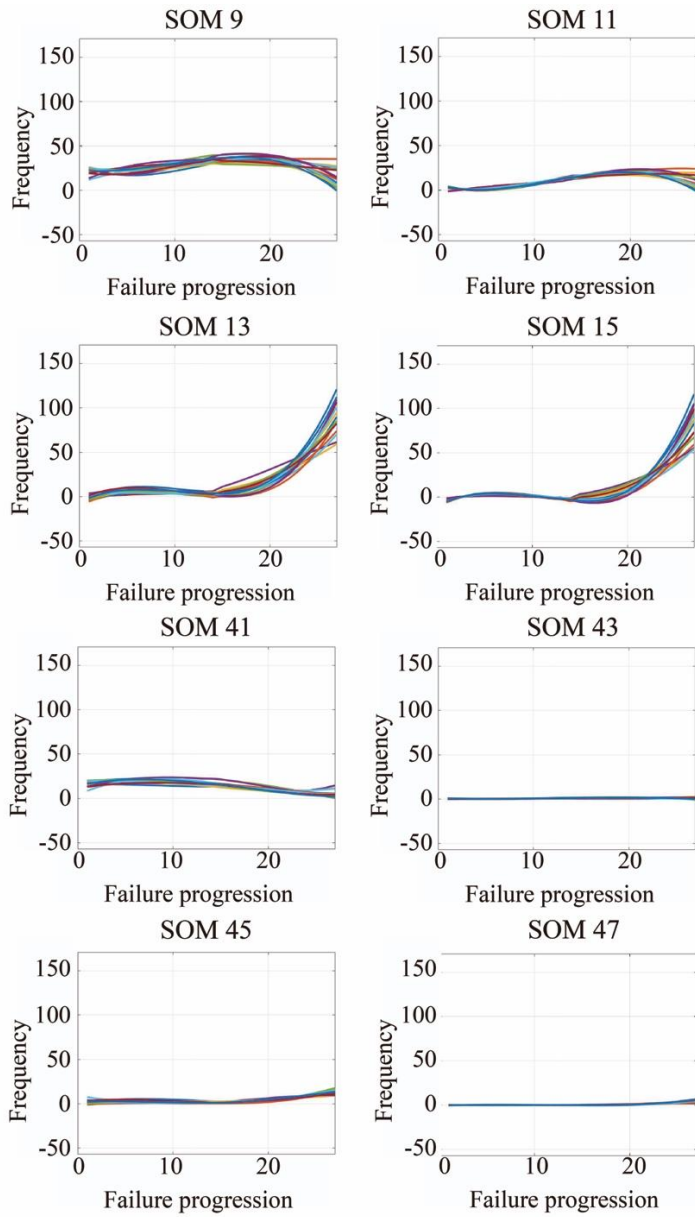


Figure B.2. Filtered trend of F_q for tank leak in the testbed

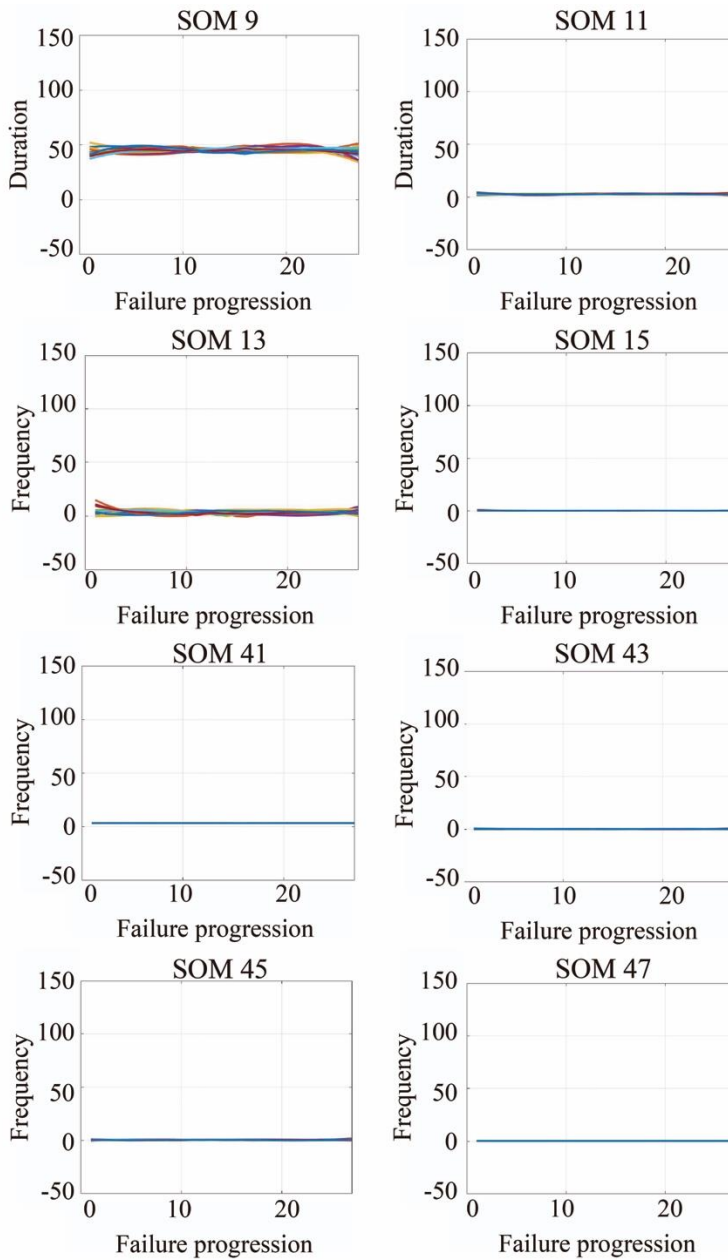


Figure B.3. Filtered trend of D for failure free scenarios in the testbed

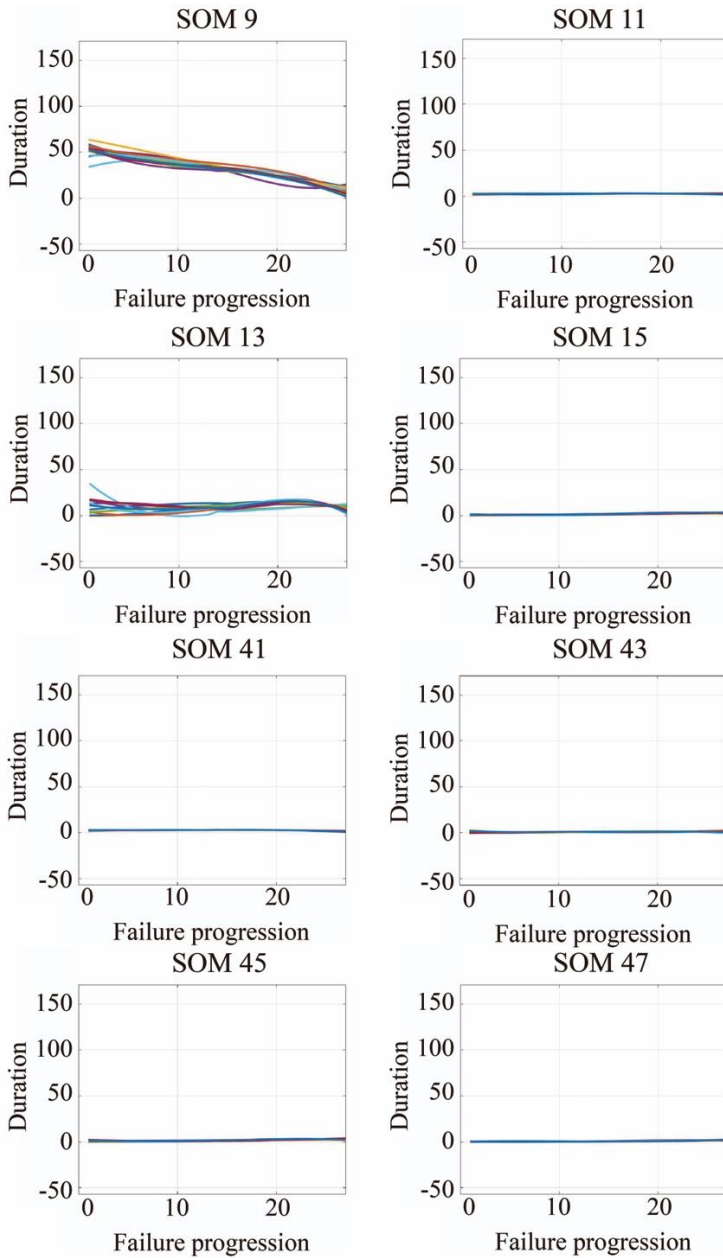


Figure B.4. Filtered trend of D for tank leak in the testbed

