Titre: Title:	Kalman Filter-based Digital Twin Framework for Anomaly Detection and System Identification of a Pipe Conveying Fluid
Auteur: Author:	Vincent Laperle
Date:	2025
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Laperle, V. (2025). Kalman Filter-based Digital Twin Framework for Anomaly Detection and System Identification of a Pipe Conveying Fluid [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/66327/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/66327/
Directeurs de recherche: Advisors:	Frederick Gosselin
Programme: Program:	Génie mécanique

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Kalman Filter-based Digital Twin Framework for Anomaly Detection and System Identification of a Pipe Conveying Fluid

VINCENT LAPERLE
Département de génie mécanique

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie mécanique

Mai 2025

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Kalman Filter-based Digital Twin Framework for Anomaly Detection and System Identification of a Pipe Conveying Fluid

présenté par **Vincent LAPERLE**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Youssef DIOUANE, président
Frédérick GOSSELIN, membre et directeur de recherche
Farhang DANESHMAND, membre

DEDICATION

N'est-ce pas dans le rêve, cependant, que naissent
la plupart des projets qui en valent la peine ?
- René Lévesque

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Frédérick Gosselin, for giving me the opportunity to work under his mentorship. Although the initial project was quite different, this did not stop him from believing in me and offering me the opportunity to pursue my academic goals. His unfailing support, humanity, and wisdom have had a profound impact on my personal and academic development.

I also want to thank Esmaeil Ghorbani for pushing me beyond my limits throughout this project. His time and dedication to the success of this project were a great source of motivation and help. Esmaeil is the teammate we all wish we had, and future students lucky enough to work with him can count on his boundless assistance and rigorous work ethic.

My sincere thanks go to Hydro-Québec and MayaHTT for their financial and technical support. A special thanks to Quentin Dollon, whose expertise, comments, and ideas helped guide this project. Thank you for your time and trust.

Many thanks are due to the guidance and support provided by the LM2 laboratory and its students. In particular, I would like to thank Ludivine Moyne, Mohammed Abda, Lucas Berthet, Sima Rishmawi, Maryam Boukor, and Hong Yan Miao for their ideas, lively discussions, attention, and mutual support. Moreover, I have to thank Fabrice Danet for his technical support and Morgan Demenois for laying the foundations for this project. Finally, I extend my gratitude to Professor Yahya Modarres-Sadeghi from the University of Massachusetts Amherst for generously providing the nonlinear code of the pipe conveying fluid, which significantly aided the expansion of ideas and analysis within this project and to further develop the DT framework.

I am also deeply indebted to my parents for their love, the values they imparted, and the autonomy they gave me, which allowed me to follow my chosen path with their constant support. To Alessia, my girlfriend, I express my heartfelt thanks for her constant emotional support and willingness to listen. Finally, I wish to acknowledge my brother, Thomas, for his generous artistic contribution to this project, despite the pressures of his own university studies. Finally, my sincere gratitude extends to my friend, Vincent Taboga, whose encouragement to pursue a master's degree and wise counsel in selecting a like-minded supervisor were pivotal in this work.

RÉSUMÉ

Au Québec, 94 % de l'énergie provient de l'hydroélectricité. Or, ce mode de production est confronté à des enjeux croissants : imprévisibilité climatique, acceptabilité sociale limitée pour de nouveaux barrages et vieillissement des turbines. L'intégration de sources énergétiques intermittentes pour varier les modes de production énergétique induit un changement d'opération des turbines hydrauliques. Initialement conçues pour un régime permanent, ces dernières sont soumises à des cycles de charge partielle et des séquences de démarrage/arrêt plus fréquentes. Ces conditions additionnées au vieillissement des machines accentuent les risques de bris causant l'arrêt de production énergétique des turbines. Les stratégies de maintenance périodique actuelles ne permettent pas d'identifier en amont les sources de défaillance. Ainsi, prédire ces défaillances dans les turbines permettrait d'optimiser leur maintenance et de réduire les charges monétaires associées à la perte de la machine et l'arrêt de production énergétique.

L'objectif général de ce projet est de développer un cadre d'applications et d'algorithmes pour la construction d'un jumeau numérique des turbines hydrauliques. À partir des données des capteurs, et des formulations mathématiques décrivant la dynamique du système, le jumeau numérique vise à reconstruire une copie digitale de la turbine. Conjointement avec les algorithmes développés, le but est de prédire les sources de défaillances dans le système via l'analyse vibratoire de la machine, l'identification des paramètres mécaniques inconnus et réaliser le suivi tout au long de la vie utile de la machine en temps réel. Construire un jumeau numérique d'une turbine s'avère fastidieux et coûteux d'où l'utilisation du systèmes académiques du tuyau parcouru par un écoulement. Outre sa simplicité, ce tuyau imite les interactions entre l'eau et la structure rencontrées dans une turbine comme l'amortissement ajouté de l'eau, le flottement de la structure, et l'excitation induite par l'écoulement. Aussi, ce montage expérimental permet d'étudier une large gamme dynamique, soit du linéaire au chaos, en variant la vitesse de l'écoulement. Ce modèle largement étudié dans les 70 dernières années a un cadre mathématique décrivant cette dynamique vérifié et validé. Ainsi, l'hypothèse du projet stipule qu'il est possible de transférer le savoir acquis aux turbines hydrauliques en testant notre cadre de jumeau numérique sur le tuyau.

Le montage consiste en un tuyau en caoutchouc siliconé et deux caméras évènementielles enregistrant la variation d'intensité lumineuse des pixels du capteur. Ces capteurs produisent un nuage continu de coordonnées des pixels excités dans le temps provenant du mouvement du tuyau. Cette nouvelle méthode de capture du mouvement des tuyaux, permet d'écarter

les traitements d'images nécessaires à la reconstruction numérique du tuyau. Ce nuage de point est discrétisé en intervalle de temps régulier de 0.025 s et assorti en ordre croissant afin d'y passer une fonction polynomiale pour reconstruire la forme du tuyau. Avec un traitement de la parallaxe, le tuyau est reconstruit numériquement en trois dimensions avec une erreur moyenne de 2.3 % par rapport à son mouvement réel. L'erreur est de 6 % lorsque la pointe du tuyau atteint son amplitude d'oscillation maximale. Ces données reconstruites sont utilisées pour identifier les paramètres inconnus du tuyau et détecter les anomalies dans le système. Un algorithme *Unscented Kalman Filter* (UKF) utilise un modèle mathématique formulant le comportement nonlinéaire du tuyau pour estimer le déplacement et la vitesse du tuyau et les paramètres inconnus comme la vitesse de l'écoulement de l'eau. Le modèle nonlinéaire permet de simuler et de suivre l'oscillation continue dans le tuyau. En comparant les estimations du filtre avec les données simulées numériquement reliées à notre système mécanique, le UKF estime avec une erreur de 0.47 % le déplacement du tuyau et de 0.2 % la vitesse de l'écoulement de l'eau. En comparant les estimations du UKF avec les données réelles des capteurs et d'un débitmètre, ces erreurs augmentent à 0.8 % pour le déplacement et à 6 % pour l'estimation de la vitesse d'écoulement de l'eau. Pour accélérer le suivi du système, un algorithme de filtre de Kalman linéaire (LKF) combiné à un modèle de décomposition des modes dynamiques (DMD) est employé. À l'aide de la DMD, le modèle nonlinéaire est réduit aux quatre modes vibratoires les plus importants dans le but de réduire la complexité du calcul. Ces modes représentent l'état stationnaire de notre système et sont sauvegardées comme référence pour réaliser la détection d'anomalie. Le LKF est employé pour estimer ces modes vibratoires dans le temps à partir des données des capteurs. À l'aide d'un algorithme *Steepest Ascent Hill Climbing* (SAHC), la distance euclidienne et leur entropie entre les modes capturées par la DMD et ceux estimés par le LKF sont comparés. Si cette distance dépasse une limite de 3σ ou que l'entropie capture une variation modale, une anomalie est signalée. Des anomalies comme l'arrêt soudain d'un capteur ou la diminution de la vitesse de l'écoulement via la perte d'efficacité dans le système sont reconnues. Le LKF-DMD accélère le suivi de la dynamique du tuyau de 12 s par itération pour le UKF a 10^{-3} s par itération.

Les algorithmes développés sont combinés dans un cadre d'applications pour tester en temps réel notre jumeau numérique. L'identification des paramètres inconnus est d'abord réalisée suivie de la détection des anomalies à partir des données obtenues des caméras évènementielles. Ce cadre itératif est fait de manière continue jusqu'à l'arrêt de l'enregistrement des données. De ce cadre, les défis identifiés et les recherches futures devront se pencher sur la puissance de calcul allouable, l'interopérabilité des algorithmes et l'automatisation des procédés via l'apprentissage machine.

ABSTRACT

Ageing of hydro turbines and the addition of intermittent energy sources to the grid imply partial load operation of the turbines under stringent conditions. The digital twin (DT) as a predictive maintenance tool could be a solution to optimise maintenance, reduce power downtime, and related economic losses. However, building a functional DT of a hydro turbine from scratch is tedious and expensive. Hence, to study the fluid-structure interaction phenomena in hydro turbines, the main objective of this project is to build a digital twin framework of a pipe conveying fluid system instead. The nonlinear pipe oscillation is captured with two event-based cameras and reconstruct in 3D through a parallax and polynomial fitting algorithm. Then, an unscented Kalman Filter (UKF) with the nonlinear formulation of the pipe is used to solve the inverse problem and estimate the unknown flow rate inside the pipe. To accelerate the monitoring procedure, we run a linear Kalman filter (LKF) combined with dynamic mode decomposition (DMD) to track the modal changes in the event-based data reconstruction. With the latter code, anomaly detection is performed by comparing the LKF estimates with the known modal patterns of the DMD using the Euclidean distance inside a steepest ascent hill-climbing algorithm and an entropy-based algorithm on the Euclidean distance evolution. The whole is merged inside a workable framework that performs in our lab the flow rate identification and anomaly detection. In this work, with event-based outputs, we reconstructed the pipe oscillation with an average error of 2.3 % in which a maximal error of 6 % was observed once the tip of the pipe reached a large amplitude. To ensure that the polynomial fitting has sufficient event coordinates, we find that the optimal time window should be set to $\Delta t = 0.025$. The parameter identification with UKF is dependent on the accuracy of the prior knowledge formulation. For simulated data, the UKF is able to estimate the deflection with an average error of 0.47 % and the unknown flow rate with an error of 0.2 %. The validation of the system identification with UKF is done by using the event-based data and a flow metre. The UKF estimates increase to 0.61 % for the deflection and 6 % for the flow rate. LKF-DMD manage to accelerate the process from 12 s per iteration to 10^{-3} s per iteration. We find that the entropy method catches well anomalies representing slow modal changes such as machine efficiency decrease. The Euclidean method alone detected instantaneously sudden perturbation peaks such as sensor defects. Overall, this work shows that the proposed DT framework is still in its infancy. Hence, building a mature DT framework will require, among others, the interoperability of multiple algorithms, hybrid modelling, high computational power, and efficient machine learning technique to generalise and automate the process.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS AND SYMBOLS

Acronyms

CN	Crank-Nicolson
DMD	Dynamic Mode Decomposition
DT	Digital Twin
EKF	Extended Kalman Filter
EVB	Event-Based
FFT	Fast Fourier Transform
FIFO	First-In First-Out
FSI	Fluid-Structure Interaction
HMI	Human-Machine Interaction
KFT	Kalman Filter Technique
LKF	Linear Kalman Filter
PA	Physic-Aware
PHM	Prognostic and Health Management
PINN	Physics Informed Neural Network
POD	Proper Orthogonal Decomposition
PSD	Power Spectral Density
RMS	Root Mean Square
ROI	Region of Interest
SAHC	Steepest Ascent Hill Climbing
SO	Sub-Objectives
SVD	Singular Value Decomposition
UKF	Unscented Kalman Filter
3D	Three-Dimension
2D	Two-Dimension

Symbols

Linear and Unscented Kalman Filter

$\mathbf{x}$	State vector
$\hat{\mathbf{x}}$	State estimate vector
$\mathbf{y}$	Measurement vector
$\hat{\mathbf{y}}$	Measurement estimate vector
$\boldsymbol{\theta}$	Parameter vector
$\hat{\boldsymbol{\theta}}$	Estimated parameter vector
$\mathbf{v}$	Process noise vector
$\mathbf{w}$	Measurement noise vector
$\mathbf{P}$	Covariance matrix
P_θ	Standard deviation associated with the estimated unknow parameter
$\mathbf{Q}$	Process noise covariance matrix
$\mathbf{R}$	Measurement noise covariance matrix
α_Q	Process forgetting factor
α_R	Measurement forgetting factor
$\mathbf{A}$	Linear transient function matrix, Best fit linear operator (DMD)
$\mathbf{H}$	Linear measurement function matrix
$\mathcal{M}$	UKF transient function matrix
$\mathcal{M}^a$	Augmented UKF transient function matrix
$\mathcal{H}$	UKF measurement function matrix
k	Time increment
ω	Weight of the sigma points
χ	Sigma points
N	Number of beam mode shapes
λ_λ	UKF scaling parameter
α_λ	UKF scaling parameter
β_λ	UKF scaling parameter
κ	UKF scaling parameter
M	Number of unknow parameter
Δt_{UKF}	Time step incremented in the UKF
$\mathcal{L}$	PA loss function for the NOMAD optimisation
Ω	Prior knowledge in the model

LKF-DMD module

$\mathbf{X}$	DMD initial data snapshot matrix
$\mathbf{X}'$	DMD data snapshot one time step latter matrix
$\mathbf{U}$	SVD left singular vector matrix
$\mathbf{\Xi}$	SVD singular value matrix
$\mathbf{V}$	SVD right singular matrix
$\tilde{\mathbf{U}}$	Reduced-rank SVD left singular vector matrix
$\tilde{\mathbf{\Xi}}$	Reduced-rank SVD singular value matrix
$\tilde{\mathbf{V}}$	Reduced-rank SVD right singular matrix
r	Rank of the vector or the matrix
$\tilde{\mathbf{A}}$	Reduced-rank DMD best-fit linear operator
Δt_{DMD}	Time step incremented in the LKF-DMD
$\tilde{\mathbf{v}}$	Reduced modal shape state from DMD training
$\hat{\mathbf{v}}$	Reduced modal shape state estimate from LKF
$\mathcal{E}$	Euclidean distance
$\bar{\mathcal{E}}$	Averaged euclidean distance
$\sigma_{\mathcal{E}}$	Standard deviation associated with the euclidean distance
$\mathcal{S}$	Entropy
$\bar{\mathcal{S}}$	Averaged entropy
$\sigma_{\mathcal{S}}$	Standard deviation associated with the entropy
CT	Computational time of the LKF-DMD
A_N	Amplitude of the simulated decay
Δt_{rand}	Randomised time period to simlulate a perturbation

Pipe conveying fluid - Dimension symbols

x	Horizontal coordinate of the pipe in the x-axis
y	Horizontal coordinate of the pipe in the y-axis
z	Vertical coordinate of the pipe
t	time coordinate
L	Lenght of the pipe
E^*	Kelvin-voigt damping parameter
E	Young modulus of the pipe silicon rubber
U	Flow rate inside the pipe
g	Gravity parameter
d	Inner diameter of the pipe

D	Outer diameter of the pipe
I	Quadrature moment of the pipe
m	Linear mass of the pipe
M	Linear mass of the pipe going through the pipe
ρ	Density of the water

Pipe conveying fluid - Dimensionless symbols

$\boldsymbol{\eta}$	Transverse Displacement vector in the xz plane
$\boldsymbol{\zeta}$	Transvers Displacement vector in the yz plane
ξ	Vertical coordinate of the pipe
τ	Time
$\Delta\tau$	Time step
α	Damping parameter
β	Mass parameter
γ	Gravity parameter
u	Flow rate

Euler-Bernoulli beam

ϕ_i	i^{th} beam mode shape in the xz plane
ψ_i	i^{th} beam mode shape in the yz plane
$\mathbf{\Phi}(\xi_j)$	Beam mode shape matrix
λ_i	i^{th} beam eigenfrequencies

Nonlinear pipe coefficients

$\tilde{\mathbf{M}}_{ij}$	Linear mass matrix
$\tilde{\mathbf{C}}_{ij}$	Linear damping matrix
$\tilde{\mathbf{K}}_{ij}$	Linear stiffness matrix
$\mathbf{B}_{ijkl}$	Nonlinear coefficient matrix
$\mathbf{D}_{ijkl}$	Nonlinear coefficient matrix
$\mathbf{E}_{ijkl}$	Nonlinear coefficient matrix
$\mathbf{H}_{ijkl}$	Nonlinear coefficient matrix
$\mathbf{M}$	Mass matrix of the linearised system
$\mathbf{C}$	Damping matrix of the linearised system
$\mathbf{K}$	Stiffness matrix of the linearised system

$\mathbf{F}$	Nonlinear contribution vector of the linearised system
$\mathbf{q}$	Generalised coordinate vector in the xz
$\mathbf{p}$	Generalised coordinate vector in the yz
$\boldsymbol{r}$	Generalised 2D coordinate vector of the system
$\boldsymbol{\mathcal{R}}$	Augmented 2D coordinate vector of the system

Crank-Nicolson scheme

Δt_{CN}	Time step inside the scheme
Δh	Space step in the scheme
$\mathcal{O}$	Convergence order of the scheme
ϵ_{CN}	Numerical error of the scheme

Event-based vision

e	Events
p	Boolean polarity
t_e	Timestamp of the event
x_e	Horizontal coordinate of the event in the pixel frame
y_e	Vertical coordinate of the event in the pixel frame
Δt	Time window in the event's cloud
x_s	Simulated horizontal coordinate for 2D benchmark
y_s	Simulated vertical coordinate for 2D benchmark
x_{fit}	Polynomial fitting horizontal coordinate for 2D benchmark
y_{fit}	Polynomial fitting vertical coordinate for 2D benchmark
x_{rc}	Reconstructed horizontal coordinate
y_{rc}	Reconstructed vertical coordinate
x_c	Corrected horizontal coordinate in the parallax step in the xz plane
y_c	Corrected horizontal coordinate in the parallax step in the yz plane
z_c	Corrected vertical coordinate within the parallax framework
D_x	Distance between the pipe and the front camera
D_y	Distance between the pipe and the side camera
N_L	Discretised pipe lenght
ϵ_{RMS}	RMS error between simulated and polynomial fit coordinates
δ_{RMS}	Tip RMS error between simulated and polynomial fit coordinates
A_{xz}	Amplitude of oscillation in the xz plane
A_{yz}	Amplitude of oscillation in the yz plane

General symbols and Mathematics

$\mathbf{I}$	Identity matrix
$\mathbf{O}$	Null matrix
$\mathbf{L}$	Lower triangular matrix
$\mathbf{D}$	Diagonal matrix
$\mathcal{I}$	Identity function
$\mathbf{S}$	Lower triangular matrix in a Cholesky decomposition
δ_{ij}	Kronecker symbol
$\mathcal{N}$	Gaussian distribution symbol
$\|\|\|_2$	L2-norm
$\|\ \|$	Absolute value
$\dot{}$	Time derivative
$'$	Space derivative
$^\top$	Matrix transpose

LIST OF APPENDICES

CHAPTER 1 INTRODUCTION

In Quebec, $77 million was invested in maintenance due to weather-related events in 2023 (Hydro-Québec, 2024a). In the same year, the impact of ageing facilities led to the shutdown of the Carillon, Rapide-Blanc, Trenche and Outardes-2 power stations. This shutdown implied an investment of $2.4 million in maintenance for the said plants, and a slowdown in their energy production for the next five years (Bergeron, 2023). In addition, with the construction of the latest dam, La Romaine, the social licence to operate for new hydroelectric projects is being divided, while the possibility of harnessing new watercourses becomes more difficult (Beaulieu, 2019). With Quebec's goal of carboneutrality by 2050 and 94% of the energy production (Gouvernement du Canada, 2024) coming from hydroelectricity, Hydro-Québec cannot rely solely on its ageing dam and on new hydroelectric projects. Hydro-Québec must now diversify its energy production to overcome the latter environmental, technical, and social challenges.

By relying on intermittent energy sources, such as wind and solar power, and investing in new technologies, hydroelectric units no longer operate continuously at the base load as originally designed. Instead, they are subject to more stringent conditions, causing them to operate more often with partial load and a start-and-stop cycle (ISO, 2020). Consequently, Hydro-Québec's already ageing units are under greater strain and prone to failure. Due to the complexity and large scale of hydro turbines, most inspections and maintenance strategies follow systematic time-based and corrective plans (Li et al., 2020). However, such a maintenance strategy often misses out on system failures. Efforts have been made to optimise the inspection of hydro turbines with different numerical and technical strategies (Li et al., 2020; Gagnon et al., 2020; Lamothe et al., 2020; Boukani et al., 2014; Gagnon et al., 2019). However, from these studies one can argue that there is a lack of standardisation, a scarcity in maintenance and failure events recorded in the available data, and a low accuracy of the predictions. In Quebec, Hydro-Québec periodically assesses the useful life of their units (Lamothe et al., 2020). Such a method represents power downtime and consequently economic losses. Furthermore, in some cases, systematic maintenance may not anticipate an imminent unit failure, which could require an unexpected unit replacement. A digital twin (DT) could be a solution for condition and predictive maintenance by accurately optimising the inspection procedure, thus reducing downtime and associated cost.

Following this introduction, we present the motivations for developing a DT of a pipe conveying fluid system. We end this Introduction by exposing this master thesis organisation.

1.1 Motivations

1.1.1 The Digital Twin as a Predictive Maintenance Tool

A digital twin as a predictive maintenance tool could overcome the lack in systematic maintenance by predicting future system failure and, consequently, optimising the inspection operation, reducing power downtime and economical losses. The idea of DT was first introduced by Michael Grieves as a product life cycle management tool in 2002 (Grieves, 2015). During the last decade, articles on the subject have increased from 1000 to more than 52,000, while the scope of DT has gone from smart manufacturing to city management, predictive maintenance, economy and healthcare, to name a few (Grieves, 2023; Semeraro et al., 2021). Depending on the DT application, its definition and specificity vary accordingly. Since there is no consensus definition of a DT, most agree that the core concept is an intelligent two-way connection between a physical entity and its virtual replica. Thus, the core components of the DT are the digital models that describe both the physical and virtual counterparts, the connection to the data network (Tao et al., 2019) and the ability of the DT to evolve in tandem with its physical asset by providing valuable information (Ayani et al., 2018). The latter ability works as a closed-loop real-time system between both physical and virtual entities to proactively update the physical system and predict future outcomes. The absence of autonomous update in DT is known as a digital shadow (Stark et al., 2017), where predictions are made in real-time, but in an open loop system. In addition to the three core components of DT, the authors have complement the DT architecture by adding a service component (Tao et al., 2018) to link the user to the DT and the distributed computing and storage available component (Ponomarev et al., 2024) to highlight the computer capabilities to run the DT. In the field of prognosis and health management (PHM) of mechanical structures or systems, DT should be able to overcome different tasks depending on its maturity level. For a mechanical system such as hydro turbines, a low-level DT could detect a system failure, while a high-level DT could furnish an automated decision with human-in-the-loop features to solve future problems of the system. Thus, a predictive maintenance DT could diagnose types of failure, predict future results, and prescribe real-time solutions.

Digital models concern every algorithm and mathematical function that discloses both physical and virtual entities. Digital models can be physical, data-driven, knowledge-based, or a combination of all, known as hybrid- or multi-model (Tidriri et al., 2016).

Virtual models are the projected results of the physical asset that includes geometric, physical, behaviour, and cognitive models (Tao and Zhang, 2017; Tao et al., 2018, 2019; Uhlenkamp et al., 2022). First, geometric modelling refers to physical properties such as size, shape, and

position. Second, physical modelling in virtual space is any derived mathematical formulation from the geometric models describing the physical phenomena. Third, the behaviour models are the dynamical responses arising from the last two models. Finally, cognitive models are any logical correlation or deduction that describes the domain knowledge in the virtual space, making the virtual model capable of reasoning and predicting.

Regarding the connection of the data network, Semeraro et al. (2021) and Uhlenkamp et al. (2022) mention the importance of a smooth and flexible exchange of information between both assets. Acting as the language of the assets, this communication guarantees the success of the DT ability to be interoperable.

Finally, DT is characterised by its (i) scalability, (ii) interoperability, (iii) fidelity, (iv) dynamicity and (v) modularity (Semeraro et al., 2021; Kapteyn and Willcox, 2021). The DT should be scalable considering all fine- and large-scale phenomena in the system. The interoperability of the DT would ensure that the architecture integrates various collaborating models. Fidelity refers to the closeness of the digital models with the reality, while the dinamicity concerns the real-time DT capability to solve the given problems and to update itself. Finally, the modularity characteristic is the flexible capacity of the DT to integrate or replace any models in its architecture so that it can act on larger systems. Together, all five characteristics make the DT a unique and personalised system capable of adapting itself to any change. From a predictive maintenance perspective, these characteristics could be beneficial for: (A) early detection of areas at risk of failure; (B) support and optimise decision making in maintenance; (C) predict more accurately the time to failure and the remaining useful lifetime of the mechanical system; (D) reduce the need for a large amount of data (Harries et al., 2023). Thus, by reducing the number of inspections, the industry can increase its efficiency, reliability, and sustainability while reducing downtime, cost, and breakdowns. In this work, our aim is to develop a framework of a Digital Twin for predictive maintenance of hydro turbines.

In the hydropower industry, DT for predictive maintenance is in an early stage with no validated technologies. In Korea, the K-Twin Sj DT system (Park and You, 2023) was developed to support decision-making on maintenance of the dam structure and predict the level of water in the river basin. In China, a DT of primary hydroelectric equipment was verified with simulated data using a radio frequency identification model (Cai et al., 2024). In Québec, Hydro-Québec tested and validated a DT on a hydro turbine speed governor for anomaly detection (Giroux et al., 2023). This study case is part of a wider project to build DT for hydro turbines. The project has been in operation since 2018 and has become a global asset management programme in 2025. The diagnosis of sensor failure was performed on a pressure

outlet in a hydroelectric plant in Turkey (Ersan and Irmak, 2024). In Wang et al. (2023), the DT is applied in a linear formulation of a Francis Turbine dynamic with an adaptive learning model. Although in the general power industry, predictive maintenance with a DT framework has been applied on wind and gas turbines, photovoltaic plants, and nuclear power systems (Yassin et al., 2023), the latter DT frameworks lack modularity, standardisation, and dynamicity (Choi et al., 2024; Yassin et al., 2023). Moreover, they are often misinterpreted as digital shadows, while no verification, validation, and uncertainty quantification procedures are performed (Giroux et al., 2023). Finally, a common limit for all DT is their computer capabilities and data storage according to the complexity of the systems.

1.1.2 Multi-model Approach for Digital Twin Implementation

Developing a DT framework for the predictive maintenance of hydro turbines requires at least the implementation of the five key DT characteristics of Semeraro et al. (2021). However, a predictive maintenance strategy that relies solely on a single digital modelling approach, such as Physics-based, Data-driven, or Knowledge-based, is limited. This limitation arises from the challenges associated with hydro turbine data, including data quality and scarcity issues, high dimensionality, and inherent complexity of such a system, but also from the challenges of each model.

Knowledge-based methods are logical reasoning procedures based on rules from expert observations (Liao and Köttig, 2014) and the inference rules algorithm (Russell et al., 2010). Knowledge-based systems are simple to understand, but often lack the accurate knowledge and experience for reliable prognostics. Moreover, this experience can be difficult to find and costly (Montero Jimenez et al., 2020).

Data-driven methods are entirely based on sensor data and are often classified as statistical models, stochastic models, machine learning methods, or a combination of the latter three (Tidriri et al., 2016; Montero Jimenez et al., 2020). Data-driven techniques can be useful in systems with nonlinearities, multiscaling in space and time, and high dimensionalities by modelling and characterising dominant patterns in the data (Brunton and Kutz, 2022). Data-driven models are fast and easy to implement, which explains why they are the main models used in industry fault diagnosis (Tidriri et al., 2016). However, in such models, the availability and quality of data are the main limits. Sensor data can be sparse, scarce, and noisy (Giroux et al., 2023; Ellis et al., 2022). In addition, to extract the dominant patterns, a pre-processing step is necessary, which adds a delay in the analysis (Tidriri et al., 2016). In the case of predictive maintenance, data-driven models alone lack explainability and interpretability, making the prognostic tedious (Giroux et al., 2023).

For the physics-based method, mathematical models are used to describe the dynamical behaviour of the observed system (Tidriri et al., 2016; Montero Jimenez et al., 2020; Giroux et al., 2023). Unlike a data-driven technique, an accurate mathematical model can solve the explainability and interpretability limits if an analytical solution exists. In most high-dimensional and nonlinear problems, analytical solutions are unsolved and assumptions are made which bring modelling uncertainty to the mathematical modelling step. Thus, a verification and validation step is needed before using the model to quantify the associated uncertainties. In addition, the physic-based method faces the dilemma of computational time versus accuracy.

The multi-models approach aims to take advantage of all three single-model benefits while bridging the gap of their respective challenges. In a review of the literature of 2018, 8% of the articles concern multi-models (Lei et al., 2018), while a second review of the literature of 2020 showed an increase of 20% on the subject demonstrating the growing interest in the matter (Montero Jimenez et al., 2020). Multi-models also known as hybrid models can be a dual or trial combination of all three single models. Choosing the right multi-model combination and configuration vary depending on the nature of the problem, the knowledge available, the data availability and quality, and the mathematical model's level of fidelity.

This study presents a DT framework based on a hybrid multi-model approach. This framework integrates information from data, physical models, and expert knowledge, the latter being incorporated through the specification of prior knowledge governing the system's dynamics. The objective is to take advantage of the complementary advantages offered by each individual modelling technique. We use Kalman filtering techniques (KFT) such as the linear Kalman filter (LKF) and the Unscented Kalman Filter (UKF) as the main algorithms for our DT framework. In fact, these latter algorithms stand out as a compelling tool because of their real-time monitoring capabilities, accuracy, scalability to various problems, and straightforward implementation. KFT is a two-step iterative process that compares noisy sensor data with a physical model to estimate unknown states and parameters of the observed system. The KF capability to estimate both states and parameters is known as the inverse problem. For a DT implementation, solving the inverse problem demonstrates its ability to reduce the need for costly sensors in complex locations such as in hydro turbines and to track change in system mechanical properties. The UKF variant uses so-called sigma points to map the nonlinear state and parameter space domain. Thus, the UKF could compare *in situ* sensor data in hydro turbines with a high-fidelity physic-based model for predicting system failure and vital component parameters. Its main drawbacks are the trade-off between accuracy and computational time coming from the high-fidelity physics-based model and the complexity associated with the resolution of the inverse problem. To overcome this limit, authors have

shown that KFT could be combined with a data-driven model instead of physics-based to increase its computation efficiency (Rosafalco et al., 2024, 2025; Ghorbani et al., 2020). Here we use a Dynamic mode decomposition (DMD) model combined with an LKF to realise real-time monitoring and assess anomaly detection. The whole system could lead to automated diagnosis through the interoperability of LKF-DMD and UKF.

1.1.3 The Nonlinear Pipe Conveying Fluid Model to Develop a DT

Performing real-time monitoring of hydro turbines in a university is tedious, demonstrating the need for a smaller and simpler system. The pipe conveying fluid set-up is a simple and economical experimental montage exhibiting rich dynamics. The ease of building the pipe conveying fluid system makes it a perfect candidate to conduct a fast and reliable experimental test in a university laboratory (Demenois, 2022b). The richness of the dynamics displayed by this fluid-structure interaction (FSI) problem is often perceived as the new dynamical paradigm of the century (Semler, 1996). As the flow rate through the pipe increases, the dynamics of the pipe-conveying-fluid system exhibits a transition from damped oscillations to periodic, then doubly periodic, quasi-periodic, and even chaotic. This initial loss of stability, known as flutter, highlights the nonlinear behaviour of the system. Moreover, the pipe system has been extensively studied in the last 70 years and various mathematical models have been validated (Païdoussis, 2022). Hence, the pipe conveying fluid system is an ideal testbed for evaluating our DT on a continuous system, providing valuable insights into its applicability and limitations. Tracking the steady-state of the pipe in real-time can help us to develop algorithms to detect abnormal behaviour disrupting the constant fluttering. Compared to hydro turbines, phenomena such as water-added mass, flutter, flow-induced excitation, and hydrodynamic damping are also observable with the pipe conveying fluid model (Trivedi and Cervantes, 2017). Hence, the pipe system is an alternative for testing general algorithms that can be scaled for industrial context. Dynamics such as fluttering are fast phenomena characterised by complex pipe curves. Observing these phenomena with cameras with high spatial or time resolution is a compromise and costly. In addition, to avoid tedious image correction with computer vision algorithms, high-quality illumination and contrast are needed (Raynaud and Mulleners, 2024). Event-based (EVB) cameras continuously tracked the change in pixel intensity, creating a pixel cloud of events over time. Such cameras have high spatio-temporal capabilities, no motion blur, are independent of illumination, and are less expensive. Recording the deflection of the pipe conveying fluid system with EVB cameras could bridge the gap between the trade-off of spatial and frame-rate resolution, while avoiding the need for time-consuming computer vision algorithms. Integrating the nonlinear model of the pipe conveying fluid system and the displacement data of the EVB inside the UKF and

the LKF would allow us to solve the inverse problem of identifying hidden parameters and monitor the system in real-time.

1.2 Thesis Goals and Organisation

Here we present the work completed in the development of a DT framework of the pipe conveying fluid set-up. We present a method to acquire data with EVB sensors and reconstruct the shape of the pipe in three dimensions (3D). Then, we show how we can perform system identification with UKF. In addition, we demonstrate acceleration of monitoring and anomaly detection through the use of LKF-DMD. Lastly, we explain and operate the whole framework in the experimental pipe setup by merging the EVB reconstructed data with the KFTs employed in this project.

In Figure 1.1, the pipe conveying fluid is our mechanical system to demonstrate our DT framework. Prior knowledge of the pipe dynamics is injected inside the model. We collect data from the pipe by recording its displacement with EVB sensors. The data is then reconstructed to reshape the pipe dynamics and saved in a *csv* format. This data is then used in the UKF for system identification and in the LKF-DMD for real-time monitoring and anomaly detection. The UKF estimate is used to train the DMD inside the LKF. We close the loop of this flow chart by informing the user with the estimated parameters and states so that action can be taken on the pipe system.

After this introduction, this master thesis is structured as follows: Chapter 2 presents a review of the literature on KFTs, pipe conveying fluid, and event-based cameras. Chapter 3 synthesises this review and states the project objectives. Chapter 4 details the methodology, numerical and experimental results of pipe shape reconstruction using EVB sensors. Chapter 5 shows the parameter identification of the pipe with an UKF. Chapter 6 demonstrates real-time monitoring and anomaly detection using our proposed LKF-DMD. Chapter 7 merges the EVB sensor data with both UKF and LKF-DMD as a proof-of-concept for a future DT framework. Finally, Chapter 8 concludes the work by summarising the results and discussing limitations and future work.

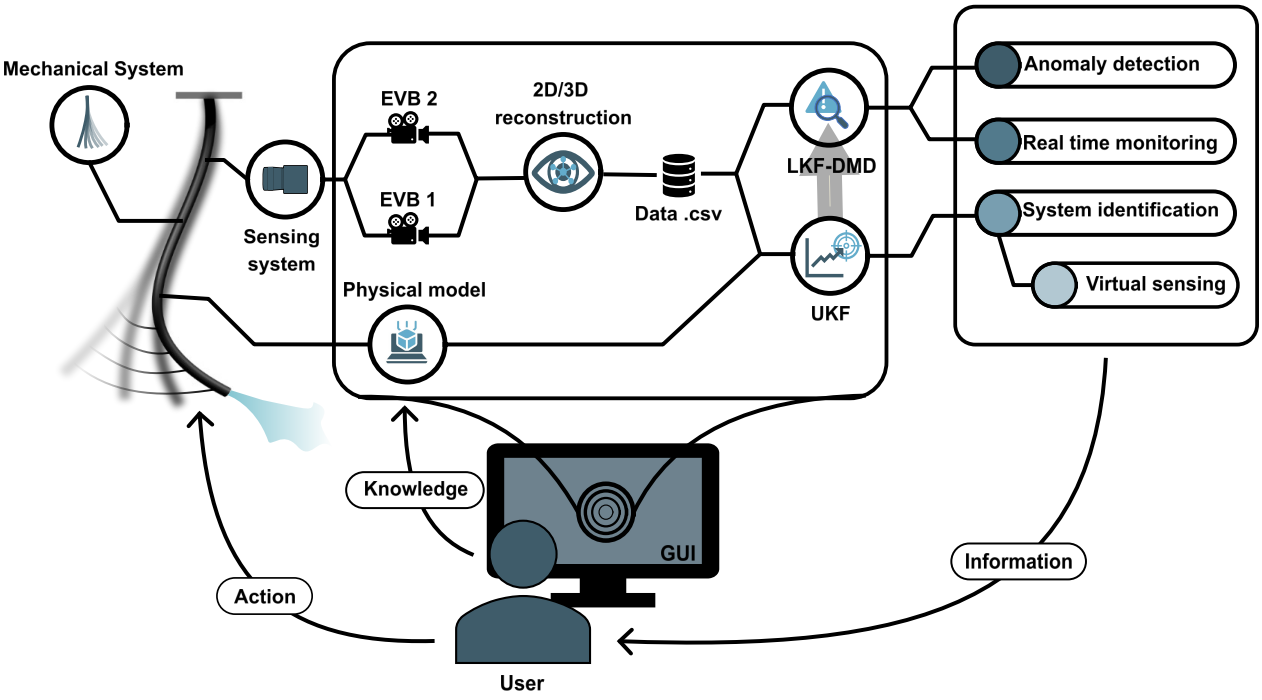

Figure 1.1 **Flowchart of the digital twin framework of the pipe conveying fluid setup**

CHAPTER 2 LITERATURE REVIEW

2.1 Kalman Filtering

Kalman filtering techniques have been widely used in industries for real-time monitoring, tracking, automation, and positioning (Khodarahmi and Maihami, 2023) to name a few. The simplicity of the mathematical formulation, the wide range of problems that can be tackled, and the high-speed and accuracy response are the main reasons for such notoriety. In a DT framework, such characteristics could be beneficial for fast and accurate prediction, making the KFT an interesting tool for implementing a DT.

2.1.1 Overview

Initially, the Linear Kalman Filter was a solution to the well-known Wiener problem, which was to predict, separate, and detect random signals for a low-dimensional linear dynamic system (Kalman, 1960; Särkkä, 2011). The Wiener filter, until then the benchmark tool, was computationally complex, and the mathematics behind the model lack transparency, making it a difficult tool to use in practice. The aim of the LKF is to minimise the error covariance of the system by following a two-step iterative process. Hence, LKF has become the basis for the development of modern control theory and real-time signal processing models by simplifying the mathematical background (Khodarahmi and Maihami, 2023). Since then, a panoply of Kalman Filters variants has been developed to tackle modern engineering problems such as nonlinearities, real-time monitoring, and optimal control. Thus, since most engineering problems encounter nonlinearities, the Extended Kalman Filter (EKF) was proposed followed by the Unscented Kalman Filter (Julier et al., 1995) as a response to the EKF limits. Indeed, to solve nonlinear problems with EKF, the system of equations has to be linearised through a first-order Taylor series expansion. However, this linearisation can introduce large errors and, in some cases, lead to divergence of the filter. In addition, in complex cases, the derivation of the Jacobian matrix is not trivial or impossible to calculate (Julier et al., 1995; Julier and Uhlmann, 2004; Menegaz et al., 2015). Finally, for highly nonlinear systems, the EKF tends to have a lower estimation performance (Ghorbani, 2021). The latter could be solved using a smaller time step, but this usually results in a higher computational cost (Julier and Uhlmann, 2004). In comparison, the UKF is a derivative-free and sampling-based algorithm that uses an Unscented Transform (UT) to map the mean and covariance domain of the target distribution (Särkkä, 2011; Julier and Uhlmann, 2004; Wan and Van Der Merwe, 2000).

In KFTs, the main architecture is similar and consists of a prediction step followed by an update step. In the prediction phase, the mean and covariance of the states at the current time step are propagated one time step ahead. This prediction is made by propagating the distribution through an *a priori* known state-transition function. The predictions are then updated by comparing to sensor measurements, resulting in a one-step advance state estimation. The algorithm iteratively processes this way until data depletion. KFT's are part of the Bayesian algorithm family since they rely on Baye's theorem to estimate the states. Hence, they can model the uncertainties in the model from the probability distribution in their initial assumptions (Särkkä, 2011). By only updating the states, *a priori* knowledge inside the model makes KFT a physics-based algorithm and an interesting tool for solving the inverse problem (Chen, 2003; Ghorbani, 2021; Ghorbani et al., 2024; Rosafalco et al., 2024). Inverse problem can be found in the system identification problem where unknown parameters are estimated with the known *a priori* physics. Inverse problem has a particularity to be under-determined, meaning that the solution of the problem can be the result of different combinations of states and parameters, and in some cases it cannot have solutions. From a DT perspective, solving the inverse problem can help identify unknown parameters in the system and follow their evolution over time to quantify and detect anomalies in the mechanical system. In this work, our aim is to solve the system identification problem, also known as joint estimation, which is considered as an example of the inverse problem (Wan et al., 1999; Ghorbani, 2021; Rosafalco et al., 2024). The system identification requires a nonlinear version of KFTs. By considering parameters as states, we inevitably transform any model with at least affine parameterisation into a nonlinear model with at least quadratic terms. Hence, the complexity and dimensionality of the problem increased with increasing number of unknowns. In this work, the UKF is employed due to its advantages over the EKF and shall be able to solve the inverse problem of any identifiable mechanical system.

However, it is important to note that the UKF is limited by the same challenges as those considered in the physics-based models. If the mathematical assumptions are too coarse and the problem is highly complex, it can add a computational burden to the model. Thus, to overcome this challenge, we propose to modify the physics prior knowledge to a data-driven one. Combining data-driven techniques with Kalman filtering is not a new topic. In Ghorbani et al. (2020), a random decrement technique is employed to reduce the unknown state and the vector of parameters. They were able to solve the inverse problem for a seven-story building. The downside of the method is its generalisation to cases outside the training range and the linear assumptions of the physics behaviour. The random decrement technique has to be retrained for every new case, implying a delay in the monitoring procedure, and they noted that nonlinear contributions were more impactful on the system than they thought.

Using a sparse identification of nonlinear dynamics method, Rosafalco et al. (2024) and Wang et al. (2022) manage to show the robustness of their data-driven Kalman Filter structure for nonlinear cases. In Wang et al. (2022), the model was applied to a time-invariant system only for state estimation. In Rosafalco et al. (2024), they study only synthetic cases. Machine learning techniques have also been combined with KFTs to increase the accuracy of target tracking (Liu et al., 2024), in vehicle performance monitoring (Tan et al., 2023) or to estimate the state of charge in the battery (Yang et al., 2023; Zeng et al., 2023). Lastly, modal decomposition techniques, such as proper orthogonal decomposition (POD), Fourier transform, and DMD, have shown promising results in minimising measurement noise while improving the accuracy of Kalman filter estimations (Nonomura et al., 2018, 2019). In all cases, the goal is to downscale the model complexity to improve computational efficiency while taking advantage of the Kalman filter's optimality, robustness, and simplicity. Here, we aim to develop a data-driven Kalman Filter model for real-time monitoring and anomaly detection in a discrete system.

In this work, we propose a dimensionality reduction with a DMD technique inside a LKF for real-time monitoring and anomaly detection. By running both UKF and LKF-DMD we aim to develop a modular DT framework that merges both algorithm advantages. This framework shall be able to integrate multiple complementary modules with only slight modifications.

2.1.2 Linear Kalman Filter

Here we use the mathematical derivation presented in Ghorbani (2021) for the development of the LKF algorithm. Hence, the current state $\mathbf{x}_{k-1}$ of the discrete state-space is incremented one time step ahead such that the predicted state $\mathbf{x}_k$ is defined as:

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{v}_k,\ \mathbf{v}_k \sim \mathcal{N}(0, \mathbf{Q}) \tag{2.1}$$

where $\mathbf{A}$ is the time-invariant transition matrix and $\mathbf{v}_k$ the process noise with the general assumption of white Gaussian distribution of zero mean and covariance $\mathbf{Q}$. The transition matrix describes how the system studied evolves over time, relying on known prior knowledge. This prior knowledge is a first-order form of the mathematical formulation that describes the observed dynamics. To update the Kalman filter state predictions, the true state $\mathbf{x}$ has to be correlated to the sensor measurement through the measurement vector $\hat{\mathbf{y}}$ such that:

$$\hat{\mathbf{y}}_k = \mathbf{H}\mathbf{x}_k + \mathbf{w}_k,\ \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{R}), \tag{2.2}$$

where $\mathbf{H}$ is the time-invariant measurement matrix and $\mathbf{w}_k$ the measurement noise vector. The measurement matrix makes the correlation between the sensor measurements and the filter observations $\hat{\mathbf{y}}$. Measurement noise is also assumed to follow a white normal distribution of zero mean and covariance $\mathbf{R}$. This white-noise assumption is motivated by the fact that the modelling and measurement uncertainty need to be incorporated into the system identification process. Hence, the motivation to use the Gaussian assumption is based on two fundamental arguments: (i) the Central Limit Theorem, which states that the distribution of the expected value of any random variables converge toward a normal distribution, and (ii) the Maximum Entropy Principle, which identifies the Gaussian family as the Shannon information maximiser under the condition of known mean and covariance (Jaynes, 1978). Also, from a mathematical point of view, the Gaussian distribution can be expressed entirely by its first- and second-order statistics, which simplifies the KFT's algorithms.

Algorithm (1) presents the overall procedure of state estimation with a LKF.

Algorithm 1: Linear Kalman Filter Algorithm

Initialise $\mathbf{x}_0$ and $\mathbf{P}_0$
Set the hyperparameters $\mathbf{Q}$ and $\mathbf{R}$
$\mathbf{x} = [\mathbf{x}_0]$
$\mathbf{P} = [\mathbf{P}_0]$
$k = 1$

while True:

$\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}\mathbf{x}_{k-1}$
$\mathbf{P}_{k|k-1} = \mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^\top + \mathbf{Q}$
$\hat{\mathbf{y}}_k = \mathbf{H}\hat{\mathbf{x}}_{k|k-1}$
$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^\top(\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^\top + \mathbf{R})^{-1}$
$\mathbf{x}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k)$
$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_{k|k-1}$
$k = k + 1$

The LKF algorithm is first initialised with a state vector $\mathbf{x}_0$ and its associated variance $\mathbf{P}_0$. The matrix $\mathbf{P}_0$ is usually set as a diagonal matrix and represents our confidence in the initial values. Low confidence in these $\mathbf{x}_0$ is expressed with large values ($\mathbf{P}_{i,j} >> 1 \ \forall \ i = j$) , whereas high confidence has lower values ($\mathbf{P}_{i,j} << 1 \ \forall \ i = j$).

To condition the KFT estimations, hyperparameters $\mathbf{Q}$ and $\mathbf{R}$ are used. They represent the noise covariance matrix for the process and measurement errors, respectively. The matrix $\mathbf{P}$ is the covariance error associated with the mean state variables in $\mathbf{x}_0$. Thus, setting the $\mathbf{P}$ diagonal with large values imposes a higher level of uncertainty than smaller values. However, in many practical implementations, these confidence levels are unquantified and

treated as tuning parameters (Naya et al., 2023; Ghorbani et al., 2024). Although in practice these hyperparameters are found by trial and error, in this work we use the optimisation procedure in (Ghorbani et al., 2024) to extract these values. The method used a physic-aware loss function within the black-box optimiser NOMAD of (Audet et al., 2022) to reduce the residual between the estimates and the true physic model.

In Algorithm (1) , the prediction step consists of lines 1 to 3 and sets the prior state vector and the covariance matrix with the transition matrix. Then, at the update step from lines 4 to 6, the prior-state vector is compared with the sensor data. The update is done at lines 5 and 6 with the so-called Kalman gain obtained in line 4. Thus, the algorithm iteratively minimises the covariance matrix $\mathbf{P}$ based on the initial confidence according to the initial states.

2.1.3 Unscented Kalman Filter

The linearisation steps in the LKF procedure can lead to numerical errors and divergence. The UKF procedure aims to overcome the LKF's limits. Thus, the initial reasoning of Julier et al. (1995) behind the nonlinear filter was that approximating a Gaussian distribution is simpler than approximating an arbitrary nonlinear function. The idea is to avoid complex derivations and to use a minimal, deterministically chosen and fixed number of points to approximate the mean and the covariance of the unknow distribution (Särkkä, 2011). These points are called sigma-points and are used to propagate through the nonlinear transition and measurement functions to approximate the mean and variance of the predicted state and measurement vectors (unscented transform). These sigma-points are calculated by considering an initial state-vector $\mathbf{x}_0$ and their associated variances in $\mathbf{P}_0$. The sigma-points are found by solving the following equations:

$$\sum_{i=0}^{2N} \omega_i = 1, \tag{2.3}$$

$$\sum_{i=0}^{2N} \omega_i \chi_i = \mathbf{x}_0, \tag{2.4}$$

$$\sum_{i=0}^{2N} \omega_i (\chi_i - \mathbf{x}_0)(\chi_i - \mathbf{x}_0)^\top = \mathbf{P}_0, \tag{2.5}$$

where N is the number of unknown, ω are weights factors and χ the so-called sigma-points. Once Equations (2.3-2.5) are solved, the sigma-points and the weights are defined as:

$$\chi_0 = \mathbf{x}_0, \tag{2.6}$$

$$\chi_i = \mathbf{x}_0 + \left(\sqrt{\mathbf{P}_0(N+\lambda)}\right)_i \quad 1 \leq i \leq N, \tag{2.7}$$

$$\chi_i = \mathbf{x}_0 - \left(\sqrt{\mathbf{P}_0(N+\lambda)}\right)_{i-N} \quad N+1 \leq i \leq 2N, \tag{2.8}$$

$$\omega_0^m = \frac{\lambda}{N+\lambda}, \tag{2.9}$$

$$\omega_0^c = \frac{\lambda}{N+\lambda} + (1 - \alpha_\lambda^2 + \beta_\lambda), \tag{2.10}$$

$$\omega_i^m = \omega_i^c = \frac{1}{2(N+\lambda)}, \quad 1 \leq i \leq 2N, \tag{2.11}$$

in which λ is a scaling parameter defined as:

$$\lambda = \alpha_\lambda^2(N+\kappa) - N, \tag{2.12}$$

where α_λ corresponds to the spread of the sigma-points around the mean $\mathbf{x}_0$ and is usually set as $0 \leq \alpha_\lambda < 1$. κ is another scaling parameter usually set to $3 - N$ (Ghorbani and Cha, 2018) and β_λ is the *a priori* knowledge of the Gaussian distribution and is usually set to 2 for optimality if and only if it is under Gaussian prior (Wan and van der Merwe, 2001; Ghorbani, 2021). In a classical UT, $2N+1$ sigma points are propagated to map the target domain. For Equations (2.7 and 2.8), one can note the need to evaluate the "square root" of the covariance matrix. Given that covariance matrices are positive semi-definite, we can express:

$$\mathbf{P} = \mathbf{L}\mathbf{D}\mathbf{L}^\top, \tag{2.13}$$

where $\mathbf{L}$, $\mathbf{L}^\top$ and $\mathbf{D}$ are respectively a lower triangular, its transpose and a diagonal matrices. We can show that we can solve the square-root of $\mathbf{P}$ via the Choleski factorisation of the form $\mathbf{P} = \mathbf{S}\mathbf{S}^\top$ such that:

$$\mathbf{P} = \mathbf{L}\mathbf{D}\mathbf{L}^\top = \mathbf{L}\sqrt{\mathbf{D}}\sqrt{\mathbf{D}}^\top\mathbf{L}^\top = \mathbf{L}\sqrt{\mathbf{D}}\left(\mathbf{L}\sqrt{\mathbf{D}}\right)^\top = \mathbf{S}\mathbf{S}^\top, \tag{2.14}$$

where $\mathbf{S}$ is a lower triangular matrix. As in the LKF's algorithm, we assume a discrete-time state-space system defined with the same assumptions but for a nonlinear model:

$$\mathbf{x}_k = \mathcal{M}(\mathbf{x}_{k-1}, \boldsymbol{\theta}) + \mathbf{v}_k, \quad \mathbf{v}_k \sim \mathcal{N}(0, \mathbf{Q}) \tag{2.15}$$

$$\hat{\mathbf{y}}_k = \mathcal{H}(\mathbf{x}_k, \boldsymbol{\theta}) + \mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{R}) \tag{2.16}$$

where $\mathcal{M}(\mathbf{x}_{k-1}, \boldsymbol{\theta})$ and $\mathcal{H}(\mathbf{x}_k, \boldsymbol{\theta})$ are the transient and the measurement functions, respectively, in which $\mathbf{x}$ is the state vector and $\boldsymbol{\theta}$, the parameter vector. The transient and measurement functions are added to their associated zero-mean white noise $\mathbf{v}_k$ and $\mathbf{w}_k$. Hence, the general Algorithm (2) presents the iterative structure of the UKF.

Algorithm 2: Unscented Kalman Filter Algorithm

Initialise $\mathbf{x}_0$ and $\mathbf{P}_0$
Set the hyperparameters $\mathbf{Q}$, $\mathbf{R}$ and λ
$\mathbf{x} = [\mathbf{x}_0]$
$\mathbf{P} = [\mathbf{P}_0]$
$k = 1$
while True:

1 $\quad \boldsymbol{\chi}_{k-1} = [\mathbf{x}_{k-1}, \quad \mathbf{x}_{k-1} \pm \sqrt{(N+\lambda)\mathbf{P}_{k-1}}]$
2 $\quad \boldsymbol{\chi}_{k|k-1} = \mathcal{M}(\boldsymbol{\chi}_{k-1}, \ \theta)$
3 $\quad \hat{\mathbf{x}}_{k|k-1} = \sum_{i=0}^{2N} \omega_i^m \boldsymbol{\chi}_{i,k|k-1}$
4 $\quad \mathbf{P}_{k|k-1}^{XX} = \sum_{i=0}^{2N} \omega_i^c [\boldsymbol{\chi}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}][\boldsymbol{\chi}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}]^\top$
5 $\quad \mathbf{P}_{k|k-1} = \mathbf{P}_{k|k-1}^{XX} + \mathbf{Q}$
6 $\quad \boldsymbol{\upsilon}_{k|k-1} = \mathcal{H}(\boldsymbol{\chi}_{k|k-1}, \theta)$
7 $\quad \hat{\mathbf{y}}_k = \sum_{i=0}^{2N} \omega_i^m \boldsymbol{\upsilon}_{i,k|k-1}$
8 $\quad \mathbf{P}_k^{YY} = \sum_{i=0}^{2N} \omega_i^c [\boldsymbol{\upsilon}_{i,k|k-1} - \hat{\mathbf{y}}_k][\boldsymbol{\upsilon}_{i,k|k-1} - \hat{\mathbf{y}}_k]^\top + \mathbf{R}$
9 $\quad \mathbf{P}_k^{XY} = \sum_{i=0}^{2N} \omega_i^c [\boldsymbol{\chi}_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}][\boldsymbol{\upsilon}_{i,k|k-1} - \hat{\mathbf{y}}_k]^\top$
10 $\quad \mathbf{K}_k = \mathbf{P}_k^{XY} \left(\mathbf{P}_k^{YY}\right)^{-1}$
11 $\quad \mathbf{x}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k)$
12 $\quad \mathbf{P}_k = \mathbf{P}_{k|k-1}^{XX} - \mathbf{K}_k \mathbf{P}_k^{YY} \mathbf{K}_k^\top$
$\quad k = k + 1$

In Algorithm (2) the two-step iterative procedure can be observed once more. The sigma points are propagated through the nonlinear prior knowledge from lines 1 to 6. Then, these estimations are used to evaluate the Kalman gain at line 9. Lastly, from lines 10 to 11, the mean and covariance of the target domain are updated with the sensor measurements.

In addition to its capacity to estimate nonlinear states, the UKF can also solve the inverse problem by considering the augmented state transition function defined in Ghorbani et al. (2024) as $\mathcal{M}^a = [\mathcal{M}, \mathcal{I}]$ with $\mathcal{I}$ the identity function. This augmented state function increases the number of unknowns N in the system and consequently the dimensionality of the model. Moreover, as the state vector increases, $\mathbf{Q}$ is augmented accordingly, but $\mathbf{R}$ remains the same. For high values of N, the computation time of the UKF can be tedious. Using a data-driven technique instead of a physics-based model could overcome this drawback.

2.1.4 Adaptive Kalman Filter Covariance and Hyperparameter Tuning

The biggest problem with classical KFT, such as LKF and UKF, is the setup of the proper matrices $\mathbf{Q}$ and $\mathbf{R}$ (Akhlaghi et al., 2017). The performance of the Kalman filter is highly dependent on the proper choice of the latter two matrices. Moreover, in lab-scale mechanical systems, the white noise assumption to describe the process and the measurement error can be inaccurate. These errors can vary in time, and the assumption of white noise can then lead to suboptimal performance of the filter. Hence, one solution is to use adaptive Kalman-filter covariance. Adaptive methods facilitate the alignment of estimates with true sensor data through dynamic adjustment of error covariances. Adaptive methods have been used in Vettori et al. (2023) for the application of structural dynamics and in Lan et al. (2024) to improve target tracking. In Akhlaghi et al. (2017), an adaptive EKF is used to adapt and adjust $\mathbf{Q}$ and $\mathbf{R}$ with forgetting factors α_Q and α_R. Forgetting factors, restricted to the interval [0, 1], serve as an approximation technique for covariance matching through a residual-based methodology (Akhlaghi et al., 2017). The idea of Akhlaghi et al. (2017) was adapted to an UKF algorithm in Ghorbani et al. (2024) and Dollon et al. (2025) such that:

$$\mathbf{Q}_{k+1} = \alpha_Q \mathbf{Q}_k + (1 - \alpha_Q)(\mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1}))(\mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1}))^\top, \tag{2.17}$$

$$\mathbf{R}_{k+1} = \alpha_R \mathbf{R}_k + (1 - \alpha_R)((\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1})(\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1})^\top + \mathbf{P}^{YY}_{k+1})^\top, \tag{2.18}$$

where $\mathbf{y}_{k+1}$ is the measurement, $\hat{\mathbf{y}}_{k+1}$ the weighted propagation of the measurement, $\mathbf{K}_{k+1}$ the Kalman gain and $\mathbf{P}^{YY}$ the covariance matrix of Algorithm (2) at time step $k+1$. In Algorithm (2), Equations (2.17) and (2.18) are solved after line 12 in the iterative procedure. The relaxation scheme could allow us to correlate the different states and parameter estimates with the true sensor measurement of an actual mechanical system.

However, this adaptive scheme adds two new hyperparameters to the UKF. In practical applications, finding the appropriate $\mathbf{Q}$, $\mathbf{R}$, α_Q and α_R for optimal KFT performance is done by trial and error, which is time-consuming and tedious. Moreover, it often assumes that

the covariance matrices are diagonal or even scalar, which imposes uncorrelated behaviour on the stochastic model. In this work, we use the NOMAD black-box optimiser of Audet et al. (2022) to find the best set of error covariance matrices and their forgetting factor as in Ghorbani et al. (2024). NOMAD is a mesh adaptive direct search optimiser that searches for the best settings to put in relation to a predefined loss function (Dollon et al., 2025). We use a Physic-Aware (PA) loss function within NOMAD to ensure the compliance of the estimate with the physical model. Unlike the traditional optimisation method that minimises the error between the predicted and measurement observations, the method of Ghorbani et al. (2024) aims to minimise the identified states from the known physical equations.

2.1.5 Data-driven Kalman Filter

For dimensionality reduction and to accelerate the algorithm's computational speed, we merge the DMD inside a LKF. We consider this adaptation of the LKF as a parallel module to the UKF. The DMD is inherently data-driven and is based on k snapshots of data such that:

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_0 & \mathbf{x}_1 & \cdots & \mathbf{x}_{k-1} \\ | & | & & | \end{bmatrix}, \tag{2.19}$$

$$\mathbf{X}' = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_k \\ | & | & & | \end{bmatrix}. \tag{2.20}$$

where $\mathbf{x}_i$ for $0 \leq i \leq k+1$ is a data column vector at different time steps Δt_{DMD}, and $\mathbf{X}$ and $\mathbf{X}'$ are snapshot matrices. With DMD, we aim to reduce the physics-based model with a modal decomposition by finding the best-fit linear operator $\mathbf{A}$ that correlates both snapshot matrices:

$$\mathbf{X}' \approx \mathbf{A}\mathbf{X} \tag{2.21}$$

$$\mathbf{x}_k \approx \mathbf{A}\mathbf{x}_{k-1} \tag{2.22}$$

From Algorithm (1) line 1, we note a similarity with Equation (2.21). The best-fit operator $\mathbf{A}$ leads to the estimation of the future state one time step ahead. To determine $\mathbf{A}$, we use the DMD code given in Brunton and Kutz (2022) based on a singular value decomposition (SVD) of $\mathbf{X}$:

$$\mathbf{X} \approx \tilde{\mathbf{U}}_{N\times r}\tilde{\mathbf{\Xi}}_{r\times r}\tilde{\mathbf{V}}^{\top}_{N\times r}, \tag{2.23}$$

where $\tilde{\mathbf{U}}$ is the reduced spatial modes of $\mathbf{X}$, $\tilde{\mathbf{\Xi}}$ the mode's energy, $\tilde{\mathbf{V}}^\top$ the modal time dynamics transpose and r the reduced rank of the matrices. Here, $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ both satisfy $\tilde{\mathbf{U}}\tilde{\mathbf{U}}^\top = \mathbf{I}$ and $\tilde{\mathbf{V}}\tilde{\mathbf{V}}^\top = \mathbf{I}$ since they are orthonormal. Thus, by inserting Equation (2.23) in (2.21) and by calculating its pseudo-inverse, we find the best-fit operator $\mathbf{A}$:

$$\mathbf{A} = \mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Xi}}^{-1}\tilde{\mathbf{U}}^\top, \tag{2.24}$$

and we can reduce $\mathbf{A}$ to an $r \times r$ sized matrix by projecting $\mathbf{A}$ on $\tilde{\mathbf{U}}$:

$$\tilde{\mathbf{A}} = \tilde{\mathbf{U}}^\top\mathbf{A}\tilde{\mathbf{U}} = \tilde{\mathbf{U}}^\top\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Xi}}^{-1}, \tag{2.25}$$

As noted in Brunton and Kutz (2022), $\tilde{\mathbf{A}}$ has the same non-zero eigenvalues as $\mathbf{A}$. It means that the whole system can be expressed via the reduced best-fit operator $\tilde{\mathbf{A}}$. Then, Equation (2.22) becomes:

$$\tilde{\mathbf{x}}_k = \tilde{\mathbf{A}}\tilde{\mathbf{x}}_{k-1} \tag{2.26}$$

where $\tilde{\mathbf{x}}$ is the reduced state vector. The reduced KFT is presented in Algorithm (3):

Algorithm 3: Linear Dynamic Mode Decomposition Kalman Filter Algorithm

Set snapshots $\mathbf{X}$ and $\mathbf{X}'$
$\tilde{\mathbf{U}}$,$\tilde{\mathbf{\Xi}}$, $\tilde{\mathbf{V}} = svd(\mathbf{X})$
$\tilde{\mathbf{A}} = \tilde{\mathbf{U}}^\top\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Xi}}^{-1}$
Initialise $\mathbf{x}_0$ and $\mathbf{P}_0$
Set the hyperparameters $\mathbf{Q}$ and $\mathbf{R}$
$\mathbf{x} = [\mathbf{x}_0]$
$\mathbf{P} = [\mathbf{P}_0]$
$k = 1$

```
while True:
```

$\hat{\mathbf{x}}_{k|k-1} = \tilde{\mathbf{A}}\mathbf{x}_{k-1}$
$\mathbf{P}_{k|k-1} = \tilde{\mathbf{A}}\mathbf{P}_{k-1}\tilde{\mathbf{A}}^\top + \mathbf{Q}$
$\hat{\mathbf{y}}_k = \mathbf{H}\mathbf{x}_{k|k-1}$
$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^\top(\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^\top + \mathbf{R})^{-1}$
$\mathbf{x}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k)$
$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_{k|k-1}$
$k = k + 1$

Algorithm (3) is similar to Algorithm (1). The only difference is the dimension reduction through $\tilde{\mathbf{A}}$ and the assumption that the measurement function $\mathbf{H}$ is equivalent to $\tilde{\mathbf{U}}\tilde{\mathbf{\Xi}}$. The LKF-DMD model iteratively updates the reduced mode vector by minimising the second-order statistic until no more data points are available.

2.2 The Pipe Conveying Fluid

As an academic problem with rich dynamic behaviour, the pipe conveying fluid with its simple experimental setup and equations of motion serves as an excellent candidate for testing, verifying, and validating our digital twin framework. In practice, the direct application of a pipe conveying fluid can be seen in oil risers (Kharami et al., 2024), deep-sea mining systems (Daneshmand et al., 2021), biomedical devices (Ghavanloo et al., 2010), and even in underwater robotic propeller systems (Paidoussis, 1976; Dai et al., 2022).

2.2.1 Overview

Flow-induced vibrations in a cantilever pipe conveying fluid system is a well-known FSI problem of almost 90 years old (Païdoussis, 2022). This simple setup can display a wide range of dynamics by only increasing the flow rate. Initial studies concern the linear behaviour of the pipe in which the water flow through the pipe tends to dampen and stabilise its perturbed oscillation. Nonlinear observation of the pipe was first observed experimentally in Benjamin (1961) on an articulated pipe and opened the door to a vast exploration on the subject. Beyond a critical flow velocity, the pipe reached a limit-cycle behaviour which could happen only through a nonlinear contribution to the system Paidoussis (2014). This transient point, known as the Hopf bifurcation, generates this flutter (Paidoussis, 2014) and appears when the damping in the pipe becomes negative (Modarres-Sadeghi, 2021). Lastly, increasing the flow rate further makes the pipe behave chaotically (Paidoussis, 2014).

The nonlinear dynamic behaviour of the pipe conveying fluid system was first studied in Holmes (1977) and is now known as a benchmark paper on two-dimensional (2D) planar nonlinear dynamics (Semler, 1996). For simplicity in the analyses and derivations, the pipe was constrained onto a plane. 3D analysis was performed in Lundgren et al. (1979) on a non-planar pipe with an end nozzle. Bajaj and Sethna (1984) contributed to the 3D analysis by exploring the behaviour of the pipe for a plain cantilever one. However, in the fluid-structure interaction community, the most complete equations for a 3D cantilevered tube are those derived by Wadham-Gagnon et al. (2007). The latter were derived considering the small lateral displacement, approximating the centerline curvature of the pipe with a Taylor series expansion of third order (Chen et al., 2020; Wadham-Gagnon et al., 2007). The equations were validated in Modarres-Sadeghi et al. (2008) for a plain cantilever pipe, in Païdoussis et al. (2007) for a pipe with intermediate spring support, and in Modarres-Sadeghi et al. (2007) for a pipe with an end mass. The only limit of the derived equations is the Taylor expansion approximation, which restricts the analysis to observations of small amplitude

(Chen et al., 2020; Païdoussis, 2022). The analysis of large amplitude requires a geometrical model that takes into account the curvature of the pipe (Chen et al., 2020, 2021; Chen, 2022; Farokhi et al., 2021; Dehrouyeh-Semnani, 2023). Chen et al. (2020, 2021); Chen (2022) demonstrated the superior performance of their model in capturing both small- and large-amplitude oscillations, compared to the work of Wadham-Gagnon et al. (2007). However, despite the high geometric fidelity of the model, it is computationally inefficient, even when it uses only half the number of degrees of freedom employed by Wadham-Gagnon (Chen et al., 2021). Li et al. (2024a) and Sarkar and Paidoussis (2002) managed to overcome this computational burden by reducing the complexity of the plain cantilevered and supported end models with a data-driven algorithm through spectral submanifolds and POD, respectively. However, both studies only concerned the 2D formulations of the pipe.

In this work, the pipe conveying fluid setup is the one built and validated by Demenois (2022b). The latter applied a physics-informed neural network (PINN) in the linear region of the pipe to identify the flow rate through the pipe. Two high-speed cameras are used to reconstruct the movement of the pipe in the linear region. PINN has shown promising results in its ability to accurately and rapidly estimate the unknown flow rate with a relative error of less than 8%. However, the model was limited by the prior knowledge injected inside the PINN, which restricted the algorithm to work exclusively in the linear region. Also, training the PINN can be complex and time consuming if it gets stuck within a local minimum. Finally, the virtual world and the physical entity were not connected by a real-time network connection. Since most engineering systems, such as hydro turbines, exhibit nonlinear behaviour, relying solely on a linear model to test our algorithms is inadequate. Furthermore, the nonlinear model enables continuous monitoring of the pipe deflection across different dynamic regimes, which the linear model cannot capture. In order to tackle the nonlinear region, we use the equations derived from Wadham-Gagnon et al. (2007) that are used for a plain cantilever pipe. Building on the work of Modarres-Sadeghi et al. (2008), we adapt their original Fortran code to Python. Leveraging modern computational capabilities and advances in multi-model approaches, our aim is to incorporate both the physical model developed by Modarres-Sadeghi et al. (2008) and the actual experimental pipe setup in our laboratory into a unified DT framework.

The pipe conveying fluid as an experimental test rig for DTs has already been tested in the marine riser industries (Kharami et al., 2024; Lee et al., 2024). In Kharami et al. (2024), to construct a DT for marine risers, vortex-induced vibration fatigue was studied on an immersed pipe conveying fluid and predicted in real-time. In Lee et al. (2024), they analysed and validated their virtual model to estimate fatigue life. In both cases, their framework combined multiple machine learning models such as PINN, long-short-term memory networks,

and deep operator networks to predict fatigue in real-time. In the field of predictive maintenance, machine learning techniques have been successively applied to rigid pipes subject to vibration (Tang et al., 2024). With proper training and models, they can predict pressure drop, erosion failure, and leak identification (Tang et al., 2024). Hence, the pipe conveying fluid model is an interesting tool for developing a DT. Only by observing its displacement can we easily derive its velocity, acceleration, and elongation to fuel our virtual model. Moreover, multiple hidden parameters such as the Young modulus, the Kelvin-Voigt damping, the flow rate, the air friction, and the fixation parameters can be used to test our UKF algorithm to solve the inverse problem.

2.2.2 The Nonlinear Model

Here, the nonlinear model is the one developed in Wadham-Gagnon et al. (2007) where an extensive and detailed derivation of the equations can be found. The given model assumes: (i) an incompressible fluid; (ii) a constant flow velocity and no pulsation; (iii) the Euler-Bernoulli cantilever beam mode shape to describe the pipe behaviour; (iv) small strain; (v) neglected rotary inertia and shear deformation; (vi) inextensibility of the pipe reducing the 3D dynamics down to two variables. To these assumptions, we suppose (vii) the cantilevered fixation perfectly rigid and (viii) the air friction negligible. From the results derived from Wadham-Gagnon et al. (2007), the only noticeable differences are the addition of the material damping (E^*) approximate via the Kelvin-Voigt assumption and the removal of the variables associated with the spring supports and the added mass. Here, the pipe is aligned given the Cartesian coordinates xyz in which the origin is fixed at the pipe clamp and the z axis points toward the tip of the pipe. The pipe oscillates in the transverse planes xz and yz, in which these transverse displacements can be expressed in their dimensionless form, respectively, by η and ζ. Thus, the dimensionless formulation of the 3D nonlinear pipe conveying fluid is in

the dimensionless xz plane:

$$\begin{aligned}
&\boldsymbol{\eta}'''' + \alpha\dot{\boldsymbol{\eta}}'''' + \ddot{\boldsymbol{\eta}} + 2u\sqrt{\beta}\dot{\boldsymbol{\eta}}' + u^2\boldsymbol{\eta}'' + \gamma\boldsymbol{\eta}' - \gamma\boldsymbol{\eta}''(1-\xi) \\
&+ \gamma\left(\frac{1}{2}\boldsymbol{\eta}'^3 + \frac{1}{2}\boldsymbol{\eta}'\boldsymbol{\zeta}'^2\right) - \gamma(1-\xi)\left(\frac{3}{2}\boldsymbol{\eta}'^2\boldsymbol{\eta}'' + \frac{1}{2}\boldsymbol{\zeta}'^2\boldsymbol{\eta}'' + \boldsymbol{\eta}'\boldsymbol{\zeta}''\right) \\
&+ 2u\sqrt{\beta}\left[\boldsymbol{\eta}'^2\dot{\boldsymbol{\eta}}' + \boldsymbol{\eta}'\boldsymbol{\zeta}'\dot{\boldsymbol{\zeta}}' - \boldsymbol{\eta}''\int_\xi^1(\boldsymbol{\eta}'\dot{\boldsymbol{\eta}}' + \boldsymbol{\zeta}'\dot{\boldsymbol{\zeta}}')d\xi\right] \\
&+ u^2\left[\boldsymbol{\eta}'^2\boldsymbol{\eta}'' + \boldsymbol{\eta}'\boldsymbol{\zeta}'' - \boldsymbol{\eta}''\int_\xi^1(\boldsymbol{\eta}'\boldsymbol{\eta}'' + \boldsymbol{\zeta}'\boldsymbol{\zeta}'')d\xi\right] \\
&+ \left[\boldsymbol{\eta}'^2\boldsymbol{\eta}'''' + 4\boldsymbol{\eta}'\boldsymbol{\eta}''\boldsymbol{\eta}''' + \boldsymbol{\eta}''\boldsymbol{\zeta}'''' + 3\boldsymbol{\eta}'\boldsymbol{\zeta}''\boldsymbol{\zeta}''' + \boldsymbol{\eta}''\boldsymbol{\zeta}''' + \boldsymbol{\eta}''\boldsymbol{\zeta}'''\boldsymbol{\zeta}''^2\right] \\
&+ \boldsymbol{\eta}'\int_0^\xi(\dot{\boldsymbol{\eta}}'^2 + \boldsymbol{\eta}'\tilde{\boldsymbol{\eta}}' + \dot{\boldsymbol{\zeta}}'^2 + \xi'\tilde{\boldsymbol{\zeta}}')d\xi \\
&- \boldsymbol{\eta}''\int_0^\xi(\dot{\boldsymbol{\eta}}'^2 + \boldsymbol{\eta}'\tilde{\boldsymbol{\eta}}' + \dot{\boldsymbol{\zeta}}'^2 + \boldsymbol{\zeta}'\tilde{\boldsymbol{\zeta}}')d\xi\ d\xi = 0.
\end{aligned} \tag{2.27}$$

and in the dimensionless yz plane:

$$\begin{aligned}
&\boldsymbol{\zeta}'''' + \alpha\dot{\boldsymbol{\zeta}}'''' + \ddot{\boldsymbol{\zeta}} + 2u\sqrt{\beta}\dot{\boldsymbol{\zeta}}' + u^2\boldsymbol{\zeta}'' + \gamma\boldsymbol{\zeta}' - \gamma\boldsymbol{\zeta}''(1-\xi) \\
&+ \gamma\left(\frac{1}{2}\boldsymbol{\zeta}'^3 + \frac{1}{2}\boldsymbol{\zeta}'\boldsymbol{\eta}'^2\right) - \gamma(1-\xi)\left(\frac{3}{2}\boldsymbol{\zeta}'^2\boldsymbol{\zeta}'' + \frac{1}{2}\boldsymbol{\eta}'^2\boldsymbol{\zeta}'' + \boldsymbol{\zeta}'\boldsymbol{\eta}''\right) \\
&+ 2u\sqrt{\beta}\left[\boldsymbol{\zeta}'^2\dot{\boldsymbol{\zeta}}' + \boldsymbol{\zeta}'\boldsymbol{\eta}'\dot{\boldsymbol{\eta}}' - \boldsymbol{\zeta}''\int_\xi^1(\boldsymbol{\zeta}'\dot{\boldsymbol{\zeta}}' + \boldsymbol{\eta}'\dot{\boldsymbol{\eta}}')d\xi\right] \\
&+ u^2\left[\boldsymbol{\zeta}'^2\boldsymbol{\zeta}'' + \boldsymbol{\zeta}'\boldsymbol{\eta}'' - \boldsymbol{\zeta}''\int_\xi^1(\boldsymbol{\zeta}'\boldsymbol{\zeta}'' + \boldsymbol{\eta}'\boldsymbol{\eta}'')d\xi\right] \\
&+ \left[\boldsymbol{\zeta}'^2\boldsymbol{\zeta}'''' + 4\boldsymbol{\zeta}'\boldsymbol{\zeta}''\boldsymbol{\zeta}''' + \boldsymbol{\zeta}''\boldsymbol{\eta}'''' + 3\boldsymbol{\zeta}'\boldsymbol{\eta}''\boldsymbol{\eta}''' + \boldsymbol{\zeta}''\boldsymbol{\eta}''' + \boldsymbol{\zeta}''\boldsymbol{\eta}'''\boldsymbol{\eta}''^2\right] \\
&+ \boldsymbol{\zeta}'\int_0^\xi(\dot{\boldsymbol{\zeta}}'^2 + \boldsymbol{\zeta}'\tilde{\boldsymbol{\zeta}}' + \dot{\boldsymbol{\eta}}'^2 + \boldsymbol{\eta}'\tilde{\boldsymbol{\eta}}')d\xi \\
&- \boldsymbol{\zeta}''\int_0^\xi(\dot{\boldsymbol{\zeta}}'^2 + \boldsymbol{\zeta}'\tilde{\boldsymbol{\zeta}}' + \dot{\boldsymbol{\eta}}'^2 + \boldsymbol{\eta}'\tilde{\boldsymbol{\eta}}')d\xi\ d\xi = 0.
\end{aligned} \tag{2.28}$$

where the dimensionless parameters are defined in Table 2.1. One can easily remark that Equations (2.27) and (2.28) are a perfect opposite of each other showing the symmetrical nature of the system. Also, as mentioned in Wadham-Gagnon et al. (2007), considering a planar motion in the xz or yz plane results only in the 2D nonlinear formulation developed in Semler (1996). For both equations, the first line corresponds to the linear formulation of the cantilevered pipe conveying fluid, whereas the lines below concern the nonlinear contributions. At low flow rate values, the nonlinear contributions become negligible and the results are similar to those obtained with the linear model (Paidoussis, 2014). The linear terms correspond, respectively, to:

- $\boldsymbol{\eta}'''', \boldsymbol{\zeta}''''$: The flexural rigidity
- $\alpha\dot{\boldsymbol{\eta}}'''', \alpha\dot{\boldsymbol{\zeta}}''''$: The visco-elasticity
- $\ddot{\boldsymbol{\eta}}, \ddot{\boldsymbol{\zeta}}$: The inertia
- $2u\sqrt{\beta}\dot{\boldsymbol{\eta}}', 2u\sqrt{\beta}\dot{\boldsymbol{\zeta}}'$: The Coriolis force
- $u^2\boldsymbol{\eta}'', u^2\boldsymbol{\zeta}''$: The centrifugal force
- $\gamma\boldsymbol{\eta}' - \gamma\boldsymbol{\eta}''(1-\xi), \gamma\boldsymbol{\zeta}' - \gamma\boldsymbol{\zeta}''(1-\xi)$: The Influence of Gravity

As detailed in Demenois (2022b); Paidoussis (2014), the Galerkin discretisation method is applied such that transverse displacements are expressed as:

$$\boldsymbol{\eta}(\xi,\tau) = \sum_{i=1}^{N} \phi_i(\xi)\mathbf{q}_i(\tau), \tag{2.29}$$

$$\boldsymbol{\zeta}(\xi,\tau) = \sum_{i=1}^{N} \psi_i(\xi)\mathbf{p}_i(\tau), \tag{2.30}$$

where $\phi_r(\xi)$,$\psi_r(\xi)$, $\mathbf{q}_i$, $\mathbf{p}_i$ are, respectively, the dimensionless cantilever beam eigenfunctions, the generalised coordinates in the x and y directions. Since the dimensionless eigenfunctions share the same boundary condition as the Euler-Bernoulli problem, one can assume $\phi_i(\xi) = \psi_i(\xi)$. Equations (2.29) can be substituted in Equation (2.27) and each term can be multiplied by the corresponding beam eigenfunction $\phi_j(\xi)$ for $1 \leq j \leq N$. The given products are then integrated in the ξ domain from 0 to 1. Since the eigenfunctions are orthonormal, one can define the Kronecker delta:

$$\delta_{ij} = \int_0^1 \phi_i\phi_j d\xi = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases} \tag{2.31}$$

and extract from the cantilever beam equation:

$$\phi_i'''' = \lambda_i^4\phi_j, \tag{2.32}$$

with λ_i being the cantilever beam eigenfrequencies. Thus, we simplify Equation (2.27) to:

$$\begin{aligned} &\tilde{\mathbf{M}}_{ij}\ddot{\mathbf{q}}_j + \tilde{\mathbf{C}}_{ij}\dot{\mathbf{q}}_j + \tilde{\mathbf{K}}_{ij}\mathbf{q}_j + \mathbf{B}_{ijkl}\mathbf{q}_j\mathbf{q}_k\mathbf{q}_l + \mathbf{D}_{ijkl}(\mathbf{q}_j\mathbf{q}_k\dot{\mathbf{q}}_l + \mathbf{q}_j\mathbf{p}_k\dot{\mathbf{p}}_l) \\ &+ \mathbf{E}_{ijkl}(\mathbf{q}_j\dot{\mathbf{q}}_k\dot{\mathbf{q}}_l + \mathbf{q}_j\dot{\mathbf{q}}_k\ddot{\mathbf{q}}_l + \mathbf{q}_j\dot{\mathbf{p}}_k\dot{\mathbf{p}}_l + \mathbf{q}_j\dot{\mathbf{p}}_k\ddot{\mathbf{p}}_l) + \mathbf{H}_{ijkl}\mathbf{q}_j\mathbf{p}_k\mathbf{p}_l = 0 \end{aligned} \tag{2.33}$$

Table 2.1 Dimensionless parameter of Equations (2.27) and (2.28)

Formulation	Signification
$\xi = \frac{z}{L}$	dimensionless vertical coordinates
$\boldsymbol{\eta} = \frac{x}{L}$	dimensionless displacement in the xzplane
$\boldsymbol{\zeta} = \frac{y}{L}$	dimensionless displacement in the yz plane
$\tau = \left(\frac{EI}{M+m}\right)^{\frac{1}{2}} \frac{t}{L^2}$	dimensionless time
$u = \left(\frac{M}{EI}\right)^{\frac{1}{2}} LU$	dimensionless flow rate
$\beta = \frac{M}{M+m}$	Mass parameter
$\gamma = \frac{(M+m)L^2}{EI} g$	Gravity parameter
$\alpha = \left(\frac{I}{E(M+m)}\right)^{\frac{1}{2}} \frac{E^*}{L^2}$	Damping parameter
$\dot{} = \frac{\partial}{\partial t}$	Time derivative
$' = \frac{\partial}{\partial z}$	Space derivative

and by doing the same mathematical steps for the second direction, Equation (2.28) becomes:

$$\begin{aligned}\tilde{\mathbf{M}}_{ij}\ddot{\mathbf{p}}_j + \tilde{\mathbf{C}}_{ij}\dot{\mathbf{p}}_j + \tilde{\mathbf{K}}_{ij}\mathbf{p}_j + \mathbf{B}_{ijkl}\mathbf{p}_j\mathbf{p}_k\mathbf{p}_l + \mathbf{D}_{ijkl}(\mathbf{p}_j\mathbf{p}_k\dot{\mathbf{p}}_l + \mathbf{p}_j\mathbf{q}_k\dot{\mathbf{q}}_l) \\ + \mathbf{E}_{ijkl}(\mathbf{p}_j\dot{\mathbf{p}}_k\dot{\mathbf{p}}_l + \mathbf{p}_j\dot{\mathbf{p}}_k\ddot{\mathbf{p}}_l + \mathbf{p}_j\dot{\mathbf{q}}_k\dot{\mathbf{q}}_l + \mathbf{p}_j\dot{\mathbf{q}}_k\ddot{\mathbf{q}}_l) + \mathbf{H}_{ijkl}\mathbf{p}_j\mathbf{q}_k\mathbf{q}_l = 0\end{aligned} \tag{2.34}$$

Where $\tilde{\mathbf{M}}_{ij}$, $\tilde{\mathbf{C}}_{ij}$ and $\tilde{\mathbf{K}}_{ij}$ are, respectively, the linear mass, damping, and stiffness matrices of size $N \times N$. Here, the Newton convention is used for simplicity and in agreement with the formulation of Wadham-Gagnon et al. (2007). Thus, $\mathbf{B}_{ijkl}$, $\mathbf{D}_{ijkl}$, $\mathbf{E}_{ijkl}$ and $\mathbf{H}_{ijkl}$ are the nonlinear contribution tensors of size $N \times N \times N \times N$. Their formulation can be found in Appendix A. Following Chen et al. (2021) method, we merge both Equations (2.33) and (2.34) into a second-order form:

$$\begin{bmatrix} \tilde{\mathbf{M}}_{ij} + \mathbf{E}_{ijkl}\mathbf{q}_j\dot{\mathbf{q}}_k & \mathbf{E}_{ijkl}\mathbf{q}_j\dot{\mathbf{p}}_k \\ \mathbf{E}_{ijkl}\mathbf{p}_j\dot{\mathbf{q}}_k & \tilde{\mathbf{M}}_{ij} + \mathbf{E}_{ijkl}\mathbf{p}_j\dot{\mathbf{p}}_k \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \ddot{\mathbf{p}} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{C}}_{ij} & 0 \\ 0 & \tilde{\mathbf{C}}_{ij} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{p}} \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{K}}_{ij} & 0 \\ 0 & \tilde{\mathbf{K}}_{ij} \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{F}(\mathbf{q}) \\ \mathbf{F}(\mathbf{p}) \end{bmatrix} \tag{2.35}$$

$$\mathbf{M}\ddot{\boldsymbol{r}} + \mathbf{C}\dot{\boldsymbol{r}} + \mathbf{K}\boldsymbol{r} = -\mathbf{F} \tag{2.36}$$

where:

$$\begin{bmatrix} \mathbf{F}(\mathbf{q}) \\ \mathbf{F}(\mathbf{p}) \end{bmatrix} = \begin{bmatrix} \mathbf{B}_{ijkl}\mathbf{q}_j\mathbf{q}_k\mathbf{q}_l + \mathbf{D}_{ijkl}(\mathbf{q}_j\mathbf{q}_k\dot{\mathbf{q}}_l + \mathbf{q}_j\mathbf{p}_k\dot{\mathbf{p}}_l) + \mathbf{E}_{ijkl}(\mathbf{q}_j\dot{\mathbf{q}}_k\dot{\mathbf{q}}_l + \mathbf{q}_j\dot{\mathbf{p}}_k\dot{\mathbf{p}}_l) + \mathbf{H}_{ijkl}\mathbf{q}_j\mathbf{p}_k\mathbf{p}_l \\ \mathbf{B}_{ijkl}\mathbf{p}_j\mathbf{p}_k\mathbf{p}_l + \mathbf{D}_{ijkl}(\mathbf{p}_j\mathbf{p}_k\dot{\mathbf{p}}_l + \mathbf{p}_j\mathbf{q}_k\dot{\mathbf{q}}_l) + \mathbf{E}_{ijkl}(\mathbf{p}_j\dot{\mathbf{p}}_k\dot{\mathbf{p}}_l + \mathbf{p}_j\dot{\mathbf{q}}_k\dot{\mathbf{q}}_l) + \mathbf{H}_{ijkl}\mathbf{p}_j\mathbf{q}_k\mathbf{q}_l \end{bmatrix}.$$

Then, we can rearrange the term to get a discretised first-order form:

$$\dot{\boldsymbol{\mathcal{R}}} = \begin{bmatrix} \mathbf{O} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix} \boldsymbol{\mathcal{R}} - \begin{bmatrix} \mathbf{O} \\ \mathbf{M}^{-1}\mathbf{F} \end{bmatrix} \tag{2.37}$$

With

$$\boldsymbol{\mathcal{R}} = \begin{bmatrix} \boldsymbol{r} \\ \dot{\boldsymbol{r}} \end{bmatrix} = \begin{bmatrix} \mathbf{q} \\ \mathbf{p} \\ \dot{\mathbf{q}} \\ \dot{\mathbf{p}} \end{bmatrix} \tag{2.38}$$

Here, the discrete first-order form of the pipe conveying fluid allows for a general interpretation and mathematical formulation of the oscillatory motion. This formulation can be scaled to other mechanical systems, such as hydro turbines.

2.2.3 Model Verification - Convergence Analysis and Comparison

Equation (2.37) can be solved using any implicit discretisation method. In both the right and left parts of Equation (2.37), $\dot{\mathbf{q}}$ and $\dot{\mathbf{p}}$ are present. In Chen et al. (2021) a fourth-order Runge-Kutta with adaptive time steps is used. In Modarres-Sadeghi et al. (2008) the Houbolt method is used to solve the implicit system of equations. In this work, a Crank-Nicolson (CN) method is used to solve Equation (2.37). To verify the convergence of our nonlinear model, we first test the algorithm to define the minimal beam's mode shape to extract reliable results. As mentioned in Modarres-Sadeghi (2021), when using the Galerkin method, a convergence test is necessary because (i) the modes used are the ones for a beam with no fluid inside, meaning the true pipe's modes are not necessarily the same. (ii) We solve a system of coupled equations. Therefore, even a small contribution of higher modes can affect the lower modes of the coupled equations. Then, we perform a convergence order analysis on the CN scheme to ensure that the method is well built. The CN method in the FSI problem is not a new topic and, in pipe conveying fluid system, it was used mainly to solve the problem of the bloodstream (Oyekole et al., 2018) or the rigid pipe system (Guo et al., 2020). The CN method is a second-order implicit scheme in both space and time. Here, we use an implicit scheme, since the explicit method required a drastically small time step to achieve convergence. However, in implicit scheme, a wrong definition of Δt_{CN} can induce suboptimal time convergence rates of the scheme or, in extreme cases, divergence in the model and unwanted oscillations (Oyekole et al., 2018; Wick, 2013). To verify the minimal number of modes of the pipe, we set $\Delta t_{CN} = 0.003$ and discretise the pipe in space with $\Delta h = 1e-4$. The initial conditions in $\boldsymbol{\mathcal{R}}$ are set to zero except for : $q_0 = 0.01, p_0 = 0.0006$

and $\dot{q}_0 = 0.01$. We start with $N = 2$ and increase the number of mode shapes until the amplitude in both x and y stabilises. The convergence test was performed for two cases (i) $u = 8, \gamma = \alpha = 0, \beta = 0.2$ and (ii) $u = 17, \gamma = 366, \alpha = 0.0057, \beta = 0.12$.

In the case of the convergence order analysis, since no analytical solution exists for the nonlinear pipe model, we use a propagation method by defining the exact solution as a fine mesh with $\Delta t_{CN} = 1/5120$. We then use the `scipy` function `interp1d` in Python to interpolate the missing points in the coarser mesh with a cubic spline. Thus, we make sure that both vectors are of the same length. The order of convergence O of the CN scheme is obtain as:

$$O = \left(\frac{log\left(\frac{\epsilon_{CN,coarser}}{\epsilon_{CN,coarser}}\right)}{log\left(\frac{\Delta t_{CN,coarser}}{\Delta t_{CN,finer}}\right)} \right), \tag{2.39}$$

where ϵ_{CN} is the numerical L2 error between the hand-made exact solution and the calculated solution. We first start the analysis with $\Delta t_{CN} = 1/20$ and then divide this initial value by two until the observed convergence order $O \to 2$. We fix $\Delta h = 1e-4$ and consider the same initial values as mentioned earlier. Figure 2.1 shows both the convergence analysis of the nonlinear model of the pipe conveying fluid.

In Figure 2.1**a)**, for a simple case without gravity and Kelvin-Voigt damping parameters, $N = 4$ is sufficient to solve Equation (2.37). However, increasing the nonlinear complexity, we see that a higher number of beam modes is necessary with $N = 6$. Hence, as we add the contribution of dimensionless parameters to Equation (2.37), the nonlinear coefficients have a contribution that cannot be ignored and a higher number of beam mode shapes is necessary to reach convergence. However, solving the nonlinear model for a number of modes greater than $N = 8$ is computationally expensive and does not drastically change the fidelity of the response. Moreover, integrating the nonlinear model within the UKF implies solving this model $2N+1$ times due to the propagation of the sigma points. The already computationally expensive resolution is consequently exarcerbated. Hence, in this project, for all the following computations of the nonlinear pipe conveying fluid model, we consider $N = 8$ beam mode shapes as a trade-off between fidelity and computational time.

In **b)**, we see that for both cases, the CN scheme converged to the desired second order. Based on the complexity of the pipe conveying fluid model, without gravity and Kelvin-voigt damping, the convergence rate is reached for $\Delta t_{CN} \leq 1/320$. Moreover, for a problem with the added contribution of γ and α, $O \approx 2$ if $\Delta t_{CN} \leq 1/640$. However, the desired time step depends on spatial discretisation Δh such that the CN results have to be stable. Hence, the convergence of the nonlinear pipe model is guaranteed if the number of beam mode shapes, the time rate, and the spatial step ensure the stability of the system. In this work, the

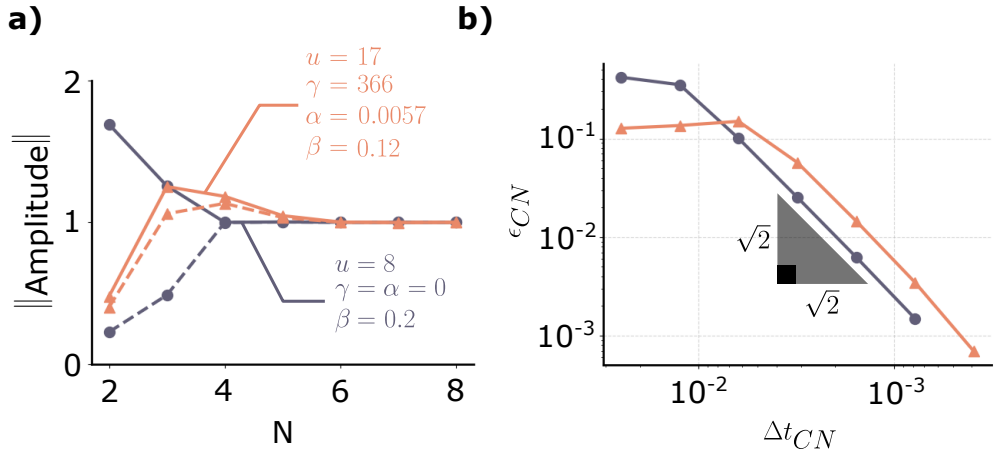

Figure 2.1 **Convergence analysis of the nonlinear pipe conveying fluid model for**: **a)** the number of beam mode shapes N according to the resulted normalised amplitude in both x (——) and y (- - - -) directions; **b)** for the Crank-Nicolson scheme given the time step Δt_{CN} employed.

appropriate Δt_{CN} and Δh according to the studied case are found by trial and error by comparing the results with the literature. Thus, we compare the 3D reconstruction of the pipe oscillation with the results in Modarres-Sadeghi et al. (2008). Figure 2.2 presents the nonlinear results for a pipe model with $\gamma = \alpha = 0$ and $\beta = 0.2$.

Figure 2.2**a)** and **b)** illustrate the phase plane trajectory of the pipe under a dimensionless flow rate of 8 and 12 respectively. We can observe the limit cycles of the nonlinear formulation of the pipe from the closed-loop pattern displayed. As the flow rate increases, the observed limit cycle goes from a circular to a rectangular pattern.

In **c)** and **d)**, we present the displacement of the pipe tip for the same two flow rates. At $u = 8$, a 3D motion is observed, whereas at $u = 12$, a 2D planar oscillation is reached. These two plots show the ability of the Python model to display the 3D behaviour of the pipe, which is also one of the main advantages of the Wadham-Gagnon et al. (2007) derivation over Semler (1996). In the planar case, the initial values in $\boldsymbol{\mathcal{R}}$ impact the orientation of the 2D planar motion.

Lastly, **e)** and **f)**, display the natural frequency of oscillation in steady state. As the flow rate increases, the dominant frequency also increases. For both cases, as expected, the model results match with Modarres-Sadeghi et al. (2008). To further push the comparison, we add the contribution of gravity in the model as illustrated in Figure 2.3.

In Figure 2.3, we compare the 3rd-order model adapted in this project with the one of Païdoussis and Semler (1998) and Farokhi et al. (2021). In addition to some small variations

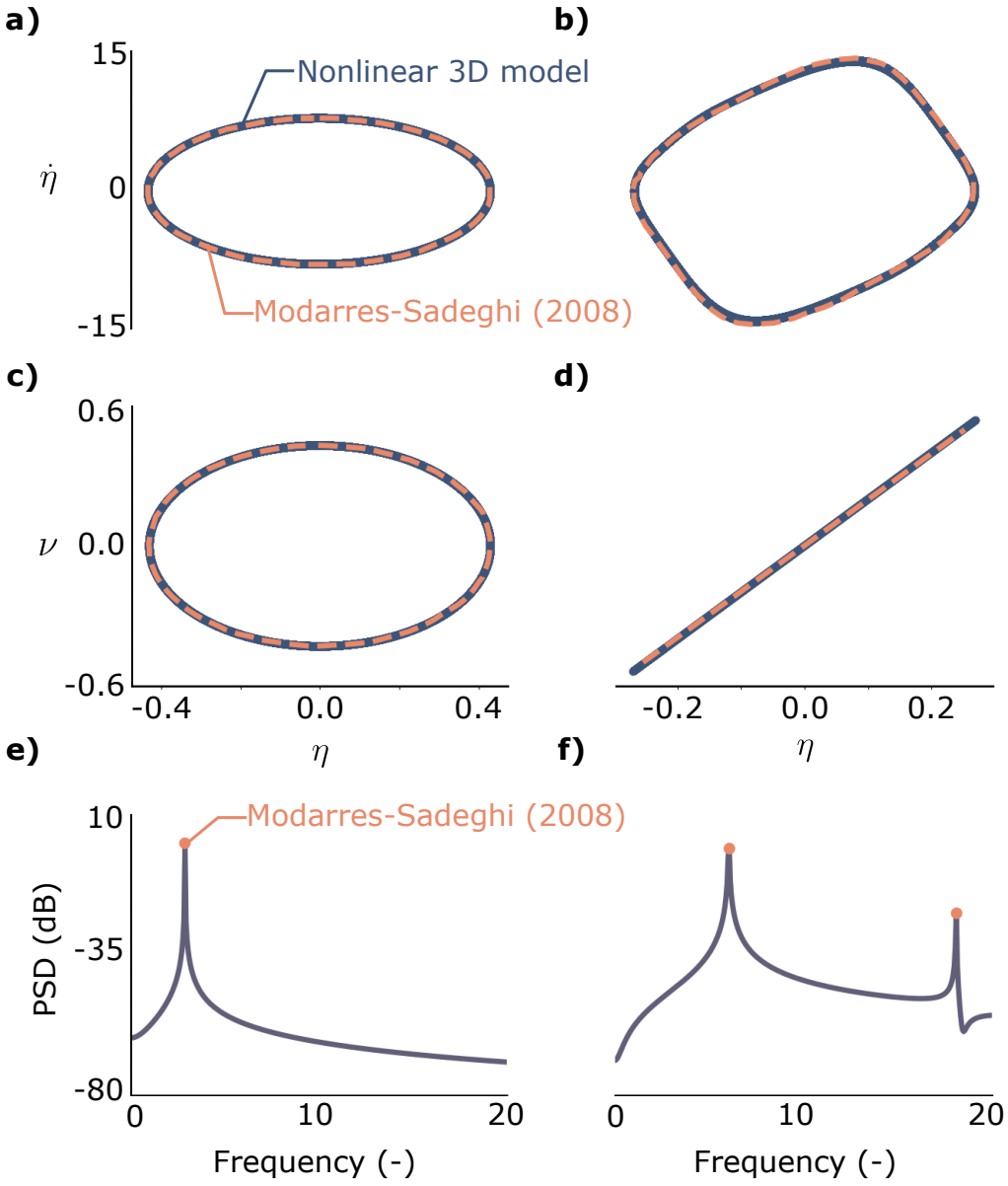

Figure 2.2 **Comparison of the nonlinear results of our Python implementation with Modarres-Sadeghi et al. (2008) for two dimensionless flow cases of $u = 8$ (a-c-e) and $u = 12$ (b-d-f) where**: **a-b)** are the phase plane; **c-d)** are pipe's tip deflection; **e-f)** are the power spectral density (PSD) given the pipe oscillation frequencies.

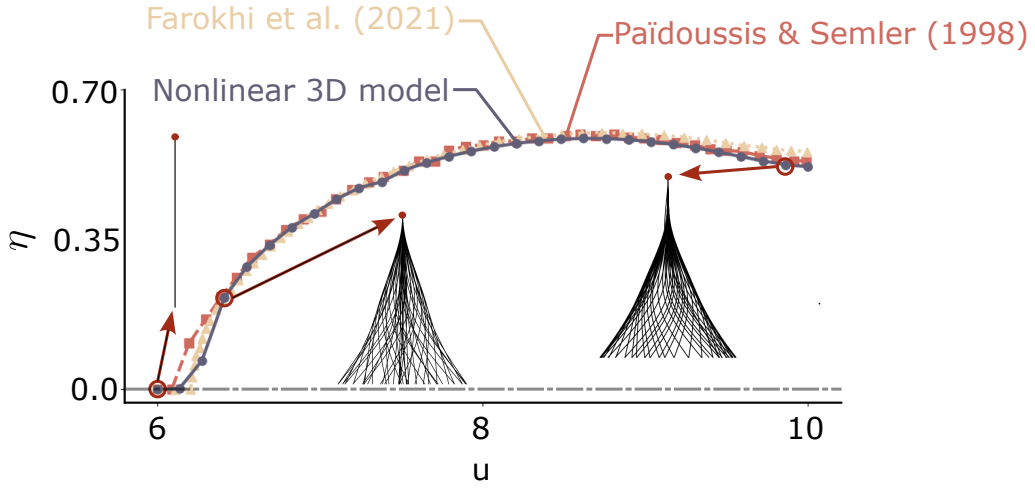

Figure 2.3 **Bifurcation diagram for the nonlinear pipe model with** $\beta = 0.142, \gamma = 18.9$ **and** $\alpha = 0$. Transverse tip amplitude $\boldsymbol{\eta}$ given the dimensionless flow rate u of the present thesis (——◯) compare with the results from Païdoussis and Semler (1998) (- - - -□) and Farokhi et al. (2021) (·····△).

with respect to the critical flow rate u_{cr}, the overall bifurcation diagram matches both models. u_{cr} of Païdoussis and Semler (1998) is 6.09, for Farokhi et al. (2021) is 6.20 and in our case 6.15. The difference in results arises from the number of beam mode shapes used in the discretisation and the numerical scheme. For a pipe oscillating in 2D planar motion from u_{cr} to chaos, increasing u to $u = 10$ generally increases the oscillation amplitude. However, the amplitude decreases slightly when $u \geq 8.63$.

2.3 Event-based Cameras: a New Sensor Paradigm

Data acquisition of high-dynamically excited systems and flexible structures cannot be done with contact sensors such as accelerometers. Contact measurements act locally on the structure, making it a sparse measurement method. Thus, the location and number of sensors on the system is essential for monitoring and extracting sufficient information about the dynamical system (Chen and Ni, 2018; Lai et al., 2020). Furthermore, for soft pipes, the addition of such sensors could significantly change the pipe behaviour by adding a local mass (Païdoussis et al., 2007). Hence, non-contact measurements such as laser vibrometers, proximity sensors, eddy current sensors, ultrasonic measurements, and digital cameras (Li et al., 2024b; Zhao et al., 2023) are necessary. In Hydro-Québec's hydro turbines, contactless measurement such as proximity metre sensors is widely used for measuring displacement of the main rotor shaft in its bearing for anomaly detection. Moreover, from a predictive maintenance perspective, contactless measurements are crucial to assess vibration-based monitoring by

tracking amplitude oscillations. With such data, engineers can detect and localise damage in mechanical systems simply by analysing the resulting modes, damping parameters, and fundamental frequency evolutions over time (Chen and Ni, 2018; Saidin et al., 2023). Among contactless measurement methods, digital cameras with computer vision-aided algorithms can capture the entire system in one frame (Lai et al., 2020). Then, it is possible to overcome the challenge of local sensing and reduce the cost of positioning multiple sensors. However, frame-based cameras involve large data acquisition and transfer, redundant information, and intensive computational cost (Litzenberger et al., 2006; Hsu et al., 2024; Raynaud and Mulleners, 2024; Gallego et al., 2022). EVBs offer a paradigm shift in continuous and real-time monitoring of systems (Hsu et al., 2024; Gallego et al., 2022; Rebecq et al., 2018; Lyu et al., 2024). This bioinspired sensor records changes in the pixel light intensity of the dynamical object. The sensor perceives the logarithmic intensity of the light at a pixel coordinate $(x_{e,i}, y_{e,i})$ and monitors the change in luminosity between the timestamp $t_{e,i}$ and $t_{e,i+1}$. This intensity shift is then denoted as an events e characterised by its pixel coordinates x_e and y_e, its timestamps t_e where the shift occurs and a boolean event polarity value p. The polarity value is either 0 or 1, if the light intensity is decreasing or increasing, respectively. This change in light intensity is due to movement in the scene of the object or the camera itself (Rebecq et al., 2018). Hence, the camera detects strictly moving edges and discards static information (Rebecq et al., 2018; Litzenberger et al., 2006).

Thus, focussing on the light intensity decreases the redundancy of information and leads to: (i) an increase in temporality resolution and dynamical range; (ii) a reduction in data transmission bandwidth; (iii) a reduced motion blur effect; (iv) low power consumption (Lyu et al., 2024; Raynaud and Mulleners, 2024; Gallego et al., 2022; Rebecq et al., 2018). The recent commercialisation of this technology (Gallego et al., 2022) has led to numerous innovations in the domain of computer vision. EVB have been used for the monitoring and estimation of the speed of moving vehicles (Litzenberger et al., 2006), the reconstruction of the motion of phenomena of fast FSI (Raynaud and Mulleners, 2024; Lyu et al., 2024; Hsu et al., 2024), the monitoring of structural health (Lai et al., 2020; Li et al., 2024b; Zhao et al., 2023), the recognition of objects (Lee et al., 2012; Lagorce et al., 2017; Iddrisu et al., 2024), and the 3D reconstruction of objects (Rebecq et al., 2018; Kogler et al., 2011), to name a few. In this project, EVB cameras are used to reconstruct the pipe displacement in 3D. Previous work in (Demenois et al., 2023) used high-speed cameras and computer vision algorithm for 3D reconstruction of the pipe. However, the whole method lacks the previously mentioned challenges of frame-based cameras while being time consuming. For a digital twin framework, such improvements offer a promising tool for a real-time monitoring of the physical asset and a real-time update of the virtual entity. The challenges of EVB

come with facing (i) uncorrelated space-time output, (ii) polarity information, and (iii) ghost noise (Gallego et al., 2022; Raynaud and Mulleners, 2024). In fact, for a given time recording window, the EVB output is an uncorrelated cloud of events. Figure 2.4 illustrates the EVB outputs for an oscillation of a pipe.

In Figure 2.4, the cloud of the event is observable over time. The data flux is continuous once the recording has started. As the pipe oscillates from right to left, the polarity of the data is triggered as $p = 0$ as the light intensity decreases and $p = 1$ as the intensity increases. The ghost noise present in the event's cloud comes from the photon flux variations and from transistor circuit noise (Gallego et al., 2022; Raynaud and Mulleners, 2024). Structuring these events cloud in usable data is a challenge. In (Raynaud and Mulleners, 2024), a coarse chain-like structure of N joints connected with piecewise polynomials is used to reconstruct a slender flag. In Lyu et al. (2024), a clustering technique of K-means is used to regroup each event into time windows. Then, a Kalman filter is employed to correlate the events and estimate the trajectory of flapping membranes. Neural networks have also been used to extract patterns in event cloud windows (Li et al., 2024b; Gallego et al., 2022). In Lai et al. (2020) beam's mode shape was used for correlating events in a physic-informed manner. In all cases, authors take advantage of the already-segmented data to extract information by clustering the events in either time- or space-windows.

Finally, 3D reconstruction from events has been largely studied (Rebecq et al., 2018; Kogler et al., 2011; Lagorce et al., 2017; Lee et al., 2012). In the initial work on the matter, multiple cameras were combined via stereo vision (Kogler et al., 2011; Lagorce et al., 2017; Lee et al., 2012). Cameras share a synchronised internal clock and a triangulation is done to reconstruct the 3D space. This method is often known as stereophotogrammetry (Hsu et al., 2024; Kogler et al., 2011). In (Rebecq et al., 2018), the reconstruction of an object is performed with one moving camera with a depth estimation algorithm. Both methods exploit the position of the cameras with the object to vary the points of view and realise a 3D reconstruction.

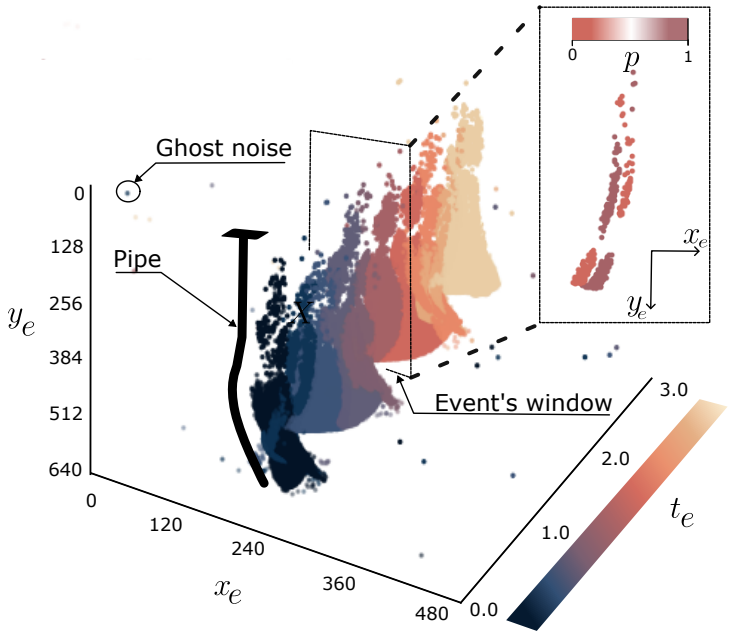

Figure 2.4 **Event-based recording of an oscillating pipe-conveying fluid in 2D for a time-interval** $t = 3$ **s.** Here coordinates (x, y) are the pixels triggered by light change and p is their polarities.

CHAPTER 3 SYNTHESIS OF THE LITERATURE REVIEW AND OBJECTIVES

3.1 Synthesis of the Literature Review

The integration of new intermittent energy sources into the power grid due to sustainability issues increases the stress on hydro turbines. A digital twin of hydro turbines could optimise their performance under stringent conditions and predict future failures in the system. This therefore motivates the global philosophy of this project: *to develop numerical tools and knowledge to build a Digital Twin framework for hydro turbines*. With the turbine sensor data, the physical model describing their dynamics, and the engineer's expertise, it is possible to build a multi-model virtual asset. Interoperable algorithms such as the UKF have the capacity to estimate hidden parameters relying on physics-based models and sensor data. By augmenting Kalman filters with a data-driven model, such as dynamic mode decomposition, the possibility of real-time monitoring and predictive maintenance of complex mechanical systems becomes more tangible.

To develop a DT framework that would scale to hydro turbines, we are working on a smaller, simpler fluid-structure interaction device: the pipe-conveying fluid. This experimental model has the advantage of (i) being easy to build while displaying a wide range of dynamics. Furthermore, (ii) it offers a simple alternative for testing and validating our algorithms in school laboratories. Finally, (iii) to identify hidden parameters and observe similar FSI phenomena encountered in both pipes and hydro turbines systems.

3.2 Research Objectives

From the summarised review of the literature, the main objective of this project is **to build and develop a DT framework of a pipe conveying fluid**. In order to realise the main objective, we propose the following sub-objectives (SO).

- **SO1 : Reconstruct in real-time the movement of a cantilevered pipe in 3D using EVB cameras.** To link both the physical system and the virtual asset, we are developing a 3D reconstruction data acquisition model with event-based cameras. This noncontact measurement method enables us to exploit the space-time event-output data, dynamic edge detection, and high spatio-temporal capability for real-time monitoring. Linear and nonlinear dynamics of the pipe is reconstructed in 3D. The event

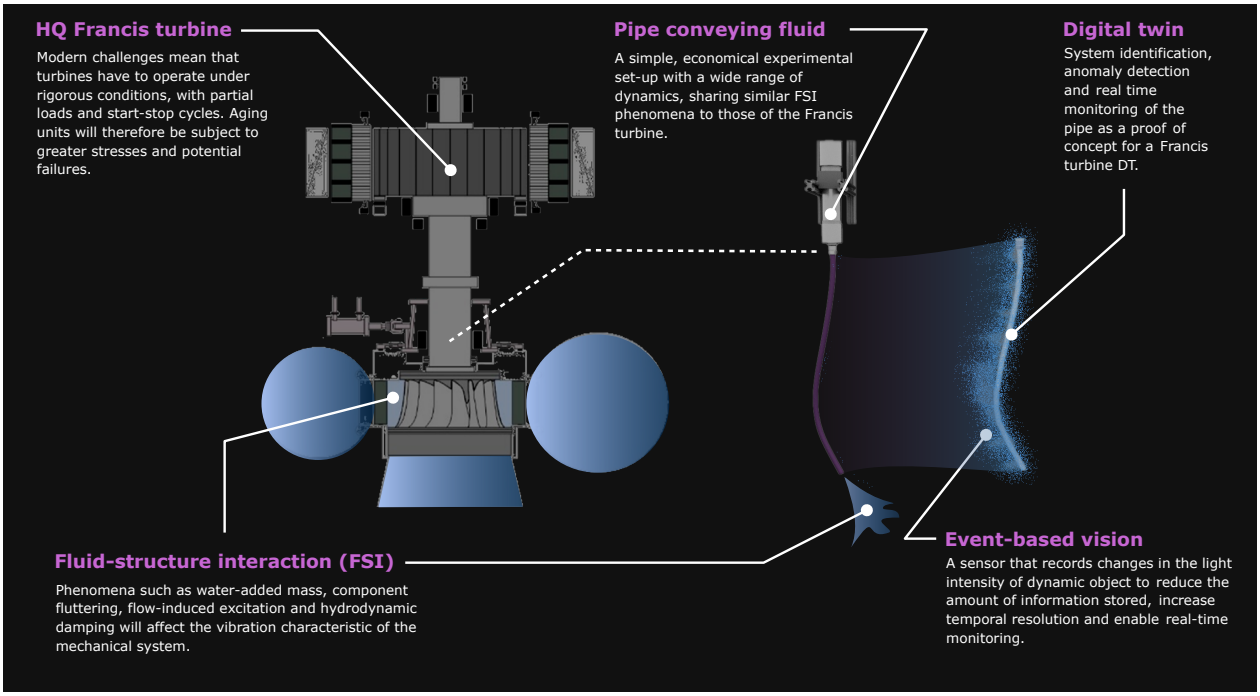

Figure 3.1 **The pipe conveying fluid as an experiment for a Francis turbine DT proof of concept.** Francis turbine figure from Hydro-Québec (2024b) and digital twin figure of the pipe conveying fluid by T. Laperle (2025)

processing method must be computationally inexpensive to enable real-time monitoring. The output data must accurately reproduce the movement of the pipe to be tested within the UKF and LDMDKF.

- **SO2 : Integrate the nonlinear pipe conveying fluid model with the UKF to solve the parameter identification problem**. The UKF estimates the hidden parameters of the pipe conveying fluid by comparing and updating its prediction from the nonlinear model with the EVB sensor data from SO1. The latter algorithm is verified and validated with synthetic and experimental data.

- **SO3 : Scale the nonlinear model with LKF-DMD for real-time monitoring and anomaly detection**. The ability of the low-order model to detect perturbations in the dynamics of the pipe conveying fluid is demonstrated with synthetic and experimental data.

- **SO4 : Combine SO1 to SO3 within a general framework that works in real-time as a DT proof of concept**. The framework must demonstrate the interoperability and modularity of the DT framework. The integrated model must be flexible enough to be enhanced or modified by subsequent algorithms.

CHAPTER 4 REAL-TIME DATA ACQUISITION WITH EVENT-BASED CAMERAS

4.1 Methodology

4.1.1 Experimental Set-up and Pipe Manufacturing

The pipe manufacturing methodology and the pipe conveying fluid setup developed and built by Demenois et al. (2023) are used. We use a room-temperature vulcanised silicone rubber (OOMOO30 Smooth On) to mould our pipe. As data acquisition is performed with EVB cameras, no image processing is required. Consequently, the step of blackening the silicone rubber to improve contrast is not necessary. Appendix B shows the slight modifications made to the procedures of Demenois et al. (2023). In this project, we cast three different pipes with the same inner diameter ($D = 6.35$ mm) and outer diameter ($D = 15.875$ mm). These pipe parameters are listed in the following Table 4.1.

The water mass per unit length parameter M, listed in Table 4.1, is obtained considering a water density of $\rho = 996 \quad \text{kg/m}^3$. For the mass of the pipe per unit length m, we weigh all three pipes and divide their mass by their length L.

Figure 4.1 shows an overview of the experimental setup used in this project. The pump ① conveys water through a reservoir ② to mitigate its impact on flow. The water then passed through a flow metre ③ and flows out through the pipe ④. The flow metre is used as a reference to validate the flow rate estimated by the UKF. Two EVB cameras (Century Arks SilkyEvCam EvC3A) ⑤ -⑤' are positioned 90 degrees from each other to capture the xz and yz planes of the translucent tank of the pipe ⑥. The cameras are mounted on PLA fixations with T-slots ⑦. This allows the operator to move the cameras vertically. A USB 3.0 cable ⑧ connects and powers both cameras from a desktop PC (Processor: 13th Gen Intel Core (R) (Tm) i9-13900H with 2.6 GHz and 32.0 GB RAM) ⑨. The cameras are physically synchronised using a hand-built 1.80 m long synchronisation cable with IX connectors

Table 4.1 Cast pipes used in this project considering an inner diameter of 6.35 mm and an outer diameter of 15.875 mm.

No.	L(m)	M(kg/m)	m(kg/m)	$I(10^{-9}\text{m}^4)$
1	0.36	0.0315	0.2427	3.038
2	0.46	0.0315	0.2302	3.038
3	0.46	0.0315	0.2321	3.038

(IX80G-B-10P: HiROSE) (10) and built according to Prophesee requirements (PROPHESEE, 2024). Their internal clock is triggered by the Sync In and Sync Out interfaces of Prophesee's Python API. These produce a 1 MHz pulsed signal and have been validated for a cable length of 1 m (PROPHESEE, 2024). We find that for a cable length of 1.80 m, signal reception is still adequate, but synchronisation takes about a second to be correctly set. Consequently, the recording start time t_0 should be set to 1 to guarantee a correct connection between the two cameras. To minimise splashing by the water that exits the pipe, a fine link net is placed at the bottom of the tank (11), which is framed by plexiglass panels. The exiting water of the pipe goes to the white plastic reservoir (12) and is pumped back through the system.

Motivation of EVB Compare to High-Speed Camera

In addition to the pipe manufacturing process and the mechanical setup used in this project, the overall methodology of the 3D reconstruction is based on the work of Demenois (2022b). In this work, the idea of polynomial fitting and parallax treatment with two sensors is directly adapted from his methodology. The only difference is the use of EVB sensors instead of high-speed cameras as done in his work. Hence, in Demenois (2022b), the recording of the pipe movement is performed once and saved in its local files. Then, the whole procedure implies to post-process the videos by (i) segmenting the pipe from the background and (ii) by converting the image to a binary format. This binary format is 1 if the pixel corresponds to the pipe or 0 if the pixel is the background.

This segmentation process combines with the image conversion to binary is time consuming and imply a stop in the overall DT procedure. Thus, before performing the system identification process with PINN, the videos have to be treated in advance. These limits motivate the use of EVB since these sensors already segment the moving object and convert the data into binary format through the polarity values p. Moreover, the EVB have the capacity to work with a smaller frame rate than the 150 fps of the high-speed cameras of Demenois (2022b) and to eliminate the motion blur effect from rapid movements. Hence, without this post-process procedure, we aim to reconstruct in real-time the pipe movement and perform anomaly detection and system identification with the incoming data flux.

4.1.2 Experimental Calibration and Data Acquisition

To correctly calibrate the cameras, an initial test is required. This involves defining a region of interest (ROI) and the system's anchoring coordinates. The main aim of this ROI is to eliminate (i) most of the ghost noise in the event window and (ii) the water ejection effect. As water ejection from the pipe is a dynamic phenomenon, the EVB captures it. However,

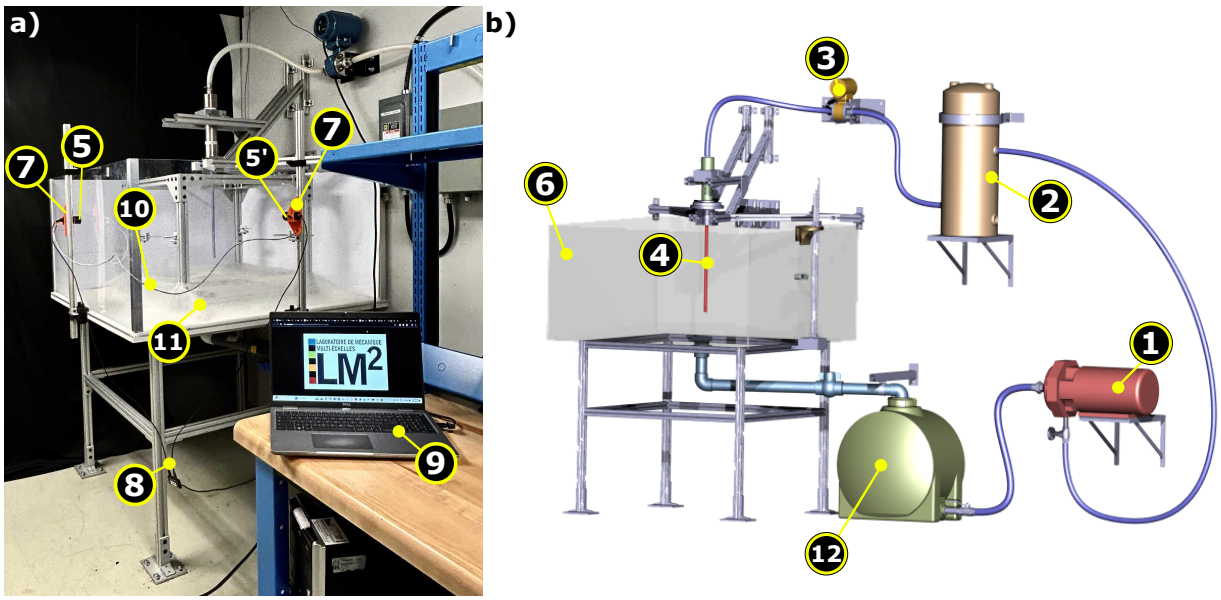

Figure 4.1 **Experimental setup of the pipe conveying fluid**. **a)** EVB recording setup and **b)** 3D model of the experimental setup by Demenois (2022b).

this event corrupts our reconstruction, giving the impression that the pipe is longer than it actually is. For the anchor coordinates, it is possible to define a scaling coefficient linking the pixel coordinates to a millimetric base (mm). We use these anchor points to (i) impose a zero displacement at the clamp and to (ii) start the fitting at a given position.

We therefore calibrate the ROI and anchor coordinate using an initial test on fluttering-pipe motion. First, we increase the flow rate until the pipe destabilises and reaches a stable periodic oscillation. Next, we visually define the most suitable rectangular mask around the pipe and further increase the flow rate. If the pipe's deflection exceeds the rectangular limits, the latter are slightly increased to ensure that the entire movement is still captured. This iterative procedure is applied until water splashing corrupts the whole recording procedure and a maximum rectangular mask is defined.

For the anchoring coordinates, we slightly perturb the pipe manually so that the EVB captures a movement. We then fit a straight line through the events. As the actual length of the pipe is known, we find the values in millimetres with a simple ratio:

$$\text{ratio}\left(\frac{\text{mm}}{\text{pixels}}\right) = \frac{\text{Pipe length} \quad (\text{mm})}{\text{Pipe length} \quad (\text{pixels})} \tag{4.1}$$

4.1.3 Proposed Workflow

2D Stereophotogrammetry

In this project, all algorithms are built and tested using `Python 3.9.19`. Figure 4.2 shows the proposed workflow for 2D/3D stereophotogrammetry of the pipe conveying fluid. For 2D reconstruction, raw event data are recorded in real-time using Metavision's `EventsIterator` function. The function is used to extract the coordinates of all events $(x_{e,i}, y_{e,i}, t_{e,i}, p_i)$ that occurred during the interval from $t_{e,0}$ to $t_{e,k}$. We define the sampling rate as $\Delta t = \frac{t_{e,k} - t_{e,0}}{k}$, where k is the maximum time length. The interval must be defined in such a way as to capture enough events to reconstruct the entire length of the pipe. Here, $\Delta t = 0.0025$ s is used.

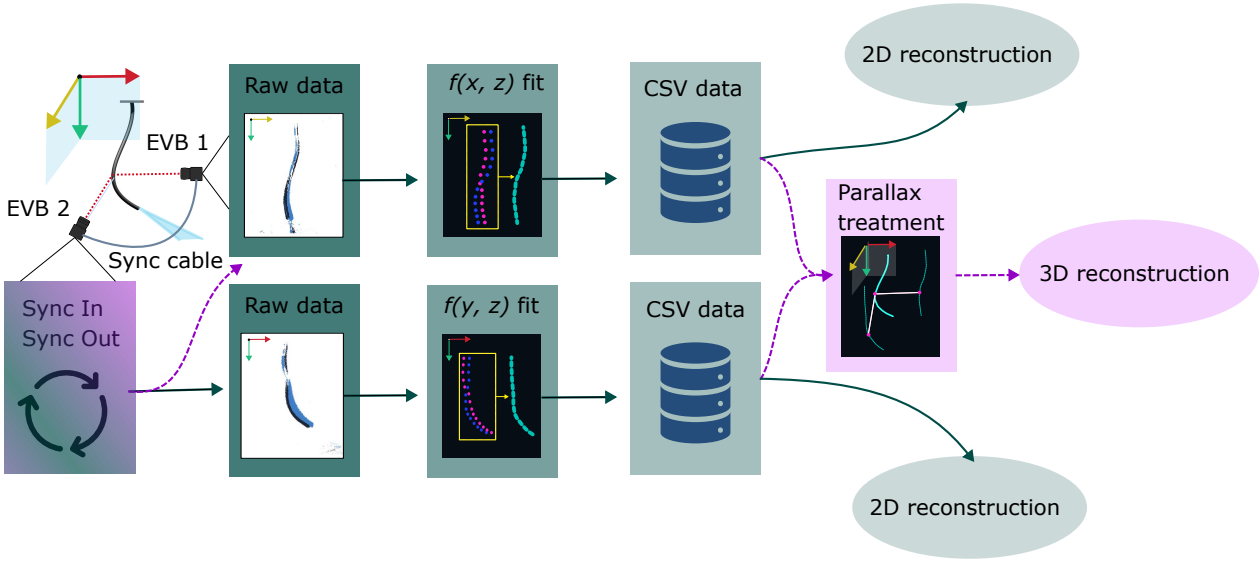

Figure 4.2 **2D/3D reconstruction algorithm of the pipe-conveying fluid recorded with EVB cameras.** The 2D reconstruction in green (——) is the direct path. The proposed 3D reconstruction in purple (- - - -) requires the synchronisation of two cameras and the merging of the two 2D reconstructions within a parallax processing function.

We reconstruct the motion of the pipe using a custom Python function. The latter code fits a polynomial function to the event cloud. It takes as input the coordinates and polarity of the event, the calibration values defined previously, and the number of joints to be fitted with the polynomial.

Using the anchor coordinates, we define a virtual set of randomly distributed points to constrain the polynomial fit so that it starts at this location. This limits the negative effects of the Runge phenomenon. This phenomenon is characterised by unwanted oscillations in the

interpolation boundary conditions that reduce the quality of the fit (Jung and Stefan, 2011). We also find that the first stage of the pipe has oscillations that are difficult to observe with the EVB. Consequently, the number of events between the anchor point and the observable dynamics of the pipe is either zero or very sparse.

Since this missing or sparse information can be poorly rebuilt, we add randomly distributed points in this area. These added points follow a straight line from the anchor region to the observed events. In doing so, we obtain a continuous set of events from the anchor to the tip. The added points and the events are then sorted according to their vertical position z. Depending on the polarity of the sorted events, we extract the polynomial coefficients using the `PolynomialFeatures` function from `Scikit-learn` for both sets of points. In this work, we assume a tenth-order polynomial and find the corresponding coefficients. For the latter, we fit the polynomial with the `LinearRegression` function from `Scikit-learn`, as shown in Figure 4.3**a)**. This gives us two curves representing the two edges of the pipe. We then average the two to extract a midline that represents the pipe deflection at a given time interval.

3D Stereophotogrammetry

For the 3D reconstruction of the pipe, we perform the 2D steps described in parallel via the synchronisation procedure. Then, a polynomial fitting is performed simultaneously, and the data is saved in a *csv* files. The output *csv* files are then merged within our `Parallax_adaptation` function to correct for parallax. In fact, as the pipe oscillates, it moves closer or farther away from the cameras, creating the illusion of a decrease in length. The error in the estimated deflection is corrected by using trigonometry as in Demenois (2022b) and the ratio (mm/pixel) obtained previously.

In addition to the two *csv* files, the parallax code also takes as input the number of points to be interpolated from the polynomial curves. Since the two curves in their respective planes have an equivalent number of points, we assume that the points with the same index in their list correspond to the same physical point. Figure 4.3**b)** illustrates the phenomenon of parallax.

In Figure 4.3**b)**, Thales' theorem is employed with the known coordinates to obtain the corrected coordinates (x_c, y_c) with:

$$x_c = \frac{x_{rc} D_x (D_y + y_{rc})}{D_y D_x + y_{rc} x_{rc}}, \tag{4.2}$$

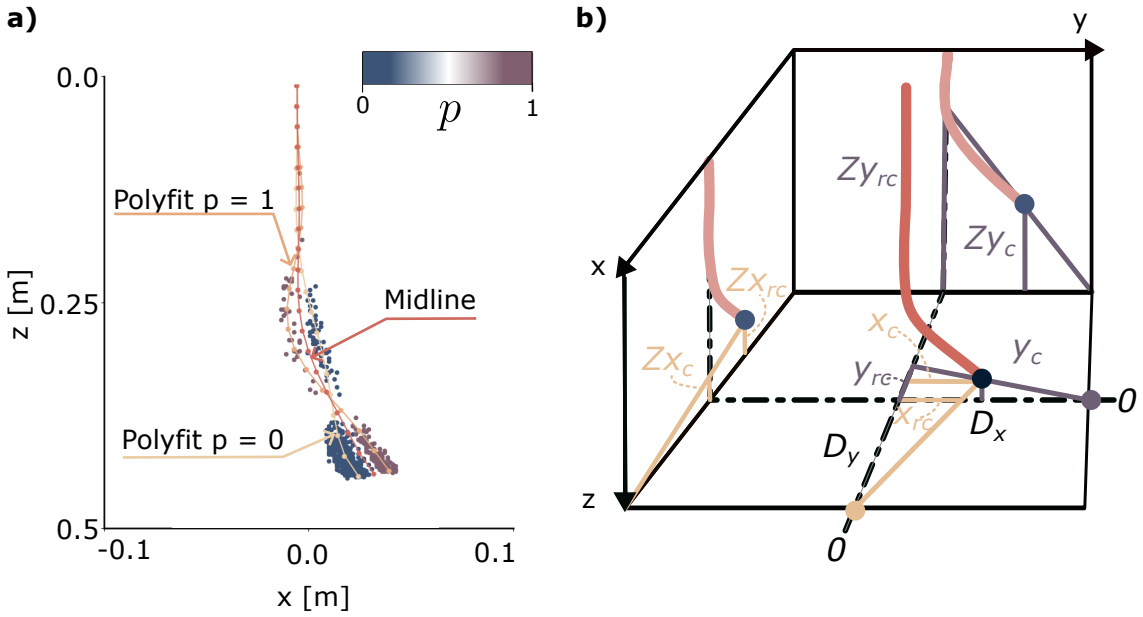

Figure 4.3 **Stereophotogrammetry of the EVB data**. **a)** is the polynomial fitting through the two polar event cloud and the average midline. **b)** is a schematic of the parallax treatment of the pipe in the center. The data coordinates are corrected by being projected on the three planes xz, yz and xy. The Thales' theorem is employed to correlated the previous coordinates with the corrected ones.

$$y_c = \frac{y_{rc} D_y (D_x - x_{rc})}{D_y D_x + y_{rc} x_{rc}}, \tag{4.3}$$

$$z_c = \frac{z_{x_{rc}} \frac{D_y + y_c}{D_y} + z_{y_{rc}} \frac{D_x - x_c}{D_x}}{2}, \tag{4.4}$$

where D_x and D_y are the known distances between the cameras and the pipe at rest. The coordinates with the subscript c are corrected, while the coordinates with the subscript rc are the data reconstructed.

4.1.4 Data Acquisition Method

The pump drive frequency is first set to a desired value to initiate the flow in the system. Once the flow velocity stabilises after 1 minute, the frequency is increased by incremental steps of 0.5 Hz. Consequently, the flow through the system increases until the pipe starts to flutter. At that point, the critical flow rate value is observed.

Above the critical flow rate value, the flutter behaviour of the pipe is observed and the cameras are synchronously started. The raw data flow is sent to the computer for 2D/3D reconstruction. The latter can be saved in a *csv* format or sent directly to the DT framework.

We carried out two experimental 3D reconstruction data campaigns with pipes No. 2 and 3. We started the data acquisition procedure by setting the pump frequency at 27 Hz, putting the pipe on the verge of destabilisation. Then we increased the frequency by 1 Hz and recorded the associated fluttering for 10 seconds considering a sampling rate of 0.025 s. We follow the same procedure until the water splashes on the transparent tank walls, corrupting the event acquisition method. For both pipes, we conducted 13 recordings from the view of both planes (xz and yz).

4.2 2D Benchmark of the Proposed Polynomial Fitting

Here, we verify the accuracy of the proposed method with the synthetic event data. The verification procedure is inspired by Raynaud and Mulleners (2024). It consists of converting mp4 videos of simulated oscillations into streams of events with the Metavision function `SimulatedEventsIterator`. These mp4 videos are generated with the nonlinear pipe conveying fluid model considering the parameters of the pipe No.1 in Demenois (2022b). The latter pipe is characterised by $L = 0.46$m, $\gamma = 355$, $\beta = 0.124$, and $\alpha = 0.006$. For ease of analysis, we impose a 2D planar motion on the nonlinear model, considering the initial displacement only in the xz plane. We employ a dimensionless time step of $\Delta\tau = 0.003$ within the CN scheme. On the other hand, we reconstruct the pipe model with $N = 100$ points and consider a time window equal to the redimensioning of $\Delta\tau$. We compare the results of our algorithm on the frame-to-event outputs with the synthetic pipe model with a fixed duration of $\tau = 3$.

Taking into account the known deflection coordinates in 2D $x_s(t), z_s(t)$ and the fitted shape $x_{fit}(t), z_{fit}(t)$ we can define error quantification metrics. The pointwise error reconstruction ϵ_{RMS} is evaluated at each time step Δt as the root-mean-square (RMS) of the difference between the known coordinates and the reconstructed one:

$$\epsilon_{RMS}(x_n(t), z_n(t)) = \sqrt{\frac{1}{N_L}((x_{fit_n}(t) - x_{s_n}(t))^2 + (z_{fit_n}(t) - z_{s_n}(t))^2)}, \quad \forall \quad 0 \leq n \leq N_L \tag{4.5}$$

where N_L is the discretised length of the pipe and n the increment of the pipe at which the motion has been reconstructed at time t. Taking into account $n = N_L$, we can extract the same tip-based metric $\delta_{RMS}(t)$ as Raynaud and Mulleners (2024),

$$\delta_{RMS}(t) = \sqrt{\frac{1}{L}((x_{fit_N}(t) - x_{s_N}(t))^2 + (z_{fit_N}(t) - z_{s_N}(t))^2)} \tag{4.6}$$

For this benchmark, we consider 6 cases by varying the dimensionless flow rate from 15 to 20.

Hence, we can analyse the impact of amplitude variations on the proposed algorithm. All the cases studied are free of noise. Figure 4.4 presents the benchmark results of the comparison of the reconstruction with the synthetic data.

In Figure 4.4**a)**, we see an increase in the error according to the flow rate observed. The relative RMS average tip error δ_{RMS}/L for the six cases is 2.87%. Regardless of the increase in flow rate, the minimal relative error δ_{RMS}/L is always 1.67%. However, maximum values vary from 2.59% for a $u \leq 15$ to 6.06% for $u \geq 20$. Since an increase in flow implies an increase in the pipe amplitude oscillation, the algorithm tends to perform less well for large motion.

Figures 4.4 **b)**, **c)**, and **d)** and snapshots **e)**, **f)**, and **g)** allow us to observe the critical zones where the maximum error arises. The colormap model presents the pointwise RMS error ϵ_{RMS}/L along the normalised pipe length $\frac{z}{L}$ given the dimensionless time τ for cases $u =$ 15, 17, and 20. We note that most of the error is concentrated at the tip and follows a wavy pattern. For a small oscillation as is the case for $u = 15$, the error ranges from 0.01 to 2.59 % with an average value of 1.04 % throughout the pipe reconstruction. For $u = 17$, the error ranges from [0.006, 3.85]% with a total average of 1.26 %. Finally, for a larger range of amplitude at u=20, the error is included in [0.009, 6.08]% and corresponds to an average value of 1.39 %. The maximum error is encountered at the maximum amplitude. Poor reconstruction is found up to 20 % from the tip. However, from the average results of all three cases, we can observe that the overall reconstruction is well executed for the other 80 % of the pipe. Since the oscillation in this region is minimal, the algorithm fits its polynomial easily. These observations are supported by the snapshot figures in **e)**, **f)**, and **g)**. We can see that the error reaches a minimal value once the pipe reaches its lowest point. Also, we can see that the length of the pipe is slightly off compared to the true value and that the curves in the pipe are often exaggerated. This is explained by the sorting data step and the degree of polynomial used in the 2D reconstruction. Indeed, the non-correlatability in the events implies the need for sorting the data given the z-axis for fitting the polynomial. However, in most cases, events with the same z coordinates are presented. This biases the fitting by aggregating points in some critical regions, such as in the curve and the tip locations. We also see one of the drawbacks of polynomial fitting by looking at snapshots **e)**, **f)** and **g)**. One can see unwanted oscillations at the anchor point due to the Runge's phenomenon. By further increasing the degree of the polynomial, we can reduce the difference in the interpolation regions of the curve. However, the effects of Runge's phenomenon increase under both boundary conditions. Hence, with a lower degree in the polynomial fitting, we lose slightly in accuracy along the pipe, but we maintain better stability at the anchor and the tip points. In general, these results demonstrate that the proposed method can reconstruct

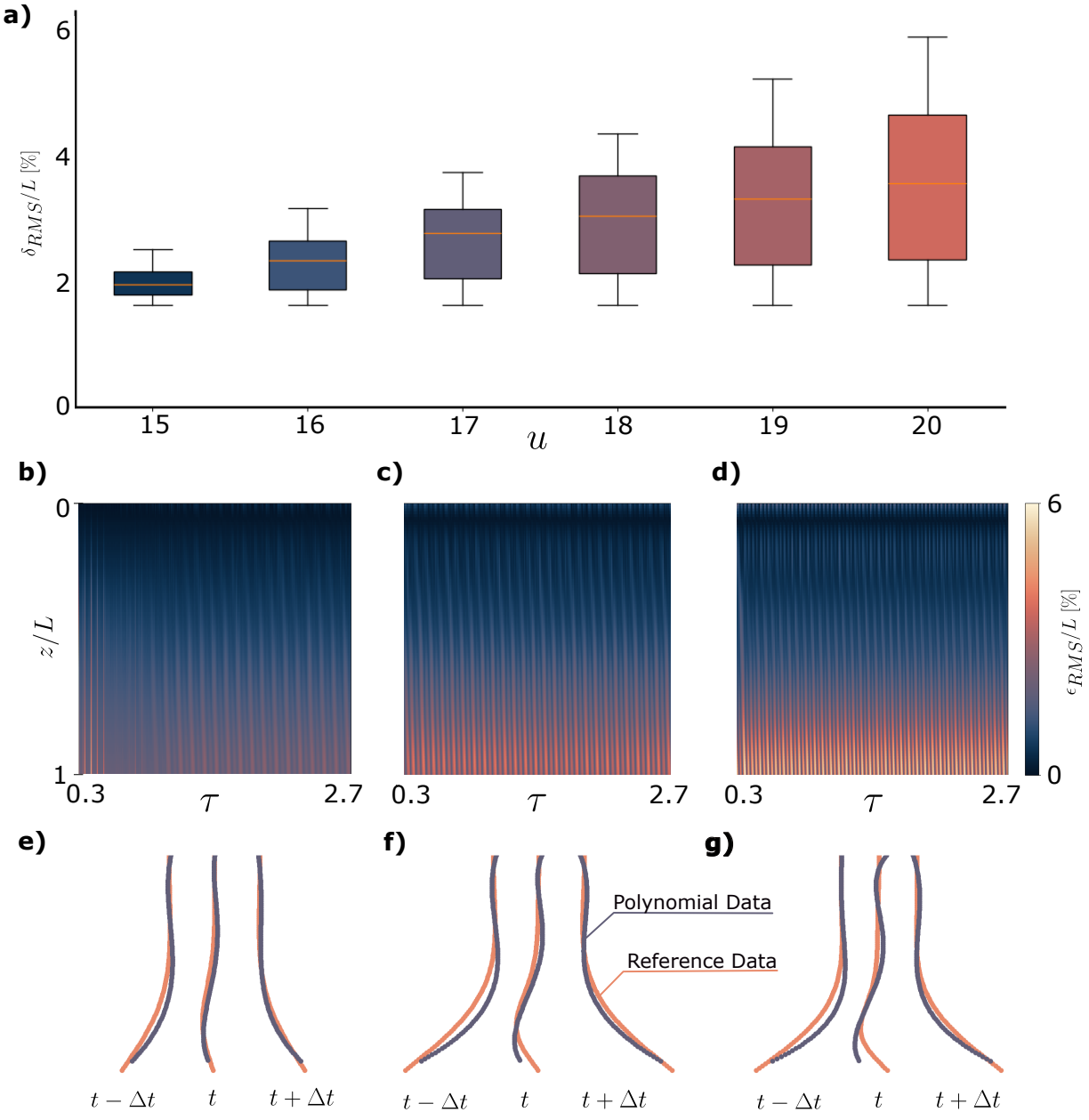

Figure 4.4 **Benchmark results from the comparison of our algorithm reconstruction with synthetic data obtain with the nonlinear pipe model**: **a)** presents the tip RMS error δ_{RMS} given the dimensionless flow rate u; **b - d)** illustrate the relative pointwise RMS error ϵ_{RMS}/L for the dimensionless pipe's length z/L given the dimensionless time τ. We showed the $u = 15$(**b**), 17(**c**) and 20(**d**) cases; **e - f)** displayed three snapshots for the same three cases at both maximum amplitude ($t \pm \Delta t$) and when the pipe tip is at its lowest point (t). In the snapshots a comparison is made with the synthetical data in pale orange and the reconstructed motion in grey.

the shape of the pipe with an average error of $1.23^{+4.85}_{-1.23}\%$. This associated model tolerance error can be seen as a reproducibility error. Next, we investigate the effect of the time window's length and of the noise on the algorithm's performance with experimental data.

4.3 Experimental Results

From the results of the experimental campaign, we tested our model on real and noisy data. Moreover, we tested the computational time of the algorithm for 3D reconstruction. The experimental results for pipe No.3 are presented in Figure 4.5.

Figures 4.5**a)** to **c)** present the experimental results for the flow range of [5.0, 7.0] m/s. In Figure 4.5**a)**, the normalised computational time to reconstructed the dynamic of 10 s is presented. We use `alive-progress 3.2.0` in Python to measure the computational time of the reconstruction. At 0 and less, the reconstruction is in real-time, while at 1, it takes twice the time. For small oscillations, the reconstruction is almost done in real-time. This is because the events present in the time window are less than in a case with large-amplitude oscillation, making it easier for the polynomial algorithm to find the best-fit function. On average, to reconstruct 10 seconds of oscillation, the proposed method takes 23 s to run with a minimal computational time of 10.1 s and a maximal of 39 s. Hence, to run 1 s, the algorithm accumulates a delay of approximately 1.3 s. This delay depends on the number of events; the higher the number in the time window, the slower the algorithm performs.

In Figures 4.5**b)** and **c)**, the evolution of the frequency of motion of the pipe and the amplitude of the tip is shown. For both planes' data sets, we measure the frequency by realising a Fast Fourier Transform (FFT) on the tip data results. For the amplitude, we extract the absolute values of the maximal tip amplitudes. As the flow rate increases, the frequency of the pipe and the amplitude increase consequently. Hence, the larger and faster the oscillations, the longer it takes to reconstruct the entire pipe. This is explained by the fact that the dynamics in the xz plane is greater than the dynamics in the second plane. Thus, the number of events to sort and the complexity of the curve to fit is greater. For a flow rate smaller than 5.5 m/s, the oscillation amplitudes and frequencies in both planes are smaller, and the computational time needed is almost in real-time. In Figure 4.5**b)**, the similarity in the frequency of both planes tells us that the cameras are well synchronised. However, the slight deviation at $u = 5.84$ m/s shows that the data from both cameras have a slight difference in their frequencies and could come from a manipulation error during the synchronisation procedure or a movement in the background that has increased the noise in the recording. This error shows that even a small change in the light intensity can affect the whole recording procedure. From Figure 4.5**d)**, we can explain the difference in amplitudes in Figure 4.5**c)**.

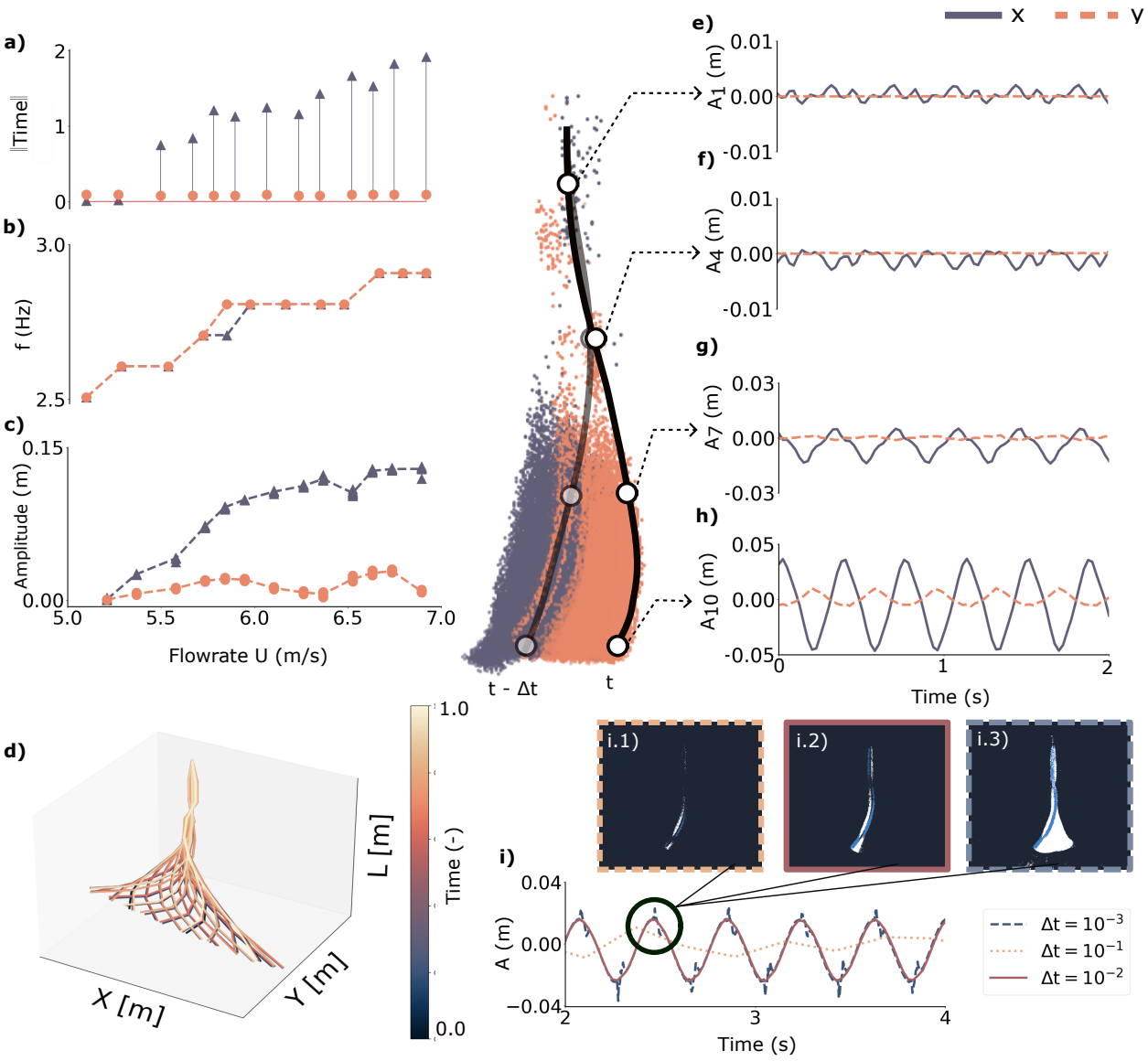

Figure 4.5 **Experimental results of 3D reconstruction for pipe campaign No. 3.** For the 13 cases, in the flow range $[5.0, 7.0]$ m/s, we extract: (**a**) The calculation time for the reconstruction; (**b**) The vibration frequency of the pipe; (**c**) the maximum amplitude. The symbols (△) and (◯) refer to data in the xz and yz planes respectively. For the case u = 5.73 m/s for 2 seconds, we show: (**d**) The global 3D reconstruction; Subfigures **e-h** The point results for four different points considering a 10-point discretisation; **i**, the impact of the choice of time window on data acquisition is demonstrated for $\Delta t = 10^{-3}$ (**i.1**), $\Delta t = 10^{-2}$ (**i.2**) and $\Delta t = 10^{-1}$ (**i.3**).

In fact, the nonlinear dynamic displayed is a 3D planar motion almost parallel to the xz plane. Hence, in the yz plane, the motion looks almost like a static pipe. Thus, fitting and sorting are computationally less expensive.

Figures 4.5**e)** to **h)** present the oscillating results at the discretised points of the pipe. Here, the results at $u = 5.73$ m/s are shown for 4 points over a distribution of 10 points. For the first two points, the oscillation is negligible. However, for data in the xz plane, we can observe a corrupted oscillation pattern. This comes from Runge's phenomenon explained in the previous section. At the anchor point, the polynomial fitting does not start at zero but displays an unwanted oscillation. At the last two points, we see an attenuation of Runge's phenomenon and cleaner oscillatory behaviour. However, we see a slight asymmetry between the maximal amplitudes. This comes from the fact that the planar motion is not exactly parallel to one or the other planes.

Based on the results obtained, we see that the ROI calibrated earlier is a simple method to discard ghost events. This is also true because of the time window. In Figure 4.5**i)**, we compare the acquisition method with a time window of $\Delta t = 10^{-3}$ s , $\Delta t = 10^{-2}$ s and $\Delta t = 10^{-1}$ s for a flow rate of 5.57 m/s. By reducing the time window's size, fewer events are captured (i.1). Hence, at the maximal amplitude, the event continuity is sparse and the fitting is negatively affected by the surrounded ghost events. For a bigger time window (i.3), too many events are captured. Then, the average of both edges polynomial fitting is biased. This results in a poor reconstruction. In our case, by trial and error, we found that the best time window should be around $\Delta t = 10^{-2}$ s.

4.3.1 Comparison of the Reconstruction for Different Pipes

Manufacturing new pipes implies the uncertainty of introducing defects during the process, such as air bubbles in the silicon melange. Thus, even though the pipes are the same length and are moulded following the same procedures, each of them is unique. On a larger scale, hydro turbines have the same limit. Unknown defects can be present and their use can vary depending on location, water flow, age, etc. To better understand the importance of a DT framework for predictive maintenance, we compared the EVB reconstructed data with the one obtained with pipe No.1 in Demenois (2022b). The pipe No.1 has a length of 0.46 m, similar to pipe No.2 and 3 of the present study. In Demenois (2022b), high-speed cameras with a frame rate of 150 fps are used. Figure 4.6 shows the results of the three pipes for their maximum tip amplitudes and their main frequencies for a flow range of [5.0 to 7.0] m/s.

Figure 4.6**a)** presents the frequency given the flow rate. An increase in flow is also observed at the frequency level. The results for pipe No.1 of Demenois (2022b) and our pipe No.3 are

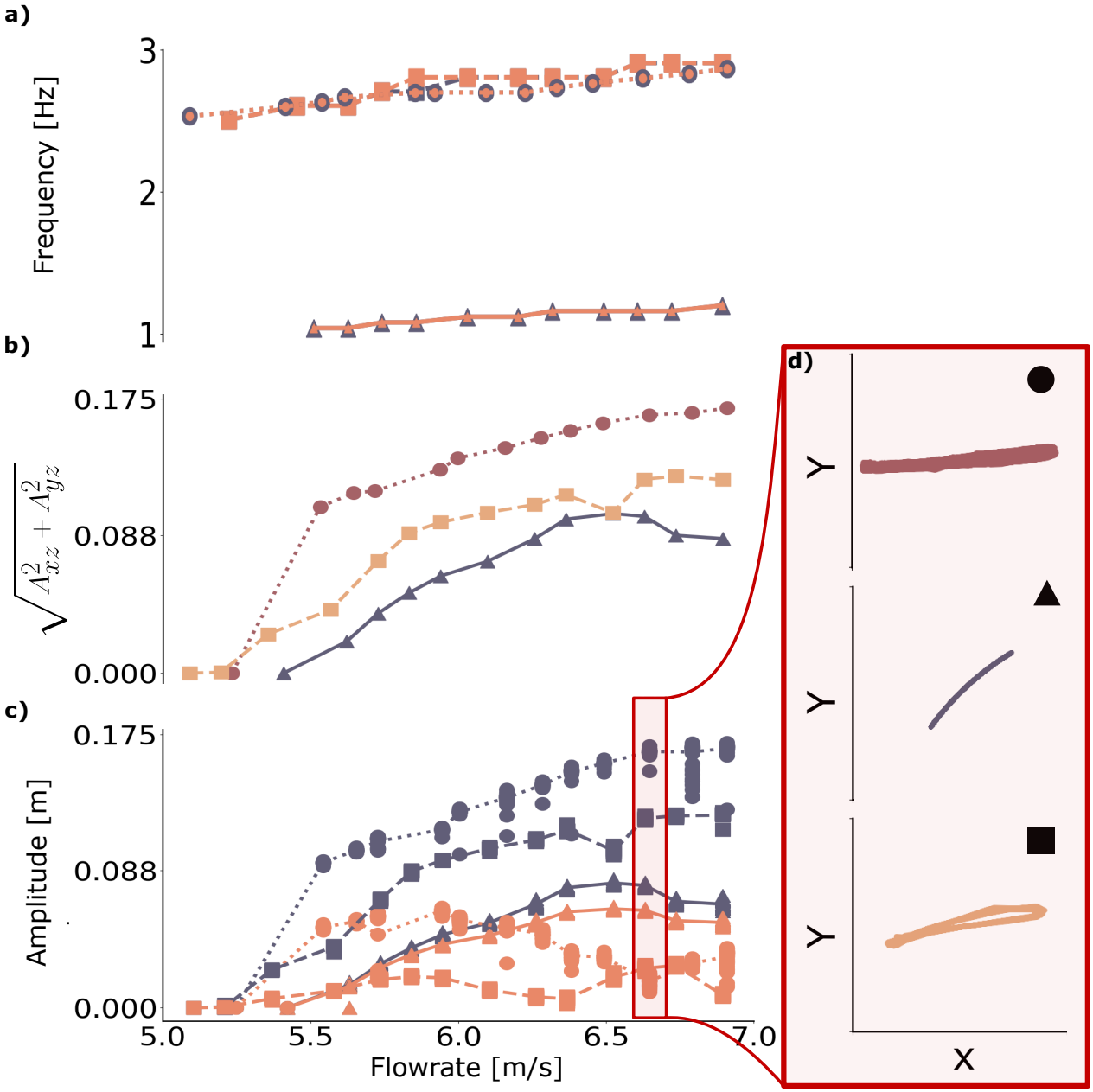

Figure 4.6 **Comparison of the EVB reconstructed data of pipe No. 2 and 3 of the present study with the pipe No. 1 of Demenois (2022b)**. Given the flow range of 5.0 m/s to 7.0 m/s we compare : **a)**, the frequencies in Hz for the pipe No. 2 (——△) and No. 3 (- - - -□) of the present study with the pipe No. 1 (·····◯) of Demenois (2022b); **b)** the RMS of the tip amplitude; **c)** the tip amplitude for all three pipes given the xz (orange) and yz (grey) planes. **d)** illustrates the trace of the tip for u = 6.63 m/s for the three pipes.

similar. All three pipes are moulded with the same procedure and the same material, yet we see that the frequencies of pipe No. 2 are half as high. Looking at Figure 4.6**b)** and **c)**, the amplitude tends to increase with the flow rate. However, we see that all three pipes have different amplitude ranges in both planes. Moreover, looking at Figure 4.6d), the 3D planar motion of the pipes is different, and they all have their own preferential plane of oscillation. All these differences come from imperfections in the pipe and in the clamping stiffness in the top boundary condition. It shows that even though the manufacturing process is the same for all three pipes, the material properties and the boundary conditions vary from one pipe to the other. The presence of air bubbles in the silicon melange can reduce the thickness of the pipe at some locations and affect the material properties. In addition, in Demenois (2022b), black pigments are used in the melange to increase the contrast between the pipe and the white background. Since the EVB recording does not need this improvement in contracts to catch the movement of the pipe, this part of the pipe manufacturing recipe was discarded. However, it could have slightly changed the general properties of the material in comparison to ours. Finally, a variation in the clamping stiffness and damping affects the general oscillation preferential plane (Paidoussis, 2014; Kheiri, 2020). Hence, it demonstrates the need of DT's and predictive maintenance in engineering system. All mechanical units are unique and behave consequently. With a well-defined DT framework one could predict failure or identify hidden parameters in its system without relying on general conservative frameworks.

CHAPTER 5 PARAMETER ESTIMATION OF THE PIPE CONVEYING FLUID SYSTEM WITH AN ADAPTIVE UNSCENTED KALMAN FILTER

Parameter estimation helps tracking mechanical system degradation by localising and quantifying the level of damage. This can be done through virtual sensing and prior knowledge about the dynamics of the mechanical system. In hydro turbines, since a significant part of the system is submerged, quantifying the level of damage requires an inspection process, and consequently a costly downtime. Here, we propose the UKF algorithm to perform parameter estimation during normal operation, which reduces downtimes and costly inspections. For the same established reasons as in Section 1, the analysis is performed on the nonlinear pipe conveying fluid equations. Figure 5.1 presents the schematic flow diagram of the proposed UKF algorithm.

Figure 5.1 illustrates the UKF used in this project. An optimisation procedure with the NOMAD algorithm of Audet et al. (2022) is needed to estimate $\mathbf{Q}$, $\mathbf{R}$, α_Q and α_R that facilitate the UKF control and estimation process. With the physic-based model and the EVB-reconstructed data, the UKF estimates both states and parameters of the pipe system through its classical prediction-update scheme. Hence, with the UKF, our aim is to solve the inverse problem through parameter identification, which could be beneficial for performance monitoring, quantification of unknown parameters, and virtual sensing in hydro turbines. In fact, in hydro turbines, many phenomena can lead to failure, and their characterisation often requires identification of latent quantities such as dynamical parameters. In addition, turbine sensors can give misleading results depending on their location and number (Trivedi and Cervantes, 2017) hence the need to solve the system identification problem.

5.1 Methodology : UKF General Architecture

NOMAD optimising: We use the NOMAD algorithm of Audet et al. (2022) and the methodology of Ghorbani et al. (2024) to find the best set of error covariance matrices and their forgetting factor. Finding the optimal hyperparameters by trial and error can be tedious. Hence, we use a Physic-Aware (PA) loss function within NOMAD to ensure the compliance of the estimate with the physical model. With this improvement to the UKF, the PA loss function $\mathcal{L}$ in NOMAD is solved by using the UKF state and parameter estimations within Equation (2.36) and by applying the L2-norm such that:

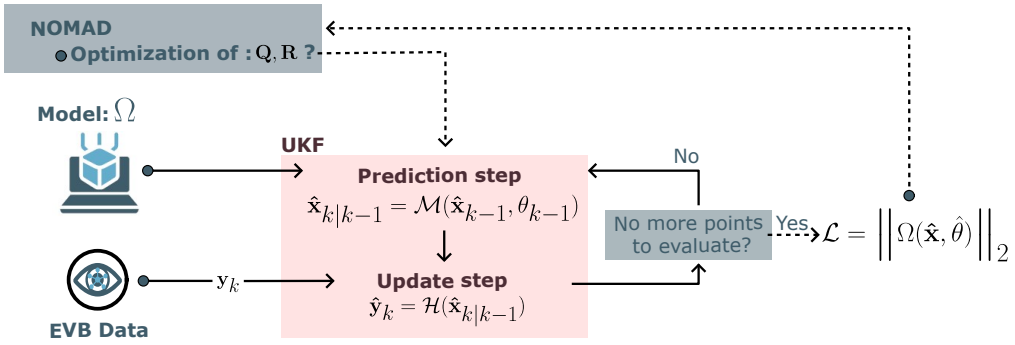

Figure 5.1 **Schematic flow of the proposed UKF algorithm.** The procedure has two paths : (·····) NOMAD optimised $\mathbf{Q}$ and $\mathbf{R}$ and minimised the PA loss $\mathcal{L}$ from the UKF estimates; (——) UKF estimates both unknown states $\hat{\mathbf{x}}$ and parameters $\hat{\boldsymbol{\theta}}$ in its prediction phase with the model Ω as prior knowledge. It then updates its estimates by comparing with measurements $\mathbf{y}$.

$$\mathcal{L} = \|\mathbf{M}\ddot{\boldsymbol{r}} + \mathbf{C}\dot{\boldsymbol{r}} + \mathbf{K}\boldsymbol{r} - \mathbf{F}\|_2 \,, \tag{5.1}$$

Accelerations $\ddot{\boldsymbol{r}}$ are evaluated by deriving the states $\dot{\boldsymbol{r}}$ with the CN scheme in a post-process. From a theoretical point of view, Equation (2.36) is conservative and should always be equal to zero. However, during inference, the UKF brings some corrections to the states that introduce a non-zero residual, the greater the corrections the higher the residual. Hence, NOMAD solves multiple UKF cases in order to find the best hyperparameters that minimise the PA loss. With such minimisation, we ensure that the UKF brings only minor corrections to the physics predicted. Consequently, the identified system can autonomously model the data, as the physics-based prediction requires only minimal Bayesian correction, indicating convergence. All these cases have the same data sensors and nonlinear pipe parameters. To simplify the optimisation process, we use scalar matrices proportional to the identity matrix. Hence, NOMAD optimise solely four scalar values to find $\mathbf{Q}$, $\mathbf{R}$, α_Q and α_R. To bound the optimisation process, we set the initial values of $\mathbf{Q}$, $\mathbf{R}$, α_Q and α_R given the range $[10^0, 10^{-11}]$ for $\mathbf{Q}_0$ and $\mathbf{R}_0$ and the range [0, 1] for forgetting factors and run the UKF. The PA loss function is calculated and a new set of covariance matrices is tested. The sensor data and the pipe parameters can be chosen given any general case as long as the pipe data (length, weight, diameters) is consistent with those of the one that is actually studied. With its optimising process, NOMAD looks for minima (Audet et al., 2022). However, for every physical dynamics, one of the minima is the trivial solution of zero. Hence, we add a heuristic to the PA loss, making sure that the averaged state and parameter estimates are always larger than zero. In this work, our objective is to use NOMAD once and reuse the

same hyperparameters for all flow cases studied. Since the procedure can be time consuming, solving it once reduces the potential downtime.

Initialising: The state $\mathbf{x}$ and the associated uncertainty $\mathbf{P}$ at t_{k-1} are used to start the UKF procedure. The Kalman Filter aims to estimate the state one time step ahead while minimising the trace of P. The state vector is as followed:

$$\mathbf{x}_{k-1} = \begin{bmatrix} \mathbf{q}_{Nx1} & \mathbf{p}_{Nx1} & \dot{\mathbf{q}}_{Nx1} & \dot{\mathbf{p}}_{Nx1} & \boldsymbol{\theta}_{M\times 1} \end{bmatrix}^\top, \tag{5.2}$$

where $\mathbf{q}, \mathbf{p}, \dot{\mathbf{q}}$ and $\dot{\mathbf{p}}$ are respectively the dynamical states in x and y directions, and their derivatives. $\boldsymbol{\theta}_{Mx1}$ is the vector of unknown parameters with M the number of unknown parameters considered. In the case of the nonlinear pipe model, such unknowns are the flow rate U, the Young modulus E, and the Kelvin-Voigt damping E^*. Lastly, the time step Δt_{UKF} has to be set according to the sensor recording rate. In this work, we consider the mode shape of the $N = 8$ first beam modes to reconstruct the motion of the pipe in both the x and y directions.

Prediction step: We perform the prediction of $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ first by propagating the sigma points in the nonlinear physic domain. Given N_λ, the unknown number of states and parameters, we propagate the $2N_\lambda + 1$ sigma points by solving the nonlinear Equation (2.37) of the pipe one time step ahead. We use a second-order implicit CN scheme to estimate the next-time step's state vector.

Update step: The second step is performed by comparing the predictions with the sensor data to get the estimates. In our experimental case, we use EVB sensors to extract displacement in points along the pipe for each time step $\Delta t = 0.025$ s. Hence, to compare the displacement sensor's data with the UKF time function's estimate we use the following measurement matrix:

$$\mathcal{H}(t_k) = \boldsymbol{\Phi}_{ij}(\xi_j)^\top \hat{\mathbf{x}}_i(t_{k-1}), \tag{5.3}$$

where $\boldsymbol{\Phi}(\xi_j)$ is a matrix of size $2N \times j$ containing the mode shapes $\phi(\xi)_i$. Here, j is the beam eigenfunction at each discretised point from 0 to L and i is the number of unknown state displacements. Following this correction, the UKF updates $\mathbf{P}$ and continues this two-step iterative procedure until no more data is available.

5.1.1 Computational Burden

The Unscented Transform in the UKF implies the resolution of the $2N_\lambda + 1$ nonlinear system of equations to map the physical domain, where $N_\lambda = 2N + M$ is the number of unknown in the state vector and M the number of unknown parameters. The higher the nonlinearities

and the larger the values of N and M, the longer it takes to compute. In our case, only considering 8 mode shapes in both directions, we have to propagate 65 sigma points per iteration for the state estimation. Increasing the state vector with unknown parameters would consequently increase this number. Figure 5.2 presents the computational time in seconds for the propagation of the sigma points per iteration.

Figure 5.2 shows that the computation of the nonlinear system of equations with the UKF for less than 50 sigma points takes less than a second. However, in the pipe system of equations, 50 sigma points represents a decomposition with 6 mode shapes. In Figure 2.1, six modes were the minimum number to achieve amplitude convergence with the fluid flow model of the pipeline. Then, it is irrelevant to consider less than 50 sigma points without compromising the model accuracy response. Despite this critical number of sigma points, an Intel Core i9 CPU alone performs with difficulty with a maximum of 60 s per iteration for $\chi = 97$. For $\chi = 65$, the computational time is around 6 s/iter. This computational complexity comes from (i) the naive implementation in a *for* loop of the propagation of sigma points, (ii) the fourth-dimensional (4D) matrices used with the nonlinear coefficients of the pipe conveying fluid model, and (iii) the inversion of the mass matrix $\mathbf{M}$ in Equation (2.37). Thus, we propose the parallelisation of the sigma points propagation with the Python framework `RAY 2.41.0` to solve (i) combine with a pre-compilation of the UKF via the Python just-in-time compiler `Numba 0.61.0` to overcome (ii-iii). By parallelising the UT we decrease the computational time by 84 % going from 6 s/iter to 1 s/iter for $\chi = 65$. Adding the just-in-time compilation procedure, we propagate 65 sigma points in 0.1 s/iter, representing an additional decrease of 90%.

5.1.2 Motivation of the UKF Compare to the PINN

The system identification process has been performed in Demenois (2022b) with a PINN algorithm. The idea was to identify the unknown flow rate by adding physical-based knowledge of the linear pipe equation to a neural network. With PINN, Demenois (2022b) manages to estimate the unknown flow rate with an average error of 3% considering the simulated data and of 7.3% with the experimental data.

First, PINN must be trained before performing the system identification. For 500 000 iterations, the PINN model was trained with 100 data points, 100 penalisation points, and 100 test points. The training took about 405 seconds. Second, the PINN used the linear model as prior. However, Demenois (2022b) noted that some experimental phenomena were not perfectly described by the linear model, making the PINN estimates a little off the ground truth. Moreover, the linear model is characterised by a damped oscillation, meaning that

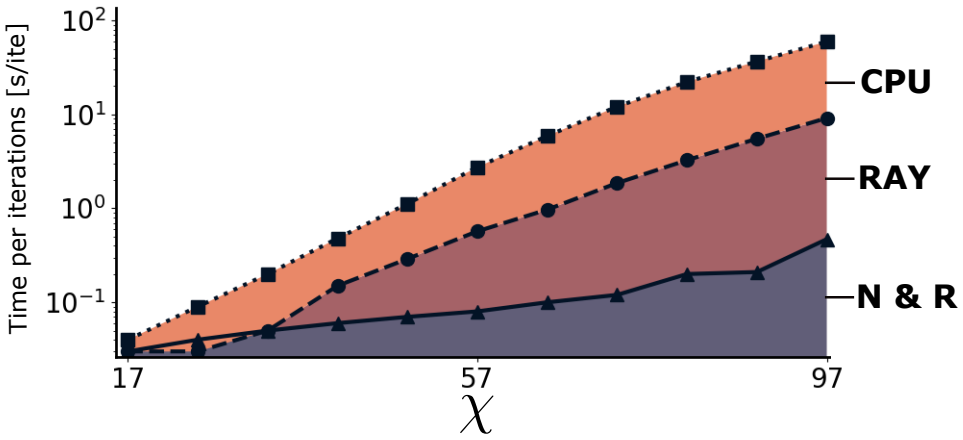

Figure 5.2 **UKF computational time per iterations according to the number of sigma points (χ)**. In **a**, we compare the computational time of the CPU intel core i9 (·····□) alone with the addition of the paralleliser RAY (- - - -◯) and the just-in-time compiler Numba (N & R)(——△).

at one point there is nothing more to observe. Hence, with the PINN of Demenois (2022b), the procedure is limited by a time-consuming step before the system identification and is restricted to observation of the linear dynamics. In this work, we use the UKF since we can perform system identification without training the algorithm at first. We introduce the nonlinear pipe model inside the UKF as prior knowledge in order to have a continuous system to study and to overcome some of the experimental phenomena unobserved with the linear model.

5.2 Simulation Study Results

We study the nonlinear pipe system considering Equation (2.37) to simulate sensor measurements. Synthetic data from the Crank-Nicolson discretisation process and sampled with $\Delta t_{UKF} = 0.004$s for a fixed time of 30s are used. We consider the parameters of pipe No. 1 of Demenois (2022b). The algorithm is tested with noisy displacement measurements on the x- and y-axes for a 10-point discretised pipe. We add 5% white Gaussian noise to the system to generalise the problem. With the DT's goal to perform system identification, we aim to estimate the unknown parameters of the pipe knowing its displacement in both directions at all 10 points. Thus, the augmented-state vector is of size 35×1 and can be summarised as:

$$\mathbf{x}_{k-1} = \begin{bmatrix} \mathbf{q}_{8\times1} & \mathbf{p}_{8\times1} & \dot{\mathbf{q}}_{8\times1} & \dot{\mathbf{p}}_{8\times1} & u & E & E^* \end{bmatrix}^\top, \tag{5.4}$$

where u is the dimensionless form of the flow rate U according to Table 2.1. The initialisation values of the covariance matrices and the joint vector x_0 are shown in Table 5.1:

In Table 5.1 each state value in $\mathbf{x}_{\mathbf{q},\mathbf{p},\dot{\mathbf{q}},\dot{\mathbf{p}}}$ is multiplied by a unit vector of size 8×1. The diagonal of the covariance matrix $\mathbf{P}$ is filled with the variance values of the given table such that $\text{diag}[\mathbf{P}_{\mathbf{x}_{0,\mathbf{q}}}, \mathbf{P}_{\mathbf{x}_{0,\mathbf{p}}}, \mathbf{P}_{\mathbf{x}_{0,\dot{\mathbf{q}}}}, \mathbf{P}_{\mathbf{x}_{0,\dot{\mathbf{p}}}}, \mathbf{P}_{E^*}, \mathbf{P}_u, \mathbf{P}_E]$. We set $\mathbf{R}$ and $\mathbf{Q}$ as $0.01 \cdot \mathbf{I}_{35\times35}$ and $0.0001 \cdot \mathbf{I}_{20\times20}$ respectively without the need for NOMAD. Since prior knowledge perfectly matches the physical equations used to build the synthetic data, the UKF is free from modelling errors. The UKF is executed over 2000 iterations. Lastly, we set the scale parameter for the sigma point λ of Equation (2.12) to 0 and the other scaling parameters $\alpha_\lambda = 0$, $\kappa = 0$, $\beta_\lambda = 2$, and $N_\lambda = 35$. The results of the system identification problem are shown in Figure 5.3 for six case trials of dimensionless flow rate in the range of 15 to 20.

Figure 5.3 shows the capacity of the UKF to solve the system identification problem for a pipe conveying fluid system as illustrated in **a)**. For flow trial $u = 16$, the Young modulus E, the Kelvin-Voigt damping parameter E^* and the flow rate u are all simultaneously estimated with an accuracy percentage of 0.8, 2.85 and 0.16 %, respectively. We show in **b)** the estimation of the parameters given a dimensionless time range τ of 0 to 8. The pale halo around the dashed lines of the estimations is the uncertainty associated to $\mathbf{P}$.

For the six case trials in **c)**, the relative error of the estimation of the unknown parameters is shown. With the same $\mathbf{Q}$ and $\mathbf{R}$ and initial values, the UKF always estimates the flow rate within the error range of [0.08, 0.4] %. The Young modulus is also accurately estimated within the error range of [0.6, 2.0] %. The Kelvin-Voigt damping estimates are within [0.05, 6.0] %. Hence, in our case, with a physical model that matches the incoming data, the UKF accurately estimates the unknown parameters with an average error of 1.52%. We see that the flow rate estimates are more accurate than the ones of the Young's modulus and the Kelvin-Voigt damping. This comes from the initial values and variance injected inside the UKF. For the flow rate, we assumed an initial value relatively close to the ground truth

Table 5.1 Initial values at time $t = k - 1$ to start the UKF.

$\mathbf{x}_{k-1}$	Initial value	Variance
$\mathbf{x}_{0,\mathbf{q}}$	0.001	0.01
$\mathbf{x}_{0,\mathbf{p}}$	0.001	0.01
$\mathbf{x}_{0,\dot{\mathbf{q}}}$	0.1	0.1
$\mathbf{x}_{0,\dot{\mathbf{p}}}$	0.1	0.1
E^*	2500	10^7
u	12	10
E	200000	10^9

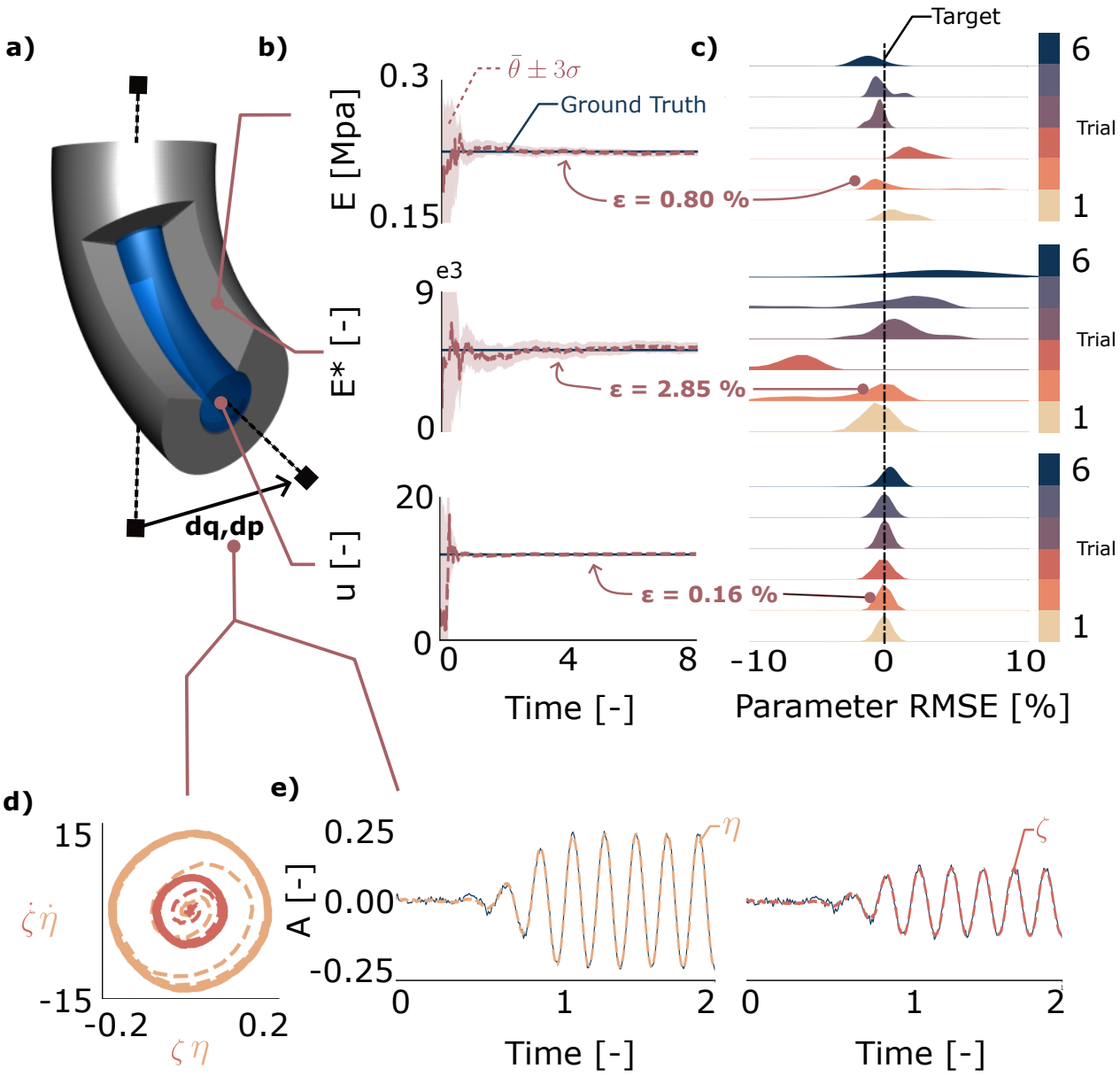

Figure 5.3 **UKF numerical results with synthetic data considering the parameters of the pipe No. 1 of Demenois (2022b)**. (**a**) illustrates a section of the pipe making a small displacement $d\mathbf{q}, d\mathbf{p}$. (**b**) presents the UKF estimation (- - - -) of the Young Modulus (E), the Kelvin-Voigt damping parameter ($E*$) and the flow rate (u) for a dimensionless time range of 8. Results come from the case trial $u = 15$. (**c**) shows the relative error from their ground truth for the unknown parameter identification for all six case trials of $u = 15$ to 20. Other than the parameter estimation, the UKF performs the estimation of the pipe velocity (**d**) and displacement (**e**) for both xz (pale orange) and yz (dark orange) planes.

and a high confidence in this value, which was not the case for the two other parameters. The bigger the difference with the ground truth and the smaller our initial confidence, the harder it is for the UKF to accurately estimate these values. Inverse problems are ill-posed, leading to non-unique and noise-sensitive solutions (Chen, 2003). The UKF addresses this through the use of a proper prior, which acts as a form of regularisation. Here, the Gaussian prior act as a Ridge regularisation and is essential for obtaining robust state estimations despite system noise, the number of unknowns, and physical nonlinearities. Indeed, nonlinear optimisation maps are often filled with minima, and finding the exact solution in the case of the inverse problem can be tedious when the initial values are uncertain and far away from their ground truth. Adding more unknowns, such as states, to the estimation procedure also complicates the resolution. Figure 5.3**d-e)** presents the estimate of the state of the pipe. In **d)**, the unknown velocity of the pipe is estimated and shown in a phase plane. in **e)**, the displacement along the discretised pipe length is known, which explains the good fitting to the ground truth. Compared with the real data, the relative RMSE of Equation (4.5) is 0.46% and 0.48% given the xz and yz planes, respectively. We found that considering some displacement states as unknown complexified the inverse problem too much for the UKF. Overall, since the measurement comes from the same prior knowledge injected inside the UKF, the latter estimates well the unknown states and parameters of the system.

5.3 Experimental Results

To validate the UKF capability to solve the system identification problem for the pipe system, we assume that the flow rate is the only unknown parameter. To experimentally define the values of E and E^*, we follow the procedure of (Paidoussis, 2014). The material properties of the silicon-rubber manufacturer are not accurate and may change depending on the accuracy of the mixing ratio employed. Thus, the method consists of characterising the pipe parameters using the free vibration response. The tracking of the pipe-free vibrations is performed with the EVB 2D reconstruction algorithm by manually letting go of the pipe perpendicularly to the camera used. We then find the natural frequency and logarithmic decrement of our pipe. With these two values, we use the `Pipe_Characterisation_Free_Motion.py` code of Demenois (2022a) to extract the values of E and E^*. With the given procedure, we find $E = 515118$Pa and $E^* = 9250$Pa. The experimental results for the characterisation of the pipe by free vibration are in Appendix C.

The best $\mathbf{Q}_0, \alpha_Q$ and $\mathbf{R}_0, \alpha_R$ are found among 180 NOMAD evolutions. To initialise NOMAD we set all the diagonal values of $\mathbf{Q}$ as 10^{-2} and of $\mathbf{R}$ as 10^{-5} and assume $\alpha_Q = \alpha_R = 1$. The initial states x_0 are $p_0 = 0.1$, $\dot{p}_0 = 0.01$, and $u = 6$. The other non-mentioned values of the

state vector are zeros. Since the experiments showed a perpendicular movement to the yz plane, the initial states are chosen to reflect a representative initial motion. The diagonal terms of covariance $\mathbf{P}$ are set to 10^{-4} for $\mathbf{q}_{8\times1}$ and $\mathbf{p}_{8\times1}$, 10^{-2} for $\dot{\mathbf{q}}_{8\times1}$ and $\dot{\mathbf{p}}_{8\times1}$, and 4 for θ_u. Since measurements are pipe displacements, we assume good confidence, whereas larger values are used for the velocity and flow rate, since they are unknown. Finally, we set the scale parameter for the sigma point λ of Equation (2.12) to -32.67 considering $\alpha_\lambda = 1e-1$, $\kappa = 0$, $N_\lambda = 33$, and $\beta_\lambda = 2$. We used the experimental data for the flow trial $U = 5.37$ m/s as the measurement for all NOMAD iterations. The time step $\Delta t_{UKF} = 0.025$ s is employed in its dimensionless form following the transformation in Table 2.1 and each NOMAD evolution is run for 400 measurement iterations. Among the 180 evolutions, NOMAD finds the best values for the diagonals of $\mathbf{Q}_0 = 0.000788^2$ and $\mathbf{R}_0 = 0.0195^2$, while $\alpha_Q = 1$ and $\alpha_R = 0.996$. With these error covariance matrices and the same initial state vector $\mathbf{x}_0$ and covariance $\mathbf{P}$, the results of the UKF are presented in Figure 5.4.

Figure 5.4**a)** shows the PA loss results and the convergent flow rate estimation in each NOMAD evolution. We can observe the convergence of the optimisation process and the minimisation of the PA loss from a maximal value of 6.62×10^2 to 2.46×10^{-5}. The best PA loss has been obtained in the NOMAD evolution $N_{NOMAD} = 93$. System identification problems imply that we deal with an under-determined problem whose output can be the result of different combinations of states. Hence, NOMAD optimises its parameters to find the best solution considering the possibility of many. However, in its optimisation process, it may encounter minima. From the converged flow rate raster plot in Figure 5.4**b)**, we see minima around the ground truth of this problem at $U = 5.37$ m/s, but also other minima at $U = 3.23$ m/s. With the NOMAD optimised values $\mathbf{Q}_0$ and $\mathbf{R}_0$, the UKF estimates the flow rate $\hat{u} = 5.51$ m/s, which represents a relative error of 2.6 % with the true value as presented in Figure 5.4**c)**. The 3D planar motion is illustrated in Figure 5.4**g)** with its respective views of the dimensionless planes $\zeta\xi$ and $\eta\xi$. The pipe shows a planar movement perpendicular to the $\eta\xi$ plane. The UKF reconstructs the movement of the pipe with a relative RMS error of 0.41% in the $\zeta\xi$ plane and 0.83% in the $\eta\xi$ plane. According to the Figures 5.4**d-f)**, the UKF manages to filter the noise from the signal. In Figures 5.4**h-j)**, where the motion is large, the UKF smooths and fits its estimate at the tip. However, from the middle region (η_5) to the anchor point (η_1), the UKF adapts its estimate according to its prior knowledge. The UKF estimates a larger amplitude at the middle point in comparison to the experimental data, while at the anchor point it estimates a solution with folded oscillation. These variations in the UKF state estimates are due to the prior knowledge used. During the prediction step of the UKF, we observe that the estimates tend to follow more the physical model. However, since the update is done by comparing the estimates with the data, the UKF

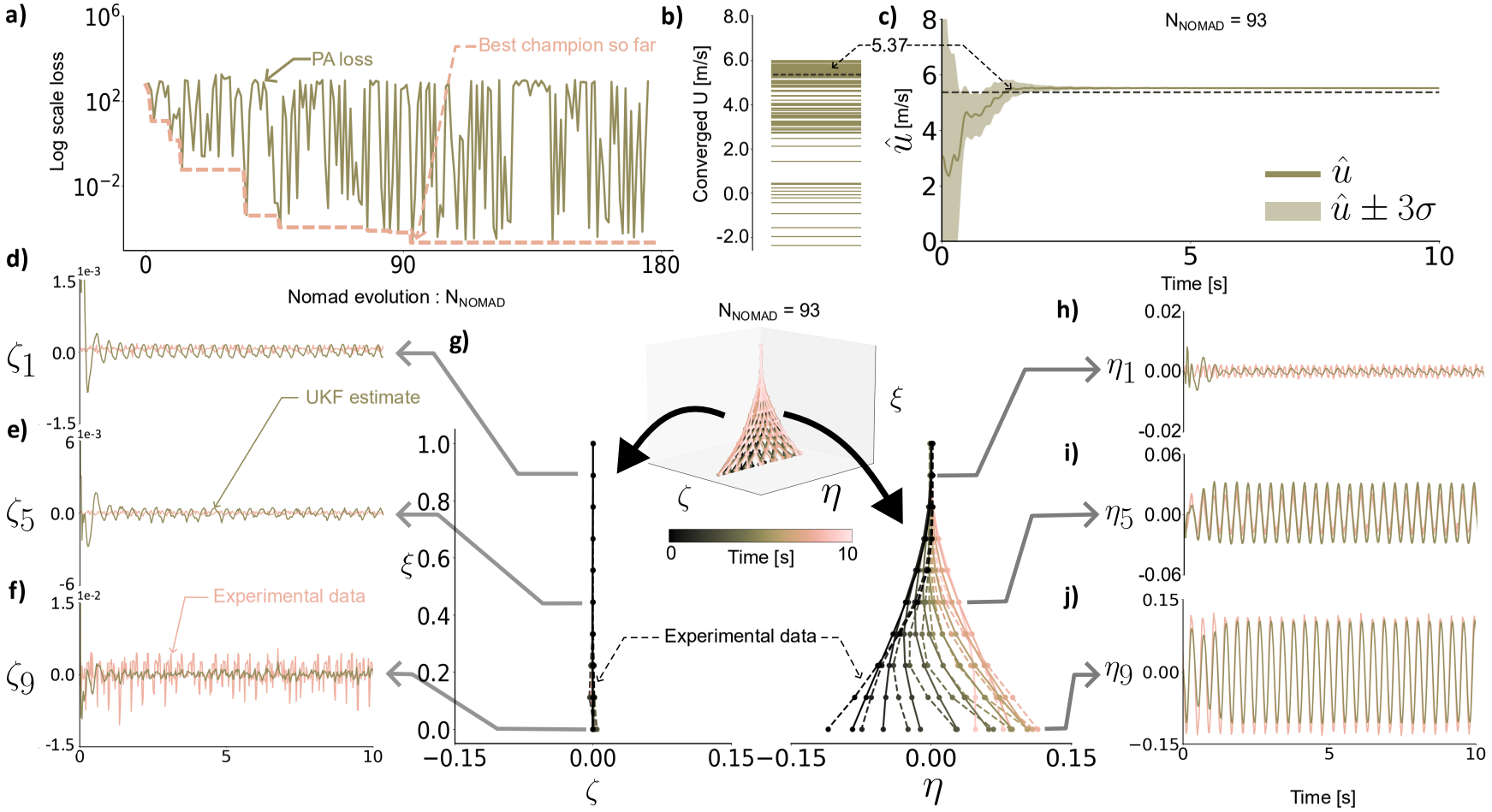

Figure 5.4 **UKF estimation of the unknown flow rate using experimental data**. **a)** is the evolution of NOMAD PA loss based on the number of iterations; **b)** is the converged value of the flow rate for each NOMAD evolution. **c)** corresponds to the best flow rate estimation that minimises the PA loss function. From **d - f)** and **h - j)**, the UKF state displacement estimate is compared with the experimental data at three locations : the pipe anchor (ζ_1, η_1), the middle of the pipe (ζ_5, η_5), and the tip (ζ_9, η_9). In **g)**, a 3D view illustrates the entire reconstruction of the state of the pipe with the UKF and their respective views of the dimensionless planes $\zeta\xi$ and $\eta\xi$ in their dimensionless form. The reconstruction of the UKF pipe (——) is compared with the experimental reconstruction (- - - -) over a time period of $t = 10$ s.

makes a trade-off. The latter results in the UKF trying to find the estimates that best fit the data while keeping the accuracy of the physical model. Since the physical model involves reductive assumptions, the estimations are biased by the latter. In Figure 5.5, this limit of prior knowledge is illustrated by comparing the experimental data with the UKF estimates.

Figure 5.5**a)** shows the comparison between the experimental amplitude of the RMS tip and the theoretical ones. The theoretical results come from the CN scheme run alone with the pipe No. 3 parameters for 10000 iterations and a dimensionless time step $\Delta\tau = 0.003$. The initial conditions are all zeros except $q_0 = 0.1$ to make sure that the oscillation is fully planar and perpendicular to one plane as previously observed. We run the theoretical model on 20 flow case trials ranging from [5.06, 6.90] m/s. Compared with the experimental data, the theoretical critical flow rate is observed at 5.45 m/s, while experimentally the flutter starts at U = 5.21 m/s. This difference represents an error of 4.6 % between the physical model and the theory. This difference is in accordance with the error limit of 10 % observed by Paidoussis (2014). Since the pipe parameters such as E and E^* are found numerically, it is possible that their values are not exact, which could explain the small error. Instead of using the Kelvin-Voigt damping assumption, Paidoussis (2014) proposes using a shaker to excite the lower modes and use the latter compared to the theoretical model. In addition to the critical flow rate difference, we can also see that the theory shows an amplitude that increases abruptly and caps at a higher value than the experimental results, which vary smoother with a lower amplitude. This difference between the model and the experiments can result from manufactured imperfection in the pipe and in the clamping support (Paidoussis, 2014; Païdoussis et al., 2007; Kheiri, 2020). Kheiri (2020) showed numerically that the added effect of both types of imperfection would decrease the amplitude bifurcation, while the slope from the linear region's flow rate and the subcritical flow rate is more gentle as observed in this project.

Since the theoretical models do not match the experimentation, it is difficult for the UKF to accurately find the right estimates. The best the algorithm can do is to minimise the error between the physical model and the data. This limit of the UKF can also be observed in Figure 5.5**b)** where the flow rate estimates are compared with the true value measured on the flow metre. We find these flow values using the same $\mathbf{Q}_0$ and $\mathbf{R}_0$ obtained with the NOMAD optimising process in Figure 5.4. However, we find that a fix $\mathbf{R}$ gives better results than its adaptive counterpart. Keeping a fixed $\mathbf{R}$, the UKF model remains general and the increase in flow rate is observable within the estimates of the UKF parameters. We refer to Appendix (D) for details on this observation on the generality of $\mathbf{R}$. We can observe that the flow estimates follow a linear increase with a slope smaller than that in the true experimental data. This difference in slopes comes from the gap between the theory and the

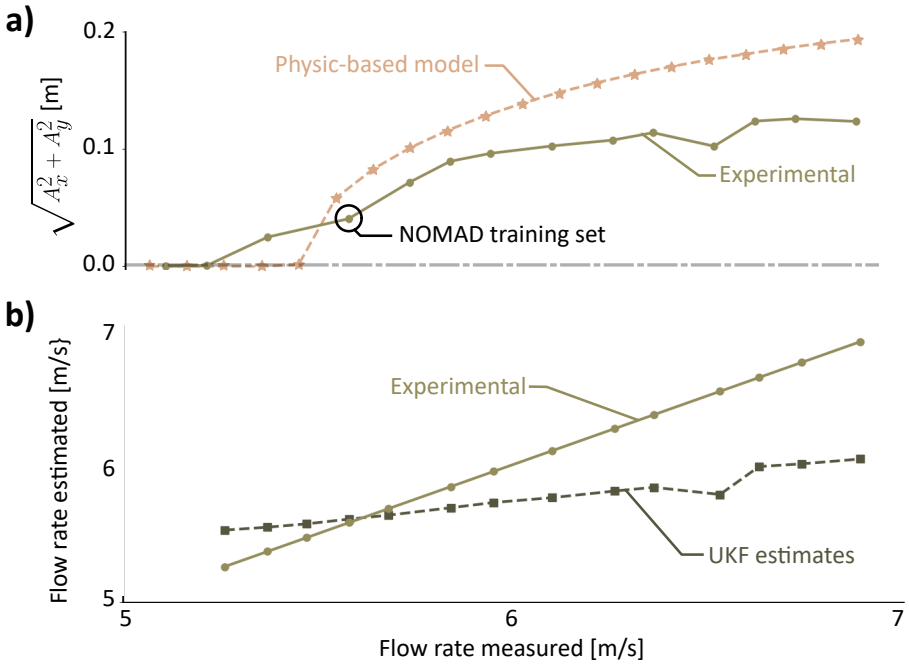

Figure 5.5 **Comparison of the experimental data results with the physic-based model.** **a)** The RMS tip amplitude of the experimental data (——◯) compare to the theoretical physic-based model (- - - -★) considering the same pipe parameters. **b)** Flow rate identified by the flow metre (——◯) compare with the UKF parameter estimation (- - - -□).

experiments. For small oscillations with an amplitude of 0.1 m and less. The relative error ranges from 0 to 6 %. This error increases further once the amplitude is greater than 0.1 m with a maximum relative error of 12 %. Hence, from Figures 5.5**a - b)** and the results obtained with the numerical simulation in Section 5.2, it is clear that the UKF estimations depend largely on the assumptions of prior knowledge. A well-defined physical model makes the UKF estimates accurate and precise. However, it is also important to note that the data quality has to be representative of the prior knowledge so the UKF can extrapolate a general solution.

Moreover, initial conditions such as the initial states, covariance $\mathbf{P}$ and time step Δt_{UKF} have to be chosen properly so the system can converge. State values far from their ground truth would imply a standard deviation in $\mathbf{P}$ large enough so that the UKF can catch the global minima and converge to it. However, because of the high nonlinearities in the function, it might converge inside minima instead. Helping the algorithm by setting the initial states and covariance $\mathbf{P}$ near the right answer can help the UKF. In Figure 5.5, for all flow rate trials, we assume the same initial flow value to start the UKF. However, as the flow rate increases, the difference between the initial value and the ground truth increases, making it harder for the algorithm to converge to the right value. Lastly, the right Δt_{UKF} is important so that the CN numerical scheme is convergent. Hence, the sensor rate has to be in accordance with the time integrator in the UKF so that the estimation process can work properly.

In addition to these initial condition importance, we note that the assumption of white noise applied to the signal response is not sufficient. White noise implies the assumption of independent noise and flat power spectral density which is not true for mechanical systems and sensors (Jerath et al., 2018). Identifying the right sources of noise and the impact on mechanical properties and sensing is beyond the scope of this project. However, it could improve the UKF capacity to accurately solve the system by better accounting for uncertainties.

Finally, the UKF used in this project is also limited by its computational time. In fact, for all of the flow trials mentioned above, the average computational time for 800 iterations is 120 s. Compared to the real-time oscillation of 20 s, the UKF is 6 times slower. From a DT perspective, this computational time delay is a limiting issue that could lead to miss some future failure events. Hence, even though the algorithm is parallelised and compiled, the computational burden of solving a high-order model remains.

CHAPTER 6 ANOMALY DETECTION THROUGH A LINEAR KALMAN FILTER EMPOWERED BY A DYNAMIC MODE DECOMPOSITION SCHEME

Here, to overcome the dynamicity limit of the UKF, we propose to monitor the pipe's data displacement with a DMD method combined with a LKF. As seen in the previous chapter, the UKF convergence is fast, but the computational time is tedious. Hence, we assume that we do not need to solve the full-order model of the pipe over the entire time length. In fact, we expect the pipe conveying fluid to repeat its limit cycle oscillation over time even after the UKF estimations. Hence, a reduced-order model of pipe dynamics is used to accelerate the monitoring to make up for the time delays caused by the UKF. With this fast solver, we also aim to detect anomalies by analysing the modal variations of the pipe over time. Early detection of anomalies is the first step before performing predictive maintenance. It could help optimising maintenance procedures by being more reactive to potential failures than periodic inspection and by giving more time to plan maintenance.

DMD is purely data-driven and reduces the problem dimensionality through a singular value decomposition (SVD). Instead of providing a modal decomposition with their energy information, such as POD, DMD models how the states evolve over time (Brunton and Kutz, 2022). Thus, compared to POD, the dimensionality reduction obtained with DMD is more useful here as it allows the prediction of the state of the system in the next time step. To perform estimation and control of the measurement data in real-time, the LKF is used. With DMD dimensionality reduction and by assuming that the unknown parameter estimations from the UKF are converged and constant, the complexity of the pipe dynamics is reduced. Thus, the computational resolution of the low-order model is faster. Figure 6.1 presents the workflow of the proposed algorithm.

In Figure 6.1, DMD is first used to evaluate the best-fit linear operator $\tilde{\mathbf{A}}$ obtained from the estimates of the UKF states. We propose training the DMD on the updated state estimates of the UKF. After k iterative steps, we observe a repetitive limit cycle oscillation of the pipe. Moreover, the system identification process is converged, meaning that the UKF does the monitoring of the pipe through high-order prior knowledge. However, the benefits of running this computationally expensive model on repeated cycles of oscillations are limited, as we expect the dynamics to remain at this steady state. Thus, we build a reduced-order model trained on the updated UKF state estimates vector. This allows us to make a trade-off between the fidelity simulations of the dynamics and the experimental observations. Thus,

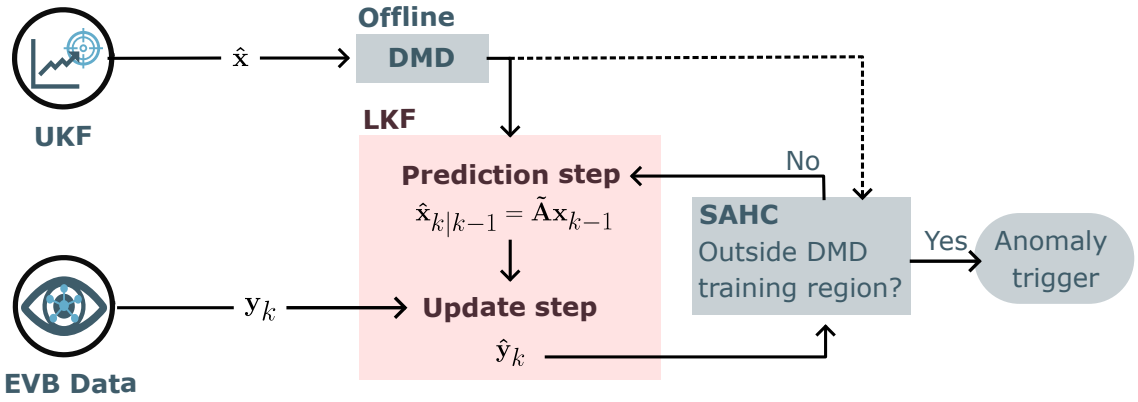

Figure 6.1 **LKF-DMD schematize workflow**. The UKF estimated states $\hat{\mathbf{x}}$ are fed into the DMD, which finds the best-fit linear operator $\tilde{\mathbf{A}}$ and reduces the system to a lower rank with SVD. With the EVB reconstructed data and the reduced-scale singular values $(\tilde{\mathbf{U}}, \tilde{\mathbf{\Xi}}, \tilde{\mathbf{V}})$, the LKF predicts and updates the next time-step modal-state vector. Anomalies are detected with a SAHC neighborhood algorithm that evaluates the Euclidian distance and the entropy between the estimate and the known phase diagram obtain with DMD.

the phase diagram extracted with DMD is less noisy and takes into account both physical understanding and data knowledge.

The associated singular value matrices from the updated state estimations represent the DMD model used as a surrogate state transition inside an LKF to perform state estimation and real-time monitoring. A Steepest Ascent-Hill Climbing (SAHC) algorithm evaluates the validity of the LKF estimation and detects if there are anomalies in the data. Hence, by performing real-time monitoring of the data, we aim to detect any perturbation to the system at the moment they occur.

In hydro turbines or any mechanical system, anomaly detection is relevant for predictive maintenance and condition maintenance procedures. Knowing and detecting such anomalies can optimise inspection strategies. In time series, anomalies can either be a point or a pattern difference. Finding these peculiarities in the overall response is known as anomaly detection (Yu et al., 2020). In hydro turbines, anomalies are various and Favrel et al. (2024) based their analysis on two possible scenarios: (I) sudden perturbation, and (II) loss of efficiency. In (I) the related perturbations can be due to sensor defects. In (II), the loss of debris from some mechanical part or the ageing of the turbine can reduce the efficiency of the machine. This decrease in efficiency is always characterised by a modification of the vibration characteristic of the machine, and therefore its mode shape is altered (Trivedi and Cervantes, 2017).

Applied to our pipe case, we use these two scenarios to define our own anomaly study cases. We can simulate scenario (I) by externally perturbing the pipe or masking one of the cameras

during the recording to create a sudden perturbation. The masking of the camera will be characterised by a short-term loss of information, but will not change the system behaviour over time. We assume that if we can detect such a small variation in the data, we could expect our method to catch more flagrant anomalies in the system. For scenario (II), we can modify the flow rate by adjusting the pump frequency over time. By adjusting the flow rate, the amplitude of oscillation of the pipe will change consequently. Finally, from the DMD training base on the UKF estimate, we aim to prove the importance of hybrid-modelling in a complex mechanical system. As stated in Giroux et al. (2023), the challenges with a single model applied solely to data are the accuracy of the data itself and the availability of historical data. Data from maintenance activities can be sparse and scarce, while lacking in standardisation, making their use difficult. In addition, without proper expertise, the data can be difficult to correlate and explain physically. Hence, training the DMD exclusively on the experimental data is limited by the data availability and quality and does not give any physical explanation of the system behaviour. From another point of view, DMD could be trained on known physics. However, it would be biased by reducing assumptions taken into account in the physical model scenario. Since the UKF can correlate and correct the data by relying on its prior knowledge, the trade-off between both physics and data is interesting for improving the data interpretation and availability.

6.1 Methodology: LKF-DMD Architecture

The LKF-DMD procedure follows the steps of Algorithm (3).

***DMD reduction*:** We use a predefined time window to extract n data snapshots. These snapshots contain the UKF state estimates obtained during a prior pipe system identification process. These snapshots are arranged as follow:

$$\mathbf{X} = \begin{bmatrix} \eta_0(t_0) & \eta_0(t_1) & \cdots & \eta_0(t_{k-1}) \\ \vdots & \vdots & & \vdots \\ \eta_1(t_0) & \eta_1(t_1) & \cdots & \eta_1(t_{k-1}) \\ \zeta_0(t_0) & \zeta_0(t_1) & \cdots & \zeta_0(t_{k-1}) \\ \vdots & \vdots & & \vdots \\ \zeta_1(t_0) & \zeta_1(t_1) & \cdots & \zeta_1(t_{k-1}) \end{bmatrix}, \quad \mathbf{X}' = \begin{bmatrix} \eta_0(t_1) & \eta_0(t_2) & \cdots & \eta_0(t_k) \\ \vdots & \vdots & & \vdots \\ \eta_1(t_1) & \eta_1(t_2) & \cdots & \eta_1(t_k) \\ \zeta_0(t_1) & \zeta_0(t_2) & \cdots & \zeta_0(t_k) \\ \vdots & \vdots & & \vdots \\ \zeta_1(t_1) & \zeta_1(t_2) & \cdots & \zeta_1(t_k) \end{bmatrix}, \tag{6.1}$$

where $\mathbf{X}$ and $\mathbf{X}'$ are the pair of shifted snapshot matrices containing the dimensionless displacement data estimated by the UKF, η and ζ are obtained through the same Galerkin decomposition of Equations 2.29- 2.30. Their index from 0 to 1 is their discretised location

along the length of the pipe in its dimensionless form. The snapshots used for DMD are acquired over a time duration t_0 to t_k, where we expect the pipe to exhibit nominal "healthy" behaviour with minimal perturbations and no anomaly. Then, we apply SVD on $\mathbf{X}$ to extract the reduced-scale singular value matrices $\tilde{\mathbf{U}}$, $\tilde{\mathbf{\Xi}}$, and $\tilde{\mathbf{V}}$. Here, $\tilde{\mathbf{V}}$ is the evolution of the mode shape in time given the length of $\mathbf{X}$ and $\mathbf{X}'$. The idea behind the proposed LKF-DMD procedure is to use LKF to estimate the reduced mode shape one time step ahead from the displacement sensor data and the snapshot information obtained from SVD. Hence, we solve Equation (2.25) to evaluate the best-fit linear operator $\tilde{\mathbf{A}}$. With the latter, we define the LKF transient function that best estimates the next reduced mode shape one time step ahead. $\tilde{\mathbf{A}}$, $\tilde{\mathbf{U}}$, $\tilde{\mathbf{\Xi}}$ and $\tilde{\mathbf{V}}$ are saved in a *.npy* format for latter use in the LKF and the SAHC procedure. These matrices are of size $r \times r$, $n \times r$, $r \times r$, and $r \times n$, respectively.

***Initialisation*:** We start the filtering procedure with a state $\mathbf{x}$ and a covariance matrix $\mathbf{P}$ at time t_{k-1}. The state vector is:

$$\mathbf{x}_{k-1} = \begin{bmatrix} \hat{\mathbf{v}}(0) & \cdots & \hat{\mathbf{v}}(r) \end{bmatrix}^\top, \tag{6.2}$$

where $\hat{\mathbf{v}}$ from 0 to r are the LKF estimated mode. In this work, we consider a reduction in dimensionality $r = 4$ since from SVD the first four modes are the most energetical. To start the LKF, we set the initial state vector equal to the time-evolutive modes at time step k in $\tilde{\mathbf{V}}$. We do so to ensure proper continuity between the data used in the training and the subsequent data input. $\mathbf{P}$ is set as $1e^{-1} \cdot \mathbf{I}_{4\times4}$. The $\mathbf{Q}$ and $\mathbf{R}$ matrices are set manually without the help of NOMAD. We find by trial and error $\mathbf{Q}_0$ and $\mathbf{R}_0$ as $1e^{-1} \cdot \mathbf{I}_{4\times4}$ and $5e^{-6} \cdot \mathbf{I}_{20\times20}$, respectively. We place such confidence in the $\mathbf{R}$ matrix to fit the incoming data as much as possible and perform anomaly detection. Finally, the time step Δt is equal to the EVB time interval of $\Delta t = 0.025$s.

***LKF prediction step*:** We no longer rely on the resolution of the complex implicit system of equations to solve the pipe system of equations. Instead, we estimate with the LKF the future modes $\mathbf{x}$ one time step ahead using the DMD best-fit operator $\tilde{\mathbf{A}}$ calculated with the time snapshots as the transition matrix. The rest of the prediction procedure is similar to the UKF where we estimate the next state vector and the covariance matrix given by $\tilde{\mathbf{A}}$.

***LKF update step*:**. The state vector is updated, and the trace of $\mathbf{P}$ is minimised by comparing the estimations with the sensor displacements. Since the estimations are the main modes from the oscillations, the measurement matrix is:

$$\mathbf{H} = \tilde{\mathbf{U}}\tilde{\mathbf{\Xi}}. \tag{6.3}$$

Multiplying $\mathbf{H}$ by $\mathbf{x}$ is equivalent to reconstructing the displacement vector from the SVD decomposition matrices. The LKF is optimal (Särkkä, 2011) and the Kalman gain correction minimises the difference between the estimations and the data. Thus, for a linear system state estimation, the procedure is one of control, and, if the LKF covariance matrices are well defined, the fitting between the estimates and data shall be optimal. We use the SAHC technique to detect any deviations in modal decomposition. From the updated estimates of LKF at time t_k, we use the neighbour search algorithm to evaluate the minimal Euclidean distance from the saved modal references $\tilde{\mathbf{V}}$. The SAHC is a naive algorithm in which the estimates are compared with all the trained modes $\tilde{\mathbf{v}}$ in $\tilde{\mathbf{V}}$. Thus, the minimal Euclidean distance $\mathcal{E}$ values at each time step is defined as:

$$\mathcal{E}_n = min\left(\sqrt{\sum_{i=0}^{r}(\tilde{\mathbf{v}}_r - \mathbf{x}_r(t_k))^2}\right), \quad \forall \quad \tilde{\mathbf{v}}_r \in \tilde{\mathbf{V}}. \tag{6.4}$$

Since the resulting $\mathcal{E}$ is a vector containing all the Euclidean distances with the estimates, we take the minimal value as the best champion so far. To detect an anomaly, we look at the variation of $\mathcal{E}$ in time from a particle point of view. If the trajectory trends of the particle should change, it would imply either a sudden extreme response in the case of scenario I or a slow decay for scenario II. Hence, to catch sudden perturbation, we ensure that the particle remains within $\bar{\mathcal{E}} \pm 3\sigma_{\mathcal{E}}$. The mean $\bar{\mathcal{E}}$ and its standard deviation are measured for the first 300 iterations of the LKF-DMD algorithm assuming the oscillation remains in a steady state. In the case of a flow rate change, the perturbation can be more subtle than a sudden burst. Hence, we used an entropy-based algorithm applied to the angle of the particle gradient. We assume that the particle has low-entropy information in a steady-state oscillation. If an anomaly should occur, the entropy would change as the angle of the particle gradient points either upward or downward. Hence, with the entropy method the anomaly is detected since a trend from the particle direction start to emerge and differ from the DMD trained modal phase steady-states. We use the Shannon entropy with the Python scipy `stats` module to catch the variation of the entropy $\mathcal{S}$ of $\mathcal{E}$. If the condition of $\bar{\mathcal{S}} \pm 3\sigma_{\mathcal{S}}$ is not respected, the system experiences a change in flow rate. Similarly to the mean Euclidean distance, the mean $\bar{\mathcal{S}}$ and its standard deviation are also measured during the first 300 iterations. The whole prediction-update-neighbour search steps are done iteratively while data are filled to the algorithm until an issue is flagged in the modes. We use the 3σ rule within the Euclidean and the entropy-based algorithm as a general criterion for anomaly detection. Figure 6.2 summarises the general methodology.

In Figure 6.2 **a-c)**, the displacement state of the UKF is integrated into the DMD procedure.

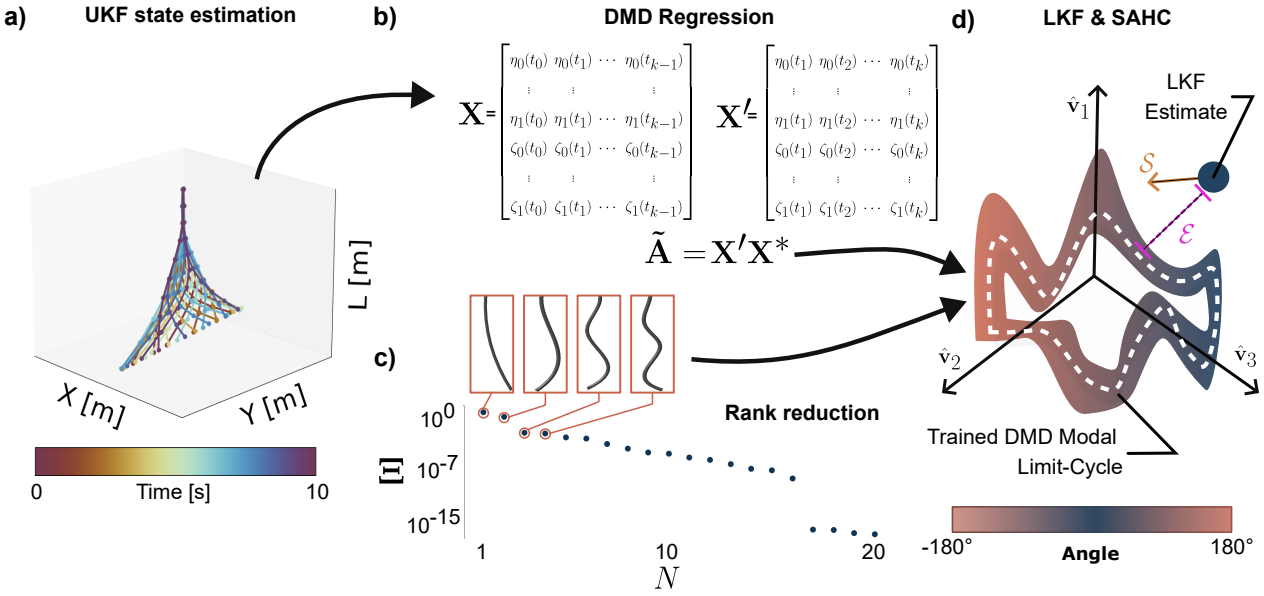

Figure 6.2 **Generalization of the LKF-DMD methodology**: **a)** updated UKF state estimations obtained from a time interval t of 10s; **b)** regression to obtain the best-fit linear operator $\tilde{\mathbf{A}}$ from the snapshots of **a)**; **c)** dimensionnality reduction to $r = 4$ modes due to their high energy; **d)** limit-cycle trajectory from the $\tilde{\mathbf{v}}$ coordinates with LKF estimated used by SAHC to detect anomalies from the Euclidean distance $\mathcal{E}$ and the entropy $\mathcal{S}$ criteria.

We perform SVD in (**c**) to reduce the dimensionality of the system to $r = 4$. We can easily observe that the pipe oscillations are dominated by the first four modes. With SVD, the evolution of the four main modes of DMD in time $\tilde{\mathbf{v}}_r$ is saved in the matrix $\tilde{\mathbf{V}}$. Since we want to detect an anomaly, we assume that any perturbation to the pipe should modify its modal response and is perceived by the SAHC in the LKF. Figure 6.2 **d**) schematises this anomaly detection procedure as described above.

6.2 Simulation Study Results

To verify the LKF-DMD, we use the synthetic data of pipe No.3 of this project (Table 4.1). We analyse anomaly detection based on the two scenarios of Favrel et al. (2024). Again, the displacement of the pipe is discretised to 10 equidistant points in the xz and yz planes. We fix a dimensionless time step $\Delta\tau = 0.004$. Synthetic data are obtained with the CN scheme described previously for the dimensionless flow case $u = 12$ and a time interval $\tau = [0, 30]$. We define the initial conditions for the CN scheme as $q_1 = 10^{-2}$, $p_1 = 10^{-3}$ and $\dot{q}_1 = 6 \times 10^{-2}$. The other unmentioned values in the initial vector are set to zero. The UKF and the high-fidelity model are used to generate the update state vector snapshots for the DMD training for 401 time steps. We inject the simulated data asynchronously into the LKF. For both

scenarios, we assume a near-perfect system with the addition of 0.5 % white noise.

6.2.1 Scenario I : Sudden Perturbation

Variations in mechanical system data can occur due to a sensor defect or a sudden shock. To simulate this scenario with synthetic data, we randomise the signal for a period of Δt_{rand} seconds using the `Numpy random` function in Python. Figure 6.3 presents the anomaly detection after a sudden burst in measurement.

Figure 6.3**a)** shows the tip amplitude in both planes. At approximately half of the simulated time, the sudden perturbation is visible. In **b)**, the SAHC Euclidean distance is calculated throughout the time interval. Once the LKF is launched, the SAHC is subsequently started. At the moment of perturbation, the loss varies because the estimations are no longer within the training range as seen in Stage III in the red box. Finally, since the perturbation is momentary, the system and the loss stabilise again. The observable peaks are due to the quality of the LKF estimates. Even though the LKF is optimal for the linear problem, it does not perfectly fit the right answer due to the uncertainty added to the model through covariance matrices. A perfect answer would imply overfitting from the algorithm. In addition to the perturbation region, the Euclidean distance remains within 0.047 ± 0.125. In some instances, $\mathcal{E}$ is outside the bounds of $3\sigma_{\mathcal{E}}$ even though no perturbation has occurred. In order to get rid of these instances, we use a naive persistency factor that triggers the anomaly if the perturbation occurs during five consecutive time steps. This persistency factor is a hyperparameter that must be set by the user according to the type of anomaly studied. Figure 6.3**c)** presents the entropy of $\mathcal{E}$. During the whole process, the entropy-based method does not see the sudden perturbation and $\mathcal{S}$ remains within 1.96 ± 0.190. This is because the entropy-based method analyses the trends in the data over a certain window length, making it less good to catch sudden irregularities.

Figures 6.3 **d)** to **f)** illustrate the modal phase diagrams. In all subfigures, modes 2 (**d**), 3 (**e**) and 4 (**f**) are shown as functions of mode 1 amplitude. Stage I is the modal phase diagrams obtained after the DMD training on the UKF state estimates. Stages II and IV show the modal phase diagrams of the LKF in the stable region. Stage III is the diagram during the perturbation time interval. We see clearly that the perturbation destroys the patterns during that short time interval, and the estimations from the LKF result in increased loss. In addition, we can see that the algorithm tracks the modal changes well for the first three modes $\hat{\mathbf{v}}_1$, $\hat{\mathbf{v}}_2$, and $\hat{\mathbf{v}}_3$. However, for the last mode $\hat{\mathbf{v}}_4$, the results do not fit well with the trained one. This comes from the fact that the UKF state estimates are devoid of noise. As shown in Section 5.2, the UKF tends to smooth its prediction based on its prior knowledge. Thus,

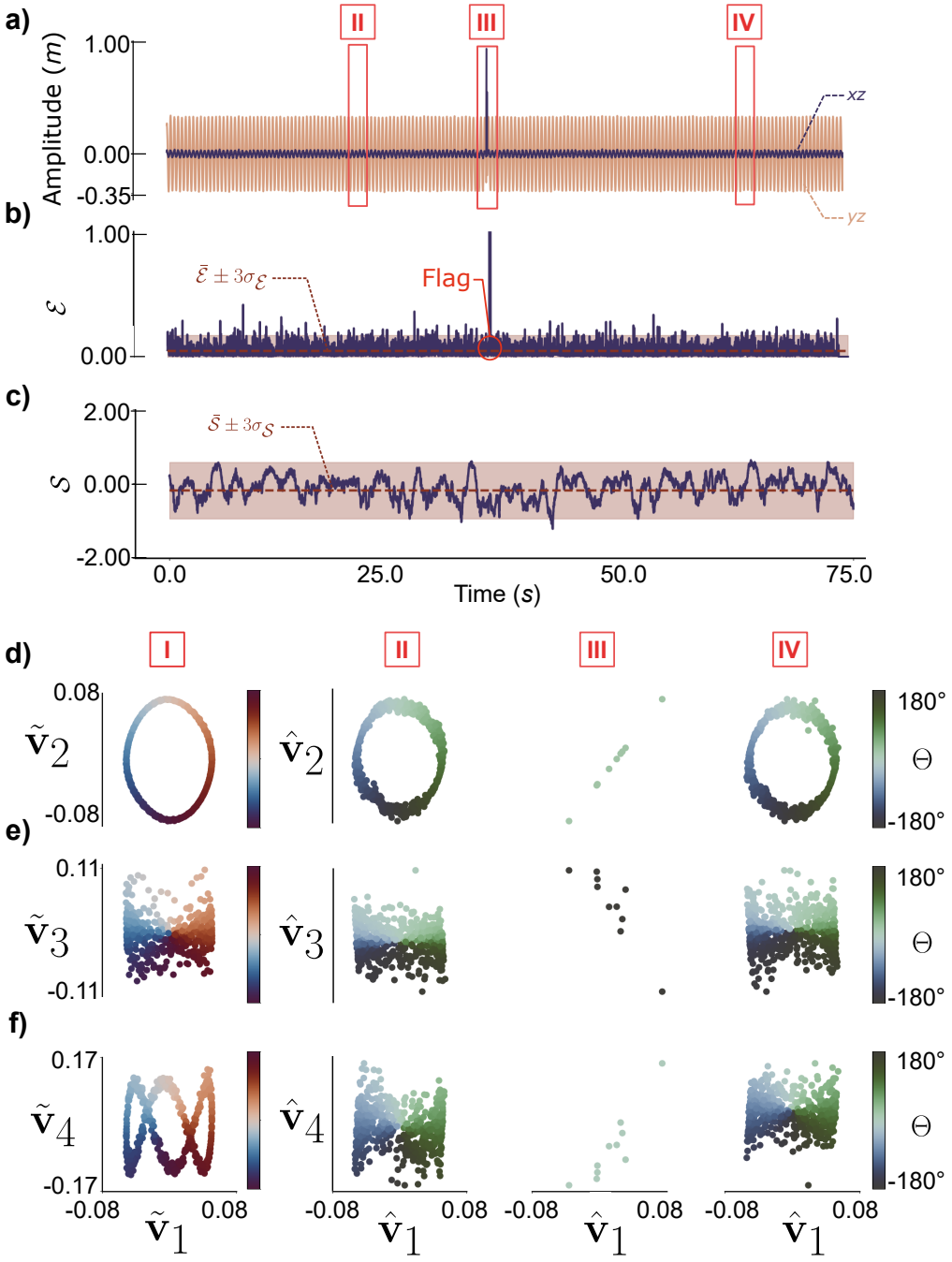

Figure 6.3 **Numerical results of the LKF-DMD applied on scenario 1 for a sudden perturbation of 0.1** s**: a)** the tip amplitude of pipe No.3 of the present study; **b)** the Euclidean distance between the trained modes from DMD and the LKF estimates; **c)** the entropy of the Euclidean distance; **d-f)** the observed modal phase diagrams at four stages, namely Stage I from the trained DMD for a time window of 400 iterations, Stage II to IV for the LKF estimations indicated on the timeline in a).

the extracted modal phase diagram from DMD is physically more meaningful than the one from the raw EVB reconstructed data affected by both noise and hardware defects. Since the comparison between the observation and the LKF estimates is done with Equation (6.3), due to their energy distribution with their singular values in $\tilde{\mathbf{\Xi}}$, the impact of reconstruction of the third and fourth modes is less important than the first two. That is, the odd reconstruction by the LKF-DMD for the last mode does not affect its capacity to capture the anomaly if the principal modes are well estimated. For the case illustrated in Figure 6.3, the instability appears at $t = 35$ s. With the given threshold, the anomaly is detected at the same time.

However, in the given case, the perturbation has an amplitude greater than the observed oscillation. Hence, the perturbation is outside the $3\sigma_{\mathcal{E}}$. To verify the robustness of the method, the algorithm is tested in two different cases. We first simulate a random response from the sensor by setting the observed oscillation to zero in a given time interval. For the second case, we scaled down the random perturbation to quantify the minimum observable perturbation amplitude. These simulations allow us to define the minimal variation observable by the proposed method. Catching such variations makes us suppose that longer changes in the signal response will be flagged as an anomaly. These simulations are in Table 6.1. For all cases, the perturbation appears at $t = 35$ s.

From Table 6.1, for a perturbation lasting for a time interval less than 0.8 s, neither the entropy-based method nor the Euclidean distance detects the perturbation. Once the time interval exceeds 1 s, the anomaly detection is performed by the entropy-based method over the Euclidean one. For the scaling-down analysis, the Euclidean method captures well the perturbation up to a factor greater than 100. As the scaled factor is further reduced, again, the entropy-based method captured the anomaly instead of the Euclidean one. Hence, as the time interval increases and the perturbation amplitude is near zero, the Euclidean method cannot flag the anomaly. Instead, the entropy method perceives this change in the sensor data as a signal angle variation. The Euclidean method evaluates the distance between the estimates and the modal phase diagrams. Due to the naive use of SAHC, as long as a point is close to one of the points of the modal phase diagram, the Euclidean distance assumes a minimal distance within its bound $3\sigma_{\mathcal{E}}$. This is true even though the corresponding point has no real correlation with the observation. Hence, if the abnormal point sequence stays near the steady-state observations, the difference is unseen. The entropy method sees the change only if the angle points in a specific direction for a time period long enough to leave the bound $3\sigma_{\mathcal{S}}$. Overall, for a sensor defect, the algorithm tends to catch the perturbation well if its amplitude is greater than the amplitude of the measurements. Also, looking at the computational time per iteration, one can remark the fast solving of the method. With an average computational time of 8.8×10^{-4} s/iter, the computational time is 11 times faster

Table 6.1 LKF-DMD robustness to perturbation time interval and to the degree of perturbation amplitude for a perturbation appearing at $t = 35$ s.

Time interval	Time lag [s]	Computational Time [s/iter]
0.1	-	8.9×10^{-4}
0.2	-	8.9×10^{-4}
0.4	-	8.9×10^{-4}
0.8	0.01	7.6×10^{-4}
1.0	0.31	7.6×10^{-4}
Scaled factor	Time lag [s]	Computational Time [s/iter]
0.0625	0.00	9.9×10^{-4}
0.03125	0.01	9.0×10^{-4}
0.015625	0.01	9.0×10^{-4}
0.0078125	-	9.0×10^{-4}

than the sampling time of 0.01 s used in the simulations.

6.2.2 Scenario II: Imposed Flow Velocity Change

The pipe conveying fluid does not have an efficiency like a hydraulic turbine does, but a change in its mechanical properties, its boundary conditions, or its flow rate can be understood as an analogous scenario. In all cases, the pipe's oscillating pattern progressively varies and consequently changes the healthy modal phase diagram on which the DMD was trained. Hence, we applied a linear decay to the displacement data to simulate scenario II. In reality, such a steep and perfect linear slope is almost impossible. However, with such gradual variations we can not only observe a change in modes but also quantify the minimal amplitude change perceived by our algorithm. Here, we assume that the imposed flow velocity change starts at $t = 40$ s and follow a linear decay given by:

$$A_N(t) = \frac{-\Delta(x, y)}{\Delta t} t + A_N(t = 40) \tag{6.5}$$

where A_N is the amplitude of the discretized pipe point, $\frac{\Delta(x,y)}{\Delta t}$ the slope of the amplitude decreased and t the time increment. Before $t = 40$ s, the maximal averaged amplitudes in both the xz and yz planes are respectively 0.33m and 0.028m. Figure 6.4 presents the results for a slope equal to 0.07.

In Figure 6.4**a)** we can observe the small linear decay of the tip of the pipe over a time interval of 75 s that is imposed at $t = 40$ s. Given the slope of Equation (6.5), the anomaly is detected at $t = 42.11$ s as shown in Figure 6.4**c)** with the entropy-based method. At that

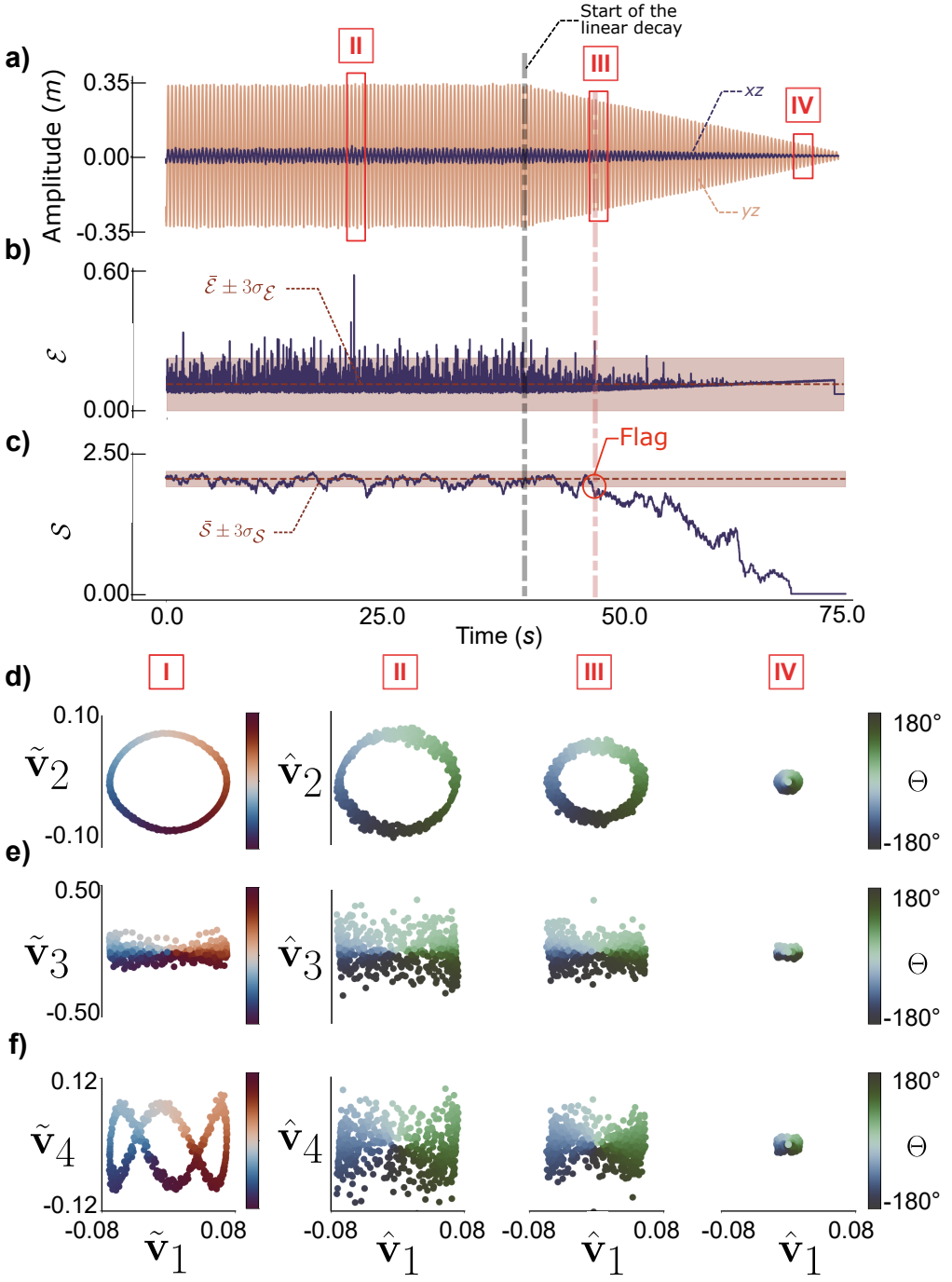

Figure 6.4 **Numerical results of the LKF-DMD applied on scenario 2 for a linear decay of the amplitude: a)** tip amplitude of pipe No.3; **b)** Euclidean distance between the DMD trained modes and the LKF estimates measured with SAHC; **c)** its entropy; **d-f)** modal phase diagrams at four stages. Stage I is the modal phase diagrams from the trained DMD for a time window of 400 iterations. Stages II to IV are the updated states of the LKF during the linear decay.

time, the amplitude in the xz plane is 0.31 m, and in the yz plane, 0.026 m. Hence, the imposed flow velocity change is observed once the amplitude has a difference of 6.6 % with its normal oscillation behaviour. Here, we see the advantage of the method compared to the Euclidean one, as the entropy variation in the angle direction is well managed. To flag the anomaly, we use a persistency factor of 100 iterations in the entropy-based method. In Figure 6.4**b)**, the Euclidean measure has a slight deviation upward as the amplitude decreases, but it is not sufficient for the algorithm to see the change. This proves the observation in the previous section in which the Euclidean method is not enough to capture small or smooth changes similar to the trained modal phase diagrams. This decrease in flow rate can be seen in the reconstruction of the mode pattern in Figures 6.4 **d)** to **f)**. Column I illustrates the modal phase diagram from the DMD training. In column II, because we are still in the stable region, the estimations agree with the patterns in column I. In columns III and IV, the decay in the pipe amplitude is observable. We can see a shrinkage of the modal phase diagram. Similarly to Scenario I, the fitting of the first three modes allows the algorithm to catch the anomaly rapidly.

To measure the efficacy of the entropy-based algorithm in various slope decays, we change the rate at which the flow velocity change occurs. Table 6.2 shows the result of the previous analysis.

From Table 6.2, the entropy-based method depends on the steepness of the slope decay. The more rapid the variations, the faster the algorithm sees the change. For slopes smaller than 0.029, the entropy-based algorithm cannot flag the anomaly, whereas for a slope larger than 0.070, the algorithm flags the anomaly near instantaneously. In neither case, the Euclidean distance captures the anomaly which shows that both methods are complementary. This is explained by the fact that from the precedent scenario, a sudden perturbation was char-

Table 6.2 Variation of the slope of the imposed amplitude decay to verify the entropy-based method efficacy to flag changes.

$\frac{\Delta(x,y)}{\Delta t}$	Time lag [s]
0.022	-
0.029	24.07
0.037	16.55
0.044	13.94
0.051	10.50
0.059	9.19
0.066	5.40
0.070	2.11

acterised by a large change in the amplitude response, which explains the efficiency of the Euclidean method to detect the perturbation. The perturbation also destroyed the observed pattern, making it easier for the algorithm to detect the change. Here, the shift is slower and the pattern remains. Hence, the Euclidean method alone is less effective in detecting a decrease in flow rate. Focussing on the entropy of the angle of the signal from a particle point of view allows one to catch more subtle variation such as a flow decay in the system.

6.3 Experimental Results

The algorithm validation is performed by running the UKF on the EVB sensor data. For all cases, we run the UKF on pipe No.3 of this project. To test Scenario I, we recorded 10 seconds of oscillation and considered a sampling rate of $\Delta t = 0.025$ s. To simulate a camera malfunction, we rapidly mask lens one of the camera with a small cardboard for a second. Table 6.3 shows the experimental results for two cases with different masking times.

Similarly to numerical tests, in Table 6.3, the perturbation is rapidly flagged. In both cases, the time lag is pretty small compared to the time in which we mask one of the cameras. However, when comparing the computational time in Table 6.1 and Table 6.3, the only difference is the computational time of the algorithm, which is 10 times slower. This difference comes from the necessity for the computer to execute both the anomaly detection algorithm and the post-treatment of the EVBs with the same processing power. Since the computational time per iteration is smaller than the sampling rate $\Delta t = 0.025$ s, the LKF-DMD algorithm could keep up with the incoming data stream of the EVB sensors in real-time.

To evaluate the code capacity to solve problems with scenario II and test its robustness when fed noisy data, we perform the change in the flow rate by increasing the pump frequency during the recording. We record the increase in flow rate for 75 s. Figure 6.5 presents the overall result for that scenario analysis. In Figure 6.5**a)** the modification in flow rate values is flagged by both methods (**b)-c)**) at time $t = 21.1$ s, as the change is steep. Figure 6.5**d)** shows the estimated modal phase diagram with LKF II - IV compared to the trained DMD patterns I. We can see that the orbit of the first two modal phase diagrams is slightly shifted

Table 6.3 Experimental results for a sensor defect obtain by masking one of the camera to simulate Scenario I.

Masking time [s]	Time Lag [s]	computational time [s/iter]
31.95	0.05	1.4×10^{-3}
48.05	0.03	1.5×10^{-3}

to the left. For the two other modes $\hat{\mathbf{v}}_3$ and $\hat{\mathbf{v}}_4$, the results do not fit well the trained ones. This discrepancy between the estimation and the training modal phase diagrams comes from the information considered in the snapshot matrices $\mathbf{X}$ and $\mathbf{X}'$. In this work, the DMD training is done only on displacement states with the assumption that it is the only available data. However, the nonlinear pipe system used is described by a second-order differential equation. Hence, the reduced-order models obtained with DMD do not capture the full system dynamics since the velocity states are not taken into account. Furthermore, the Euclidean distance does not see the perturbation due to our implementation of the 3σ rule. From the 3σ rule used in both methods, if the modal phase diagram does not correspond but has a constant bias with the ground truth, the Euclidean distance and its entropy remain in the given bound. Hence, the patterns do not need to be exactly matched by the algorithm to track changes. Instead, both methods evaluate whether the observations remain constant in time given an initial state. Lastly, in Figure 6.5**e)**, similarly to Table 6.2, a greater variation ΔU in the flow rate makes it easier and faster for the LKF-DMD method to identify the change. For all cases, the average computational time is 1.4×10^{-3} s/ite.

6.4 Understanding the Use of Hybrid Modelling in the DMD Training

To better understand the hybrid approach proposed within LKF-DMD, Figure 6.6 illustrates the modal phase diagram obtained following DMD training from I the physic-based model; II the UKF estimates; III the EVB reconstructed data.

Figure 6.6 shows the same modal combination of $\tilde{\mathbf{v}}_{1-2}$ **(a)**, $\tilde{\mathbf{v}}_{1-3}$ **(b)** and $\tilde{\mathbf{v}}_{1-4}$ **(c)** presented earlier. Considering solely the physic-based model in I, we can observe smooth cyclic modal phase diagrams. These phase diagrams are derived from a perfect simulation of the pipe conveying fluid system with no noise. Looking at the sub-figures in III shows that for the data alone, the modal phase diagram is corrupted by noise and artefacts. The latter are the result of EVB hardware defects and any corruptive dynamics such as water drops or passing shadows in the background. Hardware defects come from electric leakage and temperature change in EVB hardware as mentioned in Raynaud and Mulleners (2024). These defects cause a loss of light intensity that results in no captured information. As the measurement are tracked, the algorithm sees odd points that are not related to the system dynamics or anomalies. Training the DMD with the UKF in II acts as a trade-off between I and III. We observe a better correlation with the physics while being more representative of the data uncertainty by keeping some discrepancy with the perfect model. With the UKF estimates, the DMD method builds a surrogate model of the pipe dynamics. With this model, the uncertainty can be quantified, as the UKF estimates have their own standard deviation from

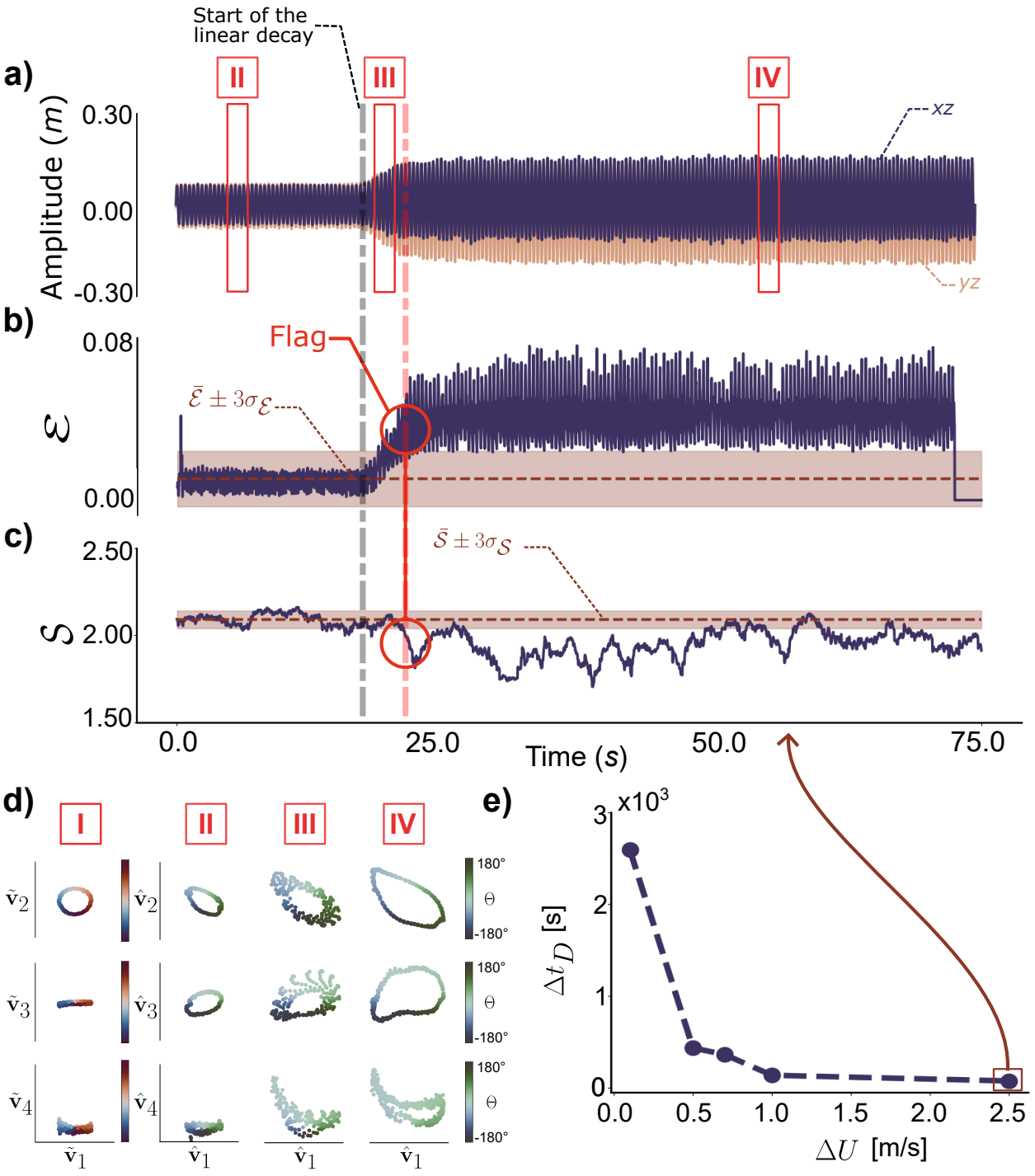

Figure 6.5 **Experimental results for an increase in flow rate: a)** tip amplitude of oscillation of the pipe in both planes for 75 s; **b)** the Euclidean distance obtained by running the SAHC with the LKF estimates; **c)** entropy-based results; **d)** modal phase diagrams from the trained DMD (I) and the estimated ones by the LKF (II-IV); **e)** entropy-based time lag Δt_D s for different flow variations ΔU m/s.

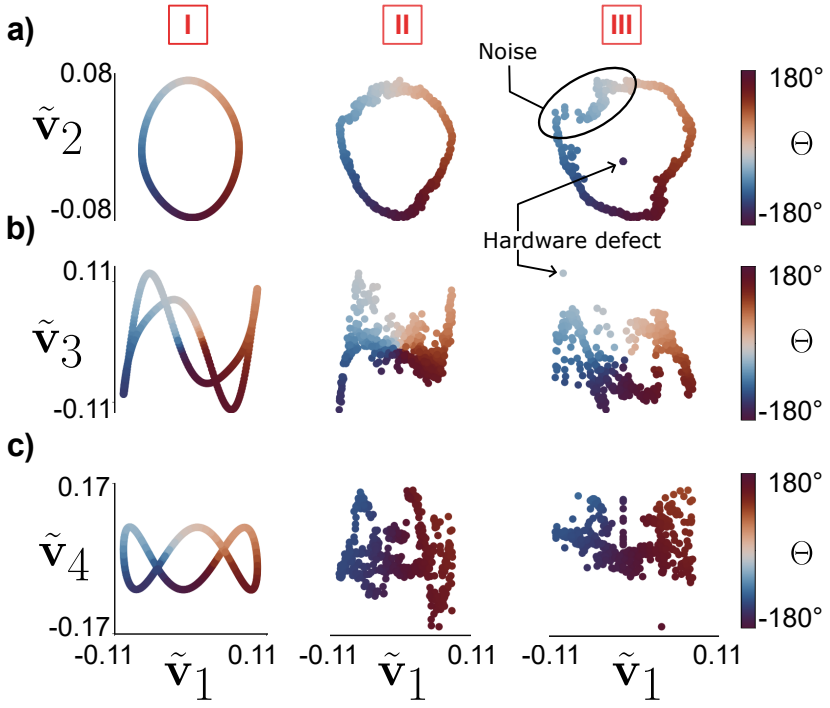

Figure 6.6 **Comparison between the DMD modal phase diagram from I the physic-based model, II the UKF estimates, and III the EVB reconstructed data for modes: a)** $\tilde{\mathbf{v}}_2$; **b)** $\tilde{\mathbf{v}}_3$; and **c)** $\tilde{\mathbf{v}}_4$ versus $\tilde{\mathbf{v}}_1$.

the covariance $\mathbf{P}$. Hence, using the UKF estimates can improve the data quality for DMD training, quantify their uncertainty, and correlate the modal phase diagram with the actual physics. In fact, for such patterns, the state estimates are correlated to a certain flow rate value, which gives a meaningful interpretation to the LKF-DMD method. However, this physical interpretation is limited by its reductive assumptions. Thus, the limits mentioned for the UKF remain and are passed on to the LKF-DMD. In our case, this is reflected in the discrepancies between the LKF-DMD estimates and the modal phase diagrams of $\tilde{\mathbf{v}}_{1-3}$ and $\tilde{\mathbf{v}}_{1-4}$.

CHAPTER 7 MODULAR DIGITAL TWIN FRAMEWORK

We propose a proof-of-concept for a DT by merging all the algorithms presented in this thesis into a workable framework. The framework performs anomaly detection, near-real-time monitoring, and system identification with the LKF-DMD and UKF algorithms. The general framework workflow was presented in Figure 1.1 in Section 1. Here, we focus on the fine details behind the proposed framework. The Python libraries and versions, and the code map of this DT framework can be found in Appendix (F) and (G) respectively.

7.1 Methodology : DT Framework Architecture

To demonstrate the DT proof of concept, the UKF is used to identify the unknown flow rate value and the LKF-DMD to flagged perturbations. Figure 7.1 presents the connective logical flow diagram of the main algorithm containing all the algorithms in this thesis.

In Figure 7.1 all the algorithms presented earlier are implemented inside a Python class function and are called into a main code. The recording and synchronisation procedure with the EVBs is performed with two independent terminals. In order to link these terminals to the LKF-DMD and the UKF, we use the Python `subprocess.popen()` function. The latter allows to connect the terminals with the main code and thus to merge both terminal data outputs.

The LKF-DMD needs to be trained on the UKF state estimates. Hence, we use the `sleep()` function of the `Time` module in Python to stack the incoming data in a First In First Out (FIFO) manner in a *csv* file. This stacking method allows us to periodically update the data injected inside the UKF and, latter, in the LKF-DMD. Also, it helps us to make sure that the system is in a steady state so that the estimations from the UKF and the DMD training are accurate enough with prior knowledge. Once the accumulated data length is greater than $t_{start}+600$ where t_{start} is the initial time step, a UKF system identification process is run until the standard deviation of the unknown parameter P_θ is convergent. Following the evolution of P_θ, we can analyse the confidence of the algorithm in its estimation. Once $P_\theta \leq 5$ %, we stop the UKF process and start the LKF-DMD anomaly detection procedure. The resulting flow value u_1 is saved as a reference for future use.

We first train the LKF-DMD on the UKF state displacement estimates. During that acquisition period, we assume that the data are not perturbed. Following the LKF-DMD training, we performed oscillating pipe monitoring to detect anomalies. When an anomaly is detected,

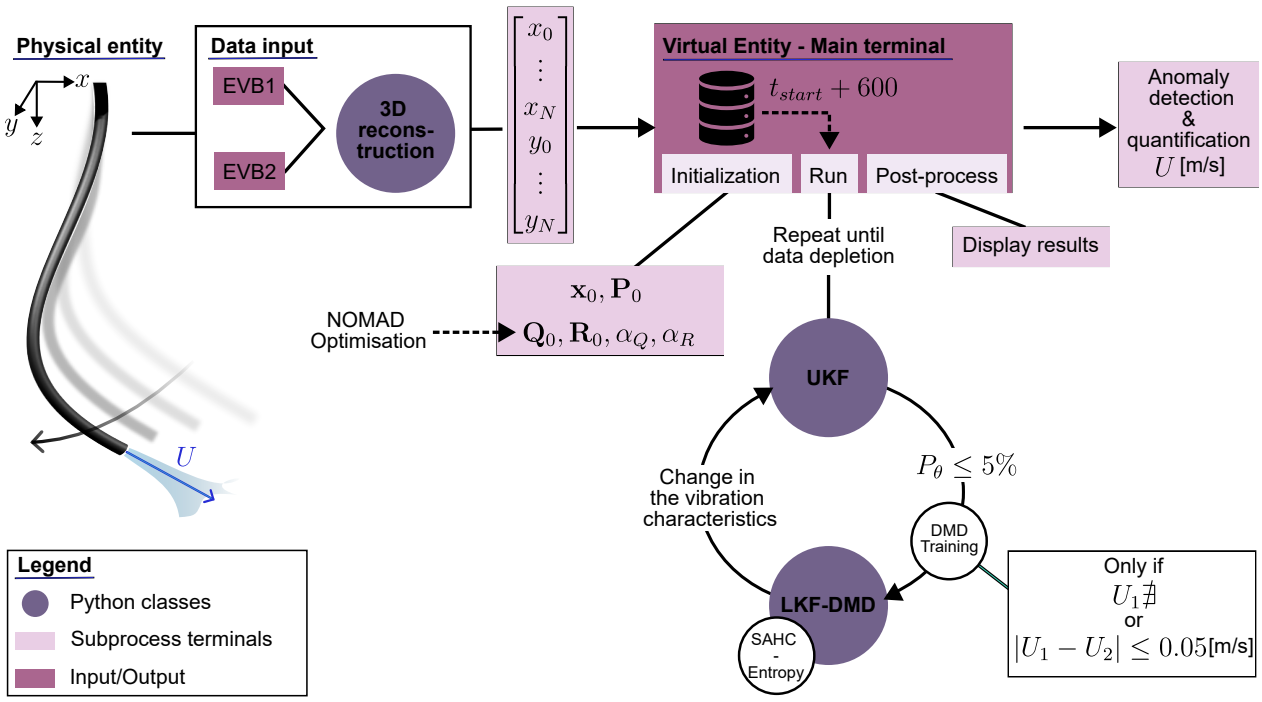

Figure 7.1 **Proposed algorithm flowchart for the proof of concept of the DT framework of the pipe-conveying fluid set-up.**

we accumulate the data again and run a new UKF system identification process. We stop the UKF procedure once the estimated covariance for the unknown parameters is converged. Then, we compare the new parameter estimations U_2 with the reference U_1. If the difference is less than $U_1 \pm 0.05$ m/s, U_2 is assumed to be equivalent to the reference, and the LKF-DMD procedure will continue to monitor it. Otherwise, a change in the parameter could signify that the system modal response is no longer the same. Hence, the LKF-DMD is trained on the new UKF state estimates, and the new reference parameter is changed to the latest UKF estimation. The difference of ± 0.05 m/s corresponds to the average experimental difference in the flow value of a 1 Hz change in pump drive frequency. The whole DT procedure ends once the data acquisition with the EVBs is stopped. This periodic update approach is inspired by (Giroux et al., 2023) where their DT is applied on a speed governor in hydro turbines. Since the evolution of anomalies in the hydro turbine is expected to be slow, a real-time procedure would be excessive and computationally expensive.

7.2 Experimental Results

7.2.1 Offline Results

To test the overall procedure, we first recorded in advance a flow rate decrease scenario during 75 s. With this test, we want to show the ability of the DT framework to automatically switch from a high-fidelity model to a lower one. This test is denoted as the *offline procedure*. The sampling rate with the EVB sensors is 0.025 s, while the initial states and covariance matrices for the UKF and the LKF-DMD are presented in Table 7.1. In the UKF, the values of $\mathbf{Q}_0$ and $\mathbf{R}_0$ are multiplied by $\mathbf{I}_{33\times33}$ and $\mathbf{I}_{20\times20}$, respectively. In addition, $\mathbf{x}_0$ is of size 33×1 and $\mathbf{P}_0$ of size 33×33. For LKF-DMD, $\mathbf{Q}_0$ and $\mathbf{R}_0$ are multiplied by $\mathbf{I}_{4\times4}$ and $\mathbf{I}_{20\times20}$, respectively. Lastly, $\mathbf{x}_0$ is of size 4×1 and $\mathbf{P}_0$ of size 4×4.

The diagonal values $\mathbf{Q}_0$ and $\mathbf{R}_0$ come from a NOMAD optimisation process carried out on 200 realisations. Both the process and the measurement forgetting factors $\alpha_{Q,R}$ are set to 1, which means that the covariance matrices are constant over time. In our case, constant covariance matrices enable us to keep the system generalisable for different cases. The results of this flow rate decrease scenario are shown in Figure 7.2.

In Figure 7.2**a)**, an anomaly associated with a change in pump frequency appears at $t =$ 18.4 s. Before detecting this anomaly, the UKF is run in (I) so we can identify the initial flow rate in the system. The UKF estimates a flow value of 5.71 m/s, which is off by an error of 2.2 % with an actual flow rate of 5.84 m/s, as shown in **b)**. Following this UKF estimation, the LKF-DMD procedure is started in (II) and the entropy-based method in **c)** detects an anomaly at $t = 28.4$ s, 10 seconds after the first real appearance of the anomaly. In (III), we can observe that the system accumulates data to ensure that the system is in a steady state. Following the new UKF procedure, the latter estimates a flow value of 5.51 m/s which corresponds to an error of 6.3 % with a true flow rate of 5.21 m/s. In (V), the LKF-DMD

Table 7.1 Initial states and covariance matrices to start both UKF and LKF-DMD algorithms in the DT framework.

UKF				
	$\mathbf{Q}_0$	$\mathbf{R}_0$	$\mathbf{x}_0$	$\mathbf{P}_0$
$(\mathbf{q},\mathbf{p})_{16\times1}$			0	1e-3
$(\dot{\mathbf{q}},\dot{\mathbf{p}})_{16\times1}$	1.4e-6	1.5e-3	0	1e-2
u			6	4
LKF-DMD				
$\mathbf{x}_{4\times1}$	1e-3	5e-6	$\tilde{\mathbf{V}}(t=k)$	1e-1

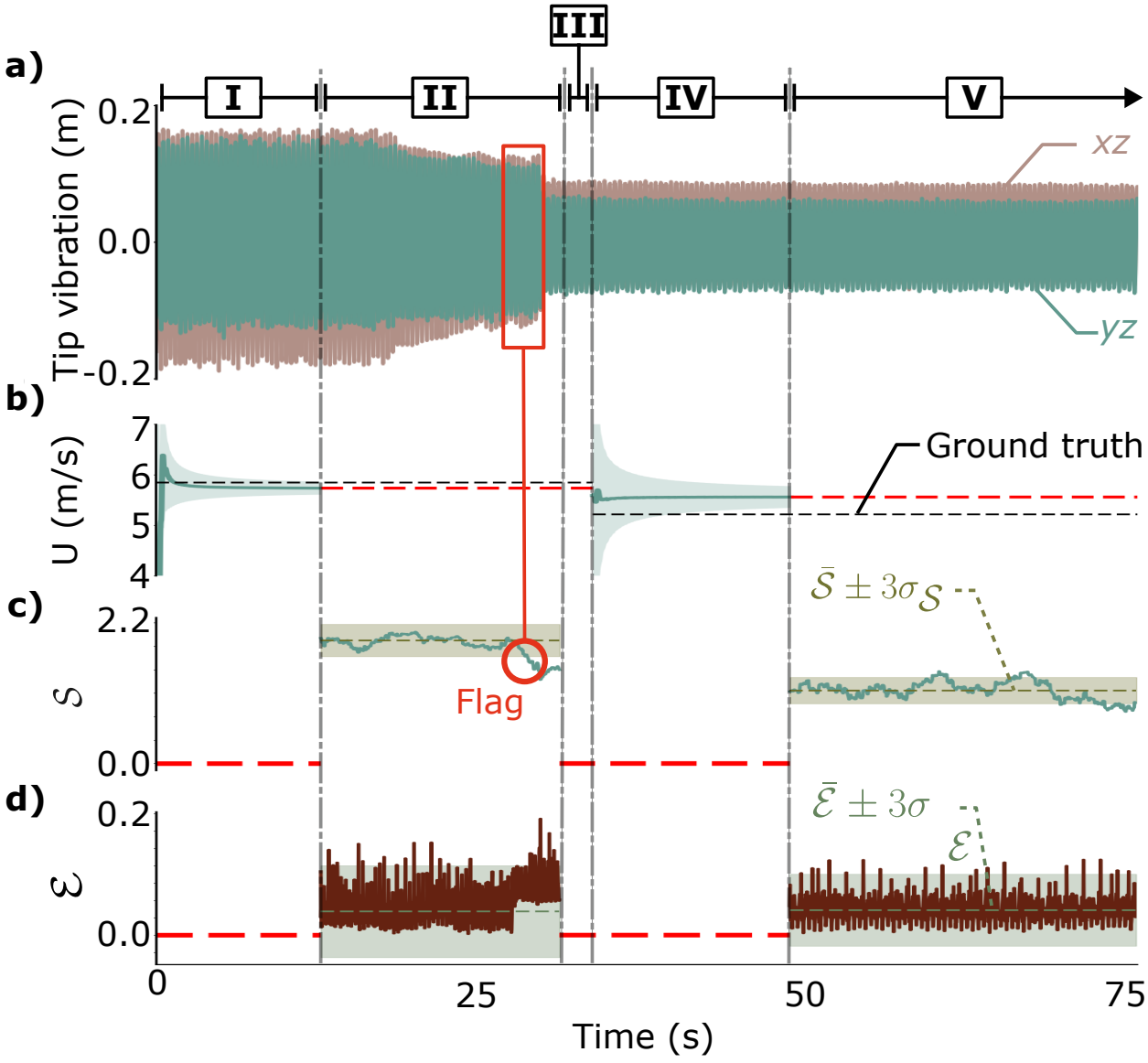

Figure 7.2 **Anomaly detection with the DT framework of a flow rate drop imposed at** $t = 18.4$ s: **a)** pipe tip amplitude oscillations; **b)** flow rate U estimation from the UKF processes (I and IV) compared with their respective ground truth; **c)** entropy based ($\mathcal{S}$); and **d)** Euclidean ($\mathcal{E}$) anomaly detection methods during the LKF-DMD (II and V) processes. The sequence initialises I) with the UKF performing the flow rate identification; then II) the LKF-DMD detects an anomaly; III) a time buffer to restart the procedure; IV) a new UKF parameter identification; V) the LKF-DMD monitoring until a new anomaly occurs.

procedure is performed until no more data are available or a new anomaly is detected.

In this scenario, a time lag of 10 s elapses between the occurrence of the anomaly and its detection. This lag can arise because the variation in the amplitude at first is not large enough. Hence, the entropy-based method in **c)** has difficulties in capturing the anomaly instantaneously. Also, since the perturbation gradually affects the dynamics of the pipe, the Euclidean method alone in **d)** is not sufficient to capture the change. Hence, the proposed DT framework is logically limited by the disadvantages of the algorithm and the models used. This is also true for the UKF, where we can observe that the second flow value once the system stabilised again has a larger difference with the ground truth than the first iteration of the UKF. Although the hyperparameters in the UKF are generalisable to other cases, their performance depends on the dynamic similarity to the trained data set injected in NOMAD. Thus, the UKF estimates can lose in accuracy as the pipe's dynamics differ more and more in amplitude.

7.2.2 Online Results: Processing Power Burden

The simultaneous operation of the EVB sensors, 3D reconstruction code, UKF, and LKF-DMD is computationally expensive for this project's computer. Since the Python codes are not optimised and run on the computer CPUs, the latter are rapidly overwhelmed with the number of tasks. Hence, to demonstrate this computational time and power limit, we use a simple test by masking one sensor. The recorded data are first filled into the DT framework in an online manner, which means that the whole framework is run simultaneously for 4000 iterations, that is, 75 s. Then, we compare the computational time of the algorithms by running the framework with the pre-recorded data. In this case, the computer does not have to power the EVB, and more CPU cores are accessible. The computational time between a *real-time* online running example and an offline post-process example is shown in Figure 7.3.

In Figure 7.3**a)**, the 75 s experimental test is separated in five zones. Initially, a UKF is run until the flow rate estimation in **c)** is converged. Here, the UKF (zone I) estimates a flow value of 5.62 m/s, which represents an error of 5.6 % with respect to the true value of 5.32 m/s considering that the pump frequency was set at 28 Hz. Following this convergence, the DT changes to the LKF-DMD (zone II) if the variance in $\mathbf{P}$ is less than 5 % and the number of iterations is greater than 500. We set this number of iterations constraint so that we can train the DMD on a sufficiently large number of limit cycles of the pipe. We make the assumption that the pipe is in a healthy state during that time interval. LKF-DMD runs until an anomaly is detected that occurs at $t = 30.2$ s. The algorithm detects the change at $t = 30.4$ s, which is a time lag of 0.2 s. Following this anomaly detection, we make sure the

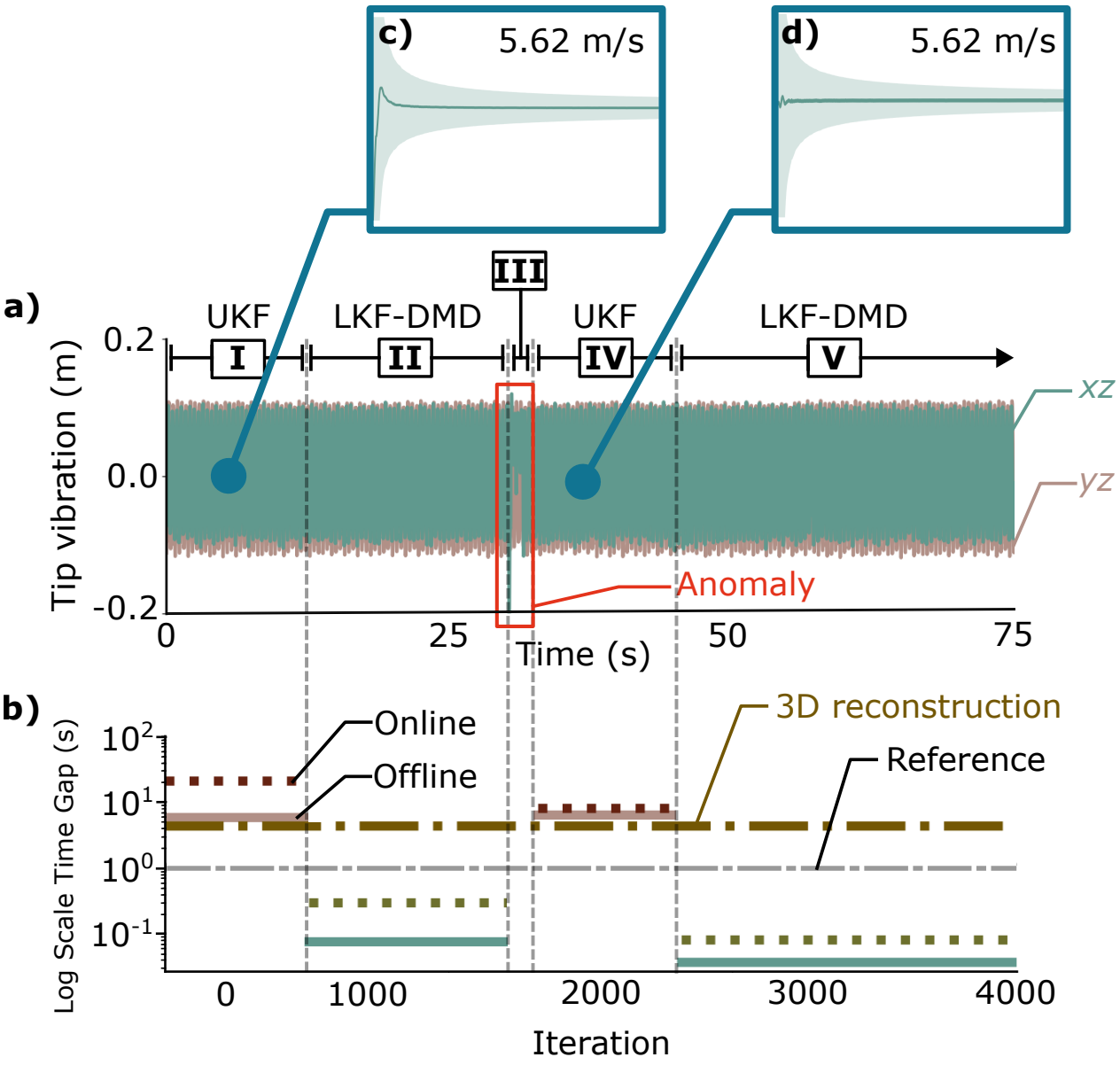

Figure 7.3 **Analysis of the computing power availabilities to run the DT framework in either an online or an offline manner for a sudden perturbation case: a)** pipe tip amplitude oscillations in both xz and yz planes; **b)** Time gap between a real-time processing (reference — · —) and the algorithm computational time run online (·····) or offline (——) for both UKF and LKF-DMD algorithms. The average computational time gap to run both EVB sensors and the 3D reconstruction algorithm is shown (— · —); **c)** UKF flow rate estimation at the beginning of the DT procedure; **d)** UKF flow rate estimation after the detected perturbation. In I), the UKF perform the flow rate identification; then II) the LKF-DMD monitor the oscillation of the pipe until an anomaly is detected; III) time-buffer to restart the procedure; IV) the UKF perform the new flow rate identification; V) LKF-DMD monitoring until a new anomaly is detected.

system is back into a new steady-state (zone III) and the UKF is re-launched (zone IV) until its convergence to the same flow value **d)**. The difference here is that the initial value for the unknown parameter is set to the previous dimensionless value obtained with the UKF in zone I. However, as expected, the second UKF makes the same estimation. Then LKF-DMD (zone V) is run again while the EVB data is available.

In **b)** the computational time to run the algorithms of this project is compared to the time in which the dynamics of the physical system is occurring. In other words, here we analyse the algorithms capacity to execute their simulation as fast as the actual wall clock time. Hence, the system identification process of the UKF or the anomaly detection of the LKF-DMD should meet the computational time reference of 1 second of CPU time for 1 second or less of dynamics time to have real-time calculations. The EVB powering from our PC and the running of the 3D reconstruction codes is on average four times slower than the real-time reference. Other than the CPU limit mentioned earlier, the number of events in each camera frame affects the algorithm's ability to run rapidly. Here, since the oscillation is slightly greater in amplitude in the yz plane than in the xz plane, the number of events is consequently also greater. Hence, the polynomial fitting is done through a larger number of points in the yz plane, and the computational time to reconstruct 75 s is 344 s compared to 298 s in the other plane. In addition to this time delay, the UKF is 21 times slower than real-time when running online and 7 times slower when offline. The EVB codes are started first, meaning that all CPUs are assigned. However, to run the UKF and both the `Numba` and `Ray` packages, enough CPU power is needed. Hence, if `Ray` cannot assign any CPUs to the UKF, the programme can terminate. For `Numba`, it takes about a minute to compile the function compared to less than 6 s in an offline procedure. For the LKF-DMD, the algorithm is still capable of running rapidly but is also slowed by the available computer power. Online, LKF-DMD is 1.3 times faster, and offline it is 3.3 times faster. In zones III and IV, we can see that both online and offline computational times are equal. This is because, after 75 s, we stop recording. Once the EVB reconstruction codes have no more data, all CPUs are allocated to the UKF and LKF-DMD algorithms.

As stated in Semeraro et al. (2021), the DT framework should be scalable, interoperable, accurate, dynamic, and modular. However, in order to achieve all five characteristics and scale this framework to industrial systems such as hydro turbines, the availability of computer power is also important and should be sufficient to run a realistic simulation (Ponomarev et al., 2024; Niederer et al., 2021). For larger and more complex systems, such as hydro turbines, this computational power burden can increase rapidly and act as a bottleneck on the ubiquity of DT in industry.

7.3 Scalability of the DT Framework to Larger Systems

To scale this framework from a simple FSI system to hydro turbines, we characterise this framework according to the maturity model of Uhlenkamp et al. (2022). Since there are no standards to assess the maturity and viability of a DT (Niederer et al., 2021), the choice of this model is based on its simplicity. The maturity model quantifies the DT given six characteristics: A) computing capabilities; B) models used; C) the data; D) the control capacities; E) human-machine interaction (HMI); F) DT integration. According to these six characteristics, the latter are divided in sub-characteristics and a relative score is assigned to them. These scores are then normalised and averaged to evaluate the maturity of the framework. The maturity model of Uhlenkamp et al. (2022) employs a weighting formula relative to user perception that in our case would not be generalisable to other mechanical systems such as hydro turbines. This weighting approach is criticised, as it does not allow for a good comparison with other DT frameworks (Boyes and Watson, 2022). Therefore, we do not consider weights in our analysis, and the results of this quantification are presented in Figure 7.4.

In Figure 7.4, the characteristics have respectively a maturity level of 0.33 for the computing capabilities, 0.125 for the model, 0.3 for the data, 0.2 for the control and 0.29 for both the HMI and the integration. On average, the maturity level of the DT framework is 0.26 which demonstrates that the proposed proof-of-concept is still at an early age. Uhlenkamp et al. (2022) Tables used to calculate the values given the subcharacteristics are in Appendix E.

Computing capabilities A): To start the DT and perform system identification and anomaly detection, the user must manually trigger the framework. The data is then periodically updated so that the framework can run. Moreover, since both UKF and LKF algorithms predict the next time step only, the look-ahead perspective of the framework is limited to the short term. Finally, the run-time of the model is on the order of minutes. Hence, the computing efficiency of the framework is limited by the available computing power. One part in which the model excels is the frequency of data update. The high frame rate of the EVB sensors ensures that new data is always available in real-time for the DT. In hydro turbines, aiming for real-time monitoring and predictive maintenance with a DT framework requires enough computing resources, which are energy-intensive and costly.

Model B): A good level for a DT model implies high accuracy, adaptability to change, and modularity. The DT creation approach is based on an existing physical asset, which means that we know how to describe its dynamics and its potential failure events. However, in many systems, this knowledge is not available. Hence, a proper DT creation approach should be

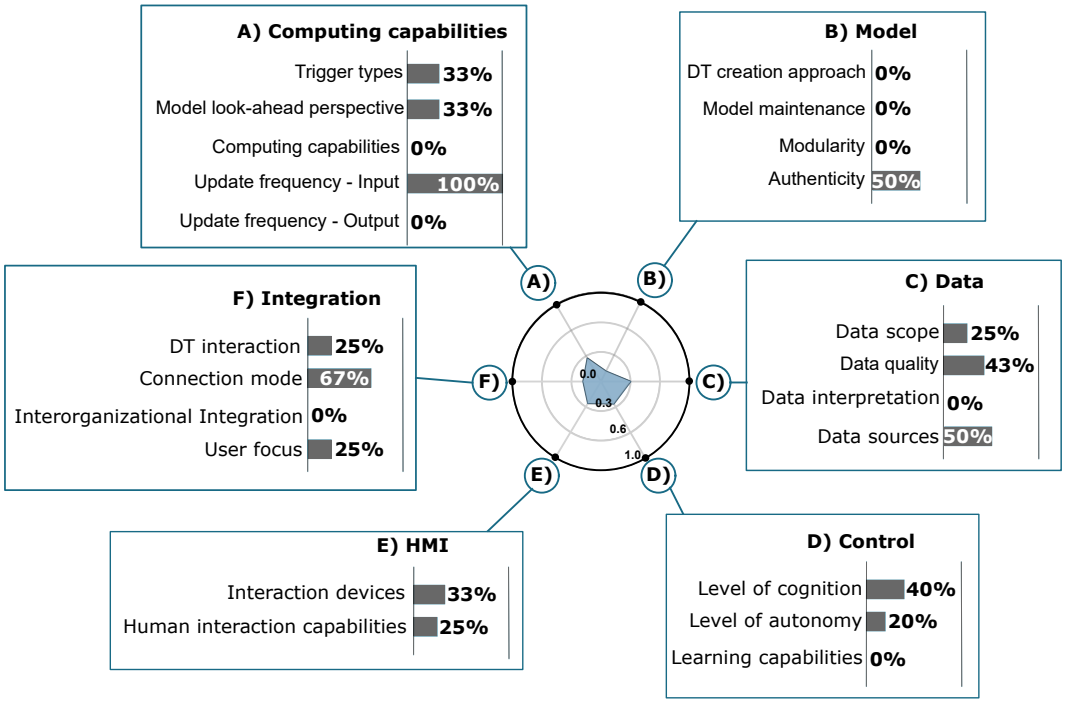

Figure 7.4 **Maturity level of the proposed DT framework of this project base on the quantification model of Uhlenkamp et al. (2022).**

generalisable and self-adaptive so that the DT framework can evolve its model throughout the life of the mechanical system. Accuracy is the fidelity of the model in describing the physical asset and interpreting the response of the virtual counterpart. Adaptability is the maintenance of the model and its capacity to evolute by itself based on the dynamics encounter. Modularity is the ability to apply the DT framework to subsequent systems. In this work, model maintenance has to be done manually, while modularity is affected by the fact that it was only tested on our pipe system. With the modular approach employed in this project, we can imagine that, by modifying the mechanical system and the prior knowledge associated, we could transfer the whole DT framework to other cases. Finally, the authenticity of the model is limited by the assumptions in the nonlinear model of the pipe conveying fluid. As presented in Section 5, some nonlinearities and imperfections in the pipe setup make the physical equations (2.27) and (2.28) of the pipe less accurate. Hence, in hydro turbines, the accuracy of the prior knowledge is of importance so that the predictions coming from the DT are reliable.

Data C): We store the data in the RAM of this computer, which can limit its memory. Second, the scope is the capacity of the DT to extract data from multiple sensors and physical assets. Here, the data sources are from the pipe alone, meaning that the scope is

local. From the results in Section 4, we assume that the data reconstruction is complete and unique by displaying the pipe dynamics without duplicating the data information. However, the data framework still lacks semantic and structural consistency and validation. From raw and uncorrelated EVB data, the 3D reconstruction code aims to structure the events. This implies that the framework could not interpret the events otherwise. Moreover, the accuracy of the 3D reconstruction is questionable with a maximum error of 6%. This error added to the model bias implies an increase in epistemic uncertainty in the framework. For larger and more complex systems, this epistemic uncertainty can be difficult to quantify and take into account.

Control D): The framework has a level of cognition of 40% as we can describe and pseudo-diagnose the anomalies encountered by the pipe. However, this level would obtain a higher score if the model was able to predict and prescribe the upcoming failure. The Digital Twin framework, as shown in Figure 1.1, requires human expertise to interpret its results and act on the physical asset. In theory, a fully autonomous DT should work without the user in the loop. However, in practice, this perspective on machine autonomy is still an ongoing debate. Thus, the development of truly autonomous DTs will depend on their ability to integrate various data types with different objectives and present multiple solutions, leaving the final decision to the user.

HMI E): The computer used in this project limits the framework to a unimodal interaction with the pipe. For Hydro-Québec, where multiple hydro-turbine assets exist, an interoperable approach would imply a shared interaction between the assets, the algorithms, the data types and the users. Also, in this work, the framework answers predefined requests through logical connectors but should be able to answer any problems given the user needs. This falls into a long-term perspective of an optimal DT framework that could encapsulate all models, various data, and reliable decision making (Kapteyn and Willcox, 2021).

Integration F): DT integration is limited to a standalone interaction that concentrates only on the pipe system. The connection mode of this framework is one of a Digital shadow, since exchange between the pipe and its virtual counterpart is unidirectional. As mentioned precedently, removing the user from the model would imply to change this one-way connection to a bidirectional one. For hydro turbines, DT integration should be done so that multiple users of the organisation can fully collaborate on the machine simultaneously.

Scaling DT to industrial uses is often associated with overcoming technical barriers such as computing power, a strong mathematical foundation, or accessible software (Niederer et al., 2021). DT is a multidisciplinary tool with various interconnected models and algorithms. Hence, to understand and quantify the DT framework, many subtleties have to be taken

into account. As the framework matures, the subtleties can increase and become complex, making the analysis of the framework tedious. Figure 7.4 demonstrates the need for multiple criteria to evaluate the framework. However, these criteria and their associated values are not enough to properly verify, validate, and quantify the uncertainty in the DT framework, marking the need for standards in the matter. In fact, here the values associated with the criteria in the maturity model of Uhlenkamp et al. (2022) are not significant themselves, but serve to understand how this DT framework could be improved. Moreover, the maturity model allows us to take a step back on this work and evaluate how qualitatively we perceive this framework. Lastly, this analysis opens up the discussion for the conclusion chapter, the limits of this framework, and the future work needed to realise the long-term goal of a DT for hydro turbines.

CHAPTER 8 CONCLUSION

8.1 Summary of Works

In this work, we develop a digital twin of a pipe conveying fluid system that not only estimates the flow rate but also detects and quantifies changes in flow rate in a near-real-time setting. The proposed framework combines an EVB-based data acquisition module for real-time measurements, an UKF for parameter estimation, and an LKF-DMD model to achieve reduced-order modelling and minimise computational load. The physical system comprises a custom-built silicone rubber pipe equipped with two EVB sensors, allowing 3D reconstruction of its nonlinear oscillatory dynamics.

With these correlated data sources and the known physical equations that describe the fluttering of the pipe, we perform the unknown flow rate estimation with an UKF. Estimates were compared and validated with real measurements of a flow metre installed with the pipe system. Alongside this system identification process, we developed and validated a low-order model to accelerate monitoring and perform anomaly detection. We applied a DMD on the UKF estimates to extract the modal phase diagrams of the oscillating pipe and reduce the physical dimensionality of the system. Then, we combined this dimensionality reduction code with an LKF to monitor the incoming data and track the perturbations. By measuring the Euclidean distance between the known modal phase diagrams and the estimated one with the UKF, we were able to detect sensor perturbation. We were also able to detect a change in flow rate in pipe amplitude by applying an entropy-based algorithm to the Euclidean results.

Finally, we merged all the algorithm models inside a DT framework that updates itself periodically. We demonstrated that the framework performs both system identification, anomaly detection, and quantification. Lastly, we characterised the DT framework throughout a maturity model to scale the proposed proof-of-concept to larger systems, such as hydro turbines.

8.2 Limitations

8.2.1 Physical Model and Numerical Scheme

In this work, we did not perform a rigourous numerical stability analysis on the Crank-Nicolson time-integration scheme inside the UKF. Existing tools such as the Courant-Friedrichs-Lewy condition method or the Von Neumann analysis could help us define and quantify the appropriate Δt_{CN} and Δh to obtain an accurate convergence of the numerical scheme during

its resolution of Equation (2.37). Moreover, a comparison of the CN results with a more robust solver such as the Runge-Kutta methods would have been needed to verify that the CN discretiser captures all nonlinear dynamics given the choice of Δt_{CN} and Δh.

For the pipe model, the validation procedure performed in Chapter 5 demonstrates discrepancies between the model amplitude reconstruction and the observation. This difference between theory and practice can come from the limiting assumptions in the physics-based model such as the relatively small amplitude, the Euler-Bernoulli cantilever beam mode shapes, or rigid fixation at the clamping point. The assumption of a small-amplitude oscillation can be overcome with the geometrically exact model developed in Chen et al. (2021) or in Farokhi et al. (2021). Moreover, as mentioned in Chapter 5, the presence of imperfection in pipe manufacturing or pipe clamp could cause the addition of inaccuracies not taken into account in the model. Imperfect pipe with unwanted curvature or with air-bubbles in their material are already studied subjects (Chen, 2022) such as pipe with elastic foundation and imperfect clamping (Kheiri, 2020). Since these imperfections have some mathematical formulation, the geometrical exact model could be improved by considering the contribution of these factors. In the case of the presence of air bubbles, it implies the reduction of the properties of the material at the given location and the variability of the mass of the pipe along its length. ElNajjar and Daneshmand (2020) have developed a nonlinear pipe model with the presence of multiple point mass and springs that could be of good use to simulate the presence of air bubbles in the pipe silicon. A model that accurately describes the actual observed dynamics would lead to better estimates via the UKF and better monitoring through the LKF-DMD. Any imperfection that can be parametrised could potentially be estimated with the UKF.

Lastly, in the pipe model, the dimensionless parameters γ, τ, and u are defined based on the Young modulus E. Although α is defined both from E and from the Kelvin-Voigt damping parameter E^*. E and E^* are obtained by a free vibration method. However, it is well known that this method is not as accurate compared to shaker characterisation as mentioned by Paidoussis (2014). The values obtained for E and E^* affect largely the values of dimensionless parameters and, consequently, the observed physics. Hence, a more accurate characterisation method could be of good use to reduce the discrepancy between theory and experimentation.

8.2.2 Data Acquisition System

The biggest limits of the proposed EVB 3D reconstruction procedure is the lack of any experimental and standardised validation. As in Raynaud and Mulleners (2024), it could be interesting to compare the point of the reconstructed pipe with the results of a distance

sensor. Also, reconstructing the whole pipe with a polynomial can show a large discrepancy with the real oscillation at the boundary conditions. In chapter 4, the Runge phenomenon limits the quality of the polynomial fitting at the clamp and the tip position. Once the amplitude increases, the phenomenon is exacerbated. Using a spline could limit the impact of the Runge phenomena, but a subsequent test should be made to verify the impact on the computational time of the procedure. In addition, improving the constraints around ROI could increase the quality of the fit by further guiding it. Techniques such as clustering and adaptive mask would be of interest in sorting the EVB point's cloud and increasing the accuracy of the fitting. With such improvements, the phenomenon of large-amplitude oscillation could be well reconstructed and studied.

Furthermore, the computational time of the proposed method is first limited by the structuring of the event cloud, so that polynomial fitting can work. The event data can be sparse and corrupted by the presence of unwanted dynamics such as water splashes or ghost events. Hence, structuring such nonuniform data can bias the fitting by aggregating unwanted events in the structured cloud. Second, to have a 3D reconfiguration, an additional parallax step is needed, and both outputs of the sensors have to be merged inside one framework. This step implies a slowdown in the data acquisition scheme and therefore prevents the EVB system from continuously sending their data in real-time to the UKF and the LKF-DMD.

8.2.3 Virtual Model Algorithms

First, the proposed UKF algorithm depends on both the data quality and the accuracy of the physic-based model. Moreover, the contribution of the nonlinearities can slow the computational time and add complexity to the algorithm estimate resolution. Such complexities added with unknown state and parameters in the system will often make the UKF converge to minima far from the actual solution. With the right hyperparameters, initial conditions and time rate, one can bypass this limit. However, it then becomes an optimisation problem that cannot be generalised to other dynamical cases. Moreover, KFTs in general lack intelligence and are restricted to short-term look-ahead prediction. Hence, in a DT framework with the perspective of predictive maintenance, KFTs alone are not necessarily sufficient for this kind of problem and should always be complement with various models and techniques chosen given the user objectives.

Second, anomaly detection with the proposed algorithms is a naive solution. Hence, small and smooth anomalies are often missed by the system or detected too late. In the case of the Euclidean distance with SAHC, the perturbation is detected only if the difference between the LKF estimates and the DMD modal phase diagrams is notable. However, this difference is

based on a modal phase diagram that does not match the sensor measurement reconstruction. Hence, the 3σ rule is bias and unrealistic, which can affect the algorithm capacity to detect smaller perturbation. Moreover, the reference modal phase diagrams are built on the UKF estimates to add physical interpretation to the low-order model. However, the difference between the model and the theory undermines this idea since there are discrepancies that make the interpretation unreliable. Such discrepancies between the estimations and the trained modal phase diagrams could also come from the DMD training itself. In this work, we train the DMD exclusively on displacement states instead of displacement and velocity states. Since model order reduction is based on second-order dynamics, the velocity is needed to capture the full system dynamics.

Finally, the general framework is limited by the computing capabilities of the computer used in this project. Since the whole framework is written in Python, the computer has to share the CPU power among the different codes. Moreover, the present study did not address code optimisation. Naive implementation and redundancy can be the main bottleneck in the available limiting computing power.

8.3 Future Research

This project has a wider scope to develop the framework and codes for a DT for hydro turbines. From the limitations mentioned previously and the categories in the maturity model of Uhlenkamp et al. (2022), we can guide future research in: (i) code profiling and optimisation; (ii) surrogate and hybrid models; (iii) implementation of intelligence and learning capability; (iv) verification, validation, and uncertainty quantification.

In (i), the computing capability can be improved by profiling the code to detect bottlenecks in the implementation. The latter could then be optimised to run faster. Using compilers such as Chapel could also be an interesting avenue, so GPUs can be used instead of CPUs. Moreover, with an optimised framework and powerful capabilities, future research could look at continuous data update and simulation instead of the batch method used in this project. This computing acceleration could improve the anomaly detection framework by identifying the perturbation right at the time it occurs. Lastly, hydro turbines are monitored with multiple source sensors for a long useful life. Hence, the amount of actual and historical data to deal with will be considerable. Challenges with the storage, fusing, interpretation, and speed of processing such an amount of data are to be expected.

(ii) Computer power and code optimisation are not sufficient for real-time monitoring and efficient 3D data reconstruction. In the 3D reconstruction, the noncorrelatability in the data

and the complexity in the pipe-curve oscillations are hard to fit properly with a polynomial. Moreover, the clamping of the pipe and the material can change the observed dynamics. Hence, surrogate models have been shown to be promising tools to reduce the computational burden of these models by using an approximated form from the data instead (Saves et al., 2024). Building such models in a DT framework could be of good use since surrogate models can also be used for sensitivity analysis and uncertainty quantification. This data-driven technique could also be combined with physical knowledge to obtain hybrid models. In Chapter 2, we presented the perspective of hybrid modelling as a trade-off tool merging the advantages of physics-based and data-driven modelling. In our case, with the uncertainty in pipe manufacturing, mechanical properties, and clamping quality, their impacts can often be misinterpreted or missed in the physics-based model. We can improve the mathematical formulation by adding more nonlinear contributions, but it would necessarily be computationally expensive. Demenois (2022b) worked on PINN for the inverse problem. This hybrid model compared to UKF is more generalisable and has a better forecasting capability due to its learning from its training. However, training is time-consuming and depends on physical prior knowledge. Future research to develop efficient hybrid models that take into account surrogate modelling and accurate physical interpretation is an interesting scope to explore and an open problem. Lastly, in the DMD method used in this project, the limit in the training on the displacement state only should be overcome by including both displacement and velocity states. Thus, the LKF-DMD method could gain accuracy by capturing the full system dynamics of the second-order differential equations.

(iii) Learning capabilities and intelligence in a DT framework is needed so it can adapt throughout the whole life cycle of the machine and recognise various situations. Moreover, such capabilities can help to produce a modular framework that works on different assets in the system. In our laboratories, pipe conveying fluid is used to coarsely simulate the FSI interaction in hydro turbines. To evaluate turbine axis rotation, a vertical axis rotary machine was developed by Serroud (2023) and parameter identification was performed to quantify the unknown bearing stiffness by Rishmawi et al. (2025). A proper DT framework could be generalised to these two systems and even work simultaneously for both through fused interaction of the system and the data. In hydro turbines this kind of interaction between multiple assets is to be expected. Furthermore, in this work, the anomalies are predefined, hence the decision. In a predictive maintenance perspective, such rigid anomaly detection and cognition limit the framework to known irregularities. With an increase in the intensity of weather-related events and human sabotage threat (Larouche, 2025), observed anomalies are subject to change. Thus, a malleable DT framework that combines multiple algorithms for multiple purposes is an avenue to explore. However, adding such capabilities

to the DT implies enough data, knowledge, and training of the algorithm in different cases so that the DT can be modular and scalable. This can be time-consuming and may require more power capabilities. In addition, choosing when and how to perform these trainings can be tedious, which marks the need for further research in this area.

(iv) Finally, a DT framework is the interoperability of multiple algorithms and potentially multiple sensors and physical assets. This huge collaborative framework is subject to multiple causes of uncertainty that have to be quantified to properly verify and validate the framework. Some verification, validation and uncertainty quantification frameworks exist, such as the VV20 standard ASME (2009). Moreover, the International Standard Organisation (ISO) is currently building a complete maturity model for DT to help quantify and generalise the DT framework (ISO, 2025). The application of standards on the DTs will not only help in improving their quality as maintenance tool, but will also ease their integration and utilisation in an organisation.

REFERENCES

Akhlaghi, S., Zhou, N., and Huang, Z. (2017). Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation. In *2017 IEEE Power & Energy Society General Meeting*, pages 1–5. ISSN: 1944-9933.

ASME (2009). VV20-Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer - ASME.

Audet, C., Le Digabel, S., Rochon Montplaisir, V., and Tribes, C. (2022). Algorithm 1027: NOMAD version 4: Nonlinear optimization with the MADS algorithm. *ACM Transactions on Mathematical Software*, 48(3):35:1–35:22.

Ayani, M., Ganebäck, M., and Ng, A. H. C. (2018). Digital Twin: Applying emulation for machine reconditioning. *Procedia CIRP*, 72:243–248.

Bajaj, A. K. and Sethna, P. R. (1984). Flow Induced Bifurcations to Three-Dimensional Oscillatory Motions in Continuous Tubes. *SIAM Journal on Applied Mathematics*, 44(2):270–286. Publisher: Society for Industrial and Applied Mathematics.

Beaulieu, C. (2019). *J'aime Hydro*, volume 13. Atelier 10, Montréal, Québec, Canada, atelier 10 edition.

Benjamin, H. H. (1961). Integration—Stimulus for Private Education. *The Educational Forum*, 25(4):457–461. Publisher: Routledge _eprint: https://doi.org/10.1080/00131726109338467.

Bergeron, U. (2023). 2,4 milliards pour changer les turbines de quatre centrales.

Boukani, H. H., Viens, M., Tahan, S. A., and Gagnon, M. (2014). On the performance of nondestructive testing methods in the hydroelectric turbine industry. *IOP Conference Series: Earth and Environmental Science*, 22(1):012018.

Boyes, H. and Watson, T. (2022). Digital twins: An analysis framework and open issues. *Computers in Industry*, 143:103763.

Brunton, S. L. and Kutz, J. N. (2022). *Data-Driven Science and Engineering Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, 2nd edition edition.

Cai, Z., Wang, Y., Zhang, D., Wen, L., Liu, H., Xiong, Z., Wajid, K., and Feng, R. (2024). Digital Twin Modeling for Hydropower System Based on Radio Frequency Identification Data Collection. *Electronics*, 13(13):2576. Number: 13 Publisher: Multidisciplinary Digital Publishing Institute.

Chen, H. and Ni, Y. (2018). *Structural Health Monitoring of Large Civil Engineering Structures*. Wiley, 1 edition.

Chen, W. (2022). Geometrically exact model and dynamics of cantilevered curved pipe conveying fluid. *Journal of Sound and Vibration*.

Chen, W., Dai, H., and Wang, L. (2021). Three-dimensional dynamical model for cantilevered pipes conveying fluid under large deformation. *Journal of Fluids and Structures*, 105:103329.

Chen, W., Hu, Z., Dai, H., and Wang, L. (2020). Extremely large-amplitude oscillation of soft pipes conveying fluid under gravity. *Applied Mathematics and Mechanics*, 41(9):1381–1400.

Chen, Z. (2003). Bayesian filtering: From Kalman filters to particle filters, and beyond. *Statistics*, 182(1):1–69.

Choi, W., Hudachek, K., Koskey, S., Perullo, C., and Noble, D. (2024). Digital twin in the power generation industry. *JMST Advances*, 6(1):103–119.

Dai, H., He, Y., Zhou, K., Peng, Z., Wang, L., and Hagedorn, P. (2022). Utilization of nonlinear vibrations of soft pipe conveying fluid for driving underwater bio-inspired robot. *Applied Mathematics and Mechanics*, 43(7):1109–1124.

Daneshmand, F., Liaghat, T., and Paidoussis, M. P. (2021). A Coupled Two-Way Fluid–Structure Interaction Analysis for the Dynamics of a Partially Confined Cantilevered Pipe Under Simultaneous Internal and External Axial Flow in Opposite Directions. *Journal of Pressure Vessel Technology*, 144(021401).

Dehrouyeh-Semnani, A. M. (2023). Nonlinear geometrically exact dynamics of fluid-conveying cantilevered hard magnetic soft pipe with uniform and nonuniform magnetizations. *Mechanical Systems and Signal Processing*, 188:110016.

Demenois, M. (2022a). Github python code. `https://github.com/lm2-poly/PCF_Setup_Article/tree/main/Python_codes`.

Demenois, M. (2022b). *The Pipe Conveying Fluid as a Model Experiment to Develop a Digital Twin*. PhD thesis, Polytechnique Montréal, Montréal, Québec, Canada.

Demenois, M., Miao, H. Y., and Gosselin, F. P. (2023). Easy to build, modular and large scale pipe conveying fluid experimental setup. *HardwareX*, 15:e00460.

Dollon, Q., Ghorbani, E., and Gosselin, F. P. (2025). Bayesian input-state-parameter inference of hydrodynamic bearings: From partial displacement measurements to force reconstruction. *Mechanical Systems and Signal Processing*, 230:112610.

Ellis, B., Heyns, P. S., and Schmidt, S. (2022). A hybrid framework for remaining useful life estimation of turbomachine rotor blades. *Mechanical Systems and Signal Processing*, 170:108805.

ElNajjar, J. and Daneshmand, F. (2020). Stability of horizontal and vertical pipes conveying fluid under the effects of additional point masses and springs. *Ocean Engineering*, 206:106943.

Ersan, M. and Irmak, E. (2024). Development and Integration of a Digital Twin Model for a Real Hydroelectric Power Plant. *Sensors*, 24(13):4174. Number: 13 Publisher: Multidisciplinary Digital Publishing Institute.

Farokhi, H., Tavallaeinejad, M., and Païdoussis, M. P. (2021). Geometrically exact dynamics of cantilevered pipes conveying fluid. *Journal of Fluids and Structures*, 106:103364.

Favrel, A., Dollon, Q., and Merleau, J. (2024). Using multiple data-driven models to diagnose abnormal behaviours in hydropower units.

Gagnon, M., Immarigeon, A., Nicolle, J., and Morissette, J.-F. (2019). Correlation between numerical simulations and measurements to assess uncertainties: a case study on a hydroelectric runner. *IOP Conference Series: Earth and Environmental Science*, 240(7):072005. Publisher: IOP Publishing.

Gagnon, M., Thibault, D., and Blain, M. (2020). On the Expected Monetary Value of Hydroelectric Turbine Start-up Protocol Optimisation. In Liyanage, J. P., Amadi-Echendu, J., and Mathew, J., editors, *Engineering Assets and Public Infrastructures in the Age of Digitalization*, Lecture Notes in Mechanical Engineering, pages 209–216, Cham. Springer International Publishing.

Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A. J., Conradt, J., Daniilidis, K., and Scaramuzza, D. (2022). Event-Based Vision:

A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):154–180. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

Ghavanloo, E., Daneshmand, F., and Rafiei, M. (2010). Vibration and instability analysis of carbon nanotubes conveying fluid and resting on a linear viscoelastic Winkler foundation. *Physica E: Low-dimensional Systems and Nanostructures*, 42(9):2218–2224.

Ghorbani, E. (2021). *Nonlinear Kalman Filtering Based Damage Quantification for Civil Infrastructure*. Doctor of Phylosophy, The University of Manitoba.

Ghorbani, E., Buyukozturk, O., and Cha, Y.-J. (2020). Hybrid output-only structural system identification using random decrement and Kalman filter. *Mechanical Systems and Signal Processing*, 144:106977. Publisher: Academic Press.

Ghorbani, E. and Cha, Y.-J. (2018). An iterated cubature unscented kalman filter for large-dof systems identification with noisy data. *Journal of Sound and Vibration*, 420:21–34.

Ghorbani, E., Dollon, Q., and Gosselin, F. P. (2024). Physics-aware tuning of the unscented Kalman filter: statistical framework for solving inverse problems involving nonlinear dynamical systems and missing data. *Nonlinear Dynamics*.

Giroux, Gauthier, and Dollon (2023). Application of the Digital Twin Concept for Operation and Maintenance of Hydro Generating Units. Edinburgh, Scotland.

Gouvernement du Canada, L. R. d. l. d. C. (2024). ONÉ - Profils énergétiques des provinces et territoires – Québec. Last Modified: 2024-09-10.

Grieves, M. (2015). Digital Twin: Manufacturing Excellence through Virtual Factory Replication. *White paper*.

Grieves, M. (2023). Digital Twins: Past, Present, and Future. In *The Digital Twin*, volume 1, pages 97–121. Springer edition.

Guo, Q., Zhou, J. X., and Guan, X. L. (2020). Fluid–structure interaction in Z-shaped pipe with different supports. *Acta Mechanica Sinica*, 36(2):513–523.

Harries, T., Hartnoll, M., Hafezianrazavi, M., Meek, H., and Nassehi, A. (2023). Digital Twins for Predictive Maintenance. *Procedia CIRP*, 118:306–311.

Holmes, P. J. (1977). Bifurcations to divergence and flutter in flow-induced oscillations: A finite dimensional analysis. *Journal of Sound and Vibration*, 53(4):471–503.

Hsu, K., Tan, J.-M., Chen, Y.-S., Hung, C.-C., Puayen Tan, Z., Currao, G., and Jiang, B.-S. (2024). Event-Based Measurement of Aeroelastic Structure in High-Speed Flow. *AIAA Journal*, 0(0):1–12. Publisher: American Institute of Aeronautics and Astronautics _eprint: https://doi.org/10.2514/1.J064591.

Hydro-Québec (2024a). Building the Future - Annual Report 2023. Technical report, Hydro-Québec, Montréal, Québec, Canada.

Hydro-Québec (2024b). Private communication.

Iddrisu, K., Shariff, W., Corcoran, P., O'Connor, N. E., Lemley, J., and Little, S. (2024). Event Camera-Based Eye Motion Analysis: A Survey. *IEEE Access*, 12:136783–136804. Conference Name: IEEE Access.

ISO (2020). ISO 19283:2020(en), Condition monitoring and diagnostics of machines — Hydroelectric generating units.

ISO (2025). Digital twin — Maturity model and guidance for a maturity assessment.

Jaynes, E. (1978). Where do we stand on maximum entropy? page 105, Massachusetts Institute of Technology.

Jerath, K., Brennan, S., and Lagoa, C. (2018). Bridging the gap between sensor noise modeling and sensor characterization. *Measurement*, 116:350–366.

Julier, S. and Uhlmann, J. (2004). Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422. Conference Name: Proceedings of the IEEE.

Julier, S., Uhlmann, J., and Durrant-Whyte, H. (1995). A new approach for filtering nonlinear systems. In *Proceedings of 1995 American Control Conference - ACC'95*, volume 3, pages 1628–1632 vol.3.

Jung, J.-H. and Stefan, W. (2011). A Simple Regularization of the Polynomial Interpolation for the Resolution of the Runge Phenomenon. *Journal of Scientific Computing*, 46(2):225–242.

Kalman, R. (1960). A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME Journal of Basic Engineering*, 82(Series D):35–45.

Kapteyn, M. G. and Willcox, K. E. (2021). Digital Twins: Where Data, Mathematics, Models, and Decisions Collide | SIAM. *Society for Industrial and Applied Mathematics*, 54(07).

Kharami, E., Mentzelopoulos, A., Prele, E., Shukla, K., Wang, Z., Bray, B., Cao, Q., Singh, A., Fan, D., Sapsis, T., Triantafyllou, M. S., and Karniadakis, G. E. (2024). FIRSTLING-DIGIMAR, a Pilot Scale Digital Twin of a Marine Riser for Field Use. In *Day 4 Thu, May 09, 2024*, page D041S052R001, Houston, Texas, USA. OTC.

Kheiri, M. (2020). Nonlinear dynamics of imperfectly-supported pipes conveying fluid. *Journal of Fluids and Structures*, 93:102850.

Khodarahmi, M. and Maihami, V. (2023). A Review on Kalman Filter Models. *Archives of Computational Methods in Engineering*, 30(1):727–747.

Kogler, J., Humenberger, M., and Sulzbachner, C. (2011). Event-Based Stereo Matching Approaches for Frameless Address Event Stereo Data. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Wang, S., Kyungnam, K., Benes, B., Moreland, K., Borst, C., DiVerdi, S., Yi-Jen, C., and Ming, J., editors, *Advances in Visual Computing*, pages 674–685, Berlin, Heidelberg. Springer.

Lagorce, X., Orchard, G., Galluppi, F., Shi, B. E., and Benosman, R. B. (2017). HOTS: A Hierarchy of Event-Based Time-Surfaces for Pattern Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(7):1346–1359. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

Lai, Z., Alzugaray, I., Chli, M., and Chatzi, E. (2020). Full-field structural monitoring using event cameras and physics-informed sparse identification. *Mechanical Systems and Signal Processing*, 145:106905.

Lamothe, T., Blain, M., Gagnon, M., Lonchampt, J., and Tahan, A. (2020). Technico-Economic Modelling of Maintenance Cost for Hydroelectric Turbine Runners. In Liyanage, J. P., Amadi-Echendu, J., and Mathew, J., editors, *Engineering Assets and Public Infrastructures in the Age of Digitalization*, Lecture Notes in Mechanical Engineering, pages 238–245, Cham. Springer International Publishing.

Lan, H., Zhao, S., Hu, J., Wang, Z., and Fu, J. (2024). Joint State Estimation and Noise Identification Based on Variational Optimization. *IEEE Transactions on Automatic Control*, pages 1–16.

Laperle, T. (2025). Private communication.

Larouche, V. (2025). Postes électriques d'Hydro-Québec: Attention : cibles de choix. *La Presse*.

Lee, J., Delbruck, T., Park, P. K. J., Pfeiffer, M., Shin, C.-W., Ryu, H., and Kang, B. C. (2012). Live demonstration: Gesture-based remote control using stereo pair of dynamic vision sensors. In *2012 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 741–745. ISSN: 2158-1525.

Lee, Y., Jin, C., Kim, M., and Xu, W. (2024). Digital twin approach with minimal sensors for Riser's fatigue-damage estimation. *International Journal of Naval Architecture and Ocean Engineering*, 16:100603.

Lei, Y., Li, N., Guo, L., Li, N., Yan, T., and Lin, J. (2018). Machinery health prognostics: A systematic review from data acquisition to RUL prediction. *Mechanical Systems and Signal Processing*, 104:799–834.

Li, M., Yan, H., and Wang, L. (2024a). Data-driven model reduction for pipes conveying fluid via spectral submanifolds. *International Journal of Mechanical Sciences*, 277:109414.

Li, R., Arzaghi, E., Abbassi, R., Chen, D., Li, C., Li, H., and Xu, B. (2020). Dynamic maintenance planning of a hydro-turbine in operational life cycle. *Reliability Engineering & System Safety*, 204:107129.

Li, X., Yu, S., Lei, Y., Li, N., and Yang, B. (2024b). Intelligent Machinery Fault Diagnosis With Event-Based Camera. *IEEE Transactions on Industrial Informatics*, 20(1):380–389. Conference Name: IEEE Transactions on Industrial Informatics.

Liao, L. and Köttig, F. (2014). Review of Hybrid Prognostics Approaches for Remaining Useful Life Prediction of Engineered Systems, and an Application to Battery Life Prediction. *IEEE Transactions on Reliability*, 63(1):191–207. Conference Name: IEEE Transactions on Reliability.

Litzenberger, M., Kohn, B., Belbachir, A., Donath, N., Gritsch, G., Garn, H., Posch, C., and Schraml, S. (2006). Estimation of Vehicle Speed Based on Asynchronous Data from a Silicon Retina Optical Sensor. In *2006 IEEE Intelligent Transportation Systems Conference*, pages 653–658. ISSN: 2153-0017.

Liu, Z.-g., Wang, Z.-k., Yang, Y.-b., and Lu, Y. (2024). A Data-Driven Maneuvering Target Tracking Method Aided With Partial Models. *IEEE Transactions on Vehicular Technology*, 73(1):414–425.

Lundgren, T. S., Sethna, P. R., and Bajaj, A. K. (1979). Stability boundaries for flow induced motions of tubes with an inclined terminal nozzle. *Journal of Sound and Vibration*, 64(4):553–571.

Lyu, Z., Cai, W., and Liu, Y. (2024). High-frequency visualization of flexible structures using an event-triggered camera: multiple flapping membranes. *Measurement Science and Technology*, 35(5):055302.

Menegaz, H. M. T., Ishihara, J. Y., Borges, G. A., and Vargas, A. N. (2015). A Systematization of the Unscented Kalman Filter Theory. *IEEE Transactions on Automatic Control*, 60(10):2583–2598. Conference Name: IEEE Transactions on Automatic Control.

Modarres-Sadeghi, Y. (2021). *Introduction to Fluid-Structure Interactions*. Springer International Publishing, Cham.

Modarres-Sadeghi, Y., Paı¨doussis, M. P., and Semler, C. (2008). Three-dimensional oscillations of a cantilever pipe conveying fluid. *International Journal of Non-Linear Mechanics*, 43(1):18–25.

Modarres-Sadeghi, Y., Semler, C., Wadham-Gagnon, M., and Païdoussis, M. (2007). Dynamics of cantilevered pipes conveying fluid. Part 3: Three-dimensional dynamics in the presence of an end-mass. *Journal of Fluids and Structures*, 23(4):589–603.

Montero Jimenez, J. J., Schwartz, S., Vingerhoeds, R., Grabot, B., and Salaün, M. (2020). Towards multi-model approaches to predictive maintenance: A systematic literature survey on diagnostics and prognostics. *Journal of Manufacturing Systems*, 56:539–557.

Naya, M. A., Sanjurjo, E., Rodriguez, A. J., and Cuadrado, J. (2023). Kalman filters based on multibody models: linking simulation and real world. a comprehensive review. *Multibody System Dynamics*.

Niederer, S. A., Sacks, M. S., Girolami, M., and Willcox, K. (2021). Scaling digital twins from the artisanal to the industrial. *Nature Computational Science*, 1(5):313–320.

Nonomura, T., Shibata, H., and Takaki, R. (2018). Dynamic mode decomposition using a Kalman filter for parameter estimation. *AIP Advances*, 8(10):105106.

Nonomura, T., Shibata, H., and Takaki, R. (2019). Extended-Kalman-filter-based dynamic mode decomposition for simultaneous system identification and denoising. *PLOS ONE*, 14(2):e0209836. Publisher: Public Library of Science.

Oyekole, O., Trenchea, C., and Bukač, M. (2018). A Second-Order in Time Approximation of Fluid-Structure Interaction Problem. *SIAM Journal on Numerical Analysis*, 56(1):590–613. Publisher: Society for Industrial and Applied Mathematics.

Paidoussis, M. P. (1976). Hydroelastic Ichthyoid Propulsion. *Journal of Hydronautics*, 10(1):30–32. Publisher: American Institute of Aeronautics and Astronautics.

Paidoussis, M. P. (2014). 5. Pipes Conveying Fluid: Nonlinear and Chaotic Dynamics - Knovel. In *Fluid-structure Interactions. Volume 1, Slender structures and axial flow*, volume 1. Academic Press, 2nd edition edition.

Park, D. and You, H. (2023). A Digital Twin Dam and Watershed Management Platform. *Water*, 15(11):2106. Number: 11 Publisher: Multidisciplinary Digital Publishing Institute.

Païdoussis, M., Semler, C., Wadham-Gagnon, M., and Saaid, S. (2007). Dynamics of cantilevered pipes conveying fluid. Part 2: Dynamics of the system with intermediate spring support. *Journal of Fluids and Structures*, 23(4):569–587.

Païdoussis, M. P. (2022). Pipes conveying fluid: A fertile dynamics problem. *Journal of Fluids and Structures*, 114:103664.

Païdoussis, M. P. and Semler, C. (1998). Non-linear dynamics of a fluid-conveying cantilevered pipe with a small mass attached at the free end. *International Journal of Non-Linear Mechanics*, 33(1):15–32.

Ponomarev, K., Kudryashov, N., Popelnukha, N., and Potekhin, V. (2024). (PDF) Main Principals and Issues of Digital Twin Development for Complex Technological Processes. In *Proceedings of the 28th DAAAM International Symposium*, pages 523–528, Vienna, Austria. DAAAM International.

PROPHESEE (2024). PROPHESEE | Metavision for Machines.

Raynaud, G. and Mulleners, K. (2024). Event-based reconstruction of time-resolved centreline deformation of flapping flags. arXiv:2409.08939 [physics].

Rebecq, H., Gallego, G., Mueggler, E., and Scaramuzza, D. (2018). EMVS: Event-Based Multi-View Stereo—3D Reconstruction with an Event Camera in Real-Time. *International Journal of Computer Vision*, 126(12):1394–1414.

Rishmawi, S., Moyne, L., Serroud, S., Rodriguez, S., Chinesta, F., Tuysuz, O., and Gosselin, F. P. (2025). Parameter identification of a nonlinear vertical axis rotating machine through reduced order modeling and data assimilation. *Nonlinear Dynamics*.

Rosafalco, L., Conti, P., Manzoni, A., Mariani, S., and Frangi, A. (2024). EKF-SINDy: Empowering the extended Kalman filter with sparse identification of nonlinear dynamics. arXiv:2404.07536 [cs, eess, math].

Rosafalco, L., Conti, P., Manzoni, A., Mariani, S., and Frangi, A. (2025). Online learning in bifurcating dynamic systems via SINDy and Kalman filtering. *Nonlinear Dynamics*.

Russell, S. J., Norvig, P., and Davis, E. (2010). *Artificial intelligence: a modern approach*. Prentice Hall series in artificial intelligence. Prentice Hall, Upper Saddle River, 3rd ed edition.

Saidin, S. S., Kudus, S. A., Jamadin, A., Anuar, M. A., Amin, N. M., Ya, A. B. Z., and Sugiura, K. (2023). Vibration-based approach for structural health monitoring of ultra-high-performance concrete bridge. *Case Studies in Construction Materials*, 18:e01752.

Sarkar, A. and Paidoussis, M. P. (2002). A Cantilever Conveying Fluid: Coherent Modes Versus Beam Modes.

Saves, P., Lafage, R., Bartoli, N., Diouane, Y., Bussemaker, J., Lefebvre, T., Hwang, J. T., Morlier, J., and Martins, J. R. R. A. (2024). SMT 2.0: A Surrogate Modeling Toolbox with a focus on hierarchical and mixed variables Gaussian processes. *Advances in Engineering Software*, 188:103571.

Semeraro, C., Lezoche, M., Panetto, H., and Dassisti, M. (2021). Digital twin paradigm: A systematic literature review. *Computers in Industry*, 130:103469.

Semler, C. (1996). *Pipe Conveying Fluid : a Paradigm of Nonlinear Dynamics*. Thesis, Mcgill University, Montréal, Québec, Canada. Publisher: McGill University.

Serroud, S. (2023). *Design of a Vertical Axis Rotating Machine for the Development of a Digital Twin*. PhD thesis, Polytechnique Montréal, Montréal, Québec, Canada.

Stark, R., Kind, S., and Neumeyer, S. (2017). Innovations in digital modelling for next generation manufacturing system design. *CIRP Annals*, 66(1):169–172.

Särkkä, S. (2011). Bayesian Estimation of Time-Varying Systems: Discrete-Time Systems. page 115.

Tan, C., Cai, Y., Wang, H., Sun, X., and Chen, L. (2023). Vehicle State Estimation Combining Physics-Informed Neural Network and Unscented Kalman Filtering on Manifolds. *Sensors*, 23(15):6665.

Tang, Y., Zhang, H.-J., Chen, L.-Q., Ding, Q., Gao, Q., and Yang, T. (2024). Recent progress on dynamics and control of pipes conveying fluid. *Nonlinear Dynamics*.

Tao, F., Liu, A., Qi, Q., and Zhang, M. (2018). (PDF) Digital twin-driven product design framework. *International Journal of Production Research*.

Tao, F., Qi, Q., Wang, L., and Nee, A. Y. C. (2019). Digital Twins and Cyber–Physical Systems toward Smart Manufacturing and Industry 4.0: Correlation and Comparison. *Engineering*, 5(4):653–661.

Tao, F. and Zhang, M. (2017). Digital Twin Shop-Floor: A New Shop-Floor Paradigm Towards Smart Manufacturing. *IEEE Access*, 5:20418–20427. Conference Name: IEEE Access.

Tidriri, K., Chatti, N., Verron, S., and Tiplica, T. (2016). Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: A review of researches and future challenges. *Annual Reviews in Control*, 42:63–81.

Trivedi, C. and Cervantes, M. J. (2017). Fluid-structure interactions in Francis turbines: A perspective review. *Renewable and Sustainable Energy Reviews*, 68:87–101.

Uhlenkamp, J.-F., Hauge, J. B., Broda, E., Lutjen, M., Freitag, M., and Thoben, K.-D. (2022). Digital Twins: A Maturity Model for Their Classification and Evaluation. *IEEE Access*, 10:69605–69635.

Vettori, S., Di Lorenzo, E., Peeters, B., Luczak, M. M., and Chatzi, E. (2023). An adaptive-noise Augmented Kalman Filter approach for input-state estimation in structural dynamics. *Mechanical Systems and Signal Processing*, 184:109654.

Wadham-Gagnon, M., Paı¨doussis, M., and Semler, C. (2007). Dynamics of cantilevered pipes conveying fluid. Part 1: Nonlinear equations of three-dimensional motion. *Journal of Fluids and Structures*, 23(4):545–567.

Wan, E. and Van Der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pages 153–158, Lake Louise, Alta., Canada. IEEE.

Wan, E., van der Merwe, R., and Nelson, A. (1999). Dual Estimation and the Unscented Transformation. In *Advances in Neural Information Processing Systems*, volume 12. MIT Press.

Wan, E. A. and van der Merwe, R. (2001). The Unscented Kalman Filter. In *Kalman Filtering and Neural Networks*, pages 221–280. John Wiley & Sons, Ltd. Section: 7 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471221546.ch7.

Wang, H., Ou, S. S., Dahlhaug, O. G., Storli, P.-T., Skjelbred, H. I., and Vilberg, I. (2023). Adaptively Learned Modeling for a Digital Twin of Hydropower Turbines with Application to a Pilot Testing System. *Mathematics*, 11(18):4012. Number: 18 Publisher: Multidisciplinary Digital Publishing Institute.

Wang, J., Moreira, J., Cao, Y., and Gopaluni, B. (2022). Time-Variant Digital Twin Modeling through the Kalman-Generalized Sparse Identification of Nonlinear Dynamics. In *2022 American Control Conference (ACC)*, pages 5217–5222. ISSN: 2378-5861.

Wick, T. (2013). Stability Estimates and Numerical Comparison of Second Order Time-Stepping Schemes for Fluid-Structure Interactions. In Cangiani, A., Davidchack, R. L., Georgoulis, E., Gorban, A. N., Levesley, J., and Tretyakov, M. V., editors, *Numerical Mathematics and Advanced Applications 2011*, pages 625–632, Berlin, Heidelberg. Springer.

Yang, Y., Zhao, L., Yu, Q., Liu, S., Zhou, G., and Shen, W. (2023). State of charge estimation for lithium-ion batteries based on cross-domain transfer learning with feedback mechanism. *Journal of Energy Storage*, 70:108037.

Yassin, M. A. M., Shrestha, A., and Rabie, S. (2023). Digital twin in power system research and development: Principle, scope, and challenges. *Energy Reviews*, 2(3):100039.

Yu, Y., Wan, D., Zhao, Q., and Liu, H. (2020). Detecting Pattern Anomalies in Hydrological Time Series with Weighted Probabilistic Suffix Trees. *Water*, 12(5):1464. Number: 5 Publisher: Multidisciplinary Digital Publishing Institute.

Zeng, Y., Li, Y., and Yang, T. (2023). State of Charge Estimation for Lithium-Ion Battery Based on Unscented Kalman Filter and Long Short-Term Memory Neural Network. *Batteries*, 9(7):358.

Zhao, M., Shen, X., and Jiang, F. (2023). Research on Mechanical Vibration Measurement Method Based on Event Camera. In *2023 3rd International Conference on Energy Engineering and Power Systems (EEPS)*, pages 528–532.

APPENDIX A NONLINEAR CONTRIBUTION MATRICES

The nonlinear contribution matrices are defined given the formulation of Wadham-Gagnon et al. (2007) and the assumptions made in this thesis.

Table A.1 Formulation of the linear and nonlinear coefficients

Coefficient	Formulation
$\tilde{\mathbf{M}}_{ij}$	δ_{ij}
$\tilde{\mathbf{C}}_{ij}$	$2u\sqrt{\beta}\int_0^1 \phi_i\phi_j^{'}d\xi$
$\tilde{\mathbf{K}}_{ij}$	$\delta_{ij}\lambda^4 + u^2\int_0^1 \phi_i\phi_j^{''}d\xi + \gamma(\int_0^1 \phi_i\phi_j^{'}d\xi - \int_0^1 \phi_i\phi_j^{''}d\xi + \int_0^1 \phi_i\xi\phi_j^{''}d\xi)$
$\mathbf{B}_{ijkl}$	$u^2\int_0^1 \left(\phi_i\phi_j\phi_k\phi_l'' - \phi_j\int_\xi^1 \phi_k\phi_l''\,\mathrm{d}\xi\right)\mathrm{d}\xi$ $+y'\int_0^1 \phi_i\left(\frac{1}{2}\phi_j\phi_k\phi_l' - \frac{3}{2}(1-\xi)\phi_j\phi_k\phi_l''\right)\mathrm{d}\xi$ $+\int_0^1 \phi_i\phi_j\phi_k\phi_l'''' + 4\phi_j\phi_k\phi_l''' + \phi_j\phi_k\phi_l''\,\mathrm{d}\xi$
$\mathbf{D}_{ijkl}$	$2u\sqrt{\beta}\int_0^1 \phi_i\left(\phi_j'\phi_k'\phi_l' - \phi_j''\int_\xi^1 \phi_k'\phi_l'\,\mathrm{d}\xi\right)\mathrm{d}\xi$
$\mathbf{E}_{ijkl}$	$\int_0^1 \phi_i\left(\phi_j'\int_0^\xi \phi_k'\phi_l'\,\mathrm{d}\xi - \phi_j''\int_\xi^1\int_0^\xi \phi_k'\phi_l'\,\mathrm{d}\xi\,\mathrm{d}\xi\right)\mathrm{d}\xi$
$\mathbf{H}_{ijkl}$	$\gamma\int_0^1 \phi_i\left[\frac{1}{2}\phi_j'\phi_k'\psi_l' + (1-\xi)\left(\frac{1}{2}\phi_j'\phi_k'\psi_l'' + \phi_j'\phi_k''\psi_l'\right)\right]\mathrm{d}\xi$ $+u^2\int_0^1 \phi_i\left(\phi_j\phi_k\psi_l'' - \phi_j\int_\xi^1 \psi_k'\psi_l'\,\mathrm{d}\xi\right)\mathrm{d}\xi$ $+\int_0^1 \phi_i\phi_j\phi_k\psi_l'''' + 3\phi_j\phi_k\psi_l''' + \phi_j\phi_k'\psi_l''' + \phi_j'\phi_k''\psi_l'\,\mathrm{d}\xi$

APPENDIX B MODIFICATION TO THE MOULDING PROCEDURE

The pipe moulding process follows the step in Demenois et al. (2023). Modifications to the mould assembly imply the following:

1. The addition of four plastic washers at the insertion end to constrain the insertion of the rod and maintain its verticality.
2. Tapering of the central hole of the plastic washer at the bottom end to direct the rod. We used a 5/8 inch drill size to mill the bottom washer.

Also, since the numerical 3D reconstruction with the EVB sensor data does not need high contrast between the pipe and the white background, the blackening step is omitted. Figure B.1 illustrates the modification made to the mould.

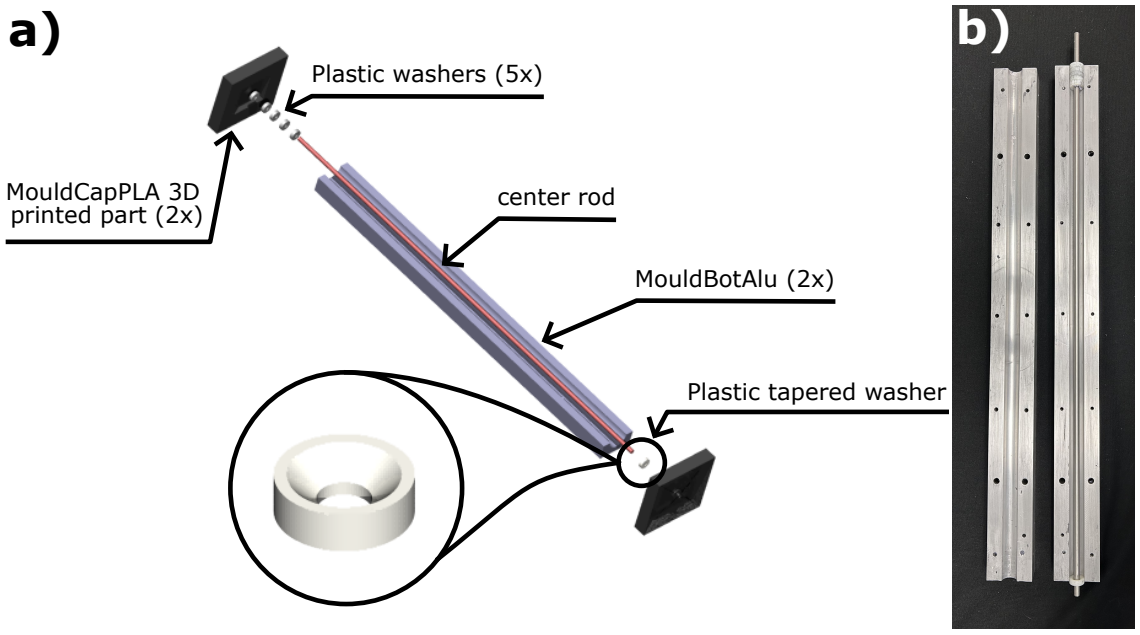

Figure B.1 **Modification to the mould assembly of Demenois et al. (2023). a)** 3D visualisation of the mould assembly. **b)** Experimental visualisation of the mould.

APPENDIX C CHARACTERISATION OF THE PIPE BY FREE VIBRATION

The method presented in Demenois (2022b) inspired by Paidoussis (2014) is used to characterise the pipe by free vibration. We use our own code to fit the polynomial through the free vibrating pipe. The code `Pipe_Characterization_Free_Motion.py` developed by Demenois (2022b) is then used to find both the Young modulus E and the Kelvin-Voigt damping parameter E^*. The latter code is available at Demenois (2022a). The general procedure is enumerate as follows:

1. To make the pipe vibrate in planar free motion, we hold the pipe perpendicularly to one of the cameras and release it from an approximately 0.2 m height.
2. We start the recording and the 2D reconstruction procedure.
3. we use the scipy.signal Butterworth low-pass filter to smooth the tip damped oscillation and perform FFT to extract the main frequency.
4. The find_peaks function of scipy.signal is used to extract the maximal amplitude value at each oscillation period. Find the logarithmic decrement :
    - The scipy.stats linregress function is employed to fit a linear function through the logarithm of the peak amplitudes.
    - The slope of the linear function is our logarithmic decrement.
5. We repeat steps 1 to 4 five times and average the frequency and the logarithmic decrement values.
6. These averaged values are then used in the pipe code Pipe_Characterization_Free_Motion.py of Demenois (2022b) to find E and E^*;

Figure C.1 presents the frequency and the logarithmic decrement plots and Table C.1, their values.

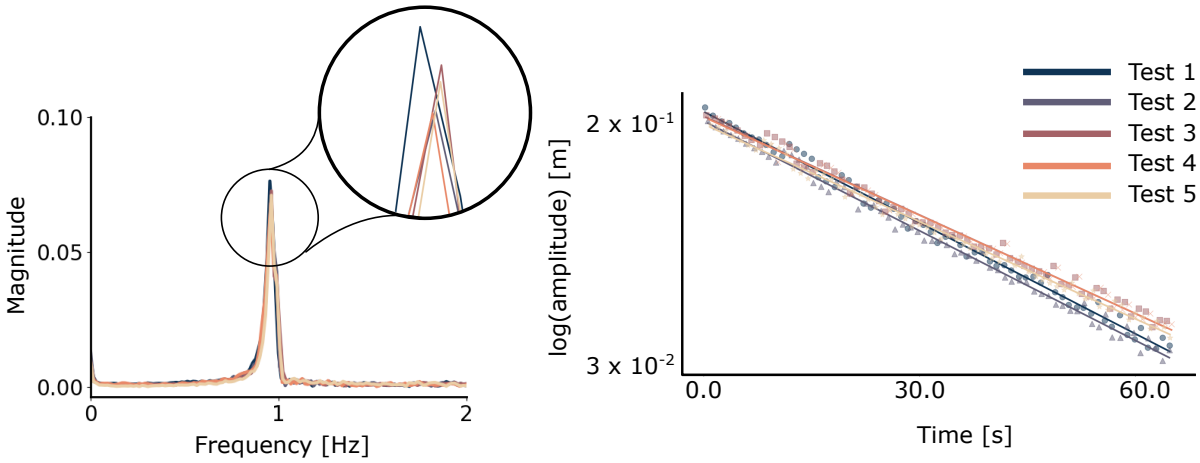

Figure C.1 **Visualisation of Frequency and logarithmic decrement δ for five free vibration analysis. a)** are the frequencies and **b)** are the linear regressions through the tip peak amplitudes of the following tests: 1 (◯); 2 (△); 3 (□); 4 (×); 5 (★)

Table C.1 Frequency and logarithmic decrement δ for five free vibration analysis.

Test	Frequency [Hz]	δ
1	0.9538	0.02741
2	0.9596	0.02707
3	0.9618	0.02434
4	0.9585	0.02424
5	0.9615	0.02401

APPENDIX D GENERALITY OF R IN THE UKF

Here, we analyse the impact of an adaptive NOMAD $\mathbf{R}$ with a forgetting factor $\alpha_{\mathbf{R}}$ on the UKF capacity to solve various flow rate cases. The initial values to start the UKF trials are the one in Chapter 5.3 with the NOMAD hypperparameters of $\mathbf{Q} = 0.000788^2$, $\mathbf{R} - 0.0195^2$, $\alpha_Q = 1$, and $\alpha_R = 0.996$. We compare the adaptive $\mathbf{R}$ flow rate estimates for a flow range of 5.2 to 6.9 m/s with the estimates made with a fixed $\mathbf{R}$ in Figure D.1.

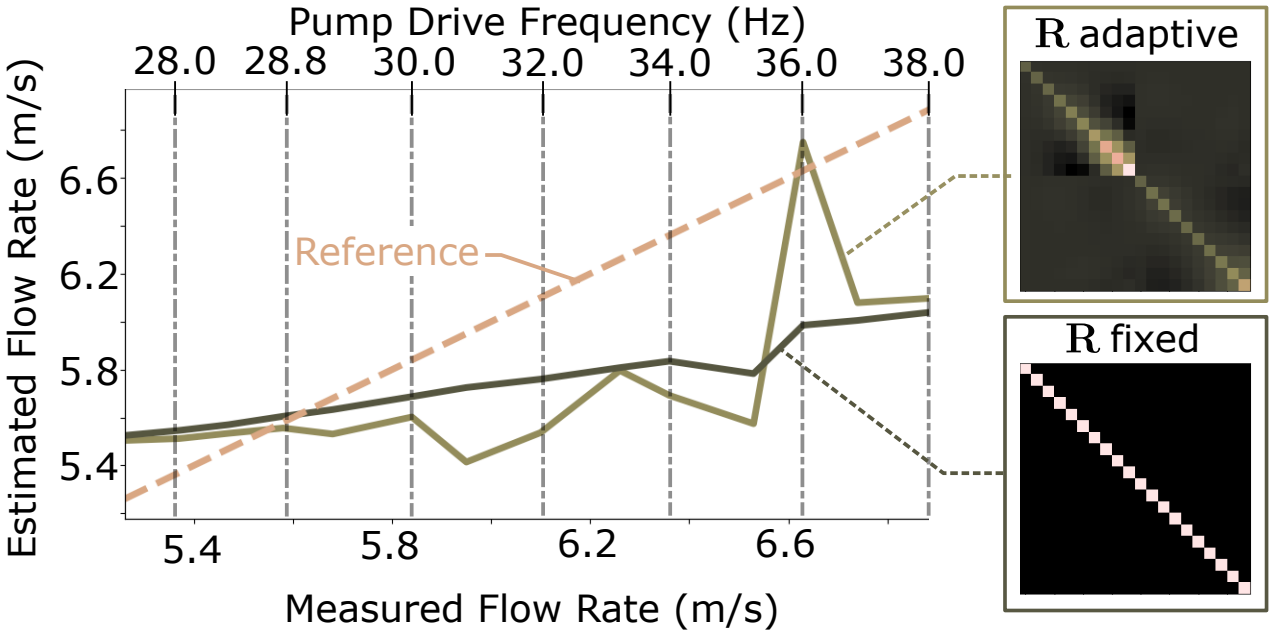

Figure D.1 **Comparison of the flow rate estimation for a UKF set with an adaptive R and a fixed R.**

In Figure D.1, both initial $\mathbf{R}$ matrices are shown on the right. The adaptive matrix $\mathbf{R}$ has a correlation between its matrix cell compared to the fixed matrix $\mathbf{R}$ which is a diagonal matrix. In the UKF, once we perform system identification, one can observe that the estimation with adaptive $\mathbf{R}$ are erratic for trials where $U \geq 5.6$ m/s. For trials where $U < 5.6$ m/s, the error with the true flow rate is slightly less than the error obtained with the fixed $\mathbf{R}$.

APPENDIX E MATURITY MODEL OF A DIGITAL TWIN

The tables from Uhlenkamp et al. (2022) are summarised in Table E.1. The first column of the present table is the subcharacteristic of the category, the second column is the attributes, and the third column is the score. A subcharacteristic with no score is marked with "-".

Table E.1 Maturity model of Uhlenkamp et al. (2022)

Context		
Reference Object	Pipe conveying fluid	-
Life cycle phase	Middle of life	-
Benefits	Service optimisation	-
Application domain	Mechanical system's maintenance	-
Computing Capabilities		
Trigger types	Static (once at a time)	1/3
Look-ahead perspective	short-term	1/3
Computing capabilities	Minutes	0/1
Update frequency - Input	real-time	2/2
Update frequency - Output	Periodic	0/1
Model		
DT Creation Approach	Rettro-fitted	0/1
Modelled Characteristics	Geometry-kinematics	-
Digital Model Types	UKF : Physic-based LKF-DMD : Hybrid Dynamic (Time is represented) Stochastic	-
Authenticity	middle	1.5/3
Maintenance	Model descriptions are not formalised and maintained manually	0/5
Modularity	Unit	0/2
Data		
Storage	Local servers	-
Scope	Local-scope data is collected regularly for one asset and its environment	1/4
Quality	Accurate, complete and unique	3/7
Sources	Physical	1/2
Interpretation	Structured	0/1
Control		
Level of cognition	Descriptive and diagnostic analytics	2/5
Autonomy	User assistance	1/5
Learning Capabilities	No learning	0/4
Human-Machine Interaction		
Interaction Devices	Traditional, unimodal interaction devices	1/3
Human Interaction Capabilities	Predefined, structures reports	1/4
Integration		
DT interaction	Standalone	1/4
Hierarchy	No-collaborative DT	-
Connection Mode	Digital shadow	2/3
User Focus	Single User	1/4
Interorganisational Integration/Collaboration	No collaboration	0/4

APPENDIX F PYTHON LIBRARIES AND VERSIONS

In this work, Python 3.9.19 is used within Spyder 5.5.1 and VS Code 1.98.2.

Table F.1 Python libraries and versions used in this project

Library	Version	Notes
alive-progress	3.1.5	Computational time evaluation
cmcrameri	1.9	Scientific color maps for color blind people
matplotlib	0.1.6	post-process
numba	0.61.0	Just-in-time compiler
numpy	2.0.2	-
pynomadbbo	4.5.0	NOMAD optimiser
ray	2.41.0	Paralleliser
scikit-learn	1.5.1	Polynomial fitting
scipy	1.13.1	-
seaborn	0.13.2	Figures

APPENDIX G DIGITAL TWIN FRAMEWORK CODE MAP

The proposed DT framework is developed in Python and follows the code map presented in Figure G.1 below.

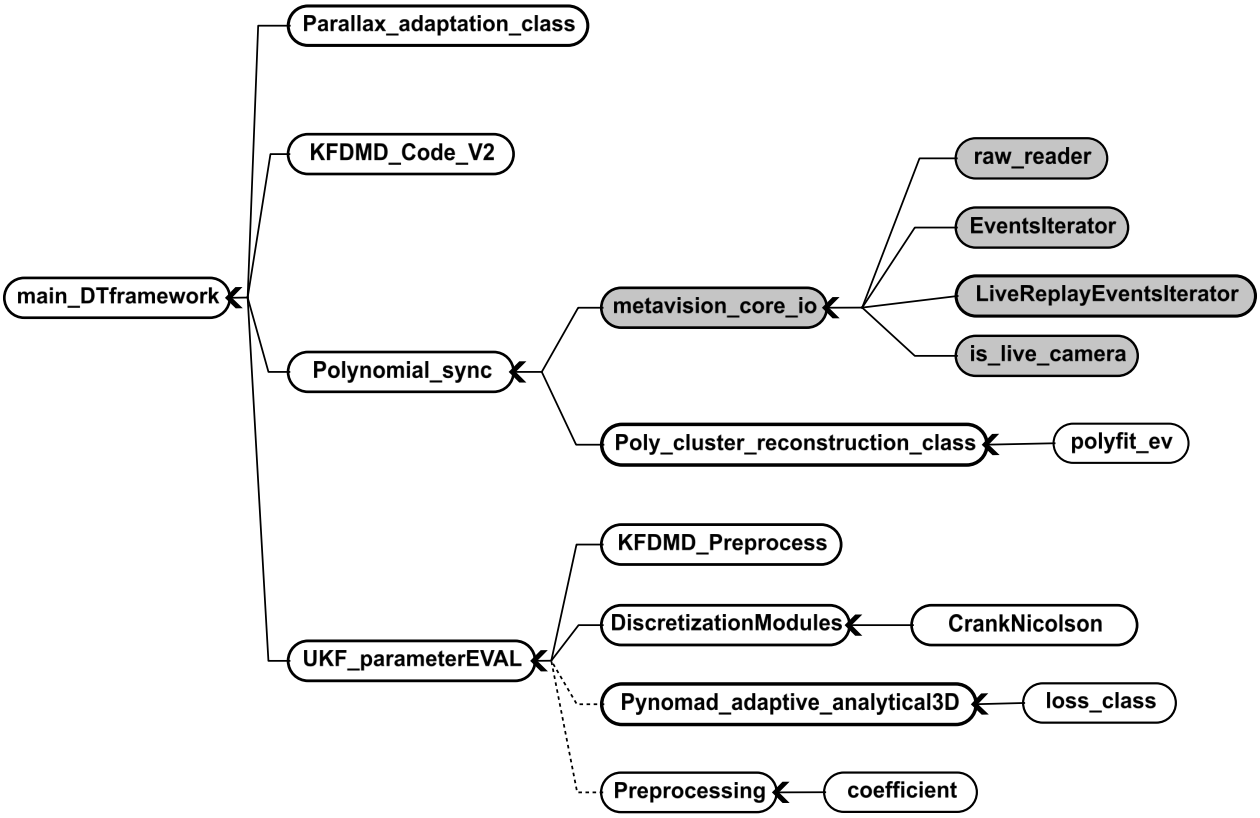

Figure G.1 **Digital Twin Framework Code Map** All the codes are implemented inside the main_DTframework code. (——) concerns direct online connexion. (- - - -) is offline procedure to run before running the DT framework code.

Metavision codes can be find in PROPHESEE (2024).