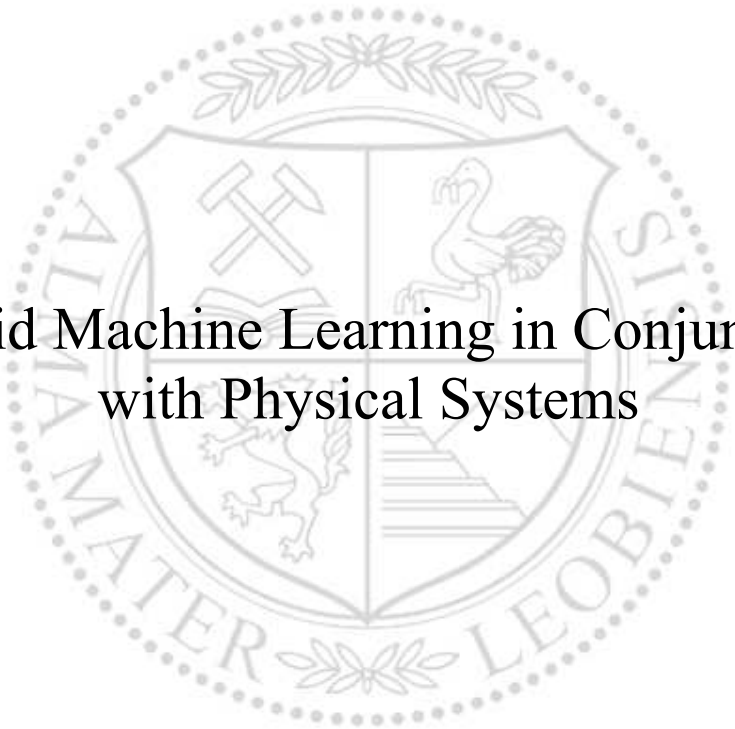




Chair of Automation

Doctoral Thesis



Hybrid Machine Learning in Conjunction
with Physical Systems

Dipl.-Ing. Anika Teresa Terbuch, BSc

November 2023



AFFIDAVIT

I declare on oath that I wrote this thesis independently, did not use other than the specified sources and aids, and did not otherwise use any unauthorized aids.

I declare that I have read, understood, and complied with the guidelines of the senate of the Montanuniversität Leoben for "Good Scientific Practice".

Furthermore, I declare that the electronic and printed version of the submitted thesis are identical, both, formally and with regard to content.

Date 04.12.2023

Anika Teresa Terbuch

Signature Author
Anika Teresa Terbuch

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor Professor Paul O’Leary and mentor Professor Peter Auer. Their support, insightful discussions and freedom to explore my ideas have been essential in bringing this thesis to completion. I appreciate the collaboration with my co-authors, whose contributions have been crucial in shaping the knowledge presented in this thesis.

A big thank you to everyone at the Chair of Automation for creating such a supportive working environment that made this journey possible.

I can’t thank my boyfriend, family and friends enough for being my rock throughout my studies. Prísrčna hvala! Thank you!

Abstract

This thesis investigates methods and architectures for machine learning that are particularly suited for the analysis of data emerging from the observation of physical processes and systems. In this context, two main areas have been addressed. The first area is centered around techniques for the detection and characterization of anomalies occurring in a physical process for multi-channel sensor data acquired in real-time. These techniques generalize to the detection of anomalies in multivariate time-series data; secondly, new mathematical approaches, together with a new machine learning architecture, are presented that enable the embedding of a-priori knowledge about the system behavior. This ensures that the solutions obtained are consistent with the physics of the system being observed. This resolves the issue that purely data-driven systems can only make statements about the data; however, to make statements about a system from data, it is necessary to embed a-priori knowledge about the behavior of the system.

The second major issue addressed is combining the Rayleigh-Ritz method with machine learning. This required the development of new synthesis algorithms for admissible functions that ensure a hard fulfillment of constraints.

The Rayleigh-Ritz Autoencoder embeds the admissible functions in the decoder portion of an autoencoder enabling the hard fulfillment of generalized constraints. This removes the necessity for an additional regularizing term in the cost function. In turn, this permits the learning of the solutions to problems modeled via the calculus of variations, whereby the cost function corresponds directly to the functional that needs to be minimized.

Both parts are supported by a hybrid architecture for hyperparameter optimization based on a genetic algorithm. All the above concepts are presented as a collection of peer-reviewed publications and supported by code published in the form of toolboxes.

Contents

1	Introduction	8
1.1	Motivation	8
1.2	Scope	10
	Bibliography	12
2	Hybrid Machine Learning for Multivariate Time-Series	13
2.1	How It All Started	13
2.2	Properties of Machine Learning for Time-Series from Physical Systems	13
2.3	Encoder-Decoder-Based Architectures	14
2.3.1	Autoencoders for Anomaly Detection	14
2.3.2	Physics and Machine Learning	16
2.3.3	Machine Learning Architectures for Time-Series	16
2.4	Hybridizations of Neural Networks	18
	Bibliography	19
3	Contributions	26
	Bibliography	27
4	Anomaly Detection in Multivariate Time-Series	28
Hybrid Machine Learning for Anomaly Detection in Industrial Time-Series Measurement		
	Data	29
1	Introduction	30
2	Anomaly detection and machine learning.	32
3	Proposed machine learning architecture	33
4	Hybrid anomaly detection	34
5	KPI classifier	34
6	LSTM-VAE classifier	35
6.1	Preselection for training the LSTM-VAE	35
6.2	Training	36
6.3	Hyperparameter optimization (HPO)	37

7	Experiments and Results	38
7.1	Training and optimization	38
7.2	Case study quasi-validation	39
8	Conclusion	42
	Bibliography	43
Quality monitoring in vibro ground improvement – A hybrid machine learning approach		46
1	Introduction	47
2	Hybrid anomaly detection framework	49
2.1	Systematic data collection and KPI	49
2.2	AI for unsupervised anomaly detection	50
3	Results of the hybrid classifier	53
3.1	KDP	53
3.2	Extended data pre-processing	54
3.3	Impact of channels on reconstruction quality	54
4	Conclusions and further work	56
	Bibliography	56
Detecting Anomalous Multivariate Time-Series via Hybrid Machine Learning		61
1	Introduction	63
1.1	New Contributions	65
1.2	Organization of the Paper	65
2	Use-Case and Nature of the Data	66
3	Structured Handling of Data	68
4	Key Performance Indicators	69
4.1	Exemplary Technical KPI	69
4.2	Heat-Maps and Outlier Detection via KPI	70
5	ML-based Anomaly Detection	71
5.1	Objective Based Pre-Processing	73
5.2	Architecture Selection	76
5.3	KPI-Supervised Training	77
5.4	Length Preserving Training	78
5.5	Hyperparameter Optimization	79
5.6	Anomaly Detection Using Autoencoders	79
6	Architecture Comparison and Results	80
6.1	Hyperparameter Optimization Results	80
6.2	Architecture Comparison	81
6.3	Subprocesswise Anomaly Detection	82
6.4	Results KPI-Supervised Training	82
7	Conclusion	85

Bibliography	85
Application of hybrid machine learning based quality control in daily site management	93
1 Introduction	95
2 Hybrid Learning	95
3 Application of the Improved Quality Assurance Procedure	96
3.1 Case Studies	97
4 Outlook	99
Bibliography	100
Instrumentation and Signal Processing for the Verification of Directional Drilling	102
1 Introduction	104
2 Instrumentation	106
2.1 Calibration and characterization	107
3 Test Measurements	107
4 Data Handling and Preprocessing	108
5 Curve Reconstruction	109
6 Differential Curve Modeling	111
6.1 Bernstein basis	111
6.2 Constrained Bézier curve reconstruction from gradients	111
7 Conclusions	113
Bibliography	114
5 Scientific Machine Learning	116
A Rayleigh-Ritz Autoencoder	117
1 Introduction	119
1.1 Calculus of Variations and Constrained ODE	121
2 Neural networks and constrained differential equations	121
3 Rayleigh-Ritz Method	122
3.1 Synthesizing admissible functions	123
4 Rayleigh-Ritz Autoencoder	124
4.1 Cost function	125
5 Example application	125
6 Conclusions	128
Bibliography	128
7 Introduction	132
Bibliography	132

Automatic Synthesis of Admissible Functions		
for Variational Learning		137
1	Introduction	138
2	Calculus of variations	139
	2.1 Rayleigh Ritz method	140
3	Polynomials as admissible functions	140
	3.1 Definition of constraints	140
	3.2 A subspace approach	141
	3.3 Two step algorithm	141
4	Numerical testing	142
5	Application in Hybrid Machine Learning	143
	5.1 Brachistochrone: A forward problem	143
	5.2 Zernike polynomials: An inverse problem	145
6	Conclusions	145
	Bibliography	147
Extended Rayleigh-Ritz Autoencoder with Distribution-Free Statistics		151
1	Introduction	153
	1.1 Distance and similarity	155
2	The Rayleigh-Ritz Method	155
	2.1 The nature of constraints	156
3	The Rayleigh-Ritz Autoencoder	156
	3.1 Synthesizing admissible functions	156
	3.2 A new approach to uncertainty in machine learning	157
4	Exemplary application	158
	4.1 Perturbed sensor data	158
	4.2 Application specific constraints	160
	4.3 Neural network hyperparameters	160
	4.4 Network training and characterization	160
	4.5 Network training statistics	160
	4.6 RRAE in the monitoring phase	161
5	Constraints and Lebesgue Functions	162
6	Conclusions	163
	Bibliography	163

Chapter 1

Introduction

This thesis presents a set of methods on the topic of hybrid machine learning, with a particular focus on modeling real-time sensor data recorded in physical processes and incorporating physical knowledge into machine learning models. The methods are presented in a collection of peer-reviewed papers. The thesis consists mainly of two parts. The first part of the thesis focuses on outlier detection in a high-risk industrial process.

This was done using hybrid machine learning algorithms, which are a combination of key performance indicator computation and autoencoders. The second part deals with the embedding of physical constraints as hard constraints in machine learning architectures, in particular autoencoders, using the theory of admissible functions. Both parts are supported by the task of hyperparameter optimization, which is the task of finding the best settings for the machine learning algorithms with respect to some metric. The papers are grouped into two main chapters according to their nature:

1. Anomaly detection methods, mainly based on variations of autoencoders,
2. Rayleigh-Ritz Machine Learning, a new architecture for scientific machine learning.

In addition to the theory, a Matlab framework has been developed to support and ease the implementation of the above theory.

The results presented in the papers were obtained by providing implementations of the techniques and methods described in this thesis. In order to allow further research on this topic, the code has been made available to the public, packaged in two toolboxes, where the first one can be used to implement autoencoders and the second one provides the hyperparameter optimization using a genetic algorithm for the aforementioned models.

1.1 Motivation

The increased computing power available in recent years has made it possible to rediscover and re-invent techniques developed before the AI winter and almost forgotten in recent decades. The

problems that machine learning (ML) algorithms tackle today were solved analytically in the past. When the research proposal was presented in October 2022, physics-based machine learning was considered a so-called *innovation trigger*. This can be seen in the annual Gartner Hype Cycle (see Figure 1): physics-informed machine learning is still at the beginning of research and work in this area brings both - great opportunities for development and a high risk of failure. In 2023, physics-based machine learning was also mentioned in the hype cycle (see Figure 2), but the name was changed to *First Principal AI*. To get solutions which are aligned with the physics of the system, the idea is to incorporate known physical properties into machine learning architectures to provide a lower dimensional problem to the learning process and enforce compliance with physical laws. First principles AI is defined in the context of the hype cycle as a synonym for physics-informed

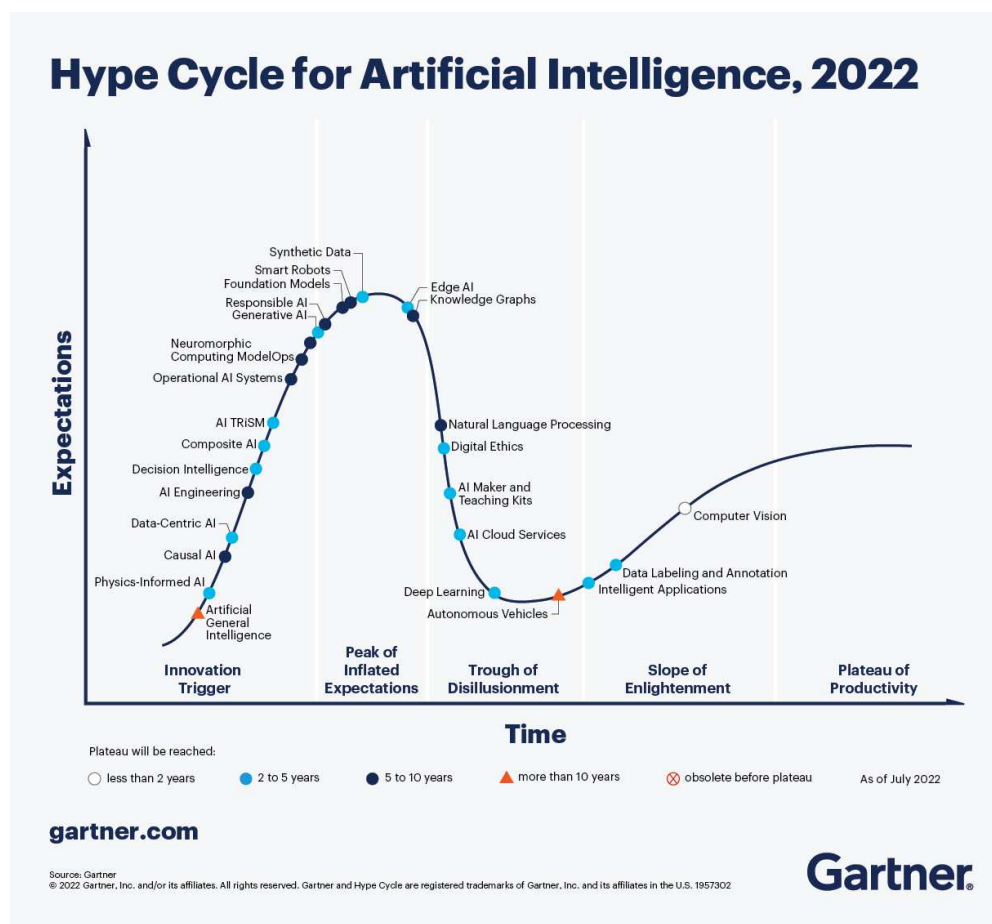


Figure 1: Hype Cycle for Artificial Intelligence, 2022 [1.1]

AI [1.2]. Not only the hybridized machine learning architectures need to be developed, but also classical machine learning architectures that can handle data from physical systems, which in many cases will be large, multivariate, with varying sampling lengths and frequencies, containing erroneous samples caused by errors in the measurement device or data processing.

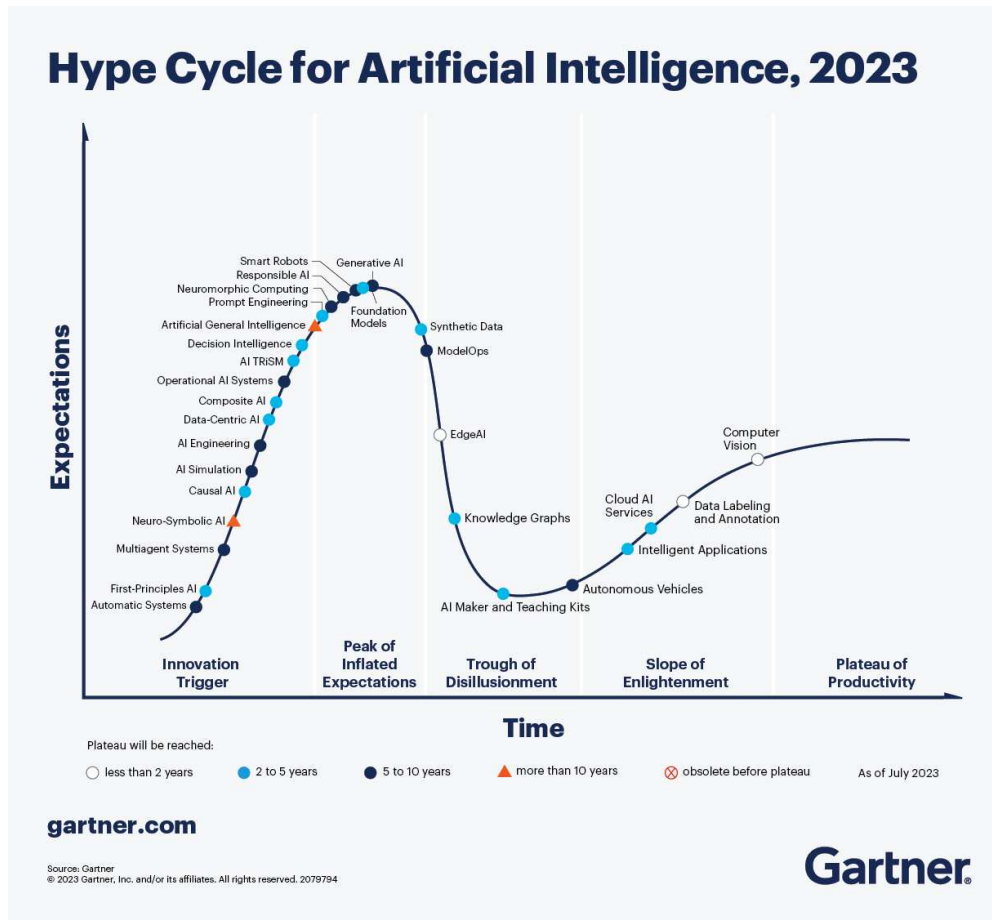


Figure 2: Hype Cycle for Artificial Intelligence, 2023 [1.2]

In addition, the machine learning architectures should, in the best case, be self-adapting to new environmental characteristics and have high reliability with low re-calibration effort. In many areas of machine learning, good performance with some performance fluctuations, e.g. translations between different languages, product recommendations in online shopping are tolerable, but in safety-relevant applications this can lead to severe repercussions.

1.2 Scope

This dissertation explores the question of applying machine learning to the analysis of real-time data recorded during the observation physical processes. It presents a collection of methods related to this topic of hybrid machine learning in the context of physical systems. The aim of all the methods presented is to model the data coming from physical systems more accurately and to incorporate a-priori physical knowledge into the hybrid machine learning. The methods are demonstrated

through a series of peer-reviewed papers and published code packaged in toolboxes.

The focus of the research is on hybrid machine learning for the analysis of possibly multivariate time-series from systems with strong physical relevance. The thesis relates to the combination of algebraic and statistical techniques with machine learning for time-series data. The scope of the thesis was limited to autoencoder-like models. There are different depths at which the proposals for hybrid machine learning are made:

1. A modified cost function of the autoencoder that more accurately reflects the characteristics of the system behavior that needs to be evaluated.
2. Anomaly detection in large amounts of real-time serial data collected during the execution of industrial production processes. In particular, the automatic decomposition of the multivariate time-series into segments corresponding to the sub-processes that, as a composite, represent the whole physical process to be monitored.
3. Combining constrained discrete orthogonal basis functions with ML techniques to implement a generalized discrete calculus of variations approach for boundary value problems (BVP) and inverse BVP.
4. Combining genetic algorithms for hyperparameter optimization with all the above mentioned ML techniques for machine learning suitable for multivariate data from physical systems.

With respect to the cost function and learning procedures for multivariate time-series, a new method has been developed that is suitable for training models on data with highly variable time series lengths. It reduces the amount of information added during resampling. The channels in multivariate time-series may have different levels of information. To address this issue, a new distribution-free regularization was proposed and tested.

In machine learning from real industrial processes, in most cases no ground truth is available. In these cases, unsupervised learning is the only option. New methods for training set construction based on additional a-priori knowledge of the process were investigated and compared to random training set construction. This was tested and evaluated in combination with hierarchical training methods that decompose the time series into segments and sub-processes and perform the anomaly detection for the decomposed time-series. The results are in the end reassembled, which leads to an additional level of redundancy.

The effect of hyperparameters and random seeds on the performance of machine learning models was shown and discussed in terms of robustness. A genetic algorithm was developed for the hyperparameter optimization of all the above machine learning architectures, which should lead to a better coverage of the search space due to the hybridization of two crossover functions. The hyperparameter optimization was used in all other hybrid architectures mentioned in this text.

Finally, a novel hybrid architecture for scientific machine learning was developed, which combines all the developments discussed above. It allows, based on admissible functions, the

fulfillment of a large class of linear constraints. It is called a Rayleigh-Ritz autoencoder because it is based on the Rayleigh-Ritz method of calculus of variations. It allows the implementation of the constraints and subsequently the admissible functions directly in the machine learning architecture. This architecture is presented together with the synthesis algorithms for the admissible functions used in the decoder and opens a new research field called *Rayleigh-Ritz Machine Learning*.

Bibliography

- [1.1] “What’s New in Artificial Intelligence from the 2022 Gartner Hype Cycle,” Sept. 2022, [Online; accessed 26. Sep. 2022]. [Online]. Available: <https://www.gartner.com/en/articles/what-s-new-in-artificial-intelligence-from-the-2022-gartner-hype-cycle>

- [1.2] “What’s New in Artificial Intelligence From the 2023 Gartner Hype Cycle™,” Oct. 2023, [Online; accessed 3. Oct. 2023]. [Online]. Available: <https://www.gartner.com/en/articles/what-s-new-in-artificial-intelligence-from-the-2023-gartner-hype-cycle>

Chapter 2

Hybrid Machine Learning for Multivariate Time-Series

This chapter presents a comprehensive survey of the literature available. Due to the diverse range of topics covered, the literature reviews pertaining to specific subtopics are incorporated within the relevant papers included in this dissertation.

2.1 How It All Started

The first computational model for neurons [2.1] was published in 1943 by McCulloch and Pitts and formed the basis for today's research on artificial neural networks.

As described in cite [2.2], there are three main problems in using neural networks:

1. the optimal feature set,
2. the architecture of the neural network,
3. the method for setting the optimal parameters of the neural network.

The automation of these tasks is also known as *Automated Machine Learning* [2.3].

This section aims to give an overview of the available literature on these subproblems. Since this topic covers a wide variety of approaches, this section will be limited to unsupervised and semi-supervised neural network approaches that are suitable for multivariate time-series data.

2.2 Properties of Machine Learning for Time-Series from Physical Systems

When applying machine learning techniques to data recorded during the observation of industrial processes, the time sequence of the recordings should also be maintained during the machine

learning process. Data recorded from physical systems have temporal dependencies and are in general multi-channel [2.4]. When recording real data from physical processes, the recordings are almost always noisy [2.4]. Moreover, in data recorded in industrial processes there is in many cases no possibility to establish ground truth and as consequence a lack of labels. A class of anomalies is not known before it is observed for the first time, using supervised learning techniques additional biases and assumptions are added to the system [2.5]. In semi-supervised approaches, the ML model is trained on a subset of the data that is assumed to contain a small number of unknown anomalies, and the known anomalies are excluded [2.5]. Therefore, unsupervised and possibly semi-supervised machine learning techniques are investigated.

Machine learning in physical systems must be reliable because the costs and consequences of false results are significant, especially in safety-relevant applications. For this reason, (parallel) hybrid models are used to reduce the risk of missing faulty elements and incorporating redundant mechanisms. Furthermore, anomalies are not known before being observed first; consequently, the hybrid models are combined with a knowledge discovery process to derive new detection mechanisms for anomalies not flagged by both parts.

2.3 Encoder-Decoder-Based Architectures

Autoencoders are the non-linear machine learning counterpart to principal component analysis [2.6], both techniques extract the features that capture most of the variance of the data by performing dimensionality reduction [2.7]. Autoencoders consist of two coupled parts: an encoder and a decoder. Each of these parts performs a mapping; the encoder maps the input to the ML architecture in most applications to a lower dimensional latent representation and the decoder tries to reconstruct the input signal as well as possible from this lower dimensional coding [2.8]. In general, the mapping performed by the encoder is the mapping from observation to parameter space, an inverse problem, and the mapping learned by the decoder, from parameter to observation space, can be described as a forward problem [2.9]. The loss function of an autoencoder is a functional to be minimized, most commonly based on the reconstruction error, which is the difference between the input passed to the encoder and the reconstructed representation obtained by the decoder [2.10]. Another member of the autoencoder class is the variational autoencoder (VAE) [2.11, 2.12], which differs from the classical autoencoder by its loss function, which is a regularized version of the reconstruction error and incorporates learning probabilities in the latent space, the space of lower dimensional encodings. For this reason, variational autoencoders are classified as generative models.

2.3.1 Autoencoders for Anomaly Detection

Anomaly detection using autoencoders relies on the assumption that anomalies/outliers contain non-representative features and therefore are not encoded in a lower dimensional space [2.13, 2.14]. In [2.15], a variation of VAEs is presented that detects outliers in the training dataset during training

and reduces their contribution to learning. Autoencoders were used in a supervised setting [2.16] as well as in an unsupervised setting for outlier detection. The disadvantage of training supervised autoencoders is that only known defects/anomalies are detected [2.16].

For time-series data; however, it may be beneficial to add layers with special properties to the networks; one possibility is to add long-short term memory (LSTM) layers [2.14, 2.17, 2.18] - yielding an LSTM autoencoder, convolutional neural network (CNN) layers [2.14] - yielding a CNN autoencoder, or mixtures of these to the encoder and/or decoder. However, CNNs have been shown to have the disadvantage of poorly capturing the spatial context of the data. To overcome this limitation of CNNs, capsule networks (CapsNet) [2.19] have been developed [2.20], primarily for image classification, and have also been hybridized to an LSTM CapsNet autoencoder [2.20]. The choice of layers is discussed in more detail in Section 2.3.3.

In the area of time series data, the datasets are small compared to vision applications or language learning; several hundred compared to several thousand available datasets. The general trend in machine learning architecture is towards deeper and deeper networks with an increasing number of learnable parameters. However, when using generative models, e.g. variational autoencoders and time-series, because of this comparatively small number of training samples, the following phenomena can occur: most of the information is stored in the decoder, this is known as *posterior collapse* [2.21] and is thought to be caused by the Kullback-Leibler divergence used in the regularization term of the cost function of VAEs. Because of this drawback of using the Kullback-Leibler (KL) divergence in latent space and the fact that it is not a symmetric measure, other measures of distribution similarity have been investigated; the distribution-free similarity index can also be used from a Bayesian perspective [2.22], or the Wasserstein distance in so-called Wasserstein autoencoders [2.23], or using the β divergence [2.24], which is a regularized version of the KL divergence, this type of VAE is also known as β -VAE [2.25].

Assumptions about the learned data are also made when variational autoencoders are used: an a-priori distribution of the learned data must be assumed. To simplify the computations, a Gaussian distribution is often chosen; the approach presented in [2.26] relaxes this assumption, although this approach requires labeled data to be available.

Most industrial processes rely on inferring information from the available data, since in most cases there is no ground truth. Almost all physical systems are governed by differential equations and are subject to constraints. For example, a bending beam that is clamped at one end and has no bending moment at the other end. The solution obtained by system identification must exactly satisfy these constraints, otherwise it is not a valid solution. In classical analysis, the mapping performed by an encoder can be seen as an inverse problem and the second mapping performed by the decoder as a forward problem. Because of this analogy, encoder-decoder structures, especially autoencoders, have been successfully used to solve forward and inverse problems [2.9].

2.3.2 Physics and Machine Learning

Classical neural networks do not employ a-priori knowledge of the physics of the system. Physics-informed neural networks [2.27] are a hybridization of classical neural networks and mathematical models that enforce physical laws. As these networks are trained, available information about the physics of the network is incorporated into the learning process. However, the disadvantage of purely data-driven models is that they may fit the observation very well, but the predictions may still contain physical inconsistencies and produce implausible solutions. In most of the physics-based approaches, presented so far, the physics are inferred from the observations [2.27]. When using autoencoder-like architectures the goal is that the latent space is the parameter space; the encoder learns the parameters of the equation governing the physical system. However, the work presented in [2.9, 2.28] does not enforce the fulfillment of constraints, which is a crucial property in modeling physical systems, since if these constraints are not fulfilled, the solution obtained is not valid. Solving inverse problems brings additional difficulties since the solutions of inverse problems are not guaranteed to be unique and stable [2.29].

In [2.30] an overview is given of the known techniques for solving differential equations with neural networks, most of the techniques presented are based on multilayer perceptrons (MLP) and radial basis functions (RBF). In one of the approaches that combines both MLP and RBF, the authors conclude that their method could solve the differential equation in a way that exactly satisfies the boundary condition, hard enforcement of constraints, but it is computationally too expensive. With this limitation, they choose to solve the system of equations approximately and exclude solutions that don't fulfill the conditions [2.30]. Other approaches to solving the constraint problems are also presented, using genetic algorithms, the Nelder-Mead simplex method and many others. The paper published in [2.31] lists three ways of solving partial differential equations with neural networks, one of which is physics-informed neural networks, but constrained systems are not treated separately in this paper. [2.32] gives an overview of recent developments in physics-based neural networks. In most of the recent literature, e.g. [2.33, 2.34], the physical constraints are incorporated into the loss function of the neural networks as a penalty term, which is also referred to as *soft boundary constraint enforcement* in the literature on physics-informed neural networks. The other approach is known as *hard enforcement* and the physical constraints are encoded into the network architecture [2.32].

More recently, the field of *Scientific Machine Learning* [2.35] has emerged. In contrast to physics-based neural networks, this term covers a broader set of concepts related to scientific applications. Examples of scientific machine learning dealing with solving differential equations and satisfying constraints can be found in [2.36, 2.37].

2.3.3 Machine Learning Architectures for Time-Series

This section discusses the architectural choice of including ML layers with special properties and their suitability for handling the properties of time-series data. For neural networks, the *universal*

approximation theorem [2.38] holds: given appropriate weights, neural networks can approximate any regular function with arbitrary accuracy; in theory with one layer, although this layer may have an infeasible large number of neurons [2.38], hence more than one layer is often used in a neural network architecture.

When dealing with time-series, the sequence of the data must be preserved. A common choice for dealing with sequential data are *Recurrent Neural Networks* (RNN), but when RNN are applied to long data sequences, such as those found in industrial processes, they face numerical difficulties, namely exploding and vanishing gradients. As a solution to this problem of gradient computation, another type of recurrent architecture; Long Short-Term Memory Layer (LSTM) [2.39], has been introduced. LSTMs have been successfully used to predict time series in [2.14, 2.40–2.42]. However, LSTMs are complex models with a large number of hyperparameters, and it has been shown that they have an impact on performance [2.43] and need to be optimized simultaneously due to dependencies between hyperparameters [2.44–2.46]. Indeed, [2.47] suggests that the performance of generative models, such as variational autoencoders or generative adversarial networks, is determined by the choice of hyperparameters. LSTMs are a common choice for inclusion in generative networks (see section 2.3.1). Another approach to solving the vanishing and exploding gradient problem is to partition the hidden layer into shorter models with different temporal granularities [2.48] and also models with a reduced number of connections between layers and neurons, known as skip connections [2.49].

In natural language processing, transformers are considered to have reached state-of-the-art performance [2.50, 2.51]. These attention-based architectures [2.52, 2.53] have emerged in ML for time-series analysis in recent years [2.54, 2.55], although pre-training can be unsupervised, most approaches require supervised fine-tuning [2.53]. Classical transformers are considered be recurrent networks and therefore suffer from the same limitations as feed-forward networks, i.e. they can only access representations from downstream layers but not from upstream layers. This limitation has been addressed e.g. by [2.56]. It has been shown that recurrent networks, in particular LSTMs, are better suited for modeling hierarchical relationships [2.57, 2.58] and have decreasing performance when dealing with long sequences [2.58]. In addition, the literature suggests that transformers have difficulty generalizing to input of lengths not seen during training, which can be a problem when dealing with time-series with widely varying time-series lengths [2.51]. Transformer-based architectures claim to have shorter training times than autoencoders with recurrent layers, e.g. LSTM, because they rely on transfer learning in most cases and are only fine-tuned to the desired problem [2.59].

Instead of using recurrent neural networks, some approaches use convolutional layers, known from machine vision applications for their ability to extract features, have also been applied to time-series problems [2.14, 2.60]. Alternative pre-trained architectures have also been presented in the recent past [2.61]. There are also approaches that combine both convolutional and recurrent features in one architecture, e.g. ConvLSTM [2.4]. According to [2.4], clustering methods, SVM and density estimation methods may not perform well due to their inability to capture temporal dependencies

in the data; however, these methods combined in a hybrid machine learning architecture may be beneficial.

2.4 Hybridizations of Neural Networks

Hybrid machine learning is defined as systems that combine at least two different techniques, one of which is a machine learning model. There are several ways to hybridize machine learning; some of the most important techniques for this dissertation project are described in the following section.

One of the most popular hybridization in time-series analysis is the combination of classical techniques, i.e. Autoregressive Integrated Moving Average (ARIMA), with machine learning models. In [2.62] the combination of ARIMA to model the linear part and support vector machines for the non-linear part is shown.

In the M4 competition [2.63] in 2020, a competition on time-series forecasting comparing classical and machine learning methods with pure statistical and pure ML approaches, it was shown that hybrid models outperformed the other competing approaches. Combining at least two methods, statistical and/or ML, also increased numerical accuracy [2.63]. The authors point out that this is consistent with previous M-competitions; it is believed that no single method can adequately capture the patterns in time series data [2.63]. The subsequent M-competition, M5, was the first in the series of M-competitions where all the top performing methods were based on machine learning approaches [2.64]. However, all such comparisons of methods need to be taken with a pinch of salt. According to [2.65], most of the benchmark datasets available for time-series anomaly detection may be unreliable. Also, many papers published on time-series ML use private data and do not show a sample plot of the data analyses [2.65].

Hyperparameter Optimization

Another way to hybridise machine learning models with another type of algorithm is to use some kind of optimization technique to determine the architecture and initial settings [2.2]. This is known as hyperparameter optimization or hyperparameter tuning and can be done, for example, using meta-heuristics, genetic algorithms [2.41, 2.66–2.69] or particle swarm optimization [2.70]. Hyperparameter optimization is particularly important for adapting systems to changing data properties as they occur in practical problems. It not only makes systems adaptable, but also allows fair comparison of different architectures and improves their performance in general [2.71].

Hyperparameters are parameters of machine learning models that determine the architecture and tuning of the optimization algorithm used for learning. In many applications, hyperparameters are set manually by trial and error or with an approach that performs an exhaustive search, the grid search [2.72]. However, the number of required evaluations of architectures grows exponentially with the number of hyperparameters to be optimized. Additionally, it should be mentioned that machine learning does not provide a unique solution in most cases, as it is highly overdetermined

and the random initialization of the weights and biases adds another source of uncertainty to the system. Evaluating the hyperparameter setting by training a model may not be sufficient due to this performance fluctuations.

Another class of hyperparameter optimization methods are Bayesian approaches [2.72, 2.73], which are especially recommended for smaller hyperparameter search spaces [2.71]. Furthermore, not all hyperparameters contribute equally to the performance [2.46, 2.74]. In addition, it was found that hyperparameters have regions of stable performance and therefore meta-heuristics [2.75] may be fit for purpose [2.44].

Bibliography

- [2.1] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, 12 1943.
- [2.2] G. Huang and L. Wang, “Hybrid Neural Network Models for Hydrologic Time Series Forecasting Based on Genetic Algorithm,” in *2011 Fourth International Joint Conference on Computational Sciences and Optimization*. IEEE, Apr. 2011, pp. 1347–1350.
- [2.3] F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated Machine Learning : Methods, Systems, Challenges*. Cham, Switzerland: Springer Nature, 2019.
- [2.4] C. Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu, W. Cheng, J. Ni, B. Zong, H. Chen, and N. V. Chawla, “A Deep Neural Network for Unsupervised Anomaly Detection and Diagnosis in Multivariate Time Series Data,” *arXiv*, Nov. 2018.
- [2.5] N. Görnitz, M. Kloft, K. Rieck, and U. Brefeld, “Toward supervised anomaly detection,” *Journal of Artificial Intelligence Research*, vol. 46, pp. 235–262, 2013.
- [2.6] B. Dai, Y. Wang, J. Aston, G. Hua, and D. Wipf, “Hidden Talents of the Variational Autoencoder,” *arXiv*, June 2017.
- [2.7] S. J. Wetzel, “Unsupervised learning of phase transitions: From principal component analysis to variational autoencoders,” *Physical Review E*, vol. 96, no. 2, p. 022140, 2017.
- [2.8] P. Baldi, “Autoencoders, Unsupervised Learning, and Deep Architectures,” in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*. JMLR Workshop and Conference Proceedings, June 2012, pp. 37–49. [Online]. Available: <https://proceedings.mlr.press/v27/baldi12a.html>
- [2.9] H. Goh, S. Sherifdeen, and T. Bui-Thanh, “Solving Forward and Inverse Problems Using Autoencoders,” *arXiv*, Dec. 2019.

- [2.10] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” in *Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations*. Cambridge, MA, USA: MIT Press, Jan. 1986, pp. 318–362.
- [2.11] D. P. Kingma, M. Welling, *et al.*, “An introduction to variational autoencoders,” *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.
- [2.12] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.
- [2.13] C. C. Aggarwal, *Outlier Analysis*. Cham, Switzerland: Springer International Publishing, 2017. [Online]. Available: <https://link.springer.com/book/10.1007/978-3-319-47578-3>
- [2.14] T. Kieu, B. Yang, and C. S. Jensen, “Outlier Detection for Multidimensional Time Series Using Deep Neural Networks,” in *2018 19th IEEE International Conference on Mobile Data Management (MDM)*. IEEE, June 2018, pp. 125–134.
- [2.15] S. Eduardo, A. Nazabal, C. K. I. Williams, and C. Sutton, “Robust Variational Autoencoders for Outlier Detection and Repair of Mixed-Type Data,” in *International Conference on Artificial Intelligence and Statistics*. PMLR, June 2020, pp. 4056–4066. [Online]. Available: <https://proceedings.mlr.press/v108/eduardo20a.html>
- [2.16] Y. Kawachi, Y. Koizumi, and N. Harada, “Complementary Set Variational Autoencoder for Supervised Anomaly Detection,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, Apr. 2018, pp. 2366–2370.
- [2.17] D. Park, Y. Hoshi, and C. C. Kemp, “A Multimodal Anomaly Detector for Robot-Assisted Feeding Using an LSTM-Based Variational Autoencoder,” *IEEE Rob. Autom. Lett.*, vol. 3, no. 3, pp. 1544–1551, Feb. 2018.
- [2.18] S. Lin, R. Clark, R. Birke, S. Schönborn, N. Trigoni, and S. Roberts, “Anomaly Detection for Time Series Using VAE-LSTM Hybrid Model,” in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, May 2020, pp. 4322–4326.
- [2.19] S. Sabour, N. Frosst, and G. E. Hinton, “Dynamic Routing Between Capsules,” *Advances in Neural Information Processing Systems*, vol. 30, 2017. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/hash/2cad8fa47bbef282badbb8de5374b894-Abstract.html>
- [2.20] A. Elhalwagy and T. Kalganova, “Hybridization of Capsule and LSTM Networks for unsupervised anomaly detection on multivariate data,” *arXiv*, Feb. 2022.

- [2.21] J. Lucas, G. Tucker, R. Grosse, and M. Norouzi, “Understanding Posterior Collapse in Generative Latent Variable Models,” *OpenReview*, July 2022. [Online]. Available: <https://openreview.net/forum?id=r1xaVLUYuE>
- [2.22] M. Pastore and A. Calcagni, “Measuring Distribution Similarities Between Samples: A Distribution-Free Overlapping Index,” *Front. Psychol.*, vol. 10, 2019.
- [2.23] I. Tolstikhin, O. Bousquet, S. Gelly, and B. Schoelkopf, “Wasserstein auto-encoders,” in *International Conference on Learning Representations*, 2018. [Online]. Available: <https://openreview.net/forum?id=HkL7n1-0b>
- [2.24] H. Akrami, A. A. Joshi, J. Li, S. Aydoore, and R. M. Leahy, “A robust variational autoencoder using beta divergence,” *Knowledge-Based Systems*, vol. 238, p. 107886, Feb. 2022.
- [2.25] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, “beta-VAE: Learning basic visual concepts with a constrained variational framework,” in *International Conference on Learning Representations*, 2017. [Online]. Available: <https://openreview.net/forum?id=Sy2fzU9gl>
- [2.26] M. Barlaud and F. Guyard, “A Non-Parametric Supervised Autoencoder for discriminative and generative modeling,” in *ICASSP*, Toronto, Canada, 2022. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-02937643>
- [2.27] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, “Physics-informed machine learning,” *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.
- [2.28] P. Peng, S. Jalali, and X. Yuan, “Solving Inverse Problems via Auto-Encoders,” *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 312–323, Mar. 2020.
- [2.29] N. Levashova, A. Gorbachev, R. Argun, and D. Lukyanenko, “The Problem of the Non-Uniqueness of the Solution to the Inverse Problem of Recovering the Symmetric States of a Bistable Medium with Data on the Position of an Autowave Front,” *Symmetry*, vol. 13, no. 5, p. 860, May 2021.
- [2.30] N. Yadav, A. Yadav, and M. Kumar, *An Introduction to Neural Network Methods for Differential Equations*. Dordrecht, The Netherlands: Springer Netherlands, 2015. [Online]. Available: <https://link.springer.com/book/10.1007/978-94-017-9816-7>
- [2.31] J. Blechschmidt and O. G. Ernst, “Three ways to solve partial differential equations with neural networks — A review,” *GAMM-Mitteilungen.*, vol. 44, no. 2, p. e202100006, June 2021.

- [2.32] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, “Scientific machine learning through physics-informed neural networks: Where we are and what’s next,” *arXiv preprint arXiv:2201.05624*, 2022.
- [2.33] Y. Xiong, R. Zuo, Z. Luo, and X. Wang, “A Physically Constrained Variational Autoencoder for Geochemical Pattern Recognition,” *Math. Geosci.*, vol. 54, no. 4, pp. 783–806, May 2022.
- [2.34] X. Meng, Z. Li, D. Zhang, and G. E. Karniadakis, “PPINN: Parareal physics-informed neural network for time-dependent PDEs,” *Comput. Methods Appl. Mech. Eng.*, vol. 370, p. 113250, Oct. 2020.
- [2.35] N. Baker, F. Alexander, T. Bremer, A. Hagberg, Y. Kevrekidis, H. Najm, M. Parashar, A. Patra, J. Sethian, S. Wild, K. Willcox, and S. Lee, “Workshop report on basic research needs for scientific machine learning: Core technologies for artificial intelligence.” [Online]. Available: <https://www.osti.gov/biblio/1478744>
- [2.36] N. Saad, G. Gupta, S. Alizadeh, and D. M. Robinson, “Guiding continuous operator learning through physics-based boundary constraints,” in *ICLR 2023*, 2023. [Online]. Available: <https://www.amazon.science/publications/guiding-continuous-operator-learning-through-physics-based-boundary-constraints>
- [2.37] D. Hansen, D. M. Robinson, S. Alizadeh, G. Gupta, and M. Mahoney, “Learning physical models that can respect conservation laws,” in *ICML 2023*, 2023. [Online]. Available: <https://www.amazon.science/publications/learning-physical-models-that-can-respect-conservation-laws>
- [2.38] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Networks*, vol. 2, no. 5, pp. 359–366, Jan. 1989.
- [2.39] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 11 1997.
- [2.40] A. Sagheer and M. Kotb, “Time series forecasting of petroleum production using deep LSTM recurrent networks,” *Neurocomputing*, vol. 323, pp. 203–213, Jan. 2019.
- [2.41] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, “Optimal Deep Learning LSTM Model for Electric Load Forecasting using Feature Selection and Genetic Algorithm: Comparison with Machine Learning Approaches †,” *Energies*, vol. 11, no. 7, p. 1636, June 2018.
- [2.42] D. Brezak, T. Bacek, D. Majetic, J. Kasac, and B. Novakovic, “A comparison of feed-forward and recurrent neural networks in time series forecasting,” in *2012 IEEE Conference on Computational Intelligence for Financial Engineering & Economics (CIFEr)*. IEEE, Mar. 2012, pp. 1–6.

- [2.43] F. Hutter, H. Hoos, and K. Leyton-Brown, “An Efficient Approach for Assessing Hyperparameter Importance,” in *International Conference on Machine Learning*. PMLR, Jan. 2014, pp. 754–762. [Online]. Available: <https://proceedings.mlr.press/v32/hutter14.html>
- [2.44] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A search space odyssey,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 28, no. 10, Mar. 2015.
- [2.45] S. L. Smith, P.-J. Kindermans, C. Ying, and Q. V. Le, “Don’t decay the learning rate, increase the batch size,” *arXiv preprint arXiv:1711.00489*, 2017.
- [2.46] P. Probst, A.-L. Boulesteix, and B. Bischl, “Tunability: Importance of hyperparameters of machine learning algorithms,” *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 1934–1965, 2019.
- [2.47] V. Škvára, T. Pevný, and V. Šmídl, “Are generative deep models for novelty detection truly better?” *arXiv*, July 2018.
- [2.48] J. Koutník, K. Greff, F. Gomez, and J. Schmidhuber, “A clockwork rnn,” in *International Conference on Machine Learning*. PMLR, 2014, pp. 1863–1871.
- [2.49] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, “Modeling long-and short-term temporal patterns with deep neural networks,” in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 95–104.
- [2.50] T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” in *EMNLP*, 2015, pp. 1412–1421. [Online]. Available: <http://aclweb.org/anthology/D/D15/D15-1166.pdf>
- [2.51] J. Pérez, J. Marinković, and P. Barceló, “On the Turing Completeness of Modern Neural Network Architectures,” *arXiv*, Jan. 2019.
- [2.52] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>
- [2.53] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, “A Transformer-based Framework for Multivariate Time Series Representation Learning,” in *KDD ’21: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. New York, NY, USA: Association for Computing Machinery, Aug. 2021, pp. 2114–2124.

- [2.54] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan, “Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting,” in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: <https://proceedings.neurips.cc/paper/2019/file/6775a0635c302542da2c32aa19d86be0-Paper.pdf>
- [2.55] S.-Y. Shih, F.-K. Sun, and H.-y. Lee, “Temporal pattern attention for multivariate time series forecasting,” *Mach. Learn.*, vol. 108, no. 8, pp. 1421–1441, Sept. 2019.
- [2.56] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan, “Enhancing the Locality and Breaking the Memory Bottleneck of Transformer on Time Series Forecasting,” *arXiv*, June 2019.
- [2.57] K. Tran, A. Bisazza, and C. Monz, “The Importance of Being Recurrent for Modeling Hierarchical Structure,” *arXiv*, Mar. 2018.
- [2.58] M. Hahn, “Theoretical Limitations of Self-Attention in Neural Sequence Models,” *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 156–171, Jan. 2020.
- [2.59] H. Meng, Y. Zhang, Y. Li, and H. Zhao, “Spacecraft Anomaly Detection via Transformer Reconstruction Error,” in *Proceedings of the International Conference on Aerospace System Science and Engineering 2019*. Singapore: Springer, Mar. 2020, pp. 351–362.
- [2.60] M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed, “DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series,” *IEEE Access*, vol. PP, no. 99, p. 1, Dec. 2018.
- [2.61] P. Malhotra, V. Tv, L. Vig, P. Agarwal, and G. Shroff, “TimeNet: Pre-trained deep recurrent neural network for time series classification,” *arXiv*, June 2017.
- [2.62] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, “Querying and mining of time series data: experimental comparison of representations and distance measures,” *Proc. VLDB Endow.*, vol. 1, no. 2, pp. 1542–1552, Aug. 2008.
- [2.63] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The M4 Competition: 100,000 time series and 61 forecasting methods,” *International Journal of Forecasting*, vol. 36, no. 1, pp. 54–74, Jan. 2020.
- [2.64] ———, “M5 accuracy competition: Results, findings, and conclusions,” *International Journal of Forecasting*, Jan. 2022.
- [2.65] R. Wu and E. J. Keogh, “Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress,” *arXiv*, Sept. 2020.

- [2.66] N. Gorgolis, I. Hatzilygeroudis, Z. Istenes, and L.-G. Gyenne, “Hyperparameter Optimization of LSTM Network Models through Genetic Algorithm,” in *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*. IEEE, July 2019, pp. 1–4.
- [2.67] S. R. Young, D. C. Rose, T. P. Karnowski, S.-H. Lim, and R. M. Patton, “Optimizing deep learning hyper-parameters through an evolutionary algorithm,” in *MLHPC '15: Proceedings of the Workshop on Machine Learning in High-Performance Computing Environments*. New York, NY, USA: Association for Computing Machinery, Nov. 2015, pp. 1–5.
- [2.68] A. Fiszlelew, P. Britos, A. Ochoa, H. Merlino, E. Fernández, and R. García-Martínez, “Finding optimal neural network architecture using genetic algorithms,” *Advances in computer science and engineering research in computing science*, vol. 27, pp. 15–24, 2007.
- [2.69] M. Reif, F. Shafait, and A. Dengel, “Meta-learning for evolutionary parameter optimization of classifiers,” *Mach. Learn.*, vol. 87, no. 3, pp. 357–380, June 2012.
- [2.70] G. H. T. Ribeiro, P. S. G. de M. Neto, G. D. C. Cavalcanti, and I. R. Tsang, “Lag selection for time series forecasting using Particle Swarm Optimization,” in *The 2011 International Joint Conference on Neural Networks*. IEEE, July 2011, pp. 2437–2444.
- [2.71] L. Yang and A. Shami, “On hyperparameter optimization of machine learning algorithms: Theory and practice,” *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020.
- [2.72] N. Reimers and I. Gurevych, “Optimal Hyperparameters for Deep LSTM-Networks for Sequence Labeling Tasks,” *arXiv*, July 2017.
- [2.73] F. Hutter, J. Lücke, and L. Schmidt-Thieme, “Beyond Manual Tuning of Hyperparameters,” *Künstl. Intell.*, vol. 29, no. 4, pp. 329–337, Nov. 2015.
- [2.74] J. N. van Rijn and F. Hutter, “Hyperparameter Importance Across Datasets,” in *KDD '18: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. New York, NY, USA: Association for Computing Machinery, July 2018, pp. 2367–2376.
- [2.75] S. N. Sivanandam and S. N. Deepa, “Genetic Algorithms,” in *Introduction to Genetic Algorithms*. Berlin, Germany: Springer, 2008, pp. 15–37.

Chapter 3

Contributions

The main parts of the research carried out for this thesis have been presented in peer reviewed publications: eight papers and two toolboxes publicly available on code servers. As this interdisciplinary research bridges several scientific fields, the work has been interdisciplinary in nature and has been in cooperation with a wide variety of people. As is the nature with interdisciplinary research, it is difficult to quantify the significance of individual contributions to success and the publication of new results; particularly, when - as is here the case - there is a combination of both simultaneous and sequential contributions. Nevertheless, my main contributions to each publication can be summarized as follows:

1. [3.1]: Implementation of the autoencoder architecture and the genetic algorithm, responsible for the LSTM-VAE classifier, execution of the presented ML experiments and evaluation of the results, literature research, main responsibility for the coordination of the paper preparation and review process, presentation of the paper at the conference.
2. [3.2]: Mainly responsible for the preparation and conceptualization of the manuscript, coordination of the writing process, implementation of the ML experiments, implementation of the new preprocessing steps, data evaluation and visualization, carried out the main part of the literature review, coordination of changes in the review process, presentation of the paper at the conference.
3. [3.3]: Mainly responsible for drafting and conceptualizing the manuscript, implementing and conceptualizing the improvements to the ML-based architecture, performing the experiments presented and collecting and visualizing the data, coordinating the review process.
4. [3.4]: Implementation of the Rayleigh-Ritz autoencoder and embedding the a-priori constraints into the architecture, as well as performing the experiments presented in the form of a case study, performing and writing the literature review on physics-informed machine learning, responsible for the sections on the Rayleigh-Ritz autoencoder, coordinating changes throughout the writing and review process, presenting the paper at the conference.

5. [3.5]: Assisting with manuscript preparation and derivation, mainly on the topics of incorporating permissible functions into the machine learning architecture, proofreading, participating in rewrites after each stage of the review process.
6. [3.6]: Main person responsible for the ML part, performing the experiments for the ML parts, draft of the literature review for the ML part, assisting the first author with the preparation of the manuscript, providing all necessary data evaluation and interpretation, and preparing all necessary figures.
7. [3.7]: Main person responsible for coordinating the writing process, performing ML experiments, implementing and conceptualizing the new distribution-free loss function and training procedure, producing ML results and literature review on physics-informed ML, coordinating the submission process
8. [3.8]: Conceptualizing and writing the main part of the introduction, supporting activities throughout the paper, proofreading, contributing to the derivation and formalization of the concepts used, coordinating the submission.

In addition to the above publications, two toolboxes (see [3.9] and [3.10]) have been programmed by me and made available for further research. These two toolboxes have been used as a basis for the research described in the aforementioned publications and are also a result of the research work carried out for this thesis.

Bibliography

- [3.1] A. Terbuch, P. O’Leary, and P. Auer, “Hybrid Machine Learning for Anomaly Detection in Industrial Time-Series Measurement Data,” in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2022, pp. 1–6.
- [3.2] A. Terbuch, A. Zöhrer, V. Winter, P. O’Leary, N. Khalili-Motlagh-Kasmaei, and G. Steiner, “Quality Monitoring in Vibro Ground Improvement — A Hybrid Machine Learning Approach,” *Geomechanics and Tunneling*, vol. 15, no. 5, pp. 658–664, Oct. 2022.
- [3.3] A. Terbuch, P. O’Leary, N. Khalili-Motlagh-Kasmaei, P. Auer, A. Zöhrer, and V. Winter, “Detecting Anomalous Multivariate Time-Series via Hybrid Machine Learning,” *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–11, Jan. 2023.
- [3.4] A. Terbuch, P. O’Leary, D. Ninevski, E. J. Hagendorfer, E. Schlager, A. Windisch, and C. Schweimer, “A Rayleigh-Ritz Autoencoder,” in *2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, Kuala Lumpur, Malaysia, May 2023.

- [3.5] P. O’Leary, D. Ninevski, and A. Terbuch, “Automatic Synthesis of Admissible Functions for Variational Learning,” in *2023 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, Gammarth, Tunisia, June 2023.
- [3.6] A. Zöhrer, V. Winter, A. Terbuch, P. O’Leary, and N. Khalili-Motlagh-Kasmaei, “Application of Hybrid Machine Learning Based Quality Control in Daily Site Management,” in *15th ISRM Congress 2023 & 72nd Geomechanics Colloquium*, W. Schubert and A. Kluckner, Eds., Salzburg, Austria, October 9-14 2023, pp. 569–574.
- [3.7] A. Terbuch, D. Ninevski, P. O’Leary, M. Harker, and M. Mücke, “Extended Rayleigh-Ritz Autoencoder with Distribution-Free Statistics,” under review.
- [3.8] P. O’Leary, A. Terbuch, D. Ninevski, D. Mevec, R. Fruhmann, M. Negin Khalili-motlagh kasmaei, and M. Habacher, “Instrumentation and Signal Processing for the Verification of Directional Drilling,” under review.
- [3.9] A. Terbuch, “Generic Deep Autoencoder for Time-Series - File Exchange - MATLAB CentralFile Exchange - MATLAB Central,” Jan. 2023, [Online; accessed 30. Nov. 2023]. [Online]. Available: https://de.mathworks.com/matlabcentral/fileexchange/111110-generic-deep-autoencoder-for-time-series?s_tid=prof_contriblnk
- [3.10] —, “Autoencoders for Time-Series with Hyperparameter Tuning - File Exchange - MATLAB CentralFile Exchange - MATLAB Central,” Sept. 2023, [Online; accessed 30. Nov. 2023]. [Online]. Available: https://de.mathworks.com/matlabcentral/fileexchange/130944-autoencoders-for-time-series-with-hyperparameter-tuning?s_tid=prof_contriblnk

Chapter 4

Anomaly Detection in Multivariate Time-Series

This chapter addresses methods for detection of anomalies, based on machine learning, during analyzing multivariate time-series, as well as the additional challenges occurring when dealing with multivariate batch-data, in contrast with univariate streaming data. The research question posed on this topic were answered with the following papers:

1. Hybrid Machine Learning for Anomaly Detection in Industrial Time-Series Measurement Data,
2. Quality Monitoring in Vibro Ground Improvement - A Hybrid Machine Learning Approach,
3. Detecting Anomalous Multivariate Time-Series via Hybrid Machine Learning,
4. Application of Hybrid Machine Learning Based Quality Control in Daily Site Management,
5. Instrumentation and Signal Processing for Verification of Directional Drilling.

Hybrid Machine Learning for Anomaly Detection in Industrial Time-Series Measurement Data

Anika Terbuch¹, Paul O’Leary¹, Peter Auer²

¹ Chair of Automation

² Chair of Information Technology

University of Leoben

Leoben, Austria

{anika.terbuch, paul.oleary, peter.auer}@unileoben.ac.at

Originally appeared as:

A. Terbuch, P. O’Leary, and P. Auer, "Hybrid Machine Learning for Anomaly Detection in Industrial Time-Series Measurement Data," 2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Ottawa, ON, Canada, 2022, pp. 1-6, DOI:10.1109/I2MTC48687.2022.9806663

©2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works

Hybrid Machine Learning for Anomaly Detection in Industrial Time-Series Measurement Data

Anika Terbuch¹, Paul O’Leary¹, Peter Auer²

¹ Chair of Automation

² Chair of Information Technology

University of Leoben

Leoben, Austria

{anika.terbuch,paul.oleary,peter.auer}@unileoben.ac.at

DOI: 10.1109/I2MTC48687.2022.9806663

Abstract

This paper presents a parallel hybrid machine learning system for the identification of anomalies in large sets of multivariate time-series (MVTs) measurement data. The goal is to achieve a more reliable detection of anomalies in safety relevant applications. Key performance indicators (KPI) are used as a measure for predicted possible sources of error. Whereas, a long short-term memory (LSTM-VAE) variational autoencoder is used to model the system behavior; the variational portion ensures the statistical uncertainty of the data is taken into account during training of the network. Combined in a parallel hybrid manner this provides a more reliable anomaly detection. The proposed structure is validated with a case study relating to a ground improvement process for building foundations. The data consists of large sets of real-time multi-variate time-series sensor data, emanating from the instrumented drilling rig. The performance of the LSTM-VAE is optimized using a genetic algorithm to select the optimal values for the hyperparameters. The implemented framework will also support future research into hybrid learning systems applied to real-time machine data analysis.

Index Terms - Hybrid Learning, Outlier Detection, Time Series

1 Introduction

This paper addresses the issue of hybrid machine learning as a means of improving the detection of anomalies in multivariate time-series, emanating from sensors on heavy machinery. The particular

focus here is on the evaluation of real-time machine data in safety relevant applications. A parallel hybrid system is implemented, combining classically computed key performance indicators and a machine learning classifier, as two independent means of identifying anomalies in MVTs. The developed methods are generic in nature and applicable to time-series evaluation in general. Here a specific case-study is used as a demonstrator and to validate the proposed methods.

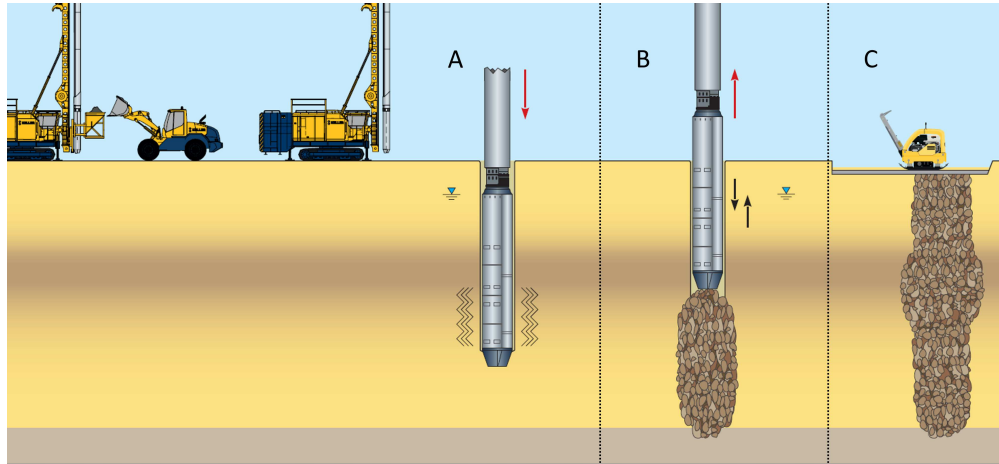


Figure 1: Schematic of the vibro-replacement ground improvement process. A) corresponds to the penetration process, B) to the compaction process and C) to a surface completion. The phases A and B are monitored in this work.

The vibro-replacement ground improvement process is used to create a large number of stable sub-surface columns for foundations of buildings. A schematic for the creation of a single column is shown in Figure 1. The process consists of two phases: a penetration phase during which the high power vibration head penetrates the ground; and the compaction phase where gravel is introduced concentrically and compacted in a vertical oscillating process.

The approach is to instrument the machine and analyze the multivariate sensor data in a manner that permits the automatic detection of a column that had anomalous characteristics. This is a safety relevant task, since the repercussions of having an instable foundation are both dangerous and very costly to rectify. This task is complicated by the fact that the subsurface conditions are non-uniform within a site and vary greatly from site to site. Furthermore, the process is manually performed by humans with different behavior and operation skills. Consequently, it is necessary to identify metrics for the process which are independent of a specific site characteristics or, are relative within a site. The machine has been instrumented with sensors for $n_s = 16$ channels, sampled at $t_s = 1$ s. A time-series of sensor data is created for each foundation-column. A site will typically have $k = 400 \dots 1000$ columns; whereby, the time required to create a column may vary from $t_c = 6 \dots 30$ min. Consequently, there are hundreds, if not thousands, of MVTs, with strongly varying lengths, associated with each building site.

The main contributions of this paper are:

1. A hybrid machine learning architecture for the detection of anomalous data sets in large numbers of multi-variate real-time machine data.
2. The identification and validation of long short-term memory variational autoencoder as a machine learning approach suitable for handling real time data with uncertainty in the temporal patterns.
3. The thorough analysis of the proposed method in a safety relevant case study. This includes the validation of the proposed methods on large volumes of real time machine data acquired in the field.

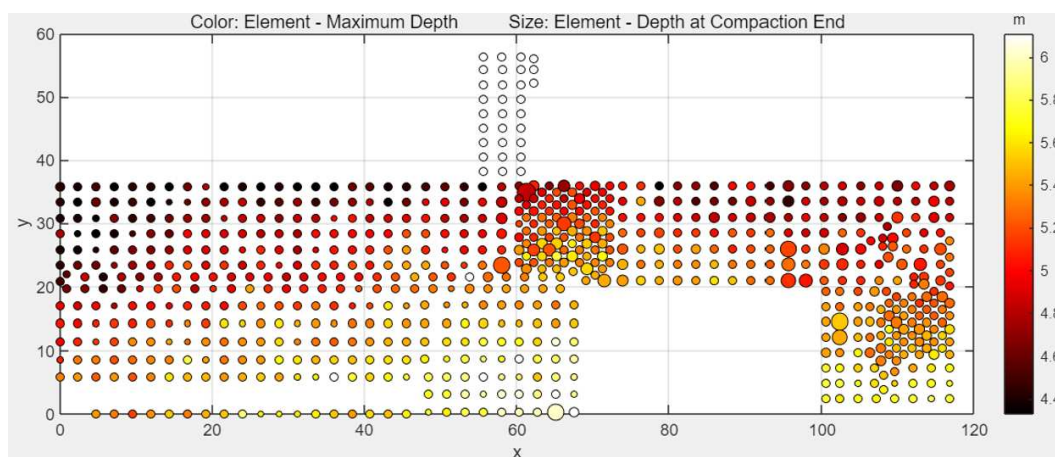


Figure 2: Example of the GPS referenced KPI for a specific building site. Each point corresponds to a foundation-column and has a MVTS associated with it. The color of the point is used to represent one KPI, while the radius of the point corresponds to a different KPI. Note: the systematic variation if the KPI from top to bottom and left to right.

2 Anomaly detection and machine learning.

The task of identifying anomalous foundation-columns, from the machine data, is currently performed manually. Manual inspection of hundreds of time-series is both error prone and costly. The goal now is to investigate a combination of key performance indicators with machine learning (ML) to automatically detect anomalous MVTS. KPI are used to determine systematic changes in the time-series data across the building site, e.g., see Fig. 2.

The machine learning classifier should be in the position to detect non-linear relationships between the sensor channels that are not considered by the KPI, but are relevant in the classification of the data. Furthermore, the ML-classifier may detect anomalies that were not addressed by the KPI directly, e.g., sources of errors that were not considered. Note: The ML-classifier is considered as augmenting the KPI not as a replacement, essentially a parallel hybrid system.

The aim is to have an ML-method that identifies anomalous time-series in an unsupervised manner, since this alleviates the necessity of labeling data. More importantly, it removes the assumption that labeling produced on one building site is valid when applied to another building site. It is necessary to verify the ML-classifier on multiple building sites.

3 Proposed machine learning architecture

There are a number of issues that need to be considered for the design of the ML-architecture:

1. The architecture must be suited to model MVTs.
2. The ML-architecture must have a mechanism to perform a regularization to deal with the uncertainty in the data.
3. Some means of optimizing the hyperparameters of the ML-method are required.
4. To define a method of classifying a data set as anomalous requires both a measure and thresholds.

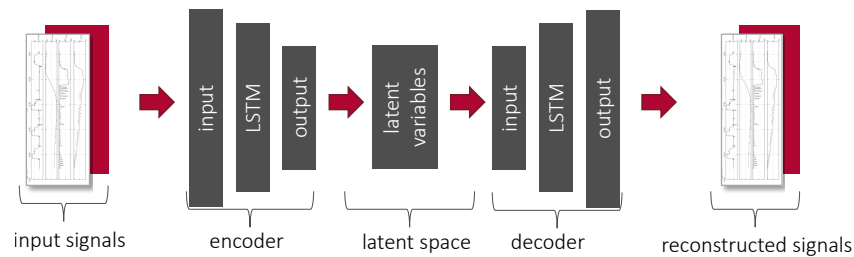


Figure 3: LSTM-VAE architecture

An autoencoder implements an unsupervised dimensionality reduction by identifying a compact representation to approximate the MVTs [4.1]. This ML-architecture consists of an encoder, which maps the input signals to a set of latent variables [4.2, 4.3] and a decoder which reconstructs an approximation. LSTM [4.4] has been selected for the encoder and decoder portions as a means of capturing the time dependencies within the time-series data, without being restricted to a specific convolution length [4.5]. A schematic of the LSTM-VAE is shown in Fig. 3.

A good representation of the data achieves a high dimensionality reduction [4.6]; that is, it requires only a low number of latent variables to create a good approximation of the data. The latent variables are akin to the coefficients of a classical model. In this analogy the encoder corresponds to solving the inverse problem and the decoder to the forward problem [4.7].

The variational autoencoder learns the joint distribution over all the latent variables and so aims to discover a generalization of the data generation process [4.2]. A common implementation of

variational autoencoders is to model the latent variables as Gaussian distributions described by their mean and standard deviation [4.8]. With this, the latent space can be considered to have a multidimensional probability density function (PDF).

Combining autoencoders, especially variational autoencoders, with LSTM layers (LSTM-VAE) is a quite recent evolution in machine learning. It has been implemented for outlier detection, see e.g. in [4.4,4.9]. Another variation of this technique using a LSTM-VAE with generative adversarial network is shown in [4.10].

An LSTM-VAE requires a set of hyperparameter values to enable training of the network; their values have a significant influence on performance; the exact mechanisms are still not fully understood [4.11]. As a result it is necessary to use some computational technique to determine an optimal set of hyperparameter values.

4 Hybrid anomaly detection

The hybrid classifier implemented here, uses both KPI and the LSTM-VAE to independently classify an MVTS as anomalous or not. Then the results of both classifications are combined. The results are combined as follows:

1. Any column that is classified as anomalous by either of the methods is considered to be anomalous. Augmenting the KPI classification with the ML-classifier leads to a more reliable detection of produced foundation-columns which are anomalous.
2. The MVTS associated with foundation-columns classified by the LSTM-VAE as anomalous, but not by the KPI, are used in an extended knowledge discovery process. Hereby, a process expert analyzes the data manually to determine as to why this classification has occurred. The goal is to establish better knowledge and understanding of the results obtained via the ML-classifier.

5 KPI classifier

Currently there are $n = 49$ different KPI defined for each foundation-column. Outlier detection is performed for each sensor channel in each KPI, based on its value being outside the upper $b_u = q_{75} + 1.5IQR$ and lower $b_l = q_{25} - 1.5IQR$ bounds; whereby q_{75}, q_{25} refer to the 75% and 25% quantiles and IQR to the interquantile range. A measure for the *outlierness*¹ of a specific MVTS is obtained by summing up the number of KPI for which it has been determined to be an outlier and normalizing by the number of KPI used, see Fig. 4.

¹This is an artificial word, it refers to a quantification as to what degree must a produced element be considered to be an outlier.

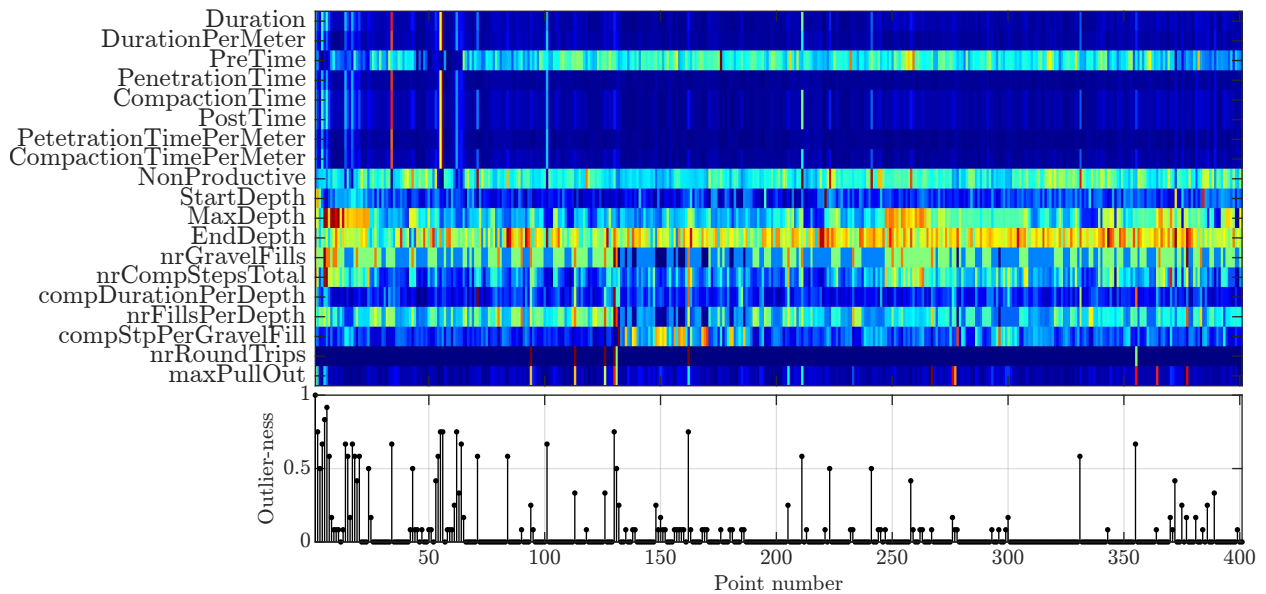


Figure 4: Evaluation of the KPI for a building site with more than $n = 400$ foundation columns. The point numbers are sorted according to execution time. Top: KPI represented as a heat-map. Bottom: degree of *outlierness*.

6 LSTM-VAE classifier

Variational autoencoders have been successfully used for outlier detection in time series, see e.g., [4.4, 4.9, 4.10, 4.12, 4.13]. The outliers are commonly detected in one of two ways:

1. In the latent space: The LSTM-VAE training yields a PDF for the latent space; see Fig. 5. The PDF of the latent space is mapped to the model for the MVTs data via the decoder. However, detection in the latent space tends to be more difficult, due to the high dimensionality reduction performed by the encoder [4.4].
2. Via the reconstruction error [4.14]: this is numerically more expensive, since both the encoder and decoder computations are required. However, it is a measure based on the quality of the reconstruction.

6.1 Preselection for training the LSTM-VAE

In the type of application being considered here, the sensor data is handled as a MVTs; consequently, for each produced foundation column there is an associated MVTs. There are many hundreds of these MVTs available for training for any one site, e.g., Fig. 4 shows data for a building site with $n > 400$ MVTs. Furthermore, the KPI computations are available prior to training the network. As defined in Section 5, a measure for the *outlierness* can be computed from the KPI. This is one of the very reasons for this piece of work: to determine how hybrid analytical and machine

learning methods can best be combined. The framework established here enables the investigation of different preselection strategies, with very little effort. Here we report on the results from the first selected strategy.

The preselection strategy selected here is, to use data which the KPI determine to be certainly non-anomalous. In this manner, the training process should identify the most compact PDF in the latent space that generalizes the description of non-anomalous data.

6.2 Training

Stochastic gradient decent (SGD) is the method used to perform the nonlinear minimization during training. The b MVTS, available for training, are grouped into mini-batches of size m using random sampling. The batch size is a hyperparameter of the network training, its value has an effect on the generalizing property of the SGD. The optimal selection of the value is addressed in Section 6.3.

If \mathbf{Y}_k is the matrix of the data, for the k^{th} sample in a batch, and $\hat{\mathbf{Y}}_k$ the corresponding network prediction, then the reconstruction cost function E_r is computed as²,

$$E_r = \sum_{k=1}^m \|\mathbf{Y}_k - \hat{\mathbf{Y}}_k\|_F^2. \quad (4.1)$$

The cost function E_t , minimized during training, is the *evidence lower bound*(ELBO) [4.2,4.15], it is the reconstruction cost function E_r regularized by the Kullback-Leibler divergence [4.2,4.16, 4.17] with respect to a normal distribution $N(0, 1)$. In this manner the uncertainty in the data is modeled by the joint probability density function (PDF) in the latent space. This regularization ensures optimal PDF for the given data. If $\boldsymbol{\mu}$ denotes the vector of the mean values of the latent variables and $\boldsymbol{\sigma}$ the vector of their corresponding standard deviations, then Kullback-Leibler divergence for all latent variables can be denoted as $D_{KL}\{N(0, 1) \| N(\boldsymbol{\mu}, \boldsymbol{\sigma})\}$

$$E_t = E_r - D_{KL}\{N(0, 1) \| N(\boldsymbol{\mu}, \boldsymbol{\sigma})\}. \quad (4.2)$$

The Kullback-Leibler regularization influences the coefficients of the network during training, with the aim of obtaining a latent space where the variables have equal variance. Consequently, given the values of the latent variables and their respective variances, the Mahalanobis distance corresponds to hyper-ellipsoids. However, with equal variance this becomes a hyper-sphere and the distances are isotropic with respect to direction, i.e., the variables are of equal importance. It is similar to covariance weighted averaging in the evaluation of measurement data [4.7].

The evidence lower bound is a lower bound on the log-likelihood of the data. The KL divergence is a quantification³ of the similarity of two statistical distributions and is a good approximation for the similarity between the true posterior distribution and the approximated posterior. With VAE it is assumed that the latent variables are mutually independent.

²Note $\|\mathbf{A}\|_F$ denotes the Frobenius norm of the matrix \mathbf{A} .

³The D_{KL} does not fulfill the properties required to formally be a measure; consequently, we do not denote it as measuring the similarity.

6.3 Hyperparameter optimization (HPO)

A popular approach to hyperparameter optimization is a technique referred to as grid search [4.11] which performs an exhaustive search on a fixed grid of candidate values. However, the dimensionality of this problem increases exponentially with the number of hyperparameters. Greff et al. showed in [4.18], that for LSTM structures, there are large regions of good performance for the hyperparameters. Consequently, exhaustive searches of the space are not required and meta-heuristics become applicable. Here a *genetic algorithm*⁴ has been selected as a meta-heuristic to optimize the hyperparameters [4.19, 4.20]. It efficiently identifies good regions in the search space [4.21, 4.22]. In comparison to grid search, genetic algorithms require fewer evaluations of the objective function [4.22] to attain similar performance.

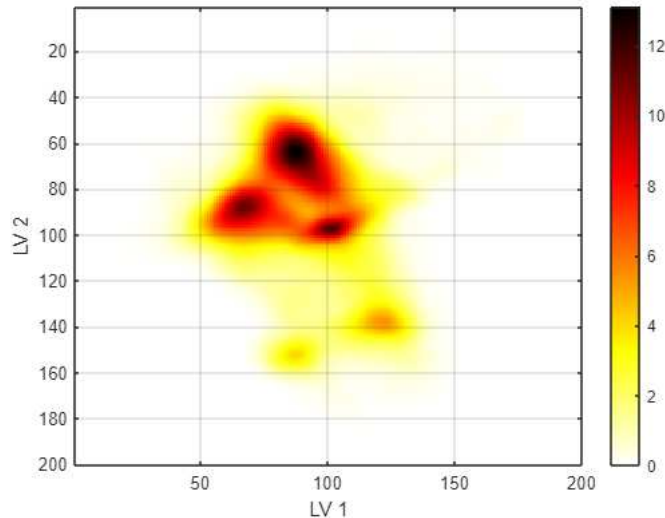


Figure 5: Example of the PDF of the latent space of the VAE-LSTM trained with the optimized hyperparameters.

The hyperparameters can be split into: training and architecture parameters. The learning rate is optimized simultaneously for the encoder and decoder. The quality of the training process is also influenced by the number of epochs p the model is trained [4.1]. The third training parameter is the mini-batch size m . Out of the set of architecture parameters the number of neurons of the LSTM layer of the encoder n_{ne} and decoder n_{nd} are also optimized. The constrained ranges for the training of these parameters are:

1. trained epochs: $p \in [1, 100]$;
2. batch size: $m \in [2, n]$;

⁴A detailed explication of the implementation of the genetic algorithm is beyond the scope of this paper, due to the limited space available here. Details of the implementation can be found in [4.19].

3. number of neurons encoder: $n_{ne} \in [1, 100]$;
4. number of neurons decoder: $n_{nd} \in [1, 100]$;
5. learning rate: $\alpha \in [0.00005, 0.01]$;
6. number of latent variables: $n_z = 2$.

The choice of these ranges is based on experience gained in previous research [4.23].

During optimization variations were investigated where additional LSTM layers could be conditionally implemented; however, the genetic algorithm found optimal solutions where these additional layers were not required. Consequently, they are not included in the final architecture.

7 Experiments and Results

The results presented here are based on data acquired at a building site where $k = 272$ foundation columns were produced; this is real production data.

The data is preprocessed prior to training the network: each time-series is resampled to have the same length, as required for the LSTM encoder. Resampling has been selected over padding due to the statistical distribution of the MVTs lengths, see Fig. 6. There are a few very long sequences and padding all others to this length would prove highly inefficient. This is followed by normalization, to create *unitless* channels with defined magnitudes.

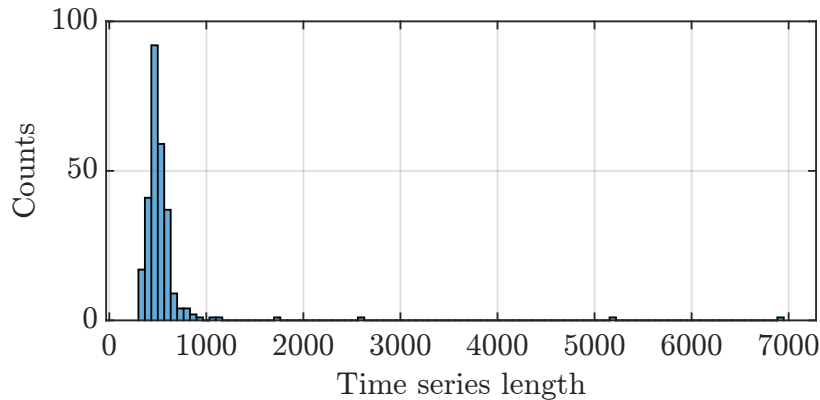


Figure 6: Example histogram of the time-series lengths associated with one building site.

7.1 Training and optimization

A subset of $n = 80$ non-anomalous data sets were selected for the training and optimization process. The evolution of the average performance of the LSTM-VAE as a function of generation in the

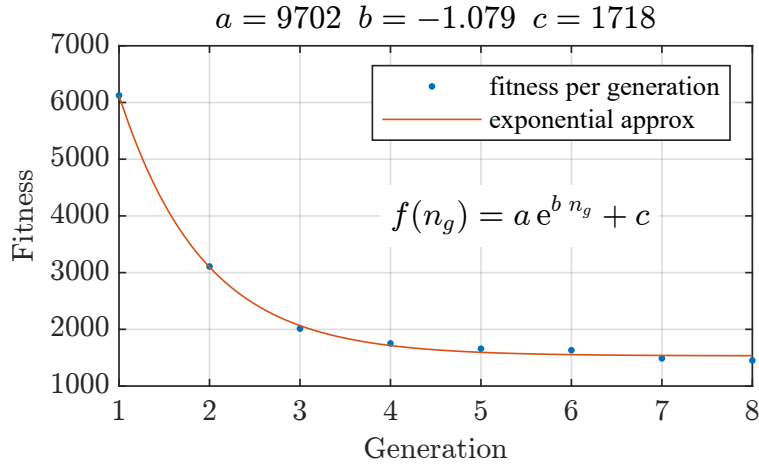


Figure 7: Average fitness of the LSTM-VAE as it evolves with generations in the genetic algorithm.

genetic algorithm is shown in Figure 7. The performance converges exponentially, as one would expect from a genetic algorithm [4.24].

In Figure 8 the spread of values for the hyperparameters p , n_{ne} , n_{nd} , α and m within each population as a function of generation is shown. It can be seen that, with the exception of α the learning rate, all the values converge to stable distributions. This behavior is consistent with the results reported in [4.18]. At this point we conclude that the learning and optimization process has converged and proceed now to evaluate selected data.

7.2 Case study quasi-validation

In this application there is fundamentally no ground truth⁵ available. A quasi-validation set was produced consisting of $n = 10$ anomalous and $n = 10$ non-anomalous. The 20 MVTs were classified to the best of human ability; four people were involved in this process, including an expert in the ground improvement process.

All time-series for the site we evaluated for their reconstruction error using the LSTM-VAE; this was performed for both the non-optimized and optimized hyperparameters. The goal is to determine if the hyperparameter optimization has improved the reliability of the classification.

The results of applying the LSTM-VAE to all the time-series available for the site are shown in Figure 9: On the left the results prior to hyperparameter optimization and on the right after optimization.

The reconstruction error from the LSTM-VAE is positive definite; as a result, it does not adhere to a standard Gaussian distribution. Given this uncertainty, skewness adjusted outlier detection is more appropriate [4.25–4.27]. Then the automatic and manual classifications are compared to see if there are erroneous classifications by the LSTM-VAE.

⁵For this reason we have not called this section validation.

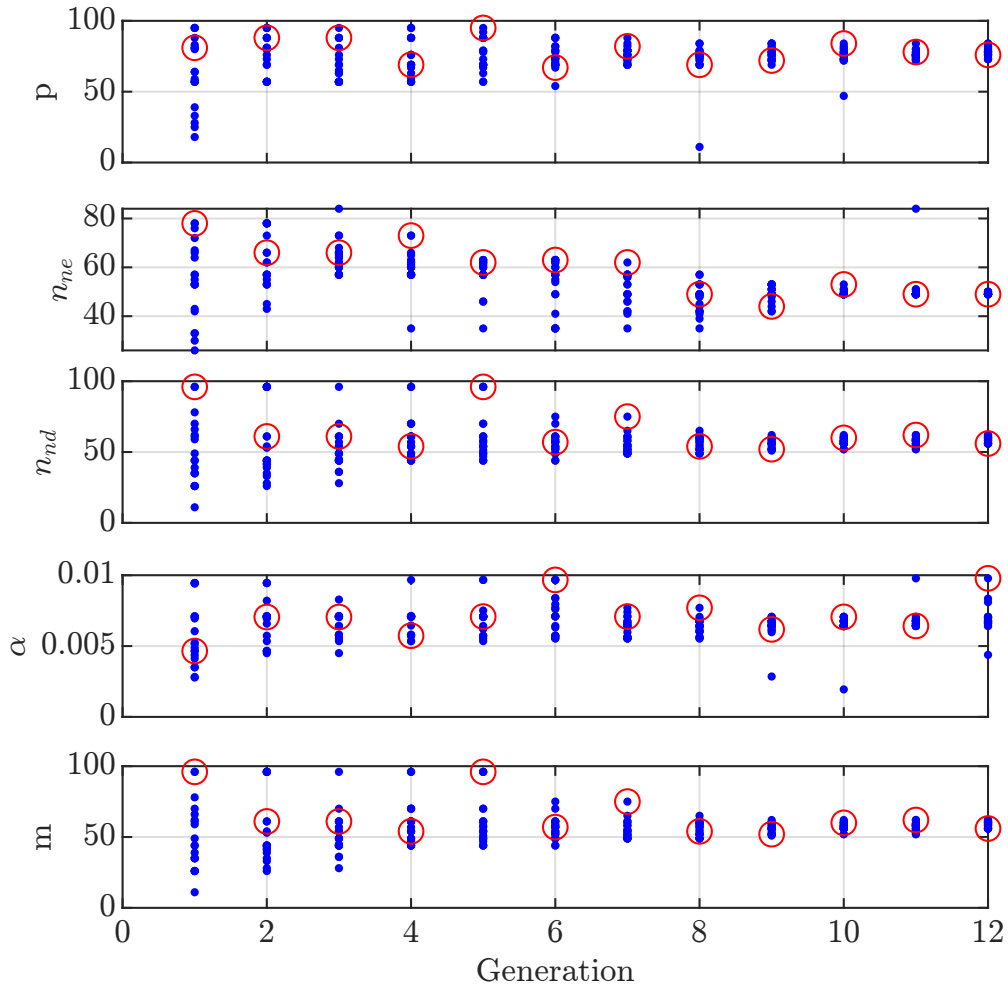


Figure 8: Hyperparameters p , n_{ne} , n_{nd} , α and m values as a function of generation. Blue: the value for each member of the population in each generation. Red-circled: the best performing member of the population at each generation. It can be seen that, with the exception of the learning rate α , all the values converge to stable values. Whereas α converges to a region, this is consistent with there being flat regions of performance with the hyperparameters [4.18].

The following observations can be made directly:

1. The reconstruction error over all the time-series for the site is significantly improved after hyperparameter optimization. This implies that the optimization is improving the representation of the data.
2. The variance and IQR of the reconstruction error are also significantly reduced through optimization. This is important since this will lead to tighter bounds for the detection of outliers after skewness adjustment.
3. After optimization, the automatic and manual classifications agree to 100%; there are no false

classification, all columns are correctly classified as anomalous or non-anomalous. This is not the case prior to optimization.

This directly concluded results are also congruent with the statistical quantities which are shown in Table 4.1. In Table 4.1 various measures related to the reconstruction error of a model trained with optimized parameters are compared to a model with non-optimized parameters. The statistical quantities were calculated for the manually labeled non-anomalous (normal) examples and anomalous examples (outliers).

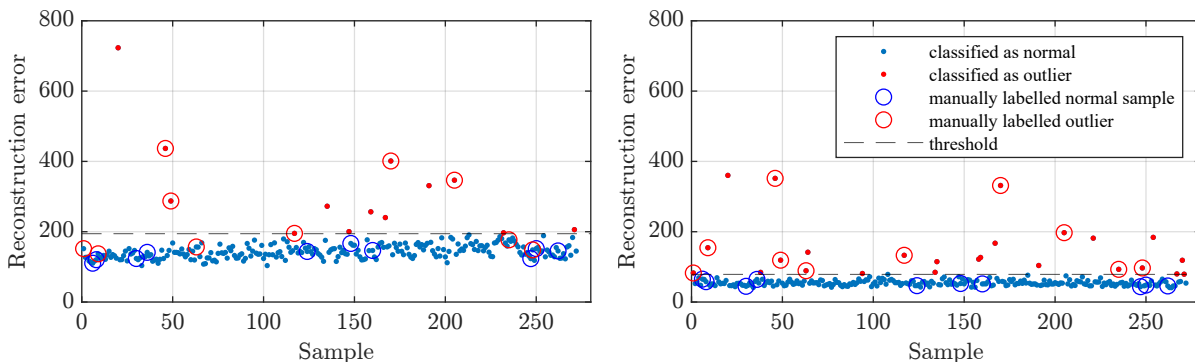


Figure 9: Results of applying the LSTM-VAE to all the time-series available for the site. Left: prior to hyperparameter optimization. Right: after optimization. The automatic classification as an outlier or not is performed using the skewness-adjusted boxplot bounds.

The uncertainty in the data can be seen in the bivariate histogram of the KPI-based *outlierness* and LSTM-VAE reconstruction error E_r , for each of the $n = 272$ time-series, is shown in Fig. 10. The two methods clearly identify a common set of MVTS, which have both low *outlierness* and low reconstruction error; this is the dominant peak in the bivariate histogram. Given the bivariate histogram thresholds can be computed based on the Mahalanobis distance and used to classify anomalous MVTS.

Table 4.1: Statistical comparison

statistical quantity	without HPO		with HPO	
	normal	outlier	normal	outlier
mean	137.38	243.80	52.04	164.85
median	142.58	186.12	49.88	125.97
σ	17.22	114.46	7.92	99.60
IQR	23.28	194.83	12.16	104.25
skewness	0.001	0.650	0.666	1.113
distance between means	106.43		112.82	
distance between medians	43.55		76.10	

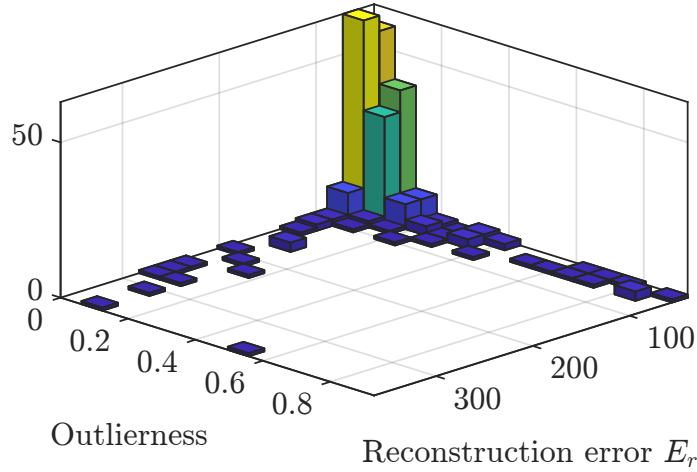


Figure 10: Bivariate histogram of the *Outlierness* and LSTM-VAE reconstruction error E_r , for each of the $n = 272$ time-series.

8 Conclusion

The proposed hybrid KPI - LSTM-VAE classifier for the multivariate time-series measurement data has functioned as proposed. The machine learning portion has identified anomalous data sets which were not detected by the KPI; i.e., there is anomalous behavior which was not detected by the KPI proposed by the experts. The hybrid statistical outlier detection and the variational nature of the encoder successfully modelled the uncertainty of the data.

It can be concluded that the application of the genetic algorithm to optimize the hyperparameters is significant; since after optimization there is a lower variance in the LSTM-VAE reconstruction error. This further permits the identification of tighter thresholds with the same confidence level as prior to optimization.

The framework implemented here has been applied to an extensive case study and provides a platform to perform further research on applying hybrid machine learning to multivariate measurement data. Issues for further research: the preselection methods applied before training; use of alternative loss functions in the autoencoder, e.g., test 1-norms rather than 2-norms and/or alternative measures for divergence. Furthermore, with this framework the preselection of MVTs to be used during training based on the KPI can be investigated; this would yield a serial hybrid system during training and a parallel hybrid system during classification.

Acknowledgment

We wish to thank Vincent Winter and Alexander Zöhrer from the company Keller Grund- und Tiefbau GmbH for their continuing support in this project and for making data available. Also our gratitude goes to Stefan Herdy who contributed to the original developments in this application area.

The authors gratefully acknowledge the financial support under the scope of the COMET program within the K2 Center “Integrated Computational Material, Process and Product Engineering (IC-MPPE)” (Project No 859480). This program is supported by the Austrian Federal Ministries for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK) and for Digital and Economic Affairs (BMDW), represented by the Austrian research funding association (FFG), and the federal states of Styria, Upper Austria and Tyrol.

Bibliography

- [4.1] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge and London: MIT Press, 2016.
- [4.2] D. P. Kingma and M. Welling., *An Introduction to Variational Autoencoders*. New York: Now Publishers, 2019.
- [4.3] W. H. Lopez Pinaya, S. Vieira, R. Garcia-Dias, and A. Mechelli, “Autoencoders,” in *Machine Learning: Methods and Applications to Brain Disorders*. Cambridge: Academic Press, 1 2020, pp. 193–208.
- [4.4] D. Park, Y. Hoshi, and C. C. Kemp, “A Multimodal Anomaly Detector for Robot-Assisted Feeding Using an LSTM-Based Variational Autoencoder,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1544–1551, 2 2018.
- [4.5] T. Chen, X. Liu, B. Xia, W. Wang, and Y. Lai, “Unsupervised anomaly detection of industrial robots using sliding-window convolutional variational autoencoder,” *IEEE Access*, vol. 8, pp. 47 072–47 081, 2020.
- [4.6] E. Alpaydin, *Introduction to Machine Learning*, 3rd ed., ser. Adaptive Computation and Machine Learning. Cambridge and London: MIT Press, 2014.
- [4.7] B. P. Bezruchko and D. A. Smirnov, *Extracting knowledge from time series: An introduction to nonlinear empirical modeling*. Springer Science & Business Media, 2010.
- [4.8] C. Doersch, “Tutorial on variational autoencoders,” 2016, [Online; accessed 21. Feb. 2021]. [Online]. Available: ”<https://arxiv.org/abs/1606.05908>”
- [4.9] S. Lin, R. Clark, R. Birke, S. Schönborn, N. Trigoni, and S. J. Roberts, “Anomaly detection for time series using VAE-LSTM hybrid model,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020*. Barcelona: IEEE, 5 2020, pp. 4322–4326.
- [4.10] Z. Niu, K. Yu, and X. Wu, “LSTM-based VAE-GAN for time-series anomaly detection,” *Sensors*, vol. 20, no. 13, 2020.

- [4.11] J. Wu, X.-Y. Chen, H. Zhang, L.-D. Xiong, H. Lei, and S.-H. Deng, “Hyperparameter optimization for machine learning models based on Bayesian optimization,” *Journal of Electronic Science and Technology*, vol. 17, no. 1, pp. 26–40, 2019.
- [4.12] T. Amarbayasgalan, V. H. Pham, N. Theera-Umpon, and K. H. Ryu, “Unsupervised Anomaly Detection Approach for Time-Series in Multi-Domains Using Deep Reconstruction Error,” *Symmetry*, vol. 12, no. 8, 08 2020.
- [4.13] F. Zhang and H. Fleyeh, “Anomaly detection of heat energy usage in district heating substations using LSTM based variational autoencoder combined with physical model,” in *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2020, pp. 153–158.
- [4.14] T. Kieu, B. Yang, and C. S. Jensen, “Outlier detection for multidimensional time series using deep neural networks,” in *2018 19th IEEE International Conference on Mobile Data Management (MDM)*. Aalborg: IEEE, 06 2018, pp. 125–134.
- [4.15] X. Hou, L. Shen, K. Sun, and G. Qiu, “Deep feature consistent variational autoencoder,” in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2017, pp. 1133–1141.
- [4.16] A. Alemi, B. Poole, I. Fischer, J. Dillon, R. A. Saurous, and K. Murphy, “Fixing a broken ELBO,” in *International Conference on Machine Learning*. PMLR, 2018, pp. 159–168.
- [4.17] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.
- [4.18] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A search space odyssey,” *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.
- [4.19] A. Terbuch, “LSTM hyperparameter optimization: Impact of the selection of hyperparameters on machine learning performance when applied to time series in physical systems,” Master’s thesis, Chair of Automation, Montanuniversitaet Leoben, 2021.
- [4.20] N. Gorgolis, I. Hatzilygeroudis, Z. Istenes, and L.-G. Gyenne, “Hyperparameter optimization of LSTM network models through genetic algorithm,” in *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*. IEEE, 7 2019, pp. 1–4.
- [4.21] S. N. Sivanandam and S. N. Deepa, *Introduction to Genetic Algorithms*. Berlin: Springer-Verlag, 2008.
- [4.22] M. Reif, F. Shafait, and A. Dengel, “Meta-learning for evolutionary parameter optimization of classifiers,” *Mach. Learn.*, vol. 87, no. 3, pp. 357–380, 6 2012.

- [4.23] S. Herdy, “Machine learning in the context of time series,” Master’s thesis, Chair of Automation, Montanuniversitaet Leoben, 2020.
- [4.24] L. Altenberg, “The schema theorem and price’s theorem,” in *Foundations of genetic algorithms*. Elsevier, 1995, vol. 3, pp. 23–49.
- [4.25] M. Hubert and S. Van der Veecken, “Outlier detection for skewed data,” *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 22, no. 3-4, pp. 235–246, 2008.
- [4.26] M. Hubert and E. Vandervieren, “An adjusted boxplot for skewed distributions,” *Computational Statistics & Data Analysis*, vol. 52, no. 12, pp. 5186–5201, 2008.
- [4.27] E. Vandervieren and M. Hubert, “An adjusted boxplot for skewed distributions,” in *COMPSTAT 2004: Proceedings in Computational Statistics*. Prague: Springer-Verlag, 2004, pp. 1933–1940.

Quality Monitoring in Vibro Ground Improvement – A Hybrid Machine Learning Approach

Anika Terbuch¹, Alexander Zöhrer², Vincent Winter²,
Paul O’Leary¹, Negin Khalili-Motlagh-Kasmaei¹, Gernot Steiner¹

¹ Chair of Automation, Montanuniversität Leoben, Leoben, Austria
{anika.terbuch, paul.oleary, negin.khalilimotlaghkasmaei}@unileoben.ac.at
g.steiner@stud.unileoben.ac.at

² Keller Grundbau GesmbH, Vienna and Söding, Austria
{alexander.zoehrer, vincent.winter}@keller.com

Originally appeared as:

A. Terbuch, A. Zöhrer, V. Winter, P. O’Leary, N. Khalili-Motlagh-Kasmaei, G. Steiner, "Quality Monitoring in Vibro Ground Improvement – A Hybrid Machine Learning Approach," *Geomechanics and Tunnelling*, vol. 15, no. 5, pp. 658–664, 2022. DOI: <https://doi.org/10.1002/geot.202200028>

This is the peer-reviewed version of the following article: [4.1], which has been published in final form at [Link to final article using the <https://doi.org/10.1002/geot.202200028>]. This article may be used for non-commercial purposes in accordance with Wiley Terms and Conditions for Use of Self-Archived Versions. This article may not be enhanced, enriched, or otherwise transformed into a derivative work, without express permission from Wiley or by statutory rights under applicable legislation. Copyright notices must not be removed, obscured or modified. The article must be linked to Wiley’s version of record on Wiley Online Library and any embedding, framing or otherwise making available the article or pages thereof by third parties from platforms, services and websites other than Wiley Online Library must be prohibited.

Quality monitoring in vibro ground improvement – A hybrid machine learning approach

Qualitätsüberwachung bei der Bodenverbesserung mittels Tiefenrüttelverfahren –
Ein Ansatz des hybriden maschinellen Lernens

Anika Terbuch ¹, Alexander Zöhrer ², Vincent Winter ²,
Paul O’Leary ¹, Negin Khalili-Motlagh-Kasmaei ¹, Gernot Steiner¹

¹Chair of Automation,
Montanuniversität Leoben,
Leoben, Austria

{anika.terbuch, paul.oleary, negin.khalilimotlaghkasmaei}@unileoben.ac.at
g.steiner@stud.unileoben.ac.at

²Keller Grundbau GesmbH
Vienna and Söding, Austria
{alexander.zoehrer, vincent.winter}@keller.com

DOI: 10.1002/geot.202200028

Abstract

This article presents a new approach of quality control to vibro ground improvement techniques based on hybrid machine learning (ML), i.e., a combination of classical analysis and ML techniques. The process is monitored with an instrumented rig equipped with multiple sensors. Key performance indicators (KPIs) are used to identify anomalous foundation columns. As the foundation columns are sub-surface, there is no direct access to ground truth; consequently, unsupervised ML is applied to the recorded time-series data. The risk of not detecting defective elements is reduced by the combination of two independent methods for anomaly detection, KPI- and ML-based classification. The ML is used to gain a deeper process understanding

and to detect anomalies which were not considered in the design phase of the KPI. New pre-processing techniques were derived from the insights gained from the ML classifier; this led to a more robust classifier. It is shown how unsupervised ML, using a multi-channel variational autoencoder (VAE) with long short-term memory (LSTM) layers, can be utilized in a knowledge discovery process (KDP).

Keywords: vibro ground improvement; quality monitoring; hybrid learning; outlier detection; time-series; KPI analysis

Zusammenfassung

In diesem Fachbeitrag wird ein neuer Ansatz für die Qualitätskontrolle der Tiefenrüttelverfahren mittels hybriden maschinellen Lernens vorgestellt. Als hybrides maschinelles Lernen wird die Kombination von klassischen analytischen Methoden mit Methoden des maschinellen Lernens beschrieben. Die Prozessüberwachung wird basierend auf einer instrumentierten Rütteltragraupe durchgeführt. Durch Berechnung von Leistungskennzahlen werden Elemente mit abweichenden Qualitätsmerkmalen identifiziert. Unüberwachtes maschinelles Lernen wird auf die aufgenommenen Zeitreihen angewandt, da keine Grundwahrheit für jedes Element ermittelt werden kann. Durch die Kombination zweier unabhängiger Methoden, maschinelles Lernen und Klassifizierung mittels Leistungskennzahlen, wird das Risiko ein fehlerhaftes Element nicht zu erkennen, minimiert. Durch die Klassifizierung basierend auf maschinellem Lernen wird ein tieferes Prozessverständnis erlangt und es werden dadurch Anomalien erkannt, welche bei der Definition von Leistungskennzahlen nicht mit einbezogen wurden. Dadurch konnten neue Datenvorverarbeitungsmethoden abgeleitet werden, welche zu einer robusteren Klassifizierung führen. Es wird gezeigt wie unüberwachtes maschinelles Lernen mittels eines "Variational Autoencoders" mit mehreren Kanälen und "Long-Short-Term Memory" Layern in einem Wissensentdeckungsprozess verwendet werden kann.

Stichworte: Tiefenrüttelverfahren; Qualitätsüberwachung; Hybrides maschinelles Lernen; Anomaliedetektion; Zeitreihenanalyse; Leistungskennzahlenanalyse

1 Introduction

Vibro ground improvement techniques are widely used to support buildings and infrastructure works. This process was developed by the Johann Keller GmbH in the 1930s. Although this technique has been used for almost 100 years, there is still no standardized, reliable method for performing quality control [4.2]. Increasing economic pressure and a higher risk awareness—if a building foundation fails, the value of damage may easily exceed the value of the foundation works—force this traditional technique to change and upgrade the quality control processes significantly [4.3].

As the produced elements are sub-surface, there is no direct possibility for inspection after execution. Consequently, the quality control is based on indirect methods. Starting in the 1990s, some key parameters from the machine data were monitored as a means of quality control, e.g.,

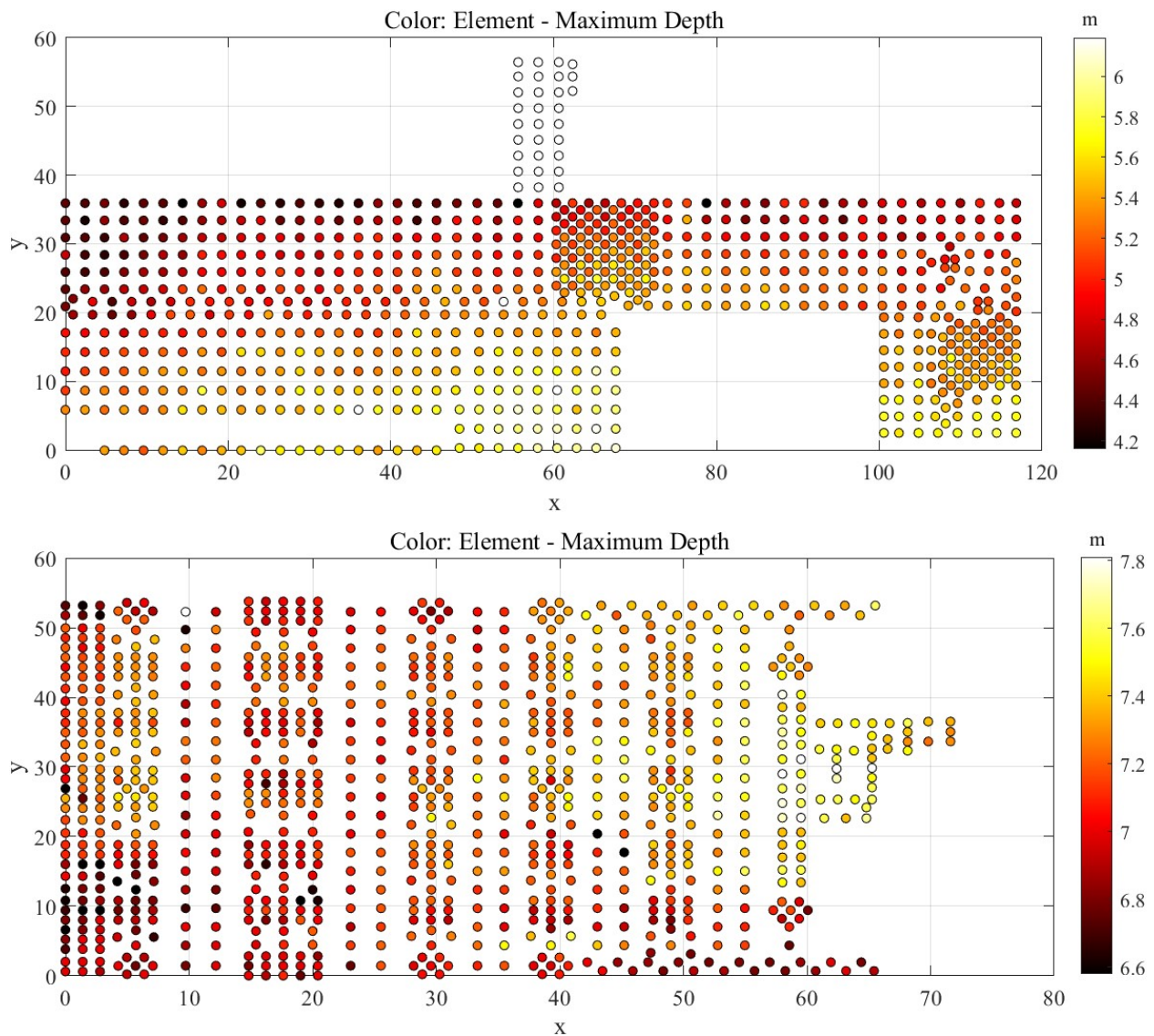


Figure 1: Geo-referenced view of the depth of two building sites. Each circle indicates an element; a deeper maximum depth is indicated with a bolder shade of the circle

depth of execution, vibrator amperage, pull down force, etc. [4.4]. This data is used in installation reports as the basis for geotechnical as well as quality evaluations [4.5]. Unfortunately, except for a few automated procedures, e.g., automatic recording of depth logs [5], the main part of the quality control is still performed manually whereby geotechnical experts analyse installation reports. In these reports, the time, depth, and compacting energy are logged [4.4, 4.6]. However, there is the necessity to manually evaluate several hundred installation reports per site with limited human resources. This creates the potential for oversights, or not all elements being evaluated with the necessary due diligence. This can lead to unforeseen failures of the future buildings [4.4]. It is very difficult to manually create an overview of the observed properties over a whole site from

hundreds of installation reports and to identify changing process properties over the site. These changes can be caused by the vibro soil improvement as it has an impact on the surrounding ground of the executed element [4.4] or by changing geotechnical properties of the soil within a building site. The understanding of the subsurface features, e.g., variations in the local bearing capacity, is crucial to ensure stable foundations, to avoid structural failure and post-construction problems in civil engineering [4.2, 4.4, 4.7]. As an indicator of changing subsurface properties within the site, the maximal required depth to obtain a stable column can be used [4.2]. A georeferenced overview of the maximum depth of the elements of two building sites is shown in Figure 1.

The results of site A (Figure 1b)¹ are discussed in more detail in the following sections.

2 Hybrid anomaly detection framework

Figure 2 gives an overview of the used methodology which is described in more detail in the following sections: Section 2.1 describes the systematic data collection and the calculation of the key performance indicators (KPIs) based on expert knowledge. Section 2.2 explains the machine learning (ML)-based anomaly detection, as well as the training set construction utilizing the outlierness, which is the result of the KPI-based module.

2.1 Systematic data collection and KPI

The rig is instrumented with multiple sensors that enable the collection of real-time machine data during performing the ground improvement process [4.8]. For each foundation column, the data are collected and organized as a multivariate time-series (MVTS). These MVTS form the basis for the knowledge discovery process (KDP). A series of KPI which capture expert knowledge, are computed for each column [4]. These KPI can be represented as a heat map (see Figure 2b) supporting the visual detection of anomalies. When fused with GPS coordinates, a georeferenced view of the data is produced (see Figure 3). This is important because it permits the modelling of systematic changes over a site. Furthermore, the statistics of the KPI are used to identify MVTS which are anomalous and possibly point at poor quality of the elements [4.9, 4.10].

From the KPI, the outlierness is derived as a parameter of the degree of an MVTS being anomalous. The outlierness is calculated as the number of KPI being an outlier, normalized by the total number of KPI. This is done for all columns of a site [4.9, 4.11]. The KPI classification of an MVTS into anomalous and non-anomalous is performed by determining a threshold based on the standard method for determining outliers. The thresholds are determined using the interquartile range (IQR), calculating an upper threshold T_u and lower threshold T_l :

$$T_l = q_{25} - c \text{ IQR}, \quad (4.3)$$

¹Due to confidentiality reasons, the data source is not disclosed.

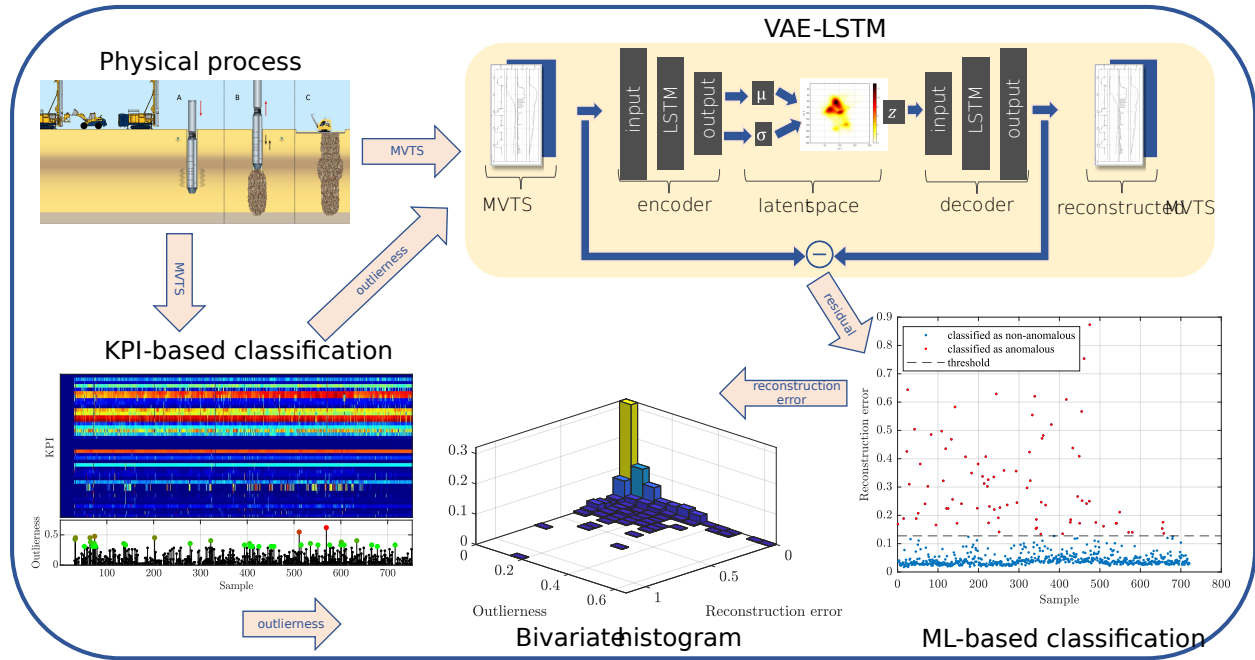


Figure 2: Overview of the used hybrid framework for anomaly detection. a) Schematic illustration of the ground improvement process with the three phases penetration (A), compaction (B), and surface completion (C). The recorded MVTs of the ground improvement process are used in the b) KPI-based anomaly detection as well as in the c) VAE with LSTM layers. The results of the KPI-based anomaly detection are utilized during training the ML model. d) The outcome of the VAE-LSTM is used in a classification into anomalous or non-anomalous. e) The results of the two means of anomaly detection are then visualized in a histogram and utilized in a KDP

$$T_r = q_{75} + c \text{ IQR}, \quad (4.4)$$

Where q_p denotes the quantile containing p per cent of the samples. A common choice for $c = 1.5$ [4.12].

2.2 AI for unsupervised anomaly detection

The goal of the ML classifier is to model repeating patterns in the data and to detect anomalies [4.13]. However, as the produced elements are sub-surface there is no possibility to establish ground truth. The manual inspection of the columns by a geotechnical expert, e.g., using dynamic probing [4.8], is time-consuming and costly; furthermore, this can only be applied after the process is finished [4.4]. As a result, no reliable labelling of the data can be performed during the production process. As a consequence, an unsupervised ML approach has been chosen [4.11]. Additionally, it is desirable that the ML-based classifier creates a generalization, which is valid, despite variations of properties over a site and from site to site [4.4].

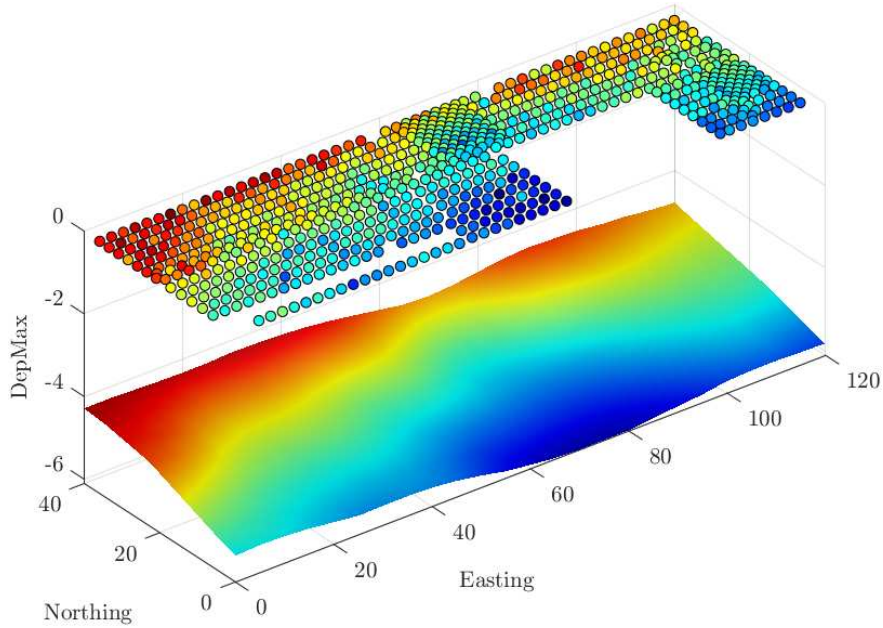


Figure 3: The upper layer corresponds to the geo-referenced data, whereas the smoothed lower layer is an inferred sub-surface model

The autoencoder (see Figure 2c) consists of a long shortterm memory (LSTM) encoder which produces the latent variables and an LSTM decoder which takes the latent encoding as input and produces the reconstructed signal \hat{y} . As the name suggests, the LSTM encodes the long- and short-term relationships within the data. The residual is given by $r = y - \hat{y}$, where y corresponds to the measurement and \hat{y} to the reconstructed signal [4.14]. The 2-norm of the residual is given as $\|r\|_2^2$. The neural networks have a large number of learnable parameters n , i.e., weights and biases; in this example, $n \approx 40.000 \dots 70.000$. Consequently, there are more degrees of freedom than required for a least squares optimum, i.e., the system is underdetermined. The variational part refers to regularizing [4.15] the cost function with a term relating to the distributions of the latent variables. The cost function C [4.14–4.17]

$$C = \|r\|_2^2 - \lambda D_{KL}(N(0, 1) | N(\mu, \sigma)) \quad (4.5)$$

consists of two terms:

1. the reconstruction error: $\|r\|_2^2$,
2. the Kullback–Leibler divergence of the latent variables from the standard, normal distribution: $D_{KL}(N(0, 1) | N(\mu, \sigma))$.

The regularization factor λ is a hyperparameter. The optimization of hyperparameters is discussed later in this section. For $\lambda = 1$, C is the cost function of a classical variational autoencoder

(VAE), called the evidence lower bound (ELBO), as described in ref. [4.14]. The weights and biases of the network are optimized during the learning process [4.14, 4.18].

Classification is simply the task of assigning a specific value to a category; for example, an MVTs can be assigned to the category anomalous or non-anomalous [4.19]. This can be performed in both a supervised and an unsupervised manner [4.20–4.23]. The ML-based classification is performed by thresholding the reconstruction error using a threshold which is skewness-adjusted [4.24, 4.25]. The probability distribution of the reconstruction error is fundamentally skewed. Therefore, the single-sided skewness adjustment is needed when computing the threshold [4.11, 4.26].

Hyperparameters

The hyperparameters are the parameters considered to be constant for any single training task. A study on the impact of the hyperparameters on the network performance was performed. This is still a topic of ongoing research. The results obtained so far suggest that the hyperparameter optimization (HPO) has a major impact on the performance of the ML-based classifier. It determines the dimensionality, i.e., the number of neurons in the encoder and decoder, number of layers used, as well as the training settings of the VAE-LSTM. Here, a genetic algorithm has been used for the HPO. More details on the obtained results can be found in refs. [4.11, 4.26, 4.27].

When applying different pre-processing techniques as described in the following sections or using data from another site, it may be necessary to repeat the optimization of the hyperparameters. Table 4.2 shows the hyperparameters used in the subsequent sections of this article.

Table 4.2: Results of the hyperparameter optimization used to train the ML models

Hyperparameter name	HPO results for data extended processing	HPO results for data before pre-processing	HPO results for data extended processing and new sub-set of channels
Neurons LSTM layer encoder	57		27
Neurons LSTM layer decoder	95		36
Learning rate	0.0109		0.0235
Mini-batch size	10		25
Number of epochs trained	154		154
Regularization factor	2		8
Latent dimension	3		2

Training

The unsupervised training of the VAE-LSTM takes advantage of the statistics of the KPI to perform an auto-selection of the MVTs with zero outlierness. Training with these improves the discriminative power of the VAE-LSTM [4.28]. In this manner, the non-representative features of the anomalous MVTs are not reconstructed accurately, leading to a higher reconstruction error and a better distinction [4.29]. The trained VAE-LSTM is now applied to the complete data set [4.11].

3 Results of the hybrid classifier

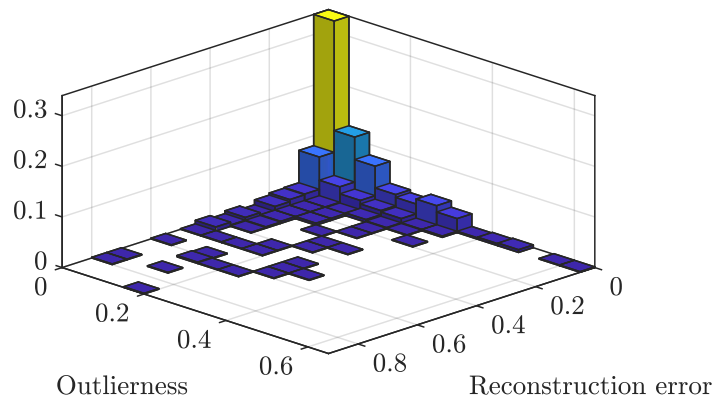


Figure 4: Normalized bivariate histogram of the outcome of the outlierness and the reconstruction error for all MVTs of a site

The bivariate histogram of the KPI outlierness and the reconstruction error of the VAE-LSTM are shown in Figure 7. This provides a mechanism to establish a hybrid classification. In Figure 7, it can be observed that the VAE-LSTM has samples with a high reconstruction error which have a low outlierness. These samples could be anomalies which were not covered by the KPI.

3.1 KDP

The purpose of knowledge discovery is to improve both the understanding of the process and the automatic detection of anomalies. The outlierness and the reconstruction error lead to a bivariate classification with four classes. The interesting class is the case that creates discrepancies between the KPI- and ML-based classification. These are used in a KDP to establish an understanding of the data, to improve pre-processing and define new KPI. This improves the reliability of the anomaly detection.

3.2 Extended data pre-processing

There are a number of pauses in the vibration process, e.g., during these pauses gravel is filled into the vibrator. There are a number of pauses in the vibration process, e.g., during these pauses gravel is filled into the vibrator.

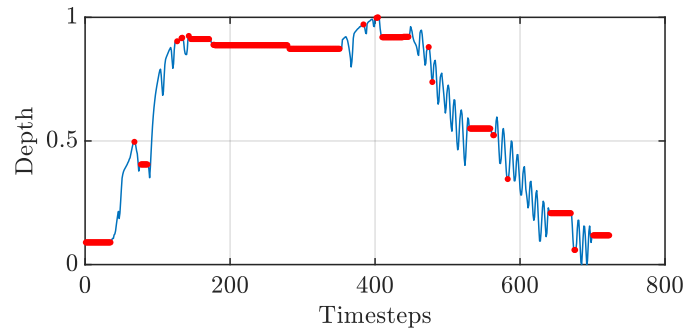


Figure 5: Example of a normalized depth profile of a drilling with marked idle time (red)

For instance, the VAE-LSTM identified MVTs as anomalous where there is a long waiting period for the delivery of gravel as anomalous because it is a significant change in the time pattern. An example of such a time-series is shown in Figure 5. As a result of this insight, the original pre-processing which was based on z-norm is extended by removing the pauses (see Figure 6). Removing the idle times yields a lower level of discrepancy between the KP I and ML-based classification without compromising quality by not flagging these columns as anomalous.

The bivariate histogram of the results of the classification after removing the idle time is shown in Figure 7. Note the fewer discrepancies between the outlieriness (KPI) and the reconstruction error (VAE-LSTM).

Removing the pauses can, however, lead to discontinuities in some of the channels of the recorded signals, e.g., the temperature of the vibrator (see Figure 6). In this example, the vibrator is stationary. Removing the pauses in the depth signal leads to discontinuities in the temperature signal. This is indicating that while filling gravel, despite the depth not changing, the temperature is still rising. This could be caused by the vibrator left running during the filling of the gravel. The quality of a foundation column is not influenced by this behaviour.

3.3 Impact of channels on reconstruction quality

The following channels were defined as having a direct impact on the quality of the foundation column, depth, vibrator amperage, and pull-down force. These channels have been selected due to their physical significance. From these channels, the work performed as a function of depth can be calculated.

Training the ML-based classifier with these signals leads to the classification, as shown in Figure 8. The described steps lead to an overall low reconstruction error of the non-anomalous

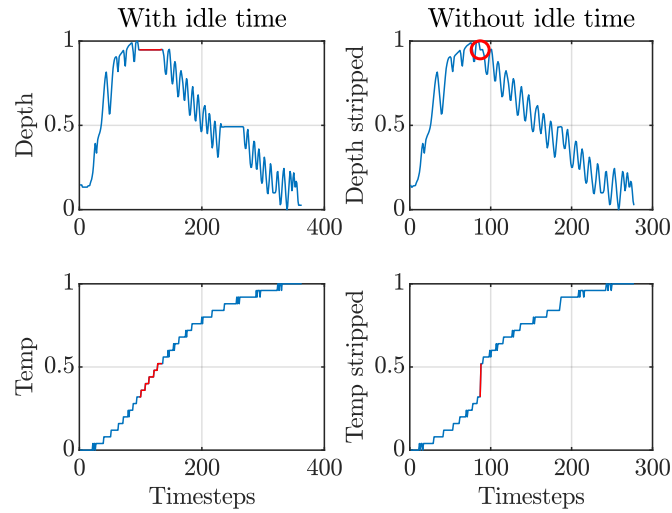


Figure 6: Normalized depth and temperature (temp) signal a) before and b) after removing the idle times. In the plots on the right, the portions with constant depth (idle times) are removed from the depth and temperature signals. Constant depth with increasing temperature indicates that the vibrator was still running while the drilling was not performed, hence the depth is not increasing. The values with these constant portions are removed to detect outliers in terms of effectiveness and not in terms of efficiency

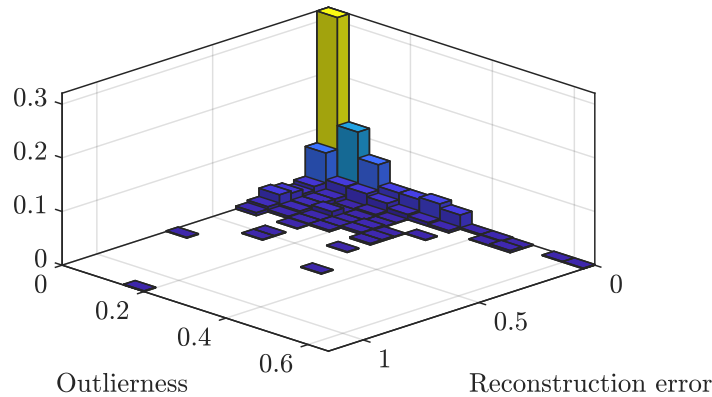


Figure 7: Normalized bivariate histogram showing the outlierness and the reconstruction error after incorporating the results from the knowledge discovery process

samples and a classification into MVTS being non-anomalous and anomalous. The MVTS which have a reconstruction error beyond the threshold determined with the skewness-adjusted boxplot [4.24] are classified as non-anomalous, and the MVTS with a reconstruction error greater than the threshold as anomalous.

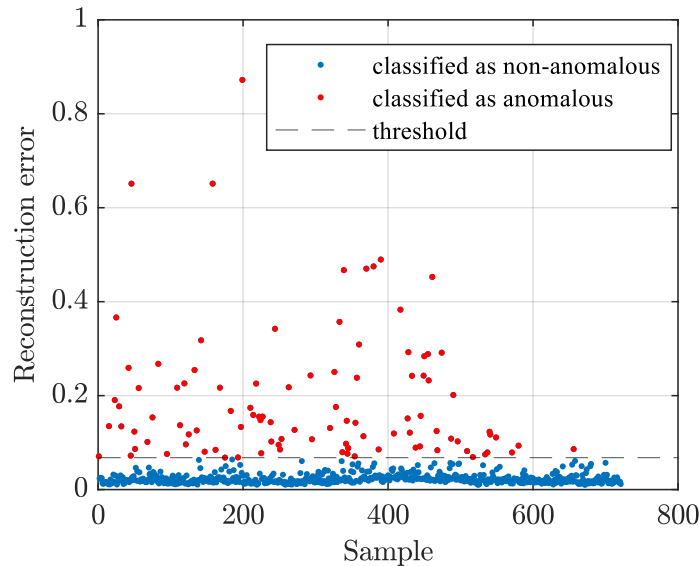


Figure 8: Classification of the ML-based classifier

4 Conclusions and further work

The combination of the KPI and ML modelling yields a higher certainty in the classification. This reduces the risk of not detecting anomalous foundation columns. The hybrid approach has also triggered a new KDP; since the VAE-LSTM indicated MVTs that the KPI – a capturing of expert knowledge – did not detect. The insights gained from the ML-based model were used to redefine the selection of channels and a modified pre-processing. The monitored process consists of two sub-processes: penetration and compaction; it may be beneficial to train separate ML models for the sub-processes and combine the results, which should be addressed in future work.

Acknowledgements

This work was partially funded by the COMET program within the K2 Center ‘Integrated Computational Material, Process and Product Engineering (IC-MPPE)’ (Project No. 859480). This program was supported by the Austrian Federal Ministries for Transport, Innovation and Technology (BMVIT) and for Digital and Economic Affairs (BMDW), represented by the Austrian research funding association (FFG), and the federal states of Styria, Upper Austria, and Tyrol.

Bibliography

- [4.1] A. Terbuch, A. Zöhrer, V. Winter, P. O’Leary, N. Khalili-Motlagh-Kasmaei, and G. Steiner, “Quality monitoring in vibro ground improvement – A hybrid machine learning approach,”

Geomechanics and Tunnelling, vol. 15, no. 5, pp. 658–664, Oct. 2022.

- [4.2] P. Nagy and D. Adam, “Quality control of deep vibro compaction based on the vibrator movement,” in *Proceedings of the XVII European Conference on Soil Mechanics and Geotechnical Engineering*, 2019.
- [4.3] R. S. Pugh, “Settlement of floor slabs on stone columns in very soft clays,” *Proceedings of the Institution of Civil Engineers - Geotechnical Engineering*, vol. 170, no. 1, pp. 16–26, 2017.
- [4.4] P. Nagy, “Rütteldruckverdichtung : dynamische Verdichtungskontrolle auf Basis der Rüttlerbewegung : dynamic compaction control based on the vibrator movement,” Ph.D. dissertation, TU Wien, 2018.
- [4.5] K. Kirsch, *Ground Improvement by Deep Vibratory Methods*, 2nd ed. Milton: Taylor & Francis Group, 2017.
- [4.6] Österreichische Gesellschaft für Geomechanik, “Empfehlungen für die ausschreibung von tiefenrüttelverfahren (rüttelstopf-/rütteldruckverdichtung),” Salzburg, 2013.
- [4.7] S. Roy and S. K. Bhalla, “Role of geotechnical properties of soil on civil engineering structures,” *Resources and Environment*, vol. 7, no. 4, pp. 103–109, 2017.
- [4.8] P. Nagy, D. Adam, F. KOPF, and P. FREITAG, “Experimental and theoretical investigation of deep vibro compaction,” *ce/papers*, vol. 2, no. 2-3, pp. 725–730, 2018.
- [4.9] N. Khalili-Motlagh-Kasmaei, D. Ninevski, P. O’Leary, C. J. Rothschedl, V. Winter, and A. Zöhrer, “A digital twin for deep vibro ground improvement,” in *International Conference on Deep Foundations and Ground Improvement: Smart Construction for the Future (DFI-EFFC 2022)*, 2022.
- [4.10] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, “A review on outlier/anomaly detection in time series data,” *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1–33, 2021.
- [4.11] A. Terbuch, P. O’Leary, and P. Auer, “Hybrid machine learning for anomaly detection in industrial time-series measurement data,” in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2022, pp. 1–6.
- [4.12] J. Yang, S. Rahardja, and P. Fränti, “Outlier detection,” in *Proceedings of the International Conference on Artificial Intelligence, Information Processing and Cloud Computing - AIIPCC ’19*, J. M. R. S. Tavares and Z. Xu, Eds. ACM Press, 2019, pp. 1–6.
- [4.13] K. P. Murphy, *Machine learning: A probabilistic perspective*, ser. Adaptive computation and machine learning series. Cambridge, Massachusetts and London, England: The MIT Press, 2012.

- [4.14] D. P. Kingma, *An Introduction to Variational Autoencoders*, ser. Foundations and Trends® in Machine Learning Ser. Norwell, MA: Now Publishers, 2019, vol. v.40.
- [4.15] H. W. Engl, M. Hanke-Bourgeois, and A. Neubauer, *Regularization of inverse problems*, ser. Mathematics and its applications. Dordrecht: Kluwer Acad. Publ, 2000, vol. 375.
- [4.16] D. P. Kingma and M. Welling, “An introduction to variational autoencoders,” *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.
- [4.17] I. Higgins, L. Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, M. Botvinick, S. Mohamed, and Alexander Lerchner, “beta-vae: Learning basic visual concepts with a constrained variational framework,” in *5th International Conference on Learning Representations (ICLR 2017)*, 2017.
- [4.18] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [4.19] S. Agrawal and J. Agrawal, “Survey on anomaly detection using data mining techniques,” *Procedia Computer Science*, vol. 60, pp. 708–713, 2015.
- [4.20] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*, ser. Adaptive computation and machine learning. Cambridge, Massachusetts and London, England: The MIT Press, 2016.
- [4.21] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, vol. 385.
- [4.22] W. H. Lopez Pinaya, S. Vieira, R. Garcia-Dias, and A. Mechelli, “Autoencoders,” in *Machine Learning*. Elsevier, 2020, pp. 193–208.
- [4.23] M. Kozak and C. H. Scaman, “Unsupervised classification methods in food sciences: discussion and outlook,” *Journal of the Science of Food and Agriculture*, vol. 88, no. 7, pp. 1115–1127, 2008.
- [4.24] M. Hubert and S. van der Veeken, “Outlier detection for skewed data,” *Journal of Chemometrics*, vol. 22, no. 3-4, pp. 235–246, 2008.
- [4.25] M. Hubert and E. Vandervieren, “An adjusted boxplot for skewed distributions,” *Computational Statistics & Data Analysis*, vol. 52, no. 12, pp. 5186–5201, 2008.
- [4.26] A. Terbuch, “LSTM hyperparameter optimization: Impact of the selection of hyperparameters on machine learning performance when applied to time series in physical systems,” Ph.D. dissertation, Chair of Automation, Montanuniversitaet Leoben, 2021.

- [4.27] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, “Lstm: A search space odyssey,” *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2017.
- [4.28] I. Steinwart, D. Hush, and C. Scovel, “A classification framework for anomaly detection,” *Journal of Machine Learning Research*, vol. 6, no. 8, pp. 211–232, 2005.
- [4.29] T. Kieu, B. Yang, and C. S. Jensen, “Outlier Detection for Multidimensional Time Series Using Deep Neural Networks,” in *2018 19th IEEE International Conference on Mobile Data Management (MDM)*. IEEE, Jun. 2018, pp. 125–134.

Authors

Anika Terbuch (corresponding author)

anika.terbuch@unileoben.ac.at

University of Leoben

Chair of Automation

Peter Tunner Strasse 25

8700 Leoben

Austria

Alexander Zöhrer

alexander.zoehrer@keller.com

Keller Grundbau GesmbH

Packer Straße 167

8561 Söding

Austria

Vincent Winter

vincent.winter@keller.com

Keller Grundbau GesmbH

Guglgasse 15

1110 Wien

Austria

Paul O’Leary

paul.oleary@unileoben.ac.at

University of Leoben

Chair of Automation

Peter Tunner Strasse 25

8700 Leoben
Austria

Negin Khalili-Motlagh-Kasmaei
negin.khalilimotlaghkasmaei@unileoben.ac.at
University of Leoben
Chair of Automation
Peter Tunner Strasse 25
8700 Leoben
Austria

Gernot Steiner
g.steiner@stud.unileoben.ac.at
University of Leoben
Chair of Automation
Peter Tunner Strasse 25
8700 Leoben
Austria

How to Cite this Paper

Terbuch, A.; Zöhrer, A.; Winter, V.; O'Leary, P.; Khalili-Motlagh-Kasmaei, N.; Steiner, G. (2022) Quality monitoring in vibro ground improvement – A hybrid machine learning approach. *Geomechanics and Tunneling* 15, No. 5, pp. 658–664.

<https://doi.org/10.1002/geot.202200028>

This paper has been peer reviewed. Submitted: 13. June 2022; accepted: 18. August 2022.

Detecting Anomalous Multivariate Time-Series via Hybrid Machine Learning

Anika Terbuch¹, Paul O’Leary¹, Negin Khalili-Motlagh-Kasmaei¹,
Peter Auer², Alexander Zöhrer³, and Vincent Winter³

¹ Chair of Automation, Montanuniversität Leoben, Leoben, Austria

² Chair of Information Technology, Montanuniversität Leoben, Leoben, Austria
{anika.terbuch, paul.oleary, negin.khalilimotlaghkasmaei,
peter.auer}@unileoben.ac.at

³ Keller Grundbau GesmbH, Vienna and Söding, Austria
{alexander.zoehrer, vincent.winter}@keller.com

Originally appeared as:

A. Terbuch, P. O’Leary, N. Khalili-Motlagh-Kasmaei, P. Auer, A. Zöhrer, and V. Winter, "Detecting Anomalous Multivariate Time-Series via Hybrid Machine Learning," IEEE Transactions on Instrumentation and Measurement, Vol. 72, 2023, DOI: 10.1109/TIM.2023.3236354

This work is licensed under the Creative Commons Attribution 4.0 International License (CC BY 4.0). To view a copy of this license, visit <https://creativecommons.org/licenses/by/4.0/>

Detecting Anomalous Multivariate Time-Series via Hybrid Machine Learning

Anika Terbuch¹, Paul O’Leary¹, Negin Khalili-Motlagh-Kasmaei¹,
Peter Auer², Alexander Zöhrer³ and Vincent Winter³

¹Chair of Automation,

²Chair of Information Technology,

Montanuniversität Leoben,

Leoben Austria

{anika.terbuch, paul.oleary, negin.khalilimotlaghkasmaei, peter.auer}
@unileoben.ac.at

³Keller Grundbau GesmbH

Vienna and Söding, Austria

{alexander.zoehrer, vincent.winter}@keller.com

IEEE Transactions on Instrumentation and Measurement, Vol. 72, 2023
Digital Object Identifier 10.1109/TIM.2023.3236354

Abstract

This article ¹ investigates the use of hybrid machine learning (HML) for the detection of anomalous multivariate timeseries (MVTS). Focusing on a specific industrial use-case from geotechnical engineering, where hundreds of MVTS need to be analyzed and classified, has permitted extensive testing of the proposed methods with real measurement data. The novel hybrid anomaly detector combines two means for detection, creating redundancy and reducing the risk of missing defective elements in a safety relevant application. The two parts are: 1) anomaly detection based on approximately 50 physics-motivated key performance indicators

¹Manuscript received 17 September 2022; revised 5 December 2022; accepted 20 December 2022. Date of publication 12 January 2023; date of current version 20 January 2023, The Associate Editor coordinating the review process was Dr. Valentina Bianchi (Corresponding author: Anika Terbuch.)

(KPIs) and 2) an unsupervised variational autoencoder (VAE) with long short-term memory layers. The KPI captures expert knowledge on the properties of the data that infer the quality of produced elements; these are used as a type of auto-labeling. The goal of the extension using machine learning (ML) is to detect anomalies that the experts may not have foreseen. In contrast to anomaly detection in streaming data, where the goal is to locate an anomaly, each MVTS is complete in itself at the time of evaluation and is categorized as anomalous or nonanomalous. The article compares the performance of different VAE architectures [e.g., long short-term memory (LSTM-VAE) and bidirectional LSTM (BiLSTM-VAE)]. The results of using a genetic algorithm to optimize the hyperparameters of the different architectures are also presented. It is shown that modeling the industrial process as an assemblage of subprocesses yields a better discriminating power and permits the identification of interdependencies between the subprocesses. Interestingly, different autoencoder architectures may be optimal for different subprocesses; here two different architectures are combined to achieve superior performance. Extensive results are presented based on a very large set of real-time measurement data.

Keywords: Artificial intelligence in measurement and instrumentation, hybrid learning, key performance indicator(KPI), long short-term memory (LSTM)-variational autoencoder (VAE), outlier detection, timeseries.

1 Introduction

This article investigates the use of hybrid machine learning (HML) [4.1] to infer the quality of industrially produced elements, for which final quality control is not possible. In the specific use-case analyzed and presented here, underground foundation columns are being produced; however, since they are sub-surface their final quality cannot be inspected directly. Consequently, the quality must be inferred through the analysis of measurement data acquired in real-time from the instrumented rig. This requires the analysis of large sets of multivariate real-time measurement data (MVRTD), emanating from instrumented industrial equipment. This is a pertinent topic since industrial IoT [4.2] is making ever-increasing volumes of measurement data available for analysis. Furthermore, HML, for example, physics-informed neural networks [4.3], is an emerging research topic and is considered the best approach to obtaining reliable results in conjunction with the analysis of the measurement data from physical systems [4.4–4.6]. Although this article is focused on a specific use-case, many of the results are relevant in other applications; in particular the examination of different autoencoder architectures for the unsupervised analysis of multivariate time-series (MVTS) data and the combination with a genetic algorithm to optimize the hyperparameters.

The importance of machine learning (ML) in the field of instrumentation and measurement [4.4, 4.7] is increasing with the growing volumes of measurement data recorded in industrial processes. However, as shown in [4.5], the two communities, i.e., the machine learning and the instrumentation and measurement community, use different terminologies. As a result, we feel it is important to have interdisciplinary publications that bridge both the topics.

As described in [4.7] ML architectures should be chosen with care when applied in safety-

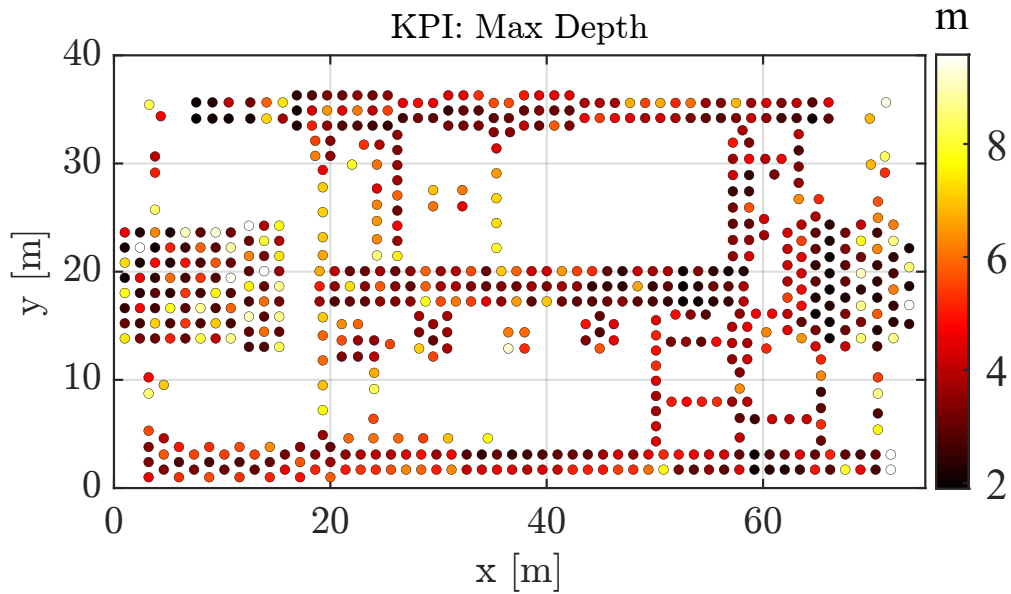


Figure 1: Georeferenced data from an exemplary building site. Each point corresponds to the location of an underground foundation column. The color of the point represents the value of some specific KPIs; in the case shown here, it is the depth required to achieve a specific load-bearing capacity. There is a real-time MVTS associated with each of these foundations, and for this site, there are $m = 637$ points.

relevant applications, as is the case here. For this reason, an HML approach was chosen, which augments an outlier detection technique based on multiple physics-based metrics, with unsupervised ML. The HML techniques were successfully applied in other fields related to instrumentation and measurement, e.g., in [4.8] and [4.9]. Many ML approaches presented in literature use publicly available datasets to perform ablation studies [4.10] and performance benchmarking of different architectures. A survey on different methods for time-series forecasting [4.11] on publicly available time-series data sets suggested that hybrid approaches are superior to pure statistical or ML approaches. The same study concluded that more complex models do not necessarily lead to higher accuracy. This is important, since high accuracy, in the field of instrumentation and measurement, is a highly desirable feature [4.5, 4.6].

Especially in anomaly detection, benchmarks need to be treated with care, since most suffer from one or all the following issues [4.12]: the problem is too trivial and can be solved in a few lines of code; the anomaly density is too high; if supervised learning is used, the ground truth is mislabeled and often there are no data available after the anomaly has occurred, and Wu and Keogh [4.12] called this as *run to failure bias*. The study done on the comparison of the classical pure ML methods, pure (deep) ML methods, and the combination of both, HML, suggests that there is no pure ML method that can outperform a hybrid system [4.13].

1.1 New Contributions

The initial results of this work were published in [4.14]; however, since then, significant modifications have been made and improved results obtained. This article presents these extensions and new results:

1. An HML framework that captures a priori knowledge in the form of KPI, which is used to implement auto-labeling, permitting a truly unsupervised learning. Furthermore, the combination of two means for anomaly detection, key performance indicators (KPI) and ML, is introduced; this leads to a redundant anomaly detection system and a higher likelihood for detecting anomalies.
2. The conceptual framework for handling MVTs measurement data in an object-oriented way is presented; covering all the tasks from data ingestion to outlier detection. The data are handled together with the pertinent metadata, events, and segments. The concept that industrial processes can be described as consisting of sub-processes is discussed. KPIs are defined for the subprocesses separately and this leads to a hierarchical anomaly detection in the subprocesses. Aggregating the results yields a more granular classification of anomalies and reveals possible interactions between the subprocesses.
3. A clear definition and algorithmic implementation for the *outlierness* is provided, i.e., a metric for the extent to which a time-series is considered to be an outlier by the KPI-based anomaly detector.
4. The statistical analysis of the performance of autoencoders with LSTM and BiLSTM layers for the sub-processes is presented.
5. The implementation of the autoencoder training has been modified, to significantly reduce the amount of padding required within the mini-batches during training.

1.2 Organization of the Paper

This article is organized as follows: in Section 1, an introduction to the topic of HML for anomaly detection in multivariate time-series is given; in Section 2, insights into the use-case are presented; Section 3 discusses the structured data handling for large amounts of industrial real-time data. The first part of the hybrid framework based on KPI is shown in Section 4, whereas the anomaly detection based on ML can be found in Section 5, and the results of the comparison of architectures are presented in Section 6. The article is concluded with insights gained and further work in Section 7.

2 Use-Case and Nature of the Data

The vibro-ground improvement techniques are used for almost 100 years to support buildings and infrastructure works, however there is still no standardized method for quality control for ground improvement techniques [4.15]. The increasing economic pressure in the construction industry and a higher risk awareness accelerated the change process of traditional, mostly manually performed techniques, toward automated solutions for quality control [4.16]. The produced foundation columns are subsurface, and therefore, the quality control needs to be based on indirect methods and reliable labeling for supervised ML is not a feasible solution. For the last 30 years monitoring KPIs, which were derived from the recorded data from instrumented production rigs built the basis for geotechnical and quality evaluations of the process [4.17]. These data are also utilized in installation reports which to this day are the core of quality control. For each foundation column, an installation report is created, which needs to be manually evaluated by geotechnical experts [4.15]. However, evaluating hundreds, if not thousands, of installation reports with limited human resources increases the risk that the controls are not performed with the due-diligence required. For this reason, our framework is used to support the geotechnical expert at this task and refining the KPI definitions to cover as many anomalies as possible. The KPI refinement is driven by new process insights, gained by the knowledge discovery process, performed after ML anomaly detection, to increase the interpretability of the anomaly detection framework, which is essential in critical infrastructure and safety-relevant applications [4.18]. The opportunity costs of false negative are much higher than of false positive detection. Therefore, the two anomaly detection results are combined with a logical or-if an MVTS is flagged by one of the means for anomaly detection as anomalous it is considered as anomalous and undergoes further investigations by geotechnical experts.



Figure 2: Here a series of three instrumented rigs can be seen on a building site. This gives an impression of the scale of operations and the relevance of location registered data.

In more detail, the use-case presented here is the vibro ground improvement process. The goal is to improve the bearing capacity of the ground so that more stable foundations can be produced for a building. However, this leads to a safety-relevant system, since failure to detect an anomalous

foundation column can lead to a local instability of a building. The process consists of four phases: *run-in*, process preparation is performed; *penetration*, whereby a rig penetrates the ground with a large vibrator until a predefined bearing capacity is reached; *compaction*, is the repetitive process of withdrawing the vibrator, introducing gravel into the ground and compacting with the vibrator; and *run-out* finalizing the process.

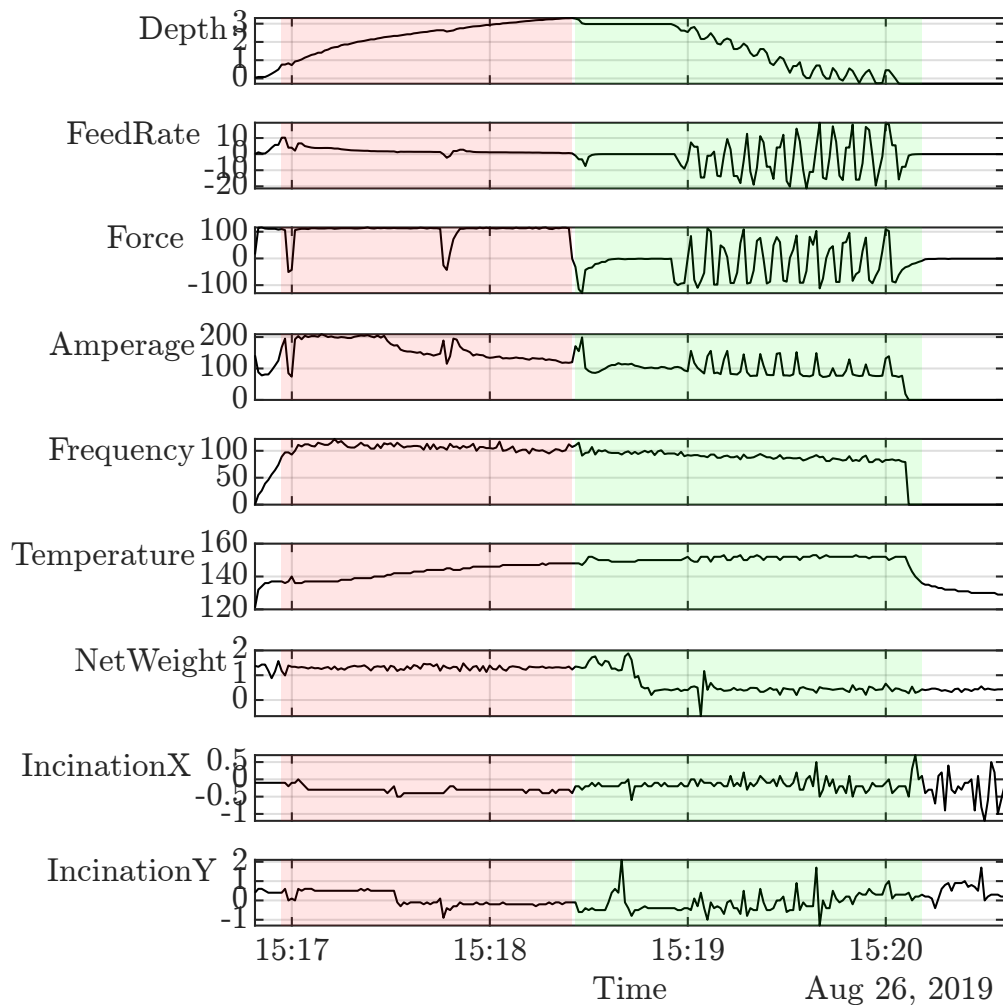


Figure 3: Example of the real-time data acquired from the rig while producing a single foundation column. One such MVTS is acquired per foundation column. The four phases of the process can be seen: run-in (white), penetration (red), compaction (green), and run-out (white). Only the penetration and compaction phases are relevant for the quality of the foundation column.

A typical building site may have between $250 \leq m \leq 1500$ such foundation columns. Fusing the location data with machine data enables the georeferenced viewing of the data associated with the foundation columns. This supports the geotechnical experts in establishing an overview of the geotechnical properties and enables an automatic detection of the changing subsurface properties. An example of such a view is shown in Figure 1; this site has $m = 637$ foundation columns. The

georeferencing permits the identification of systematic changes in the subsurface ground properties over a site. This is important since reliable detection of anomalies requires the separation of systematic changes from local anomalous behavior.

A series of instrumented rigs work in parallel on a single building site; see Figure 2, producing the required foundation columns. The instrumented rigs have $n = 9$ sensor channels used in this process. The sensor data acquired in real-time are sampled at $f_s = 1 \text{ Hz}$. This yields an MVTS for each foundation column; an example of such data can be seen Figure 3. Due to changes in the subsurface properties, the time required to produce the foundation may vary significantly; as a result, the time-series have strongly varying lengths. For the site data shown in Figure 1, there is a median number of time-steps per foundation of $t_{med} = 345$ and an interquartile range (IQR) of $t_{IQR} = 169$. A t_{med} of 345 time-steps with 1 Hz sampling frequency corresponds to an average lead time of 345 s. Consequently, the computational methods must be capable of analyzing, the execution of the production process with strongly varying lead time, which leads to recordings of the process with widely varying lengths. To summarize, there are approximately 1000 MVRTD associated with a single building site, each of these time-series having $n = 9$ sensor channels and several hundred time-steps recorded per sensor. Furthermore, there are multiple building sites to be monitored.

3 Structured Handling of Data

A significant portion of this project is associated with managing such volumes of multivariate measurement data, together with the corresponding metadata, in a systematic manner. There are tens of thousands of MVRTD that need to be handled. Consequently, a systematic data ingestion process has been defined. In this process, each of the MVTD is ingested. In the same step, events and segments are determined by applying a rule-based system. By performing data fusion, the data from computer-aided design (CAD) are merged with machine data for each of the foundations. This provides for the georeferencing of the data and enables the comparison of planned versus executed work. An object class for an MVTS has been defined. It provides a container for the MVRTD and augments this with the respective metadata, events, and segments. In this manner, all data required to segment and process the data from a specific foundation column are contained in a single MVTS object. These objects are saved in binary HDF5 [4.19] format and made available via a standard interface. This facilitates the exchange of data and fast loading into memory.

In addition, an index has been implemented with one column per foundation element and one row per indexing value, e.g., KPI. This permits the identification of MVTS from either its metadata or the values of the associated KPI or by an indexing value emanating from the ML evaluation of the data.

4 Key Performance Indicators

Given the collection of MVTS, the task now is to individually characterize them by a series of metrics, called KPIs here. The goal of the technical KPI is to capture expert knowledge about the process and physical properties that can be expressed in terms of scalar metrics computed from the real-time machine data. These provide the physics-based side of the hybrid learning process.

The penetration and compaction phases are definitive for the quality of the executed foundation column, whereas the complete process, including run-in and run-out, may be relevant for the efficiency. Consequently, KPIs are defined for and categorized according to: penetration, compaction, and complete element. The categorization is achieved via metadata and not hard coded. In previous work [4.14], these phases were analyzed together. However, defining KPI in separate categories provides the possibility of a more granular classification of the MVTS according to the subprocesses. For example, the penetration phase may be anomalous, whereas the compaction is not. Currently, there are more than $p = 50$ different KPIs computed.

4.1 Exemplary Technical KPI

Here, two new exemplary KPIs² are presented. They are physics-based metrics and permit the identification of certain types of anomalies. Furthermore, they can be computed using the map-reduce paradigm [4.20], opening the door to parallel computation, if required.

The incremental work as a function of time $w(t)$ is computed from the amperage $a(t)$ multiplied with the operating voltage (electrical work) and the product of force $f(t)$ and penetration rate $dd(t)/dt$ (mechanical work). This computation has the same temporal resolution as the MVTS data and corresponds to the map step in map-reduce. Then the total work W , required to perform penetration, is the integral over $w(t)$

$$W = \int_{t_s}^{t_e} w(t) dt, \quad (4.6)$$

where by t_s and t_e correspond to the start and end times for penetration, respectively, and this aggregation corresponds to the reduce step. The total work required to complete penetration W is used as a metric to characterize each MVTS.

The second KPI presented L is the ratio of the traversed length to the depth penetrated. Given $d(t)$ the depth as a function of time, L is computed as follows,

$$L = \frac{\int_{t_s}^{t_e} |d(t)| dt}{\int_{t_s}^{t_e} d(t) dt}. \quad (4.7)$$

Consider the exemplary MVTS shown in Figure 4: this corresponds to a straightforward penetration of the ground and yields $L = 1$. In contrast, consider Figure 5: as can be seen, there

²It is not possible to present details on more of the KPIs due to the limited space available in this paper

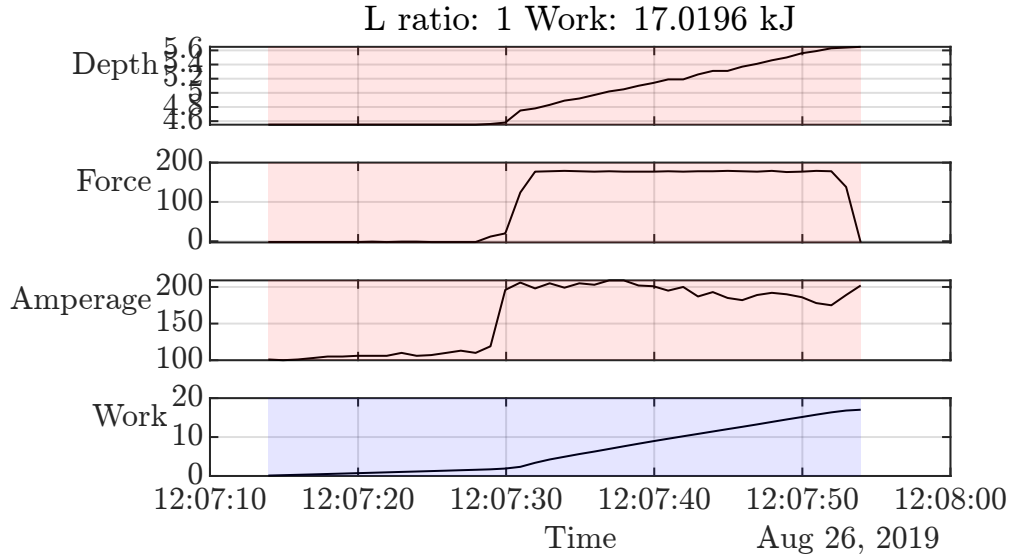


Figure 4: Depth $d(t)$, force $f(t)$ and amperage $a(t)$ are the channels that form the MVTs of a single foundation column. The *work* channel $w(t)$, underlaid in blue, is a derived variable, computed from the above three channels and has the same temporal resolution as the real-time data from the machine. The integral over $w(t)$ yields the total work required to perform penetration. This dataset corresponds to the almost ideal process of creating a column. The *L ratio*, indicated in the header, refers to the ratio of traversed distance to depth penetrated.

were difficulties penetrating the ground and the vibrator had to be retracted several times and the process restarted. This behavior is considered anomalous; however, it has predictive value, since together with the georeferencing it can be used to determine whether this is a local random anomaly, or if the subsurface properties are changing systematically over a site.

4.2 Heat-Maps and Outlier Detection via KPI

Currently, more than $p = 50$ KPI are computed, in three categories, for each foundation column. The KPI can be organized as a matrix $T(j, k)$, whereby each row j corresponds to a KPI and each column k to a specific foundation point. Normalizing the KPI matrix T , using min-max normalization, by row provides a uniform scaling for the graphical representation, e.g., in Figure 6 the KPI heat-map for the compaction phase and in Figure 7 the heat-map of the KPI relevant for all phases is shown.

In addition, the KPI matrix T is used to detect MVTs that are statistical outliers; this infers the possibility of an anomalous foundation column. The foundation columns are subsurface, the precludes access to the *ground-truth*, and consequently, an inference must be used at this point. The term *outlierness* is coined here, and it is a relative measure of the degree to which the MVTs can be considered to be an outlier. It is the number of KPI for which the column data are a statistical

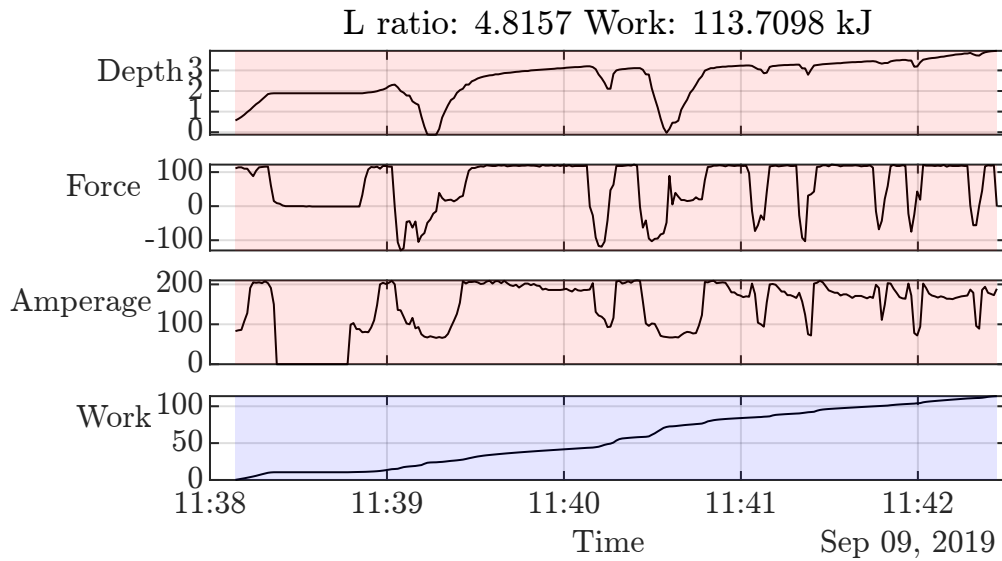


Figure 5: Data similar to Figure 4 but for an anomalous foundation column. This is an example where the operator had difficulties penetrating the ground with the vibrator. Both the L ratio and the total work performed are physics-relevant metrics and are suitable to identify this type of anomaly.

outlier divided by the total number of KPI being considered. The pseudo-code for the computation of the outlierness is shown in Algorithm 1.

5 ML-based Anomaly Detection

As described in [4.18], autoencoders are a popular choice for anomaly detection in critical infrastructure. The task of identifying patterns that are not present in data from normal operations is called anomaly detection [4.18]. Anomaly detection can also be described as a binary classification problem with one class containing the anomalous data and one class containing the nonanomalous samples [4.21]. Modeling the normal behavior in physical systems is often not a feasible solution, if it is unknown or simply too complex. In ML approaches for anomaly detection, this normal behavior is learned/inferred from provided data samples [4.21]. A simplified drawing of the architecture used is shown in Figure 8.

Most of the lately invented architectures for anomaly detection in time-series data are designed for streaming data for the multivariate [4.22, 4.23] or univariate case [4.24] using sliding window approaches. The industrial use-case presented in this paper, however, deals with a high number of multivariate time-series data files with recordings of varying length. Each MVTs is complete at the time of evaluation. In this manner it is more closely related to boundary value problems, than locating a changing pattern in streaming data.

To apply the autoencoder to the samples of varying length, a novel resampling approach was developed (see Section 5.4), which reduces the effects of resampling to a minimum.

Algorithm 1 Pseudo/Code for the computation of *outlierness*

procedure OUTLIERNESS(T)▷ Given the matrix T of KPI compute the *outlierness* for each foundation column. $B \leftarrow \text{false}(\text{size}(T))$

▷ Initialize binary matrix

 $[o, p] \leftarrow \text{size}(T)$ **for** $j = 1, \dots, o$ **do**

▷ For each row compute upper and lower bounds

 $t \leftarrow T(j, :)$ ▷ Extract row vector t ▷ $q_{25}(t)$ and $q_{75}(t)$ refer to the 25% and 75% percentile of the vector t , whereas IQR refers to the inter quantile range. $b_u \leftarrow q_{75}(t) + 1.5IQR(t)$ ▷ Compute upper bound b_u $b_l \leftarrow q_{25}(t) - 1.5IQR(t)$ ▷ Compute lower bound b_l **for** $k = 1, \dots, p$ **do**

▷ For each column

▷ Detect individual outliers if v is outside the bounds $v \leftarrow T(j, k)$ **if not** $(b_l \leq v \leq b_u)$ **then** $B(j, k) \leftarrow \text{true}$ **end if****end for****end for**▷ Sum up each column to obtain the result vector r **for** $k = 1, \dots, p$ **do** $r(k) \leftarrow \sum_{j=1}^o B(j, k)$ **end for****return** r

▷ The vector of outlierness values.

end procedure

This approach differs from the previously mentioned, since the goal is not to localize the position of the occurrence of an anomaly, but to detect whether a built column, an MVTs, is anomalous as a whole. In contrast to approaches (e.g., [4.25]) which apply ML on KPI data, in this framework the KPI data are used as a separate mean for anomaly detection and for the unsupervised training set construction. The learning algorithm, however, is applied on the recorded time-series data. In additionally, the available process knowledge should be incorporated into the hybrid anomaly detector to acquire explainable and physically consistent results.

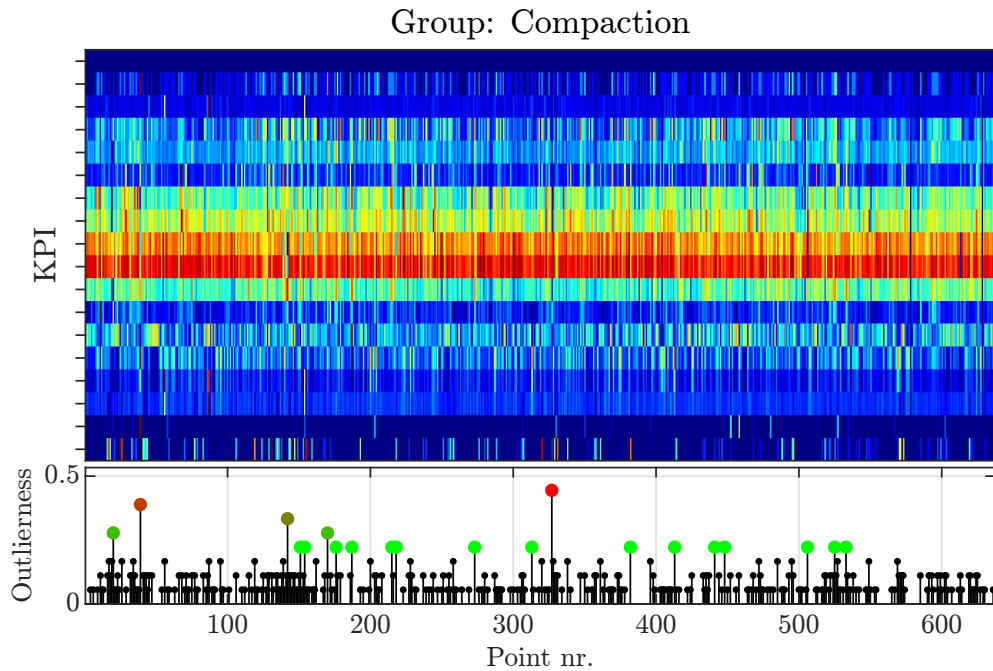


Figure 6: Heat-map and outlieriness for the KPI relevant to the compaction phase of the process, and the data are for all foundation columns on the building site. Top: Heat-map, whereby the horizontal dimension refers to the number of the individual points shown in Fig. 1; the KPIs are mapped to the vertical axes. The coloring of the elements on each row corresponds to the relative value of that KPI at each point. Bottom: The outlieriness for each foundation column.

5.1 Objective Based Pre-Processing

To meet different objectives, different preprocessing steps need to be performed to extract and enhance the special characteristics of the data. The objective can also be dependent on the subprocess monitored. In this section, the preprocessing steps regarding outlier detection in terms of the quality of the foundation columns are shown.

Feature Selection

Using the whole set of channels is in most cases not beneficial for the ML performance. A higher number of features used results in a higher number of learnable parameters, increased model complexity, and training time. The goal is to reduce the number of features used while preserving the relevant information and eliminating redundant features [4.26].

Despite using unsupervised ML, knowledge about the monitored system and the different recorded channels should be used when deriving the subset of channels. In this approach, the channels from which the work performed as function of time (see Figs. 4 and 5) can be derived were chosen. Depending on the goal that should be archived with ML, a different subset of channels may be required as input to the ML model. Two possible examples of goals that require a different

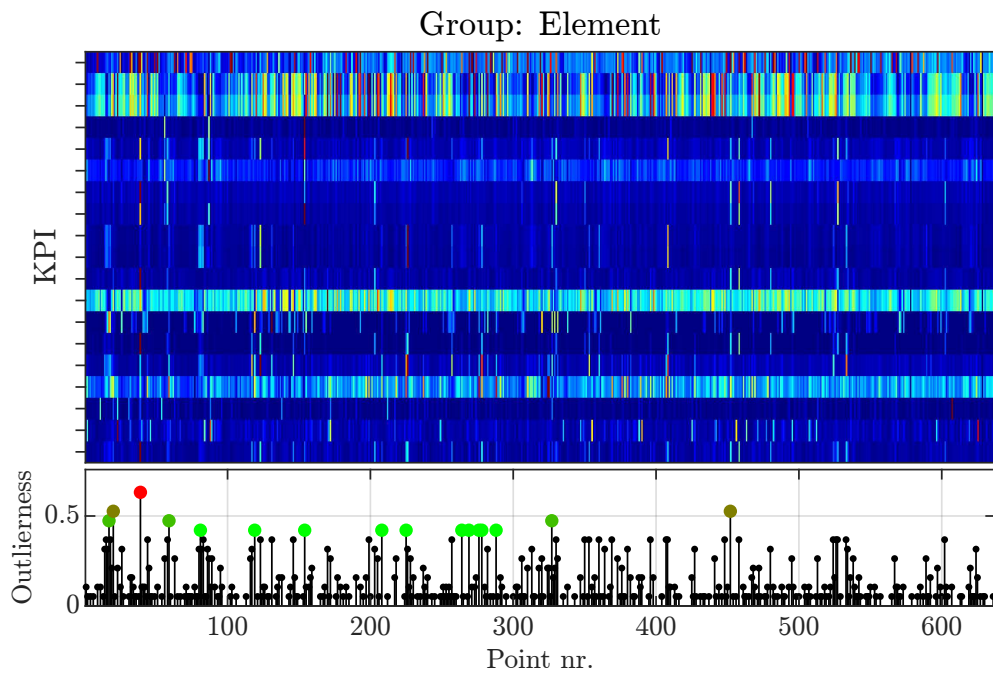


Figure 7: Heat-map and outlieriness for the KPI relevant to the complete element produced; see also Fig. 6

channel selection are anomaly detection regarding the foundation column quality or anomaly detection regarding the production efficiency of the ground improvement process.

Rule-Based Segmentation

Based on the definition of rules and events using expert knowledge, the MVTS are segmented into sub-processes. This is done because separate ML models are trained for the separate phases. The sub-processes are indicated in Figure 3 with different colors.

Data Trimming

The objective of anomaly detection for the case study is to detect anomalies in the sense of quality. The ground improvement process contains process pauses, e.g., logistical difficulties in gravel delivery. These pauses do not have an impact on the effectiveness of the performed process but the process efficiency. To meet our objective, these pauses when no drilling is performed, are removed from the MVTS used for ML. After trimming the data by removing the process pauses, in MVTS with anomalous process behavior discontinuities can occur. These discontinuities caused by operating errors and other process anomalies can be detected using appropriate techniques [4.27, 4.28]. This is done to use these anomalous cases in a knowledge discovery process. This is based on previous work published in [4.29].

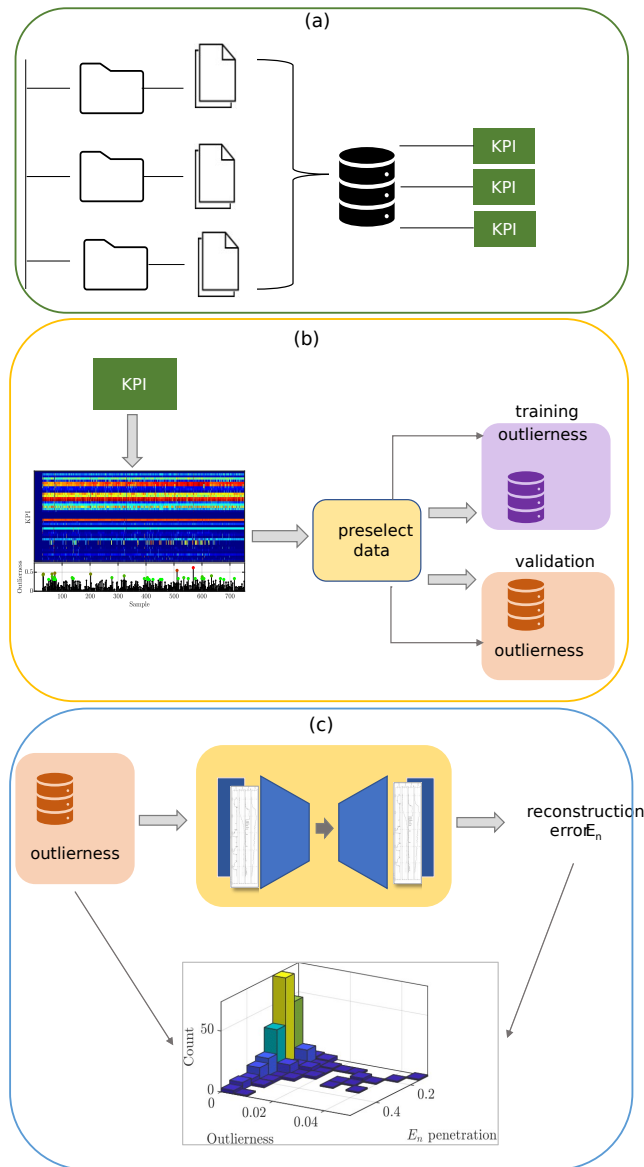


Figure 8: Interplay of the components of our hybrid framework. (a) Files are recorded at different sites and mapped to a storage location. From these data, the KPIs are calculated as part of the data-ingestion process. (b) From KPI visualizations—e.g., the heat-map is created and the outlieriness—rank of the MVTS—is calculated. Based on the outlieriness, the files are separated in a training and validation set. (c) Process of getting the autoencoder anomaly score for the data of the validation set using a trained ML model. By passing the validation set through the autoencoder, the reconstruction error can be calculated. In the end, the two anomaly scores per MVTS for the KPI-based anomaly detection and the autoencoder-based anomaly detection are combined in a bivariate histogram as a starting point for the knowledge discovery process.

5.2 Architecture Selection

An autoencoder is an ML structure performing two mappings [4.30].

1. Encoder: Mapping Ψ from the original data to the latent variables in the latent space.
2. Decoder: Mapping Σ from the latent space to the reconstructed signal.

The goal is to extract latent variables that allow for a minimal amount of distortion in the reconstructed signal [4.30, 4.31]. The dimension of the latent space n_z is for anomaly detection chosen to be $n_z < b$, b being the dimension of the encoded data [4.32]. Autoencoders and variational autoencoders (VAEs), which differ in the regularization of their cost function, were successfully used for anomaly detection in time-series in the past [4.33]. A schematic representation of the ML architecture is shown in Figure 9.

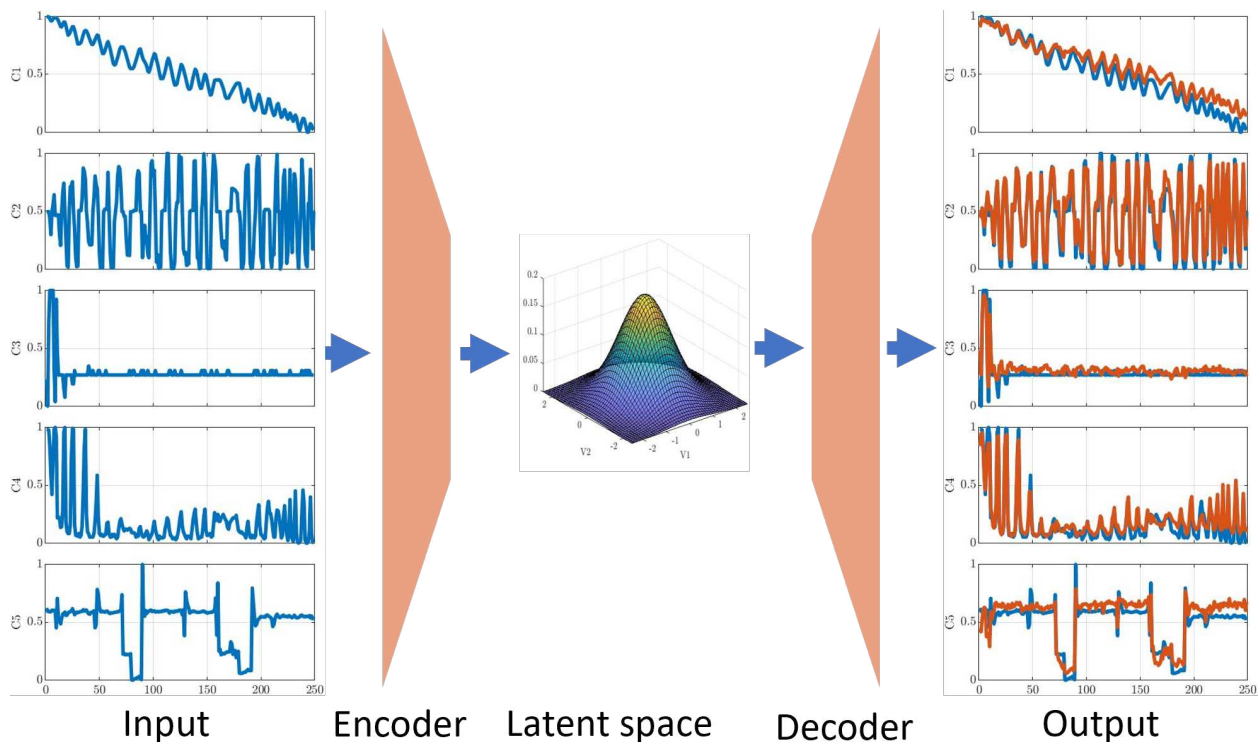


Figure 9: Schematic representation of an autoencoder with the encoder that encodes the input features into latent variables and the decoder that reconstructs the lower dimensional representation available in the latent space.

The assumption behind anomaly detection using VAEs is performing some kind of dimensionality reduction and the assumption that anomalies contain non-representative features which cannot be encoded into the lower dimensional space [4.34, 4.35]. As described previously, the three-channel MVTs used for ML is converted into a lower dimensional latent representation. The degree of

dimensionality reduction is determined by the dimension of the latent space, the latent dimension n_z , in this work $n_z = 1$ was chosen to archive a sufficiently high degree of dimensionality reduction.

A recurrent neural network (RNN) was incorporated into both, the encoder and the decoder, to handle the sequential data [4.36], namely a long-short term memory (LSTM) [4.37] layer. Unidirectional LSTM is processing the input data in the forward direction, whereas bidirectional LSTM (BiLSTM) is processing it in a forward and backward direction [4.38]. Bidirectional models can only be used when the whole data sample that they are applied to is available [4.36]. This condition is fulfilled in the use-case presented here, since the data analyzed are not streaming data. BiLSTM performs well, especially when the value of time-step t depends on both, prior and past time-steps [4.39]. This is the case in boundary value problems. The drilling process described in this article can be seen as a (homogeneous) boundary value problem since the drilling starts at the surface and ends at the surface.

5.3 KPI-Supervised Training

As described in Section 5, most of the data, gathered with physical systems, are unlabeled and reliable labeling is hard to archive [4.21]. Because of these circumstances, an unsupervised anomaly detection approach was chosen. Unsupervised anomaly detection is done under the assumption that the training set $\Gamma \subset \Phi$, whereas Φ is the set of all available samples, contains mostly samples from normal operations but could contain a small number of still unknown anomalies [4.13, 4.40]. This training set is constructed using the outlierness calculated with the KPI [4.13] for this reason, and we call it *KPI-supervised training* as the anomalous samples are excluded and autolabeled using the outcome of the KPI classifier. This excludes the already known anomalies covered by the KPI framework.

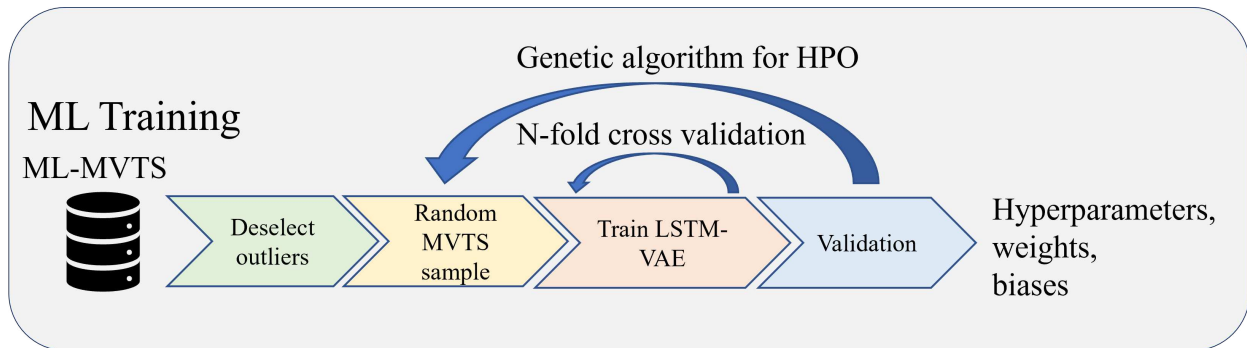


Figure 10: Process from ML-MVTS to trained ML-model incorporating HPO.

The training set is constructed using the outcome of the KPI anomaly detection. The proposed framework does not require human labeling, since it uses the results gained with the KPI classifier to construct a training set only consisting of MVTS which are considered to be non-anomalous. The whole process of preprocessing and training is illustrated in Figure 10

5.4 Length Preserving Training

The training procedure based on stochastic gradient descent was adjusted to apply on MVTS of different time-series lengths and minimize the effects of resampling to a common length which is required for gradient calculation. During training, the samples are sorted according to their time-series length and only resampled to a common length, downsampled to the shortest length of the samples forming a mini-batch; see Figure 11. After training the autoencoder in this new way it can reconstruct samples while preserving their real length. This is desirable since an unusual time-series length, the process length of an industrial process, can be a sign of anomalous process execution. This avoids the necessity to resample the time-series to a common time-series length [4.41,4.42] or to pad the data with zeros to the length of the longest time-series [4.43,4.44], which adds information and has an impact on the learning performance, especially when used with LSTM. [4.45].

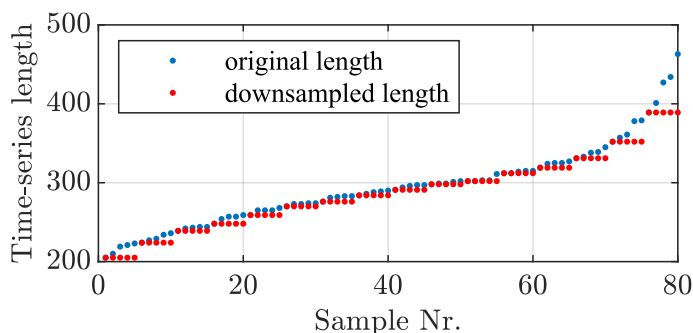


Figure 11: Samples forming a mini-batch of the training dataset are downsampled to the shortest time-series length of the mini-batch and the training dataset is not resampled to a common time-series length.

The cost function optimized with stochastic gradient descent is a regularized [4.46] version of the least-squares cost function minimizing the reconstruction error. Being Y_k the k -th sample of a mini-batch and \hat{Y}_k the reconstructed MVTS, i.e., the output of the autoencoder the reconstruction error E_r of a mini-batch of size m is given as³

$$E_r = \sum_{k=1}^m \|Y_k - \hat{Y}_k\|_F^2. \quad (4.8)$$

In the VAE the regularization is based on the Kullback-Leibler divergence⁴ between two distributions Ω and Π given as $D_{KL}(\Omega||\Pi)$ [4.47,4.48]. The loss function of a VAE can be written as [4.47,4.48]

$$E_t = E_r - \beta D_{KL}(\Omega||\Pi), \quad (4.9)$$

³ $\|A\|_F$ denotes the Frobenius norm of the matrix A .

⁴The Kullback-Leibler divergence is not symmetric and therefore cannot be used as a measure.

with Ω being a prior chosen by the user, often a multivariate Gaussian distribution with zero mean and a diagonal covariance matrix. The second distribution Π is the learned distribution in the latent space. In the formulation introduced by [4.48] $\lambda = 1$, in this case, the objective of the learning algorithm is minimizing the *evidence lower bound* (ELBO) [4.47]. In our approach we use a so-called β -VAE [4.49] where the Lagrangian multiplier β is a hyperparameter that is optimized during hyperparameter optimization (HPO), described in Section 5.5, to ensure that the balancing between the two terms of the loss function is optimal.

5.5 Hyperparameter Optimization

The hyperparameters have a significant impact on the performance of the network [4.50]. The optimal performance of an ML architecture on a dataset is obtained by optimizing the hyperparameters. The available hyperparameters depend on the architecture chosen and their optimal value on the problem and the corresponding data. The search space of the HPO grows exponentially with the number of parameters optimized. This combined with the long training times of ML architectures mostly heuristics or meta-heuristics are used [4.36, 4.43].

In this framework, a genetic algorithm is used for the HPO and the following hyperparameters were optimized,

1. Trained epochs: $e \in [50, 150]$
2. Mini-batch size $s \in [2, n]$
3. Number of neurons encoder $n_{ne} \in [10, 100]$
4. Number of neurons decoder $n_{nd} \in [10, 100]$
5. Learning rate $\alpha \in [0.003, 0.1]$
6. Regularization factor $\beta \in [0, 50]$.

The genetic algorithm adjusted for this framework is described in more detail in [4.51] and was executed with the following settings: 15 individuals per generation, threefold cross-validation for the fitness estimation, and a maximum of ten generations. Moreover, a combination of two crossover functions was used to obtain a better exploration of the search space. ⁵

5.6 Anomaly Detection Using Autoencoders

The last step of anomaly detection using autoencoders is performing the binary classification task. In this step, the model trained on the subset Γ is applied to the whole available dataset Φ .

⁵Further information on the genetic algorithm and implementation details can be found in [4.51].

Table 4.3: Results of the HPO

Architecture	sub-process	e	n_{ne}	n_{nd}	$\alpha * 10^5$	s	β
VAE-LSTM	penetration	69	40	56	1998	11	5
VAE-LSTM	compaction	132	39	47	3689	15	8
VAE-BiLSTM	penetration	117	28	39	5987	17	9
VAE-BiLSTM	compaction	132	39	47	3689	15	8

When using autoencoders, the output is the reconstructed signal from which the reconstruction error can be derived [4.33,4.44,4.52]. From the reconstruction error, the anomaly score is calculated. The anomaly score used here for the sample y with a time-series length t is calculated as:

$$E_a = \sum_{c=1}^t \|y^{(c)} - \hat{y}^{(c)}\|_F \quad (4.10)$$

with $y^{(c)}$ denoting the measurement taken at time-step c and $\hat{y}^{(c)}$ the reconstructed measurement at time-step c . Because of the changing time-series length, the anomaly score E_a is normalized by the number of time-steps t , which leads to a normalized anomaly score E_n

$$E_n = \frac{E_a}{t}. \quad (4.11)$$

To get the classification, a threshold u is needed to represent the boundary between the classes. Since the distribution of reconstruction errors is positive semi-definite, a skewness adjustment needs to be performed [4.53–4.55]. The MVTS with $E_n < u$ get assigned to the class of nonanomalous samples and MVTS with $E_n \geq u$ are marked as anomalous.

6 Architecture Comparison and Results

The novel hybrid framework was applied to the case study introduced in Section 2. Separate autoencoders were trained for the analyzed sub-processes, compaction, and penetration. The following architectures were compared.

1. VAE-LSTM: VAE with one LSTM layer in the encoder and the decoder.
2. VAE-BiLSTM: VAE with one Bi-LSTM layer in the encoder and decoder.

6.1 Hyperparameter Optimization Results

The HPO was executed four times to cover all the two compared architectures; the two phases and the results are shown in Table 4.3.

6.2 Architecture Comparison

In this section, different architectures are evaluated phase-wise with metrics based on the anomaly score which is derived from the reconstruction error of trained autoencoders. Because of the nature of ML problems, random initialization of learnable parameters, and an overdetermined solution space, their performance varies when executing them multiple times on the same dataset using the same hyperparameters. These observations were previously described in [4.51]. The visualization

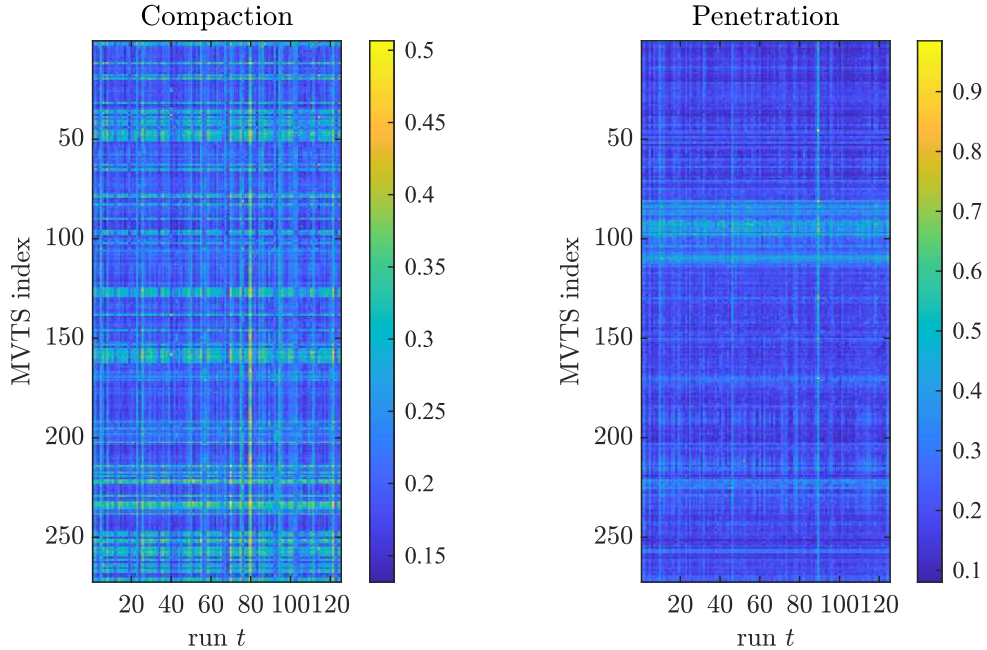


Figure 12: Visualization of the fluctuation in the anomaly scores E_r of MVTS of a site when the training procedure is repeated $r = 125$ times for the sub-processes compaction (left) and penetration (right).

of the anomaly score of the trained architectures is shown in Figure12. The performance fluctuation of the training on the same data with the same hyperparameters can be observed on the vertical lines. The horizontal lines indicate that the sample was reconstructed with a similar anomaly score in the majority of the t trained autoencoders.

To get an estimate of the performance of the architectures, $r = 125$ instances of each ML architecture were trained and then an outlier detection was performed. The goal was to identify the best architecture in terms of reliability, the variance of the outcome should be low, and at the same time, the distance between the two groups should be maximized. The obtained results are shown in Table 4.4.⁶

⁶ E_{nn} indicates the normalized anomaly score of samples that are assigned to the class non-anomalous; correspondingly, E_{na} indicates the normalized anomaly score of samples that are assigned to the class anomalous.

Table 4.4: Subprocesswise ML architecture comparison

statistical quantity	penetration		compaction	
	VAE-LSTM	VAE-BiLSTM	VAE-LSTM	VAE-BiLSTM
$mean(mean(E_{nn}))$	0.1997	0.2175	0.1986	0.2319
$median(median(E_{nn}))$	0.1883	0.1992	0.1966	0.2247
$mean(mean(E_{na}))$	0.3918	0.4303	0.3161	0.2880
$median(median(E_{na}))$	0.3821	0.4053	0.2353	0.2483
$mean(distance\ means)$	0.1921	0.2128	0.1251	0.0550
$median(distance\ medians)$	0.1932	0.2110	0.0516	0.0275

The results of the comparison of the ML architectures as shown in Table 4.4 are interpreted as follows.

1. The mean of the means of the reconstruction errors of samples classified by the autoencoder architectures as nonanomalous is lower for VAE-LSTMs in both the subprocesses; the corresponding results occur calculating the median of medians.
2. The mean of the means of the anomaly scores of samples classified by the autoencoder architectures as anomalous is higher for BiLSTM in the penetration subprocess. This can also be seen in the box-plot shown in Figure13. A higher mean of the distance between nonanomalous and anomalous samples is desirable, since it makes the two groups better distinguishable.

Based on the results, the following selection of architectures was made: for the two subprocesses, different architectures were chosen; for penetration, autoencoders incorporating BiLSTM layers outperformed autoencoders using LSTM layers; for compaction, the architectures using LSTM layers performed significantly better.

6.3 Subprocesswise Anomaly Detection

The bivariate histogram shown in Figure 14 indicates that a high anomaly score in one of the two phases does not correlate with a high anomaly score in the other phase. A correlation coefficient $r = -0.0628 \ll -0.3$ indicates a very weak negative linear relationship [4.56] between the anomaly scores of the two phases. This is consistent with the experts' institution of this case study; if the penetration phase is anomalous, this does not result in an anomalous compaction sub-process.

6.4 Results KPI-Supervised Training

As described in Section 5.3, the training sets were constructed using the outlieriness. However, when analyzing the distribution of the anomaly scores of both the groups, MVTS which are

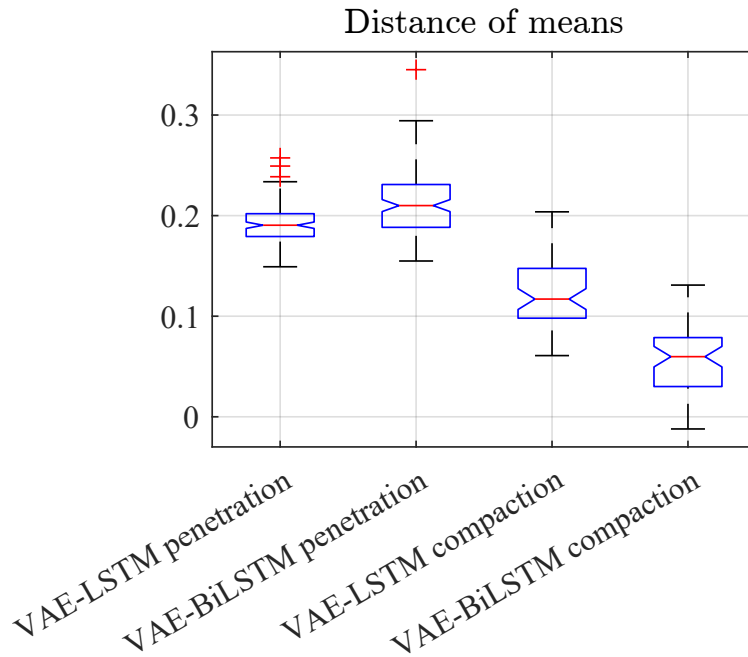


Figure 13: Box-plot showing the distance of the means of the two classes; nonanomalous and anomalous; for the compared architectures.

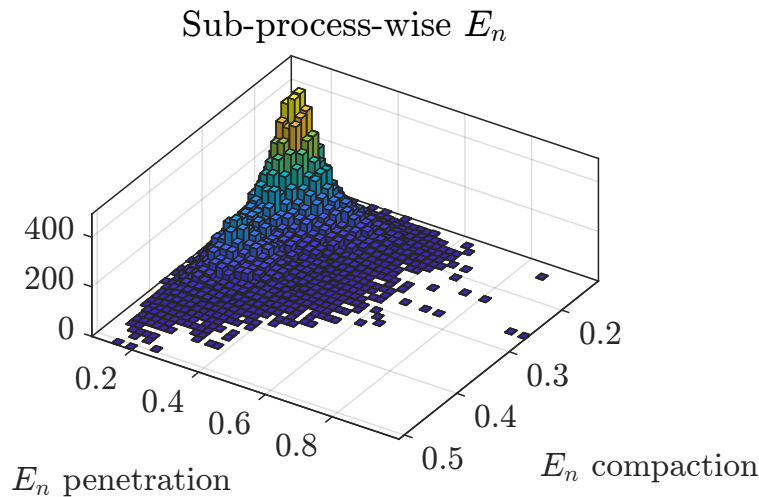


Figure 14: Bivariate histogram of the anomaly score E_n for the two sub-processes gained by training a population of $t = 125$ autoencoders with optimized hyperparameters for each of the subprocesses.

considered, based on the KPI analysis, to be anomalous and those considered to be non-anomalous, the distributions shown in Figure 15 of the anomaly score were obtained. It can be seen that the outcome is two nearly identical probability density functions. Based on these results the KPI-supervised training needs further refinement to produce meaningful, distinctive results. This could be caused by the KPI covering all the sensor channels or by the fact that the KPI classifier and

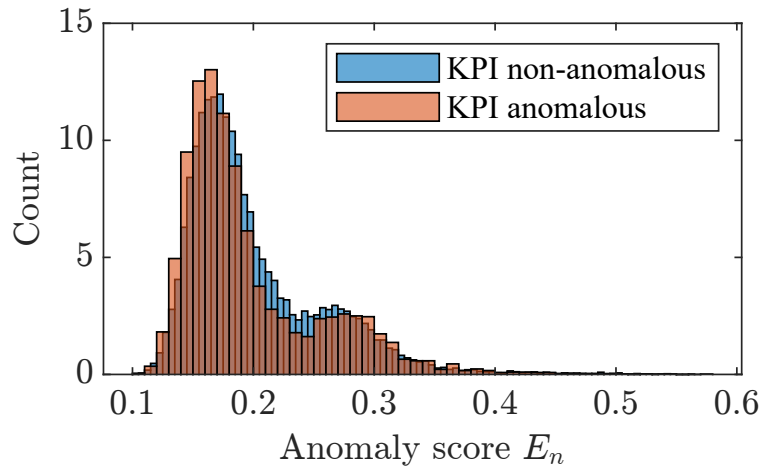


Figure 15: The probability density function of the reconstructions of a trained VAE-BiLSTM architecture for the compaction phase; the two sets of MVTs were constructed using the KPI classifier.

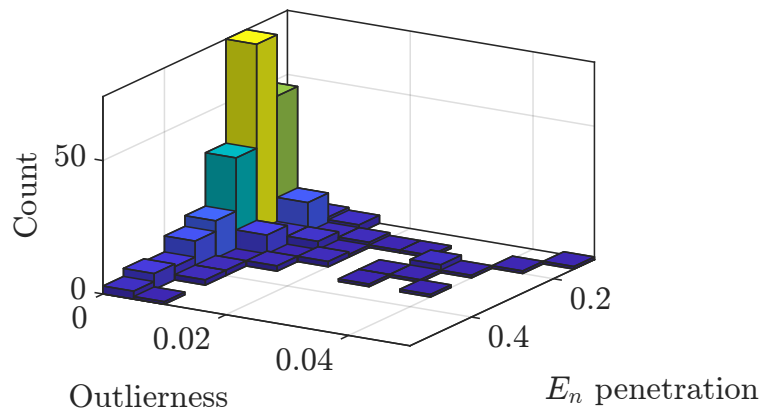


Figure 16: Bivariate histogram of the anomaly score E_n for a trained VAE-BiLSTM and the KPI outlierness for the penetration phase.

autoencoder anomaly detection cover other anomalies; see Figure 16.

However, summed up it can be said that a high reconstruction error of an optimized and trained autoencoder does not lead to a high outlierness and vice versa. This can be seen in Figure 16, where the best performing architecture for compaction (VAE-BiLSTM) was trained with optimized hyperparameters to obtain the anomaly score and the KPIs covering the penetration phase were used to calculate the outlierness. As already reported in [4.14], the two methods identify a common set of MVTs having both low anomaly scores and low outlierness. Based on these, two means of anomaly detection two classifications can be derived and combined to cover a wider variety of anomalies; since the application shown is safety-relevant, every additionally detected anomaly is strengthening the process safety.

7 Conclusion

A new framework for the handling of MVTs data recorded in industrial applications was presented. The presented framework standardizes the data handling and makes the HML classifier applicable to a wide variety of use cases. Multiple technical KPI that capture physical knowledge were developed. Furthermore, additional refinements of (Bi-)LSTM autoencoders were performed to make them better suitable for MVTs data with varying lengths. This permits the construction of ML architectures that preserve the real length of the MVTs after applying ML.

A new approach of unsupervised training set construction using the outcome of the KPI anomaly detection was investigated, which needs further refinement. A comparison of LSTM-VAE and BiLSTM-VAE was performed and led to using different recurrent layers for the present subprocesses of the investigated case study. It can be concluded that training separate ML models for the subprocesses is beneficial since the anomaly scores of the two phases are not correlated and it also gives insights into which of the subprocesses the recorded data is anomalous.

Further work should be done in the direction of unsupervised training set construction and evaluation metrics for unsupervised ML architectures and training functions which are robust against outliers in the training set. The comparison shows that different architectures of different sizes are optimal for the different subprocesses. It should be noted that it is concluded after this study that due to the extensive knowledge discovery process and the derived preprocessing techniques only a low number of not excluded anomalies is still present in the compaction phase. One indicator for that is the overall low number of detected anomalies and the overall low anomaly score of the subprocess.

Acknowledgment

This work was supported by the COMET Program within the K2 Center “Integrated Computational Material, Process and Product Engineering (IC-MPPE)” (this program is supported by the Austrian Federal Ministries for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK) and for Labor and Economy (BMAW), represented by the Austrian Research Promotion Agency (FFG), and the Federal States of Styria, Upper Austria, and Tyrol) under Project 886385.

Bibliography

- [4.1] C. Alippi, A. Ferrero, and V. Piuri, “Artificial intelligence for instruments and measurement applications,” *IEEE Instrumentation and Measurement Magazine*, vol. 1, no. 2, pp. 9–17, 1998.
- [4.2] A. Morato, S. Vitturi, F. Tramarin, and A. Cenedese, “Assessment of Different OPC UA Industrial IoT Solutions for Distributed Measurement Applications,” in *2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2020, pp. 1–6.

- [4.3] M. Raissi, P. Perdikaris, and G. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686 – 707, 2019.
- [4.4] M. Khanafer and S. Shirmohammadi, “Applied AI in instrumentation and measurement: The deep learning revolution,” *IEEE Instrumentation & Measurement Magazine*, vol. 23, no. 6, pp. 10–17, 2020.
- [4.5] S. Shirmohammadi and H. Al Osman, “Machine learning in measurement part 1: Error contribution and terminology confusion,” *IEEE Instrumentation & Measurement Magazine*, vol. 24, no. 2, pp. 84–92, 2021.
- [4.6] H. Al Osman and S. Shirmohammadi, “Machine learning in measurement part 2: uncertainty quantification,” *IEEE Instrumentation & Measurement Magazine*, vol. 24, no. 3, pp. 23–27, 2021.
- [4.7] A. Rashed and S. Shirmohammadi, “A novel method to estimate measurement error in AI-assisted measurements,” in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE, 2022, pp. 1–5.
- [4.8] O. Rippel, N. Schönfelder, K. Rahimi, J. Kurniadi, A. Herrmann, and D. Merhof, “Panoptic segmentation of animal fibers,” in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE, 2022, pp. 1–6.
- [4.9] P. Assumpção, C. Oliveira, W. Melo, and L. Carmo, “Sensors fingerprints using machine learning: a case study on dam monitoring systems,” in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE, 2022, pp. 1–6.
- [4.10] P. R. Cohen and A. E. Howe, “How Evaluation Guides AI Research: The Message Still Counts More than the Medium,” *AI Magazine*, vol. 9, no. 4, p. 35, Dec. 1988.
- [4.11] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The m4 competition: 100,000 time series and 61 forecasting methods,” *International Journal of Forecasting*, vol. 36, no. 1, pp. 54–74, 2020.
- [4.12] R. Wu and E. Keogh, “Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress,” *IEEE Transactions on Knowledge and Data Engineering*, 2021.
- [4.13] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, “Do Deep Neural Networks Contribute to Multivariate Time Series Anomaly Detection?” *ArXiv e-prints*, Apr. 2022.

- [4.14] A. Terbuch, P. O’Leary, and P. Auer, “Hybrid machine learning for anomaly detection in industrial time-series measurement data,” in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2022, pp. 1–6.
- [4.15] P. Nagy and D. Adam, “Quality control of deep vibro compaction based on the vibrator movement,” in *Proceedings of the XVII European Conference on Soil Mechanics and Geotechnical Engineering, Reykjavik, Island*, 2019.
- [4.16] R. S. Pugh, “Settlement of floor slabs on stone columns in very soft clays,” *Proceedings of the Institution of Civil Engineers - Geotechnical Engineering*, Jan. 2017.
- [4.17] K. Kirsch and F. Kirsch, *Ground Improvement by Deep Vibratory Methods*. Andover, England, UK: Taylor & Francis, 2017.
- [4.18] H. S. Mavikumbure, C. Wickramasinghe, D. Marino, V. Cobilean, and M. Manic, “Anomaly Detection in Critical-Infrastructures using Autoencoders: A Survey,” in *IECON 2022 – 48th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, Oct. 2022, pp. 1–7.
- [4.19] M. Folk, A. Cheng, and K. Yates, “HDF5: A file format and I/O library for high performance computing applications,” in *Proceedings of Supercomputing*, vol. 99, 1999, pp. 5–33.
- [4.20] J. Dean and S. Ghemawat, “Mapreduce: simplified data processing on large clusters,” *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008.
- [4.21] I. Steinwart, D. Hush, and C. Scovel, “A classification framework for anomaly detection,” *Journal of Machine Learning Research*, vol. 6, no. 8, pp. 211–232, 2005.
- [4.22] L. Zhou, Q. Zeng, and B. Li, “Hybrid Anomaly Detection via Multihead Dynamic Graph Attention Networks for Multivariate Time Series,” *IEEE Access*, vol. 10, pp. 40 967–40 978, Apr. 2022.
- [4.23] M. Munir, S. A. Siddiqui, M. A. Chattha, A. Dengel, and S. Ahmed, “FuseAD: Unsupervised Anomaly Detection in Streaming Sensors Data by Fusing Statistical and Deep Learning Models,” *Sensors*, vol. 19, no. 11, p. 2451, May 2019.
- [4.24] Z. Ji, J. Gong, and J. Feng, “A Novel Deep Learning Approach for Anomaly Detection of Time Series Data,” *Scientific Programming*, vol. 2021, Jul. 2021.
- [4.25] Y. Zhao, X. Zhang, Z. Shang, and Z. Cao, “DA-LSTM-VAE: Dual-Stage Attention-Based LSTM-VAE for KPI Anomaly Detection,” *Entropy (Basel, Switzerland)*, vol. 24, no. 11, p. 1613., Nov. 2022.
- [4.26] R.-C. Chen, C. Dewi, S.-W. Huang, and R. E. Caraka, “Selecting critical features for data classification based on machine learning methods,” *J. Big Data*, vol. 7, no. 1, pp. 1–26, Dec. 2020.

- [4.27] D. Ninevski and P. O’Leary, “A convolutional method for the detection of derivative discontinuities,” in *2020 21th International Carpathian Control Conference (ICCC)*, 2020, pp. 1–5.
- [4.28] ———, “Detection of derivative discontinuities in observational data,” in *International Symposium on Intelligent Data Analysis*. Springer, 2020, pp. 366–378.
- [4.29] A. Terbuch, A. Zöhrer, V. Winter, P. O’Leary, N. Khalili-Motlagh-Kasmaei, and G. Steiner, “Quality monitoring in vibro ground improvement – A hybrid machine learning approach,” *Geomechanics and Tunnelling*, vol. 15, no. 5, pp. 658–664, Oct. 2022.
- [4.30] P. Baldi, “Autoencoders, unsupervised learning, and deep architectures,” in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, ser. Proceedings of Machine Learning Research, I. Guyon, G. Dror, V. Lemaire, G. Taylor, and D. Silver, Eds., vol. 27. Bellevue, Washington, USA: PMLR, 02 Jul 2012, pp. 37–49. [Online]. Available: <https://proceedings.mlr.press/v27/baldi12a.html>
- [4.31] D. E. Rumelhart and J. L. McClelland, *Learning Internal Representations by Error Propagation*, ser. A Bradford book. MIT Press, 1987, pp. 318–362.
- [4.32] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge and London: MIT Press, 2016.
- [4.33] D. Park, Y. Hoshi, and C. C. Kemp, “A Multimodal Anomaly Detector for Robot-Assisted Feeding Using an LSTM-Based Variational Autoencoder,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1544–1551, 2 2018.
- [4.34] T. Kieu, B. Yang, and C. S. Jensen, “Outlier detection for multidimensional time series using deep neural networks,” in *2018 19th IEEE International Conference on Mobile Data Management (MDM)*. Aalborg: IEEE, 06 2018, pp. 125–134.
- [4.35] T. Amarbayasgalan, V. H. Pham, N. Theera-Umpon, and K. H. Ryu, “Unsupervised Anomaly Detection Approach for Time-Series in Multi-Domains Using Deep Reconstruction Error,” *Symmetry*, vol. 12, no. 8, 08 2020.
- [4.36] J. F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, and A. Troncoso, “Deep Learning for Time Series Forecasting: A Survey,” *Big Data*, vol. 9, no. 1, pp. 3–21, Feb. 2021.
- [4.37] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 11 1997.
- [4.38] F. U. M. Ullah, A. Ullah, I. U. Haq, S. Rho, and S. W. Baik, “Short-term prediction of residential power energy consumption via cnn and multi-layer bi-directional lstm networks,” *IEEE Access*, vol. 8, pp. 123 369–123 380, 2020.

- [4.39] S. Siami-Namini, N. Tavakoli, and A. S. Namin, “The performance of lstm and bilstm in forecasting time series,” *2019 IEEE International Conference on Big Data (Big Data)*, pp. 3285–3292, 2019.
- [4.40] A. Garg, W. Zhang, J. Samaran, R. Savitha, and C.-S. Foo, “An evaluation of anomaly detection and diagnosis in multivariate time series,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 6, pp. 2508–2517, 2021.
- [4.41] Z. C. Lipton, D. C. Kale, C. Elkan, and R. C. Wetzel, “Learning to diagnose with LSTM recurrent neural networks,” in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2016.
- [4.42] W. Zhang, Z. Lin, and X. Liu, “Short-term offshore wind power forecasting - A hybrid model based on Discrete Wavelet Transform (DWT), Seasonal Autoregressive Integrated Moving Average (SARIMA), and deep-learning-based Long Short-Term Memory (LSTM),” *Renewable Energy*, vol. 185, pp. 611–628, Feb. 2022.
- [4.43] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, “Multi-Sequence LSTM-RNN Deep Learning and Metaheuristics for Electric Load Forecasting,” *Energies*, vol. 13, no. 2, p. 391, Jan. 2020.
- [4.44] M. Thill, W. Konen, H. Wang, and T. Bäck, “Temporal convolutional autoencoder for unsupervised anomaly detection in time series,” *Applied Soft Computing*, vol. 112, 2021.
- [4.45] M. Dwarampudi and N. V. S. Reddy, “Effects of padding on LSTMs and CNNs,” *ArXiv e-prints*, Mar. 2019.
- [4.46] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*. Springer Netherlands, 2000.
- [4.47] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” *ArXiv e-prints*, Dec. 2013.
- [4.48] D. P. Kingma and M. Welling., *An Introduction to Variational Autoencoders*. New York: Now Publishers, 2019.
- [4.49] I. Higgins, L. Matthey, A. Pal, C. P. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner, “beta-vae: Learning basic visual concepts with a constrained variational framework,” in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. [Online]. Available: <https://openreview.net/forum?id=Sy2fzU9gl>

- [4.50] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A search space odyssey,” *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.
- [4.51] A. Terbuch, “LSTM hyperparameter optimization: Impact of the selection of hyperparameters on machine learning performance when applied to time series in physical systems,” Master’s thesis, Chair of Automation, Montanuniversitaet Leoben, 2021.
- [4.52] M. Liu, Z. Xu, and Q. Xu, “DeepFIB: Self-Imputation for Time Series Anomaly Detection,” *ArXiv e-prints*, Dec. 2021.
- [4.53] E. Vandervieren and M. Hubert, “An adjusted boxplot for skewed distributions,” in *COMP-STAT 2004: Proceedings in Computational Statistics*. Prague: Springer-Verlag, 2004, pp. 1933–1940.
- [4.54] M. Hubert and S. Van der Veeken, “Outlier detection for skewed data,” *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 22, no. 3-4, pp. 235–246, 2008.
- [4.55] M. Hubert and E. Vandervieren, “An adjusted boxplot for skewed distributions,” *Computational Statistics & Data Analysis*, vol. 52, no. 12, pp. 5186–5201, 2008.
- [4.56] B. Ratner, “The correlation coefficient: Its values range between +1/-1, or do they?” *J. Target. Meas. Anal. Mark.*, vol. 17, no. 2, pp. 139–142, Jun. 2009.



Anika Terbuch received the B.Sc. and M.Sc. degrees in industrial logistics from Montanuniversität Leoben, Leoben, Austria, in 2019 and 2021, respectively.

She is currently a University Assistant with the Chair of Automation, Montanuniversität Leoben, Leoben, Austria. Her research focuses on embedding physical a priori knowledge into hybrid machine learning architectures.

Paul O’Leary received the B.A. degree in mathematics and the B.A.I. degree in engineering from Trinity College Dublin, Dublin, Ireland, both in 1982, the M.Sc. degree in electronic engineering from PII, Eindhoven, The Netherlands, in 1984, and the Ph.D. degree from the University of Pavia, Pavia, Italy, in 1991.

He is currently a Professor of automation with the University of Leoben, Leoben, Austria. His research focuses on mathematical methods to extract information, knowledge, and understanding from data.



Negin Khalili-Motlagh-Kasmaei received the Master's degree in urban engineering from Azad University, Zanjan, Iran, in 2015. She is currently pursuing the Ph.D. degree with the Chair of automation, Montanuniversität Leoben. She is working as a Project Assistant at the Chair of automation, Montanuniversität Leoben. Her research focuses on multivariate time-series data analysis, anomaly detection, rule-based key performance indicators, and developing observational digital twins.



Peter Auer received the Ph.D. degree in mathematics from the Vienna University of Technology, Vienna, Austria, in 1992.

From 1988 to 1992, he was a Researcher at the Institute for Statistics and Probability Theory and the Institute for Applied Computer Science, Vienna University of Technology. From 1992 to 2002, he was an Assistant Professor and later Associate Professor at the Institute for Theoretical Computer Science, Graz University of Technology, Graz, Austria. From 1995 to 1996, he was a Visiting Research Scholar at the University of California at Santa Cruz, Santa Cruz, CA, USA. Since 2003, he is a Full Professor of information technology with the Montanuniversität Leoben, Leoben, Austria.

Dr. Auer is a member of the ACM. He is an Action Editor of the Journal of Machine Learning Research and an Editorial Board Member of the Machine Learning Journal.



Alexander Zöhrer received the master's degree in civil engineering and the Ph.D. degree from the Graz University of Technology, Graz, Austria, in 2003 and 2006, respectively. In 2007, he joined Company Keller Grundbau Ges.m.b.H., Vienna, Austria, as a Design Engineer. Since 2015, he has been the Head of the Engineering Department of the Business Unit South-East Europe and Nordics. In this position he is responsible for any research and development activities as well as the engineering support including design calculations for all construction sites. Dr. Zöhrer is a member of the EFFC Sustainability Working Group as well as a member of various global working groups at Keller.



Vincent Winter received the M.Sc. degree in civil engineering from the Vienna University of Technology, Vienna, Austria, in 2002. Since 2000, he has been working in various positions with Company Keller Grundbau Ges.m.b.H., Vienna, starting as a site manager and later on as a project manager for sites of all special foundation works. Currently, he is a Product Line Manager for Vibro Techniques for Keller South East Europe and Nordics, supporting big and difficult projects and being responsible for the development of the Vibro Techniques at Keller. As an expert for vibro ground improvement techniques, he was involved in many national as well as a number of international projects.

Application of Hybrid Machine Learning Based Quality Control in Daily Site Management

Alexander Zöhrer¹, Vincent Winter¹, Anika Terbuch²,
Paul O’Leary², Negin Khalili-Motlagh-Kasmaei²

¹ Keller Grundbau GesmbH, Vienna and Söding, Austria
{alexander.zoehrer, vincent.winter}@keller.com

² Chair of Automation, Montanuniversität Leoben, Leoben, Austria
{anika.terbuch, paul.oleary, negin.khalilimotlaghkasmaei}@unileoben.ac.at

Originally appeared as:

A. Zöhrer, V. Winter, A. Terbuch, P. O’Leary, N. Khalili-Motlagh-Kasmaei,
”Application of Hybrid Machine Learning Based Quality Control in Daily Site
Management,” 15th ISRM Congress 2023 & 72nd Geomechanics Colloquium,
Schubert & Kluckner (eds), Salzburg, Austria, October 9-14, 2023, pp. 569-574,
ISBN 978-3-9503898-3-8

15th ISRM Congress 2023 & 72nd Geomechanics Colloquium. Schubert & Kluckner (eds) © ÖGG
Copyright 2023: Österreichische Gesellschaft für Geomechanik, Innsbrucker Bundesstraße 67, 5020
Salzburg, Austria, P.: 0043 662 875519, F.: 0043 662 886748, E.: salzburg@oegg.at, Web:
<https://www.oegg.at/en/>

Application of hybrid machine learning based quality control in daily site management

Alexander Zöhrer ¹, Vincent Winter ¹, Anika Terbuch ²,
Paul O’Leary ², Negin Khalili-Motlagh-Kasmaei ²

¹Keller Grundbau GesmbH
Vienna and Söding, Austria
{alexander.zoehrer, vincent.winter}@keller.com

²Chair of Automation,
Montanuniversität Leoben,
Leoben, Austria
{anika.terbuch, paul.oleary, negin.khalilimotlaghkasmaei}@unileoben.ac.at

IEEE Transactions on Instrumentation and Measurement, Vol. 72, 2023
Digital Object Identifier 10.1109/TIM.2023.3236354

15th ISRM Congress 2023 & 72nd Geomechanics Colloquium. Schubert & Kluckner (eds) ©ÖGG

Abstract

This paper presents a system that combines KPI with autoencoders to implement a hybrid machine learning system. The goal here is to investigate workflows which permit the site manager to use the hybrid machine learning systems as a decision support tool. The workflows are explained by means of case studies, demonstrating the application of the hybrid system to detect both element as well as site related quality issues. In addition to that, the detection of anomalies regarding execution efficiency assist the project manager to optimize the sequence of work on site.

Keywords: ground improvement, quality control, machine learning, key performance indicators

1 Introduction

The evaluation and analysis of machine data to support project managers in their daily business was introduced by Keller more than 10 years ago [4.1]. Improving of the quality assurance procedures was a focus right from the start. Falk et al. [4.2] describe the development and challenges of machine data acquisition and the geotechnical interpretation of this data. Key performance indicators (KPI) were introduced with goal of characterizing soil and execution parameters; these can be calculated directly from the machine data. The tool named “VibroScan” was the result of this work and it was made available to all of Keller’s project managers and their clients.

The concept of a digital twin for vibro-ground improvement was later introduced [4.3] to improve quality control. Then hybrid machine learning was added, i.e., a combination of classical analysis and machine learning (ML), see [4.4]. The hybrid system computes a set of (KPI) for each element, based on the corresponding multivariate time series data (MVTs), i.e., the real-time machine data. The KPI capture the current status of expert knowledge relating machine data to process properties. The KPI are categorized into groups corresponding to different aspects of the process being executed. The unsupervised machine learning (U-ML) creates a data-model (MLDM) which corresponds to a generalization of the process and influences of a specific site. The MLDM is then used to identify MVTs which are uncharacteristic for the process and/or site. This infers the possibility that the corresponding point is anomalous.

2 Hybrid Learning

The hybrid architecture presented here is a refinement of previous work [4.3–4.6]. The focus was mainly on variational autoencoders (VAE) [4.7]; however, literature suggests that more complex architectures do not necessary lead to better accuracy [4.8]. Consequently, simpler autoencoders (AE), [4.9] are also considered here. Due to the limited space for this paper, only the architecture with the best performance is presented. The metrics for comparison of the architectures can be found in [4.6].

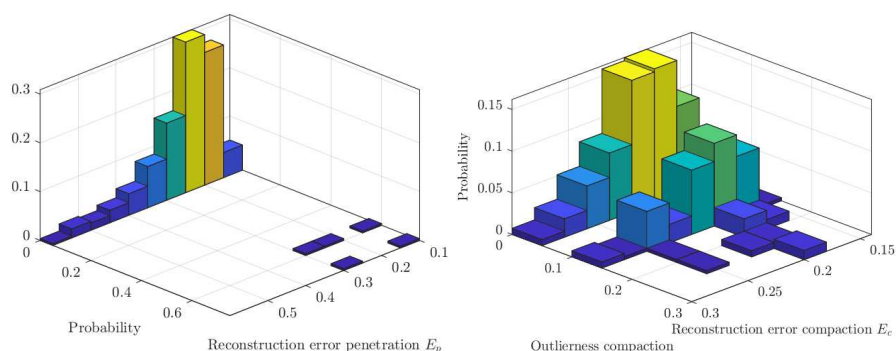


Figure 1: Normalized histogram of the reconstruction error and the outliers. The ML results were generated using the winning architecture - a VAE-LSTM for the penetration phase on the left and a VAE-BiLSTM for the compaction phase on the right.

The goal of anomaly detection is to identify patterns in data which are not observed in normal operations [4.10]. Hybrid machine learning combines functionalities and advantages of multiple techniques into one

architecture [4.8]. Here a combination of KPI and an unsupervised machine learning approach, based on autoencoders, has been selected. The combination of two independent and concomitant techniques adds a layer of redundancy to the system. This reduces the risk of overseeing an anomaly.

The KPI capture the expert knowledge relating properties of the produced element to characteristics that can be calculated directly from the machine data. The statistics of each KPI can be used to identify outliers; subsuming the outliers for each point yields a measure of the *outlierness*; for convenience, this value is normalized, see [4.6] for more details. An autoencoder architecture, implementing unsupervised machine learning, has been selected here to combine with the KPI. The encoders and decoders include recurrent networks; whereby, unidirectional Long-Short term memory (LSTM) [4.11], and bi-directional LSTM (BiLSTM) [4.12] layers were considered.

Each MVTS is characterized by a set of lower dimensional latent variables computed by the encoder. The decoder reconstructs an approximation for the MVTS from the latent variables [4.13]. The norm of the difference between the original MVTS and its approximation, is called the reconstruction error. The training phase of the ML aims to minimize this reconstruction error so as to identify an abstraction for the process as a whole. The MVTS used for training are selected using the results of the KPI anomaly detection, therefore it is called KPI-supervised training [4.6]. In this manner the network learns to reconstruct the data corresponding to normal operation. Conversely, it will produce a higher reconstruction error for non-normal data patterns [4.14, 4.15]. This approach is truly unsupervised, since the selection of training samples is done fully automatically without any need of manually labelling data.

The vibro-ground improvement consists of two sub-processes: penetration and compaction. The MVTS are segmented correspondingly. This enables separate, sub-process specific analysis; it also ensures the correct temporal and spatial localization of the source of the anomaly. Different ML architectures [4.16] may be beneficial for the different sub-process. A focus was placed on physics informed [4.17] hybrid learning. For this reason, the signals: *vibrator amperage*, *depth* and *pull-down force* were selected, since they permit the computation of work (in a physical sense) as a function of time and depth.

The hyperparameters for each architecture were optimized [4.18]. Training a network multiple times often leads to differing performance; this is due to the random initialization of the network parameters. Consequently, each architecture was trained 125 times and the stability of the performance was also considered as a criterion for selection. The performance comparison suggested that: the penetration process was best modelled using a BiLSTM-VAE; whereas, optimal performance was achieved for the compaction process with an LSTM-VAE.

3 Application of the Improved Quality Assurance Procedure

During a manual comparison of hundreds of installation reports, the project manager can easily lose the overview, especially if complex coherences need to be interpreted. Supported by the hybrid learning system of the Digital Twin such tasks can be improved significantly. Typically, the first points on a site are used to create an overview of expected site properties and machine behaviour. The corresponding MVTS can now be used to initialize the hybrid system, establishing the MLDM and limit values for the KPI. Consequently, there are no changes required to the current working procedures; nevertheless, the improvement of the level of quality control is achieved. Furthermore, given the geo-referencing, a model for the systematic variations

of the KPI over the site can be computed. This enables a prediction of behaviour at new locations and yields a statistically more significant detection of anomalous data. It is important to note that the KPI capture expert knowledge; consequently, decisions are not based solely on machine learning; i.e. the machine learning serves as an additional mechanism to detect anomalies possibly overseen by the experts.

3.1 Case Studies

Detection of an element related quality issue

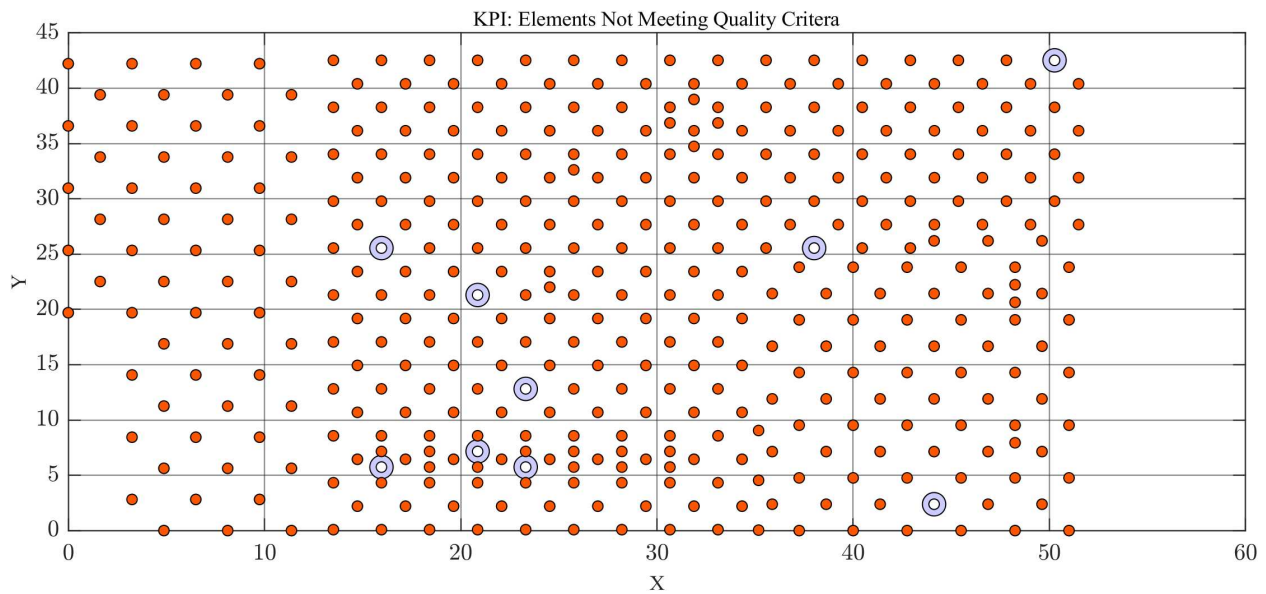


Figure 2: GPS references data: each point corresponds to a produced foundation column. The elements that do not fulfil the quality criteria for compaction are marked with grey circles. These are considered to be outliers require manual evaluation of the corresponding MVTs.

At the beginning, the project manager defines the minimum quality criteria for the compaction of the columns. These are minimum values for amperage, pull-down force and the ratio of pulling and pushing distances of the vibrator for each compaction step. A column is flagged as a quality outlier if it does not fulfil all criteria and needs to be reviewed manually by the project manager (Figure 2).

In the current example, all but one, of the flagged outliers were classified as columns with sufficient quality after the manual review. The installation report for the one exception is shown in Figure 3. Whereas low Amperage and Pull-Down-Force values are recognized at a depth of 1-3 m, the pull-out and push-in ratio of the Vibrator movement is within the range of the Quality-KPI (green in Figure 3). However, in the upper meter the pull-push-ratio is also not fulfilled due to exceeding pullout distances (violet in Figure 3). Consequently, the required execution quality of the column is not achieved. This was detected automatically by the Digital Twin, showing red points at the concerned compaction steps. As a result, the upper part of the column had to be reworked.

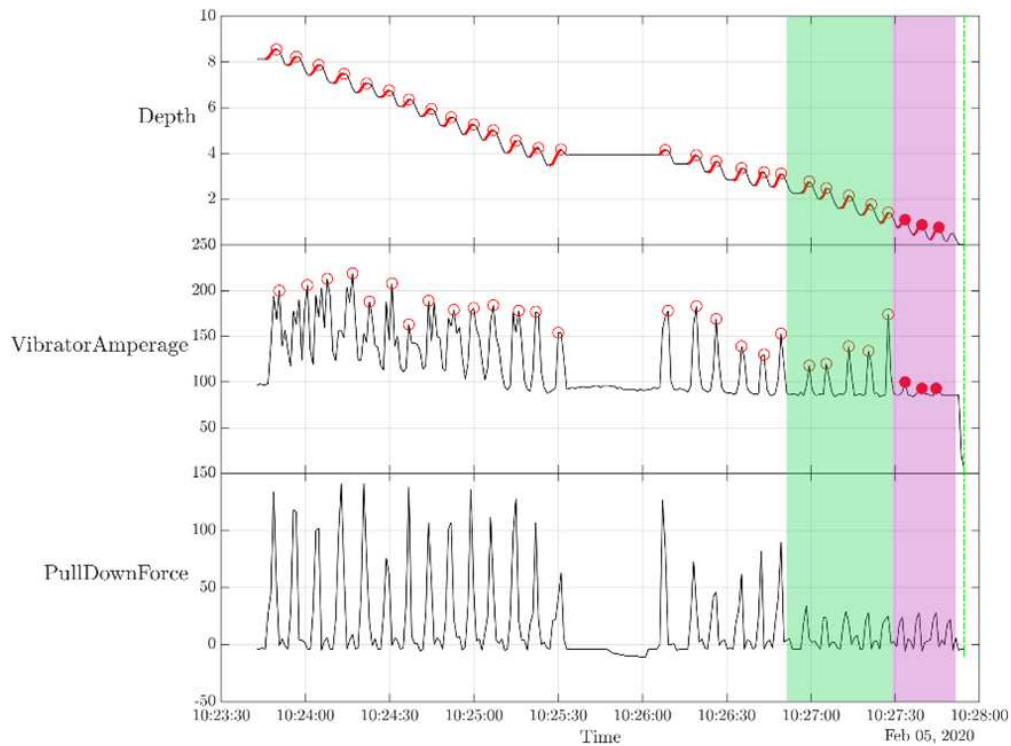


Figure 3: Installation report of a single column in the compaction phase with the analysis of every single compaction step regarding the compaction quality.

Detection of an efficiency issue

For the project illustrated in Figure 4 the estimated shift performance was not reached during execution, however, the reason for the lower performance was not obvious.

The Digital Twin detected a correlation between the temperature of the vibrator and idle times during the execution of the columns (Figure 5). It turned out, that the stiffness of the soil was higher than expected, leading to a higher resistance against penetration and thus resulting in an overheating of the vibrator's motor. Idle time was required to avoid this overheating. This led to a loss in production time and a reduction in process efficiency. This hidden lost time was identified reliably in all cases.

The penetration phases of two installation reports from the same site are shown in Figure 6, where the achieved production rate was lower than expected. The evaluation of the KPI did not show any outliers or other reasons for that behaviour. However, the machine learning algorithm identified several outliers. Review of the corresponding installation reports revealed differences in the working behaviours during the penetration phase. For correctly executed columns, the penetration into the bearable soil layer, i.e. when approaching the maximum depth, took ca. 20 seconds (right in Figure 6). The outliers found by the ML showed a significant longer penetration phase into the bearable soil of ca. one minute (left in Figure 6). That means that the rig operator was overfulfilling the required penetration quality and therefore losing time which was finally leading to a reduced shift performance. As a consequence, a new KPI was defined to describe the quality of the embedding of the column into the bearable soil layer. This can be regarded as a knowledge discovery process.

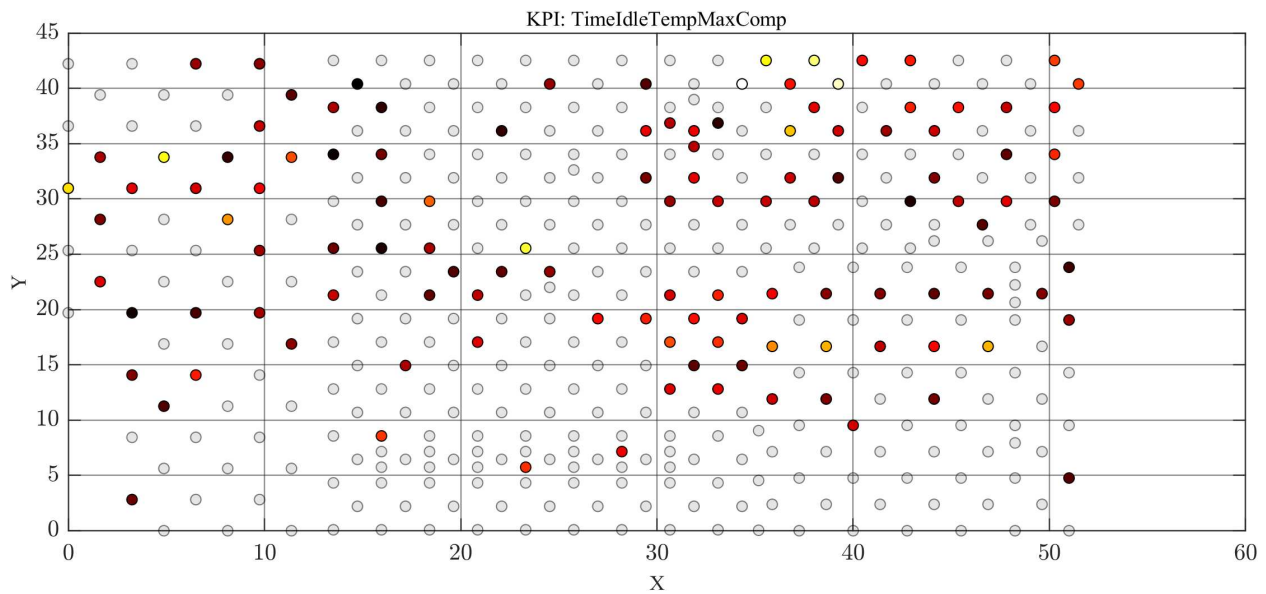


Figure 4: The coloured points correspond to foundation columns where unusual idle times were detected. The machine learning detected that this is a result of overheating of the vibrator.

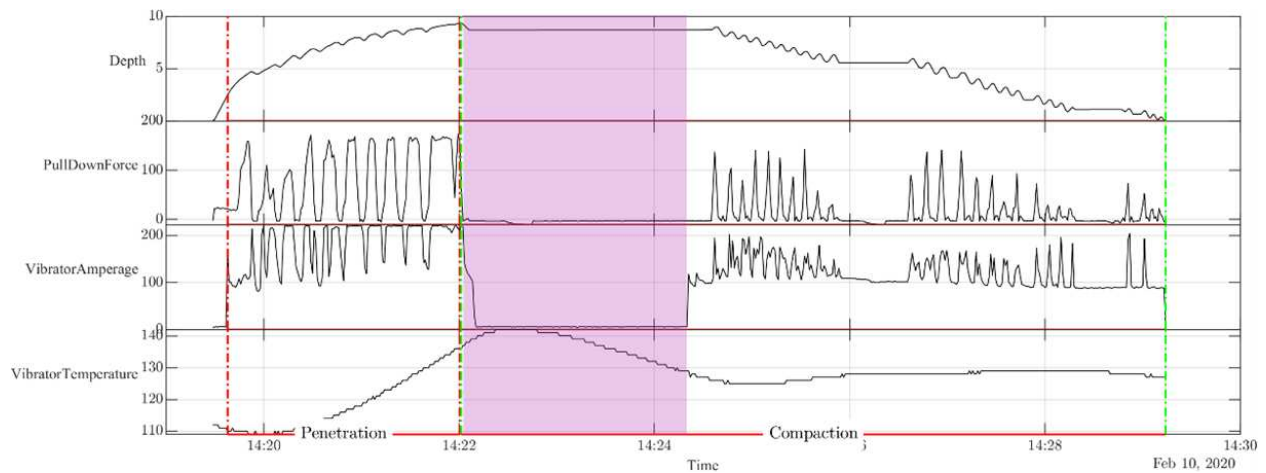


Figure 5: Example installation report exhibiting unplanned Idle time (violet). This was the result of overheating of the vibrator.

4 Outlook

As a next step the new quality application will be fully integrated into Keller’s digital site management system. It will enhance the daily site management routines and support the project and site managers to detect potential quality issues both more reliable and efficient. Also, site or machine related deviations from the planned working procedures, e.g. unusual idle times, will be displayed automatically.

In a next phase, the improved quality application will support the rig operator during execution of the elements. KPI, either pre-defined by the project manager or automatically referenced to the parameters of

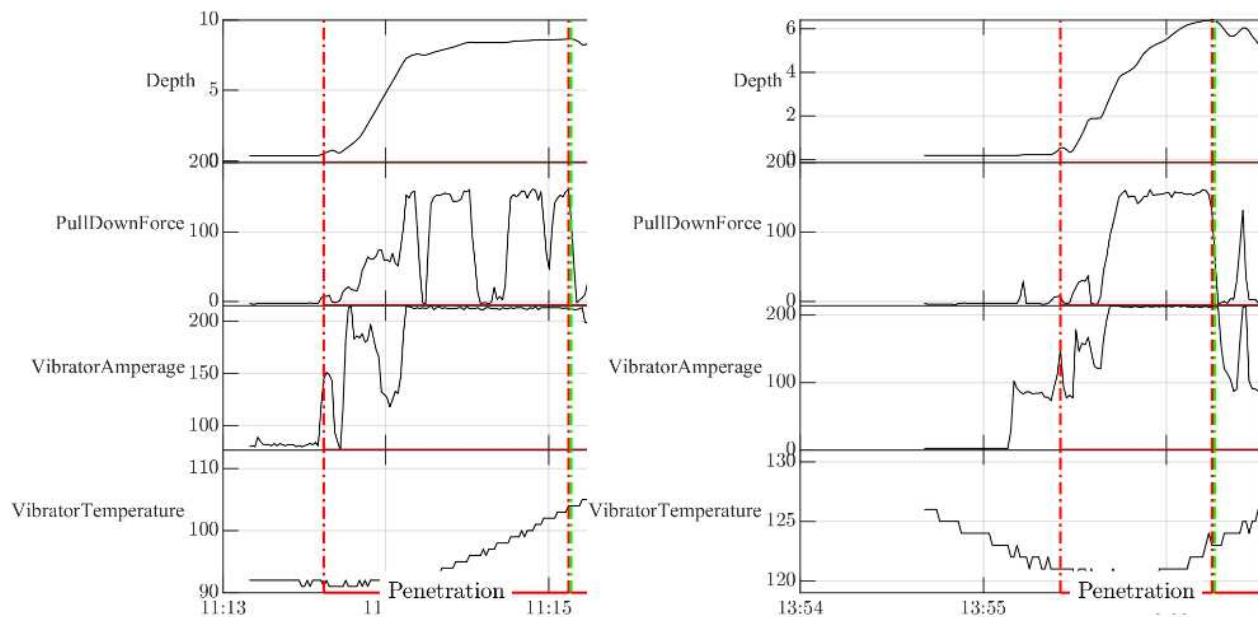


Figure 6: Segments of two installation reports for one and the same site. Left: abnormal case, the operator has overfulfilled the penetration requirement. The required bearing capability of the soil had been reached but penetration was nevertheless continued. Right: Normal operation where the penetration is terminated when the required bearing capacity is reached.

already executed points, will define the quality criteria for each element. The operator will be informed, in a timely manner, by the rig computer if one or several quality criteria are not fulfilled. This will enable the immediate adjustment of the working procedures. The timely detection of anomalous points permits alleviating measures to avoid finishing a site with low quality issues.

Of course, the implementation of the improved quality application for deep vibro techniques will serve as a role model to develop similar procedures for various other geotechnical techniques, e.g. jet grouting. For any new technique the definition of meaningful KPI will be the biggest effort, but due to the hybrid approach this step will be more and more supported by the results delivered by the ML algorithms.

Bibliography

- [4.1] A. Zöhrer and J. Wondre, “Neue entwicklungen und anwendungen von vibroscan,” in *2. Symposium Baugrundverbesserung in der Geotechnik*, D. Adam and H. Richard, Eds., vol. 2. Vienna, Austria: Eigenverlag, Wien, 2012, pp. 179–193.
- [4.2] E. Falk, A. Zöhrer, and G. Strauch, “Entwicklung der tiefenverdichtung – von der händischen datenaufzeichnung zur automatischen visualisierung,” in *Österreichischer Geotechniktag 2011 Tagungsband*, Vienna, Austria, 2011.

- [4.3] N. Khalili-Motlagh-Kasmaei, D. Ninevski, P. O’Leary, C. J. Rothschedl, V. Winter, and A. Zöhrer, “A digital twin for deep vibro ground improvement,” in *International Conference on Deep Foundations and Ground Improvement: Smart Construction for the Future (DFI-EFFC 2022)*, Berlin, Germany, Technical University of Berlin, 2022.
- [4.4] A. Terbuch, A. Zöhrer, V. Winter, P. O’Leary, N. Khalili-Motlagh-Kasmaei, and G. Steiner, “Quality monitoring in vibro ground improvement – A hybrid machine learning approach,” *Geomechanics and Tunneling*, vol. 15, no. 5, pp. 658–664, Oct. 2022.
- [4.5] A. Terbuch, P. O’Leary, and P. Auer, “Hybrid machine learning for anomaly detection in industrial time-series measurement data,” in *2022 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2022, pp. 1–6.
- [4.6] A. Terbuch, P. O’Leary, N. Khalili-Motlagh-Kasmaei, P. Auer, A. Zöhrer, and V. Winter, “Detecting anomalous multivariate time-series via hybrid machine learning,” *IEEE Transactions on Instrumentation and Measurement*, 2023.
- [4.7] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” *ArXiv e-prints*, Dec. 2013.
- [4.8] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The M4 Competition: 100,000 time series and 61 forecasting methods,” *International Journal of Forecasting*, vol. 36, no. 1, pp. 54–74, Jan. 2020.
- [4.9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” in *Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations*. Cambridge, MA, USA: MIT Press, Jan. 1986, pp. 318–362.
- [4.10] H. S. Mavikumbure, C. Wickramasinghe, D. Marino, V. Cobilean, and M. Manic, “Anomaly Detection in Critical-Infrastructures using Autoencoders: A Survey,” in *IECON 2022 – 48th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, Oct. 2022, pp. 1–7.
- [4.11] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 11 1997.
- [4.12] A. Graves, S. Fernández, and J. Schmidhuber, “Bidirectional recurrent neural networks,” in *International conference on artificial neural networks*. Springer, 2005.
- [4.13] P. Baldi, “Autoencoders, unsupervised learning, and deep architectures,” in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, ser. Proceedings of Machine Learning Research, I. Guyon, G. Dror, V. Lemaire, G. Taylor, and D. Silver, Eds., vol. 27. Bellevue, Washington, USA: PMLR, 02 Jul 2012, pp. 37–49. [Online]. Available: <https://proceedings.mlr.press/v27/baldi12a.html>
- [4.14] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, “Do Deep Neural Networks Contribute to Multivariate Time Series Anomaly Detection?” *ArXiv e-prints*, Apr. 2022.
- [4.15] A. Garg, W. Zhang, J. Samaran, R. Savitha, and C.-S. Foo, “An evaluation of anomaly detection and diagnosis in multivariate time series,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 6, pp. 2508–2517, 2021.

- [4.16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge and London: MIT Press, 2016.
- [4.17] M. Raissi, P. Perdikaris, and G. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686 – 707, 2019.
- [4.18] A. Terbuch, “LSTM hyperparameter optimization: Impact of the selection of hyperparameters on machine learning performance when applied to time series in physical systems,” Master’s thesis, Chair of Automation, Montanuniversitaet Leoben, 2021.

Instrumentation and Signal Processing for the Verification of Directional Drilling

Paul O’Leary¹, Anika Terbuch¹, Dimitar Ninevski¹,
Negin Khalili-Motlagh-Kasmaei¹, Daniel Mevec², Robert Fruhmann², Michael
Habacher²

¹ Chair of Automation, University of Leoben, Leoben, Austria
{paul.oleary, anika.terbuch, dimitar.ninevski,
negin.khalilimotlaghkasmaei}@unileoben.ac.at

² eSENSEial Data Science GmbH, Leoben, Austria
{daniel.mevec, robert.fruhmann, michael.habacher}@esenseial.at

Submitted to:
IEEE International Instrumentation and Measurement Technology Conference
(I2MTC) 2024

Submission Date: November 16, 2023

Status: Under Review

Instrumentation and Signal Processing for the Verification of Directional Drilling

Paul O’Leary¹, Anika Terbuch¹, Dimitar Ninevski¹,
Negin Khalili-Motlagh-Kasmaei¹
Daniel Mevec², Robert Fruhmann², Michael Habacher²

¹ Chair of Automation

University of Leoben

Leoben, Austria

{paul.oleary, anika.terbuch, dimitar.ninevski,
negin.khalilimotlaghkasmaei}@unileoben.ac.at

² eSENSEial Data Science GmbH,

Leoben, Austria

{daniel.mevec, robert.fruhmann, michael.habacher}@esenseial.at

Abstract

This paper presents the development of a new instrument and signal processing algorithm that enable the measurement of the trajectories of underground holes during directional drilling. It uses gyroscopes and accelerometers to determine when the drill string is stationary and at these locations to determine the local orientation of the trajectory with respect to Earth’s gravity. Hierarchical indexing of the multivariate time-series data has been implemented to ensure high numerical efficiency. The statistical behavior of the sensors is characterized to ensure the correct application of least squares approximation to the reconstruction process. A differential geometric approach is taken to the curve reconstruction from local gradients. The instrument was tested on a full-scale drilling rig performing directional drilling. The goal is to acquire test data under fully realistic operational conditions and to ensure that the sensors withstand the harsh drilling conditions. The signal processing methods are demonstrated on the data acquired during the real drilling process; yielding the trajectory of the test holes.

Keywords: Gyroscopes and accelerometers, differential geometry, directional drilling, measurement while drilling

1 Introduction

This paper presents a new instrumentation and measurement procedure for the monitoring of directional drilling, the directional drill rig is shown in Figure 1. The task of monitoring borehole trajectories can be described as calculating the drilled trajectory based on the data recorded while performing the drill. In most cases, the data is only recorded at a limited number of discrete points and contains basic parameters such as depth and inclination but does not provide the trajectory of the borehole [4.1]. Due to the high-risk application, the execution of the planned trajectories needs to be ensured. This measurement task is often performed by instrumented inclinometers, known as borehole inclinometers [4.2], which are a common choice for geotechnical instrumentation [4.3]. One approach to instrumenting deep boreholes is using fiber-optic gyros [4.4–4.6]. Another use-case of instrumentation in civil engineering is the a-posteriori evaluation of the drilling; to monitor soil improvement as presented in [4.7]. Directional drilling, which is the main technical way in geological exploration also relies on trajectory prediction; in [4.8] an approach based on random forests is presented. In *measurement while drilling* systems commonly magnetometers, more precisely fluxgates are used, as described in [4.6]. However, their downside when performing measurements for borehole trajectory using magnetometers is, that they are very sensitive to the external magnetic fields, e.g., the magnetic field of the earth.

The instrumentation approach presented in this paper was developed to simplify the measurement process. The starting point was the following procedure: each time a series of measurements is performed, the drilling needs to be stopped and the tool removed from the borehole. After the measurement is completed, the measurement equipment is removed and the boring equipment re-inserted into the hole to continue the drilling.

The problem with this approach is that it is time-consuming and loose stones can trickle into the borehole. To eliminate the need to retract the boring equipment to perform the measurements at different depths, the drilling head was instrumented.

In this work, the drilling head was instrumented with inclinometers and gyroscopes. Gyroscopes were used to determine when the drilling was stopped to perform the measurement. For this task special hardware was developed; including six sensors, see Figure 2.

Once the data has been collected, the drilling trajectory curve needs to be reconstructed. Curve reconstruction for boreholes using polynomials was presented in [4.9], however, this was used to model seismic activities in glaciers. In [4.1] some of the commonly used methods for reconstructing the borehole are named: minimum curvature method, cylindrical helix method, and natural curve method. It is also mentioned that no ground truth for the borehole trajectory model is available and that the task of calculating the borehole trajectory can also be seen as the task of calculating the coordinates in the space where the drilling takes place. A trigonometric approach was chosen in [4.1]; with this, the trajectory at any point is modeled as a cylindrical helix, similar to screw theory. Here an approach based on differential geometry (DG) [4.10] is presented.



Figure 1: Rig in operation during the drilling of test holes for the acquisition of representative data. This should also verify the sensor's capability to survive the harsh environment during drilling.



Figure 2: Prototype instrument designed to be inserted in the drill string, to enable testing under real operating conditions. Top: 3D rendering of the design. The three accelerometers and three gyroscopic sensors are contained within the blue module, together with a processing unit and Li-Ion batteries. The sensor module is autonomous. Bottom: physical prototype of the drill head used during testing.

2 Instrumentation

A prototype instrumentation was designed and manufactured to enable the on-site testing of the concept under real operating conditions. A rendering of the design and an image of the initial implementation are shown in Figure 2. The measurement unit contains a TDK ICM-20948 MEMS device, an ARM Cortex-M3 CPU with 512 MB RAM, a lithium-ion accumulator, and a wireless charging interface. Data is also transferred in a wireless manner. This enables the device to be hermetically sealed, essential for operation in the harsh environmental conditions encountered on underground drill rods.

In this application the accelerometers were parameterized to have a full-scale sensitivity of $\pm 8g$ with 16-bit resolution; while, the gyroscopes were set to $\pm 1000 dps$ also with 16-bit resolution. The temperature sensor is used to monitor the system and avoid misinterpretation of data when temperatures are too high.

A sampling frequency $f_s = 4500 Hz$ was selected and the code separates the data into $T = 1 s$ segments and saves these as separate JSON files, including the required metadata.

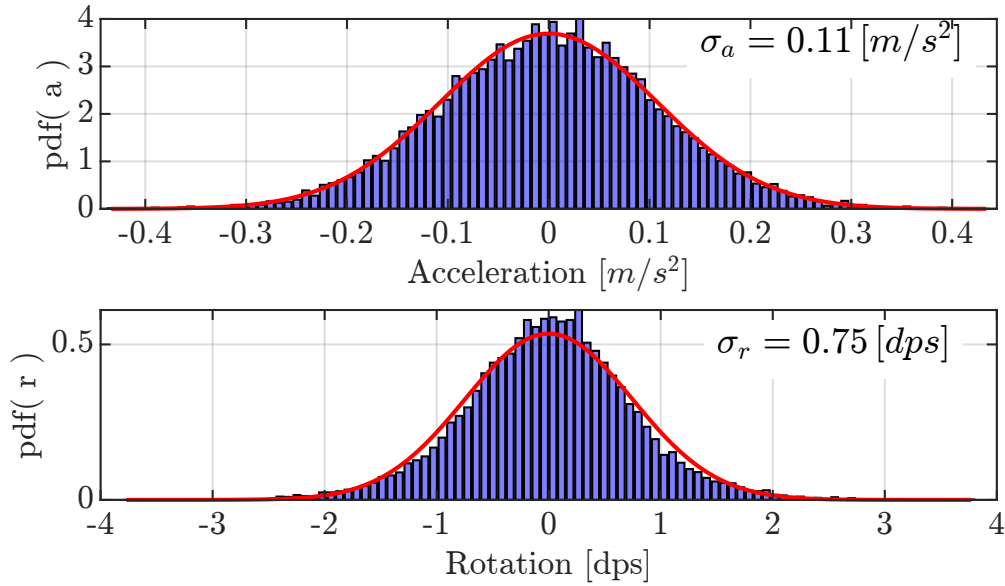


Figure 3: Histograms for the perturbations of acceleration (top) and rotation (bottom), normalized to yield probability density functions. This data is acquired while the drill string is stationary. These are for $m = 13500$ samples, the respective standard uncertainties are $\sigma_a = 0.11 m/s^2$ and $\sigma_r = 0.75 dps$. The red lines are the approximations by a normal distribution. Both cases pass the Kolmogorov-Smirnov test.

2.1 Calibration and characterization

The newly developed instrument was mounted on the drill string and held in a vertical stationary position. During this phase, the accelerometers and gyroscopes were calibrated and their noise behavior was characterized. To ensure statistically relevant results $m = 13500$ samples were acquired from the sensors. The corresponding histograms, normalized to yield probability density functions, are shown in Figure 3, together with the approximations by a normal distribution: Both pass the Kolmogorov-Smirnov test [4.11] for being Gaussian. Consequently, least squares minimization methods are appropriate during the computation of approximations. The respective standard uncertainties [4.12, 4.13] are: for acceleration $\sigma_a = 0.11 m/s^2$ and $\sigma_r = 0.75 dps$ for the gyroscopes.

3 Test Measurements

Test drilling was performed on-site, see Figure 1, with a full-scale drilling rig. The goal is not only to test the functionality of the sensors and analysis methods, but to also verify that the instrumentation withstands the very harsh conditions experienced during drilling.

Several test holes were drilled; whereby, approximately every two meters the drilling was paused for a few seconds, with the drill string in a reference orientation. The data for these pauses is automatically identified during processing and used to compute the current position and orientation of the drill head.

During the first test campaign, a total of $n_f = 6278$ data sets were collected, each with $n_c = 7$ data

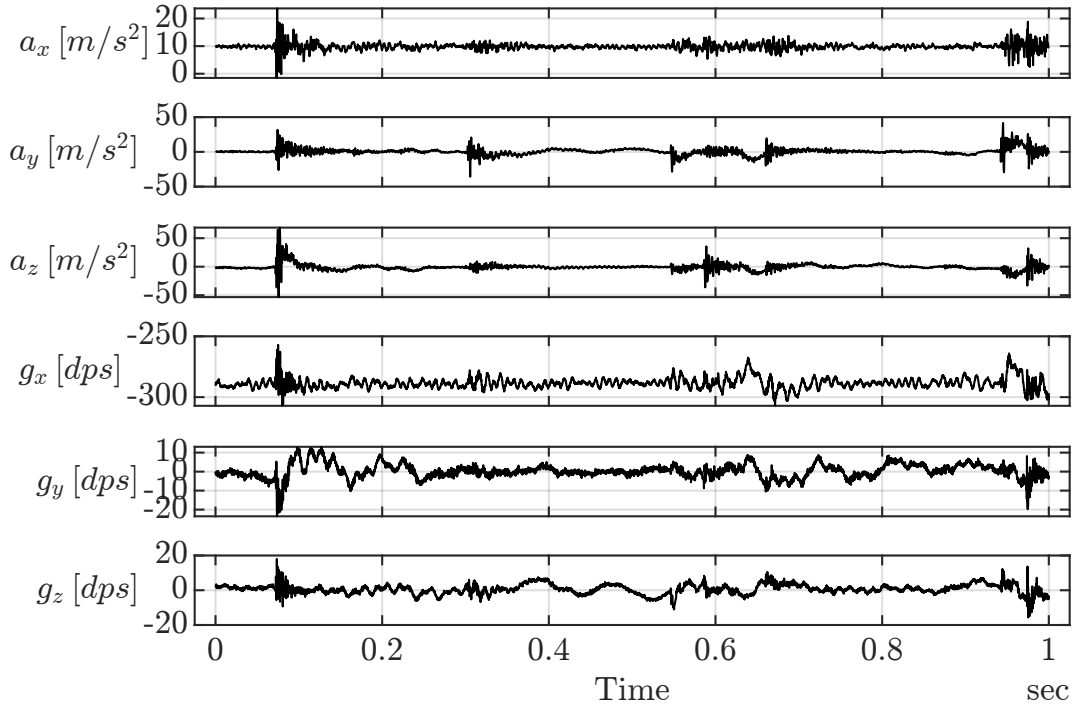


Figure 4: Example of a single measurement time-series. The accelerations A_x , A_y and A_z , together with the gyroscope data G_x , G_y and G_z . The sensor ICM-20948 was used during these tests with the settings: sensitivity-gyroscope $s_g = 1000$ and -accelerometer $s_a = 8$. Data acquired with sampling frequency $f_s = 4.5kHz$ and there are $m = 4500$ samples per channel.

channels and each channel having $m = 4500$ samples. A single example of such a multivariate time-series (MVTs) is shown in Figure 4. This data set has been chosen to illustrate the accelerations observed during drilling. Large accelerations can be observed, these are due to interactions between the drill-head and the underground geological structures. The complete data is separated into batches, each corresponding to the drilling of a specific hole; typically containing 500...600 MVTs. One such batch is used in this paper to illustrate the data analysis methods. The interval between measurements resulting in an MVTs is $\Delta(t) = 4 \pm 20ms$.

4 Data Handling and Preprocessing

The exemplary case chosen to illustrate the data handling and preprocessing contains $n_f = 525$ MVTs and corresponds to the drilling of a single hole. Each MVTs, has seven channels and $m = 4500$ samples per channel, corresponding to $T = 1s$ of measurement time. Hierarchical indexing was implemented to improve numerical efficiency. Each entry in the index contains a time stamp indicating the time when the measurement was performed; a reference to the respective MVTs; and a series of values that summarize the data. The key values relevant in this paper are \bar{r} the mean rotation of the drill string, the mean accelerations \bar{a}_x , \bar{a}_y , \bar{a}_z ; together, with their respective standard deviations σ_x , σ_y and σ_z .

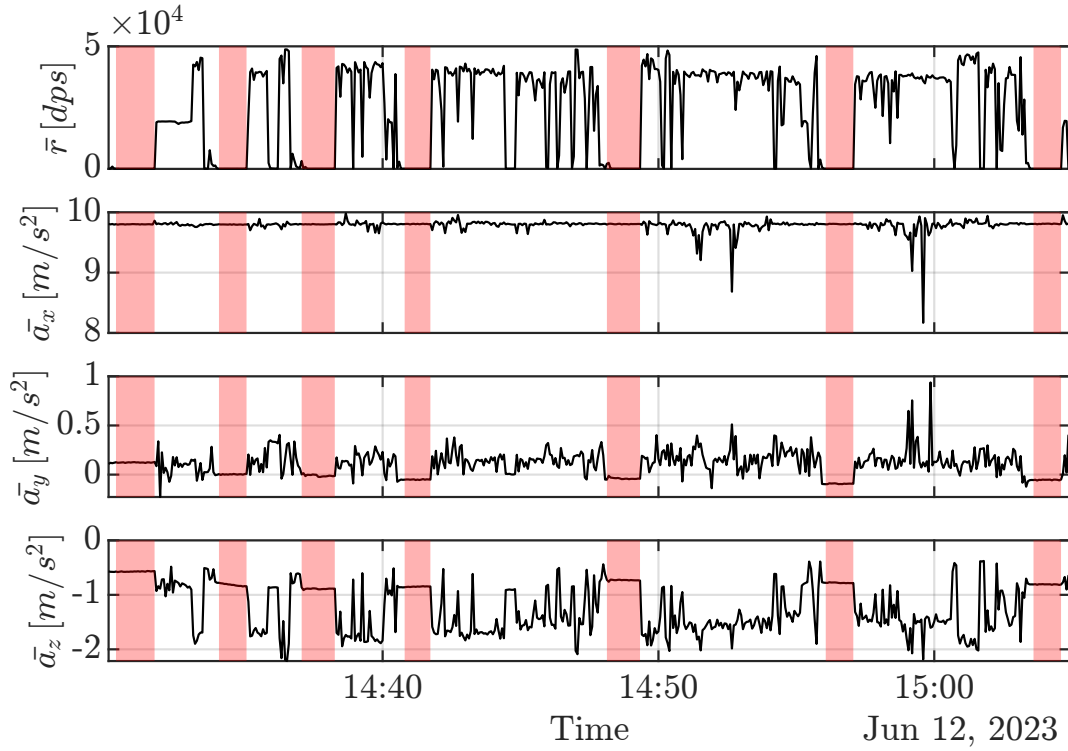


Figure 5: A portion of the index data is used to identify the time intervals when the drill string is stationary, these are underlayed in red. Note that the time required to traverse $l = 2m$ is not constant across the drill hole, this is due to local variations in the subsurface geo-mechanical properties.

The stationary state of the drill string is initially identified as $|\bar{r}| \leq \varepsilon$; whereby, the value for ε is selected to be consistent with the statistical data shown in Figure 3. Additionally it was observed that the vibration of the drill rods may need a few seconds to abate. As defined for the drilling process there are seven stationary segments in the data, they are characterized in Table 4.5. Each segment is defined by a start i_s and end i_e index, n the number of MVTs contained in the segment and the total number of samples available for that segment m . There are a minimum of $m = 63000$ samples per segment where the drill rod is stationary; combined with the measured standard uncertainty, shown in Figure 3, this ensures statistically relevant results.

5 Curve Reconstruction

The goal now is to reconstruct the trajectory of the hole, given the length of rod l in the hole at each measurement location and the corresponding measurements, $\bar{a}_x(l)$, $\bar{a}_y(l)$ and $\bar{a}_z(l)$. The drill hole can be modeled as a cylindrical helix [4.9], see Figure 6; whereby, the sensor module glides along the trajectory of the hole. In the stationary state, the accelerometers measure g the Earth's gravitation. Assuming, that gravity defines the vertical direction, implies that $\bar{a}_x(l)/g$, $\bar{a}_y(l)/g$ and $\bar{a}_z(l)/g$ correspond the portions in the x , y and z directions respectively. This in turn, defines the components of the directional vector of the tangent to the curve at each measurement location, i.e., $dx/dl = \bar{a}_x(l)/g$, similarly for y and z . The fact that

Table 4.5: Segment table for the index data presented in Figure 5.

segment	i_s	i_e	n	m
1	5	26	21	94500
2	61	76	15	67500
3	106	124	18	81000
4	162	176	14	63000
5	272	290	18	81000
6	391	406	15	67500
7	504	519	15	67500

for each location l the tangent directions are known, defines this as the problem of reconstructing curves from gradients [4.14, 4.15]. Differential geometry [4.10], combined with matrix algebraic formulations for least squares approximation provide an efficient numerical implementation.

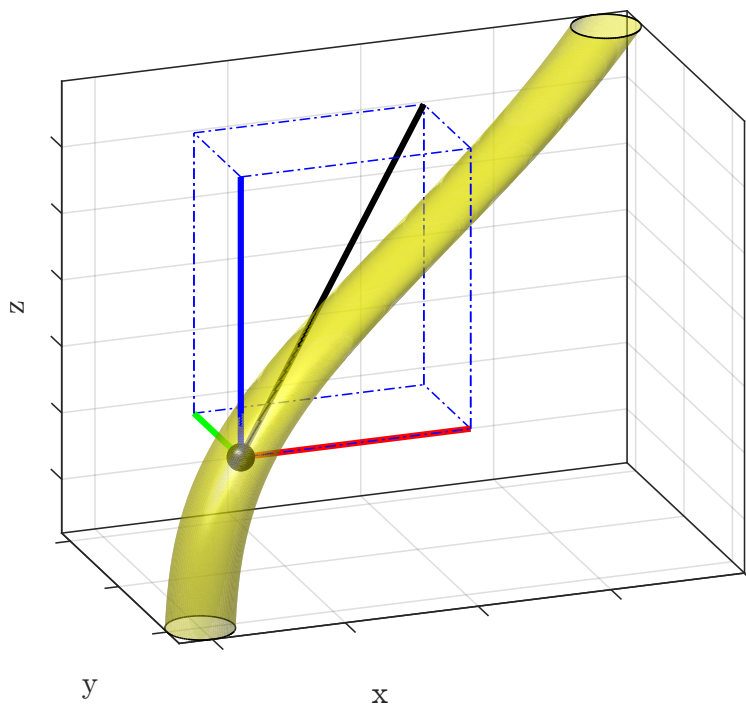


Figure 6: Drill hole modeled locally as a cylindrical helix. The black sphere indicates a measurement point and the black line indicates a directional vector of the tangent at that point. The accelerometers measure the components of the directional vector in x , y and z , i.e., red, green and blue respectively.

6 Differential Curve Modeling

Bézier curves can be used to model trajectories in higher dimensional spaces [4.16]. They are formed from linear combinations of the Bernstein polynomials [4.17]. That is, the Bézier curve $\mathbf{G}(u)$, in j dimensions is obtained by evaluating,

$$\mathbf{G}(u) = \mathbf{B}(u) \mathbf{C}. \quad (4.12)$$

Where, $\mathbf{B}(u)$ has column vectors corresponding to the Bernstein basis and \mathbf{C} , a $(d+1) \times j$ matrix, defining the control points for polynomials of degree d in j dimensions. Each row in \mathbf{C} defines a control point in j dimensions.

The local tangents $\mathbf{G}'(u)$ to the curve $\mathbf{G}(u)$ can be calculated as, $\mathbf{G}'(u) = \mathbf{B}'(u) \mathbf{C}$, where $\mathbf{B}'(u)$ are the derivatives of the Bernstein polynomials. This makes the Bézier curves particularly amenable to curve reconstruction from gradients.

6.1 Bernstein basis

The n^{th} polynomial in u of degree d is defined as,

$$b_{n,d}(u) := \binom{d}{n} u^n (1-u)^{d-n}, \quad n = 0, \dots, d, \quad (4.13)$$

given $0 \leq u \leq 1$, otherwise $b_{n,d}(x) = 0$. Additionally, their derivatives can be computed as a combination of two polynomials of lower degree,

$$b'_{(n,d)}(u) = d \{b_{(n-1,d-1)}(u) - b_{(n,d-1)}(u)\}. \quad (4.14)$$

Consequently, given code to synthesize $b_{n,d}(u)$ it is possible to calculate $b'_{(n,d)}(u)$ with minimal additional effort. Note these are the analytic derivatives of the polynomials and not finite difference approximations. This yields the Bernstein bases $\mathbf{B}(u) = [b_{0,d}(u), \dots, b_{d,d}(u)]$ and their derivatives $\mathbf{B}'(u)$. Note since $\text{rank}(\mathbf{B}(u)) = d+1$ the $\text{rank}(\mathbf{B}'(u)) = d$. Consequently, $\mathbf{B}'(u)$ is rank-one deficient in terms of determining the matrix of coefficients \mathbf{C} .

There is the need for a coordinate transformation between the length of drill rod l in the hole and u , such that,

$$l \mapsto u = \frac{l - \min(l)}{\max(l) - \min(l)}. \quad (4.15)$$

The effects of this transformation on the derivatives must also be accounted for,

$$\frac{dx}{dl} = \frac{dx}{du} \frac{du}{dl}, \quad (4.16)$$

similarly for y and z .

6.2 Constrained Bézier curve reconstruction from gradients

The location of the top of the hole is known and here the orientation of the drill rod is initially assumed to be vertical. These can be formulated as constraints on the control points, i.e.,

$$\mathbf{B}(1, :) \mathbf{C} = \mathbf{0}, \text{ and} \quad (4.17)$$

$$\mathbf{B}'(1, :) \mathbf{C} = \mathbf{0}. \quad (4.18)$$

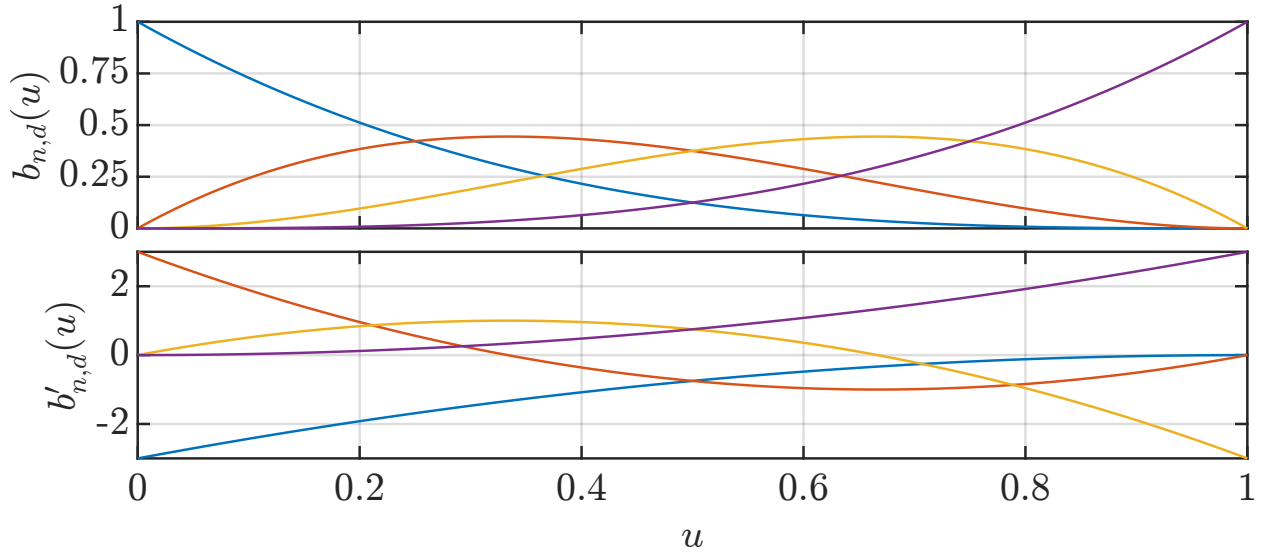


Figure 7: Example of the degree $d = 3$ Bernstein polynomials (top) and their first derivatives (bottom).

Consequently, there is one value and one derivative constraint at the start of the hole trajectory. Defining the constraining matrix \mathbf{K} as,

$$\mathbf{K} = \begin{bmatrix} \mathbf{B}(1, :) \\ \mathbf{B}'(1, :) \end{bmatrix}, \quad (4.19)$$

yields, $\mathbf{K}\mathbf{C} = \mathbf{0}$. Consequently, to fulfill these constraints $\mathbf{C} \in \text{null}(\mathbf{K})$.

The algorithm to reconstruct the curve proceeds as follows:

1. Given the lengths of rod l in the hole at each measurement point, map $l \mapsto u = (l - \min(l)) / (\max(l) - \min(l))$, this ensures $0 \leq u \leq 1$.
2. Now given u compute the Bernstein basis $\mathbf{B}(u)$ and its first derivative $\mathbf{B}'(u)$ according to equations 4.13 and 4.14.
3. Define the constraints, represented by the constraining matrix \mathbf{K} defined in Eqn. 4.19, that need to be observed at the start of the hole.
4. Apply singular value decomposition $\mathbf{K} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, and extract an ortho-normal basis set \mathbf{N} for the null space of \mathbf{K} , i.e., $\mathbf{N} = \mathbf{V}(:, r+1 : d+1)$ where r is the rank of \mathbf{K} .
5. Compute the constrained Bernstein polynomials $\mathbf{B}_c(u) = \mathbf{B}(u)\mathbf{N}$ and their first derivatives $\mathbf{B}'_c(u) = \mathbf{B}'(u)\mathbf{N}$.
6. Assemble the column vectors of measured accelerations to form $\mathbf{G}' = [\bar{a}_x, \bar{a}_y, \bar{a}_z]$.
7. Compute the matrix of control points $\mathbf{C} = \{\mathbf{B}'\}^+(u)\mathbf{G}'(u)$.
8. Since \mathbf{B}' is rank-one deficient there is still one degree of freedom. This is used to position the whole curve so that it is consistent with the location of the start point of the drilling.

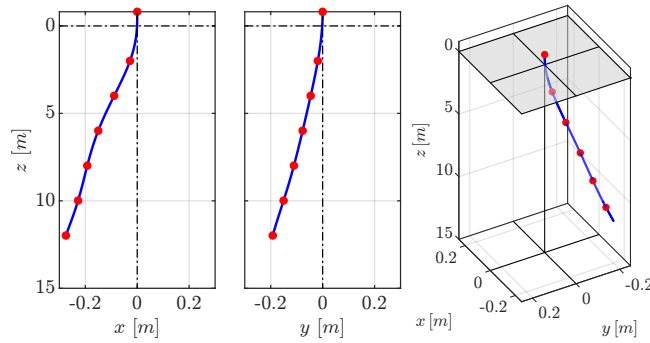


Figure 8: Trajectory of the hole in $x(l)$, $y(l)$ and $z(l)$ reconstructed from the measurement data.

9. Given \mathcal{C} interpolate the curve at the locations u_i .

The results of this process, applied to the previously presented data, is shown in Figure 8. The trajectory of the curve is exactly fulfilled at the measurement points and the interstitial curve points are interpolated using the Bernstein polynomials.

7 Conclusions

This paper has presented the development and testing of a new instrument for the measurement of hole trajectories during directional drilling. The new approach permits *measurement while drilling*, that is, it is not necessary to remove the drill string to perform a measurement. The new instrument was tested during two days of directional drilling operations involving a full-scale drilling rig. The instrument withstood this long period of very harsh conditions without any issues. The new signal processing methods were successfully applied to the data acquired during these tests.

This simplifies the drilling process and shortens the time required to execute a drilling and grouting process. The trajectory of the hole is modeled as a cylindrical helix and the sensors are used to measure local gradients of the curve. A new approach to reconstructing a curve in multiple dimensions from gradients was present. It takes advantage of the properties of Bernstein polynomials to obtain a set of equations that are simple to solve using matrix algebra. The linear nature of these equations opens the door to an extension that permits the computation of both confidence and prediction intervals for the trajectory.

Acknowledgements

We wish to thank the company Keller Grund- und Tiefbau GmbH for their continuing support in this project and enabling the test measurements.

The authors gratefully acknowledge the financial support under the scope of the COMET program within the K2 Center “Integrated Computational Material, Process and Product Engineering (IC-MPPE)” (Project No 886385). This program is supported by the Austrian Federal Ministries for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK) and for Labour and Economy (BMAW), represented by the Austrian Research Promotion Agency (FFG), and the federal states of Styria, Upper Austria and Tyrol.

Bibliography

- [4.1] X. Liu, “Quantitative recognition method for borehole trajectory models,” *Petroleum Exploration and Development*, vol. 45, no. 1, pp. 154–158, Feb. 2018.
- [4.2] L. Wu, T. Li, Z. Chen, and H. Li, “A new capacitive borehole tiltmeter for crustal deformation measurement and its performance analysis,” *International Journal of Mining Science and Technology*, vol. 25, no. 2, pp. 285–290, Mar. 2015.
- [4.3] G. Machan and V. G. Bennett, “Use of inclinometers for geotechnical instrumentation on transportation projects: State of the practice,” *Transportation Research Circular*, no. E-C129, 2008.
- [4.4] Y. Liu, C. Wang, G. Luo, and W. Ji, “Design and applications of drilling trajectory measurement instrumentation in an ultra-deep borehole based on a fiber-optic gyro,” *Geoscientific Instrumentation, Methods and Data Systems*, vol. 9, no. 1, pp. 79–104, Mar. 2020.
- [4.5] Y. Liu, J. Wang, W. Ji, and G. Luo, “Temperature field finite element analysis of the ultra-high temperature borehole inclinometer based on fog and its optimization design,” *Chemical engineering transactions*, vol. 51, pp. 709–714, 2016. [Online]. Available: <https://api.semanticscholar.org/CorpusID:114422403>
- [4.6] G. Shuang, W. Pan, and L. Ling-ling, “Research on fast measurement method of borehole trajectory based on fiber optic gyro mwd system,” in *2017 2nd International Conference on Cybernetics, Robotics and Control (CRC)*, 2017, pp. 149–154.
- [4.7] A. Greinmeister, J. Golser, A. Radinger, G. Jedlitschka, M. Eder, M. Feldinger, F. Rathmair, and E. Sauer, “Monitoring and analysis of the behavior of the salzburg-clay during the jet grouting works for the s-link project,” in *Proceedings of the ISRM 15th International Congress on Rock Mechanics and Rock Engineering and 72nd Geomechanics Colloquium*, 2023, pp. 1436–1441.
- [4.8] B. Yan, X. Zhang, C. Tang, X. Wang, Y. Yang, and W. Xu, “A random forest-based method for predicting borehole trajectories,” *Mathematics*, vol. 11, no. 6, p. 1297, 2023.
- [4.9] S. Hellmann, M. Grab, C. Patzer, A. Bauder, and H. Maurer, “A borehole trajectory inversion scheme to adjust the measurement geometry for 3D travel-time tomography on glaciers,” *Solid Earth*, vol. 14, no. 7, pp. 805–821, Jul. 2023.
- [4.10] W. Blaschke and K. Leichtweiß, *Elementare Differentialgeometrie (Grundlehren der mathematischen Wissenschaften, I, Band 1)*. Berlin, Germany: Springer, Jan. 1973.
- [4.11] F. J. Massey, Jr., “The Kolmogorov-Smirnov Test for Goodness of Fit,” *Journal of the American Statistical Association*, pp. 68–78, Apr. 2012. [Online]. Available: <https://www.tandfonline.com/doi/abs/10.1080/01621459.1951.10500769>
- [4.12] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML, “Guide to the expression of uncertainty in measurement — Part 6: Developing and using measurement models,” Joint Committee for Guides in Metrology, JCGM GUM-6:2020.

- [4.13] ———, “International vocabulary of metrology — Basic and general concepts and associated terms (VIM),” Joint Committee for Guides in Metrology, JCGM 200:2012. (3rd edition).
- [4.14] P. O’Leary, M. Harker, and J. Golser, “Direct discrete variational curve reconstruction from derivatives and its application to track subsidence measurements,” in *2011 IEEE International Instrumentation and Measurement Technology Conference*, 2011, pp. 1–6.
- [4.15] P. O’Leary and M. Harker, “A framework for the evaluation of inclinometer data in the measurement of structures,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 5, pp. 1237–1251, 2012.
- [4.16] L. Piegl and W. Tiller, *The NURBS Book*. Berlin, Heidelberg: Springer-Verlag, 1995.
- [4.17] G. Lorentz, *Bernstein Polynomials*, ser. AMS Chelsea Publishing Series. Chelsea Publishing Company, 1986. [Online]. Available: <https://books.google.at/books?id=nXiRAKzpdjUC>

Chapter 5

Scientific Machine Learning

This chapter gives an overview of the architectures and synthesis algorithms developed for scientific machine learning. The progress in the field of scientific machine learning, especially around the novel architecture called *Rayleigh-Ritz Autoencoder*, is demonstrated with the following papers:

1. A Rayleigh-Ritz Autoencoder,
2. Automatic Synthesis of Admissible Functions for Variational Learning,
3. Extended Rayleigh-Ritz Autoencoder with Distribution-Free Statistics.

A Rayleigh-Ritz Autoencoder

Anika Terbuch¹, Dimitar Ninevski¹, Paul O’Leary¹, Elias Jan Hagendorfer²,
Elke Schlager³, Andreas Windisch³, Christoph Schweimer³

¹ Chair of Automation, University of Leoben, Leoben, Austria
{anika.terbuch,dimitar.ninevski,paul.oleary}@unileoben.ac.at

² Materials Center Leoben, Leoben, Austria
eliasjan.hagendorfer@mcl.at

³ Know-Center GmbH, Graz, Austria
{eschlager,awindisch,cschweimer}@know-center.at

Originally appeared as:

A. Terbuch et al., "A Rayleigh-Ritz Autoencoder," 2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Kuala Lumpur, Malaysia, 2023, pp. 1-6, DOI: 10.1109/I2MTC53148.2023.10176014

©2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works

A Rayleigh-Ritz Autoencoder

Anika Terbuch¹, Dimitar Ninevski¹, Paul O’Leary¹, Elias Jan Hagendorfer²,
Elke Schlager³, Andreas Windisch³, Christoph Schweimer³

¹ Chair of Automation
University of Leoben
Leoben, Austria

{anika.terbuch,dimitar.ninevski,paul.oleary}@unileoben.ac.at

² Materials Center Leoben
Leoben, Austria

eliasjan.hagendorfer@mcl.at

³ Know-Center GmbH
Graz, Austria

{eschlager,awindisch,cschweimer}@know-center.at

Abstract

This paper presents a new architecture for unsupervised hybrid machine learning, called a Rayleigh-Ritz Autoencoder (RRAE). It is suitable for applications in instrumentation and measurement where the system being observed by multiple sensors is well modeled as a boundary value problem. The embedding of the admissible functions in the decoder implements a truly physics-informed machine learning architecture. The RRAE provides an exact fulfillment of Neumann, Cauchy, Dirichlet, or periodic constraints. Only the encoder needs to be trained; consequently, the RRAE is numerically more efficient during training than traditional autoencoders.

The new Rayleigh-Ritz Autoencoder has been applied to an instrumentation and measurement problem in structural monitoring. It involves the fusion of data from multiple sensors and the solution of a boundary value problem. A 1-norm minimization has been chosen to minimize

the effects of non-Gaussian perturbations and to demonstrate the non-linear abilities of the RRAE. The results from the tunnel monitoring application over months of work are presented in detail.

Index Terms - Physics informed machine learning, Rayleigh-Ritz, Sturm Liouville, Calculus of Variations, Admissible functions, Structural monitoring

1 Introduction

This paper presents a new architecture for unsupervised hybrid machine learning applied to instrumentation and measurement. The name Rayleigh-Ritz Autoencoder (RRAE), see Figure 1, has been selected, since it is an autoencoder [5.1, 5.2] permitting unsupervised learning; admissible functions are used in the *decoder* to ensure the exact fulfillment of the equations governing the behavior of the physical system. It is a form of *physics-informed* neural networks [5.3]. The *encoder* is implemented as a neural network [5.4], this permits the modeling of any well-behaved function [5.5]. The loss function, used during training, implements the functional with its corresponding cost, as required in the calculus of variations (CoV). It is an autoencoder that implements the Rayleigh-Ritz method, with exact fulfillment of the system's physical constraints.

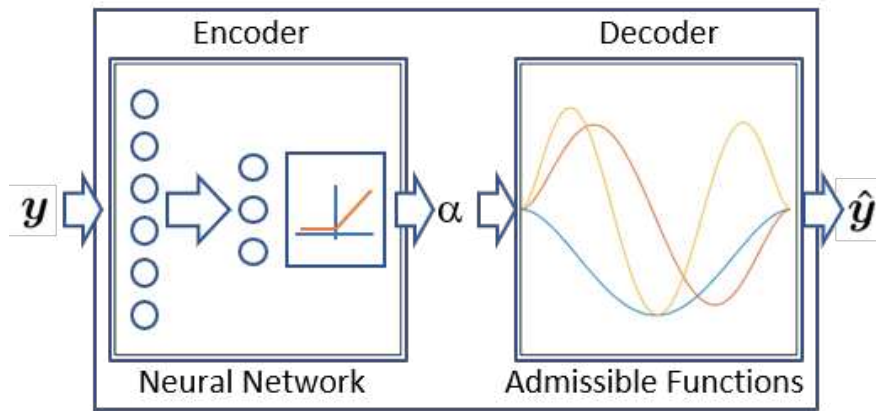


Figure 1: Schematic of the Rayleigh-Ritz Autoencoder structure. The *encoder* is implemented as a neural network and maps the inputs to the latent variables α . The *decoder* uses admissible functions \mathbf{A} , that fulfill the constraints of the physical system in an exact manner, to generate $\hat{\mathbf{y}} = \mathbf{A}\alpha$.

Some key features of the Rayleigh-Ritz Autoencoder developed here are:

1. The a-priori physical constraints on the system are fulfilled exactly and not just approximated.
2. Constraints on the functional or its derivatives, i.e. $f(x)^{(k)}$ at arbitrary locations x are possible. They are not limited to the boundaries of the system.
3. Relational constraints between different locations of the solution can be implemented, $f(x_1)^{(k_1)} = f(x_2)^{(k_2)}$.

4. The admissible functions are automatically synthesized from the sensor locations x and the a-priori constraints.
5. A key feature for instrumentation and measurement applications is that the RRAE provides a computation for error propagation.
6. Only the encoder needs to be trained, this improves the numerical efficiency and convergence during the training of the network.

The focus here is on the solution of inverse problems which require the fusion of data from multiple sensors and the embedding of the equations governing the physical system. Consider the example shown in Figure 2: the task is to monitor the deformation of a tunnel during the construction of a new side access. The new access is to be constructed while the primary tunnel is still in use; consequently, reliable monitoring is important, since there is an immediate danger to life and limb. A finite element simulation was performed to predict the deformations of the wall. The metal ribs supporting the tunnel, have each been instrumented with five sensors, additionally, two fixed points are identified. A reliable and physically relevant measurement of the deformation requires the fusion of the data from the sensors and the embedding of the equations for the deformation of the beam. The two endpoints are defined so that they are fixed in both position and orientation.

The characteristics of such applications [5.6–5.8], are: there are multiple sources of data that need to be merged, there are explicit constraints on the solution and the results must be consistent with the equations governing the behavior of the system.

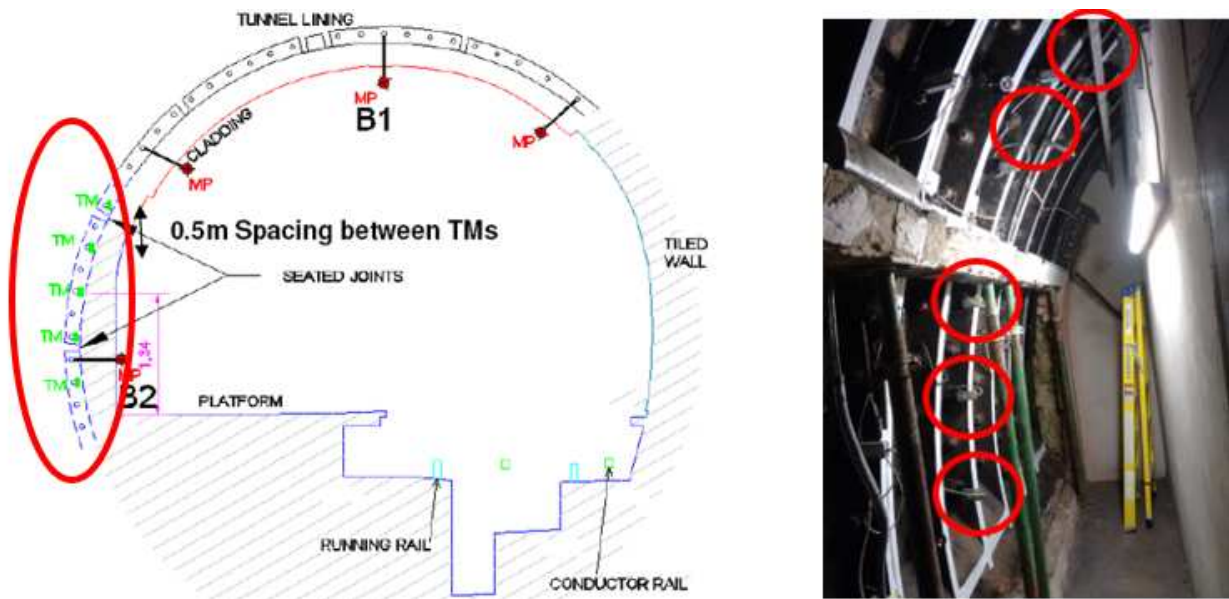


Figure 2: Left: Finite element simulation of the tunnel wall showing expected deformations. Center: Schematic of the planned locations for the sensors; encircled in red. Right: Photograph showing the locations of the individual sensors installed on the structural beams.

A number of approaches have been published addressing boundary value problems in a classical manner [5.9]. The methods of shooting can be used where the constraints are limited to the boundaries of the

system [5.10]. Solutions to more general cases for forward problems are also available [5.11, 5.12].

1.1 Calculus of Variations and Constrained ODE

The goal here is to investigate a new autoencoder [5.1, 5.2] architecture, that can take advantage of the generalizing properties of machine learning, while fulfilling the requirements of a measurement system [5.13–5.15]. Taking an approach based on *the principle of least action* [5.16][Ch. 19] is logical, since it has become the standard approach to modeling systems in modern physics. The principle is generally formulated in terms of the calculus of variations (CoV) [5.17]. The action $S(y)$, i.e. the cost function, is formulated for continuous systems as,

$$S(y) = \int_{x_a}^{x_b} f(x, y, y') dx. \quad (5.1)$$

whereby, $f(x, y, y')$ is the functional, i.e., the cost associated with a specific solution. The task is to find $y(x)$, here a curve, that corresponds to a stationary value of $S(y)$, subject to a set of constraints. This is equivalent to solving the Euler-Lagrange differential equation,

$$\frac{\partial f}{\partial y} - \frac{d}{dx} \left(\frac{\partial f}{\partial y'} \right) = 0, \quad (5.2)$$

subject to the same constraints. The following types of constraints are considered here:

1. Constraints at specific locations:

$$y(x)^{(d)} = w, \quad (5.3)$$

i.e., the d^{th} derivative of the curve $y(x)$, at the location x has the constant value w .

2. Relational constraints,

$$\sum_{i=1}^m y(x_i)^{(d_i)} = w. \quad (5.4)$$

Neumann, Cauchy, Dirichlet and periodic constraints are all special cases of these two definitions. Note: the constraints are not restricted to the boundaries of the system.

2 Neural networks and constrained differential equations

Neural network (NN) approaches have been used to solve differential equations [5.18] and constrained differential equations [5.19, 5.20]. They take advantage of the denoising and generalizing properties of machine learning. However, exclusively data-driven models have no embedded knowledge of the physical system. Consequently, predictions may contain physical inconsistencies and produce implausible solutions.

Physics-informed neural networks [5.3] are a hybridization of classical NN and mathematical models that enforce physical laws. In most approaches presented so far, the physics are inferred from the observations [5.21]. The goal is that the latent space is the parameter space; the encoder learns the parameters of the equation by which the physical system is governed. However, the works presented in [5.22, 5.23] do not enforce the fulfillment of constraints; this issue is crucial, since not fulfilling constraints exactly may invalidate the results.

Solving inverse problems brings additional difficulties since the solutions of inverse problems are not guaranteed to be unique and stable [5.24]. Yadav et al. [5.25] provide a good overview of the known techniques for solving differential equations with neural networks. Most of the presented techniques are based on multilayer perceptrons (MLP) and radial basis functions (RBF). In one of the approaches which combines both, MLP and RBF, the authors conclude that their method could solve the differential equation in a manner that fulfills the boundary condition exactly; however, it is computationally too expensive. With this limitation, they choose to solve the system of equations approximately and exclude solutions that don't fulfill the conditions [5.25].

The paper [5.26] states that the three approaches suitable for solving partial differential equations are: physics-informed neural networks, methods based on the Feynman-Kac formulae [5.27] and methods based on the solution of backward stochastic differential equations. A further overview of physics-informed neural networks can be found in [5.28]. In most of the recently published literature, see e.g. [5.29–5.31,5.31,5.32], the physical constraints are incorporated into the loss function as an additional penalty term, this is referred to as *soft boundary constraint enforcement*; this is akin to weighted constraints in algebraic least squares [5.33,5.34]. The other approach is known as *hard enforcement* and the physical constraints are encoded into the network architecture [5.28].

3 Rayleigh-Ritz Method

The Rayleigh-Ritz method starts from a set of admissible functions, here the columns of the matrix \mathbf{A} , which are assumed to be complete. These functions fulfill the constraints in a homogeneous manner. Consequently, any linear combination is also an admissible function. Furthermore, \mathbf{A} is complete, i.e., $\mathbf{A}\mathbf{A}^+ = \mathbf{I}$ and with this spans the complete space Ω of all admissible solutions. This ensures $\mathbf{y} \in \Omega$ and $\mathbf{y} \in \text{span}(\mathbf{A})$. The model is computed as $\hat{\mathbf{y}} = \mathbf{A}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha}$ is the vector of coefficients, in ML parlance, these are the latent variables. Given these conditions, there exists a set of values for $\boldsymbol{\alpha}$ such that,

$$\min_{\boldsymbol{\alpha}} S(\mathbf{A}\boldsymbol{\alpha}) = \min_{\mathbf{y}} S(\mathbf{y}) \quad \text{given } \mathbf{y} \in \Omega. \quad (5.5)$$

The task at hand now is to compute a set of admissible functions and to determine $\boldsymbol{\alpha}$ to ensure, $S(\mathbf{A}\boldsymbol{\alpha})$ is stationary.

Some machine learning approaches have been presented to this task: in both [5.32, 5.35] a version of *Deep Ritz Method* was used to learn the admissible functions. However, the constraints are taken into account by adding a penalty term to the loss function. Consequently, the constraints are not fulfilled exactly. Chen et al. [5.36] in their comparison of deep-Galerkin (DGM) and deep-Ritz methods (DRM) state that the enforcement of boundary conditions in deep neural networks is highly nontrivial. The main difference between DGM and DRM is the loss function. In both of these methods, the boundary conditions are treated using the penalty method - adding a penalty term to the loss function. Only for the Dirichlet boundary conditions a strategy was found to avoid the penalty term. The special case of periodic constraints was solved [5.37] using the Fourier basis functions.

None of these methods support the exact fulfillment of generic constraints. As a result, the approaches are not generally applicable.

3.1 Synthesizing admissible functions

There are a number of properties that the admissible functions should have:

1. Constraints of the types defined in equations 5.3 and 5.4, should be fulfilled. The exact fulfillment of the constraints is particularly important in structural monitoring, where it is necessary to detect deformations that have a scale of milliliters over distances of meters.
2. The matrix of admissible functions should be unitary $\mathbf{A}^T \mathbf{A} = \mathbf{I}$. This will ensure optimal noise behavior when approximating.
3. The admissible functions should be complete, i.e., span the space of all admissible solutions.
4. It is desirable to have the admissible functions ordered according to their number of zeros, as are the eigenfunctions for all self-adjoint ODE. This is related to optimal regularization using truncated basis functions.

In the following, we shall derive an algorithm that ensures the synthesis of admissible functions that fulfill these properties. In addition to the admissible functions \mathbf{A} , we shall also create a set of interpolating functions \mathbf{A}_I that also fulfill the constraints exactly.

Step 1: Starting from the $m \times 1$ vector \mathbf{x} , the locations of the points where the approximation is to be performed, the Vandermonde matrix \mathbf{V} is computed ordered from left to right according to increasing degree.

Step 2: Compute the QR decomposition of

$$\mathbf{V} = \mathbf{Q}\mathbf{R} \quad (5.6)$$

Since \mathbf{R} is upper triangular, its inverse \mathbf{R}^{-1} must also be upper triangular. Consequently, $\mathbf{Q} = \mathbf{V}\mathbf{R}^{-1}$ maintains the ordering of the polynomials in \mathbf{Q} according to degree. The columns of \mathbf{Q} form a set of ordered discrete orthogonal polynomials (DOP). To avoid confusion later we shall denote the DOP by $\mathbf{B} \triangleq \mathbf{Q}$. An unconstrained approximation would be computed as $\hat{\mathbf{y}} = \mathbf{B}\boldsymbol{\beta}$.

The matrix \mathbf{V} can become poorly conditioned as the degree of the polynomial rises; this can be determined from the condition number $\kappa(\mathbf{R})$. This permits an estimate of the number of significant digits in the computation. From Figure 3 it can be seen that single precision computations are possible with $d \approx 20$, which is compatible with the training performed on a GPU also using single precision computations.

Step 3: Compute the constraining matrix \mathbf{C} , whereby each row corresponds to one of the k constraints, such that $\mathbf{C}\boldsymbol{\beta} = \mathbf{0}$; this requires

$$\boldsymbol{\beta} \in \text{null}(\mathbf{C}). \quad (5.7)$$

Derivatives are computed using a pseudo-spectral approach [5.38], since finite differences yield poor estimates for derivatives when only a low number of points is available. Using singular value decomposition, a unitary basis set \mathbf{N} can be computed for the $\text{null}(\mathbf{C})$. The matrix \mathbf{N} has the dimensions $m \times m - k$. The calculation $\mathbf{B}\mathbf{N}$ would lead to a set of admissible functions. However, at this point their ordering can not be guaranteed.

Step 4: Apply an RQ¹ to $\mathbf{N} = \mathbf{R}_N \mathbf{Q}_N$, the notation \mathbf{R}_N and \mathbf{Q}_N is used to differentiate wrt \mathbf{R} and \mathbf{Q} used above. The matrix of ordered admissible functions \mathbf{A} is obtained as,

$$\mathbf{A} = \mathbf{B}\mathbf{R}_N. \quad (5.8)$$

¹N.B.: this is an RQ decomposition and not a QR decomposition.

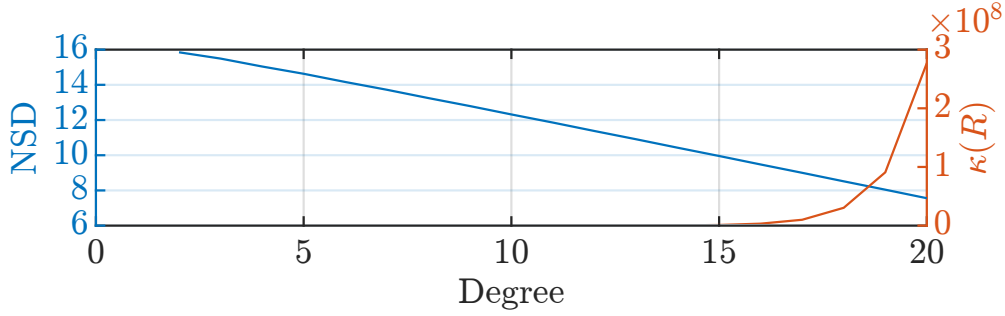


Figure 3: Invertibility of the Vandermonde matrix V and the resulting number of significant digits (NSD), left, due to the poor conditioning of $\kappa(R)$, right.

They span the complete space of solutions given m locations and k constraints; they fulfill the constraints exactly; $A^T A = I$ and they have the same ordering of zeros as would the corresponding eigenfunctions. Consequently,

$$\hat{y} = A \alpha. \quad (5.9)$$

Step 5: Given the vector of locations x_I where the interpolation of the result is desired, compute V_I the corresponding Vandermonde matrix. The interpolating admissible functions A_I are now obtained as, $A_I = V_I R^{-1} R_N$ and the interpolated curve is computed as

$$\hat{y}_I = A_I \alpha. \quad (5.10)$$

4 Rayleigh-Ritz Autoencoder

An autoencoder (AE) [5.1, 5.2] is a machine learning architecture suitable for unsupervised learning. The mapping performed by an encoder can be seen as an inverse problem and the second mapping performed by the decoder as a forward problem in classical analysis [5.22]. An AE can be described by the two mappings it performs [5.39]:

1. The encoder: $y \mapsto \alpha$, where y is the input vector α is the vector of latent variables.
2. The decoder: $\alpha \mapsto \hat{y}$, where \hat{y} is the model for the data.

The *Rayleigh-Ritz Autoencoder* proposed here uses the admissible functions presented above for the *decoder* so that,

$$\hat{y} = A \alpha. \quad (5.11)$$

This ensures that all mappings from the latent space create valid solutions for \hat{y} , consistent with the physics of the problem. It is not possible to create an invalid solution. Additionally, this structure permits the explicit calculation of error propagation. If the covariance of α , i.e. Λ_α is maintained during the progress of the computations — this can be done numerically — then the covariance of the prediction is obtained as,

$$\Lambda_{\hat{y}} = A \Lambda_\alpha A^T. \quad (5.12)$$

Both the prediction and confidence intervals for the solution $\hat{\mathbf{y}}$ can be computed from the covariance $\mathbf{\Lambda}_{\hat{\mathbf{y}}}$. This is a surprising and very positive feature since it is highly unusual for autoencoders to provide error bounds on a solution.

From the training process we have a means of estimating both α and its covariance $\mathbf{\Lambda}_{\alpha}$; consequently, for each additional measurement, we can calculate the marginal likelihood of it belonging to the previous observations. As a result, the proposed RRAE can also be used directly to implement a Gaussian Process (GP). It is beyond the scope of this paper, due to lack of space, to go into more details on GP, see e.g., [5.40].

The *encoder* is implemented as a neural network [5.4], since neural networks are known to be capable of representing any well-behaved function with arbitrary accuracy [5.5].

4.1 Cost function

The cost function to be minimized corresponds to the functional $f(x, y, y')$ as used in Eqn. 5.1. It is common to have an additional regularizing term in the cost function, e.g., variational autoencoders [5.41] performs regularization based on the Kullback–Leibler divergence $D_{KL}(Q \parallel P)$ of the distribution of the latent variables P with respect to a standard Gaussian Q .

The admissible function method proposed here is particularly well suited to regularization via truncation of the basis, i.e, removing the higher degree bases. The ordering of the DOP according to the number of zeros in the function, ensures the same ordering as the eigenfunctions of mechanical systems. Consequently, regularization via truncation of the basis, corresponds to approximating the response of the system by lower order eigenfunctions².

5 Example application

The exemplary application used to demonstrate the proposed method is shown in Figure 2 and briefly presented in the introduction. The task is to reconstruct the deformation of the tunnel wall, given the measurements from five sensors. The two additional fixed points provide four constraints,

$$y(x_a) = 0, \quad y'(x_a) = 0, \quad y(x_b) = 0, \quad \text{and} \quad y'(x_b) = 0. \quad (5.13)$$

The metal beam forming the ribs ensures a C^n continuous deformation. Multiple ribs were instrumented; however, only the data from one rib is presented here. Given, the locations of the five sensors and two fixed points there are $m = 7$ nodes for the computation of the admissible functions. To ensure completeness, the starting point is a set of degree $d = 6$ polynomials. The four independent constraints remove four degrees of freedom. Consequently, there will be three admissible functions in \mathbf{A} . These span the complete space of valid solutions, for this constellation of the physical system. The admissible functions are shown in Figure 4.

The sensor data, see Figure 5, is split into three portions: for training, normal operation, and data with a known anomaly. The box plot of the data used to train the network is shown in 6.

The functional selected here for minimization is the 1-norm of the reconstruction error,

$$S(\mathbf{y}) = \sum |\mathbf{y} - \mathbf{A}\alpha|. \quad (5.14)$$

²Unfortunately, there is not sufficient space available here to elaborate on this any further here. It is an important issue and justifies further investigation.

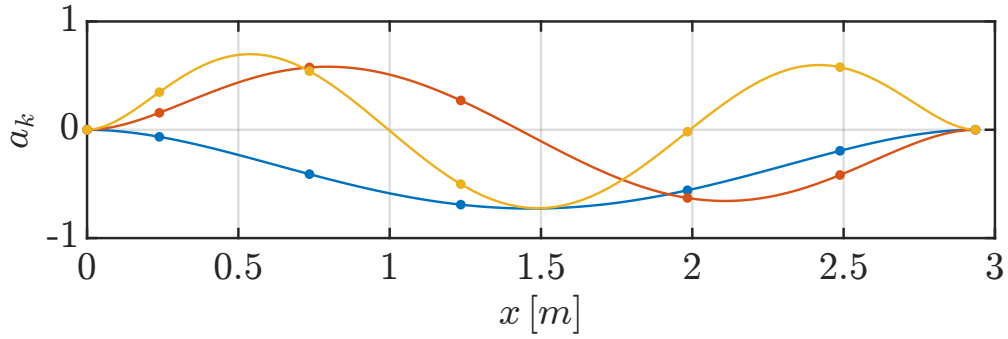


Figure 4: The three valid admissible functions for the exemplary application shown in Figure 2. The dots indicate the values of \mathbf{A} that form the unitary admissible functions. The continuous lines are the interpolated admissible functions \mathbf{A}_I at $p = 100$ points.

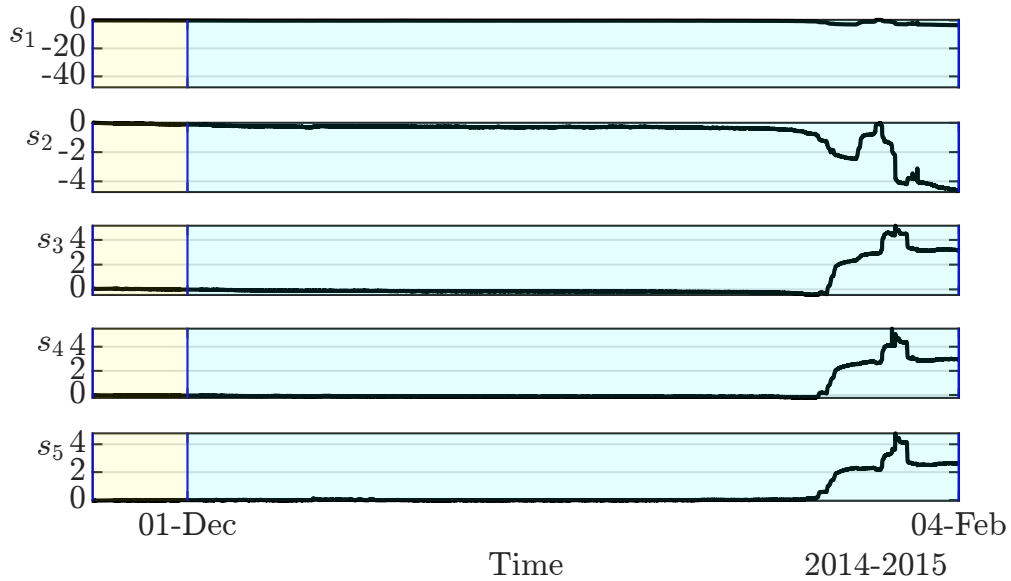


Figure 5: Raw data from the five tilt sensors at the tunnel section *SBP2*, the tilt data is measured in $[mm/m]$. The data has been segmented into three portions: (Green) The training data, with $n = 779$. (Red) Normal operative region, $n = 9159$ and (Blue) data with a known anomaly, $n = 1815$; n refers to the number of samples per sensor. Measurements was performed every $t_s = 10$ mins over the period from 23-Nov-2014 to 16-Feb-2015.

The 1-norm is selected, since they are less sensitive to outliers than 2-norm approximations, an important aspect in this application. Explicit solutions to the 2-norm case have been developed [5.8]. However, they are not easily extended to 1-norms, or other nonlinear-functionals, which require a non-linear optimization; here the advantage of the Rayleigh-Ritz autoencoder comes into play.

The portion of the sensor data corresponding to the normal operation, with its n samples, is put through the trained RRAE yielding a matrix of latent variables, $\mathbf{L} = [\alpha_1, \dots, \alpha_n]$, whereby α_j corresponds to the vector of latent variables for the j^{th} measurement. The homogeneously constrained polynomials form a

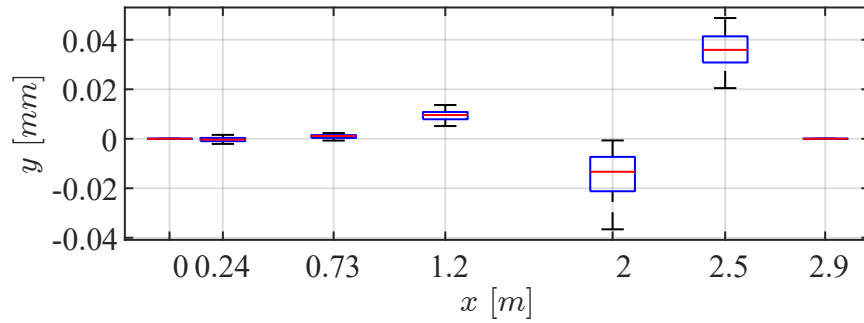


Figure 6: Boxplot of the training data at their specific locations. Note the horizontal scale is in *meters*, whereas the vertical is in *millimeters*. There are small variations that need to be computed.

vector space; consequently, it is permissible to compute distances in the latent space, i.e., wrt to polynomial coefficients. The distance of α_j from α_1 in the vector space of the latent variables is shown in Figure 7, this is a measure of the change in the latent variables as a function of time. The point in time where the deformation of the wall is triggered by the tunneling process is clearly visible.

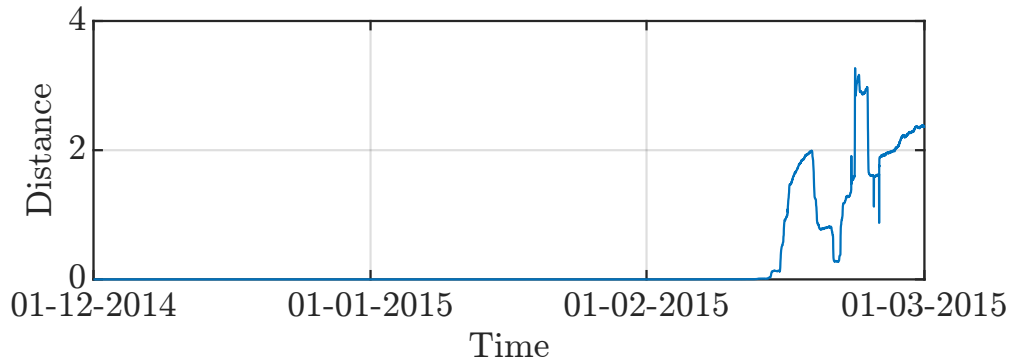


Figure 7: Distance in the latent space from α_1 to α_j , i.e., a measure for the change in the latent variables as a function of time.

Given the matrix of latent variables L and the interpolating constrained polynomials A_I , it is now possible to compute the interpolated reconstruction of the deformation of the surface $\hat{Y}_I = A_I L$. The result of monitoring the deformation of the tunnel wall over the period of approximately three months is shown in Figure 8. The critical time point where the deformation of the rib becomes significant is clearly visible. The interpolation provides a representation for the shape of the complete deformation. Note that the deformation is in millimeters, whereas the elevation is in meters. For computations to be reliable at this scale, it is essential that the a-priori physical constraints are fulfilled exactly.

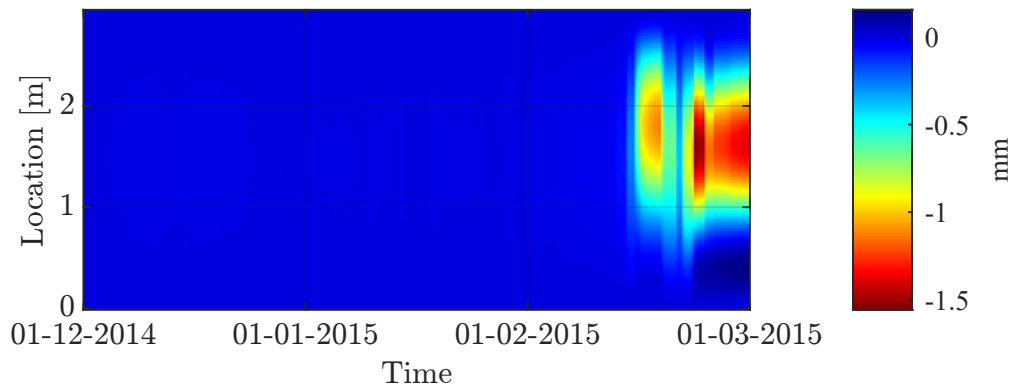


Figure 8: Interpolated reconstruction of the deformation of the surface of the tunnel $\hat{\mathbf{Y}}_I$ as a function of time. Note that the deformation is in millimeters, whereas the elevation is in meters.

6 Conclusions

The new Rayleigh-Ritz Autoencoder has been presented. The constraints and behavior of the physical system are embedded in the admissible functions used in the decoder. This ensures that modeling the system from data yields the exact and full conformity of the solution with the a-priori knowledge about the behavior of the system. This is truly a physics-informed machine learning architecture.

The new architecture has been applied to an advanced instrumentation and measurement problem; involving the fusion of data from multiple sensors. The Rayleigh-Ritz Autoencoder was used to implement a 1-norm minimizing approximation of the deformations. The constraint consistent interpolation was used to create a view of the complete surface deformation and not just the results at the sensor locations.

Acknowledgment

The authors gratefully acknowledge the financial support under the scope of the COMET program within the K2 Center “Integrated Computational Material, Process and Product Engineering (IC-MPPE)” (Project No 886385). This program is supported by the Austrian Federal Ministries for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK) and for Labour and Economy (BMAW), represented by the Austrian Research Promotion Agency (FFG), and the federal states of Styria, Upper Austria and Tyrol.

Bibliography

- [5.1] D. E. Rumelhart and J. L. McClelland, “Learning Internal Representations by Error Propagation,” in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*. Cambridge, MA, USA: MIT Press, 1987, pp. 318–362. [Online]. Available: <http://ieeexplore.ieee.org/document/6302929>
- [5.2] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.

- [5.3] M. Raissi, P. Perdikaris, and G. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686 – 707, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0021999118307125>
- [5.4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge and London: MIT Press, 2016.
- [5.5] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Networks*, vol. 2, no. 5, pp. 359–366, Jan. 1989.
- [5.6] J. Van Cranenbroeck, “Continuous beam deflection monitoring using precise inclinometers,” in *FIG Working Week 2007, Hong Kong, SAR, 13..17 May, 2007*, pp. 1–6.
- [5.7] G. Machan and V. G. Bennett, “Use of inclinometers for geotechnical instrumentation on transportation projects,” *Transportation Research E-Circular*, vol. E-C129, 2008. [Online]. Available: <http://worldcat.org/issn/00978515>
- [5.8] P. O’Leary and M. Harker, “A framework for the evaluation of inclinometer data in the measurement of structures,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 5, pp. 1237–1251, 2012.
- [5.9] P. B. Bailey and A. Zettl, “Eigenvalue and eigenfunction computations for sturm-liouville problems,” *ACM Trans. Math. Softw.*, vol. 17, no. 4, p. 491–499, dec 1991. [Online]. Available: <https://doi.org/10.1145/210232.210238>
- [5.10] S. Roberts and J. Shipman, *Two-point boundary value problems: shooting methods*, ser. Modern analytic and computational methods in science and mathematics. American Elsevier Pub. Co., 1972.
- [5.11] M. Harker and P. O’Leary, “Aproximation of physical measurement data by discrete orthogonal eigenfunctions of linear differential operators,” *Transactions of the Canadian Society for Mechanical Engineering*, vol. 41, pp. 804 – 824, 2017.
- [5.12] W. Auzinger, E. Karner, O. Koch, and E. Weinmüller, “Collocation methods for the solution of eigenvalue problems for singular ordinary differential equations,” *Opuscula Mathematica*, vol. 26, pp. 229–241, 2006.
- [5.13] M. Khanafer and S. Shirmohammadi, “Applied AI in instrumentation and measurement: The deep learning revolution,” *IEEE Instrumentation & Measurement Magazine*, vol. 23, no. 6, pp. 10–17, 2020.
- [5.14] S. Shirmohammadi and H. Al Osman, “Machine learning in measurement part 1: Error contribution and terminology confusion,” *IEEE Instrumentation & Measurement Magazine*, vol. 24, no. 2, pp. 84–92, 2021.
- [5.15] H. Al Osman and S. Shirmohammadi, “Machine learning in measurement part 2: Uncertainty quantification,” *IEEE Instrumentation & Measurement Magazine*, vol. 24, no. 3, pp. 23–27, 2021.

- [5.16] R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics, Vol. II: The New Millennium Edition: Mainly Electromagnetism and Matter*, ser. The Feynman Lectures on Physics. Basic Books, 2011. [Online]. Available: <https://books.google.at/books?id=hlRhwGK40fgC>
- [5.17] I. Gelfand and S. Fomin, *Calculus of Variations*, ser. Dover Books on Mathematics. Dover Publications, 2012.
- [5.18] J. Fojdl and R. W. Brause, “The Performance of Approximating Ordinary Differential Equations by Neural Nets,” in *2008 20th IEEE International Conference on Tools with Artificial Intelligence*. IEEE, Nov. 2008, vol. 2, pp. 457–464.
- [5.19] K. Günel and I. Gör, “Gravitational search algorithm solutions of initial value problems for ordinary differential equations,” *Adv. Math. Models Appl*, vol. 4, no. 3, pp. 232–242, 2019.
- [5.20] C. R. Gin, D. E. Shea, S. L. Brunton, and J. N. Kutz, “DeepGreen: deep learning of Green’s functions for nonlinear boundary value problems,” *Scientific Reports*, vol. 11, no. 21614, pp. 1–14, Nov. 2021.
- [5.21] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, “Physics-informed machine learning,” *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.
- [5.22] H. Goh, S. Sherifdeen, and T. Bui-Thanh, “Solving Forward and Inverse Problems Using Autoencoders,” *arXiv*, Dec. 2019.
- [5.23] P. Peng, S. Jalali, and X. Yuan, “Solving Inverse Problems via Auto-Encoders,” *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 312–323, Mar. 2020.
- [5.24] N. Levashova, A. Gorbachev, R. Argun, and D. Lukyanenko, “The Problem of the Non-Uniqueness of the Solution to the Inverse Problem of Recovering the Symmetric States of a Bistable Medium with Data on the Position of an Autowave Front,” *Symmetry*, vol. 13, no. 5, p. 860, May 2021.
- [5.25] N. Yadav, A. Yadav, and M. Kumar, *An Introduction to Neural Network Methods for Differential Equations*. Dordrecht, The Netherlands: Springer Netherlands, 2015. [Online]. Available: <https://link.springer.com/book/10.1007/978-94-017-9816-7>
- [5.26] J. Blechschmidt and O. G. Ernst, “Three ways to solve partial differential equations with neural networks — A review,” *GAMM-Mitteilungen.*, vol. 44, no. 2, p. e202100006, Jun. 2021.
- [5.27] P. Moral, *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*, ser. Probability and Its Applications. Springer New York, 2004. [Online]. Available: <https://books.google.at/books?id=8LypfuG8ZLYC>
- [5.28] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, “Scientific machine learning through physics-informed neural networks: Where we are and what’s next,” *arXiv preprint arXiv:2201.05624*, 2022.
- [5.29] Y. Xiong, R. Zuo, Z. Luo, and X. Wang, “A Physically Constrained Variational Autoencoder for Geochemical Pattern Recognition,” *Math. Geosci.*, vol. 54, no. 4, pp. 783–806, May 2022.

- [5.30] X. Meng, Z. Li, D. Zhang, and G. E. Karniadakis, “PPINN: Parareal physics-informed neural network for time-dependent PDEs,” *Comput. Methods Appl. Mech. Eng.*, vol. 370, p. 113250, Oct. 2020.
- [5.31] C. Anitescu, E. Atroshchenko, N. Alajlan, and T. Rabczuk, “Artificial neural network methods for the solution of second order boundary value problems,” *Computers, Materials and Continua*, vol. 59, no. 1, pp. 345–359, 2019.
- [5.32] H. Wang, B. Zou, J. Su, D. Wang, and X. Xu, “Variational methods and deep Ritz method for active elastic solids,” *ArXiv e-prints*, Mar. 2022.
- [5.33] G. H. Golub and C. F. Van Loan, *Matrix Computations (Johns Hopkins Studies in Mathematical Sciences)*, 3rd ed. The Johns Hopkins University Press, 1996.
- [5.34] G. W. Stewart, “On the weighting method for least squares problems with linear equality constraints,” *BIT Numerical Mathematics*, vol. 37, no. 4, pp. 961–967, Dec 1997. [Online]. Available: <https://doi.org/10.1007/BF02510363>
- [5.35] W. E and B. Yu, “The Deep Ritz Method: A Deep Learning-Based Numerical Algorithm for Solving Variational Problems,” *Communications in Mathematics and Statistics*, vol. 6, no. 1, pp. 1–12, Mar. 2018.
- [5.36] J. Chen, R. Du, and K. Wu, “A comparison study of deep Galerkin method and deep Ritz method for elliptic problems with different boundary conditions,” *ArXiv e-prints*, May 2020.
- [5.37] J. Han, J. Lu, and M. Zhou, “Solving high-dimensional eigenvalue problems using deep neural networks: A diffusion Monte Carlo like approach,” *Journal of Computational Physics*, vol. 423, p. 109792, Dec. 2020.
- [5.38] D. A. Kopriva, *Implementing Spectral Methods for Partial Differential Equations: Algorithms for Scientists and Engineers*, 1st ed. Springer Publishing Company, Incorporated, 2009.
- [5.39] P. Baldi, “Autoencoders, unsupervised learning, and deep architectures,” in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, ser. Proceedings of Machine Learning Research, I. Guyon, G. Dror, V. Lemaire, G. Taylor, and D. Silver, Eds., vol. 27. Bellevue, Washington, USA: PMLR, 02 Jul 2012, pp. 37–49. [Online]. Available: <https://proceedings.mlr.press/v27/baldi12a.html>
- [5.40] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning.*, ser. Adaptive computation and machine learning. MIT Press, 2006.
- [5.41] D. P. Kingma, M. Welling *et al.*, “An introduction to variational autoencoders,” *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

7 Introduction

Bibliography

- [5.1] D. E. Rumelhart and J. L. McClelland, “Learning Internal Representations by Error Propagation,” in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*. Cambridge, MA, USA: MIT Press, 1987, pp. 318–362. [Online]. Available: <http://ieeexplore.ieee.org/document/6302929>
- [5.2] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.
- [5.3] M. Raissi, P. Perdikaris, and G. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686 – 707, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0021999118307125>
- [5.4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge and London: MIT Press, 2016.
- [5.5] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Networks*, vol. 2, no. 5, pp. 359–366, Jan. 1989.
- [5.6] J. Van Cranenbroeck, “Continuous beam deflection monitoring using precise inclinometers,” in *FIG Working Week 2007, Hong Kong, SAR, 13..17 May, 2007*, pp. 1–6.
- [5.7] G. Machan and V. G. Bennett, “Use of inclinometers for geotechnical instrumentation on transportation projects,” *Transportation Research E-Circular*, vol. E-C129, 2008. [Online]. Available: <http://worldcat.org/issn/00978515>
- [5.8] P. O’Leary and M. Harker, “A framework for the evaluation of inclinometer data in the measurement of structures,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 5, pp. 1237–1251, 2012.
- [5.9] P. B. Bailey and A. Zettl, “Eigenvalue and eigenfunction computations for sturm-liouville problems,” *ACM Trans. Math. Softw.*, vol. 17, no. 4, p. 491–499, dec 1991. [Online]. Available: <https://doi.org/10.1145/210232.210238>
- [5.10] S. Roberts and J. Shipman, *Two-point boundary value problems: shooting methods*, ser. Modern analytic and computational methods in science and mathematics. American Elsevier Pub. Co., 1972.
- [5.11] M. Harker and P. O’Leary, “Approximation of physical measurement data by discrete orthogonal eigenfunctions of linear differential operators,” *Transactions of the Canadian Society for Mechanical Engineering*, vol. 41, pp. 804 – 824, 2017.
- [5.12] W. Auzinger, E. Karner, O. Koch, and E. Weinmüller, “Collocation methods for the solution of eigenvalue problems for singular ordinary differential equations,” *Opuscula Mathematica*, vol. 26, pp. 229–241, 2006.

- [5.13] M. Khanafer and S. Shirmohammadi, “Applied AI in instrumentation and measurement: The deep learning revolution,” *IEEE Instrumentation & Measurement Magazine*, vol. 23, no. 6, pp. 10–17, 2020.
- [5.14] S. Shirmohammadi and H. Al Osman, “Machine learning in measurement part 1: Error contribution and terminology confusion,” *IEEE Instrumentation & Measurement Magazine*, vol. 24, no. 2, pp. 84–92, 2021.
- [5.15] H. Al Osman and S. Shirmohammadi, “Machine learning in measurement part 2: Uncertainty quantification,” *IEEE Instrumentation & Measurement Magazine*, vol. 24, no. 3, pp. 23–27, 2021.
- [5.16] R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics, Vol. II: The New Millennium Edition: Mainly Electromagnetism and Matter*, ser. The Feynman Lectures on Physics. Basic Books, 2011. [Online]. Available: <https://books.google.at/books?id=hlRhwGK40fgC>
- [5.17] I. Gelfand and S. Fomin, *Calculus of Variations*, ser. Dover Books on Mathematics. Dover Publications, 2012.
- [5.18] J. Fojdl and R. W. Brause, “The Performance of Approximating Ordinary Differential Equations by Neural Nets,” in *2008 20th IEEE International Conference on Tools with Artificial Intelligence*. IEEE, Nov. 2008, vol. 2, pp. 457–464.
- [5.19] K. Günel and I. Gör, “Gravitational search algorithm solutions of initial value problems for ordinary differential equations,” *Adv. Math. Models Appl*, vol. 4, no. 3, pp. 232–242, 2019.
- [5.20] C. R. Gin, D. E. Shea, S. L. Brunton, and J. N. Kutz, “DeepGreen: deep learning of Green’s functions for nonlinear boundary value problems,” *Scientific Reports*, vol. 11, no. 21614, pp. 1–14, Nov. 2021.
- [5.21] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, “Physics-informed machine learning,” *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.
- [5.22] H. Goh, S. Sherifdeen, and T. Bui-Thanh, “Solving Forward and Inverse Problems Using Autoencoders,” *arXiv*, Dec. 2019.
- [5.23] P. Peng, S. Jalali, and X. Yuan, “Solving Inverse Problems via Auto-Encoders,” *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 312–323, Mar. 2020.
- [5.24] N. Levashova, A. Gorbachev, R. Argun, and D. Lukyanenko, “The Problem of the Non-Uniqueness of the Solution to the Inverse Problem of Recovering the Symmetric States of a Bistable Medium with Data on the Position of an Autowave Front,” *Symmetry*, vol. 13, no. 5, p. 860, May 2021.
- [5.25] N. Yadav, A. Yadav, and M. Kumar, *An Introduction to Neural Network Methods for Differential Equations*. Dordrecht, The Netherlands: Springer Netherlands, 2015. [Online]. Available: <https://link.springer.com/book/10.1007/978-94-017-9816-7>
- [5.26] J. Blechschmidt and O. G. Ernst, “Three ways to solve partial differential equations with neural networks — A review,” *GAMM-Mitteilungen.*, vol. 44, no. 2, p. e202100006, Jun. 2021.

- [5.27] P. Moral, *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*, ser. Probability and Its Applications. Springer New York, 2004. [Online]. Available: <https://books.google.at/books?id=8LypfuG8ZLYC>
- [5.28] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, “Scientific machine learning through physics-informed neural networks: Where we are and what’s next,” *arXiv preprint arXiv:2201.05624*, 2022.
- [5.29] Y. Xiong, R. Zuo, Z. Luo, and X. Wang, “A Physically Constrained Variational Autoencoder for Geochemical Pattern Recognition,” *Math. Geosci.*, vol. 54, no. 4, pp. 783–806, May 2022.
- [5.30] X. Meng, Z. Li, D. Zhang, and G. E. Karniadakis, “PPINN: Parareal physics-informed neural network for time-dependent PDEs,” *Comput. Methods Appl. Mech. Eng.*, vol. 370, p. 113250, Oct. 2020.
- [5.31] C. Anitescu, E. Atroshchenko, N. Alajlan, and T. Rabczuk, “Artificial neural network methods for the solution of second order boundary value problems,” *Computers, Materials and Continua*, vol. 59, no. 1, pp. 345–359, 2019.
- [5.32] H. Wang, B. Zou, J. Su, D. Wang, and X. Xu, “Variational methods and deep Ritz method for active elastic solids,” *ArXiv e-prints*, Mar. 2022.
- [5.33] G. H. Golub and C. F. Van Loan, *Matrix Computations (Johns Hopkins Studies in Mathematical Sciences)*, 3rd ed. The Johns Hopkins University Press, 1996.
- [5.34] G. W. Stewart, “On the weighting method for least squares problems with linear equality constraints,” *BIT Numerical Mathematics*, vol. 37, no. 4, pp. 961–967, Dec 1997. [Online]. Available: <https://doi.org/10.1007/BF02510363>
- [5.35] W. E and B. Yu, “The Deep Ritz Method: A Deep Learning-Based Numerical Algorithm for Solving Variational Problems,” *Communications in Mathematics and Statistics*, vol. 6, no. 1, pp. 1–12, Mar. 2018.
- [5.36] J. Chen, R. Du, and K. Wu, “A comparison study of deep Galerkin method and deep Ritz method for elliptic problems with different boundary conditions,” *ArXiv e-prints*, May 2020.
- [5.37] J. Han, J. Lu, and M. Zhou, “Solving high-dimensional eigenvalue problems using deep neural networks: A diffusion Monte Carlo like approach,” *Journal of Computational Physics*, vol. 423, p. 109792, Dec. 2020.
- [5.38] D. A. Kopriva, *Implementing Spectral Methods for Partial Differential Equations: Algorithms for Scientists and Engineers*, 1st ed. Springer Publishing Company, Incorporated, 2009.
- [5.39] P. Baldi, “Autoencoders, unsupervised learning, and deep architectures,” in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, ser. Proceedings of Machine Learning Research, I. Guyon, G. Dror, V. Lemaire, G. Taylor, and D. Silver, Eds., vol. 27. Bellevue, Washington, USA: PMLR, 02 Jul 2012, pp. 37–49. [Online]. Available: <https://proceedings.mlr.press/v27/baldi12a.html>

- [5.40] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning.*, ser. Adaptive computation and machine learning. MIT Press, 2006.
- [5.41] D. P. Kingma, M. Welling *et al.*, “An introduction to variational autoencoders,” *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

Automatic Synthesis of Admissible Functions for Variational Learning

Paul O’Leary, Dimitar Ninevski, Anika Terbuch

Chair of Automation

University of Leoben

Leoben, Austria

{paul.oleary, dimitar.ninevski, anika.terbuch}@unileoben.ac.at

Originally appeared as:

P. O’Leary, D. Ninevski and A. Terbuch, ”Automatic Synthesis of Admissible Functions for Variational Learning,” 2023 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), Gammarth, Tunisia, 2023, pp. 1-6, DOI: 10.1109/CIVEMSA57781.2023.10231012

©2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works

Automatic Synthesis of Admissible Functions for Variational Learning

Paul O’Leary, Dimitar Ninevski, Anika Terbuch, Chair of Automation
University of Leoben
Leoben, Austria
{paul.oleary, dimitar.ninevski, anika.terbuch}@unileoben.ac.at

10.1109/CIVEMSA57781.2023.10231012

Abstract

This paper presents a new algorithm for the automatic synthesis of admissible functions, which fulfill generalized derivative constraints. The goal is to establish a generic approach to hybrid machine learning based on the calculus of variations. This permits the embedding of a-priori knowledge into the solutions involving measurement data. The approach is optimized to be compatible with single precision as used on most GPUs. This supports the simple use of common GPUs to accelerate the computations.

The new algorithm uses a two-step approach to synthesizing the subspace for the admissible functions, ensuring the exact fulfillment of all constraints. The subspace approach avoids the necessity for Lagrange multipliers and reduces the dimensionality of the space in which the problem is solved. This approach achieves higher modes of admissible functions than past methods, while maintaining single precision. This is due to the improved numerical stability.

The proposed methods are verified by synthesizing admissible functions for a Rayleigh-Ritz approach to solving problems formulated as the calculus of variations. The brachistochrone is solved to demonstrate the solution of a forward problem. This requires the minimization of a cost function, that is an integral over an irrational function. The Zernike polynomials are synthesized and applied to the approximation of measurement data on a circular disk by a surface. This corresponds to using anisotropic bases to solve a partial differential equation (PDE) on a circular disk.

Index Terms - Physics informed machine learning, Rayleigh-Ritz, Calculus of Variations, Admissible functions

1 Introduction

This paper presents a new algorithm for the synthesis of *admissible functions* (AF) which fulfill generalized constraints. In [5.1] it is shown that admissible functions can be combined with machine learning techniques to implement physics-informed learning [5.2] in measurement systems. It is known that hybrid machine learning is an important approach when considering measurement systems [5.3]. The paper extends the authors previous work in an effort to establish a generic approach to hybrid machine learning (hybrid-ML) for the solution of problems in the calculus of variations (CoV), this we call *Variational Learning* (VL). Special attention is paid to numerical accuracy so that the admissible functions are consistent with the single precision computations common on GPUs. This work extends methods presented in [5.1] by improving numerical behavior; additionally, here both forward and inverse problems are addressed.

Some valuable and important work was carried out in the 1980s on the use of polynomials as admissible functions by Baht [5.4] and extended by others [5.5, 5.6]. They showed their potential and pointed at issues relating to numerical stability and the difficulties of computing higher-order eigenfunctions. However, the difficulties encountered are, to a large extent, associated with the approach taken and are not principally associated with the use of polynomials.

It was standard practice in the analysis of vibrational modes in engineering mechanics [5.7, 5.8] to use a coordinate system where $0 \leq x \leq 1$, since this simplifies the analytic equations. However, this leads to numerical instability of the polynomials. The use of the Gram nodes [5.9] $x_k := -1 + (2k - 1)/m$ for k points yields significantly better stability. Note: the end nodes are not exactly at $x = [-1, 1]$, but half an interval in; this avoids potential end-point singularities.

In the papers [5.4–5.6], the initial polynomials are handcrafted to fulfill the boundary conditions. Unfortunately, the combination with Gram-Schmidt orthogonalization, as used in [5.4–5.6], does not maintain the fulfillment of derivative constraints. This was observed in [5.10] and the concept of complementary boundary conditions was introduced to alleviate this issue. Furthermore, the Gram-Schmidt algorithm is known to be numerically unstable [5.11]. This all leads to admissible functions that are poorly conditioned.

In this paper we wish to address more general cases of boundary value problems (BVP) and not just vibrational modes. Classical approaches to general boundary value problems can be found in [5.12]. The methods of shooting can be used where the constraints are limited to the boundaries of the system [5.13]. Solutions to more general cases for forward problems are also available [5.14, 5.15]. These approaches are suitable for forward- but not for inverse problems.

BVP have also been addressed using machine learning, in particular Physics-informed neural networks. In many previous studies, the underlying physics have been inferred from observations [5.16]. The ultimate aim of these methods is for the latent space of an autoencoder to correspond to the parameter space, whereby the encoder learns the parameters of the governing equation for the physical system. However, the works presented in [5.17, 5.18] do not enforce the fulfillment of constraints; which is crucial, since failure to meet these constraints precisely can render the results invalid.

A review of known techniques for solving differential equations with neural networks can be found in [5.19]. In one of the approaches, which combines multilayer perceptrons and radial basis functions, the authors conclude that their method could solve the differential equation in a manner that fulfills the boundary condition exactly; however, it is computationally too expensive. With this limitation, they choose to solve the system of equations approximately and exclude solutions which don't fulfill the constraints [5.19].

Another review of hybrid-ML can be found in [5.20]. In most cases, e.g. [5.21–5.23, 5.23, 5.24], the physical constraints are incorporated into the loss function as an additional penalty term, called *soft boundary constraint enforcement*; this is akin to weighted constraints in algebraic least squares [5.11, 5.25].

In [5.26] a new approach to synthesizing constrained discrete orthogonal polynomials was introduced; this leads to unitary bases with excellent numerical stability. However, the implementation of derivative constraints is limited to using co-locative finite differences. To be more general, interstitial derivative constraints are required and pseudo-spectral definitions produce better results in sensor systems where a limited number of sensors is available.

The main contributions of this paper are:

1. A new two-step algorithm with improved numerical stability for the automatic synthesis of admissible functions with generalized derivative constraints. Exact fulfillment of the constraints is guaranteed over all the admissible functions.
2. The improved numerical stability permits single precision accuracy, compatible with common GPU formats while enabling higher-order admissible functions. This is part of the strategic decision to optimize these methods for hybrid-ML.
3. Geometric polynomials have been selected, since they are sufficient for this context and offer a more balanced computational effort than discrete orthogonal polynomials [5.27, 5.28]. They are also simpler when implementing interstitial constraints
4. A pseudo-spectral [5.29] approach to implementing the derivatives constraints was selected; since this yields better approximations where there is only a modest number of x locations, as is the case in sensor applications.
5. An application to a two-dimensional inverse problem in polar coordinates is presented involving the use of Zernike polynomials and the Fourier basis.

The approach is verified on both forward and inverse problems. The paper extends and improves the work presented in [5.30] and [5.1] yielding better numerical stability and suitability for use in hybrid-ML.

2 Calculus of variations

The focus of this paper is on general boundary value problems. The calculus of variations is defined in terms of minimizing a functional $S(y(x))$,

$$S(y(x)) = \int_{x_1}^{x_2} L(x, y(x), y'(x) \dots y^{(p)}(x)) dx, \quad (5.15)$$

i.e. minimizing the integral over the Lagrangian $L(\cdot)$ between the boundary locations x_1 and x_2 . Additionally, there are boundary, and possibly isoperimetric, conditions that must be fulfilled. Consequently, the problem can be formulated as,

$$\min_{y(x)} \left\{ \int_{x_1}^{x_2} L(x, y(x), y'(x) \dots y^{(p)}(x)) dx \right\}$$

subject to constraints on $y(x), y'(x), \dots, y^{(p)}(x)$. (5.16)

In the general case, the Lagrangian is a nonlinear function of $y(x)$ and its derivatives. This precludes the use of direct solution approaches, such as linear least squares.

2.1 Rayleigh Ritz method

The Rayleigh Ritz method computes an approximation \hat{y} to y as a linear combination of admissible functions, i.e., $\hat{y} = \mathbf{A}\alpha$, whereby $\mathbf{A} \triangleq [\mathbf{a}_1, \dots, \mathbf{a}_k]$. For a continuous system, there is an infinite set of AF that span the complete space Ω of all feasible solutions.

Theorem 1. If the functional $S(y)$ is continuous and the admissible functions are complete in Ω , then:

$$\lim_{k \rightarrow \infty} S(\hat{y}) = \inf_{y \in \Omega} S(y). \quad (5.17)$$

Problems for which there are no analytic solutions can be addressed with this method.

In discrete computations, with m distinct solution locations, there can be no more than m linearly independent AF that span the discrete Hilbert space Ω_d . If fewer AF are used, i.e., $k < m$ then the approximation lies within a subspace of Ω .

3 Polynomials as admissible functions

The Weierstrass approximation theorem states:

Theorem 2. Suppose $f(x)$ is a continuous real-valued function defined on the real interval $[a, b]$. For every $\varepsilon > 0$, there exists a polynomial $p(x, \alpha)$ such that for all $x \in [a, b]$, we have, $|f(x) - p(x, \alpha)| < \varepsilon$.

That is, from a theoretical perspective, polynomials fulfill the condition of C^n continuity and can achieve arbitrary accuracy. Consequently, the use of polynomials as admissible functions is primarily a question of computational methodology to achieve the required numerical stability.

3.1 Definition of constraints

Assuming the use of the Rayleigh-Ritz method Eqn. 5.16 can be rewritten as.

$$\begin{aligned} \min_{y(x)} \left\{ \int_{x_1}^{x_2} L(x, \mathbf{A}\alpha, \mathbf{A}^{(1)}\alpha \dots \mathbf{A}^{(p)}\alpha) dx \right\} \\ \text{subject to constraints on } \mathbf{A}\alpha, \mathbf{A}^{(1)}\alpha \dots \mathbf{A}^{(p)}\alpha. \end{aligned} \quad (5.18)$$

This indicates that we require the admissible functions \mathbf{A} and their generalized derivatives $\mathbf{A}^{(p)}$. Polynomials provide a simple and stable method of implementing this. If \mathbf{V} is the Vandermonde matrix, then its generalized derivatives, consistent with Eqn 5.18, are given by $\mathbf{V}^{(p)} = \mathbf{V}\mathbf{M}^p$; using MATLAB notation $\mathbf{M} = \text{diag}(1 : d, 1)$, with d being the degree of the polynomial.

Each of the n located constraints¹ is defined at a location x_i with a derivative degree p_i , such that $y(x_i)^{(p_i)} = 0$ with $i \in 1 \dots n$. This is mapped to a row vector $\mathbf{c}_i \mapsto \mathbf{v}(x_i)\mathbf{M}^{p_i}$ and with this we obtain $\mathbf{c}_i\alpha = 0$;

¹Note: any non-homogeneous problem can be reformulated so that all constraints are homogeneous.

whereby $v(x_i)$ corresponds to one row of the Vandermonde matrix evaluated at the x_i . The vectors c_i are stacked to form the constraining matrix C . Computing the function \mathbf{y} involves solving,

$$\mathbf{y} = V \boldsymbol{\alpha} \quad \text{given} \quad C \boldsymbol{\alpha} = \mathbf{0}. \quad (5.19)$$

Classically, this is solved using Lagrange multipliers [5.31]. However, each constraint then adds an additional dimension to the problem. Here we take a subspace approach; whereby, each constraint will reduce the dimensionality of the subspace in which the solution must lie.

3.2 A subspace approach

Given a polynomial of degree d and n independent constraints, the matrix C has the dimensions $(n \times d + 1)$ and an orthonormal vector basis set N that spans the $\text{null}(C)$ [5.11] has the dimensions $(d + 1 \times d + 1 - n)$. Consequently, starting from a polynomial vector space of dimension $d + 1$, there exists a subspace of V , i.e., admissible functions $A = V N$ of dimension $d + 1 - n$, such that,

$$\mathbf{y} = A \boldsymbol{\beta}. \quad (5.20)$$

The vector $\boldsymbol{\beta}$ now only has $d + 1 - n$ entries; whereas $\boldsymbol{\alpha}$ had $d + 1$. However, direct computation in this manner loses the ordering of the admissible functions. This can be rectified via an intermediate computation: first applying an RQ decomposition² to $N = RQ$, then computing,

$$A = V R, \quad (5.21)$$

yields an ordered set of admissible functions, since R is upper triangular. The matrix, Q being orthonormal, will not change the span of A and can be ignored. The required derivatives of the admissible functions $A^{(p)}$ are computed as,

$$A^{(p)} = V M^p R. \quad (5.22)$$

In this manner an ordered set of admissible functions A and their derivatives $A^{(p)}$ are obtained.

3.3 Two step algorithm

The subspaces proposed above can be calculated more efficiently and with better numerical stability using a two-step approach. The constraints can be split into *value* and *derivative constraints*. In the first step, a subspace B for all s value constraints is synthesized directly. This is equivalent to forcing the polynomials to have a set of s roots r_i , at the corresponding locations x_i . Then, in the second step, the subspace approach, from Section 3.2, is applied to B to implement the derivative constraints. In this manner two lower-degree problems are solved rather than one higher-degree problem; this proved to be numerically more stable.

²This is an RQ and not a QR decomposition

Step 1

A function $y_r(x)$, that fulfills the value constraints can be computed as,

$$y_r(x) = \prod_{j=1}^s (x - r_j) \sum_{k=0}^{(d-s)} \gamma_k x^k. \quad (5.23)$$

This can be formulated in vector form by now evaluating,

$$\mathbf{x}_r = \prod_{j=1}^s (x - r_j), \quad (5.24)$$

and defining the diagonal matrix $\mathbf{X}_r \triangleq \text{diag}(\mathbf{x}_r)$, we obtain

$$\mathbf{y}_r = \mathbf{X}_r \mathbf{V} \boldsymbol{\gamma}. \quad (5.25)$$

Note: it is possible to directly synthesize a constrained Vandermonde matrix $\mathbf{B} = \mathbf{X}_r \mathbf{V}$, using Horner form, in a numerically efficient and stable manner [5.32], furthermore, $\mathbf{B}^{(p)} = \mathbf{B} \mathbf{M}^p$.

Step 2

The derivative constraints are now applied to \mathbf{B} , as done in Section 3.2 and we obtain,

$$\mathbf{A}^{(p)} = \mathbf{B} \mathbf{M}^p \mathbf{R}. \quad (5.26)$$

Note: this two-step approach involves computing Vandermonde matrices of lower degrees. It is as a consequence numerically more stable and efficient.

4 Numerical testing

The first test is to synthesize a set of admissible functions with three roots; one at each end and one in the center of the support. Three versions of the admissible functions were synthesized: $\mathbf{Y}_1(x)$ with m nodes such that $0 \leq x \leq 1$; $\mathbf{Y}_2(x)$ using the Gram nodes $x_j = -1 + (2j - 1)/m$ and $\mathbf{Y}_3(x)$ using the Gram nodes but synthesized using the two-step algorithm. The admissible functions $\mathbf{Y}_1(x)$ and $\mathbf{Y}_3(x)$ are shown in Figure 1; note, the asymmetry of the AF for $0 \leq x \leq 1$.

The quality of the respective admissible functions is quantified by estimating the number of digits of accuracy that are achievable when inverting the bases. Ideally, $\mathbf{G} = \mathbf{A}^+ \mathbf{A}$ should correspond to the identity matrix. Consequently, a matrix of errors can be computed as $\mathbf{E} = \mathbf{I} - \mathbf{A}^+ \mathbf{A}$. The Frobenius norm $\varepsilon = \|\mathbf{E}\|_F$ is a measure over all the errors and $-\log_{10}(\varepsilon)$ can be considered as the number of digits that are maintained. The first derivatives of the admissible functions $\mathbf{A}^{(1)}$ should be exactly rank-1 deficient. Otherwise, there are additional null spaces, which will falsify inversion. We considered it necessary to maintain single precision accuracy to be compatible with most common GPU calculations. The results of these calculations are shown in Figure 2. As can be seen, the two-step algorithm on the Gram nodes yields the best results. It was possible to synthesize polynomials of degree $d = 27$ while maintaining single precision. We considered it necessary to maintain single precision accuracy to be compatible with most common GPU calculations.

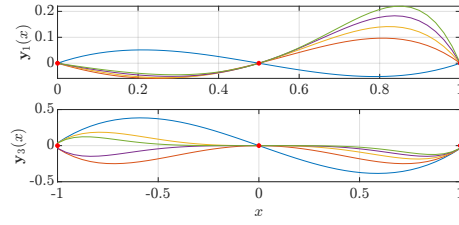


Figure 1: The first five admissible functions with the three roots synthesized using — Top: $Y_1(x)$ with m on $0 \leq x \leq 1$. Bottom: $Y_3(x)$ on the Gram nodes and the two-step algorithm.

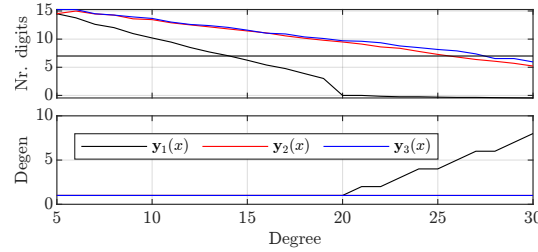


Figure 2: Measure of quality for the sets of admissible functions, Y_1 , Y_2 and Y_3 . Top: The number of significant digits maintained during the inversion of the respective bases. The horizontal line indicates the boundary for single precision. Bottom: rank degeneracy of the differentials $Y^{(1)}$.

5 Application in Hybrid Machine Learning

5.1 Brachistochrone: A forward problem

The Brachistochrone problem [5.33, Page 324] has been chosen as a demonstration for the hybrid-ML for a number of reasons:

1. This is a well-known problem first posed by Johann Bernoulli to the readers of Acta Eruditorum in June, 1696 to which Newton [5.34] published a solution. The availability of parametric solutions enables the comparison with the hybrid-ML approach.
2. The minimization problem involves the integral over an irrational function, see Eqn 5.27. This makes it suitable for the non-linear optimization algorithms made available by machine learning environments.
3. It is a non-homogeneous forward problem, demonstrating the application of hybrid-ML to such cases.
4. A non-linear coordinate transformation is required wrt. the parametric solution, if good numerical stability is to be achieved.

These features make it a suitable case to test the hybrid-ML. The equation for the action $S(y)$ in the case of the Brachistochrone is:

$$S(y) = \frac{1}{\sqrt{2g}} \int_{x_1}^{x_2} \sqrt{\frac{1 + (dy/dx)^2}{y}} dx \quad (5.27)$$

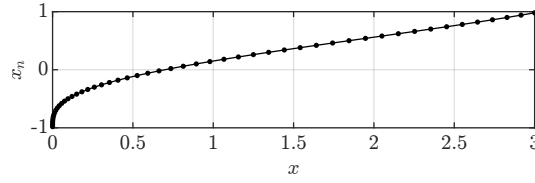


Figure 3: The coordinate transformation for the Brachistochrone admissible functions; x are the values obtained from the parametric solution, whereas x_n are the nodes on which the admissible functions are synthesized.

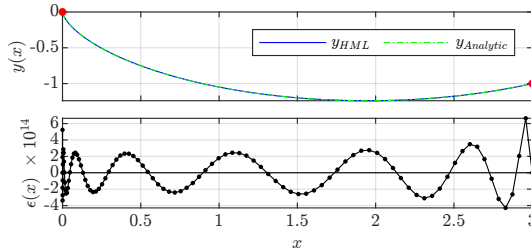


Figure 4: Top: The analytic solution and the hybrid-ML solution, with $d = 15$ and with $m = 100$ nodes. Bottom: the difference $\varepsilon(x)$ between the parametric solution and hybrid-ML as a function of x ; note, this is scaled by 10^{14} .

with the boundary conditions $y(x_1) = 0$ and $y(x_2) = b$. The endpoint solution is,

$$b = \frac{2\phi_B - \sin(2\phi_B)}{1 - \cos(2\phi_B)}, \quad (5.28)$$

This is solved in a non-linear manner to obtain values of ϕ_B . Then for $0 \leq \phi \leq \phi_B$ the parametric solutions for x and y are computed as:

$$x = \frac{2\phi - \sin(2\phi)}{1 - \cos(2\phi_B)} \quad \text{and} \quad y = \frac{2\phi - \cos(2\phi)}{1 - \cos(2\phi_B)}.$$

The parametric solution yields values for x which are highly non-uniform in their spacing. A dramatic improvement in the numerical stability of the admissible functions is obtained by introducing the coordinate transformation shown in Figure 3. These solutions will later be compared with the results from solving the problem via hybrid-ML. The admissible functions are now synthesized on the nodes x_n , the Gram nodes, and then applied at the locations x . This points to a future area of research on optimized node placement for the calculus of variations.

In this non-homogeneous problem, \mathbf{y} is considered as consisting of a particular solution \mathbf{y}_p and a homogeneous portion \mathbf{y}_h , i.e., $\mathbf{y} = \mathbf{y}_p + \mathbf{y}_h$. Whereby, \mathbf{y}_p is the minimum degree polynomial that interpolates the boundary constraints, in this example, simply a straight line; $\mathbf{y}_h = \mathbf{A} \boldsymbol{\alpha}$ is the homogeneous solution [5.35]. The task now is to find $\boldsymbol{\alpha}$ that minimizes the discrete equivalent of Eqn 5.35, i.e.,

$$\min_{\boldsymbol{\alpha}} S(\mathbf{y}) = \frac{1}{\sqrt{2g}} \text{trapz} \left\{ \sqrt{\left(\mathbf{1} + (\mathbf{y}^{(1)})^2 \right) \circ / (\mathbf{y})} \right\}, \quad (5.29)$$

here $\text{trapz}(\cdot)$ indicates the trapezoidal equivalent to integration and the symbol $\circ/$ denotes the Hadamard division, i.e., an element-wise division of the two vectors.

In the hybrid-ML solution $S(y)$ is used as the loss function. In this example, it is an irrational function of $y(x)$ and its first derivative $y^{(1)}$. Machine learning offers efficient and stable solutions to such non-linear tasks.

The comparison of the results obtained from the hybrid-ML and the known parametric solution is shown in Figure 4. As can be seen, the hybrid-ML agrees with past solutions within an error of 10^{-14} , this is beyond single precision and can be considered as an exact solution in this context. That is, the hybrid-ML has successfully learnt the solution to the boundary value problem.

5.2 Zernike polynomials: An inverse problem

The Zernike polynomials are applied in optics [5.36] to model and describe distortions of lenses, i.e., deformations on a spherical field. The even and odd Zernike polynomials are defined as:

$$Z_n^m(r, \phi) = R_n^m(r) \cos(m\phi), \quad (5.30)$$

$$Z_n^{-m}(r, \phi) = R_n^m(r) \sin(m\phi), \quad (5.31)$$

whereby the radial polynomials are given by,

$$R_n^m(r) = \sum_{k=0}^{(n-m)/2} \frac{(-1)^k (n-k)!}{k! \{(n+m)/2 - k\}! \{(n-m)/2 - k\}!} r^{n-2k} \quad (5.32)$$

and fulfill the orthogonality condition,

$$\int_0^1 R_n^m(r) R_{n'}^m(r) r dr = \frac{1}{2(n+1)} \delta_{n,n'}, \quad (5.33)$$

where $\delta_{p,q}$ is the Kronecker delta. The specific selection of the polynomials in conjunction with a corresponding cyclic mode ensures that the constraints required at zero to avoid discontinuities are fulfilled. The polynomials for modes $m = 0 \dots 2$ up to degree $d = 6$ are shown in Figure 5. These polynomials have different constraints depending on the mode number: $y^{(1)}(0) = 0$ for mode 0; $y^{(1)}(0) = 0$ and $y(0) = 0$ for mode 1; $y^{(2)}(0) = 0$, $y^{(1)}(0) = 0$ and $y(0) = 0$ for mode 2.

The polynomials are used here to model distortions of a disk. The measurement data was acquired on $n_r = 6$ concentric rings, with $m = 298$ samples per ring. The measurement data and the composite Zernike surface model are shown in Figure 6; whereas, the individual mode $m = 0 \dots 3$ approximations are shown in Figure 7.

More details on applying the admissible functions in hybrid-ML can be found in [5.1], together with additional examples.

6 Conclusions

The newly proposed two-step synthesis algorithm for synthesizing admissible functions is numerically more stable than the past solutions. It provides a mechanism to automatically convert constraint definitions to

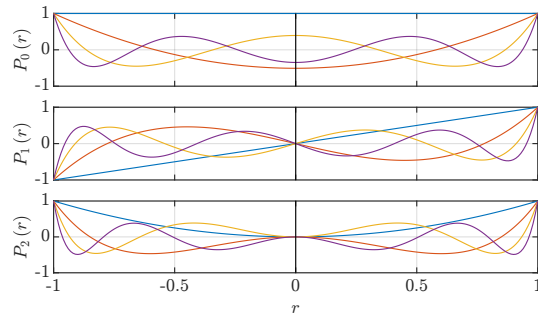


Figure 5: The radial Zernike polynomials for the modes $m = 0 \dots 2$, from top to bottom. They are defined on the range $0 \leq r \leq 1$; however, they are plotted here $-1 \leq r \leq 1$ to make their even and odd symmetries visible. Note the alternating even odd nature of the polynomials.

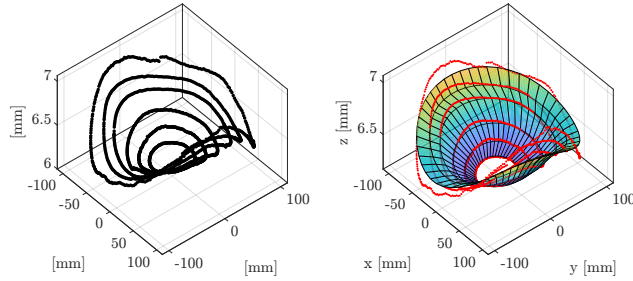


Figure 6: Left: Elevation measurement data for a disk acquired on six concentric circular paths. The data forms a regular lattice with $m = 298$ samples per track. Right: The Zernike surface approximation as a composition of the modes $z_m = 0 \dots 4$, the red dots are the data. Higher modes would be required for a more complete modeling of the data.

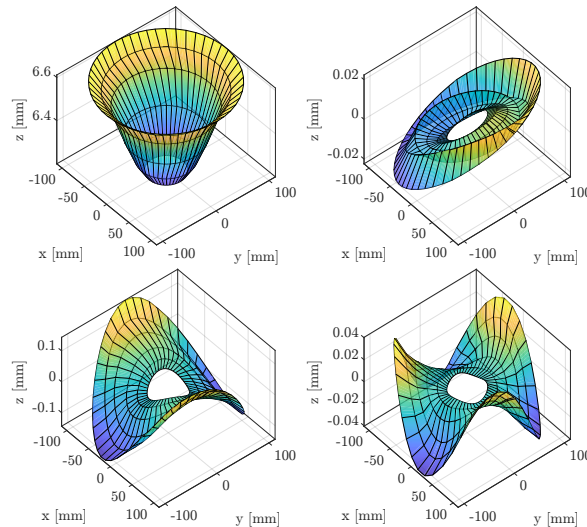


Figure 7: The first four modes $z_m = 0 \dots 3$ of the Zernike approximation of the data, with a polynomial of degree $d = 6$. The final approximation is the sum of these modes. Each mode corresponds to a specific type of warping of the disk.

admissible functions, via a subspace approach. The solution of the brachistochrone problem demonstrates the successful use of hybrid-ML to solve a strongly non-linear forward problem. The Zernike approximations show the applicability to two-dimensional inverse problems. Future work will focus on identifying optimal node placements for the synthesis process since this has been observed to offer order of magnitude improvement in the numerical stability.

Acknowledgment

The authors gratefully acknowledge the financial support under the scope of the COMET program within the K2 Center “Integrated Computational Material, Process and Product Engineering (IC-MPPE)” (Project No 886385). This program is supported by the Austrian Federal Ministries for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK) and for Labour and Economy (BMAW), represented by the Austrian Research Promotion Agency (FFG), and the federal states of Styria, Upper Austria and Tyrol.

Bibliography

- [5.1] A. Terbuch, P. O’Leary, D. Ninevski, E. J. Hagendorfer, E. Schlager, A. Windisch, and C. Schweimer, “A Rayleigh-Ritz Autoencoder,” in *2023 IEEE International Instrumentation and Measurement Technology Conference Conference (I2MTC 2023)*, Kuala Lumpur, Malaysia, 2023, accepted.
- [5.2] M. Raissi, P. Perdikaris, and G. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686 – 707, 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0021999118307125>
- [5.3] M. Khanafer and S. Shirmohammadi, “Applied AI in instrumentation and measurement: The deep learning revolution,” *IEEE Instrumentation & Measurement Magazine*, vol. 23, no. 6, pp. 10–17, 2020.
- [5.4] R. B. Bhat, “Plate deflections using orthogonal polynomials,” *Journal of Engineering Mechanics-ASCE*, vol. 111, pp. 1301–1309, 1985.
- [5.5] G. Oosterhout, P. van der Hoogt, and R. Spiering, “Accurate calculation methods for natural frequencies of plates with special attention to the higher modes,” *Journal of Sound and Vibration*, vol. 183, no. 1, pp. 33–47, 1995. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0022460X85702374>
- [5.6] R. Bhat, “Transverse vibrations of a rotating uniform cantilever beam with tip mass as predicted by using beam characteristic orthogonal polynomials in the rayleigh-ritz method,” *Journal of Sound and Vibration*, vol. 105, no. 2, pp. 199–210, 1986. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0022460X86901495>
- [5.7] J. Meriam, L. Kraige, and J. Bolton, *Engineering Mechanics: Dynamics, 8th Edition*. Wiley, 2015. [Online]. Available: <https://books.google.at/books?id=xrl2CwAAQBAJ>

- [5.8] J. Meriam and L. Kraige, *Statics*, ser. Engineering Mechanics: Statics. Wiley, 2008. [Online]. Available: <https://books.google.at/books?id=0gLE-EdUa9oC>
- [5.9] R. Barnard, G. Dahlquist, K. Pearce, L. Reichel, and K. Richards, “Gram polynomials and the kummer function,” *J. Approx. Theory*, vol. 94, no. 1, p. 128–143, jul 1998. [Online]. Available: <https://doi.org/10.1006/jath.1998.3181>
- [5.10] H. Baruh and S. Tadikonda, “Another look at admissible functions,” *Journal of Sound and Vibration*, vol. 132, no. 1, pp. 73–87, 1989. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0022460X89908729>
- [5.11] G. H. Golub and C. F. Van Loan, *Matrix Computations (Johns Hopkins Studies in Mathematical Sciences)*, 3rd ed. The Johns Hopkins University Press, 1996.
- [5.12] P. B. Bailey and A. Zettl, “Eigenvalue and eigenfunction computations for sturm-liouville problems,” *ACM Trans. Math. Softw.*, vol. 17, no. 4, p. 491–499, dec 1991. [Online]. Available: <https://doi.org/10.1145/210232.210238>
- [5.13] S. Roberts and J. Shipman, *Two-point boundary value problems: shooting methods*, ser. Modern analytic and computational methods in science and mathematics. American Elsevier Pub. Co., 1972.
- [5.14] M. Harker and P. O’Leary, “Aproximation of physical measurement data by discrete orthogonal eigenfunctions of linear differential operators,” *Transactions of the Canadian Society for Mechanical Engineering*, vol. 41, pp. 804 – 824, 2017.
- [5.15] W. Auzinger, E. Karner, O. Koch, and E. Weinmüller, “Collocation methods for the solution of eigenvalue problems for singular ordinary differential equations,” *Opuscula Mathematica*, vol. 26, pp. 229–241, 2006.
- [5.16] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, “Physics-informed machine learning,” *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, 2021.
- [5.17] H. Goh, S. Sherifdeen, and T. Bui-Thanh, “Solving Forward and Inverse Problems Using Autoencoders,” *arXiv*, Dec. 2019.
- [5.18] P. Peng, S. Jalali, and X. Yuan, “Solving Inverse Problems via Auto-Encoders,” *IEEE Journal on Selected Areas in Information Theory*, vol. 1, no. 1, pp. 312–323, Mar. 2020.
- [5.19] N. Yadav, A. Yadav, and M. Kumar, *An Introduction to Neural Network Methods for Differential Equations*. Dordrecht, The Netherlands: Springer Netherlands, 2015. [Online]. Available: <https://link.springer.com/book/10.1007/978-94-017-9816-7>
- [5.20] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, “Scientific machine learning through physics-informed neural networks: Where we are and what’s next,” *arXiv preprint arXiv:2201.05624*, 2022.
- [5.21] Y. Xiong, R. Zuo, Z. Luo, and X. Wang, “A Physically Constrained Variational Autoencoder for Geochemical Pattern Recognition,” *Math. Geosci.*, vol. 54, no. 4, pp. 783–806, May 2022.

- [5.22] X. Meng, Z. Li, D. Zhang, and G. E. Karniadakis, “PPINN: Parareal physics-informed neural network for time-dependent PDEs,” *Comput. Methods Appl. Mech. Eng.*, vol. 370, p. 113250, Oct. 2020.
- [5.23] C. Anitescu, E. Atroshchenko, N. Alajlan, and T. Rabczuk, “Artificial neural network methods for the solution of second order boundary value problems,” *Computers, Materials and Continua*, vol. 59, no. 1, pp. 345–359, 2019.
- [5.24] H. Wang, B. Zou, J. Su, D. Wang, and X. Xu, “Variational methods and deep Ritz method for active elastic solids,” *ArXiv e-prints*, Mar. 2022.
- [5.25] G. W. Stewart, “On the weighting method for least squares problems with linear equality constraints,” *BIT Numerical Mathematics*, vol. 37, no. 4, pp. 961–967, Dec 1997. [Online]. Available: <https://doi.org/10.1007/BF02510363>
- [5.26] P. O’Leary and M. Harker, “A framework for the evaluation of inclinometer data in the measurement of structures,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 5, pp. 1237–1251, 2012.
- [5.27] M. Harker and P. O’Leary. (2014) Discrete orthogonal polynomial toolbox: Dopbox version 1.8. [Online]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/41250>
- [5.28] ———, “Approximation of physical measurement data by discrete orthogonal eigenfunctions of linear differential operators,” *Transactions of the Canadian Society for Mechanical Engineering*, vol. 41, pp. 804–824, 12 2017.
- [5.29] D. Gottlieb, S. A. Orszag, and E. Turkel, “Stability of pseudospectral and finite-difference methods for variable coefficient problems,” *Mathematics of Computation*, vol. 37, no. 156, pp. 293–305, 1981. [Online]. Available: <http://www.jstor.org/stable/2007427>
- [5.30] P. O’Leary and D. Ninevski, “A computational framework for generalized constrained inverse problems,” in *2022 IEEE 5th International Conference on Industrial Cyber-Physical Systems (ICPS)*, 2022, pp. 01–06.
- [5.31] V. Jovanovic and S. Koshkin, “The ritz method with lagrange multipliers,” *arXiv: Numerical Analysis*, 2013.
- [5.32] P. O’Leary, R. Ritt, and M. Harker, “Constrained polynomial approximation for inverse problems in engineering,” in *Proceedings of the 1st International Conference on Numerical Modelling in Engineering*, M. Abdel Wahab, Ed. Singapore: Springer Singapore, 2019, pp. 225–244.
- [5.33] J. Oprea, *Differential Geometry and Its Applications*, ser. Classroom resource materials. Mathematical Association of America, 2007. [Online]. Available: <https://books.google.at/books?id=xb48zk0wJfIC>
- [5.34] I. Newton, “De ratione temporis quo grave labitur per rectam data duo puncta conjungentem, ad tempus brevissimum quo, vi gravitatis, transit ab horum uno ad alterum per arcum cycloidis,” *Philosophical Transactions (1683-1775)*, vol. 19, pp. 424–425, 1695. [Online]. Available: <http://www.jstor.org/stable/102342>

- [5.35] J. Burns, *Introduction to the Calculus of Variations and Control with Modern Applications*, ser. Chapman & Hall/CRC Applied Mathematics & Nonlinear Science. Taylor & Francis, 2013. [Online]. Available: <https://books.google.at/books?id=pah7AAAAQBAJ>
- [5.36] V. Lakshminarayanan and A. Fleck, “Zernike polynomials: a guide,” *Journal of Modern Optics*, vol. 58, no. 7, pp. 545–561, 2011.

Extended Rayleigh-Ritz Autoencoder with Distribution-Free Statistics

**Anika Terbuch¹, Dimitar Ninevski¹, Paul O’Leary¹,
Matthew Harker², Manfred Mücke³**

¹ Chair of Automation, University of Leoben, Leoben, Austria
{anika.terbuch, dimitar.ninevski, paul.oleary}@unileoben.ac.at

² Departement de génie des systèmes, École de Technologie Supérieure, Montreal,
Canada
matthew.harker@etsmtl.ca

³ Materials Center Leoben, Leoben, Austria
manfred.muecke@mcl.at

Submitted

IEEE International Instrumentation and Measurement Technology Conference
(I2MTC) 2024

Submission Date: November 7, 2023

Status: Under Review

Extended Rayleigh-Ritz Autoencoder with Distribution-Free Statistics

Anika Terbuch¹, Dimitar Ninevski¹, Paul O’Leary¹,
Matthew Harker², Manfred Mücke³

¹ Chair of Automation
University of Leoben
Leoben, Austria

{anika.terbuch,dimitar.ninevski,paul.oleary}@unileoben.ac.at

² Département de génie des systèmes, École de Technologie Supérieure
Montreal, Canada,
matthew.harker@etsmtl.ca

³ Materials Center Leoben
Leoben, Austria
manfred.muecke@mcl.at

Abstract

This paper presents a detailed analysis of an extended Rayleigh-Ritz Autoencoder which uses distribution-free statistics to achieve stability with respect to non-Gaussian data. This provides consistent results for sensor data with both Gaussian and non-Gaussian perturbations. The necessity for handling non-Gaussian data in sensor applications is documented by the behavior of inclinometer sensors where the perturbations are characterized by Cauchy-Lorentz distribution. In such cases variance does not provide a reliable measure for uncertainty; consequently, 1-norm error measures are investigated thoroughly. Furthermore, the stability of the basis functions is improved via a new synthesis approach; enabling the use of single precision computations while achieving polynomials of higher degree. The concept of Lebesgue functions

and constants is extended to constrained bases, yielding a theoretical upper bound on the interpolation error of the autoencoder. The method is applied to a previously well-documented set of real measurement data, this enables the object determination of the improvement obtained by the new calculation approach.

Keywords: Physics-informed machine learning, Rayleigh-Ritz, Admissible functions, Structural health monitoring, Measurement uncertainty, Distribution-free statistics, Lebesgue constant

1 Introduction

The concept of the Rayleigh-Ritz Autoencoder (RRAE) was first presented at the I2MTC in 2023, see [5.1]. This paper presents the results of new research on extensions to the RRAE, a new architecture for physics-informed machine [5.2–5.4]. A schematic diagram of the RRAE architecture is shown in Figure 1.

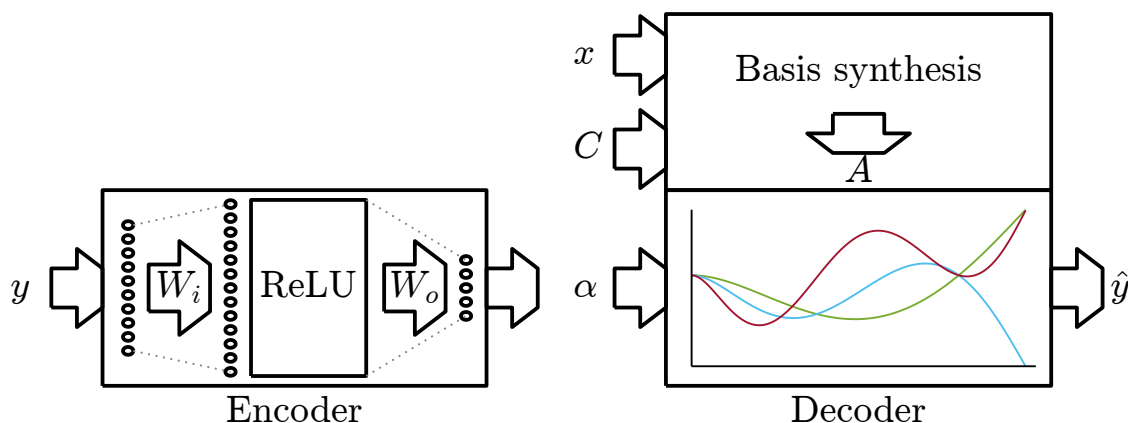


Figure 1: Schematic of the Rayleigh-Ritz autoencoder. The *encoder* is a neural network, its purpose is to map the inputs y to the latent variables α . The rectified linear unit layer (ReLU) is used as the non-linear element. The *basis synthesis* computes a set of admissible functions A , given the vector of locations x and a matrix C defining the constraints. This is performed prior to training the network. The *decoder* computes the approximation $\hat{y} = A\alpha$. The learnable parameters of the encoder are denoted with W_i and W_o .

The first major extension is the introduction of distribution-free statistical methods [5.5] to achieve good stability with respect to sensor data having non-Gaussian perturbations. In the context of the Guide to the expression of uncertainty in measurement, also known as GUM [5.6] and the International vocabulary of basic and general terms in metrology (VIM) [5.7] the term uncertainty is defined as *non-negative parameter characterizing the dispersion of the quantity values being attributed to a measurand, based on the information used*. Commonly, uncertainty is expressed in terms of standard deviation and referred to as *standard uncertainty*. However, there are proposals, e.g. [5.8] that GUM should be revised. The current version of GUM is strongly focused on standard deviation as a measure of variability which is implicitly a 2-norm context.

To document the necessity for the introduction of more general measures for uncertainty, consider data measured with the Murata SCA103T 3D-mems based inclinometer: it uses two accelerators, delivering the signals a_x and a_y respectively. The total acceleration is calculated as $a = a_x + a_y$ and the inclination as $\phi = a_x - a_y$. The bivariate histogram for $m = 183000$ measurements is shown in Figure 2. The individual histograms for a and ϕ are also shown in Figure 2. The large number of observations ensures that the results are statistically significant.

The issue is that the perturbations of inclination ϕ are not Gaussian in nature. They are, however, correctly modeled by a three-term Cauchy-Lorentz distribution,

$$f(x; x_o, \gamma, I_o) = I_o \left\{ \frac{\gamma^2}{(x - \bar{x})^2 + \gamma^2} \right\}, \quad (5.34)$$

with the values $x_o = -0.728$, $\gamma = 5.606$ and $I_o = 0.061$. In this real measurement case, the mean provides an unstable measure for the central tendency of the data, and the standard uncertainty is a very poor measure for the spread of the data. Non-Gaussian data is surprisingly common; in [5.9] it was documented that only a small percentage (5.5%) of the data samples fulfilled the conditions for normality, due mainly to their kurtosis or skewness [5.10].

Consequently, a thorough investigation of distribution-free statistical methods and 1-norms, in conjunction with the RRAE is presented. The new approaches yield stable and consistent results for both Gaussian and non-Gaussian perturbations.

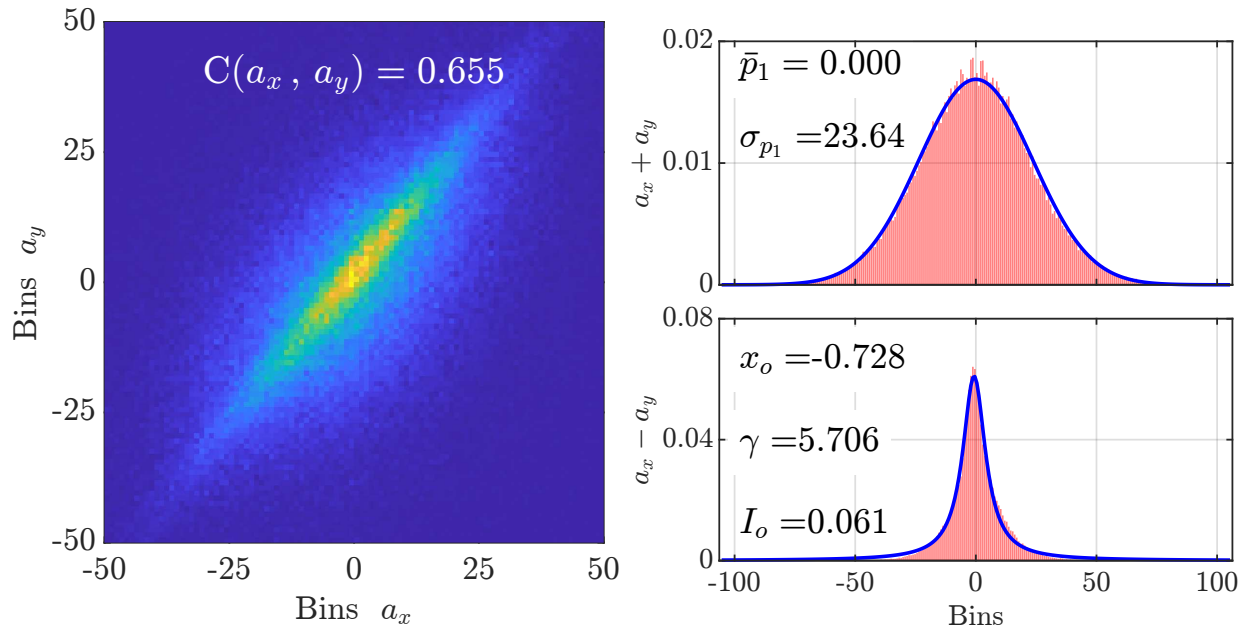


Figure 2: Left: Bivariate histogram for a_x and a_y . The color indicates the frequency and is proportional to the probability $p(a_x, a_y)$. Right: the histograms of the total acceleration $a = a_x + a_y$ and the inclination $\phi = a_x - a_y$, together with their respective probability density functions. A total of $m = 183000$ measurement samples were used for these statistics. Note the probability distribution function for inclination ϕ is Cauchy-Lorentz, and *not* Gaussian.

The second major extension is the utilization of a new approach [5.11] to synthesizing constrained admissible functions; these are required for the Rayleigh-Ritz method [5.12]. The goal is to achieve improved numerical stability, using single-precision computations, while calculating higher-order admissible functions. This enables the use of GPU for high throughput applications without the loss of quality. Additionally, the concept of Lebesgue functions and constants [5.13] is extended to constrained admissible functions. This provides theoretical upper bounds on the interpolation error of the RRAE. It also provides a means of numerically quantifying the improvement in possible uncertainty through the implementation of the constraints.

The new approaches are verified using the same data set as presented in [5.1]. This enables an objective evaluation of the improvements with respect to the previous RRAE implementation.

1.1 Distance and similarity

The goal of modeling is to maximize the similarity between the data \mathbf{y} and the corresponding model $\hat{\mathbf{y}}$, while obtaining dimensionality reduction; consequently, measures of similarity are required. In addition to minimizing some distance measure, we may also wish to minimize the spread of the model values across a complete training set, i.e., to minimize the uncertainty of the model prediction.

Classically, the 2-norm distance [5.14] is used for autoencoders as a measure, i.e., the residual \mathbf{r} is defined $\mathbf{r} \triangleq \mathbf{y} - \hat{\mathbf{y}}$ and the loss function is the $|\mathbf{r}|_2^2$. However, for non-Gaussian data, the 2-norm is not an appropriate measure of similarity and often leads to unstable results. In this paper 1-norms, median, and inter-quantile ranges (IQR) are thoroughly investigated as alternative measures for distance, central tendency, and spread of distributions. Minimizing with respect to these measures is fundamentally non-linear; this is where the use of the neural networks (NN) [5.15, 5.16] and modern training algorithms has its advantages. The proposed measures are particularly interesting since they yield stable results for both Gaussian and non-Gaussian cases.

2 The Rayleigh-Ritz Method

A typical calculus of variations problem [5.17] can be formulated as

$$\min_y C(y) = \min_y \int_a^b f(t, y, y') dt \quad (5.35)$$

Subject to a set of constraints.

Here the solution y is a function of time. The Rayleigh-Ritz method [5.12] is a direct approach to finding approximate solutions to such problems; whereby, \mathbf{y} is replaced by an approximation $\hat{\mathbf{y}}$, a linear combination of admissible functions \mathbf{a}_k , i.e.,

$$\hat{\mathbf{y}} = \sum_{k=0}^d \mathbf{a}_k(x) \alpha_k. \quad (5.36)$$

or in matrix form $\hat{\mathbf{y}} = \mathbf{A}(x)\boldsymbol{\alpha}$; whereby $\boldsymbol{\alpha}$ is the vector of coefficients. This requires all the admissible functions, $\mathbf{A}(x)$, to fulfill the constraints exactly in a homogeneous manner, if the correspondence to the physical system is to be maintained. Conversely, if this condition is fulfilled, the solution $\hat{\mathbf{y}}$ provided by the RRAE, using the admissible functions, cannot violate the physical constraints defined by the system. The

task has now been transformed to finding the values of the coefficient vector α that minimize $C(\hat{\mathbf{y}})$. This will be the task of the encoder portion of the RRAE.

A number of papers claim to implement the Ritz method; however, strictly speaking, the papers are proposing an additional term in the cost functions to fulfill constraints [5.18–5.21]; this is undesirable, since it is tantamount to modifying the functional. That is, they are redefining the problem to be solved and the exact correspondence to the physics of the problem is lost. A further problem in these papers is that the trial functions used are not strictly admissible in and of themselves. This leads to a further divergence between the model solution on the physics.

2.1 The nature of constraints

Dirichlet, Neumann, Cauchy, or Robin constraints are the most common found in literature [5.22]. They can be summarized as linear combinations of values $\hat{\mathbf{y}}$ and derivatives of the model $\hat{\mathbf{y}}^{(1)}$ at the boundaries $[a, b]$ of the support:

$$k_{11}\hat{y}(a) + k_{12}\hat{y}^{(1)}(b) = c_1 \quad (5.37)$$

$$k_{21}\hat{y}(a) + k_{22}\hat{y}^{(1)}(b) = c_2. \quad (5.38)$$

Additionally, isoperimetric constraints must be considered, i.e., integrals over functions along the curve. For example, the constraint on the catenary curve, is,

$$\int_a^b \sqrt{1 + y'(x)^2} dx = c \quad (\text{a constant}). \quad (5.39)$$

3 The Rayleigh-Ritz Autoencoder

The RRAE is in principle an autoencoder [5.14, 5.23], consisting of an encoder and decoder. The key issue in the RRAE is the *decoder*, it is implemented as a pre-trained fully connected layer, in which the values of $A(x)$ are stored. These values are static and not modified during training. This ensures that the solution $\hat{y}(x)$ can not violate the constraints defined by the physics of the system, since $\hat{\mathbf{y}} = A(x)\alpha$ and the admissible functions $A(x)$ fulfill the constraints exactly.

The *loss function* corresponds to a direct implementation of the functional defined by physics, see Eqn 5.35; these is no need for additional terms to deal with constraints.

The *encoder* is implemented as a classical neural network [5.24, 5.25]. It is known that neural networks can theoretically approximate arbitrary nonlinear functions [5.25]. The task of training is to determine the weights and biases of the encoder that directly minimize the loss function.

3.1 Synthesizing admissible functions

The algorithmic synthesis of admissible functions for general constraints is considered to be one of the major challenges [5.11] when implementing the Rayleigh-Ritz method. In the approach presented in [5.26] the goal is to learn the continuous operator that fulfills the constraints. However, for the different groups of constraints different algorithms are suggested; consequently, this approach does not generalize easily. In [5.27] penalty

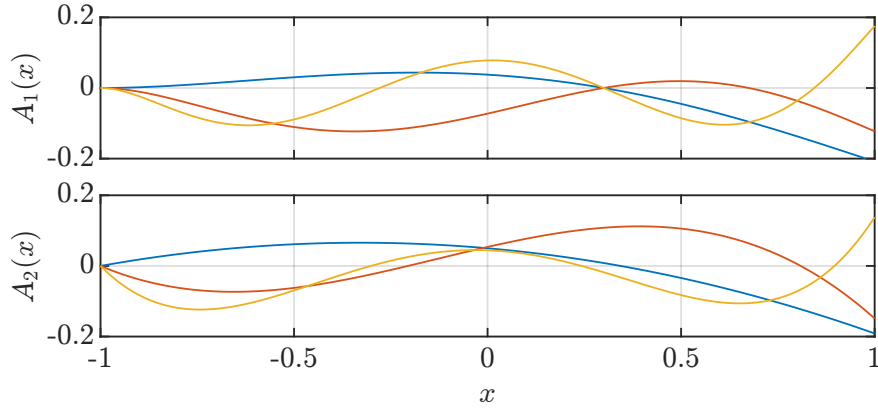


Figure 3: Two examples of orthonormal admissible functions synthesized using the proposed technique. In each case the first three admissible functions are plotted, i.e., these are truncated basis function sets. Case 1: Corresponds to a cantilever with an additional support, i.e., an internal constraint at $y(0.3) = 0$. Case 2: Demonstrates a value constraint at the boundary, combined with integro-differential constraint.

terms are added to the loss function; this is undesirable since the functional is modified and the one-to-one relationship to the physics is lost.

Here, the hard enforcement of constraints is solved in a systematic manner using constrained discrete orthogonal polynomials [5.28] and a new synthesis approach [5.11] that achieves better numerical stability when computed in single precision, compatible with the use of GPUs. The approach enables the implementation of any constraint that can be modelled as a linear operator L applied to \hat{y} , i.e., $L\hat{y} = 0$, this includes all the cases discussed in Section 2.1.

Two examples of the synthesis of admissible functions for such constraints are demonstrated here, see Figure 3:

1. A cantilever with an additional support. This requires constraints on the function and the first two derivatives, i.e., $y(-1) = 0$, $y^{(1)}(-1) = 0$, $y^{(2)}(1) = 0$, and an internal constraint $y(0.3) = 0$, which is not located at the boundaries.
2. Implementing an isoperimetric integro-differential constraint of the form:

$$\hat{y}^{(2)}(1) + \int_{-1}^1 \hat{y}(x) dx = 0. \quad (5.40)$$

An in-depth description of the synthesis algorithm for the discrete orthogonal basis functions can be found in [5.11].

3.2 A new approach to uncertainty in machine learning

Distribution-free statistical methods [5.5] provide a means of characterizing data without requiring a specific probability distribution function (PDF). The metrics are valid for both Gaussian and non-Gaussian data. The

key values here are the *median* as a measure for central tendency and inter-quantile range (IQR) for the spread of the data. Additionally, 1-norm distances are used due to their lower sensitivity to outliers, compared to 2-norms.

In classical least squares approximation of data, with known covariance, $\mathbf{\Lambda}$, the Mahalanobis distance [5.29] $E_2 = (\mathbf{y} - \hat{\mathbf{y}})^T \mathbf{\Lambda}^{-1} (\mathbf{y} - \hat{\mathbf{y}})$ is minimized, i.e. an inverse covariance weighted cost function. This will yield a maximum likelihood predictor in the presence of Gaussian perturbations. However, this is implicitly assuming the validity of a Gaussian PDF.

Here we introduce the concept of inverse IQR weighting. Given the diagonal matrix $\mathbf{\Delta}$, of IQR for each sensor, the weighted 1-norm distance is defined as:

$$E_1 = |\mathbf{\Delta}^{-1} (\mathbf{y} - \hat{\mathbf{y}})|_1. \quad (5.41)$$

The proposed approach is also consistent with the classical definitions for upper- U_b and lower-bounds L_b to detect outliers [5.30], i.e.

$$U_b = q_{25} - 1.5 \text{ IQR}, \quad L_b = q_{75} + 1.5 \text{ IQR}. \quad (5.42)$$

Whereby, q_{25} and q_{75} correspond to the 25% and 75% quantiles respectively.

4 Exemplary application

The extended RRAE is demonstrated on the same application and data sets as presented in [5.1]. This maintains the relevance to instrumentation and measurement, by using sensor data acquired in a real application; while enabling a direct determination of the improvements achieved in comparison to previous work.

The task is to measure the deformation of a tunnel wall during active tunneling in the vicinity. The sensors are mounted on the tunnel struts; providing a physical interpolation of the deformations. It is now necessary to implement mathematical interpolation to obtain the complete curve from the sensor values. The physical and mathematical interpolations must be consistent. This is achieved here by applying constrained polynomials to implement the interpolation; whereby, the constraints reflect the physics of the system being measured. The spatial measurement has been extended on both ends to locations where no motion is to be expected. Additional sensors are placed to ensure the fulfillment of this assumption. This has the additional advantage of converting the measurement problem to a boundary value problem, for which the Rayleigh-Ritz method provides an excellent solution approach.

4.1 Perturbed sensor data

A boxplot for the sensor data acquired during the training period is shown in Figure 4: the median, IQR together with the upper and lower outlier bounds are shown for each sensor channel. This maintains consistency with distribution-free statistics. The data is clearly heteroscedastic; furthermore, it is non-Gaussian (see Figure 2). There was no construction activity during this period; consequently, this data represents some fundamental uncertainty in the sensor data. This must be taken into account during training.

This nature of the sensor data questions the general, and most times implicit, assumption in machine learning that large data sets can be regarded as Gaussian [5.31–5.33]. Most work on machine learning and statistical learning [5.31] assumes independent and identically distributed (iid) data [5.34]. This is a further justification for the new approach taken here.

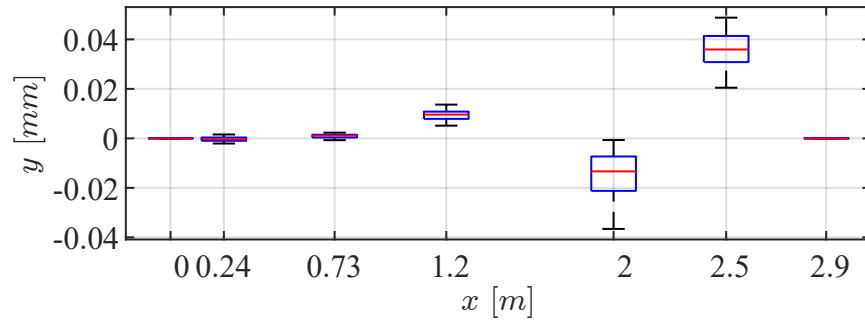


Figure 4: Boxplots of the training data, the input to the RRAE during training, at their specific locations. Note the horizontal scale is in *meters*, whereas the vertical is in *millimeters*. The boxplots characterize the central tendency and variability of the sensor signals in a distribution-free manner.

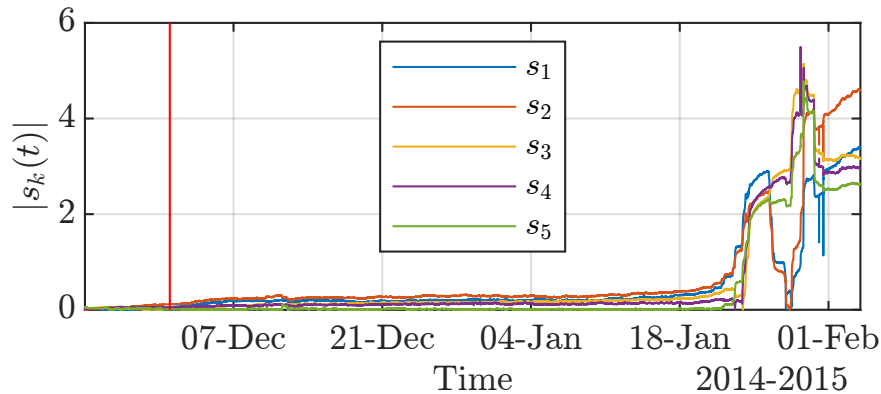


Figure 5: The tunnel is monitored along a number of struts, each with five sensors. Raw data from one of the struts, the tilt data is measured in $[mm/m]$. The data has been segmented into two portions separated by the red line: the training data, with $n = 779$ samples, and the normal operative region, $n = 9159$ samples. Measurements were performed every $t_s = 10$ mins over the period from 23-Nov-2014 to 4-Feb-2015.

The data was recorded using a total of seven tilt sensors. The first and last sensors measure points where there should be no motion. These can be considered as boundary conditions; the two end sensors monitor the fulfillment of these conditions. The measurement data is acquired from the remaining five sensors.

4.2 Application specific constraints

Returning to the example application: the model has four constraints, two value constraints, $y(-1) = 0$ and $y(1) = 0$, together with two derivative constraints, $y'(-1) = 0$ and $y'(1) = 0$. These correspond to the physical configuration of the system; additionally, the deformation of the struts is assumed to be continuous. This permits the synthesis of the admissible functions for the decoder portion of the RRAE.

4.3 Neural network hyperparameters

The hyperparameters of the encoder, including the dimension of the network are optimized using a genetic algorithm. This is a numerically time-consuming task since multiple instances of the same network need to be trained to identify hyperparameter values that yield both optimal performance and stability with respect to random initializations [5.32]. This is not discussed here in any further detail, due to the limited paper length. A description of the genetic algorithm used can be found in [5.35,5.36].

4.4 Network training and characterization

The sensor data acquired during the training and operation phases can be seen in Figure 5. Given the architecture and the admissible functions discussed previously, together with the optimized hyperparameters, the network was trained twice: First with a 1-norm distance metric for the loss function and a second time with the inverse IQR weighted 1-norm metric. This enables an objective evaluation of the performance difference. The results obtained with the inverse IQR weighting correspond more closely to the expectation associated with the physics of the system, this is discussed in detail below.

4.5 Network training statistics

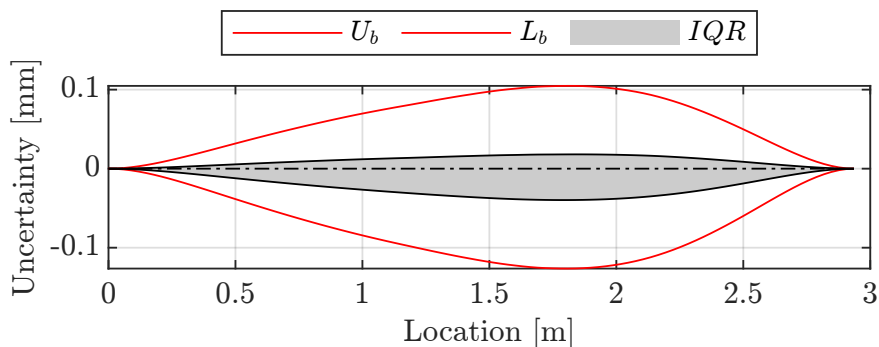


Figure 6: The upper- U_b and lower- L_b bounds for the detection of outliers and the IQR, for the training data when processed with the RRAE, trained using the 1-norm metric for the cost function. The asymmetry in the figure emanates from the asymmetry in the sensor perturbations, see Figure 4. This is not what would be expected from the physics of the system.

Given a trained network, it would ideally be expected that applying the RRAE to the training data once again would deliver results that correspond exactly to the initial state of the structure. However, since the

sensor data is perturbed there will be some uncertainty. Distribution-free statistical methods are being used here, rather than standard uncertainty, this is to ensure that the non-Gaussian nature of the data is adequately addressed.

The IQR, together with upper- U_b and lower-bounds L_b , for the detection of outliers are shown in Figure 6 when using the 1-norm metric for the loss function during training. A significant asymmetry is visible; however, we would expect a perfectly left-right symmetry given the symmetric nature of the constraints. In contrast, the results obtained using the inverse IQR weighted 1-norm are shown in Figure 7 and are now symmetric. This is exactly what would be expected from the physics of the system being monitored. This implies that the new distribution-free statistics and weighted metrics constitute a good approach and yields results that are consistent with the objectives of physics-informed machine learning.

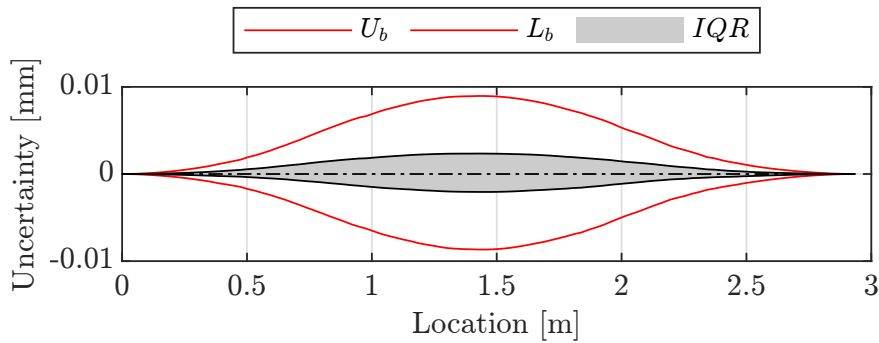


Figure 7: Similar visualization as in Figure 6; however, here the inverse IQR weighting has been applied to the 1-norm minimization. Note: this delivers left-right symmetric error bounds as the physics of the system would predict.

4.6 RRAE in the monitoring phase

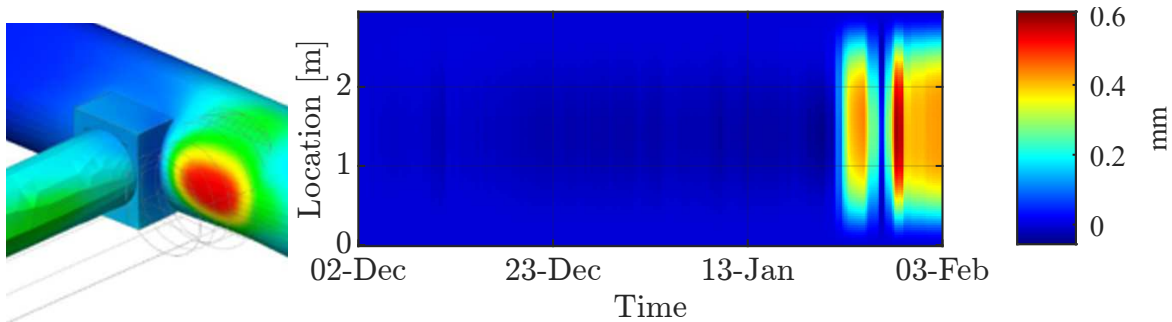


Figure 8: Left: Finite element simulation to predict the tunnel deformation. Right: Monitoring reconstruction of the tunnel deformation from the sensor data shown in Figure 5. The RRAE has been trained using the inverse IQR weighted 1-norm as a metric. Note the symmetric prediction and corresponding reconstruction.

The results of applying the RRAE to the monitoring data are shown in Figure 8, together with a finite element simulation (FEM) to predict the deformation of the tunnel as construction proceeds. The symmetries predicted by the FEM are now also present in the monitoring reconstruction. Once again suggesting that the new approach is delivering results that are much closer to the physical systems' expected behavior. It should be noted that deflections with a magnitude in millimeters are being reliably detected over ranges of several meters, in a very harsh environment.

5 Constraints and Lebesgue Functions

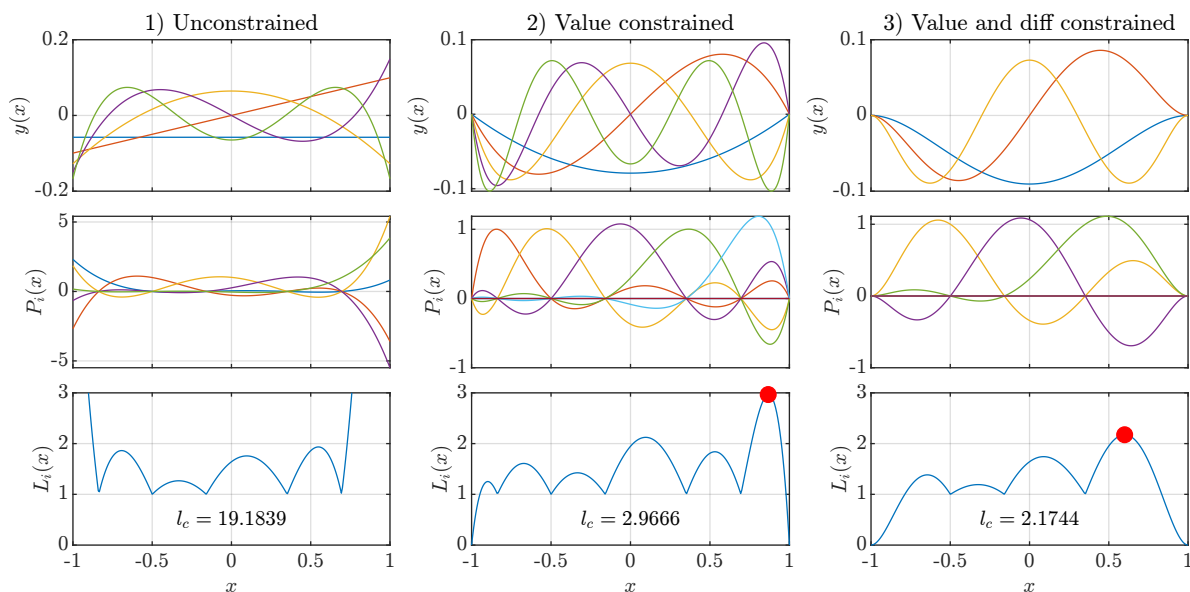


Figure 9: This figure shows the stability improvement when using constrained polynomials. The values of x have been normalized such that $(-1 \leq x \leq 1)$, while maintaining the relative sensor positions. From left to right the bases with increasing constraints: 1. the set of unconstrained polynomials; 2. the value-constrained polynomials and 3. the constrained polynomials with both value and differential constraints. From top to bottom: the bases $y(x)$, their corresponding interpolating functions $P_i(x)$ and the respective Lebesgue function $L_i(P)$ and Lebesgue constant l_c . The values of the respective Lebesgue constants indicate that the addition of constraints reduces the error bound due to the Runge phenomena when interpolating by a factor of $k \approx 8.84$.

It is now possible to put theoretical upper bounds on the interpolation error of the RRAE and to quantify this via the Lebesgue constant. As mentioned in Section 4, it is necessary to implement a mathematical interpolation to obtain the complete curve from the sensor values. Polynomial interpolation is subject to the Runge phenomena [5.37] at the ends of the interval, due to the large derivatives of the polynomials at these locations. The potential interpolation error can be quantified via the Lebesgue function $l_c(x)$ and -constant l_c for the interpolating functions [5.13]. Originally this was defined for Lagrange polynomials; however, its use can be extended to constrained polynomials.

The physical structure is characterized by four constraints: $y(0) = 0$, $y'(0) = 0$, $y(1) = 0$ and $y'(1) = 0$. The use of constrained polynomials stabilizes the derivatives at the ends of the interpolation. This in turn minimizes the Runge phenomenon.

The stabilizing effect of the constraints can be seen in Figure 9. Three different cases are shown to demonstrate the addition of value and derivative constraints; whereby, the sensor locations are used as the computational nodes for the polynomials. The unconstrained polynomial interpolation has a Lebesgue constant of $l_c \approx 19.18$, adding the value constraints reduced this value to $l_c \approx 2.97$, and the derivative constraint further reduced the value to $l_c \approx 2.17$. Consequently, the constraints reduce the interpolation error by a factor of $k \approx 8.84$.

6 Conclusions

An extended Rayleigh-Ritz autoencoder has been presented. Distribution-free statistical methods are used to ensure that the results are reliable and more robust for both Gaussian and non-Gaussian perturbed data. The extended approach has been tested on previously published sensor data. This has enabled a more objective evaluation of the achieved improvements. The results indicate that the use of an IQR weighted 1-norm metric during training leads to results that are more consistent with the predicted behavior of the physical system; this was verified by comparison with FEM simulation of the tunnel wall. Finally, it has been possible to compute theoretical upper bounds for the interpolation error of the RRAE based on Lebesgue functions. The implementation of the constraints via admissible functions has improved this error by a factor $k \approx 8.84$.

The improved statistical behavior, together with the exact fulfillment of generalized linear constraints makes the RRAE and interesting architecture for physics-informed machine learning.

Acknowledgment

The authors gratefully acknowledge the financial support under the scope of the COMET program within the K2 Center “Integrated Computational Material, Process and Product Engineering (IC-MPPE)” (Project No 886385). This program is supported by the Austrian Federal Ministries for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK) and for Labour and Economy (BMAW), represented by the Austrian Research Promotion Agency (FFG), and the federal states of Styria, Upper Austria and Tyrol.

Bibliography

- [5.1] A. Terbuch, P. O’Leary, D. Ninevski, E. J. Hagendorfer, E. Schlager, A. Windisch, and C. Schweimer, “A Rayleigh-Ritz Autoencoder,” in *2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2023, pp. 1–6.
- [5.2] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations,” 2017. [Online]. Available: <https://doi.org/10.48550/arXiv.1711.10561>

- [5.3] ———, “Physics informed deep learning (part ii): Data-driven discovery of nonlinear partial differential equations,” 2017. [Online]. Available: <https://doi.org/10.48550/arXiv.1711.10566>
- [5.4] M. Raissi, P. Perdikaris, and K. George, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.
- [5.5] J. S. Matitz, *Distribution-Free Statistical Methods*, 2nd ed. London: Chapman and Hall/CRC, 1995.
- [5.6] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML, “Guide to the expression of uncertainty in measurement — Part 6: Developing and using measurement models,” Joint Committee for Guides in Metrology, JCGM GUM-6:2020.
- [5.7] ———, “International vocabulary of metrology — Basic and general concepts and associated terms (VIM),” Joint Committee for Guides in Metrology, JCGM 200:2012. (3rd edition).
- [5.8] G. B. Rossi, F. Crenna, and A. Palazzo, “A Proposal for a More User-Oriented GUM,” *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 5, pp. 1343–1352, Mar. 2019.
- [5.9] M. Blanca, J. Arnau, D. López-Montiel, R. Bono, and R. Bendayan, “Skewness and kurtosis in real data samples,” *Methodology*, vol. 9, no. 2, 2013.
- [5.10] Q. Zhang and Y. Zhou, “Recent advances in non-gaussian stochastic systems control theory and its applications,” *International Journal of Network Dynamics and Intelligence*, vol. 1, no. 1, p. 111–119, 2022.
- [5.11] P. O’Leary, D. Ninevski, and A. Terbuch, “Automatic synthesis of admissible functions for variational learning,” in *2023 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, 2023, pp. 1–6.
- [5.12] S. Ilanko, L. E. Monterrubio, and Y. Mochida, *The Rayleigh–Ritz Method for Structural Analysis*. London: John Wiley and Sons, Ltd, 2014.
- [5.13] L. Brutman, “On the lebesgue function for polynomial interpolation,” *SIAM Journal on Numerical Analysis*, vol. 15, no. 4, pp. 694–704, 1978.
- [5.14] D. E. Rumelhart and J. L. McClelland, “Learning internal representations by error propagation,” in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*. Cambridge, MA, USA: MIT Press, 1987, pp. 318–362.
- [5.15] G. Zhang, B. Eddy Patuwo, and M. Y. Hu, “Forecasting with artificial neural networks:: The state of the art,” *International Journal of Forecasting*, vol. 14, no. 1, pp. 35–62, Mar. 1998.
- [5.16] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*. Berlin: Springer-Verlag, 2012.
- [5.17] I. M. Gelfand and S. V. Fomin, *Calculus of Variations*, ser. Dover Books on Mathematics. Dover Publications, 2000.

- [5.18] Y. Xiong, R. Zuo, Z. Luo, and X. Wang, “A physically constrained variational autoencoder for geochemical pattern recognition,” *Math. Geosci.*, vol. 54, no. 4, pp. 783–806, May 2022.
- [5.19] X. Meng, Z. Li, D. Zhang, and G. E. Karniadakis, “PPINN: Parareal physics-informed neural network for time-dependent pdes,” *Comput. Methods Appl. Mech. Eng.*, vol. 370, p. 113250, Oct. 2020.
- [5.20] H. Wang, B. Zou, J. Su, D. Wang, and X. Xu, “Variational methods and deep ritz method for active elastic solids,” *Soft Matter*, vol. 18, no. 32, pp. 6015–6031, Aug. 2022.
- [5.21] C. Anitescu, E. Atroshchenko, N. Alajlan, and T. Rabczuk, “Artificial neural network methods for the solution of second order boundary value problems,” *Computers, Materials and Continua*, vol. 59, no. 1, pp. 345–359, 2019.
- [5.22] G. Arfken, *Mathematical Methods for Physicists*, 3rd ed. San Diego: Academic Press, Inc., 1985.
- [5.23] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.
- [5.24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge and London: MIT Press, 2016.
- [5.25] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Networks*, vol. 2, no. 5, pp. 359–366, Jan. 1989.
- [5.26] N. Saad, G. Gupta, S. Alizadeh, and D. C. Maddix, “Guiding continuous operator learning through physics-based boundary constraints,” in *The Eleventh International Conference on Learning Representations*, 2023.
- [5.27] D. Hansen, D. C. Maddix, S. Alizadeh, G. Gupta, and M. W. Mahoney, “Learning physical models that can respect conservation laws,” in *ICLR 2023 Workshop on Physics for Machine Learning*, 2023.
- [5.28] P. O’Leary and M. Harker, “A framework for the evaluation of inclinometer data in the measurement of structures,” *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 5, pp. 1237–1251, 2012.
- [5.29] P. C. Mahalanobis, “On the generalised distance in statistics,” *Proceedings of the National Institute of Sciences of India*, vol. 2, no. 1, pp. 49–55, 1936.
- [5.30] D. C. Hoaglin, F. Mosteller, and J. W. Tukey, *Understanding Robust and Exploratory Data Analysis*. Hoboken, NJ, USA: Wiley, Jun. 2000.
- [5.31] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics)*. New York, NY, USA: Springer, Jan. 2016.
- [5.32] S. Raschka, “Model evaluation, model selection, and algorithm selection in machine learning,” 2018. [Online]. Available: <https://doi.org/10.48550/arXiv.1811.12808>

- [5.33] I. Steinwart, D. Hush, and C. Scovel, “A classification framework for anomaly detection,” *Journal of Machine Learning Research*, vol. 6, no. 8, pp. 211–232, 2005.
- [5.34] A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun, “The loss surfaces of multi-layer networks,” in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, G. Lebanon and S. V. N. Vishwanathan, Eds., vol. 38. San Diego, California, USA: PMLR, 09–12 May 2015, pp. 192–204.
- [5.35] A. Terbuch, P. O’Leary, N. Khalili-Motlagh-Kasmaei, P. Auer, A. Zöhrer, and V. Winter, “Detecting anomalous multivariate time-series via hybrid machine learning,” *IEEE Transactions on Instrumentation and Measurement*, 2023.
- [5.36] A. Terbuch, “LSTM hyperparameter optimization: Impact of the selection of hyperparameters on machine learning performance when applied to time series in physical systems,” Master’s thesis, Chair of Automation, Montanuniversitaet Leoben, 2021.
- [5.37] C. Runge, “Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinate,” *Zeitschrift für Mathematik und Physik*, vol. 46, pp. 224–243, 1856.