Data-driven approaches for improving failure resilience of engineered systems

Mohammadkazem Sadoughi

Iowa State University
Data-driven approaches for improving failure resilience of engineered systems

by

Mohammadkazem Sadoughi

A dissertation submitted to the graduate faculty
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Mechanical Engineering

Program of Study Committee:
Chao Hu, Major Professor
William Meeker
Eliot Winer
Simon Laflamme
Cameron MacKenzie
Shan Hu

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University
Ames, Iowa
2019

Copyright © Mohammadkazem Sadoughi, 2019. All rights reserved.
DEDICATION

I would like to dedicate this thesis to my parents for their love, support, patience, and sacrifice.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>LIST OF FIGURES</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>v</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>viii</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>ix</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>x</td>
</tr>
</tbody>
</table>

CHAPTER 1. GENERAL INTRODUCTION ................................................................. 1

CHAPTER 2. A HIGH-DIMENSIONAL RELIABILITY ANALYSIS METHOD FOR SIMULATION-BASED DESIGN UNDER UNCERTAINTY .................................................. 4

1.1. Abstract ..................................................................................................... 4
1.2. Introduction .............................................................................................. 5
1.3. Review of previous methods ................................................................... 12
   A. Univariate Dimension Reduction ....................................................... 12
   B. Sequential Exploration-Exploitation with Dynamic Trade-off .......... 12
1.4. High-Dimensional Reliability Analysis .................................................. 16
   A. Building global surrogate with AUDR (Step 1) ............................ 17
   B. Refining global surrogate with SEEDT (Step 2) ............................ 21
   A. Discussion ...................................................................................... 23
1.5. Examples .................................................................................................. 25
   A. Example 1: influence of function nonlinearity ............................ 27
   B. Example 2: influence of reliability level with strong variate
      interactions .................................................................................... 28
   C. Example 3: influence of dimensionality with strong variate interactions ... 31
   D. Example 4: reliability analysis of piezoelectric energy harvester .... 32
   E. Discussion ...................................................................................... 35
1.6. Conclusion ............................................................................................. 37
1.7. References .............................................................................................. 38

CHAPTER 3. PHYSICS-BASED CONVOLUTIONAL NEURAL NETWORK FOR
FAULT DIAGNOSIS OF ROLLING ELEMENT BEARINGS ........................................ 42

2.1. Abstract .................................................................................................. 42
2.2. Introduction ............................................................................................ 43
2.3. Background ............................................................................................ 47
   A. Spectral Kurtosis analysis ............................................................... 47
   B. Envelope analysis ............................................................................. 49
   C. Convolutional neural network (CNN) ............................................. 51
2.4. Physics-based convolutional neural network (PCNN) ......................... 52
   A. Physics-based convolution ............................................................. 54
2.5. Case studies ........................................................................................... 57
   A. Case study 1: machinery fault simulator ........................................ 58
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>A schematic comparison between squared exponential kernel and additive kernel in a 2-D space.</td>
<td>16</td>
</tr>
<tr>
<td>2.2</td>
<td>Overall flowchart of the proposed HDRA method.</td>
<td>18</td>
</tr>
<tr>
<td>2.3</td>
<td>A comparison of the sampling strategies in classic UDR (a) and AUDR (b) for the 2-D mathematical example in equation (16).</td>
<td>20</td>
</tr>
<tr>
<td>2.4</td>
<td>Reliability errors at different levels of nonlinearity, ( b ), for Example 1.</td>
<td>28</td>
</tr>
<tr>
<td>2.5</td>
<td>Reliability errors by different methods at different reliability levels.</td>
<td>30</td>
</tr>
<tr>
<td>2.6</td>
<td>Reliability error versus number of input variables (or input dimension) for UDR, SEEDT and HDRA in Example 3.</td>
<td>32</td>
</tr>
<tr>
<td>2.7</td>
<td>Components of a piezoelectric energy harvester (cantilever-type).</td>
<td>32</td>
</tr>
<tr>
<td>2.8</td>
<td>Comparison of reliability levels estimated by HDRA and MCS at different ( P_0 ) levels. The widths of error bars are ± one standard deviation in reliability estimates. For the ease of visualization, the error bars by UDR and SEEDT are slightly shifted along the ( x )-axis to the left and right, respectively.</td>
<td>35</td>
</tr>
<tr>
<td>3.1</td>
<td>frequency/frequency resolution) plane.</td>
<td>49</td>
</tr>
<tr>
<td>3.2</td>
<td>Bearing fault pulses: (a) time-domain signal, and (b) power spectra.</td>
<td>50</td>
</tr>
<tr>
<td>3.3</td>
<td>The architecture of CNN.</td>
<td>51</td>
</tr>
<tr>
<td>3.4</td>
<td>The proposed PCNN architecture.</td>
<td>54</td>
</tr>
<tr>
<td>3.5</td>
<td>A sample of physics-based kernel with characteristic frequency of ( f_0 ).</td>
<td>55</td>
</tr>
<tr>
<td>3.6</td>
<td>Convolutional operation on a sample pre-processed vibration signal of a defective bearing. The first plot from the top shows the sample vibration signal of a defective bearing with a fault characteristic frequency of ( f_0 ). The second, third, and fourth plots from the top show the convolved signals using three physics-based kernels generated with the frequencies less than ( f_0 ), equal to ( f_0 ), and higher than ( f_0 ), respectively.</td>
<td>57</td>
</tr>
</tbody>
</table>
Figure 3.7 Machinery fault simulator: (a) experimental setup, and (b) sensor locations........................................................................................................................................ 59

Figure 3.8 Example of two raw signals and their corresponding pre-processed signals. Both input signals are from channel 6 and the operating condition during the data collection was 10 Hz shaft speed, zero level of misalignment, zero level of rotor unbalance, no defect on Bearing R and (a) inner race defect in Bearing L and (b) outer race defect in Bearing L. ........................................................................................................................................ 62

Figure 3.9 The accuracy of PCNN in fault detection and localization at the nine testing speeds........................................................................................................................................ 65

Figure 3.10 Accuracy of different methods in fault detection and localization in Case study 1 ........................................................................................................................................ 66

Figure 3.11 The hay baler under test in Case study 2 and the locations of the sensors that monitor the health conditions of the two bearings. ........................................ 67

Figure 3.12 Bearing faults in Case study 2: (a) inner race fault, (b) outer race fault, and (c) ball fault........................................................................................................................................ 68

Figure 3.13 A sample pre-processed input to PCNN in Case study 2 ........................................................................................................................................ 68

Figure 3.14 Comparison of fault detection accuracy between two existing (SVM and ANN) and the proposed PCNN approaches under various testing speeds in Case study 2 ........................................................................................................................................ 71

Figure 4.1 A soft elastomeric capacitor (SEC) sensor with key components and reference axes annotated........................................................................................................................................ 82

Figure 4.2 Flowchart of unidirectional strain map reconstruction using a traditional Kriging method ........................................................................................................................................ 88

Figure 4.3 Flowchart of the proposed ISF method ........................................................................................................................................ 89

Figure 4.4 Experimental setup used for validating the proposed method: (a) picture of the test bench with key components annotated; (b) schematic of the test bench showing the locations of the sensors, loading point, and added mass ........................................................................................................................................ 93

Figure 4.5 SEC signal for sensors A, B, C, and D under load case 1; here, only every other data point is shown for clarity. ........................................................................................................................................ 97

Figure 4.6 Strain maps obtained through the FEA and the ISF method using the experimental data with 4, 8 and 12 RSGs ........................................................................................................................................ 98
Figure 4.7 ISF reconstruction error as a function of the number of RSGs used in the algorithm formulation. The error is calculated using both load cases 1 and 2. .............................................................................................................. 100

Figure 4.8 Strain maps obtained through the ISF method using the experimental data with 4, 8, and 12 RSGs...................................................... 101

Figure 4.9 Flowchart of unidirectional strain map reconstruction using a traditional Kriging method........................................................................ 102
LIST OF TABLES

Table 2.1 Procedure of UDR for reliability analysis ........................................................ 13
Table 2.2 Procedure of SEEDT for reliability analysis .................................................... 14
Table 2.3 The pseudo-code of the proposed HDRA method............................................ 22
Table 2.4 Summary of the four examples in this study .................................................... 26
Table 2.5 Statistical properties of random variables in Example 2 .................................. 29
Table 2.6 Reliability estimates and relative errors for different $F_s$ values ....................... 30
Table 2.7 Statistical properties of random variables in Example 4 ............................... 34
Table 2.8 Main features of various reliability analysis methods ........................................ 36
Table 3.1 Summary of sensor characteristics ................................................................. 60
Table 3.2 Fault characteristic frequencies of the simulator bearings............................. 60
Table 3.3 Design parameters and their values in Case study 1 ....................................... 61
Table 3.4 PCNN architecture in Case study 1 ................................................................. 63
Table 3.5 Classification scenarios in Case study 1 ......................................................... 64
Table 3.6 Fault characteristic frequencies of the bearing used in Case study 2 ............... 69
Table 3.7 Design of experiments for Case study 2 ......................................................... 69
Table 4.1 Procedure of ISF using Kriging to construct the strain maps ........................ 90
Table 4.2 Parameters used in constructing the FEA model ............................................. 95
Table 4.3 Locations of RSGs used in the ISF method .................................................... 96
I would like to take this opportunity to express my many thanks to those who helped me with various aspects of conducting this research. I have been very fortunate to have been afforded the best advisor, Dr. Chao Hu. His guidance, patience, and support have been instrumental to this research. Special thanks to Dr. Simon Laflamme, Dr. Cameron MacKenzie, and Dr. Austin Downey who have been greatly helpful in providing advice many times during my graduate school career. I hope that I can be as lively, enthusiastic, and energetic as them. My committee members previously not mentioned, Dr. William Meeker, Dr. Eliot Winer, and Dr. Shan Hu, for challenging me to strive for higher heights. My collaborators, Meng Li, Sheng Shen, Ali Nahvi, Yu Hui Lui, Hao Lu, Yifei Li and Amin Eshghi. It was my pleasure working with you. Others, whose names I may have forgotten, thank you.
ABSTRACT

Since the 1980s, major industries and government agencies worldwide have faced increasing challenges in ensuring the reliability and resilience of engineered systems. The goal of this dissertation is to create novel probabilistic analysis and design methodologies that enable engineered systems to achieve and sustain near-zero breakdown performance. Specifically, this dissertation is focused on developing new methods for simulation-based design and sensor-based diagnostics and prognostics that can be used to design engineered systems for failure resilience. The research contributions are in the areas of engineering design under uncertainty and post-design fault diagnostics which focuses on applications within wind turbines (energy generation), rotating machinery, and large-scale structural systems.
CHAPTER 1. GENERAL INTRODUCTION

The goal of this dissertation is to create novel probabilistic analysis and design methodologies that enable engineered systems to achieve and sustain near-zero breakdown performance. Specifically, this dissertation is focused on developing new methods for simulation-based design and sensor-based diagnostics and prognostics that can be used to design engineered systems for failure resilience. The research contributions are in the areas of engineering design under uncertainty and post-design fault diagnostics which focuses on applications within wind turbines (energy generation), rotating machinery, and large-scale structural systems. The main contributions of this dissertation are summarized as follow.

High-dimensional reliability analysis

Reliability-based design optimization attempts to find the optimum design of an engineered system that minimizes the cost and satisfies a target reliability, while accounting for uncertainty in parameters and design variables. Reliability analysis is an important step and often the most time-consuming step in simulation-based design under uncertainty. Numerous methods have been proposed to analyze engineering product reliability while considering various sources of uncertainty (e.g., loads, material properties, and geometric tolerances). In order to formulate reliability analysis in a mathematical framework, design parameters are usually considered as random variables. A multi-dimensional integration of the joint probability density function (PDF) of these random variables is utilized to determine the reliability:

\[ R = \int_{\Omega} f(\mathbf{x}) d\mathbf{x} \]
where $R$ is the reliability; $\mathbf{x}$ is the vector of random variables; $f(\mathbf{x})$ is the joint PDF of $\mathbf{x}$; $\Omega^S$ denotes the safety region, and is defined based on a performance function (or response) $G(\mathbf{x})$ as $\Omega^S = \{\mathbf{x}: G(\mathbf{x}) < 0\}$. Here, the boundary, $G(\mathbf{x}) = 0$, that separates the safety region from the failure region is called the limit-state function (LSF). In practice, it is often challenging to perform the multi-dimensional numerical integration in equation (1), especially when the performance function is expensive to evaluate and has a large number of dimensions. In the first part of this dissertation, this challenging problem is tackled by proposing a new method, high-dimensional reliability analysis (HDRA), in which a surrogate model is built to approximate a performance function that is high dimensional, computationally expensive, implicit and unknown to the user. A detailed description of the proposed method is provided in Chapter 2.

**Physics-based deep learning for fault diagnosis**

During the past few years, deep learning has been recognized as a useful tool in condition monitoring and fault detection of rotating machinery. Unlike traditional machine learning, deep learning can learn complex patterns relevant to fault diagnosis from massive volumes of sensor data without manually extracting features of fault. Although existing deep learning approaches to fault diagnosis are able to intelligently detect and classify the faults in rotating machinery, they still face one or both of two challenging issues: (i) most of these approaches rely exclusively on data and thus are purely data-driven and do not incorporate physical knowledge into the learning and prediction processes and (ii) these approaches often focus exclusively on the fault diagnosis of a single bearing in a rotating machine, while in reality many rotating machines contain multiple bearings.
In the second part of this dissertation, a new approach for fault diagnosis of rotating machinery is developed. In the developed approach an exclusively data-driven deep learning architecture, convolutional neural network, is modified to incorporate useful information from engineering domain knowledge. Chapter 3 provides a technical description of the developed approach.

**Structural health monitoring**

Traditionally, mesoscale structural systems, including aerospace structures, energy systems and civil infrastructures, are investigated and maintained using reactive or time-based strategies. An alternative is condition-based maintenance, which is known to have strong economic benefits for owners, operators, and society. Structural health monitoring (SHM) and failure prognostics are among the key components of the condition-based maintenance. SHM is defined as the automation of damage detection, localization, and prognostics of structural systems and components. Of particular importance in the development of an SHM system is the consideration of sensor density. The use of dense sensor networks (DSNs) for SHM applications have attracted interest in recent years. Building an accurate strain map over the surface is still a challenging problem. In the third part of this dissertation, a new technique is developed to enhance the capability of DSNs for SHM applications using a novel surrogate modeling technique. A detailed description of the developed technique is provided in Chapter 4.
CHAPTER 2. A HIGH-DIMENSIONAL RELIABILITY ANALYSIS METHOD FOR SIMULATION-BASED DESIGN UNDER UNCERTAINTY

Modified from the paper published in the Journal of Mechanical Design [1]

1.1. Abstract

Reliability analysis involving high-dimensional, computationally expensive, highly nonlinear performance functions is a notoriously challenging problem in simulation-based design under uncertainty. In this study, we tackle this problem by proposing a new method, high-dimensional reliability analysis (HDRA), in which a surrogate model is built to approximate a performance function that is high dimensional, computationally expensive, implicit and unknown to the user. HDRA first employs the adaptive univariate dimension reduction (AUDR) method to construct a global surrogate model by adaptively tracking the important dimensions or regions. Then, the sequential exploration-exploitation with dynamic trade-off (SEEDT) method is utilized to locally refine the surrogate model by identifying additional sample points that are close to the critical region (i.e., the limit-state function) with high prediction uncertainty. The HDRA method has three advantages: (i) alleviating the curse of dimensionality and adaptively detecting important dimensions; (ii) capturing the interactive effects among variables on the performance function; and (iii) flexibility in choosing the locations of sample points. The performance of the proposed method is tested through three mathematical examples and a real world problem, the results of which suggest that the method can achieve an accurate and computationally efficient estimation of reliability even when the performance function exhibits high dimensionality, high nonlinearity, and strong interactions among variables.
1.2. Introduction

Reliability is usually defined as the probability that an engineered system will function properly under stated operating conditions. Reliability analysis, as an essential step in simulation-based design under uncertainty, assesses the probability that a system’s performance (e.g., fatigue, corrosion, fracture) meets its specification while considering various uncertainty sources (e.g., material properties, loads, and geometries). A mathematical framework of reliability analysis models these uncertainties as random variables [1-4]. In equation (1), the boundary, \( G(x) = 0 \), that separates the safety and failure regions is known as the limit-state function (LSF). In engineering practice, the increasing dimension and complexity of the performance function often make the solution of equation (1) computationally intractable. An alternative solution in light of this is to perform the direct Monte Carlo simulation (MCS), where the random variable \( (x) \) space is randomly discretized to a large number of samples, \( x^{(k)}, k = 1:N, \) based on the joint PDF of \( x, f(x) \). Then, the reliability can be estimated based on these random samples as:

\[
R = \hat{R}_x[I_{\Omega_s}(x)] = \lim_{N \to \infty} \frac{1}{N} \sum_{k=1}^{N} I_{\Omega_s}(x^{(k)})
\]

where the system safety indicator \( I_{\Omega_s} \) equals 1 (0) given a safety (failure) state. Although the direct MCS can yield an accurate reliability estimate, achieving satisfactory accuracy often requires a large number of \( G \) function evaluations [3]. A popular alternative to the direct MCS is to use a surrogate model, \( \hat{G}(x) \), which is built with a much smaller number of sample points, to approximate the original performance function, \( G(x) \) [4]. A large number of surrogate modeling methods have been proposed in the literature, among which our review is mainly
focused on two categories of these methods: high-dimensional model representation (HDMR)-based methods [5-20] and kriging-based methods [21-25].

The HDMR-based methods simplify the construction of the surrogate model by decomposing a high-dimensional performance function into several low-dimensional functions (or low-order component functions) [5, 6]. In most practical applications, the response of an engineered system can be largely determined by the low-order interactions among the input variables, which makes HDMR an attractive option in these applications. HDMR was initially applied for efficient multivariate representations in chemical system modeling [7-9]. Numerical experiments showed that HDMR-based methods are promising for simulation-based design of high-dimensional systems [10]. Two of the primary and well-known HDMR-based methods are univariate dimension reduction (UDR) and bivariate dimension reduction (BDR), which get their names on the basis of whether the performance function is decomposed into first- or second-order component functions [11, 12]. Each component function can be approximated with an interpolation technique, and a global surrogate model of the performance function can then be built based on the summation of the approximated component functions. The HDMR-based methods thereafter have been further developed for the application in an expanded range of areas (e.g., design optimization [13], metamodeling [14], and reliability analysis [15]). In design optimization, Li et al [13] integrated HDMR with an expected improvement (EI) sampling strategy that compensates for the underestimation of the response
uncertainty by the kriging predictor. The resulting HDMR-based method achieved satisfactory search performance in deterministic optimization. In metamodeling, Foo and Karniadakis [14] proposed the combined use of HDMR and stochastic collocation, where HDMR is first used to decompose a high-dimensional function into several low-dimensional functions, and multi-element probabilistic collocation is then used to approximate these low-dimensional functions. In reliability analysis, the random input variables could be statistically dependent and in the form of intervals. Xie et al [15] proposed an efficient interval analysis algorithm based on the HDMR expansion to handle dependent interval input variables in reliability analysis. Hu and Youn [16] proposed the adaptive dimension decomposition and reselection method that can automatically identify potentially important low-order component functions and adaptively reselect the truly important ones. Li and Wang [17] coupled HDMR with an intelligent sampling strategy for building global surrogate models on nonlinear problems. To integrate HDMR with the sampling strategy, a projection-based intelligent method was introduced to locate the sample points along the corresponding cuts (lines, planes, or hyperplanes of high orders). This method treats all dimensions equally and is able to track the nonlinear regions in the input space. The component functions in HDMR, resulting from the decomposition of a performance function, can also be approximated by a family of linearly independent basis functions and then metamodeling or reliability analysis can be performed over the approximated component functions [18-20]. The basis functions, such as orthogonal
polynomials and cubic splines, are defined as uni- or multivariate functions in a way that a
proper combination of these basis functions can reflect the true behavior of the performance
function. Hajikolaei and Wang [18] proposed the integration of principal component analysis
with HDMR for finding the coefficients of orthogonal basis functions that provide the best
linear combination of the bases with minimum error. Liu et al [19] proposed the generalized
radial basis function (RBF)-based HDMR method, which employs the virtual regular points,
projected from the random points, to update the predictions by the basis functions. The RBF-
based HDMR method was further developed by using a recently proposed proportional
sampling strategy that allows adding first-order sample points to efficiently construct each
component basis function [20].

Given the negligibly weak high-order interactions, HDMR-based methods with low-order
component functions would provide an efficient and accurate approximation of the
performance function. However, several limitations of these HDMR-based methods do exist:
(i) they usually consider only the low-order interactive effects among input variables; (ii)
increasing the order of component functions may require a dramatic increase in the number of
function evaluations (e.g., building the first-order component functions for a $d$-dimensional
performance function only needs $nd + 1$ function evaluations with $n + 1$ equal to the number
of sample points along each dimension; however building the $m^{th}$-order component functions
with $m \geq 2$ requires $\binom{d}{m} n^m$ function evaluations$^1$; and (iii) most of these methods build a surrogate of the performance function based on the function values determined at a set of structured and predefined grid points, thus lacking the flexibility in choosing the locations of the sample points.

Kriging-based methods first construct an initial surrogate model of the performance function with an initial set of observations (e.g., at a set of sample points generated by Latin Hypercube sampling), and then continuously refine the surrogate model by identifying and adding new sample points until adequate accuracy is achieved [21-30]. Two well-known kriging-based reliability analysis methods are efficient global reliability analysis (EGRA) and maximum confidence enhancement (MCE), both of which consider the prediction uncertainty and LSF proximity in selecting a new sample point [21-23]. A recently developed method, namely Sequential Exploration-Exploitation with Dynamic Trade-off (SEEDT), introduces an exploration-exploitation trade-off coefficient that adaptively weighs exploration and exploitation throughout sampling process to achieve a more optimal balance between these two criteria [24,25].

Two unique features that kriging-based methods possess are: (i) they are flexible in choosing the location of a new sample point, which enables these methods to sample in an optimized sequence; and (ii) they provide a probabilistic surrogate model that quantifies

$^1$ To put this into perspective, it requires a total number of 3,040 function evaluations to build the second-order component functions ($m = 2$) for a 20-dimensional performance function ($d = 20$) with 5 sample points along each dimension ($n = 4$).
both the prediction of a performance function and its uncertainty. With these features, kriging-based methods have been successful in reliability analysis involving system performance functions that are computationally expensive to evaluate, and implicit and unknown to the user. Additionally, kriging-based surrogate modeling has also been successfully applied in real-time processing of multi-sensor signals for structural health monitoring [26]. However, these success stories have been on problems of low (typically < 10) dimension. When it comes to high-dimensional problems, the efficiency of these methods diminishes dramatically. It has been reported in the machine learning area that kriging does not scale to high-dimensional problems [31,32]. This severely limits the application of kriging to reliability analysis of systems that involve the use of complex simulation models with large dimensions (e.g., > 30 parameters) and expensive simulations (e.g., hours of run-time for numerically solving differential equations in one simulation).

In this study, we tackle the above challenges by addressing efficient reliability analysis involving high-dimensional, computationally expensive, highly nonlinear performance functions. Due to the above mentioned limitations, neither HDMR- nor kriging-based methods are capable of providing an accurate and computationally efficient solution for these challenging problems. However, we find that appropriately modifying these two methods and optimally combining their strengths can lead to the development of a feasible solution that is capable of (i) capturing the strong interactions among variables; (ii) alleviating the curse of
dimensionality; and (iii) flexibly choosing the locations of new sample points. Built upon this finding, this study proposes a new method for reliability analysis, named as high-dimensional reliability analysis (HDRA). The proposed HDRA method decomposes the task of surrogate model construction for reliability analysis into two sequential steps: (i) adaptive univariate sampling (with UDR and SEEDT) for building a global surrogate model; and (ii) kriging-based multivariate sampling (with SEEDT) to refine the global surrogate model in highly uncertain regions close to the LSF. At the first step, a newly proposed method, named as adaptive UDR (AUDR), adaptively locates significant univariate sample points by decomposing the important multivariate points identified by SEEDT. A UDR-based global surrogate model is then built based on these univariate sample points. At the second step, the global surrogate model is used as the trend function in the kriging model, and SEEDT further refines the surrogate model by sequentially locating critical multivariate sample points in highly uncertain regions close to the LSF.

The reminder of this chapter is organized as follows. Section 2 gives a brief review on two related methods, UDR and SEEDT. Section 3 introduces the proposed HDRA method, a hybrid of AUDR and SEEDT, for reliability analysis. The effectiveness of the proposed method is demonstrated using four mathematical and engineering examples in Section 4. Section 5 provides several concluding remarks.
1.3. Review of previous methods

A. Univariate Dimension Reduction

A $d$-dimensional performance function can be decomposed in a hierarchical manner as [12]

$$G(x) = G_0 + \sum_{i=1}^{d} G_i(x_i) + \sum_{1 \leq i_1 < i_2 \leq d} G_{i_1i_2}(x_{i_1}, x_{i_2}) + \cdots + \sum_{1 \leq i_1 < \cdots < i_d \leq d} G_{i_1\cdots i_d}(x_{i_1}, \ldots, x_{i_d}) + \cdots + G_{1\cdots d}(x)$$  \hspace{1cm} (3)

where $G_0 = G(\mu)$ is the function value at the mean of the input variables, $\mu$, and acts as the zeroth-order component function, $G_i(x_i)$ is the first-order component function and captures the effect of a single variable $x_i$ on the performance function [12]. The interactive effects among variables are represented by the rest of the terms which are the second- and higher-order component functions. Considering only the zeroth- and first-order component functions leads to the definition of the UDR method, which is described by [11]:

$$G(x) = G_0 + \sum_{i=1}^{d} G_i(x_i) + \epsilon(x)$$  \hspace{1cm} (4)

where $\epsilon(x)$ is the truncation error of the univariate approximation, caused by dropping the interactive effects among variables on the response. Table 2.1 describes explicitly the UDR method in a pseudo-code. The algorithm starts by generating 3 or 5 sample points along each dimension (line 2). After evaluating the performance function at the sample points, each component function $G_i(x_i)$ is approximated based on the cubic spline interpolation (line 5). At the end, the global surrogate model is built based on the equation (4) (line 7).

B. Sequential Exploration-Exploitation with Dynamic Trade-off

The SEEDT method first builds an initial surrogate model with kriging and then sequentially refines the model. For the updating of the surrogate model, expected utility is used
as the acquisition function to locate the next sample point. The sample points are sequentially chosen according to a new exploration-exploitation scheme that adaptively weighs exploring the region with high prediction uncertainty and exploiting the probable location of the LSF throughout the sampling process [24]. The exploration criterion ensures that the sequential sample points would not cluster in one region because the algorithm in SEEDT explores the whole input space for selecting the new sample points. In other words, if the algorithm places a number of sample points in one nonlinear area, the acquisition function will tend to guide the exploration to other nonlinear areas with higher prediction uncertainty. The algorithm starts by generating several initial Latin hypercube points. The initial surrogate model is built with kriging upon the initial sample points (line 1). At each iteration, a new sample point is located by maximizing the expected utility function $EU$ (lines 3-6). The surrogate model is then updated with the augmented data set (lines 7, 8).

Table 2.1 Procedure of UDR for reliability analysis

<table>
<thead>
<tr>
<th>Algorithm 1: Univariate dimension reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 for $i = 1 : d$ do</td>
</tr>
<tr>
<td>2 Initialize sample points with an initial set of univariate points:</td>
</tr>
<tr>
<td>${x_{i,j}}_{j=1}^3 = {\mu_i, \mu_i \pm 3\sigma_i}$ for $n = 2$, or</td>
</tr>
<tr>
<td>${x_{i,j}}_{j=1}^5 = {\mu_i, \mu_i \pm 1.5\sigma_i, \mu_i \pm 3\sigma_i}$ for $n = 4$</td>
</tr>
<tr>
<td>3 Observe the performance function at $x_{i,j}$:</td>
</tr>
<tr>
<td>$y_{i,j} = G(x = (x_{i,j} - \mu_i) \cdot e_i + \mu)$</td>
</tr>
<tr>
<td>4 Build a sample data set for surrogate modeling:</td>
</tr>
<tr>
<td>$D_i = {(x_{i,j}, y_{i,j})}_{j=1:n+1}$</td>
</tr>
<tr>
<td>5 Approximate the $i^{th}$ first-order component function with cubic spline interpolation:</td>
</tr>
<tr>
<td>$\hat{G}<em>i(x_i) = G</em>{\text{spline}}(x_i</td>
</tr>
<tr>
<td>6 end for</td>
</tr>
<tr>
<td>7 Build the global surrogate model:</td>
</tr>
<tr>
<td>$\hat{G}(x) = G_0 + \sum_{i=1}^d \hat{G}_i(x_i)$</td>
</tr>
<tr>
<td>8 Perform MCS on the approximated performance function for reliability estimation</td>
</tr>
</tbody>
</table>
Table 2.2 Procedure of SEEDT for reliability analysis

**Algorithm 2: Sequential exploration-exploitation with dynamic trade-off**

1. Construct an initial surrogate model of the performance function \( G(x) \) with initial data set \( \mathcal{D}^0 \)
2. for \( t = 1 : T \) do
   3. Compute the exploration-exploitation trade-off (EET) coefficient, \( \alpha_t' \)
   4. Construct the acquisition function \( EU \) over \( x \)
   5. Locate the new sample point \( x^t \) that maximizes \( EU \):
      \[
      x^t = \text{argmax } EU(x | \mathcal{D}^{t-1})
      \]
   6. Observe the performance function at \( x^t \):
      \[
      y^t = G(x^t)
      \]
   7. Supplement the data set with the new data:
      \[
      \mathcal{D}^t = \mathcal{D}^{t-1} \cup \{(x^t, y^t)\}
      \]
   8. Refine the surrogate model according to the updated data set \( \mathcal{D}^t \)
3. end for
4. Evaluate the reliability with the surrogate model by performing MCS

The expected utility function is expressed as [24]:

\[
EU = f_x \cdot \frac{\alpha^t \sigma_{\hat{\xi}}}{\sqrt{1 + \alpha^t} \sqrt{2\pi \sigma_{\hat{\xi}}}} \exp\left(-\frac{\mu^2_{\hat{\xi}}}{2\sigma_{\hat{\xi}}^2}\right)
\]  

(5)

where \( \alpha^t \) is the EET coefficient, defined as the probability that a realization of the random variables falls within the regions close to LSF, \( f_x \) is the value of the PDF at \( x \), and \( \mu_{\hat{\xi}} \) and \( \sigma_{\hat{\xi}} \) are respectively the mean and standard deviation of the prediction derived from the kriging model. The kriging model builds a Gaussian process using the residual \( Z \) over the trend function \( m(x) \), which can be expressed as [27]:

\[
\hat{G}(x) = m(x) + Z(x)
\]

(6)

where the Gaussian process \( Z \) has zero mean and correlation matrix \( \Psi \). The general trend of \( G(x) \) is captured by \( m(x) \), while \( Z \) predicts the process noise. Given a sample data set \( \mathcal{D}^t = \{(x^1, y^1), ..., (x^t, y^t)\} \), the correlation matrix \( \Psi \) is expressed as:
\[
\Psi = \begin{pmatrix}
    k(x', x^1) & \cdots & k(x', x^i) \\
    \vdots & \ddots & \vdots \\
    k(x', x^1) & \cdots & k(x', x^i)
\end{pmatrix}
\]  

(7)

where \(k(\cdot, \cdot)\) is known as correlation function or the kernel function which measures the similarity between two real values of random variables and governs the sensitivity of the Gaussian process with respect to changes in the random space. In this study, we use two types of kernel functions, each serving a specific purpose. One kernel function chosen in this study is the squared exponential kernel [35].

\[
k_{se}(x, x') = \exp\left(-\frac{1}{2}(x - x')^T \text{diag}(\theta)^{-2}(x - x')\right)
\]

(8)

The other one is the additive kernel function [36], which decomposes the kernel function into a sum of one-dimensional functions.

\[
k_{add}(x, x') = \sum_{i=1}^{d} \exp\left((x_i - x'_i)^2 \text{diag}(\theta)_i^{-2}\right)
\]

(9)

In both kernel functions, \(\theta\) is the hyper-parameter vector that determines the smoothness of the prediction. The maximum likelihood estimation (MLE) technique is performed to estimate these hyper-parameters. Figure 2.1 schematically illustrates the correlation between two arbitrary points \(x\) and \(x'\), using both the squared exponential kernel \((k_{se})\) and additive kernel \((k_{add})\) functions with \(\theta = (1,1)^T\) in a two-dimensional (2-D) space. It can be observed that \(k_{se}\) is related to the effect of the direct bivariate distance, \(l\), between \(x\) and \(x'\), while \(k_{add}\) corresponds to the sum of the effects of the decomposed univariate distances, \(l_1\) and \(l_2\), between these two points.
Once the hyper-parameters are identified with MLE, the Sherman-Morrison-Woodbury formula is then used to predict the performance function at a new point $\mathbf{x}$. The mean and standard deviation of the prediction are expressed respectively as [27]:

$$\mu_\varphi(\mathbf{x}) = m(\mathbf{x}) + \mathbf{r}(\mathbf{x}) \cdot \Psi^{-1} \cdot (\mathbf{y} - M)$$  \hspace{1cm} (10)$$

$$\sigma_\varphi^2(\mathbf{x}) = 1 - \mathbf{r}(\mathbf{x}) \cdot \Psi^{-1} \cdot \mathbf{r}(\mathbf{x})^T$$ \hspace{1cm} (11)$$

where $\mathbf{y} = (y^1, ..., y^t)^T$ is a vector of $t$ responses, $M$ is the trend function value at sample points $\{x^1, ..., x^t\}$, and $\mathbf{r}(\mathbf{x}) = (\psi(x, x^1), ..., \psi(x, x^t))^T$ is a correlation vector [27].

### 1.4. High-Dimensional Reliability Analysis

The overall flowchart of the proposed HDRA method is shown in Figure 2.2. The method is composed of two sequential steps. At the first step, AUDR adaptively identifies significant univariate sample points by first identifying important multivariate sample points with SEEDT and then decomposing the multivariate points into univariate points along all active dimensions. This step builds a global surrogate model with equation (4) while ensuring satisfactory approximation accuracy of the univariate component function along each
dimension. Upon the completion of adaptive univariate sampling in AUDR, the global surrogate model is used in Step 2 as the trend function in the kriging model, and SEEDT further refines the surrogate model by sequentially locating critical multivariate sample points in regions near the LSF and with high prediction uncertainty. Table 2.1 describes explicitly the procedure of HDRA. Steps 1 and 2 are described in lines 1-14 and 15 of the pseudo-code, respectively. In what follows, the two main steps are explained in further detail.

A. Building global surrogate with AUDR (Step 1)

The response surface built by UDR is often based on the function evaluations at structured grid points. In many real-world applications, the performance function is high-dimensional but depends only on a reduced unknown set of dimensions. Motivated by this observation, we propose the AUDR method that follows an adaptive sampling scheme to detect the important dimensions. In the adaptive sampling schedule, SEEDT adaptively identifies the important multivariate points, which are then decomposed into the corresponding univariate points along all important dimensions through UDR. This can be done by introducing a $d$-dimensional index vector, $I = (I_1, \ldots, I_d)^T$, indicating which dimensions are active. Let $\hat{G}_i^t(x_i)$ and $\hat{G}_i^{t+1}(x_i)$ be the approximation of the first-order component function at two sequential iterations, $t$ and $t+1$, along the $i^{th}$ dimension. The amount of improvement in $\hat{G}_i(x_i)$ by adding a new sample point $x_i^{t+1}$ to the data set, can be considered as the amount of information gained. The information gain takes the following form:

$$
\xi_i^t = E_{x_i} \left( \frac{\hat{G}_i^{t+1}(x_i) - \hat{G}_i^t(x_i)}{\hat{G}_{\max} - \hat{G}_{\min}} \right) = \int \frac{\hat{G}_i^{t+1}(x_i) - \hat{G}_i^t(x_i)}{\hat{G}_{\max} - \hat{G}_{\min}} \cdot f_i(x_i) dx_i
$$

(12)
where $\xi^t_i$ is the expected improvement (or information gain) from adding the new point and measures the relative importance of the $i^{th}$ dimensions, $\hat{G}_{\text{max}}$ and $\hat{G}_{\text{min}}$ are respectively the maximum and minimum predicted values of the performance function at the latest iteration (i.e., iteration $t$), and $E_{f_{x_i}}(\cdot)$ is the expectation operator with respect to the marginal probability distribution of the $i^{th}$ random variable $x_i$. Let $e_0$ denote a pre-defined threshold. If $\xi^t_i < e_0$, then adding more sample points hardly changes our current belief about the $i^{th}$ component function. On the contrary, if $\xi^t_i > e_0$, more sample points are required to be added to enhance our belief about the $i^{th}$ component function. The $i^{th}$ element in the index vector at iteration $t$ indicates the activeness of the $i^{th}$ dimension, and can be evaluated based on the information gain along the $i^{th}$ dimension as
$I_i = \begin{cases} 1, & \text{if } \xi_i > e_a \\ 0, & \text{otherwise} \end{cases}$ (13)

As shown in Table 2.3, Step 1 of the HDRA method starts by locating three or five initial sample points along each dimension (lines 2 and 3). Then a trend function over the performance function, $\hat{G}_{\text{AUDR}}(x)$, is built using equation (4). After that, the kriging model is built based on the $\hat{G}_{\text{AUDR}}(x)$ as the trend function (line 8).

$$\hat{G}(x) = \hat{G}_{\text{AUDR}}(x) + Z(x)$$ (14)

It should be mentioned that the first part of the algorithm (lines 1-14) does not consider the interactive effects among variables, while the second part (line 15) efficiently captures the interactions among variables by identifying multivariate sample points in regions near the LSF and with high uncertainty in prediction. We use the additive kernel function, equation (9), in the correlation matrix $\Psi$ for determining the correlation between sample points.

The new point $x^t$ that maximizes the expected utility function in equation (5) is selected as the important multivariate sample point (line 9). SEEDT suggests that the accurate prediction of the performance function at $x^t$ can likely lead to the maximum improvement in the accuracy of reliability prediction. Since only univariate points can be added to the data set in AUDR, the multivariate point $x^t$ is decomposed into a number of univariate points $x_i^t$ along all the important dimensions defined in the current index vector.

$$x_i^t = (x^t - \mu_i) \cdot e_i + \mu_i \text{ if } I_i = 1$$ (15)

No sample points are added along the inactive dimensions (i.e., dimensions with $I_i = 0$), because the current approximation of the first-order component functions along those dimensions has reached the target accuracy. At the next step, the new point, $x_i^t$, is supplemented to the data set for updating the surrogate model using equation (4) (lines 10-12).
This process is continually executed until all dimensions become inactive (i.e., the first-order component functions along all dimensions are approximated with satisfactory accuracy). Figure 2.3 graphically compares the sampling strategies in classic UDR and AUDR with a simple 2-D performance function:

$$G(x) = \frac{(x_1^2 + 4)(x_2 - 1)}{20} - \cos\left(\frac{7x_1}{2}\right) - 1.5$$

(16)

It can observed from the LSF that the performance function is more highly nonlinear along the $x_1$ dimension. UDR treats both dimensions to be of equal importance, while AUDR identifies the higher degree of nonlinearity along the $x_1$ dimension and adaptively assigns more univariate points to the $x_1$ dimension and fewer points to the $x_2$ dimension. Furthermore, AUDR allocates more points to regions that potentially contribute more to improving the accuracy in the LSF and reliability predictions.

Figure 2.3 A comparison of the sampling strategies in classic UDR (a) and AUDR (b) for the 2-D mathematical example in equation (16).
B. Refining global surrogate with SEEDT (Step 2)

AUDR does not incorporate any multivariate sample points during the sampling process, and thus the resulting global surrogate of the performance function lacks the ability to capture the interactive effects among variables. To address this limitation, after Step 1 is completed, the SEEDT method is utilized again to adaptively refine the surrogate model by sequentially locating multivariate sample points in regions close to the LSF and with high prediction uncertainty. For the refinement of the global surrogate, multivariate sample points are chosen according to a sequential exploration-exploitation scheme in SEEDT that dynamically weighs exploring the regions with high prediction uncertainty and exploiting the ones close to the LSF. The procedure of SEEDT (see Algorithm 2 in Chapter 2, Section 3) is used here as Step 2 of the HDRA method, and the surrogate model built in Step 1, \( \hat{G}_{\text{AUDR}}(\mathbf{x}) \), is employed as the trend function of the kriging model in equation (6). To allow for the consideration of the interactions among variables, this step uses the squared exponential kernel in equation (8) for building the correlation matrix. After building the initial kriging model, a new sample point \( \mathbf{x}^k \) is chosen by maximizing the expected utility function (line 18). This new point is then added to the data set, and the kriging model is updated with the augmented data set (lines 19-21). This procedure is repeated until the target accuracy (see the convergence estimator in line 1) is achieved or the maximum number of iterations is reached (line 17). Interested readers are referred to [24] for more information about the convergence estimator in SEEDT.
Algorithm 3: High-dimensional reliability analysis

**Step 1: Building Global Surrogate with AUDR**

1. Initialize active index vector: \( I = \text{ones}(d, 1) \)
2. Initialize sample points with an initial set of univariate points:
   \[
   \{x_{i,j}^0\}_{j=1:3} = \{\mu_i, \mu_i \pm 3\sigma_i\}, \text{ for } n = 3, \text{ or }
   \{x_{i,j}^0\}_{j=1:5} = \{\mu_i, \mu_i \pm 1.5\sigma_i, \mu_i \pm 3\sigma_i\}, \text{ for } n = 5
   \]
3. Observe the performance function at the initial univariate points:
   \[
   y_{i,j}^0 = G_i(x = (x_{i,j}^0 - \mu_i) \cdot e_l + \mu)
   \]
4. Build an initial sample data set for surrogate modeling:
   \[
   D^0 = \{(x_{i,j}^0, y_{i,j}^0)\}_{i=1:d, j=1:n+1}
   \]
5. \( t = 1 \)
6. while \( t < T \& I \neq 0 \) do
   7. Update the active index vector \( I \) and the surrogate model:
      \[
      [I, \hat{G}_{\text{AUDR}}] = \text{update}(D^{t-1})
      \]
   8. Construct the kriging model of \( G(x) \) based on \( D^{t-1} \):
      \[
      \hat{G}(x) = \text{GP}(x|\hat{G}_{\text{AUDR}}, \Psi, D^{t-1})
      \]
   9. Identify the new multivariate point \( x^t \) that maximizes the acquisition function \( EU \):
      \[
      x^t = \text{argmax } EU(x|D^{t-1})
      \]
   10. Generate univariate sample points according to \( I \):
      \[
      x^t_l = (x^t - \mu).e_l + \mu \quad \text{if } I_l = 1
      \]
   11. Observe the performance function at \( x^t_l \):
      \[
      y^t_l = G(x^t_l = (x^t_l - \mu_l).e_l + \mu)
      \]
   12. Supplement the data set with the new data:
      \[
      D^t = D^{t-1} \cup \{(x^t_l, y^t_l)\}_{1 \leq l \leq d|I_l=1}
      \]
   13. \( t = t+1 \)
   14. end while

**Step 2: Refining Global Surrogate with SEEDT**

15. Build an initial kriging model based on \( D^t \) from Step 1:
    \[
    \hat{G}(x) = \text{GP}(x|\hat{G}_{\text{AUDR}}, \Psi, D^t)
    \]
16. \( k = 1 \)
17. while \( k < K \& CE > CE_{th} \) do
   18. Select the new multivariate point \( x^{t+k} \) that maximizes \( EU \):
      \[
      x^{t+k} = \text{argmax } EU(x|D^{t+k-1})
      \]
   19. Observe the performance function at \( x^{t+k} \):
      \[
      y^{t+k} = G(x^{t+k})
      \]
   20. Supplement the data set with the new data:
      \[
      D^{t+k} = D^{t+k-1} \cup \{(x^{t+k}, y^{t+k})\}
      \]
   21. Update the kriging model with \( D^{t+k} \):
      \[
      \hat{G}(x) = \text{GP}(x|\hat{G}_{\text{AUDR}}, \Psi, D^{t+k})
      \]
Evaluate the convergence estimator (CE):

\[
CE = \int_{\hat{f}(x) \approx \text{est}(x)} f_i(x)\,dx
\]

22 \( k = k+1 \)

23 \textbf{end while}

24 Evaluate the reliability with the kriging model \( \hat{G}(x) \) by performing MCS.

25 \textbf{function} \([l, \hat{G}_{\text{AUDR}}] = \text{update}(\mathcal{D}^t)\)

26 \textbf{for} \( i = 1 : d \) \textbf{do}

27 Approximate the \( i \)th first-order component function with spline interpolation:

\[
\hat{G}_i(x_i) = G_{\text{spline}}(x_i|\mathcal{D}_i^t)
\]

28 Compute \textit{information} gain \( \xi_i^t \) for the \( i \)th dimension with equation (12).

29 Evaluate \( I_i \) with equation (13).

30 \textbf{end for}

31 Build the global surrogate model

\[
\hat{G}_{\text{AUDR}}(x) = G_0 + \sum_{i=1}^d \hat{G}_i(x_i)
\]

32 \textbf{end function}

A. Discussion

As described earlier, HDRA consists of two sequential steps: (i) adaptive univariate sampling (with AUDR) for building a global surrogate that captures the global trend of a performance function and (ii) kriging-based multivariate sampling (with SEEDT) to refine the global surrogate for capturing the variate interactions in the performance function. AUDR in Step 1 does not add any multivariate samples to the sample data set and thus the resulting global surrogate of the performance function lacks the ability to capture the interactive effects among variables. To address this limitation of the global surrogate, SEEDT in Step 2 sequentially identifies multivariate samples in regions close to the LSF and with high prediction uncertainty. The addition of these multivariate samples produces a refined surrogate that is capable of capturing the variate interactions in the performance function.
It is important to note that the initial samples (i.e., the univariate samples added in Step 1) greatly influence the accuracy and efficiency of the proposed HDRA method. In what follows, we will elaborate this influence based on two extreme scenarios:

- **Scenario 1 – Considering too many initial samples:** The addition of more initial samples in Step 1 (AUDR) would result in higher accuracy in the approximation of the first-order component functions in the UDR model of the performance function (see equation (4)). This UDR model, regardless of how many univariate samples are incorporated, cannot capture the interactive effects among variables. If the performance function contains strong variate interactions, the kriging-based multivariate sampling in Step 2 (SEEDT) is always essential, as it overcomes the limitation of the UDR approximation in Step 1. In this regard, considering too many initial univariate samples in Step 1 could exhaust the computing budget, thereby allowing for the addition of only few multivariate samples in Step 2 and producing an inaccurate surrogate model that does not adequately represent the variate interactions.

- **Scenario 2 – Considering too few initial samples:** The smallest number of initial samples in Step 1 is $2d+1$, where $d$ is the number of random variables in the performance function. Assuming only $2d+1$ univariate samples are considered under this scenario, Step 1 produces an initial global surrogate model built with UDR without adaptive enrichment of univariate samples along any dimension. This initial UDR model may fail to capture the global trend of the performance function, especially along dimensions with high degrees of nonlinearity. In such cases, no matter how many multivariate samples are added in Step 2 to refine the global surrogate, the surrogate may always fail to capture both the high dimensionality and strong variate interactions.
Under this scenario, the final HDRA model would likely be a slightly improved version of ordinary kriging with the UDR model as the trend function.

To conclude, adaptive univariate sampling (AUDR) is used in Step 1 to efficiently build a global trend function for dealing with high dimensionality, and kriging-based multivariate sampling (SEEDT) is adopted in Step 2 to efficiently refine the global trend function for capturing the interactive effect among variables. It is important to determine a proper number of initial samples in Step 1 and this sample size determination is automatically performed in AUDR by stopping the adaptive univariate sampling when the information gain in all the dimensions by adding a new sample is lower than a pre-defined threshold (see equations (12) and (13)).

1.5. Examples

Three mathematical examples and one real-world problem are used to evaluate the performance of the proposed HDRA method. The purpose of each example is summarized in Table 2.4. Example 1 compares the reliability errors by four methods (UDR, AUDR, SEEDT and HDRA) for different degrees of response nonlinearity. Example 2 employs a highly nonlinear performance function and investigates the effect of the reliability level on the efficiency of UDR, AUDR, SEEDT and HDRA. Example 3 is also a mathematical example chosen to evaluate the effect of dimensionality on the performance of the three methods. Then, Example 4 estimates the practicality of the proposed HDRA method with a real-world application. In this example, the performance function is defined as the power generation of a piezoelectric energy harvester and can be represented by a function of 6 intrinsic input variables (i.e., 3 geometric design variables and 3 material property variables), embedded in a
space of extrinsic dimensionality \( d = 15 \). That is, we add 9 extrinsic input variables, each of which contains uncertainty but does not have any effect on the performance function.

Table 2.4 Summary of the four examples in this study

<table>
<thead>
<tr>
<th>Example</th>
<th># of dimensions</th>
<th>Purpose</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40</td>
<td>Influence of function nonlinearity</td>
<td>UDR, AUDR, SEEDT and HDRA</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>Influence of reliability level</td>
<td>UDR, AUDR, SEEDT and HDRA</td>
</tr>
<tr>
<td>3</td>
<td>2-14</td>
<td>Influence of function dimensionality</td>
<td>UDR, SEEDT and HDRA</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>Practicality in real-world application</td>
<td>UDR, SEEDT, HDRA and MCS</td>
</tr>
</tbody>
</table>

In these four examples, the reliability estimate by each method is compared with that by the direct MCS to evaluate the accuracy of the method. If one wants to directly estimate the accuracy of the surrogate model produced by one method, the LSF error estimator defined in [24] can be used that may address the potential issue that two methods give very similar reliability estimates but produce very different surrogate models and thus estimates of the LSF. In the examples of this study, the reliability estimation results by the proposed method have consistently shown better accuracy than those by the benchmark methods. As a result, only the reliability estimation accuracy is used to compare the performance of these methods.

Due to the randomness in the initial sample selection and MCS (for reliability analysis), the reliability estimation errors by UDR, AUDR, SEEDT and HDRA contain uncertainty. To capture this uncertainty, we repeatedly run each method with the same parameter setting for 10 times for all the examples. The performance metric of each method is presented by the mean and uncertainty \((\pm \sigma)\) of the metric over the 10 repeated runs.
A. Example 1: influence of function nonlinearity

Example 1 illustrates how the nonlinearity of a performance function affects the accuracy of reliability analysis. In this example, we consider a performance function that consists of 40 random variables, which is defined as [21]:

\[
G(x) = \frac{(x_1^2 + 4)(x_2 - 1)}{20} - \cos\left(\frac{bx_1}{2}\right) + \sum_{i=1}^{40} x_i^2 - 141.5
\]

(17)

where the coefficient \(b\) is used to adjust the nonlinearity of the performance function along the first direction. All 40 input variables are independent and normally distributed (mean \(\mu = 1.5\), and standard deviation \(\sigma = 1\)). The corresponding reliability is defined as \(R = P(G(x) < 0)\), which is kept in the same level for all chosen values of \(b\). MCS is performed to set the benchmark results of reliability analysis for each level of nonlinearity (sample size: 1 million). The relative reliability error, which is the ratio of the absolute difference between the reliability estimate by each method and that by the direct MCS to the reliability estimate by the direct MCS, serves as the criterion for evaluating the performance of each method. The relative reliability errors by UDR, AUDR, SEEDT and HDRA for different nonlinearity levels are graphically compared in Figure 2.4. The numbers of sample points (or function evaluations) required by UDR, SEEDT and HDRA are 161, 95 and 92, respectively. The number of sample points for AUDR varies from 83 to 86. From Fig. 4, HDRA in general is more accurate than UDR, with considerably fewer sample points, and more accurate than AUDR and SEEDT, with approximately the same number of sample points. SEEDT, as a kriging-based reliability analysis method, fails to produce satisfactory accuracy (i.e., < 1% reliability error) in this high-dimensional problem, and this suggests that the limitation of kriging in high-dimensional problems reported in the machine learning area also applies to reliability analysis.
B. Example 2: influence of reliability level with strong variate interactions

The second example considers a two degree-of-freedom primary-secondary system with uncertain damped oscillators in the presence of a white noise [37]. The reliability level of the system can vary depending on the load $F_s$. This problem has been investigated by many researchers due to its highly nonlinearity [38,39]. The mathematical expression of the performance function of this problem is given by:

$$G(x) = 3k \left[ \frac{\pi S_0}{4\xi^3} \left( \frac{\xi_p \xi_s}{\xi_p \xi_s (4\xi^2 + \theta^2) + \gamma \xi^2_s} \right) \left( \xi_p \omega_p^3 + \xi_s \omega_s^3 \right) \right]^{1/2} - F,$$

where $\omega_p = \sqrt{k_p/m_p}$, $\omega_s = \sqrt{k_s/m_s}$, $\omega_a = (\omega_p + \omega_s)/2$, $\xi_a = (\xi_p + \xi_s)/2$, $\gamma = m_s/m_p$ and $\theta = (\omega_p - \omega_s)/\omega_a$, $x=[\omega_p, \omega_s, \omega_a, \xi_a, \gamma, \theta]$. Table 2.5 summarizes the statistical information of the input variables.
Table 2.5 Statistical properties of random variables in Example 2

<table>
<thead>
<tr>
<th>Variable</th>
<th>Distribution</th>
<th>Mean</th>
<th>St. dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k_p$</td>
<td>Lognormal</td>
<td>1</td>
<td>0.2</td>
</tr>
<tr>
<td>$k_s$</td>
<td>Lognormal</td>
<td>0.01</td>
<td>2e-3</td>
</tr>
<tr>
<td>$m_p$</td>
<td>Lognormal</td>
<td>1.5</td>
<td>0.15</td>
</tr>
<tr>
<td>$m_s$</td>
<td>Lognormal</td>
<td>0.01</td>
<td>1e-3</td>
</tr>
<tr>
<td>$\xi_p$</td>
<td>Lognormal</td>
<td>0.05</td>
<td>0.02</td>
</tr>
<tr>
<td>$\xi_s$</td>
<td>Lognormal</td>
<td>0.02</td>
<td>0.01</td>
</tr>
<tr>
<td>$F_s$</td>
<td>Lognormal</td>
<td>13-16</td>
<td>1.5</td>
</tr>
<tr>
<td>$S_0$</td>
<td>Lognormal</td>
<td>100</td>
<td>10</td>
</tr>
</tbody>
</table>

With the reliability estimate by MCS as the benchmark, changing the mean value of $F_s$ from 15.0 N to 27.5 N results in a reliability interval between 98.22% and 99.77%. Figure 2.5 graphically compares the relative reliability errors of UDR, AUDR, SEEDT and HDRA at seven reliability levels within the interval. Table 2.6 summarizes the numbers of function evaluations, reliability estimates and relative errors by the three methods. As can be seen in the figure and table, the proposed HDRA method yields the minimum error among the four methods with the fewest function evaluations. It is important to emphasize that although UDR and AUDR succeed in accurately approximating all first-order component functions by considering only a small number of sample points at each dimension, they fail to produce an accurate prediction of the reliability. This indicates that the performance function in equation (18) may not show high nonlinearity along each dimension, but it is most affected by high-order interactions among the random variables. HDRA captures these interactive effects by adding to the sample data set multivariate sample points in regions close to the LSF and with high prediction uncertainty.
### Table 2.6 Reliability estimates and relative errors for different $F_s$ values

<table>
<thead>
<tr>
<th>Method</th>
<th>No. of $G$ evaluations</th>
<th>Reliability and error (both in %)</th>
<th>$F_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Reliability</td>
<td>13</td>
</tr>
<tr>
<td>MCS</td>
<td>1,000,000</td>
<td>Reliability</td>
<td>98.22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Error</td>
<td>1.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Error</td>
<td>1.82</td>
</tr>
<tr>
<td>UDR</td>
<td>65*</td>
<td>Reliability</td>
<td>96.40</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Error</td>
<td>1.68</td>
</tr>
<tr>
<td>AUDR</td>
<td>19-21</td>
<td>Reliability</td>
<td>96.54</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Error</td>
<td>0.32</td>
</tr>
</tbody>
</table>

*We do not observe any noticeable improvement in accuracy when further increasing the number of function evaluations.*

Figure 2.5 Reliability errors by different methods at different reliability levels.
C. Example 3: influence of dimensionality with strong variate interactions

Example 3 investigates the influence of dimensionality on the performance of reliability estimation. The performance function considered in this example contains $d$ independent normal random variables, all with zero means and standard deviations 0.2. The input dimension $d$ varies from 2 to 14. The performance function is expressed as:

$$G(x) = \prod_{i=0}^{(d-2)/2} \left[ 500 - \left[ 1 + \left( x_{2i+1} + x_{2i+2} + 1 \right)^2 \left( 19 - 14 x_{2i+1} + 3 x_{2i+1}^2 - 14 x_{2i+2} - 6 x_{2i+1} x_{2i+2} + 3 x_{2i+2}^2 \right) \right] \right]$$

The above definition allows us to create performance functions of varying dimensions by simply changing the value of input dimension $d$. Figure 2.6 compares the reliability errors of UDR, SEEDT and HDRA under four settings of input dimension $d$. To make the comparison fair, we keep the numbers of function evaluations the same (i.e., $20d+1$) for the three methods. The number of initial sample points for HDRA is set as $2d+1$, and in most cases, AUDR automatically considers three more sample points along each direction to refine the univariate component functions. As can be seen in Figure 1.6, HDRA yields the lowest error value among the three methods when $d = 6, 10$ and 14. SEEDT produces slightly better accuracy than HDRA when $d = 2$, and somewhat comparable accuracy when $d = 6$. These results suggest that kriging-based methods have a strong capability to handle low-dimensional reliability analysis problems. However, SEEDT fails to accurately predict the reliability for the cases of $d = 10$ and $d = 14$. Furthermore, the relative performance of UDR, as compared to SEEDT, improves with the input dimension, $d$, which shows the unique strength of UDR in handling high-dimensional problems. By taking the advantages of the unique strengths of SEEDT and UDR for low- and high-dimensional problems, HDRA is able to achieve satisfactory performance in all cases.
D. Example 4: reliability analysis of piezoelectric energy harvester

Vibration energy harvesting is widely used to transform commonly wasted energy of vibration into accessible energy, which can then be applied to charge supercapacitors, batteries, or enable self-powering sensors. The typical configuration of an energy harvester is shown in Figure 2.7. It consists of a shim laminated by piezoelectric materials at one end and a tip mass attached at the other end. The tip and shim mass are constructed from tungsten/nickel alloy and Blue steel, respectively.

Figure 2.6 Reliability error versus number of input variables (or input dimension) for UDR, SEEDT and HDRA in Example 3

Figure 2.7 Components of a piezoelectric energy harvester (cantilever-type)
Under the piezoelectric effect, mechanical strain is transferred into electric voltage or current. This study considers 31 modes, which allows for higher longitudinal strain when energy harvester is subject to smaller input forces. Under longitudinal stress/strain, voltage is produced along the thickness direction. The piezoelectric harvester plays similar role as a transformer circuit. Per the Kirchhoff’s Voltage Law, the circuit can be expressed by the coupled differential equations that describe the conversion from mechanical stress/strain to electrical voltage. The conversion process can be simulated by MATLAB Simulink. The harvester output is determined by the geometries of input and material properties. A detailed description of the energy harvester model can be found in the authors’ previous publication [41]. The study in [41] aims to optimize the design of the energy harvester with reliability-based design optimization (RBDO). Three geometric terms, \( \mathbf{x} = [l_b, l_m, h_m] \), and three material properties, \( \mathbf{u} = [u_1, u_2, u_3] \), were considered as the random design variables and the random parameters, respectively. For all these random terms, the means and standard deviations were considered fixed. In this RBDO problem, the size of the energy harvester was minimized while fulfilling the reliability requirement with respect to power generation.

In this study, we evaluate the reliability of the energy harvester at the optimum design acquired by RBDO in [41]. For the energy harvest to function properly, the harvester output power, \( P \), needs to be higher than the minimum required power, \( P_0 \). Then, the reliability of the energy harvester can be expressed as the probability that \( P \) is larger than \( P_0 \) given the random design variables \( \mathbf{x} \) and parameters \( \mathbf{u} \). The performance function is expressed as

\[
G = P_0 - P(\mathbf{x}, \mathbf{u})
\]  

(20)

Besides \( \mathbf{x} \) and \( \mathbf{u} \), we add 9 random input variables \((z_1, ..., z_9)\) that do not affect the power output of the energy harvester, making the performance function of high extrinsic dimension.
(i.e., $d = 15$). This treatment is to demonstrate the capability of HDRA in dealing with high-dimensional reliability analysis problems. The distributional information of the 15 random variables is summarized in Table 7.

<table>
<thead>
<tr>
<th>Random input variable</th>
<th>Description</th>
<th>Dist. type</th>
<th>Mean</th>
<th>Std. dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1(m)$</td>
<td>Length of shim ($l_p$)</td>
<td>Normal</td>
<td>$8.75 \times 10^{-2}$</td>
<td>$1.92 \times 10^{-3}$</td>
</tr>
<tr>
<td>$x_2(m)$</td>
<td>Length of tip mass ($l_m$)</td>
<td>Normal</td>
<td>$1.42 \times 10^{-2}$</td>
<td>$2.40 \times 10^{-4}$</td>
</tr>
<tr>
<td>$x_3(m)$</td>
<td>Height of tip mass ($h_m$)</td>
<td>Normal</td>
<td>$8.00 \times 10^{-3}$</td>
<td>$1.27 \times 10^{-4}$</td>
</tr>
<tr>
<td>$v_1 (m/V)$</td>
<td>Piezoelectric strain coefficient</td>
<td>Normal</td>
<td>$-153.9 \times 10^{-12}$</td>
<td>$7.7 \times 10^{-12}$</td>
</tr>
<tr>
<td>$v_2 (Pa)$</td>
<td>Young’s modulus for PZT_5A</td>
<td>Normal</td>
<td>$66 \times 10^{9}$</td>
<td>$3.3 \times 10^{9}$</td>
</tr>
<tr>
<td>$v_3 (Pa)$</td>
<td>Young’s modulus for the shim</td>
<td>Normal</td>
<td>$20 \times 10^{10}$</td>
<td>$1.00 \times 10^{10}$</td>
</tr>
<tr>
<td>$z_1, ..., z_9$</td>
<td>Ineffective dimension</td>
<td>Normal</td>
<td>0</td>
<td>$10^{-3}$</td>
</tr>
</tbody>
</table>

The reliability level is estimated with the UDR, SEEDT and HDRA method with a total of 121, 120 and 120 sample points respectively. In Step 1 of HDRA, 37 initial sample points are selected to build the global trend function. In HDRA, it is observed that in Step 1 of the algorithm, all 9 ineffective dimensions are considered as inactive dimensions. Therefore, no more sample points are selected along these directions. This feature helps the algorithm focus more on the effective variables (i.e., the first 6 variables) by locating more sample points along them. The direct MCS serves as the benchmark method for the reliability estimation. Since the uncertainty of mean reliability estimates of the direct MCS with 10,000 sample points is much smaller than that of the HDRA method and the process of function evaluation is computationally expensive, it is concluded that 10,000 sample points are enough for the benchmark estimation of reliability. Figure 2.8 graphically summarizes the reliability estimates.
by UDRA, SEEDT, HDRA, and MCS for different levels of $P_0$. It can be observed that the reliability estimates by HDRA are closer to the benchmark reliability levels acquired by the direct MCS in comparison with those by UDR and SEEDT. The reliability analysis results of this real-world engineering application further demonstrate the effectiveness and accuracy of the proposed method in high-dimensional reliability analysis.

![Figure 2.8](image)

**Figure 2.8** Comparison of reliability levels estimated by HDRA and MCS at different $P_0$ levels. The widths of error bars are ± one standard deviation in reliability estimates. For the ease of visualization, the error bars by UDR and SEEDT are slightly shifted along the x-axis to the left and right, respectively.

**E. Discussion**

Table 2.8 lists the main features of four reliability analysis methods, UDR, AUDR, SEEDT, and HDRA. These features are identified by analyzing the intrinsic structures of the algorithms in the methods and interpreting the results from the above examples. UDR is capable of alleviating the curse of dimensionality, and AUDR further enhances this capability by
incorporating an adaptive sampling strategy. Both methods are particularly useful for high-dimensional problems with weak variate interactions. However, neither possesses the capability to handle performance functions with strong variate interactions. Although this limitation can be alleviated by considering higher-order component functions (e.g., in BDR and trivariate DR), the computational efficiency quickly diminishes with the increase of the input dimension. Kriging-based methods such as SEEDT can capture strong variate interactions and are favored in low-dimensional problems (typically \( d < 6 \)) with strong variate interactions. But similar to the limitation of BDR and trivariate DR, these methods lose efficiency in high-dimensional problems. HDRA optimally combines the strengths of UDR and SEEDT, and can achieve satisfactory accuracy and efficiency for problems of both ranges of dimension and with various degrees of variate interactions. The HDRA method provides a better alternative to these popular existing methods by alleviating the fundamental limitation of UDR in tackling strong variate interactions and high computational cost of SEEDT in high-dimensional problems.

<table>
<thead>
<tr>
<th>Features</th>
<th>UDR</th>
<th>AUDR</th>
<th>SEEDT</th>
<th>HDRA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alleviate the curse of dimensionality</td>
<td>✔</td>
<td>✔</td>
<td>⊗</td>
<td>✔</td>
</tr>
<tr>
<td>Adaptive sampling strategy for variable selection</td>
<td>⊗</td>
<td>✔</td>
<td>⊗</td>
<td>✔</td>
</tr>
<tr>
<td>Flexibility in locating the sample points</td>
<td>⊗</td>
<td>⊗</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Capture the interactions among variables</td>
<td>⊗</td>
<td>⊗</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>
1.6. Conclusion

The high-dimensional reliability analysis (HDRA) method has been proposed to solve reliability analysis problems involving high dimensionality, computationally expensive simulations, high nonlinearity, and strong variate interactions. The basic idea of HDRA is to first build a global surrogate model with AUDR and then locally refine the surrogate model using SEEDT. AUDR adaptively identifies important univariate sample points considering both response nonlinearity and criticality (with respect to accurate reliability prediction), while SEEDT captures the interactive effects among variables on the performance function by adding multivariate sample points in highly uncertain regions close to the LSF. Results from four mathematical and real-world examples with up to 40 dimensions suggest that the HDRA method can achieve significantly higher efficiency in reliability analysis than the existing DR- and kriging-based methods, and is especially useful for high-dimensional, computationally expensive problems with strong interactions among random variables. Future research will further develop the proposed method for a wider range of applications, such as sequential experimental design for high-dimensional uncertainty quantification and simulation-based design under high-dimensional uncertainty.
1.7. References


CHAPTER 3. PHYSICS-BASED CONVOLUTIONAL NEURAL NETWORK FOR FAULT DIAGNOSIS OF ROLLING ELEMENT BEARINGS

Modified from the paper published in the IEEE Sensors Journal [1].

2.1. Abstract

During the past few years, deep learning has been recognized as a useful tool in condition monitoring and fault detection of rolling element bearings. Although existing deep learning approaches are able to intelligently detect and classify the faults in bearings, they still face one or both of the following challenges: (i) most of these approaches rely exclusively on data and do not incorporate physical knowledge into the learning and prediction processes; and (ii) the approaches often focus on the fault diagnosis of a single bearing in a rotating machine, while in reality a rotating machine may contain multiple bearings. To address these challenges, this study proposes a novel approach, namely physics-based convolutional neural network (PCNN), for fault diagnosis of rolling element bearings. In PCNN, an exclusively data-driven deep learning approach, called convolutional neural network, is carefully modified to incorporate useful information from physical knowledge about bearings and their fault characteristics. To this end, the proposed approach (1) utilizes spectral kurtosis and envelope analysis to extract sidebands from raw sensor signals and minimize non-transient components of the signals and (2) feeds the information about the fault characteristics into the CNN model. With the capability to process signals from multiple sensors, the proposed PCNN approach is
capable of concurrently monitoring multiple bearings and detecting faults in these bearings.

The performance of PCNN in machinery fault diagnosis is compared with that of traditional machine learning- and deep learning-based approaches reported in the literature.

### 2.2. Introduction

Condition-based maintenance of rotating machinery is of great importance in modern industry. Early faults are often observed in machinery due to the vulnerability of rolling element bearings to various damage mechanisms. An unanticipated machinery failure can lead to significant financial losses due to unplanned maintenance and downtime. The continuous monitoring of bearing health can contribute to avoiding unanticipated failures and improving the reliability and availability of rotating machinery [2, 3]. Over the past few decades, various sensing techniques including vibration [4, 5], acoustic emission (AE) [6], motor current [7], and stray flux [8] have been applied to bearing health monitoring and fault diagnosis. Vibration sensors have been broadly used due to their widespread availability and sensitivity in fault detection of bearings. AE sensors have been shown to be able to detect a crack underneath the surface of either inner or outer ring raceway (or simply inner or outer race) even before the fault grows up to the surface. Recently, Jarek et al. [9] have shown that a consumer microphone can be utilized for fault detection of rotating machinery. Compared to the high-precision vibration sensors and AE sensors, the commercial microphones in general have lower costs and are readily available for deployment in detection of faults that can be acoustically detected. Although each of these sensing techniques has its unique advantages and benefits, there is still no consensus on which single technique can be the best choice for all applications and their operating environments.
Since the raw signals are not adequate to effectively detect a fault, signal processing techniques are often utilized to pre-process the raw signals for facilitating the extraction of the desired diagnostic information. Among the numerous existing signal processing techniques, spectral kurtosis (SK) analysis [10] and envelope analysis [11] have been recognized as two strong tools for improving the accuracy and robustness of machinery fault detection. By quantifying the noise to signal ratio, SK can indicate the frequency band that contains the most diagnostic information; consequently, the signal can be filtered to minimize the noise level [10]. Envelope analysis consists of a bandpass filtering step around a frequency range where the impulsive excitation is magnified followed by a demodulation step to extract the signal envelope [11]. The desired diagnostic information, such as the fault characteristic frequencies and potential modulations, can then be extracted from the spectrum of the signal envelope [12].

Traditionally, after extracting characteristic features of fault from the raw or pre-processed sensor signals, a machine learning technique such as support vector machine (SVM) [13], K-nearest neighbor [10], and artificial neural network (ANN) [14] is often utilized to classify the health state of a bearing (e.g., healthy, inner race fault, ball fault, and outer race fault) [10].

Over the recent years, a new branch of machine learning called deep learning has attracted increasing attention from researchers in the field of bearing fault diagnosis and has also been recognized as a powerful tool for bearing health monitoring. Unlike traditional machine learning techniques, deep learning techniques can directly learn the diagnostic information in the raw or pre-processed sensor signals, without manually extracting features [15].

The unsupervised learning ability of deep learning has recently inspired many researchers to build deep learning models that mine the often large volumes of unlabeled sensor data to achieve higher accuracy and robustness in fault diagnosis of rolling element bearings. As one
of the earliest studies, Tamilselvan et al. [16] proposed a multi-sensor fault diagnosis method that uses deep belief network (DBN) for health-state classification. The DBN model employs a hierarchical structure consisting of multiple stacked restricted Boltzmann machines and works through a layer-by-layer successive learning process. Zhu et al. [17] proposed a deep neural network (DNN)-based technique for hydraulic pump fault diagnosis that leverages frequency characteristics produced by the Fourier transform. Their study shows that frequency spectra may help DNN better discriminate the health conditions of hydraulic pumps. Recently, Lu et al. [18] performed an empirical study on the use of stacked denoising autoencoders with multiple hidden layers to diagnose the faults of rotating machine systems based on vibration signals.

Convolutional neural network (CNN) has also attracted attention from researchers in the field of machinery fault diagnosis [19, 20]. Janssens et al. [21] used a shallow CNN structure with one convolutional layer consisting of wide kernels and a fully connected layer to classify the health state of a bearing. In their study, the discrete Fourier transform (DFT) was employed to process normalized vibration signals collected by two accelerometers placed perpendicular to each other. Zhang et al. [22] proposed a 5-layer CNN architecture with wide kernels in the first convolutional layer and narrow kernels in the subsequent layers to detect bearing faults. By using an extremely large amount of training data, they were able to achieve high detection accuracy even in noisy environments.

Although a number of deep learning-based techniques have recently been implemented and applied to monitor the health of rolling element bearings, little research effort has been devoted to carefully examining how to incorporate physical knowledge into a data-driven deep learning model (e.g., DBN, DNN, and CNN), and more importantly, how to build the knowledge into
the model's architecture, making the model physically meaningful. Without such an examination, the deep learning model is purely data-driven and may not be able to reveal and exploit many pieces of useful information for fault diagnosis that are hidden in the input sensor signals (e.g., vibration and acoustics). For instance, as shown by Maruthi and Hegde [4], the characteristic frequency pertaining to a specific fault of a bearing can be both measured from the peak locations in the power spectrum of a vibration signal, acquired in the presence of the fault, and calculated based on the rotational speed and geometry of the bearing. The calculated frequency serves as part of the physical knowledge about the bearing and its fault characteristics. This knowledge can be useful and even essential to accurate and robust fault diagnosis. Therefore, instead of simply implementing an existing deep learning architecture, more attention should be paid to developing new architectures that explicitly incorporate useful diagnostic information from physical knowledge.

To solve the aforementioned problem, this study proposes a new deep learning approach called physics-based CNN (PCNN) for fault diagnosis of rolling element bearings. The proposed PCNN approach has three unique features: (i) three signal processing techniques, namely SK analysis, envelope analysis, and fast Fourier transform (FFT) analysis, are added to the front of the CNN architecture as new layers; (ii) the physics-based convolutional layer in the CNN architecture explicitly considers the rotational speed and fault characteristic frequencies as the inputs in building the convolutional filters; and (iii) a recently proposed multi-channel CNN is adopted to make the PCNN model compatible when multiple sensors are used to monitor the health of rotating machine systems. This study is conducted based on our earlier study presented in Ref. [23].
2.3. Background

A. Spectral Kurtosis analysis

Spectral kurtosis (SK) analysis is a well-known signal processing technique for characterization of the transients in a signal. In the analysis, the signal is band passed into several subsignals with different frequency ranges and the kurtosis of each band passed subsignal is computed in order to identify the hidden non-stationaries as well as determine in which frequency band these occur. After that, the subsignal that has the highest kurtosis value is selected as the filtered signal.

The main objective is to extract a fault signal $X(t)$ from raw measurement $S(t)$ in presence of some strong additive noise $N(t)$ [24].

$$S(t) = X(t) + N(t)$$

(21)

where it is assumed that $X(t)$ has a transient nature whereas $N(t)$ is stationary. Following the Wold-Cramer representation [25], the zero-mean non-stationary signal $S(t)$ can be decomposed as

$$S(t) = \int_{-\infty}^{+\infty} F(t, f)e^{j2\pi f t} dX(f)$$

(22)

where $F(t, f)$ is the Fourier transform of $h(t)$ with $h(t)$ denoting the time-varying impulse response of series of repetitive short transient forces (e.g., spalls, cracks, and other types of defect) that in turn excite the structural resonances [26]. $F(t, f)$ can be interpreted as the complex envelope of $S(t)$ at frequency $f$. $dX(f)$ is the spectral process associated with $X(t)$, i.e.,

$$X(t) = \int_{-\infty}^{+\infty} e^{j2\pi f t} dX(f)$$

(23)
Then, the SK can be defined as

\[ K_S(f) = \frac{\langle |F(n, f)|^4 \rangle}{\langle |F(n, f)|^2 \rangle^2} - 2 \]  \hspace{1cm} (24)

where \(<.>\) denotes the temporal average. Antoni [26] has shown that the SK of a stationary process is a constant function of frequency, while the SK of a stationary Gaussian process is exactly zero. Therefore, \( K_S(f) \) can be used as an indicator to identify the non-stationaries in a signal.

One of the key questions in SK analysis is how to best decompose an original signal into subsignals. The kurtogram technique is one of the most powerful spectral analysis tools for decomposing a signal into different band passed subsignals in \( n \) sequential steps (see Figure 3.1). At level 1 the original signal is decomposed into two subsignals, each of which covers one-half of the original signal's frequency band. The two subsignals are considered as the subsignals at level 1. Then, each of the two subsignals is decomposed to produce its own subsignals in level 2, and each new subsignal covers one-half of the frequency band of the level 1 subsignal. This process continues until level \( n \), where the original signal is decomposed into \( 2^n \) band passed subsignals [26]. This sequential decomposition of the original signal can provide the so-called frequency/frequency resolution \((f, \Delta f)\) plane. In Figure 3.1, \( S_{k,j} \) is the kurtosis of the jth subsignal at the kth level. After performing the SK analysis using the kurtogram technique, the signal can be band pass filtered based on the frequency/frequency resolution that maximizes its kurtosis.
Envelope analysis, also known as amplitude demodulation, is one of the well-known vibration signal processing techniques for detection of bearing incipient faults. Fundamentally, each time a localized defect in a rolling element bearing makes contact with another surface in the bearing under load, an impulse of vibration is generated. The periodic impulses excite various resonances of the bearing and its surrounding structure. As such the vibration signal is often modulated based on the natural frequency of the system [27]. A sample of simulated vibration signal, coming from a system with a fault characteristic frequency of $f_0$ and a natural frequency of $f_n$, is shown with the blue waveform in Figure 3.2(a). Using the FFT analysis, it can be seen that the spectrum of the raw signal contains little or no diagnostic information about bearing faults (see the blue spectrum in Figure 3.2(b)).

Hilbert transform is a signal demodulation technique that can reveal this diagnostic information by constructing the analytic signal of the sample vibration signal. An analytic
signal is a complex time signal whose imaginary component is the Hilbert transform of the real part. Thus, if \( a(t) \) represents the time signal and \( H(a(t)) \) is its Hilbert transform, then the analytic signal can be defined as \( A(t) = a(t) + jH(a(t)) \), where \( j \) is the unit imaginary number. It can be shown that the Hilbert transform corresponds to a 90 degree phase shift in the time domain. A general modulated signal can be represented as the real part of the analytic signal and the amplitude of the analytic signal (\(|A(t)|\)) represents the signal envelope in the time domain. The orange waveform in Figure 3.2(a) shows a sample of signal envelope, demodulated using Hilbert transform. From the orange spectrum in Figure 3.2 (b), it can be seen that the spectrum of the signal envelope reveals useful information for fault diagnosis, such as the fault characteristic frequency \( f_0 \), its harmonic, and potential modulations.

Figure 3.2 Bearing fault pulses: (a) time-domain signal, and (b) power spectra.
C. **Convolutional neural network (CNN)**

As a multi-stage neural network, CNN starts with multiple convolutional layers, batch normalization (BN) layers, activation layers, and pooling layers, and ends with a classification layer (see Figure 3.3) [28, 29, 30].

The convolutional layer convolves the inputs with a set of unknown filters called kernels and then the activation layer generates the output features from the convolved inputs. Each kernel is a matrix with a fixed length and dimension and is learned during the training process. The kernels are used to extract the local features within a local region of the inputs. Since the same kernels are used to convolve the input units at each layer, the number of hyperparameters in CNN is much smaller than that of ANN, leading to a lower risk of over-fitting in the training process.

![Figure 3.3 The architecture of CNN.](image)
The BN layer is designed to reduce the shift of internal covariance and accelerate the training process of CNN. This layer is usually added between the convolutional layer and the activation layer.

The activation layer enhances the ability of the network to represent the nonlinearity of the input signals [28]. In recent years, a number of activation functions have been developed by researchers in the machine learning community. In this study, we implement Rectified Linear Unit (ReLU) since it can accelerate the convergence of network training. Similar to a down sampling operation, the pooling layer can be added after the activation layer to reduce the number of hyperparameters. In this study, we use max-pooling which performs the local max operation over the input features. The CNN architecture ends with a classification layer that is similar to those used in ANNs.

### 2.4. Physics-based convolutional neural network (PCNN)

As shown in Figure 3.4, the proposed PCNN architecture contains multiple layers, among which some in the front are built based on the physical knowledge about bearings and their fault characteristics and the others are purely data-driven. The first layer implements SK analysis to denoise the input signal using the optimum frequency/frequency resolution that maximizes the kurtosis of a subsignal. In the second layer, the denoised signal is demodulated to remove the carrier frequencies and only keep the diagnostic information.

The next layer, which serves as the most unique part of PCNN, is a convolutional layer that functions based on the similarity between the input signals (data) and the fault characteristic signals (physics). The kernels in this physics-based convolutional layer are generated based the rotational speed and fault characteristic frequencies of a bearing. Thus, no hyperparameters
are involved in generating the physics-based kernels. This feature makes it possible to have wide kernels without introducing additional hyperparameters to the PCNN model.

The outputs of the first three layers are in the time domain. However, due to the use of multiple sensor signals as the network inputs and the presence of unknown time delays between these signals, it is essential to convert the signals from the time domain to the frequency domain (see the fourth layer in Figure 3.4).

The remaining layers in Figure 3.4 (from the second convolutional layer to the last classification layer) are similar to those explained in Background section. Since the first four layers are predetermined based on the physics of bearings and their faults, these layers do not add any new hyperparameters to the set of hyperparameters that need to be learned during the training process. This can help greatly reduce the risk of over-fitting, which is a common issue in deep learning models with high-dimensional inputs [31, 32].
The proposed PCNN architecture.

A. Physics-based convolution

The first convolutional layer in the proposed PCNN architecture (the third layer in Figure 3.4) is one of the key parts of PCNN. Convolutional operator in CNN aims to extract the features from an input signal by quantifying the local similarities between the kernels and input signal. For instance, if the objective is to extract the vertical lines in a 2D image, the kernels with different vertical lines convolve the input image to find their similarities (vertical lines). In our task of bearing fault diagnosis, a good kernel should help identify whether an input signal has a modulating frequency close to a fault characteristic frequency of a bearing.
Therefore, the kernel should ideally simulate the signal produced by a faulty bearing. As such, the kernel can be pre-defined based on a sample fault characteristic signal. Suppose a bearing has M critical failure modes (or fault types). For the mth fault type \( m = 1, 2, \ldots, M \), the fault characteristic to be mapped into a PCNN is represented as a hand-constructed kernel that emulates the sensor signal from the bearing in the presence of the mth fault type. These M kernels, once generated, become the reference signals that capture physical knowledge about bearing fault characteristics. A reference signal can be generated by using a simulation model developed in earlier studies on bearing fault physics [33] and takes the following form

\[
\phi(k) = a_0 \sum_{s=0}^{K_w} \left[ \chi(k - s / f_0) \cdot e^{-\xi(k - s / f_0)} \right]
\]

(25)

where \( k \) is the time index \( k = 1, 2, \ldots, K_w \) with \( K_w \) being the kernel width and \( K_w \) is much less than the length of an input signal, \( a_0 \) is a constant amplitude that accounts for the radial load and fault severity, \( \phi(k) \) is the unit step function and simulates fault-induced impulses that switch on at time \( k = 0, \ldots, K_w \) is the fault characteristic frequency of the defective bearing, and \( \chi \) is the damping coefficient. Based on the geometry of the target bearing and its fault types, the fault characteristic frequencies can be easily calculated as described in [34]. A sample simulated kernel by assuming a fault characteristic frequency of \( f_0 \) is shown in Figure 3.5.

![Figure 3.5 A sample of physics-based kernel with characteristic frequency of \( f_0 \).](image)
After generating the reference kernels, an input signal $x$ can be convolved using the following formula

$$(x \otimes \phi)[k] = \sum_{v=0}^{K_c} x(k-v) \cdot \phi(v)$$

where $(x \otimes \phi)[k]$ denotes the convolution of $x$ with $\phi$. Figure 3.6 shows a sample convolution in the first convolutional layer of the proposed PCNN architecture. The input signal, which has already been filtered and demodulated, contains the fault-relevant information at a fault characteristic frequency of $f_0$. This signal is convolved using three physics-based kernels that are generated using equation (26) with the frequencies less than $f_0$, equal to $f_0$, and larger than $f_0$. By comparing the corresponding convolved signals, it can be clearly seen that the second convolved signal has the highest magnitude. Thus, it can be concluded that convolving an input signal by a kernel generated based on the characteristic frequency of a fault can help reveal the information relevant to the fault that is carried by the signal.
Figure 3.6 Convolutional operation on a sample pre-processed vibration signal of a defective bearing. The first plot from the top shows the sample vibration signal of a defective bearing with a fault characteristic frequency of $f_0$. The second, third, and fourth plots from the top show the convolved signals using three physics-based kernels generated with the frequencies less than $f_0$, equal to $f_0$, and higher than $f_0$, respectively.

2.5. Case studies

Two experimental studies were conducted to verify the performance of the proposed PCNN approach. The first case study was performed on a machinery fault simulator to assess the performance of PCNN in detecting artificially seeded bearing faults in the presence of other
types of malfunction including rotor unbalance and shaft misalignment. The second case study was performed on an agricultural baler, most often called a hay baler, to verify the performance of PCNN in a real-world application. It should be mentioned that most of the existing approaches for bearing fault diagnosis have been designed to handle a small number of sensor signals and often only one signal (e.g., acquired from a single vibration or AE sensor). However, there is often no absolute answer for the best type of sensor or the optimum number of sensors. Therefore, the two case studies leverage multiple sensors of different types and at different locations (i.e., vibration signals, AE signals and commercial microphones), and fuse these multiple sources of sensor information when diagnosing bearing faults. The detailed descriptions of the two case studies are provided in the following subsections.

A. Case study 1: machinery fault simulator

Experimental setup

An experiment was carried out on a machinery fault simulator to evaluate the performance of the proposed PCNN approach in detecting artificially seeded bearing faults. Two test bearings were mounted on the shaft of the simulator that was driven by an electric motor (see Figure 3.7(a)). Eight sensors including four vibrations sensors, two AE sensors and two consumer microphones were used to monitor the health of the bearings. The characteristics of the sensors and their positions are shown in Table 3.1 and Figure 3.7 (b), respectively. The sampling rate was selected to ensure the Nyquist frequency is much higher than the resonance frequency of the simulator (around 10 kHz) to gain some margin [10].
Inner race defect, outer race defect, and ball defect are known to be common types of bearing fault which could cause bearing failures [12]. As such, these three types of defect, along with their combination, were introduced to the test bearings. The size of the defect was approximately $1.5 \times 1.0 \times 0.1$ mm, similar to the size of a spall at an early stage of degradation. The test bearings were rolling element bearings, each with 13 balls. The inner race, outer race and ball diameters of each bearing are 22.1 mm, 29.1 mm, and 3.5 mm, respectively. Based on the bearing specifications and geometry, the fault characteristic frequencies of each bearing were calculated and are presented in Table 3.2.
Table 3.1 Summary of sensor characteristics

<table>
<thead>
<tr>
<th>channel #</th>
<th>type</th>
<th>max frequency (kHz)</th>
<th>sampling rate (kHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>vibration</td>
<td>12</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>vibration</td>
<td>12</td>
<td>100</td>
</tr>
<tr>
<td>3</td>
<td>AE</td>
<td>900</td>
<td>500</td>
</tr>
<tr>
<td>4</td>
<td>microphone</td>
<td>20</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>microphone</td>
<td>20</td>
<td>100</td>
</tr>
<tr>
<td>6</td>
<td>AE</td>
<td>900</td>
<td>500</td>
</tr>
<tr>
<td>7</td>
<td>vibration</td>
<td>12</td>
<td>100</td>
</tr>
<tr>
<td>8</td>
<td>vibration</td>
<td>12</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 3.2 Fault characteristic frequencies of the simulator bearings

<table>
<thead>
<tr>
<th></th>
<th>f_{o} 3.048 \times f_{R}</th>
<th>f_{i} 4.950 \times f_{R}</th>
<th>f_{bc} 1.992 \times f_{R}</th>
<th>f_{tr} 0.378 \times f_{R}</th>
</tr>
</thead>
</table>

The shaft was run under rotational speeds ranging from 10 Hz to 30 Hz. To simulate the true behavior of rotating machines in real-world applications, the rotor unbalance and shaft misalignment were also considered in the test plan. Additionally, the environmental sources of noise such as the electric motor were not mitigated to keep the testing condition close to real operating conditions, under which there is often no control on the sources of environmental noise. It should be mentioned that the level of noise that a sensor was exposed to varied depending on the type of the sensor and its distances from the noise sources (e.g., electric motor). Thus, the amount of useful diagnostic information carried in the sensor signal varied from one sensor to the other. Table 3.3 summarizes the design of experiments used in this case study. In total the simulator was run 2,340 times under different testing conditions.
<table>
<thead>
<tr>
<th>parameter</th>
<th>values</th>
</tr>
</thead>
<tbody>
<tr>
<td>shaft speed (Hz)</td>
<td>10, 12.5, ..., 30</td>
</tr>
<tr>
<td>misalignment level (in)</td>
<td>0, 0.01</td>
</tr>
<tr>
<td>rotor unbalance (gr)</td>
<td>0, 5</td>
</tr>
<tr>
<td>bearing 1 condition</td>
<td>no defect, inner race defect, outer race defect, ball defect, combination of defects</td>
</tr>
<tr>
<td>bearing 2 condition</td>
<td>no defect, inner race defect, outer race defect, ball defect, combination of defects</td>
</tr>
<tr>
<td>trials</td>
<td>1, 2, ..., 10</td>
</tr>
</tbody>
</table>

**PCNN architecture**

Table 3.4 shows the architecture of the PCNN model used in this case study. In the first two layers, the input data with eight signals, each of which has 30,000 by 1 elements, are filtered using SK and envelope analysis. These two layers are essential parts of the PCNN model since they can maximize the signal-to-noise ratio in the input signals. Figure 3.8 shows two sample raw signals (on the top plots), the outputs of SK analysis (on the middle plots) and the outputs of envelope analysis (on the bottom plots). It can be seen that the sequential execution of the signal processing techniques greatly reduces the noise in both signals.
Figure 3.8 Example of two raw signals and their corresponding pre-processed signals. Both input signals are from channel 6 and the operating condition during the data collection was 10 Hz shaft speed, zero level of misalignment, zero level of rotor unbalance, no defect on Bearing R and (a) inner race defect in Bearing L and (b) outer race defect in Bearing L.

Three reference kernels were generated, based on the shaft rotational speed and bearing fault characteristic frequencies, to simulate the vibration signals of bearings with inner race defect, outer race defect, and ball defect. The amplitude of a reference signal is determined by the parameter $a_0$ in equation (25). In this study, $a_0 = 1$ for all the three kernels. These kernels were used as the physics-based kernels in the third layer (i.e., physics-based convolutional layer) of PCNN. The size of a kernel depends on the fault characteristic frequency and the rotational speed of the shaft. In the fourth layer, the convolved time-domain signals are transformed into frequency-domain signals and the first 3,000 elements of the power spectrum
of each signal envelope (corresponding to a frequency range of 0 Hz to 500 Hz) are kept as the input to the fifth layer of PCNN.

In the fifth layer, eight wide kernels, each of length 48, convolve the input signals with a stride size of 8. The remaining layers followed the traditional CNN architecture explained in the background section.

The training and testing processes of PCNN were implemented using the Tensorflow package in Python 3.6 [36]. It should be mentioned that the first four layers of PCNN were built based on the physics of the bearings and their faults, and their parameters/kernels were pre-defined and did not need to be learned during the training process. However, all subsequent layers (i.e., layers 5-15) are purely data-driven and their parameters needed to be learned during the training process.

<table>
<thead>
<tr>
<th>no.</th>
<th>layer (type)</th>
<th>kernel size</th>
<th>number of kernels</th>
<th>kernel stride</th>
<th>output shape (width, depth)</th>
<th>padding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SK</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30000, 8</td>
<td>false</td>
</tr>
<tr>
<td>2</td>
<td>envelope</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30000, 8</td>
<td>false</td>
</tr>
<tr>
<td>3</td>
<td>convolution 1</td>
<td>[6100-10640]</td>
<td>3</td>
<td>1</td>
<td>30000, 24</td>
<td>false</td>
</tr>
<tr>
<td>4</td>
<td>FFT</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>500, 24</td>
<td>false</td>
</tr>
<tr>
<td>5</td>
<td>convolution 2</td>
<td>48</td>
<td>8</td>
<td>3</td>
<td>151, 24</td>
<td>true</td>
</tr>
<tr>
<td>6</td>
<td>max pooling 1</td>
<td>2</td>
<td>8</td>
<td>2</td>
<td>75, 24</td>
<td>false</td>
</tr>
<tr>
<td>7</td>
<td>convolution 3</td>
<td>9</td>
<td>16</td>
<td>3</td>
<td>20, 48</td>
<td>true</td>
</tr>
<tr>
<td>8</td>
<td>max pooling 2</td>
<td>2</td>
<td>16</td>
<td>2</td>
<td>10, 48</td>
<td>false</td>
</tr>
<tr>
<td>9</td>
<td>convolution 4</td>
<td>3</td>
<td>32</td>
<td>1</td>
<td>8, 96</td>
<td>true</td>
</tr>
<tr>
<td>11</td>
<td>max pooling 3</td>
<td>2</td>
<td>32</td>
<td>2</td>
<td>4, 96</td>
<td>false</td>
</tr>
<tr>
<td>12</td>
<td>convolution 5</td>
<td>3</td>
<td>32</td>
<td>1</td>
<td>2, 96</td>
<td>true</td>
</tr>
<tr>
<td>13</td>
<td>global max pooling</td>
<td>2</td>
<td>32</td>
<td>2</td>
<td>192</td>
<td>false</td>
</tr>
<tr>
<td>14</td>
<td>fully connecter</td>
<td>100</td>
<td>1</td>
<td>1</td>
<td>100</td>
<td>true</td>
</tr>
<tr>
<td>15</td>
<td>softmax</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>2-4</td>
<td>-</td>
</tr>
</tbody>
</table>
Results

As shown in Table 3.5, the performance of the proposed PCNN approach was evaluated via the use of two classification scenarios. The first scenario was used to evaluate the ability of PCNN to detect bearing faults in the simulator and the second scenario to assess the ability of PCNN to identify the locations of these faults (i.e., which bearing(s) are faulty). The number of samples used in each class are presented for both classification scenarios in Table 3.5.

<table>
<thead>
<tr>
<th>class name</th>
<th>number of data points</th>
</tr>
</thead>
<tbody>
<tr>
<td>fault detection</td>
<td></td>
</tr>
<tr>
<td>class 1: both bearings are healthy</td>
<td>540</td>
</tr>
<tr>
<td>class 2: at least one bearing is faulty</td>
<td>1800</td>
</tr>
<tr>
<td>fault localization</td>
<td></td>
</tr>
<tr>
<td>class 1: both bearings are healthy</td>
<td>540</td>
</tr>
<tr>
<td>class 2: only bearing R is faulty</td>
<td>360</td>
</tr>
<tr>
<td>class 3: only bearing L is faulty</td>
<td>360</td>
</tr>
<tr>
<td>class 4: both bearings are faulty</td>
<td>1080</td>
</tr>
</tbody>
</table>

The experiment was performed at nine different speeds ranging from 10 Hz to 30 Hz. To estimate the accuracy of the proposed PCNN, one set of data corresponding to one speed was left out as the testing data and the remaining data was first shuffled and then 70% of that was used for training the PCNN model (to optimize the unknown hyperparameters in layers 5-15) and the rest for validating the trained model (to identify possible opportunities for refinement of the model). This process was performed for each of the nine speeds (at each time one speed was considered as the testing speed). Figure 3.9 shows the accuracy of PCNN in fault detection and fault localization when the testing speed was varied from 10 Hz to 30 Hz. The results suggest that for all nine speeds, the proposed PCNN is able to detect and localize the faults with high accuracy under the noisy environment and in the presence of other sources of malfunction including rotor unbalance and shaft misalignment.
Figure 3.9 The accuracy of PCNN in fault detection and localization at the nine testing speeds.

Figure 3.10 graphically compares the performance of the proposed PCNN approach to that of multi-channel CNN and two traditional machine learning-based approaches (i.e., SVM and ANN). In multi-channel CNN, the input signals were first converted from the time domain to the frequency domain and were then fed into the CNN model. For the machine learning-based approaches, we followed the procedure described in [13]. First, the signal was filtered using SK analysis and then the RMS, crest factor, and kurtosis of the envelope signals from all eight channels were considered as the input features to SVM and ANN models. It can be seen that for both classification scenarios, the proposed PCNN outperforms the existing approaches. A comparison of the classification accuracy between multi-channel CNN and PCNN suggests that incorporating physical knowledge directly into the architecture of a deep learning model has the potential to significantly improve the performance of the purely data-driven model.
B. Case study 2: machine testing

The second case study explores the use of vibration and AE sensors in early fault detection of rolling element bearings used in hay balers. A hay baler is a farming machine used to compact cut hay into bales that are more efficient to transport and store. Continuous monitoring of the health conditions of the bearings in such agricultural equipment is a critical task as an unexpected failure of a bearing may lead to a large increase in the maintenance cost, incur a high downtime cost, and reduce the customer satisfaction. Therefore, practical solutions for monitoring the bearing health condition are important for customers and equipment manufacturers.

Experimental setup

Figure 3.11 shows the hay baler (504R Signature Baler manufactured by Vermeer Corporation) under test in this case study. Two bearings (FH208 manufactured by PEER Bearing Company with the inner race, outer race, and ball diameters being 47.5 mm, 60.2 mm,
and 12.7 mm, respectively) were identified to assess the health condition of the hay baler and are enlarged in Figure 3.11. For each bearing, two accelerometers and one AE sensor were installed on the outer surface of the bearing flange. These sensors are the same as those used in the first case study (see Table 3.6 for characteristics). Due to the high level of environmental sound and noise produced by the machine, the commercial microphones were not considered. The accelerometers were bolted to the flange perpendicular to the main shaft. Two magnetic holders were designed and manufactured to provide rigid contact between the AE sensors and the bearing flange surfaces. The two bearings and the sensor locations are shown in Figure 3.11.

Figure 3.11 The hay baler under test in Case study 2 and the locations of the sensors that monitor the health conditions of the two bearings.
Case study 2 considered four different types of fault including an inner race defect, outer race defect, ball defect, and combination of inner and outer race defects. The defects were artificially introduced to the inner race, outer race, and ball (see Figure 3.12). Table 3.6 presents the fault characteristic frequencies of the bearings used in this case study.

Figure 3.12 Bearing faults in Case study 2: (a) inner race fault, (b) outer race fault, and (c) ball fault.

Figure 3.13 A sample pre-processed input to PCNN in Case study 2.
Table 3.6 Fault characteristic frequencies of the bearing used in Case study 2

| Ball pass frequency of outer race $f_o$          | $3.552 \times f_R$ |
| Ball pass frequency of inner race $f_i$        | $5.448 \times f_R$ |
| Ball spin frequency $f_{bc}$                   | $4.537 \times f_R$ |
| Fundamental train frequency $f_{tr}$           | $0.394 \times f_R$ |

The hay baler main shaft was run under six different speeds from 300 rpm (5 Hz) to 550 rpm (9.16 Hz). The main shaft was connected to the shafts of the testing bearings via a gearbox with the speed ratio of 2.5. Thus, the frequencies of the bearing shafts were varied from 2.00 Hz to 3.67 Hz. At each operating condition, the vibration and AE data were collected for 10 s. In order to increase the number of data points, the data augmentation technique explained in Ref. [24] was utilized and each training sensor signal was sliced into 30 training samples with a predefined shift of 0.1 s. Table 3.7 summarizes the operating conditions used in this study. In total, the system was run 72 times under various operating conditions, which yielded a total 2,160 sliced training samples.

Table 3.7 Design of experiments for Case study 2

<table>
<thead>
<tr>
<th>Operating condition</th>
<th>Bearing health condition</th>
<th>Number of sliced samples $\times$ number of operating speeds</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 4</td>
<td>healthy</td>
<td>$30 \times 6$</td>
</tr>
<tr>
<td>5</td>
<td>healthy</td>
<td>$30 \times 6$</td>
</tr>
<tr>
<td>6</td>
<td>healthy</td>
<td>$30 \times 6$</td>
</tr>
<tr>
<td>7</td>
<td>healthy</td>
<td>$30 \times 6$</td>
</tr>
<tr>
<td>8</td>
<td>healthy</td>
<td>$30 \times 6$</td>
</tr>
<tr>
<td>9</td>
<td>inner race defect</td>
<td>$30 \times 6$</td>
</tr>
<tr>
<td>10</td>
<td>outer race defect</td>
<td>$30 \times 6$</td>
</tr>
<tr>
<td>11</td>
<td>ball defect</td>
<td>$30 \times 6$</td>
</tr>
<tr>
<td>12</td>
<td>defects combination</td>
<td>$30 \times 6$</td>
</tr>
</tbody>
</table>
Implementation of the proposed PCNN approach

In Case study 2, we followed the same fault diagnosis strategy. First, vibration and AE signals from six input channels were filtered by using SK and envelop analysis and then the first 3,000 elements of the power spectrum of each signal envelope (corresponding to the frequency range of 0 Hz to 500 Hz) was considered as input to the multi-channel CNN model. Similar to the first case study, a set of data corresponding to one speed was left out for use as a testing data set and 70% (30%) of the remaining data was used for the model training (validation). This process was performed for each of the six rotational speeds.

Results

A sample of pre-processed signal to the CNN model from the accelerometer mounted on the left side of the top bearing is shown in Figure 3.14. This signal was collected when the bottom bearing had no defect and the top bearing had an inner race defect. For the range of the rotational speed considered in this case study, the maximum fault characteristic frequency is expected to be less than 150 Hz. To cover the fault characteristic frequencies and their first two harmonics, we chose to use the frequency range of 0 Hz to 500 Hz when plotted the pre-processed input to PCNN in this figure. From the spectrum of the pre-processed signal it can be observed that the frequency components related to the inner race fault ($f_i = 12.71$ Hz) and its first two harmonics ($2f_i$ and $3f_i$) have higher magnitudes than the other frequency components. Figure 3.14 shows the accuracy of SVM, ANN and PCNN in fault detection at the six bearing shaft speeds. When implementing SVM and ANN, the same procedure explained in case study 1 was used. The accuracy of the proposed PCNN approach is higher
than both SVM and ANN at all testing speeds except the first one (i.e., 2.05 Hz). For the first testing speed, PCNN produced comparable accuracy to ANN.

![Comparison of fault detection accuracy between two existing (SVM and ANN) and the proposed PCNN approaches under various testing speeds in Case study 2.](image.png)

Figure 3.14 Comparison of fault detection accuracy between two existing (SVM and ANN) and the proposed PCNN approaches under various testing speeds in Case study 2.

2.6. Conclusion

In this study, a novel approach named physics-based convolutional neural network (PCNN) is proposed for fault diagnosis of rolling element bearings, with a special application on fault detection of rotating-element bearings. In the proposed approach, the conventional CNN is modified to incorporate useful information from physical knowledge about bearings and their fault characteristics. To this end, multiple physics-based layers are added to the front part of a conventional CNN model, making the model physically meaningful. Based on the bearing fault characteristic frequencies and the shaft speed, new physics-based kernels are generated for use in the convolutional layer of the proposed PCNN model. Replacing the data-driven kernels with the physics-based kernels in a deep learning model would reduce the number of training
parameters in the model and thus lower the risk of over-fitting, while improving the accuracy and robustness of the purely data-driven deep learning model.

Experiments were carried out on a machinery fault simulator and a hay baler to examine the performance of the proposed PCNN approach in monitoring the health of multiple bearings. Compared to the conventional machine learning- and CNN-based approaches, PCNN is able to detect and localize the faults with consistently higher accuracy.

In the proposed approach, the fault characteristic frequencies need to be pre-computed and fed into a PCNN model as pre-defined inputs. In our future work, we will attempt to treat the fault characteristic frequencies as part of the hyperparameters to be learned via the model training, with an aim to further improve the performance of PCNN.

2.7. References


CHAPTER 4. RECONSTRUCTION OF UNIDIRECTIONAL STRAIN MAPS VIA ITERATIVE SIGNAL FUSION FOR MESOSCALE STRUCTURES MONITORED BY A SENSIGN SKIN

Modified from the paper published in the journal of Mechanical Systems and Signal Processing [1].

3.1. Abstract

Flexible skin-like membranes have received considerable research interest for the cost-effective monitoring of mesoscale (large-scale) structures. The authors have recently proposed a large-area electronic consisting of a soft elastomeric capacitor (SEC) that transduces a structure's change in geometry (i.e. strain) into a measurable change in capacitance. The SEC sensor measures the summation of the orthogonal strain (i.e. $\varepsilon_x + \varepsilon_y$). It follows that an algorithm is required for the decomposition of the sensor signal into unidirectional strain maps. In this study, a new method enabling such decomposition, leveraging a dense sensor network of SECs and resistive strain gauges (RSGs), is proposed. This method, termed iterative signal fusion (ISF), combines the large-area sensing capability of SECs and the high-precision sensing capability of RSGs. The proposed method adaptively fuses the different sources of signal information (e.g. from SECs and RSGs) to build a structure's best fit unidirectional strain maps. Each step of ISF contains an update process for strain maps based on the Kriging model. To demonstrate the accuracy of the proposed method, an experimental test bench is developed, which is the largest deployment of the SEC-based sensing skin to date both in terms of size and sensor count. A network of 40 SECs deployed on a grid ($5 \times 8$) is utilized and an optimal sensor placement algorithm is used to select the optimal RSG sensor locations within the network of SECs. Results show that the proposed ISF method is capable of reconstructing
unidirectional strain maps for the experimental test plate. In addition to the experimental data, a numerical validation for the ISF method is provided through a finite element analysis model of the experimental test bench.

3.2. Introduction

Traditionally, mesoscale structural systems, including aerospace structures, energy systems and civil infrastructures are investigated and maintained using break-down based and time-based [2] strategies. An alternative is condition-based maintenance, which is known to have strong economic benefits for owners, operators, and society [3, 4]. Structural health monitoring (SHM) and life prediction are among the key components of the condition based maintenance [5, 6]. SHM is defined as the automation of damage detection, localization, and prognosis of structural systems and components.

A major challenge in the SHM of mesoscale structural systems is the distinction of global versus local faults [7, 8]. Also, since the monitored mesoscale structures can be geometrically complex [8], the selection of sensors and models capable of performing SHM can be challenging [7]. Of particular importance in the development of an SHM system is the consideration of sensor density. The use of dense sensor networks (DSNs) for SHM applications have attracted interest in recent years [9, 10, 11, 12, 13]. When compared to traditional sparse sensor networks, a DSN will provide for greater detection and localization of localized damage, including cracks [14, 15], material delamination [16, 17], corrosion [18], and loosening of bolts [19]. While a DSN has its advantages, it faces challenges in terms of high hardware requirements, complex installation, and high data management costs. Recently, through the use of micro-fabrication techniques [20] and advances in the field of flexible electronics [21], skin-like sensing membranes have been proposed as a solution for simplifying
the deployment and utilization of DSNs. These DSNs would fully integrate sensing, data
acquisition, data transmission, and power management into a sensing skin. The term sensing
skin is used because of their ability to mimic the capability of biological skin to detect and
localize events (e.g. damage, contact, temperature changes) over a large area [22].

Sensing skins for SHM applications have attracted significant attention in the last few years
and various sensing skins have been proposed and prototyped. These efforts have leveraged
various technologies, including: resistive strain gauges (RSGs) [14]; piezoceramic transducers
and receivers [23, 24]; carbon nanotube thin film strain sensors [25, 26]; electrically conductive
paint [27]; graphitic porous sensor arrays on polyimide; and photoactive nanocomposites [28].
The authors have previously proposed a fully integrated sensing skin [29] based on a novel
large-area electronic termed the soft elastomeric capacitor (SEC) [30]. The SEC was designed
to be inexpensive and benefits from an easily scalable manufacturing process.

In contrast with traditional strain sensors (e.g. RSG, fiber optic, and vibrating wire) that
measure unidirectional strain at discrete points, the SEC measures the additive strain over an
area. The SEC and its additive strain signal have been used for fatigue crack detection as well
as damage detection and localization over large areas. However, in cases where the
unidirectional strain maps of a structure are desired, it is imperative that the sensor's additive
signal be decomposed into its unidirectional components. Examples where a structure's
unidirectional strain maps are needed include: the incorporation of strain data into existing
strain based displacement [31] and damage detection [32] algorithms; model updating,
including finite element analysis (FEA) and analytical surrogate models [33]; and material
characterization [34, 35].
In situations where the structure's unidirectional strain maps are needed, the main challenge is to decompose the SEC's additive strain map into its linear strain components along two orthogonal directions. To address this challenge, the authors have previously developed an algorithm that leverages a dense sensor network (DSN) of SECs to decompose the additive strain maps. The algorithm assumes a polynomial deflection shape and appropriate boundary conditions and uses a least squares estimator (LSE) to estimate unidirectional strain maps over the DSN's area [36]. However in certain cases, such as the complex loading conditions present in a wind turbine blade [37, 38], accurate knowledge of the boundary conditions can be difficult or impossible to determine. To alleviate this challenge, RSGs were added to the DSN to allow for the real-time updating of boundary conditions at key locations, therefore, forming a hybrid DSN (HDSN). This extended LSE algorithm has been demonstrated for damage detection, both numerically and experimentally [38]. While computationally efficient, the extended LSE algorithm lacks the ability to reproduce nonlinear or complex strain maps due to its selection of a polynomial deflection shape function. The capability to reproduce nonlinear or complex strain maps is important as damage often manifests itself as nonlinearities in a unidirectional strain map (e.g., a thin crack in a plate) [38].

In this study, the authors propose a generic method, termed iterative signal fusion (ISF), that overcomes the difficulty of capturing high nonlinearities in strain responses and makes strain map reconstruction suitable for local damage detection. The method adaptively fuses the different sources of strain information from an HDSN containing both SECs and RSGs to build optimum and unique unidirectional strain maps. Each step of the ISF contains an update process for the strain maps based on a surrogate modeling technique. Various potential surrogate modeling techniques are based on radial basis functions [39], support vector
machines [40], artificial neural networks [41], and Kriging [42]. In the field of surrogate modeling, Kriging, or sometimes called Gaussian process regression, is a method of spatial interpolation for which the approximations are modeled by a Gaussian process derived by proper covariance [43]. The authors' previous studies showed that Kriging has strong benefits when it comes to processing data with a small number of sample points, a small number of input variables, and/or when the response shows a highly nonlinear behavior [43, 44]. Due to these benefits, Kriging is selected in this study as the surrogate modeling technique. Since the RSG and SEC sensors are located at different locations on the surface of a structure, a simple Kriging model cannot directly be used to generate the unidirectional strain maps based on the available data.

To address this issue, the proposed ISF method adaptively finds the best unbiased prediction of unidirectional strain data at the SEC sensor locations to virtually expand the set of strain data. Consequently, the unidirectional strain maps can be generated directly from this expanded data set using Kriging, or any other surrogate modeling techniques. As with any sensor network, the placement of sensors is a critical component of an SHM system [45]. The optimal sensor configuration is one that minimizes the presence of type I (false positive) or type II (false negative) errors. Therefore, it is critical to implement an optimal sensor placement strategy for determining the locations of sensors within an HDSN. For the particular case under study, a network of 40 SECs deployed on a grid (5 × 8) is utilized on an experimental test bench and an optimal sensor placement algorithm is used to select the optimal RSG sensor locations within the network of SECs. This optimal sensor placement algorithm [37], previously developed by the authors for use within the extended LSE algorithm, leverages the intuitive idea that not all potential sensor locations hold the same level of information. The key
contributions of this chapter are twofold: 1) it introduces an effective strain decomposition algorithm for the previously proposed SEC-based sensing skin that does not require the assumption of a shape function; and 2) it validates the proposed SEC-based sensing skin and the newly proposed ISF method through experimental results obtained from the largest deployment of the SEC-based sensing skin, both in terms of size and sensor count.

3.3. Background

This section covers the background that is needed to implement the ISF method. This includes a review of the SEC sensors and its electromechanical model, a previously investigated genetic algorithm to determine the optimal placement of RSGs within a network of SECs, and the discussion of a generic Kriging model.

A. Soft elastomeric capacitor (SEC)

The SEC is a robust large-area electronic that is inexpensive and easy to fabricate. Its architecture, manufacturing process, and electromechanical models are presented in Refs. [29, 30] and reviewed here for completeness. The SEC sensor takes the form of a parallel plate capacitor, as shown in Figure 4.1, where the dielectric is composed of a styrene-ethylene-butadiene-styrene (SEBS) block co-polymer matrix filled with titania (TiO₂) to increase both its durability and permittivity. Its conductive plates are fabricated using a conductive paint, made from the same SEBS, but filled with carbon black particles, painted onto each side of the SEBS matrix. Copper contacts, with an electrically conductive adhesive, are added to the conductors on both the top and bottom plates. These contacts are used for connecting the data acquisition to the SECs with a secure solder connection. Lastly, a thin layer of conductive paint is applied over the copper contacts to ensure a good connection between the copper contacts
and the conductors, as seen in Figure 4.1. Manufacturing of the SEC sensor in various shapes and sizes is relatively simple and does not require any highly specialized equipment or techniques, therefore allowing the technology to be easily scaled. To ensure the SEC is capable of monitoring the substrate in both tension and compression, the sensor is pre-stretched during its adhesion to the monitored substrate.

![Figure 4.1 A soft elastomeric capacitor (SEC) sensor with key components and reference axes annotated](image)

The capacitance \( C \) of a parallel plate capacitor can be modeled as a non-lossy parallel plate capacitor assuming a sampling rate of less than 1 kHz:

\[
C = \varepsilon_0 \varepsilon_r \frac{A}{h}
\]

where \( \varepsilon_0 = 8.854 \text{ pF/m} \) is the vacuum permittivity, \( \varepsilon_r \) is the polymer's relative permittivity, \( A \) is the sensor area of width \( d \) and length \( l \), and \( h \) is the thickness of the dielectric as annotated in Figure 4.1. Assuming small changes in strain, equation (27) leads to a differential equation that relates a change in strain to a change in capacitance (\( \Delta C \)):

\[
\frac{\Delta C}{C} = \frac{\Delta d}{d} + \frac{\Delta l}{l} - \frac{\Delta h}{h}
\]
where $\Delta d/d$, $\Delta l/l$, and $\Delta h/h$, can be expressed as strain $\varepsilon_x$, $\varepsilon_y$, and $\varepsilon_z$, respectively. Assuming a plane stress condition, $\varepsilon_z = -\nu \left( \varepsilon_x + \varepsilon_y \right)$ where $\nu$ is the sensor material's Poisson's ratio taken as $\nu \sim 0.49$. The relative change in capacitance $\Delta C$ can be related to a change in the sensor's deformation as:

$$\frac{\Delta C}{C} = \lambda (\varepsilon_x + \varepsilon_y)$$

(29)

where $\lambda = \frac{1}{1 - \nu}$ represents the gauge factor of the sensor. Since $\nu \sim 0.49$, the gauge factor can be estimated as $\lambda \sim 2$. Equation (29) shows that the signal of the SEC varies as a function of the sensor's additive strain, $\varepsilon_x + \varepsilon_y$.

The SEC's electro-mechanical model presented in equation (29) has been validated for both static and quasi-static loading conditions [29]. The linearity of the electro-mechanical model has been validated for mechanical excitation under 15 Hz [29].

**B. Optimal Sensor Placement**

The sensing skin used in this work consists of a network of SECs with a few RSGs distributed into the SEC grid to form an HDSN. The numbers and locations of RSGs within an HDSN affect the accuracy of the decomposed strain fields. Therefore, it is important to consider an optimal sensor placement scheme for the RSGs when validating the ISF method. The authors have previously developed a genetic algorithm with a learning gene pool for selecting optimal RSG sensor locations within a network of SECs [37]. The genetic algorithm leverages the intuitive idea that for a set of potential sensor locations ($P$), some sensor locations (p) add little or no information to the estimated system. Conversely, some sensor locations add a measurable level of information to the system. Therefore, the goal of the genetic algorithm
is to build a set of optimal sensor locations \( \mathbf{P} = [p_1, \ldots, p_m] \) that minimize the error between the system and its estimated state. This goal is achieved through linking sensor locations to genes. The probability of these genes (sensor locations) reoccurring are then mutated over generations by the genetic algorithm. After a sufficient number of generations, only the strongest genes remain and these form a set of sensor locations that constitute an optimal set of RSG locations within the network of the SECs.

In this work, the system is the true strain maps of the monitored substrate while the estimated state is the strain maps obtained through the ISF method. The error between the true strain field and its estimated state can be expressed in terms of type I and type II errors. In the case where strain maps are obtained for a structure with the intention of detecting damage, a type I error (false positive) is the incorrect classification of a healthy state as a damage state, while, a type II error (false negative) is the failure to detect a structural fault.

Here, a previously developed single objective function, borrowed from the field of robust design, is used in the multi-objective problem for decreasing the likelihood of type I and type II errors through the optimal placement of RSGs in the HDSN. The occurrence of type I errors within the HDSN's extracted strain maps is reduced through minimizing the mean absolute error (MAE) between the system and its estimated state. The use of MAE for selecting sensor locations provides an effective representation of how a structure will perform under various loading conditions. However, if the placement of RSGs is based solely on the MAE of the system, locations of high disagreement between the estimated and real systems will develop.

In the case of a load-carrying structural component, such an occurrence could result in the component being stressed passed its design limit (i.e. type II error). Therefore, to reduce the occurrence of type II errors, a second optimization problem based on minimizing the maximum
difference between the system and its estimated state (i.e. strain value) at any point on a strain map is introduced, defined as $E_{\text{max}}$. The bi-objective optimization problem (type I and II errors) can be simplified into a single objective function through a straightforward scalarization approach formulated as a linear combination of the bi-objective optimization problem. Considering $n$ possible sensor locations in $\mathbf{P}$, a single objective problem for optimizing the placement of $m$ sensors can be formulated as,

$$\text{minimize } \text{fit} = (1 - \alpha) \frac{\text{MAE}(\mathbf{P})}{\text{MAE}} + \alpha \frac{\text{E}_m(\mathbf{P})}{\text{E}_m}$$  \quad (30)$$

subject to

$$0 \leq m \leq n$$

$$0 \leq \alpha \leq 1$$

where $\alpha$ is a user-defined scalarization factor used to weight both objective functions and MAE and $E_{\text{max}}$ are factors used for normalizing MAE and $E_{\text{max}}$. While the selection of $\alpha$ depends on the structure's ability to tolerate type I or type II errors, the value of 0.5 has been shown to be a suitable value for similar problems [37].

C. Kriging (Gaussian Process Regression)

Kriging performs two main steps simultaneously: 1) it builds a trend function $\mathbf{h}(\mathbf{x}) \mathbf{\beta}$ based on the available data; and 2) it constructs a Gaussian process using the residuals $Z$ [42]. The Kriging-approximated model of the true response $\mathbf{G}(\mathbf{x})$ takes the following form

$$\hat{\mathbf{G}}(\mathbf{x}) = \mathbf{h}(\mathbf{x}) \mathbf{\beta} + \mathbf{Z}(\mathbf{x})$$  \quad (31)$$

where $\mathbf{Z}(\mathbf{x})$ is a Gaussian process with zero mean, variance $\sigma^2$, and a correlation matrix $\psi$. The objective is to capture the general trend or the largest variance in the data using the regression function and interpolate the residuals using the Gaussian process. The elements of matrix $\psi$
are derived by the kernel function that can take different forms to model the spatial correlation in random space. One popular choice is the squared exponential kernel with a vector of hyper-parameters $\theta$ [43]:

$$
\psi(x_i, x_j) = \exp \left( -\frac{1}{2} (x_i - x_j)^T \text{diag}(\theta)^{-2} (x_i - x_j) \right)
$$

(32)

where $x_i$ and $x_j$ are two arbitrary points in the input space. The hyper-parameters determine the smoothness of the prediction, and are estimated by maximizing the likelihood of observations given $\psi$. Subsequently, using the Sherman-Morrison-Woodbury formula, the prediction mean $\mu_G$ and uncertainty $\sigma_G^2$ of Kriging are expressed as:

$$
\mu_G(x) = h(x) \beta + r(x) \cdot \Psi^{-1} \cdot (y - F\beta)
$$

(33)

$$
\sigma_G^2(x) = s^2 \left[ 1 - r(x) \Psi^{-1} r(x)^T + \frac{(1-F^T\Psi^{-1}r(x)^T)}{F^T\Psi^{-1}F} \right]
$$

(34)

where $h(x)$ is a vector of $p$ trend functions, $y$ is a vector of $t$ responses, $\beta$ is the $p$-element vector of the coefficients of the trend functions, and $r(x) = [\psi(x, x_1), ..., \psi(x, x_t)]^T$ is a correlation vector between training and testing points. The process variance can be determined as $s^2 = \frac{1}{t} \cdot (y - F\beta)^T \Psi^{-1} (y - F\beta)$. More details about the Kriging model can be found in reference [42].

### 3.4. Iterative Signal Fusion (ISF)

This work proposes the new ISF method for strain map reconstruction, with the objective to minimize the loss of information when fusing the various signals from HDSNs. To build strain maps from a DSN with a single type of sensor, one may simply use traditional surrogate
modeling techniques (e.g. Kriging) when the source of strain data is limited to the one type of sensor, provided the strain data is obtained for the correct orientation. However, in the HDSN of interest both unidirectional and additive strain data are collected at different RSG and SEC sensor locations (see Figure 4.4(b)). Therefore, a direct implementation of any traditional surrogate modeling technique would not leverage all potential information in the unidirectional strain map reconstruction. To overcome this challenge, the proposed ISF method adaptively fuses the multiple sources of strain information from both the SECs and RSGs to build an optimal prediction of the unidirectional strain maps. In what follows, the traditional method and proposed methods are explained in the form of two scenarios.

A. Scenario 1 - Traditional Method

First, consider the scenario where no information fusion is applied. The strain measurements collected by an HDSN can be grouped into three data sets (see the solid-line boxes in Figure 4.2) 1) x-direction strains ($\varepsilon_x$) at the location of RSG-x sensors ($I_{RSG-x}$), 2) y-direction strains ($\varepsilon_y$) at the location of RSG-y sensors ($I_{RSG-y}$), and 3) additive strains ($\varepsilon_x + \varepsilon_y$) at the location of SEC sensors ($I_{SEC}$). Next, taking $O$ as the measured strain data, superscripts are added to denote sensor type/locations and subscripts are added to denote strain map type. For example, $O_{\varepsilon_x+\varepsilon_y}^{SEC}$ represents the additive strain data at the locations of the SEC sensors while $O_{\varepsilon_x+\varepsilon_y}^{RSG-x}$ represents the additive strain data at the location of the RSG-x sensors.
As shown in Figure 4.2, traditional surrogate modeling techniques such as Kriging build the model for each of the three available strain data sets (i.e. $\varepsilon_{x\text{RSG-x}}$, $\varepsilon_{y\text{RSG-y}}$, and $\varepsilon_{x+y\text{SEC}}$) separately and independently. Therefore, the $\varepsilon_x$ strain map, the $\varepsilon_y$ strain map, and the $\varepsilon_x+\varepsilon_y$ strain map at an arbitrary point $(x, y)$ on the surface of a structure are defined as:

\[
\varepsilon_x = GP((x, y) \mid D = \{(x_{\text{RSG-x}}, y_{\text{RSG-x}})\})
\]

(35)

\[
\varepsilon_y = GP((x, y) \mid D = \{(x_{\text{RSG-y}}, y_{\text{RSG-y}})\})
\]

(36)

\[
\varepsilon_x + \varepsilon_y = GP((x, y) \mid D = \{(x_{\text{SEC}}, y_{\text{SEC}})\})
\]

(37)

where $GP((x, y) \mid D)$ denotes the prediction at the arbitrary point $(x, y)$ in the 2-D input space using the Gaussian process or Kriging model which is trained based on the data set $D$. Each model is built using separate data sets and thus there is no correlation between the outputs. For instance, the SEC sensor data is not used for constructing either the $\varepsilon_x$ strain map nor the $\varepsilon_y$ strain map.
B. Scenario 2 - Proposed Method

Second, consider a scenario that leverages the correlation among the different sources of data in constructing the unidirectional strain maps. The ISF method is proposed based on this premise. To fuse the different sources of information, the proposed ISF method iteratively exploits all three strain measurement sets to estimate the strain responses at sensor locations where such responses are not measured. Figure 4.3 shows the flowchart of the ISF method. A solid-line box denotes a directly measured strain and a dashed-line box denotes a strain that is not directly measured and needs to be estimated using the ISF method. To this end, Kriging is used to find the best unbiased prediction of strain data at the dashed-line boxes using the available data sets (i.e. RSG-x, RSG-y, and SEC sensor data) as the training data sets. It follows that the unidirectional strain maps can be generated directly from Kriging or any other surrogate modeling techniques based on the expanded data sets at all solid-line and dashed-line boxes.

Figure 4.3 Flowchart of the proposed ISF method.
Algorithm 1: Procedure of ISF using Kriging to construct the strain maps

1. Build the initial Kriging model for all three strain maps
2. Calculate the error estimator $\zeta$
3. while $\zeta < \zeta_0$ do
   4. Step 1: $\varepsilon_y$ map at RSG–x sensors location.
   5. Step 2: $\varepsilon_x + \varepsilon_y$ strain map at RSG–x sensors location
   6. Step 3: $\varepsilon_x + \varepsilon_y$ strain map at RSG–y sensors location
   7. Step 4: $\varepsilon_x$ strain map at RSG–y sensors location
   8. Step 5: $\varepsilon_x$ strain map at SEC sensors location
   9. Step 6: $\varepsilon_y$ strain map at SEC sensors location
10. Calculate the error estimator $\zeta$
end while
11. Build the final Kriging models

In Figure 4.3, all nine possible strain data sets are shown in the main middle block. The three solid-line boxes represent the available strain data sets and the six dashed-line boxes show the unavailable data sets for which one attempts to find the best unbiased predictions (called virtual data set). A pseudo-code of the proposed method is provided in Table 4.1. The algorithm starts with finding initial guesses for the virtual data sets using the available data sets:

$$[O^{RSG-y}_{x}, O^{SEC}_{x}] = GP(I^{RSG-y}_{x}, I^{SEC}_{x} | \mathcal{D} = \{I^{RSG-x}_{x}, O^{RSG-x}_{x}\})$$

$$[O^{RSG-x}_{y}, O^{SEC}_{y}] = GP(I^{RSG-x}_{y}, I^{SEC}_{y} | \mathcal{D} = \{I^{RSG-y}_{y}, O^{RSG-y}_{y}\})$$

$$[O^{RSG-x}_{x+y}, O^{RSG-y}_{x+y}] = GP(I^{RSG-x}_{x+y}, I^{RSG-y}_{x+y} | \mathcal{D} = \{I^{SEC}_{x+y}, O^{SEC}_{x+y}\})$$

After finding the initial guesses, the virtual data sets are updated iteratively until the optimal prediction is achieved. As shown by the small arrows in Figure 4.3, each iteration consists of six sequential steps, each of which updates a Kriging (or strain response) model with the most recent strain measurements/estimates and uses the updated model to estimate the
strain responses pertaining to one of the virtual data sets. Step 1 estimates \( \varepsilon_y \) at \( \mathbf{I}_{RSG-y} \) (virtual data set \( \mathbf{O}_{\varepsilon_y}^{RSG-x} \)) based on all available y-strain measurements/estimates, \( \mathbf{O}_{\varepsilon_y}^{RSG-y} \) and \( \mathbf{O}_{\varepsilon_y}^{SEC} \), with the following form:

\[
\mathbf{O}_{\varepsilon_y}^{RSG-x} = GP\left(\mathbf{I}_{RSG-x} \mid \mathcal{D} = \{(\mathbf{I}_{RSG-y}, \mathbf{O}_{\varepsilon_y}^{RSG-y}), (\mathbf{I}_{RSG-y}, \mathbf{O}_{\varepsilon_y}^{SEC})\}\right) \tag{41}
\]

At Step 2, \( \mathbf{O}_{\varepsilon_y}^{RSG-x} \) (i.e. \( \varepsilon_y \) at \( \mathbf{I}_{RSG-x} \)) is used to update the additive strain data \( \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{RSG-x} \) at the same locations:

\[
\mathbf{O}_{\varepsilon_x+\varepsilon_y}^{RSG-x} = \mathbf{O}_{\varepsilon_x}^{RSG-x} + \mathbf{O}_{\varepsilon_y}^{RSG-x} \tag{42}
\]

At Step 3, the virtual data set \( \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{RSG-y} \) is updated using a Kriging model trained with the true data set \( \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{SEC} \) and virtual data set \( \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{RSG-x} \).

\[
\mathbf{O}_{\varepsilon_x+\varepsilon_y}^{RSG-y} = GP\left(\mathbf{I}_{RSG-y} \mid \mathcal{D} = \{(\mathbf{I}_{RSG-y}, \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{SEC}), (\mathbf{I}_{RSG-x}, \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{RSG-x})\}\right) \tag{43}
\]

At Step 4, the updated \( \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{RSG-y} \) is used to predict \( \mathbf{O}_{\varepsilon_x}^{RSG-y} \):

\[
\mathbf{O}_{\varepsilon_x}^{RSG-y} = \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{RSG-y} - \mathbf{O}_{\varepsilon_y}^{RSG-y} \tag{44}
\]

At Step 5, \( \mathbf{O}_{\varepsilon_x}^{SEC} \) is updated using the following equation:

\[
\mathbf{O}_{\varepsilon_x}^{SEC} = GP\left(\mathbf{I}_{SEC} \mid \mathcal{D} = \{(\mathbf{I}_{RSG-x}, \mathbf{O}_{\varepsilon_x}^{RSG-x}), (\mathbf{I}_{RSG-y}, \mathbf{O}_{\varepsilon_y}^{RSG-y})\}\right) \tag{45}
\]

Lastly, Step 6 updates \( \mathbf{O}_{\varepsilon_y}^{SEC} \) using the values solved for in Steps 4 and 5 (equations (44) and (45)):

\[
\mathbf{O}_{\varepsilon_y}^{SEC} = \mathbf{O}_{\varepsilon_x+\varepsilon_y}^{SEC} - \mathbf{O}_{\varepsilon_x}^{SEC} \tag{46}
\]

After performing the 6 sequential steps, the strain estimates in all virtual data sets (dashed-line boxes) will be updated. The iteration continues until the level of change in the strain values
pertaining to all dashed-line boxes converges close to zero. To this end, an error estimator is defined as:

$$\xi = \mathbf{O}_{\varepsilon_y}^{RSG-x} - GP\left(\mathbf{I}_{RSG-x}^{RSG-y}, \mathbf{D} = \{\mathbf{I}_{RSG-y}^{RSG-y}, \mathbf{O}_{\varepsilon_y}^{RSG-y}, (\mathbf{I}_{SEC}, \mathbf{O}_{\varepsilon_y}^{SEC})\}\right)$$

(47)

If the change in $\mathbf{O}_{\varepsilon_y}^{RSG-x}$ over sequential iterations converges to a same number (i.e. $\xi < \zeta_0$), then the algorithm is stopped and the final Kriging models are built based on all measured/estimated $\varepsilon_x$ and $\varepsilon_y$ strain data to reconstruct the unidirectional strain maps, expanded for the entire surface area of the structure:

$$\varepsilon_x = GP\left(\mathbf{(x, y)}\mathbf{D} = \{\mathbf{I}_{RSG-x}^{RSG-x}, \mathbf{O}_{\varepsilon_x}^{RSG-x}, (\mathbf{I}_{RSG-y}^{RSG-y}, \mathbf{O}_{\varepsilon_y}^{RSG-y}, (\mathbf{I}_{SEC}, \mathbf{O}_{\varepsilon_x}^{SEC})\}\right)$$

(48)

$$\varepsilon_y = GP\left(\mathbf{(x, y)}\mathbf{D} = \{\mathbf{I}_{RSG-x}^{RSG-x}, \mathbf{O}_{\varepsilon_y}^{RSG-y}, (\mathbf{I}_{RSG-y}^{RSG-y}, \mathbf{O}_{\varepsilon_y}^{RSG-y}, (\mathbf{I}_{SEC}, \mathbf{O}_{\varepsilon_y}^{SEC})\}\right)$$

(49)

3.5. Methodology

This section presents the methodology used in validating the ISF method. First, an experimental test bench specifically designed for validating the ISF method is introduced, followed by the introduction of an FEA model of the test bench that is used for the numerical validation of the ISF method.

A. Experimental setup

The test bench developed for validating the ISF method is shown in Figure 4.4. An HDSN consisting of 40 SECs and 20 RSGs (10 measuring $\varepsilon_x$ and 10 measuring $\varepsilon_y$) was deployed onto the surface of a fiberglass plate with a geometry of $500 \times 900 \times 2.6$ mm$^3$, as shown in Figure 4.4(a). Figure 4.4(b) is a schematic of the sensor layout showing the locations of the SECs and
RSGs, where each RSG location has two RSGs (model #FCA-5-350-11-3LJBT, manufactured by Tokyo Sokki Kenkyujo), individually measuring $\varepsilon_x$ and $\varepsilon_y$.

Figure 4.4 Experimental setup used for validating the proposed method: (a) picture of the test bench with key components annotated; (b) schematic of the test bench showing the locations of the sensors, loading point, and added mass.

The RSG locations are numbered 1-10, for later use in selecting RSGs to be utilized as part of the ISF method. Additionally, the four SECs denoted A, B, C, and D are used for investigating temporal strain data. A yellowing is present on some of the sensors' dielectrics (see sensors A and C for example). This yellowing does not appear to affect the sensors' strain measurements, as it will be discussed later in this work. The plate's left-hand side is bolted to an aluminum support ($12.7 \times 76.2 \times 500$ mm$^3$) to form a rigid connection. The rigid connection was added to eliminate the strain complexities that would be present if the hinges were connected directly to the fiberglass plate. This rigid connection is then attached to the frame through a pinned connection. The right-hand side of the plate is restrained in the vertical direction through the use of two lightly greased rods of diameter 12.7 mm to form a roller
connection. Each SEC covers an area of 38×38 mm² and these SEC sensors are deployed in a 5×8 grid array. The center of an SEC sensor is used as the location of the sensor in the ISF method. The SEC and RSG data are sampled simultaneously at 17 samples per second. The SECs are measured using a custom-built data acquisition system that includes active shielding in the cable to remove the cable's parasitic capacitance. The RSGs are measured using three quarter bridge analog input modules (NI-9236, manufactured by National Instruments) mounted in a chassis (cDAQ-9178, manufactured by National Instruments). Additionally, the same chassis is used to obtain measurements from the LVDT (model #0244, manufactured by Trans-Tek) measured through a 16-bit analog input module (NI-9205) while also providing a simultaneous trigger source for the SEC and RSG DAQs through a sourced digital output (NI-9472, manufactured by National Instruments).

Two experimental load cases are considered during the course of this work. For load case 1, the plate is excited with a displacement controlled force at the center of the plate, as annotated in Figure 4.4, sourced from a stepper motor located under the plate. The excitation force is a 20 mm sinusoidal load at 0.25 Hz. Load case 2 uses the same driving displacement and frequency, but includes a 0.5 kg mass added to the edge of the plate (see Figure 4.4(b)) to introduce some complexities into the strain maps. To eliminate any high-frequency noise in the SEC signal, a fifth order Butterworth filter with a cutoff frequency of 10 Hz was applied to the raw SEC signals. No filtering was applied to the RSG signals.

B. FEA model

Numerical validation for the ISF method is performed through an FEA model of the experimental test bench created in Abaqus. The FEA model included the fiberglass plate and the rigid aluminum connection that connects the pinned connections to the fiberglass plate.
was constructed using 298,065 eight-node brick elements with 1 integration point to allow for simple modeling of the connection between the fiberglass plate and the rigid aluminum connector. Constraints were modeled as a pinned connection at the plate's left-hand side and a roller connection on the plate's right-hand side. All materials were considered to be isotropic. In the fiberglass plate, nine elements are used through its thickness to prevent shear locking. A convergence test was performed and the selected model parameters yielded an error of less than 1% when compared to the FEA model with 1.2 million elements. The key parameters of the FEA model used in this numerical validation are listed in Table 4.2, where the material constants for the aluminum were taken from the material's data sheet and the material properties for the fiberglass were obtained experimentally from material drops. Similar to the experimental validation, two load cases are considered: 1) load case 1 consists of the plate displaced 20 mm upward at the middle; and 2) load case 2 consists of the same displacement but with the addition of a 0.5 kg load at the center along the top edge, as shown in Figure 4.4(b).

Table 4.2 Parameters used in constructing the FEA model.

<table>
<thead>
<tr>
<th>parameter</th>
<th>value</th>
<th>parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abaqus element type</td>
<td>C3D8R</td>
<td>Young's Modulus (aluminum)</td>
<td>68.9 GPa</td>
</tr>
<tr>
<td>element type</td>
<td>linear brick</td>
<td>Poisson's ratio (aluminum)</td>
<td>0.33</td>
</tr>
<tr>
<td>element nodes</td>
<td>8</td>
<td>density (aluminum)</td>
<td>2,700 $kg/m^3$</td>
</tr>
<tr>
<td>element integration points</td>
<td>1</td>
<td>Young's Modulus (fiberglass)</td>
<td>15 GPa</td>
</tr>
<tr>
<td>elements total</td>
<td>298,065</td>
<td>Poisson's ratio (fiberglass)</td>
<td>0.21</td>
</tr>
<tr>
<td>elements (aluminum connection)</td>
<td>32,340</td>
<td>density (fiberglass)</td>
<td>4,500 $kg/m^3$</td>
</tr>
<tr>
<td>elements (fiberglass plate)</td>
<td>265,725</td>
<td>plate dimensions</td>
<td>500×900×3.18 mm$^3$</td>
</tr>
</tbody>
</table>
Optimal sensor placement using the genetic algorithm was performed using the results for both load case 1 and 2. The genetic algorithm was solved over 500 generations using a population of 50 offsprings per generation. An initial guess for the genetic algorithm was generated by finding the lowest MAE for a set of 50 randomly selected RSG sensor locations. Then, MAE and $E_{\text{max}}$ were set by solving for the MAE and point of maximum disagreement ($E_{\text{max}}$) for the initial guess. Table 4.3 reports the RSG sensors, numbered to correspond with the RSG sensor locations depicted in Figure 4.4(b), used for developing the strain maps. When calculating the error between the FEA and ISF generated strain maps, every point on the FEA model was used excluding a $50 \times 50 \, \text{mm}^2$ square around the loading point. This was excluded as the FEA creates relatively high, highly localized strain values around the 30 mm circular loading point used in the FEA model to simulate the washer used in the real experimental setup.

Table 4.3 Locations of RSGs used in the ISF method.

<table>
<thead>
<tr>
<th>Number of RSGs used</th>
<th>RSGs locations user for $\varepsilon_x$</th>
<th>RSGs locations user for $\varepsilon_y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>8, 10</td>
<td>3, 7</td>
</tr>
<tr>
<td>8</td>
<td>1, 4, 5, 7</td>
<td>1, 8, 9, 10</td>
</tr>
<tr>
<td>12</td>
<td>3, 5, 6, 7</td>
<td>1, 4, 5, 6, 7, 8, 10</td>
</tr>
</tbody>
</table>

3.6. Results

A. Temporal strain data

Figure 4.5 presents the temporal data results for sensors A, B, C, and D as denoted in Figure 4.4(b). These sensors were selected to demonstrate the range of SEC sensor signals under
varying strain conditions, including two sensors that experienced the yellowing of the dielectric. For clarity, only every other strain data point is reported for an individual SEC sensor with its corresponding marker type. The raw SEC signal is presented as a hollow marker while the signal filtered with a low-pass Butterworth filter is presented as a filled marker on a dotted line.

Figure 4.5 SEC signal for sensors A, B, C, and D under load case 1; here, only every other data point is shown for clarity.

As shown in Figure 4.5, sensor D experiences the highest level of noise, which is due in part to the sensor having the longest cable at 1.2 meters and the lowest level of strain. However, even with the high noise level and relatively low localized strain, the filtered signal provides a smooth signal that can be used for the ISF method.
B. Numerical investigation of strain maps

Figure 4.6 reports the strain maps obtained through the FEA analysis (first row) for both load cases and the strain maps estimated using the ISF method with 4, 8, and 12 RSGs. From the FEA analysis, a difference can be observed in strain maps developed using load case 1 and those using load case 2, particularly in the $\varepsilon_x$ strain maps.

Figure 4.6 Strain maps obtained through the FEA and the ISF method using the experimental data with 4, 8 and 12 RSGs.

The added mass exerts a compressive force on the top of the plate where the mass is added (see Figure 4.4(b) for the location of the added mass). This compressive force reduces the
magnitude of the tensile strain along the top of the plate. The unidirectional strain maps
developed using the ISF method with 4, 8, and 12 optimally placed RSGs (Table 4.3) are
presented on rows 2, 3, and 4, respectively.

Load case 1, a simpler loading configuration, is generally easily solved for using any numbers
of RSGs. The largest points of disagreement between the FEA and ISF strain maps are around
the loading connection. This is as hypothesized, because the sensor network is relatively sparse
compared to the complexity of this local strain topology. In comparison, the more complicated
strain topology caused by load case 2 benefits more from the increasing number of strain
gauges.

With a sufficient number of RSGs, the reconstructed strain maps try to fit the complex
strain topologies around the location of the mass. In particular, the ISF method with 8 and 12
RSGs benefits from the higher numbers of RSGs as the optimal sensor placement algorithm
selected RSG location #1 for the RSGs that measure $\varepsilon_y$. This added strain information at a
location close to the added mass allows the ISF method to greatly increase its ability to track
the complex strain topology, although this highly localized information causes the ISF method
to overestimate the spatial distribution of the compressive load at top of the plate, as depicted
by the large purple area in the $\varepsilon_y$ strain maps for 8 and 12 RSGs in load case 2.

C. Effect of RSGs on strain maps

Next, the effect of increasing the number of RSGs used in the ISF method is investigated
and presented in Figure 4.7. This study uses the FEA model's derived strain maps to better
investigate the effect of adding RSGs to the ISF method without considering the effect of other
complications found in experimental testing (i.e., noise). Results are quantified using the error
in the FEA strain maps (MAE and $E_{\text{max}}$) for both loading conditions. As before, a $50 \times 50$ mm$^2$ square around the loading point is excluded when calculating the error to prevent the highly localized strain results from complicating the investigation. As expected and plotted in Figure 4.7, the introduction of more RSGs into the ISF method results in a reduction of both quantifiable error values.

![Graph showing MAE and $E_{\text{max}}$ as a function of RSGs used](image)

Figure 4.7 ISF reconstruction error as a function of the number of RSGs used in the algorithm formulation. The error is calculated using both load cases 1 and 2.

**D. Experimental investigation of strain maps**

Here, the experimental implementation of the ISF method is presented. Figure 4.8 shows that the ISF method is capable of reconstructing strain maps for the experimental test plate. While no full-field strain data is available for the experimental test bench, the algorithm does generate strain maps close to those predicted by the FEA model. This is particularly true in load case 2 where the ISF method is capable of capturing the complex topology caused by adding the mass to the top of the plate. Deviations between the FEA model results and the experimental data could be caused by various factors, including material variations, imperfect
loading conditions, and the fact that the FEA model does not account for the added mass and stiffness from the sensor wires.

Figure 4.8 Strain maps obtained through the ISF method using the experimental data with 4, 8, and 12 RSGs.

Figure 4.9 reports the temporal error results for the case with the ISF method with 12 RSGs over a typical load cycle. For this figure, the error is calculated by using the readings from all 20 RSGs. The RSGs were selected due to their high reliability and low level of noise. As expected, the error parameters increase when the magnitude of the displacement increases. This is due to the higher levels of strain in the system. Figure 4.9(a) presents the temporal error data for load case 1 while Figure 4.9(b) presents the temporal error data for load case 2. As expected, the errors are consistently higher for load case 2 (Figure 4.9(b)) due to the more complex loading configuration.
3.7. Conclusion

We have proposed a robust method for the development of unidirectional strain maps from the additive strain signal of a novel large-area electronic, termed the soft elastomeric capacitor (SEC). When deployed in a network configuration, SECs can cover large-scale surfaces and can be used to reconstruct physics-based features for condition assessment, such as strain maps.
and deflection shapes. Given that each SEC measures the summation of the orthogonal strains (i.e. \( \varepsilon_x + \varepsilon_y \)), the proposed method retrieves the magnitude and directional information of strain prior to reconstructing strain maps. The proposed method, termed iterative signal fusion (ISF), adaptively fuses the different sources of signal information (e.g. from SECs and RSGs) to build best-fit unidirectional strain maps for the monitored structure. Each step of ISF contains an update process for strain maps based on a Kriging model. We have investigated the accuracy of the proposed method by developing an experimental test bench which is the largest deployment of the SEC-based sensing skin to date. We have utilized a network of 40 SECs deployed on a grid (5×8) and an optimal sensor placement algorithm to select the optimal RSG sensor locations within the network of SECs. This optimal sensor placement algorithm, previously developed by the authors, leverages the intuitive idea that not all potential sensor locations hold the same level of information. Two experimental load cases were considered during the course of this work. These load cases consist of a displacement controlled force at the center of the experimental plate and a similar load case but with a mass added to the edge of the plate to introduce some complexities into the strain maps. For both load cases, the results show that the proposed ISF method successfully develops strain maps for the experimental test plate. In addition, a finite element analysis model of the experimental test bench was developed to numerically verify the accuracy of the proposed ISF method. While no full-field strain data is available for the experimental test bench, we have shown that the results of unidirectional strain maps reconstructed using the ISF method strongly correlate with the results generated by the numerical finite element analysis model.
3.8. References


10. A. Downey, S. Laflamme, F. Ubertini, Experimental wind tunnel study of a smart sensing skin for condition evaluation of a wind turbine blade, Smart Materials and


CHAPTER 5. GENERAL CONCLUSION

In this research, novel data-driven methods for simulation-based design and sensor-based diagnostics of engineering systems have been developed. These methods can be used to improve the failure resilience of engineering systems prior and posterior to the design process. In particular, the contributions that this research made to the scientific community are listed as follows.

- A new method named as high-dimensional reliability analysis (HDRA) has been proposed to solve reliability analysis problems involving high dimensionality, computationally expensive simulations, high nonlinearity, and strong variate interactions.

- A robust data-driven method for the development of unidirectional strain maps from the additive strain signals has been proposed. The proposed method, termed iterative signal fusion (ISF), adaptively fuses the different sources of signal information to build best-fit unidirectional strain maps for the monitored structure.

- A novel approach named physics-based convolutional neural network (PCNN) has been proposed for fault diagnosis of rolling element bearings, with a special application on fault detection of rotating-element bearings.

In this research, the devolved fault diagnosis approaches require sensors with high resolution and sampling rates to monitor the health condition of the engineering systems. However, in many industrial applications, installing and maintaining these high-end sensors is too costly. Additionally, most fault diagnosis solutions require either wired instrumentation or manual inspection on individual machines and do not scale well to production settings where large quantities of machines need to be monitored. Future studies will tackle these challenges
by examining the feasibility of the fault diagnosis methods on the industrially hardened Industrial Internet of Things (IIoT) platforms. Future work will focus on fully developing and validating the low-cost and scalable IIoT platforms, where the smart devices run the developed diagnosis models via edge processing and send fault diagnosis results to a web-based Maintenance Hub that provides real-time analytics, dashboards and alert capabilities.