Master of Science Thesis in Mechanical Engineering
Department of Electrical Engineering, Linköping University, 2017

Wheel Brake Noise Analysis

Teodor Hamnholm Löfgren

LINKÖPING UNIVERSITY

Master of Science Thesis in Mechanical Engineering

Wheel Brake Noise Analysis

Teodor Hamnholm Löfgren

LiTH-ISY-EX--17/5062--SE

Supervisor: **Kamiar Radnosrati**
ISY, Linköpings universitet
Tomas Nordin
Scania

Examiner: **Fredrik Gustafsson**
ISY, Linköpings universitet

Division of Automatic Control
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden

Abstract

The scope of this thesis is to investigate methods of recording, processing and analysing sound data from wheel brake testing in dynamometers with focus on detecting and measuring squeal. The desired outcome is a method that Scania can use to record and analyse brake sound.

A literature study was made to find relevant methodologies and tools proposed in papers, books and industry standards. These methods were tried and evaluated by recording and analysing real sound data and other signals from one of Scanias dynamometers.

The resulting method includes directions on what hardware to use, how to set it up and an algorithm that computes a spectral limit based on normal sound data. This limit is then used as reference when evaluating other recordings. To increase signal to noise ratio, an adaptive filter is proposed to attenuate background noise in the recordings, in particular from the dynamometer and ventilation system.

The conclusion is that it is possible to find squeal using spectral limits based on normal data. The performance of the algorithm is a compromise between being very effective but rather complex, or slightly less effective but also less complex. Its performance is also highly dependent on how squeal is defined. A very narrow definition will only find certain types of squeal while a more broad definition will find more squeal, but also potentially mislabel some recordings.

Contents

1 Introduction **1**
1.1 Background 1
1.2 State of the Art 2
1.3 Report Structure 2

2 Measurements **5**
2.1 Data Acquisition 5
2.1.1 Hardware Setup 5
2.1.2 Hardware Settings and Calibration 7
2.1.3 Squeal Search Testing 8
2.2 Processing and Analysing the Data 9
2.2.1 Validation of Data 10

3 Spectral Estimation **13**
3.1 Discrete Fourier Transform 13
3.2 Welch's Method for Smoothing 14

4 Squeal Detection **17**
4.1 Defining Squeal 17
4.1.1 Squeal Detection Algorithm 18
4.1.2 Parameter Tuning 19
4.1.3 Accelerometer Validation 20
4.2 Squeal Detection Results 21
4.2.1 Accelerometer Validation Results 28
4.3 Squeal Detection Discussion 30
4.A Appendix 31

5 Adaptive Noise Cancellation **37**
5.1 Noise Cancellation 37
5.1.1 Least Mean Square Algorithm 38
5.1.2 Filter Length Estimation 39
5.2 Noise Cancellation Results 40
5.3 Noise Cancellation Discussion 42

6 Conclusion **43**
6.1 Results 43
6.2 Future Work 44

Bibliography **45**

1

Introduction

1.1 Background

Scania is a major manufacturer of trucks, buses and engines. Their products are sold in the premium segment and it is important that they maintain high quality. A major aspect of quality is comfort, which can be negatively influenced by noise. Sources of noise can be for example the engine, gearbox or brakes. This family of problems is known as NVH (Noise, Vibration and Harshness).

The department responsible for most of the brake testing at Scania is called RTCS Brake Performance. Currently, RTCS do not perform any type of testing where the sound or noise characteristics of brakes are analysed and would like a method for doing so. An international standard known as J2521 developed by SAE [17] for lighter vehicles will be used as main reference during the project, as well as an internal document provided by Scania [19].

The current state of the art for brake noise investigation includes theory from many different fields such as vibrations and acoustics, brake testing, strength of materials, signal processing, statistics and psychoacoustics. These different fields all have their own problems and theories and investigating brake noise is therefore a highly interdisciplinary subject. For example, psychoacoustical considerations might be whether a certain sound is perceived as unpleasant or not while someone specialised in materials want to know which materials produce squeal and why. The signal processing aspect include finding the best ways of processing the recorded data to get something useful out of it. The statistical viewpoint might be to find patterns in driving behaviour or environmental conditions that contribute to squeal. The goal from a brake testing perspective is to set up tests that cover a wide range of conditions without being unnecessarily time consuming. The field of vibration analysis deals with the theory of the structural dynamics causing the squeal.

This thesis will mostly discuss issues in the domain of brake testing and signal processing. The main goal is to develop a method for investigating brake noise with focus on detecting and measuring squeal. This method should include both theoretical and practical aspects. In other words, suggestions on what hardware and software to use, how to set it up, how to conduct the testing and recording and how to process and analyse the results.

My hope is that RTCS will be able to use this report and the proposed methodology in the future for investigating the acoustic properties of Scanias brakes. The results should also include a tool that can perform the analysis described in the method and how to interpret the output. It should also be possible to expand or change the method if required, for example by modifying the criteria for squeal detection.

1.2 State of the Art

Brake noise investigation techniques can be divided in two major categories, numerical simulation using FE software and experimental evaluation [3]. The experimental subgroup can be further divided in vehicle testing [15], dynamometer testing [17] and component analysis [8], where dynamometer testing is studied in this project. Testing in dynamometer compared to on road vehicle testing has the benefit of high repeatability and efficiency since environmental effects can be kept more constant and driver and traffic influences are removed. On the other hand, dynamometer testing excludes parts of the assembly such as suspension, rim and tire and might therefore give different results than vehicle testing.

The problem of brake noise has likely existed as long brakes have been around, and can have many different causes and explanations. Therefore, much research have been done on the subject by the automobile and brake industry with the aim of understanding and reducing noise. However, the main physical explanation can be summed up as a result of high amplitude vibrations that arise when the brake setup is excited near one of the systems natural frequencies. According to [14] there are two major types of modes that cause brake squeal.

The first one of these modes is known as stick slip and is caused by the difference between the static and dynamic friction coefficient between the brake pad and the disc. This causes the brake pad to start oscillating forward and backward in the tangential direction of the disc, i.e. it 'sticks' and 'slips'.

The second major mode that causes brake squeal is caused by geometric coupling of the vibrational modes of the disc and pad. The excitation of this mode also originates from variations in the friction coefficient between the brake pad and the disc. Most studies in the literature consider the second mode to be the major cause of brake squeal [1].

1.3 Report Structure

The report consists of chapters divided in sections and subsections. Each chapter is dedicated to a separate part of the project and the chapters with specific results will present them along with a discussion.

Chapter 2 describes the measurements that are done in the project. This includes the equipment, settings and the test cases that are used.

Chapter 3 introduces some basic signal processing theory that is helpful in understanding the report. This chapter is meant for readers who have little to no knowledge of signal processing and can be skipped by people with advanced knowledge in the field.

Chapter 4 proposes a method for detecting and measuring squeal. It also presents some results from applying this method to data recorded in the dynamometer. In the last section of the chapter the results are discussed and the method evaluated.

Chapter 5 presents the concept of adaptive noise cancellation and how it can be used in the brake dynamometer to filter out unwanted noise. Some results are also shown and discussed.

Chapter 6 offers a general discussion of the project, with a finishing conclusion and some suggestions on the direction of future work.

2 Measurements

This chapter provides a detailed description of how the measurements are done in this work. It includes describing the hardware that is used and how it is set up and calibrated. It also describes the tests that are performed and which signals are measured during those tests and how they are measured. The purpose of the chapter is to give useful and practical information on the testing as well as an idea of the possibilities and limitations of the setup that is used.

2.1 Data Acquisition

The brake testing is performed on site in one of Scanias dynamometers manufactured by Schenk. Apart from sound, other signals such as rotational velocity, temperature, torque, pneumatic air pressure and vibrations are recorded and analysed.

2.1.1 Hardware Setup

The data acquisition system that is used for the measurements is made by Dewesoft. The system consists of the hardware module Dewe43 [5] (Figure 2.1) for data acquisition and the software interface DEWESoft X2 for processing and analysing the data. Properties of the applied system can be briefly summarized as:

- Dewe43 can sample and record up to 8 analogue channels with a sample rate of up to 200 kHz. High sample rates are required when recording sound and vibrations.
- The multi channel capabilities facilitate the analysis since all data from the different channels is gathered in the same framework.
- It has the capability of exporting the data files to various formats including *.mat* for analysis in Matlab.

- Additional packages for analysing sound and vibration can be installed.
- Already available at Scania.
- It can perform real time fft analysis and allows user defined functions.

The sound is measured using an ECM 990P condenser microphone (Figure 2.2) connected to a Behringer Xenyx 802 mixer. This is not ideal since the mixer adds an extra element to the setup chain, but it is required to feed the microphone with phantom power. Microphone specifications are shown in Table 2.1.

Table 2.1: *Microphone specifications.*

Frequency range	30-20000 Hz
Nominal impedance	200 Ω
Sensitivity	12 mV/Pa
Max SPL	130 dB
Power supply	DC current 9-48 V phantom power
Connection	XLR

Figure 2.1: *Dewesoft Dewe43 data acquisition system.*

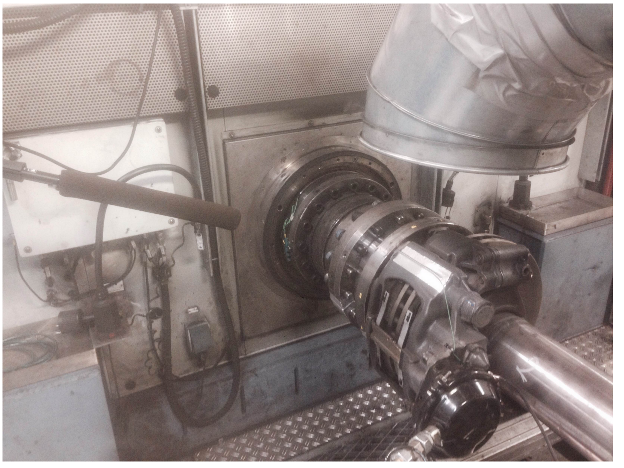

Figure 2.2: *Microphone, brake assembly and ventilation duct.*

Instructions on microphone setup given by the SAE J2521 standard is followed to ensure consistency. That is, the microphone is placed above the brake at 50 cm from the rotational axis of the brake and 10 cm from the plane of the disc. Given that the dynamometer is equipped with the air ventilation placed just above the brake, the microphone is instead placed at an angle from the vertical line, as can be seen in Figure 2.2. It is important that the microphone is not placed in the airflow of the ventilation system, since this leads to undesired noise.

The temperature of the brake disc is measured using a thermo element in the middle of the friction surface. The signal from the thermoelement is given as a voltage. This voltage is converted to degrees using the internal calibration report [21] for the dynamometer. Similarly, the voltage signals for brake torque, rotational velocity and air pressure of the pneumatic brake system are recorded and converted from voltage to kNm, RPM (or km/h) and bar, respectively. The choice of which signals to record are based on [17] and [9], but can be altered depending on the aim and scope of the test.

2.1.2 Hardware Settings and Calibration

The calibration phase using a "Bruel & Kjaer" calibrator is performed regularly between tests or upon any change of the equipment setup as follows. A pure 1 kHz tone at 94 dB is emitted by the calibrator and recorded by the microphone. The amplification of the mixer should be set to 0 dB and the sound pressure level in the recording software is set to match the 94 dB played by the calibrator. This procedure is based on the instructions of the SoundLevel manual from Dewesoft [6].

An important setting when recording data is the sampling frequency. By setting the sampling frequency too low some important information might be lost, while setting it too high creates unnecessarily large amounts of data. The Nyquist theorem [11] is fundamental in signal processing and gives the following relation between sampling frequency ω_s and the Nyquist frequency ω_N

$$\omega_N = \frac{\omega_s}{2} \tag{2.1}$$

The theorem states that the sample rate of a signal should be at least twice as high as the highest frequency component of the signal being sampled. If this condition is not met, the components of higher frequency will be 'folded' into the frequency interval below the Nyquist frequency. Since the audible spectrum of the human ear is roughly 20-20000 Hz, the sound should be sampled using at least 40 kHz according to the Nyquist theorem. The dewe43 uses Sigma-Delta-ADC technology with anti-aliasing (or anti folding) filter, which means that an even higher sample rate of 50 kHz is needed to recreate 19531 Hz using the current setup.

The accelerometer that is used for validating the squeal sounds has a frequency range of 2 Hz to 10 kHz. Therefore, its signal is sampled at 25 kHz to keep most of its useful spectrum below the Nyquist frequency.

Other signals, i.e. temperature, rotational velocity, brake torque and pneumatic air pressure have a larger timescale. They are also different in the sense that they do not oscillate, and therefore there is no interesting frequency content. For these reasons they are sampled at a much lower frequency of 20 Hz to limit the amount of stored data without any significant information loss.

2.1.3 Squeal Search Testing

Several tests are made to investigate which parameters have most impact on the squeal. A dynamometer-program that allows manual pressure variation is created and the values shown in Table 2.2 are used in one of the tests and is used as the main data reference for the squeal search and detection part of the project.

Following brake components are mounted on the brake during this test: Knorr SN7HP brake, TMD 3030 brake pads and a standard disc. The brake pads are worn in using bedding during 100 brakes from 50 to 2 km/h at 3 bar pressure and 100°C initial temperature.

The idea is to try different temperatures, velocities and application pressures to cover a large variety of possible conditions. The temperature T_s refers to the initial temperature of the brake disc and the velocity v refers to the equivalent truck speed in km/h, which was kept constant during each drag application. The pressure p is ramped between 0.9 and 1.4 bar and the direction of the arrow indicates if the ramp is increasing or decreasing, i.e. $\nearrow$ indicates that the pressure is ramped from 0.9 to 1.4 bar and $\searrow$ means that the pressure is ramped from 1.4 to 0.9 bar. Around 0.4 bar is needed in the pneumatic system for the brake pad to apply, 11 bar is the maximum pressure and 1.4 bar corresponds to 2.2-2.3 kNm torque.

These pressures are used since no squeal could be found at higher or lower application pressures. The duration of each drag application is 36 seconds and the entire duration is recorded including 1 second before and 1 second after the brake is applied, resulting in a total of 38 seconds per drag application. After each drag application the brake is cooled down to the initial temperature T_s again. No velocities above 40 km/h are tested since no squeal can be found above 35 km/h.

Table 2.2: *Data table for manual squeal search.*

$T_s = 60°$			$T_s = 90°$			$T_s = 120°$		
N_{drag}	v	p	N_{drag}	v	p	N_{drag}	v	p
1	3	$\nearrow$	19	3	$\nearrow$	37	3	$\nearrow$
2	3	$\searrow$	20	3	$\searrow$	38	3	$\searrow$
3	5	$\nearrow$	21	5	$\nearrow$	39	5	$\nearrow$
4	5	$\searrow$	22	5	$\searrow$	40	5	$\searrow$
5	10	$\nearrow$	23	10	$\nearrow$	41	10	$\nearrow$
6	10	$\searrow$	24	10	$\searrow$	42	10	$\searrow$
7	15	$\nearrow$	25	15	$\nearrow$	43	15	$\nearrow$
8	15	$\searrow$	26	15	$\searrow$	44	15	$\searrow$
9	20	$\nearrow$	27	20	$\nearrow$	45	20	$\nearrow$
10	20	$\searrow$	28	20	$\searrow$	46	20	$\searrow$
11	25	$\nearrow$	29	25	$\nearrow$	47	25	$\nearrow$
12	25	$\searrow$	30	25	$\searrow$	48	25	$\searrow$
13	30	$\nearrow$	31	30	$\nearrow$	49	30	$\nearrow$
14	30	$\searrow$	32	30	$\searrow$	50	30	$\searrow$
15	35	$\nearrow$	33	35	$\nearrow$	51	35	$\nearrow$
16	35	$\searrow$	34	35	$\searrow$	52	35	$\searrow$
17	40	$\nearrow$	35	40	$\nearrow$	53	40	$\nearrow$
18	40	$\searrow$	36	40	$\searrow$	54	40	$\searrow$

2.2 Processing and Analysing the Data

The sound data gathered during the testing is measured by sampling an analogue voltage output from the microphone. This voltage corresponds to the pressure difference from normal atmospheric pressure in Pascal, with a scaling determined by the calibration process. The pressure value is used to calculate the sound dB value, which is given by the following relation

$$L_p = 20\log(\frac{p}{p_0}) \quad dB \tag{2.2}$$

where p is the root mean square of the air pressure and p_0 is the reference pressure. The reference pressure is 20 μPa for air and it is the defined threshold for human hearing referring to a mosquito flying at a distance of 3 meters. This relation means that an air pressure perturbation of 1 Pa is equal to a sound pressure level of 94 dB.

To account for the difference in sensitivity of the human ear to different parts of the sonic spectra it is common to use frequency weighting when measuring sound. The most common weighting scale for representing the perceived sound spectra of the human ear is the A-scale, which is shown in Figure 2.3. A-weighted sound pressure level is written dB(A).

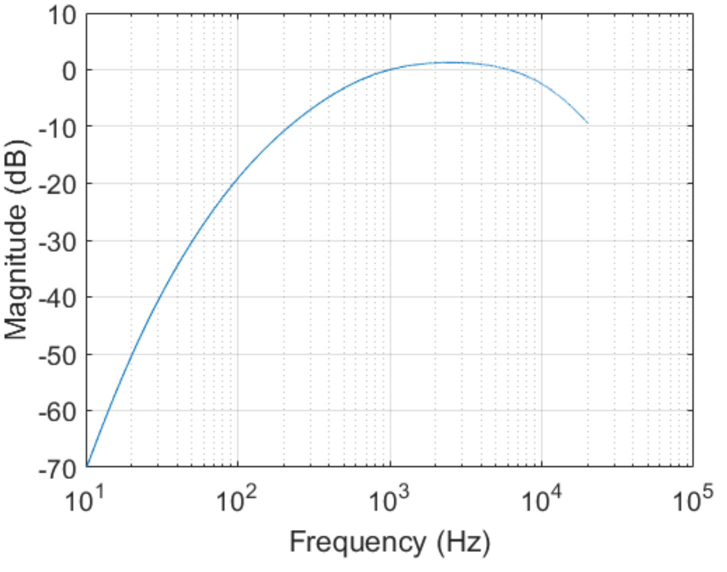

Figure 2.3: *A-weighting frequency scale.*

2.2.1 Validation of Data

When screening for squealing sounds during the brake testing it is of interest to be able to verify the source of the noise. For instance, a squealing sound is not of much interest if it originates from the brake dynamometer machinery or a nearby dynamometer conducting some other test. A method of validation [12], is to use an accelerometer mounted on the brake assembly as shown in Figure 2.4. The vibrations measured by this accelerometer can then be correlated in frequency domain to the signal from the microphone.

It was decided to mount the accelerometer in the out of plane direction of the disc. The amplitude of the vibrations should be highest in this direction, assuming an out of plane vibrational mode as is most common according to [1]. Additional accelerometers could be added in other directions if the shape of the mode is of particular interest. If the frequency components of the squealing sound then appears in both signals it can be assumed with higher certainty that the source of the sound is the brake assembly. Unfortunately, the accelerometer has certain restrictions. For example, it can not be mounted on the brake disc and temperature restrictions make it hard to find an accelerometer that can be mounted on the brake pad. Mass loading effects should also be considered. By mounting an accelerometer the vibrational characteristics of an object change since the mass of the accelerometer is added to the mass of the measured object. A rule of thumb is that the accelerometer should be kept below 10 percent of the weight of the object. This is not considered a problem for the accelerometer currently used, since its weight of 10.5 grams is negligible compared to the 35 kg of the brake caliper. Another aspect of accelerometer validation that should be considered is that vibrations from the machinery could be picked up by the accelerometer as well, leading to incorrect validations.

The accelerometer that was used was an Isotron 7251A, see Table 2.3 for specifications.

Figure 2.4: *Accelerometer mounted on the brake caliper with measurement direction orthogonal to the plane of the disc.*

Table 2.3: *Accelerometer specifications.*

Frequency range	20-10000 Hz
Weight	10.5 g
Temperature range	-55°C to 125°C
Connection	bnc

3

Spectral Estimation

This chapter describes how the sound spectrum is calculated and discusses some of the main issues in transform based spectral analysis. It also explains a spectral smoothing technique known as Welch's method.

The chapter is not meant to give a rigorous theoretical derivation of transform theory, but just a brief introduction to readers of other backgrounds. It may be skipped by people with advanced knowledge in the field.

3.1 Discrete Fourier Transform

The signals from the microphones and accelerometer are, as previously mentioned, given as analogue voltages. These analogue signals are a superposition of sinusoids of different frequencies and amplitudes as well as noise. In the data acquisition system this analogue signal is sampled and converted to a discrete signal. This results in keeping samples of the recorded signals while removing the rest. To extract the frequencies of a discrete signal we need the Discrete Fourier Transform. To arrive at this, we start with the Discrete Time Fourier Transform.

The Discrete Time Fourier Transform (DTFT) [11] is given by

$$X_T(e^{i\omega T}) = T \sum_{k=-\infty}^{\infty} x[k]e^{-i\omega kT} \tag{3.1}$$

where T is the sampling interval, ω is the frequency, x is the signal beeing transformed and k is the time index.

In the physical world a signal can not be infinite and the expression has to be truncated, leading to the following formula

$$X_T^{(N)}(e^{i\omega T}) = T \sum_{k=0}^{N-1} x[k] e^{-i\omega kT} \tag{3.2}$$

where N is the number of samples in our discrete signal. Truncation means that some information is lost leading to the so-called leakage phenomenon. Leakage means that the energy of a frequency component is spread to nearby frequencies leading to less accurate results. The amount of leakage is proportional to the length of the truncation, and therefore leads to a trade off between leakage and signal length. It is still necessary to perform these steps to be able to use these methods on physical signals.

One last step is needed before the transform theory can be used in practice, and that is discretizing Equation (3.2) in the frequency domain as well, which is done in the following expression

$$X[n] = \sum_{k=0}^{N-1} x[k] e^{-2\pi ikn/N}, \qquad n = 0, 1, ..., N-1 \tag{3.3}$$

Equation (3.3) is known as the DFT or Discrete Fourier Transform.

The DFT is a central part of spectral analysis and can be used to directly calculate the frequency spectrum of a signal. However, in practice it is rarely used since the complexity of directly applying the DFT is $O(N^2)$. This can be realized by noting that the sum of N multiplcations needs to be calculated N times. In practice, a more efficient method known as FFT or Fast Fourier Transform is used. The FFT reduces the complexity to $O(N \log N)$, which is an enormous improvement when N is large. This improvement is crucial for performing real time frequency analysis. For further discussions on the FFT and its implementation see [13].

3.2 Welch's Method for Smoothing

When there is significant noise in signals their spectra also have high variance. This can lead to difficulties when trying to visualise the data. To decrease this problem and get nicer plots it is common to use spectral smoothing.

One method for spectral smoothing is Welch's method [11]. The first step is to divide the signal of length N in R temporal batches of length M, giving the relation $N = RM$. The idea is then to compute the periodogram for each batch and average the batches. The periodogram can be calculated by first using the DFT and then squaring the result as shown in Equation (3.4).

$$\hat{\hat{\Phi}}_N[n] = \frac{T}{N} |X[n]|^2 \tag{3.4}$$

The periodogram is then averaged in Welch's method according to

$$\hat{\Phi}_N[n] = \frac{1}{R} \sum_{k=1}^{R} \hat{\hat{\Phi}}_M^{(k)}[n], \qquad n = 0, 1, ..., M-1 \tag{3.5}$$

The benefit of using Welch's method is decreasing the variance by a factor R, at the price of decreasing the frequency resolution by the same factor. An example is provided in Figure 3.1 and 3.2.

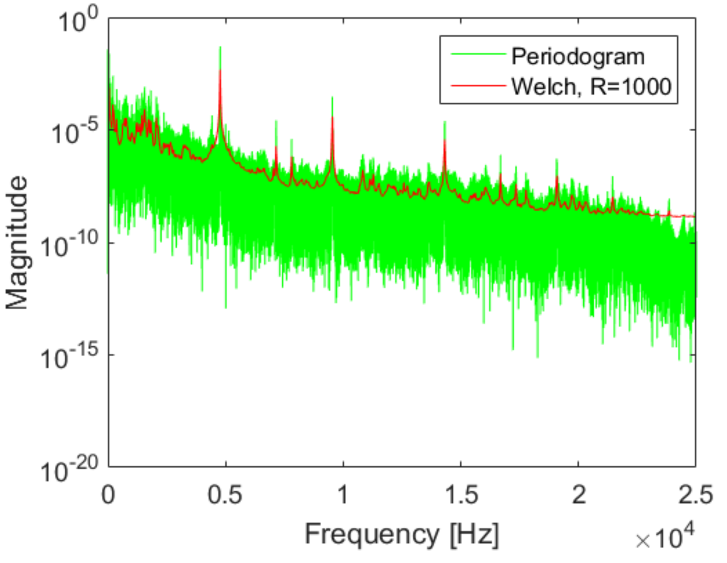

Figure 3.1: *Periodogram of brake sound with squeal*

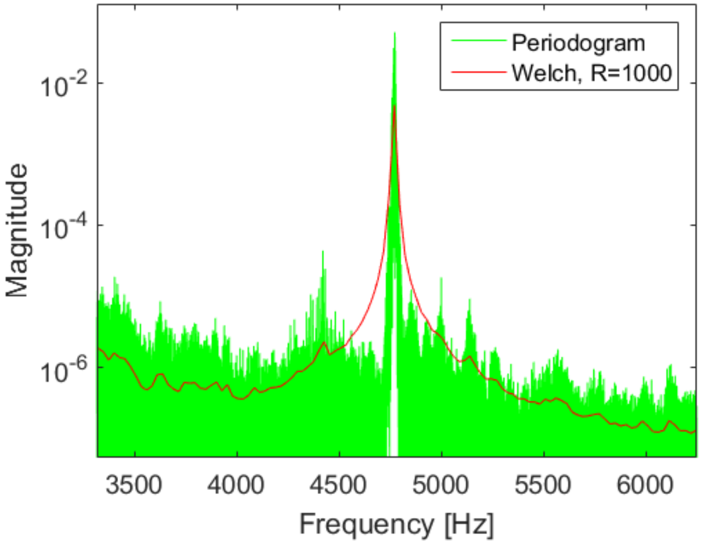

Figure 3.2: *Zoom of Figure 3.1*

4

Squeal Detection

One of the main goals of the project is to develop a way of detecting and quantifying abnormal sound, with focus on squeal. The aim of this chapter is to propose and evaluate such a method. The idea is that the method can be used either directly as it is by the department or for further development and ideas.

4.1 Defining Squeal

To detect squeal it is important to define what squeal is and there are many ways of doing this. For further discussions on human perception and measurement of squeal, please see [2] and [20]. As proposed in the SAE J2521 standard [17] squeal can be defined as spectral peaks above 70 dB(A) between 2 and 16 kHz. This approach is straightforward and easy to implement, however it has certain restrictions. The biggest drawback is that it is not necessarily suitable for the type of sound being investigated. For example, depending on the velocity and pressure of the brake/drag application the average sound pressure level is going to be different. For higher velocities and pressures squeal is more likely to be found since the overall sound pressure level is higher.

A more sophisticated method of detecting squeal is to use recordings of sound that has been rated as normal for the relevant test and compute a type of limit from the spectra of these data. The spectra of other sound recordings could then be compared to this limit, and if it exceeds the limit, it is considered squeal. This approach is more complex, but gives a more customised definition of squeal. The following section will propose a method for computing the limit based on normal sound, and the results section will show the results of applying this method to the data from Table 2.2 with a comparison to the standard SAE method.

4.1.1 Squeal Detection Algorithm

The main idea can be explained in the following way. First the data that should be evaluated and the data that should be considered as normal (without squeal) is chosen. The spectra of the data that is to be evaluated is then averaged for each drag application, leaving a much smaller amount of data. The normal data is used to create an upper spectral limit by taking the maximum value at all frequencies. These two quantities, the average spectra of the data to be evaluated and the upper limit that normal sound never exceeds are then compared. If any part of the data being evaluated exceeds the spectral limit of the normal data, it is considered as squeal. The difference between the limit and the parts of the spectra that exceeds it are then summed up and used as a quantification of the amount of squeal.

The general structure of the implementation can be seen in Figure 4.1. The interesting parts are how the data is processed and what comparisons are made between the evaluated data and the normal data. For the complete Matlab implementation see the chapter appendix.

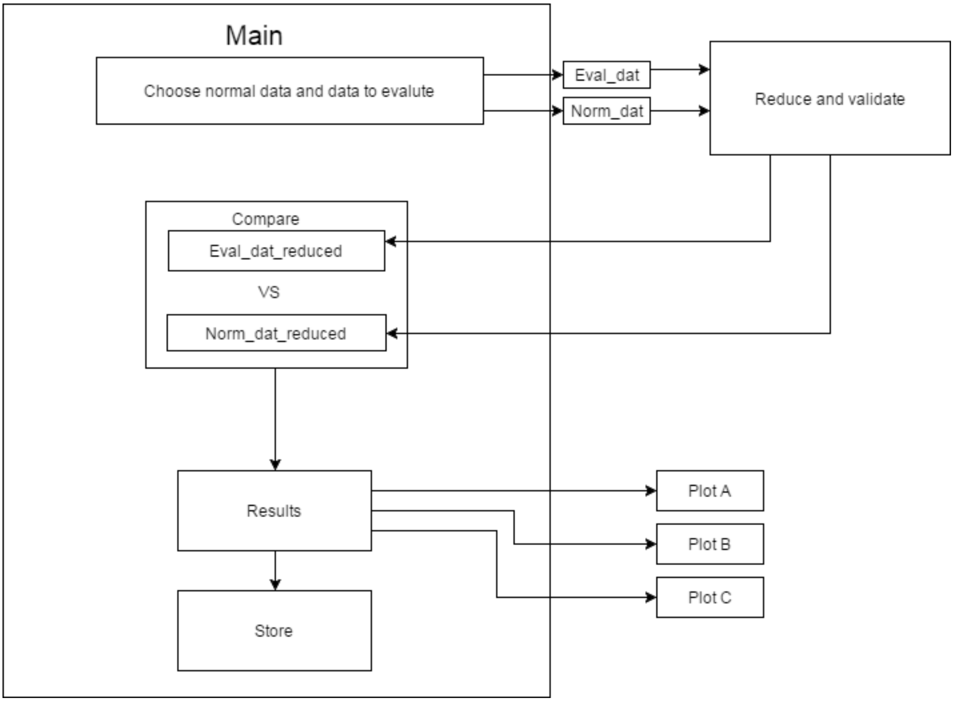

Figure 4.1: *Program structure of the matlab squeal detection/quantification algorithm.*

The 'reduce and validate' function receives the path to the directory with the raw data from the main program. This data includes the signals in time domain and the FFTs of the sound and accelerometer signals that have already been computed in real time by the Dewesoft system. The resolution of the sound signal FFTs is set to 1024 samples, while the resolution of the FFTs of the accelerometer signal is set to 512 samples to make them align in time (the accelerometer signal is sampled with half the sample rate of the sound). These FFTs are then averaged much like in Welch's method (Section 3.2), except that instead of first calculating the periodogram the magnitude of the FFT is used directly according to

$$X[n]_N = \frac{1}{R}\sum_{k=1}^{R}|\hat{X}_M^{(k)}[n]|, \qquad n = 0, 1, ..., M-1 \tag{4.1}$$

where M, which is the length of each FFT, is set to 1024 for the sound signal as previously mentioned. This has three major benefits. It reduces the variance by a factor R as explained in Section 3.2. It does not alter the sound pressure unit like a periodogram would by squaring it. It also reduces the amount of data to store in the working memory by a factor R since only the averaged spectrum is kept in the memory, the rest is overwritten for each recording. The last point is important if we want to process hundreds or thousands of recordings that are each dozens or hundreds of megabytes. The major drawback is that we lose the evolution of the FFTs over time.

Next step is calculating the upper limit of the normal sound. This is done by extracting the maximum possible value for every frequency point out of all FFTs of the normal rated sound. The result is a spectral limit which normal sound never exceeds. This line is then filtered using a zero phase filter to smooth it out. This gives a nice shape of the line, and the last step is to adjust it if required to increase or decrease the sensitivity of the detection. A plot showing this upper limit can seen in Figure 4.7.

After performing these calculations on the normal-rated data and the data to be evaluated the results are returned to the main program. The average spectrum of the sound signal for each brake/drag application is then compared to the computed upper limit of the normal sound. At the frequency points where the average spectrum of the data that is being evaluated lies above the upper limit of normal data the difference is calculated by taking the spectral component exceeding the limit and subtracting the limit. The result is then summed up. This sum is then used as the measure of squeal for each drag/brake application.

4.1.2 Parameter Tuning

A statistical approach can be used to set the parameters and fine-tune the algorithm. One particularly helpful tool is Receiver Operating Characteristic or ROC curves used in detection theory, see [10] and [16]. The idea is to plot the probability of false alarm (in our case this means detecting squeal when there is no squeal) against the probability of detection (detecting squeal when there is squeal) for different threshold values.

The notation can be written

$$\text{decide squeal if: } \sum \Delta > 0 \tag{4.2}$$

$$\text{decide no squeal if: } \sum \Delta = 0 \tag{4.3}$$

where $\sum \Delta$ is the sum of the difference between the spectral components lying above the computed max limit and the max limit itself, i.e. the spectral components exceeding the limit minus the actual limit. This means that the threshold used for the ROC curve is the limit that is either computed using normal sound or the SAE limit. In other words, spectral peaks only contribute to the sum if they exceed the limit. The next step is to compare these algorithmic results with a list of drag applications that have been annotated as 'with squeal' or 'without squeal' and set the following

$$\text{TP if (Algorithm = squeal) and (annotation = squeal)} \tag{4.4}$$

$$\text{FP if (Algorithm = squeal) and (annotation = no squeal)} \tag{4.5}$$

$$\text{TN if (Algorithm = no squeal) and (annotation = no squeal)} \tag{4.6}$$

$$\text{FN if (Algorithm = no squeal) and (annotation = squeal)} \tag{4.7}$$

where TP refers to True positive, FP False Positive, TN to True Negative and FN to False Negative. The True Positive Rate and False Positive Rate are then calculated according to:

$$\mathrm{TPR} = \frac{\sum \mathrm{TP}}{\sum \mathrm{TP} + \sum \mathrm{FN}} \tag{4.8}$$

$$\mathrm{FPR} = \frac{\sum \mathrm{FP}}{\sum \mathrm{FP} + \sum \mathrm{TN}} \tag{4.9}$$

The plot is created by calculating and plotting these two values against each other for different values of a threshold parameter. In the discussed case this threshold was the height of the normal sound limit, as shown in Figure 4.7. For further discussion on sensitivity and specificity of detecting brake squeal see [15].

4.1.3 Accelerometer Validation

A validation of the sound data can also be made by a comparison of the frequency spectrum of the sound and accelerometer signal. Two algorithms were tested and evaluated for this.

The first one is a more simple algorithm. It works by checking if the maximum peak of the average FFT of the accelerometer signal lies within ±2% of the maximum peak of the average FFT of the sound signal. A field called validated is appended to the result structure and set to true if the validation checks out and false if it does not.

The second algorithm computes the coherence between the sound signal and the accelerometer signal in the following way

$$\gamma^2 = \frac{|\Phi_{sa}(f)|^2}{\Phi_{ss}(f)\Phi_{aa}(f)} \tag{4.10}$$

where $\Phi_{sa}(f)$ is the averaged crosspectrum of the sound signal and the accelerometer signal, while $\Phi_{ss}(f)$ and $\Phi_{aa}(f)$ are just the averaged squared amplitudes of the sound signal and the accelerometer signal, respectively. The result is then used to compute the COP or Coherent Output Power according to

$$COP = \gamma^2 \Phi_{ss}(f) \tag{4.11}$$

This COP value can then be compared to a threshold value, and if it exceeds the value it is considered squeal.

4.2 Squeal Detection Results

Table 4.1 shows the same data as Table 2.2 but annotated with checkmarks for squeal and '-' for no squeal. This annotation is based on the author's subjective opinion of what squeal is based on listening to hundreds of drag applications and knowing the conventional definitions of squeal. Drag applications that produce other types of noise than squeal has been annotated as no squeal.

The data that was used as normal data for this test consist of 5 of the drag applications from the actual test that didn't have any squeal.

Some initial observations can be made: squeal is more common for lower velocities, lower temperatures and increasing pressures. The cutoff limits of 2 and 16 kHz proposed in J2521 were changed to 2.5 and 15 kHz to exclude more low frequency noise and some strange high frequency phenomena occurring during some of the drag applications.

Table 4.1: *Data table for manual squeal search with annotations for squeal or no squeal made subjectively by operator.*

$T_s = 60°$				$T_s = 90°$				$T_s = 120°$			
N_{drag}	v	p	$Squeal$	N_{drag}	v	p	$Squeal$	N_{drag}	v	p	$Squeal$
1	3	$\nearrow$	✓	19	3	$\nearrow$	✓	37	3	$\nearrow$	-
2	3	$\searrow$	✓	20	3	$\searrow$	-	38	3	$\searrow$	-
3	5	$\nearrow$	✓	21	5	$\nearrow$	✓	39	5	$\nearrow$	-
4	5	$\searrow$	✓	22	5	$\searrow$	-	40	5	$\searrow$	-
5	10	$\nearrow$	✓	23	10	$\nearrow$	✓	41	10	$\nearrow$	-
6	10	$\searrow$	✓	24	10	$\searrow$	-	42	10	$\searrow$	-
7	15	$\nearrow$	✓	25	15	$\nearrow$	✓	43	15	$\nearrow$	-
8	15	$\searrow$	-	26	15	$\searrow$	-	44	15	$\searrow$	-
9	20	$\nearrow$	✓	27	20	$\nearrow$	✓	45	20	$\nearrow$	-
10	20	$\searrow$	-	28	20	$\searrow$	-	46	20	$\searrow$	-
11	25	$\nearrow$	✓	29	25	$\nearrow$	-	47	25	$\nearrow$	-
12	25	$\searrow$	-	30	25	$\searrow$	-	48	25	$\searrow$	-
13	30	$\nearrow$	✓	31	30	$\nearrow$	-	49	30	$\nearrow$	-
14	30	$\searrow$	-	32	30	$\searrow$	-	50	30	$\searrow$	-
15	35	$\nearrow$	-	33	35	$\nearrow$	-	51	35	$\nearrow$	-
16	35	$\searrow$	-	34	35	$\searrow$	-	52	35	$\searrow$	-
17	40	$\nearrow$	-	35	40	$\nearrow$	-	53	40	$\nearrow$	-
18	40	$\searrow$	-	36	40	$\searrow$	-	54	40	$\searrow$	-

Figure 4.2 shows the ROC plot when altering the threshold value, or max limit value (shown in Figure 4.7) with values from +10 to -10 dB. A perfect squeal detection algorithm would have 0 FPR and 1 TPR.

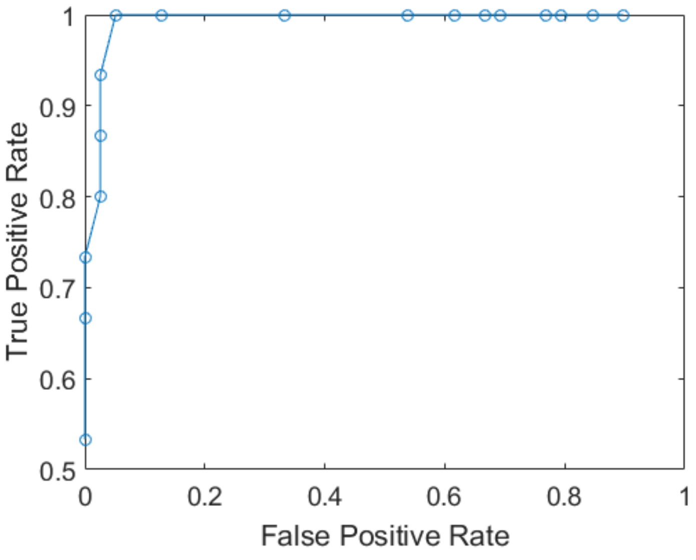

__Figure 4.2:__ ROC curve when adjusting the height of the spectral limit with +10 to -10 dB.

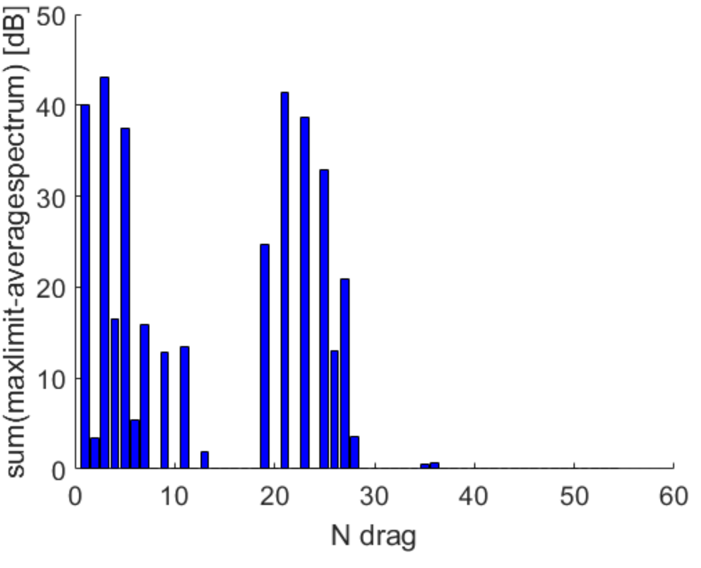

__Figure 4.3:__ Results from the squeal detection algorithm for all 54 drag applications with unaltered max limit.

Figure 4.3, 4.4 and 4.5 shows the result of the squel detection algorithm used on the sound recordings from all drag applications in Table 4.1 with the max limit (shown in Figure 4.7) unaltered, raised by 10 dB and lowered by 10 dB respectively.

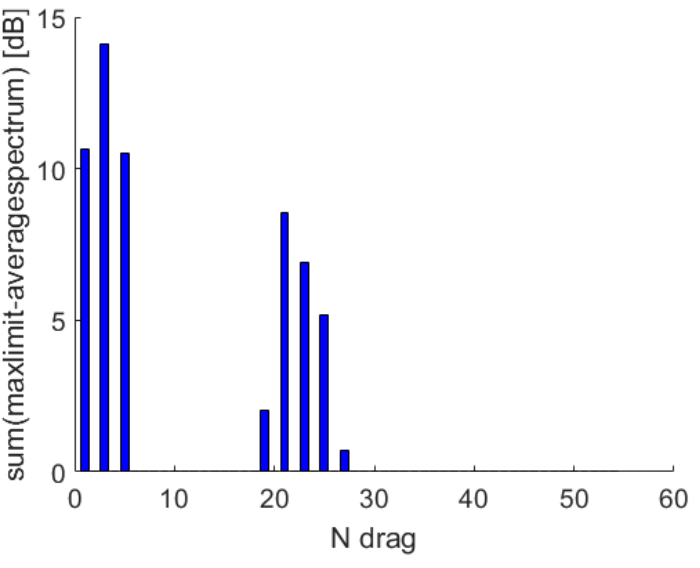

Figure 4.4: *Results from squeal detection algorithm for all 54 drag applications with 10 dB added to max limit.*

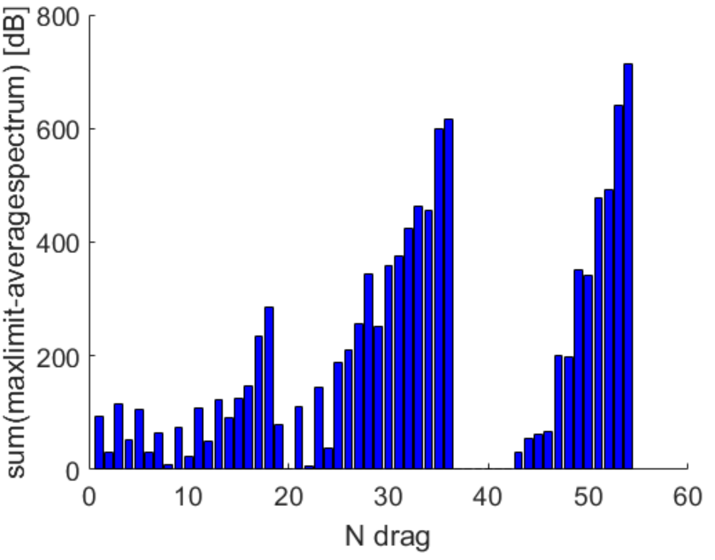

Figure 4.5: *Results from squeal detection algorithm for all 54 drag applications with 10 dB subtracted from max limit.*

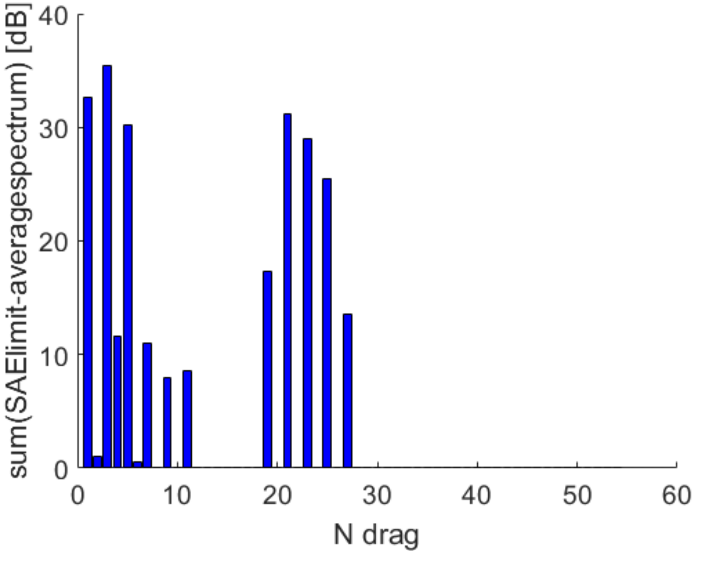

Figure 4.6: *Results from squeal detection algorithm using the regular SAE squeal definition with peaks above 70 dB between 2500 and 15000 Hz.*

Figure 4.6 shows the results of running the algorithm with the regular SAE definition of squeal as spectral peaks above 70 dB between 2500 and 15000 Hz. Se Figure 4.9 for a closer look at drag application 3 from the figure.

Figure 4.7 shows a comparison of the max limit of normal data and the average sound spectrum for drag application number 3.

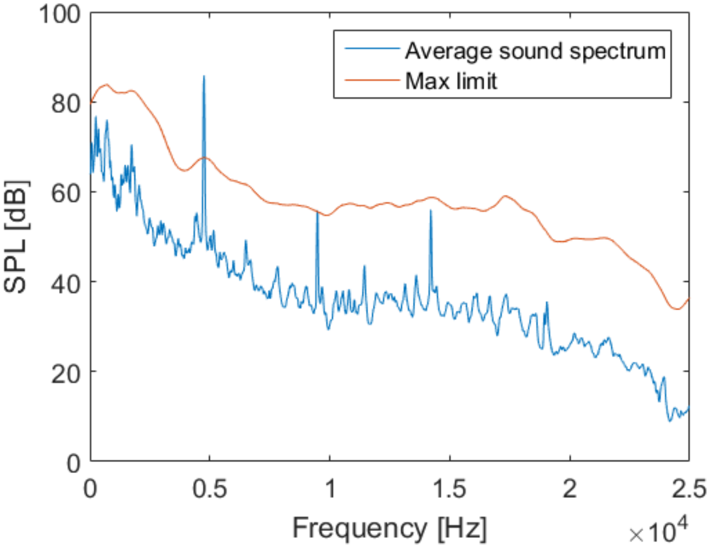

Figure 4.7: *Averaged sound spectra and max limit for drag application 3.*

Figure 4.8 shows the corresponding spectra of the accelerometer signal for drag application 3. It is presented to show the strong correlation between the spectral peaks of the sound and the accelerometer during a drag application with squeal.

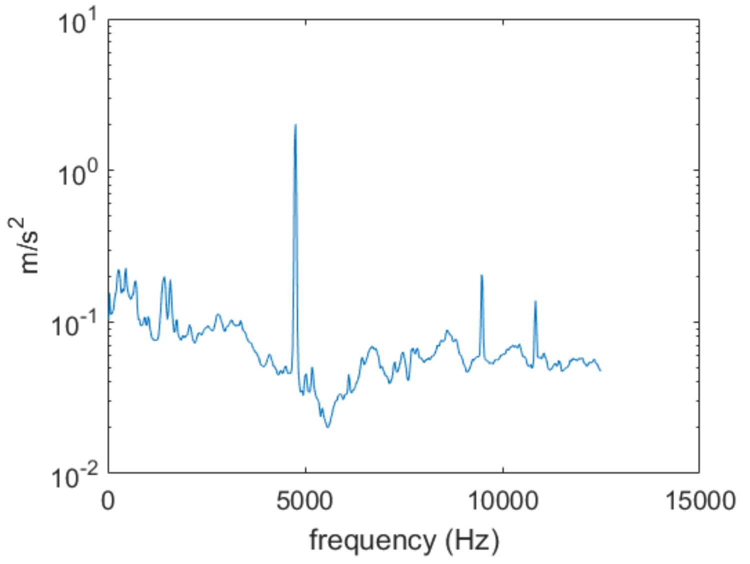

Figure 4.8: *Averaged accelerometer spectra for drag application 3.*

Figure 4.9 shows the SAE counterpart to the max limit in comparison to the average sound spectra of drag application 3.

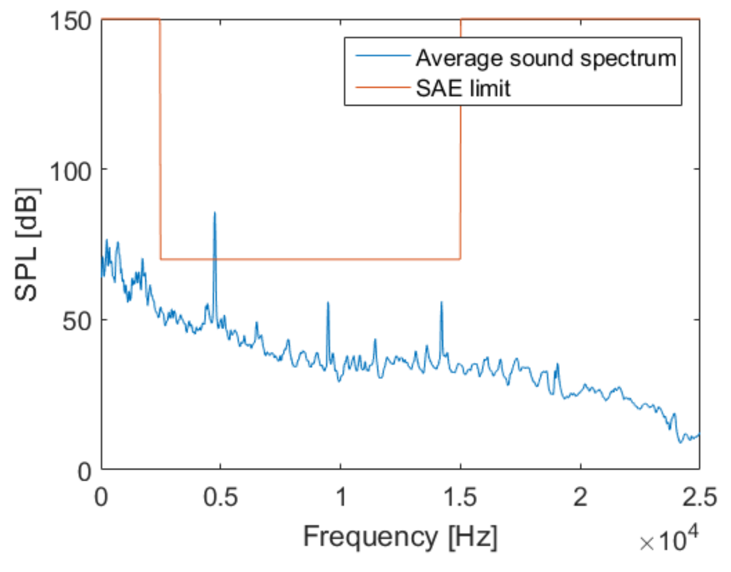

Figure 4.9: *Averaged sound spectra and SAE limit for drag application 3.*

Figure 4.10 shows the average sound spectra of drag application 8 in comparison to the max limit of normal sound. It is shown as an example of a sound recording without squeal.

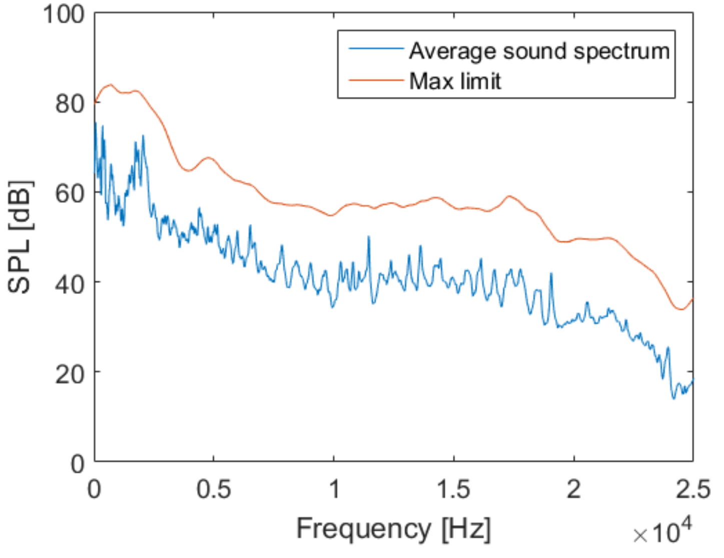

Figure 4.10: *Averaged sound spectra and max limit for drag application 8.*

4.2.1 Accelerometer Validation Results

Figure 4.11 shows the result of using the more simple accelerometer validation algorithm that compares the maximum peaks of the accelerometer and sound signals. Figure 4.12 shows the result from the same data using the coherence algorithm. Green color indicates that the measured squeal is validated by the accelerometer signal. The results are very similar, but the simpler algorithm mislabels drag application 13.

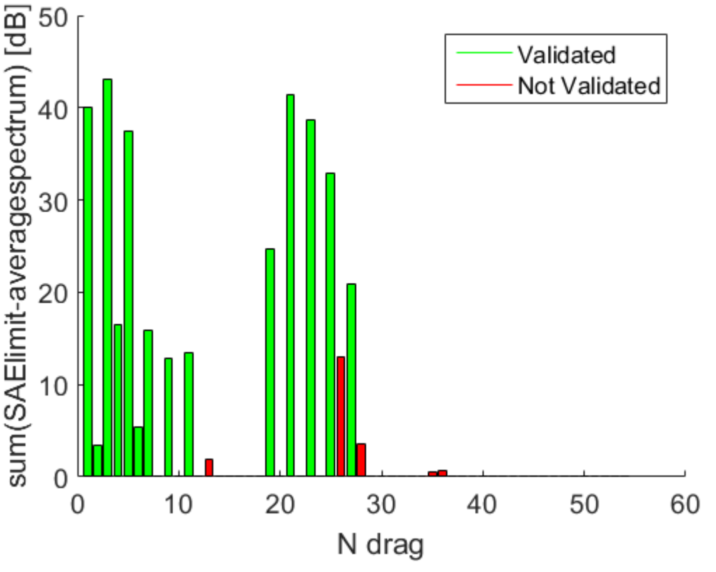

Figure 4.11: *Results of squeal detection using the simple accelerometer validation for all 54 drag applications with unaltered max limit and accelerometer validation.*

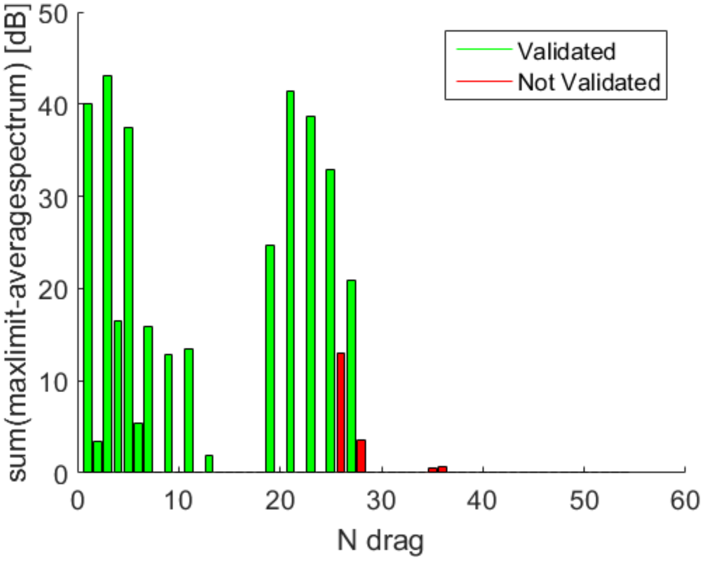

Figure 4.12: *Results of squeal detection using the coherence algorithm for all 54 drag applications with unaltered max limit and accelerometer validation.*

4.3 Squeal Detection Discussion

The results from the squeal detection algorithm agree well with the annotations. The ROC curve shown in Figure 4.2 suggests that the algorithm is best tuned when leaving the max limit unaltered. However, there is no guarantee that this would be the case for all types of tests since the limit is based on normal data and if that data would differ then so would the limit.

In Figure 4.5 the result of lowering the max limit, or threshold, by 10 dB is shown. The lowered max limit leads to a significant increase in detected squeal. This can be understood when looking at Figure 4.7. After a certain point the threshold, or limit, reaches the noise floor of the spectrum, and by then almost all recordings are labelled as containing squeal. In Figure 4.5 we can also see the result of the increasing speed of the 3x18 recordings, especially for application 19 and above. What happens is that the overall noise floor becomes slightly higher for each increase in velocity, resulting in the pairwise increase in measured noise.

When adding validation of the accelerometer as shown in Figure 4.11 and 4.12 using the methods described in Section 4.1.3 we see that the accuracy becomes even better with only one error for the simpler validation algorithm and no errors for the coherence algorithm. The price of this increase in precision is the inconvenience of mounting an accelerometer on the brake.

When looking at Figure 4.3 and 4.6 we see that the results from using the proposed limit computed from normal data is rather similar to those from using the SAE limit. This can be understood when looking at Figure 4.7 and 4.9 where the spectral peaks of higher frequenies does not really reach the limit in any of the cases. The SAE limit misses one squeal application (number 13) that the proposed method detects, and is close to missing two others (number 2 and 6). On the other hand it does not mislabel application 26, 28, 35 and 46. In other words, the proposed method is more sensitive. The best performing alternative is using the proposed method with limits based on normal data together with accelerometer validation. If this is not an alternative, the methods perform similarly and could both be used with similar outcomes. However, if the squeal appears at higher frequencies the SAE settings would prove way less sensitive, since there is no decrease in the limit for higher frequencies where the magnitudes of the spectra are always smaller.

It should also be pointed out the SAE version as used in this case is a fusion between the proposed method and the actual SAE method. The limit from the SAE method is used, but the same way of calculating the squeal measure by adding upp the difference of the spectral components that exceeds the limit is the same in both cases. In the actual SAE standard they propose some other measures of squeal, for example the magnitude of the maximum spectral peak between the defined frequency limits.

One should be careful when interpreting the meaning of the unit of the Y-axis in the plot type shown in Figure 4.3 since its a linear sum of logarithmic values.

Appendix

4.A Appendix

This is the main script in 4.1.

```
annotations= [1 1 1 1 1 1 1 0 1 0 1 0 1 0 0 0 0 0   1 0 1 0 1 0 1 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ];

% Set the directory for the data to be evaluated.
eval_dir='C:\DEWESoft\Exports\renamed_channel_data_correctnumbering';
% eval_dir='G:\RTCS\50_Medarbetare egna mappar\tlouu7\Exjobb\filer20170516\manualsear
% eval_dir='C:\DEWESoft\Exports\all_dat';
% eval_dir='C:\DEWESoft\Exports\manualsearch5_60grader';
% eval_dir='C:\DEWESoft\Exports\manualsearch6_120grader';
% eval_dir='C:\DEWESoft\Exports\manualsearch7_90grader';

% Select if you want to use accelerometer or not.
selected_validate='use accelerometer';
% selected_validate='dont use accelerometer';

% Set the cutoff frequencies for the analysis.
low_freqlim=2500;
high_freqlim=15000;
low_ind = floor( low_freqlim * 1024 / 25000 ); %convert frequencies to corresponding
high_ind =floor( high_freqlim * 1024 / 25000 );
freqlims=[low_ind high_ind];

%% Use these two lines for SAE runmode
runmode='SAE';
threshold_lim=70; %Set detection threshold for SAE testing
%% Use these two lines for prepared data
% runmode='prepared';
% norm_dir='C:\DEWESoft\Exports\renamed_normal_data';
%% Use these two lines for defining your own data
% runmode='user defined';
% norm_dir='C:\DEWESoft\Exports\renamed_normal_data'; %Provide path to directory with

%% this section reduces data for respective runmode
%Please see norm calc for more comments on what is done.
reduced_struct=norm_calc5_reduced(eval_dir,freqlims,selected_validate);
```

```
if isequal(runmode,'prepared') | isequal(runmode,'user defined')
    norm_struct_reduced=norm_calc5_reduced(norm_dir,freqlims,selected_validate);
elseif isequal(runmode,'SAE')
    Noise_threshold=150*ones(1,1024);
    Noise_threshold(low_ind:high_ind)=threshold_lim;
end
%% This section compares the reduced data from evaluate directory and normal data
% ROCcount=1;
% for j=-10:10;
reduce_db=0; %use this constant to add or subtract decibels from the max bound
for i=1:length(reduced_struct)-1

    if isequal(runmode,'prepared') | isequal(runmode,'user defined')
        reduced_struct(i).ampdiff_average_db = reduced_struct(i).Data1_MIC1_FFT_average_
        reduced_struct(i).ampdiff_raw_db = reduced_struct(i).Data1_MIC1_FFT_average_db(l
        reduced_struct(i).ampdiff_average_sum_db = sum(reduced_struct(i).ampdiff_average
        reduced_struct(i).ampdiff_raw_sum_db = sum(reduced_struct(i).ampdiff_raw_db(redu

    elseif isequal(runmode,'SAE')
        reduced_struct(i).ampdiff_average_db = reduced_struct(i).Data1_MIC1_FFT_average_
        reduced_struct(i).ampdiff_average_sum_db = sum( reduced_struct(i).ampdiff_averag
    end
end

%% This section performs plotting
close all
num=3;%specify which drag/brake application you want to compare in the compare plot
if isequal(runmode,'prepared') | isequal(runmode,'user defined')
    reference_line=norm_struct_reduced(end).maxbound_raw_filtered_dba-reduce_db;
elseif isequal(runmode,'SAE')
    reference_line=Noise_threshold;
end
compare_plot2(reduced_struct,reference_line,num)

%plot accelerometer spectra for drag/brake application specified by num
if isequal(selected_validate,'use accelerometer')
    figure
    semilogy((1:512)*12500/512,reduced_struct(num).Data1_ACCELEROMETER_FFT_average)
    ylabel('m/s^2');xlabel('frequency (Hz)');%title('Accelerometer average fft');
    set(gca,'fontsize',14)
end

%Call to plot suggested by me function
teo_plot(reduced_struct,runmode,selected_validate)

%Call to A14plot function
% A14plot(reduced_struct)

%Call to A13plot function
% A13plot(reduced_struct)
% TPcount=0;
% FNcount=0;
% TNcount=0;
% FPcount=0;
%
% for i=1:length(reduced_struct)-1
%     if annotations(i) == 1 & reduced_struct(i).ampdiff_raw_sum_db >0
%         reduced_struct(i).roc='TP';
```

```
%         TPcount=TPcount+1;
%     elseif annotations(i) == 1 & reduced_struct(i).ampdiff_raw_sum_db == 0
%         reduced_struct(i).roc='FN';
%         FNcount=FNcount+1;
%     elseif annotations(i) == 0 & reduced_struct(i).ampdiff_raw_sum_db == 0
%         reduced_struct(i).roc='TN';
%         TNcount=TNcount+1;
%     elseif annotations(i) == 0 & reduced_struct(i).ampdiff_raw_sum_db > 0
%         reduced_struct(i).roc='FP';
%         FPcount=FPcount+1;
%     end
% end
%
% TPR(ROCcount)=TPcount/(TPcount+FNcount);
% FPR(ROCcount)=FPcount/(FPcount+TNcount);
% ROCcount=ROCcount+1;
% end
% %%
% figure
% plot(FPR,TPR,'-o')
% xlabel('False Positive Rate');ylabel('True Positive Rate');%title('Receiver Operati
% set(gca,'fontsize',14)
```

This is the function referred to as reduce and validate in 4.1.

```
function [reduced_struct]=norm_calc5_reduced(dir_name,freqlims,selected_validate)
%This function loads the files in dir_name and performs some calculations
%and saves the results as a structure with reduced data.
%It overwrites the loaded file for every iteration to
%keep the memory from filling up.

low_ind = freqlims(1);
high_ind =freqlims(2);

%get files from directory in dir_name
file_dir=dir(dir_name);
addpath(dir_name);

%allocate for average and raw matrices (see bottom of funtion for
%calculations.
average_matrix=[];
raw_matrix=[];
tot_max_spect=zeros(1,1024);
tot_max_spect_db=zeros(1,1024);

%filter out to only read files with .mat extension (exclude directory names
%and other file types)
files = dir( fullfile(dir_name,'*.mat') )
nfiles=size(files);

%Loop through all files and perform calculations
for i=1:nfiles

    %This line loads a new file for every iteration and overwrites the old file
    input_struct=load(files(i).name);

    %average spectrum of the fft channels for each drag/brake application
```

```
    reduced_struct(i).Data1_MIC1_FFT_average=mean(input_struct.Data1_MIC1_FFT,1);

    input_struct.Data1_MIC1_FFT_db=20*log10(abs(input_struct.Data1_MIC1_FFT)/(20*10^-6))

    reduced_struct(i).Data1_MIC1_FFT_average_db=20*log10 (abs (reduced_struct(i).Data1_M

    %Average accelerometer signal fft.
    reduced_struct(i).Data1_ACCELEROMETER_FFT_average=mean(input_struct.Data1_ACCELEROME

    %Magnitude and frequency of max-peaks of average spectra.
    [reduced_struct(i).average_maxpeak_sound_db, reduced_struct(i).maxpeak_index_sound_d

    vib_max_index=floor(5000*1024/25000);%It is necessary to only look at vibrations bel

    [reduced_struct(i).average_maxpeak_vib, reduced_struct(i).maxpeak_index_vib ]=max(re

    %compute absolute maxpeak in dB between the lower and upper frequency limits
    reduced_struct(i).absolute_max_db=max(max(input_struct.Data1_MIC1_FFT_db(:,low_ind:h

    %convert index of maxpeak to correct frequency
    reduced_struct(i).maxpeak_index_sound_db = reduced_struct(i).maxpeak_index_sound_db
    reduced_struct(i).maxpeak_index_vib = reduced_struct(i).maxpeak_index_vib + low_ind;

    %Convert indices to actual frequencies
    reduced_struct(i).maxpeak_freq_sound = reduced_struct(i).maxpeak_index_sound_db / 10
    reduced_struct(i).maxpeak_freq_vib = reduced_struct(i).maxpeak_index_vib / 1024 * 25


    %This if statements performs accelerometer validation if chosen
    if isequal(selected_validate,'use accelerometer')

        %% First type of validation checks if maxpeak lies at same frequency
%         if reduced_struct(i).maxpeak_index_sound_db < 1.02 * reduced_struct(i).maxpeak
%             reduced_struct(i).validated=true;
%         else
%             reduced_struct(i).validated=false;
%         end
%
        %% Second type of validation computes coherence with Tri-Spectrum averaging

        %Get some necessary variable sizes
        viblen=length(reduced_struct(i).Data1_ACCELEROMETER_FFT_average);
        nSoundFFT=size(input_struct.Data1_MIC1_FFT,1);
        nVibFFT=size(input_struct.Data1_ACCELEROMETER_FFT,1);

        min_index=min(nSoundFFT,nVibFFT); %make sure we don't exceed any matrix dimensio

        %These lines below compute the coherence function
        SAB = input_struct.Data1_MIC1_FFT(1:min_index,1:viblen) .* input_struct.Data1_AC
        SBB = input_struct.Data1_MIC1_FFT(1:min_index,1:viblen) .^2;
        SAA = input_struct.Data1_ACCELEROMETER_FFT(1:min_index,:) .^2;

        SABa = mean(SAB,1);
        SAAa = mean(SAA,1);
        SBBa = mean(SBB,1);

        %Coherence
        reduced_struct(i).gamma2 = (SABa.^2) ./(SAAa .* SBBa);
```

```
        %Coherent output power in dB
        reduced_struct(i).COP =  20*log10( ( reduced_struct(i).gamma2 .* SBBa)/(20*10

        maxpeak_COP=max(reduced_struct(i).COP(low_ind:end));%get the maximum COP abov

        if maxpeak_COP >  60 %This value of 60 can be adjusted if necessary
            reduced_struct(i).validated=true;
        else
            reduced_struct(i).validated=false;
        end

    end

    %add other signals you want to save from input_struct:
    reduced_struct(i).Data1_ACCELEROMETER=input_struct.Data1_ACCELEROMETER;

    %these two lines just creates matrices for maxbound calculations
    average_matrix=[average_matrix; reduced_struct(i).Data1_MIC1_FFT_average];
    average_matrix_db=[average_matrix; reduced_struct(i).Data1_MIC1_FFT_average_db];
    raw_matrix=[raw_matrix; input_struct.Data1_MIC1_FFT];
    raw_matrix_db=[raw_matrix; input_struct.Data1_MIC1_FFT_db];


    %these lines computes the absolute max frequency components of all
    %recordings.
    current_max_spect=max(input_struct.Data1_MIC1_FFT,[],1);
    index_matrix=current_max_spect>tot_max_spect;
    tot_max_spect(current_max_spect>tot_max_spect)=current_max_spect(current_max_spec

    current_max_spect_db=max(input_struct.Data1_MIC1_FFT_db,[],1);
    index_matrix_db=current_max_spect_db>tot_max_spect_db;
    tot_max_spect_db(current_max_spect_db>tot_max_spect_db)=current_max_spect_db(curr


end
%this line finds the maximum for each frequency component among all averaged
%spectras
reduced_struct(i+1).maxbound_average=max(average_matrix,[],1)
reduced_struct(i+1).maxbound_average_db=max(average_matrix_db,[],1)
%This line finds the maximum for each frequency component among all
%spectras without averaging them
reduced_struct(i+1).maxbound_raw=tot_max_spect;
reduced_struct(i+1).maxbound_raw_db=tot_max_spect_db;


%These lines zero phase IIR filtering in the frequency domain to smooth the data.
reduced_struct(i+1).maxbound_raw_filtered=filtfilt(ones(1,30)/30,1,double((max(raw_ma
reduced_struct(i+1).maxbound_raw_filtered_dba=filtfilt(ones(1,30)/30,1,double((max(ra
```

5

Adaptive Noise Cancellation

This chapter describes adaptive noise cancellation and how it can be implemented in the brake dynamometer. When recording brake sound in the dynamometer there is usually a lot of background noise coming from the machinery and the ventilation system. All this background noise can make the brake sound hard to distinguish. A way of filtering out some of the background noise is by using adaptive noise cancellation. The implementation was done in Simulink and could with some slight modifications be used both on-line and off-line. The on-line version can be used for listening to the brake sound more clearly during operation and the off-line version can be used for cleaning recorded signals.

5.1 Noise Cancellation

This section briefly presents the theory and practical implementation. For more details on adaptive filters and noise cancellation please see [11] and [7].

The main idea in active noise cancellation is to model the background noise and subtract it from the signal. This is done by using two microphones, one for recording a mixture of interesting sound and background noise and one for recording background noise exclusively. In the case of a brake dynamometer this would correspond to one microphone close to the brake and one further away. Of course in practice the microphone placed further away will also record some brake sound, but the idea is that the brake sound will be somewhat weaker while the background noise is of the same magnitude.

Since the two microphones are located at different distances from the source of the desired sound, i.e. the brake, there will be a slight offset in time between the two signals. This delay can be estimated in different ways. For example, one could simply take the distance between the microphones and divide it by the speed of sound, or one could correlate the two signals to see the exact delay. There are benefits and drawbacks to both methods, and it is advised to use a combination of both. For example, if one of the microphones is located at 50 cm from the brake and one at 150 cm from the brake, the distance between them is 100 cm (assuming they are mounted on the same line from the brake). Sound travels at about 340 m/s at atmospheric pressure and room temperature, so it would take around 1/340 = 0.0029 seconds for it to travel the distance between the microphones. At a sample rate of 50 kHz this would correspond to around 0.0029*50000=147 sample delay between the two signals. This means that to align the signals from the two microphones in time, we would have to delay the signal from the microphone closer to the brake with around 147 samples. This will not be the only difference between the two signals, the sound will also be reflected differently because of the difference in the placement of the microphones. If there were no other differences than the time delay it would be enough to subtract the signals from each other and no adaptive filter would be needed. Since this is not the case, we have to model the signal from the noise microphone so that it better resembles the noise picked up by the microphone recording both background noise and the interesting brake sound. This will be described in the following sections, but first we need to introduce some notation. Consider

$$y_1 = s + u_1 \tag{5.1}$$

where y_1 is the signal recorded by the first microphone (close to the brake), s is the interesting sound and u_1 is the disturbing noise. The second microphone records only disturbing noise denoted u_2 (only in theory, in practice it will also record some brake sound) which is a delayed and altered version of u_1. The idea is now to filter u_2 using an FIR (Finite Impulse Response) filter so that it better resembles u_1 and then subtract it from y_1 giving the following approximation of s

$$\hat{s} = y_1 - \hat{u}_1 \tag{5.2}$$

The modelling of u_1 that will be used for this part can be written

$$\hat{u}_1(t) = \theta_1 u_2(t - n_k) + .. + \theta_{n_b} u_2(t - n_b - n_k) \tag{5.3}$$

where θ are the parameters of the FIR filter, n_k is the number of samples corresponding to the time delay and n_b is the length of the filter. To be precise, the time delay will actually be negative (or applied to y_1) since it is recorded before u_2, but this notation does the job for now.

5.1.1 Least Mean Square Algorithm

The Least Mean Square or LMS algorithm was used to compute the parameter vector θ for the FIR filter used in the noise cancellation. The LMS block filter from Simulink was used for the actual implementation of the noise cancellation system. Good descriptions of adaptive noise cancellation and the LMS algorithm can be found in [18], [11] and [7]. LMS works by minimising the expected value of the squared prediction error by recursively updating θ. The algorithm can be derived in the following way. Consider

$$\hat{\theta}^{LMS}(t) = \arg_\theta \min V_t(\theta) \tag{5.4}$$

meaning that $\hat{\theta}^{LMS}$ is the parameter vector minimising $V_t(\theta)$ given by

$$V_t(\theta) = \frac{1}{2}E((y_1(t) - \phi^T(t)\theta)^2) \tag{5.5}$$

where $y_1(t)$ is the signal we want to model. By differentiating the expression in Equation (5.5) with respect to θ we get the following expression

$$-\frac{d}{d\theta}V_t(\theta) = E(\phi(t)(y_1(t) - \phi^T(t)\theta)) \tag{5.6}$$

We now use the right hand side of Equation (5.6) without expectation as an approxmation of $-\frac{d}{d\theta}V_t(\theta)$ to update our parameter vector using the following expression

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \mu\phi(t)(y_1(t) - \phi^T(t)\hat{\theta}(t-1)) \tag{5.7}$$

Where μ is a step size usually set between 0.1-0.001. In the discussed noise case $\phi^T(t)$ will be given by

$$\phi^T(t) = (u_2(t-n_k), ..., u_2(t-n_b-n_k)) \tag{5.8}$$

5.1.2 Filter Length Estimation

This section will describe the method used to estimate the number of parameters n_b, or length, of the FIR filter giving the best results in the adaptive noise cancellation.

The purpose of the LMS algorithm in this case is to compute the parameters of an FIR filter that minimizes the error between y_1 and u_2. Therefore, to evaluate the order of the FIR filter, the RMSE or Root Mean Square of the error was calculated for different filter orders. The results can be seen in Figure 5.1. The order was set to 100, for higher orders the filter became unstable.

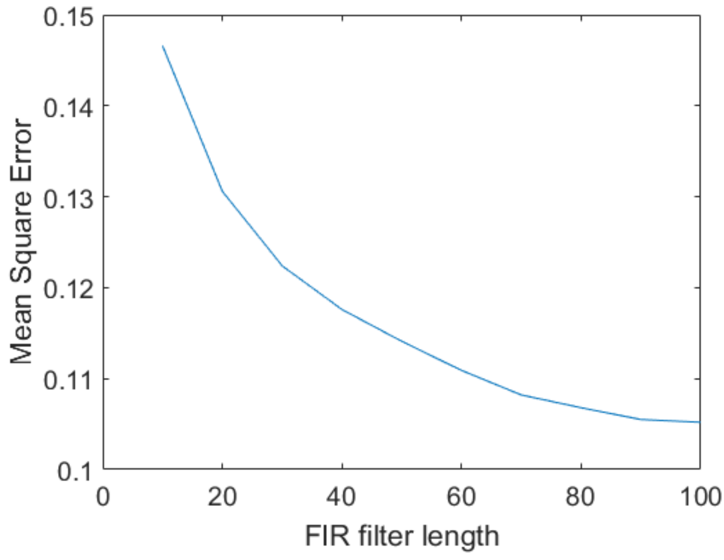

Figure 5.1: *RMS of the error for the LMS algorithm for different filter lengths.*

5.2 Noise Cancellation Results

Figure 5.2-5.4 shows some results from the attempt on implementing adaptive noise cancellation in the brake dynamometer. A step length of μ=0.001 was used for the LMS algorithm. The step length was conceived by normalizing 0.01 with the average energy of the signal computed using

$$E = \frac{1}{N}\sum_{1}^{N}|x[n]|^2 \tag{5.9}$$

Larger step lengths made the algorithm unstable and smaller steplengths led to significant amounts of time before the parameters reached somewhat steady values.

In Figure 5.2 the signals y_1 and $\hat{s}$ are shown. Seconds 23-30 contain a clearly audible squealing sound visible by the increased amplitude.

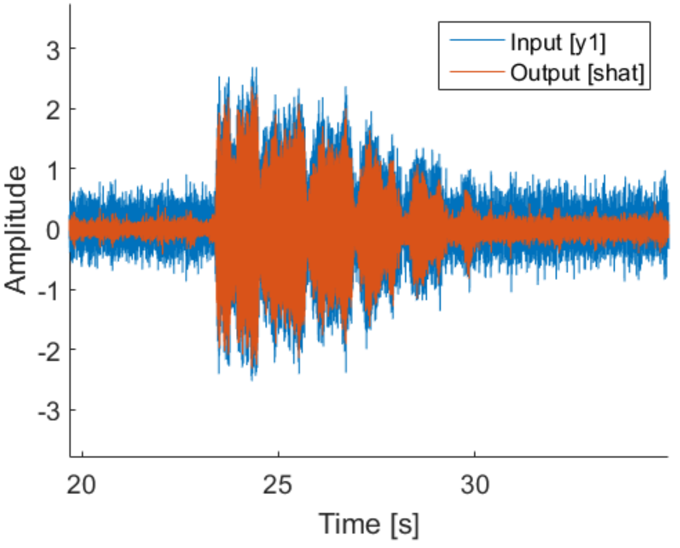

Figure 5.2: *The input (y1) and the output (shat) as function of time, with squeal from second 23 to 30.*

Figure 5.3 shows the evolution of the 100 FIR filter parameters over time. It can be observed that the parameters change drastically when the squealing sound starts.

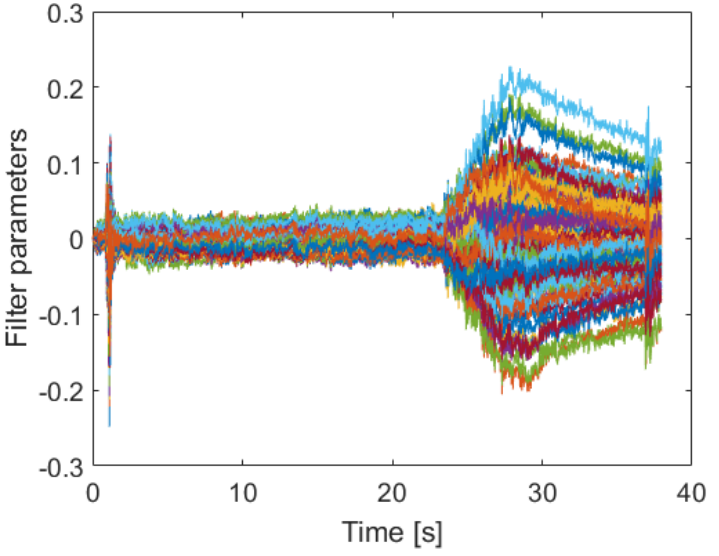

Figure 5.3: *LMS filter parameters over time using μ=0.001.*

Figure 5.4 shows the magnitude of the spectra in dB for y_1 and $\hat{s}$. It is clear that $\hat{s}$ has a highly attenuated spectrum in almost all frequencies except for the squealing frequency. The signal to noise ratio was also computed by taking the mean power of the spectrum between 4600 abd 4800 Hz and divided by the mean power of the signal up to 4600 Hz before and after the filter. The resulting signal to noise ratio was 4.26 without filter and 8.27 when using the adaptive filter.

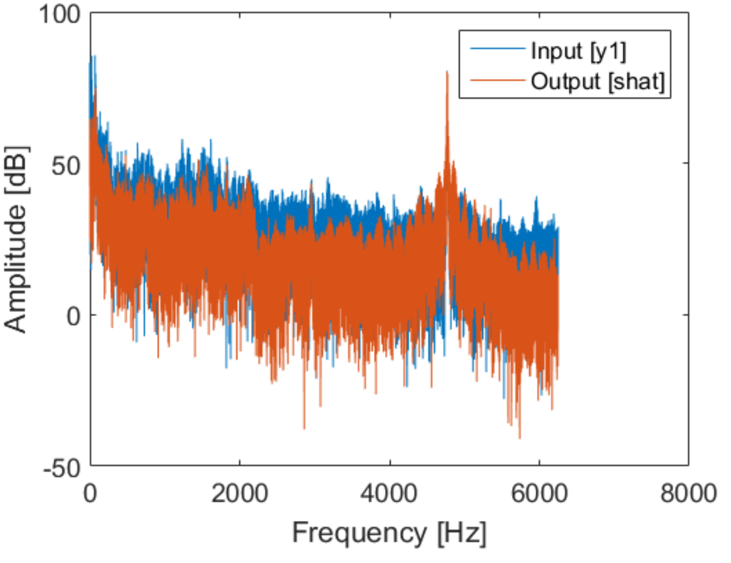

Figure 5.4: *The input (y1) and the filtered signal (shat) as function of frequency with the first squeal harmonic visible at around 4760 Hz.*

5.3 Noise Cancellation Discussion

Implementing noise cancellation can be helpful to filter out undesired noise in brake squeal testing. The major issue is that its very hard to prevent the noise microphone, or u_2, from recording s. It turns out that it works quite well anyway if we do not do the subtraction of the signals, and only regard the filtered version of y_1 as $\hat{s}$. The reason why this works so well is likely because the noise is harder to model, while the squealing sound is much easier since it is very narrow in frequency and of much higher amplitude. One interpretation is that the FIR filter removes u_1 very well while leaving s nearly intact.

6 Conclusion

This chapter will provide some conclusions from the project and results in general and some directions and suggestions for future work. For more specific discussions of the obtained results see corresponding chapter.

6.1 Results

It is definitely possible to use algorithms and software to detect and measure squealing sounds from wheel brakes.

The suggested method based on calculating limits from normal data detects all 15 squeal events out of the 54 drag applications, with 4 false detections. If accelerometer coherence validation is added all the falsely detected squeal events are removed. If the more simple accelerometer validation method is used one squeal event of 15 in total is mislabeled as non squeal. If the SAE definition of squeal is used with modified frequency limits of 2500 to 15000 Hz instead of 2000-16000 Hz, one squeal event is missed, but none of the drag applications without squeal is mislabeled. In other words, the proposed method is more sensitive than the SAE method.

To summarize the results of the proposed algorithm; the most precise method is to use the spectral limit calculated from normal data with accelerometer validation. If accelerometer validation is not an option, the SAE performs very well, although it's a bit less sensitive.

Adaptive nosie cancellation can also be implemented in the system. It almost doubles the signal to noise ratio when using a filter order of 100.

6.2 Future Work

For the future of testing brakes with respect to noise and squeal at Scania it is suggested that a more extensive and rigorous brake test matrix is developed. A good starting point for this would definitely be the SAE J2521 test matrix. The methodologies proposed in this report could then be used to record and analyse the generated data. The number of brake/drag applications from such a matrix would provide an excellent data set for developing statistics around the squeal and noise phenomenon for different brake setups and different conditions. These statistics could then be used to recommend certain setups to customers with certain predominant drive cycles, or simply to remove or change certain components or combinations of components in Scanias brake hardware.

The methodology could also be used on vehicles, although it would need some modifications. For example more accelerometers would be needed to determine from which brake the squealing sound originates.

It would also be of value for the department to make the current methodology easier to use. For example developing a more extensive user interface so that people without knowledge of Matlab can use the program. Another idea is to further automate the dewesoft system through scripting in for example Matlab or Python. A description of how this is done can be found in [4].

Bibliography

[1] Dr. Mohamed Khalid Abdelhamid, Dr. Peter Blaschke, Dr. W. Alex Wang, and Dr. Shuo Yang. An overview of brake noise and vibration problems. *inter-noise*, 2007. Cited on pages 2 and 10.

[2] Zoeller T. Abdelhamid M. Bretz U. et al. Attia, F. The psycho acoustical approach behind the brake squeal evaluation procedure boni. *W. SAE Technical Paper*, 2006. Cited on page 17.

[3] P. Blaschke and W Rumold. Global nvh matrix for brake noise - a bosch proposal. *W. SAE Technical Paper*, 1999. Cited on page 2.

[4] Dan Sundström Daniel Ahlberg. Feldiagnos av växellådor med ljud- och vibrationsanalys. Cited on page 44.

[5] Dewesoft. Dewe43 technical reference manual. . Cited on page 5.

[6] Dewesoft. Soundlevel manual. . Cited on page 7.

[7] Scott C. Douglas. *Digital Signal Processing Handbook*. CRC Press LLC, 1 edition, 1999. Cited on pages 37 and 38.

[8] Oliviero Giannini Francesco Massi, Laurent Baillet. Experimental analysis on squeal modal instability. *International Modal Analysis Conference*, 2006. Cited on page 2.

[9] Winfried Krebber Gunter Mauer. Brake squeal and moan detection: Important requirements concerning mobile recording systems. *Automobile Comfort Conference*, 2006. Cited on page 7.

[10] Fredrik Gustafsson. *Statistical Sensor Fusion*. Studentlitteratur, 2 edition, 2012. Cited on page 19.

[11] Fredrik Gustafsson, Lennart Ljung, and Mille Millnert. *Signal Processing*. Studentlitteratur AB, 1:3 edition, 2010. Cited on pages 8, 13, 14, 37, and 38.

[12] M. Haverkamp. Brake noise assessment by means of vehicle road tests subjective parameters and objective measurement concepts. *W. SAE Technical Paper*, 2005. Cited on page 10.

[13] B.P. Lathi. *Linear Systems and Signals*. Oxford university press, 2 edition, 2005. Cited on page 14.

[14] S. N. Y. Gerges M. Triches Jr and R. Jordan. Reduction of squeal noise from disc brake systems using constrained layer damping. (3):348, 2004. Cited on page 2.

[15] Gunter Mauer and Michael Haverkamp. Measurement and assessment of noise caused by vehicle brake systems. *inter-noise*, 2007. Cited on pages 2 and 20.

[16] David Powers. Evaluation: From precision, recall and f-measure to roc, informedness, markedness correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011. Cited on page 19.

[17] SAE. Disc and drum brake dynamometer squeal noise test procedure j2521. *W. SAE Technical Paper*, 2013. Cited on pages 1, 2, 7, and 17.

[18] Aarti Singh. Adaptive noise cancellation. *Netaji Subhas Institute of Technology*, 2001. Cited on page 38.

[19] STAC. Eventscan brake. *Internal Scania Document*. Cited on page 1.

[20] Andre Garcia Lima Suetti. Objective noise index for brake tests using analytical equations. *W. SAE Technical Paper*, 2013. Cited on page 17.

[21] Per Åberg. Bd4 calibration report. *Internal Scania document*. Cited on page 7.