UNIVERSITY OF CALIFORNIA SAN DIEGO

# Optimized Structural Health Monitoring for Inland Waterways Infrastructure Using Model-Based Diagnostics and Prognostics

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Structural Engineering

by

Zihan Wu

Committee in charge:

Professor Michael D. Todd, Chair
Professor Nikolay Atanasov
Professor Joel P. Conte
Professor Charles R. Farrar
Professor Zhen Hu

2024

The Dissertation of Zihan Wu is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

# DEDICATION

This dissertation is dedicated to my wife, Yening Shu, whose unwavering support and patience have been my anchor. To my father, Shunying Wu, and my mother, Hui Guo, who have been my pillars of strength and encouragement throughout this journey. Their love and sacrifices have illuminated my path during the most challenging times.

# EPIGRAPH

*What we observe is not nature itself,*
*but nature exposed to our method of questioning.*

— *Werner Heisenberg*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

Completing my PhD has been a journey I'll never forget, and it wouldn't have been possible without the support and guidance of many. I'm incredibly grateful to everyone who played a part.

First and foremost, I extend my heartfelt thanks to Prof. Michael Todd, my committee chair. His support and trust, both academically and personally, have been a cornerstone of my development. Prof. Todd's mentorship has not only guided me through my research but has also inspired me to strive for excellence in all aspects of my life. I am truly proud of being his student and grateful for the role model he has been to me.

I also want to thank my committee members, Prof. Nikolay Atanasov, Prof. Joel Conte, Prof. Charles Farrar, and Prof. Zhen Hu. Their invaluable suggestions and insights have significantly enriched my research journey. A huge thank you goes to Prof. Zhen Hu, whose expertise and readiness to share his knowledge from the start of my PhD have been key to my learning and growth.

My gratitude extends to Dr. See Yenn Chong, who opened the doors of research to me and provided patient guidance and support as a friend, especially through challenging times. This journey would have been much harder without his encouragement.

I'm grateful to Dr. Manuel Vega for his mentorship, teaching me how to think and work as a researcher. To my colleagues and collaborators, Dr. Mayank Chadha, Dr. Yichao Yang, Dr. Guofeng Qian, Adrielly Hokama Razzini, and David Najera, I'll always treasure the memories and achievements we've shared.

The research presented in this dissertation was financially supported by the U.S. Army Corps of Engineers, specifically through program managers Matt Smith and Brian Eick. This support has been crucial in advancing my research and is deeply appreciated.

Portions of this dissertation have been published or are currently being considered for publication.

Chapter 2 is composed of a second-authored publication: Fillmore, T. B., **Wu, Z.**,

Vega, M. A., Hu, Z., & Todd, M. D. (2022). A surrogate model to accelerate non-intrusive global–local simulations of cracked steel structures. Structural and Multidisciplinary Optimization, 65(7), 208.

Chapter 3 is composed of a first-authored publication: **Wu, Z.**, Fillmore, T. B., Vega, M. A., Hu, Z., & Todd, M. D. (2022). Diagnostics and prognostics of multi-mode failure scenarios in miter gates using multiple data sources and a dynamic Bayesian network. Structural and Multidisciplinary Optimization, 65(9), 270.

Chapter 4 is currently in preparation for publication, the dissertation author was the primary investigator and author of this paper: **Wu, Z.**, Hu, Z., & Todd, M. D. Deep Learning-Based Automatic Crack Detection in the Underwater Environment of Miter Gates. Structural Health Monitoring.

Chapter 5 is composed of a first-authored publication: **Wu, Z.**, Zeng, J., Hu, Z., & Todd, M. D. (2023). Optimization of unmanned aerial vehicle inspection strategy for infrastructure based on model-enabled diagnostics and prognostics. Mechanical Systems and Signal Processing, 204, 110841.

## VITA

2018 B.S., Chang'an University, China

2020 M.S., University of California San Diego

2024 Ph.D., University of California San Diego

## PUBLICATIONS

**Wu, Z.**, Zeng, J., Hu, Z., & Todd, M. D. (2023). Optimization of unmanned aerial vehicle inspection strategy for infrastructure based on model-enabled diagnostics and prognostics. Mechanical Systems and Signal Processing, 204, 110841.

Zeng, J., **Wu, Z.**, Todd, M. D., & Hu, Z. (2023). Bayes risk-based mission planning of Unmanned Aerial Vehicles for autonomous damage inspection. Mechanical Systems and Signal Processing, 187, 109958.

**Wu, Z.**, Fillmore, T. B., Vega, M. A., Hu, Z., & Todd, M. D. (2022). Diagnostics and prognostics of multi-mode failure scenarios in miter gates using multiple data sources and a dynamic Bayesian network. Structural and Multidisciplinary Optimization, 65(9), 270.

Fillmore, T. B., **Wu, Z.**, Vega, M. A., Hu, Z., & Todd, M. D. (2022). A surrogate model to accelerate non-intrusive global–local simulations of cracked steel structures. Structural and Multidisciplinary Optimization, 65(7), 208.

**Wu, Z.**, Chong, S. Y., & Todd, M. D. (2021). Laser ultrasonic imaging of wavefield spatial gradients for damage detection. Structural Health Monitoring, 20(3), 960-977.

ABSTRACT OF THE DISSERTATION

Optimized Structural Health Monitoring for Inland Waterways Infrastructure Using Model-Based Diagnostics and Prognostics

by

Zihan Wu

Doctor of Philosophy in Structural Engineering

University of California San Diego, 2024

Professor Michael D. Todd, Chair

Inland waterways infrastructure such as miter gates are subject to damage like cracking and corrosion due to long (∼50 years) service lives with extensive water exposure. With the advancement of modern sensing technologies, there's a vast potential for Structural Health Monitoring (SHM) to transition into a more intelligent and efficient technology that can integrate multiple data sources for enhanced damage diagnostics and inform predictive inspection and maintenance strategies. This research presents a comprehensive optimization framework for the diagnosis and prognosis of such infrastructure. The framework first proposes a novel iterative global-local method for efficient and accurate forward modeling of

structural damage in miter gates. It then develops an innovative diagnostic and prognostic framework that not only integrates multiple data sources for structures with multi-failure modes but also analyzes the environmental factors influencing SHM, offering insights into the challenges and solutions for real-world inspections. Furthermore, it introduces a physics-informed inspection planning framework, underpinned by model-based diagnostics and prognostics, leveraging the benefits of digital twin and deep learning technologies. This work represents a significant advancement for a certain class of SHM, providing a robust methodology for improving the lifespan and ensuring the safety of critical waterway infrastructure, marking a crucial step toward the future of infrastructure inspection and maintenance.

# Chapter 1

# Introduction

## 1.1 Overview of Structural Health Monitoring Systems

Structural Health Monitoring (SHM) refers to the process of implementing a damage detection strategy for built structures. Such a process involves the integration of sensors that collect data from structures over time, the extraction of damage-sensitive features from the collected data, and the statistical analysis of those features to derive actionable insights about structural health conditions. For long-term SHM, the objectives are twofold: 1) to ensure the structure's safety and functionality considering degradation from operational environments, and 2) to optimize inspection and maintenance strategies for timeliness and cost-effectiveness (Brownjohn, 2007; Farrar and Worden, 2007; Sohn and Farrar, 2001).

Inland waterways infrastructure presents unique challenges for SHM due to their expansive dimensions, complex geometry, and unique operating conditions. To illustrate, the United States Army Corps of Engineers (USACE) operates and maintains 236 lock chambers at 191 lock sites on 41 US waterways (Eick, Treece, Spencer Jr, Smith, Sweeney, Alexander and Foltz, 2018). These structures require a well-designed monitoring strategy that can effectively detect damage both on a global and local scale. Given their multifunctional roles, unexpected failure can result in severe consequences. The closure of a lock could cost the US economy up to $3 million per day (Eick, Treece, Spencer Jr,

Smith, Sweeney, Alexander and Foltz, 2018). As a crucial component of inland waterway systems, miter gates facilitate essential tasks including waterway transportation, flood control, and water supply. A significant portion of miter gates remain underwater, which gives rise to varied damage degradation patterns, such as support structure material loss (leading to loss of proper load transfer), pitting corrosion, and fatigue cracks. Presently, maintenance relies on periodic inspections that require fully dewatering the chamber for manual examination, a process that is time-intensive, costly, and inefficient. Furthermore, the underwater context complicates inspection and maintenance due to factors such as water turbidity, restricted accessibility, and limited visibility. Given these challenges, there is an increasing need for a specialized SHM framework tailored for inland waterways infrastructure.

## 1.2 SHM-Informed Asset Life-Cycle Management

SHM systems are a central component in the life-cycle management (i.e., operation, inspection, and maintenance) of inland waterways infrastructure. The overall goal is to maximize the remaining useful life (RUL) of these structures while minimizing the total life-cycle costs, which typically can be characterized by three phases: early detection and diagnosis, predictive analysis and prognostics, and life-cycle management.

### 1.2.1 Early detection and diagnosis

As a fundamental component of SHM, diagnosis refers to the process of identifying and characterizing damage in a structure based on measured data. Diagnosis offers insights into the current health state of a structure, where the primary goal is to detect, locate, and quantify damage or deterioration in the structure (Farrar and Worden, 2012; Sohn, Farrar, Hemez, Shunk, Stinemates, Nadler and Czarnecki, 2003). Numerous methodologies have been developed for damage diagnosis in SHM. The following are some of the widely used diagnostic techniques.

### Model-based diagnosis

Model-based diagnosis relies on a computational model or mathematical model representing the structure's behavior. The potential damage locations and severities can be identified from the discrepancies between the model's predictions and the measurements from the actual assets. In recent years, the "digital twin" concept has drawn intensive attention to model-based diagnosis. A Digital Twin (DT) is a digital counterpart of a physical object or system, which in SHM, serves as an advanced computational model that "mirrors" the current state of the structure, providing a platform for real-time monitoring, diagnostics, and prognostics. The integration of DT in SHM has enabled more detailed and accurate damage detection, owing to their ability to incorporate and process vast amounts of data from various sources (Glaessgen and Stargel, 2012; Li, Mahadevan, Ling, Choze and Wang, 2017a; Tao, Cheng, Qi, Zhang, Zhang and Sui, 2018). Finite Element (FE) analysis, a numerical technique used for finding approximate solutions to boundary value problems for differential equations, serves as the backbone for many DT representations of structures. These FE models can capture intricate details about a structure's geometry, material properties, boundary conditions, and loading scenarios, providing more accurate insights into potential damage mechanisms (Moravej, Jamali, Chan and Nguyen, 2017; Tao, Xiao, Qi, Cheng and Ji, 2022).

### Data-driven diagnosis

The rise of computational capacities and smart sensing technologies have quickly advanced the evolution of data-driven approaches in SHM. These methods, as opposed to traditional model-based techniques, rely on the features and patterns present in the collected observations to identify and diagnose damage in structures. Among them, Machine Learning (ML) has become an indispensable tool, which can be trained to automatically learn from data, to recognize patterns corresponding to damage, and to improve the model performance as more data becomes available (Bishop, 2006). The data-driven approaches

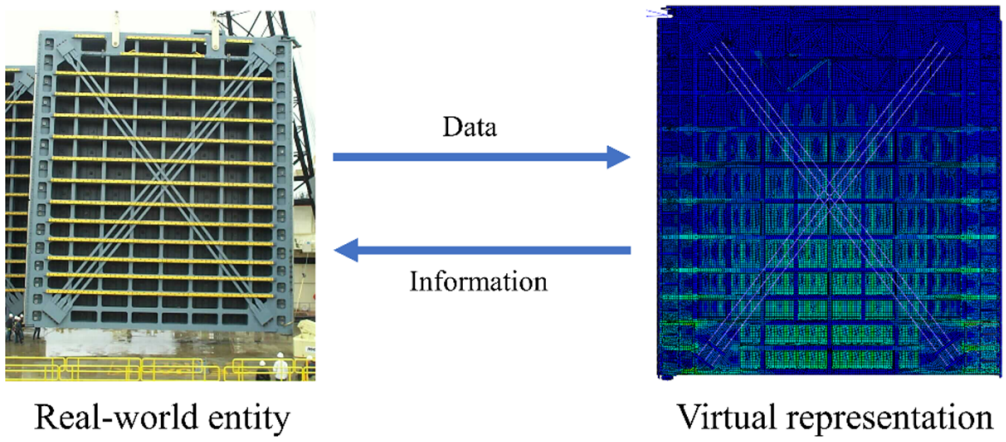



**Figure 1.1.** Digital twin concept.

can be broadly categorized into supervised and unsupervised learning, each serving unique roles in SHM.

Unsupervised learning techniques do not require labeled data and are primarily used for dimensionality reduction and clustering. For instance, Singular Value Decomposition (SVD) and Principal Component Analysis (PCA) are some dimensionality reduction techniques widely used in SHM to transform measurement data into lower-dimensional subspaces without significant losses in data variance. Such techniques can be crucial for identifying anomalies in structural behavior, enhancing computational efficiency, and highlighting the most dominant modes of the data (García-Macías and Ubertini, 2021). Besides, K-Means Clustering partitions the dataset into K non-overlapping clusters to minimize the within-cluster sum of squares. In the context of SHM, K-means clustering can be applied to digital signals to categorize different material properties or structural conditions, aiding in the identification of damage-related anomalies (Liu, Malinowski, Pawłowski, Wu and Todd, 2023a). In unsupervised approaches, data is only available from nominal (baseline) conditions, and comparisons are made between test data patterns and these reference patterns. They may sometimes only indicate the presence of an anomaly, struggling to provide detailed diagnostics or prognostics information.

In supervised learning, the model is trained on a labeled dataset, from all the targeted classes of health. This approach is particularly powerful for tasks such as damage classification and quantification. For example, Gaussian Processes (GP) is a type of non-parametric Bayesian method used for regression and classification tasks. In SHM, they can model the uncertainty in structural responses, making them suitable for structures with complex interactions and noise in the measurements (Worden and Manson, 2007). GP models have been employed for predicting structural responses, modeling the evolution of damage, and uncertainty quantification in SHM (Avendano-Valencia, Chatzi and Tcherniak, 2020). Convolutional Neural Networks (CNN), a deep learning model primarily used for image processing, have a specific architecture designed to learn pixel-level features automatically and adaptively from data. With the increasing use of visual inspections via drones or other imaging systems, CNNs have been employed to detect and localize damage such as cracks, corrosion, and other defects in various structural elements (Azimi, Eslamlou and Pekcan, 2020; Cha, Choi and Büyüköztürk, 2017).

In conclusion, the applicability of data-driven diagnosis in SHM depends on the availability of data and the specific monitoring objectives. While supervised methods offer detailed insights when sufficient labeled data is available, unsupervised methods provide capabilities for anomaly detection with the scarcity of damaged data.

### 1.2.2 Predictive analysis and prognostics

Prognosis, or predictive modeling of future structural state, is crucial in SHM for informed decision-making and long-term structural reliability by using the gathered damage information to simulate the evolution of the damage and predict useful limit states that inform RUL. Such a process helps to avoid unexpected failures and downtime and has been particularly critical in inland waterways infrastructure such as miter gates, dams, and other such large-scale structures.

RUL estimation refers to the process of predicting the time remaining until a

structural component reaches its end of life (EOL) or a predefined failure threshold. Figure 1.2 shows an illustration of how to perform RUL estimation based on failure prognosis. The EOL is defined as the time when the damage state of the structure reaches a predefined failure threshold. The RUL is determined as the difference between the predicted EOL and the current time step.

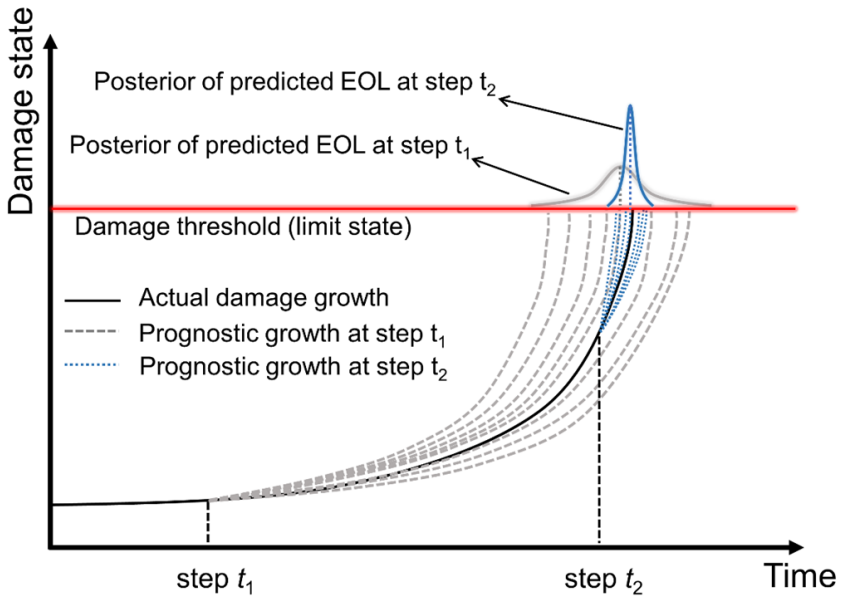



**Figure 1.2.** Illustration of RUL prediction in prognosis.

While RUL estimation stands as a crucial component in predictive maintenance, the challenge lies in the uncertainties in modeling the underlying deterioration processes, which arise primarily from the selected methodology and the quality of the data (Farrar and Lieven, 2007).

Similarly, prognosis can be categorized into two main types: model-based and data-driven prognosis. Model-based methods rely on mathematical models that describe the deterioration or damage evolution over time in a structure. An illustrative example of this is the use of Paris's Law to estimate the evolution of fatigue cracks (Suresh, 1998). These methods often require accurate model parameters and loading conditions, which

highly depend on the estimation accuracy during the diagnostic phase (Farrar and Worden, 2012). Model-based prognosis is particularly prevalent in the domain of inland waterways infrastructure, where accurate modeling of material degradation under environmental loading is crucial (Brownjohn, 2007).

On the other hand, data-driven prognosis leverages ML and statistical methods to predict future states of the structure based on historical and real-time data. Techniques such as Bayesian approaches, time series analysis, and neural networks have gained popularity in this domain, driven by advances in sensing technologies and data acquisition capabilities. These methods have shown promise in inland waterway applications, providing valuable insights for maintenance and decision-making processes (Negi, Kromanis, Dorée and Wijnberg, 2024; Vega, Hu, Yang, Chadha and Todd, 2022).

In practice, a hybrid approach combining both model-based and data-driven methods is often adopted to enhance the accuracy and robustness of predictions, particularly in the presence of uncertainties and limited data (Azimi, Eslamlou and Pekcan, 2020). This integration allows for the compensation of potential shortcomings in each individual approach, ensuring a more comprehensive and reliable prognosis.

### 1.2.3 Life-cycle management

Once damage is diagnosed and its future progression is predicted, the next challenge is to determine an appropriate response to the structure's life-cycle span, i.e., life-cycle management. Generally, life-cycle management refers to the systematic process of managing the entire life cycle of a structure to maximize its value and performance while minimizing costs and risks (Frangopol, Saydam and Kim, 2012). It involves the integration of various processes including inspection, maintenance, and repair. With accurate diagnosis and prognosis, inspection and maintenance activities can be optimized to ensure the longevity and performance of a structure.

Various inspection techniques such as ultrasonic testing, radiography, and thermog-

raphy have been widely used in various industries to identify and characterize structural deficiencies. However, in the context of large-scale structures like inland waterway systems, emerging technologies such as drone inspections and sensor networks are usually more popular. These technologies enable more frequent and comprehensive monitoring of structures, providing richer datasets for diagnosis and prognosis. The integration of machine learning and computer vision techniques further enhances the capability of these systems, automating the detection and analysis of structural issues.

With the advancement of various inspection techniques, predictive maintenance is gradually being intensively adopted instead of periodic maintenance (Kim, Frangopol and Soliman, 2013; Carvalho, Soares, Vita, Francisco, Basto and Alcalá, 2019). By continuously monitoring the condition of the structure and predicting the onset of failure, maintenance activities can be scheduled adaptively based on the risk profile of the structure, thereby preventing unexpected failures and minimizing downtime. This approach not only ensures the optimal performance of the structure but also contributes to cost savings by reducing the need for emergency repairs and extending the service life of the structure.

In essence, SHM serves as the integrated decision-making architecture for structural asset life-cycle management. It provides continuous feedback on the health status of structures, predicts their future states, and suggests optimal strategies based on continuous observation and interpretation.

## 1.3 Critical Concerns in Modern SHM Techniques

Despite the indispensable role of SHM for inland waterways infrastructure, contemporary SHM systems face a myriad challenges. Addressing these challenges forms the core of this thesis and corresponds to the content of subsequent chapters. The primary challenges can be defined into four themes.

### 1.3.1 Multi-scale modeling

Inland waterways infrastructure, such as miter gates, display a variety of damage features across different scales, ranging from global deformations that impact the entire structure to local cracks and corrosion. Thus, multi-scale modeling has become increasingly important in SHM for accurately understanding and simulating the behavior of large structures. However, the discrepancy in length scales between the overall structure and the localized damage features poses significant challenges in terms of numerical modeling and computational demand (Allix and Gosselet, 2020; Oden, Vemaganti and Moës, 1999).

The challenge primarily arises from the need for fine discretization to accurately capture the small-scale damage features (e.g., cracks), while also modeling the global structure. This results in unaffordably large increases in the computable degrees of freedom in the simulations, leading to extremely high computational costs and time-consuming calculations.

To address this computational challenge, there is a growing need for surrogate models. Surrogates serve as approximate models that can significantly reduce the required computational time and resources, while still providing sufficiently accurate predictions of structural behavior. To bridge the gap between the different scales without sacrificing accuracy, a surrogate-based multi-scale modeling method is desired to capture the critical features and behaviors of the structure at both the global and local levels, ensuring that the physics of small-scale damage are adequately represented in the overall structural response.

A surrogate model to accelerate non-intrusive global–local simulations of cracked steel structures is discussed in depth in Chapter 2.

### 1.3.2 Diagnostics and prognostics with diverse data sources

In the rapidly advancing field of SHM, the integration of diverse data sources has become an indispensable element. One prominent advance has been the adoption of Unmanned Aerial Vehicles (UAVs) equipped with vision cameras, which have transformed the traditional observational data from strain measurements and manual visual inspections into high-resolution digital images, opening the doors to a new realm of data acquisition. Each of these data types offers a unique lens through which the structural state can be observed. For instance, while strain gauges provide a measure of stress and strain distribution, vision-based inspection is capable of capturing surface abnormalities and degradation patterns in the early stage.

However, with the diversity in data acquisition comes a crucial challenge: how to synergistically leverage these varied data sources for a robust SHM framework, considering different types of observational data inherently have unique sensitivities and detectability to different damage features. In the particular application of large-scale structures such as miter gates, which are subject to complex degradation and multi-mode failure scenarios, a SHM system that overcomes the limitations of individual data sources and performs diagnosis and prognosis with multiple data sources to enhance the accuracy and reliability of damage detection, localization, and characterization is eventually desired.

Chapter 3 will detail the diagnostics and prognostics of multi-mode failure scenarios in miter gates using multiple data sources and a dynamic Bayesian network.

### 1.3.3 Data challenges in ML-based SHM

The integration of Machine Learning (ML) and Deep Learning (DL) in SHM has created new avenues for advanced damage detection and assessment. For example, the application of image processing techniques and CNNs has shown great potential in enhancing the performance of image-based damage identification.

Transitioning from the capabilities of unsupervised learning, which excels in anomaly and outlier detection within structures as we just described, we encounter the necessity for supervised learning to achieve more detailed and informative SHM tasks, such as diagnostics and predictive prognostics. However, one of the primary issues in ML/DL-based SHM is the scarcity of damaged data for training these sophisticated models. For example, the most difficult task in constructing a deep CNN model is to collect a large training dataset that covers different aspects and states of the target structures (Tsai, Tsai, Hsu and Wu, 2017). In fact, most structures fortunately do not experience damage at a frequency sufficient for training, resulting in an imbalanced dataset that is predominantly composed of healthy state data. This imbalance can lead to models that are not well-trained to recognize and diagnose damaged states, which is the critical function of SHM.

Moreover, the quality of the available data is often compromised, especially in the context of underwater infrastructure. Environmental variables such as water turbidity, light scattering, and inconsistencies in-camera focus can severely degrade the probability of detection of image-based damage detection methods.

To address the lack of damaged data, one promising solution is the use of synthetic data generated through tools like Blender (Mayershofer, Ge and Fottner, 2021). Blender enables the creation of realistic images of structures with control of various damage scenarios and parameters, augmenting the limited real-world datasets. This synthetic data can be used to train CNN models, ensuring that they are trained by a wide variety of damage cases and learn to identify subtle signs of deterioration.

In summary, ML/DL-based SHM introduces a set of challenges related to data availability and quality. Addressing these challenges requires a combination of synthetic data generation, advanced image processing techniques, and appropriate deep learning models, ensuring accurate damage detection even in complex environments such as underwater infrastructure.

Chapter 4 is dedicated to understanding the environmental complexities inherent

in underwater SHM.

### 1.3.4 Bridging predictive analytics and decision-making in SHM

Translating predictive analytics from diagnosis and prognosis into feasible and strategic inspection and maintenance actions is a vital aspect that ensures the practical application of SHM insights. However, this translation process, especially when enabled by model-based diagnostics and prognostics, is rarely studied compared with conventional data-driven lifecycle management frameworks.

The challenge is brought by the dilemma of the trade-off between inspection and maintenance. On one hand, there's the question of inspection frequency: how often should a structure be inspected to ensure its integrity? On the other hand, there's the question of action protocols: at what point does the damage state require repair or maintenance?

Answering these questions requires the integration of diagnostics, prognostics, and strategic optimization. Frequent inspections, while ensuring higher detection probabilities and reduced uncertainties, come with much higher costs. Conversely, simply increasing the interval between inspections may lower costs but at the potential expense of reliability (e.g., failure consequence costs can dramatically increase) and possibly increase maintenance expenses in the long run.

Addressing this challenge calls for statistical approaches like Bayes risk analysis, which offers a quantitative framework to evaluate the trade-offs between inspection, maintenance, and associated risks. Optimization techniques, such as Bayesian optimization, offer a powerful tool for identifying optimal strategies that balance cost, risk, and reliability. Thus, actionable decision-making in SHM that integrates diagnostics, prognostics, and optimization is highly desired for model-informed, time-efficient, and cost-effective inspection/maintenance decisions.

Chapter 5 will explain the optimization of unmanned aerial vehicle inspection

strategy for infrastructure based on model-enabled diagnostics and prognostics. Through the lens of the UAV inspection scenario, the chapter will explore how SHM predictions can be actualized into optimal strategies for asset maintenance.

## 1.4 Research Objectives, Organizations of the Thesis, and Contributions

The high-level aim of this research is to address the challenges particularly focusing on inland waterways infrastructure as identified in Section 1.3. To achieve this, the research was structured with four specific objectives: 1. Integrate multiscale modeling, 2. Data fusion for multi-failure structure, 3. Automatic crack detection, and 4. Optimize inspection strategies.

The remainder of the dissertation has been organized as follows: Chapter 2 delves into the development of the global-local iterative process, focusing on addressing the forward problem in SHM. Chapter 3 introduces the integration of diverse data sets for diagnostics and prognostics, explaining the novel methodology of merging sensor data and image data. Chapter 4 explores the optimization of UAV-based inspection strategies, focusing on the balance between inspection frequency, associated costs, and the quality of data obtained. Chapter 5 dives deep into the impact of environmental variables on SHM, particularly through the lens of underwater imaging and deep learning models. Chapter 6 presents a comprehensive conclusion, drawing from all the findings and highlighting the practical implications and potential future research directions.

The main contributions presented in the following chapters can be summarized:

- This work proposes a novel iterative global-local method for efficient and accurate structural health monitoring. The challenge of computational efficiency for scale separation has been addressed by a GP-based surrogate model, enhancing the simulation speed of fine-discretized modeling.

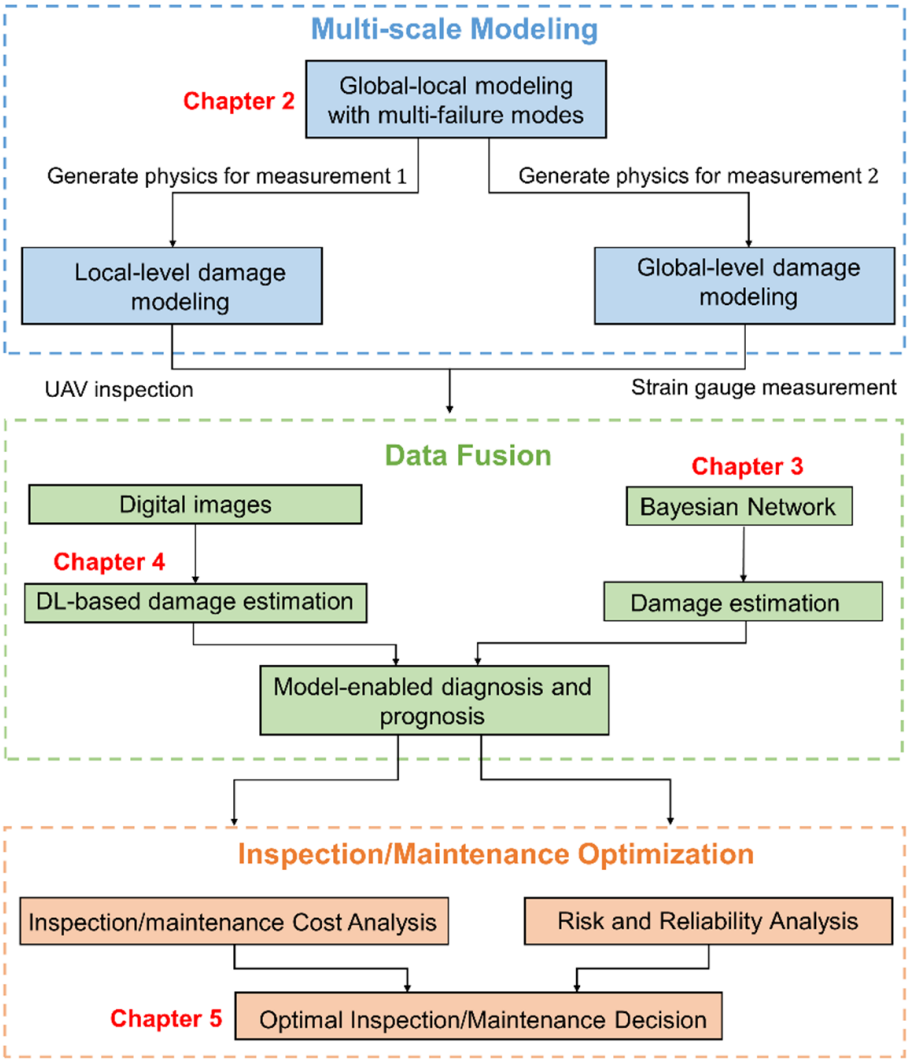



**Figure 1.3.** The optimized SHM framework.

- This work develops an innovative diagnostic and prognostic framework that integrates multiple data sources for structures with multi-failure mode. Sensor data and image data are merged using a dynamic Bayesian network, to achieve accurate damage diagnostics and prognostics for multiple damage features at different scales.

- This work analyzes the environmental factors influencing SHM, providing insights into the challenges and solutions for real-world inspections. Synthetic underwater images of miter gates are generated to study the effects of various factors on deep

learning-based crack detection accuracy.

- This work presents a physics-informed inspection planning framework based on model-based diagnostics and prognostics. Such a framework facilitates a bidirectional information exchange between UAV inspection planning and structural health assessment. A comprehensive analysis of how key UAV inspection parameters affect the overall structural life-cycle cost is performed.

# Chapter 2

# A Surrogate Model to Accelerate Non-intrusive Global-Local Simulations of Cracked Steel Structures

## 2.1 Abstract

Physics-based digital twins often require many computations to diagnose current and predict future damage states in structures. This research proposes a novel iterative global-local method, where the local numerical model is replaced with a surrogate to simulate cracking quickly on large steel structures. The iterative global-local method bridges the scales from the operational level of a large steel structure to that of a cracked component. The linear global domain is efficiently simulated using static condensation, and the cracked local domain is quickly simulated using the adaptive surrogate modeling method proposed herein. This work compares the solution time and accuracy of the proposed surrogate iterative global-local method with a reference model, a submodeling model, and an iterative global-local method with no surrogate model for the local domain. It is found that the surrogate iterative global-local method gives the fastest solution time with comparatively accurate results.

## 2.2 Introduction

With current computational mechanics technology, physics-based digital twins can diagnose and predict crack damage in structures. However, techniques to infer such information require the exploration of many crack state possibilities, which is computationally expensive. This research proposes a surrogate iterative global-local method to quickly simulate many instances of a stationary crack on a large steel structure.

This research uses a miter gate problem as its case study, as shown in Fig. 2.1, although the developed approach is easily applicable to other structures. Miter gates are critical components of river navigation that swing open and shut to allow boat passage through a navigation lock chamber. When closed, miter gates act as damming surfaces, allowing the water in the lock chamber to rise or fall. These two processes combined allow the lock chamber to act as a boat elevator, allowing boats to bypass dams and their accompanying water level differences. Some of the most important structural parts of the gates are submerged during operation, making visual inspection difficult, and leaving an information gap that digital twin technology aims to fill. This chapter describes the miter gate example problem more fully in Section 2.4.

The main problem in simulating large steel structural performance (e.g. miter gate in Fig. 2.1) with component-scale cracks is the separation in length scales. Miter gate structures may be tens of meters tall and wide, but their (possibly stable) cracks may be as long as a few cm. Thus the structure and crack features are two orders of magnitude different in scale, complicating numerical model discretization and increasing computational cost. In the miter gate numerical model, a small solid cracked part (local region) is tied to a pristine shell structure (global region) in Abaqus. A fully coupled at-scale simulation must include the global region's behavior to find crack effects, greatly adding to the computational burden of the digital twin.

Zooming or submodeling can be used to separate the cracked portion of the

**Figure 2.1.** An open miter gate.

structure from the pristine portion for reduced computational cost. The submodeling method transfers the global region solution to the shared boundary with the local region. This is computationally cheap and built into several commercial software including Ansys and Abaqus, but it fundamentally relies on Saint-Venant's principle, which states the difference between the effects of two different but statically equivalent loads becomes very small at sufficiently large distances from load. It will be shown that this principle does not hold for the miter gate example.

A generalization of the submodeling method is the iterative global-local (IGL) method (Allix and Gosselet, 2020). The IGL method provides a mechanism to obtain much more accurate solutions than submodeling via a similar numerical strategy: the global gate's displacements are imposed on the local model boundary. Then a feedback loop finds the local boundary reaction, compares it to the global boundary traction, and

applies the calculated immersed surface force to a new global computational job. Iterations can be performed until the solution is sufficiently accurate. The basic process is shown in Fig. 2.2. The IGL method converges to the reference combined problem given enough iterations. Thus the IGL method can be viewed as a bridge between the submodeling and tying methods, providing increased accuracy over submodeling at the expense of increased computational cost over submodeling.

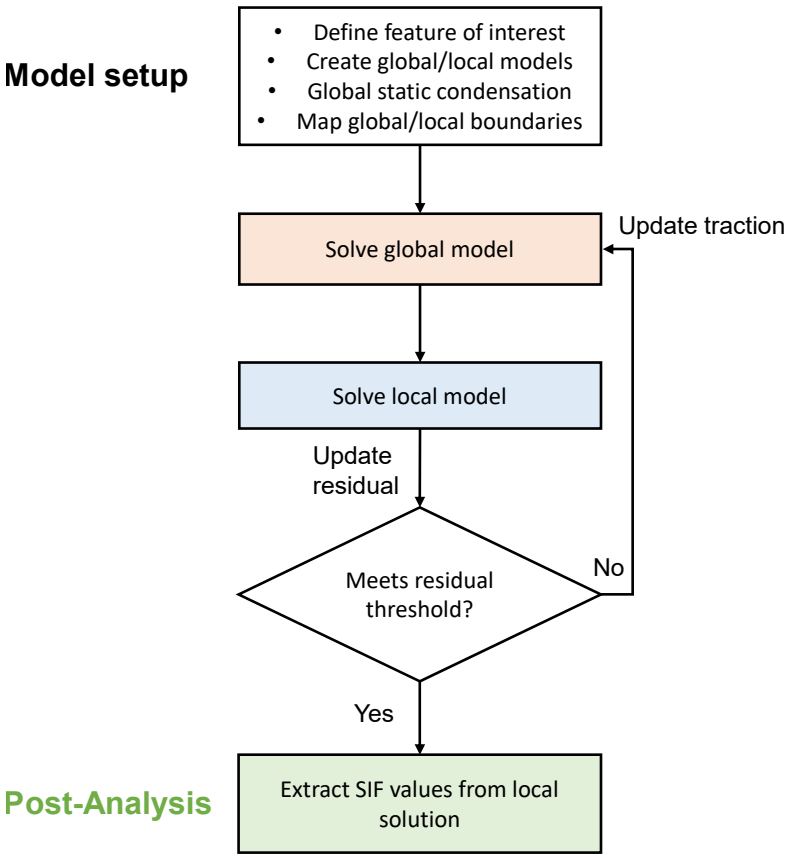



**Figure 2.2.** Iterative global local overview.

Previous IGL method work has looked at non-intrusively enhancing solid domains with XFEM cracks using XFEM/GFEM (Duarte, Hamzeh, Liszka and Tworzydlo, 2001a) (Moes, Dolbow and Belytschko, 1999) (Fillmore and Duarte, 2018a) (Gupta, Pereira, Kim, Duarte and Eason, 2012a). However, those methods each encountered limitations not affecting the IGL method. XFEM/GFEM crack modeling requires less particular local

mesh refinement than quarter node elements (Duarte, Hamzeh, Liszka and Tworzydlo, 2001a) (Moes, Dolbow and Belytschko, 1999) (Henshell and Shaw, 1975a) (Barsoum, 1976a), and therefore it is used in this work. Additionally, the IGL method has been used successfully to simulate nonlinearities in the local domain with a linear global domain (Gendre, Allix, Gosselet and Comte, 2009a).

The IGL method can be described as non-intrusive because of the ease with which research software may be combined with commercial software. This allows synergy between the robustness and broad applicability of the commercial software and the specificity of the research software. Also, the nonlinear case clearly lends itself to speed increases since Newton-Raphson iterations may be performed locally, reducing the problem size dramatically. Within the context of large structures typically modeled as shells, the IGL method has been successfully used to connect shell global domains to solid local domains with welds (Li, O'Hara and Duarte, 2021a). The IGL method has also been used to tie a shell aircraft geometry to a shell local domain with a sub-local solid domain tied into the local domain (Guinard, Bouclier, Toniolli and Passieux, 2018). This chapter describes the IGL methodology in Section 2.3. Finally, an alternative method to IGL for crack representation (i.e. a multigrid XFEM method) is proposed in (Passieux, Réthoré, Gravouil and Baietto, 2013).

The local cracked region in the miter gate will be modeled linearly using XFEM/GFEM, so the speed advantages of quarantining nonlinear regions to the local domain cannot be exploited. Therefore, the IGL method cannot be assumed to be faster than the reference tying method. To accelerate the local domain solution, this research proposes the novel modeling the IGL local domain using a surrogate model rather than a physics-based model. The surrogate model is trained on the local physics-based (crack) numerical results (not necessarily from a linear analysis). This surrogate model may then be used within the IGL framework to dramatically reduce local domain solution time. In fact, the non-intrusive nature of the IGL method (Allix and Gosselet, 2020) (Gendre, Allix, Gosselet and Comte,

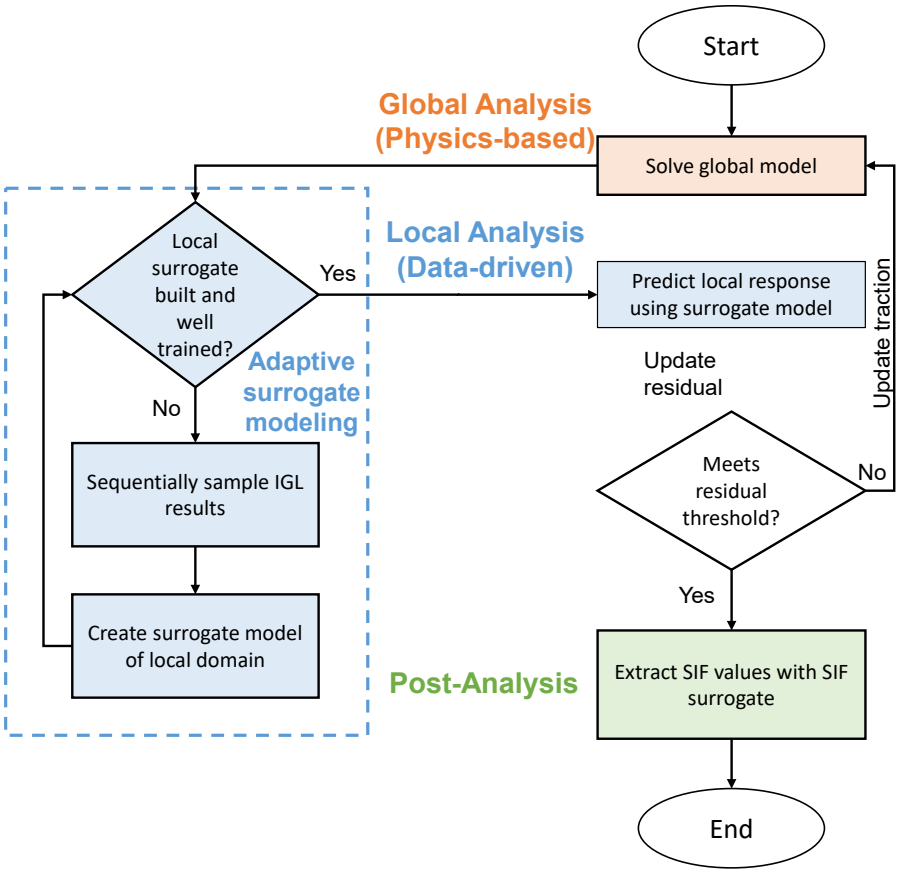



**Figure 2.3.** Surrogate iterative global local overview

2009a) (Gendre, Allix and Gosselet, 2011) (Gosselet, Blanchard, Allix and Guguin, 2018) (Li, O'Hara and Duarte, 2021a) facilitates easy implementation of the surrogate model.

Surrogate models, such as a Kriging method (Hu and Mahadevan, 2016; Li, Shen, Barzegar, Sadoughi, Hu and Laflamme, 2021c), neural networks (Li, Gao, Gu, Gong, Jing and Su, 2017b), and deep learning approaches (Chen, Chen, Zhou, Zhang and Yao, 2020), have been extensively studied in structural analysis and design optimization to reduce the required computational effort, especially in the presence of uncertainty (Hu and Mahadevan, 2017; Zhang, J. and Taflanidis , A.A., 2019). Various approaches have been proposed in the past decade to build an efficient yet accurate surrogate model (Sadoughi, Hu, MacKenzie, Eshghi and Lee, 2018; Li, O'Hara and Duarte, 2021a; Viana, Gogu and

Goel, 2021). Some multiscale frameworks simulate material-scale damage by using a surrogate model handle material properties and damage information (Yan, Zou, Ilkhani and Jones, 2020)(El Said and Hallett, 2018). To the best of our knowledge, however, surrogate modeling in an IGL framework has not been reported. This chapter describes a surrogate-based IGL methodology in Section 2.5 to fill this void.

Accelerating global domain linear solutions is somewhat easier using static condensation. Interactions between sub-regions to solve the aggregate problem can be accelerated using static condensation (Bjorstad and Widlund, 1986) (Gendre, Allix, Gosselet and Comte, 2009a) (Wyart, Duflot, Coulon, Martiny, Pardoen, Remacle and Lani, 2008). Within the IGL framework, this research utilizes static condensation to accelerate solution of the linear global problem, as discussed in Section 2.3.4. The use of a surrogate model for the local domain and static condensation for the global domain results will be referred to as the surrogate iterative global-local (SIGL) method for the rest of this chapter. SIGL has trivial computational time for each IGL method iteration, making the IGL method extremely fast, relatively speaking. The basic SIGL process is shown in Fig. 2.3

Four possible techniques have been mentioned to solve a problem in the class posed within this work: 1) reference tying method, 2) submodeling method, 3) IGL method, and 4) a proposed surrogate IGL method. The accuracy and speed of each of these approaches are shown and discussed in Section 2.6.

The remainder of this chapter is organized as follows: Section 2.3 presents the reference tying, submodeling method, and the IGL algorithm. Using the miter gate problem given in Section 2.4 as an example, Section 2.5 presents the proposed surrogate model-based IGL framework. Results comparing the different modeling methods is presented and discussed in Section 2.6. Section 2.7 gives the conclusions.

## 2.3 Modeling methodologies

### 2.3.1 Reference problem (shell-solid tying)

The reference problem to be solved is a shell geometry tied to a small solid geometry with a feature of interest and boundary conditions, as shown in Fig. 2.4. Large steel structures will have much larger shell domain $\Omega_{SH}$ compared to the solid domain $\Omega_S$. Also, body loads may be included although they are not shown in the figure. Commercial software such as Abaqus provide the tools to solve this problem for many different features of interest, including cracks. The tying method couples a solid surface to a shell edge where the shell normal is perpendicular to the solid surface normal. The constraints couple the displacement and rotation of each shell node to the average displacement and rotation of the solid surface near the shell node (Abaqus, 2021).

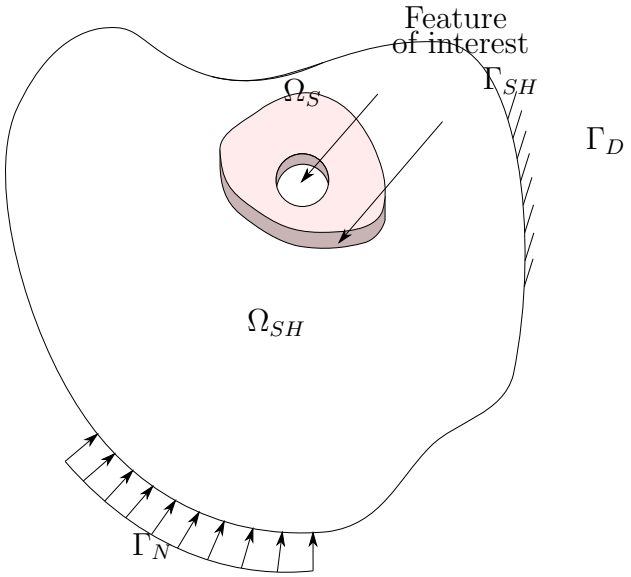



**Figure 2.4.** Reference problem with shell domain $\Omega_{SH}$, solid domain $\Omega_S$, shell-to-solid tied boundary $\Gamma_{SH}$, feature of interest, Neumann boundary condition $\Gamma_N$, and Dirichlet boundary condition $\Gamma_D$

### 2.3.2 Submodeling methodology

The submodeling method has a coarsely discretized global domain $\Omega_G$ and a finely discretized local domain $\Omega_{LSH} \cup \Omega_{LS} = \Omega_L$ containing the feature of interest within $\Omega_{LS}$ as shown in Fig. 2.5. The displacements and rotations are solved for in the global domain and then the displacements and rotations along $\Gamma_{GL-G}$ are applied to the local domain along $\Gamma_{GL-L}$. The solution of the local domain reflects the effects of the feature of interest. Due to the lack of any feedback mechanism to the global domain, this solution tends to underestimate the effects of the feature of interest and may not be sufficiently accurate. However, the numerical solution time is likely faster than the reference problem. If it is assumed that the number of flops is on the order of the number of degrees of freedom cubed $f_{ref} = O(n^3)$ due to factorization, then dividing $n$ into $n \approx n_G + n_L$, gives the submodeling number of flops as $f_{sub} = O(n_G^3) + O(n_L^3)$. Therefore $f_{ref} > f_{sub}$.

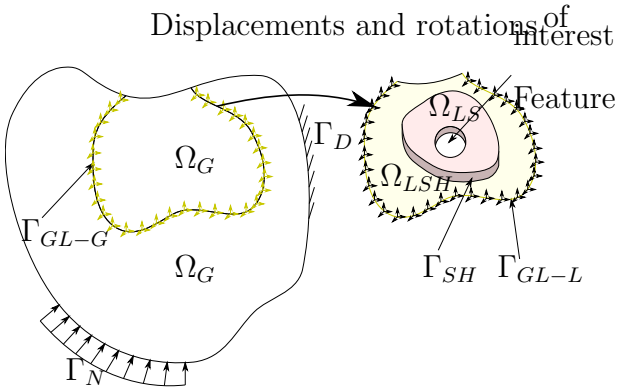



**Figure 2.5.** Submodeling problem with global domain $\Omega_G$, local shell domain $\Omega_{SH}$, local solid domain $\Omega_S$, and feature of interest.

As shown in Figure 2.5, the local domain $\Omega_L$ is subdivided into a solid local domain $\Omega_{LS}$ with the feature of interest and a shell local domain $\Omega_{LSH}$ to act as a buffer zone between the global discretization and local discretization. The two subdomains are tied along their shared boundary $\Gamma_{SH}$ using built-in Abaqus shell-to-solid tie constraints.

### 2.3.3 Iterative global-local methodology

The IGL method is a generalization of the zooming/submodeling method which incorporates a feedback loop into the global domain. This feedback loop improves accuracy but increases computational cost. The IGL method utilizes a local domain with local features of interest and fine discretization along with a global domain with a coarse discretization. The corresponding problem to Fig. 2.5 is shown in Fig. 2.6. The boundary between the global and local domains $\Gamma_{GL}$ facilitates exchange of displacement and reaction forces between the global and local problems. Note that the local domain utilizes the technique in (Guinard, Bouclier, Toniolli and Passieux, 2018) to facilitate shells in the global region and solids in the local region near the feature of interest.

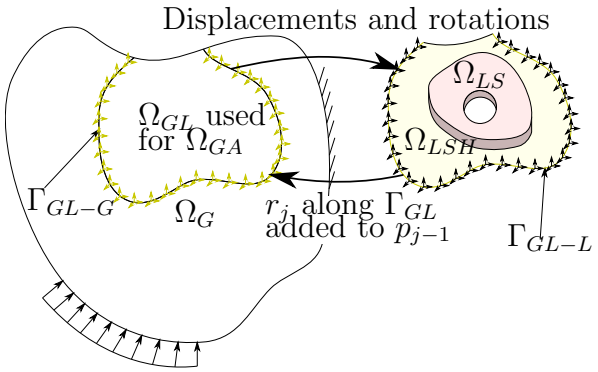



**Figure 2.6.** Iterative global-local algorithm illustrated using $\Omega_G$, $\Omega_{LSH}$, $\Omega_{LS}$, feature of interest, $\Gamma_N$, Dirichlet boundary condition $\Gamma_D$, $\Gamma_{GL-G}$, $\Gamma_{GL-L}$, and $\Gamma_{SH}$.

The IGL algorithm is given in Algorithm 1 where $\mathbf{p}_j$ is immersed surface force at $\Gamma_{GL-G}$, $\omega_j$ is relaxation parameter that accelerates convergence, $\mathbf{u}_j^G$ is global displacement at $\Gamma_{GL-G}$, $\boldsymbol{\lambda}_j^L$ is local reaction at $\Gamma_{GL-L}$, and $\boldsymbol{\lambda}_j^L$ is the auxiliary reaction at $\Gamma_{GL-A}$ shown in Fig. 2.7. Algorithm 1 expounds upon the values exchanged between $\Omega_G$ and $\Omega_L$. It is generally accepted that Aitken's Delta-Squared method provides robust convergence for this algorithm (Allix and Gosselet, 2020)(Duval, Passieux, Salaun and Guinard, 2016)(Gosselet, Blanchard, Allix and Guguin, 2018)(Liu, Sun and Fan, 2014).

This algorithm shows that part of the first iteration of IGL constitutes the zoom-

**Algorithm 1.** Iterative global-local fixed point iteration algorithm with Aitken's delta-squared method (Gosselet, Blanchard, Allix and Guguin, 2018)

**procedure** IGL(tolerance,$m$,$\mathbf{f}^G$,$\mathbf{f}^L$) ▷ $\mathbf{f}^G$ and $\mathbf{f}^L$ are glob. and loc. load vectors
  Arbitrary initialization $\mathbf{p}_0$
  Arbitrary initialization $\omega_0 \approx 1.0$
  **for** $j \in [0, ..., m]$ **do**
    $\mathbf{u}_j^G = \text{SolveGlobal}(\mathbf{p}_j; \mathbf{f}^G)$
    $\boldsymbol{\lambda}_j^L = \text{SolveLocal}(\mathbf{u}_j^G; \mathbf{f}^L)$
    $\boldsymbol{\lambda}_j^{GA} = \text{SolveAux}(\mathbf{u}_j^G; \mathbf{f}^{GA})$
    $\mathbf{r}_j = -\left(\boldsymbol{\lambda}_j^L + \mathbf{p}_j - \boldsymbol{\lambda}_j^{GA}\right)$
    $e_j = ||r_j||_\infty$
    **if** $e_j <$ tolerance **then**
      exit for loop
    **end if**
    $\mathbf{p}_{j+1} = \mathbf{p}_j + \mathbf{r}_j$
    Ait. $\Delta^2$: $\omega_{j+1} = -\omega_j \frac{r_{j-1} \cdot (r_j - r_{j-1})}{(r_j - r_{j-1}) \cdot (r_j - r_{j-1})}$
    $\mathbf{p}_{j+1} = \omega_{j+1} \mathbf{p}_{j+1} + (1 - \omega_{j+1})\, \mathbf{p}_j$
  **end for**
**end procedure**

ing/submodeling method. To compare accuracy between the two and the reference solution, values specific to the feature of interest will be used, specifically for cracks stress intensity factors (SIF).

For a linear problem IGL may be faster than the reference problem under ideal conditions, e.g. the commercial software can save the factorized matrix. The speed increase depends on the number of iterations $i$ and $n_G$, $n_L$, and $n$. Considering factorization and forward and backward substitution since it may be significant for iterations within IGL, $f_{IGL} = O(n_L^3 + n_G^3 + i \times (n_L^2 + n_G^2)$. Directly comparing this with the tying method $f_{ref} = O((n_L + n_G)^3 + (n_L + n_G)^2)$ one can see the rather precarious situations under which IGL may be faster than the tying method. Now, assume that these estimates on the order of solution perfectly represent solution time. We take that IGL solution time must be less than the tying, $n_L^3 + n_G^3 + i \times (n_L^2 + n_G^2) < n_L^3 + 3n_L^2 n_G + 3n_L n_G^2 + n_G^3 + n_L^2 + 2n_L n_G + n_G^2$.

Solving this for $i$ gives

$$i < \frac{3n_L^2 n_G + 3n_L n_G^2}{n_L^2 + n_G^2} + 1. \tag{2.1}$$

If either $n_L$ or $n_G$ is much greater than the other, $i < 4$ for faster IGL solution of the system of equations. If $n_L$ approaches 0 (local problem disappears) $i < 1$. This motivates the numerical context within which IGL is useful: the global domain is so large that the local domain likely requires immense detail for the feature of interest. Now, the global maximum for $m$ is along the line $n_L = n_G$ which gives $i < 6n_L + 1$ iterations. Now this is not exact arithmetic on the time to solution of the system, but demonstrates the likely speed advantage of IGL for linear problems. In this research such ideal conditions are achieved using static condensation, which has the added benefit of reducing the degrees of freedom in the solve for the global system of equations.

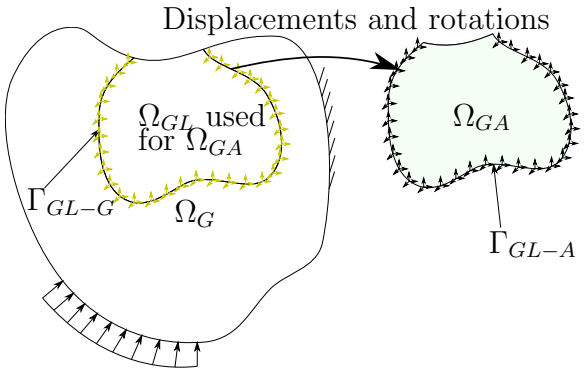



**Figure 2.7.** Iterative global-local algorithm auxiliary domain $\Omega_{GA}$ used for the calculation of global traction

### 2.3.4 Static condensation of global domain in IGL method

The IGL algorithm provides clear computational benefits with localized nonlinearity, since Newton-Raphson iterations need to be performed on only the much smaller local domain. However, this research shows that the IGL method may be much slower than the tying method in a linear local problem. In this research the XFEM crack local problem is linear and computationally expensive, which makes IGL possibly slower than the tying

method. In an attempt to accelerate the IGL method, static condensation can be applied to both the global and local stiffness matrices since both are linear. Statically condensing $\Omega_G$ requires leaving the degrees of freedom at the nodes along $\Gamma_{GL}$ uncondensed. Then global Neumann boundary conditions can be applied at those degrees of freedom as well as the immersed surface force at each iteration $p_j$. Then, the global displacement along $\Gamma_{GL}$ ($\boldsymbol{u}_j^G$) is obtained directly from the condensed matrix.

In the example problem presented in this research, there may be damage in one boundary condition region. When damage is not present, a pin boundary condition is applied; when damage is present, the pin boundary condition is removed. This is compatible with static condensation by leaving all nodes along the boundary condition uncondensed and applying the pins on the static condensation system of equations.

While static condensation demands a large upfront cost, the speed improvement comes with the many IGL iterations performed over the many permutations of a Monte Carlo analysis or training process. However, applying static condensation to the local domain stiffness matrix has some caveats. First, the used commercial software does not support static condensation with XFEM, although it is theoretically possible. Second, a unique static condensation must be computed for each crack length, which may be faster than the reference solution.

Since the static condensation of a global stiffness matrix can consider different load cases and levels of damage along the boundary condition, only one global static condensation step is necessary per local domain. Then the same reduced stiffness matrix can be used over the permutations of load cases, damaged boundary conditions, and IGL iterations. Thereby the static condensation of the FEM discretization of the global domain saves computational cost. However, it is shown in Section 2.6 that the IGL solution of a problem with only one permutation of the load cases and damaged boundary conditions saves computational effort.

## 2.4 Problem definition: application to a miter gate

### 2.4.1 Miter gate operation, load state, and feature of interest

Miter gates are navigational hydraulic steel structures critical to river traffic. They function as boat "elevators" that allow boats to bypass dams and navigate up or down river. Figure 2.8 shows how miter gates open and close. The gudgeon and pintle (seen in Fig. 2.9) form a hinge about which the gate rotates. More detailed information about pintle behavior can be found in (Fillmore and Smith, 2021a). When open, boats can enter or leave the lock chamber. When closed, the lock chamber can be filled or emptied (on the upstream side) while the miter gate acts as a damming surface. The resulting hydrostatic pressure pushes the two leaves together along their miter and pushes each leaf into the wall along the quoin. More detailed information about quoin behavior can be found in (Eick, Fillmore and Smith, 2019a).

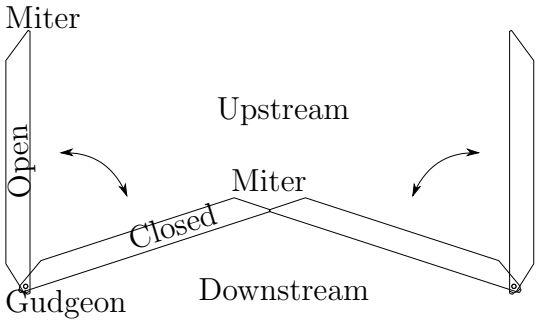



**Figure 2.8.** Miter gate top view with swinging motion

Miter gates' largest cyclic loads are from the filling and emptying of lock chambers as boats are lifted or lowered. The resulting cyclic stresses contribute to fatigue cracking. Miter gates are welded structures, so the heat-affected zones greatly accentuate the cyclic stresses. However, in this example, a region of the leaf is selected that naturally experiences tension to reduce complexity resulting from weld residual stresses. This portion of the leaf is near the bottom center. If each leaf is viewed as a beam (Fig. 2.8) with distributed load,

the greatest tension in the leaf will occur on the downstream side at the middle of the leaf.

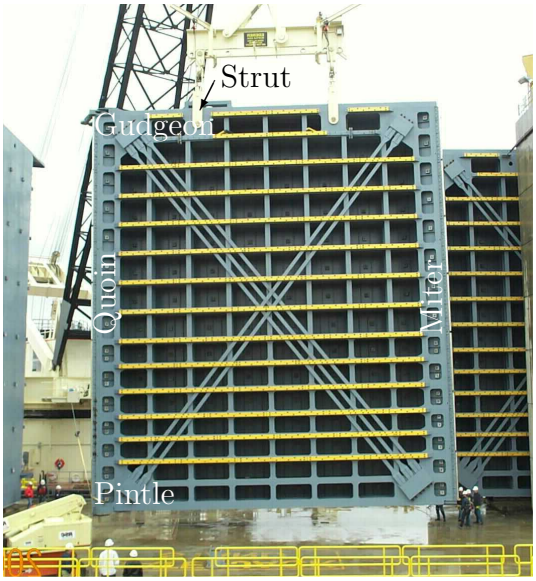



**Figure 2.9.** Miter gate downstream side view. Photograph courtesy of John Cheek, USACE.

A finite element model representing the Greenup downstream miter gate is shown in Fig. 2.10. The boundary conditions of the miter gate are set up to simulate the in-situ environment of a hanging gate. The miter gate rotates around the axis created by the anchorage pin and pintle as shown in Fig. 2.8. The pintle, a ball and socket joint, takes all of the vertical gravity load. The pintle is represented by applying a multi-point constrant (MPC) from the center of the ball to the portions of the horizontal girder with which the socket connects. Then, the center of the ball is restrained from translating in the $x$, $y$, and $z$-directions. The anchorage links are embedded in concrete at the top of the gate. This is represented by restraining translation in the $x$, $y$, and $z$-directions.

The strut pin is attached to a strut arm that opens and closes the gate. The strut pin can rotate around the z-axis. When the gate is closed, the strut arm applies resistance at $-43^o$ from the negative $x$-direction on the gate. The strut pin is modeled by applying an MPC from the top of the strut pin to the enveloping top lug and a separate MPC from the bottom of the strut pin to the enveloping bottom lug., Then the center of the strut

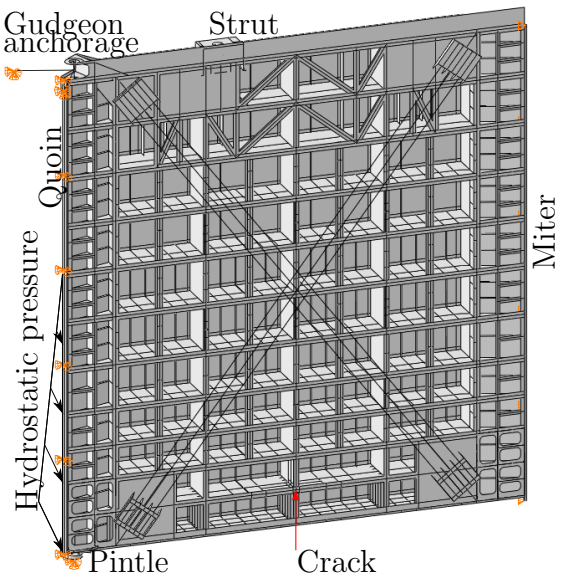



**Figure 2.10.** Miter gate boundary conditions

pin is restrained from translating $-43^o$ from the negative $x$-direction.

Hydrostatic pressure is applied on the upstream plate of the gate, called the skin plate as shown in Fig. 2.11. The upstream hydrostatic pressure is denoted $h_{up}$ and the downstream hydrostatic pressure is denoted $h_{down}$. When the gate holds enough water in the lock chamber, the miter contact block of both gate leaves come into contact and a symmetric pin is assumed preventing translational movement $-18^o$ from the x-direction. The two gate leaves act as an arch, experiencing more axial compression under more hydraulic head. This compression causes the gate to thrust in the lock wall contact block. The wall resists horizontal movement in the $x$ and $y$-directions, which is represented in the model with pins that resist translation in the $x$ and $y$-directions.

The contact of the quoin contact block with the wall is idealized using pin boundary conditions as shown in Fig. 2.12. Often, the bottom portion of the quoin becomes damaged so that it cannot properly contact the wall. This lack of contact is idealized by not applying the pin boundary conditions. The length of this damaged region is denoted $l_{dmg}$.

Figure 2.13 shows the reference discretization with a zoom-in of the crack region.

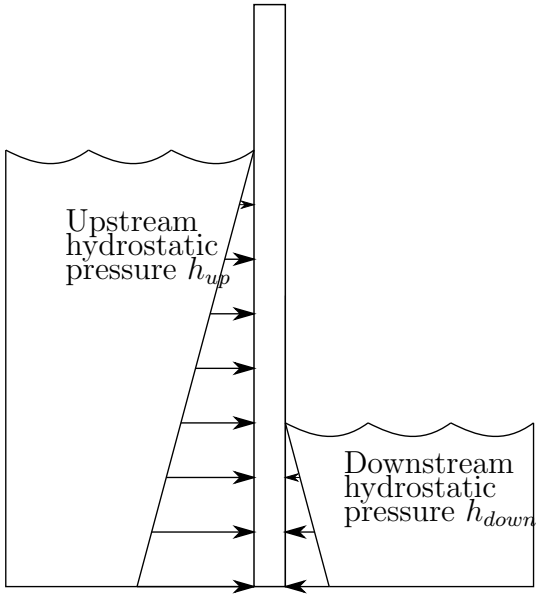



**Figure 2.11.** Miter gate hydrostatic pressure from upstream and downstream water levels

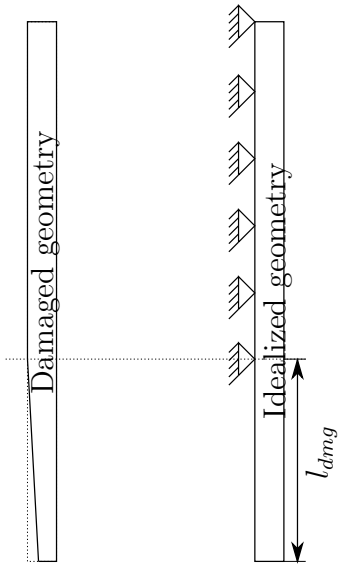



**Figure 2.12.** Damage in the quoin contact block

The shell elements used over much of the gate are reduced-integration quadrilaterals with element size six inches. Where the crack is defined linear hexahedral elements are used with element size 0.0625 in. The mesh discretization has $201,463$ elements and $211,372$

nodes. The IGL discretization is effectively identical to the reference, with an identical mesh discretization in $\Omega_G$ and $\Omega_L$.

Figure 2.13 also shows the location of the crack used for this example. The crack occurs along the bottom web edge of the second from bottom girder as shown in Fig. 2.13. The crack has a straight front, extending through the entire 3/4 in thickness of the plate. The crack length is variable, but the largest possible length through the web bottom is 4 in.

Miter gates are fabricated by welding mild steel plates together. The local weld geometries are ignored in this research. A linear material model is used with the Young's modulus as $E = 29,000$ ksi and the Poisson ratio as 0.3.

Figure 2.14, Figure 2.15, Figure 2.16, and Figure 2.17 help clarify how IGL is used in this example. Figure 2.14 shows the global domain along with the immersed surface force $p_j$ resulting from the IGL algorithm. The global domain does not contain the crack and is only coarsely discretized in the crack's coordinates $\Omega_{GA}$. Outside of $\Omega_{GA}$, the global domain's geometry and discretization matches up exactly with the reference model.

The feature of interest is the crack, which is only explicitly represented in the local model in Fig. 2.15. The geometry and discretization of the shell domain $\Omega_{LSH}$ and the solid domain $\Omega_S$ line up exactly with the reference model in the corresponding region. Also, the nodes of the FEM global mesh along $\Gamma_{GL}$ line up with the nodes of the FEM local mesh along $\Gamma_{GL-L}$ exactly. Because shell elements are used, such matching meshes along the 1-dimensional interface are easy to produce in Abaqus. Displacements and rotations from the global model at the IGL step are applied along $\Gamma_{GL-L}$.

Figure 2.16 shows the global auxiliary domain. This domain matches the $\Omega_{GA}$ in Fig. 2.16 exactly, i.e. there is no crack and has the same mesh discretization. When the displacements and rotations from the global solution are applied along $\Gamma_{GL}$, this domain helps to calculate the reaction forces of $\Omega_{GA}$ easily, particularly when sophisticated post-processing capabilities are not available.

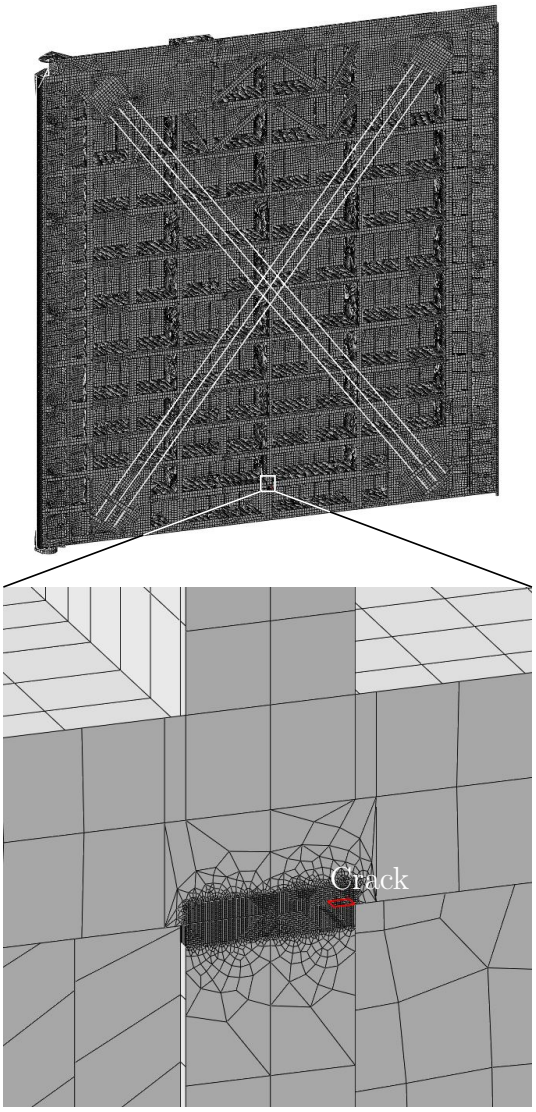



**Figure 2.13.** Mesh discretization of reference miter gate model

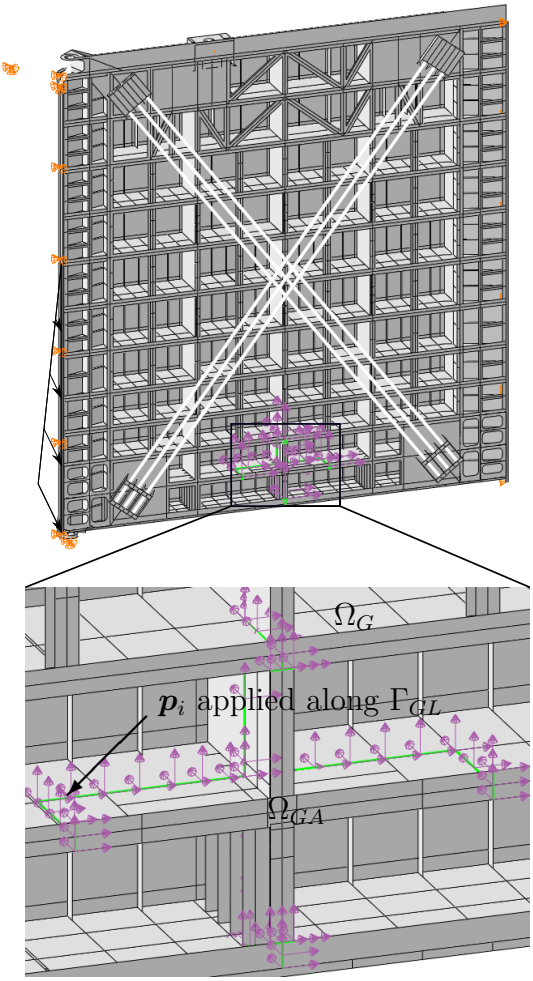



**Figure 2.14.** IGL global miter gate model with zoom-in of area of interest. No crack is included in the area of interest, but the shown purple arrows along boundary $\Gamma_{GL}$ are the $p_j$ forces that relay the effects to the global model.

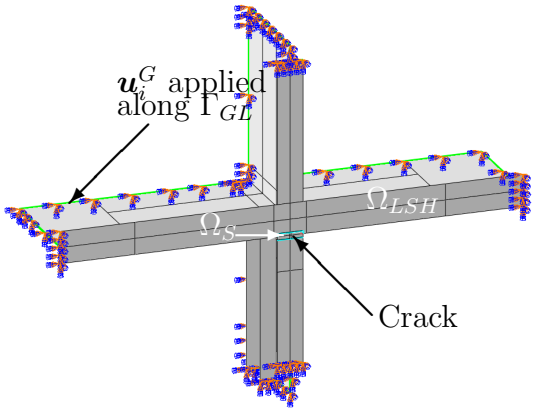



**Figure 2.15.** Local miter gate model with contour integral crack representation. The crack is located in the solid subdomain $\Omega_S$. The global displacement solution $u_j^G$ is applied along the global-local boundary $\Gamma_{GL}$.

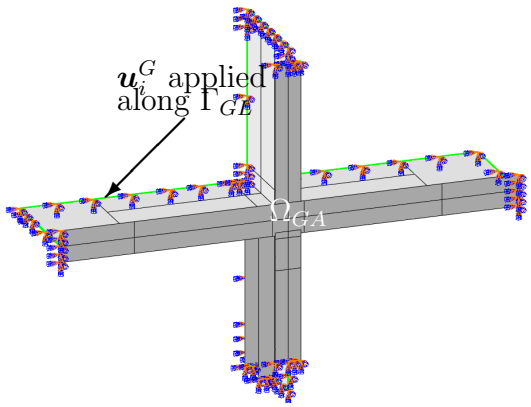



**Figure 2.16.** Global auxiliary miter gate model. The global displacement solution $u_j^G$ is applied along the global-local boundary $\Gamma_{GL}$.

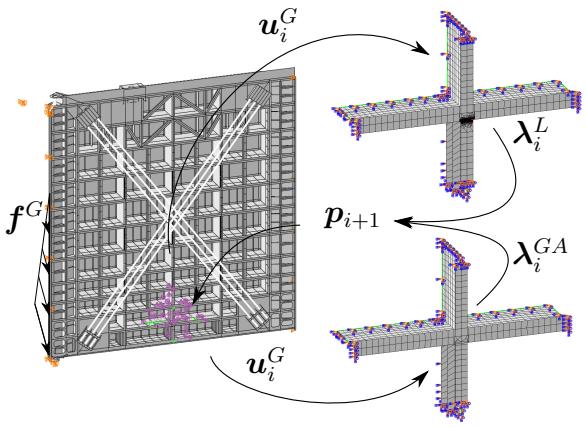



**Figure 2.17.** Illustrated IGL fixed point iteration algorithm with Aitken's Delta-Squared method for miter gate with global, local, and global auxiliary mesh discretizations. The global domain has $l_{dmg}$, $h_{up}$, and $h_{down}$ parameters. The local domain has parameter $a$.

The IGL fixed point iteration algorithm with Aitken's Delta-Squared method is shown in the context of the cracked miter gate in Fig. 2.17. The boundary conditions for the global domain include damaged gap length $l_{dmg}$, and upstream $h_{up}$ and downstream $h_{down}$ water heights that result in $\boldsymbol{f}^G$. The local domain has a certain crack length $a$. For the first iteration of IGL or the submodeling method, $p_j = 0$. The resulting displacements and rotations along $\Gamma_{GL}$ are applied to the local and auxiliary global models. These models give the local reactions and global reactions respectively. The residual between them is found. The local model and global auxiliary model also have the displacements from the global model solution $u_j^G$ applied along $\Gamma_{GL}$.

### 2.4.2 Calculating the stress intensity factor

The stress intensity factor is calculated using built-in Abaqus technology. There are 13 nodes through the thickness of the cracked plate, and through the rest of this chapter the middle node will be considered. Four contours are generated per node, and the first SIF mode, $K_1$, is recorded. The Abaqus default contour integral method, the line integral method is used. In order to measure error of the methods considered in this research, the SIF relative error with the reference solution is calculated as $e_K = \frac{\|K_{1-ref} - K_{1-IGL}\|}{K_{1-ref}}$, where

$K_{1-ref}$ is the SIF value extracted from the reference model and $K_{1-IGL}$ is from IGL.

## 2.5 Surrogate iterative global-local methodology

The aforementioned IGL algorithm is computationally expensive for probabilistic analysis (e.g. reliability analysis, uncertainty quantification, model updating), since the model needs to be executed thousands of times. A straightforward way to overcome the computational challenge of the IGL method is to directly build a surrogate model for the IGL model as a whole by treating the model as a black box. The direct surrogate modeling method, however, has the following three major drawbacks:

1. Whenever there is a change in the global model, the original surrogate model will become inapplicable and needs to to be retrained;

2. As a purely data-driven approach, the direct surrogate modeling method does not preserve the physical information in the global model;

3. The training time is much longer. Generating training samples for the direct surrogate modeling method requires full runs of the IGL, which itself requires a number of iterations to converge.

This section proposes a hybrid surrogate modeling method to tackle the computational challenge in the IGL method and overcome the limitations of the direct surrogate modeling method. In the proposed method, the FE process of global domain is kept to capture the physical response of the global domain under several boundary conditions, while the non-linear behavior of the local domain is modeled using Gaussian Process Regression (GPR)(Santner, Williams, Notz and Williams, 2003)(Williams and Rasmussen, 2006). As for this chapter, the GP package from scikit-learn is used to build the surrogate models (Pedregosa, Varoquaux, Gramfort, Michel, Thirion, Grisel, Blondel, Prettenhofer, Weiss and Dubourg, 2011). Because that the local FEM process was used repeated in IGL

iterations, the performance of the surrogate model must be carefully calibrated without introducing additional system error. The proposed method consists of two main steps, namely (1) surrogate modeling and refinement in the local domain; and (2) integration of physics-based global model and data-driven local model for IGL implementation. In what follows, more details are provided about the proposed surrogate-based iterative global-local methodology.

### 2.5.1 Surrogate modeling in the local domain

As shown in Fig. 2.17, the input of the local-domain FE model consists of the displacements $\boldsymbol{u}^G$ from the global model solution and given crack length $a$ imposed to the local domain. The output, correspondingly, is composed of the reaction forces $\boldsymbol{\lambda}^L$ from the local model solution. The goal of surrogate modeling in the local domain is to efficiently map $\boldsymbol{u}^G$ and $a$ to $\boldsymbol{\lambda}^L$ using surrogates without solving the computationally expensive local FE model repeatedly.

**Training data collection**

The local boundary condition solved from the global model is dominated by the hydrostatic pressure and the quoin block damage. Meanwhile, the crack length determines the corresponding response from the local model. Overall, the physics of the whole structural system in this case is affected by four parameters, i.e., $h_{up}$, $h_{down}$, $l_{dmg}$, and $a$. Different combinations of such parameters induce different physical behaviors of local model, leading to input-output (IO) relations for the surrogate models. Directly building and training surrogate models can be time-intensive as such high-dimensional space is hard to be sufficiently sampled. To overcome this challenge, we first generate $N$ samples in the 4-D space constructed by $h_{up}$, $h_{down}$, $l_{dmg}$, and $a$ using the Latin hyper-cube sampling method. In this study, 400 samples are firstly generated with As shown in Fig. 2.18, assume that the IGL algorithm needs $n_i$ iterations to converge for the $i$-th sample,

$\forall i = 1,\ 2,\ \cdots,\ N$. The intermediate training data can be denoted as $\boldsymbol{u}_i^G \in \mathbb{R}^{(n_i \times M_{DOF})}$ and $\boldsymbol{\lambda}_i^L \in \mathbb{R}^{(n_i \times M_{DOF})}$, where $M_{DOF}$ is the total DOFs of the local boundary $\Gamma_{GL}$. In this example, 200 samples are generated with the parameter ranges as: $h_{up} \sim [432, 720]$in, $h_{down} \sim [120, 360]$in, $l_{dmg} \sim [0, 150]$in, and $a \sim [0.5, 4]$in, where $[lb, ub]$ represents variation lower bound $lb$ and upper bound $ub$.

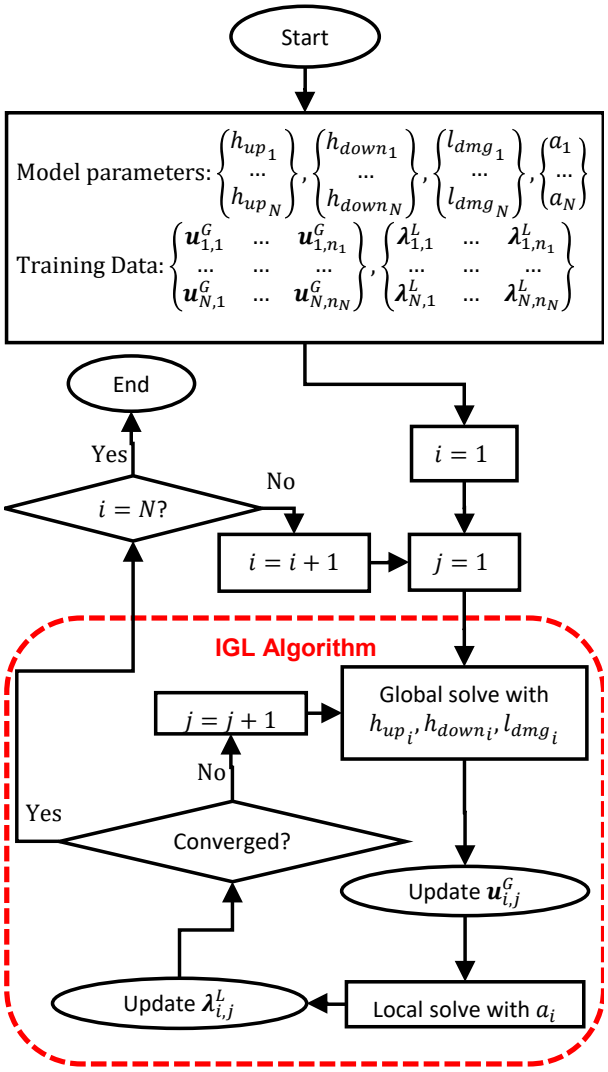



**Figure 2.18.** Collecting $\boldsymbol{u}^G$, $\boldsymbol{\lambda}^L$, and $SIF$ training data from the physics model.

Let the total number of data collected for $\boldsymbol{u}^G$ and $\boldsymbol{\lambda}^L$ be $N_T$ (i.e. $N_T = \sum_{i=1}^{N} n_i$), we then have training data from the $N$ simulations as

$$
\begin{aligned}
\mathbf{X} &= (\underline{\boldsymbol{u}}^G, \mathbf{a}) \\
&= [(\boldsymbol{u}_1^G, a_1), (\boldsymbol{u}_2^G, a_2), ..., (\boldsymbol{u}_{N_T}^G, a_{N_T})] \\
&\in \mathbb{R}^{(N_T \times (M_{DOF}+1))}, \\
\mathbf{Y} &= \underline{\boldsymbol{\lambda}}^L = [\boldsymbol{\lambda}_1^L, \boldsymbol{\lambda}_2^L, ...\boldsymbol{\lambda}_{N_T}^L] \in \mathbb{R}^{(N_T \times M_{DOF})},
\end{aligned}
\tag{2.2}
$$

where $M_{DOF}$ is the total DOFs of the local boundary $\Gamma_{GL}$. Note that the data are organized in rows, i.e., the total number of rows represents the length of the data and the total number of columns represents the dimension of the data.

**Data compression and latent space representation**

In general, as the input dimension of the surrogate model increases, the training data required to fully characterize the IO relationship grows exponentially. According to the collected training data in the case of miter gate, the local boundary $\Gamma_{GL}$ contains $120 \times 6$ DOFs ($M_{DOF} = 720$) resulting in a 721-dimensional input and a 720-dimensional output. The high-dimensional input and output make the construction of accurate surrogate models in the local domain very challenging. Thus, instead of directly building surrogate models for $\boldsymbol{u}^G$ and $\boldsymbol{\lambda}^L$, dimension reduction method is necessary to map the IO relationship into a low-dimensional latent space. Numerous contributions have been made to compress dataset from higher dimensional matrix to lower dimensional matrix with various dimension reduction techniques, such as singular value decomposition, independent component analysis, auto-encoder, etc. (Fodor, 2002; Vega, Hu, Yang, Chadha and Todd, 2021a). Considering its computational cost as well as stability, SVD is adopted in this chapter. However, compression in the developed approach is not limited to SVD, but can be accomplished with other dimension reduction techniques as well.

In SVD, the data collected in Eq. (2.2) is decomposed as

$$\begin{aligned} \underline{\boldsymbol{u}}^G &= \boldsymbol{W}_u \boldsymbol{E}_u \boldsymbol{V}_u^T, \\ \underline{\boldsymbol{\lambda}}^L &= \boldsymbol{W}_\lambda \boldsymbol{E}_\lambda \boldsymbol{V}_\lambda^T, \end{aligned} \tag{2.3}$$

where $\boldsymbol{W}_u, \boldsymbol{W}_\lambda \in \mathbb{R}^{(N_T \times N_T)}$ and $\boldsymbol{V}_u^T, \boldsymbol{V}_\lambda^T \in \mathbb{R}^{(M_{DOF} \times M_{DOF})}$are orthogonal matrices, and $\boldsymbol{E}_u, \boldsymbol{E}_\lambda \in \mathbb{R}^{(N_T \times M_{DOF})}$ are rectangular diagonal matrices. Note that the crack length in the input data is not compressed with the whole matrix due to its significance.

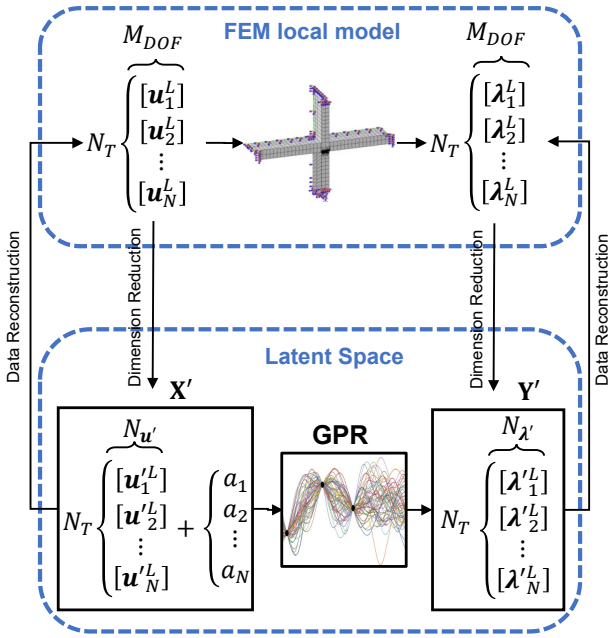



**Figure 2.19.** Dimension reduction strategy of the proposed method compared with IO of FEM local model.

After the decomposition given in Eq. (2.3), a low-rank matrix approximation can be further determined, namely

$$\begin{aligned} \underline{\tilde{\boldsymbol{u}}}'^G &= \boldsymbol{W}_u \tilde{\boldsymbol{E}}_u \boldsymbol{V}_u^T, \\ \underline{\tilde{\boldsymbol{\lambda}}}'^L &= \boldsymbol{W}_\lambda \tilde{\boldsymbol{E}}_\lambda \boldsymbol{V}_\lambda^T, \end{aligned} \tag{2.4}$$

where $\tilde{\boldsymbol{E}}_u \in \mathbb{R}^{(N_T \times N'_u)}$ and $\tilde{\boldsymbol{E}}_\lambda \in \mathbb{R}^{(N_T \times N'_\lambda)}$ are the same matrices as $\boldsymbol{E}_u, \boldsymbol{E}_\lambda$ except that they contain only $N'_u$ and $N'_\lambda$ largest singular values, respectively (the other singular values are replaced by zero). Figure 2.20 illustrates how the important features of the data can be represented by a low-rank matrix from the SVD.

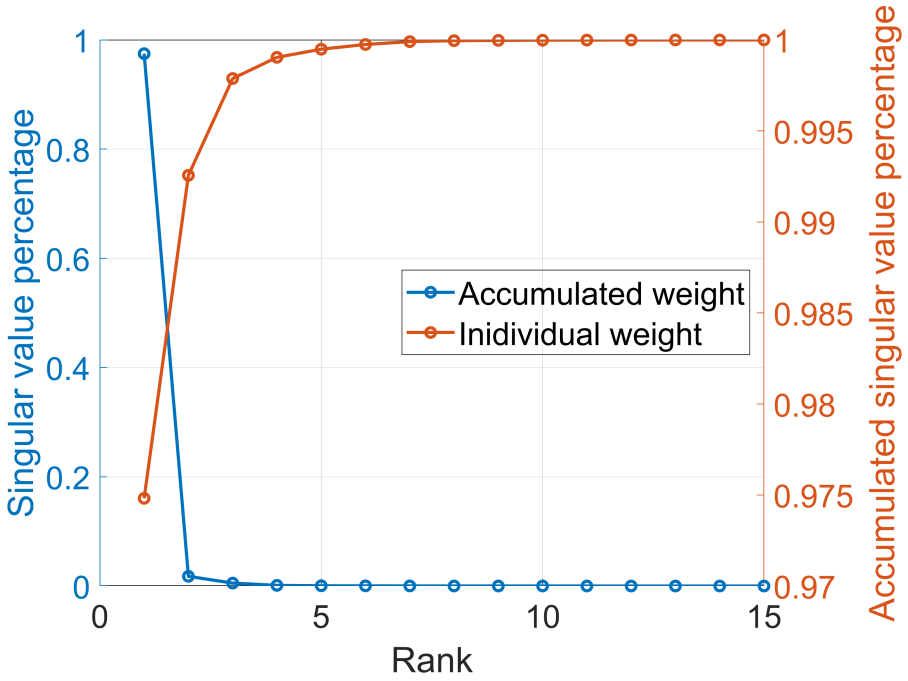



**Figure 2.20.** Illustration of the importance values of different features in the matrix represented by the singular values.

As shown in Fig. 2.19, the corresponding reduced-order the displacement and reaction force, denoted $\boldsymbol{u}'^G$ and $\boldsymbol{\lambda}'^L$, can be represented by truncating the orthogonal matrices $\boldsymbol{W}_u, \boldsymbol{W}_\lambda$ based on their ranks,

$$\begin{aligned} \boldsymbol{W}_u \in \mathbb{R}^{(N_T \times N_T)} \rightarrow \boldsymbol{W'}_u \in \mathbb{R}^{(N_T \times N'_u)} \rightarrow \boldsymbol{u}'^G, \\ \boldsymbol{W}_\lambda \in \mathbb{R}^{(N_T \times N_T)} \rightarrow \boldsymbol{W'}_\lambda \in \mathbb{R}^{(N_T \times N'_\lambda)} \rightarrow \boldsymbol{\lambda}'^L, \end{aligned} \tag{2.5}$$

where $N_{u'}$ and $N_{\lambda'}$ are the dimensions of $\boldsymbol{u}'^G$ and $\boldsymbol{\lambda}'^L$ after reduction. The decoders and encoders are defined as the matrices that allows the data to transform between

low-dimensional and high-dimensional spaces through matrix multiplication. Written explicitly,

$$
\begin{aligned}
&\text{Decoder}_\lambda = \boldsymbol{E'}_\lambda \boldsymbol{V'}_\lambda^T, \\
&\text{Encoder}_u = (\boldsymbol{E'}_u \boldsymbol{V'}_u^T)^\dagger, \\
&\boldsymbol{u}^G = \boldsymbol{u'}^G \times \text{Decoder}_u, \\
&\boldsymbol{\lambda}^L = \boldsymbol{\lambda'}^L \times \text{Decoder}_\lambda, \\
&\boldsymbol{u'}^G = \boldsymbol{u}^G \times \text{Encoder}_u, \\
&\boldsymbol{\lambda'}^L = \boldsymbol{\lambda}^L \times \text{Encoder}_\lambda,
\end{aligned}
\tag{2.6}
$$

where $(\boldsymbol{E'}_u \boldsymbol{V'}_u^T)^\dagger$ is the (Moore-Penrose) pseudo inverse (Moore, 1920) of $\boldsymbol{E'}_u \boldsymbol{V'}_u^T$, which extends matrices inversion to non-square matrices (the decoders are non-square matrices in most cases).

The latent space of the surrogate model now can be presented as

$$
\begin{aligned}
\mathbf{X'} &= \underline{\boldsymbol{u}}'^G \\
&= [(\boldsymbol{u}_1'^G, a_1), (\boldsymbol{u}_2'^G, a_2), ..., (\boldsymbol{u}_{N_T}'^G, a_{N_T})] \\
&\in \mathbb{R}^{(N_T \times (N'_u + 1))}, \\
\mathbf{Y'} &= \underline{\boldsymbol{\lambda}}'^L = [\boldsymbol{\lambda}_1'^L, \boldsymbol{\lambda}_2'^L, ..., \boldsymbol{\lambda}_{N_T}'^L] \in \mathbb{R}^{(N_T \times N'_\lambda)}.
\end{aligned}
\tag{2.7}
$$

In this example, the 720-DOF displacement vector $\boldsymbol{u}^G$ is compressed into a 4-dimensional vector ($N_{u'} = 4$), which forms a 5-dimensional input combining with crack length parameter in the latent space. Similarly, the 720-DOF reaction force $\boldsymbol{\lambda'}^L$ is compressed into a 4-dimensional output ($N_{\lambda'} = 4$) in latent space. The GPR-based surrogate models are then built and trained in the designed latent space with training

samples $\mathbf{X}'$ and $\mathbf{Y}'$. Since $\mathbf{Y}' \in \mathbb{R}^{(N_T \times N'_\lambda)}$, GPR surrogate models are constructed for each dimension of $\mathbf{Y}'$ as follows

$$Y'_i = \hat{G}_i(\mathbf{X}'),\ i = 1,\ \ldots,\ N'_\lambda, \tag{2.8}$$

where $\hat{G}_i(\cdot), \forall i = 1,\ \ldots,\ N'_\lambda$ is the $i$-th GPR surrogate model. Note that the surrogate modeling in this chapter is not limited to GPR. Because GPR can be computationally inefficient when handling high-dimensional data, the GPR can be replaced by a neural network architecture or other deep learning methods in a case that low-dimensional data can not be accurately generated.

For any given value of $\mathbf{X}'$, we have the prediction from the $i$-th GPR surrogate model as follows

$$\hat{G}_i(\mathbf{X}') \sim N(\mu_{Y_i},\ \sigma^2_{Y_i}), \forall i = 1,\ \ldots,\ N'_\lambda \tag{2.9}$$

in which $N(\cdot)$ is Gaussian distribution, $\mu_{Y_i}$ and $\sigma_{Y_i}$ are respectively the mean and standard deviation of the prediction.

Due to the imbalance of the initial training data collected in Sec. 2.5.1, the GPR surrogate models given in Eq. (2.8) may not accurately represent the original local-domain FE model. Using the GPR surrogate models to replace the original local-domain model in the global-local iterative scheme will lead to large prediction errors due to error accumulation over iterations. To overcome this issue in surrogate model-based IGL algorithm, we present a framework to refine the local-domain surrogate models in the subsequent section.

### 2.5.2 Refinement of local-domain surrogate models

An important issue in surrogate modeling is how to achieve a good accuracy with a reasonable number of sample points in the latent space. Due to the error accumulation over iterations as mentioned above, the performance of the surrogate model that replaces

the FE process must be carefully calibrated in order to avoid additional system error. As an example shown in Fig. 2.21, an input that locates in the area with sufficient training points (i.e., well-trained area) will result a low model error when passing the GPR model, while poor-trained area will result a high model error.

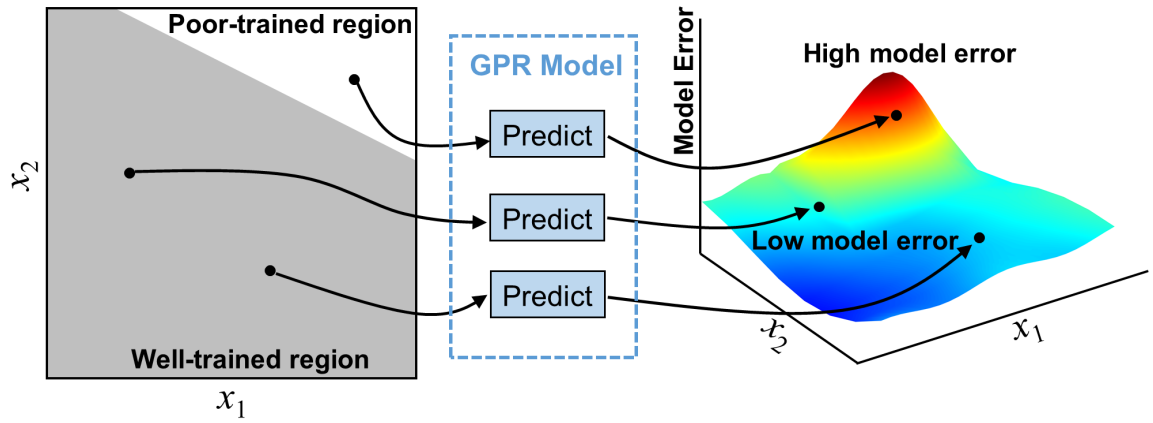



**Figure 2.21.** Model error from differently trained regions.

Three sequential sampling approaches, i.e., the Maximin approach, Variance Minimization (VM) method, and the Voronoi method are proposed to identify the sample points that need to be trained in the latent space. The goal of this section is to adaptively identify new training points $\mathbf{x}^*$ in the input space, utilizing the information obtained from the existing input space $\mathbf{X_C}$.

### Global refinement

The global refinement is defined as a strategy which adds essential points based on current well-trained region to extend the cover range of the latent space. In this case, the Maxmin approach (Jin, Chen and Sudjianto, 2002) is adopted which adaptively determines new training points by maximizing the minimum distance between the new point and all current available training points

$$d_{new} = max\{min[\|d - d^t\|_2]\}, \tag{2.10}$$

where $d_{new}$ is the identified location of the new training point $x^*$, $d^t$ denotes all current available training points, and $\|\cdot\|_2$ is the $l_2$-norm of a vector. By using this method, a larger training space of the design domain can be evenly sampled with training points.

**Local refinement**

The local refinement is defined as a strategy which optimizes the training space by further sampling the regions with the largest prediction error. Given the different definitions of "prediction error", two local refine strategies are developed. In VM method, the new training point in each iteration is selected by minimizing the maximum mean square error (MSE) or prediction variance as

$$d_{new} = max\{MSE(d)\}, \tag{2.11}$$

where $MSE(\cdot)$ is the prediction variance of the surrogate model. The VM methods can effectively construct a global surrogate model when the variation of the response is similar across the design domain. However, VM is limited to the GP-based surrogate modeling method because it requires additional information from model outputs. Besides, when the underlying black box function is highly nonlinear in only certain design regions, the VM methods become inefficient (Hu and Mourelatos, 2018).

We then further proposed Voronoi method, serving as an alternative to the VM method. The Voronoi method finds the most sensitive Voronoi cell to sample more points in this region. Such sensitive region when removed, the predicted response constructed by the rest of existing points will be far away from the actual response (Xu, Liu, Wang and Jiang, 2014).

As shown in Fig. 2.22, the design space is partitioned into $N_C$ Voronoi cells in each

iteration, where $N_C$ is the number of training data at current iteration, as follows

$$\Omega = \underset{i=1,\ldots,N_C}{\cup} R_i \tag{2.12}$$

where $R_i$ is the domain of the $i-th$ cell defined as below

$$R_i = \underset{\boldsymbol{d}_j \in \boldsymbol{D}/\boldsymbol{d}_i}{\cap} \left\{ \boldsymbol{x} \in \boldsymbol{X}, \|\mathbf{x} - \mathbf{d}_i\|_2 \leq \|\mathbf{x} - \mathbf{d}_j\|_2 \right\}, \tag{2.13}$$

in which $\mathbf{D}/\mathbf{d}_i$ represents the training data excluding $\mathbf{d}_i$.

From the $N_C$ Voronoi cells, the most important cell is identified in each iteration as follows

$$i^* = \underset{i \in \{1,,\cdots N_C\}}{\max} \{e^i_{LOO}\}, \tag{2.14}$$

where $e^i_{LOO}$ is the leave-one-out (LOO) prediction bias given by

$$e^i_{LOO} = \left\| f(\mathbf{d}_i) - \hat{G}_{\mathbf{D}/\mathbf{d}_i}(\mathbf{d}_i) \right\|, \tag{2.15}$$

in which $f(\mathbf{d}_i)$ represent the true response of the training data $\mathbf{d}_i$ and $\hat{G}_{\mathbf{D}/\mathbf{d}_i}(\mathbf{d}_i)$ is the prediction of a GPR model trained using training data $\mathbf{D}$ excluding $\mathbf{d}_i$.

After the important cell (i.e. $i^*$) is determined, the new training input is identified in that cell by maximizing the distance between the new training data and the current training data (i.e. $\mathbf{d}_{i^*}$) as follows

$$\mathbf{x}^* = \underset{\mathbf{x} \in R_{i^*}}{\arg\max} \{\|\mathbf{x} - \mathbf{d}_{i^*}\|\}, \tag{2.16}$$

where $R_i$ is the Voronoi cell defined in Eq. (2.13).

Once the new training point is added, the design space is then re-partitioned into $N_C + 1$ Voronoi cells in the next iteration as illustrated in Fig. 2.22. Voronoi method takes use of the information from the existing surrogate models and are not limited to the

GP-based surrogate modeling method.

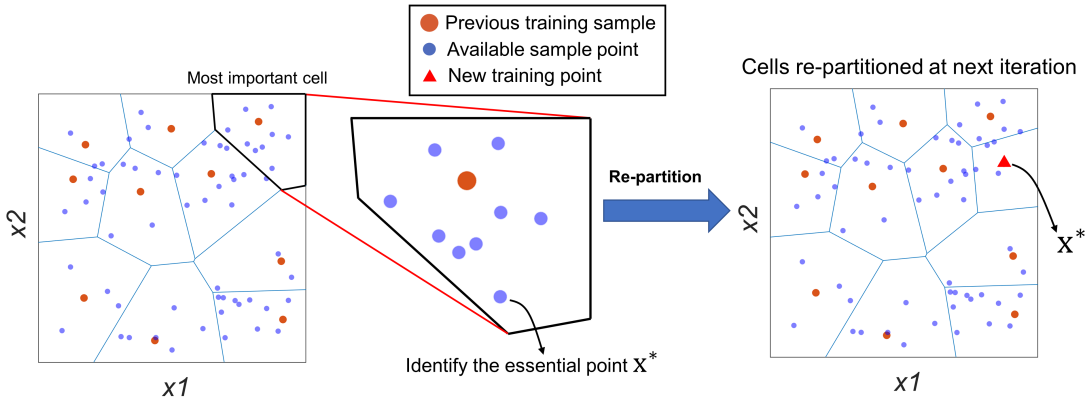



**Figure 2.22.** Iteratively adding new training points by the Voronoi method.

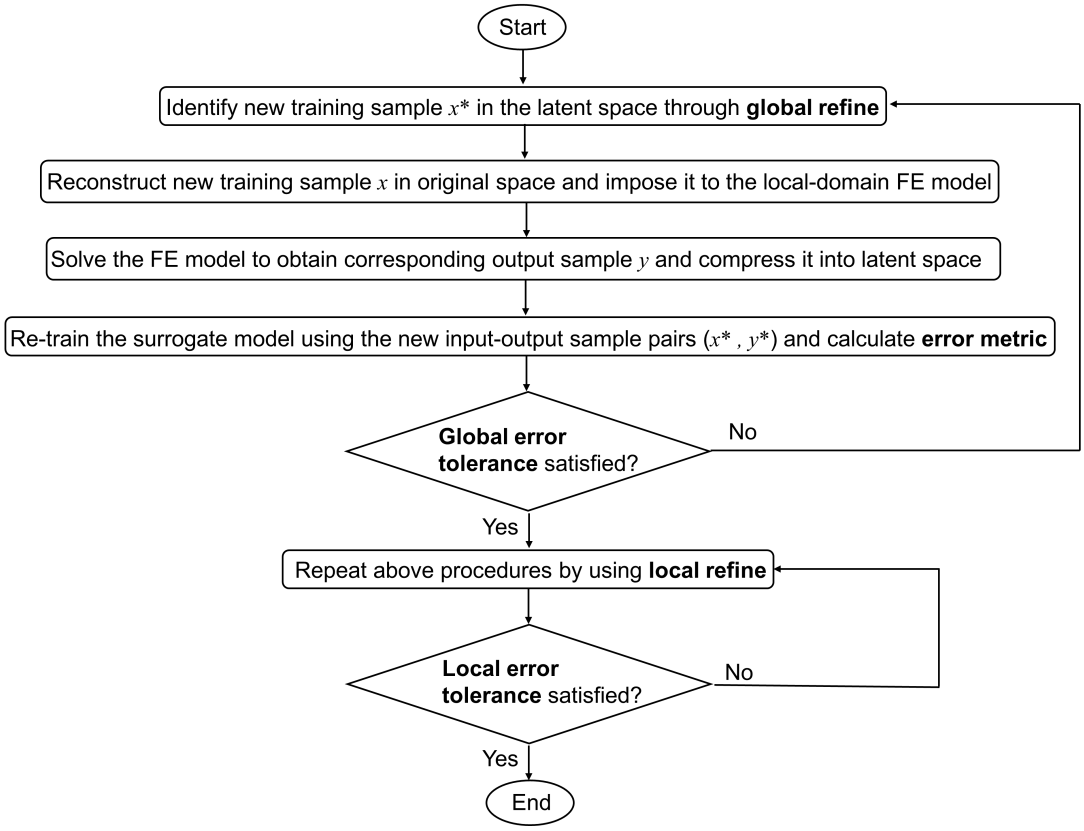



**Figure 2.23.** Flowchart of the overall procedure from initial surrogate modeling to well-trained model.

The overall process of combining global refine and local refine from initial surrogate modeling to well-trained model is shown in Fig. 2.23. The input space of the GPR is

sufficiently sampled by adaptively identifying new training points in the poor-trained region, which improves the surrogate performance without filling the training space blindly. Denoting all the identified new sample point in the input space as $\boldsymbol{x}_{new}$, to form a complete training dataset for GPR, the corresponding training points in the output space $\boldsymbol{y}_{new}$ have to be obtained. Firstly, the displacement in physical domain $\boldsymbol{u}^G_{new}$ is reconstructed from the compressed displacement $\boldsymbol{u}'^G_{new}$ determined in $\boldsymbol{x}_{new}$. By imposing $\boldsymbol{u}^G_{new}$ into the local ABAQUS FE model, the local reaction $\boldsymbol{\lambda}^L_{new}$ can be solved. The new training point in the output space of GPR model $\boldsymbol{\lambda}'^G_{new}$ is then obtained by compressing the full-dimensional data into latent space using output encoder as follows

$$
\begin{aligned}
&\text{Decoder}_u = \boldsymbol{E'}_u \boldsymbol{V'}^T_u, \\
&\text{Encoder}_\lambda = pseudoinverse(\boldsymbol{E'}_\lambda \boldsymbol{V'}^T_\lambda), \\
&\boldsymbol{u}^G_{new} = \boldsymbol{u}'^G_{new} \times \text{Decoder}_u, \\
&\boldsymbol{\lambda}'^L_{new} = \boldsymbol{\lambda}^L_{new} \times \text{Encoder}_\lambda,
\end{aligned}
\tag{2.17}
$$

The updated training dataset in the latent space of GPR after after refinement is now obtained as follows,

$$
\begin{aligned}
\mathbf{X}'_{updated} &= [(\boldsymbol{u}'^G_1, a_1), (\boldsymbol{u}'^G_2, a_2), ... \\
&\quad , (\boldsymbol{u}'^G_N, a_N), (\boldsymbol{u}'^G_{new}, a_{new})] \\
&\quad \in \mathbb{R}^{((N_T+N_{new})\times(N'_u+1))}, \\
\mathbf{Y}'_{updated} &= [\boldsymbol{\lambda}'^L_1, \boldsymbol{\lambda}'^L_2, ..., \boldsymbol{\lambda}'^L_N, \boldsymbol{\lambda}'^L_{new}] \\
&\quad \in \mathbb{R}^{((N_T+N_{new})\times N'_\lambda)},
\end{aligned}
\tag{2.18}
$$

where $N_{new}$ is the total number of added training points in the refinement. The GPR model is then considered as fine-developed as it covers a larger well-trained training region.

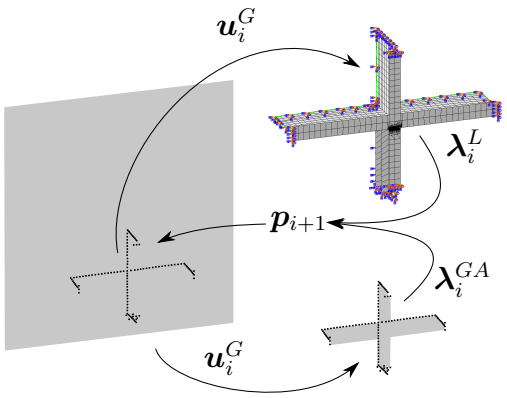



**Figure 2.24.** Illustrated IGL algorithm for miter gate with 1) global and global auxiliary static condensation uncondensed nodes and 2) local domain mesh discretization.

### 2.5.3 Surrogate IGL method combining statically condensed physics-based model in global domain and data-driven surrogate model in local domain

For the miter gate example, solution of the global domain reduced system of equations (in this example the matrix is $720 \times 720$) takes less than a second. This is particularly attractive when considering the IGL iterations of a nonlinear problem. For example, say a crack-propagation is discretized to $c$ crack lengths and each crack length takes 5 IGL iterations. The static condensation reduced matrix can be used $5c$ times with only one front-end cost, resulting in dramatic time savings.

No special handling is required for the inputs and outputs of the statically condensed global model, making it a plug-in replacement for the FEM global model in IGL as shown in Fig. 2.24. With these improvements, the bottleneck for IGL solution time is now the local problem. However, while calculating $\boldsymbol{p}_j$ the reaction forces of $\Omega_{GA}$ cannot be pulled directly from the FEM model without element information around the boundary, so the statically condensed global auxiliary domain is used instead.

The proposed surrogate local model in Section 2.5 removes the local solution bottleneck. The GPR surrogate local model receives $\boldsymbol{u}_i^G$ and outputs $\boldsymbol{\lambda}_i^L$, making it a plug-in replacement for the FEM local model. The surrogate iterative global-local method

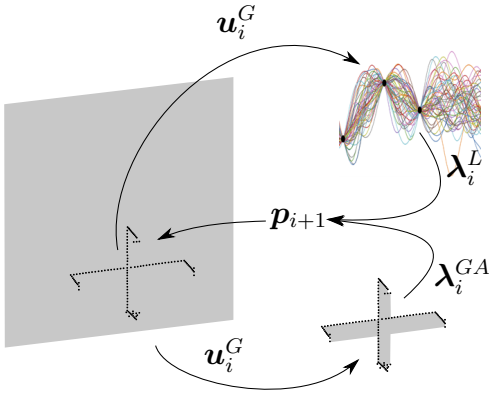



**Figure 2.25.** Illustrated surrogate IGL algorithm for miter gate with 1) global and global auxiliary static condensation uncondensed nodes and 2) local domain GPR surrogate.

is illustrated in Fig. 2.25.

### 2.5.4 Extracting SIF Values After Convergence

After the IGL reaches its convergence, the local FE model after the last iteration is considered to preserve a true physics. As mentioned above, the SIF value at the middle node through the thickness of the is extracted from the local model after post-processing. In SIGL, however, due to the physics is replaced by surrogate modeling, it is important to fill the gap between local reaction forces $\lambda^L_{convergence}$ and SIF. Given that, another surrogate model is built and trained to increase the running efficiency.

In this case, the 720-DOF reaction force vector $\boldsymbol{\lambda}_j^L$ is compressed into a 4-dimensional vector, which forms a 5-dimensional input combining with crack length parameter in the latent space. The output is then defined as the desired SIF value $K_{1-SIGL}$. The GPR-based surrogate models are then built and trained in the designed latent space with training samples $\boldsymbol{X}$ and $K_{1-SIGL}$.

$$K_{1-SIGL} = G_{SIF}(\mathbf{X}), \tag{2.19}$$

where $G_{SFIF}(\cdot)$ is the GPR surrogate model connecting local reaction forces with SIF.

For any given value of local reaction forces $\mathbf{X}$, we have the prediction of SIF from the GPR surrogate model as follows

$$G_{SIF}(\mathbf{X}) \sim N(\mu_K,\ \sigma_K^2), \tag{2.20}$$

in which $N(\cdot)$ is Gaussian distribution, $\mu_K$ and $\sigma_K$ are respectively the mean and standard deviation of the SIF prediction.

Next, we will use the miter gate example presented in Section 2.4 to compare the different approaches including submodeling, IGL, and surrogate-based IGL (SIGL).

## 2.6 Results and Discussion

Several solution methods have been covered: 1) Reference tying method, 2) submodeling, 3) IGL, 4) IGL with static condensation for global and auxiliary domains, and 4) SIGL. The methods for which accuracy is considered in this research are the IGL and SIGL methods. An example of their accuracy with a miter gate and $a = 1$ in, $h_{up} = 50$ ft, $h_{down} = 16$ ft, and $l_{dmg} = 0.5$ in is shown in Fig. 2.26. The IGL method with and without static condensation gives the same solution, so it is not shown in the figure.

For the IGL method the error drops below $10^{-5}$ after only three iterations. The SIGL method takes more iterations, but reaches an error below $10^{-4}$. While the error definition in Algorithm 1 is convenient for defining the IGL method convergence, the example problem proposed depends on the accuracy of the stress intensity factors along the crack front as compared with the reference tying method. Figure 2.27 shows the relative stress intensity factor error $e_K$ evaluated at each IGL method and SIGL method iteration.

The IGL method quickly converges to below $10^{-3}$ while the SIGL method lags somewhat, but still achieves an $e_K$ near 2%. Since the residual convergence $e$ showed better convergence, it seems likely that this is due to lack of accuracy in the SIF surrogate model. As for the higher $e_K$ than $e$, it can be helpful to look at the physical quantities each

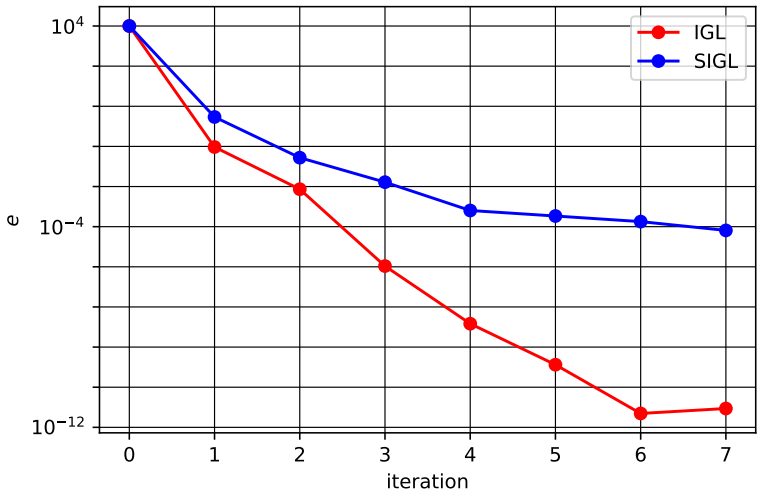



**Figure 2.26.** Error convergence of IGL and SIGL methods for $a = 1$ in, $h_{up} = 50$ ft, $h_{down} = 16$ ft, and $l_{dmg} = 0.5$ in.

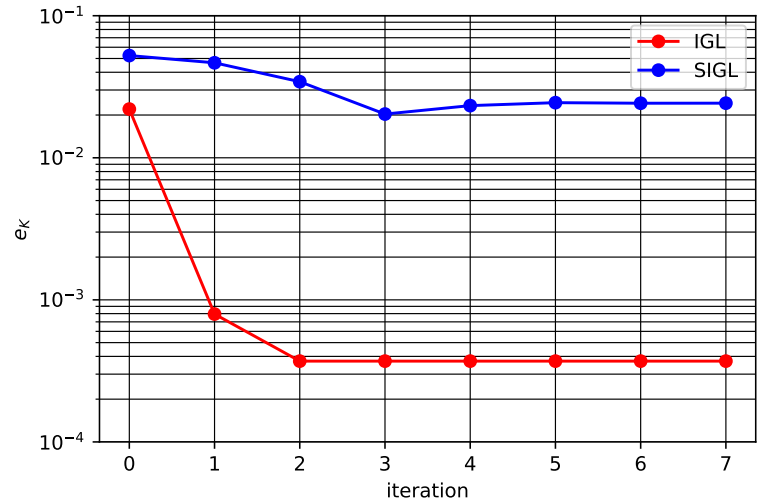



**Figure 2.27.** SIF error $e_K$ convergence of IGL and SIGL methods for $a = 1$ in, $h_{up} = 50$ ft, $h_{down} = 16$ ft, and $l_{dmg} = 0.5$ in.

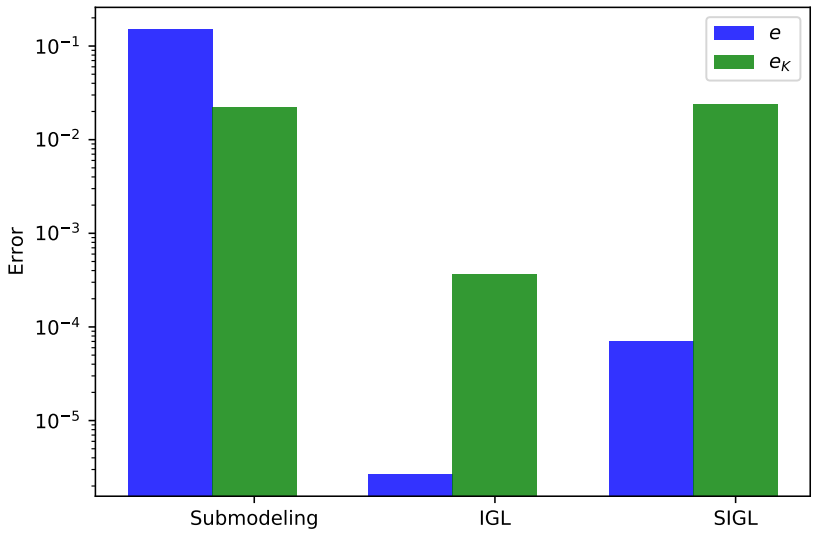



**Figure 2.28.** SIF error $e_K$ accuracy of submodeling, IGL and SIGL methods for $a = 1$ in, $h_{up} = 50$ ft, $h_{down} = 16$ ft, and $l_{dmg} = 0.5$ in.

deal with. The error $e$ deals with residual forces at nodes, quantities solved for directly in the system of equations. However, $e_K$ depends on contour integrals involving evaluation of stress, a derived quantity from the displacements. Therefore, the error will be higher for SIF outputs than for residual forces. However, the SIF error stagnates at around $4 \times 10^{-4}$. This may be due to computer precision error between the reference tied model definition and IGL model definition, e.g. the geometry in Abaqus seems to only have single precision. Figure 2.28 shows the SIF error $e_K$ accuracy of the converged IGL and SIGL compared with the submodeling solution.

It can be seen that while $e_K$ is similar for submodeling and the SIGL method, $e$ is much smaller for the SIGL method. Since the IGL method is much more accurate than the submodeling model, this points to room for improvement in the surrogate SIF model. Also, the crack length is very small ($a = 1$), helping the submodeling St. Venant's assumption hold. As the crack length grows, the submodeling solution will become much less accurate. The SIGL method accuracy for varying $a$ and $l_{dmg}$ will be explored later in this section. For now, Fig. 2.29 shows the time to solution for several methods on the

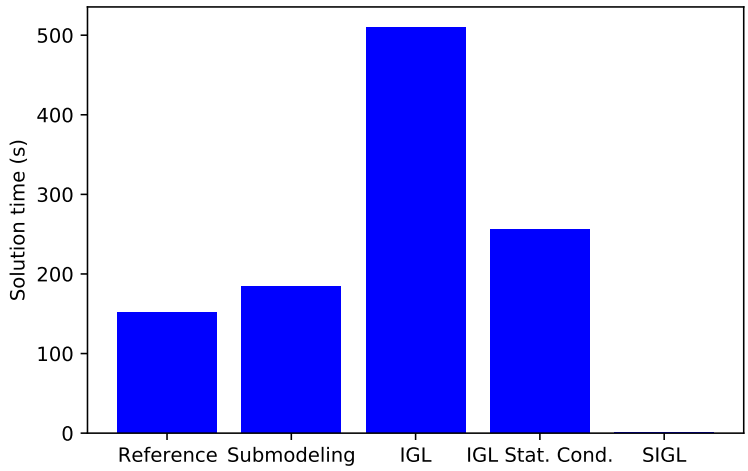



**Figure 2.29.** Solution time for reference tying model, submodeling, IGL, IGL with global and auxiliary static condensation, and SIGL methods for $a = 1$ in, $h_{up} = 50$ ft, $h_{down} = 16$ ft, and $l_{dmg} = 0.5$ in.

same desktop computer using 2 processors (CPUs) with a RAM of 32 GB.

The reference solution takes about 150 seconds. Interestingly the submodeling solution takes longer than the reference solution, which bodes ill for the IGL method. A possible explanation for this may be inefficiencies in multiple Abaqus calls versus one in input file analysis, assembly, solution, and post-processing. The IGL method takes three iterations and about three times the time of the reference solution. Using static condensation on the global and auxiliary problems cuts the IGL solution time in half, but IGL with static condensation is still slower than the reference solution. This is surprising given some of the discussion in Section 2.3.2 claiming a potential speed advantage for IGL. However, this can be explained by the limitations of performing analysis with Abaqus. Abaqus does not store the factorized stiffness matrix between jobs, so every job called after the first IGL iteration requires an (unnecessary to IGL) stiffness matrix assembly and factorization. However, The SIGL method has such a small time to solution 1.06 s that the bar is not visible.

Figure 2.30 shows the reference SIF results over cracks from 0.5 in to 4.0 in and

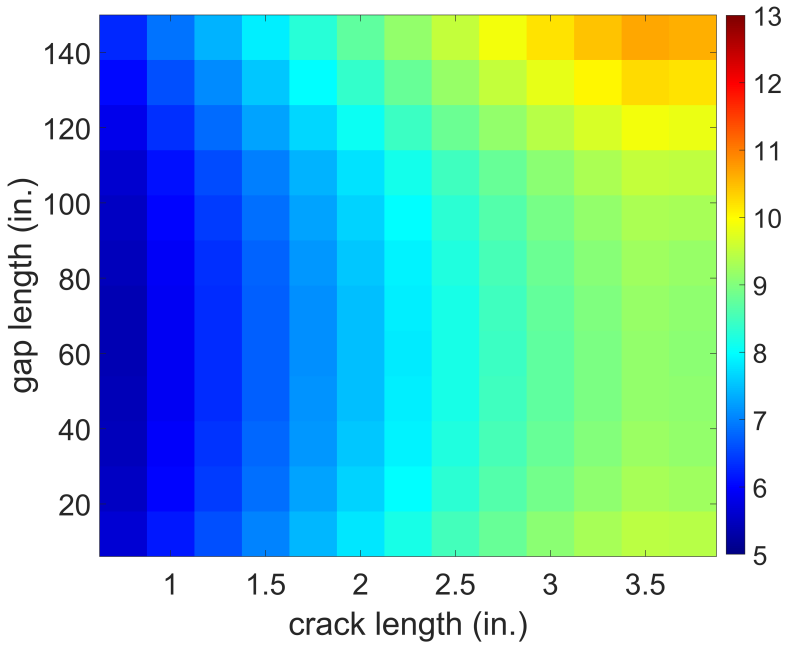
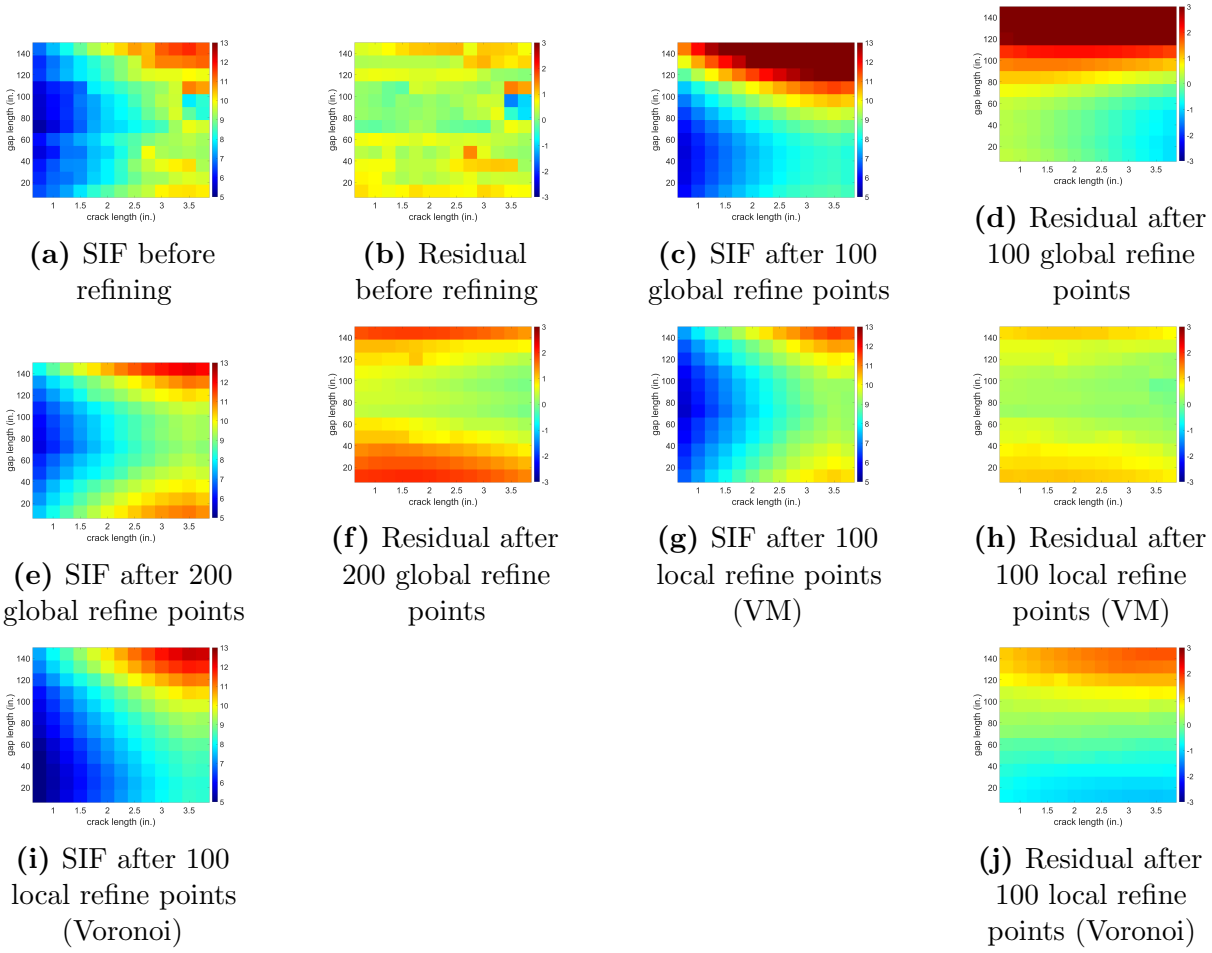



**Figure 2.30.** Heatmap of SIFs for reference solution in ksi$\sqrt{\text{in}}$.

damage gap height from 10 in to 150 in. The SIF values get higher for longer cracks, but the behavior for higher damage gaps is more complicated. In the range of 60 in to 120 in the SIF values are actually smaller, showing the importance of the location of the crack on the miter gate. Since the damage gap is at the bottom of the gate, the load path travels up and around it, and coincidentally the crack as well. However, the pintle (bottom hinge support) can take load, so as the damage gap grows higher the load paths somewhat divert back down through the crack.

Figure 2.31 shows the ability of the surrogate iterative global local method to model the miter gate. Interestingly, globally refining the surrogate local model leads to overestimation of the SIF value for large damage gaps. Considering that local refinement improves the solution drastically, the solution must be very sensitive for large damage gap heights. In fact, looking at the residual map for global refine, the error clearly depends on damage gap height more than crack length, peaking at the extremes. Higher error near the edges. Both local refine methods manage to control the prediction error based the




**(a)** SIF before refining

**(b)** Residual before refining

**(c)** SIF after 100 global refine points

**(d)** Residual after 100 global refine points

**(e)** SIF after 200 global refine points

**(f)** Residual after 200 global refine points

**(g)** SIF after 100 local refine points (VM)

**(h)** Residual after 100 local refine points (VM)

**(i)** SIF after 100 local refine points (Voronoi)

**(j)** Residual after 100 local refine points (Voronoi)

**Figure 2.31.** SIF heatmaps showing S-IGL performance in ksi$\sqrt{\text{in}}$.

global refine improvement.

## 2.7 Conclusion

A surrogate iterative global-local methodology has been proposed to reduce computation time for problems with cracked large steel structures. This research novelly represents the local domain in an IGL problem using a surrogate model rather than a physics-based model. It was shown that for the example problem (with a linear local domain) IGL was extremely accurate, while the required computational cost is high which is not suitable for probabilistic analysis such as failure diagnostics under the Bayesian

framework. However, SIGL achieves acceptable accuracy and is extremely fast. This makes SIGL well-suited for diagnosis and prognostic tasks in digital twins.

Future research will look at handling nonlinear global problems and utilization of SIGL to probabilistically infer crack length given sensor readings.

## 2.8 Remarks

This chapter is composed of a second-authored publication: Fillmore, T. B., **Wu, Z.**, Vega, M. A., Hu, Z., & Todd, M. D. (2022). A surrogate model to accelerate non-intrusive global–local simulations of cracked steel structures. Structural and Multidisciplinary Optimization, 65(7), 208.

# Chapter 3

# Diagnostics and Prognostics of Multi-Mode Failure Scenarios in Miter Gates Using Multiple Data Sources and a Dynamic Bayesian Network

## 3.1 Abstract

Current health monitoring approaches for large structures mostly rely on a combination of distributed sensor networks and in-situ inspection. This chapter presents a novel online diagnostics and prognostics framework for structures subject to multiple failure modes and demonstrates the proposed method with a high-fidelity finite element model using multiple data sources (i.e., strain gauges and images). The approach aims at an accurate simulation of the interaction between different failure features, and subsequently at the effective estimation and prediction of the damage states based on the generated structural physics. A dynamic Bayesian network is used which incorporates different data sources to evaluate the structures under different kinds of deterioration mechanisms. In diagnosis, the dynamic Bayesian network is used to approximate the damage-related parameters and estimate the time-dependent variables. In prognosis, the dynamic Bayesian network gives a probabilistic prediction of the remaining useful life of the structure based on the evolution of the failures. It is found that the proposed framework is highly effective

in performing online diagnosis and prognosis using combined data sources.

## 3.2 Introduction

The United States inland waterway system contributed $33.8 billion to GDP in 2014 (PricewaterhouseCoopers, 2017). Locks and navigation dams play an important role in inland waterway systems by providing a consistent navigable channel in a series of pools along the entire waterway. Locks open a gate to give boats entry and allow boats to travel between pools. If a lock gate cannot perform this function, barge traffic shuts down in that portion of the waterway. The most common type of lock gate within the United States Army Corps of Engineers is the miter gate, more than half of which have exceeded their economic design life of 50 years (Foltz, 2017), increasing the risk of major impacts to barge traffic.

The long life of many miter gates presents difficult life-cycle management decisions. To proactively schedule the maintenance of structures and thus reduce the overall life-cycle costs, numerous structural health monitoring (SHM) and damage prognosis (DP) strategies have been developed (Sabatino and Frangopol, 2017; Leser, Hochhalter, Warner, Newman, Leser, Wawrzynek and Yuan, 2017; Yang, Madarshahian and Todd, 2020; Vega and Todd, 2020; Su, Wan, Dong, Frangopol and Ling, 2021). In SHM, damage diagnosis aims to detect and quantify the potential damage, which provides essential information about the current health state. Failure prognosis, in the meanwhile, uses the gathered damage information to simulate the evolution of the damage, and further predicts the remaining useful life (RUL) of the structures. In recent years, the "digital twin" concept has drawn intensive attention because of its ability to inform damage diagnostic and failure prognostic strategies by simulating life cycle scenarios (Tuegel, Ingraffea, Eason and Spottswood, 2011; Li, Mahadevan, Ling, Choze and Wang, 2017a; Ye, Butler, Calka, Iangurazov, Lu, Gregory, Girolami and Middleton, 2019).

For the digital twin concept, one of the most commonly used physics-based approaches in digital twin execution is high-fidelity finite element (FE) analysis, which computationally reflects the evolving physical system. For instance, Zhang, Song, Lim, Akiyama and Frangopol (2019) presented a reliability estimation procedure for RC structures at different corrosion levels which used X-ray and digital image processing technique to infer the spatial variability of steel corrosion. With a focus on the seismic cracking identification, Pirboudaghi, Tarinejad and Alami (2018) developed a damage detection procedure for concrete gravity dam by integrating the FE numerical model with the wavelet transform system identification. Jiang, Vega, Todd and Hu (2022) proposed a model correction and updating scheme to improve the accuracy of failure prognostics by recovering the missing physics in the boundary condition degradation of miter gates. Eick, Levine, Smith and Spencer Jr (2021) suggested a fatigue life updating method for embedded miter gate anchorages. Commonly, in a digital twin framework as shown in Fig. 3.1, a physical asset (i.e. the miter gate) is connected to its digital counterpart core (i.e., the FE model) through Bayesian updating methods and real-time SHM monitoring data. Bayesian updating methods infer the damage state based on monitoring data and thereby allow the digital twin to not only estimate the current damage level but also to forecast potential failure before it happens.

Even though current efforts have shown the promising potential of the digital twin in optimizing the maintenance activity of large-scale assets, they mainly focus on a single-mode failure scenario (e.g., boundary condition degradation of miter gates). For steel structures such as miter gates, fatigue cracks are another very common structural deterioration mechanism. As fatigue cracks propagate, they may interact with other failure modes. Cracks may be computationally modeled using XFEM/GFEM (Moës, Dolbow and Belytschko, 1999; Duarte, Hamzeh, Liszka and Tworzydlo, 2001b), which is much more mesh independent than quarter node element crack representation (Henshell and Shaw, 1975b; Barsoum, 1976b). Therefore, practitioners widely use XFEM/GFEM for crack

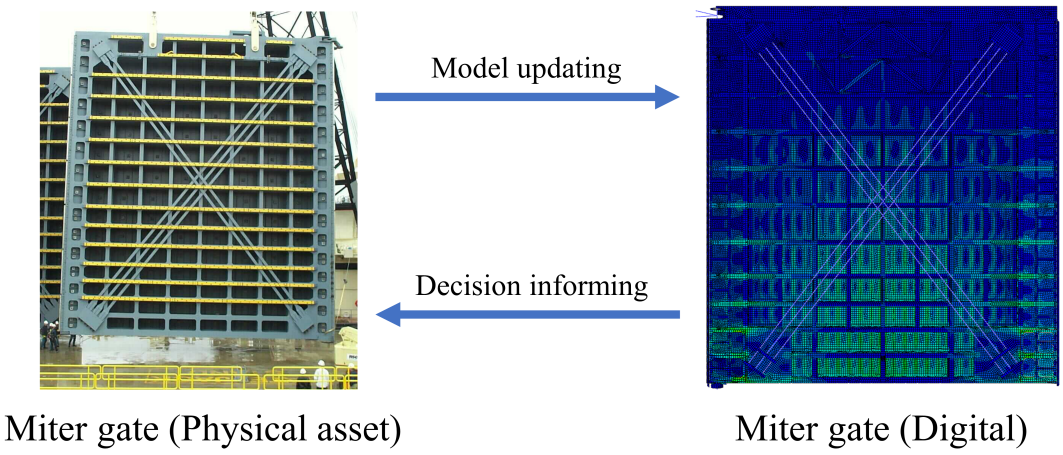



**Figure 3.1.** Digital twin concept of miter gates.

modeling/analysis. Although FE analysis offers high interpretability (Moës, Gravouil and Belytschko, 2002; Gravouil, Moës and Belytschko, 2002; Shi, Chopp, Lua, Sukumar and Belytschko, 2010; Xie, Bott, Sutton, Nemeth and Tian, 2018), the separation in length scales between structural scale (e.g., at the scale of miter gates) and damage scale (e.g. at the scale of cracks) may increase numerical model discretization and add computational cost. Moreover, the existing methods rely upon strain measurements for model updating (Parno, O'Connor and Smith, 2018; Hoskere, Eick, Spencer Jr, Smith and Foltz, 2020). With novel measurement techniques, such as cameras, and drones, developed for the monitoring of miter gates in recent years, there is an urgent need to develop an integrated diagnostic/prognostic capability that uses multiple data sources (including strain gauges) to simultaneously account for multiple failure modes.

In this chapter, we focus on two failure scenarios at different scales, including the boundary condition degradation at a global scale and the crack growth of a cruciform at a very localized scale. Two types of measurements are considered: strain measurement data from strain gauges and displacement observations extracted from digital images. In order to develop the framework for multiple failure modes and data sources, two main challenges need to be addressed, namely: (1) how to properly model different failure modes of miter

gates; and (2) how to fuse multiple data sources for the model updating of the digital model.

The iterative global-local (IGL) method is employed to address the first challenge. Dealing with separation of scales is a broad field of research in crack modeling. Of particular interest are methods bridging scales non-intrusively with XFEM cracking represented in the local domain (Gupta, Pereira, Kim, Duarte and Eason, 2012b; Fillmore and Duarte, 2018b). The IGL method offers particularly good non-intrusive characteristics (Allix and Gosselet, 2020), requiring only the exchange of reaction and displacement related quantities along the local boundary. Despite the relatively simple coupling of global and local models, the IGL method can simulate nonlinearities in the local model with a linear global model (Gendre, Allix, Gosselet and Comte, 2009b). Within the context of large structures modeled as shells, such as a miter gate, the IGL method has been successfully used to connect shell global domains to solid local domains with welds (Li, O'Hara and Duarte, 2021b).

To address the second challenge of fusing multiple data and uncertainty sources for model updating of miter gates with multiple failure modes, a dynamic Bayesian network (DBN) model is developed in this chapter. DBNs have been widely used for studies where the topology structure represents causal relationships, and for building digital twins of complex engineering systems such as aircraft structures (Li, Mahadevan, Ling, Choze and Wang, 2017a) and nuclear power plants (Agarwal, Neal, Mahadevan and Adams, 2017). For example, Li et al. (2017a) suggested a digital twin framework for diagnosis and prognosis of an aircraft wing using a DBN as a versatile probabilistic model. A detailed discussion on using DBN as a unifying mathematical tool for digital twins at scale is available in Kapteyn, Pretorius and Willcox (2021). As a probabilistic graphic model, DBN allows for information fusion of various data and uncertainty sources (both aleatory and epistemic uncertainty sources) using Bayesian inference and conditional probabilistic models. The recursive updating scheme supports the digital twin's need for real-time

updating and prediction over time, which plays an essential role in fully realizing the promising potential of digital twin of miter gates.

The main objective of this chapter is to develop a framework that utilizes image-based observations and strain sensor data to diagnose and predict failure features in large-scale structures. The physics of two types of failure modes is represented in an FE model of the miter gate: the boundary condition loss represents the large-scale damage; the fatigue crack growth represents the small-scale local damage. For illustration purposes, this chapter takes the crack growth on the bottom flange edge of a horizontal girder on a miter gate as an example. The underlying concepts, however, can be extended and applied to other locations and different types of structures. The proposed framework includes two main steps, as shown in Figure 3.2: (1) effective simulation of failure modes in different length scales using a global-local modeling method with surrogate modeling to increase computational efficiency, and (2) online diagnostics and prognostics based on the two types of observations.

The rest of the chapter is arranged as follows. Section 3.3 presents the modeling of miter gate failure scenarios based on an FE model. The proposed diagnostic and prognostic framework using multiple data sources and a DBN is described in Sec. 3.4. Section 3.5 gives the key results and discussion, followed by Sec. 3.6 which draws the conclusions.

## 3.3 Modeling of Miter Gate Failures

### 3.3.1 Boundary condition degradation

Figure 3.3 shows the downstream side view of a miter gate in a dewatered state. The gudgeon and pintle function as pivots for the miter gate's rotation. Normally, the bottom of the miter gates are submerged below water, resulting in hydrostatic pressure pushing the two leaves of the gate together. Hydrostatic pressures are applied on the upstream plate of the gate as shown in Fig. 3.4, where the upstream water level is denoted

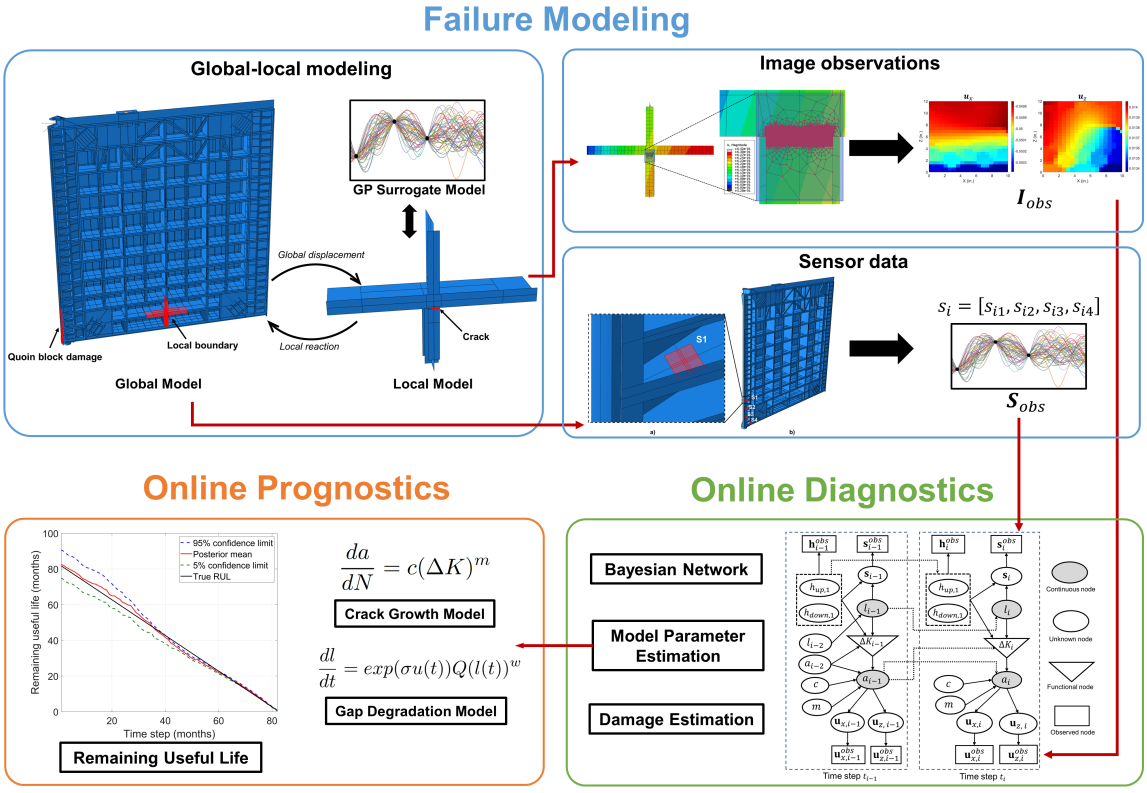



**Figure 3.2.** Overview of the proposed framework.

$h_{up}$ and the downstream water level is denoted $h_{down}$. Since the hydrostatic pressures is considered to be fully described by the water levels, the loading condition resulted by hydrostatic pressure will be symbolized by parameter $\mathbf{h} = [h_{up}, h_{down}]$ for the rest of the chapter. When the gate holds enough water in the lock chamber, the miter contact block of both gate leaves come into contact and a symmetric pin is assumed preventing translational movement. The two gate leaves act as an arch, experiencing more axial compression under more hydraulic head. This compression causes the miter gate's quoin contact block to thrust into the lock wall contact block. The quoin often experiences damage so that only part of it comes into contact with the lock wall. When the miter gate is open, boats can enter or leave the lock chamber. When the miter gate is closed, the lock chamber can be filled or emptied while the miter gate acts as a damming surface. More detailed information about miter gates may be found in (Daniel and Paulus, 2019;

Eick, Smith and Fillmore, 2019b; Fillmore and Smith, 2021b).

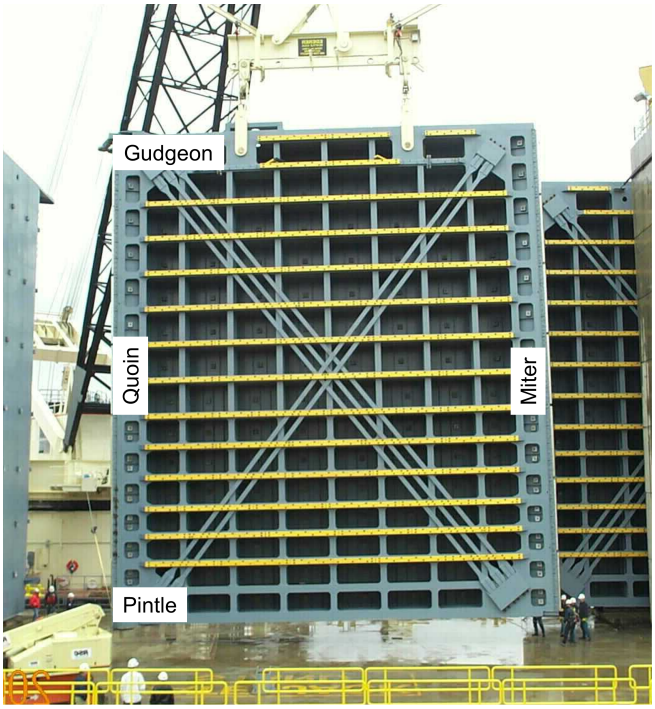



**Figure 3.3.** Miter gate downstream side view. Photograph courtesy of John Cheek, USACE.

The aging of the gate is manifested by multiple forms of damage. Often, the bottom portion of the quoin becomes damaged so that it cannot properly contact the wall. To account for the effects of quoin block damage, a simplified gap degradation model (Vega, Hu, Fillmore, Smith and Todd, 2021b) is generalized below,

$$\frac{dl}{dt} = exp(\sigma U(t))Q(l(t))^w, \tag{3.1}$$

where $U(t)$ is a random variable with a standard normal distribution; $\sigma$, $Q$, and $w$ are empirical parameters based on previous research (Yang and Manning, 1996; Jiang, Vega, Todd and Hu, 2022).

The discrete-time form of Eq. (3.1) can be written as

$$l_i = l_{i-1} + exp(\sigma U_i)Q(l_{i-1})^w, \tag{3.2}$$

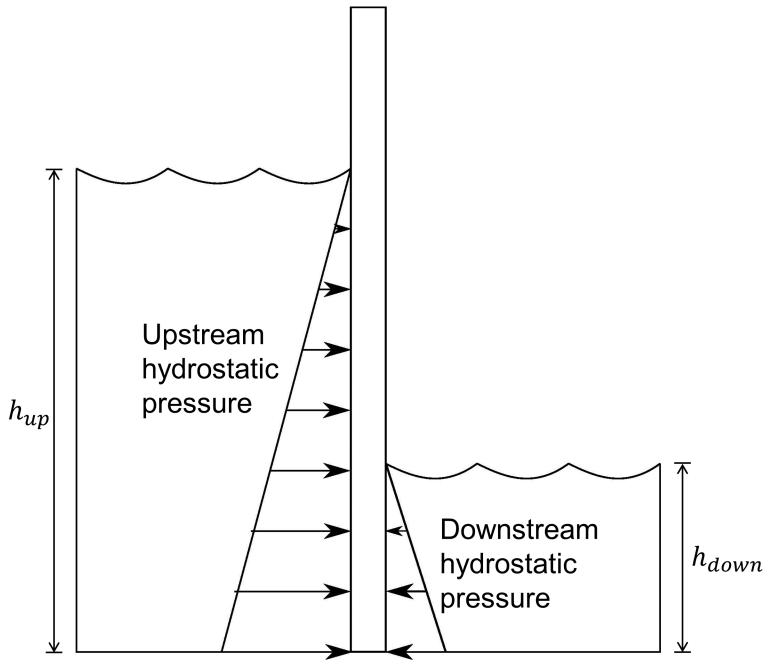



**Figure 3.4.** Miter gate hydrostatic pressure from upstream and downstream water levels.

where $l_i$ and $l_{i-1}$ are the state variable (gap length) at time steps $t_i$ and $t_{i-1}$ respectively, and $U_i$ is a standard normal random variable at $t_i$.

An FE model was generated using Abaqus 2021 as shown in Fig. 3.5. The model represents the Greenup downstream miter gate, which has been previously validated with the field data to provide accurate physics (Eick, Treece, Spencer Jr, Smith, Sweeney, Alexander and Foltz, 2018). This model is employed in this chapter in order to capture the global behavior and predict the strain responses of the gate. The quoin block contact loss is modelled by not applying the pinned boundary conditions along a certain length. For a more detailed description on the quoin block mechanism, refer to Fig. 8.37b in Daniel and Paulus (2019). For the rest of the chapter, the length of the contact loss interface is referred to as the *gap length* denoted $l_i$. The lengthening of this gap leads to a global re-distribution of the stress, which escalates crack evolution of the miter gates at different local regions. The gap damage state is connected with the strain responses as follows

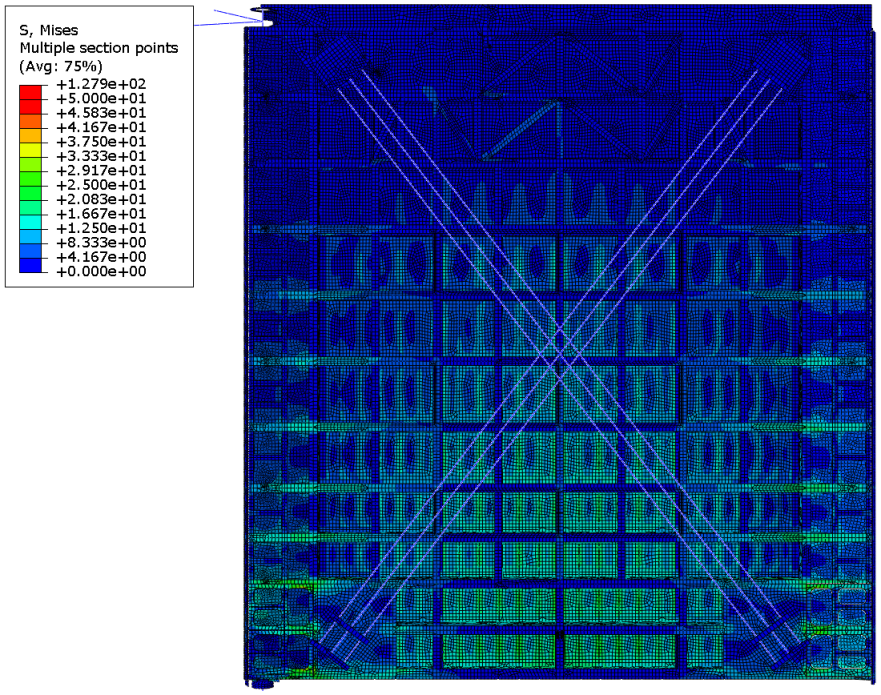



**Figure 3.5.** Finite element model of Greenup miter gate, showing global strain distribution.

$$\begin{aligned} &\text{State}: \ l_i = l_{i-1} + \exp(\sigma U_i) Q(l_{i-1})^w, \\ &\text{Measurement}: \ \mathbf{s}_i = g(l_i, \ \mathbf{h_i}) + \varepsilon_i, \end{aligned} \tag{3.3}$$

where $\mathbf{h}$ is the loading condition at a given time step $t_i$, $g(l_i, \ \mathbf{h_i})$ is the response of the FE model, $\varepsilon_i \sim N(\mathbf{0}, \ \sigma^2_{\mathbf{obs}}\mathbf{I})$ are the uncorrelated measurement noise contributions characterized by standard deviation $\sigma^2_{\mathbf{obs}}$, and $\mathbf{I}$ is an identity matrix.

### 3.3.2 Crack growth modeling using an iterative global-local algorithm

Besides the quoin block damage discussed above, fatigue cracks are a common form of miter gate damage due to the cyclic loads when the lock chambers are filled and emptied. Since the sparsely distributed strain gauge sensor network is fairly insensitive to crack

presence at an initial stage, conventional crack detection methods are mostly operated by in-situ inspectors, which makes the inspection somewhat subjective and labor-dependent. Besides, much of the gate is always submerged under water which increases the difficulty and accessibility of in-situ inspections. Thus, an accurate crack analysis using the miter gate FE model is necessary to understand the behavior of such localized effect. First, Paris' law–one of the most commonly used crack growth models–is adopted to generate the physics of the model, or

$$\frac{da}{dN} = c(\Delta K)^m \tag{3.4}$$

where $a$ is the crack length and $da/dN$ is the fatigue crack growth for a load cycle $N$, $c$ and $m$ are the empirical parameters of Paris' law, and $\Delta K$ is maximum stress intensity factor (SIF) difference in a loading cycle at the crack front, as shown in Fig. 3.6. The discrete-time form of Eq. (3.4) can be written as

$$a_i = a_{i-1} + c(\Delta K_i)^m, \tag{3.5}$$

in which $\Delta K_i$ stands for the SIF range at time step $t_i$.

Three main assumptions are made here to the FE model simplify the problem:

1. The crack can only propagate in one direction with a fixed crack front shape;

2. The 13 nodes (12 elements) through the 0.625 in. thickness of the cracked plate (solid geometry with linear hexahedral elements with XFEM enrichment functions shown in Fig. 3.6(b)) are sufficient to represent the crack physics, where only the first cracking mode of the middle node, $K_1$, is considered;

3. The geometry, boundary conditions, and discretization represent the Greenup gate leaf well enough for the diagnosis and prognosis in this research;

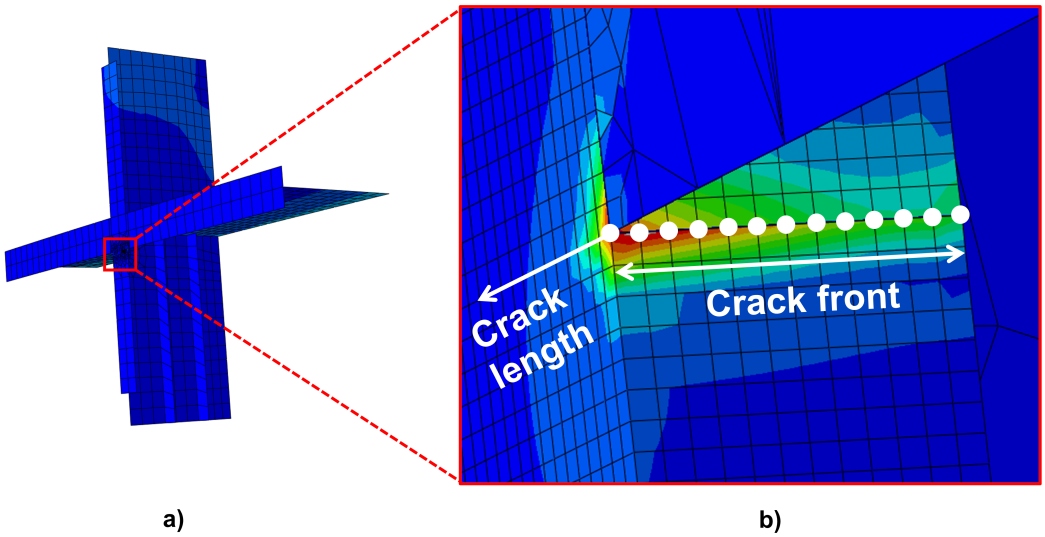



**Figure 3.6.** FE representation of the simulated crack front: (a) cruciform where crack initiates, and (b) a close view of crack front

With all the above assumptions, the crack geometry can be described by one single parameter, $a$. This chapter aims to provide a general framework for multi-mode failures of large-scale structures, the explicit form of crack representation is beyond the scope of this chapter. Thus, the crack problem is simplified in this study.

The maximum SIF difference in a loading cycle, $\Delta K$, is a variable that is affected by gap length, crack length, and load conditions, where gap length is a global-scale damage and crack is a local-scale damage. The fatigue crack modeling requires the calculation of accurate SIF values at each time step to indicate the crack growth pattern. The SIF at $t_i$ is a function of multiple factors,

$$\Delta K_i = g_{\Delta K}(l_i,\ a_i,\ \Delta s_i), \tag{3.6}$$

in which $\Delta s_i$ is the loading condition caused by the cyclic fluctuation of the hydrostatic pressure $\mathbf{h}$, and $g_{\Delta K}(l_i,\ a_i,\ \Delta s_i)$ is an FE model to predict the SIF range $\Delta K_i$ for given gap length, crack length, and loading cycle. Although the built-in Abaqus technology calculates SIF values through the contour integral method, crack analysis for the complicated and

large-scale miter gate model is computationally expensive due to the fact that the crack can only be simulated with finely-discretized solid elements in Abaqus. Given that, a coupled global-local FE model was generated using Abaqus 2021 as shown in Fig. 3.7. The IGL-based model is developed in order to address the challenge in estimating SIF caused by the two damage features in different length scales.

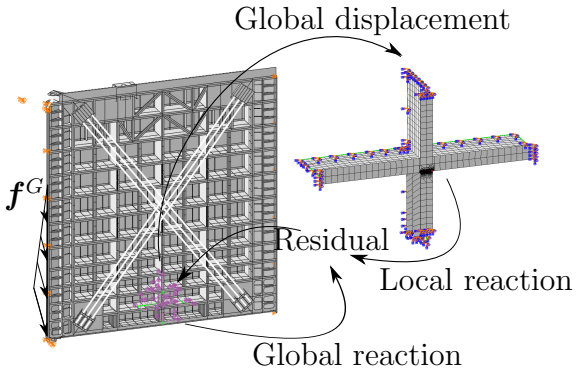



**Figure 3.7.** Illustrated IGL algorithm for miter gate with global, and local mesh discretizations. The global domain has parameters as $l$ and $\mathbf{h}$, with boundary condition described by parameter $\boldsymbol{f}^G$. The local domain has parameter $a$.

All the elements of the global model are 3D linear reduced-integration shell elements which lowers the computational cost. The local model is defined as a cruciform whose local boundary is shared with the global model. The local model takes the displacements from the global model as its boundary condition. The local model is divided into two parts: One is the crack affected zone with Abaqus XFEM 3D solid geometry which allows for crack analysis; the second part of the local model is the rest of the cruciform which uses the 3D linear reduced-integration shell elements. The feature of interest is the crack, which is only explicitly represented in the solid area of the local model. More detailed IGL implementation information may be found in Fillmore et al. (2022). For any given $l_i$, $\Delta s_i$, and $a_i$, the SIF value may be obtained. It is assumed that since a surrogate model trained on an identical FE model showed acceptable error (less than 10%), the surrogate model in this research also has acceptable accuracy.

## 3.4 Diagnosis and Prognosis of Miter Gates Using Multiple Data Sources and DBN

### 3.4.1 Structural health monitoring (SHM) data sources

The physics of the miter gate in this study is parameterized by three factors: the loading condition $\mathbf{h}$, the quoin block damage $l_i$ that is imposed on the global domain, and the crack length $a_i$ that is assigned to the local domain. Different combinations of such parameters induce different physical behaviors that are reflected in different observations. Image-based observations enable computer vision techniques to capture the cracks in the early stage while the strain sensor network detects the quoin block damage, resulting in load re-distribution within the whole structure. In this chapter, two types of surrogate models are built in order to efficiently perform probabilistic analysis based on the different measurements.

**Strain sensor network data**

To generate synthetic strain measurements, four sensor locations are selected in this chapter, which are close to the location that quoin block damage most likely will happen, as shown in Fig. 3.8. The sensors are located in compression regions, and thus negative strain values are recorded.

At any time step $t_i$, the strain measurements from the four strain gauge sensors are related to the FE model shown in Sec. 3.3.1 as follows

$$[s_{i1},\ s_{i2},\ s_{i3},\ s_{i4}] = g(l_i,\ \mathbf{h}_i) + \varepsilon_i, \tag{3.7}$$

where $s_{i1}$ represents the response of the first selected strain gauge at time step $t_i$, and $g(l_i,\ \mathbf{h}_i)$ is the strain output of the FE model for a given gap length and loading cycle. The measurement noise $\varepsilon_i$ is considered statistically independent and identical distributed Gaussian random variables.

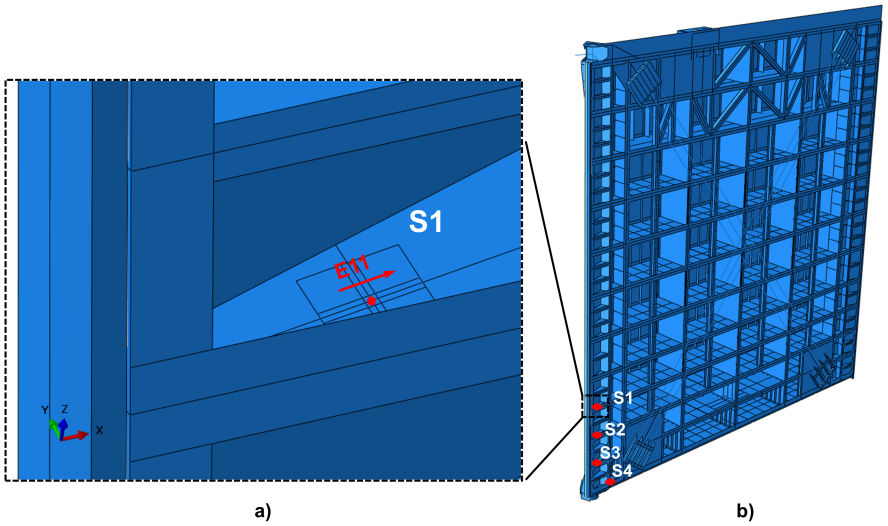



**Figure 3.8.** Sensor locations: (a) Individual sensor location and corresponding value, and (b) locations of the selected four sensors.

Since the original FE model $g(l_i,\ \mathbf{h}_i)$ is computationally expensive for damage diagnostics and failure prognostics, Gaussian process regression (GPR)-based surrogate models are constructed to replace the original model. Considering that there are only four strain gauges, we construct a GPR model for each sensor response separately. After that, Eq. (3.7) is rewritten as

$$s_{ij} = \hat{G}_j(l_i,\ \mathbf{h}_i) + \varepsilon_i,\ \forall j = 1,\ \cdots,\ 4, \tag{3.8}$$

where $\hat{G}_j(l_i,\ \mathbf{h}_i)$ is the GPR model for the FEA response of the $i$-th strain gauge and is given by

$$\hat{G}_j(l_i,\ \mathbf{h}_i) \sim N(\mu_{ij},\ \sigma_{ij}^2), \tag{3.9}$$

in which $N(\cdot, \cdot)$ is Gaussian distribution, $\mu_{ij}$ and $\sigma_{ij}$ are respectively the mean and standard deviation of the prediction of the $j$-th surrogate model at time step $t_i$.

Based on Eq. (3.8), the likelihood of observing $\mathbf{s}_i = [s_{i1},\ s_{i2},\ s_{i3},\ s_{i4}]$ at $t_i$ for given

$l_i$ and $\mathbf{h}_i$ is then given by

$$f(\mathbf{s}_i|l_i,\ \mathbf{h}_i) = \prod_{j=1}^{4} \phi\left(\frac{s_{ij} - \mu_{ij}}{\sqrt{\sigma_{ij}^2 + \sigma_{\varepsilon i}^2}}\right), \tag{3.10}$$

where $\phi(\cdot)$ is the PDF of a standard normal random variable.

**Image monitoring data**

The digital image is the key data source for crack detection. The evolution of the crack results in a displacement redistribution of the surface which may be captured by cameras or drones. Note that the cruciform on which the crack initiates is located at the center of the second-from-bottom horizontal girder, which is always underwater during lock chamber filling and emptying. In reality, photos obtained underwater usually have lower contrast and may be blurred out by the water reflection. The optical flow method (Alvarez, Weickert and Sánchez, 2000) and digital image correlation (DIC) (Pan, 2011) methods can obtain the measured dense displacement field assumed in this research. Correspondingly, a simplified digital image model is developed to represent the process of obtaining the displacement measurements from images (a "measurement model") using the optical flow method. Given the fact that Drews, Nascimento, Xavier and Campos (2014) found turbidity increases the error of optical flow fields and Madjidi and Negahdaripour (2006) proved that the low-contrast photo underestimates the magnitude of the optical flow field, the model down-sizes the displacement measurements and assigns a noise that represents the noise level of photos taken underwater. This noise also accounts for environmental factors such as camera vibration and light source movement over a lock filling event. The process of getting the displacement field can be expressed as follows

$$[\boldsymbol{u_x}, \boldsymbol{u_z}] = G_{OP}(l_i, a_i, \mathbf{h}_i), \tag{3.11}$$

where $\boldsymbol{u_x}$ and $\boldsymbol{u_z}$ are the localized displacements related to $\mathbf{h}$, $l_i$, and $a_i$, and $G_{OP}(l_i, a_i, \mathbf{h}_i)$ is the displacement field prediction from optical flow model. The transformation $G_{OP}$ depends on camera location, focal length, and other camera parameters. For simplicity here, the camera angle is normal to the crack location on the gate and the transformation from 3D to pixel coordinates is a linear scaling. Since the IGL algorithm developed in Sec. 3.3.2 offers an accurate way of measuring loading condition and the two different-scale damage states, the process of using optical flow model to generate synthetic displacements is represented by IGL model developed in Sec. 3.3.2.

$$[\boldsymbol{u_x}, \boldsymbol{u_z}] = G_{IGL}(l_i, a_i, \mathbf{h}_i) + \epsilon_i, \tag{3.12}$$

in which $G_{IGL}(l_i, a_i, \mathbf{h}_i)$ is the IGL algorithm.

First, a surface of interest around the crack in the cruciform Abaqus model is determined with a dimension of $10 \times 12$ inches. The built-in Abaqus post-processing provides the nodal displacements of all the nodes within the area, shown in Fig. 3.9. The irregular quadrilateral meshing elements generate nodal displacements at scattered locations. To simulate the uniformly distributed displacement field obtained from camera images, the scattered nodal displacements are interpolated onto a uniformly gridded surface as vectors $(x, z, v)$ using the "nearest" method, where the point of interpolation specified by location $(x, z)$ is assigned by the value of closest nodal displacement $v$.

Fig. 3.10 shows an example of the displacements in two directions obtained from IGL and linear interpolation with a pixel length of 0.1 inches when $\mathbf{h} = [h_{up}, h_{down}] = [506.8, 339.8]$, $l = 27.2$, and $a = 2.16$.

Since the IGL algorithm requires global-local model analysis, which is computationally expensive, we construct surrogate models for the localized displacements, similar to the surrogate models for the strain response. Since the high-dimensional displacement field is computationally impractical for surrogate modeling, singular value decomposition

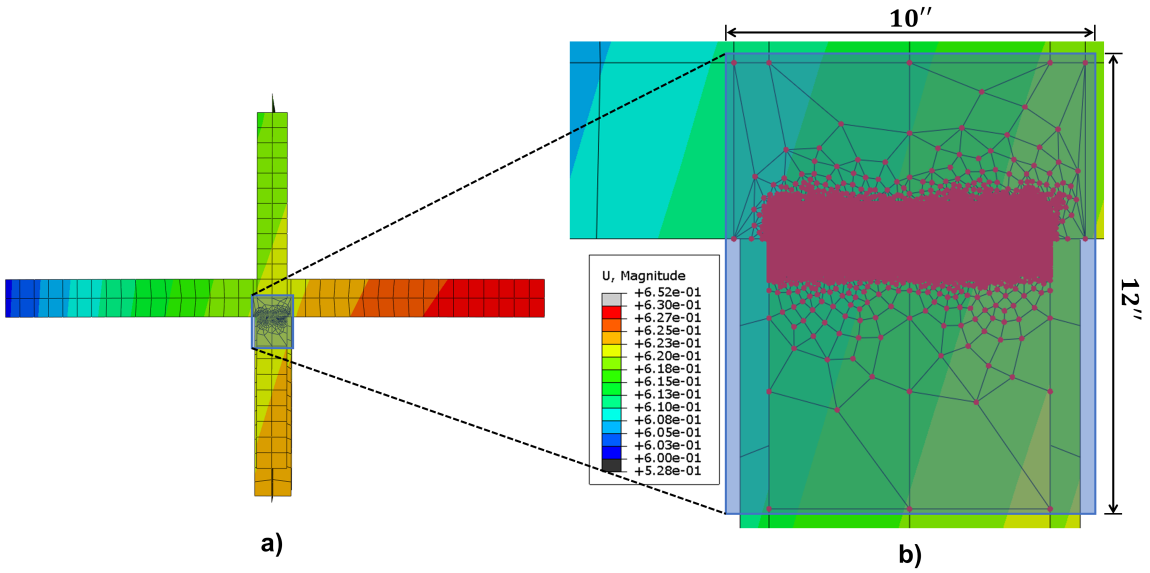



**Figure 3.9.** Area of Interest: a) Cruciform where the crack is evolving, and b) the area in which all the nodal displacements are extracted.

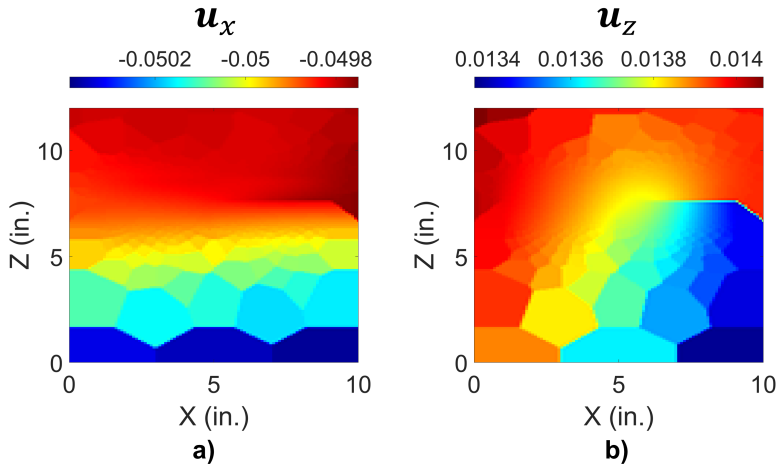



**Figure 3.10.** Displacement-based observation: a) displacement in $x-$direction, and b) displacement in $z-$direction.

(SVD) is employed to construct the GPR models by following the procedure suggested in Vega, Hu and Todd (2020). The surrogate modeling gives $\boldsymbol{u_x}$ and $\boldsymbol{u_z}$ as

$$\begin{aligned} \mathbf{u}_x &= \sum_{j=1}^{N_x} \hat{G}_{x,j}(l_i,\ a_i,\ \mathbf{h}_i)\eta_j + \varepsilon_{x,i}, \\ \mathbf{u}_z &= \sum_{k=1}^{N_z} \hat{G}_{z,k}(l_i,\ a_i,\ \mathbf{h}_i)\eta_k + \varepsilon_{z,i}, \end{aligned} \tag{3.13}$$

where $\hat{G}_{x,j}(l_i,\ a_i,\ \mathbf{h}_i) \sim N(\mu_{x,j}, \sigma^2_{x,j})$ is the $j$-th GPR surrogate model in the latent space, $\eta$ is the vector that transforms the latent space prediction back into full-dimensional displacement, and $\varepsilon_{\mathbf{x},\mathbf{i}}$ is the corresponding noise assigned to the observation model.

Based on the surrogate modeling and following the derivations given in Eq. (3.13), the likelihood of observing $\boldsymbol{u_x}$ is computed by

$$f(\mathbf{u}_x|l_i,\ a_i,\ \mathbf{h}_i) = \frac{\exp\left(-0.5(\mathbf{u}_x - \mu_x)^{\mathrm{T}} \boldsymbol{\Sigma}_x^{-1} (\mathbf{u}_x - \mu_x)\right)}{\sqrt{(2\pi)^{N_P} |\boldsymbol{\Sigma}_x|}}, \tag{3.14}$$

where $\mu_x$ and $\boldsymbol{\Sigma}_x$ are given by

$$\mu_x = \sum_{j=1}^{N_x} \mu_{x,j}(l_i,\ a_i,\ \mathbf{h}_i)\eta_j, \tag{3.15}$$

and $\boldsymbol{\Sigma}_x$ is a co-variance matrix with the $(i,j)$-th, $\forall i,\ j = 1,\ \cdots,\ N_P$ element given by

$$\Sigma_{q,r} = \sum_{j=1}^{N_x} \sigma^2_{j,x}(l_i,\ a_i,\ \mathbf{h}_i)\eta_{jq}\eta_{jr}, \tag{3.16}$$

in which $\eta_{jq}$ and $\eta_{jr}$ are respectively the $q$-th and $r$-th element of the $j$-th basis $\eta_j$. The likelihood function $f(\mathbf{u}_z|l_i,\ a_i,\ \mathbf{h}_i)$ of observing $\boldsymbol{u_z}$ is computed similarly to $\boldsymbol{u_x}$,

$$f(\mathbf{u}_z|l_i,\ a_i,\ \mathbf{h}_i) = \frac{\exp\left(-0.5(\mathbf{u}_z - \mu_z)^{\mathrm{T}} \boldsymbol{\Sigma}_z^{-1} (\mathbf{u}_z - \mu_z)\right)}{\sqrt{(2\pi)^{N_P} |\boldsymbol{\Sigma}_z|}}. \tag{3.17}$$

The focus of this research with regard to image monitoring data is its utilization for diagnosis and prognosis. Therefore, a simplified transformation from 3D to image coordinates is presented and synthetic camera measurements are generated. The accurate

collection of camera displacement measurements to achieve the research's diagnosis and prognosis results is left to future work. In particular, future work would define the correct transformation from the FE displacement results into image displacements. Then, a fatigue experiment on a cruciform similar to Figure 3.9 here could be observed using high resolution cameras. The diagnosis and prognosis proposed in this research could be applied to estimate the crack length and parameters. Then this research could be validated against other experimental techniques.

The two types of the observations based on the miter gate physics are now fully described. We next consider the integration of multi-mode damage diagnosis and failure prognosis using a DBN framework.

### 3.4.2 SHM Using DBN

#### DBN for miter gates with multiple failure modes

We assume that there is uncertainty from noise in the two data sources, i.e., sensor noise and camera image quality; thus, a dynamic Bayesian network (DBN) is constructed which accommodates measurement uncertainty of observations along with probabilistic transitions of damage modes over time. Fig. 3.11 shows the feature of interest represented by different types of DBN nodes and their connections at two consecutive time steps ($t_{i-1}$ and $t_i$). The continuous nodes represents the two state variables that quantify the two failure modes of miter gate at time step $t_i$, referred to as $l_i$ and $a_i$. The observed nodes described the measurements associated with the unknown nodes. Besides the strain reading $S^{obs}$ and digital images $I^{obs}$, the upstream and downstream water levels can be also measured at any time step. Thus, the hydrostatic pressure condition $\mathbf{h}$ is assumed to be observable and static at each time step without measuring error (The staff gauge measurement error is so low that it is ignored in this study). The arrows, meanwhile, indicate the probabilistic connection and interaction between different variables, i.e., the dashed lines represent the connection between continuous nodes in two consecutive time

steps and the solid lines represent the interaction between nodes in individual time step. For example, the crack length $a_i$ is dependent on not only the crack length at previous time step, i.e., $a_{i-1}$, but also the crack increment that dominated by Paris' law, i.e., $c$, $m$, $\Delta K_i$. Table 3.1 summarizes the variables of the DBN.

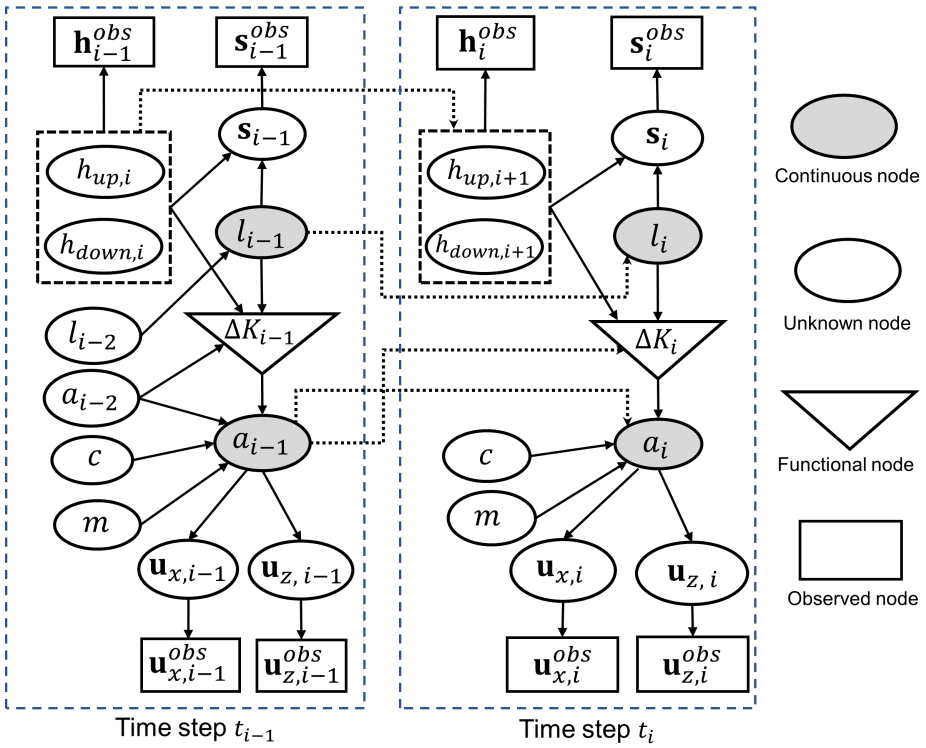



**Figure 3.11.** Dynamic Bayesian network for miter gate with multi-failure modes.

### Surrogate-based IGL in DBN

As mentioned above, the physics of the crack is simulated by the IGL algorithm, which provides the SIF value at any time step for the Paris' law. However, a single run of getting the SIF result from the IGL global and local analysis takes up to 10 minutes due to the complex local XFEM model. Generally, probabilistic analysis for damage diagnostics and failure prognostics, such as Bayesian updating and uncertainty propagation, requires the model to be executed thousands of times. Thus, for a fast yet accurate calculation of SIF given the parameters $\mathbf{h}$, $l_i$ and $a_i$, a surrogate-based IGL (SIGL) algorithm is

**Table 3.1.** Nomenclature for the DBN.

| Symbol | Parameter explanation |
|---|---|
| $a_i$ | Crack length at time step $t_i$ |
| $l_i$ | Gap length at time step $t_i$ |
| $\Delta K$ | Stress intensity factor range |
| $\mathbf{h^{obs}}$ | Load observation |
| $c$, $m$ | Paris' law parameter |
| $\mathbf{s^{obs}}$ | Strain measurement |
| $\mathbf{u_x^{obs}}$ | Displacement measurement in $x$ direction |
| $\mathbf{u_z^{obs}}$ | Displacement measurement in $z$ direction |
| $\theta$ | Interested parameters |
| $\mathbf{y}$ | Jointed observations |
| $\alpha$, $\beta$ | coefficients for the likelihood functions |
| $N_p$ | Total number of particles |
| $w_j$ | Weight of $j$-th particle |

used for the purpose of computational efficiency. Algorithm 1 presents a pseudocode of surrogate-based IGL method. Details of the SIGL method are available in Fillmore et al. (2022).

As shown in Fig 3.12, the global FE analysis is accelerated by using static condensation (denote FastGlobal Algorithm 2) where the global displacement along global-local boundary is obtained directly from a static-condensed matrix; while the local FE analysis is replaced by GP-based surrogate model (denote SurrogateLocal in Algorithm 2). Such setup shortens the computational time for one global-local simulation from 10 minutes to less than 0.1 seconds, enabling damage diagnostics and failure prognostics.

$$\Delta K_i = G_{SIGL}(l_i,\ a_i,\ \Delta s_i), \tag{3.18}$$

where $G_{SIGL}(l_i,\ a_i,\ \Delta s_i)$ is the SIGL algorithm that enables a fast calculation of SIF range $\Delta K_i$.

With the capability of acquiring SIFs via SIGL model in affordable amounts of time, the process of using a dynamic Bayesian network (DBN) with two synthetic observations

**Algorithm 2.** Surrogate-Based Iterative global-local (SIGL) algorithm

**procedure** SIGL(tolerance,$m$,$\mathbf{f}^G$,$\mathbf{f}^L$) ▷ $\mathbf{f}^G$ and $\mathbf{f}^L$ are glob. and loc. load vectors
  Arbitrary initialization $\mathbf{p}_0$
  Arbitrary initialization $\omega_0 \approx 1.0$
  **for** $j \in [0, ..., m]$ **do**
    $\mathbf{u}_j^G = \text{FastGlobal}(\mathbf{p}_j; \mathbf{f}^G)$
    $\boldsymbol{\lambda}_j^L = \text{SurrogateLocal}(\mathbf{u}_j^G; \mathbf{f}^L)$
    $\boldsymbol{\lambda}_j^G = \text{FastGlobal}(\mathbf{u}_j^G; \mathbf{f}^G)$
    $\mathbf{r}_j = -\left(\boldsymbol{\lambda}_j^L + \mathbf{p}_j - \boldsymbol{\lambda}_j^G\right)$
    $e_j = ||r_j||_\infty$
    **if** $e_j <$ tolerance **then**
      exit for loop
    **end if**
    $\mathbf{p}_{j+1} = \mathbf{p}_j + \mathbf{r}_j$
  **end for**
  $\Delta K = \text{SurrogateSIF}(\mathbf{u}_m^G; \mathbf{f}^L)$
**end procedure**

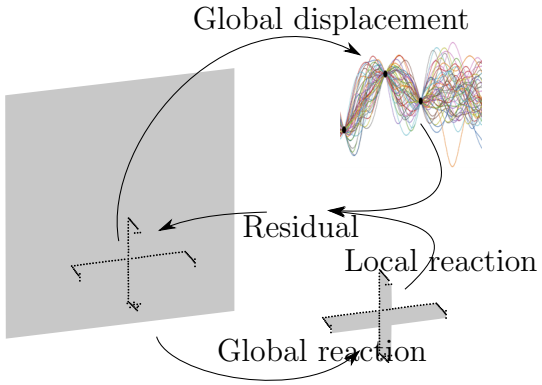



**Figure 3.12.** Surrogate-based IGL with global static condensation.

is represented in the following section.

### Diagnostics and prognostics of multiple failure modes with DBN and maintenance optimization

Based on the surrogate modeling, functional representation, and probabilistic modeling of different nodes in the DBN, we now present the diagnostics and prognostics of miter gates with multiple failure modes using the DBN and multiple data sources (i.e.

strain measurements and image monitoring data).

**(a) Damage diagnostics with DBN**

Under the Bayesian updating framework, the damage states including gap degradation $l_i$ and crack length $a_i$ at $t_i$ are estimated along with the uncertain crack growth model parameters as follows

$$\begin{aligned} & f(c_i,\ m_i,\ l_i,\ a_i|\mathbf{y}_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs}) \\ & = \frac{f(\mathbf{y}_i^{obs}|\mathbf{h}_i^{obs},\ \theta_i)f'(\theta_i)}{\int\cdots\iint f(\mathbf{y}_i^{obs}|\mathbf{h}_i^{obs},\ \theta_i)f'(\theta_i)\mathbf{d}\theta_i} \\ & \propto f(\mathbf{y}_i^{obs}|\mathbf{h}_i^{obs},\ \theta_i)f'(\theta_i), \end{aligned} \tag{3.19}$$

where

$$\begin{aligned} \theta_i &\doteq [c_i,\ m_i,\ l_i,\ a_i] \\ \mathbf{y}_{1:i}^{obs} &\doteq [\ \mathbf{s}_{1:i}^{obs},\ \mathbf{u}_{x,1:i}^{obs},\ \mathbf{u}_{z,1:i}^{obs}] \\ \mathbf{y}_i^{obs} &\doteq [\ \mathbf{s}_i^{obs},\ \mathbf{u}_{x,i}^{obs},\ \mathbf{u}_{z,i}^{obs}] \end{aligned} \tag{3.20}$$

$f(\mathbf{y}_i^{obs}|\mathbf{h}_i^{obs},\ \theta_i)$ is the likelihood function of observing the two types of data sources (i.e. strain measurements and displacement images), and $f'(\theta_i)$ is the prior distribution at time $t_i$ given by

$$\begin{aligned} & f'(\theta_i) = f(c_i,\ m_i,\ l_i,\ a_i|\mathbf{y}_{1:i-1}^{obs},\ \mathbf{h}_{1:i-1}^{obs}), \\ & = \int\cdots\iint f(\theta_i|\theta_{i-1})f(\theta_{i-1}|\mathbf{y}_{1:i-1}^{obs},\ \mathbf{h}_{1:i-1}^{obs})\mathbf{d}\theta_{i-1}, \end{aligned} \tag{3.21}$$

in which $f(\theta_i|\theta_{i-1})$ represents the state transition between two time steps.

Considering the two different types of data sources and according to the graphic model given in Fig. 3.11, the likelihood function $f(\mathbf{y}_i^{obs}|\mathbf{h}_i^{obs},\ \theta_i)$ is computed based on the

chain rule of Bayesian networks as follows

$$\begin{aligned} & L(\mathbf{s}_i^{obs},\ \mathbf{u}_{x,i}^{obs},\ \mathbf{u}_{z,i}^{obs}|\mathbf{h}_i^{obs},\ c_i,\ m_i,\ l_i,\ a_i) \\ & = f(\mathbf{s}_i^{obs}|\mathbf{h}_i^{obs},\ l_i) f(a_i|\mathbf{h}_i^{obs},\ l_i,\ c_i,\ m_i) \\ & \quad \times f(\mathbf{u}_{x,i}^{obs}|\mathbf{h}_i^{obs},\ l_i,\ a_i) f(\mathbf{u}_{z,i}^{obs}|\mathbf{h}_i^{obs},\ l_i,\ a_i), \end{aligned} \tag{3.22}$$

where $f(\mathbf{s}_i^{obs}|\mathbf{h}_i^{obs},\ l_i)$ is given in Eq. (3.10), $f(\mathbf{u}_{x,i}^{obs}|\mathbf{h}_i^{obs},\ l_i,\ a_i)$ and $f(\mathbf{u}_{z,i}^{obs}|\mathbf{h}_i^{obs},\ l_i,\ a_i)$ are given in Eq. (3.14), and $f(a_i|\mathbf{h}_i^{obs},\ l_i)$ is obtained through uncertainty propagation using the surrogate-based IGL method, which first propagates the uncertainty of $l_i$ to the uncertainty of SIF range $\Delta K_i$ using Eq. (3.18) and then to crack length $a_i$ using Eq. (3.5).

In this chapter, the particle filter (PF) is used as the Bayesian inference algorithm which enables a quantitative way to track and evaluate the evolution of the state variables in the DBN. The PF is designed to achieve an optimum estimate of the posterior probability density functions $f(l_i|S_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs})$ and $f(a_i|S_{1:i}^{obs},\ I_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs})$ based on observations $S_{1:i}^{obs}$, $I_{1:i}^{obs}$, and $\mathbf{h}_{1:i}^{obs}$. It starts with prior samples of state variables in the network. For the first time step, the prior samples are generated according to empirical research and prior knowledge of the physics. For the other time steps, the prior samples are obtained through uncertainty propagation from the previous time step (i.e. Eq. (3.21)).

Assuming that $N_p$ particles are generated at each time step, we have the particles of the state variables at $t_i$ as

$$\begin{aligned} & \theta_i^p \doteq [\mathbf{l}_i^p,\ \mathbf{a}_i^p,\ \mathbf{c}_i^p,\ \mathbf{m}_i^p], \\ & \mathbf{l}_i^p \doteq [l_{i1}^p,\ \cdots,\ l_{iN_p}^p];\ \mathbf{a}_i^p \doteq [a_{i1}^p,\ \cdots,\ a_{iN_p}^p]; \\ & \mathbf{c}_i^p \doteq [c_{i1}^p,\ \cdots,\ c_{iN_p}^p];\ \mathbf{m}_i^p \doteq [m_{i1}^p,\ \cdots,\ m_{iN_p}^p]; \end{aligned} \tag{3.23}$$

in which $a_{ij}^p, a_{ij}^p, c_{ij}^p,$ and $m_{ij}^p, \forall j = 1, \cdots, N_p$ is the $j$-th particle at $t_i$.

The likelihood function of each particle is then computed using Eq. (3.22) as

$$\begin{aligned}
&\log L(\mathbf{s}_i^{obs},\ \mathbf{u}_{x,i}^{obs},\ \mathbf{u}_{z,i}^{obs}|\mathbf{h}_i^{obs},\ \theta_{ij}) \\
&= \alpha \cdot \log(f(\mathbf{s}_i^{obs}|\mathbf{h}_i^{obs},\ l_{ij})) \\
&+ \beta \cdot [\log(f(\mathbf{u}_{x,i}^{obs}\ |\mathbf{h}_i^{obs},\ \theta_{ij})) \\
&+ \log(f(\mathbf{u}_{z,i}^{obs}\ |\mathbf{h}_i^{obs},\ \theta_{ij}))],
\end{aligned} \tag{3.24}$$

where $\alpha$ and $\beta$ are the coefficients for two likelihood functions. When the importance of two measurements are equally considered, $\alpha = 1$ and $\beta = 1$, respectively. When only image data is considered, $\alpha = 0$ and $\beta = 1$.

Based on the above likelihood function, the weight of each particle is computed by

$$\begin{aligned}
w_j &= \frac{L(\mathbf{s}_i^{obs},\ \mathbf{u}_{x,i}^{obs},\ \mathbf{u}_{z,i}^{obs}|\mathbf{h}_i^{obs},\ \theta_{ij})}{\sum_{j=1}^{N_p} L(\mathbf{s}_i^{obs},\ \mathbf{u}_{x,i}^{obs},\ \mathbf{u}_{z,i}^{obs}|\mathbf{h}_i^{obs},\ \theta_{ij})}, \\
&\forall j = 1,\ \cdots,\ N_p.
\end{aligned} \tag{3.25}$$

The joint posterior distribution given in Eq. (3.19) is then approximated based on the particles based on re-sampling using the weights given in Eq. (3.25) as

$$f(c_i,\ m_i,\ l_i,\ a_i|\mathbf{y}_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs}) \approx \sum_{j=1}^{N_p} w_j \delta_{\theta_i}, \tag{3.26}$$

where $\delta_{\theta_i}$ is a delta function at $\theta_i = [c_i, m_i, l_i, a_i]$.

Let the posterior particles of $l$, $a$, $c$, and $m$ at $t_i$ after re-sampling be $\mathbf{l}''_i = [l''_{i1},\ \cdots,\ l''_{iN_p}]$, $\mathbf{a}''_i = [a''_{i1},\ \cdots,\ a''_{iN_p}]$, $\mathbf{c}''_i = [c''_{i1},\ \cdots,\ c''_{iN_p}]$, and $\mathbf{m}''_i = [m''_{i1},\ \cdots,\ m''_{iN_p}]$. These particles are then used to obtain the prior samples for $t_{i+1}$ based on state transition probability $f(\theta_i|\theta_{i-1})$. For unknown constant parameter such as $c$ and $m$, a very small noise amount is added to prevent particle degeneration during PF implementation. The

transition equations are defined as below:

$$\begin{aligned} c^p_{(i+1)j} &= c''_{ij} + \varepsilon_{c,i+1}, \\ m^p_{(i+1)j} &= m''_{ij} + \varepsilon_{m,i+1}, \end{aligned} \tag{3.27}$$

in which $c_{(i+1)j}$ and $m_{(i+1)j}$ are respectively the $j$-th prior sample of $c$ and $m$ at $t_{i+1}$, $\varepsilon_{c,i+1}$ and $\varepsilon_{m,i+1}$ are very small noises to avoid sample degeneration as mentioned above.

For the gap length state variable, the posterior samples of $l_i$ is used to obtain the prior samples at $t_{i+1}$ as

$$\begin{aligned} l^p_{(i+1)j} &= l''_{ij} + exp(\sigma u_j) Q (l''_{ij})^w, \\ \forall j &= 1, \cdots, N_p \end{aligned} \tag{3.28}$$

where $u_j$ is a random sample of a standard normal random variable.

For state variable $a_i$, as shown in Fig. 3.11, the posterior samples are first passed through Eq. (3.18) (i.e. a functional node) to obtain samples of $\Delta K_i$. The prior samples $a^p_{(i+1)j}$ are then obtained through Eq. (3.5) given in Sec. 3.3. The above process (i.e. Eqs. (3.19) through (3.28)) is implemented recursively over time to perform damage diagnostics of miter gate with multiple failure modes.

**(b) Failure prognostics with DBN**

Failure prognostics is a process of predicting the remaining useful life (RUL) of structural assets based on all the information available at the current time step. The RUL information gives the engineers insight into life-cycle maintenance. Figure 3.13 shows an illustration of how to perform RUL prediction based on failure prognostics.

Based on the state estimation from failure diagnostics at time step $t_i$, the end of life (EOL) can be determined which is defined as the intersection point between feature limit state and predicted curve of damage growth path.

Through particles obtained at time step $t_i$, a family of degradation curves can be

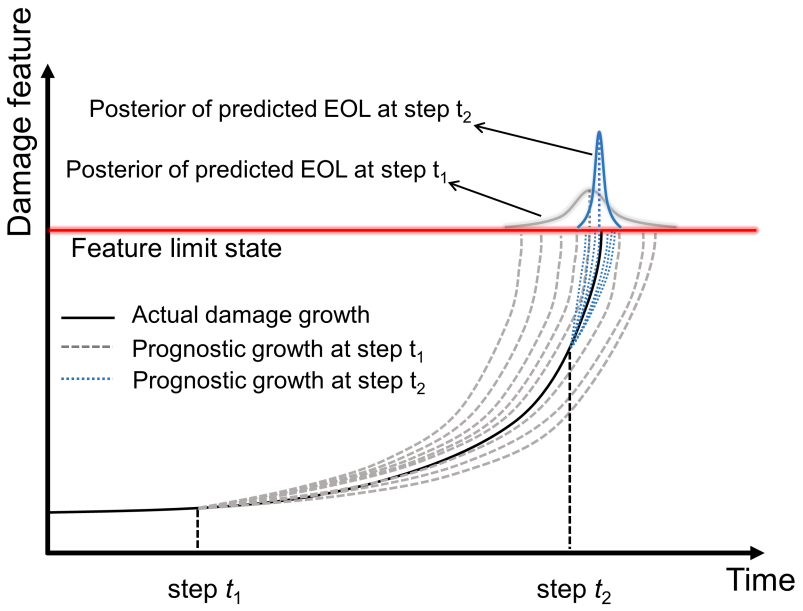



**Figure 3.13.** Illustration of EOL to obtain RUL prediction based on failure prognostics

obtained as illustrated in Fig. 3.13. Based on that, a distribution of EOL of the structures can be approximated by collecting all the intersection points. The RUL is determined as the difference between EOL and the current time step. The probability that the RUL at $t_i$ is less than $p$ conditioned on current observations is given by

$$\begin{aligned} &\Pr\{RUL_{l,i} \leq p | \mathbf{y}_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs}\} \\ &= \int \Pr\{l_{i+p} \geq l_e | l_i\} f(l_i | \mathbf{y}_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs}) dl_i, \end{aligned} \tag{3.29}$$

where $RUL_{l,i}$ is the RUL at time step $t_i$ for failure mode of gap degradation, $l_e$ is the failure threshold of gap length and $f(l_i|\mathbf{y}_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs})$ is the posterior distribution of gap length at time step $t_i$.

Eq. (3.29) is approximated using the Monte Carlo simulation method based on the

posterior particles from DBN as follows

$$\begin{aligned}&\Pr\{RUL_{l,i} \leq p|\mathbf{y}_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs}\}\\&\approx \frac{1}{N_p}\sum_{k=1}^{N_p}\Lambda(l_{(i+p)k} \geq l_e|l_{ik}''),\end{aligned} \tag{3.30}$$

in which $N_p$ is the number of particles in the inference using DBN, $l_{ik}''$ is the $k$-th posterior particle of gap length at $t_i$, and $\Lambda(l_{(i+p)k} \geq l_e|l_{ik}'') = 1$ if $l_{(i+p)k} \geq l_e|l_{ik}''$ is true, otherwise $\Lambda(l_{(i+p)k} \geq l_e|l_{ik}'') = 0$, and $l_{(i+p)k} \geq l_e|l_{ik}''$ stands for a trajectory of random gap growth curve conditioned on initial state $l_{ik}''$ as indicated in Fig. 3.13.

Similarly, the RUL at $t_i$ corresponding to failure mode of fatigue crack is estimated by

$$\begin{aligned}&\Pr\{RUL_{a,i} \leq p|\mathbf{y}_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs}\}\\&\approx \frac{1}{N_p}\sum_{k=1}^{N_p}\Lambda(a_{(i+p)k} \geq a_e|l_{ik}'',\ a_{ik}'',\ c_{ik}'',\ m_{ik}''),\end{aligned} \tag{3.31}$$

where $RUL_{a,i}$ is the RUL corresponding to crack at $t_i$, $a_e$ is the failure threshold for fatigue crack, and $a_{ik}'',\ c_{ik}'',\ m_{ik}''$ are the $k$-th posterior particle.

The overall system RUL is then obtained based on Eq. (3.29) through (3.31) as

$$\begin{aligned}&\Pr\{RUL \leq p|\mathbf{y}_{1:i}^{obs},\ \mathbf{h}_{1:i}^{obs}\}\\&= \int F_{i|\mathbf{y}_{1:i}^{obs},\,\mathbf{h}_{1:i}^{obs}}(p)]\\&\approx \frac{1}{N_p}\sum_{k=1}^{N_p}\Lambda(l_{(i+p)k} \geq l_e|l_{ik}''\\&\qquad \cup\ a_{(i+p)k} \geq a_e|l_{ik}'',\ a_{ik}'',\ c_{ik}'',\ m_{ik}''),\end{aligned} \tag{3.32}$$

where $F_{i|\mathbf{y}_{1:i}^{obs},\,\mathbf{h}_{1:i}^{obs}}(p)]$ is the failure probability in the future $p$ time steps conditioned on observations $\mathbf{y_{1:i}^{obs}}$ and $\mathbf{h_{1:i}^{obs}}$, and "$\cup$" indicates "union" of two events which means that

the gate fails if either one of the two failure modes occurs.

**(c) Optimal maintenance planning based on failure prognostics**

The RUL estimation in failure prognostics provides an informative way of understanding how damage progresses in time. Consequently, maintenance decisions may be optimized and updated based on the structural condition assessment. In this chapter, the cost per unit of time (CPUT) is employed for maintenance optimization based on failure prognostics. CPUT is a cost function proposed by Barlow and Hunter (1960), which defines the cost of performing preventive maintenance at time $t$ as

$$CPUT(t) = \frac{C_p[1 - F_i(t)] + C_u[F_i(t)]}{[\int_0^t [1 - F_i(\tau)]\, d\tau]}, \tag{3.33}$$

where $C_p$ is the preventative action cost, $C_u$ is the unplanned action cost, and $F_i(t)$ is the failure probability given in Eq. (3.32) (i.e., $F_{i|\mathbf{y}_{1:i}^{obs},\, \mathbf{h}_{1:i}^{obs}}(p)$). Note that Eq. (3.33) is meaningful only if the cost ratio, $C_u/C_p$, is greater than 1, otherwise no maintenance operation is needed. It is suggested in Vega et al. (2020) that the corresponding cost ratio for some miter gates is close to 5. A larger cost ratio would represent the case that unplanned failure may have a much more severe consequence cost compared to preventative action. The optimal time for maintenance planning is then defined as the time when CPUT is minimized, given the different values of $C_p$ and $C_u$. In addition, the optimal maintenance time is decreasing over time as suggested in Vega et al. (2020).

Next, we will use a case study to illustrate the proposed framework for damage diagnostics and failure prognostics of multi-mode failure using multiple data sources.

## 3.5 Case Study

### 3.5.1 Prior Information and Measurements

With the above formulated training process, the test case is carried below. In this chapter, the physical value of one time step is set to be one month. The true values of parameters $c$ and $m$ are set as $c = 3 \times 10^{-4}$ and $m = 2.2$, respectively. The number of particles in the PF is set as $N_p = 50,000$. Based on our best engineering assumptions, the truncated uniform prior distributions of $c$ and $m$ are set as $c \sim U[1 \times 10^{-4}, 1 \times 10^{-3}]$ and $m \sim U[1, 3]$, where $U[lb, ub]$ represents uniform distribution with lower bound $lb$ and upper bound $ub$. The initial gap length and crack length are set as $l_0 = 50$ inches and $a_0 = 1$ inch, respectively. Fig. 3.14 shows the gap and crack growth curves used to generate synthetic data. The failure thresholds of gap length and crack length are set to $l_e = 100$ inches and $a_e = 3$ inches. Correspondingly, the true EOLs are determined as 82.7 months and 87.7 months, respectively. Note that the true EOLs for the two damage features are selected on purpose to have similar values, in order to show the performance of damage prognostics using jointed observations.

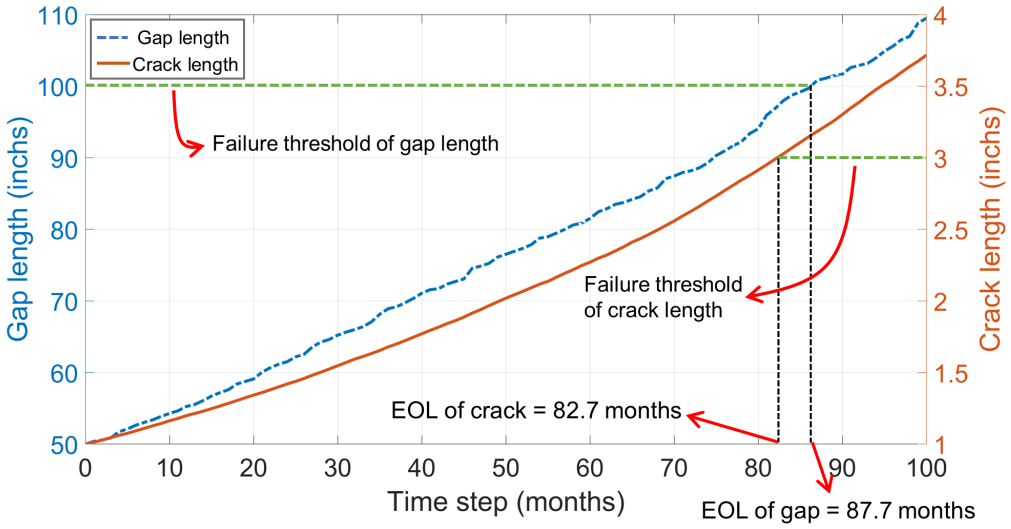



**Figure 3.14.** True gap and crack growth curves for synthetic data generation.

The true states of the two failure modes are assumed to be unknown during the diagnosis and prognosis. To validate the proposed DBN framework, two sets of synthetic measurements are firstly generated based on the structure under crack and quoin block degradation. Figure 3.15 presents 1000 readings of the four strain gauges obtained based on the synthetic gap data given in Fig. 3.14 and water level data where $h_{up} \sim N(550, 20)$, $h_{down} \sim N(150, 20)$. Figure 3.16 depicts the displacement measurements at each time step with a pixel size of 0.5 inches where the crack grows from 0.5 inches to 4 inches. As indicated in this figure, the displacement in the $z$ direction increases with the growth of crack length, which is manifested in the displacement images as more red colors in the upper part and more blue colors in the lower part (surface fractures into opposite directions).

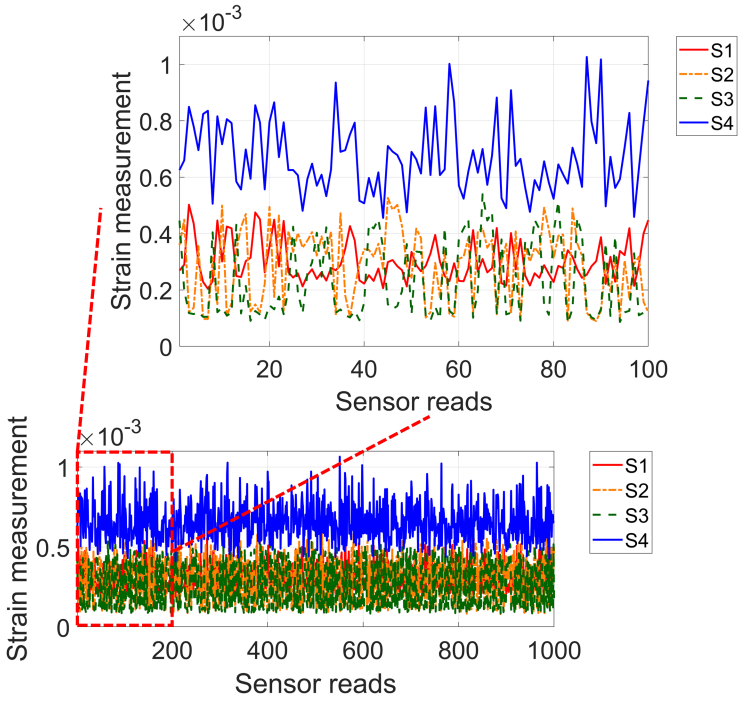



**Figure 3.15.** Synthetic strain measurements from the four sensor locations.

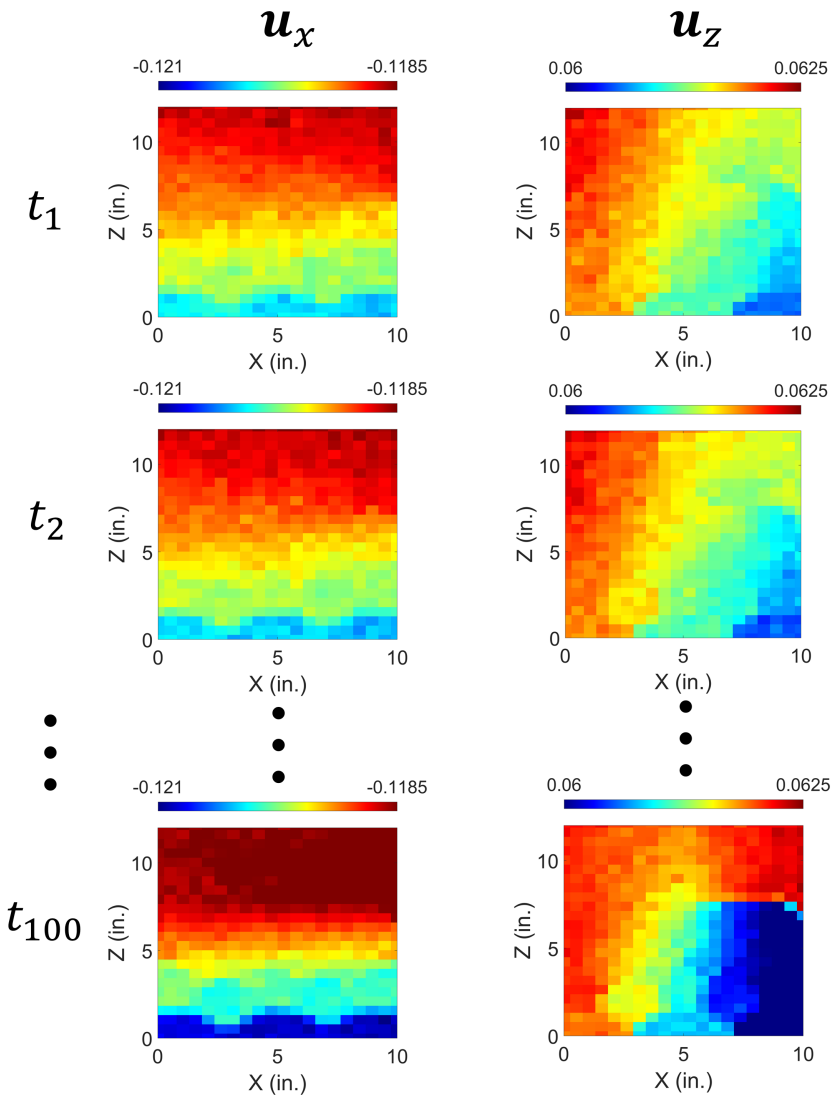



**Figure 3.16.** Displacement measurement with pixel size 0.5 *in.* and Gaussian noise.

### 3.5.2 Results and Discussion

Based on the synthetic data presented in Sec.3.5.1, the DBN model takes in the two measurements to calculate corresponding likelihood functions base on their weights of importance. By setting the coefficients for the two likelihood functions, $\alpha$ and $\beta$, the distributions of the state variables are updated at each time step. Figure 3.17 presents the diagnostic result of the two damage features, crack length $a$ and quoin damage length $l$, when different measurement inputs of DBN are defined. In Fig. 3.17 a) and b), two

types of observations are used, i.e., $\alpha = 1$ and $\beta = 1$. The mean prediction and the 95% confidence intervals suggest that both two variables $a$ and $l$ are estimated with high accuracy. For the case when only displacement data are available ( $\alpha = 1$ and $\beta = 1$) as shown in Fig. 3.17 c) and d), the proposed damage estimation method is still able to accurately estimate the crack length length. However, the accuracy of the gap length estimation significantly drops, reflected by the error of mean prediction and increased confidence intervals. While the images taken far from the bottom quoin are not sensitive enough to detect the quoin block deterioration compared to the strain measurements, incorporating multiple data sources with different sensitivities to damages features are essentially required to obtain accurate prediction.

The failure prognosis depends upon the target damage feature and the definition of failure. The following prognostic results are carried out based on considering crack and quoin block damage individually and considering two damage features jointly. Figure 3.18 shows the prognostic result when considering the crack only. The true RUL of the structure is 82.7 months. Four cases are shown here which represent the four stages of the structural life cycle: Figure 3.18 a) is an early stage of the crack initiation, where the prognostic result overestimates the RUL of the structures by 10 months. The error between mean prediction and true value is improved to around 1 month after 20 months, shown in 3.18 b). At the 60th month and 80th month, the prediction becomes more and more accurate.

The prognostic result when considering gap length only is shown in Fig. 3.19. The true RUL of the structure is 87.5 months based on quoin block damage. In this case, the prediction is consistently accurate, as the prediction of the gap length follows the true RUL in all life stages.

Figure 3.20 shows the prognostic result when the joint failure threshold is determined as the smaller of the crack and gap length damage limits. The predicted RUL outperformed both results of using a single failure threshold. In the first stage (before 35 months), the

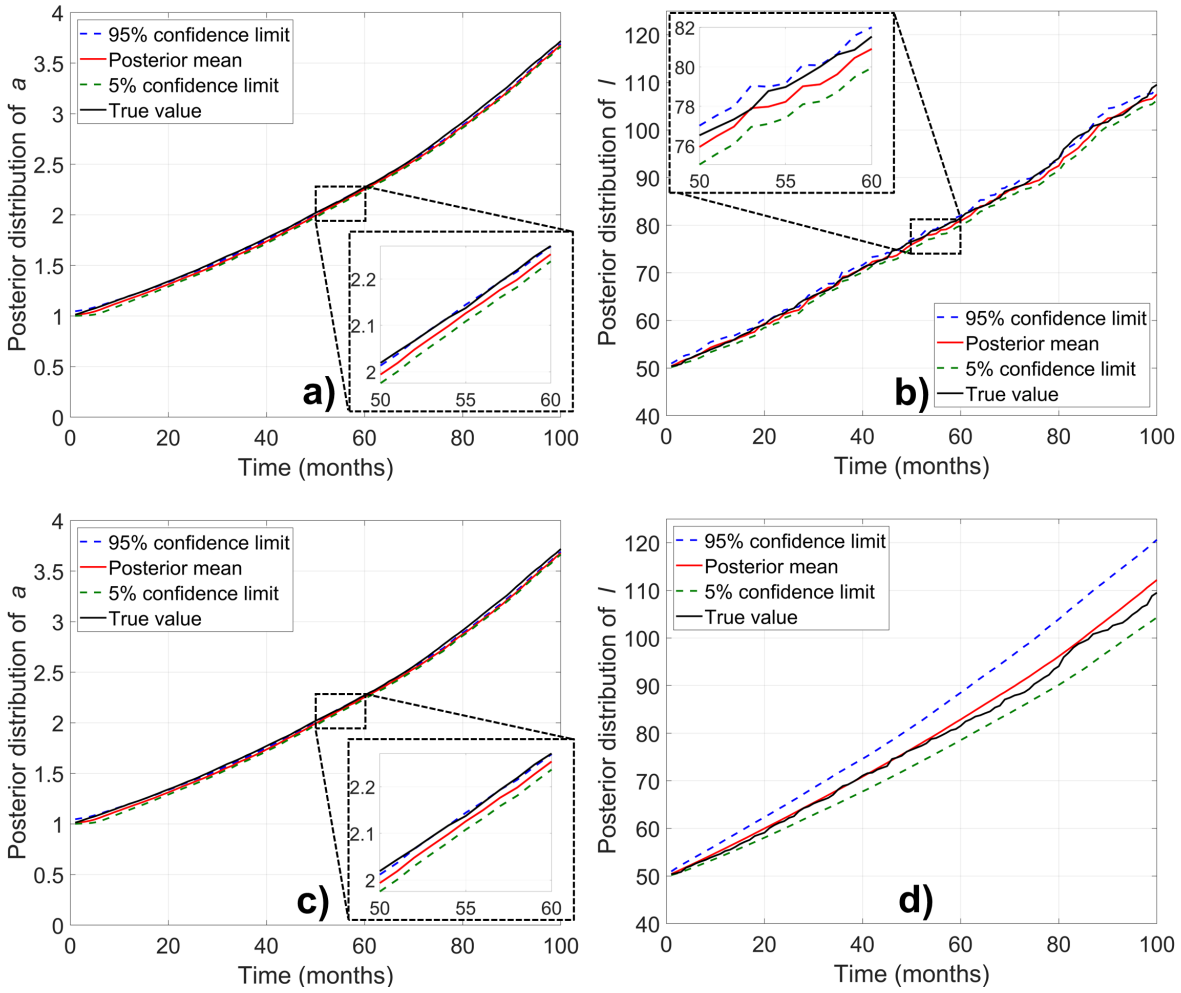



**Figure 3.17.** Diagnostic results with different inputs: a) Posterior distribution of $a$ using both $S^{obs}$ and $I^{obs}$, b) posterior distribution of $l$ using both $S^{obs}$ and $I^{obs}$, c) posterior distribution of $a$ using $I^{obs}$ only, and d) posterior distribution of $l$ using $I^{obs}$ only.

prediction slightly overestimates the RUL of the structure; in the second stage (after 35 months), the model tends to be more conservative about the prediction as the predicted RULs are smaller than the true values. Such a pattern will lead to different risk-based life-cycle managements during the optimal maintenance planning process, considering the different combinations of preventative action cost and unplanned action cost.

Figure 3.21 shows the overall RUL estimation at each time step and its confidence limits when considering the crack as the only damage feature. Although the prediction of the crack is very inaccurate in the early months, the model manages to converge the

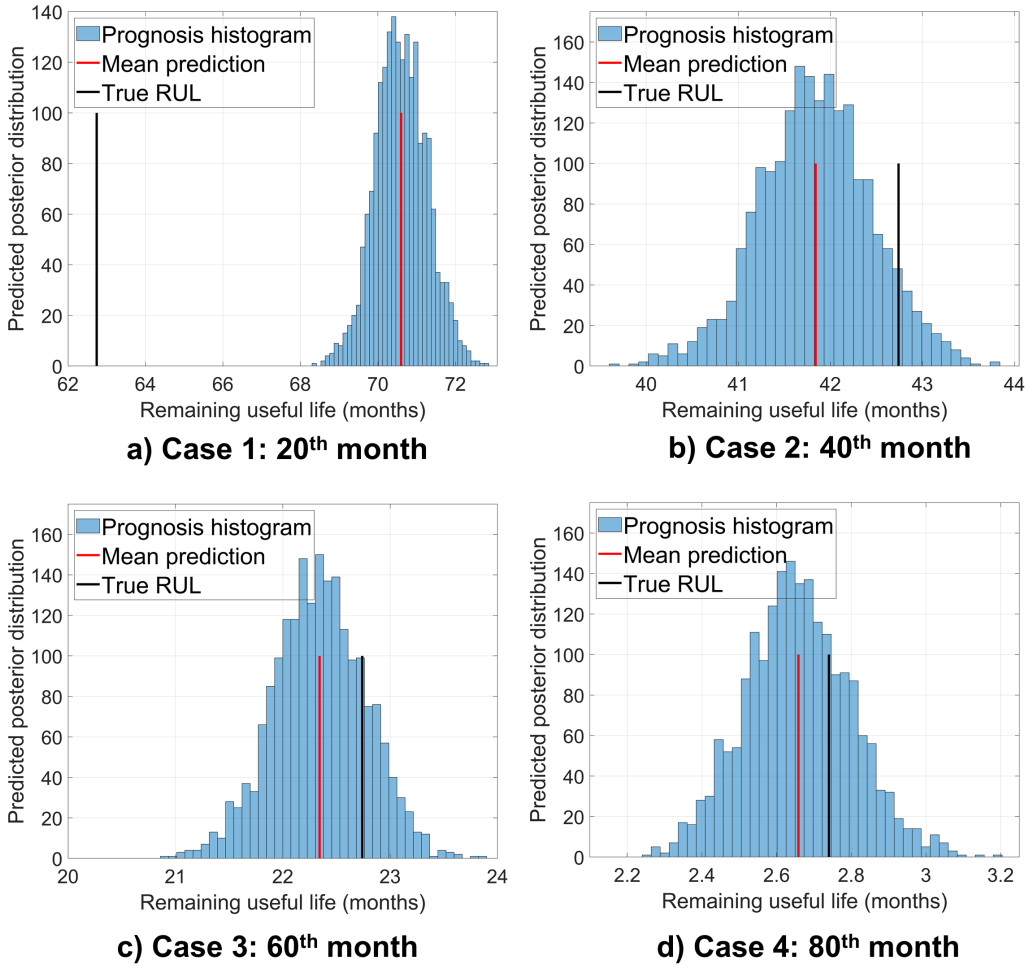



**Figure 3.18.** RUL results based on crack prognostics only.

prediction to the true value after around 30 months with a high confidence level.

Figure 3.22 shows the RUL estimation at each time step and its confidence limits when considering gap as the only damage feature. The gap prediction fluctuates around the true gap RUL, and both prediction error and confidence limit converge to at the final time step.

Considering both the crack and the gap as damage features, Fig. 3.23 shows the RUL estimation at each time step and its confidence limits. Similarly, the prediction outperformed both results of using single failure threshold when the EOL is jointly determined from the two damage features.

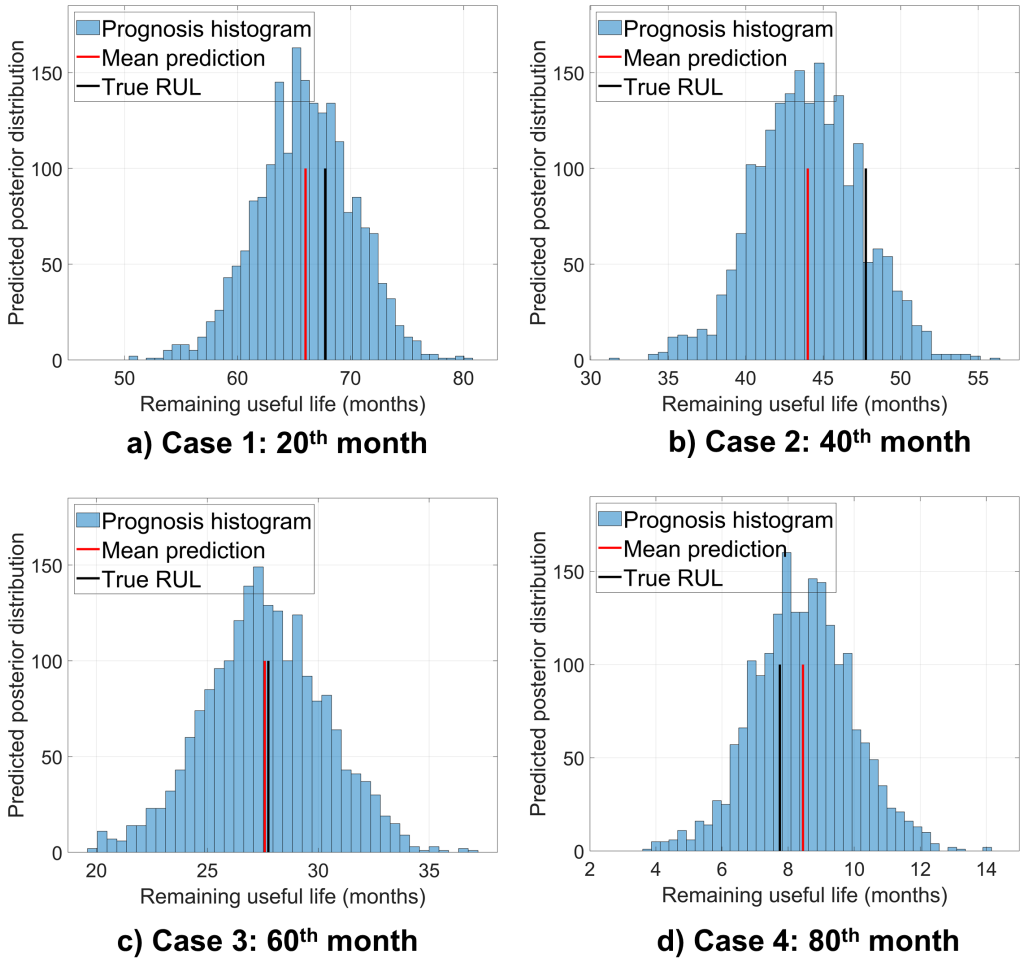



**Figure 3.19.** RUL results based on gap prognostics only.

The RUL prediction from failure prognostics is actually related to the reliability. Based on the reliability function obtained from predicted RULs at each time step, the CPUT can be calculated and updated as time evolves. Figure 3.24 shows CPUT at time step 50th month with different cost ratios. It can be seen that as the unplanned action cost grows, i.e., $C_u$ increases, the optimal maintenance time decreases, and the corresponding CPUT becomes stable at a large value.

To understand the impact of different monitoring techniques (e.g., strain gauges and camera images) on decisions related to maintenance planning, the optimal maintenance time and minimum CPUT are calculated based on the prognostics results using measurements

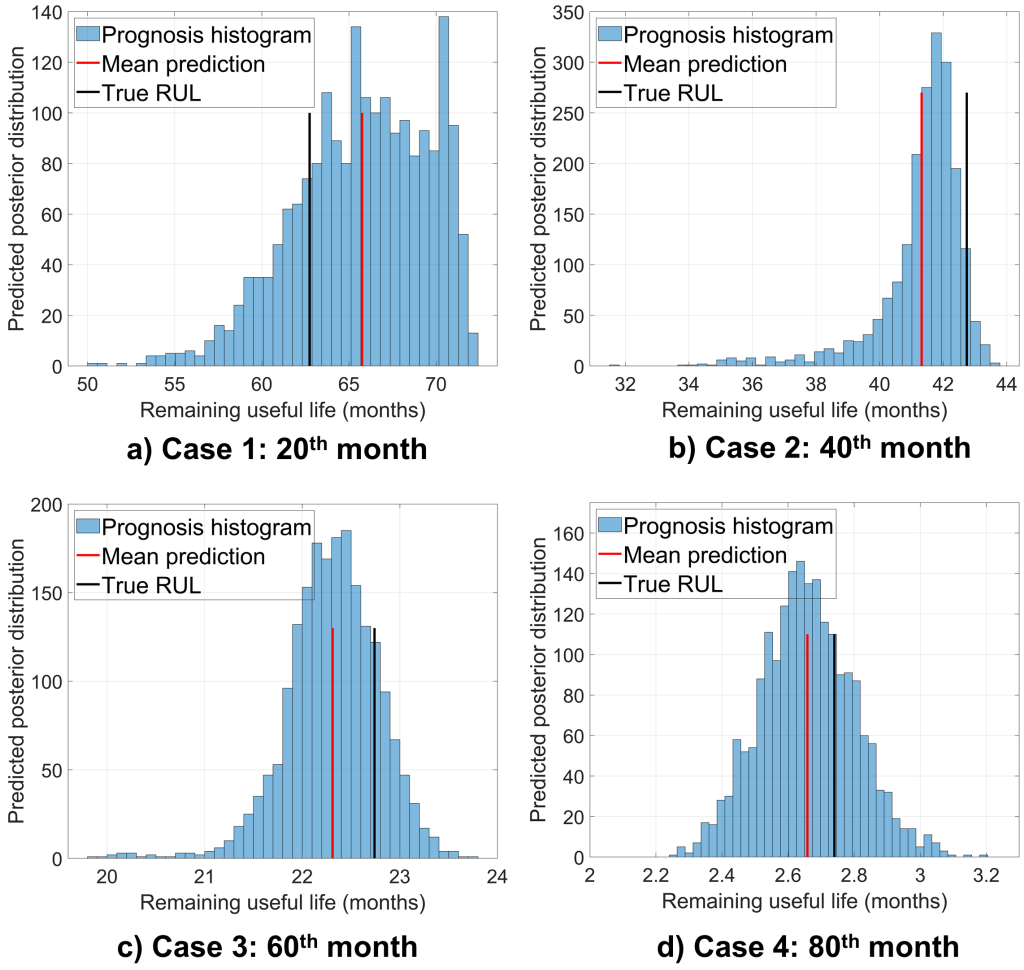



**Figure 3.20.** RUL results based on jointed failure threshold.

from both monitoring techniques and from camera images only. Figure 3.25 shows how the optimal maintenance time (i.e., the time when CPUT is minimized) are updated from the measurements over time, when $C_u = 1$ and $C_p = 50$. The vertical line in the figure represents the true end of life, which is the time that one of the two competing damage features first reaches its failure threshold. As noted, the two curves of optimal maintenance time are very similar, which is due to the high accuracy of the failure prognostics results. Figure 3.26 a) shows the minimum CPUT when $C_u = 1$ and $C_p = 50$. By zooming into the curve after 65 months, the result clearly proves that the uncertainty in Fig. 3.17 d) consequently leads to a higher minimum CPUT compared to that of Fig. 3.17 b).

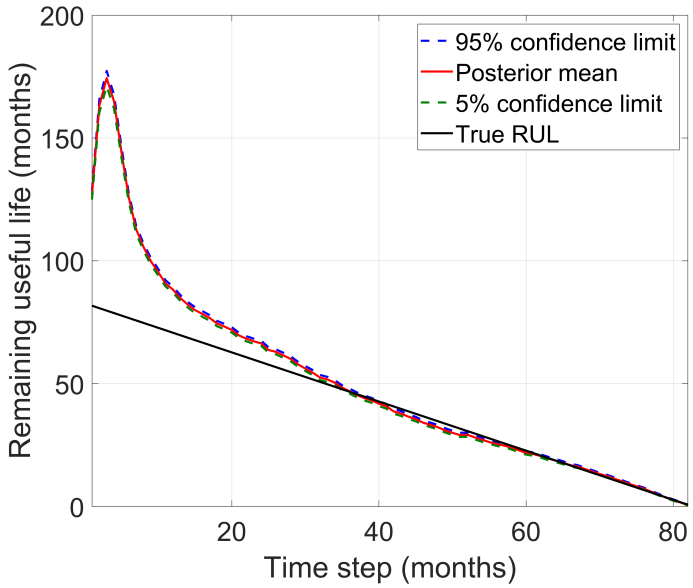



**Figure 3.21.** RUL estimation at all time steps based on crack prognostics only.

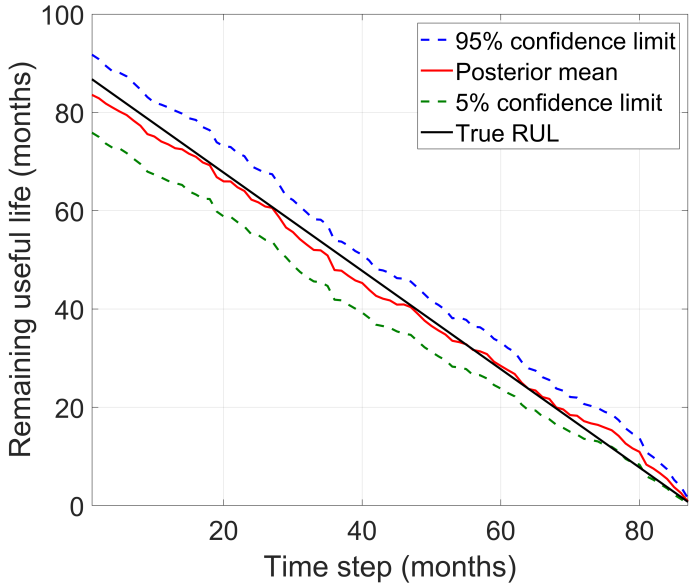



**Figure 3.22.** RUL estimation at all time steps based on gap prognostics only.

It implies that including multiple monitoring techniques can help reduce the minimum CPUT, which will result in a minimized overall maintenance cost. This demonstrates the value of adopting an additional monitoring technique. It is worth noting that the amount of cost savings by adding an additional monitoring technique should be compared

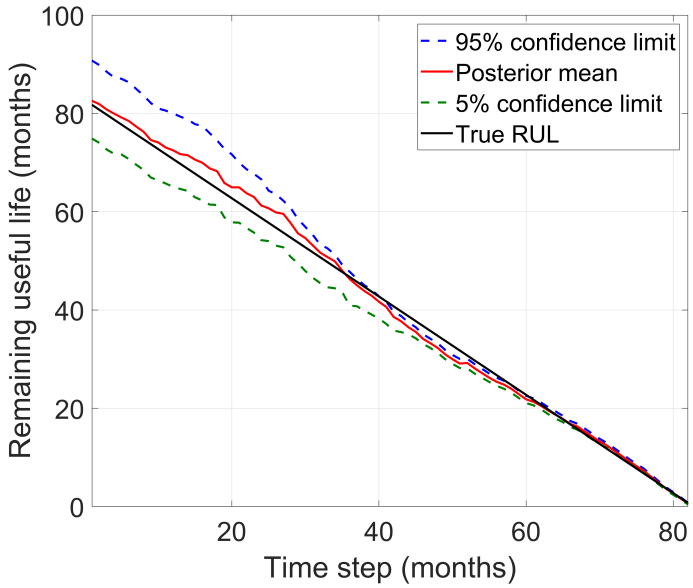



**Figure 3.23.** RUL estimation at all time steps based on jointed failure threshold.

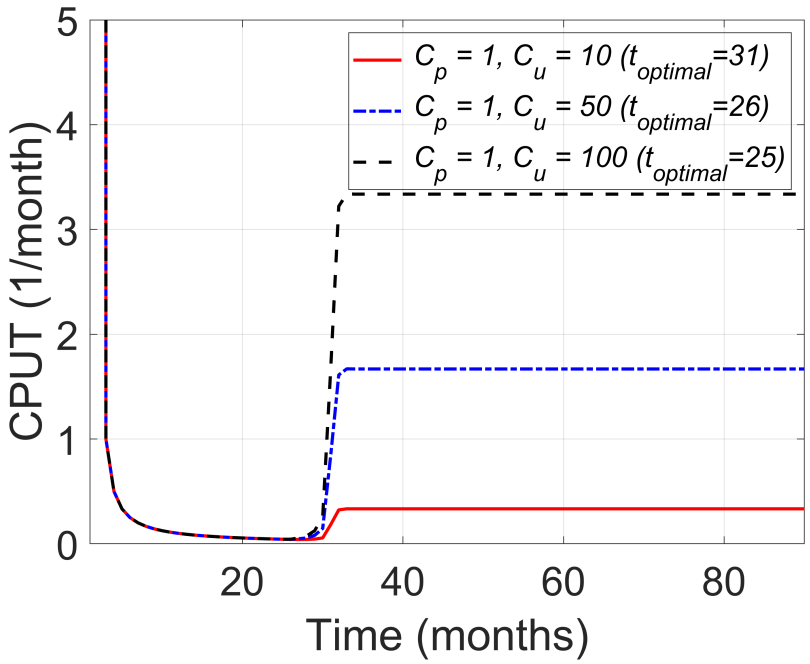



**Figure 3.24.** CPUT at 50 months corresponding to different values of $C_u$ and $C_p$.

against the cost of installing the system to justify the adoption of the technique. It is an interesting topic that worth investigating in future work, using a value-of-information analysis.

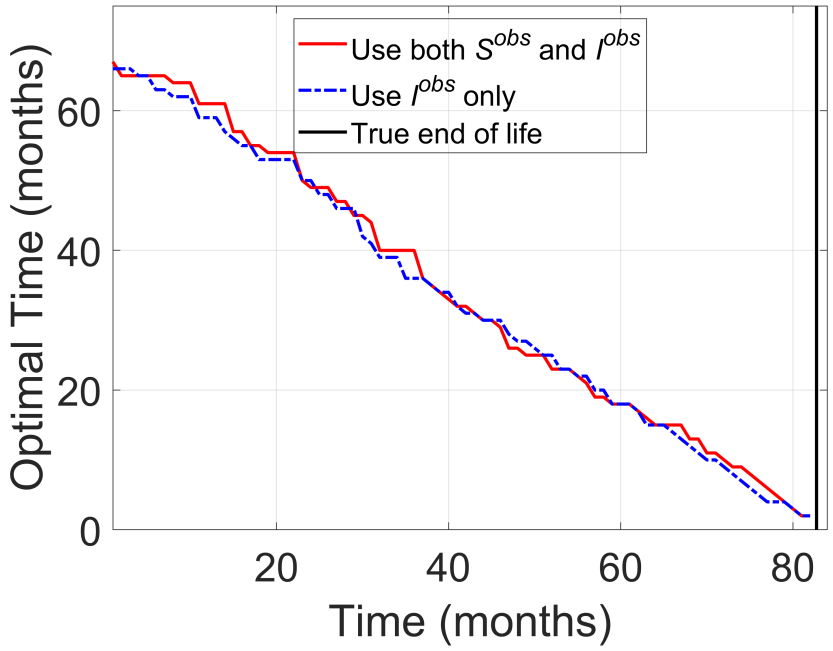



**Figure 3.25.** Optimal maintenance time corresponding to $C_u = 1$ and $C_p = 50$.

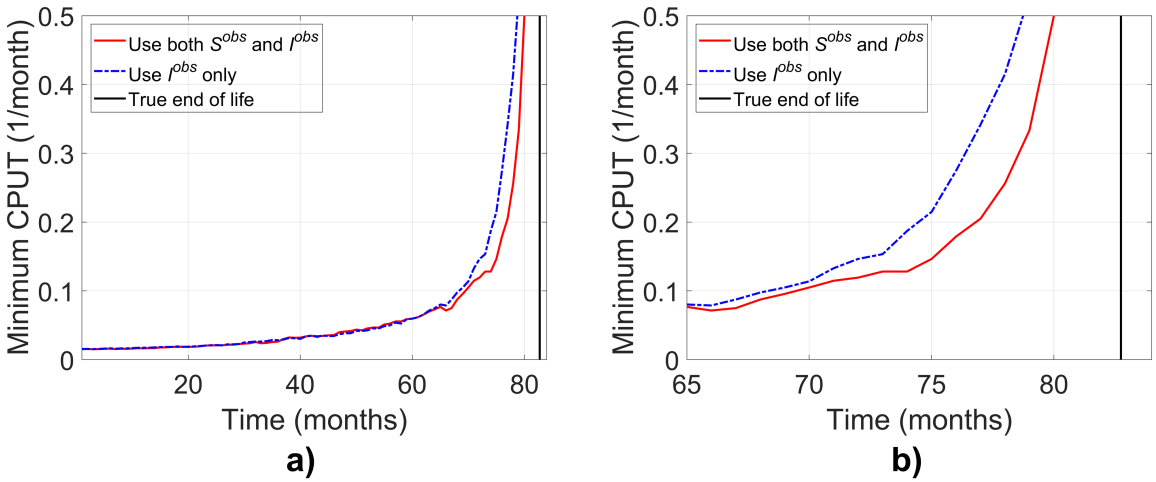



**Figure 3.26.** a) Minimum CPUT corresponding to $C_u = 1$ and $C_p = 50$, b) minimum CPUT approaching end of life.

## 3.6 Conclusions

In this chapter, an online diagnostic and prognostic framework that efficiently used multi-source data was proposed for structures with multiple failure modes. A high-fidelity

FE model was used as a physics-based emulator of two different kinds of deterioration mechanisms, the loss of contact "gap" and fatigue crack growth. The separation of damage scales has been carefully studied through global-local analysis. Two surrogate models were created and trained to generate synthetic observations (digital images and sensor data), which replaced the time-consuming FE model and enables the extensive model-based analysis of miter gates. The multi-source observations were passed through a dynamic Bayesian network for online diagnostics and prognostics. In diagnostics, the framework successfully determined the damage-related parameters as well as estimated damage conditions. In prognostics, the RUL of both failure modes was accurately predicted as time evolved. Based on the RUL results, the impact of the optimal maintenance planning of the miter gate was studied. It is found that including multiple monitoring techniques can help reduce the maintenance cost. The contributions of this chapter can be summarized as: (1) Implementation of a digital twin concept for a practical engineering problem with complicated degradation behaviors, which requires extensive model-based analysis to capture the interactions between multiple damages; (2) The extension of the widely DBN framework to fuse information from strain gauges and camera for damage diagnostics and failure prognostics of miter gates; and (3) The investigation of the impact of using multiple structural health monitoring data sources (i.e. strain sensor and camera) on the final maintenance decision making process.

To conclude, the proposed framework provides a new approach of using DBN to incorporate multiple data sources for structures under different scales of failure modes. Although the synthetic failure mechanisms and measurement data were simplified for illustration purposes, such DBN framework can be extended to more complicated structures for more informative life-cycle management and risk-based decision analysis. Future research will look at a more thorough study at the impact of digital image quality and more accurate failure representation.

## 3.7 Remarks

This chapter is composed of a first-authored publication: **Wu, Z.**, Fillmore, T. B., Vega, M. A., Hu, Z., & Todd, M. D. (2022). Diagnostics and prognostics of multi-mode failure scenarios in miter gates using multiple data sources and a dynamic Bayesian network. Structural and Multidisciplinary Optimization, 65(9), 270.

# Chapter 4

# Deep Learning-Based Automatic Crack Detection in the Underwater Environment of Miter Gates

## 4.1 Abstract

Detecting damage for large-scale infrastructure has been shifting from traditional human visual inspection to methods benefiting from advanced deep learning and computer vision techniques in recent years. However, such a transition poses unique challenges for underwater structures due to the complexity and uncertainty existing in underwater environments and a scarcity of quality training data. To overcome the limited availability of underwater imagery, this study develops a graphics-based digital twin model based on a high-fidelity finite element model that generates synthetic underwater images under diverse conditions. Coupled with data augmentation strategies, this approach enriches the dataset for the fine-tuning of current deep learning models to specifically target the underwater environment. In addition, the chapter quantitatively assesses how various environmental factors affect the performance of deep learning-based crack detection in terms of the probability of crack detection, paving the way for more effective inspection strategies for large underwater structures. The efficacy of the proposed framework is demonstrated through a case study on a miter gate. The results highlight the potential of

this approach in enhancing the reliability and detectability of image-based underwater crack detection using a limited amount of synthetic datasets.

## 4.2 Introduction

Structural inspection of large-scale infrastructure has evolved significantly over recent years, transitioning from traditional human visual inspection methods to more sophisticated image-based damage identification, facilitated by the advent of drones and advanced camera technology. Consequently, the field of structural health monitoring has increasingly come to rely on deep learning models considering their exceptional abilities in image recognition and analysis. These models can learn and identify intricate patterns in visual data, enabling the detection of subtle damage indicators that might be missed by traditional methods. For example, Cha et al. employed a 4-layer convolutional neural network trained with 277 4K images to detect cracks on concrete images with 98% accuracy (Cha et al., 2017). Furthermore, advanced models such as ResNet and Unet, known for their deep and complex architectures, have been successfully applied to detect concrete cracks and corrosion in various structures. For instance, Tan et al. utilized a ResNet model to identify multiple types of damage on bridge surfaces (Tan and Yang, 2021). Nash et al. explored the use of a High-Resolution Network (HRNetV2) with Bayesian variants for corrosion classification with uncertainty quantification, using metallic surface images taken from an industrial site (Nash, Zheng and Birbilis, 2022).

Although the above studies underscore the versatility and robustness of deep learning models in identifying structural damage under diverse conditions, inspecting underwater structures like miter gates and dams introduces a unique set of challenges. The complexities of the underwater environment, including the scattering and turbulence of water and variable lighting conditions, hinder the task of obtaining high-quality images for accurate damage detection. Besides, the complex geometries of underwater structures

and potential obstructions, further increase the difficulty of thorough scanning of all areas with adequate resolution. In addressing these specific challenges, Chen et al. proposed a CNN model with a Naïve Bayes data fusion scheme for crack detection on the underwater metallic surfaces of nuclear power plant reactors (Chen and Jahanshahi, 2017), proving its ability to detect even low-contrast tiny cracks. Nevertheless, a notable limitation of such advanced models is their reliance on extensive volumes of labeled training data. Deep learning models, particularly those with a large number of parameters, require extensive datasets to learn effectively. It is stated in Chen's paper that the model requires over 100,000 images to train the model, which makes the industrial application impractical most of the time. This requirement poses a challenge where acquiring a substantial amount of labeled data is difficult due to the uniqueness of the underwater environment conditions for each large-scale structure.

To mitigate these challenges, this chapter introduces the concept of transfer learning in machine learning, which involves adapting a pre-trained model to a new but related problem. This approach is particularly beneficial in scenarios with scarce labeled data or where training a model from scratch is computationally expensive. Leveraging the knowledge gained from a previous task, transfer learning enables rapid progress and improved performance in the new task, even with smaller datasets. In the context of underwater structural damage detection, transfer learning significantly narrows the gap between the limited availability of data and the need for robust, accurate models. In this chapter, a GBDT model is first developed to generate synthetic underwater images for training purposes. A pre-trained CNN model is then trained on the generated images for fine-tuning. The performance of the model is assessed using a testing dataset generated for this purpose.

While the deep learning-based crack detection model offers significant promise for automating damage identification from image data, its performance could be considerably affected by environmental conditions such as lighting intensity and water turbidity. These

factors can compromise the accuracy of damage detection, underscoring the necessity of analyzing how these uncertainty sources will affect the performance of the deep learning model. Performing global sensitivity analysis is crucial for understanding the robustness and reliability of deep learning-based crack detection models, especially given the impact of image quality on model performance. Global sensitivity analysis allows us to systematically quantify and rank the impact of these factors, identifying the most influential parameters that degrade image quality. By understanding these parameters, we can improve data preprocessing, model training strategies, and ultimately, the model's ability to generalize across different environmental conditions. This analysis not only enhances the model's practical applicability but also guides the development of more resilient deep learning algorithms for structural health monitoring.

The main contributions of this chapter are outlined as follows:

- First, the chapter introduces a graphic-based digital twin (GBDT) model that is capable of generating synthetic images under various scenarios. This model serves as a valuable source of specific training data for pre-trained models, effectively addressing the challenge of data scarcity in underwater environments.

- Second, the chapter presents an exhaustive analysis of different parameters affecting underwater unmanned vehicle (UUV) inspections. It provides a quantitative study of the impact of these parameters, enhancing the understanding of their effects on image quality and damage detection probabilities. A detailed sensitivity analysis is conducted for each parameter, offering insights into their respective influences on the structural damage detection process.

- Third, the practical applicability of the proposed framework is demonstrated through a case study of a miter gate. This real-world example highlights the framework's effectiveness and the significant improvements it brings to detection probability

and accuracy. Valuable insights are driven that could shape future strategies and technologies for underwater structural inspection.

The remainder of this chapter is structured as follows: Section 4.3 details the development of the graphics-based digital twin model, providing the foundation for synthetic data generation. Section 4.4 elaborates on the adopted pre-trained deep learning models and the transfer learning strategies employed. Section 4.5 introduces the methodology for uncertainty quantification and conducts a sensitivity analysis for each environmental parameter. Section 4.6 presents the results and discussions. Finally, Section 4.7 concludes the chapter, summarizing the findings and suggesting directions for future research.

## 4.3 Graphics-Based Digital Twin Development

This section outlines the development of a Graphics-Based Digital Twin (GBDT) model, employing Blender to create a synthetic underwater image generation framework. A schematic representation of the framework is presented in Fig.4.1, detailing the sequential steps and components involved.

### 4.3.1 Finite element model introduction

In the pursuit of simulating and understanding the complex behaviors of underwater structural components, this work chooses the Greenup miter gate located on the Ohio River. Miter gates, serving as crucial navigational hydraulic structures, facilitate the transition of cargo ships across varying water levels, enabling them to bypass dams and navigate rivers. Figure 4.2 (a) shows a field image of the miter gate, where a substantial portion of the miter gate remains submerged, making it susceptible to damage such as fatigue cracking and pitting corrosion. To comprehensively understand the physics behaviors of the gate, a finite element (FE) model of the Greenup downstream miter gate was developed, as shown in Figure 4.2 (b). The FE model is constructed from the design drawings of the

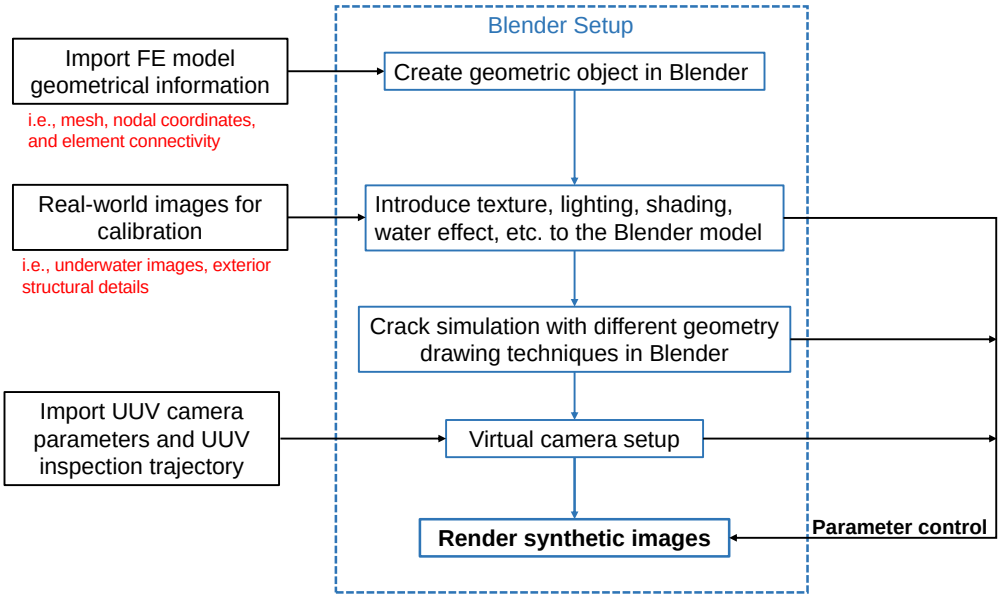



**Figure 4.1.** Synthetic underwater image generation framework with Blender.

miter gate, incorporating all relevant geometric details and material properties. It has been previously validated with the field data proving its ability to accurately predict the strain response of the structure under various loading conditions (Eick et al., 2018).

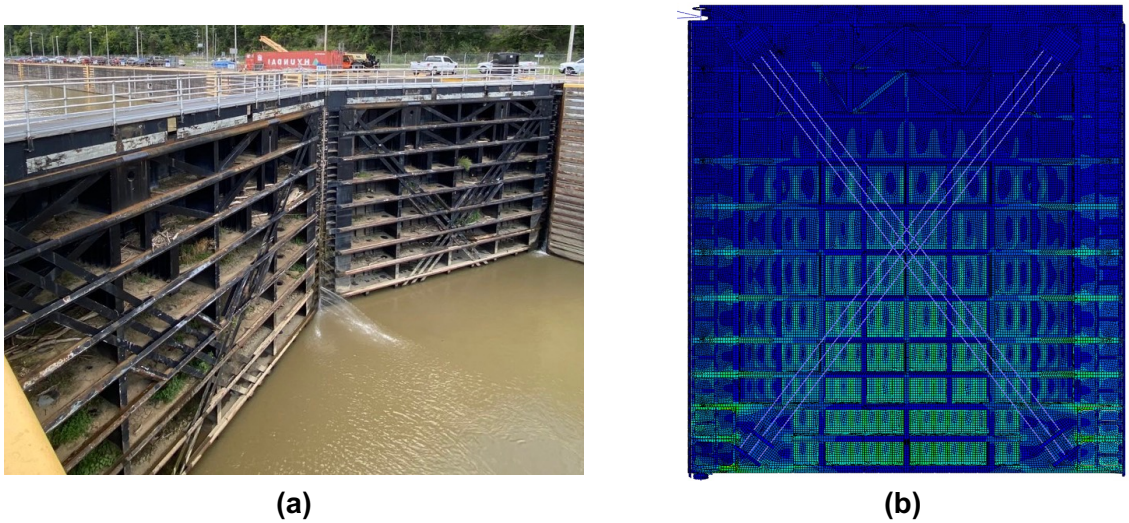



**Figure 4.2.** Miter gate structure and finite element representation.

The FE model of the Greenup miter gate serves as the cornerstone for the devel-

opment of the Graphics-Based Digital Twin (GBDT) model. After solving the physics simulation in Abaqus, the ".obj" file that contains all the geometric data — meshing, nodal coordinates, element connectivity, and material properties — is exported from Abaqus and subsequently imported into the Blender software. This process is not a mere geometric duplication but involves the dynamic state of the miter gate under time-dependent loading and damage conditions. In essence, this subsection not only introduces the FE model but also establishes its role as a critical component in the subsequent development of the GBDT model.

### 4.3.2 Blender GBDT model construction

The imported geometry from Abauqs serves as the foundational "skeleton" for constructing the Graphics-Based Digital Twin (GBDT) model in Blender. This subsection outlines the process of simulating real-world underwater conditions where multiple layers of effects are added.

Texturing as the first layer imparts a realistic appearance to structural surfaces. This process involves not just assigning color and material properties but also the integration of real field images for enhanced calibration. Following texturing, lighting, and water effects are integrated into the model. These effects are realized by basic brightness and shadow considerations, incorporating diverse light sources such as sunbeams and UUV-mounted torches. The model simulates the complex interactions between light and water, influenced by factors like turbidity and scattering, to accurately reflect the underwater environment.

To authentically replicate underwater images captured by UUV inspection equipment, the GBDT model also incorporates precise camera specifications, including focal distance, resolution, and aperture, sourced from literature and commercial product data. This ensures the synthetic images generated are as close to real inspection images as possible. The model also simulates the UUV's trajectory and movement, offering a variety of angles and distances for comprehensive visualization.

Throughout the chapter, five environmental variables in this section are considered to have a nontrivial impact on crack detection as follows,

$$\theta = [w_t,\ w_s,\ p_s,\ p_c,\ f] \tag{4.1}$$

where $w_t$ and $w_s$ stand for the coefficients of water turbidity and scattering, correspondingly. $p_s$ and $p_c$ stand for the power of sunlight and lighting source of the UUV. $f$ represents the focus distance of the camera. The process of constructing the GBDT model is exemplified in Figure 4.3.

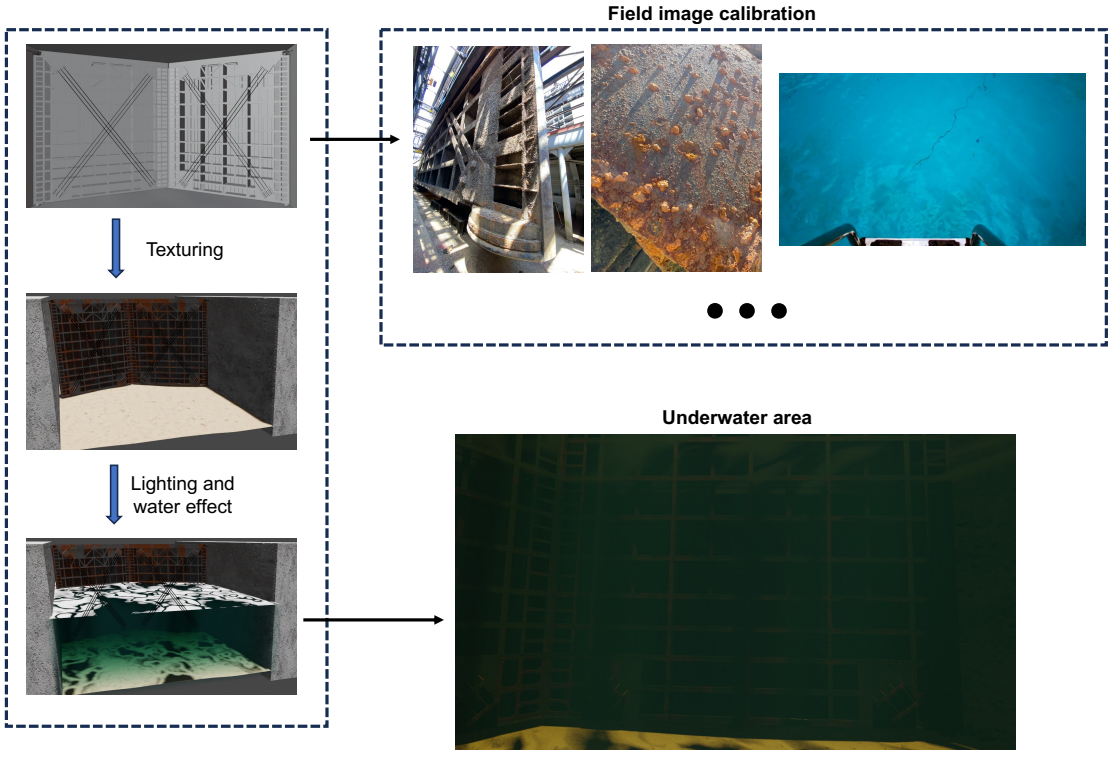



**Figure 4.3.** Blender GBDT model construction.

### 4.3.3 Rendering synthetic images

With the GBDT model established, this subsection focuses on the process of rendering synthetic images. The rendering process in Blender is a complex outcome from

mixing the factors of camera, lighting, and environmental parameters, each contributing to the final image's quality and uncertainty. The objective is to determine a practical range for the parameter vector $\theta$, and systematically manipulate $\theta$ within its range to produce a wide range of synthetic images for efficiently training the deep learning model.

The surface coverage of the miter gate is determined on the inspection route of the UUV. By aligning the virtual camera's movement in Blender based on the USACE-provided route data, a series of viewpoints are established. Images are rendered at each viewpoint, considering the camera's specifications and how it's positioned and moved within the model to simulate various inspection scenarios. The lighting and environmental settings are then adjusted, reflecting different underwater conditions — from clear to murky waters, still to turbulent conditions. Additionally, synthetic cracks of different sizes and shapes are created on the surface of the structure along the camera's path to introduce "damage features" into the images.

The rendering process is not just about producing images but also generating samples useful in training and evaluating deep learning models. The rendering process transcends mere image production, focusing on generating visuals that are instrumental in training and evaluating deep learning models. Figure 4.4 showcases examples of these rendered images with changing variable vector $\theta$ and different surface areas of the miter gate, illustrating the GBDT model's flexibility in creating a wide spectrum of synthetic images.

### 4.3.4 Impact of environmental variables on synthetic images

Environmental factors such as lighting intensity and water turbidity significantly impact the quality of images used for crack detection. As shown in Figure 4.5, altering lighting intensity can greatly affect the visibility of structural damage, with insufficient lighting masking critical details. Conversely, increased water turbidity diminishes image clarity and contrast, complicating the identification of subtle damages.

**Figure 4.4.** Blender GBDT model construction.

Moreover, the influence of these parameters on image quality is not always straightforward; it often results from the interplay between multiple factors. For instance, while high water scattering combined with strong sunlight might introduce noise and uncertainty into images, the same sunlight intensity under low scattering conditions could enhance image quality, facilitating better crack detection. This complexity underscores the necessity of a nuanced understanding of environmental impacts on image acquisition and analysis.

## 4.4 Crack Detection with Transfer Learning

This section presents a robust framework for crack detection using synthetic data augmentation and transfer learning with a fully convolutional U-Net CNN. The section is structured to guide through the process of employing a pre-trained U-Net model, preparing data, and leveraging transfer learning to enhance the model's performance for underwater

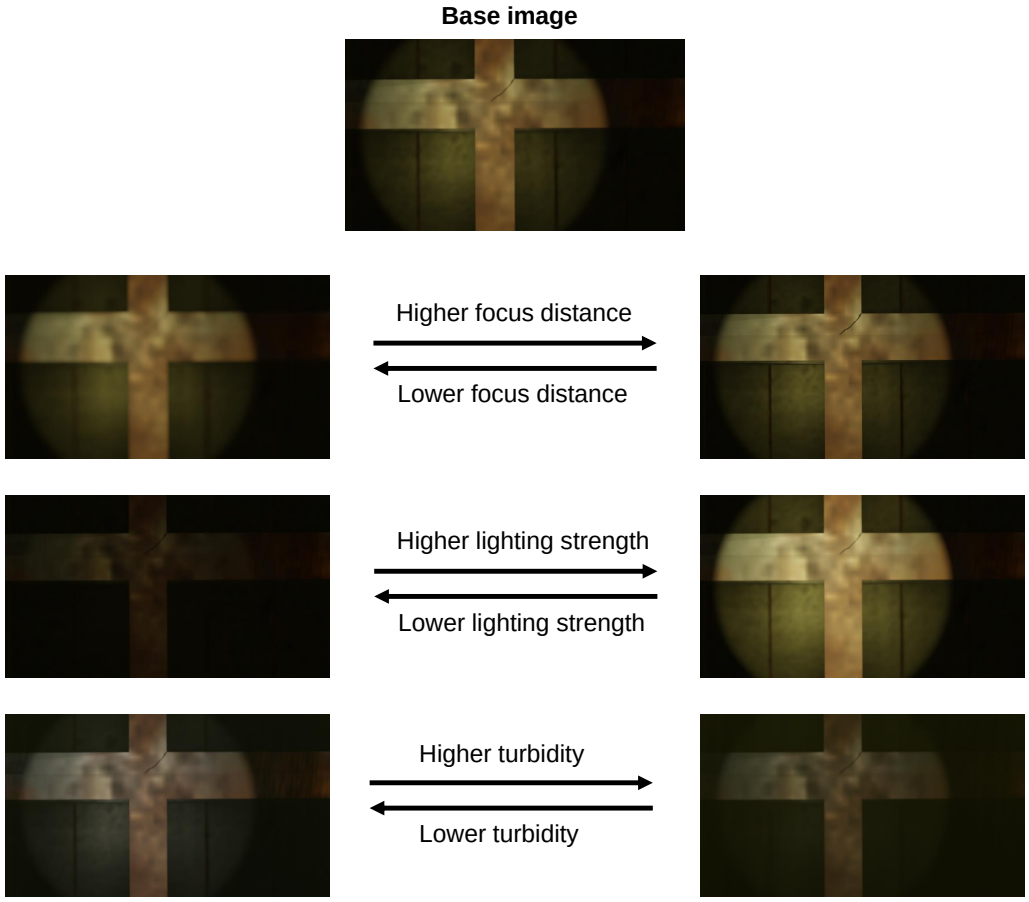



**Figure 4.5.** Environmental factor impact demonstration.

crack detection.

### 4.4.1 Introducing the Pre-trained U-Net CNN Model

U-Net is a convolutional neural network (CNN) model originally designed for biomedical image segmentation. Its architecture is characterized by a symmetric expanding path that enables precise localization, making it suitable for tasks like crack detection where capturing the objective feature is crucial. As shown in Figure 4.6, the architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. The network uses a series of convolutions, each followed by a rectified linear unit (ReLU) and a max-pooling operation to reduce the spatial dimensions.

In the expansive path, transposed convolutions are used to increase the spatial dimensions. Skip connections from the contracting path are concatenated with the feature map in the expansive path to provide local information while upsampling. This design allows the network to predict at a pixel level, making it particularly effective for segmenting images into crack and non-crack regions.

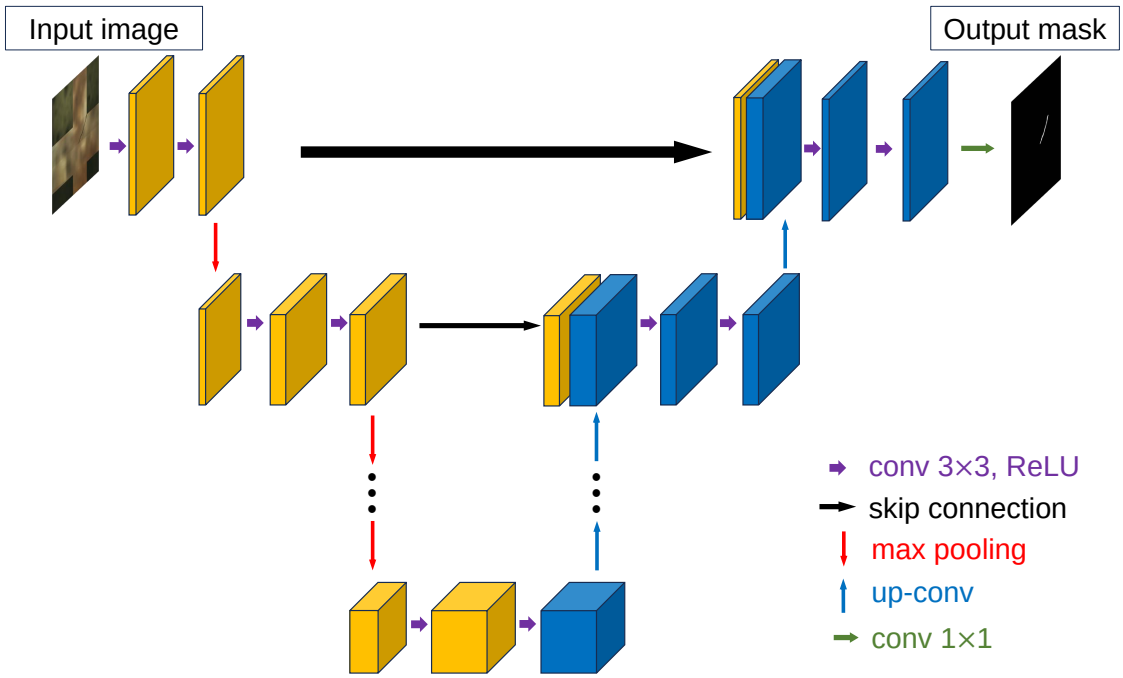



**Figure 4.6.** Blender GBDT model construction.

### 4.4.2 Data Collection and Preparation

To effectively train the U-Net model for underwater crack detection, the creation and preparation of a comprehensive dataset is crucial. Utilizing the GBDT framework developed in Section 4.3, synthetic images replicating a variety of underwater conditions are generated. As shown in Figure 4.7, these images simulate various crack sizes, shapes, and environmental settings.

It is important to note that while real-world conditions seldom present multiple cracks within a single vicinity, the simulation intentionally disperses cracks randomly

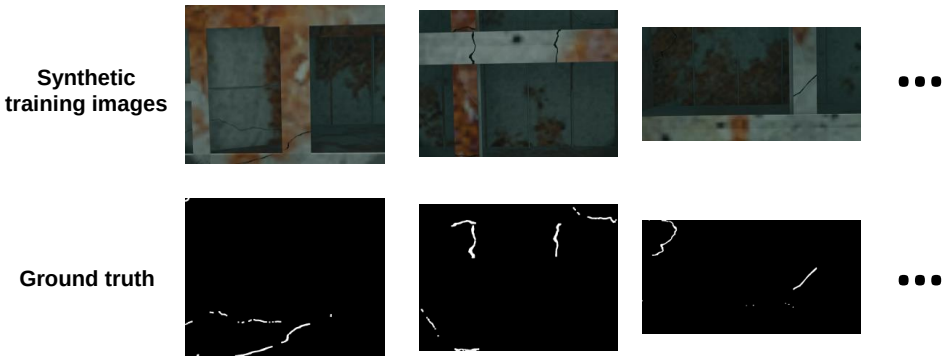



**Figure 4.7.** Images (first row), with corresponding ground truth label maps (second row), white = crack, black = background.

.

across the structure's surface. This strategy ensures a thorough representation of potential crack configurations and backgrounds. Besides, data augmentation techniques such as rotation, scaling, and brightness adjustments are employed to further increase the diversity and quantity of the training data. Corresponding ground truths for each image are simultaneously created using Blender, followed by post-processing to label cracks in white and the background in black. This prepares the data for direct integration into the deep learning model. The resulting dataset equips the U-Net model with the necessary variety and detail to accurately identify features of underwater cracks.

### 4.4.3 Transfer Learning Implementation

Transfer learning is a machine learning technique where a model developed for "task 1" is reused as the starting point for a model on "task 2". This approach is particularly preferred in scenarios when the domain gap between two tasks is small and there's limited labeled data for "task 2". In the context of this chapter, "task 1" can be considered as general crack detection for which plenty of deep learning models are already trained with abundant training data."Task 2" is mainly to detect cracks in underwater environments. While both tasks aim to identify similar features (crack patterns), "task 2" (underwater crack detection) faces challenges such as limited data availability and greater levels of

noise and uncertainty. Here, a U-Net model initially trained on crack segmentation issues is employed, demonstrating the application of transfer learning to bridge the gap between these tasks. A detailed introduction of the U-net model can be found here (Liu, Yao, Lu, Xie and Li, 2019). Figure 4.8 shows the strategy employed in this chapter to train the model to be adapted to underwater images with the least amount of training data.

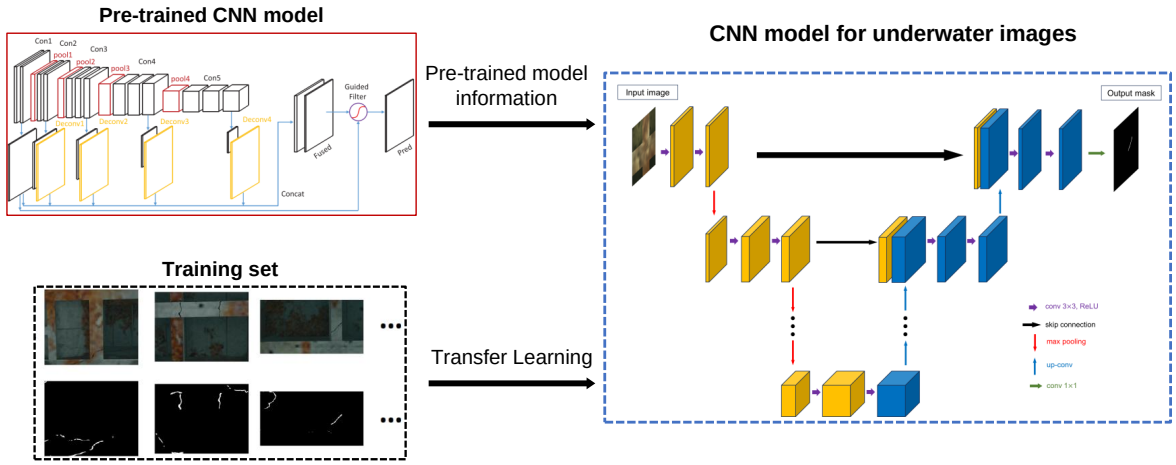



**Figure 4.8.** Transfer learning with U-net model

.

The model is fine-tuned using the synthetic underwater crack images. This approach allows for leveraging the learned features and knowledge from the pre-trained model and adapting it to the underwater domain.

## 4.5 Uncertainty quantification and Global Sensitivity Analysis of the DL model for crack detection

Once the U-net model is trained for automatic crack detection for underwater images, the results of the identification will be evaluated. In this section, the chapter delves into methodologies for evaluating and understanding the influence of various parameters and their uncertainties on the performance of the deep learning model for crack detection in

underwater structures.

### 4.5.1 Uncertainty quantification of the DL model performance

The Monte Carlo Simulation (MCS) process is one of the most common and efficient ways to assess how environmental variables influence crack detection accuracy and for uncertainty quantification of the DL model performance. The MCS is conducted by first identifying the environmental variables. As introduced in Section 4.3.2, the parameter of interest in this chapter is defined as $\theta = [w_t,\ w_s,\ p_s,\ p_c,\ f]$, where $w_t$ and $w_s$ stand for the coefficients of water turbidity and scattering, correspondingly. $p_s$ and $p_c$ stand for the power of sunlight and lighting source of the UUV. $f$ represents the focus distance of the camera. While fixing the camera, the images are rendered by systematically varying the five environmental variables given the range of interest. 2,000 synthetic images are generated using the Blender-based GBDT model, and each image is tagged with its combination of environmental variables to simulate a wide range of underwater conditions. These images serve as input to the deep learning crack detection model, and the accuracy metric will be calculated based on ground truth for performance evaluation. The data matrix for input variables can be constructed as follows

$$\mathbf{X} = \begin{pmatrix} w_t^1 & w_s^1 & p_s^1 & p_c^1 & f^1 \\ w_t^2 & w_s^2 & p_s^2 & p_c^2 & f^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ w_t^n & w_s^n & p_s^n & p_c^n & f^n \end{pmatrix} \tag{4.2}$$

In the context of underwater crack detection, traditional accuracy metrics may not fully capture the model's performance and the detectability of each image due to class imbalances in the images (e.g., many non-crack pixels versus fewer crack pixels). Therefore, recall is used instead of accuracy as a more suitable metric for comparing model predictions with ground truth. Recall emphasizes the model's ability to correctly identify

positive instances (crack pixels) among all actual positive instances, which is crucial for ensuring no damage goes undetected. The recall for the image generated from each input variable vector can be denoted as

$$\mathbf{Y} = [y^1, y^2, \cdots, y^n]^T \tag{4.3}$$

This comprehensive dataset enables a thorough investigation into the sensitivity of the deep learning model to changes in environmental factors, laying the groundwork for subsequent global sensitivity analysis. A detailed flowchart of the MCS is shown in Figure 4.9, from parameter specification to synthetic image generation and analysis preparation, ensuring a methodical approach to understanding the model's performance under diverse conditions.

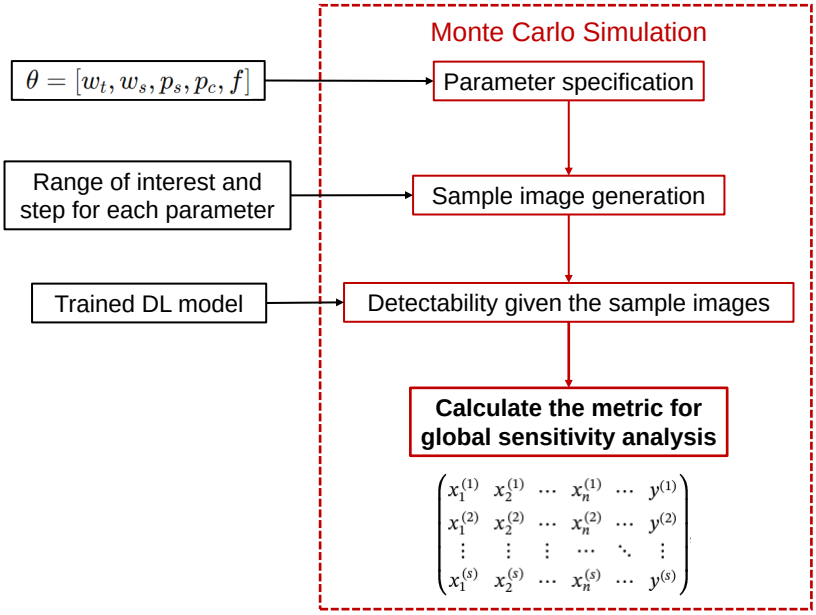



**Figure 4.9.** Monte Carlo simulation (MCS) flowchart

.

By analyzing the model's performance across this diverse dataset and calculating the first-order Sobol' index, we can quantitatively assess the impact of each uncertainty source on the model's ability to detect cracks. This comprehensive sensitivity analysis

provides insights into the robustness of the model and guides further refinements to enhance its accuracy and reliability in real-world underwater inspection scenarios.

### 4.5.2 Global Sensitivity Analysis from Monte Carlo Samples

Global sensitivity analysis plays an integral role in unraveling the relationship between input variables and the output in computational models. Among various methods, the Sobol' indices are widely recognized for their capability to decompose the output variance into contributions from individual input variables and their interactions.

Defining $Y \in \mathbb{R}$ as the quantity of interest (QoI), its underlying physics with input variables $X$ is given by $Y = g(\mathbf{X})$. The variance $Var(Y)$ of $Y$ then can be decomposed as follows (Sudret, 2008; Sobol, 2001)

$$\mathrm{Var}(Y) = \sum_{i=1}^{n} V_i + \sum_{1 \le i < j}^{n} V_{ij} + \cdots + V_{12\ldots n}, \tag{4.4}$$

where $V_i$ represents the variance of $Y$ cased by $X_i$, excluding the its interaction with other input variables.

Because the number of indices grows dramatically with higher-order Sobol' indices, the first-order Sobol' index for an input $X_i$ is commonly used which is defined as follows

$$S_i = \frac{\mathrm{Var}_{X_i}[E_{\mathbf{X}_{\sim i}}(Y|X_i)]}{\mathrm{Var}(Y)}, \tag{4.5}$$

where $S_i$ is the first-order Sobol' index of $X_i$ and $\mathrm{Var}_{X_i}$ denotes the variance over $X_i$, $E_{\mathbf{X}_{\sim i}}$ represents the expected value over all input variables except $X_i$, and $\mathrm{Var}(Y)$ is the total variance of the model output. A more detailed variance decomposition literature review can be found in (Sudret, 2008; Sobol, 2001; Hu and Mahadevan, 2019).

A novel approach of sensitivity analysis using Monte Carlo samples for Global Sensitivity Analysis (GSA) is used in this study for estimating the first-order Sobol' index (Li and Mahadevan, 2016). This method is particularly effective for understanding the

influence of individual input variables on the variability of the output in models, a crucial aspect when dealing with the complexity of underwater crack detection.

The algorithm works as follows:

1. Input transformation: Input samples are converted into cumulative distribution function (CDF) domains to deal with their stochastic nature. For each input dimension $i$, the algorithm sorts the samples and assigns them CDF values, ensuring that the transformed inputs are uniformly distributed.

2. Local variance computation: The function divides the range of each input variable into pre-defined sub-domains. For each sub-domain in input dimension $i$, it computes the variance of the output, considering only the samples where the input lies within the current sub-domain. This step results in a local variance matrix, $\mathbf{Var}(\mathbf{X})$, with dimensions corresponding to the number of sub-domains and input variables.

3. Sobol' index estimation: The first-order Sobol' index for each input variable is estimated by comparing the mean of the local variances for each input with the total output variance. The indices represent the normalized effect of each input variable on the output variance, with higher values indicating a stronger influence.

The above procedure efficiently estimates the first-order Sobol' indices, enabling analysts to identify and rank the most significant variables affecting the output. The results of these evaluations will be detailed in the following section.

## 4.6 Results and Discussion

Figure 4.10 presents three scatter plots illustrating how recall values correlate with camera focus distance, turbidity, and illumination strength. Among them, the camera focus distance has a significant impact. Specifically, shorter focus distances, where the camera is not properly aimed at the structure's surface, are associated with lower recall values. This suggests that achieving high recall values is almost impossible with a low

focus distance. As the focus distance increases, which means the camera is better aligned with the structure's surface, recall values tend to improve. Although turbidity's effect on predictions is not as pronounced as that of focus distance, it still offers valuable insights. At low turbidity levels, recalls cluster more densely on high values, indicating higher detection probabilities. However, as turbidity rises, the frequency of high recalls decreases. Unlike the focus distance case, high recalls are still attainable with a lower likelihood. In contrast, illumination strength has the opposite effect of turbidity. Enhanced illumination, leading to clearer images, fosters an increase in detection probability.

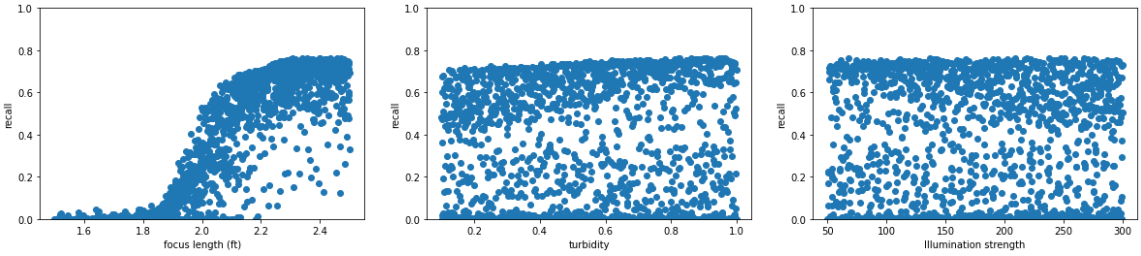



**Figure 4.10.** Selected Monte Carlo simulation (MCS) results

.

The sensitivity analysis, informed by the recall data, reinforces these findings. Table 4.1 shows that the focus distance has a dominant Sobol' index of 0.89, which is significantly higher than the indices for the other two variables. The Sobol' index is a quantitative tool used to measure how much each environmental variable contributes to the overall outcome. This method further confirms the insights derived from the recall figure.

**Table 4.1.** Selected first-order Sobol' index.

| **Input variable** | Focus distance ($f$) | turbidity ($w_t$) | light power ($p_c$) |
|---|---|---|---|
| **First-order Sobol' index** | 0.89 | 0.005 | 0.0004 |

## 4.7 Conclusions

Detecting fatigue cracks in underwater structures requires a robust inspection framework under environmental variables like turbidity, lighting, and focus discrepancies. This research employed a novel approach, integrating a graphics-based digital twin model with deep learning techniques, to enhance inspection accuracy. Blender 3.4 was used to generate synthetic underwater images, simulating diverse conditions. Subsequently, the U-net model, adapted for underwater environments through transfer learning and data augmentation, was trained using these images. Comprehensive results revealed the camera's focus distance as the most influential factor in crack detection. While turbidity and illumination strength also played roles, their impacts were less pronounced. This study's findings offer a foundation for understanding underwater environmental conditions for damage detection, providing valuable insights for optimizing future underwater inspections.

## 4.8 Remarks

This chapter is currently in preparation for publication, the dissertation author was the primary investigator and author of this paper: **Wu, Z.**, Hu, Z., & Todd, M. D. Deep Learning-Based Automatic Crack Detection in the Underwater Environment of Miter Gates. Structural Health Monitoring.

# Chapter 5

# Optimization of Unmanned Aerial Vehicle Inspection Strategy for Infrastructure Based on Model-Enabled Diagnostics and Prognostics

## 5.1 Abstract

The use of unmanned aerial vehicles (UAVs) for structural health inspection has become a promising technique to perform labor-intensive, accessibility-challenged, and sometimes dangerous inspection tasks. This chapter presents a novel physics-informed UAV inspection planning framework for infrastructure structural health assessment based on model-based diagnostics and prognostics enabled by physics-based probabilistic analysis. It bridges the gap between UAV mission planning and inspection with model-based probabilistic analysis, by allowing bidirectional information exchange, namely (1) structural damage state diagnostics using UAV inspection data and (2) UAV inspection optimization through model-based failure prognostics. Based on the bidirectional communication, the impacts of the three key UAV inspection parameters (i.e., inspection distance, inspection interval, and critical maintenance threshold) on structural life-cycle cost are analyzed. The optimization of key UAV inspection parameters is achieved by minimizing the cost per unit time (CPUT) through model-based pre-posterior analysis. In this analysis, synthetic

observations are generated using predictive models according to the prior distributions of various uncertainty sources (e.g., detection rate, damage state evolution, etc.). The generated synthetic observations are then used to obtain the posterior distributions of uncertain parameters, enabling the integration of prior information and Bayesian model updating into inspection optimization through a cost function. The proposed model offers a robust method to accommodate the inherent uncertainties in failure prognostics, leading to a more effective optimization of the UAV inspection parameters. The practical application of the framework is demonstrated through a miter gate example. The results show that the proposed method is able to efficiently determine the optimal UAV inspection parameters and continuously update the information model.

# Nomenclature

$C_F(\cdot)$ Structural failure cost model

$C_I(\cdot)$ Inspection cost model

$C_R(\cdot)$ Maintenance cost model

$G_c(\cdot)$ Overall objective function

$G_K(\cdot)$ Surrogate model for stress intensity factor

$\alpha$ Uncertain coefficient in the observation model

$\mathbf{h}$ Loading condition vector

$\boldsymbol{\epsilon}$ Environmental noises in image processing

$\boldsymbol{\theta}$ Unknown damage degradation parameters

$\boldsymbol{\omega}$ Vector of UAV inspection parameters

$\Delta K_i$ Stress intensity factor at time step $t_i$

$\hat{e}$ Random variable of the estimated damage state

$\hat{g}(\cdot)$ Surrogate model for objective function

$\mathcal{X}(\cdot)$ UAV image model

$\mathcal{Z}$ Detectability indicator

$\Psi(\cdot)$ Computer vision algorithm

$d$ UAV inspection distance

$e$ True damage state

$e_d^{(k)}$ $k^{th}$ damage sample generated through uncertainty propagation

$e_h$ Local failure limit state

$e_m$ Critical maintenance threshold

$e_o$ Observed damage state

$e_{o,j+1}$ Observed damage state at $j^{th}$ inspection

$G_{UP}$ The process of uncertainty propagation by executing the state-transition equation

$i$ Continuous time index

$j$ UAV inspection time index

$p_{\mathcal{Z}}(\cdot)$ Probability of detection model

$T_I$ Next inspection interval

$w^{(k)}$ Weight of $k^{th}$ particle

## 5.2 Introduction

Periodic inspection and regular maintenance play a vital role in guaranteeing the integrity of civil infrastructure (Spencer Jr, Hoskere and Narazaki (2019); Gibb, La, Le, Nguyen, Schmid and Pham (2018)). For larger-scale structures such as bridges, dams, and miter gates, these tasks can be especially time-consuming, accessibility-challenged, and labor-intensive due to their complicated geometries, extensive inspection areas, and

the presence of multiple degradation mechanisms at multiple locations. In recent years, the increased use and accessibility of unmanned aerial vehicle (UAV) platforms, coupled with the development of user-friendly image processing software, has led to a growing adoption of UAV-based inspection techniques for structural health monitoring (SHM). Such techniques enable the efficient and effective assessment of structural integrity, as well as the detection of potential damage, through the use of images and subsequent data analysis (Freimuth and König (2018)) provided by UAV imaging payloads. It provides a promising way to alleviate labor-intensive and/or dangerous inspection tasks.

UAV payloads span various sensing mechanisms, e.g., high-speed cameras or Light Detection and Ranging (LiDAR), enabling rapid and cost-effective scanning of the structure resulting in diverse, information-rich images from multiple angles and perspectives. Such images can be analyzed to capture damage-correlated features that can inform proactive measures for maintenance and repair (Zhao, Kang and Li (2022)). Depending on the specific sensing payload, UAVs can detect damage on the surface (camera/videography, LiDAR, etc.) or those hidden beneath the surface (ultrasonics or magnetometry, etc.). For example, Zhao, Kang, Li and Ma (2021) developed a high-precision 3D dam monitoring model using overlapped images from UAVs with cameras to perform surface damage detection for emergency situations. Zhang, Zuo, Xu, Wu, Zhu, Zhang, Wang and Tian (2022) introduced a multi-level attention mechanism-based UAV inspection network for pavement damage detection with various crack types. In addition, Zhang, Watson, Dobie, MacLeod and Pierce (2018) implemented thickness measurement on an aluminum plate with an ultrasonic probe payload on a UAV carried, allowing it to detect invisible discontinuities and defects beneath the outer facade. Mu, Zhang, Xie and Zheng (2020) presented a workflow of large-scale UAV-borne magnetic survey where magnetic sensors were mounted on UAV to inspect the subsurface of metallic targets by removing the UAV interference field.

In many applications, UAV inspection systems are designed to enable autonomous

damage detection and consistent data acquisition. However, the efficiency of detection is often compromised (or sub-optimized, at best) when UAVs are given a fixed flying route designed to cover an entire structure. In the case of large-scale structures such as miter gates, dams, and bridges, the inspection efficiency of each UAV mission becomes a crucial factor that must be carefully considered. Motivated by improving the efficacy of UAV inspection of civil infrastructure, extensive research has been conducted in recent years to optimize UAV inspection strategies (Chung, Maharjan, Zhang, Eliassen and Strunz (2020); Yan, Zhang, Zhang, Wang, Liao and Liang (2018); Tan, Li, Liu, Chen and Zhou (2021); Li, Han, Ge, Xu and Liu (2020)). For instance, Bolourian and Hammad (2020) proposed a bridge inspection framework for surface defects based on LiDAR-equipped UAVs, in which the shortest obstacle-free UAV flight path is determined by considering different regions of risk levels. Ellenberg, Kontsos, Bartoli and Pradhan (2014) considered the appropriate distance to be maintained between the camera and the target surface as a key factor in achieving minimum acceptable detection sensitivity during a UAV inspection mission. Zeng, Wu, Todd and Hu (2023) developed a UAV path planning algorithm by formulating it as a multi-objective optimization problem informed by physics-based risk analysis. Even though these UAV inspection optimization strategies are able to optimize the inspection from certain perspectives, they are largely dedicated to using physics-based analysis to guide the minimization of flight path length or the maximization of the inspection coverage. The incorporation of model-based probabilistic analysis and structural life-cycle optimization into UAV mission planning has been rarely studied; this chapter seeks to address this by fully incorporating the UAV inspection process into an optimal life-cycle monitoring strategy.

The most common goal in life cycle monitoring of large-scale civil infrastructure is to minimize the life-cycle cost; as such, risk-based inspection and maintenance planning have been extensively studied in the risk and reliability research community (Liu, Gao, Jiang and Zeng (2023b); Shafiee and Sørensen (2019); Yeter, Garbatov and Soares

(2020)). For example, Okasha and Frangopol (2010) proposed an automated maintenance optimization approach that considers multiple preventive maintenance types and time intervals while incorporating criteria such as redundancy and life-cycle cost. Onoufriou and Frangopol (2002) discussed the application of reliability-based assessment in the inspection optimization of complex structures. Dong and Frangopol (2015) presented a risk-informed methodology that combines multi-objective optimization and life-cycle probabilistic risk assessment to optimize inspection and repair planning for ship structures, resulting in decisions that balance risk and cost factors. Luque and Straub (2019) suggested a dynamic Bayesian network-based approach to enable risk-based optimal inspection. By continuously tracking damage and triggering maintenance based on the health state of the aircraft, Yiwei, Christian, Binaud, Christian, Haftka and Kim (2017) designed a cost-driven maintenance policy that balances aircraft safety and maintenance costs by leveraging the benefit from both scheduled and unscheduled maintenance. Recently, Vega et al. (2020) incorporated failure prognosis and a cost function that considered both preventive and reactive (emergency) costs associated with the reliability of the structural components to determine the optimal time for maintenance planning based on both monitoring data and historical condition assessments. Hughes, Barthorpe, Dervilis, Farrar and Worden (2021) provided a framework for maximizing expected utility in engineering applications by utilizing probabilistic risk assessment, specifically fault tree modeling, which enables risk-based decision-making that facilitates the comparison of actions and the development of strategies. While these risk-informed inspection optimization frameworks have been applied to various civil infrastructure, most of them did not consider model-based diagnostics and prognostics in the optimization. The integration of these frameworks with the aforementioned emerging UAV-based inspection planning techniques has not been yet substantially studied in the literature.

This chapter aims to bridge the gap between UAV inspection planning and risk-informed inspection optimization by making the following major contributions.

- First, it presents a novel physics-informed framework for planning UAV inspections of deteriorating infrastructure. This framework leverages model-based diagnostics and prognostics enabled by physics-based probabilistic analysis, connecting UAV mission planning to structural inspection goals.

- Second, the chapter accounts for the impact of UAV inspection results on subsequent model-based damage diagnostics and failure prognostics. This approach optimizes mission planning for UAV inspections by considering the physics of the inspection objectives and the risk-based damage profile of the structure being inspected. The resulting model-informed inspection optimization framework provides a more informative and effective approach to UAV inspection planning over the structural life cycle.

- Third, Bayesian optimization is incorporated to accelerate the process of UAV inspection optimization, greatly enhancing computational efficiency and practical viability.

- Fourth, the practical applicability of the proposed framework is demonstrated using a miter gate example, highlighting its real-world effectiveness.

Fig. 5.1 shows an overview of the proposed framework. As illustrated in this figure, the proposed framework consists of a bi-directional information flow. In Direction 1 (i.e., Sec. 3), to simulate the process of damage evolution, a high-fidelity finite element (FE) model is developed. It emulates the structure of interest (in this case, a miter gate hydraulic structure) that experiences multiple fatigue cracking locations for demonstration purposes. Based on the FE model, a Bayesian model updating framework is proposed to perform model-based diagnostics and prognostics using UAV inspections. Building upon the framework in Sec. 3, in Direction 2 (i.e., Sec. 4), an objective function is constructed as the cost per unit time (CPUT) where several factors that relate to UAV inspection are

considered: the inspection distance, inspection interval, and critical maintenance threshold. Based on the cost analysis, Bayesian optimization is then performed to find the optimal inspection parameters for the UAV based on model-based diagnostics and prognostics. Even though the miter gate is employed for demonstration in this chapter, the underlying concepts can be extended and applied to other locations and different types of structures.

The remainder of this chapter is arranged as follows. Section 5.3 presents the background of UAV-based damage inspection for large-scale structures. The approach of coupling UAV inspections with model-based failure prognostics is then described in Sec. 5.4. Section 5.5 introduces the objective function and the optimization of the UAV inspection program, followed by Sec. 5.6 which demonstrates the proposed framework through a miter gate. The results are shown and discussed in Sec. 5.7, and Sec. 5.8 draws the conclusions.

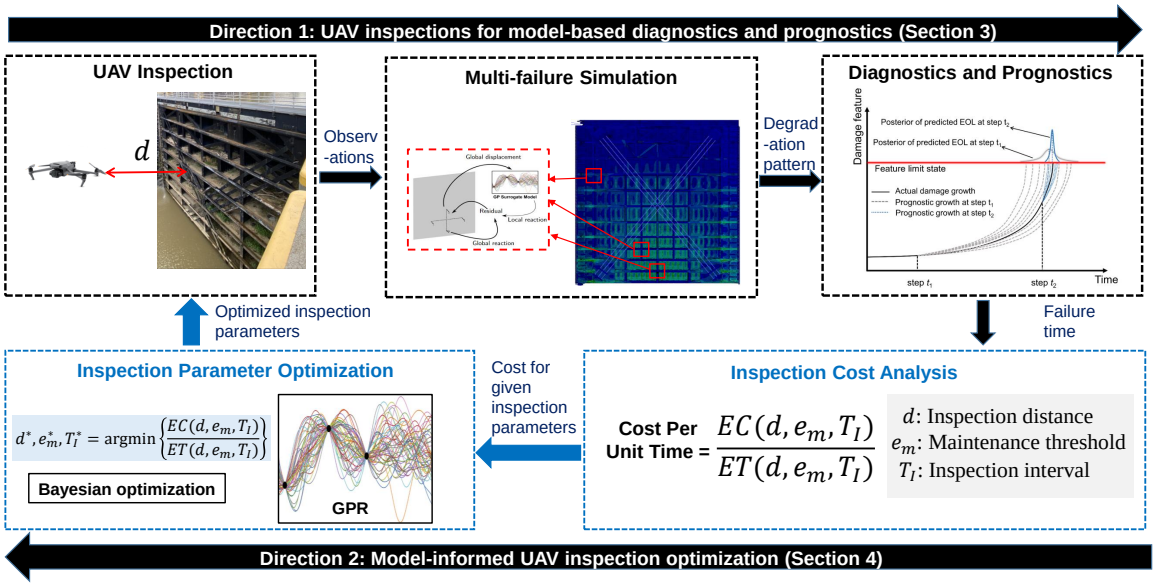



**Figure 5.1.** Overview of the proposed framework.

## 5.3 UAV-Based Damage Inspection of Large Civil Infrastructure

There is little doubt that using UAV systems to enable labor-extensive, inaccessible, and/or tedious structural inspection duties has substantially increased, as UAVs have undergone radical transformations in efficiency, size, and controllability (Zeng et al. (2023); Phung, Quach, Dinh and Ha (2017); Bolourian and Hammad (2020)). Even though the current UAV mission planning methods for damage inspection differ from each other, the procedure generally consists of three main steps, namely three-dimensional structural modeling, physics-based structural analysis, and UAV mission planning based on structural analysis. Fig. 5.2 illustrates the three steps for UAV-based damage inspection planning for a miter gate structure, which is also the focus of this chapter. Among the three steps, UAV mission planning is the primary research focus in the literature. The goal is to optimize the UAV inspection parameters to minimize or maximize a certain quantity of interest such as flight time, path length, or view area coverage. For instance, Bolourian and Hammad (2020) optimized the flight path of a UAV to increase the coverage of UAV inspection using LiDAR. Phung et al. (2017) formulated the UAV inspection problem as an extended traveling salesman problem that accounts for both the coverage and obstacle avoidance of UAVs. In our previous work (Zeng et al. (2023)), we proposed a Bayes risk-based UAV path planning algorithm for damage detection by accounting for not only the probability of detection (PoD), but also the impact of the flight path on decision-making costs related to SHM.

As an intelligent and potentially autonomous inspection technique, the performance of UAV damage assessment is affected by various inspection parameters, including the length of the flight path, inspection distance, view angle(s), and inspection frequency:

- *Length of flight path*: The length of the flight path is mainly correlated to the cost of inspection and coverage. A long flight path could lead to an increased inspection

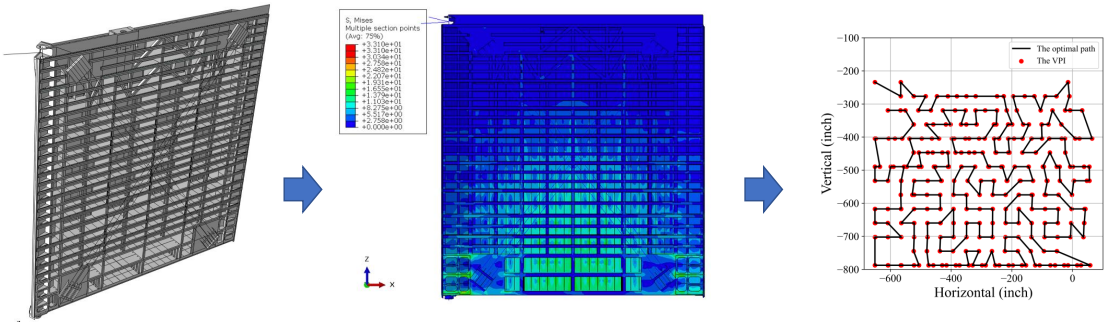
cost, while it can potentially help increase the coverage and effectiveness of UAV inspection if the path is properly designed. Most current UAV inspection research, therefore, focuses on minimizing the flight path and at the same time accounting for other quantities of interest to achieve a trade-off between the inspection cost and inspection accuracy.

- *Inspection distance*: As a non-contact inspection method, UAV inspection collects information from a structure of interest. The quality of the data usually depends upon the UAV inspection distance. Due to safety and the length of flight path concerns, a UAV cannot fly too close to the structure, even though flying closer to the structure's surface typically allows for collecting higher-resolution data. An appropriate flying distance could help reduce the inspection cost and increase inspection accuracy.

- *View angle of UAV inspection*: Structural damage manifestation usually has complicated and irregular presentation. Inspecting the damage from different view angles could increase the probability of detection. The optimization of the viewpoints of interest (VPIs) as illustrated in Fig. 5.2 is also an important research topic that has been extensively studied in the literature.

- *Inspection frequency*: Structural damage inspection needs to be performed periodically to detect the target damage at a stage that is early enough to inform corrective actions and avoid catastrophic failures. However, inspections that are too frequent may significantly increase the inspection costs, while inspections that are too infrequent may increase the risk to an unacceptable level. Determining an optimal inspection frequency is vital to the research topic that needs to be considered in UAV inspection planning.

The authors acknowledge that the above list is not exhaustive, and most of the current methods use one, two, or part of these parameters to optimize the mission of a




**Step 1**: Structural model of miter gate

**Step 2**: Physics-based structural analysis

**Step 3**: Identification of VPIs and UAV path planning

**Figure 5.2.** Model-informed UAV Damage Inspection of a Miter Gate.

UAV inspection. While this chapter concentrates on optimizing UAV inspection distance and inspection frequency, the proposed framework presented in this chapter could be extended to include other UAV inspection parameters. In addition, existing methods in the literature are mainly dedicated to using physics-based analysis to guide the minimization of flight path length or the maximization of the inspection coverage, but they fail to consider the impact of UAV inspection results on subsequent model-based damage diagnostics and failure prognostics; we would argue this is the ultimate goal of inspections over the life cycle. The following section aims to fill this void by developing a novel UAV inspection optimization framework that integrates inspection optimization with model-based diagnostics and prognostics to minimize the overall life-cycle cost. To the best of our knowledge, this is the first attempt to incorporate post-inspection decision-making guided by physics-based analysis into the UAV inspection planning of large civil infrastructure.

## 5.4 Coupling UAV Inspections with Model-Based Failure Prognostics

### 5.4.1 Overview

UAV inspection enables the effective acquisition of high-resolution images of structural damage. Such high-resolution images obtained from UAVs are typically utilized

to train machine-learning or computer vision algorithms for automated damage detection/assessment. Although the UAV inspection system provides a reliable way to acquire image observations, inevitable noise and other sources of uncertainty challenge accurate SHM assessment that informs prognostics. The focus of this section is to build a relationship between UAV inspection and model-based diagnostics/prognostics under uncertainty. In particular, detection theory is first implemented to identify damage features while taking into account the sources of uncertainty present in UAV observations. Next, we discuss how the prior information about the damage and damage-related parameters are updated using UAV observations, which further inform physics-based simulations. Based on that, we investigate the impact of the uncertainty sources in UAV inspections on model-enabled failure prognostics, which eventually allows for the prediction of the remaining useful life (RUL). This paves the way to providing essential information for life cycle cost optimization. We provide detailed explanations of each step in the following sections.

### 5.4.2 UAV inspection distance impact on damage detection

Significant research has been devoted to automating the feature extraction and damage identification process from UAV images. With the ability to generate high-quality labeled training image datasets through both field data and high-fidelity simulation models (e.g., the graphics-based digital twin model (Wang, Rodgers, Zhai, Matiki, Welsh, Najafi, Wang, Narazaki, Hoskere and Spencer Jr (2022))), machine-learning algorithms offer a promising solution to capture and learn different damage patterns hidden in the data and make corresponding damage assessments (Xu, Tian and Li (2022); Ji, Xue, Wang, Luo and Wang (2021); Chen and Jahanshahi (2017)). However, the effectiveness of such algorithms often highly depends on the image quality obtained from the UAV. In the presence of environmental noise and the limitation of computer vision techniques, the true damage state $e$ may not be accurately estimated. Denote the inspection distance, $d$, as

the distance between the UAV camera and the surface of the scanning region, a larger inspection distance normally increases the uncertainty and noise in the observations. As illustrated in Fig. 5.3, the images become increasingly blurred as the inspection distance increases.

For any given flying distance $d$ and a true damage state $e$, we denote the collected image from UAV as $\mathcal{X}(d, e)$. Denoting environmental noise as $\boldsymbol{\epsilon}$, we represent the relationship between the true damage state $e$ and the observed damage level from the UAV as

$$\hat{e} = \Psi(\mathcal{X}(d, e), \boldsymbol{\epsilon}), \tag{5.1}$$

where $\hat{e}$ is the estimated damage state based on the image, $\Psi(\cdot)$ stands for a machine learning (ML)-based computer vision algorithm (e.g., a deep convolutional neural network), and $\mathcal{X}(d, e)$ is an image taken by a UAV at a distance $d$, given true damage state $e$. The variable $\hat{e}$ is an uncertain quantity, due to the uncertainty in ML-based computer vision algorithms and environmental noises $\boldsymbol{\epsilon}$, such as contamination on the structural surface (see Fig. 5.3), light conditions, and weather conditions.

The various uncertainty sources make it difficult to precisely detect and evaluate the damage. The damage estimated from observations $e_o$ may be considered as one realization of a random variable (i.e., $\hat{e}$) that is characterized by various uncertainty sources in the detection process for any given true damage state $e$. For the sake of explanation in this chapter, we assume that the difference between observations $e_o$ obtained from the UAV and the true damage state $e$ follows a normal distribution

$$f_{\hat{e}}(e_o, d) = \frac{\exp(-0.5((e_o - \mu_e)/\sigma_e(e, d))^2)}{\sigma_e(e, d)\sqrt{2\pi}} = \frac{1}{\sigma_e(e, d)} \phi\left(\frac{e_o - e}{\sigma_e(e, d)}\right), \tag{5.2}$$

where $\phi(\cdot)$ is the probability density function (PDF) of a standard normal random variable, the mean value $\mu_e$ and standard deviation $\sigma_e(e, d)$ are assumed to be a function of the

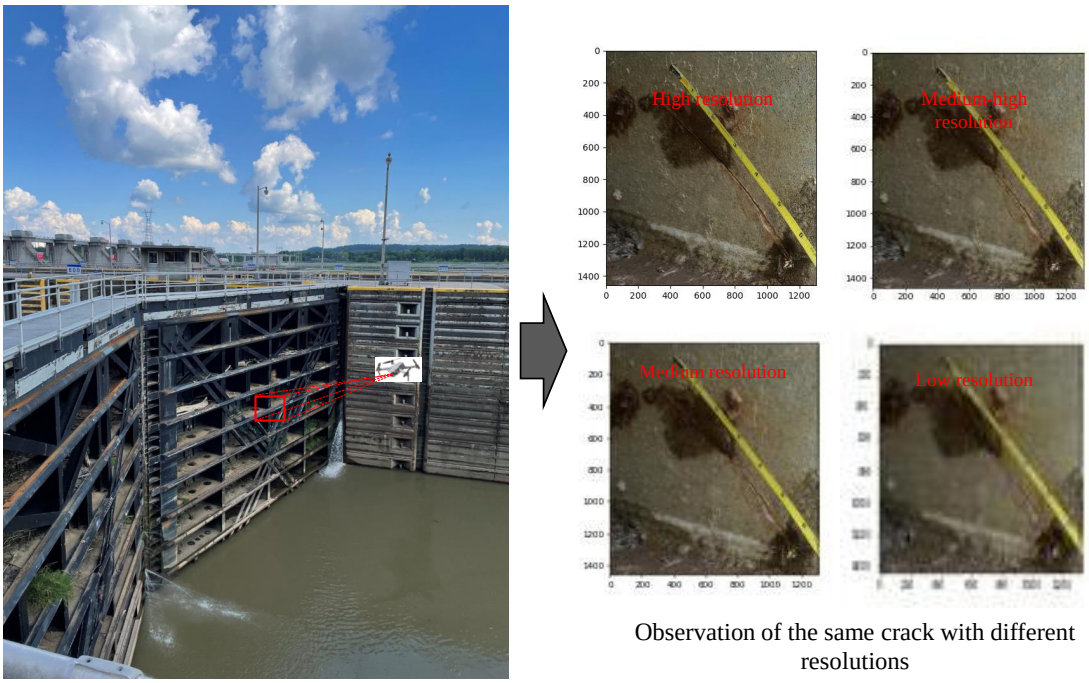



**Figure 5.3.** Illustration of impact of UAV flying distance on damage detection (a crack is used as an example).

damage state and the UAV flying distance as

$$\mu_e(e) = e; \sigma_e(e, d) = \alpha \frac{d}{e}, \tag{5.3}$$

where $\alpha$ is the coefficient that defines the contribution of the two variables $d$ and $e$, which should be given before the inspection mission. The contribution coefficient $\alpha$ depends on the resolution of the UAV-mounted camera which can be estimated by capturing a series of repeated images. An assumption is made here for the standard deviation that more severe damage with a smaller inspection distance can relate to more reliable detection of damage. By adjusting the coefficient $\alpha$, the assumption naturally accounts for the joint effect of the current damage state and inspection distance on the uncertainty level of acquired observation. In practice, obtaining the distribution given by Eq. (5.2) requires considering uncertainty sources in the computer vision algorithms, machine learning models, and

UAV inspection environments. Accurately calculating the distribution of observations will be a focus of our future work, as this chapter is primarily concerned with optimizing UAV inspections based on model-based analysis rather than uncertainty quantification of computer vision algorithms. The assumption in Eq. (5.2) is made solely for the purpose of explanation in this chapter, and any suitable density model form may be substituted.

To statistically quantify the damage-detection capability of observations from different UAV inspection distances, the probability of detection (PoD) is introduced that links inspection distance with observation uncertainty. Among a number of statistical models that were experimentally carried out to fit the PoD curve, the log-logistic distribution is one of the most widely used (Kwon and Frangopol (2011)). The PoD function can be expressed as follows (Georgiou (2007)),

$$p_{\mathcal{Z}}(e,d) = \Pr\{\mathcal{Z} = 1|(e,d)\} = \frac{\exp(\gamma(e,d) + \delta(e,d)\ln(e))}{1 + \exp(\gamma(e,d) + \delta(e,d)\ln(e))}, \tag{5.4}$$

where $\mathcal{Z}$ is the detectability indicator, i.e., $\{\mathcal{Z} = 1\}$ denotes that a damage is detected, while $\{\mathcal{Z} = 0\}$ denotes that a damage is not detected. $\gamma(e,d)$ and $\delta(e,d)$ are statistical parameters estimated according to Eq. (5.2), $\gamma(e,d) = -\delta(e,d) \cdot \mu_e(e)$ and $\delta(e,d) = -\pi/(\sqrt{3} \cdot \sigma_e(e,d))$.

### 5.4.3 Damage diagnostics and failure prognostics of miter gates based on UAV inspections

The observation model presented above provides observations of structural damage states from UAV inspections with uncertainty. This model can be used in conjunction with physics-based probabilistic analysis for damage diagnostics and failure prognostics of the miter gates. Fig. 5.4 illustrates how UAV observations can be integrated with physics-based analysis to update simulation models and compute the RUL. This computation necessitates a physics degradation model for future damage growth simulation based on current data.

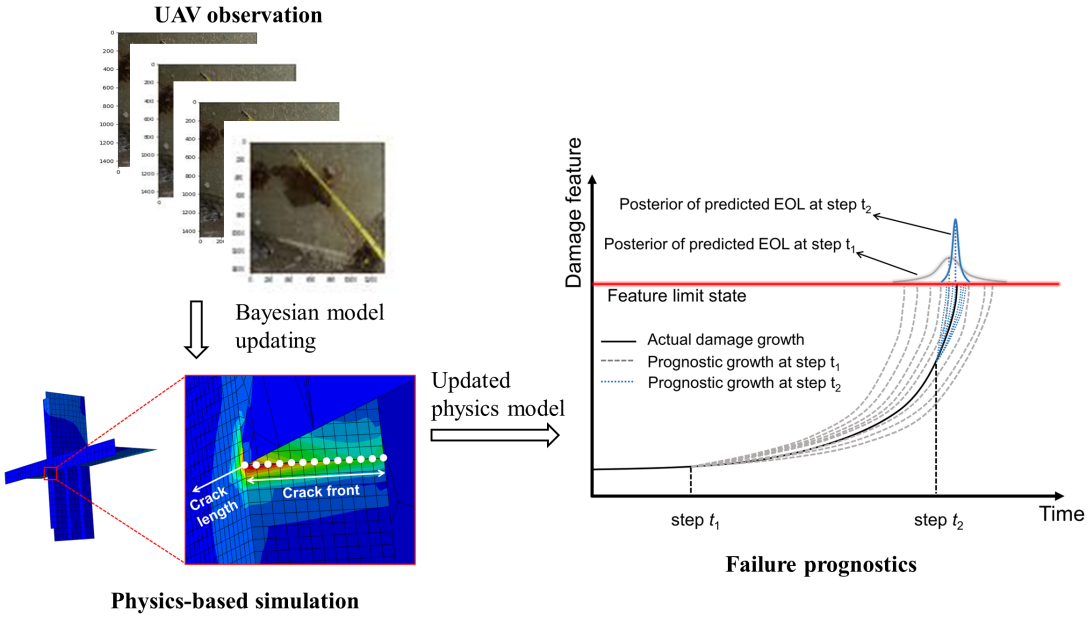



**Figure 5.4.** Main steps for damage diagnostics and failure prognostics using UAV observations.

While there are various structural degradation mechanisms that can be modeled using both empirical and analytical/physics-based approaches, the research scope for this chapter focuses on problems with self-accelerating degradation patterns, such as fatigue crack growth. Therefore, the degradation model is expressed in the form of Paris' Law as a reference to probe the performance of the proposed framework. The discrete-time form of the model is given by

$$\begin{aligned} e_i &= e_{i-1} + \Delta e_i, \\ \Delta e_i &= f(\Delta K_i, \boldsymbol{\theta}) = c(\Delta K_i)^m, \end{aligned} \tag{5.5}$$

where $\boldsymbol{\theta} = \{c,\ m\}$, $c$ and $m$ are unknown degradation model parameters, $\Delta K_i$ is the stress intensity factor (SIF) range at time step $t_i$, which indicates the trend of the growth of the damage based on the loading condition and current damage level. We represent $\Delta K_i$ as

$$\Delta K_i = G_K(e_{i-1},\ \Delta s_i(\mathbf{h}_i)), \tag{5.6}$$

where $\Delta s_i(\mathbf{h}_i)$ is the loading cycle range obtained using physics-based simulation as illustrated in Fig. 5.4, and $\mathbf{h}_i$ is the observed load vector at time step $t_i$. More details about this are available in Ref. (Wu, Fillmore, Vega, Hu and Todd (2022)). It is worth noting that the finite element used to compute $\Delta K_i$ as illustrated in Fig. 5.4 is computationally expensive to be directly used for diagnostics and prognostics. In our previous research (Wu et al. (2022)), we have constructed a surrogate model to reduce the required computational effort in predicting SIF. We direct interested readers to Wu et al. (2022) and Fillmore et al. (2022) for details of the surrogate modeling. In this chapter, the focus is given to explaining Bayesian model updating and failure prognostics using UAV inspections.

**Degradation model updating using UAV observations**

In order to update the aforementioned degradation model using UAV observations, a state-space model is first formulated that connects the UAV observation model with the degradation state-transition model as follows:

$$\begin{aligned} &\text{State transition} : \boldsymbol{\theta}_i = \boldsymbol{\theta}_{i-1} + \mathbf{r}_i,\ \boldsymbol{\theta}_i = [c_i, m_i], \\ &\qquad\qquad e_i = e_{i-1} + c_i(\Delta K_i)^{m_i}, \\ &\text{Measurement} : \hat{e}_i = \Psi(X(d, e_i), \boldsymbol{\epsilon}_i), \end{aligned} \tag{5.7}$$

where $\mathbf{r}_i$ denotes the process noise, which is modeled as a Gaussian distribution with zero mean and a known standard deviation. Note that the state and parameter estimations are performed together in the above equation, following the method discussed in Ref. Thelen, Zhang, Fink, Lu, Ghosh, Youn, Todd, Mahadevan, Hu and Hu (2022).

The loading condition $\mathbf{h}$ is measured at every time step, while observations $e_o$ are considered as not continuously available for all time steps because the UAV inspection may not be performed at every time step, and/or the time intervals between different

UAV inspections may be different. Define the inspection interval between the $(j+1)^{th}$ inspection and its previous inspection as $T_j$, we can represent the state-space equation given in Eq. (5.7) as Fig. 5.5.

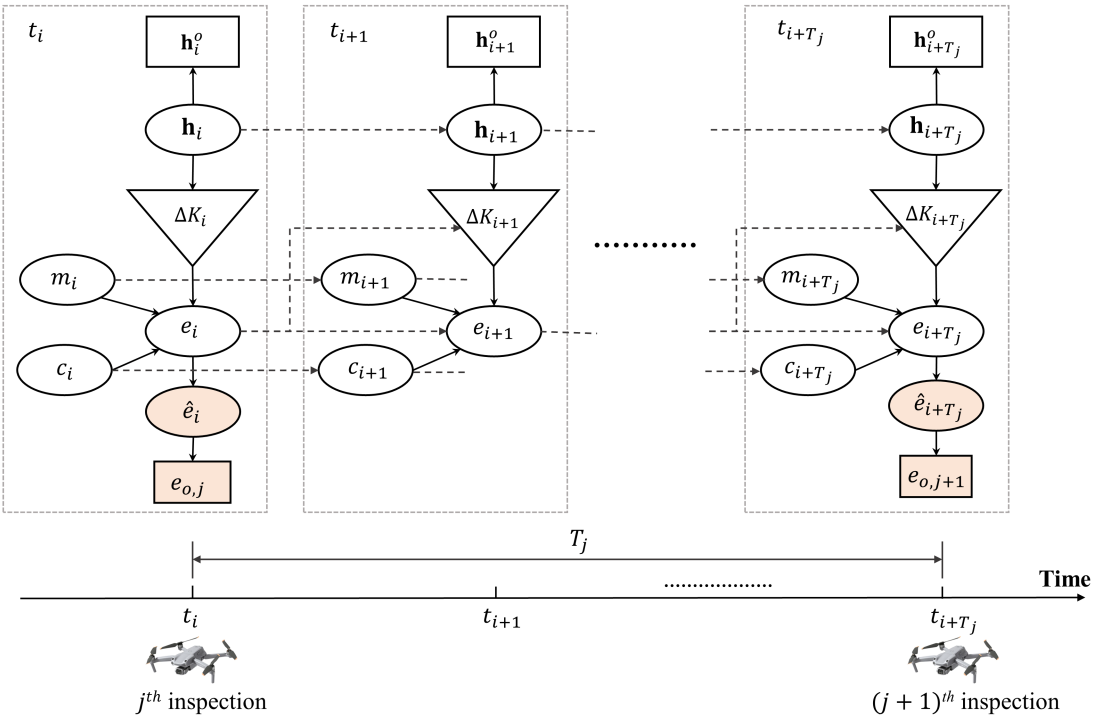



**Figure 5.5.** Connection between UAV inspection and physics-based degradation model ("Triangle node" represents deterministic function relationship; "Elliptical node" represents probabilistic relationship; and "Rectangle node" denotes observations.)

Assume that the $j^{th}$ UAV inspection is performed at $t_i$ (as illustrated in Fig. 5.5), we denote the observations from the previous inspections as $e_{o,1:j} \doteq \{e_{o,1}, e_{o,2}, \cdots, e_{o,j}\}$. If a damage is detected from the observation at $(j+1)^{th}$ inspection as $e_{o,j+1}$, according to the probabilistic connection given in Fig. 5.5, we can then use $e_{o,j+1}$ at time step $t_{i+T_j}$ (i.e., the time when the $(j+1)^{th}$ inspection is performed) to update $\boldsymbol{\theta}$ as follows

$$f((\boldsymbol{\theta}_{i+T_j}, e_{i+T_j})|e_{o,1:j+1},\ \mathbf{h}^o_{1:(i+T_j)}) \propto f(e_{o,j+1}|\ e_{i+T_j}) f((\boldsymbol{\theta}_{i+T_j}, e_{i+T_j})|e_{o,1:j},\ \mathbf{h}^o_{1:(i+T_j)}), \quad (5.8)$$

in which $f(e_{o,j+1}|\ e_{i+T_j})$ is obtained by some computer vision techniques as in Eq. (5.1).

By assuming the relationship between the true damage state and observed damage as a Gaussian distribution in this chapter, $f(e_{o,j+1}|\ e_{i+T_j})$ is then computed using Eq. (5.2).

$f((\boldsymbol{\theta}_{i+T_j}, e_{i+T_j})|e_{o,1:j},\ \mathbf{h}^o_{1:(i+T_j)})$ is given by

$$\begin{aligned} &f((\boldsymbol{\theta}_{i+T_j}, e_{i+T_j})|e_{o,1:j},\ \mathbf{h}^o_{1:(i+T_j)}) \\ &= \int\int f((e_{i+T_j}, \boldsymbol{\theta}_{i+T_j})|\ \boldsymbol{\theta}_i,\ e_i, \mathbf{h}^o_{(i+1):(i+T_j)}) f(\boldsymbol{\theta}_i, e_i|e_{o,1:j},\ \mathbf{h}^o_{1:i}) de_i d\boldsymbol{\theta}_i. \end{aligned} \tag{5.9}$$

Eqs. (5.8) and (5.9) are analytically intractable. In this chapter, the particle filter method is employed to solve the above two equations to obtain the posteriors of $\boldsymbol{\theta}$ and $e$ at time step $t_{i+T_j}$ (i.e., $(j+1)^{th}$ inspection). Denote the posterior particles of $\boldsymbol{\theta}$ and $e$ at time step $t_i$ (i.e., after the $j^{th}$ inspection) as ${\boldsymbol{\theta}_i}^{(k)}, {e_i}^{(k)},\ k = 1, \cdots, N_p$, where $N_p$ is the number of particles, the prior particles of $\boldsymbol{\theta}$ at time step $t_{i+T_j}$ (i.e., $(j+1)^{th}$ inspection) can be first obtained through uncertainty propagation based on the following equation

$$\boldsymbol{\theta}^{(k)}_q = \boldsymbol{\theta}^{(k)}_{q-1} + \mathbf{r}_q,\ \forall q = (i+1), \cdots, (i+T_j); k = 1, \cdots, N_p. \tag{5.10}$$

After that, we obtain the prior particles of $e_{i+T_j}$ through uncertainty propagation based on the state-transition equation given in Eq. (5.7) and Fig. 5.5 as

$$e^{(k)}_q = e^{(k)}_{q-1} + c^{(k)}_q (\Delta K^{(k)}_q)^{m^{(k)}_q},\ \forall q = (i+1),\ \cdots, (i+T_j);\ k = 1, \cdots, N_p, \tag{5.11}$$

where $\Delta K^{(k)}_q = G_K(e^{(k)}_{q-1},\ \Delta s_q(\mathbf{h}^o_q)),\ \forall q = (i+1),\ \cdots, (i+T_j);\ k = 1, \cdots, N_p$, as given in Eq. (5.6)

From the above two equations, we have the prior particles of $\boldsymbol{\theta}$ and $e$ at time $t_{i+T_j}$ as $\theta^{(k)}_{i+T_j}, e^{(k)}_{i+T_j},\ k = 1, \cdots, N_p$, and denote the corresponding marginal prior distribution for the $(j+1)^{th}$ inspection as $f'_{\boldsymbol{\theta}_{i+T_j}|e_{o,1:j}}(\boldsymbol{\theta})$ and $f'_{e_{i+T_j}|e_{o,1:j}}(e)$. Based on the prior particles $e^{(k)}_{i+T_j},\ k = 1, \cdots, N_p$, we have $L^{(k)} = f(e_{o,j+1}|e^{(k)}_{i+T_j})$ and the particle weights can be

computed by

$$w^{(k)} = \frac{L^{(k)}}{\sum_{r=1}^{N_p} L^{(r)}} = \frac{\phi\left(\frac{e_{o,j+1} - e_{i+T_j}^{(k)}}{\sigma_e(e_{i+T_j}^{(k)}, d)}\right)}{\sum_{r=1}^{N_p} \phi\left(\frac{e_{o,j+1} - e_{i+T_j}^{(r)}}{\sigma_e(e_{i+T_j}^{(r)}, d)}\right)}, \forall k = 1, \cdots, N_p. \tag{5.12}$$

Using $\{w^{(k)}\}_{k=1}^{N_p}$ as particle weights, all the particles can then be re-sampled using the particle filter method. Thus, the posterior particles of $\boldsymbol{\theta}$ and $e$ at time step $t_{i+T_j}$ can be obtained using the $(j+1)^{th}$ UAV inspection. Denoting the posterior particles after the $(j+1)^{th}$ UAV inspection as $\boldsymbol{\theta}_{po,\, i+T_j}^{(k)}, e_{po,\, i+T_j}^{(k)},\ k = 1, \cdots, N_p$ and the corresponding marginal posterior distributions as $f''_{\boldsymbol{\theta}_{i+T_j}|e_{o,1:(j+1)}}(\boldsymbol{\theta}|e_{o,j+1})$ and $f''_{e_{i+T_j}|e_{o,1:(j+1)}}(e|e_{o,j+1})$.

Please note that in Fig. 5.5, for the sake of simplicity, we only demonstrated a one-time diagnostic process based on two consecutive inspections. This one-time updating involves the inspection distance and the subsequent inspection time. In practice, the process of online diagnostics involves continuous iterations while each subsequent iteration follows the same pattern: using observations from the current inspection to update prior information, which further informs model-based failure prognostics.

The next section explains the procedure of performing failure prognostics accounting for UAV detectability and maintenance protocols.

**Failure prognostics accounting for UAV detectability and maintenance protocols**

Model-based failure prognostics predict the RUL of the structure, providing essential information for the system reliability and safety. To optimize the entire integrated workflow for minimal risk/cost, both the UAV detectability and maintenance protocols are incorporated through the prognostic process. The UAV detectability is introduced to include the uncertainties existing in the observations based on detection theory. Considering the impact of maintenance protocols during the failure prognostics enables the reduction

of maintenance costs and eventually maximizes operational availability.

**(a) Generalized failure prognostics using a model-based method**

We assume those $N_p$ particles are simulated based on the current priors of the damage degradation model parameters $\boldsymbol{\theta}$ and damage state $e$. By numerically collecting the intersection points, i.e., end of life (EOL), between simulated damage growth curves and damage limit state, a distribution of RUL can be obtained by subtracting the current time step from EOL. The probability of failure is then defined as follows

$$F_{\mathrm{RUL}}(t;d) = \Pr\{\,\mathrm{RUL} \leq t|\, d\} \approx \frac{N_{t,d}}{N_p}, \tag{5.13}$$

where $N_{t,d}$ is the number of samples with a damage level greater than the limit state (denote as $e_h$) at time step $t$, and $N_p$ is the total number of samples. Please note that $d$ in the subscript of $N_{t,d}$ is employed to imply that the estimated number of failure samples is affected by the inspection distance $d$. $N_{t,d}$ is given by

$$N_{t,d} = \sum_{k=1}^{N_p} (I(e_d^{(k)}(t) > e_h)), \tag{5.14}$$

where $I(\mathcal{E})$ is an indicator function given by

$$I(\mathcal{E}) = \begin{cases} 1, & \mathcal{E} \text{ is true} \\ 0, & \text{otherwise} \end{cases} \tag{5.15}$$

and $e_d^{(k)}(t)$ is the $k^{th}$ sample generated through uncertainty propagation (UP) using the following equation

$$e_d^{(k)}(t) = G_{\mathrm{UP}}(e_{0,d}^{(k)},\ \boldsymbol{\theta}_{0,d}^{(k)},\ t),\ \forall k = 1, \cdots, N_p \tag{5.16}$$

in which $G_{\text{UP}}(\cdot)$ represents uncertainty propagation by executing the state-transition equation in Eq.(5.7) and accounting for uncertainty in the load conditions (i.e., $\mathbf{h}$) for a time duration of $t$ with an initial damage state $e_{0,d}^{(k)}$ and initial model parameters $\boldsymbol{\theta}_{0,d}^{(k)}$. During the uncertainty propagation, there is no model updating performed since no observations are collected during this time period.

Due to the impact of damage detectability in the UAV inspection and maintenance protocols, the PDFs used to generate $e_{0,d}^{(k)}$ and $\boldsymbol{\theta}_{0,d}^{(k)}, k = 1, \cdots, N_p$ will be different for different scenarios. The difference in the initial damage state and model parameters ultimately leads to different failure prognostic results. In what follows, we discuss how to select appropriate PDFs to generate $\{e_{0,d}^{(k)}\}_{k=1}^{N_p}$ and $\{\boldsymbol{\theta}_{0,d}^{(k)}\}_{k=1}^{N_p}$ for failure prognostics by accounting for the impact of damage detectability and maintenance protocols on prognostics.

**(b) Impact of damage detectability on failure prognostics after the $j^{th}$ inspection**

If an observation $e_{o,j+1}$ is collected during the $(j+1)^{th}$ UAV inspection, it means that damage diagnostics have been performed by following the procedure presented in Sec. 5.4.3 and we have $e_{0,d}^{(k)}$ and $\boldsymbol{\theta}_{0,d}^{(k)}$ as

$$e_{0,d}^{(k)} = e_{po,i+T_j}^{(k)} \text{ and } \boldsymbol{\theta}_{0,d}^{(k)} = \boldsymbol{\theta}_{po,i+T_j}^{(k)}, \ \forall k = 1, \cdots, N_p. \tag{5.17}$$

In other words, the posterior distributions $f''_{\boldsymbol{\theta}_{i+T_j}|e_{o,1:(j+1)}}(\boldsymbol{\theta}|e_{o,j+1})$ and $f''_{e_{i+T_j}|e_{o,1:(j+1)}}(e|e_{o,j+1})$ after model updating using observation $e_{o,j+1}$ from the $(j+1)^{th}$ inspection are used to generate samples of $\{e_{0,d}^{(k)}\}_{k=1}^{N_p}$ and $\{\boldsymbol{\theta}_{0,d}^{(k)}\}_{k=1}^{N_p}$ for failure prognostics using Eqs. (5.13) through (5.16).

If no damage is detected in the $(j+1)^{th}$ inspection (i.e., $e_{o,j+1} = \emptyset$ or $\Pr\{\mathcal{Z} = 0\}$), $e_{0,d}^{(k)}, \ \forall k = 1, \cdots, N_p$ should be generated based on both the prior distribution (i.e.,

$f'_{e_{i+T_j}|e_{o,1:j}}(e)$) of the damage state at the $(j+1)^{th}$ inspection and the fact that no damage is detected.

To this end, the PDF of the initial damage state used for failure prognostics is derived as follows

$$\begin{aligned} f_{e|\mathcal{Z}=0}(e|\mathcal{Z}=0,d) &= \frac{f(\mathcal{Z}=0|e)f'_{e_{i+T_j}|e_{o,1:j}}(e)}{\int f(\mathcal{Z}=0|e)f'_{e_{i+T_j}|e_{o,1:j}}(e)de}, \\ &= \frac{(1-p_{\mathcal{Z}}(e,\ d))f'_{e_{i+T_j}|e_{o,1:j}}(e)}{\int (1-p_{\mathcal{Z}}(e,\ d))f'_{e_{i+T_j}|e_{o,1:j}}(e)de}, \end{aligned} \tag{5.18}$$

where $f'_{e_{i+T_j}|e_{o,1:j}}(e)$ is a prior distribution of $e_{i+T_j}$ obtained through uncertainty propagation in Sec. 5.4.3, and $p_{\mathcal{Z}}(e,\ d)$ is given in Eq. (5.4).

Since during the planning of the $(j+1)^{th}$ UAV inspection, we do not have any observation of $e_{o,j+1}$ yet. Thus, the detectability is uncertain and depends on the underlying true damage state $e_{i+T_j}$. For given true damage state $e_{i+T_j}$, after considering the uncertainty in the observation (i.e., Eq. (5.2)), we have the PDF of the initial damage state (i.e., $e^{(k)}_{0,d}$) for failure prognostics as

$$\begin{aligned} f_{e_0}(e|(d,e_{i+T_j})) &= \begin{cases} \int f''_{e_{i+T_j}|e_{o,1:(j+1)}}(e|e_{o,j+1})\phi\left(\frac{e_{o,j+1}-e_{i+T_j}}{\sigma_e(e_{i+T_j},d)}\right)de_{o,j+1}, & \mathcal{Z}=1 \\ f_{e|\mathcal{Z}=0}(e|\mathcal{Z}=0,d), & \mathcal{Z}=0 \end{cases} \\ &= p_{\mathcal{Z}}(e_{i+T_j},d)\int f''_{e_{i+T_j}|e_{o,1:(j+1)}}(e|e_{o,j+1})\phi\left(\frac{e_{o,j+1}-e_{i+T_j}}{\sigma_e(e_{i+T_j},d)}\right)de_{o,j+1} \\ &+ (1-p_{\mathcal{Z}}(e_{i+T_j},d))f_{e|\mathcal{Z}=0}(e|\mathcal{Z}=0,d), \end{aligned} \tag{5.19}$$

where $f_{e|\mathcal{Z}=0}(e|\mathcal{Z}=0,d)$ is given in Eq. (5.18) and $f''_{e_{i+T_j}|e_{o,1:(j+1)}}(e|e_{o,j+1})$ is the posterior distribution of $e_{i+T_j}$ if there is an observation $e_{o,j+1}$.

Figure 5.6 summarizes the procedure of determining an appropriate PDF of $e_{0,d}$ for failure prognostics by accounting for UAV detectability. Note that the cumulative density

function (CDF) of the RUL becomes a function of $d$ and $e_{i+T_j}$ as shown in this figure because there is no observation available yet during the planning of the $(j+1)^{th}$ UAV inspection.

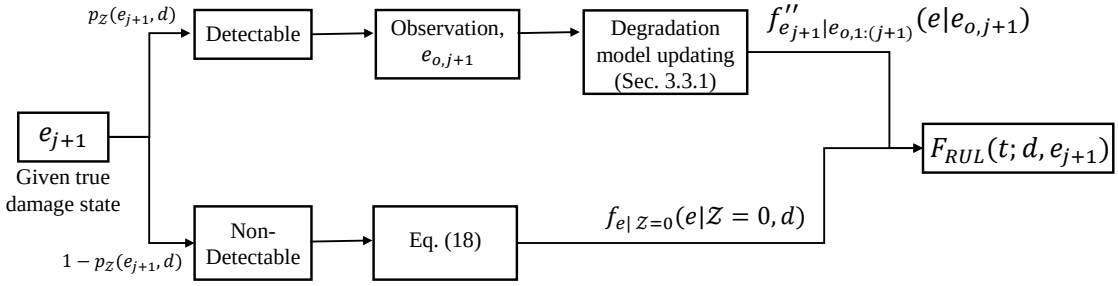



**Figure 5.6.** Flowchart of determining an appropriate PDF of $e_{0,d}$ for failure prognostics by accounting for UAV detectability.

Similarly, we determine the PDF used to generate samples of $\{\boldsymbol{\theta}_{0,d}^{(k)}\}_{k=1}^{N_p}$ as

$$
\begin{aligned}
&f_{\boldsymbol{\theta}_0}(\boldsymbol{\theta}|(d, e_{i+T_j})) \\
&= p_{\mathcal{Z}}(e_{i+T_j}, d) \int f''_{\boldsymbol{\theta}_{i+T_j}|e_{o,1:(j+1)}}(\boldsymbol{\theta}|e_{o,j+1}) \phi\left(\frac{e_{o,j+1} - e_{i+T_j}}{\sigma_e(e_{i+T_j}, d)}\right) de_{o,j+1} \\
&+ (1 - p_{\mathcal{Z}}(e_{i+T_j}, d)) f'_{\boldsymbol{\theta}_{i+T_j}|e_{o,1:j}}(\boldsymbol{\theta}).
\end{aligned} \tag{5.20}
$$

**(c) Impact of maintenance protocols on failure prognostics after the $(j+1)^{th}$ inspection**

Let the maintenance threshold be $e_m$, which means if observed damage is greater than $e_m$, a repair action will be performed. Otherwise, no repair is needed. Thus, if $e_{o,j+1} > e_m$, a repair will be performed. Generally, the damaged area will be considered "sufficiently healthy" after a repair action. Such a sufficiently healthy state can be represented by assigning a very small initial damage state $e_{0,d}$ (instead of zero), without ruling out the possibility of failure re-generation. $e_{0,d}$ follows a normal distribution with a standard deviation of $\sigma_{ini}$. Thus, the samples of the initial damage state (i.e., $e_{0,d}^{(k)}$) for failure

prognostics are generated through the following PDF

$$f_{e_0}(e|d, e_{o,j+1}) = \frac{1}{\sigma_{ini}}\phi\left(\frac{e - e_{ini}}{\sigma_{ini}}\right), \tag{5.21}$$

where $e_{ini}$ and $\sigma_{ini}$ are the mean and standard deviation of the initial damage state after repair.

If $e_{o,j+1} \leq e_m$, no repair action is taken, $e_0^{(k)}$ and $\boldsymbol{\theta}_d^{(k)}$ are generated from the posterior distributions $f''_{\theta_j|e_{o,1:(j+1)}}(\theta|e_{o,j+1})$ and $f''_{e_{i+T_j}|e_{o,1:(j+1)}}(e|e_{o,j+1})$ after degradation model updating using observation from the $(j+1)^{th}$ inspection. This is the same as that in Eq. (5.17). Therefore, for given observation $e_{o,j+1}$, the PDF of the initial damage state (i.e., $e_{0,d}^{(k)}$) for failure prognostics after accounting for maintenance protocols is given by

$$f_{e_0}(e|e_{o,j+1}, e_m) = \begin{cases} f''_{e_{i+T_j}|e_{o,1:(j+1)}}(e|e_{o,j+1}), \mathcal{Z} = 1,\ e_{o,j+1} \leq e_m \\ \frac{1}{\sigma_{ini}}\phi\left(\frac{e-e_{ini}}{\sigma_{ini}}\right), \mathcal{Z} = 1,\ e_{o,j+1} > e_m \end{cases}. \tag{5.22}$$

Similar to what we discussed above, there is no observation available yet during the planning of the $(j+1)^{th}$ UAV inspection. $e_{o,j+1}$ will be an uncertain quantity conditioned on the true damage state. For given true damage state $e_{i+T_j}$, after accounting for both damage detectability and maintenance protocols, the PDF of the initial damage state (i.e., $e_{0,d}^{(k)}, k = 1, \cdots, N_p$) for failure prognostics given in Eq. (5.19) is written as

$$\begin{aligned} f_{e_0}(e|(d, e_{i+T_j}, e_m)) = p_{\mathcal{Z}}(e_{i+T_j},\ d) \Bigg\{ & \int_0^{e_m} f''_{e_{i+T_j}|e_{o,1:(j+1)}}(e|e_{o,j+1})\phi\left(\frac{e_{o,j+1} - e_{i+T_j}}{\sigma_e(e_{i+T_j}, d)}\right) de_{o,j+1} \\ & + \frac{1}{\sigma_{ini}}\phi\left(\frac{e - e_{ini}}{\sigma_{ini}}\right)\left(1 - \Phi\left(\frac{e_m - e_j}{\sigma_e(e_{i+T_j}, d)}\right)\right)\Bigg\} \\ & + (1 - p_{\mathcal{Z}}(e_{i+T_j},\ d)) f_{e|\mathcal{Z}=0}(e|\mathcal{Z} = 0, d), \end{aligned} \tag{5.23}$$

where $\Phi(\cdot)$ is the CDF of a standard Gaussian random variable.

The procedure of determining an appropriate PDF for the initial damage state for failure prognostics given in Fig. 5.6 is revised accordingly as shown in Fig. 5.7. As indicated in this figure, the CDF function of RUL becomes a function of $d$, $e_{i+T_j}$, and $e_m$ after considering both damage detectability and maintenance protocols.

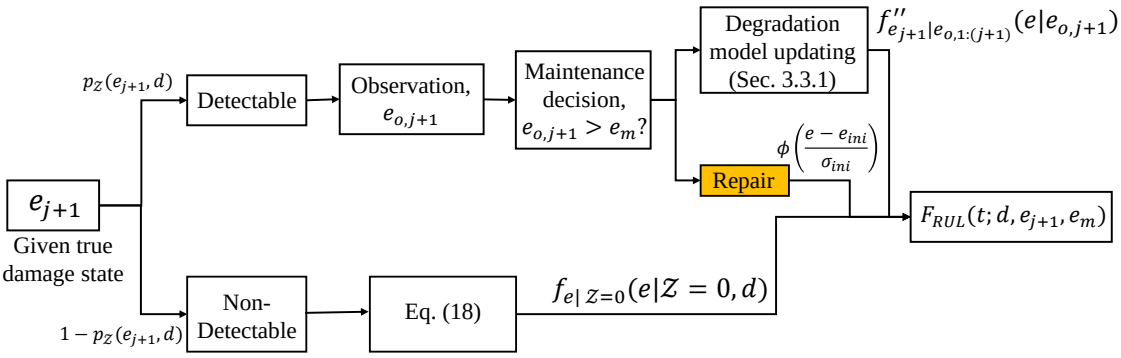



**Figure 5.7.** Flowchart of model-based prognostics accounting for both damage detectability and maintenance protocols.

The PDF used to generate samples of $\{\boldsymbol{\theta}_{0,d}^{(k)}\}_{k=1}^{N_p}$ is derived similarly as

$$f_{\boldsymbol{\theta}_0}(\boldsymbol{\theta}|(d, e_{i+T_j}, e_m)) = p_{\mathcal{Z}}(e_{i+T_j},\ d)\left\{\int_0^{e_m} f''_{\boldsymbol{\theta}_{i+T_j}|e_{o,1:(j+1)}}(\boldsymbol{\theta}|e_{o,j+1})\phi\left(\frac{e_{o,j+1}-e_{i+T_j}}{\sigma_e(e_{i+T_j}, d)}\right)de_{o,j+1}\right.$$
$$\left. + \frac{1}{\sigma_{\boldsymbol{\theta},ini}}\phi\left(\frac{\boldsymbol{\theta}-\boldsymbol{\theta}_{ini}}{\sigma_{\boldsymbol{\theta},ini}}\right)\left(1-\Phi\left(\frac{e_m-e_{i+T_j}}{\sigma_e(e_{i+T_j}, d)}\right)\right)\right\}$$
$$+ (1 - p_{\mathcal{Z}}(e_{i+T_j},\ d))f'_{\boldsymbol{\theta}_{i+T_j}|e_{o,1:j}}(\boldsymbol{\theta}), \tag{5.24}$$

where $\boldsymbol{\theta}_{ini}$ and $\sigma_{\boldsymbol{\theta},ini}$ are respectively the mean and standard deviation of $\boldsymbol{\theta}$ after performing a repair.

### 5.4.4 Failure prognostics under multi-mode failure based on UAV inspection

During the $j^{th}$ inspection, the UAV may detect multiple damage modes (as illustrated in Fig. 5.8). Note that damage modes here refer to different types of damage

mechanisms that can occur at the same or different sites, or the same damage mechanism presented at different locations. In this section, we denote the damage state of the $c$-th damage mode at the $j^{th}$ inspection as $e_{j,c}, c = 1, \cdots, N_c$, where $N_c$ is the total number of damage modes. The structure will fail if any of the damage states reaches its damage threshold $e_{h,c}$, as described in Sec. 5.4.3. For each of the damage modes, the degradation model updating procedure will be the same as what we discussed in Sec. 5.4.3. Again, the observations of the $(j+1)^{th}$ inspection for these damaged modes are uncertain since the inspection has not been performed yet.

For given inspection distance $d$, maintenance protocols $e_m$, and true damage states $e_{i+T_j,c}, c = 1, \cdots, N_c$, the probability of failure in the presence of multiple failure modes can be computed by

$$\begin{aligned} &F_{sys}(t; d,\ e_m,\ \{e_{i+T_j,c}\}_{c=1}^{N_c}) \\ &\approx \frac{\sum_{k=1}^{N_p} I\left(((e_{d,1}^{(k)}(t)|e_m, e_{i+T_j,1}) > e_{h,1}) \cup \cdots \cup ((e_{d,N_c}^{(k)}(t)|e_m, e_{i+T_j,N_c}) > e_{h,N_c})\right)}{N_p}, \end{aligned} \tag{5.25}$$

where "$\cup$" stands for "union" of events, and $(e_{d,c}^{(k)}(t)|e_m, e_{i+T_j,c}),\ \forall c = 1, \cdots, N_c$ is the damage state of the $c$-th damage mode forecast after $t$ time steps conditioned on $e_m$ and the true damage state $e_{i+T_j,c}$, is

$$\begin{aligned} &(e_{d,c}^{(k)}(t)|e_m, e_{i+T_j,c}) \\ &= G_{c,\mathrm{UP}}((e_{0,d,c}^{(k)}|e_m, e_{i+T_j,c}),\ (\boldsymbol{\theta}_{0,d,c}^{(k)}|e_m, e_{i+T_j,c}),\ t),\ \forall k = 1, \cdots, N_p;\ c = 1, \cdots, N_c; \end{aligned} \tag{5.26}$$

where $G_{c,\mathrm{UP}}(\cdot)$ represents uncertainty propagation using the state-transition equation of the $c$-th damage mode (see the explanation in Eq. (5.16)) and the PDFs used to generate the samples of the initial damage state $e_{0,d,c}^{(k)}$ and degradation model parameters $\boldsymbol{\theta}_{0,d,c}^{(k)}$ are determined similarly using the methods discussed in Sec. 5.4.3 (i.e., Eqs. (5.23) and (5.24)).

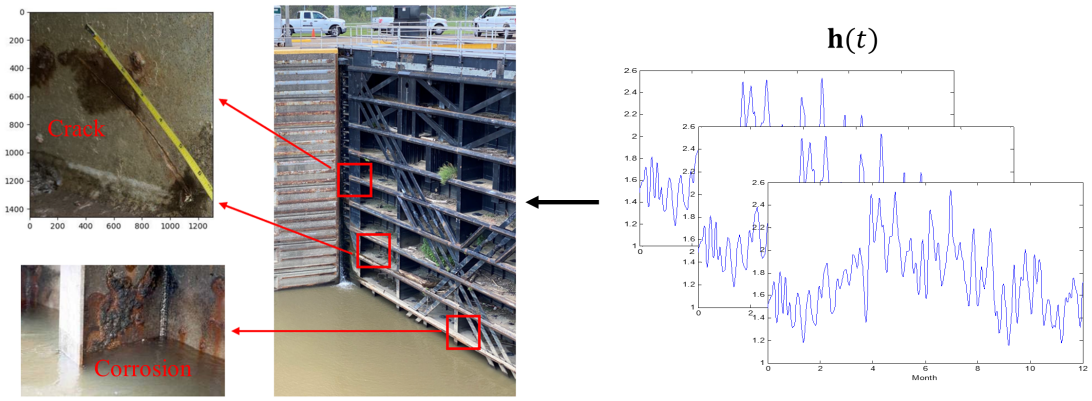



**Figure 5.8.** Illustration of multiple damage modes identified during an inspection.

In addition, it is worth noting that due to the shared load conditions (e.g., $\mathbf{h}(t)$ in Eq. (5.6)) among different damage modes (as illustrated in Fig. 5.8), samples of $e_{d,c}^{(k)}(t), \forall k = 1, \cdots, N_p;\ c = 1, \cdots, N_c$ have complicated correlations over time. This correlation is automatically accounted for during the sampling process by using the same group of samples of $\mathbf{h}(t)$ during failure prognostics. In the presence of multiple damage modes (e.g., cracking, corrosion, or cracking at different locations), the damage levels may have different physical meanings with different magnitudes. To address this issue in making maintenance protocols, a non-dimensional maintenance threshold $e_m$ can be used. This universal maintenance threshold can be converted to an individualized maintenance threshold ($e_{r,c},\ c = 1, \cdots, N_c$) for each damage type as follows

$$e_{r,c} = e_m(e_{c,U} - e_{c,L}) + e_{c,L},\ c = 1, \cdots, N_c, \tag{5.27}$$

where $e_{c,L}$ and $e_{c,U}$ are respectively the lower and upper bounds of the damage level of the $c$-th damage type.

After the conversion of the universal maintenance threshold to individualized maintenance thresholds, $e_{r,c}$ can be applied in Sec. 5.4.3 for failure prognostics (i.e.,

replacing $e_m$ with $e_{r,c}$ for each damage mode in Eqs. (5.23) and (5.24)). For example, Eq. (5.23) can be rewritten for the $c$-th damage type as

$$\begin{aligned} & f_{e_{0,c}}(e|(d, e_{i+T_j,c}, e_m)) \\ & = p_{\mathcal{Z}}(e_{i+T_j,c},\ d) \left\{ \int_0^{e_{r,c}} f''_{e_{i+T_j,c}|e_{o,1:(j+1),c}}(e|e_{o,j+1,c}) \phi\left(\frac{e_{o,j+1,c} - e_{i+T_j,c}}{\sigma_e(e_{i+T_j,c}, d)}\right) de_{o,j+1,c} \right. \\ & \qquad \left. + \phi\left(\frac{e - e_{ini,c}}{\sigma_{ini,c}}\right)\left(1 - \Phi\left(\frac{e_{r,c} - e_{i+T_j,c}}{\sigma_e(e_{i+T_j,c}, d)}\right)\right)\right\} \\ & \qquad + (1 - p_{\mathcal{Z}}(e_{i+T_j,c},\ d)) f_{e|\mathcal{Z}=0}(e|\mathcal{Z} = 0, d), \end{aligned} \tag{5.28}$$

where $e_{r,c}$ is a function of $e_m$ given in Eq. (5.27), $e_{ini,c}$ and $\sigma_{ini,c}$ are respectively the mean and standard deviation of the initial damage state of the $c^{th}$ damage mode after repair, and $e_{o,j+1,c}$ is the observation of the $c^{th}$ damage mode at the $(j+1)^{th}$ inspection.

For the scenario that all the damage mechanisms are the same (i.e., cracks), the above transformation given in Eq. (5.27) is not necessary and $e_m$ can be directly used as a variable to determine maintenance decision. As shown in the above discussions, the probability of failure of the structure of interest will be affected by the UAV flying distance $d$, the critical maintenance threshold $e_m$, and the next inspection interval $T_I$. Next, we discuss how to optimize these parameters to minimize the expected life-cycle cost.

## 5.5 Optimization of the UAV Inspection Program

### 5.5.1 Overview of the optimization model

The above process of damage diagnostics and prognostics of the miter gate indicates that selecting a smaller inspection distance and repair action limit may result in a more reliable observation and lower failure probability. However, such a strategy may come at a high cost in terms of inspection and maintenance expenses. In order to quantitatively evaluate the impact of each decision, an optimization model is introduced

that comprehensively takes into account both the expected risk and cost. The primary purpose of the optimization model is to minimize the cost per unit time (CPUT) by optimizing the values of the inspection distance $d$, the critical maintenance threshold $e_m$, and the next inspection interval $T_I$. The optimization model can be expressed as follows

$$d^*, e_m^*, T_I^* = \text{argmin}\left\{\frac{EC(d, e_m, T_I)}{ET(d, e_m, T_I)}\right\}, \tag{5.29}$$

where $EC(d, e_m, T_I)$ is the expected operational cost, and $ET(d, e_m, T_I)$ is the expected operational time.

### 5.5.2 Cost model

In this chapter, the cost model incorporates the cost of UAV inspections, repair actions, and potential structural failure consequence costs prior to the next inspection. These costs are affected by the UAV inspection parameters, i.e., the inspection distance $d$, the inspection interval $T_I$, and the repair damage limit $e_m$. In what follows, we provide a detailed explanation of each term in the cost model.

**Inspection cost**

The inspection cost is assumed to be a parameter solely related to distance. As illustrated in Fig. 5.2 and discussed in Ref. Zeng et al. (2023), a smaller inspection distance results in a smaller inspection view range and therefore a smaller number of viewpoints of interest. This indicates that a larger inspection distance will lead to a smaller flying distance of the UAV to cover the structure of interest, while a smaller inspection distance increases the inspection cost $C_I$ due to an increase in the flight path. In theory, the UAV flying distance is a function of the inspection distance, and consequently, the inspection cost is also a function of the inspection distance (Zeng et al. (2023)). For the purposes of

illustration in this chapter, we assume that $C_I$ is a function of UAV inspection distance as

$$C_I(d) = \alpha_{C_I} \frac{1}{d}, \tag{5.30}$$

where $\alpha_{IC}$ is the coefficient of the inspection distance.

**Repair action cost**

The repair action cost is defined as the expected cost of repairing all components that have damage $e_{i+T_j,c}$ exceeding the repair threshold $e_{r,c}$ during $(j+1)^{th}$ UAV inspection mission. As mentioned above, $e_{r,c}$ is a function of $e_m$ as shown in Eq. (5.27). The expected repair cost $C_R$ can be calculated as follows

$$C_R((d, c_m)|\ \{e_{i+T_j,c}\}_{c=1}^{N_c}) = \sum_{c=1}^{N_c} C_{R,c}((d, c_m)|e_{i+T_j,c}), \tag{5.31}$$

where $C_{R,c}((d, c_m)|e_{i+T_j,c})$ is the expected repair cost of the $c$-th damage mode and is computed by

$$\begin{aligned}
&C_{R,c}((d, c_m)|e_{i+T_j,c}) \\
&= \Pr\{\mathcal{Z} = 1\} \left\{ \left(1 - \Phi\left(\frac{e_{r,c} - e_{i+T_j,c}}{\sigma_e(e_{i+T_j,c}, d)}\right)\right) C_r + \Phi\left(\frac{e_{r,c} - e_{i+T_j,c}}{\sigma_e(e_{i+T_j,c}, d)}\right) \times 0 \right\} + \Pr\{\mathcal{Z} = 0\} \times 0, \\
&= p_{\mathcal{Z}}(e_{i+T_j,c}, d) \left(1 - \Phi\left(\frac{e_{r,c} - e_{i+T_j,c}}{\sigma_e(e_{i+T_j,c}, d)}\right)\right) C_r,\ c = 1, \cdots, N_c,
\end{aligned} \tag{5.32}$$

where $C_r$ is the repair cost of the corresponding region/ component, $e_{r,c} = e_m$ if the studied damages belong to the same damage mechanism, otherwise, $e_{r,c}$ is computed based on $e_m$ using Eq. (5.27).

**Structural failure cost**

The expected structural failure cost $C_F$ is defined as the potential cost of failure if the subsequent inspection is performed after a period of $T_I$ following the current inspection. $C_F$ is computed by

$$C_F((d, c_m, T_I)|\ \{e_{i+T_j,c}\}_{c=1}^{N_c}) = C_{sys}F_{sys}(T_I; d,\ e_m,\ \{e_{i+T_j,c}\}_{c=1}^{N_c}), \tag{5.33}$$

where $C_{sys}$ is the failure cost ($C_{sys} >> C_r$) and $F_{sys}(T_I; d,\ e_m,\ \{e_{i+T_j,c}\}_{c=1}^{N_c})$ is computed using the method discussed in Sec. 5.4.4. The cost value $C_{sys}$ could include replacement costs, liability, life safety costs, or whatever other consequence losses exist.

**Total cost conditioned on possible true damage state vector**

Eventually, the total expected cost conditioned on the possible true damage state vector is expressed as

$$\begin{aligned} EC((d, e_m, T_I)|\{e_{i+T_j,c}\}_{c=1}^{N_c}) = \alpha_{C_I}\frac{1}{d^2} + \sum_{c=1}^{N_c} p_{\mathcal{Z}}(e_{i+T_j,c}, d)\left(1 - \Phi\left(\frac{e_{r,c} - e_{i+T_j,c}}{\sigma_e(e_{i+T_j,c}, d)}\right)\right) C_r \\ + C_{sys}F_{sys}(T_I; d,\ e_m,\ \{e_{i+T_j,c}\}_{c=1}^{N_c}). \end{aligned} \tag{5.34}$$

### 5.5.3 Operational time model

UAV inspections for large-scale structures are usually conducted at discrete intervals, spanning several months to years. The actual operational time, $T_O$, in such a long duration, is subject to uncertainty, represented by two scenarios illustrated in Fig. 5.9. The operational time $T_O$ of the structure depends on the time sequences of two events, namely structural failure and the subsequent inspection. Based on Fig. 5.9, we have the

operational time as

$$T_O = \begin{cases} T_F, & T_F \leq T_I \\ \\ T_I, & T_F > T_I \end{cases}, \tag{5.35}$$

where $T_F$ is the duration from the current inspection time to the time that failure occurs.

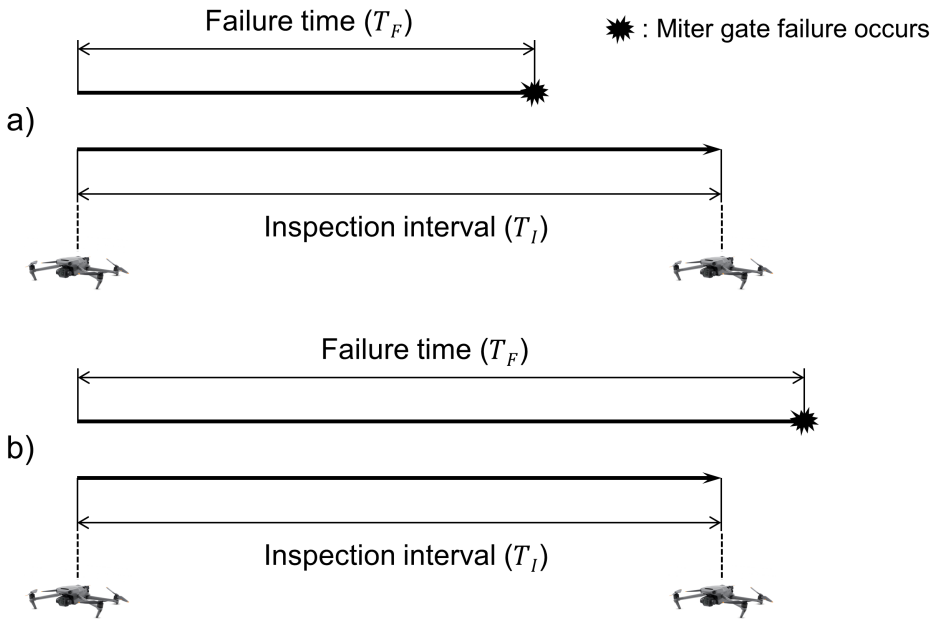



**Figure 5.9.** Operation time model: (a) Failure occurs before the next inspection, and (b) failure does not occur before the next inspection.

The expected operational time is therefore derived as

$$\begin{aligned}
&ET(d, e_m, T_I | \{e_{i+T_j,c}\}_{c=1}^{N_c}) = E(T_O | d, e_m, T_I, \{e_{i+T_j,c}\}_{c=1}^{N_c}), \\
&= T_I \Pr\{(T_F | (d, e_m, \{e_{i+T_j,c}\}_{c=1}^{N_c})) > T_I\} + \int_0^{T_I} t f_{sys}(t | d, e_m, \{e_{i+T_j,c}\}_{c=1}^{N_c})\, dt, \\
&= T_I (1 - F_{sys}(T_I | d, e_m, \{e_{i+T_j,c}\}_{c=1}^{N_c})) + \int_0^{T_I} (1 - F_{sys}(t | d, e_m, \{e_{i+T_j,c}\}_{c=1}^{N_c}))\, dt \\
&\quad - T_I (1 - F_{sys}(T_I | d, e_m, \{e_{i+T_j,c}\}_{c=1}^{N_c})), \\
&= \int_0^{T_I} (1 - F_{sys}(t | d, e_m, \{e_{i+T_j,c}\}_{c=1}^{N_c}))\, dt, \\
&= T_I - \int_0^{T_I} F_{sys}(t | d, e_m, \{e_{i+T_j,c}\}_{c=1}^{N_c})\, dt.
\end{aligned} \tag{5.36}$$

To consider the uncertainty in the true damage states $e_{i+T_j,c}, c = 1, \cdots, N_c$, i.e., $\{e_{i+T_j,c}\}_{c=1}^{N_c}$, the optimization model given in Eq. (5.29) is rewritten based on the prior information as

$$\begin{aligned} d^*, e_m^*, T_I^* &= \arg\min_{d,e_m,T} \{G_c(d, e_m, T_I)\}, \\ &= \text{argmin} \Bigg\{ \int \cdots \int \frac{EC(d, e_m, T_I | \{e_{i+T_j,c}\}_{c=1}^{N_c})}{ET(d, e_m, T_I | \{e_{i+T_j,c}\}_{c=1}^{N_c})} \\ & f'_{\{e_{o,1:j,c}\}_{c=1}^{N_c}}(e_{1,i+T_j}, \cdots, e_{N_c,i+T_j})\, de_{1,i+T_j} \ldots de_{N_c,i+T_j} \Bigg\}, \end{aligned} \tag{5.37}$$

where $G_c(d, e_m, T_I)$ is the objective function for UAV inspection optimization, $f'_{\{e_{o,1:j,c}\}_{c=1}^{N_c}}(e_{1,i+T_j}, \cdots, e_{N_c,i+T_j})$ is the joint prior PDF of $e_{1,i+T_j}, \cdots, e_{N_c,i+T_j}$.

In this chapter, the objective function in the above equation is approximated based on the prior samples from the $j$-th inspection as follows

$$G_c(d, e_m, T_I) \approx \frac{1}{N_p} \sum_{k=1}^{N_p} \frac{EC(d, e_m, T_I | \{e_{c,i+T_j}^{(k)}\}_{c=1}^{N_c})}{ET(d, e_m, T_I | \{e_{c,i+T_j}^{(k)}\}_{c=1}^{N_c})}, \tag{5.38}$$

in which $\{e_{c,i+T_j}^{(k)}\}_{c=1}^{N_c}, \forall k = 1, \cdots, N_p$ are prior samples of damage state for the $(j+1)^{th}$ inspection, generated using the method discussed in Sec. 5.4.3 and illustrated in Fig. 5.5.

### 5.5.4 Surrogate Modeling

The multi-layered integral composition of the objective function results in a substantial computational demand in calculating the objective function. To optimize the UAV inspection parameters $d, e_m, T_I$ effectively, Bayesian optimization is employed here to enhance the efficiency of the process without consuming excessive computing power. This synergistic approach allows us to devise optimal strategies for subsequent UAV inspections without being impeded by computational constraints.

For simplicity, the vector symbol $\boldsymbol{\omega} = [d, e_m, T_I]$ will be used to denote the three UAV inspection parameters. Since directly evaluating the objective function, $G_c(\boldsymbol{\omega})$, is

computationally extensive, a Gaussian process regression (GPR)-based surrogate model is constructed as a more computational-efficient alternative to study the non-linear behavior, which can be expressed as follows

$$G_c(\boldsymbol{\omega}) \approx \hat{y} = \hat{g}(\boldsymbol{\omega}), \tag{5.39}$$

where $\hat{y}$ is the predicted objective function outcome and $\hat{g}(\boldsymbol{\omega})$ is the GPR model that represents the objective function which is given by

$$\hat{g}(\boldsymbol{\omega}) \sim N(\mu(\boldsymbol{\omega}),\ \sigma(\boldsymbol{\omega})^2), \tag{5.40}$$

in which $N(\cdot, \cdot)$ is Gaussian distribution, $\mu(\boldsymbol{\omega})$ and $\sigma(\boldsymbol{\omega})$ are respectively the mean and standard deviation of the prediction of $\hat{y}$.

To find the global minimum of $G_c(\boldsymbol{\omega})$, the improvement is defined by $I = \max(y^* - y, 0)$, where $y^*$ is the current best solution (the current minimum CPUT) obtained by all the sampled training space. The expectation value or expected improvement (EI) is then computed by (Jones, Schonlau and Welch (1998); Hu and Du (2015))

$$EI(\boldsymbol{\omega}) = (y^* - \mu(\boldsymbol{\omega}))\, \Phi\left(\frac{y^* - \mu(\boldsymbol{\omega})}{\sigma(\boldsymbol{\omega})}\right) + \sigma(\boldsymbol{\omega})\, \phi\left(\frac{y^* - \mu(\boldsymbol{\omega})}{\sigma(\boldsymbol{\omega})}\right), \tag{5.41}$$

where $y^*$ is defined as

$$y^* = \min_{i=1,2,\ldots,k} \{\hat{g}(\boldsymbol{\omega}^{(i)})\}, \tag{5.42}$$

in which $k$ is the number of current training data points.

The new training point (also the possible minimum cost) is identified by maximizing the EI function as

$$\boldsymbol{\omega}^{(k+1)} = \arg\max_{\boldsymbol{\omega} \in \boldsymbol{\Omega}} \mathrm{EI}(\boldsymbol{\omega}). \tag{5.43}$$

The procedure outlined above performs a comprehensive search across the entire

input space, which sequentially identifies the UAV inspection parameters that minimize the output of the objective function until a pre-defined stopping criterion is met. Algorithm 1 summarizes the major steps of the proposed UAV inspection optimization framework.

Next, we present a practical case study of a miter gate to demonstrate the proposed UAV inspection optimization framework. The miter gate is part of the Greenup lock system located on the Ohio River in the Commonwealth of Kentucky, USA.

## 5.6 Case Study

In this section, we begin by presenting the degradation modeling of the miter gate using physics-based simulation. Following that, we show how the proposed framework can be used to optimize the UAV inspection based on the physics-based simulation.

### 5.6.1 Multi-location degradation model and loading condition

As a crucial component of the digital twin paradigm, a high-fidelity FE model was developed using Abaqus 2021 to enable physics-based simulation, as illustrated in Fig. 5.10. The model has been previously validated with the field data which is able to accurately predict the strain response of the structure under various load conditions (Eick et al. (2018)).

Fig. 5.11 shows the hydrostatic pressures applied to both surfaces of the gate, which present a major component of the loading conditions. The upstream and downstream hydrostatic pressures are denoted respectively as $h_{up}$ and $h_{down}$. We therefore have $\mathbf{h} = [h_{up},\ h_{down}]$. Through the daily watering and dewatering process, the cyclic hydrostatic pressure leads to multiple forms of damage along with the aging of the gates. For the sake of illustration, in this chapter, both $h_{up}$ and $h_{down}$ are assumed to follow a Gaussian distribution with different mean values ($\mu_{up} = 550$ and $\mu_{down} = 150$) and the same standard deviation ($\sigma_{up} = \sigma_{down} = 20$) (Vega et al. (2021b)).

Fatigue cracks are among the most concerning damage mechanisms encountered

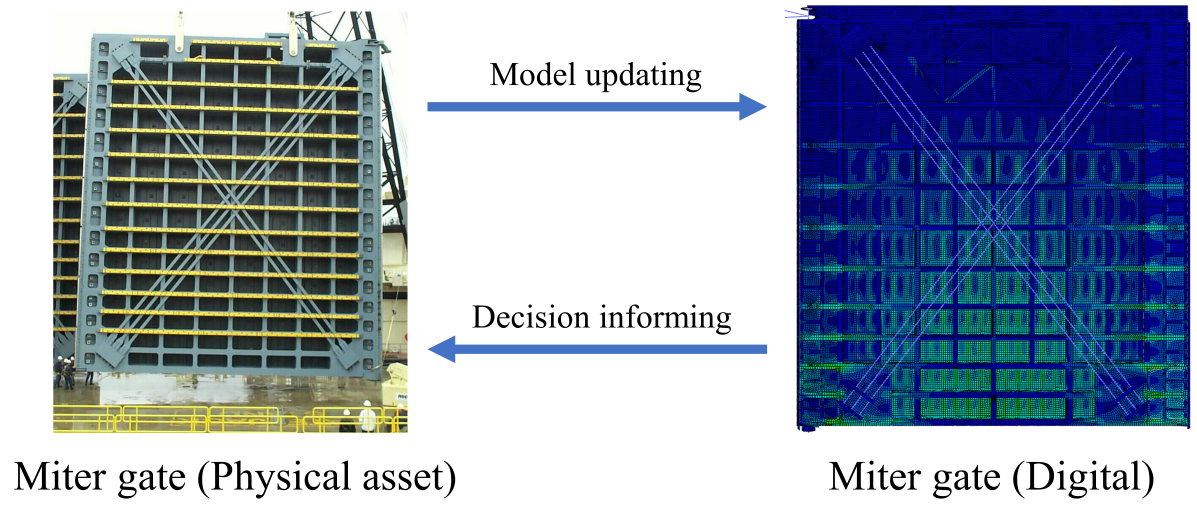



**Figure 5.10.** Finite element analysis model of a miter gate and its connection with the actual gate.

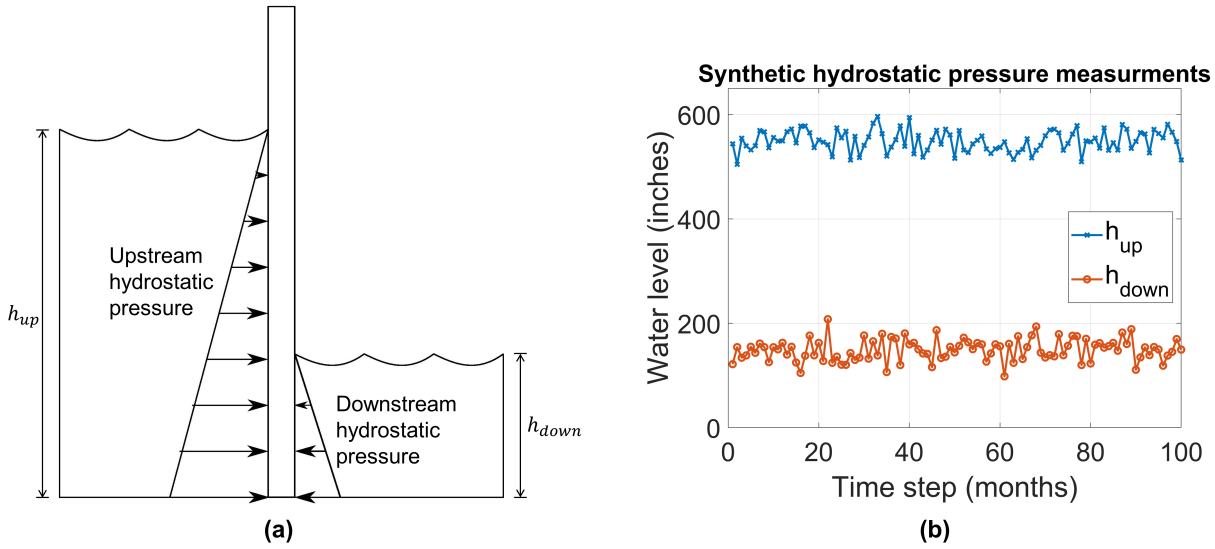



**Figure 5.11.** Hydrostatic pressure: (a) loading condition profile, and (b) synthetic cyclic hydrostatic pressure data in 100 months.

during miter gate inspections, making them a suitable focus for illustration purposes. Explicit crack modeling with finely-discretized meshing has been successfully implemented in the FE model using a surrogate iterative global-local (SIGL) method in Ref. (Fillmore et al. (2022)). Specifically, this approach simulated the local cracked region using XFEM and accelerates the process using static condensation and surrogate modeling. In this

chapter, three cracks were introduced to the FE model at the welding area of the cruciform components of the gate. The SIGL method developed in our previous work (as illustrated in Fig. 5.12) was employed for modeling the cracks.

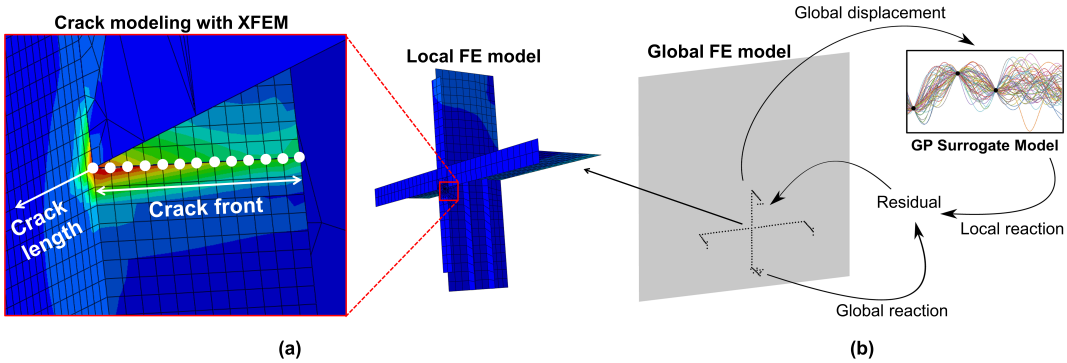



**Figure 5.12.** Implementation of multi-location cracking using IGL: (a) crack simulation using XFEM, and (b) apply to multiple locations with fast and accurate computational cycle using SIGL. (Fillmore et al. (2022))

Fig. 5.13(a) presents the locations of the three cracks, identified as high-risk or low-risk regions based on the global strain distribution during the operational stage of the miter gate. It is worth noting that a small crack detected in a high-risk region may sometimes result in more severe failure than a large crack detected in a low-risk region. In this study, three locations are selected to represent three different scenarios:

1. A large crack in a high-risk region;
2. A small crack in a high-risk region;
3. A small crack in a low-risk region.

The initial length for the three cracks was set to $e_0 = [2.5, 1, 1]$ inches. Fig. 5.13(b) shows the three synthetically generated crack growth curves without maintenance interference. Define the failure threshold of the structure for the crack damage as $e_h = 4\,in.$, the three regions will reach local failure at 9.25, 28.90, and 91.74 months, respectively.

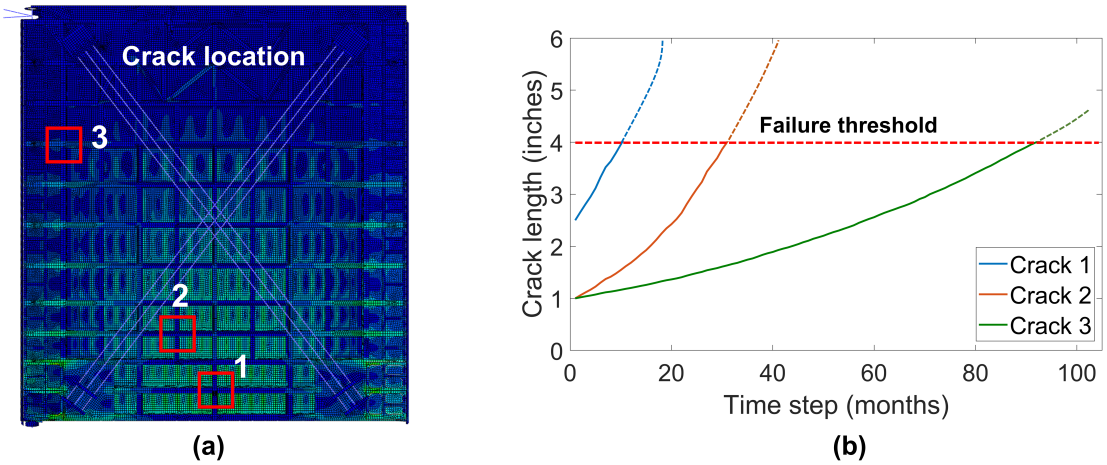



**Figure 5.13.** Synthetic crack growth modeling: (a) the locations of the three cracks, and (b) corresponding crack growth curves without maintenance interference.

### 5.6.2 Inspection and maintenance protocols, and corresponding SHM associated costs

To numerically evaluate the performance of the proposed framework, Table 5.1 presents the design intervals of the inspection parameters and the maintenance protocols. The contribution coefficient $\alpha$ was assigned a value of 0.2 to generally represent the uncertainty level in observation mode. Following that, Fig. 5.14 shows the log-logistic PoD curves obtained using Eq. (5.4). It can be observed that the PoD level increases as the crack size evolves, and a smaller inspection distance leads to a higher PoD level for a given crack size. Additionally, if the crack size is large enough (e.g., for crack sizes of 4 inches or greater), the PoD level will reach 1 for all three scenarios illustrated in Fig. 5.14.

**Table 5.1.** Inspection and maintenance protocols.

| Inspection parameter | Design range |
|---|---|
| Inspection distance ($d$) | 1 ft. $\sim$ 10 ft. |
| Maintenance threshold ($e_m$) | 0 in. $\sim$ 4 in. |
| Inspection interval ($T_I$) | 1 month $\sim$ 40 months |

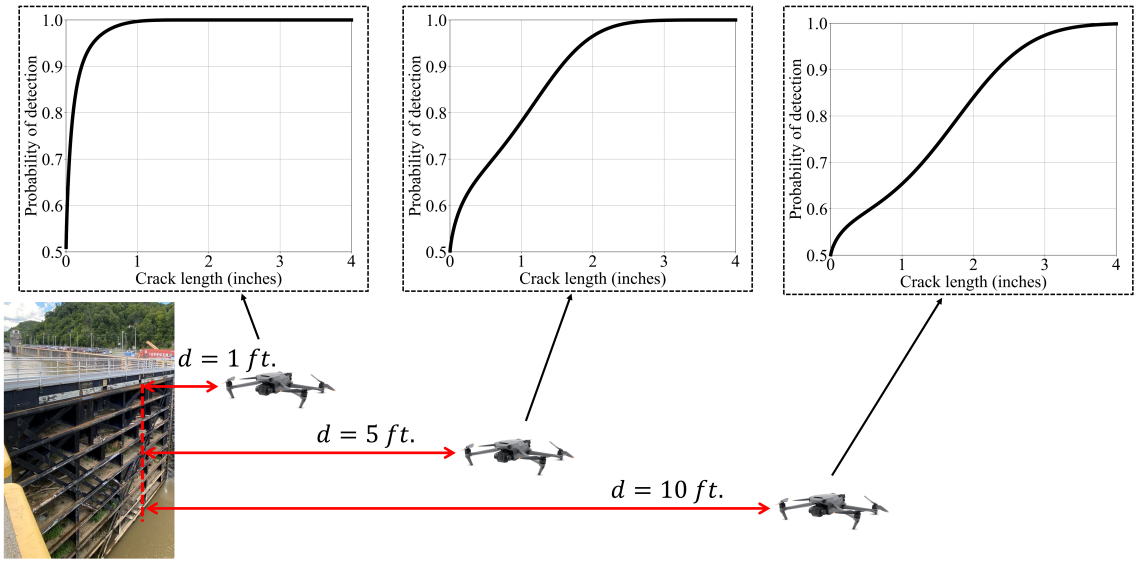



**Figure 5.14.** PoD curves under different inspection distances.

## 5.7 Results and Discussion

The initial priors for the crack-related parameters are set to $f_{e_0}(e)$ and $f_{\theta_0}(\theta)$ for the three damage sites and are updated based on the latest inspection observation $e_{o,j}$. Before we perform UAV inspection optimization using the proposed method, the impact of various inspection parameters on life-cycle costs and failure prognostics is investigated. Fig. 5.15 demonstrates the effect of detectability on the prognostic results. As described in Sec. 5.4.3, the sample $e_{0,d}^{(k)}$ is generated through UP based on the prior distribution (i.e., $f'_{e_{i+T_j}|e_{o,1:j}}(e)$) of the damage state. For Crack 1 and Crack 2 in the "Non-detectable" case, the predicted EOLs are close to the true EOLs. However, the prediction still has significant uncertainty, as seen by the dispersion of the sample curves. In the "Detectable" case, both predicted EOLs of Crack 1 and Crack 2 are more accurate, and the prediction has higher confidence, as indicated by the convergence of the generated samples. Note that obtaining new observations after each inspection does not always lead to better results. As shown in the EOL prediction for Crack 3 in Fig. 5.15, the prediction uncertainty decreases because of the diagnosis process based on the new observations, but the mean

value deviates further from the true value. This deviation occurs because the prognostic accuracy depends on the estimated damage level from the observation, which is a single realization from the observation model $f_{\hat{e}}(e_o, d)$ with randomness. In other words, the obtained observation may not always contribute to the prognostic process, as the image quality taken by UAVs may vary from time to time. However, continuous inspections will eventually rule out the uncertainty and improve the accuracy of the predictions. In the case that a repair is performed after the damage is detected, the EOL of the structure will be extended significantly, as the maintenance action guarantees the reliability of the structure, allowing the system to operate for a longer time.

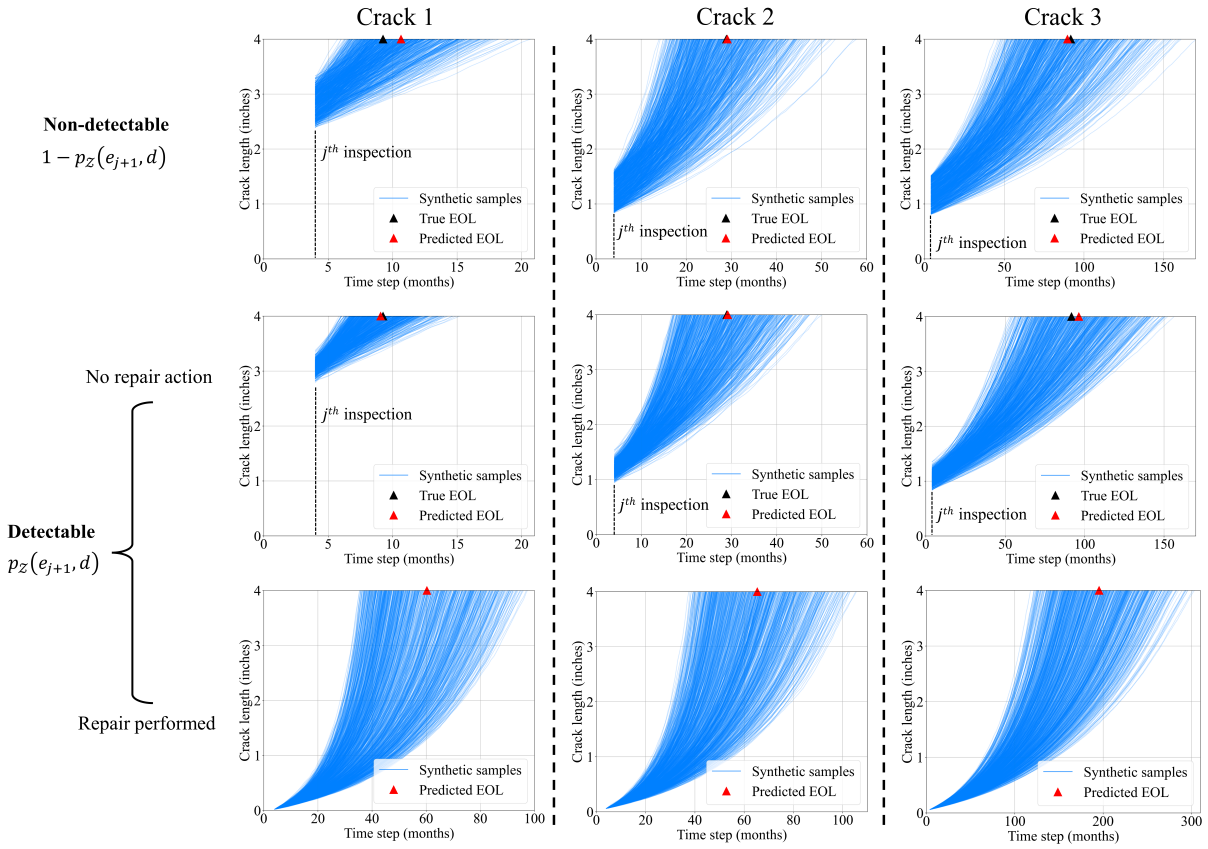



**Figure 5.15.** Effect of detectability on prognostics results.

### 5.7.1 Parametric study of the developed framework

To numerically show the impacts of different inspection and maintenance strategies on CPUT using the proposed framework, we randomly generated ten samples in the 3-D input space constructed by $d$, $e_m$, and $T_I$, using the Latin hyper-cube sampling method. The 10 samples are presented in Table 5.2. We then identified the designs corresponding to the highest and lowest CPUTs for three different cost scenarios. Finally, we analyzed the reasons why these designs resulted in high or low CPUTs.

**Table 5.2.** 10 samples of different inspection and maintenance strategies

| Inspection distance $d$ (ft.) | Maintenance threshold $e_m$ (in.) | Inspection interval $T_I$ (month) |
|---|---|---|
| 5.50 | 2 | 6 |
| 3.25 | 3 | 10 |
| 7.75 | 1 | 2 |
| 4.38 | 1.5 | 6 |
| 8.88 | 3.5 | 7 |
| 2.13 | 2.5 | 13 |
| 6.63 | 0.5 | 13 |
| 3.81 | 0.75 | 14 |
| 8.31 | 2.75 | 21 |
| 1.56 | 3.75 | 29 |

**Case 1: $C_{sys}/C_r = 100$, average $C_I = 55$**

The cost ratio between repairing and system failure is 100. The inspection cost coefficient is 10, which means the cost of performing a UAV inspection varies from 1/10 to 1 of the repair cost of a single region. The extreme results for the two lowest CPUTs and two highest CPUTs are shown in Table 5.3.

As mentioned previously in Section 5.6, region 1 (where Crack 1 exists) will fail at month 9, which is 5 months later after the current inspection. Both strategies correspond to a low CPUT (i.e., designs labeled with an "L" for Case 1 in Table 5.3) set a damage

repair level lower than the length of crack 1. It means that region 1 will get repaired once the crack is detected. By repairing the region with a high failure possibility, both strategies (i.e., $[d, e_m, T_I] = [4.38, 1.5, 6]$ and $[d, e_m, T_I] = [5.50, 2, 6]$) reduce the potential of system failure, thus minimizing the CPUT.

The extremely high CPUTs from the strategies labeled with an "H" for Case 1 in Table 5.3 (i.e., $[d, e_m, T_I] = [1.56, 3.75, 29]$ and $[d, e_m, T_I] = [8.88, 3.5, 7]$) are caused by setting a relatively large damage repair level, which means most of the regions will not get repaired even if a crack is detected. By leaving the region with a high failure possibility unrepaired, both strategies expose the whole system to a high risk of system-level failure, thus "maximizing" the CPUT.

**Case 2: $C_{sys}/C_r = 2$, average $C_I = 55$**

The cost ratio between repairing and system failure is 2. The inspection cost coefficient is 10, which means the cost of performing a UAV inspection now varies from only 1/500 to 1/50 of the repair cost of a single region. The extreme results for the two lowest CPUTs and two highest CPUTs are shown as Case 2 in Table 5.3.

As shown in Case 2 of Table 5.3, due to the high cost of a repair action, a higher damage repair limit is now saving more money compared to Case 1. More specifically, a repair limit of $e_m = 2$ or $e_m = 1.5$ leads to the lowest CPUTs amongst the studied 10 designs for Case 1, while $e_m = 2.5$ or $e_m = 2.75$ results in the lowest CPUTs amongst the same 10 designs for Case 2.

Unlike Case 1, the two extremely high CPUTs observed in Case 2 now can be attributed to two distinct strategies. One strategy (i.e., $[d, e_m, T_I] = [1.56, 3.75, 29]$) again, sets a large damage repair level and an extremely long inspection interval, leaving regions remaining unrepaired without timely inspection. The other strategy (i.e., $[d, e_m, T_I] = [7.75, 1, 2]$), however, causes an outstanding CPUT by setting a very small damage repair level and a short inspection interval. Some regions get repaired even if the cracks are

not dangerous, and another inspection is carried out shortly after such conservative maintenance. As a result, when the repair cost increases, performing excessive and unnecessary maintenance would significantly elevate the CPUTs.

**Case 3: $C_{sys}/C_r = 2$, average $C_I = 275$**

In this case, the cost ratio between repairing and system failure is 100, while the inspection cost coefficient is 50, making the UAV inspection cost range from 1/2 to 5 times the local repair cost. As presented in Table 5.3, the same four inspection/maintenance parameters result in the two lowest and two highest CPUTs, as in Case 1. However, the increasing inspection cost impacts the order of the two lowest CPUTs. In Case 1, a smaller inspection distance with a more conservative damage repair limit is more economically viable, whereas in Case 3, a larger inspection distance with a riskier damage repair limit now leads to a lower CPUT. To conclude, this case highlights the trade-off between allocating resources for thorough UAV inspections and investing in consistent maintenance efforts.

**Table 5.3.** Parametric study of the impact of costs and design variables on CPUT

| Case | Description | $d$ (ft.) | $e_m$ (in.) | $T_I$ (Months) | CPUT | High /Low |
|---|---|---|---|---|---|---|
| 1 | $C_{sys}/C_r = 100$, average $C_I = 55$ | 4.38 | 1.5 | 6 | 19.05 | L |
| | | 5.50 | 2 | 6 | 26.73 | L |
| | | 1.56 | 3.75 | 29 | 1049.93 | H |
| | | 8.88 | 3.5 | 7 | 584.26 | H |
| 2 | $C_{sys}/C_r = 2$, average $C_I = 55$ | 2.13 | 2.5 | 13 | 427.69 | L |
| | | 8.31 | 2.75 | 21 | 604.88 | L |
| | | 7.75 | 1 | 2 | 5710.72 | H |
| | | 1.56 | 3.75 | 29 | 1250.96 | H |
| 3 | $C_{sys}/C_r = 100$, average $C_I = 275$ | 5.50 | 2 | 6 | 34.90 | L |
| | | 4.38 | 1.5 | 6 | 47.28 | L |
| | | 1.56 | 3.75 | 29 | 1286.88 | H |
| | | 8.88 | 3.5 | 7 | 705.33 | H |

### 5.7.2 Bayesian optimization for optimal inspection and maintenance planning

Using Case 1 from Table 5.3 as an example, it's important to note that the lowest CPUT out of the 10 generated samples is not necessarily the optimal solution. With infinite inspection and maintenance strategies that can be implemented for the system, exhaustively calculating all possible combinations for optimal solutions is extremely time-consuming and impractical. Therefore, Bayesian optimization is employed to sequentially find the optimal inspection and maintenance strategies, namely the best $d$, $e_m$, and $T_I$ for the system. Fig. 5.16 shows the convergence history of the optimization results over iterations.

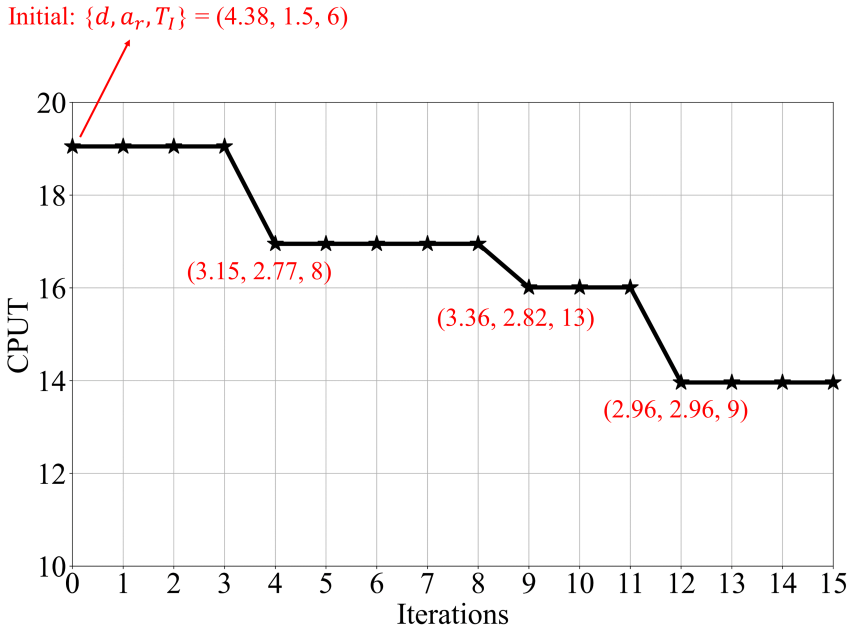



**Figure 5.16.** The obtained minimal CPUT at each iteration using Bayesian optimization.

The results presented in Figure 5.16 demonstrate the effectiveness of the proposed Bayesian optimization framework in identifying optimal inspection and maintenance strategies. Prior to optimization, the currently known "optimal" strategy for Case 1 was identified as using parameters resulting in the lowest CPUT: $\{d, e_m, T_I\} = (4.38, 1.5, 6)$.

However, after only 15 iterations, a new optimal solution was found as $\{d, e_m, T_I\} = (2.96, 2.96, 9)$, which reduced the CPUT from 19.05 to 13.96. The iterative updating process sequentially led to the identification of better strategies, as detailed in Table 5.4. Notably, the first two improvements indicate that a longer operational time after proper maintenance can lead to reduced CPUT. However, the benefits of increasing the operational time should be weighed against the risks of prolonged operation without inspection and maintenance.

**Table 5.4.** The optimal CPUT found at each stage during Bayesian optimization.

| Iteration | $d$ (ft.) | $e_m$ (in.) | $T_I$ (months) | CPUT |
|---|---|---|---|---|
| Initial | 2.13 | 2.5 | 7 | 19.05 |
| 4 | 3.15 | 2.77 | 8 | 16.95 |
| 8 | 3.36 | 2.82 | 13 | 16.01 |
| 11 | 2.96 | 2.96 | 9 | 13.96 |

The subsequent evaluation section will elaborate on the economic performance implications of optimally determined UAV inspection parameters.

### 5.7.3 Performance evaluation

With the determination of the optimal UAV inspection parameters, the proposed optimization framework guides future UAV inspections optimally. This section reuses Case 1 to exhibit the economic enhancements compared to the traditional fixed-period inspection approach. For demonstration purposes, we use a fixed inspection period of $T_I = 10$ months, a constant UAV inspection distance $d = 5$, and a constant maintenance threshold $e_m = 2$. In contrast, the proposed framework dynamically adjusts all inspection parameters after each inspection. The performance comparison of our framework and the fixed-period inspection method is displayed in Fig. 5.17. Fig. 5.17(a) shows that the optimized inspection strategy keeps the structure's failure probability low, offering a clear improvement over the fixed-period method. Fig. 5.17(b) presents the expected cost due to

repair actions, UAV inspection, and potential failure probabilities. Here, the economic benefits of our optimized framework are more significant in terms of expected cost by reducing the potential failure probability. Although the strategy incurs slightly higher costs in some periods due to choosing a smaller inspection distance, this approach ensures observation reliability at the expense of inspection cost.

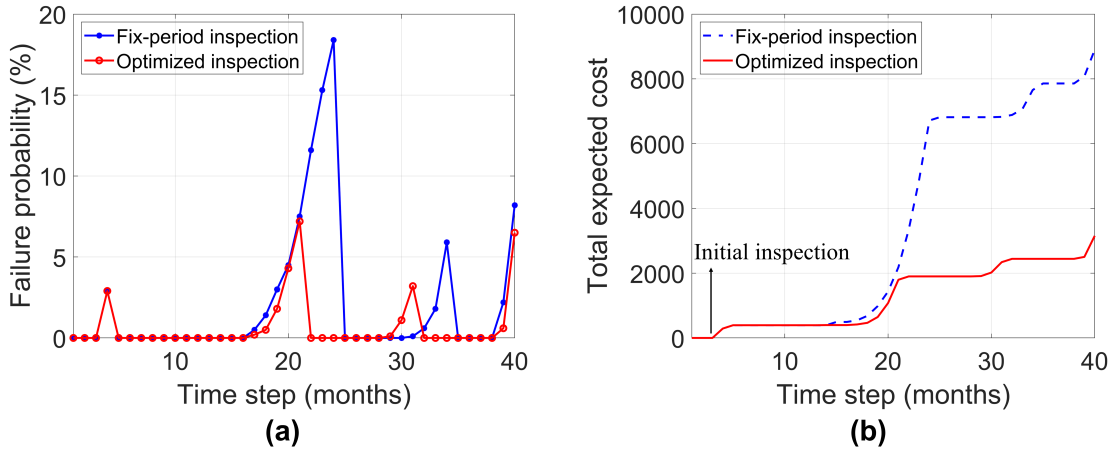



**Figure 5.17.** Comparison between optimized inspection and fix-period inspection in 40 months: (a) the structural failure probability, and (b) the overall inspection and maintenance cost.

Note that the performance shown in Fig. 5.17 is obtained from only a single realization from the Monte Carlo simulation, which can not capture all uncertainties. As such, it may not fully reflect the traditional fixed-period inspection's negative impact. For instance, a longer inspection distance may lead to a low probability of detection, potentially leading to structural failure due to undetected dangerous cracks. Overall, the proposed framework manifests its potential to exploit model-based diagnostics and prognostics for the efficient optimization of UAV inspection and maintenance protocols for large-scale structures.

## 5.8 Conclusions

The proposed framework offers a promising approach to optimizing UAV inspection and maintenance protocols for large-scale structures under complex deteriorating conditions. By leveraging model-informed diagnostics and prognostics, this framework allows for the identification of optimal inspection and maintenance strategies that can significantly reduce the cost of unscheduled maintenance and system failure. Through numerical simulations, we have demonstrated the effectiveness of this framework in identifying the best combination of inspection distance, damage repair limit, and inspection interval to minimize the cost of unscheduled maintenance, system failure, and inspection. The incorporated Bayesian optimization successfully improves the computational efficiency of finding the optimal solution for UAV missions. The results of this study underscore the importance of using model-based diagnostics and prognostics to inform the design of inspection and maintenance protocols, as well as the value of optimizing these protocols to minimize life-cycle costs and improve system reliability.

Further development of this framework could follow one of two primary ways:

1. Integration with data-driven methods: Data-driven methods enable predicting the system's behavior based on additional data sources, such as sensor data. The integration of physics-based and data-driven methods can improve the accuracy and reliability of the health state prediction and inspection and maintenance optimization.

2. Realization of complex environmental conditions: Real-world structures operate under more complex environmental conditions. Environmental factors such as wind turbulence, light/shadow variations, and even underwater conditions (if, for example, underwater unmanned vehicles are used) can significantly affect the accuracy and effectiveness of the proposed framework. To improve the framework's robustness and applicability to real-world scenarios, it is necessary to incorporate more realistic environmental conditions into the model.

## 5.9 Remarks

This chapter is composed of a first-authored publication: **Wu, Z.**, Zeng, J., Hu, Z., & Todd, M. D. (2023). Optimization of unmanned aerial vehicle inspection strategy for infrastructure based on model-enabled diagnostics and prognostics. Mechanical Systems and Signal Processing, 204, 110841.

**Algorithm 3.** Pseudo-code of Model-Based Inspection Optimization Framework

**procedure** MODEL UPDATING AND FAILURE PROGNOSTICS($e_{o,j+1}$, $e_{o,1:j}$, $\boldsymbol{\theta}_i$, $d$, $e_m$, $T_j$)

▷ *Update $\boldsymbol{\theta}_i$ for given $e_{o,j+1}$ and $e_{o,1:j}$, and analyze impacts of $d$, $e_m$ and $T_j$ on subsequent failure prognostics.*

  **if** a damage is detected from observations $e_{o,1:j+1}$ **then**

    Calculate the posterior $f''_{\boldsymbol{\theta}_{i+T_j}|e_{o,1:(j+1)}}$ and $f''_{e_{i+T_j}|e_{o,1:(j+1)}}$ using Eqs. (5.13) - (5.16)

  **else**

    Calculate $f_{e|\mathcal{Z}=0}(e|\mathcal{Z}=0,d)$ using Eq. (5.18)

  **end if**

  Calculate the PDF of $e_{0,d}$ for failure prognostics using Eq. (5.19)

  **if** repair action is taken: $e_{o,j+1} \leq e_m$ **then**

    Calculate $f_{e_0}(e|e_{o,j+1}, e_m)$ using Eq. (5.22)

  **else**

    Calculate $f_{e_0}(e|e_{o,j+1}, e_m)$ using Eq. (5.21)

  **end if**

  Calculate the PDF of $e_{0,d}$ considering damage detectability using Eq. (5.23)

  Calculate $F_{sys}(t; d,\ e_m,\ \{e_{i+T_j,c}\}_{c=1}^{N_c})$ using Eq. (5.25)

**end procedure**

**procedure** INSPECTION COST FUNCTION EVALUATION($d, e_m, T_I$)

▷ *Evaluate the CPUT for any given $d$, $e_m$, and $T_j$ based on MODEL UPDATING AND FAILURE PROGNOSTICS.*

  Calculate the inspection cost $C_I(d)$ using Eq. (5.30)

  Calculate the repair action cost $C_R(d, c_m)$ using Eq. (5.31)

  Calculate the structural failure cost $C_F(d, c_m, T_I)$ using Eq. (5.33)

  Calculate the total expected cost $EC((d, e_m, T_I)|\{e_{i+T_j,c}\}_{c=1}^{N_c})$ using Eq. (5.34)

  Calculate the expected operational time $ET((d, e_m, T_I)|\{e_{i+T_j,c}\}_{c=1}^{N_c})$ using Eq. (5.36) based on model-based failure prognostics.

  Determine the objective function $G_c(d, e_m, T_I)$ using Eqs. (5.37) - (5.38)

**end procedure**

**procedure** INSPECTION PARAMETER OPTIMIZATION(tolerance, $d, e_m, T_I$)

▷ *Optimize $d$, $e_m$, and $T_j$, based on INSPECTION COST FUNCTION EVALUATION.*

  Train a GPR-based surrogate model $\hat{g}(\boldsymbol{\omega})$ to represent the objective/cost function

  **for** $i \in [0, ..., N]$ **do**

    Generate $EI(\boldsymbol{\omega})$ according to Eq. (5.41)

    Find new training point $\boldsymbol{\omega}_{new}$ according to Eq. (5.43)

    **if** $EI(\boldsymbol{\omega}) <$ tolerance **then**

      exit for loop

    **end if**

  **end for**

  Return the optimized inspection parameters

**end procedure**

# Chapter 6

# Conclusions and Future Research

Inland waterways infrastructures like miter gates have served beyond their anticipated lifespans. The challenges these structures face – primarily cracks and corrosion induced by prolonged water exposure – underscore the critical need for an evolved SHM system. This thesis proposes a comprehensive optimization framework that not only enhances damage diagnostics and prognostics but also paves the way for predictive inspection and maintenance strategies.

As shown in Figure 6.1, the key contributions of this research can be summarized as 1) Enhancing the efficiency of solving forward problems through a surrogate-based global-local model, bypassing the need for computationally intensive finely-discretized physics modeling; 2) Developing a model-based diagnostic and prognostic framework with a dynamic Bayesian network that accounts for the uncertainty in multiple observations; 3) Integrating diverse data sources to estimate structure damage state with various failure modes, given that each data type has varying sensitivities to different damages; 4) Fine-tuning a deep learning model for underwater crack detection with a limited dataset by employing a graphics-based digital twin model and transfer learning; 5) Constructing a model-enabled optimized inspection and maintenance framework that facilitates bidirectional information exchange between UAV inspection planning and structural health assessment.

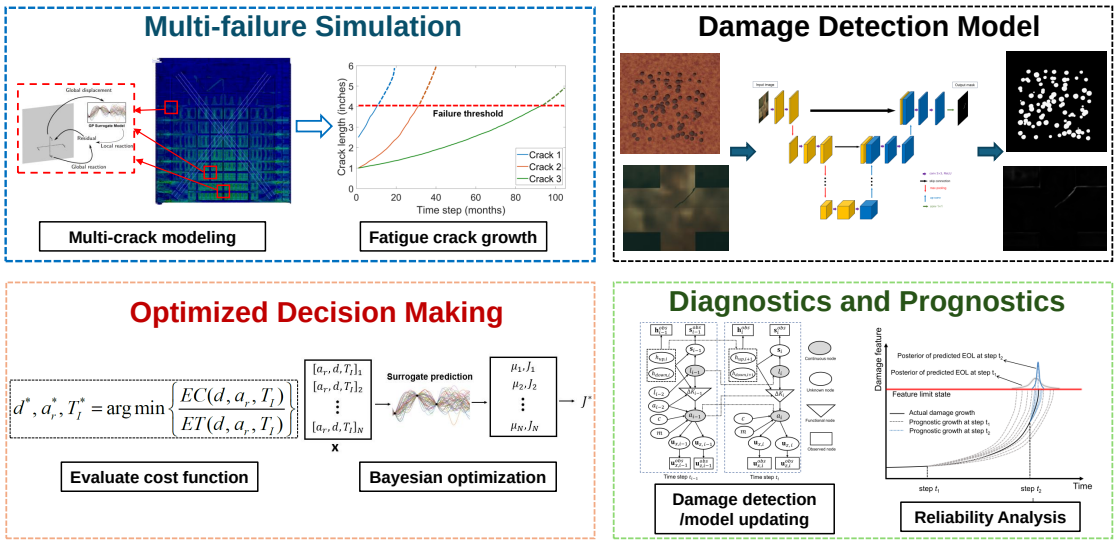



**Figure 6.1.** Key contribution areas of this research.

Chapter 2 addresses the computational challenges of fine-discretized modeling in forward problem-solving. An iterative global-local modeling approach is designed to manage the difference in scale between the coarse global model and the finely-discretized local model. The interaction between the global and local models is accurately calculated by iteratively solving the displacement and reaction force along their shared local boundary. Gaussian Process Regression (GPR) is used to accelerate these repetitive simulation processes. Additionally, to maintain the surrogate model's accuracy, a sequential sampling training strategy is proposed which incrementally gathers training samples based on the current model's performance, enhancing the training space's coverage without over-relying on excessive training samples.

Chapter 3 presents an online diagnostic and prognostic framework that integrates various data sources for structures subject to multiple types of failures. By employing a dynamic Bayesian network, it effectively combines sensor and image data across continuous time steps, facilitating precise damage detection and forecasting across a range of damage magnitudes. This approach also considers the uncertainty inherent in various data types. By updating the digital twin model and tracking the damage state of the structure with

continuous observation, the framework enables a robust damage assessment and the overall reliability of structural health evaluations.

Chapter 4 addresses the challenges faced by existing vision-based damage detection methods in underwater environments. It introduces a graphics-based digital twin model designed to create synthetic underwater images of miter gates. Utilizing transfer learning, these synthetic images allow for the fine-tuning of deep learning models for specific underwater scenarios, bypassing the need for extensive initial training data and computational resources. Additionally, the impact of different environmental factors on the detectability of the deep learning model is studied through a global sensitivity analysis. This analysis provides valuable insights into potential solutions for conducting inspections in real-world underwater settings.

Chapter 5 introduces a physics-informed inspection planning framework based on model-based diagnostics and prognostics. This framework facilitates a bidirectional information exchange between UAV inspection planning and structural reliability. This includes a comprehensive analysis of how key UAV inspection parameters affect the overall cost over the structure's life cycle. It is noted that using Monte Carlo simulations to quantitatively assess every potential inspection and maintenance strategy combination could demand considerable time and computational resources. This chapter applied Bayesian optimization to effectively minimize the time and effort needed to optimize the inspection and maintenance strategy.

The inherent flexibility of this framework, while designed to tackle the complex requirements of miter gate SHM, can be simplified and adapted to address more straightforward, practical problems across different fields. This adaptability underscores the potential of the proposed high-level SHM framework to not only revolutionize the maintenance and inspection of miter gates but also to serve as a foundational model for solving diverse engineering problems.

While this thesis marks a significant advancement in SHM, it also opens several

avenues for future research:

- Advanced machine learning models: The exploration of more sophisticated machine learning models, particularly those capable of handling sparse and imbalanced datasets, could enhance the accuracy of damage detection and classification.
- Real-world validation: Extending the application of the proposed SHM framework to more real-world scenarios would provide valuable insights into its adaptability and scalability.
- Longitudinal studies on SHM performance: Conducting long-term studies on the performance and reliability of SHM systems in varying environmental conditions could help in improving the models and algorithms for better durability.

This research lays the groundwork for an integrated and optimized SHM framework for infrastructure operation, inspection, and maintenance. The future of SHM is bright, with the potential to significantly impact the maintenance, safety, and longevity of critical infrastructure worldwide, contributing to the stage for an era of more intelligent and resilient infrastructure systems.

# Bibliography


Abaqus, 2021. Abaqus verification guide. Dassault Systemes.

Agarwal, V., Neal, K.D., Mahadevan, S., Adams, D., 2017. Concrete Structural Health Monitoring in Nuclear Power Plants. Technical Report. Idaho National Lab.(INL), Idaho Falls, ID (United States).

Allix, O., Gosselet, P., 2020. Non intrusive global/local coupling techniques in solid mechanics: An introduction to different coupling strategies and acceleration techniques, in: Modeling in Engineering Using Innovative Numerical Methods for Solids and Fluids. Springer, pp. 203–220.

Alvarez, L., Weickert, J., Sánchez, J., 2000. Reliable estimation of dense optical flow fields with large displacements. International Journal of Computer Vision 39, 41–56.

Avendano-Valencia, L.D., Chatzi, E.N., Tcherniak, D., 2020. Gaussian process models for mitigation of operational variability in the structural health monitoring of wind turbines. Mechanical Systems and Signal Processing 142, 106686.

Azimi, M., Eslamlou, A.D., Pekcan, G., 2020. Data-driven structural health monitoring and damage detection through deep learning: State-of-the-art review. Sensors 20, 2778.

Barlow, R., Hunter, L., 1960. Optimum preventive maintenance policies. Operations research 8, 90–100.

Barsoum, R., 1976a. On the use of isoparametric finite elements in linear fracture mechanics. International Journal for Numerical Methods and Engineering , 25–37.

Barsoum, R.S., 1976b. On the use of isoparametric finite elements in linear fracture mechanics. International journal for numerical methods in engineering 10, 25–37.

Bishop, C., 2006. Pattern recognition and machine learning. Springer google schola 2, 5–43.

Bjorstad, P., Widlund, O., 1986. Iterative methods for the solution of elliptic problems on

regions partitioned into substructures. SIAM J. Numer. Anal. .

Bolourian, N., Hammad, A., 2020. Lidar-equipped uav path planning considering potential locations of defects for bridge inspection. Automation in Construction 117, 103250.

Brownjohn, J.M., 2007. Structural health monitoring of civil infrastructure. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 365, 589–622.

Carvalho, T.P., Soares, F.A., Vita, R., Francisco, R.d.P., Basto, J.P., Alcalá, S.G., 2019. A systematic literature review of machine learning methods applied to predictive maintenance. Computers & Industrial Engineering 137, 106024.

Cha, Y.J., Choi, W., Büyüköztürk, O., 2017. Deep learning-based crack damage detection using convolutional neural networks. Computer-Aided Civil and Infrastructure Engineering 32, 361–378.

Chen, F.C., Jahanshahi, M.R., 2017. Nb-cnn: Deep learning-based crack detection using convolutional neural network and naïve bayes data fusion. IEEE Transactions on Industrial Electronics 65, 4392–4400.

Chen, X., Chen, X., Zhou, W., Zhang, J., Yao, W., 2020. The heat source layout optimization using deep learning surrogate modeling. Structural and Multidisciplinary Optimization 62, 3127–3148.

Chung, H.M., Maharjan, S., Zhang, Y., Eliassen, F., Strunz, K., 2020. Placement and routing optimization for automated inspection with unmanned aerial vehicles: A study in offshore wind farm. IEEE Transactions on Industrial Informatics 17, 3032–3043.

Daniel, R., Paulus, T., 2019. Lock gates and other closures in hydraulic projects. Elsevier, Amsterdam.

Dong, Y., Frangopol, D.M., 2015. Risk-informed life-cycle optimum inspection and maintenance of ship structures considering corrosion and fatigue. Ocean Engineering 101, 161–171.

Drews, P., Nascimento, E.R., Xavier, A., Campos, M., 2014. Generalized optical flow model for scattering media, in: 2014 22nd International Conference on Pattern Recognition, IEEE. pp. 3999–4004.

Duarte, C., Hamzeh, O., Liszka, T., Tworzydlo, W., 2001a. A generalized finite element method for the simulation of three-dimensional dynamic crack propagation. Computer Methods in Applied Mechanics and Engineering , 2227–2262.

Duarte, C., Hamzeh, O., Liszka, T., Tworzydlo, W., 2001b. A generalized finite element method for the simulation of three-dimensional dynamic crack propagation. Computer methods in applied mechanics and engineering 190, 2227–2262.

Duval, M., Passieux, J.C., Salaun, M., Guinard, S., 2016. Non-intrusive coupling: Recent advances and scalable nonlinear domain decomposition. Arch Computat Methods Eng 72, 173–196.

Eick, B., Fillmore, T., Smith, M., 2019a. Feasibility of discontinuous quoin blocks for usace miter gates. Engineer Research & Development Center .

Eick, B.A., Levine, N.M., Smith, M.D., Spencer Jr, B.F., 2021. Fatigue life updating of embedded miter gate anchorages of navigation locks using full-scale laboratory testing. Structure and Infrastructure Engineering , 1–17.

Eick, B.A., Smith, M.D., Fillmore, T.B., 2019b. Feasibility of discontinuous quoin blocks for USACE miter gates. Technical Report. ENGINEER RESEARCH AND DEVELOPMENT CENTER VICKSBURG MS VICKSBURG.

Eick, B.A., Treece, Z.R., Spencer Jr, B.F., Smith, M.D., Sweeney, S.C., Alexander, Q.G., Foltz, S.D., 2018. Automated damage detection in miter gates of navigation locks. Structural Control and Health Monitoring 25, e2053.

El Said, B., Hallett, S.R., 2018. Multiscale surrogate modelling of the elastic response of thick composite structures with embedded defects and features. Composite Structures 200, 781–798.

Ellenberg, A., Kontsos, A., Bartoli, I., Pradhan, A., 2014. Masonry crack detection application of an unmanned aerial vehicle, in: Computing in Civil and Building Engineering (2014), pp. 1788–1795.

Farrar, C.R., Lieven, N.A., 2007. Damage prognosis: the future of structural health monitoring. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 365, 623–632.

Farrar, C.R., Worden, K., 2007. An introduction to structural health monitoring. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 365, 303–315.

Farrar, C.R., Worden, K., 2012. Structural health monitoring: a machine learning perspective. John Wiley & Sons.

Fillmore, T., Duarte, C., 2018a. A hierarchical non-intrusive algorithm for the generalized finite element method. Advanced Modeling and Simulation in Engineering Sciences ,

1–28.

Fillmore, T., Smith, M., 2021a. Behavior of flexible pintles for miter gates. Journal of Waterway, Port, Coastal, and Ocean Engineering 147.

Fillmore, T.B., Duarte, C.A., 2018b. A hierarchical non-intrusive algorithm for the generalized finite element method. Advanced Modeling and Simulation in Engineering Sciences 5, 1–28.

Fillmore, T.B., Smith, M.D., 2021b. Behavior of flexible pintles for miter gates. Journal of Waterway, Port, Coastal, and Ocean Engineering 147, 04021018.

Fillmore, T.B., Wu, Z., Vega, M.A., Hu, Z., Todd, M.D., 2022. A surrogate model to accelerate non-intrusive global–local simulations of cracked steel structures. Structural and Multidisciplinary Optimization 65, 1–20.

Fodor, I., 2002. A survey of dimension reduction techniques. no. ucrl-id-148494. Lawrence Livermore National Lab., CA (US) .

Foltz, S.D., 2017. Investigation of mechanical breakdowns leading to lock closures. Technical Report. ERDC-Cerl Champaign United States.

Frangopol, D.M., Saydam, D., Kim, S., 2012. Maintenance, management, life-cycle design and performance of structures and infrastructures: a brief review. Structure and infrastructure engineering 8, 1–25.

Freimuth, H., König, M., 2018. Planning and executing construction inspections with unmanned aerial vehicles. Automation in Construction 96, 540–553.

García-Macías, E., Ubertini, F., 2021. Integrated shm systems: Damage detection through unsupervised learning and data fusion, in: Structural health monitoring based on data science techniques. Springer, pp. 247–268.

Gendre, L., Allix, O., Gosselet, P., 2011. A two-scale approximation of the schur complement and its use for non-intrusive coupling. International Journal for Numerical Methods in Engineering , 889–905.

Gendre, L., Allix, O., Gosselet, P., Comte, F., 2009a. Non-intrusive and exact global/local techniques for structural problems with local plasticity. Computational Mechanics , 233–245.

Gendre, L., Allix, O., Gosselet, P., Comte, F., 2009b. Non-intrusive and exact global/local techniques for structural problems with local plasticity. Computational Mechanics 44, 233–245.

Georgiou, G.A., 2007. Pod curves, their derivation, applications and limitations. Insight-Non-Destructive Testing and Condition Monitoring 49, 409–414.

Gibb, S., La, H.M., Le, T., Nguyen, L., Schmid, R., Pham, H., 2018. Nondestructive evaluation sensor fusion with autonomous robotic system for civil infrastructure inspection. Journal of Field Robotics 35, 988–1004.

Glaessgen, E., Stargel, D., 2012. The digital twin paradigm for future nasa and us air force vehicles, in: 53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA, p. 1818.

Gosselet, P., Blanchard, M., Allix, O., Guguin, G., 2018. Non-invasive global-local coupling as a schwarz domain decomposition method: acceleration and generalization. Adv. Model. and Simul. in Eng. Sci. .

Gravouil, A., Moës, N., Belytschko, T., 2002. Non-planar 3d crack growth by the extended finite element and level sets—part ii: Level set update. International journal for numerical methods in engineering 53, 2569–2586.

Guinard, S., Bouclier, R., Toniolli, M., Passieux, J.C., 2018. Multiscale analysis of complex aeronatical structures using robust non-intrusive coupling. Adv. Model. and Simul. in Eng. Sci. .

Gupta, P., Pereira, J., Kim, D.J., Duarte, C., Eason, T., 2012a. Analysis of three-dimensional fracture mechanics problems: A non-intrusive approach using a generalized finite element method. Engineering Fracture Mechanics 90, 41–64.

Gupta, P., Pereira, J., Kim, D.J., Duarte, C., Eason, T., 2012b. Analysis of three-dimensional fracture mechanics problems: A non-intrusive approach using a generalized finite element method. Engineering Fracture Mechanics 90, 41–64.

Henshell, R., Shaw, K., 1975a. Crack-tip elements are unnecessary. International Journal for Numerical Methods in Engineering , 495–507.

Henshell, R., Shaw, K., 1975b. Crack tip finite elements are unnecessary. International journal for numerical methods in engineering 9, 495–507.

Hoskere, V., Eick, B., Spencer Jr, B.F., Smith, M.D., Foltz, S.D., 2020. Deep bayesian neural networks for damage quantification in miter gates of navigation locks. Structural Health Monitoring 19, 1391–1420.

Hu, Z., Du, X., 2015. Mixed efficient global optimization for time-dependent reliability analysis. Journal of Mechanical Design 137, 051401.

Hu, Z., Mahadevan, S., 2016. Global sensitivity analysis-enhanced surrogate (gsas) modeling for reliability analysis. Structural and Multidisciplinary Optimization 53, 501–521.

Hu, Z., Mahadevan, S., 2017. A surrogate modeling approach for reliability analysis of a multidisciplinary system with spatio-temporal output. Structural and Multidisciplinary Optimization 56, 553–569.

Hu, Z., Mahadevan, S., 2019. Probability models for data-driven global sensitivity analysis. Reliability Engineering & System Safety 187, 40–57.

Hu, Z., Mourelatos, Z., 2018. Efficient global surrogate modeling based on multi-layer sampling. SAE International Journal of Materials and Manufacturing 11, 385–400.

Hughes, A.J., Barthorpe, R.J., Dervilis, N., Farrar, C.R., Worden, K., 2021. A probabilistic risk-based decision framework for structural health monitoring. Mechanical Systems and Signal Processing 150, 107339.

Ji, A., Xue, X., Wang, Y., Luo, X., Wang, L., 2021. Image-based road crack risk-informed assessment using a convolutional neural network and an unmanned aerial vehicle. Structural Control and Health Monitoring 28, e2749.

Jiang, C., Vega, M.A., Todd, M.D., Hu, Z., 2022. Model correction and updating of a stochastic degradation model for failure prognostics of miter gates. Reliability Engineering & System Safety 218, 108203.

Jin, R., Chen, W., Sudjianto, A., 2002. On sequential sampling for global metamodeling in engineering design. International design engineering technical conferences and computers and information in engineering conference .

Jones, D.R., Schonlau, M., Welch, W.J., 1998. Efficient global optimization of expensive black-box functions. Journal of Global optimization 13, 455.

Kapteyn, M.G., Pretorius, J.V., Willcox, K.E., 2021. A probabilistic graphical model foundation for enabling predictive digital twins at scale. Nature Computational Science 1, 337–347.

Kim, S., Frangopol, D.M., Soliman, M., 2013. Generalized probabilistic framework for optimum inspection and maintenance planning. Journal of Structural Engineering 139, 435–447.

Kwon, K., Frangopol, D.M., 2011. Bridge fatigue assessment and management using reliability-based crack growth and probability of detection models. Probabilistic Engineering Mechanics 26, 471–480.

Leser, P.E., Hochhalter, J.D., Warner, J.E., Newman, J.A., Leser, W.P., Wawrzynek, P.A., Yuan, F.G., 2017. Probabilistic fatigue damage prognosis using surrogate models trained via three-dimensional finite element analysis. Structural Health Monitoring 16, 291–308.

Li, C., Mahadevan, S., 2016. An efficient modularized sample-based method to estimate the first-order sobol' index. Reliability Engineering & System Safety 153, 110–121.

Li, C., Mahadevan, S., Ling, Y., Choze, S., Wang, L., 2017a. Dynamic bayesian network for aircraft wing health monitoring digital twin. Aiaa Journal 55, 930–941.

Li, H., O'Hara, P., Duarte, C., 2021a. Non-intrusive coupling of a 3-d generalized finite element method and abaqus for the multiscale analysis of localized defects and structural features. Finite Elements in Analysis and Design .

Li, H., O'Hara, P., Duarte, C., 2021b. Non-intrusive coupling of a 3-d generalized finite element method and abaqus for the multiscale analysis of localized defects and structural features. Finite Elements in Analysis and Design 193, 103554.

Li, K., Han, Y., Ge, F., Xu, W., Liu, L., 2020. Tracking a dynamic invading target by uav in oilfield inspection via an improved bat algorithm. Applied Soft Computing 90, 106150.

Li, M., Shen, S., Barzegar, V., Sadoughi, M., Hu, C., Laflamme, S., 2021c. Kriging-based reliability analysis considering predictive uncertainty reduction. Structural and Multidisciplinary Optimization , 1–17.

Li, X., Gao, W., Gu, L., Gong, C., Jing, Z., Su, H., 2017b. A cooperative radial basis function method for variable-fidelity surrogate modeling. Structural and Multidisciplinary Optimization 56, 1077–1092.

Liu, J., Malinowski, P.H., Pawłowski, P., Wu, Z., Todd, M.D., 2023a. Damage assessment with laser ultrasonics in 3d-printed plate, in: Society for Experimental Mechanics Annual Conference and Exposition, Springer. pp. 51–55.

Liu, Y., Gao, J., Jiang, T., Zeng, Z., 2023b. Selective maintenance and inspection optimization for partially observable systems: An interactively sequential decision framework. IISE Transactions 55, 463–479.

Liu, Y., Sun, Q., Fan, X., 2014. A non-intrusive global/local algorithm with non-matching interface: Derivation and numerical validation. Comp. Methods Appl. Mech. Engrg., , 81–103.

Liu, Y., Yao, J., Lu, X., Xie, R., Li, L., 2019. Deepcrack: A deep hierarchical feature learning architecture for crack segmentation. Neurocomputing 338, 139–153.

Luque, J., Straub, D., 2019. Risk-based optimal inspection strategies for structural systems using dynamic bayesian networks. Structural Safety 76, 68–80.

Madjidi, H., Negahdaripour, S., 2006. On robustness and localization accuracy of optical flow computation for underwater color images. Computer Vision and Image Understanding 104, 61–76.

Mayershofer, C., Ge, T., Fottner, J., 2021. Towards fully-synthetic training for industrial applications, in: LISS 2020: Proceedings of the 10th International Conference on Logistics, Informatics and Service Sciences, Springer. pp. 765–782.

Moes, N., Dolbow, J., Belytschko, T., 1999. A finite element method for crack growth without remeshing. International Journal for Numerical Methods in Engineering , 131–150.

Moës, N., Dolbow, J., Belytschko, T., 1999. A finite element method for crack growth without remeshing. International journal for numerical methods in engineering 46, 131–150.

Moës, N., Gravouil, A., Belytschko, T., 2002. Non-planar 3d crack growth by the extended finite element and level sets—part i: Mechanical model. International journal for numerical methods in engineering 53, 2549–2568.

Moore, E., 1920. On the reciprocal of the general algebraic matrix. Bull. Am. Math. Soc. 26 , 394–395.

Moravej, H., Jamali, S., Chan, T., Nguyen, A., 2017. Finite element model updating of civil engineering infrastructures: A literature review, in: Proceedings of the 8th International Conference on Structural Health Monitoring of Intelligent Infrastructure 2017, International Society for Structural Health Monitoring of Intelligent .... pp. 1–12.

Mu, Y., Zhang, X., Xie, W., Zheng, Y., 2020. Automatic detection of near-surface targets for unmanned aerial vehicle (uav) magnetic survey. Remote Sensing 12, 452.

Nash, W., Zheng, L., Birbilis, N., 2022. Deep learning corrosion detection with confidence. npj Materials degradation 6, 26.

Negi, P., Kromanis, R., Dorée, A.G., Wijnberg, K.M., 2024. Structural health monitoring of inland navigation structures and ports: a review on developments and challenges. Structural Health Monitoring 23, 605–645.

Oden, J.T., Vemaganti, K., Moës, N., 1999. Hierarchical modeling of heterogeneous solids. Computer Methods in Applied Mechanics and Engineering 172, 3–25.

Okasha, N.M., Frangopol, D.M., 2010. Novel approach for multicriteria optimization of life-cycle preventive and essential maintenance of deteriorating structures. Journal of Structural Engineering 136, 1009–1022.

Onoufriou, T., Frangopol, D.M., 2002. Reliability-based inspection optimization of complex structures: a brief retrospective. Computers & structures 80, 1133–1144.

Pan, B., 2011. Recent progress in digital image correlation. Experimental mechanics 51, 1223–1235.

Parno, M., O'Connor, D., Smith, M., 2018. High dimensional inference for the structural health monitoring of lock gates. arXiv preprint arXiv:1812.05529 .

Passieux, J., Réthoré, J., Gravouil, A., Baietto, M., 2013. Local/global non-intrusive crack propagation simulation using a multigrid x-fem solver. Computational Mechanics 52, 1381–1393.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., 2011. Scikit-learn: Machine learning in python. the Journal of machine Learning research 12, 2825–2830.

Phung, M.D., Quach, C.H., Dinh, T.H., Ha, Q., 2017. Enhanced discrete particle swarm optimization path planning for uav vision-based surface inspection. Automation in Construction 81, 25–33.

Pirboudaghi, S., Tarinejad, R., Alami, M.T., 2018. Damage detection based on system identification of concrete dams using an extended finite element–wavelet transform coupled procedure. Journal of Vibration and Control 24, 4226–4246.

PricewaterhouseCoopers, 2017. Economic contribution of the us tugboat, towboat, and barge industry .

Sabatino, S., Frangopol, D.M., 2017. Decision making framework for optimal shm planning of ship structures considering availability and utility. Ocean Engineering 135, 194–206.

Sadoughi, M., Hu, C., MacKenzie, C., Eshghi, A., Lee, S., 2018. Sequential exploration-exploitation with dynamic trade-off for efficient reliability analysis of complex engineered systems. Structural and Multidisciplinary Optimization 57, 235–250.

Santner, T.J., Williams, B.J., Notz, W.I., Williams, B.J., 2003. The design and analysis of computer experiments. volume 1. Springer.

Shafiee, M., Sørensen, J.D., 2019. Maintenance optimization and inspection planning of wind energy assets: Models, methods and strategies. Reliability Engineering & System

Safety 192, 105993.

Shi, J., Chopp, D., Lua, J., Sukumar, N., Belytschko, T., 2010. Abaqus implementation of extended finite element method using a level set representation for three-dimensional fatigue crack growth and life predictions. Engineering Fracture Mechanics 77, 2840–2863.

Sobol, I.M., 2001. Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. Mathematics and computers in simulation 55, 271–280.

Sohn, H., Farrar, C.R., 2001. Damage diagnosis using time series analysis of vibration signals. Smart materials and structures 10, 446.

Sohn, H., Farrar, C.R., Hemez, F.M., Shunk, D.D., Stinemates, D.W., Nadler, B.R., Czarnecki, J.J., 2003. A review of structural health monitoring literature: 1996–2001. Los Alamos National Laboratory, USA 1, 16.

Spencer Jr, B.F., Hoskere, V., Narazaki, Y., 2019. Advances in computer vision-based civil infrastructure inspection and monitoring. Engineering 5, 199–222.

Su, L., Wan, H.P., Dong, Y., Frangopol, D.M., Ling, X.Z., 2021. Efficient uncertainty quantification of wharf structures under seismic scenarios using gaussian process surrogate model. Journal of Earthquake Engineering 25, 117–138.

Sudret, B., 2008. Global sensitivity analysis using polynomial chaos expansions. Reliability engineering & system safety 93, 964–979.

Suresh, S., 1998. Fatigue of materials. Cambridge university press.

Tan, G., Yang, Z., 2021. Autonomous bridge detection based on resnet for multiple damage types, in: 2021 IEEE 11th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), IEEE. pp. 555–559.

Tan, Y., Li, S., Liu, H., Chen, P., Zhou, Z., 2021. Automatic inspection data collection of building surface based on bim and uav. Automation in Construction 131, 103881.

Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., Sui, F., 2018. Digital twin-driven product design, manufacturing and service with big data. The International Journal of Advanced Manufacturing Technology 94, 3563–3576.

Tao, F., Xiao, B., Qi, Q., Cheng, J., Ji, P., 2022. Digital twin modeling. Journal of Manufacturing Systems 64, 372–389.

Thelen, A., Zhang, X., Fink, O., Lu, Y., Ghosh, S., Youn, B.D., Todd, M.D., Mahadevan, S., Hu, C., Hu, Z., 2022. A comprehensive review of digital twin—part 1: modeling and

twinning enabling technologies. Structural and Multidisciplinary Optimization 65, 1–55.

Tsai, C.J., Tsai, Y.W., Hsu, S.L., Wu, Y.C., 2017. Synthetic training of deep cnn for 3d hand gesture identification, in: 2017 International Conference on Control, Artificial Intelligence, Robotics & Optimization (ICCAIRO), IEEE. pp. 165–170.

Tuegel, E.J., Ingraffea, A.R., Eason, T.G., Spottswood, S.M., 2011. Reengineering aircraft structural life prediction using a digital twin. International Journal of Aerospace Engineering 2011.

Vega, M., Hu, Z., Yang, Y., Chadha, M., Todd, M., 2021a. Diagnosis, prognosis, and maintenance decision making for civil infrastructure: Bayesian data analytics and machine learning, in: Structural Health Monitoring Based on Data Science Techniques. Springer, Cham, pp. 45–73.

Vega, M.A., Hu, Z., Fillmore, T.B., Smith, M.D., Todd, M.D., 2021b. A novel framework for integration of abstracted inspection data and structural health monitoring for damage prognosis of miter gates. Reliability Engineering & System Safety 211, 107561.

Vega, M.A., Hu, Z., Todd, M.D., 2020. Optimal maintenance decisions for deteriorating quoin blocks in miter gates subject to uncertainty in the condition rating protocol. Reliability Engineering & System Safety 204, 107147.

Vega, M.A., Hu, Z., Yang, Y., Chadha, M., Todd, M.D., 2022. Diagnosis, prognosis, and maintenance decision making for civil infrastructure: Bayesian data analytics and machine learning. Springer.

Vega, M.A., Todd, M.D., 2020. A variational bayesian neural network for structural health monitoring and cost-informed decision-making in miter gates. Structural Health Monitoring , 1475921720904543.

Viana, F., Gogu, C., Goel, T., 2021. Surrogate modeling: tricks that endured the test of time and some recent developments. Structural and Multidisciplinary Optimization 64, 2881–2908.

Wang, S., Rodgers, C., Zhai, G., Matiki, T.N., Welsh, B., Najafi, A., Wang, J., Narazaki, Y., Hoskere, V., Spencer Jr, B.F., 2022. A graphics-based digital twin framework for computer vision-based post-earthquake structural inspection and evaluation using unmanned aerial vehicles. Journal of Infrastructure Intelligence and Resilience 1, 100003.

Williams, C., Rasmussen, C., 2006. Gaussian processes for machine learning. Cambridge, MIT press 2.

Worden, K., Manson, G., 2007. The application of machine learning to structural health

monitoring. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 365, 515–537.

Wu, Z., Fillmore, T.B., Vega, M.A., Hu, Z., Todd, M.D., 2022. Diagnostics and prognostics of multi-mode failure scenarios in miter gates using multiple data sources and a dynamic bayesian network. Structural and Multidisciplinary Optimization 65, 1–20.

Wyart, E., Duflot, M., Coulon, D., Martiny, P., Pardoen, T., Remacle, J.F., Lani, F., 2008. Substructuring fe-xfe approaches applied to three-dimensional crack propagation. Journal of Computational and Applied Mathematics , 626–638.

Xie, M., Bott, S., Sutton, A., Nemeth, A., Tian, Z., 2018. An integrated prognostics approach for pipeline fatigue crack growth prediction utilizing inline inspection data. Journal of Pressure Vessel Technology 140.

Xu, S., Liu, H., Wang, X., Jiang, X., 2014. A robust error-pursuing sequential sampling approach for global metamodeling based on voronoi diagram and cross validation. Journal of Mechanical Design 136, 071009.

Xu, Y., Tian, Y., Li, H., 2022. Unsupervised deep learning method for bridge condition assessment based on intra-and inter-class probabilistic correlations of quasi-static responses. Structural Health Monitoring , 14759217221103016.

Yan, S., Zou, X., Ilkhani, M., Jones, A., 2020. An efficient multiscale surrogate modelling framework for composite materials considering progressive damage based on artificial neural networks. Composites Part B: Engineering 194.

Yan, Y., Zhang, H., Zhang, W., Wang, B., Liao, Q., Liang, Y., 2018. An optimization model for uav inspection path of oil and gas pipeline network, in: International Pipeline Conference, American Society of Mechanical Engineers. p. V003T04A007.

Yang, J., Manning, S., 1996. A simple second order approximation for stochastic crack growth analysis. Engineering fracture mechanics 53, 677–686.

Yang, Y., Madarshahian, R., Todd, M.D., 2020. Bayesian damage identification using strain data from lock gates, in: Dynamics of Civil Structures, Volume 2. Springer, pp. 47–54.

Ye, C., Butler, L., Calka, B., Iangurazov, M., Lu, Q., Gregory, A., Girolami, M., Middleton, C., 2019. A digital twin of bridges for structural health monitoring .

Yeter, B., Garbatov, Y., Soares, C.G., 2020. Risk-based maintenance planning of offshore wind turbine farms. Reliability Engineering & System Safety 202, 107062.

Yiwei, W., Christian, G., Binaud, N., Christian, B., Haftka, R.T., Kim, N.H., 2017. A cost driven predictive maintenance policy for structural airframe maintenance. Chinese Journal of Aeronautics 30, 1242–1257.

Zeng, J., Wu, Z., Todd, M.D., Hu, Z., 2023. Bayes risk-based mission planning of unmanned aerial vehicles for autonomous damage inspection. Mechanical Systems and Signal Processing 187, 109958.

Zhang, D., Watson, R., Dobie, G., MacLeod, C., Pierce, G., 2018. Autonomous ultrasonic inspection using unmanned aerial vehicle, in: 2018 IEEE International Ultrasonics Symposium (IUS), IEEE. pp. 1–4.

Zhang, M., Song, H., Lim, S., Akiyama, M., Frangopol, D.M., 2019. Reliability estimation of corroded rc structures based on spatial variability using experimental evidence, probabilistic analysis and finite element method. Engineering Structures 192, 30–52.

Zhang, Y., Zuo, Z., Xu, X., Wu, J., Zhu, J., Zhang, H., Wang, J., Tian, Y., 2022. Road damage detection using uav images based on multi-level attention mechanism. Automation in Construction 144, 104613.

Zhang, J., Taflanidis , A.A., 2019. Multi-objective optimization for design under uncertainty problems through surrogate modeling in augmented input space. Structural and Multidisciplinary Optimization 59, 351–372.

Zhao, S., Kang, F., Li, J., 2022. Concrete dam damage detection and localisation based on yolov5s-hsc and photogrammetric 3d reconstruction. Automation in Construction 143, 104555.

Zhao, S., Kang, F., Li, J., Ma, C., 2021. Structural health monitoring and inspection of dams based on uav photogrammetry with image 3d reconstruction. Automation in Construction 130, 103832.