NTNU
Norwegian University of
Science and Technology

Failure analysis and prediction in compound system by wavelets

Tesfaye Amare Zerihun

Master of Telematics - Communication Networks and Networked Services
Submission date: August 2016
Supervisor: Bjarne Emil Helvik, ITEM

Norwegian University of Science and Technology
Department of Telematics

NTNU – Trondheim
Norwegian University of
Science and Technology

Failure analysis and prediction in compound system by wavelets

Tesfaye Amare Zerihun

Submission date: August 2016
Responsible professor: Bjarne Emil Helvik, ITEM
Supervisor:

Norwegian University of Science and Technology
Department of Telematics

Abstract

The current overall ICT infrastructure mainly the Internet and Telecom networks can be looked upon as an ecosystem, which is the result of the cooperation between a huge number of Autonomous systems (ASes). The interconnection and interdependence between ASes become large and complex as technology advances. This interdependence of ASes or subsystems create vulnerabilities in such a way that problems in one of the interconnected networks affect the normal operation of other networks and even might result in a failure of services across the whole system. The aim of this study is twofold. The first is to discuss about the basic features and trends in the logs of failure data to get some insight about the network's behaviour. In addition to this, the study looks into failure prediction by using the primary failure data to model normal behaviours and predict the system level(critical) failures. Failure log data will be used to model the normal(expected) behaviours of the failures and hence for prediction when there happens a change in the normal behaviour.

The report first discusses the conceptual model mainly about some related works as well as a background knowledge on wavelet technique. Then, a simple failure data analysis and brief discussion on the main trend observed during the preliminary study is presented. Lastly, a simple approach for failure prediction using wavelet technique is presented followed by evaluation and discussion of results. The report focuses in using a frequency domain approach which is called wavelet technique. A wavelet based failure prediction approach is proposed which uses some frequencies in the failure log data to characterize the normal operation and hence identify deviations(abnormal behaviours) from the variation in those frequencies when something bad occurs in the network. Once the deviations are identified, a root cause analysis can be conducted for a detail investigation of the problem areas.

Contents

Appendices

List of Figures

List of Tables

Chapter 1
Problem description

Today's ICT systems are extensively complex and consist of a large number of subsystems which co-operate in order to provide the intended services. These systems undergo a continuous evolution/change, with respect to structure, functionality, organisation and management. Furthermore, these subsystems may be operated by separate organisations, which also may belong to different market actors. In the latter case, there will be a limitation the information flow and the co-ordination of actions. The dependability requirements for such systems are extremely high. Due to the size and complexity of the systems, there are frequent element failures, mis-operations, glitches, etc. For these reasons, the existence of hidden channels for error propagation in the above outlined context, it not possible to understand the system level consequences of primary faults on system level failures, and traditional methodologies like FMEA (failure mode effect analysis) is totally inadequate. To manage the systems efficiently and to prevent disastrous system level failures, it is desirable to get insight in the evolution of primary faults into system level failures without insight into the detailed design and operation of the system. A promising approach, based on wavelets, towards a similar objective was identified in [1]. The objective of this master thesis is to investigate and extend this approach based on the extensive logs of failure data provided by Telenor. The work will consist of pre-processing and filtering of the raw data, statistical modelling and Wavelet transform analysis, development of algorithms, models, hypothesis, etc. in order to gain the outcome outlined below. The expected outcomes of the work are improved insight into:

- The failure processes within large compound systems.
- The evolution of primary failures (low priority alarms) into system level failures, and
- The strengths and shortcomings by using wavelets for describing failure patterns and relationships in large networks.

It is of specific interest to find if system failures can be predicted, i.e. to which degree, and how system level failures, that may have consequences for the service provided, may be predicted by less significant primary (low level) failure indications.

Chapter 2 Introduction

The current overall ICT infrastructure such as the Internet and Telecom networks can be looked upon as an ecosystem, which are the result of the cooperation between a huge number of Autonomous systems(ASes). [Zer15] Through time, as the technology advances, these systems are undergoing a continuous evolution and change, with respect to structure, functionality, organisation and management.

An Autonomous system can be large ICT enterprise such as, the worldwide on-line shopping company Amazon, which usually has several worldwide data centers. Each data center has tens of thousands of servers, switches, routers, firewalls, as well as other affiliated systems like power supply systems or cooling systems. The ICT network infrastructure for carriers is even more complex. For example, besides data centers, there are nationwide communication networks in a 3G/4G network infrastructure. Each communication network includes access network equipment, core network equipment, transport network equipment, and other application systems, containing tens of thousands of network elements that provide authentication, billing, data/voice communications, and multimedia services. These large scale complex networks introduce many difficulties in designing, architecting, operating, and maintaining the corresponding network infrastructures, on which multiple complex systems are coordinated to ensure that the computation and communication functions work smoothly.[Jun16]

Furthermore, these subsystems may be operated by separate organisations, which also may belong to different market actors. In the latter case, there will be a limitation the information flow and the co-ordination of actions. Such situations made the ICT ecosystem to become complex through time.

The interconnection and interdependence of ASes or subsystems create vulnerabilities in such a way that problems in one of the interconnected networks affect the normal operation of other networks and even might result in a failure of services across the whole system. [Zer15] This makes the dependability requirements of such complex systems to be extremely high.

A service failure, or simply a failure, is an event that occurs when the delivered service deviates from correct service. It is a transition from correct service to incorrect service, i.e., to not implementing the system function. Since a service is a sequence of the system's external states, a service failure means that at least one (or more) external state of the system deviates from the correct service state. The deviation is called an error. The adjudged or hypothesized cause of an error is called a fault. A fault is active when it causes an error, otherwise it is dormant.[ALRL04]

Figure 2.1 shows how fault evolve to an error and how errors propagate resulting in a failure of service. An error might successively transformed into other errors. Error propagation from component A to component B that receives service from A (i.e., external propagation) occurs when, through internal propagation, an error reaches the service interface of component A. At this time, service delivered by A to B becomes incorrect, and the ensuing service failure of A appears as an external fault to B and propagates the error into B via its user interface. [ALRL04] Errors in sub systems or some components can propagate through the system and they might result in the failure of the service delivered by the system.

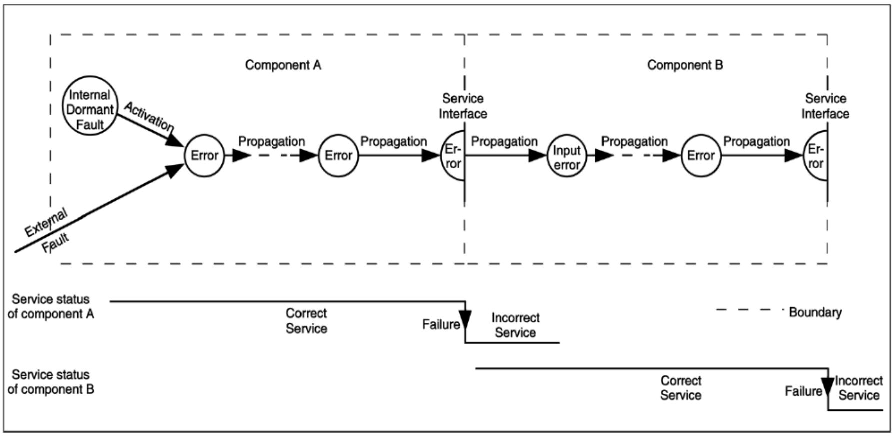


Figure 2.1: Error propagation [ALRL04]

Due to the size and complexity of large ICT systems, there are frequent element failures, mis-operations, glitches, etc. Having frequent faults and the existence of hidden channels for error propagation in the above outlined context, it not possible to understand the system level consequences of primary failures(i.e. failures in subsystems and/or components of the system) on system level failures, and traditional methodologies like FMEA (failure mode effect analysis) is totally inadequate.

There are many research papers such as [MYC08] [DTHS09] on failure prediction of large scale networks which often rely on measuring the traffic such as using BGP message data, IP traffic and so on. A summary of some papers is included in section 3. Most of these papers are based on assessing the log files which contain data representing both the normal and failed behaviours. There hasn't been much work on how to predict the failures by basing only on the failure log. Therefore, an alternative technique is to examine the failure log, mainly the primary failures, and to predict system level failures.

The aim of this study is twofold. The first is to discuss about the basic features and trends in the logs of failure data to get some insight about the network's behaviour. The other main objective is about failure prediction by using the primary failure data to model normal behaviours and predict the system level failures by looking for any changes in the normal(expected) behaviours of the failure data from different perspectives.

In addition, unlike the common approaches, this study tries to use wavelet techniques to analyse and predict failures. Applying Wavelet techniques, frequency domain transformation, on failure logs has a lot of advantages over other popular approaches such as scalability. Wavelet doesn't need to process much information as compared to data mining techniques or it doesn't need to have prior knowledge compared to Bayesian approaches.

Figure 2.2 shows the overall outline of the report. The report first discusses the conceptual model mainly about some related works(previous works on failure log analysis and failure prediction) as well as a background on wavelet technique on chapter 3 and 4 respectively. A brief description of the failure log data and how the the data is filtered is discussed in chapter 5. Chapter 5 also presents representative time series models for the logs of failure data and a simple analysis and brief discussion on the trends and behaviours observed during the preliminary study.

Chapter 6 first discusses the expected functionalities from a better failure prediction algorithm and then it introduces a wavelet based approach that use primary faults to predict system level (critical) failures. Following the presentation of a better approach, many experiments are conducted and chapter 7 presents evaluation and discussion of results from some selected scenarios of the proposed approach. Lastly, summarize the main findings and concludes about the strength and drawbacks of the proposed wavelet based approach.

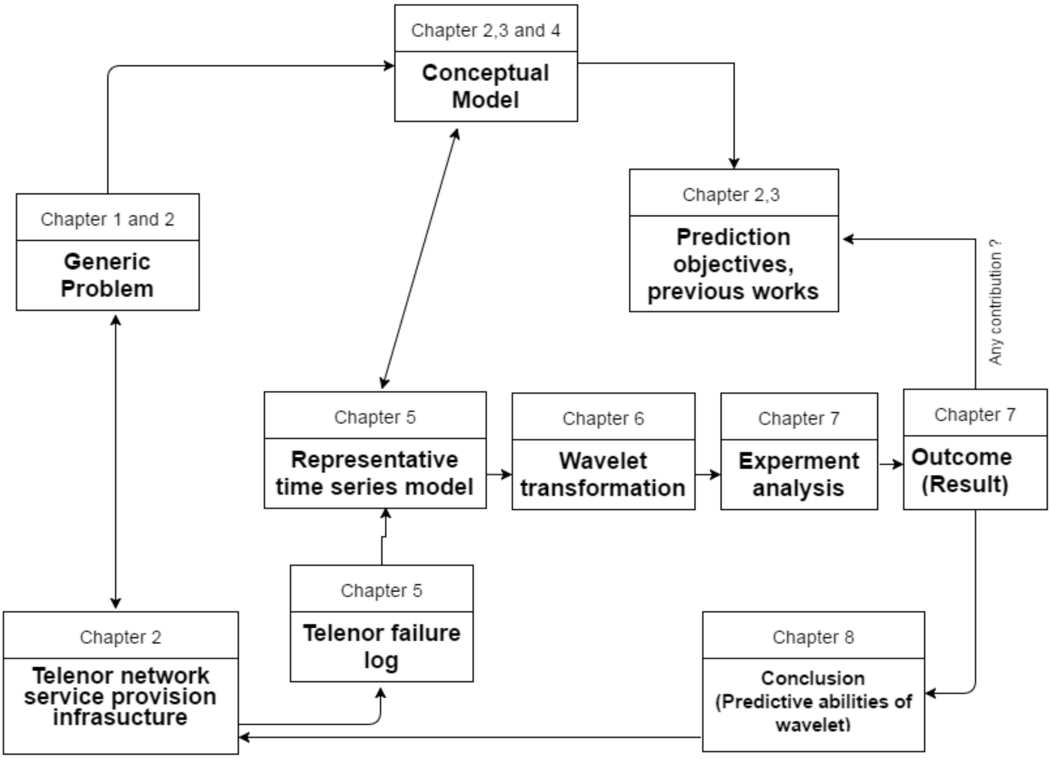


Figure 2.2: Outine of the study approach

Chapter 3 Related work

3.1 Introduction

There are a lot of studies for network failure analysis and prediction such as [MYC08] [DTHS09] that rely on measuring the traffic such as using BGP message data, IP traffic and so on. But, there hasn't been done much on using failure log as an input for the prediction. This chapter presents previous works on failure log analysis and failure prediction that has use similar data considering a similar environment(network). Failure analysis and prediction techniques discussed below are mainly those techniques which are used to automatically and effectively discover valuable knowledge from historical event/log data. Finally, a brief discussion of why a wavelet technique is used is presented.

3.2 Previous works

Failure analysis in compound systems has basically the following main procedures; Event generation (i.e., converting messages in log files into structured events), Root cause analysis to locate the faulty elements/components without relying much on experienced domain experts. Failure prediction for proactive fault management which improves network reliability. The summary below is mainly based on [Zer15] and [Jun16].

Nowadays, several industry organizations have already paid attention to these issues and put lots of efforts on making specifications related to best practices in operating and maintaining largescale complex systems/networks. In the IT service area, Information Technology Infrastructure Library (ITIL) such as [AXE15] and [TLS$^+$13] are a collection of specifications for service management, with which the best practices are organized according to the full life cycle of IT services including incident management, failure management, problem management, configuration management, and knowledge management. In the carrier service area, international organizations, such as ITUT [ITU15b] and TM Forum [ITU15a], also make recommended similar specifications. [Jun16]

However, the best practices in those specifications can not address the challenges in managing large scale complex networks/compound systems. This is because Large complex network infrastructures are often heterogeneous with respect to equipment type, software type and so on. And the different network elements generates huge amount of messages and alerts in different types and formats. The heterogeneity complicates the management work [HAB^{+}05], [BH08], and understanding these messages and alerts is not an easy task. In a small network, system administrators can analyze the messages and alerts one by one, and understand their corresponding event types. Apparently, it is not practical in large complex networks. Automatic event generation is important for reducing the maintenance cost with limited human resources. [Jun16]

In addition, malfunction of certain network elements can cause alerts in both upper/system level business applications and other connected network elements. The scale and complexity of root cause analysis [ZTL^{+}14a] in such networks are often beyond the ability of human operators. Therefore, automatic root cause analysis is necessary in managing large complex networks. [Jun16]

Event Generation (Extraction)

According to the survey paper [Jun16], recent research studies on event generation(extracting important information from log files) can be classified into three categories: log parser, classification, and clustering.

Log parser based approach: In Log parser based approach, system administrators with prior knowledge about type and format of raw messages, can develop text parsers to extract the detailed semantic information from these messages accurately. This takes fair amount of human effort but it gives good accuracy. [Jun16]

Classification based approach: The classification based approach does not require extracting all possible field variable values from log messages. Sometime it is enough to know event types of raw messages and focus on discovering the unknown relationship between different event types [Li15]. A simple classifier can be built using regular expression patterns. For each event type, there is a corresponding regular expression pattern [Sec15]. But similar to the issue in logparser based approaches, using regular expression for classification requires experienced domain experts to write the expression in advance, which is inefficient in large complex network infrastructures with heterogeneous network elements. [Jun16]

There are a lot of researches on classification algorithms that assume labeled log messages available for training such as paper [Sch08] using support vector machine (SVMs) algorithm, [ACP09] [KMRV03] focusing on security log classification. The classification based methods are accurate, but they need the labeled log messages for training. Obtaining the labeled data requires human efforts, which is often time consuming and costly. Classification based methods are inappropriate for large complex networks due to the lack of experienced domain experts for labeling. [Jun16]

Clustering based approach: Labeled training data is not required for clustering based methods, because such the methods infer event types from raw log messages.There are some studies [ABCM09], [MZHM09] on applying clustering techniques to partition log messages into separated groups, each of which represents an event type. To have a better performance, the studies on clustering based methods focus on the structured log messages. There are also other cluster based techniques discussed on paper [TL10] (based on building tree patterns for log messages), [TLP11](based on some signatures from the log messages), [MBZHM08] and k mean clustering algorithm on paper [SP13] [Jun16].

The advantage of clustering based methods is that they do not require lots of human efforts, but they are not as accurate as log parser based or classification based approaches. So clustering based approaches should be applied when the applications are error tolerant or the log files are noisy. [Jun16]

Root Cause Analysis

When a system error occurs at a lower level network element, it might propagate to upper level network elements and cause system errors at different levels. To find the root cause of the fault, it is not possible to check the network elements one by one to verify whether there is a hardware failure or a software exception. Therefore, automatic root cause analysis is needed.

Most root cause analysis methods are based on the dependency graph of network elements [BRM02], [KF05]. Dependency graphs could be built by experts if the network architecture is simple. For large complex networks, dependency graphs are built by finding the dependencies of network elements using event mining techniques. Root cause analysis can be done by locating the deepest element with alert messages on dependency graphs. Dependency might be bidirectional in practice, in which case it is needed to build a Bayesian network to calculate the probability of an element's status. Then the key step in root cause analysis is to discover the dependencies between events from log messages. Some of these approaches do not consider the time lag between events while others do. The research studies along this direction are divided into two categories: pattern based methods and temporal based methods. [Jun16]

Pattern Based Methods: Pattern Based approaches such as [KLA$^+$14] discusses how to find bugs in wireless sensor networks which are usually not caused by a particular component but the unexpected interactions between multiple working components. The tool performs root cause analysis by discovering event sequences that are responsible for the faulty behavior. All log messages are divided into two categories, good and bad. Then all frequent event sequences up to a predefined length are generated. The good and bad frequent event sequences are used to perform discriminative analysis and these discriminative sub-sequences are used for bug analysis by matching. There are also papers such as [LFWL10] proposing an approach to find the hidden dependencies between components from unstructured logs using Bayesian decision theory and paper [NKN12] presenting a tool to find the most possible system components which might cause the performance issue in modern largescale distributed systems using machine learning techniques. [Jun16]

Temporal Based Methods: Paper [ZTL$^+$14b] proposed to mine time lags of hidden temporal dependencies from sequential data for root cause analysis. Unlike traditional methods using a predefined time window, this method is used to find fluctuating, noisy, interleaved time lags. The randomness of time lags and the temporal dependencies between events are formalized as a parametric model. The parameters of the maximal likelihood model are inferred using an EM based approach. Another paper [TLS12] presented a non parametric method for finding the hidden temporal dependencies. By investigating the correlations between temporal dependency and other temporal patterns, both the pattern frequency and the time lag between events are considered in their proposed model. Two algorithms utilizing the sorted table in representing time lags are proposed to efficiently discover the appropriate lag intervals. [Jun16]

Failure Prediction

So far there were some methodologies/techniques for predicting failures in ICT systems by examining the behaviour(such as IP traffic or failure patterns) of the system studied. Some of the most popular techniques such as statistical, Bayesian, Machine learning based approaches etc. are discussed on the semester project report[Zer15]. Here below summarized are some recent works on applying the techniques using log files in a large scale networks.

Paper [SM07] presented an approach for online failure prediction in telecommunication systems using eventdriven data sources. Hidden Semi Markov Models (HSMMs) are used to model the failure event flow. The historical event sequence for failure and non failure are collected for building two HSMMs. The failure likelihood of current event sequence is calculated using the two HSMMs. [Jun16]

Paper [SFMW14] and [FSS$^+$13] use log files to predict failures. Paper [SFMW14] presented a data driven approach based on multiple instance learning for failure prediction using equipment events. The log files contain both the daily operation records and the service details. Predictive features include event keywords, event codes, variations, sequence of event codes, etc. which are generated by parsers. A sparse linear classifier is trained with selected stable features for failure prediction. In paper [FSS+13], event sequences are first extracted from log files. Supported Vector Machines (SVMs) are used to classify these event sequences into two categories: fail and non fail. The process of extracting the event sequences is done in an incremental way. Each word in log files is assigned to a unique high dimensional index vector. When the log message is scanned, a context vector is calculated by summarizing index vectors in the sliding window. [Jun16]

paper [LZXS07] applied several classification methods on event logs collected from supercomputer IBM BlueGene/L and tried to predict the fatal event in the near future based on events in current window and historical observation period. There are six different types of events in the log files and for each event type. The following features are extracted from log files for training the classifiers: event number, accumulated event number, event distribution, interval between failures, and entry keywords in log messages. [Jun16]

Paper [SOR+03] described a framework of a proactive prediction and control system for large clusters. Event logs and system activity reports are collected from a 350 node cluster for one year. A filtering technique is applied to remove the redundant and misaligned event data. They evaluated three different failure prediction approaches: linear time series models, rule based classification algorithms, and Bayesian network models.[Jun16]

Paper [FX07] developed a spherical covariance model and a stochastic model to qualify the temporal correlation and the temporal correlation between events, respectively. The failure events are clustered into groups based on the correlations. Each group is represented by a failure signature which contains various attributes. Failure prediction is done by predicting the future occurrences of each group. [Jun16]

Paper [ZLO01] developed an approach for predicting failure and in categorical event sequences. Sequential data mining techniques are applied on the historical plan failure information for generating predictive rules. Normative, redundant, and dominated patterns are removed in order to select the most predictive rules for failure prediction. [Jun16]

Most of the papers are based on assessing the log files which contain data representing both the normal and failed behaviours. There hasn't been much work on how to predict the failures by basing only on the failure log. Therefore, an alternative technique is to examine the failure log, mainly the primary faults, and to predict system level failures. This report discusses about using the primary failure data to model normal behaviours and predict the system level failures by looking for any changes in the normal behaviours of the failure data from different perspectives.

3.3 Why wavelet techniques?

There has not been much work for predicting failures soley depending on the failure log. Section 3.2 tries to discuss some of the approached used so far. Unlike those common approaches, this paper tries to use wavelet techniques to analyse and predict failures looks promising as it fulfills most of the criteria mentioned in Chapter 6, section 6.2. The basic advantages of using wavelet techniques for predicting abnormal deviations is also discussed on the semester project [Zer15] and most of the arguments to use the technique for failure predictions are similar.

Failure prediction approaches that use failure logs such as paper [ZLO01] use sequential data mining which needs to analyse large amount of data. Using such techniques has some complexity and the implementation is also not easy. Likewise, Bayesian based approaches such as [SOR$^+$03] and papers that use classification based on signatures such as [FX07] needs some prior knowledge. For wavelet based approaches, for instance in paper [MYC08], it is mentioned that wavelet technique can be used without to rely on detailed information, and serves as a complementary tool to reduce the candidate data set for further detailed root cause analysis. Having such property of detection analysis on a reduced dataset makes it more scalable to use. Furthermore, the wavelet-based algorithm for the temporal localization of anomalies requires only minimum processing, [Zer15] and it doesn't also need much assumption and prior knowledge.

Wavelet techniques help to locate anomalies both in time domain and space. Though wavelet technique has not been used with failure logs, there are some papers that use the technique to predict abnormal deviations in the network based on BGP data exchange. For instance, on paper [MYC08], the wavelet algorithm (MODWT) detect anomalies temporally while the two-dimensional clustering procedure opens up further possibilities in locating anomalies spatially. The BAlet wavelet techinique used in this paper complements existing approaches by locating a smaller set of BGP data (through temporal and spatial localization) that can later be processed by other signature-based sophisticated root cause analysis algorithms. [MYC08]

The usage of thresholding mechanism in wavelet techniques is rare. There are some variants of wavelet technique algorithms which require neither auto regression nor thresholds to detect changes such as abrupt change detection using hypothesis testing. Wavelet techniques are shown to be able to detect and locate subtle changes in variance from time series, and performs better than adaptive thresholding techniques and auto-regressive models [MYC08] [Zer15].

In addition, frequency domain analysis is not extensively investigated in detecting failure patterns and behaviors when compared to failure analysis and prediction techniques in time domain representation. [Zer15] Hence, if investigated much more in detail, frequency domain analysis could be a good alternative to investigate failure logs and tackle current problems from a different viewpoint.

Having all the above functionalities, it is worth to use wavelet techniques for analysing and predicting failure patterns using a failure log collected from compound systems or large networks. Wavelet approaches can use frequency of different attributes in the failure log such as priority levels, consequences aspects of the failure and so on to monitor and characterize the normal operation as well as severe conditions.

When something goes wrong in the network, there will be a change in those frequencies and wavelet techniques can extract and expose the various frequencies with the respective time (when those frequencies has occurred) nicely. The anomalies or sudden changes are characterized by high frequencies for a relatively short period of time. This in turn means we can identify deviations from the normal operation when something bad occurs in the network [Zer15].

Chapter 4 Background

4.1 Wavelet transform

Mostly raw data signals are represented as a function of time (i.e in time domain). But, sometimes the information needed might be in the frequency domain. So, frequency transforms are useful to get more insight. It could also be easy for detailed analysis of complex equations which could be difficult if we use time domain representation. Wavelet transform is a frequency transform technique which is capable of providing the time and frequency information simultaneously, hence giving a time-frequency representation of the signal. [Zer15]

There are different popular frequency transform techniques which are often used such as Fourier transform (FT) and Short term Fourier transform (STFT). FT gives the frequency information of the signal, which means that it tells us how much of each frequency exists in the signal, but it does not tell us when in time these frequency components exist. Therefore, Fourier transform is not suitable if the signal has time varying frequency, i.e., the signal is non-stationary. [Pol96]

When the time localization of the spectral components is needed, a transform giving the time-frequency representation of the signal is needed. Wavelet transform and Short term Fourier transform (STFT) can provide such time-frequency representation. However, STFT has problems related to resolution and Wavelet transform is able to overcome some resolution problems of the STFT as discussed below. [Zer15]

The frequency and time information of a signal at some certain point in the time-frequency plane cannot be known. (I.e. it is difficult to know what spectral component exists at any given time instant). The best solution would be to investigate what spectral components exist at any given interval of time. This brings a problem of resolution, and it is the main reason why researchers have switched to WT from STFT [Pol96].

STFT uses a window of finite length and it gives a fixed resolution at all times. Whereas, WT gives a variable resolution as follows: Higher frequencies are better resolved in time, and lower frequencies are better resolved in frequency. In other words, a certain high frequency component can be located better in time (with less relative error) than a low frequency component. On the contrary, a low frequency component can be located better in frequency compared to high frequency component. [Pol96]

Continuous wavelet transform (DWT) The continuous wavelet transform is defined as shown in equation 4.1

$\mathrm{CWT}_x^{\psi}(\tau, s) = \psi_x^{\psi}(\tau, s) = \frac{1}{\sqrt{S}} \int X(t) \psi^*(\frac{t-\tau}{S})$ (3.1)

As seen in the equation 4.1, the transformed signal is a function of two variables, *tau* and s, the translation and scale parameters, respectively. $\psi(t)$ is the transforming function, and it is called the *mother wavelet* . The mother wavelet is a prototype for generating the other window functions. [Zer15]

$\tau(t)$ (translation) parameter is related to the location of the window, as the window is shifted through the signal. It corresponds to time information in the transform domain. S (scale) parameter is defined as $1/frequency$. The parameter scale in the wavelet analysis is similar to the scale used in maps. Low frequencies (high scales) correspond to a global information of a signal (that usually spans the entire signal), whereas high frequencies (low scales) correspond to a detailed information of a hidden pattern in the signal (that usually lasts a relatively short time). [Pol96]

The mother wavelet is chosen to serve as a prototype for all windows in the process. All the windows that are used are the dilated (or compressed) and shifted versions of the mother wavelet. There are a number of functions that are used for this purpose such as the Morlet wavelet and the Mexican hat functions. Once the mother wavelet is chosen the continuous wavelet transform is computed for different values of s. CWT is simply a correlation between a wavelet at different scales and the signal with the scale (or the frequency) being used as a measure of similarity [Pol96].

Discrete wavelet transform (DWT)

In order to have a practical computation of the analytical equations on computers, it is necessary to have a discretized transform and Discrete wavelet transform (DWT) is one with a significant reduction in the computation time.

In DWT, a time-scale representation of a digital signal is obtained using digital filtering techniques. The continuous wavelet transform was computed by changing the scale of the analysis window, shifting the window in time, multiplying by the signal, and integrating over all times. In the discrete case, filters of different cutoff frequencies are used to analyze the signal at different scales. The signal is passed through a series of high pass filters to analyze the high frequencies, and it is passed through a series of low pass filters to analyze the low frequencies [Pol96]. [Zer15]

The resolution of the signal, which is a measure of the amount of detail information in the signal, is changed by the filtering operations, and the scale is changed by upsampling and downsampling (subsampling) operations. Upsampling a signal corresponds to increasing the sampling rate of a signal by adding new samples to the signal while subsampling a signal corresponds to reducing the sampling rate, or removing some of the samples of the signal. [Pol96]

The DWT analyzes the signal at different frequency bands with different resolutions by decomposing the signal into a coarse approximation and detail information. DWT employs two sets of functions, called scaling functions and wavelet functions, which are associated with low pass and highpass filters, respectively. The decomposition of the signal into different frequency bands is simply obtained by successive high pass and low pass filtering of the time domain signal. [Pol96]

The original signal x[n] is first passed through a half band high pass filter g[n] and a low pass filter h[n]. After the filtering, half of the samples can be eliminated according to the Nyquist's rule, since the signal now has a highest frequency of p/2 radians instead of p. The signal can therefore be subsampled by 2, simply by discarding every other sample. This constitutes one level of decomposition and can mathematically be expressed as shown in 4.2 and 4.3 where $\mathrm{Y}_{\mathrm{high}}$[k] and $\mathrm{Y}_{\mathrm{low}}$[k] are the outputs of the high pass and low pass filters, respectively, after subsampling by 2. [Pol96]

$\mathrm{Y}_{high}[K] = \sum_n X[n].g[2k-n]$ (4.2)

$\mathrm{Y}_{low}[K] = \sum_n X[n].h[2k-n]$ (4.3)

This decomposition halves the time resolution since only half the number of samples now characterizes the entire signal. However, this operation doubles the frequency resolution, since the frequency band of the signal now spans only half the previous frequency band, effectively reducing the uncertainty in the frequency by half. The above procedure, which is also known as the subband coding, can be repeated for further decomposition. At every level, the filtering and subsampling will result in half the number of samples (and hence half the time resolution) and half the frequency band spanned (and hence doubles the frequency resolution). Figure 4.1 illustrates this procedure, where x[n] is the original signal to be decomposed while h[n] and g[n] are low pass and high pass filters, respectively. The bandwidth of the signal at every level is marked on the figure as "f" [Pol96]. [Zer15]

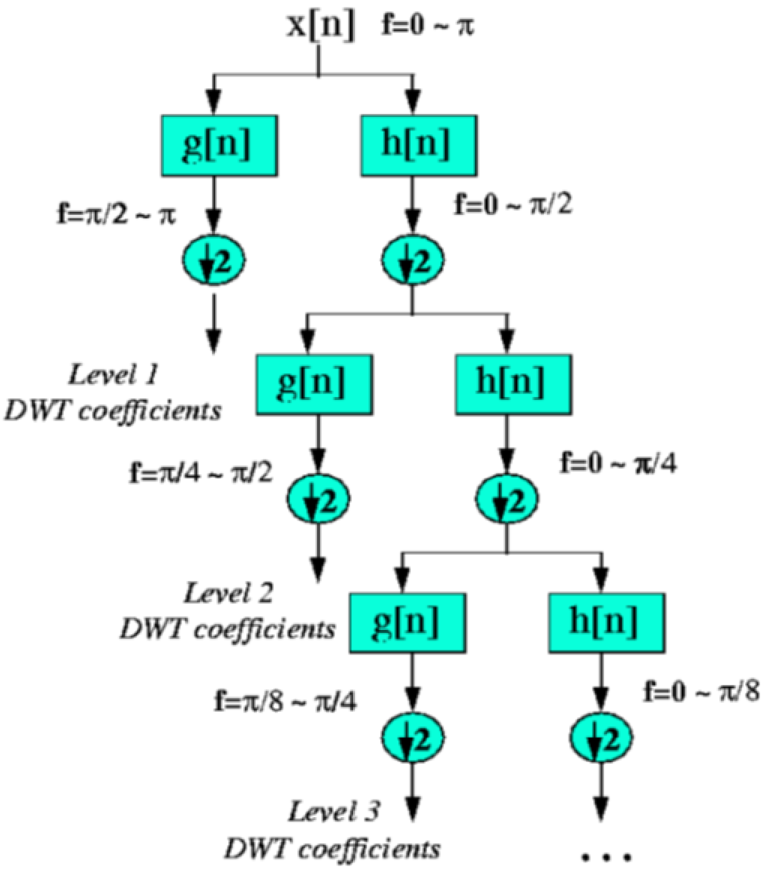


Figure 4.1: Discrete wavelet transform from [Pol96].

The DWT of the original signal is then obtained by concatenating all coefficients starting from the last level of decomposition. The DWT will then have the same number of coefficients as the original signal.

The frequencies that are most prominent in the original signal will appear as high amplitudes in that region of the DWT signal that includes those particular frequencies. The time localization will have a resolution that depends on which level they appear. If the main information of the signal lies in the high frequencies, as happens most often, the time localization of these frequencies will be more precise, since they are characterized by more number of samples. If the main information lies only at very low frequencies, the time localization will not be very precise, since few samples are used to express signal at these frequencies. This procedure in effect offers a good time resolution at high frequencies, and good frequency resolution at low frequencies. Most practical signals encountered are of this type [Pol96] [Zer15].

Chapter 5 Failure data analysis

5.1 Introduction

One of the main objective of this master thesis is to investigate the extensive logs of failure data provided by Telenor. This chapter briefly discusses about the basic features and trends in the logs of failure data to get some insight about the behaviour of the network to be studied. It first presents an explanation about how the pre-processing and filtering of the raw data is made. And then, the behaviour of the failure process in compound systems including a demonstration of how the failure distribution looks like is discussed.

5.2 Pre-processing and filtering of Failure data

The study uses logs of failure data provided by Telenor. The data is automatically collected failure records(alarms) with different level of severity from the Telenor network throughout Norway. The failure log consists two types of raw data collected in a period of 3 month duration. One is a dumped text file while the other is partially filtered data and presented in an Excel format(A sample of this format is attached in Appendix B). As most of the important aspects of the raw data were captured by this partially filtered raw data, the study is mainly based on using the second type of data, the excel files. Afterwards, the use of the term raw data is referring to the screened excel files.

As the log records contain messages in natural language, it is impossible to extract key knowledge without some semantic analysis. It has also some events recorded repetitively. To address these issues, some pre-processing measures have been taken and the following procedures are used for pre-processing the automatically generated records(log files).

- Filtering the records to decrease the data to be analyzed by removing some repetitive failure records. Some failures has the same root cause but they are reported by different network elements with the occurrence time being the same. So, only one record will be taken for such cases.

- Some failure records with empty value were represented and treated differently. An assumed value close to the neighbouring points were given to these records.

The log file is a collection of record of failure information from different aspects. Every failure records has information corresponding to the different aspects(dimensions) such as where the failure happened, what type of network elements failed, the consequence due to the failure and so on. So, each time we extract information to form a data set, it is necessary to take one of the those dimensions together with the time information.

After conducting this procedure a concise, comprehensive and well organized failure record sets were achieved, and these refined data sets also helps to improve the accuracy of failure prediction. A simple example in Appendix B is attached to show how the pre-processing and filtering is done.

Classification of failure data

The log file used in this study has somehow well structured record format with paring of records. Each failure is recorded with a lot of information about it such as where and when it occur, what consequence it results and so on. However it is needed to classify the so called organized record as it is difficult to use all those information directly. A sample of failure log raw data used for the study can be found in Appendix B.1. The failure record has the following basic dimensions/aspects:

- Priority level
- Failure registration date
- Problem type
- Problem area
- Consequence
- Outrage duration
- Municipality
- County

In this study, the failure data set is mainly divided into two sets; one representing the high priority failures or simply failures with a severe consequence while the other data set is low priority(primary) failures. There are seven priority levels, represented by P1 - P7. The first two priorities, P1 and P2, are considered high priority failures while the rest priority levels are considered to be a low priority failures. The study will be focusing on the analysing the second data set(low priority failures) to predict the high priority failures with severe consequence.

When analysing the low priority data set, there can be different ways of organizing the failure log based on the different aspects mentioned above as well as based on the level hierarchy to be considered. For instance, based on the failure aspects, we can have data sets considering one aspect at a time such as the priority alone or considering the consequence aspect alone. Whereas, with respect to the level hierarchy, one can consider to analyse data sets separately for each counties(regions) or on the system level considering the whole data set.

Constructing time series

In the log files, most of the records of the different aspects mentioned above are in a text form. This will make it difficult to use it as a time series in the wavelet analysis later. Therefore, when the information is extracted, it is tried to quantify each aspects into a numeric value. By representing all the information with a numeric value, it becomes easy to transform the data sets into a wavelet domain representation and study the frequencies of those information represented by numerical value.

The study uses two ways of quantification. One is for aspects/dimensions that can be ranked such as Priority level and consequence resulted, it is tried to assign a value based on their respective severity magnitude. Some fixed value is assumed for the initial (starting) rank and a constant increment is used between ranks as we do not have enough detail information about the difference in severity level among the ranks/levels.

Meanwhile, other aspects/dimensions such as those that represent spatial information and problem type can not be ranked(from the failure log data). These aspects are also mapped into some fixed numeric values so that it become easy to use wavelet technique. Hence, by using wavelet technique, it is hypothesized that we are able to study the frequencies of the numbers assigned (which also means studying the frequencies of the information represented by the numeric values) and possibly point out deviations.

The study uses different time series constructed both at the regional level and at the system level. Four out of the eight aspects/dimensions of the failure log data are used in this study. And, four time series corresponding to these aspects/dimensions of the failure data are created both at a regional level and at the system level.

For the priority aspect, it is assumed that both the high priority failures P1 and P2 are considered as a similar situation which result into a sever consequence. For simplicity, the study will not differentiate between the two, rather it tries to predict their occurrence as if they are of the same type. Hence, they are quantified into same numeric value.

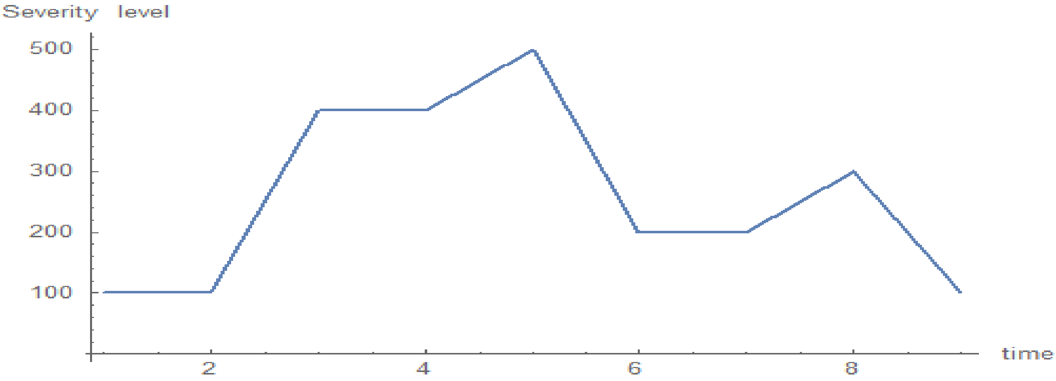


Figure 5.1: An example of creating time series from priority level

Whereas, for the low priority data set, a numeric value is given for each of the records based on their priority level. Failure record P3, having a higher priority than the rest, is given a highest value among the low priority failures while Failure record P7 being the least priority is given the smallest numeric value. This representation helps serious failure alarms such as P3 to contribute more while alarms with a small effect will contribute less when a threshold is used later in the prediction algorithms. It is hypothesized that this will improve the performance and accuracy of the prediction. Figure 5.1 shows an example of how the quantification is done in creating time series representing low priority failures. A low priority failure records of failure sequence {P3, P3, P6, P6, P7, P4, P4, P5, P3} can be represented as shown in figure 5.1. The detail quantification of priority level can be found in Appendix A.

Similarly, when constructing a time series for the consequence aspect, the quantification is based on the severity. Failure records that result in few seconds interruptions are given smaller numeric values while failures that result in an interruption of a longer time span are given a larger numeric value. For time series representing location of failures, a numeric value is assigned for each cities without consideration of closeness between locations.

Overall there are four dimensions that are used to create time series in this study. One failure log can be represented in different ways; what priority does it have?, where does it occur?, how severe consequence it resulted? and what type of subsystem failed?. Figure 5.2 shows how a typical high priority failure can be represented at least in a three dimensional context. The detail quantification related to all the aspects/dimensions used in the study can be found in Appendix A.

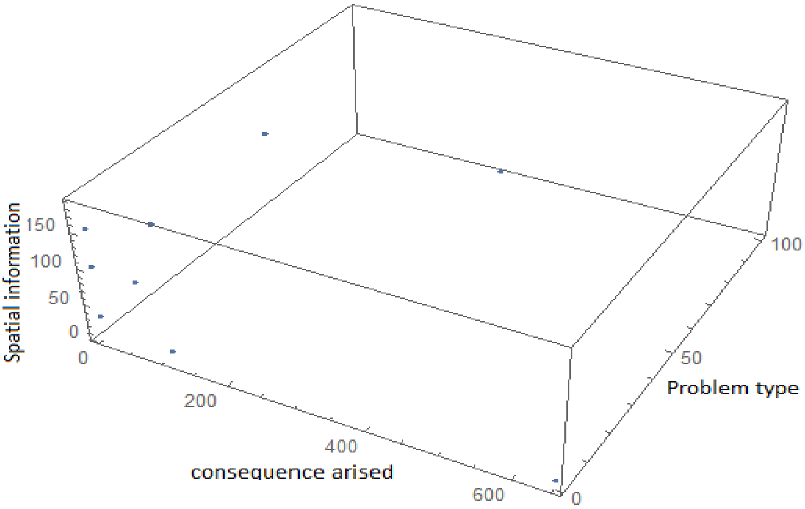


Figure 5.2: An example showing the other three dimensions of a typical high priority failure

Once the data sets are quantified, an event series is first created and then it is combined with the time information to create a time series. A simple example in B shows how this is done. All the time series discussed above are constructed on a time granularity of 1 hour.

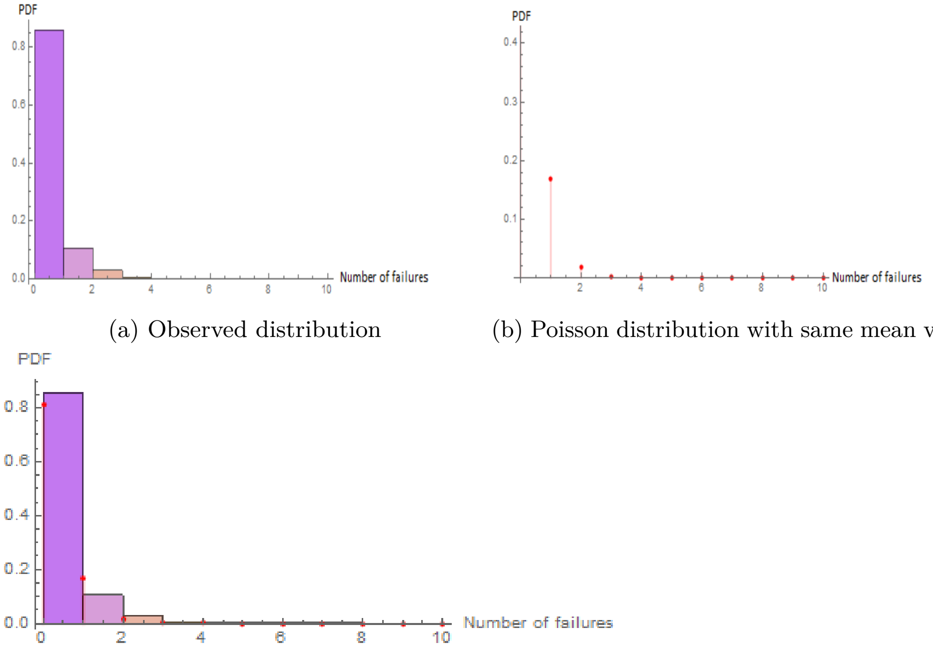
5.3 Failure process(behaviour) in compound systems

The failure process can be studied from two perspectives; One is to look at the behaviour of high priority failures (P1 and P2) which are of the focus point in the study and the other perspective will be to look at the behaviour of low priority failures which are going to be used to propose mechanisms to predict high priority failures so that a severe consequences can potentially be avoided.

5.3.1 Distribution of high priority failures

There are two types of high priority failure records, P1 and P2, with the first priority P1 occurring very rarely as compared to P2. Both priorities are considered as critical situations and this study looks at their behaviour/distribution collectively. It is sought to look at the failure distribution of these critical failures in a sub-system level(considering counties) and at the system level(considering the whole network).

Regional level: Looking at a regional level, the distribution of high priority failures based on failure records collected for a time span of 3 month is shown in Figure 5.3. The failure data used is the aggregated number of failures recorded every one hour. Figure 5.3 shows how the distribution of high priority failures in Sør-trondelag county looks like. It is also shown in Figure 5.4 that the distribution of high priority failures in other counties also follow a similar pattern.



(a) Observed distribution

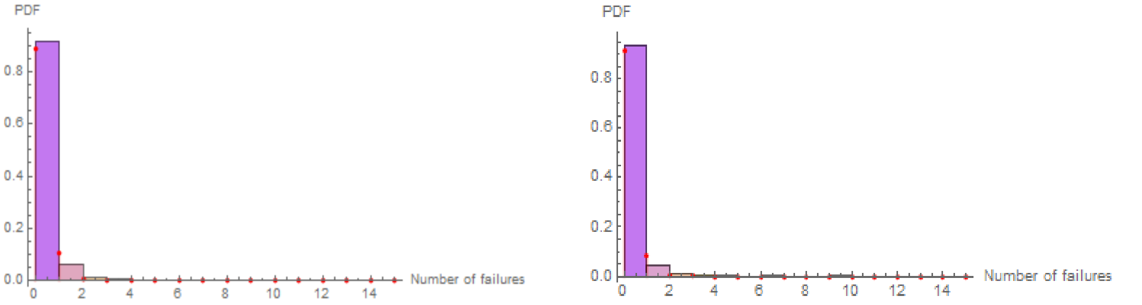
(b) Poisson distribution with same mean value

(c) Comparison between Poisson distribution with the same mean and observed distribution

Figure 5.3: High priority failure distribution in Sør-trondelag county

Here, the main interest is to study whether the distribution has a Poisson property or if it has a bursty behaviour. The knowledge of this behaviour is important for the analysis and assumptions to be used in the later chapters.

While Figure 5.4a shows the observed PDF of high priority failures in Sør-trondelag region, Figure 5.4b shows the Poisson distribution with a mean value (number of failures) calculated over the same period. It can be seen on Figure 5.3c that the observed distribution of high priority failures in counties is somehow similar to the



(a) Comparison between Poisson distribution with the same mean and observed distribution in Telemark county

(b) Comparison between Poisson distribution with t same mean and observed distribution in Nord-trondel county

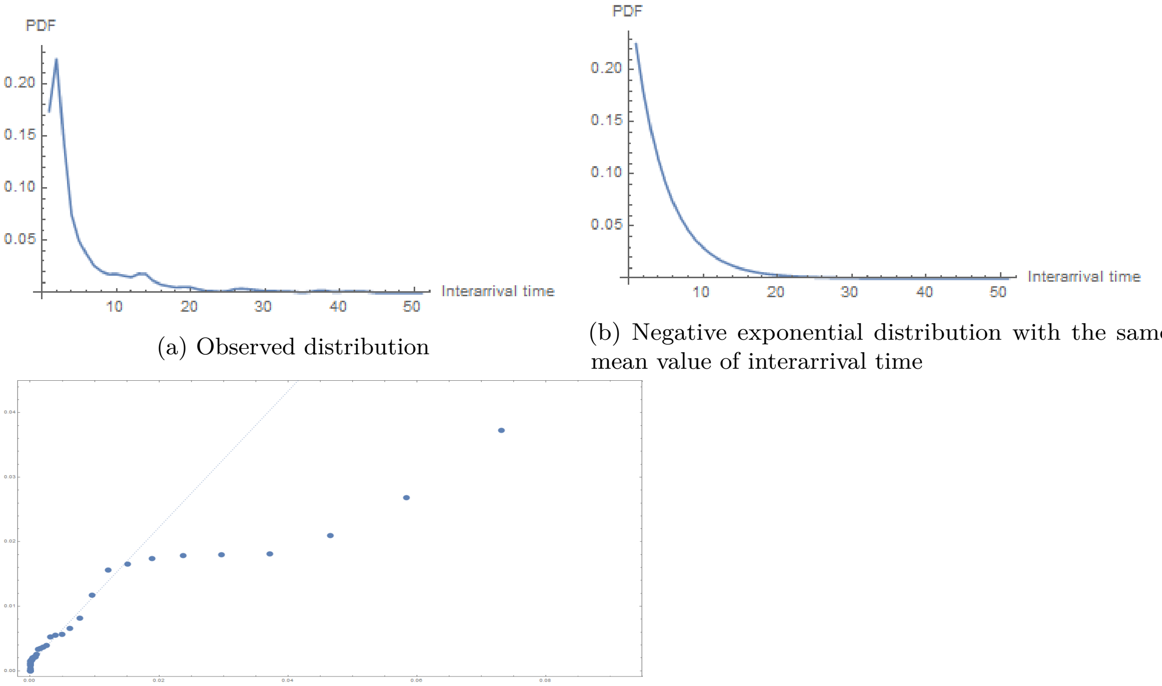
Figure 5.4: High priority failure distribution in different regions

Poisson distribution with the same mean. However, With such simple investigation, it is still difficult to conclude that the occurrence of critical failures at a regional level is Poissonly distributed.

An alternative way to study the distribution is to look at the time between failures. For a poissonly distributed distribution, the time between the failures is negative exponentially distributed. It is possible to calculate the average time between failures from the failure log data. Figure 5.5 shows the interarrival time between high priority failures in Sør-trondelag region. Simple tests for fitness of the empirical distribution with a standard distribution using Q-Q is also presented on the figure.

Figure 5.5a shows the observed distribution of time between high priority failures while figure 5.5b shows the negative exponential distribution considering the mean value of interarrival time calculated from the failure log data. The Q-Q plot in figure 5.5c shows that the observed distribution follows the negative exponential distribution with same mean for most of the data points, mainly on smaller range of values. However, for large values which has lesser probability of occurrence, the distributions are not similar. A kolmogorov-simonorv test on the observed and negative exponential distribution with same mean interarrival time results in rejection of the hypothesis(at the 5% level) that the observed and negative exponential distribution are the same.

The test on the interarrival time between failures shows that there is a good similarity between the observed distribution and negative exponential distribution with same mean which supports the above argument that the high priority failure distribution is from a poisson process. There are some cases where a test failed such as the kolmogorov simonorov test. Though it is not a sufficient condition to conclude,



(a) Observed distribution

(b) Negative exponential distribution with the sam mean value of interarrival time

(c) Q-Q plot for comparison between observed distribution and negative exponential distribution with same mean interarrival time

Figure 5.5: Interarrival time between high priority failures in Sør-trondelag region

this might gives a hint that the failure process might have some bursty behaviours. Figure 5.6 supports this argument. As we can see figure 5.6, there are days where we have quite many failures and there are also other days where we have a very small number of failures. On the figure, Saturday and Sunday have small number of failures while the rest weekdays have more failures than the weekend days. It can clearly be seen that the highest number of failures (15 failure per day in average) is recorded on few days, especially Monday and Tuesday.

System level:

Similarly, at the system level, the distribution of high priority failures(considering the whole network) based on failure records collected for a time span of 3 month is shown in figure 5.7. The failure data used is the aggregated number of failures recorded every one hour.

On the system level, there are many critical failures every hour. Figure 5.7a shows the observed PDF of high priority failures throughout the whole network and

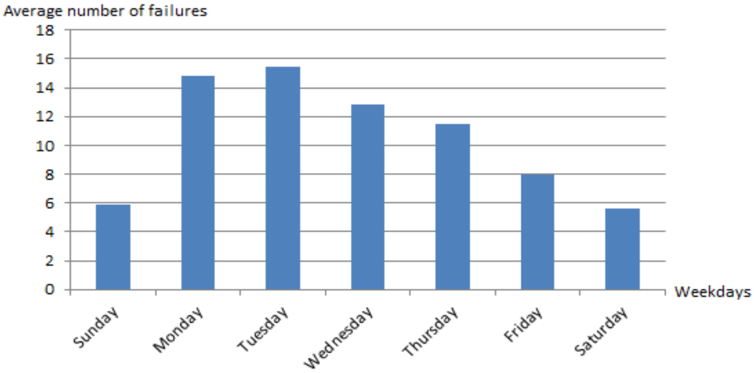


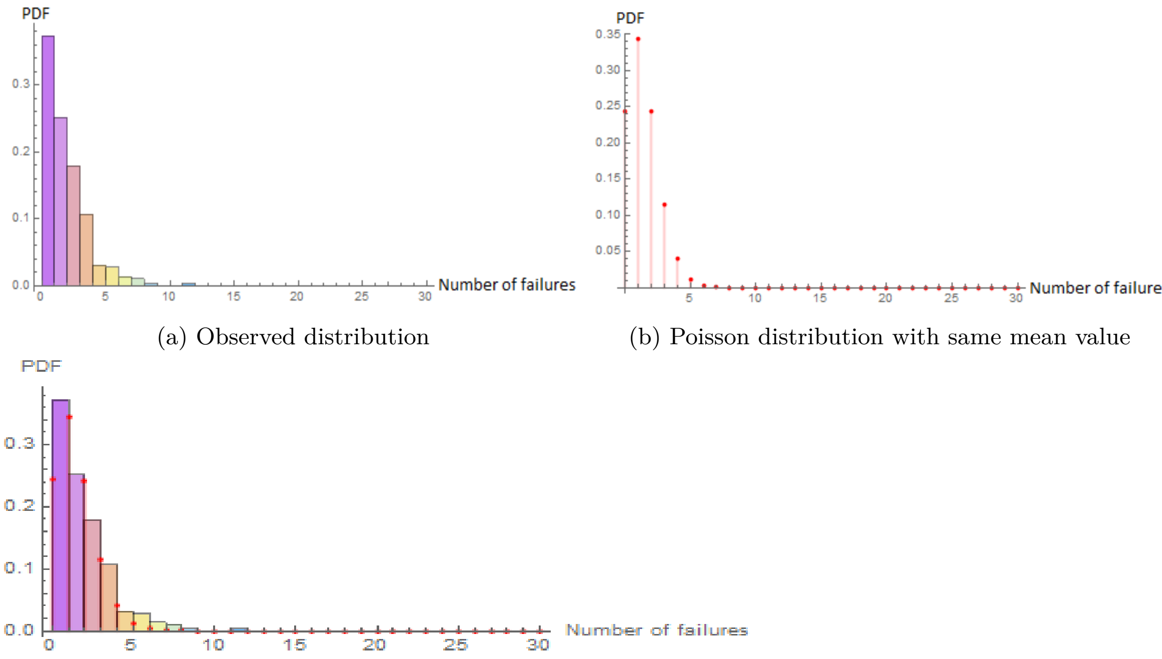
Figure 5.6: Daily pattern of critical failures in Sør-trondelag region

Figure 5.7b shows the Poisson distribution with a mean value (number of failures) calculated over the same period. The observed distribution of high priority failures closely follows the Poisson distribution with same mean value as shown on Figure 5.7c.

Here again, an alternative way to study the distribution is to look at the time between failures. For a Poisson process, time between high priority failures shall have a negative exponential distribution. Figure 5.8 shows how the observed interarrival time distribution looks like and a result from a test for fitness of the empirical distribution with a standard distribution using Q-Q plot as well as result from kolmogorov-simonorv test is also included.

Figure 5.8a shows the observed distribution of time between high priority failures considering the whole network while Figure 5.8b shows the negative exponential distribution considering the mean interarrival time between failures calculated from the failure log data. Here again, the Q-Q plot in Figure 5.8c shows that the observed distribution follows the negative exponential distribution with the same mean for most of the data points, mainly on smaller range of values. And, for large values which has lesser probability of occurrence, the distributions are not similar. A kolmogorov-simonorv test is also conducted which results acceptance of the hypothesis(at the 5% level) that the observed and negative exponential distribution (with same mean value of interarrival time) are the same.

The above tests shows that there is a good similarity between the observed distribution and negative exponential distribution with the same mean in most cases, but it fails for some large values on the Q-Q plot. Here also, there is some bursty behaviour which can be somehow explained by Figure 5.9. The burstiness on the system level is lower than what we have seen on the regional level. As we can see



(a) Observed distribution

(b) Poisson distribution with same mean value

(c) Comparison between the observed distribution and Poisson distribution with same mean value

Figure 5.7: High priority distribution considering the whole network

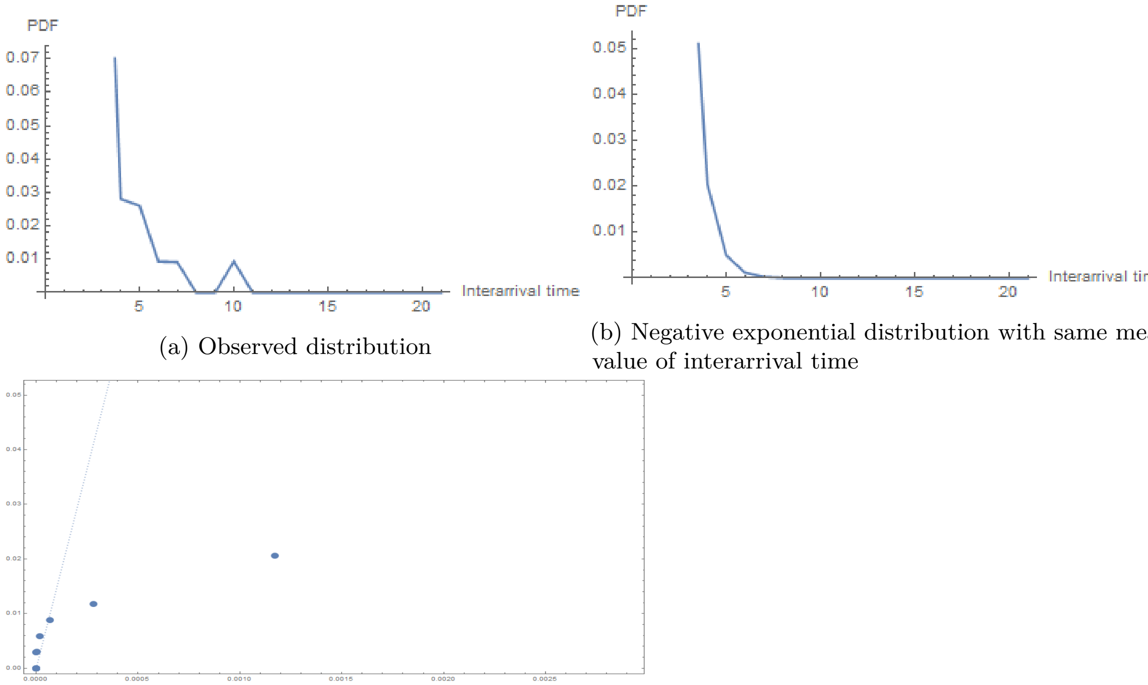
on Figure 5.9, there are days where we have quite many failures and there are also other days with too few critical failures. Saturday and Sunday have small number of failures while the rest weekdays have more failures than the weekend days. Similar to the pattern we have seen on a regional level, Figure 5.9 shows that there are more failures (up to 800 failure per day in average) on few days, especially Monday, Tuesday and Wednesday.

5.3.2 Distribution of Low priority failures

The distribution of low priority alarms(P3 - P7) is also studied both at the system as well as at the sub system(regional) level. It is tried to look at the failure distribution of these low priority failures in a regional level as well as at the system level (considering the whole network).

Regional level: Looking at a regional level, Figure 5.10 shows how the distribution of low priority failures looks like on the regional level considering the whole network.

While Figure 5.10a shows the observed PDF of low priority failures in Sør-



(a) Observed distribution

(b) Negative exponential distribution with same me value of interarrival time

(c) Q-Q plot for comparison between observed distribution and negative exponential distribution with same mean interarrival time

Figure 5.8: Distribution of time between high priority failures considering the whole network

trondelag region, Figure 5.10b shows the Poisson distribution with a mean value calculated over the same period. It can be seen on Figure 5.10c that the observed distribution of low priority failures in counties is somehow similar to the Poisson distribution with the same mean.

System level:

Similarly, at the system level, the distribution of low priority failures(considering the whole network) based on failure records collected for a time span of 3 month is shown in Figure 5.11. The failure data used is the aggregated number of failures recorded every one hour.

On the system level, there are quite a lot low priority failures every hour. Figure 5.11a shows the observed PDF of these low priority failures throughout the whole network and Figure 5.11b shows the Poisson distribution with a mean value (number of failures) calculated over the same period. As it can clearly be shown on Figure

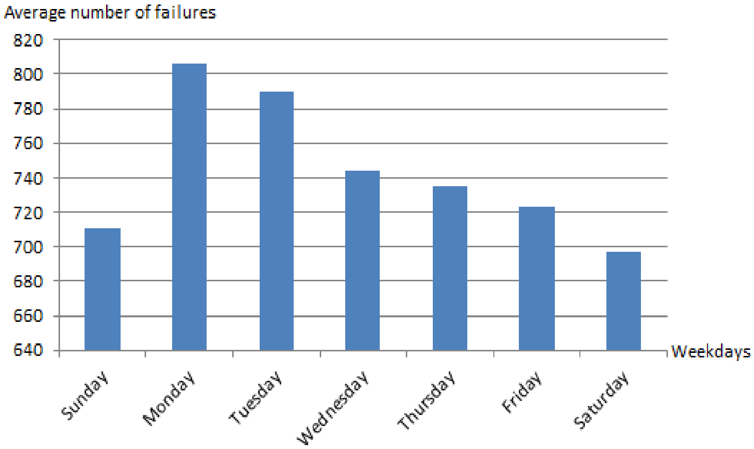


Figure 5.9: Daily pattern of critical failures in the whole network

5.11c, the observed distribution of high priority failures closely follows the Poisson distribution with the same mean. As we have many logs of low priority failures every hours(on the system level), the time between low priority failures is negligible in terms of hours as all the data sets used in this study are aggregated on hourly basis.

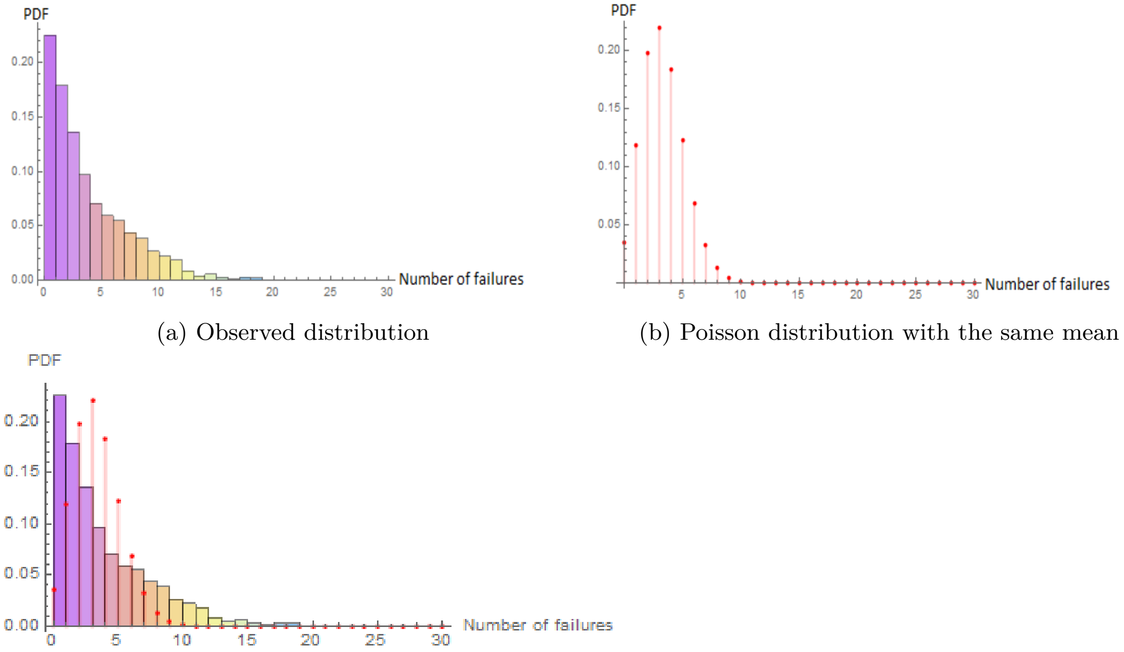
5.3.3 A closer look at the failure frequencies using wavelet

This section aims to give some insight about the interpretations of the result we get after the data sets are transformed into the wavelet domain. As wavelet method is a technique that transform a time domain data in to a frequency domain representation, it can be used to conduct a simple investigation/study of the different frequencies and hence the behaviour of the failure log.

Considering the critical failures at the system level, transformation of the time series representing the priority level of critical(high priority) failures into a frequency domain representation is presented in Figure 5.12.

As we can see on Figure 5.12, there are almost 8 wavelet coefficients showing the different frequencies of high probability failures in the network. Lower wavelet coefficients, such as coefficients from 1 up to 3 represent short term events occurring in a hourly and daily basis. On the other hand, the higher wavelet coefficients (coefficients above 6) represent a long term events/occurrences of failures such as weekly and monthly patterns.

The wavelet coefficients 4 and 5 are events with middle frequencies occurring in a range of few days, less than a week. and these medium frequency ranges are mostly



(a) Observed distribution

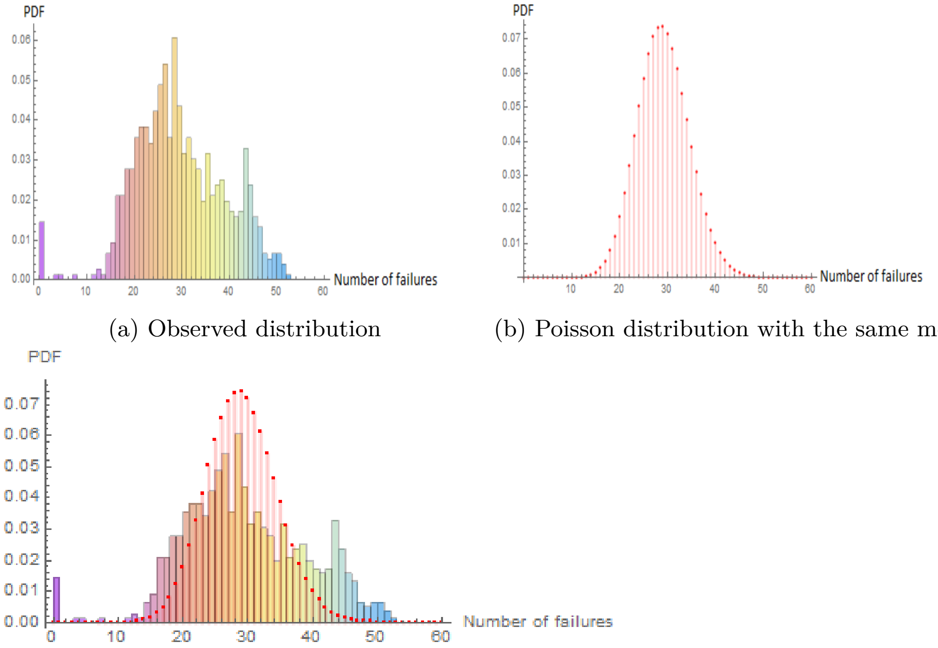
(b) Poisson distribution with the same mean

(c) Comparison between Poisson distribution with the same mean and observed distribution

Figure 5.10: Low priority failure distribution in Sør-trondelag county

used in this study especially for failure prediction in chapter 7.

Similarly, transforming the low priority failure time series using wavelet techniques gives some insight about the effects of daily, weekly etc. patterns of the low priority failures and hence pointing out activities that are responsible for the failures.



(a) Observed distribution

(b) Poisson distribution with the same mean

(c) Comparison between Poisson distribution with same mean and observed distribution

Figure 5.11: Low priority failure distribution considering the whole network

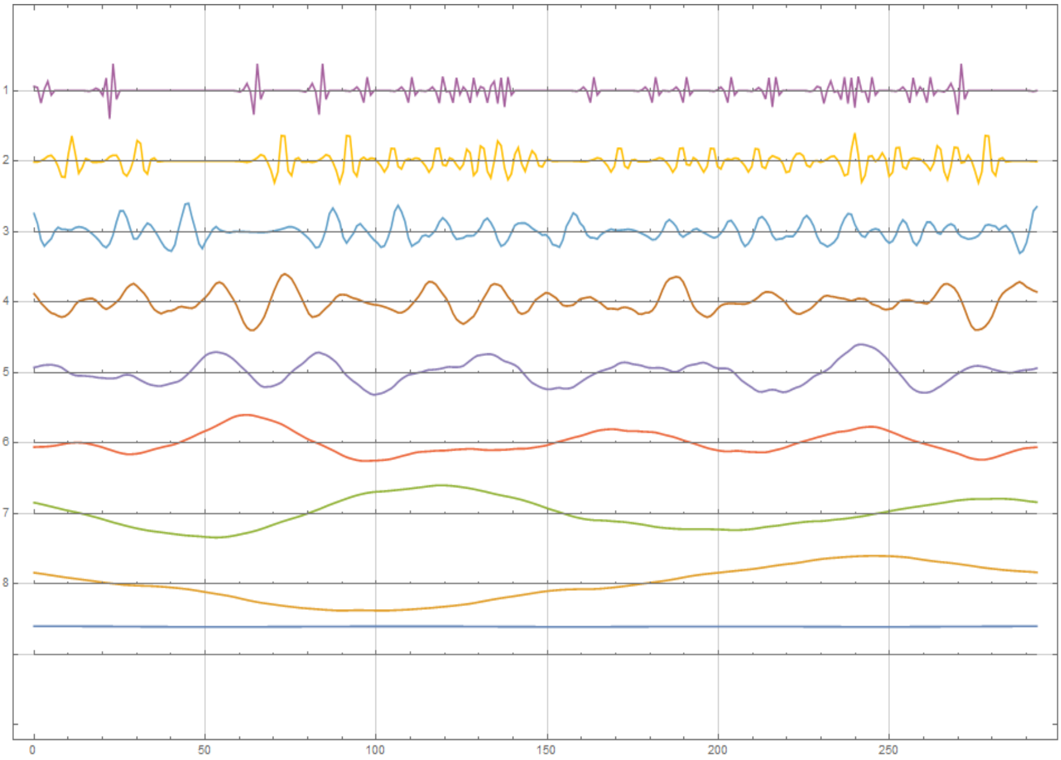


Figure 5.12: A sample wavelet domain representation of high priority failures

Chapter 6
Failure prediction: Methodology

6.1 Introduction

As failure prediction is a key step in proactive fault management of large complex networks, this section discusses about the proposed approach to predict high priority failures by looking at the patterns of low priority alarms in the failure log. Failure prediction tries to avoid service interrupt by applying resolution before fault happens. The main steps in most failure prediction approaches are similar. However, the specific techniques which are used for learning patterns and prediction are different. The section first discusses the basic functionalities expected from the approach. Then, a simple description of the proposed approach is presented.

6.2 Functionalities needed

There are some approaches which use failure logs and tries to avoid service interrupt by applying resolution before fault happens. A simple list of functionalities needed from a prediction algorithm is presented on [Zer15] most of which is also an interest for this study. Generally, the prediction algorithm shall have at least the following functionalities:

- Learning capability
- Low false alarm
- Low missed alarm
- Ability to identify the type of faults and/or the faulty network element
- Less assumption of thresholds
- Easy to deploy
- Moderate processing time

Learning Capability: The proposed approach should be able to learn and adapt the various failure patterns through time and improve the detection performance.

Low false alarm: The technique should also be accurate in detecting real failure patterns. False alarm simply means generating an alarm when there is no real failure occurring. The method should be designed in such a way that it should not generate too much alarm during the system's normal operation.

Low missed alarm: While keeping the number of false alarms low, the technique should not miss a real failure occurring. There must not be too much alarm just not to miss a failure. At the same time, the number of alarms should not be too few which results in missed alarms.

Ability to identify the type of faults and/or the faulty AS: The approach should be capable of identifying the location of occurrence of the root-cause event that caused the instabilities. If possible, it should also pinpoint the exact network elements as well as the time when the root cause event happens.

Less assumption of thresholds: As the networks to be considered are very large and complex, the use of threshold to detect failures should be minimized or avoided if possible.

Easy to deploy: It should not be a complex task to deploy the detection algorithm. It shall not need modification of existing platforms; rather it shall operate on top of them.

Moderate processing time: The required processing time shall be as minimum as possible so that the results would be significant to take protective actions.

6.3 Methodology

The following proposed methodology is mostly depending on the anomaly detection model on the semester project [Zer15]. Relying on the same basic principles, it is tried to customize the model for failure prediction using failure logs. The description of the architecture of the proposed system and its components with their functions is shown in Figure 6.1. The architecture is also basically based on the system described in the semester project report [Zer15].

The proposed approach will have basically three parts;

- *Feature extraction*: This stage takes the pre-processed failure log data, form different data sets which can capture the failure patterns/behavior.

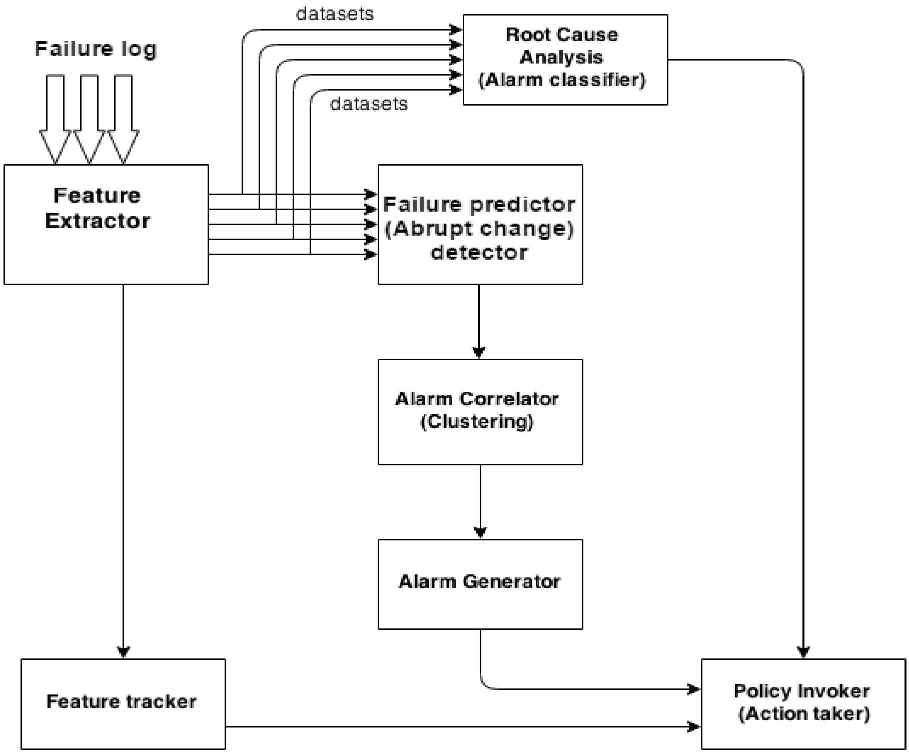

Failure log
datasets
datasets
Root Cause Analysis (Alarm classifier)
Feature Extractor
Failure predictor (Abrupt change) detector
Alarm Correlator (Clustering)
Alarm Generator
Feature tracker
Policy Invoker (Action taker)

Figure 6.1: Proposed system architecture

- *Deviation detection/Actual prediction*: Applying wavelet transform technique on each of the data sets constructed in the Feature extraction stage, this stage is to detect abnormal patterns in each of them.

- *Alarm generation and root cause analysis stages*: Finally, there will be a stage to decide about the deviations found using the wavelet technique and to generate an alarm so that root cause analysis can be initiated to tackle the problem.

6.3.1 Feature extraction

There are characteristics and patterns that are specific to periods of instability. The main purpose of this stage is to extract features that are used to differentiate between the failure log's behavior during normal(with out failures that result in severe consequence) and abnormal(with failure that results in severe consequence) periods.

As pointed out in section 5.2, it is planned to consider at least two level of hierarchies. One at the system level(considering the whole network) and the second considering part of the network with a smaller area(counties/regional level). This will help us to study about both local patterns that result in high priority failures such as an adjacent network element failure and global patterns that occur in other sub networks with their effect propagating in the network. Therefore, with respect to hierarchy level, we will have two scenarios;

Scenario a: The analysis/prediction will be conducted on the regional level (Counties).

Scenario b: The analysis/prediction will be conducted on the system level.

Meanwhile, in each of these hierarchies, it is possible to consider either one data set at a time or many data sets and later to use clustering techniques.

Scenario 1: one data set will be considered at a time.

Scenario 2: two or more data sets will be considered at a time.

As mentioned in [MYC08], it is better to have a reduced data set by just considering a simple count of number of messages (message volume) . Considering one data set/feature at a time will help us to study how much each failure affect or contribute to high priority failures.

However, relying only on one data set (such as the number of message) could make our model weak in such a way that we might miss certain anomaly behaviors. Rather, it is better to have more attributes so that we can have a better capability to capture the instabilities and irregularities. When we have many attributes as the case in scenario 2, the probability that the instability events affect one of the attributes is higher. This will increase the accuracy as compared to scenario 1, but we might need much processing and analysing. Having additional attributes can also be helpful for root cause analysis in locating the anomalous point. [Zer15].

In order to capture all effects, it is proposed to consider all the above scenarios. hence, we will have the following four scenarios:

Regional level

Scenario 1a: The analysis/prediction will be conducted on the regional level (Counties) considering one data set at a time. (Simple approach)

Scenario 2a: The analysis/prediction will be conducted on the regional level (Counties) considering more than two data sets.(clustering)

System level

Scenario 1b: The analysis/prediction will be conducted on the system level considering one data set at a time. (Simple approach)

Scenario 2b: The analysis/prediction will be conducted on the system level considering more than two data sets. (clustering)

For Scenario 2a and Scenario 2b, it is planned to consider at least three data sets; data sets related to priority level, consequences and location of the low priority failures/alarms. It is hypothesized that these three attributes can capture instability periods as there will be a change in their values or frequencies when sudden changes occur in the network.

6.3.2 Actual detection

Applying the wavelet based change detection technique to the selected data sets in the feature extraction stage will help us to locate potential anomalies temporally. As we want also not to miss any abnormal deviation and to have a better accuracy, it is also planned to have multi-dimensional anomaly detection. Thus, we will have two cases. A simple approach for detecting on a single data set and a multidimensional approach which has clustering for identifying time correlated patterns.

Temporal localization

It is proposed to use the technique mentioned in paper [MYC08]. The technique used is *The Maximal Overlap Discrete Wavelet Transform.* This is the same wavelet technique suggested on the semester project [Zer15]. Maximal overlap discrete wavelet transform (MODWT) is a Wavelet analysis which facilitates multi-resolution analysis of traffic time-frequency characteristics. The discrete wavelet transform of signal $X[n]$ with length N involves the computation of the convolution between the signal and a family of wavelets. [MYC08] A detail explanation on how discrete wavelet transform is used to get frequency –time representation which in turn can be used to detect anomalies is mentioned in chapter 4. More detail explanation about wavelet transform can also be found on Paper [PG94].

Although DWT can detect abrupt changes in a time series, it may introduce ambiguities in the time domain. A change in the starting point for a time series can yield quite different results due to the alignment of the time series with the averaging intervals predefined by the DWT. In contrast, MODWT is translation invariant in the sense that it preserves regularity information at each point in time for each scale, and it may be computed for an arbitrary length time series. This translation invariant property allows alignment of events in a multi-resolution analysis with respect to the original time series. [MYC08] [Zer15]

Percival and Guttorp [PG94] showed that the wavelet variance could be more efficiently estimated, not by subsampling the convolution of the filters with the data, but rather by retaining all the values. They called this new procedure the maximal overlap discrete wavelet transform (MODWT), and they denote the sequence of MODWT coefficients by $d_{j,k}$. For a sequence of MODWT coefficients the locations k progress in unit steps rather than steps of $2j$ as in the DWT case. MODWT coefficients are obtained without the subsampling step means that the new coefficients at any scale are no longer orthogonal to each other. In addition to the increased efficiency, the MODWT is shift-invariant and can be readily applied to N data when N is not an integer power of 2. [LW01] [Zer15]

Near the beginning and the end of the data sequence the wavelet filters overlap the ends. Percival and Guttorp [PG94] simply discarded the locations where this occurs so that $n_j < N$ MODWT coefficients $d_{j,k}$ are generated at this scale. [LW01] The paper [LW01] uses MODWT technique to detect changes in a different application area. The maximal overlap (MO) wavelet variance is defined on this paper as shown in equation(6.1). [Zer15]

$$\delta^2_{u,j} = \frac{1}{2^j n_j} \sum_{n=1}^{n_j} d^2_{j,n} \text{ (6.1)}$$

By varying window sizes, wavelet analysis can extract multi-resolution properties of the data at different scales. The choice of mother wavelets, or filters, determines the quality of time and frequency localization.[MYC08] In this paper it is proposed to use filter known as the Daubechies family wavelets. If Daubechies' wavelet with two vanishing moments are used as mentioned in [LW01], the MODWT coefficient $d_{j,k}$ can be obtained by equation (6.2):

$$\mathrm{d}_{j,k} = \sum_{i=1}^{2^j+2(2^j-1)} h_{i,j} f(k - 2^j + i) \text{ (6.2)}$$

Change detection algorithm

There are different wavelet-based anomaly detection algorithms. A summary of most common algorithms that are used to detect abrupt changes is found on paper [Zer15]. In this study, two detection algorithms are proposed; One for the simple approach scenario considering one data set at a time and another for multidimensional approach considering more than two data sets.

Simple approach

The basic step in this approach is to detect anomalous points in time domain by applying wavelet technique on each data sets (Temporal localization). After pre-processing and creating time series, the basic procedure of this approach is as follows:

Algorithm 01

1. Select one data set(time series) which represent a low priority failures.

2. Take part of the selected data set(time series) corresponding to a period of maximum 1 and half month(or less up to 25 days). The investigation has shown that using a very long period has a poor performance.

3. Similary, take part of the high priority data set corresponding to a period selected in step 2.

4. If the study is conducted at the system level, only critical high priority failures with four or more failures occurring at a time (in one hour) will be considered. The reason is that there are quite many high priority failures almost every hour and this makes it difficult to predict them in a hourly basis. Hence, for a system level prediction, the study focuses on predicting cases where we have more than four high priority failures at a time(in one hour). Therefore, a time series corresponding to the period in step 2 with points where 4 or more high priority failures occurred is prepared.

5. Apply MODWT transform on the low priority data set using Daubechies wavelet of order 4.

6. Select wavelet coefficients that tends to follow the high priority failure pattern, mainly wavelet coefficient 4 and 5 (for some scenarios coefficient 3 is also used).

7. Locate the peak points in the selected wavelet coefficients and form a new time series for each coefficients that has only peak values.

8. Combine the peak points of the selected wavelet coefficients and form one time series.

9. Use a simple threshold value to screen out noises with very small peak values. This has been investigated for different values and mostly the threshold is set to a small positive value as there are peaks with a very small positive values as well as significant number of peaks with negative values and these all have to be avoided.

10. Check if there are high priority failures following the peak points within some fixed duration T, scanning time (for instance average time between high priority failures). The algorithm is tested for various time ranges starting from the minimum duration(one hour) up to atleast two times the average time between failures. For comparison of the different scenarios T is set to the average time between high priority failures

11. If there are high priority failures within the time duration we set in step 10 of the detected peak points, generate(record) a positive alarm. Otherwise, consider the detection as a false alarm.

Multidimensional approach

For this approach, the first step is similar to the simple approach; to detect anomalous points in time domain by applying wavelet technique on each data sets (Temporal localization). Once abnormal patterns are detected on each data set, the second step will be to identify time-correlated anomalies among the data sets by using clustering mechanism. An alarm will be generated if we have anomalous behaviors detected within a small time range across many data sets (at least more than one data set).

Clustering

The simple approach procedure mentioned above use wavelet techniques to locate anomalies on each of the data sets, but it is hypothesized that there can be some complex abnormal patterns whose effects might not be captured just by considering a single feature. Anomalies that affect two or more features/aspects can be identified by using clustering. Having as many features as possible might also help for further root cause analysis to locate problematic elements so that action can be taken before a severe consequence arises.

The very first steps of the clustering technique are similar to what have been mentioned for the simple approach. It is planned to use at least four data sets representing different perspectives from the failure log (such as priority level, consequence data set, location/spatial information and type of problem). These data sets are going to be transformed into the wavelet domain and once the data sets are transformed into the wavelet domain, representative wavelet coefficient (mainly wavelet coefficient 3, 4 and 5 as it is shown in Figure 7.9 of Chapter 7 that they have a good prediction capability) are going to be selected from each of the transformed representations. The study uses different coefficients for the different scenarios. For example, if we consider two coefficient 4 and 5 from each data set, the clustering takes at least a total of 8 wavelet coefficient as an input.

The simple clustering aims to set a fine location of adjacent peak points and look for patterns that follow the high priority failures. The peak points from the selected wavelet coefficients are going to be extracted and a new data set will be formed by combining these peak points. Similar to the simple approach proposed above, a simple threshold will be used on the newly formed data set to screen out noises with very small peak values.

Lastly, the approach tries to cluster the peak points that are aggregated from all selected wavelet coefficients by first setting the maximum number of clusters into M, where M is the average number of critical failures expected in the observed period. This is not an ideal approach however it is hypothesized that it helps to see if there is pattern for detected points that can be associated with the failures. The peak points in the wavelet coefficients are somehow located sparsely as can be shown in Figure 7.3 of chapter 7, and mostly they are very close to each other. In addition, the number of high priority failures is not small compared to the number of peak points. Hence, setting the maximum cluster size into M roughly groups adjacent peak points between two failures into one cluster and help us to see if there is a pattern that coincide with the occuerence of critical failures.

The proposed clustering approach is also investigated for scenarios where the number of clusters is increased (which in turn means distance considered between neighbouring peak points is decreased).

A k-means clustering technique is used which considers a squared euclidean distance between adjacent peak points(i.e. which considers two dimensional, both y-axis/magnitude and x-axis/time, distance between peak points). When there are low priority failures occurring nearly at the same time (or following each other within a short time), their effect could add up and cause a high priority failure. Hence, it is important to group such points into one cluster(with respect to time). In this study, a failure is represented by different aspect/dimensions and one failure could be detected by those different aspects where the detection points are very close in time domain. Thus, it is trivial to group such adjacent detected points(in time domain) into one cluster.

Meanwhile, it is also important to group detected points based on the severity level. It is not enough to only consider how close are the detected points in time but also it is important to consider how much of the low priority failures are serious ones. Low priority alarms with higher severity rank (higher magnitude) are going to draw the centroid(center of the cluster) towards them and it is logical that they affect the result significantly. In addition, adjacent detected points that has high severity levels and close in time are more likely grouped into one clusters and their effect can also be investigated later. Hence, considering the squared euclidean distance helps to cluster detected points in both dimension.

6.3.3 Alarm Generation and Root Cause Analysis

The generation of an alarm is somehow dependent on the thresholds used in detecting the anomaly temporally across each data set. Having a small threshold in detecting anomalies on each data set will result too many false alarms, while having a high threshold will result in missed alarms. Therefore, system administrators should be able to set an appropriate threshold points in detecting anomalies across each data set. And based on the clustering among the data sets, the system administrator can decide whether further investigation is needed or not to find the root causes. [Zer15]

For the simple approach, threshold setting for detecting peak values of the wavelet coefficients should be done carefully. The threshold values depend on how we set the values during quantification of the different features of the failure log data. Once a threshold is set, peaks higher than the threshold value from the selected wavelet coefficients will be used for generating an alarm. When a peak value higher than the threshold is detected, the high priority data set will be scanned if a failure happens within the time duration we set in step 10 of Algorithm 01(say for instance 3 hours) following the detected peak point. If there is such critical failure on this duration a positive alarm will be generated, otherwise a false alarm will be recorded.

As we are able to track the time information even in the transformed data, root cause analysis can be used to find the exact location of the instability event. The technique takes into account that we store the original data sets we had for each features/aspects and further investigation can be done on these data sets when needed.

For the multidimensional approach, the same threshold setting is needed in each of the selected wavelet coefficients as we have in the simple approach. This will help us to generate alarms for wavelet coefficients from one typical feature data set. Then, using clustering, an alarm will be generated for peak values detected by different wavelet coefficients of the different aspects/features considered. Here also it is considered that all data sets related to extracted features as well as transformed wavelet coefficients are stored for the root cause analysis investigation.

Chapter 7
Evaluation and Discussion

This section is a discussion of the results obtained using the proposed approach presented in chapter 6. The discussions are presented for each of the scenarios identified in chapter 6; both at a regional level and at a system level, considering the whole network.

7.1 Prediction on a regional level

7.1.1 Simple approach

The analysis/prediction is conducted on the regional level (Counties) considering one data set at a time. A data set covering one and half month(47 days) failure log of sør-trondelag region is used for this test. Figure 7.1 shows the the high priority(critical) failures during this period while Figure 7.2 shows the wavelet coefficients of the primary faults after applying a wavelet technique on the low priority data set.

The wavelet domain representation of low priority failures as shown on the Figure 7.1, indicates that the medium frequencies, specifically wavelet coefficient 4 and 5, somehow have a similar patterns with the occurrence of critical failures.

For a readable comparison, Figure 7.3 is presented below with a shorter time range showing the comparison between high priority failures and wavelet coefficients selected for prediction. Figure 7.3a shows how the 4th wavelet coefficient follows/predict the pattern in high priority failures while figure 7.3b shows how the 5th wavelet coefficient follows/predict the pattern in high priority failures. Figure 7.3c shows the relationship between the two coefficients and their prediction capability towards the high priority failures. The figures shows that the peak values of the selected wavelet coefficients somehow predict high priority failures.

In most cases, the bumps on the selected wavelet coefficients occur right before the critical failures. This means we can probably use the peak points in the selected

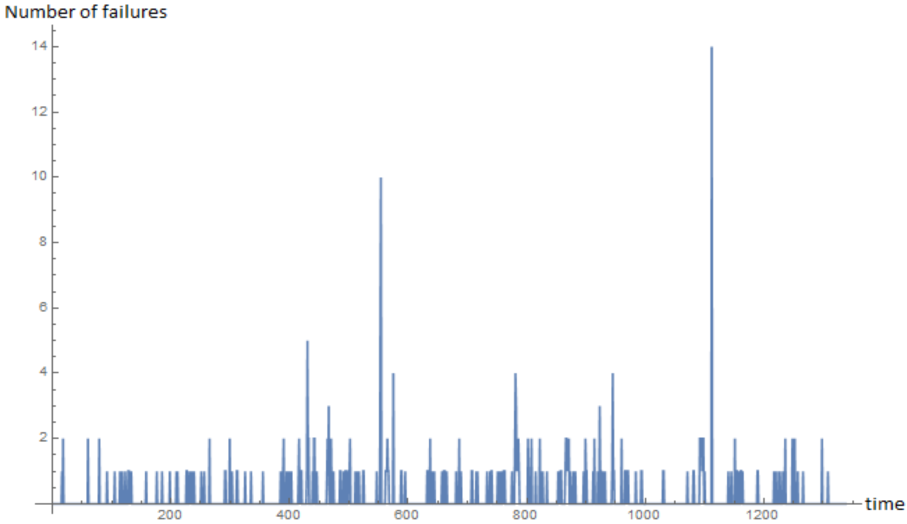


Figure 7.1: High priority failures in Sør-trondelag region

wavelet coefficients to predict critical failures. And, the peak values at the selected wavelet coefficients can be further tested to measure how good is the performance of the failure prediction.

A simple performance test of the prediction capability is to compare the proposed approach's prediction with a random occurrence of the failures. The prediction performance/capability is simply the percentage of failures detected by the specific approach used (in this case the simple approach, Algorithm 01) within a scanning time range of t. It shows how much of high priority failures can be predicted(fall within a scanning time range, t) following an alarm(peak point) on the low priority data set.

The prediction performance is calculated for various scanning time durations and presented graphically. Figure 7.4 shows the prediction performance of the proposed approach at various scanning time ranges as compared to random occurrence of the failures.

In addition to assessing the prediction performance, it is also important to look at how much of the generated alarms end up detecting high priority failures and how much were false alarms. It is tried to calculate the false alarm rate(percentage of peak/detection points that did not predict high priority failures). For easy comparison between the different scenarios, the false alarm rates are calculated at a scanning time range of mean interrarival time between high priority failures.

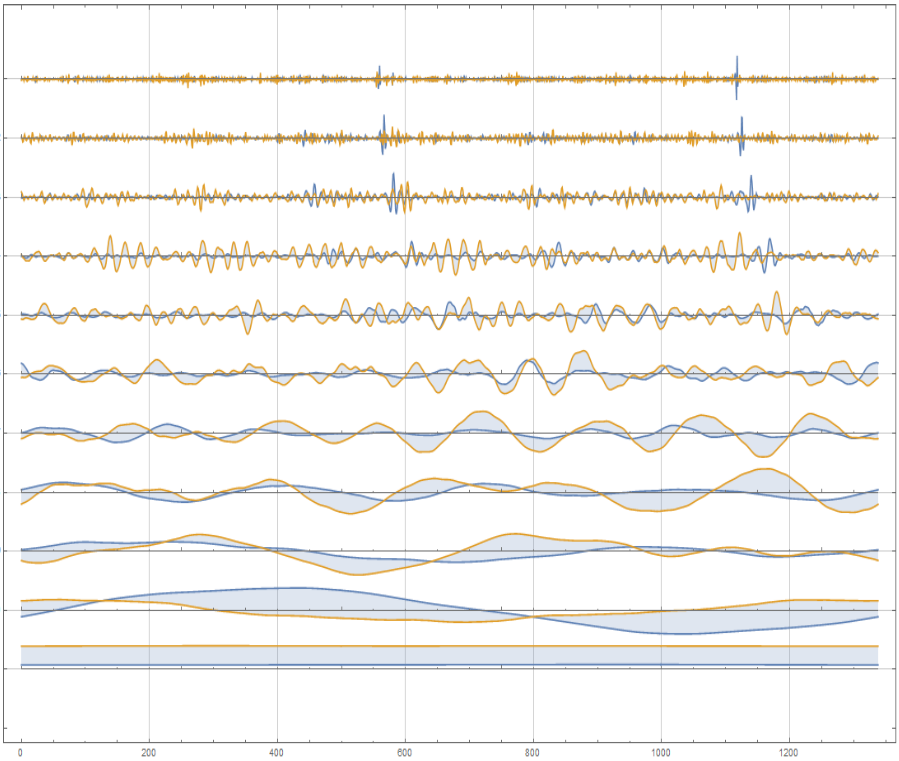
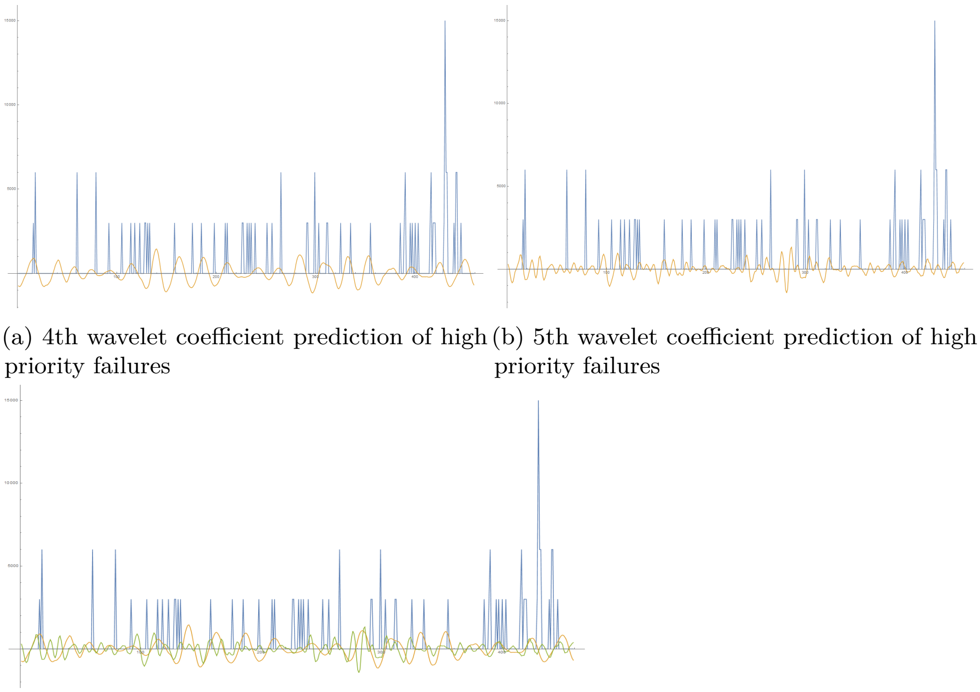


Figure 7.2: A sample wavelet domain representation of low priority failures

Figure 7.4a shows the prediction performance of the simple approach (using wavelet coefficient 4 and 5) to predict critical failures in sør-trondelag region. The prediction performance is presented for various time ranges with an average false alarm rate around 23%(measured at scanning time range equal to the mean interarrival time between critical failures). Figure 7.4b shows the random(poisson process assumption) of failure occurrence with the mean value equal to the average interarrival time between critical failures in the observation period.

Figure 7.4c shows the comparison between the prediction performance of the random occurrence of failures and the performance of the proposed "simple approach" using the selected wavelet coefficient 4 and 5. As we can see on this figure, the prediction performance of the proposed approach is slightly better than the random prediction. However, as we stated in chapter 5, the failure process has some bursty patterns and the exact performance of the random prediction could be somehow lower which means the proposed approach might slightly be more better than the random prediction.



(a) 4th wavelet coefficient prediction of high priority failures

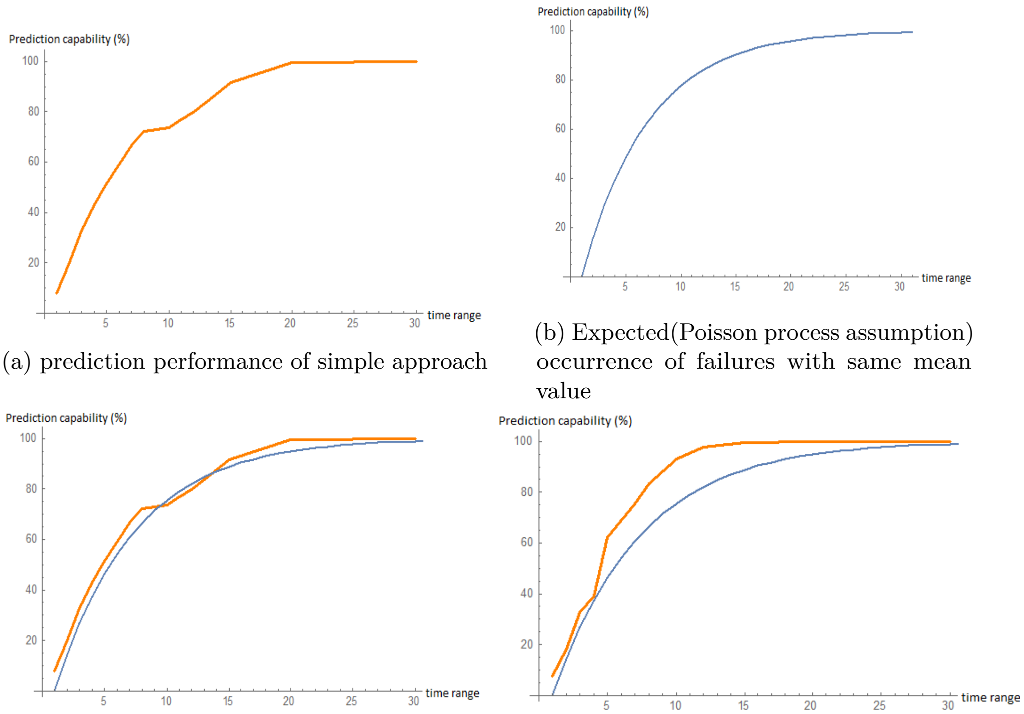
(b) 5th wavelet coefficient prediction of high priority failures

(c) Selected(4th and 5th) wavelet coefficient prediction of high priority failures

Figure 7.3: wavelet coefficient prediction of high priority failures in Sør-trondelag region

In Figure 7.4d, the prediction performance of the simple approach is tested by considering additional wavelet coefficient (wavelet coefficient 3) in addition to the selected wavelet coefficients 4 and 5. The figure shows the comparison between the prediction performance of the random occurrence of failures and the performance of the "simple approach" using the three wavelet coefficients in sør-trondelag region. As we can see on the figure, the prediction performance of using the three wavelet coefficients is better than the above version of the proposed approach using the two wavelet coefficients 4 and 5. However, the false alarm rate(measured at scanning time range equal to the mean interarrival time between critical failures) is around 35% which is somehow larger than the above version of the proposed approach using wavelet coefficients 4 and 5. Though, the false alarm rate is larger in both cases, the simple approach using two wavelet coefficients has a lower false alarm rate with a performance not that much different from the second version using three coefficients.

The prediction performance of the "simple approach" in Sør-trondelag region is also tested by using different data sets. As explained in chapter 5, the failure log



(a) prediction performance of simple approach

(b) Expected(Poisson process assumption) occurrence of failures with same mean value

(c) comparing the performance of the "simple approach" using coefficient 4 5 to a random occurrence of failures

(d) comparing the performance of the "simple approach" using coefficient 3,4 and 5 to a random occurrence of failures

Figure 7.4: Measuring the performance of the "simple approach" in predicting critical failures in sør-trondelag region

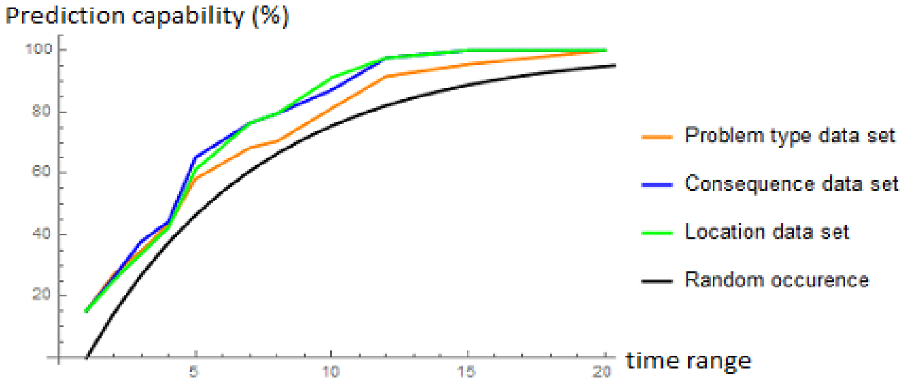


Figure 7.5: Prediction performance of the "simple approach" in Sør-trondelag region using four different dataset

can be represented in different ways such as based on their priority level, location information, type of problem and so on. Here, the proposed approach is tested for three datasets; consequence data set, location data set and type of problem. And, the results are compared to the random prediction performance. As we can see on Figure 7.5, using these data sets gives some how similar result to the priority level data sets we used above on Figure 7.4. But, there are marginal improvements from using these data sets. Using data sets created from the consequence resulted and failure location information gives a better result while using the problem type data set gives a little bit smaller prediction performance.

7.1.2 Simple clustering

The failure prediction is conducted on the regional level (Counties) considering more than two data sets. A simple clustering technique is used to specify peak points that are indicated at least by two or more data sets. In this test, four data sets that are extracted from the failure log(priority level, consequence data set, location data set and type of problem) are used. First, the data sets are transformed into the wavelet domain. Once the data sets are transformed into the wavelet domain, representative wavelet coefficient are selected (different selection for the different scenarios, in this case let's first assume coefficient 4 and 5) from each of the transformed representations. Hence, the clustering takes a total of 8 wavelet coefficient as an input.

The peak points on wavelet coefficient 4 from all the data sets are very close to each other and there is a similar pattern for wavelet coefficient 5(as shown in Figure 7.6). Since the peak points from similar wavelet coefficients are very close to each other, the clustering will group them one cluster. Overall, the clustering aims to set a fine location of such adjacent peak points and look for patterns that follow the high priority failures.

This approach first tries to cluster the peak points from all selected wavelet coefficients into a number M, where M is the average number of critical failures expected in the observed period. It is also tested for the scenario where three wavelet coefficients are used and for two other scenarios with higher M values and the result is shown in Table 7.1. As discussed in chapter 6, k-means clustering technique is used which considers a squared euclidean distance between adjacent peak points.

Figure 7.7 shows the centroids of the clustered data set with the critical failures in sør-trondelag region. As it can clearly seen on this figure, in most cases, the centroids of the clusters somehow occur right before the critical failures. To further investigate the prediction performance, a pictorial comparison of the simple clustering technique with the random occurrence of failures is shown in Figure 7.8.

As seen on figure 7.8, the use of the proposed simple clustering technique improves

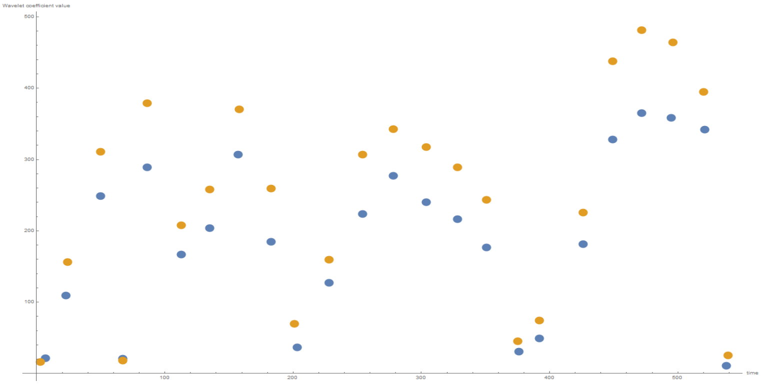


Figure 7.6: comparison of the 5th wavelet coefficient from two different data sets representing priority level and spatial information.

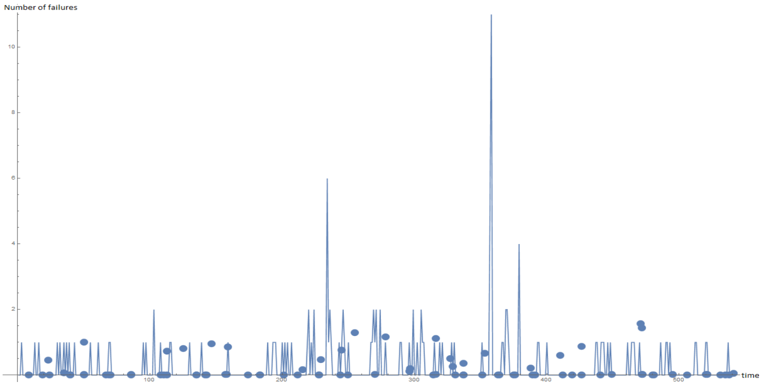


Figure 7.7: Prediction of critical failures in Sør-trondelag region using a simple clustering technique

the performance relative to the simple approach. The false alarm rate in using the simple clustering technique is almost similar to the simple approach with the clustering having a slightly lower false alarm rate in some cases(a decrease by 4%). As we can see from the result, using different data sets and clustering does not give much different result. This is because of the reason that even if we represent the failure log with different data sets from different perspectives, in all the cases the failures have similar frequencies. And, the wavelet domain representation has similar pattern and clustering could not give has new locations other than refining the placement of the peak points.

A similar result is obtained for the scenario where three wavelet coefficients are used as shown in Table 7.1. However, when the number of clusters is increased (distance considered between adjacent peak points is decreased), the result is improved

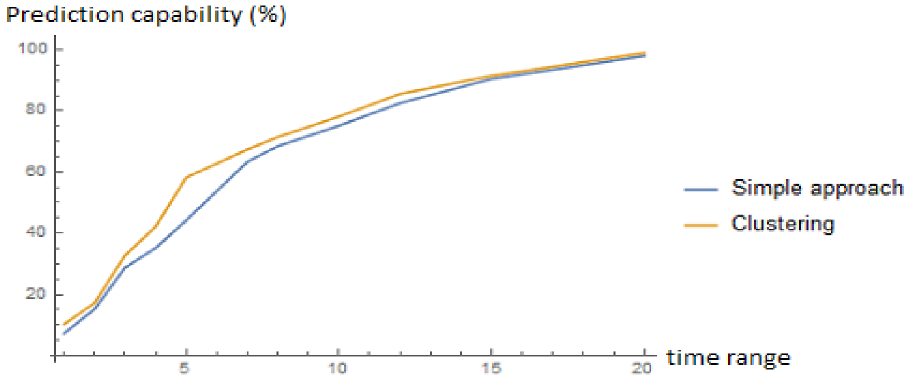


Figure 7.8: Prediction performance using a simple clustering technique in Sør-trondelag region

as shown in Table 7.1. This study does not look into setting an optimal cluster size(distance between adjacent points), but a more better approach could be designed to find an optimal value of cluster size/distance considered between adjacent points to be clustered.

7.2 Prediction on a system level

The analysis/prediction conducted on the system level considering one data set at a time. A data set covering 1 month failure log of the whole network is used for this test. For system level prediction, very critical high priority failures are used for the study. The high priority failures are filtered so that only those points where four or more critical failures occurred at a time (in 1 hour) are used.

The wavelet domain representation of low priority failures, mainly the medium frequencies, wavelet coefficient 3, 4 and 5 together with the critical failures in the whole system during the observation period is shown on the figure 7.9.

Figure 7.9 is presented with a shorter time range showing the comparison between high priority failures and the wavelet coefficients selected for prediction. Figure 7.9a shows how the 3rd wavelet coefficient follows/predict the pattern in high priority failures while figure 7.9b shows how the 4th wavelet coefficient follows/predict the pattern in high priority failures and figure 7.9c for the the 5th wavelet coefficient prediction ability. Figure 7.9d shows the relationship between the three coefficients and their prediction capability towards the high priority failures. The figures shows that the peak values of the selected wavelet coefficients somehow tend to follow the critical failures pattern and hence can be used to predict these high priority failures on the system level.

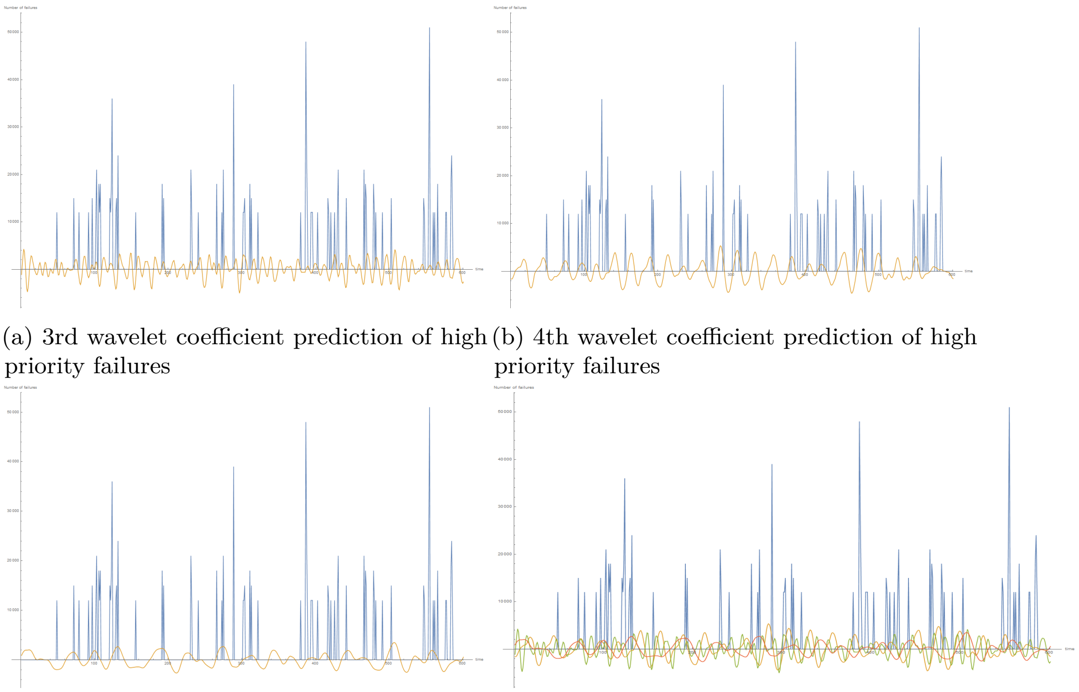
Table 7.1: Comparison of different approaches/scenarios based on false alarm rate and prediction capability (Regional level)

Approach and scenario	**False alarm** (measured at the average time between failures)	**Prediction capability** (measured at the average time between failures)
Simple approach(considering coefficient 4 and 5)	23%	67%
Simple approach(considering coefficient 3, 4 and 5)	35%	78%
Clustering (into, M clusters where M is average number of critical failures) considering coefficient 4 and 5	20%	70%
Clustering (into M clusters where M is average number of critical failures) considering coefficient 3, 4 and 5	35%	80%
Clustering (into 2M clusters where M is average number of critical failures) considering coefficient 3, 4 and 5	33%	84%
Clustering (into 3M clusters where M is average number of critical failures) considering coefficient 3, 4 and 5	33%	84%

And, the peak values at the selected wavelet coefficients can be further tested to measure how good is the performance of the failure prediction.

A simple performance test of the prediction capability(using wavelet coefficient 4 and 5) is conducted and the result is presented in figure 7.10. The figure compares the prediction performance of the proposed simple approach with a random occurrence of the failures. The percentage of detected failures for various time lengths are shown in the figure where the random occurrence of critical failures is calculated based on the average interarrival time between critical failures considering the failure process is poisson.

As we can see on the figure 7.10, the prediction performance of the proposed approach is slightly better than the random prediction. The prediction capability on the system level is also somehow better than the prediction capability we have got when considering on a regional level. The false alarm rate is around 20%(measured at the mean interarrival time) which is slightly lower than what we have on the regional level. As stated in chapter 5, the failure process on a system level has also



(a) 3rd wavelet coefficient prediction of high priority failures

(b) 4th wavelet coefficient prediction of high priority failures

(c) 5th wavelet coefficient prediction of high priority failures

(d) Selected(3rd, 4th and 5th) wavelet coefficient prediction of high priority failures

Figure 7.9: wavelet coefficient prediction of high priority failures in the whole network

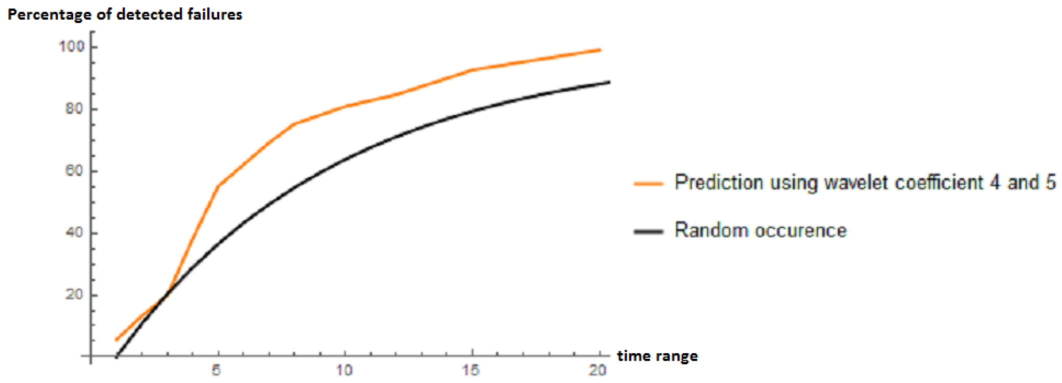


Figure 7.10: comparing the performance of the "simple approach" using wavelet coefficient 4 and 5 to a random occurrence of failures.

some bursty patterns and the exact performance of the random prediction could be somehow lower due to this burstiness which means the proposed approach might more slightly better than the random prediction.

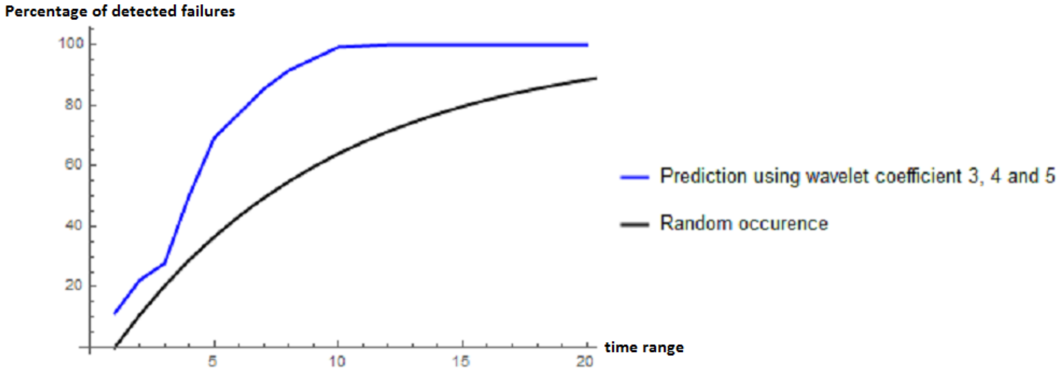


Figure 7.11: comparing the performance of the "simple approach" using wavelet coefficient 3, 4 and 5 to a random occurrence of failures.

In Figure 7.11, the prediction performance of the simple approach is tested by considering additional wavelet coefficient (wavelet coefficient 3) in addition to the selected wavelet coefficients 4 and 5. The figure shows the comparison between the prediction performance of the random occurrence of failures and the "simple approach"'s prediction using the three wavelet coefficients on the system level. It can be clearly seen that the prediction performance of using the three wavelet coefficients is better than the above version of the proposed approach using the two wavelet coefficients 4 and 5. However, the false alarm rate(measured at scanning time range equal to the mean interarrival time between critical failures) is around 34% which is somehow larger than the above version of the proposed approach using wavelet coefficients 4 and 5.

The simple approach on a system level using the three wavelet coefficients gives a good prediction especially in the middle areas near to the mean interarrival time of failures. In this analysis and comparisons, the failure process is considered as Possion which might not be completely true as explained in chapter 5 where there is also some bursty behaviours on week days. This implies that the prediction performance might be slightly better than what is presented here as the comparison of the result with random occurrence of failures is optimistic.

7.2.1 Simple clustering

The failure prediction is conducted on the system level considering more than two data sets. Similar to what we have done for the regional level, the simple clustering technique is used to locate optimized peak points that are indicated by the four data sets extracted from the failure log(priority level, consequence data set, location/spatial information and type of problem). Here again, once the data sets are transformed into the wavelet domain, representative wavelet coefficient are selected (different selection for the different scenarios, in this case let's first assume coefficient 4 and 5) from each of the transformed representations. Then, the proposed simple clustering approach first tries to cluster the peak points from all selected wavelet coefficients into a number M, where M is the average number of critical failures expected in the observed period.

The approach is also tested for the scenario where three wavelet coefficients are used and for two other scenarios with higher M values and the result is shown in Table 7.2. As discussed in chapter 6, a k-means clustering technique is used which considers a squared euclidean distance between adjacent peak points. Figure 7.12 shows a pictorial comparison of the prediction capability(percentage of detected high priority failures) of simple clustering technique with the prediction capability of the simple approach discussed above on a system level considering the whole network.

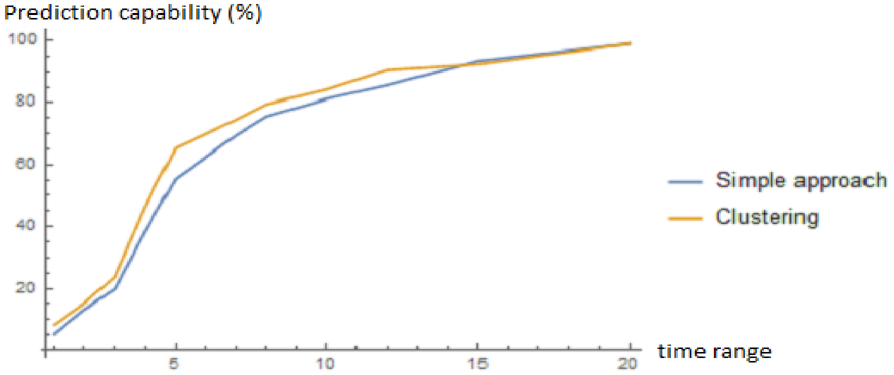


Figure 7.12: Prediction performance using a simple clustering technique(considering coefficient 4 and 5) at the system level.

As seen on figure 7.12, the use of the proposed simple clustering technique gives a very slightly better performance than the simple approach. The false alarm rate in using the simple clustering technique is slightly lower than the simple approach explained above as show in in Table 7.2. As we can see from the result, using different data sets and clustering(for cluster size equal to average number of failures) does not give much different result. This is because of the reason mentioned in section 7.1.2 that the different data sets used in the simple clustering has different information but whose frequencies are more or less the same.

A similar result is obtained for the scenario where three wavelet coefficients are used as shown in Table 7.2. Here again, when the number of clusters is increased (distance considered between adjacent peak points is decreased), the result is somehow

Table 7.2: Comparison of different approaches/scenarios based on false alarm rate and prediction capability (System level)

Approach and scenario	**False alarm** (measured at the average time between failures)	**Prediction capability** (measured at the average time between failures)
Simple approach(considering coefficient 4 and 5)	20%	78%
Simple approach(considering coefficient 3, 4 and 5)	34%	91%
Clustering (into M clusters where M is average number of critical failures) considering coefficient 4 and 5	20%	82%
Clustering (into M clusters where M is average number of critical failures) considering coefficient 3, 4 and 5	36%	94%
Clustering (into 2M clusters where M is average number of critical failures) considering coefficient 3, 4 and 5	35%	95%
Clustering (into 3M clusters where M is average number of critical failures) considering coefficient 3, 4 and 5	35%	95%

improved marginally as shown in Table 7.2.

Chapter 8 Conclusion

The paper first gives an overview of the previous works of failure analysis and prediction mainly using log files. The basic features and trends in the logs of failure data are studied briefly which gives some insight about the failure process and the behaviour of the network to be studied. It is also tried to specify the basic criteria and functionalities to be considered in proposing a better approach to predict system level critical failures by studying primary(low level) failure records. And lastly, the proposed better approach is presented. Unlike previous failure prediction approaches which often use time domain analysis, the proposed approach is based on wavelet technique which is not extensively investigated in failure prediction.

The proposed approach has two ways of predicting critical failures; one is a simple approach where a wavelet technique is applied on one data set while the second type takes 4 types of data sets as an input and use a simple clustering technique. The prediction is tested both at a regional/sub system level as well as on a system level considering the whole network and an explanation to the results is included.

The prediction capabilities of the proposed approaches is somehow slightly better than what we could have got from a random occurrence of failures. The result showed that there is a relatively better prediction capability on a system level than a regional one. This might be due to the reason that we have much more data and hence more dynamics(changes every few minutes and hour) on the system level. The results points that wavelet techniques are more efficient in pointing out anomalous points and prediction when the data considered has some dynamics.

In using a simple clustering technique where the cluster size is set to average number of failures, it was possible to get again to some degree a better result. Even though the data sets used in clustering represent different information, they have almost similar frequencies and it could not give much better result except fine placement of anomalous points in the time domain. Using large cluster size somehow improves the prediction a little bit. This study did not look into setting an optimal cluster size(distance between adjacent points), but a more better approach could be designed to find an optimal value of cluster size/distance considered between adjacent detection (peak) points to be clustered.

In using the wavelet technique, all the raw data (text form) information was quantified to some numeric value. This has not affect the prediction as it has been investigated for different assumptions of values and same result(same frequency pattern) was obtained. However, the choice of threshold values and the period considered (how much of the data set is used) for the analysis matters. The study tried to investigate some scenarios for these values and put suggestions. But, still tuning these values could affect results and hence a more systematic approach could improve the result.

The study does not look into root cause analysis of the anomalous points identified by the proposed approach. Though the performance of the proposed technique is not very impressive, a detailed root cause analysis of the anomalous points could still reveal important findings. As all the original data sets(before the wavelet domain transformation) are stored, they can be retrieved and anomalous points can be further investigated which can potentially improve the results.

Overall wavelet approach is powerful technique in revealing short term variations. However, in this study, the failure log data was not ideal when it comes to the dynamics in the data as there were not much varying frequencies in the lower priority failure data that can be used to model normal and abnormal behaviours. If it was also possible to get more knowledge about the system, for instance information related to dependencies and interconnection between components(topology), the results could have been improved.

References

[ABCM09] Michal Aharon, Gilad Barash, Ira Cohen, and Eli Mordechai. One graph is worth a thousand logs: Uncovering hidden structures in massive system event logs. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 227–243. Springer, 2009.

[ACP09] Georgios Androulidakis, Vassilis Chatzigiannakis, and Symeon Papavassiliou. Network anomaly detection and classification via opportunistic sampling. *IEEE network*, 23(1):6–12, 2009.

[ALRL04] Algirdas Avizienis, J-C Laprie, Brian Randell, and Carl Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE transactions on dependable and secure computing*, 1(1):11–33, 2004.

[AXE15] AXELOS. ITIL—information technology infrastructure library. https://www.axelos.com/best-practice-solutions/itil, 2015.

[BH08] Philip A Bernstein and Laura M Haas. Information integration in the enterprise. *Communications of the ACM*, 51(9):72–79, 2008.

[BRM02] Mark Brodie, Irina Rish, and Sheng Ma. Intelligent probing: A cost-effective approach to fault diagnosis in computer networks. *IBM systems journal*, 41(3):372–385, 2002.

[DTHS09] Shivani Deshpande, Marina Thottan, Tin Kam Ho, and Biplab Sikdar. An online mechanism for bgp instability detection and analysis. *IEEE Transactions on Computers*, 58(11):1470–1484, 2009.

[FSS$^+$13] Ilenia Fronza, Alberto Sillitti, Giancarlo Succi, Mikko Terho, and Jelena Vlasenko. Failure prediction based on log files using random indexing and support vector machines. *Journal of Systems and Software*, 86(1):2–11, 2013.

[FX07] Song Fu and Cheng-Zhong Xu. Exploring event correlation for failure prediction in coalitions of clusters. In *Supercomputing, 2007. SC'07. Proceedings of the 2007 ACM/IEEE Conference on*, pages 1–12. IEEE, 2007.

[HAB$^+$05] Alon Y Halevy, Naveen Ashish, Dina Bitton, Michael Carey, Denise Draper, Jeff Pollock, Arnon Rosenthal, and Vishal Sikka. Enterprise information integration:

successes, challenges and controversies. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 778–787. ACM, 2005.

[ITU15a] ITU. Forum, frameworx—TM [online]. https://www.tmforum.org/tm-forum-frameworx, 2015.

[ITU15b] ITU. ITU - T recommendations. http://www.itu.int/en/ITU-T/publications/Pages/recs.aspx, 2015.

[Jun16] LIU Zheng LI Tao WANG Junchang. A survey on event mining for ICT network infrastructure management. *ZTE Communications*, 14(2), 2016.

[KF05] Emre Kiciman and Armando Fox. Detecting application-level failures in component-based internet services. *IEEE Transactions on Neural Networks*, 16(5):1027–1041, 2005.

[KLA$^+$14] Mohammad Maifi Hasan Khan, Hieu Khac Le, Hossein Ahmadi, Tarek F Abdelzaher, and Jiawei Han. Troubleshooting interactive complexity bugs in wireless sensor networks using data mining techniques. *ACM Transactions on Sensor Networks (TOSN)*, 10(2):31, 2014.

[KMRV03] Christopher Kruegel, Darren Mutz, William Robertson, and Fredrik Valeur. Bayesian event classification for intrusion detection. In *Computer Security Applications Conference, 2003. Proceedings. 19th Annual*, pages 14–23. IEEE, 2003.

[LFWL10] Jian-Guang Lou, Qiang Fu, Yi Wang, and Jiang Li. Mining dependency in distributed systems through unstructured logs analysis. *ACM SIGOPS Operating Systems Review*, 44(1):91–96, 2010.

[Li15] Tao Li. *Event Mining: Algorithms and Applications*, volume 38. CRC Press, 2015.

[LW01] RM Lark and R Webster. Changes in variance and correlation of soil properties with scale and location: analysis using an adapted maximal overlap discrete wavelet transform. *European journal of soil science*, 52(4):547–562, 2001.

[LZXS07] Yinglung Liang, Yanyong Zhang, Hui Xiong, and Ramendra Sahoo. Failure prediction in ibm bluegene/l event logs. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pages 583–588. IEEE, 2007.

[MBZHM08] Adetokunbo Makanju, Stephen Brooks, A Nur Zincir-Heywood, and Evangelos E Milios. Logview: Visualizing event log clusters. In *Sixth Annual Conference on Privacy, Security and Trust, 2008. PST'08.*, pages 99–108. IEEE, 2008.

[MYC08] Jianning Mai, Lihua Yuan, and Chen-Nee Chuah. Detecting bgp anomalies with wavelet. In *IEEE Network Operations and Management Symposium, NOMS 2008-2008*, pages 465–472. IEEE, 2008.

[MZHM09] Adetokunbo AO Makanju, A Nur Zincir-Heywood, and Evangelos E Milios. Clustering event logs using iterative partitioning. In *ACM International Conference on Knowledge Discovery and Data Mining*, pages 1255–1264. ACM, 2009.

[NKN12] Karthik Nagaraj, Charles Killian, and Jennifer Neville. Structured comparative analysis of systems logs to diagnose performance problems. In *Presented as part of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*, pages 353–366, 2012.

[PG94] Donald B Percival and Peter Guttorp. Long-memory processes, the allan variance and wavelets. *Wavelets in geophysics*, 4:325–344, 1994.

[Pol96] Robi Polikar. Fundamental concepts & an overview of the wavelet theory. *The Wavelet Tutorial Part I, Rowan University, College of Engineering Web Servers*, 15, 1996.

[Sch08] Hinrich Schütze. Introduction to information retrieval. In *Proceedings of the international communication of association for computing machinery conference*, 2008.

[Sec15] BalaBit IT Security[Online]. Pattern DB — real time syslog message classification. http://www.balabit.com/network-security/syslog-ng/opensource-logging-system/features/pattern-db, 2015.

[SFMW14] Ruben Sipos, Dmitriy Fradkin, Fabian Moerchen, and Zhuang Wang. Log-based predictive maintenance. In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1867–1876. ACM, 2014.

[SM07] Felix Salfner and Miroslaw Malek. Using hidden semi-markov models for effective online failure prediction. In *Reliable Distributed Systems, 2007. SRDS 2007. 26th IEEE International Symposium on*, pages 161–174. IEEE, 2007.

[SOR$^+$03] Ramendra K Sahoo, Adam J Oliner, Irina Rish, Manish Gupta, José E Moreira, Sheng Ma, Ricardo Vilalta, and Anand Sivasubramaniam. Critical event prediction for proactive management in large-scale computer clusters. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–435. ACM, 2003.

[SP13] Preeti Sharma and Thaksen J Parvat. Network log clustering using k-means algorithm. In *Proceedings of the Third International Conference on Trends in Information, Telecommunication and Computing*, pages 115–124. Springer, 2013.

[TL10] Liang Tang and Tao Li. Logtree: A framework for generating system events from raw textual logs. In *2010 IEEE International Conference on Data Mining*, pages 491–500. IEEE, 2010.

[TLP11] Liang Tang, Tao Li, and Chang-Shing Perng. Logsig: Generating system events from raw textual logs. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 785–794. ACM, 2011.

[TLS12] Liang Tang, Tao Li, and Larisa Shwartz. Discovering lag intervals for temporal dependencies. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 633–641. ACM, 2012.

[TLS$^+$13] Liang Tang, Tao Li, Larisa Shwartz, Florian Pinel, and Genady Ya Grabarnik. An integrated framework for optimizing automatic monitoring systems in large it infrastructures. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1249–1257. ACM, 2013.

[Zer15] Tesfaye Amare Zerihun. Network anomaly detection using BGP updates. 2015.

[ZLO01] Mohammed J Zaki, Neal Lesh, and Mitsunori Ogihara. Predicting failures in event sequences. In *Data Mining for Scientific and Engineering Applications*, pages 515–539. Springer, 2001.

[ZTL$^+$14a] Chunqiu Zeng, Liang Tang, Tao Li, Larisa Shwartz, and Genady Ya Grabarnik. Mining temporal lag from fluctuating events for correlation and root cause analysis. In *10th International Conference on Network and Service Management (CNSM) and Workshop*, pages 19–27. IEEE, 2014.

[ZTL$^+$14b] Chunqiu Zeng, Liang Tang, Tao Li, Larisa Shwartz, and Genady Ya Grabarnik. Mining temporal lag from fluctuating events for correlation and root cause analysis. In *10th International Conference on Network and Service Management (CNSM) and Workshop*, pages 19–27. IEEE, 2014.

Appendix A

Appendix A

Table A.1: Quantification used to create time series based on priority levels

Priority level	**Value assigned**
P1	1000
P2	1000
P3	100
P4	200
P5	300
P6	400
P7	500

Table A.2: Quantification used to create time series based on Problem types

Problem type	**Value assigned**
Aksessnett	4
Annet	8
Data Aksess	12
DXX	16
Energiteknikk	20
GPON	24
IP Nett Brum	28
IP Nett Brut	32
IP Nett Gips	36
IPTV	40
Kystradio	44
Mobil	48
Mobile tjenester BMO	52
Radiolinje	56
Server/Tjeneste	60
Støttesystemer	64
Svitsjing	68
Transport	72
TVinPeaks	76
Wimax	80
xDSL	84

Table A.3: Quantification used to create time series based on location information

Location	**Value assigned**
Agdenes	104
Bjugn	108
Frøya	112
Hemne	116
Hitra	120
Holtålen	124
Klæbu	128
Malvik	132
Meldal	136
Melhus	140
Midtre Gauldal	144
Oppdal	148
Orkdal	152
Osen	156
Rennebu	160
Rissa	164
Roan	168
Røros	172
Selbu	176
Skaun	180
Snillfjord	184
Trondheim	188
Tydal	192
Ørland	196
Åfjord	200

Table A.4: Quantification used to create time series based on consequences resulted due to failures

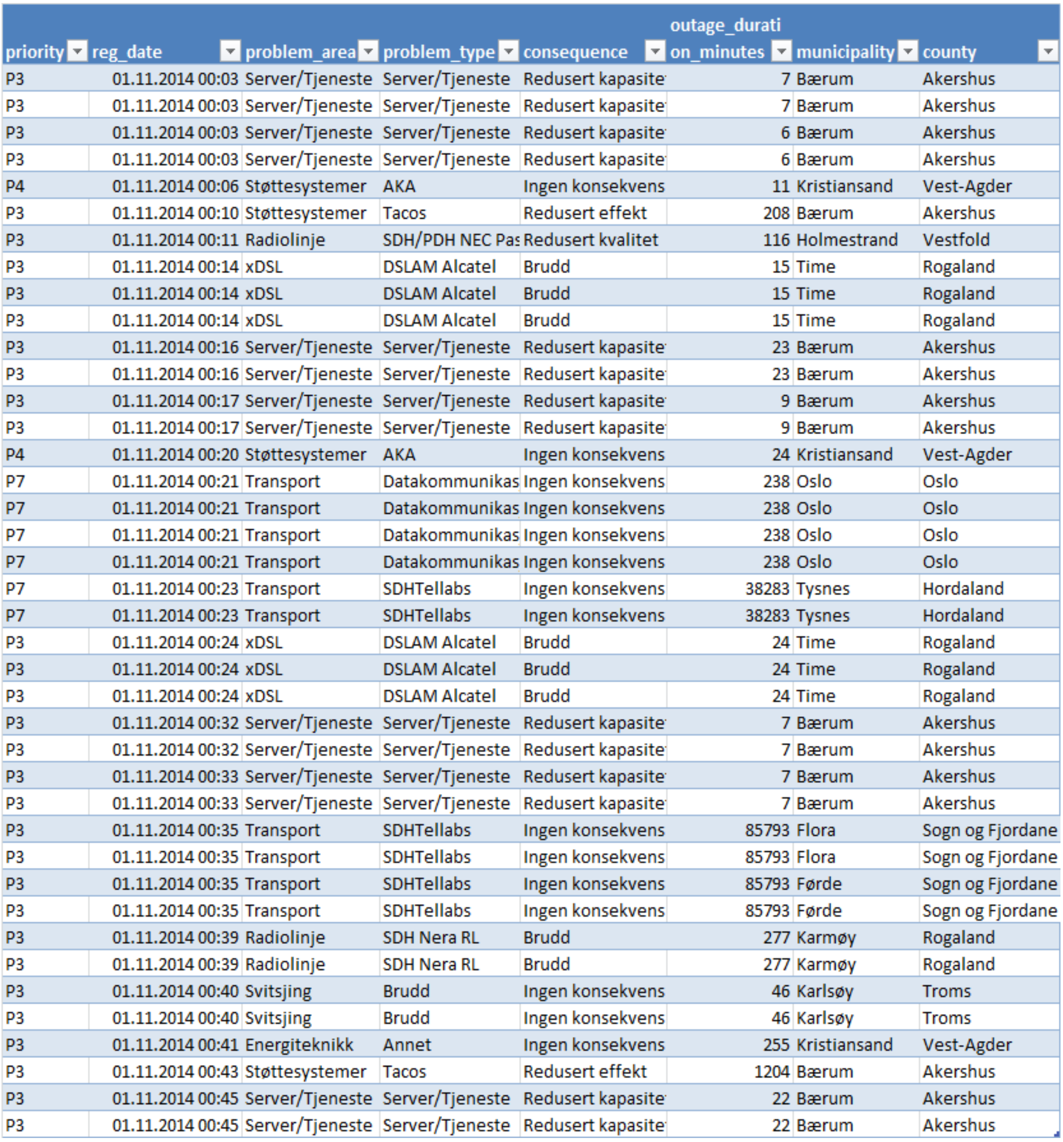
Consequence resulted	**Value assigned**
Ingen konsekvens	50
1 sekund brudd	150
15 minutter brudd	200
30 minutter brudd	250
Korte brudd	300
Alarmkonsekvens	350
Brudd	400
Brudd. Større feil i nettet	450
Ikke klarlagt. Enterprenør tilbakemelder	500
Redusert effekt	550
Redusert kapasitet	600
Redusert kvalitet	650

Appendix B

B.0.1 Pre processing and filtering of raw data

This is a sample to show how the pre-processing is done and how the time series used in the study looks like. The example is based on the log file shown in appendix Table B.1.

1. First, one aspect/dimension is selected from the log file, let's take priority level data set. The series corresponding to priority level is selected together with the time information(when it happended)as shown in Figure B.2 below.
2. Both the priority level and the time information are imported separately. the Priority level has the form $\{''P3'', ''P3'', ''P3'', ...\}$ while the time information has the form {"01.11.2014 00:03", "01.11.2014 00:03", "01.11.2014 00:03",....} as shown in Figure B.3.
3. The time information is put into a list with standard date form. {{2014, 11, 1, 0, 3, 0.}, {2014, 11, 1, 0, 3, 0.}, {2014, 11, 1, 0, 3, 0.}, ..., {2014, 11, 1, 0, 47, 0.}}. The entries represent Year, Month, Day, Hour and Minute respectively.
4. The Priority level is quantized based on the mapping shown in A.1. {500, 500, 500, 500, 400......., 500}.
5. The newly created list for time information is mapped into the respective newly formed priority level to form a series. (Now, the time information is in standard form). {{{2014, 11, 1, 0, 3, 0.}, 500}, {{2014, 11, 1, 0, 6, 0.}, 400}, {{2014, 11, 1, 0, 10, 0.}, 500},.........,{{2014, 11, 1, 0, 47, 0.}, 500}}
6. The data set is filtered to remove repetitive failure records. As it can be seen on Figure B.2, some failures has the same root cause but they are reported by different network elements with the occurrence time being the same. After filtering, only one record will be taken as shown below in Figure B.3.

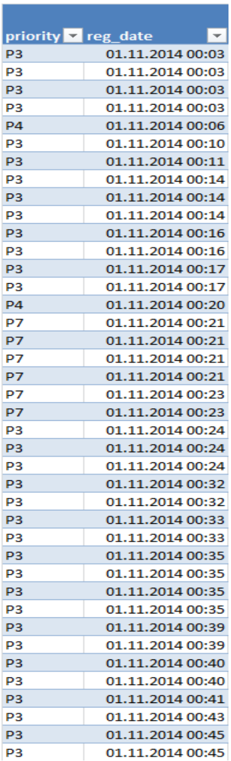


priority	reg_date	problem_area	problem_type	consequence	outage_duration_minutes	municipality	county
P3	01.11.2014 00:03	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	7	Bærum	Akershus
P3	01.11.2014 00:03	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	7	Bærum	Akershus
P3	01.11.2014 00:03	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	6	Bærum	Akershus
P3	01.11.2014 00:03	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	6	Bærum	Akershus
P4	01.11.2014 00:06	Støttesystemer	AKA	Ingen konsekvens	11	Kristiansand	Vest-Agder
P3	01.11.2014 00:10	Støttesystemer	Tacos	Redusert effekt	208	Bærum	Akershus
P3	01.11.2014 00:11	Radiolinje	SDH/PDH NEC Pas	Redusert kvalitet	116	Holmestrand	Vestfold
P3	01.11.2014 00:14	xDSL	DSLAM Alcatel	Brudd	15	Time	Rogaland
P3	01.11.2014 00:14	xDSL	DSLAM Alcatel	Brudd	15	Time	Rogaland
P3	01.11.2014 00:14	xDSL	DSLAM Alcatel	Brudd	15	Time	Rogaland
P3	01.11.2014 00:16	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	23	Bærum	Akershus
P3	01.11.2014 00:16	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	23	Bærum	Akershus
P3	01.11.2014 00:17	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	9	Bærum	Akershus
P3	01.11.2014 00:17	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	9	Bærum	Akershus
P4	01.11.2014 00:20	Støttesystemer	AKA	Ingen konsekvens	24	Kristiansand	Vest-Agder
P7	01.11.2014 00:21	Transport	Datakommunikas	Ingen konsekvens	238	Oslo	Oslo
P7	01.11.2014 00:21	Transport	Datakommunikas	Ingen konsekvens	238	Oslo	Oslo
P7	01.11.2014 00:21	Transport	Datakommunikas	Ingen konsekvens	238	Oslo	Oslo
P7	01.11.2014 00:21	Transport	Datakommunikas	Ingen konsekvens	238	Oslo	Oslo
P7	01.11.2014 00:23	Transport	SDHTellabs	Ingen konsekvens	38283	Tysnes	Hordaland
P7	01.11.2014 00:23	Transport	SDHTellabs	Ingen konsekvens	38283	Tysnes	Hordaland
P3	01.11.2014 00:24	xDSL	DSLAM Alcatel	Brudd	24	Time	Rogaland
P3	01.11.2014 00:24	xDSL	DSLAM Alcatel	Brudd	24	Time	Rogaland
P3	01.11.2014 00:24	xDSL	DSLAM Alcatel	Brudd	24	Time	Rogaland
P3	01.11.2014 00:32	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	7	Bærum	Akershus
P3	01.11.2014 00:32	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	7	Bærum	Akershus
P3	01.11.2014 00:33	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	7	Bærum	Akershus
P3	01.11.2014 00:33	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	7	Bærum	Akershus
P3	01.11.2014 00:35	Transport	SDHTellabs	Ingen konsekvens	85793	Flora	Sogn og Fjordane
P3	01.11.2014 00:35	Transport	SDHTellabs	Ingen konsekvens	85793	Flora	Sogn og Fjordane
P3	01.11.2014 00:35	Transport	SDHTellabs	Ingen konsekvens	85793	Førde	Sogn og Fjordane
P3	01.11.2014 00:35	Transport	SDHTellabs	Ingen konsekvens	85793	Førde	Sogn og Fjordane
P3	01.11.2014 00:39	Radiolinje	SDH Nera RL	Brudd	277	Karmøy	Rogaland
P3	01.11.2014 00:39	Radiolinje	SDH Nera RL	Brudd	277	Karmøy	Rogaland
P3	01.11.2014 00:40	Svitsjing	Brudd	Ingen konsekvens	46	Karlsøy	Troms
P3	01.11.2014 00:40	Svitsjing	Brudd	Ingen konsekvens	46	Karlsøy	Troms
P3	01.11.2014 00:41	Energiteknikk	Annet	Ingen konsekvens	255	Kristiansand	Vest-Agder
P3	01.11.2014 00:43	Støttesystemer	Tacos	Redusert effekt	1204	Bærum	Akershus
P3	01.11.2014 00:45	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	22	Bærum	Akershus
P3	01.11.2014 00:45	Server/Tjeneste	Server/Tjeneste	Redusert kapasite	22	Bærum	Akershus

Figure B.1: Sample failure log raw data used for the study

7. In this example data set we do not have empty values for the priority level. If there were failure records with empty values, an assumed value close to the neighbouring will be given.

8. An event series is formed which places each failure record with respect to absolute time as shown in B.4.

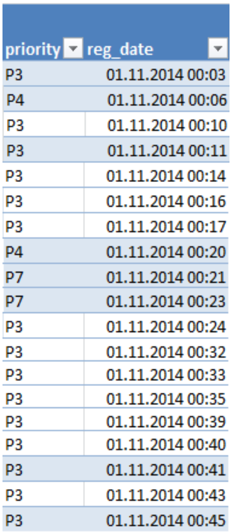


priority	reg_date
P3	01.11.2014 00:03
P3	01.11.2014 00:03
P3	01.11.2014 00:03
P3	01.11.2014 00:03
P4	01.11.2014 00:06
P3	01.11.2014 00:10
P3	01.11.2014 00:11
P3	01.11.2014 00:14
P3	01.11.2014 00:14
P3	01.11.2014 00:14
P3	01.11.2014 00:16
P3	01.11.2014 00:16
P3	01.11.2014 00:17
P3	01.11.2014 00:17
P4	01.11.2014 00:20
P7	01.11.2014 00:21
P7	01.11.2014 00:21
P7	01.11.2014 00:21
P7	01.11.2014 00:21
P7	01.11.2014 00:23
P7	01.11.2014 00:23
P3	01.11.2014 00:24
P3	01.11.2014 00:24
P3	01.11.2014 00:24
P3	01.11.2014 00:32
P3	01.11.2014 00:32
P3	01.11.2014 00:33
P3	01.11.2014 00:33
P3	01.11.2014 00:35
P3	01.11.2014 00:35
P3	01.11.2014 00:35
P3	01.11.2014 00:35
P3	01.11.2014 00:39
P3	01.11.2014 00:39
P3	01.11.2014 00:40
P3	01.11.2014 00:40
P3	01.11.2014 00:41
P3	01.11.2014 00:43
P3	01.11.2014 00:45
P3	01.11.2014 00:45

Figure B.2: Sample failure log raw data after filtering to remove repetitive records.

9. An event series is formed which places each failure record with respect to absolute time as shown in B.4.

10. Lastly, the series is re sampled into a hourly basis. And the corresponding priority level values are expressed for each hour starting from the starting point. The starting point where we have the first failure is considered as the first hour. The priority level values recorded for every hour (i.. starting from the first failure to the point where we have the final failure) are extracted and used for wavelet analysis. In this case, since all the failures happen with in a few minutes, we have only one entry with one hour data, {9000}.



priority	reg_date
P3	01.11.2014 00:03
P4	01.11.2014 00:06
P3	01.11.2014 00:10
P3	01.11.2014 00:11
P3	01.11.2014 00:14
P3	01.11.2014 00:16
P3	01.11.2014 00:17
P4	01.11.2014 00:20
P7	01.11.2014 00:21
P7	01.11.2014 00:23
P3	01.11.2014 00:24
P3	01.11.2014 00:32
P3	01.11.2014 00:33
P3	01.11.2014 00:35
P3	01.11.2014 00:39
P3	01.11.2014 00:40
P3	01.11.2014 00:41
P3	01.11.2014 00:43
P3	01.11.2014 00:45

Figure B.3: Sample failure log raw data used for the study

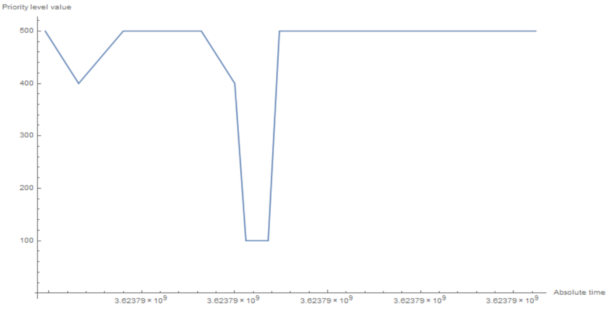


Figure B.4: Event series of sample priority level data set