

CHALMERS



Operational Transfer Path Analysis

A Study of Source Contribution Predictions at Low Frequency

Master's Thesis in the Master's programme in Sound and Vibration

MIHKEL TOOME

Department of Civil and Environmental Engineering

Division of Applied Acoustics

Vibroacoustics Group

CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2012

Master's Thesis 2012:141

MASTER'S THESIS 2012:141

Operational Transfer Path Analysis:
A Study of Source Contribution Predictions
at Low Frequency

Mihkel Toome

Department of Civil and Environmental Engineering
Division of Applied Acoustics
Vibroacoustics Group
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2012

Operational Transfer Path Analysis:
A Study of Source Contribution Predictions at Low Frequency

© Mihkel Toome, 2012

Master's Thesis 2012:141

Department of Civil and Environmental Engineering
Division of Applied Acoustics
Vibroacoustics Group
Chalmers University of Technology
SE-41296 Göteborg
Sweden

Tel. +46 (0)31 772 1000

Reproservice / Department of Civil and Environmental Engineering
Göteborg, Sweden 2012

Operational Transfer Path Analysis:
A Study of Source Contribution Predictions at Low Frequency
Master's Thesis in the Master's Programme Sound and Vibration
MIHKEL TOOME
Department of Civil and Environmental Engineering
Division of Applied Acoustics
Vibroacoustics Group
Chalmers University of Technology

Abstract

The operational transfer path analysis (OTPA) method is a variation of the classical transfer path analysis (TPA) method, both of which are used particularly for noise, vibration and harshness (NVH) testing in the vehicle industry. The methods differ such that classical TPA requires a physical isolation of the critical source-receiver paths to eliminate cross-talk prior to determining the frequency response functions (FRFs), whereas OTPA uses singular value decomposition (SVD) and principal component analysis (PCA) for cross-talk cancellation (CTC), prior to computing the transmissibility of each critical path.

In some cases, it has been found that the OTPA results yield an over-prediction of the airborne sound contribution in the low frequency range (20 to 100 Hz). This over-prediction seems to occur when several microphones are placed within close proximity of each other (i.e. within a fraction of the wavelength).

This thesis begins with a study of the underlying theory for the classical TPA and the OTPA methods. The limitations and potential sources of error for each method are discussed. The OTPA method is further studied by way of an implementation using MATLAB. The OTPA script is tested and validated using a *virtual test setup* (VTS), consisting of two sources and one receiver.

Next, the VTS is used to study several simplified OTPA scenarios. Subsequently, the over-prediction of the airborne sound contribution is recreated using the VTS and OTPA scripts. The reasons for the over-prediction of the airborne sound contribution are identified and discussed.

Keywords: Operational Transfer Path Analysis, OTPA, Cross-Talk Cancellation, CTC, Low Frequency, Source Contribution,

Acknowledgements

This master's thesis topic was proposed by Müller-BBM Scandinavia AB, and the work has been carried out with guidance from engineers from both Müller-BBM Scandinavia AB and Müller-BBM VibroAkustik Systeme GmbH.

First and foremost, I would like to thank my advisor at Müller-BBM Scandinavia, Dr. Juha Plunt, who's technical expertise, experience and guidance have been invaluable in completing this thesis work. Also, thank-you to Martin Lohrmann at Müller-BBM VibroAkustik Systeme for taking the time to meet with me to discuss OSPA and for his suggestions on what aspects of the method would be interesting to study.

This thesis project is the culmination of two years of study at the Division of Applied Acoustics at Chalmers University of Technology. Thank-you to all my colleagues and to all the staff for the informative lectures, labs, exercises, projects and discussions. In particular, thank-you to my thesis supervisor, Professor Wolfgang Kropp, who's insight, guidance and encouragement have helped make this thesis project a rewarding experience.

Most importantly, thank-you to my wife, Heili, for her constant love and support, especially during these past two years.

Mihkel Toome
Göteborg, June 2012

Contents

Abstract	iii
Acknowledgements	v
Contents	vi
List of Figures	ix
List of Abbreviations	xi
1. Introduction	1
1.1. Case Study: Airborne Sound Over-Prediction Using OTPA	1
1.1.1. OTPA Measurement Setup	2
1.1.2. OTPA Contribution Analysis Results	3
1.1.3. Case Study Conclusions	4
1.2. Thesis Structure	5
1.2.1. Hypotheses	5
1.2.2. Overview	6
2. Basic Principles of (Operational) Transfer Path Analysis	7
2.1. The Classical Transfer Path Analysis (TPA) Method	7
2.1.1. The Frequency Response Function (FRF)	7
2.1.2. Determining the Operational Excitation Force Function	9
2.1.3. Limitations of The Classical TPA Method	10
2.2. The Operational Transfer Path Analysis (OTPA) Method	11
2.2.1. The Operational Transfer Function	11
2.2.2. The Cross-Talk Cancellation Method	14
2.2.3. Response Synthesis	17
2.2.4. OTPA Limitations and Potential Sources of Error	19
3. Implementation of OTPA and CTC	21
3.1. MATLAB Script: Description of Key Functions	23
3.1.1. Singular Value Decomposition	23
3.1.2. Principal Component Analysis and Cross-Talk Cancellation	23
3.1.3. Response Synthesis and Contribution Analysis	24

4. The Virtual Test Setup	25
4.1. Virtual Test Setup Layout	25
4.2. Source Signals - Background Theory	25
4.3. VTS and OTPA Parameters	27
4.3.1. VTS Paramaters	27
4.3.2. OTPA Parameters	28
5. Analysis	29
5.1. Analysis Outline	29
5.2. OTPA Validation: CTC and Noise Removal Using PCA	31
5.3. OTPA of Two Airborne Noise Sources	33
5.3.1. Varying Distance Between Two Correlated Sources	34
5.3.2. Varying Distance Between Two Uncorrelated Sources	38
5.3.3. Discussion and Summary of Results	42
5.4. OTPA of Correlated Structure-Borne and Airborne Noise Sources	43
5.4.1. Effects of Varying Distance of Additional Airborne Source MPs	43
5.4.2. Varying The Number of Reference MPs	46
5.4.3. Discussion and Summary of Results	49
6. Summary and Conclusions	51
6.1. Suggestions for Future Work	52
References	53
A. MATLAB Script	55
A.1. MATLAB Script - Two Airborne Noise Sources	55
A.1.1. OTPA Script (<i>otpa.m</i>)	55
A.1.2. Function to Define The Virtual Test Setup Layout (<i>dist.m</i>)	58
A.1.3. Function to Define The Source Signal (<i>source.m</i>)	59
A.2. MATLAB Script - Airborne and Structure-Borne Sources	61
A.2.1. OTPA Script (<i>otpa_sbs.m</i>)	61
A.2.2. Function to Define The Virtual Test Setup Layout (<i>dist_sbs.m</i>)	65
A.2.3. Function to Define The Source Signal (<i>source_sbs.m</i>)	66
A.3. Function to Plot Sound Field (<i>contourmesh.m</i>)	68
A.4. MATLAB Script - Airborne and Structure-Borne Sources, Varied Number of Reference MPs	69
A.4.1. OTPA Script (<i>otpa_sbs_num.m</i>)	69
A.4.2. Function to Define The Virtual Test Setup Layout (<i>dist_sbs_num.m</i>)	72
A.4.3. Function to Define The Source Signal (<i>source_sbs_num.m</i>)	73
A.4.4. Function to Plot Sound Field (<i>contourmesh_num.m</i>)	74

List of Figures

1.1. OTPA microphone positions	2
1.2. Response contribution analysis (12 microphone positions)	3
1.3. Response contribution analysis (6 microphone positions)	4
3.1. OTPA Program Flow Chart	22
4.1. Virtual Test Setup Schematic	26
5.1. VTS layout (4 reference MPs)	31
5.2. Response synthesis and principal component contribution	32
5.3. Sound field – correlated sources ($Q_A = Q_B, f_A = f_B$)	34
5.4. Source contribution – correlated sources ($Q_A = Q_B, f_A = f_B$)	35
5.5. Sound field – correlated sources ($Q_A = 2Q_B, f_A = f_B$)	36
5.6. Source contribution – correlated sources ($Q_A = 2Q_B, f_A = f_B$)	37
5.7. Sound field at f_A – uncorrelated sources ($f_A \neq f_B$)	38
5.8. Sound field at f_B – uncorrelated sources ($f_A \neq f_B$)	39
5.9. Source contribution at f_A – uncorrelated sources ($f_A \neq f_B$)	40
5.10. Source contribution at f_B – uncorrelated sources ($f_A \neq f_B$)	41
5.11. VTS layout – varying additional ABS MP distance	44
5.12. Source contribution – varying additional ABS MP distance	45
5.13. VTS layout – varying number of ABS reference MPs	46
5.14. Source contribution – varying number of ABS reference MPs ($L_{SBS} = L_{ABS}$)	47
5.15. Source contribution – varying number of ABS reference MPs ($L_{SBS} =$ $2L_{ABS}$)	48

List of Abbreviations

ABS	Airborne Sound
CTC	Cross-Talk Cancellation
DFT	Discrete Fourier Transform
DOF(s)	Degree(s) of Freedom
FRF	Frequency Response Function
MBBMS	Müller-BBM Scandinavia
MBBM-VAS	Müller-BBM VibroAkustik Systeme
MIMO	Multiple Input/Multiple Output
MP(s)	Measurement Point(s)
NTF	Noise Transfer Function
OTPA	Operational Transfer Path Analysis
PCA	Principal Component Analysis
PC(s)	Principal Component(s)
SBS	Structure-Borne Sound
SVD	Singular Value Decomposition
TPA	Transfer Path Analysis
VTF	Vibration Transfer Function
VTS	Virtual Test Setup

1. Introduction

Operational transfer path (OTPA) analysis was introduced circa the mid-2000's as an alternative to the classical transfer path analysis (TPA) method, which has been used for NVH analysis in the automotive industry since the 1970's. The main advantage of the OTPA method over the classical TPA method is that measurement of the frequency response functions (FRFs) of the physically isolated critical paths is not necessary – therefore, a considerable amount of measurement and analysis time is saved by using the OTPA method.

The fundamental difference between the TPA and OTPA is that the TPA method determines a force-response transfer path relationship (an FRF), whereas the OTPA method determines a response-response transfer path relationship (a transmissibility). Both methods have their advantages and disadvantages – these, along with the underlying theory behind both analysis methods, are discussed in Chapter 2.

The focus of this master's thesis is on a phenomenon that has recently transpired in the OTPA measurement results for several analysis cases¹: The OTPA method seems to over-predict the airborne sound contribution when several microphones are used in close proximity². Furthermore, this over-prediction of airborne sound seems to lead to an under-prediction of structure-borne sound in the same frequency range – the energy in that frequency range seems to be erroneously shifted from the structure-borne sound contribution to the airborne sound contribution.

The objective of this thesis work is to study the OTPA method and to determine what is causing the erroneous source contribution prediction in the low frequency range.

1.1. Case Study: Airborne Sound Over-Prediction Using OTPA

In the following, a Müller-BBM Scandinavia AB (MBBMS) project for which the OTPA results include the over-prediction of airborne sound (and under-prediction of structure-borne sound) is described.

¹The phenomenon has been noticed for both stationary and vehicle pass-by OTPA results on Müller-BBM Scandinavia AB projects.

²In this report, references to distances, such as "close proximity", are always relative to the wavelength of the low frequency airborne sound.

This particular project involved the OTPA of a trailer-less road tractor¹ during an idling operating condition.

1.1.1. OTPA Measurement Setup

Accelerometers and microphones were placed such that all of the critical sources and paths were included in the analysis. The microphone positions are indicated in Figure 1.1, which is a schematic of the tractor as seen from above. The microphones were mounted to be approximately 30-40 cm from the surface of the source, and measurements were taken outdoors (i.e. the free field can be assumed).

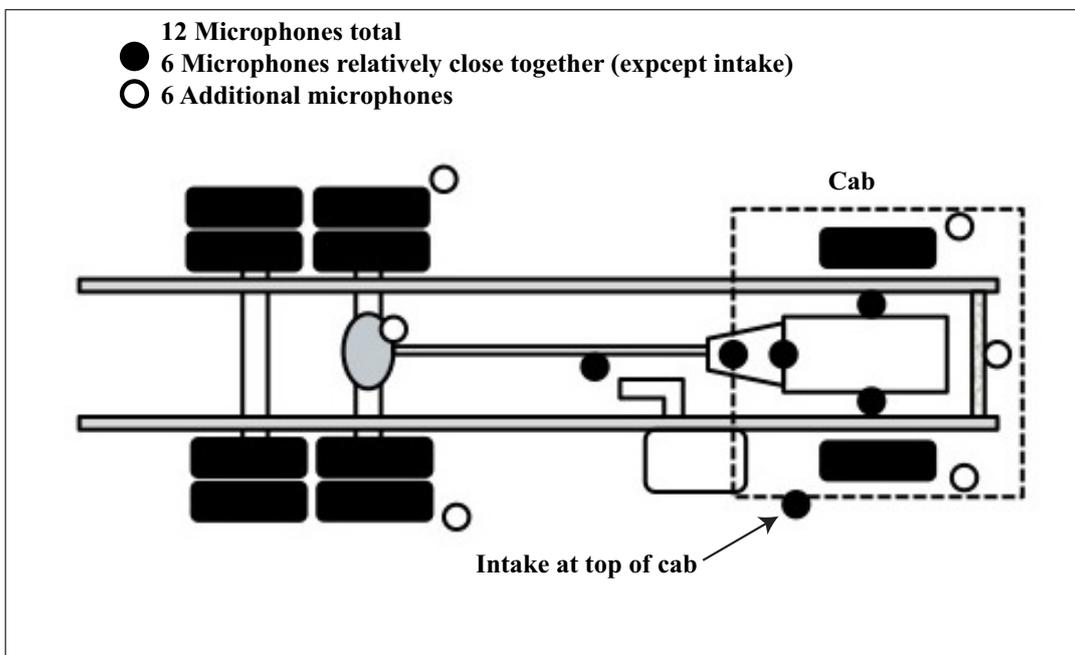


Figure 1.1.: OTPA microphone positions

To compute the transfer functions, measurements of several operating conditions were included in the analysis to ensure excitation of all relevant modes. The operating conditions used for the transfer function computation were: neutral run-up, neutral run-down and idle. The tractor was stationary for all measurements (i.e. no wind noise or tire rolling noise).

The measurement data was collected and analyzed using the MBBM-VAS measurement interface PAK[®] [15].

¹a.k.a. lorry, transport truck, rig.

microphones included in the analysis. Referring to Figure 1.3b, note that the airborne and structure-borne contributions are approximately equal at 25 Hz. The total response at the receiver remains unchanged from the previous analysis case.

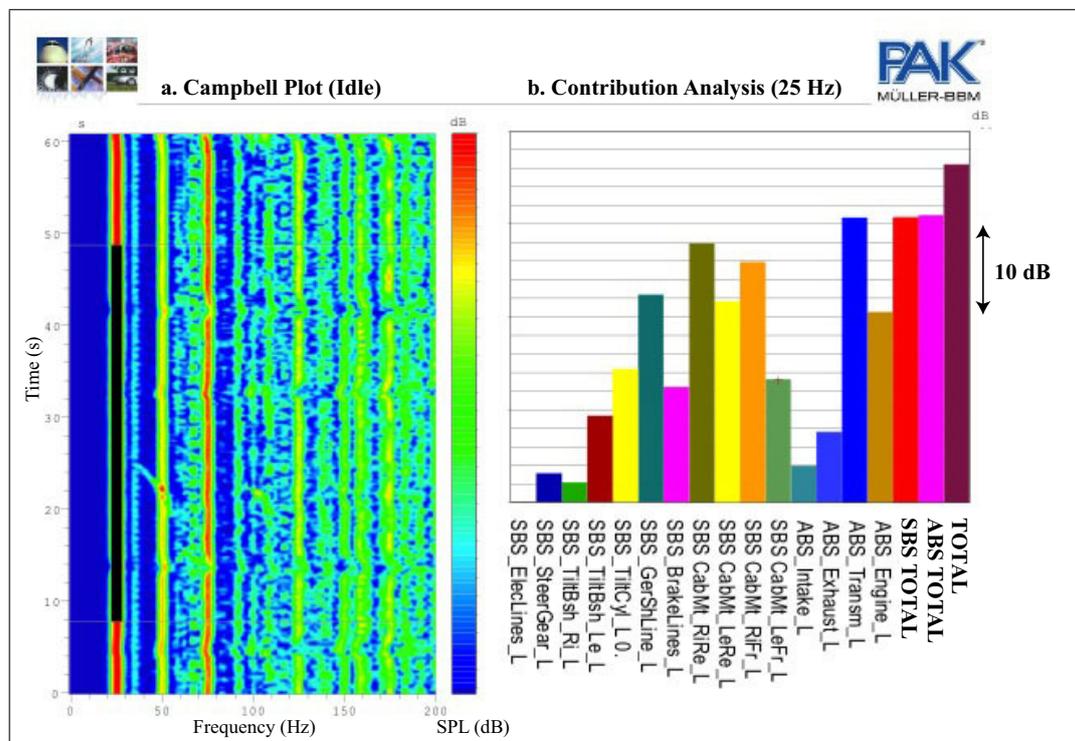


Figure 1.3.: Response contribution analysis (6 microphone positions)

By excluding the six additional microphone positions from the OTPA, the predicted total airborne sound contribution dropped by approximately 5 dB, while the predicted total structure-borne sound contribution increased by approximately 15 dB. Further, most of the individual structure-borne source contributions also increased significantly for the latter case.

1.1.3. Case Study Conclusions

The results of these two cases lead to very different conclusions – For the first case (12 microphones positions), the contribution analysis results indicate that the airborne sound of the transmission is the critical source. For the second case (6 microphone positions), the contribution analysis indicates that both the structure-borne (the cab mounts) and airborne (transmission) sources are significant.

Past experience by MBBMS engineers lead to a partiality towards the analysis results which excluded the six additional microphone positions. Consequently, modifications

were made to the tractor in order to reduce the structure-borne sound contribution. The modifications were successful in reducing the overall sound pressure level at the receiver, thus confirming that the contribution analysis results with the six additional microphone positions excluded are more accurate than the results using all twelve microphone positions.

Comparing the results of the two OTPA cases, refer to Figures 1.2 and 1.3, observe that although the total airborne and structure-borne contributions change, the total sound pressure level remains the same. This indicates that by excluding the six additional microphones, the acoustic energy is shifted from the airborne sound to the structure-borne sound in the contribution calculation.

1.2. Thesis Structure

This thesis is based on an investigation of why the erroneous source contribution described in the tractor OTPA case study has occurred. In the following, the hypotheses are presented, and the thesis structure is outlined.

1.2.1. Hypotheses

The main hypotheses drawn from the results of the tractor OTPA case study are based on the observation that the six additional microphones are not measuring a significant source, therefore, it is possible that cross-talk from the other nearby significant contributors creates a bias towards the airborne contribution in the results.

The bias towards airborne sound contribution occurs at low frequencies – in this case study at 25 Hz, however it can be generalized to the range of 20 to 100 Hz. This could be for one or more of the following reasons:

- The structure-borne sound contribution typically occurs in this low frequency range – It is possible that the energy is shifted from structure-borne to airborne sound, since both could include acoustic energy and may also be highly correlated in this frequency range.
- All of the microphone positions are within a fraction of the wavelength of the source (e.g. for dry air at 20 °C the wavelength of a 25 Hz tone is 13.7 m). It is possible that cross-talk cancellation between the reference microphone positions is not effective since the sound pressure and phase is more or less the same at all microphone positions in the low frequency range.

- The reference microphones are positioned close to the sources – It is possible that the results are influenced because the measurements are taken in the near field of the low frequency range.

1.2.2. Overview

The outline of this thesis is described as follows:

Chapter 2 begins with an explanation of the theory behind both the classical TPA method and OTPA method, including advantages and disadvantages for both. Also included in this chapter is an explanation of the cross-talk cancellation method as well as reconstructing the signal as the response synthesis and for contribution analysis.

In Chapter 3, an implementation of the OTPA theory is discussed. The implementation is conducted using MATLAB[®], with the relevant script included in Appendix A.

To validate the OTPA implementation, and to investigate the hypotheses proposed in Section 1.2.1, a virtual test setup (VTS) consisting of two sources and one receiver is developed using MATLAB. Various VTS configurations are setup in order to study the OTPA method, and to attempt to recreate the airborne sound over-prediction. The VTS implementation and various analyses are discussed in Chapters 4 and 5. An example of the relevant MATLAB script is included in Appendix A.

The conclusions drawn from the OTPA study are summarized in Chapter 6. The conclusions include suggestions for practical rules of thumb for setting up an OTPA measurement.

2. Basic Principles of (Operational) Transfer Path Analysis

2.1. The Classical Transfer Path Analysis (TPA) Method

The transfer path analysis (TPA) method is used particularly in the automotive industry to for the analysis of the different contributions of noise and/or vibration at a particular receiver position (e.g. the driver and passenger positions in a vehicle). The method aims to identify propagation paths from the various airborne and structure-borne noise sources (e.g. engine, gearbox, intake, exhaust, tire noise) by using a load-response linear relationship model.

2.1.1. The Frequency Response Function (FRF)

The classical TPA method identifies the frequency response functions (FRF) of the various source-receiver paths by using a known artificial excitation, for example, using a shaker as a vibration source or using a loudspeaker as an airborne source, and measuring the response at the receiver. The resulting FRFs are uniquely determined by the properties of the linear multiple input/multiple output (MIMO) system that relates the response to the input excitation force (structure-borne) and/or volume velocity (airborne) [1]. Note that the classical TPA method can also be applied reciprocally to determine the FRFs - Meaning that the excitation is applied at the receiver position and the response is measured at the source position [20].

In some cases, the use of a shaker/loudspeaker is not possible, and the operational force is used to determine the FRF. However, in this case, to accurately determine the transfer function, the input force must also be accurately determined by measurement and/or calculation [8, 11].

Once the FRF has been determined, the predicted response at the receiver is calculated by multiplying the operational force with the measured FRF. Assuming a linear relationship between excitation point and response point, the response at the receiver (sound pressure level or vibration) is determined by:

$$\mathbf{y}(j\omega) = \mathbf{H}(j\omega)\mathbf{x}(j\omega) \quad (2.1)$$

Where the response (or output), $\mathbf{y}(j\omega)$, is related to the input, $\mathbf{x}(j\omega)$, by the transfer function (FRF), $\mathbf{H}(j\omega)$. The dependency on phase and frequency is indicated by $(j\omega)$. Note that in the classical TPA, the transfer function is the FRF between input and output, although sometimes this may also be referred to as the noise transfer function (NTF) and/or the vibration transfer function (VTF).

In order to calculate the total response at the receiver, all source-path-receiver configurations must be included in the analysis. This requires that all potential noise sources and related paths must be measured (e.g. engine, transmission, exhaust and intake, tires, etc). The total response¹ (i.e. sound pressure and/or vibration) at the receiver is a superposition of all the individual contributions and is obtained by summing the contribution of the product of all input excitations and transfer paths, as is described by Equation 2.2.

$$y_m(j\omega) = \sum_{n=1}^N H_{mn}(j\omega)x_n(j\omega) \quad (2.2)$$

Where:

$y_m(j\omega)$ is the response at the receiver located at point m ;

$H_{mn}(j\omega)$ is the FRF between point m and the input excitation applied to path n ;

$x_n(j\omega)$ is the input excitation force function applied to path n

In general, the input is a force (F_n) or volume velocity (Q_j), and the response is a sound pressure (p_m) or vibration (\ddot{x}_m , for acceleration). If both structural and acoustic input loads are considered, the response at the receiver is expressed as:

$$p_m(\omega) = \sum_{n=1}^k H_n(\omega)F_n(\omega) + \sum_{j=1}^r H_j(\omega)Q_j(\omega) \quad (2.3)$$

Where:

$p_m(\omega)$ is the response (sound pressure) at the receiver point m ;

F_n is the force input, or structural load, at point n (source of structure borne noise);

Q_j is the volume velocity input, or acoustic load, at point j (source of airborne noise);

H_n and H_j are the corresponding FRFs.

Note that in Equation 2.3 the airborne and structure borne contributions are treated similarly. For the classical TPA method, the measured response is typically sound or

¹Note that the contribution of wind noise is usually not included in the TPA model since the excitation force/noise is distributed over an area (over the entire exterior area of the vehicle).

vibration, and the input is a force. For the sake of simplicity, the following discussion on the TPA theory will focus on the structure-borne case.

2.1.2. Determining the Operational Excitation Force Function

The force input could be easily measured directly using a force transducer when the source is a shaker. However, for operational cases, the use of direct force measurements are prohibitive due to engine mounts or vibration isolators for which the local stiffness can be significantly affected by the transducer. Further, the three translational DOFs are usually considered, meaning that the force transducer would need to be able to measure all three directions simultaneously. To get around the difficulties, indirect methods are used to determine the operational forces.

There are three commonly used methods to determine the force input:

The first and most common method, called the *Complex Stiffness Method*, is used when the source (e.g. the engine) is mounted to the structure via a resilient connection - The forces are obtained by measuring the difference in displacement between the source and receiver sides of resilient connecting elements (with known dynamic transfer stiffness), refer to Equation 2.4.

$$F_n(\omega) = k_n(\omega)[x_{n1}(\omega) - x_{n2}(\omega)] \quad (2.4)$$

Where:

$F_n(\omega)$ is the operational force transmitted to path n ;

$k_n(\omega)$ is the complex transfer stiffness for the resilient mount for path n ;

$x_{n1}(\omega)$ is the displacement for path n on the source side;

$x_{n2}(\omega)$ is the displacement for path n on the receiver side;

The second method, called the *Matrix Inversion Method*, is used when the mountings at the connecting points are rigid leading to zero or very small relative displacements between the source and receiver sides, and therefore difficulties in measuring the displacements. Inaccurate displacement measurements would in turn lead to inaccuracy in the determination of the force. Using this method, the forces are determined as shown in Equation 2.5.

$$\begin{bmatrix} F_1 \\ \vdots \\ F_N \end{bmatrix} = \begin{bmatrix} H_{11} & \dots & H_{1N} \\ \vdots & \ddots & \vdots \\ H_{M1} & \dots & H_{MN} \end{bmatrix}^{-1} \begin{bmatrix} \ddot{x}_1 \\ \vdots \\ \ddot{x}_M \end{bmatrix} \quad (2.5)$$

Here, the number of outputs (i.e. response measurement points), M , must be equal to or larger than the number of force DOFs, N , in order to avoid numerical problems due

to the ill-conditioned inverse matrix calculation. A redundancy of response measurements by a factor of approximately two is usually used and has been found to improve the results. Further, measurement noise must be treated carefully as it may cause erroneous results upon the inverse matrix calculation. Note that the dependency on frequency, ω , has been omitted for the sake of clarity, however the equation is carried out separately for each frequency component [11, 17, 18].

The third method is a simplified version of the *Matrix Inversion Method*. The force for each DOF is simply estimated by multiplying the measured point transfer function with the measured response at the receiver, refer to Equation 2.6.

$$F_i = H_{ij}\ddot{x}_j \quad (2.6)$$

Where:

F_i is the estimated force input at point i ;

H_{ij} is the transfer function of the force at point i to the response at point j ;

\ddot{x}_j is the measured response (acceleration) at point j .

Although the method may seem attractive due to its simplicity, a high error may occur in the estimation of the force, particularly at lower frequencies, due to the cross-talk contribution from forces acting at other DOFs which are quite large [18].

The response of the system can also be determined by calculating the transfer function by a combination of numerical modelling (e.g. the finite element method) and measurement data, however this is not as widely used as the described methods [11].

2.1.3. Limitations of The Classical TPA Method

The classical TPA method uses techniques that are limited to measuring the transfer path of one sub-system at a time; therefore, each transfer path must be isolated in order to eliminate flanking paths. This is typically done by disassembling the system in order to make the measurements. The major disadvantages of having to disassemble the system are that the measurement process is time consuming, the boundary conditions of the sub-systems are changed and, in almost all cases, the vehicle can not be fully operational during the measurements [8, 14, 18].

These disadvantages have led to a modified TPA method, called operational transfer path analysis (OTPA). With the OTPA method, it is possible to identify transfer paths without disassembling the system, makes use of the operational forces, and measure several sub-systems simultaneously. The theory behind the OTPA is discussed in Section 2.2.

2.2. The Operational Transfer Path Analysis (OTPA) Method

A variation of the classical TPA method has been developed to get around the disadvantage of having to individually measure the FRFs - A time consuming and sometimes awkward process. Further, to determine the transfer function using the classical TPA method, the measured FRFs are multiplied by the estimated excitation force and/or source strength (i.e. volume velocity), which is also error-prone [23].

The Operational Transfer Path Analysis (OTPA) method was developed by Honda Research and Development, and was first described in the mid-2000's [16]. The OTPA method only requires measurement data of the operating vehicle in order to perform the analysis. The key difference between TPA and OTPA is that OTPA is based on a response-response relationship model, whereas classical TPA is based on a load-response relationship. This implies that the OTPA method does not involve the calculation or use of FRFs – OTPA uses measured transmissibilities to characterize the operational transfer functions.

This section describes how OTPA uses a technique called Singular Value Decomposition (SVD) to solve for and linearize the transfer functions between a chosen source and receiver such that the source/path contributions are independent uncorrelated (orthogonal) quantities. Subsequently, a signal processing technique called Principal Component Analysis (PCA) is used to discard measurement noise and cross-talk from the signals. The result of the SVD and PCA are measurement noise removed cross-talk cancelled excitation signals. From this, the transfer paths and source contributions can be accurately determined.

The resulting linearized uncorrelated transfer functions are used to determine the contribution of the identified sources to the total response at the receiver. Note that since the operational force is unknown, the calculated transfer function is not an FRF. The transfer function is analogous to a transmissibility function [5, 11, 10, 16].

2.2.1. The Operational Transfer Function

The OTPA system model is similar to the classical TPA method, and can be described as:

$$\mathbf{Y}(j\omega) = \mathbf{X}(j\omega)\mathbf{H}(j\omega) \quad (2.7)$$

Where:

$\mathbf{Y}(j\omega)$ is the vector of response (output) measurements at the receivers;

$\mathbf{X}(j\omega)$ is the vector of reference (input) measurements at the sources;

$\mathbf{H}(j\omega)$ is the operational transfer function matrix;

$(j\omega)$ denotes the dependency on phase and frequency;

Typically for NVH analysis the reference and response measurements consist of vibration (acceleration, velocity or displacement), forces and sound pressures. If these quantities are defined as $\mathbf{u}(j\omega)$, $\mathbf{f}(j\omega)$ and $\mathbf{p}(j\omega)$ respectively, the input and output vectors of Equation 2.7 can be described as:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{u}_y \\ \mathbf{f}_y \\ \mathbf{p}_y \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} \mathbf{u}_x \\ \mathbf{f}_x \\ \mathbf{p}_x \end{bmatrix} \quad (2.8)$$

Where the quantities in Equation 2.8 are vectors as well:

$$\mathbf{u}_y = \begin{bmatrix} u_y^{(1)} \\ \vdots \\ u_y^{(k)} \end{bmatrix} \quad \mathbf{u}_x = \begin{bmatrix} u_x^{(1)} \\ \vdots \\ u_x^{(n)} \end{bmatrix} \quad \mathbf{f}_y = \begin{bmatrix} f_y^{(1)} \\ \vdots \\ f_y^{(l)} \end{bmatrix} \quad \mathbf{f}_x = \begin{bmatrix} f_x^{(1)} \\ \vdots \\ f_x^{(o)} \end{bmatrix} \quad \mathbf{p}_y = \begin{bmatrix} p_y^{(1)} \\ \vdots \\ p_y^{(m)} \end{bmatrix} \quad \mathbf{p}_x = \begin{bmatrix} p_x^{(1)} \\ \vdots \\ p_x^{(p)} \end{bmatrix} \quad (2.9)$$

Note that the dependency on frequency has been omitted for clarity, and that the indices (k, l, m, n, o, p) indicate that the number of measurement points (also referred to as DOFs) for each quantity may be different. Also, not all quantities have to be included as a measurement.

For example, the number of sound pressure measurements will be dependant on the number of significant airborne sound sources, and the number of force and/or vibration measurements will be dependant on the number of significant structure-borne sources/paths. The response measurements for a vehicle could simply be sound pressure levels at two locations - at the driver's left and right ear.

A key aspect to the OTPA method is that it is up to the engineer to design the measurement setup. Care must be taken to properly measure all significant sources. The effects of a poor measurement setup (i.e. neglected sources/paths in the measurement) will lead to erroneous contribution estimates - this is discussed further in Section 2.2.4.

The main advantage of OTPA over classical TPA is that it is possible to determine all of the transfer paths simultaneously, when operational excitations are present. Physical isolation of sources/paths is not required for the OTPA method.

Several measurement points can be included in the analysis by taking the transpose of Equation 2.7:

$$\begin{bmatrix} y^{(1)} & \dots & y^{(n)} \end{bmatrix} = \begin{bmatrix} x^{(1)} & \dots & x^{(m)} \end{bmatrix} \begin{bmatrix} H_{11} & \dots & H_{n1} \\ \vdots & \ddots & \vdots \\ H_{1m} & \dots & H_{mn} \end{bmatrix} \quad (2.10)$$

Where m and n represent the number of response (output) and reference (input) measurement points, or DOFs.

For operational measurements, particularly in NVH analysis of vehicles, it is often desirable to relate the response at the receiver at several operational states, e.g. with respect to the RPM of the engine. In this case, the input and output measurement data is constantly changing. However, at a particular instant, or for a particular block of measurement data, the reference (input) and response (output) are related by the corresponding transfer function.

This relationship is particularly useful in the analysis of a vehicle run-up, for example. Assuming that the relationship between reference (input) and response (output) are linear and constant for each measurement block, Equation 2.10 can be modified to show the measurement blocks, denoted r , as shown in Equation 2.11.

$$\begin{bmatrix} y_1^{(1)} & \dots & y_1^{(n)} \\ \vdots & \ddots & \vdots \\ y_r^{(1)} & \dots & y_r^{(n)} \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & \dots & x_1^{(m)} \\ \vdots & \ddots & \vdots \\ x_r^{(1)} & \dots & x_r^{(m)} \end{bmatrix} \begin{bmatrix} H_{11} & \dots & H_{n1} \\ \vdots & \ddots & \vdots \\ H_{1m} & \dots & H_{mn} \end{bmatrix} \quad (2.11)$$

Recall that Equation 2.11 is in the frequency domain, $j\omega$, and therefore the computation must be conducted for each frequency component – that is to say for each line of the discrete Fourier transform (DFT) of the sampled reference and response measurement point data.

Equation 2.11 is defined in compact form as:

$$\mathbf{Y} = \mathbf{X}\mathbf{H} \quad (2.12)$$

If the reference (input) matrix, \mathbf{X} , is square, i.e. in Equation 2.11 $m = r$, then the solution for the transfer functions can be determined by multiplying the inverse to both sides, which would yield:

$$\mathbf{H} = \mathbf{X}^{-1}\mathbf{Y} \quad (2.13)$$

However, in most cases, the input matrix is not square (i.e. $m \neq n$). In this case, the solution for the transfer function can be found using the least-squares method. The least-squares solution, in compact form, can be written as:

$$\mathbf{H} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{Y} = \mathbf{X}^+ \mathbf{Y} \quad (2.14)$$

Where \mathbf{X}^+ is the pseudo-inverse of \mathbf{X} , i.e. $\mathbf{X}^+ = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T$.

Note that in order for Equation 2.11 to be solvable for the transfer functions (in the least-squares sense), the number of measurement blocks, r , must be greater than the number of measurement DOFs, m , i.e. $r > m$ [22].

Solving for the transfer functions using this method could be error-prone if the reference (input) signals are highly coherent and include measurement noise. This is due to the amplification of measurement noise when calculating the pseudo-inverse, \mathbf{X}^+ (the noise is amplified in the term $(\mathbf{X}^T \mathbf{X})^{-1}$). To prevent poor estimates of the transfer function, singular value decomposition (SVD) is used to determine the pseudo-inverse of the input matrix, \mathbf{X}^+ , further, principal component analysis (PCA) methods are used to disregard measurement noise and cross-talk between measurement channels [5, 7, 16].

2.2.2. The Cross-Talk Cancellation Method

Since for OTPA all of the measurements are taken simultaneously, cross-talk between measurement channels is expected. To eliminate the cross-talk, singular value decomposition (SVD) and principal component analysis (PCA) techniques are used. The SVD is used for two main reasons - First, to solve the linear least-squares problem of finding \mathbf{X}^+ , the inverse of the input matrix, \mathbf{X} . Second, the SVD is used because it is a computationally efficient method of finding the principal components (PCs) [13]. In the following, the mathematics behind the CTC method are discussed.

Singular Value Decomposition

The SVD, a commonly used method for solving most linear least-squares problems, is used to define an expression for the reference (input) measurements, shown in Equation 2.15. The purpose of the SVD is to solve for the singular values (i.e. eigenvalues) of the data set. Essentially, the SVD finds the correlated components of a data set, such that each singular value is uncorrelated (i.e. orthogonal) to the other singular values. The strongest correlated component of the data is ranked as the first singular value, leaving behind an uncorrelated residual data set. The SVD is then conducted on the residual to find the next singular value, again leaving a residual. The computation is repeated until the full data set is represented by the SVD [3, 19].

$$\mathbf{X} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \quad (2.15)$$

Where:

\mathbf{X} is the an $r \times m$ matrix consisting of the reference, or input, measurements;

\mathbf{U} is an $r \times m$ unitary column-orthogonal matrix¹;

$\mathbf{\Sigma}$ is an $m \times m$ diagonal matrix with the singular values (positive or zero elements);

\mathbf{V}^T is the transpose of an $m \times m$ unitary column-orthogonal matrix, \mathbf{V} ;

For OTPA, the number of reference (input) measurement blocks are assumed to be greater than the number of reference DOFs, i.e. $r > m$.

N.B. If \mathbf{X} includes a mix of sound pressure and vibration measurements, the data must be normalized prior to conducting the SVD. This is particularly important for the PCA and consequently for proper calculation of the source contribution (discussed in Section 2.2.3).

In order to find the matrices \mathbf{U} , $\mathbf{\Sigma}$, \mathbf{V} satisfying Equation 2.15, the following relations are used:

$$\mathbf{X}^T \mathbf{X} = \mathbf{V} \mathbf{\Sigma}^T \mathbf{U}^T \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T = \mathbf{V} (\mathbf{\Sigma}^T \mathbf{\Sigma}) \mathbf{V}^T \quad (2.16)$$

$$\mathbf{X} \mathbf{X}^T = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \mathbf{V} \mathbf{\Sigma}^T \mathbf{U}^T = \mathbf{U} (\mathbf{\Sigma} \mathbf{\Sigma}^T) \mathbf{U}^T \quad (2.17)$$

In Equation 2.16, \mathbf{V} is the eigenvector matrix and the term $(\mathbf{\Sigma}^T \mathbf{\Sigma})$ is the eigenvalue matrix, with the squares of the singular values along the diagonal, for the term $\mathbf{X}^T \mathbf{X}$.

Similarly, in Equation 2.17, \mathbf{U} is the eigenvector matrix and the term $(\mathbf{\Sigma} \mathbf{\Sigma}^T)$ is the eigenvalue matrix, with the squares of the singular values along the diagonal, for the term $\mathbf{X} \mathbf{X}^T$.

Note that the squares of the singular values along the diagonals of $(\mathbf{\Sigma}^T \mathbf{\Sigma})$ and $(\mathbf{\Sigma} \mathbf{\Sigma}^T)$ are the same.

The pseudo-inverse, \mathbf{X}^+ , can be found using the SVD expression of \mathbf{X} , shown in Equation 2.15.

$$\mathbf{X}^+ = \mathbf{V} \mathbf{\Sigma}^{-1} \mathbf{U}^T \quad (2.18)$$

Combining Equations 2.14 and 2.18, an estimate of the solution for the transfer path matrix, $\tilde{\mathbf{H}}$, can be described in the following form:

$$\tilde{\mathbf{H}} = \mathbf{V} \mathbf{\Sigma}^{-1} \mathbf{U}^T \mathbf{Y} \quad (2.19)$$

¹A unitary matrix has the properties that $\mathbf{V} \mathbf{V}^T = \mathbf{I}$ and $\mathbf{V}^T \mathbf{V} = \mathbf{I}$. This also implies that $\mathbf{V}^T = \mathbf{V}^{-1}$. If $r \geq m$, the same holds true for matrix \mathbf{U} (for OTPA, $r > m$) [19].

Equation 2.19 indicates an estimate for the solution of the transfer path matrix, however, the measurement noise and cross-talk between measurement channels has not yet been addressed, and is still included in the analysis data. The measurement noise and cross-talk cancellation should be addressed prior to calculating the pseudo-inverse, \mathbf{X}^+ , thus the PCA (discussed in the next sub-section) should be conducted after the SVD of matrix \mathbf{X} [7, 16, 19].

Principal Component Analysis

To determine what data is not relevant to the overall response at the receiver (e.g. measurement noise), and to avoid numerical problems in calculating the transfer functions, principal component analysis (PCA) methods are used.

PCA is a technique used to reduce a data set that consists of a large number of inter-related variables to a smaller data set that preserves most of the original information. More specifically, the PCA involves an orthogonal transformation (i.e. SVD) that converts the data so that it is uncorrelated.

These uncorrelated variables are called the principal components (PC), and are ranked such that the PC with the largest variance within the data set is the first PC, the next variable is the second PC and so on. The lowest PCs exhibit very little variation - that is to say that the relationship among the variables is almost constant and linear.

For example, when conducting PCA on vibration data, and the strongest contribution is due to bending, then that would be the first PC. The PCA would continue on the remaining data (i.e. the residual), and the next strongest contribution, for example torsional waves, would be considered the second PC. The analysis would continue in this manner for all vibrational DOFs. This analysis is conducted similarly for sound pressure, with the ranking of PCs related to phase relationship.

In the case of OTPA, the PCA is conducted in the frequency domain - the DFT spectra are the input data (variables) for each measurement point. Further, the PCA is conducted on all the measurement blocks (observations), r , which are considered to be part of vectors $[x^{(1)} \dots x^{(m)}]$. In essence, the PCA determines to what extent the measurement points share common signals (i.e. to what extent the signals are correlated) [15].

It is possible to find as many PCs as there are data sets. For the OTPA, the maximum number of PCs would be the number of reference measurements positions, m . More specifically, the number of significant PCs will correspond to the number of significant DOFs of all sources and/or paths (if all sources/paths are properly identified and included in the measurement setup).

The PCs can be determined from the SVD of a matrix. In fact the PCs are the singular values found along the diagonal of the matrix Σ , refer to Equation 2.15. Recall that the SVD separates the correlated component of a signal from the uncorrelated component, therefore It is important to keep in mind that the SVD can only separate sources that are either uncorrelated or only partly correlated.

The PC scores are obtained by multiplying the matrix of the PCs, Σ , by the eigenvector, \mathbf{U} . Referring to Equation 2.15, this yields the relationship shown in Equation 2.20, where \mathbf{Z} is the matrix of PC scores.

$$\mathbf{Z} = \mathbf{X}\mathbf{V} = \mathbf{U}\Sigma \quad (2.20)$$

At this stage, the matrix of PC scores, \mathbf{Z} , should be analyzed and the smaller PCs (i.e. measurement noise and cross-talk) noted so that they can be discarded in subsequent computations.

The contribution of each PC to the overall signal can be investigated by dividing \mathbf{Z} by the total sum of the PC scores, which yields a percentage contribution for each PC score. The smaller PCs can be discarded from the analysis by setting them to zero, however these values will have to be redefined when calculating the inverse of Σ_r .

The noise removed PC matrix, Σ_r , can now be substituted into Equation 2.19 to yield an expression for the cross-talk cancelled and noise removed estimate of the transfer function matrix:

$$\tilde{\mathbf{H}}_r = \mathbf{V}\Sigma_r^{-1}\mathbf{U}^T\mathbf{Y} \quad (2.21)$$

Where Σ_r^{-1} is the inverse of the noise removed PC matrix and is a diagonal matrix. The values along the diagonal of Σ_r^{-1} are defined as:

$$\Sigma_r^{-1} = \begin{cases} 1/\sigma_n & \text{if } \sigma_n > \text{thres} \\ 0 & \text{otherwise} \end{cases} \quad (2.22)$$

Where *thres* is a threshold for the minimum size of the principal components to be included in the analysis. The inverse of the PCs must be defined as shown in Equation 2.22 to avoid the result of a singular matrix upon taking the inverse of Σ_r [3, 12, 13, 16].

2.2.3. Response Synthesis

Upon solving for the the noise removed transfer function, $\tilde{\mathbf{H}}_r$, the response synthesis at the receiver can be calculated by taking the convolution of the response matrix with the transfer function matrix.

$$\mathbf{Y}_s(t) = \mathbf{X}(t) \otimes \tilde{\mathbf{H}}_r(t) \quad (2.23)$$

Where:

$\mathbf{Y}_s(t)$ is the matrix of the response syntheses at the receivers in the time domain;

$\tilde{\mathbf{H}}_r(t)$ is the operational transfer function matrix in the time domain;

$\mathbf{X}(t)$ is the matrix of the reference (input) measurements, in the time domain ;

\otimes denotes that the operation is a convolution of $\tilde{\mathbf{H}}_r(t)$ and $\mathbf{X}(t)$;

Note that Equation 2.23 is in the time domain, and the convolution must be conducted for each time component. $\tilde{\mathbf{H}}_r(t)$, which is computed by taking the inverse DFT of $\tilde{\mathbf{H}}_r(j\omega)$, is used in the computation as a vector of finite length impulse response (FIR) filters which are convolved with the time domain reference (input) measurements to determine the response (output).

In the frequency domain, the convolution may be written as the point-wise multiplication between $\tilde{\mathbf{H}}_r$ with \mathbf{X} , refer to Equation 2.24.

$$\mathbf{Y}_s(j\omega) = \mathbf{X}^T(j\omega) \cdot \tilde{\mathbf{H}}_r(j\omega) \quad (2.24)$$

Where:

$\mathbf{Y}_s(j\omega)$ is the matrix of the response syntheses at the receivers in the frequency domain;

$\mathbf{X}^T(j\omega)$ is the transpose matrix of the reference (input) measurements, in the frequency domain;

$\tilde{\mathbf{H}}_r^T(j\omega)$ is the operational transfer function matrix in the frequency domain;

The "." represents a point-wise multiplication between $\mathbf{X}(j\omega)$ and $\tilde{\mathbf{H}}_r^T(j\omega)$

The resulting response syntheses, $\mathbf{Y}_s(t)$ and $\mathbf{Y}_s(j\omega)$, are vectors of the response contributions at each reference measurement position, for each measurement block. The total response at the receiver is the sum of the response contributions from each reference measurement position. The response synthesis of each measurement block can then be assembled to plot the response synthesis over the entire measurement time (e.g. in a Campbell diagram).

The contribution of each identified source to the overall response sound pressure level can be analyzed by summing the response contributions of the corresponding synthesis vectors. For example, if *Source A* includes four reference measurement positions, the corresponding synthesis contribution of these four measurements should be summed to yield the contribution of *Source A* to the overall response at the receiver.

The overall response synthesis (i.e. the sum of all synthesis contributions) is used to determine the effectiveness of the OTPA computations by comparing the synthesis to the actual measured response. However, the results of this comparison should be used

with care - the overall response synthesis will almost always match the measured response at the receiver. This does not, however, necessarily mean that the calculated contributions are correct [24]. The limitations of the OTPA method are discussed in Section 2.2.4.

2.2.4. OTPA Limitations and Potential Sources of Error

The main characteristic of the OTPA method is that the resulting transfer functions are not FRFs, but are analogous to transmissibilities. Therefore the transfer functions are specifically related to the operational excitation signals used in the computation. If certain modes are not excited by the excitation signals, these modes will not be identified in the analysis. This fact is usually considered acceptable for the analysis of a vehicle since the typical operational forces can be included in the analysis (e.g. idle, acceleration, coast-down), thus all relevant excitations are included.

A potential source of error in using the OTPA method stems from the fact that the reliability of the results is highly dependant on properly setting up the measurement. This is also true for the classical TPA method, however an error in identifying all of the significant sources/paths is apparent during analysis of the classical TPA results, whereas for OTPA it may not be so clear.

The reliability of the OTPA model is often checked by comparing the actual measured response to the response synthesis. A common claim for its accuracy is based on the comparison of the the sum of all source/path contributions to the actual measured signal. This comparison, however, does not prove that the individual source/path synthesis contributions are correct [14].

In fact, the overall response synthesis and the actual measured response will always have very similar, if not matching, results. This occurs because similar measured inputs are used in the computation of the transfer function.

A literature review indicates that there are three critical elements that could lead to erroneous transfer path estimates [9, 10, 14, 24]:

1. Neglected sources/paths in the measurement setup.
2. Cross-coupling between input measurements (i.e. CTC is not able to separate the sources/paths).
3. Incorrect estimation of the transmissibilities (transfer paths).

The result of these three elements is that the source/path contribution calculation is erroneous, however, as stated earlier in this section, the overall response synthesis may still appear to be correct (i.e. the overall response synthesis matches the measured response). These elements are briefly discussed in the following.

Neglected sources/paths in the measurement setup

Neglected sources/paths can introduce errors particularly if they are correlated with those included in the analysis, as its energy will be spread over to the other sources/-paths. Consequently, the overall response synthesis will still match the measured response – the synthesis contributions will be skewed to account for the energy of the source/path contributions that were not included in the measurement setup.

If the neglected sources/paths are not correlated with those that are included in the analysis, the error will show up in the overall response synthesis – it will not match the measured response.

Cross-coupling between input measurements

Difficulties in the cross-talk cancellation using SVD and PCA arise when two or more strong and highly coherent sources/paths exist for the system. The CTC technique is not able to accurately separate two completely coherent sources/paths, and therefore the results yield erroneous contribution estimates.

If there is cross-talk between non-coherent, or even partly coherent sources/paths, the CTC technique can be used successfully for source/path separation.

Incorrect estimation of transfer paths

The third critical element refers to the fact that the OTPA method relies on SVD and PCA in order to calculate the pseudo-inverse of the matrix of reference/input measurements. Because the pseudo-inverse calculation is an approximation, the results may not be accurate.

Further, because the OTPA is based on a transmissibility, if all of the significant input excitations are not included in the analysis, the calculated transfer paths will be incorrect. The potential for this error can be alleviated by including measurements for several operational conditions in the reference/input matrix (e.g. run-up and run-down at several gear positions).

3. Implementation of OTPA and CTC

Based on the OTPA theory discussed in Section 2.2, an OTPA script has been implemented using MATLAB. The OTPA program flow is summarized in the following:

1. Load measurement data. If the data is in the time domain, transform the data to frequency domain (i.e. DFT)¹.
2. Assemble measurement data into reference matrix, $\mathbf{X}(j\omega)$ and response matrix $\mathbf{Y}(j\omega)$.
3. SVD of matrix $\mathbf{X}(j\omega)$ into its principal components.
4. PCA of matrix \mathbf{S} , which consists of the PCs along its diagonal.
5. Remove less important PCs (i.e. noise and cross-talk from the other source).
6. Compute the pseudo-inverse of $\mathbf{X}_r(j\omega)$ – the noise removed and cross-talk cancelled matrix of reference measurements.
7. Calculate the cross-talk cancelled transfer function, $\tilde{\mathbf{H}}_r(j\omega)$.
8. Load a separate set of input data, \mathbf{X} so that the calculation of the response synthesis is not circular logic (i.e. the same input data used to compute the transfer function, is not used again to compute the response synthesis).
9. Calculate the response synthesis, $\mathbf{Y}_s(j\omega)$, as well as the source contributions by adding up the relevant vectors of $\mathbf{Y}_s(j\omega)$ (i.e. those that correspond to the source's reference MPs).
10. Calculate the transfer function in the time domain, $\tilde{\mathbf{H}}_r(t)$, by taking the inverse DFT of $\tilde{\mathbf{H}}_r(j\omega)$.
11. Calculate the response synthesis in the time domain by convolving $\mathbf{X}(t)$ and $\tilde{\mathbf{H}}_r(t)$.

A flow chart of the OTPA script organization is displayed in Figure 3.1.

¹The virtual test setup, described in Chapter 4, is implemented in the frequency domain, thus signal processing to transform the signal from the time domain to the frequency domain is not required, and is not further discussed in this thesis.

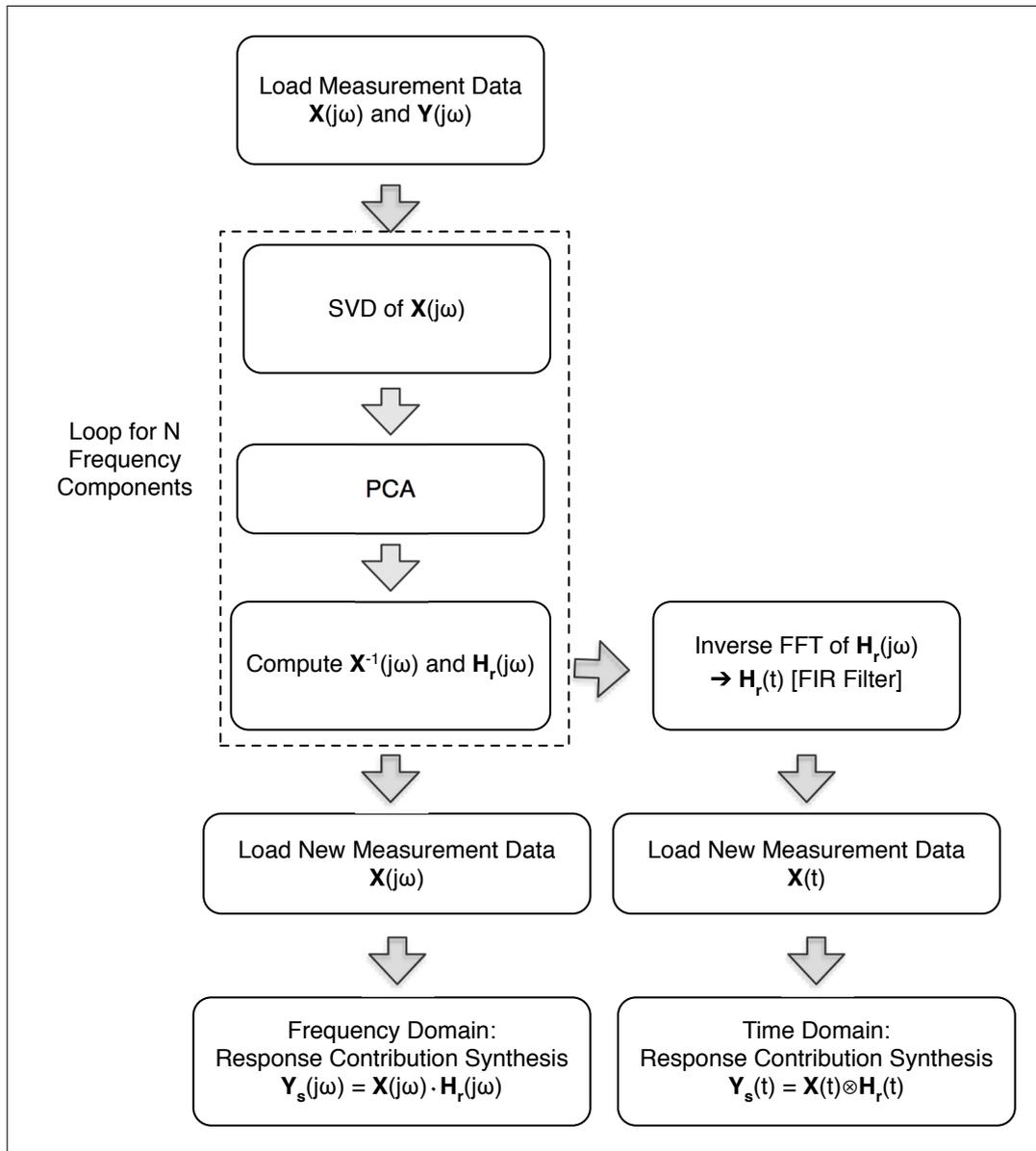


Figure 3.1.: OTPA Program Flow Chart

3.1. MATLAB Script: Description of Key Functions

A description of the key functions used in the OTPA implementation in MATLAB are described in the following.

3.1.1. Singular Value Decomposition

The OTPA script is made fairly simple by using the embedded MATLAB function for the singular value decomposition of the matrix $\mathbf{X}(j\omega)$. The command to compute the SVD is:

$$[\mathbf{U}, \mathbf{S}, \mathbf{V}] = \text{SVD}(\mathbf{X}) \quad (3.1)$$

Where \mathbf{U} and \mathbf{V} are the eigenvector matrices and \mathbf{S} is the matrix of singular values (i.e. principal components). Observe that in Equation 2.15, the matrix of singular values, Σ , is represented by \mathbf{S} in the MATLAB script. Refer to Section 2.2.2 for the theory behind the SVD computation.

The output of this function reveals the principal components and eigenvector matrices from which the principal component scores as well as cumulative contribution are determined, see Equation 2.20.

3.1.2. Principal Component Analysis and Cross-Talk Cancellation

There are several methods to conduct the PCA. Four of these methods are included in the OTPA script, however, to maintain consistency, only one method is used for the analyses. The PCA methods included in the script are described in the following:

The first option is to simply define the number of PCs that are to be included in the analysis. For example, if that number is defined as two, the two highest ranked PCs are included in matrix \mathbf{S} , and the rest are set to discarded (e.g. set to zero).

The second option, which is the method used for the analyses described in this thesis report, is to define a threshold, as a percentage, for the minimum contribution of the PCs towards the total sum of the PC scores (i.e. towards the total contribution). For example, if a threshold of 5% is defined, any PCs that contribute 5% or less to the overall contribution will be excluded.

The third option is to define a minimum cumulative contribution that the PCs included in the analysis must meet. For example, if a cumulative contribution of 90% is defined, the highest ranked PCs are included in the analysis until the cumulative contribution is equal to 90%. Any left over PCs are set to zero, and thus discarded from the analysis.

The fourth option allows the user to simply identify the minimum magnitude of the principal components to be included in the the analysis (i.e. the minimum magnitude of the values in S that are to be included).

3.1.3. Response Synthesis and Contribution Analysis

Once the transfer function, H , has been determined, the response synthesis is calculated by a point-wise multiplication between X and H .

$$Y = X' .* H \quad (3.2)$$

Where the matrix of input reference measurements, X is transposed so that the MPs are multiplied by the corresponding transfer function for each measurement block. The resulting response matrix, Y , consists of the contribution to the total response from each measurement position. Therefore, if a source includes two MPs, these two MPs must be summed to yield the contribution of that particular source. Similarly, the overall synthesis at a particular receiver is the sum of all the contributions of that corresponding vector in matrix Y .

Note that these functions are repeated for each frequency component (i.e. looped for N iterations, where N corresponds to the number of frequency components).

The full Matlab script for the OTPA program is included in Appendix A. Note that the included script is customized for the analyses included in this report. Further, the OTPA script has its limitations – as the number of sources/paths increases, it may become impractical to use the provided script.

4. The Virtual Test Setup

A virtual test setup (VTS) has been created in order to study the OTPA method in a highly controlled environment. The VTS was chosen over a real test setup so that all of the "measurement" parameters could be controlled. Further, the analysis results obtained using the VTS are not influenced by any unknown external sources of error.

The main objective of the study is to recreate the over-prediction of airborne sound and consequently to determine if limitations and/or rules of thumb can be clearly defined for the OTPA method in this frequency range.

4.1. Virtual Test Setup Layout

The virtual test setup has been created in order to address the hypotheses stated in Section 1.2.1. Although various scenarios are investigated, the basic VTS layout can be described as two spherically radiating sources and one receiver position in a completely free-field environment. The two sources, *Source A* and *Source B*, can be moved away from each other, while the distances from *Source A* to the receiver and *Source B* to the receiver remains equal. Each source is "measured" by one reference MP for most of the analyses, however, in some cases more than one reference MP is included. The distance between reference MPs and the source may be defined in the program, but it is generally at the same distance for most of the analyses. A schematic of the virtual test setup is shown in Figure 4.1.

The VTS layout was implemented using MATLAB by defining the source, reference and response positions as coordinates. The distances between points is then easily determined.

The MATLAB script used to define the layout is included in Appendix A.

4.2. Source Signals - Background Theory

The sources are defined so that all computations can be conducted in the frequency domain – This is done for simplicity, since the OTPA analysis is generally conducted in the frequency domain (although it is sometimes conducted in the time domain [25]).

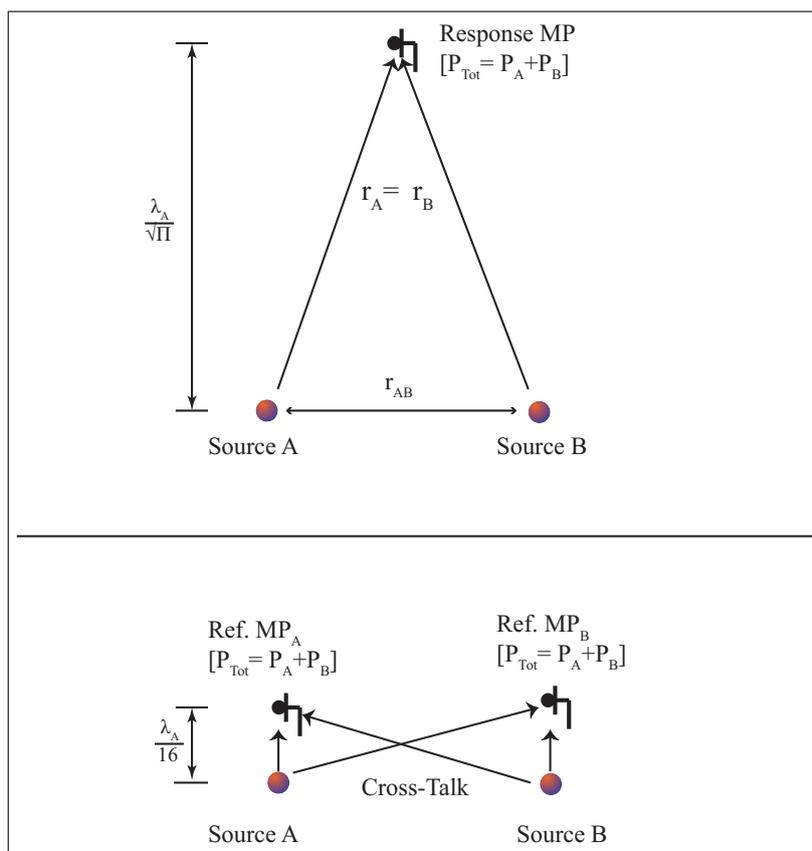


Figure 4.1.: Virtual Test Setup Schematic

The two sources are modelled as radiating spherical sources of radius a and b (where $a = b$). The sound pressure level at a radius r_A from the *Source A* is described by Equation 4.1.

$$\hat{p}_A(r, \omega) = v_A \frac{j\omega\rho_0 a^2}{1 + jk_A a} \frac{e^{-jk_A(r_A - a)}}{r_A} \quad (4.1)$$

Where:

v_A is the vibrational velocity of the surface of *Source A* at frequency ω ;

$j\omega$ denotes the complex angular frequency;

ρ_0 is the density of the surrounding medium (i.e. air). k_A is the wave number, where $k_a = \omega/c = 2\pi/\lambda$, and c is the speed of sound;

r_A is the distance from *Source A* to the MP;

a is the radius of the spherical source.

The resulting sound pressure level at the reference and response measurement posi-

tions can then be described as the vector addition of the contribution of the two sources, as shown in Equation 4.2 [2, 4].

$$\hat{p}_{total}(r, \omega) = \left[v_A \frac{j\omega\rho_0 a^2}{1 + jk_A a} \frac{e^{-jk_A(r_A - a)}}{r_A} + v_B \frac{j\omega\rho_0 b^2}{1 + jk_B b} \frac{e^{-jk_B(r_B - b)}}{r_B} \right] \quad (4.2)$$

The MATLAB script for the computation of the source signal and resulting sound pressure levels at the receiver MPs and response MP is included in Appendix A.

4.3. VTS and OTPA Parameters

4.3.1. VTS Parameters

The parameters used in the VTS were kept to reasonable values. For example, the size of the vibrating spheres, that is, the radii for Sources A and B are defined as $\frac{\lambda_A}{40}$ (at 25 Hz this is 0.35 m), which is a fairly close approximation of the size of the engine of a vehicle.

The reference MPs for the sources are positioned at a radius of $\frac{\lambda_A}{16}$ from the centre of the sources. At 25 Hz, this distance puts the MPs at a distance of 0.5 m from the surface of the sources. Again, this distance is realistic when compared to microphone placement for actual OTPA measurement setups.

The response MP is equidistant from both sources, with the right-angle distance of $\frac{\lambda_A}{\sqrt{\pi}}$. This distance was chosen simply so that the distance is not an integer multiple of λ_A , as using an integer multiple could lead to unique results specific to that position.

The vibration velocity of the sphere's surface is arbitrary, and is defined simply to yield a good scale for the plots. The actual magnitude of the source is not of importance since only the relative values are of interest in this study. Both sources are in phase, unless a phase shift is specifically mentioned in the analysis.

The OTPA analysis calls for several measurement blocks of data – this is treated by simply looping the VTS output so that the reference matrix includes several measurement blocks with the same response. This is done to keep the analysis of the OTPA simple.

Noise is introduced to the signals by adding a random sound pressure to the VTS throughout the full frequency range of interest (up to 200 Hz). The random noise is different for each "measurement block".

Some of the input variables for the sources may vary depending on the analysis, however any significant changes are noted in the discussion for each individual analysis.

4.3.2. OTPA Parameters

The input parameters for the OTPA relate to the principal component analysis, which is used for the cross-talk cancellation. The VTS is quite simple, such that with only two reference MPs (one for each source), at any frequency the maximum number of PCs would be two. For the sake of consistency between the analyses, the PCA technique used is a minimum threshold towards the total contribution at each frequency component, which was set to 5% for all analyses (unless otherwise stated).

The MATLAB script for analyses are included in Appendix A.

5. Analysis

5.1. Analysis Outline

The study of the OTPA method is set up to first validate both the OTPA and VTS programs, and then to investigate the hypotheses discussed in Section 1.2.1.

The validation of the OTPA program is conducted by examining a simple case consisting of two airborne sources radiating random noise throughout the frequency range of interest, as well as a dominant tone. The effects of varying the PCA threshold value on the overall source synthesis are also examined.

Next, two simple cases consisting of two airborne sources and one receiver are studied. These cases are used to study the effects of cross-talk between reference MPs.

The VTS is then slightly modified so that it includes one airborne sound path and one structure-borne sound path. This setup is designed to be a simplified setup of the tractor OTPA case study (refer to Section 1.1). Furthermore, a "falsely identified" airborne source is also included in this VTS – this is analogous to including the tire airborne sound reference MPs during the neutral OTPA of the tractor (i.e. the tires are not a significant noise source when the vehicle is stationary).

Finally, because the number of reference MPs included in the actual measurement appeared to have an effect on the tractor OTPA results¹, the effect of the number of reference MPs included in the analysis is also studied.

In summary, the OTPA study using the VTS is structured as follows:

1. Validation of the OTPA MATLAB program: Noise removal using PCA.
2. The effect of varying distance between two fully correlated airborne sources on the contribution prediction results.
3. The effect of varying the distance between two fully uncorrelated airborne sources on the contribution prediction results.
4. The effect of including a structure-borne source/path which is correlated with the airborne source as well as a second "falsely identified" airborne source (i.e. a source that is not radiating sound) on the contribution prediction results.

¹Recall that the calculated source contribution was different for the analysis with twelve airborne reference MPs when compared to the case with six airborne reference MPs

5. The effect of varying the number of airborne source reference MPs on the contribution prediction results.

5.2. OTPA Validation: CTC and Noise Removal Using PCA

To confirm that the VTS and the OTPA program is functioning properly, a simple case is examined: Two spherical sources that radiate different tones (i.e. fully uncorrelated sources). To verify that the PCA routine is functioning, the effects of varying threshold levels for the PCA are studied.

Upon conducting the SVD of the input data, the smaller PCs, which are considered noise and cross-talk, must be removed from the analysis in order to avoid errors in the calculation of the pseudo-inverse of the input matrix, \mathbf{X}^+ , and subsequently the matrix of transfer functions, \mathbf{H} . This is done by PCA – the PCs that do not meet the pre-defined threshold are discarded from the analysis. Naturally, it is important to define the PCA threshold correctly.

In order to test the PCA of the OTPA implementation, noise was added to all of the MPs throughout the frequency range of interest (up to 200 Hz), while the sources were defined to radiate tones at 25 Hz and 50 Hz in addition to the random noise throughout the rest of the frequency range. To increase the number of PCs in the PCA, each source was associated with four reference MPs¹. The VTS layout, which also shows the sound fields at 25 Hz and 50 Hz, is displayed in Figure 5.1. A distance between *Source A* and *Source B* of $\frac{\lambda_A}{6}$ was used for this study.

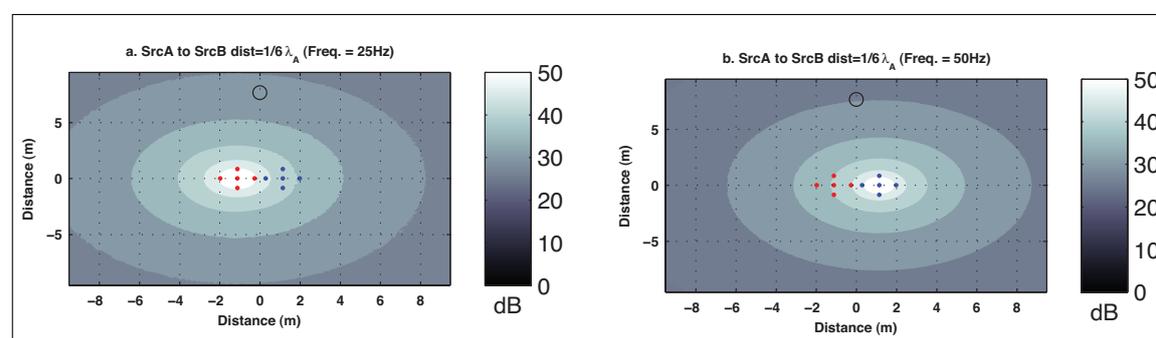


Figure 5.1.: VTS layout (4 reference MPs)

For comparison, four cases with different threshold settings for the PCA were conducted – the response synthesis results corresponding to PCA threshold levels of 1%, 5%, 15% and 40% are plotted in Figures 5.2a, c, e and g respectively. The corresponding contribution of the PCs are plotted in Figure 5.2b, d, f and h, respectively.

Referring to Figure 5.2b, observe that at 25 Hz and 50 Hz, the first PC is contributing to approximately 80% of total magnitude of the signal. Throughout the rest of the frequency range (i.e. where the signal is defined as random noise), the PCs are all quite close in terms of relative contribution.

¹There can be as many PCs as there are reference MPs, i.e. up to eight PCs in this case.

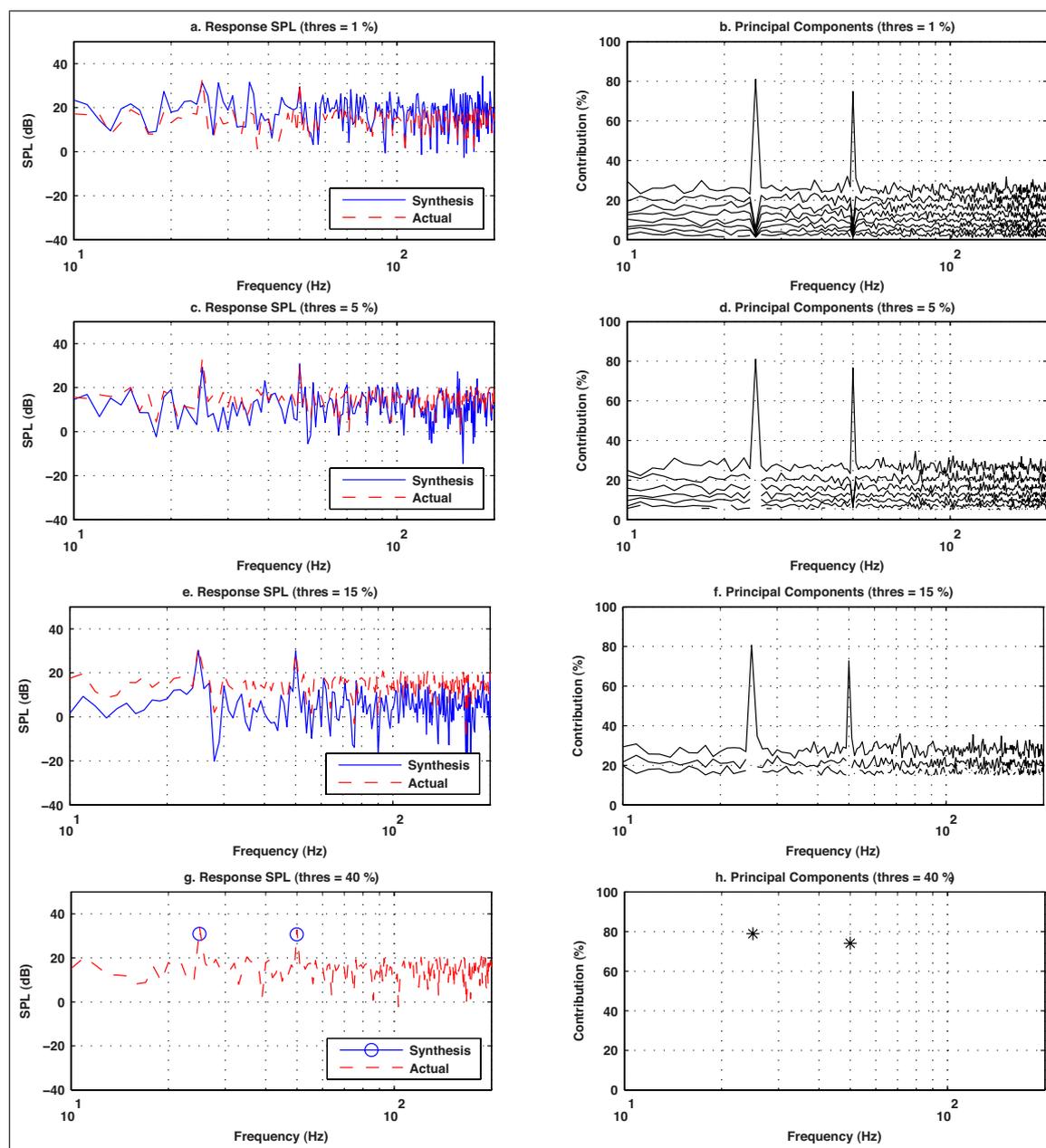


Figure 5.2.: Response synthesis and principal component contribution

For the first case, the threshold for the PCA was set at 1%. From Figure 5.2b, it appears as though almost all PCs are included in the computation throughout the frequency range (in fact only six of eight are included at 25 Hz and 50 Hz, and all PCs are included at the other frequencies). Referring to Figure 5.2a, we see that the response synthesis does not match the actual response very well (there are significant peaks in the synthesis results at around 28 Hz and 35 Hz).

Increasing the threshold to 5% yields the response synthesis displayed in Figure 5.2c. For this case, we see from Figure 5.2d that only two or three PCs are included in the computation at the radiated tones (25 Hz and 50 Hz), and six or seven PCs are included for the rest of the frequency components. Note that with only a slight increase in the threshold level, the response synthesis matches the actual response quite accurately. The response synthesis has been improved because the small PCs have now been discarded – thus the uncorrelated noise component (the noise) has been removed from the analysis, which in turn reduces the potential error in the computation of the pseudo-inverse of the input matrix, and subsequently the transfer function.

For the next case, the threshold was again increased slightly, to 15%. Referring to Figure 5.2f, we see that only the first PCs remain at 25Hz and 50Hz, while throughout the rest of the frequency range two to three PCs are included. Looking at the response synthesis, refer to Figure 5.2e, we see that although the tones at 25Hz and 50Hz match with the actual response, the synthesis throughout the rest of the frequency range does not agree very well.

Upon examination of the PCs that are included in the analysis, we see that the two or three remaining PCs only represent approximately 45% of the total signal – thus the response synthesis is an under-prediction of the actual response.

If it is desired to remove the noise completely while keeping the dominant tones in the analysis, a higher threshold level could be used. This case was tested by setting a threshold level of 40% for the analysis. Referring to Figure 5.2h, we observe that only the first PCs at 25Hz and 50Hz remain – all other PCs have been discarded from the analysis. Therefore, the resulting response synthesis, displayed in Figure 5.2g, only includes the tones at 25 Hz and 50 Hz, as expected.

The results of this study confirm that the PCA of the OTPA program is functioning as expected – thus validating the implementation. Also, the results show that the PCA is a useful tool to identify the strong contributions in a signal – however, it is important to appropriately set the threshold for the PCA in order to yield accurate results.

5.3. OTPA of Two Airborne Noise Sources

The following case studies consist of two spherical airborne sources, labeled *Source A* and *Source B*, which have a certain surface vibration that causes a radiated sound power at a specified tone. In the first case, two correlated sources are studied, and in the second case two uncorrelated sources are studied.

5.3.1. Varying Distance Between Two Correlated Sources

This case study consists of two airborne sources, labeled *Source A* and *Source B*, that radiate sound power at a single tone, 25 Hz. Both sources are the same size, have the same amplitude, and are in phase. The response MP (i.e. the receiver) is equidistant from both sources, with an initial distance of $\frac{\lambda_A}{\sqrt{\pi}}$.

A plot of the VTS layout that includes the sound field is displayed in Figure 5.3. Note that the response MP is denoted by a circle at the top of the figures, while the sources are in the centre; *Source A* is moved to the left, and *Source B* is moved to the right. Each source is represented by one reference MP – located to the left of *Source A*, and to the right of *Source B*.

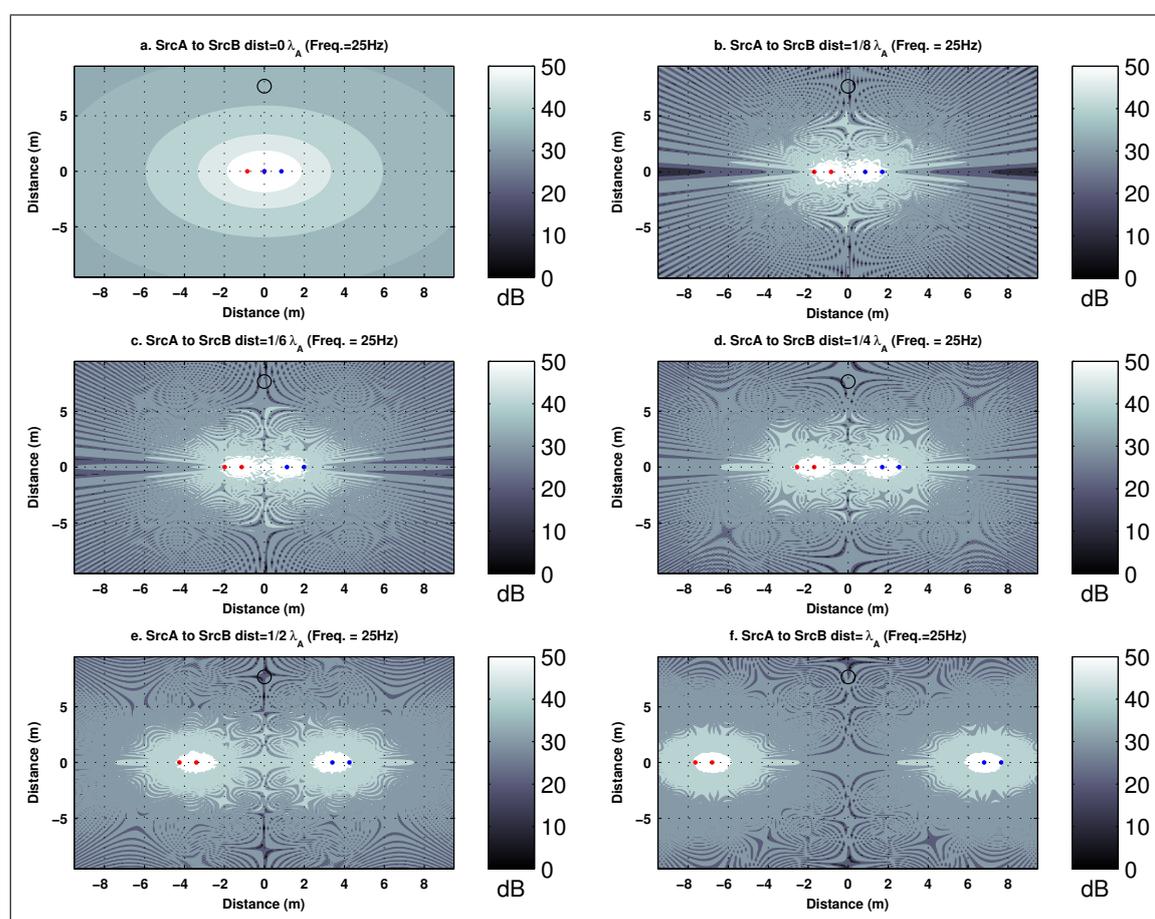


Figure 5.3.: Sound field – correlated sources ($Q_A = Q_B, f_A = f_B$)

Referring to Figure 5.3a, where the sources are at the exact same location, the sound field appears as if there is only one source, which is as expected for two fully coherent sources at exactly the same location (i.e. this can be considered a single source). Once

the sources are moved apart, refer to Figures 5.3b-f, the interaction between the sources can be seen, resulting in an interference pattern in the sound field.

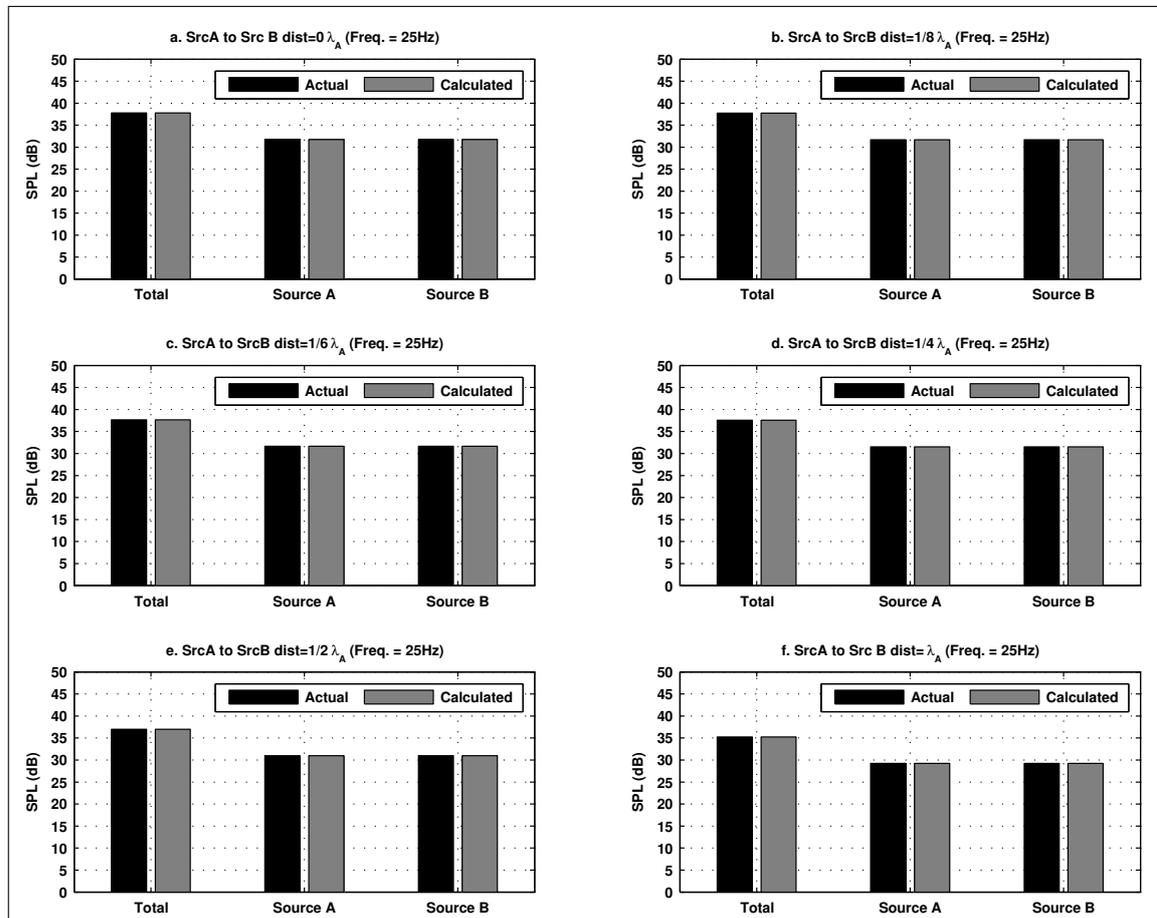


Figure 5.4.: Source contribution – correlated sources ($Q_A = Q_B, f_A = f_B$)

From Figure 5.4, note that the contributions of *Source A* and *Source B* are the same for all configurations, and that the calculated source contribution agrees with the actual source contributions. The results are as expected, however, this case may be considered as trivial in the sense that cross-talk between the reference MPs would not be identifiable with two identical (fully correlated) sources.

Therefore, a supplementary study, similar to the one just described, was setup for further investigation of the OPA method. In this study, the amplitude of *Source B* was halved, so that $Q_A = 2Q_B$. The results are displayed in Figures 5.5 and 5.6.

Referring to the source contribution plots in Figure 5.6, observe that, as expected, when the sources are at the same position (Figure 5.6a), the OPA algorithm can not distinguish between the sources, thus, the results indicate that each source exhibits an equal contribution towards the overall sound level at the response position.

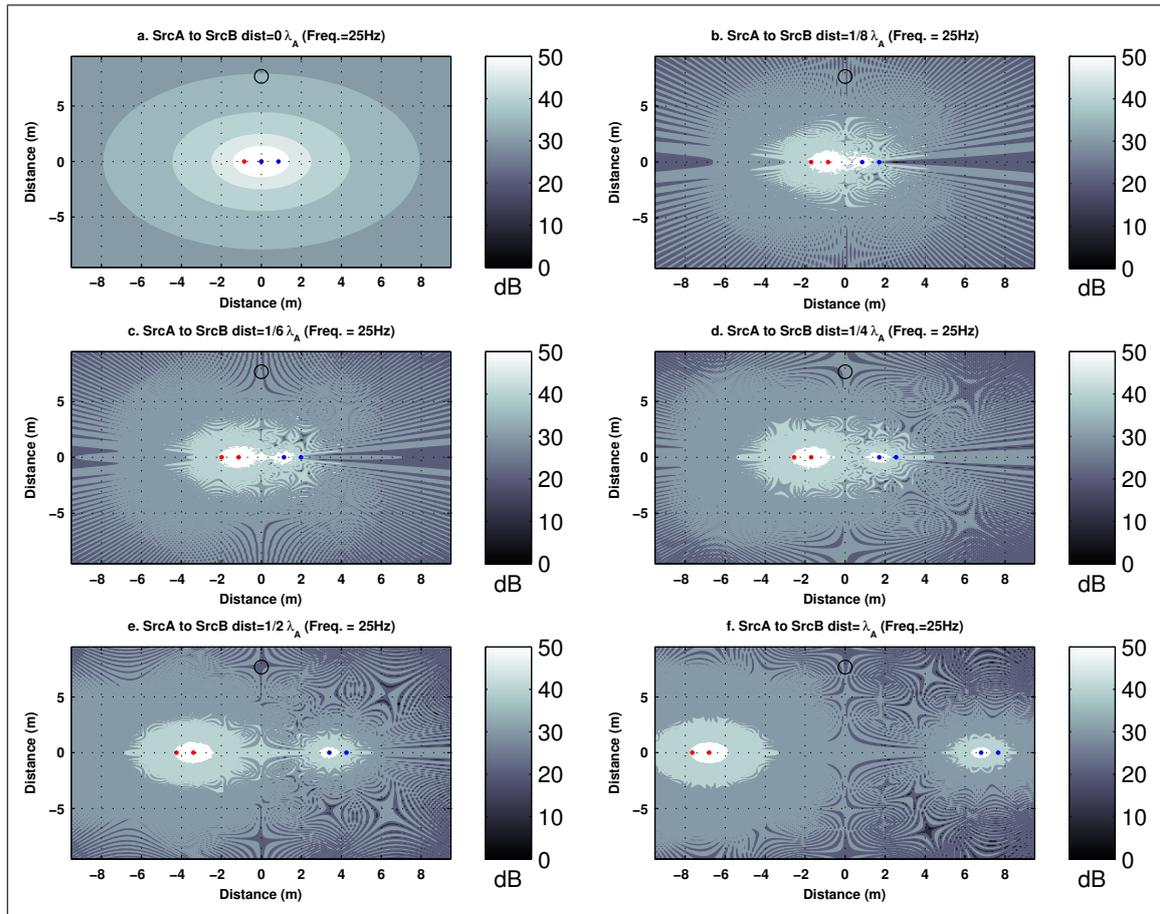


Figure 5.5.: Sound field – correlated sources ($Q_A = 2Q_B, f_A = f_B$)

Once the sources are moved apart, the overall response synthesis remains accurate, however, the source contribution estimates are not accurate – the stronger source strength is over estimated, and the weaker source is under estimated. Referring to Figure 5.6c-f, note that as the sources are moved further apart, the cross-talk between the reference MPs is reduced and the source contribution estimates begin to agree more with the actual source contributions (although the results remain biased towards the stronger source). At a distance of $\frac{\lambda_A}{4}$, the source contribution estimate is accurate to within a few dB (refer to Figure 5.6d). The source contribution estimate is accurate only because of the reduction in cross-talk, since the amplitude of the cross-talk from *Source A* is reduced by 6 dB for each doubling of distance. Note that the calculated total contribution matches the actual total contribution for all cases.

This result indicates that the reference MPs should be placed as close as possible to the source – thus improving the signal-to-noise ratio (i.e. reducing the effects of cross-talk) and improving the calculated results.

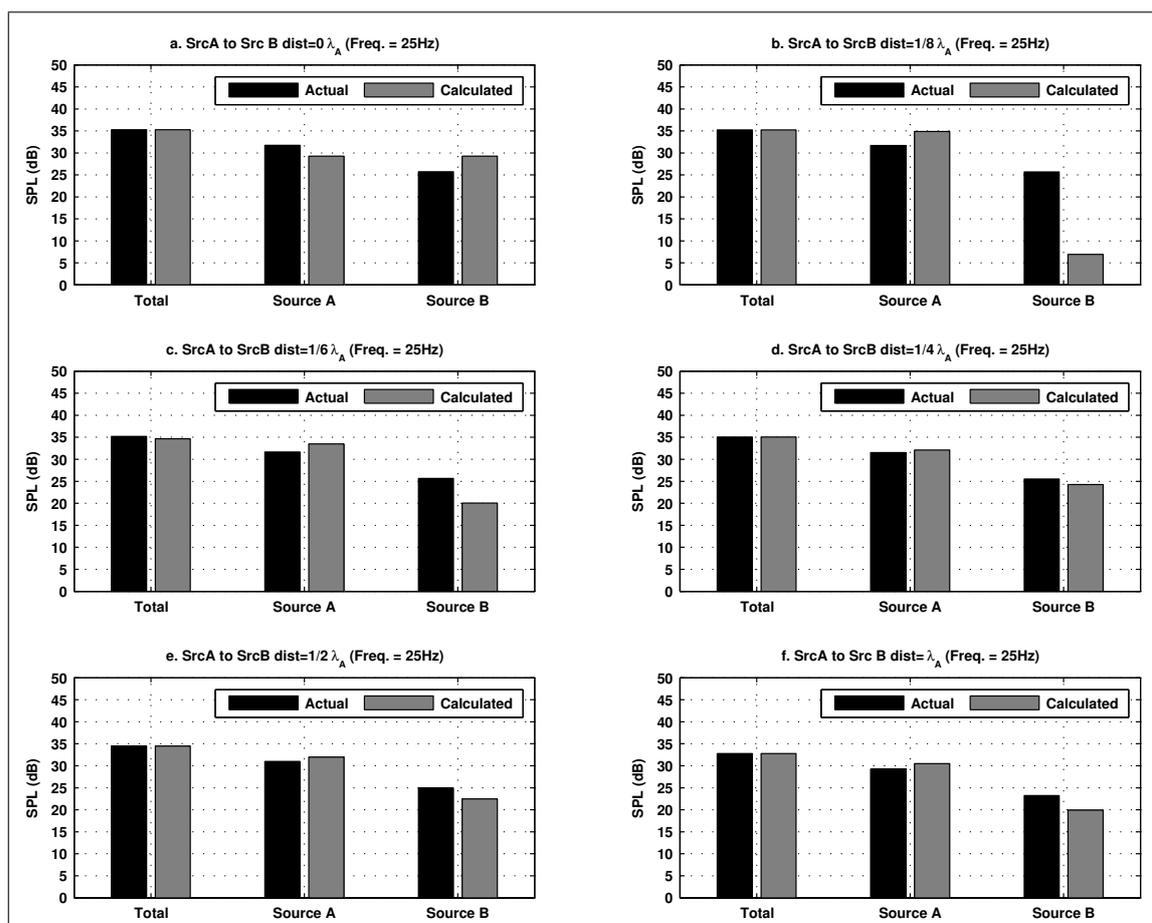


Figure 5.6.: Source contribution – correlated sources ($Q_A = 2Q_B$, $f_A = f_B$)

Even when the two sources are very close (refer to Figure 5.6b), the conclusions that would be drawn from the OPA results are useful – although the calculated source contribution is not accurate (the synthesized source contribution deviates from the actual contribution by approximately 3 dB for *Source A* and by approximately 18 dB for *Source B*), the results clearly indicate which source is the major contributor to the overall sound level at the response position¹.

¹The CTC is not effective for fully correlated data, therefore it is expected that the same conclusions would be drawn from the unprocessed data of the sound level measurements.

5.3.2. Varying Distance Between Two Uncorrelated Sources

This case study involves two spherical sources – *Source A* and *Source B*, which radiate different tones, 25 Hz and 50 Hz, respectively, at approximately the same amplitude. This scenario represents two uncorrelated sources, and therefore there should be no interference in the discrete sound fields at these tones. Referring to the plots of the sound fields, shown in Figures 5.7 and 5.8, the results are as expected.

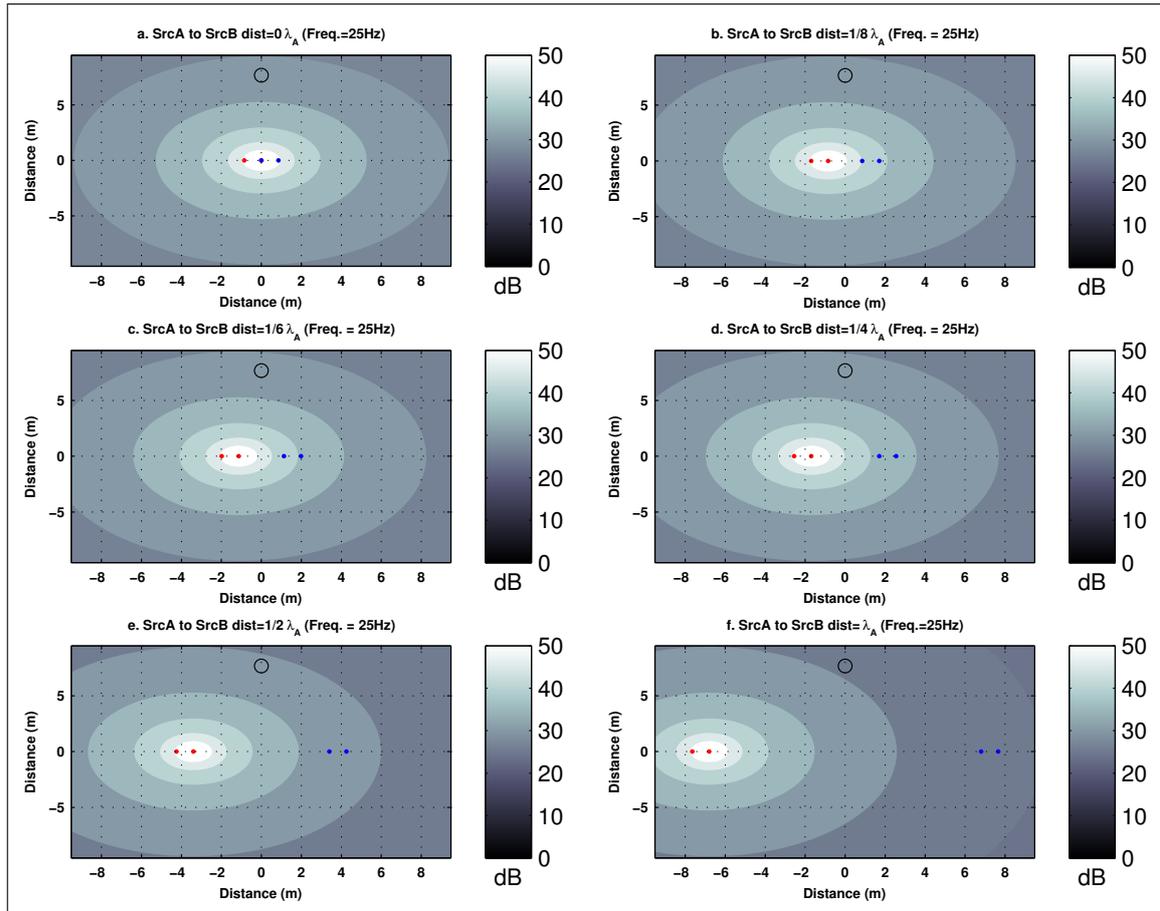


Figure 5.7.: Sound field at f_A – uncorrelated sources ($f_A \neq f_B$)

The calculated source contributions are included in Figures 5.9 and 5.10. Note that the source contribution results are similar to the case with two correlated sources – That is to say that when both sources are at the same location (and therefore the reference MPs are equidistant from the radiating source), the contribution synthesis indicates that both sources are contributing equally to the overall sound level at the response position. This, of course, is incorrect, however, as the distance between the two sources is increased, the response synthesis becomes increasingly accurate. Note that at a separation distance of only $1/8\lambda_A$ (refer to Figures 5.9a and 5.10a), the response synthesis

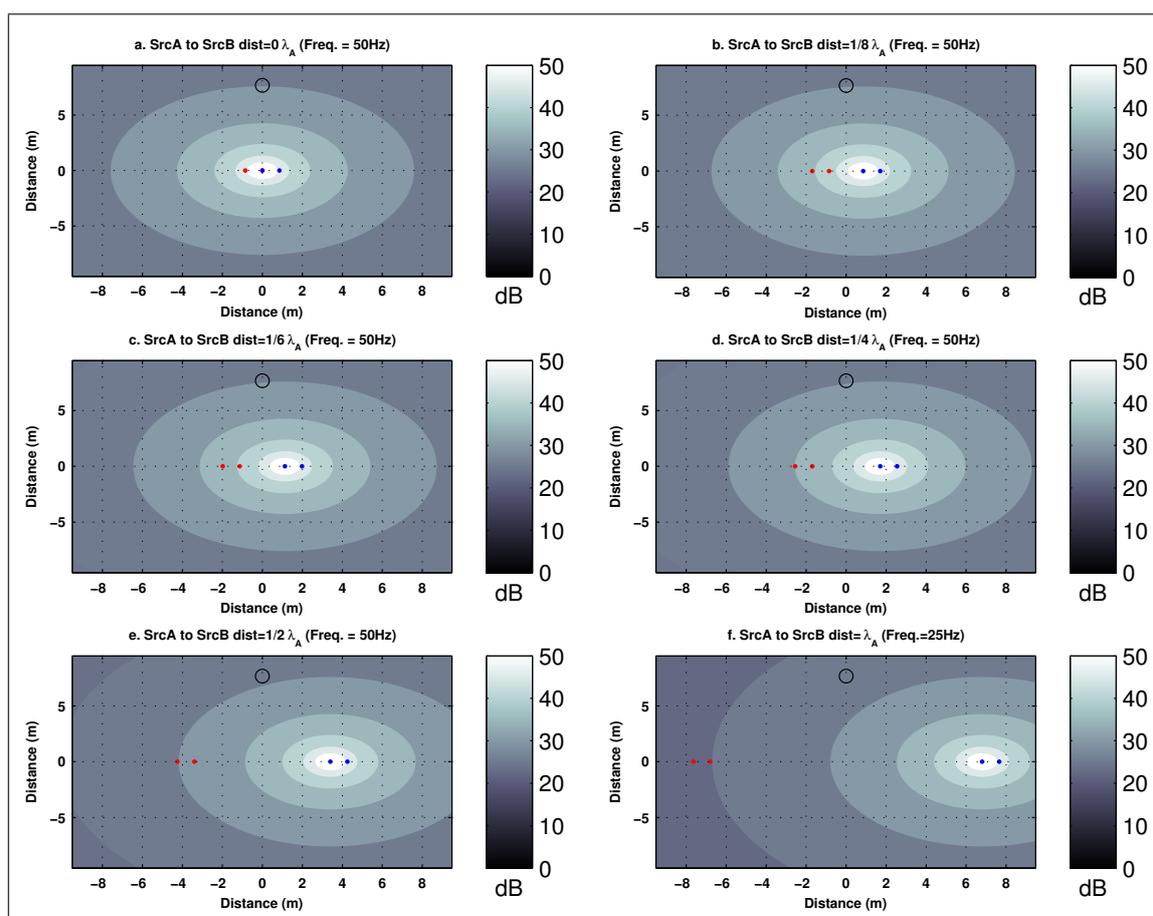


Figure 5.8.: Sound field at f_B – uncorrelated sources ($f_A \neq f_B$)

is already fairly accurate.

Cross-referencing the sound field plots (Figures 5.7 and 5.8) with the source contribution prediction results (Figures 5.9 and 5.10), we see that the incorrect contribution is simply related to the amplitude of the cross-talk at each particular distance.

Because the sources are only radiating a single tone, the reference MPs, regardless of the source, pick up this signal – when introduced into the OTPA, this is of course assumed to be coming from the source that the reference MP is assigned to. Further, the CTC algorithm yields a very strong first principal component which accounts for the the majority of the contribution towards the total sound pressure level at the response MP.

This case is analogous to a system which has two correlated sources as input to the OTPA algorithm. Thus, it is not possible to effectively cancel the cross-talk. To eliminate the cross-talk, one possibility is to filter the measurement data, however this would require more information about the radiated sound from sources.

5.3. OSPA of Two Airborne Noise Sources

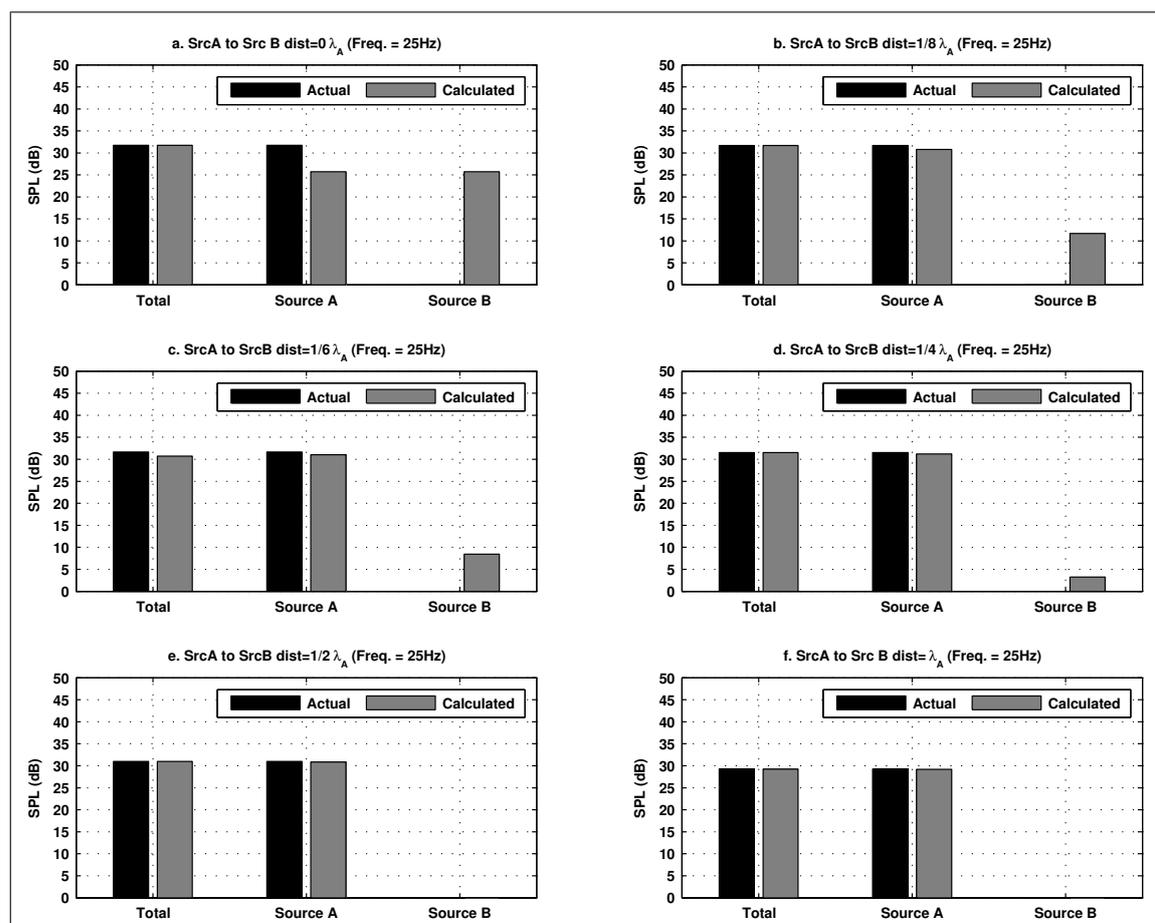


Figure 5.9.: Source contribution at f_A – uncorrelated sources ($f_A \neq f_B$)

Although it has been established that some errors may arise if airborne sources are within a close proximity, it has been shown that even at very close distances (e.g. Figures 5.9b and 5.10b), the results do give an accurate indication of which source is the main contributor. The analysis of these results could therefore lead to the correct strategy for any noise attenuation measures.

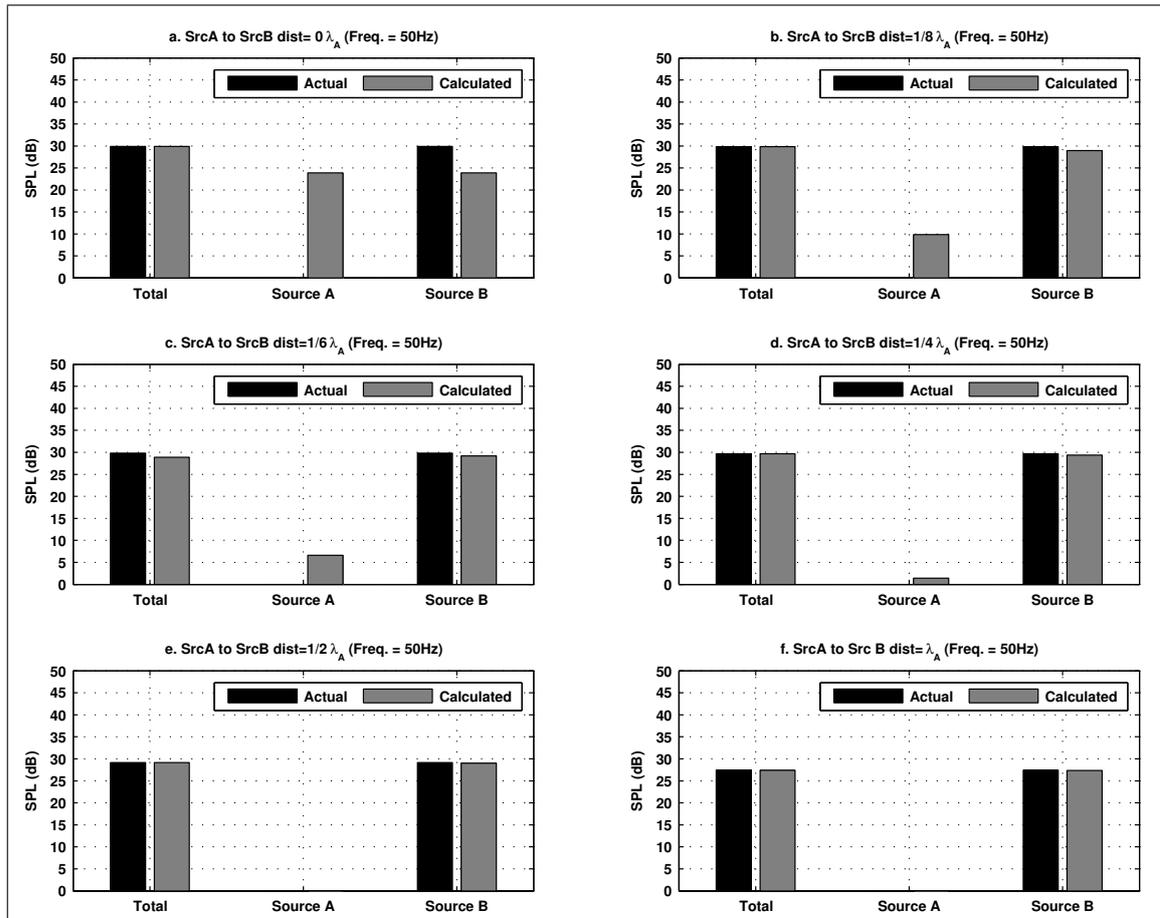


Figure 5.10.: Source contribution at f_B – unrelated sources ($f_A \neq f_B$)

5.3.3. Discussion and Summary of Results

The results of the two case studies presented in this sub-section lead to the following conclusions:

- The sound field behaves according to what is expected, therefore the VTS implementation is considered to be valid.
- The results of the case study with two correlated sources (i.e. the same frequency and phase for *Source A* and *Source B*) indicate that the reference MPs must mainly measure the noise radiated from the source for which they are assigned (i.e. good signal to noise ratio) in order for the OTPA to compute an accurate source contribution. Further, the results indicate that for two strongly correlated sources/-paths, the source contribution results will likely yield an over-estimation of the stronger source strength, and under estimation of the weaker source strength.
- The results of the case study with two uncorrelated sources (i.e. different tones for *Source A* and *Source B*) indicate that the contribution analysis results could be misleading if there is strong cross-talk to a reference MP for which its source is not radiating sound at this frequency (since the cross-talk would be correlated with the other source). However, at a certain signal to noise ratio, the results indicate which source is the greatest contributor, thus leading to a correct conclusion.

5.4. OTPA of Correlated Structure-Borne and Airborne Noise Sources

Recall the tractor OTPA case study presented in Section 1.1 as the background to this thesis. The OTPA consisted of several reference MPs, both for airborne sources (using microphones) and for structure-borne sources (using accelerometers). The over-prediction of the airborne sound contribution (and under-prediction of the structure-borne sound contribution) was corrected by excluding data from the six non-critical airborne MPs from the analysis.

In an attempt to recreate the over-prediction of the airborne sound contribution, the VTS was slightly modified by including one airborne sound (ABS) source/path and one structure-borne sound (SBS) source/path.

Recall from Section 2.2.2 that if an OTPA measurement setup includes mixed units, such as sound pressure (Pa) and vibration (e.g. acceleration, m/s^2), the units must be normalized prior to the computations. In the VTS, the simulated reference and response data is normalized by scaling the SBS to be consistent with the ABS. To preserve a simple model, the SBS path is assumed to be correlated with the ABS path such that the two paths comprise of the same normalized amplitude and phase contribution at both the reference MP and the response MP.

5.4.1. Effects of Varying Distance of Additional Airborne Source MPs

This first simulation consists of a simplified setup of the case study presented in Section 1.1. The OTPA is conducted on a VTS that includes one ABS path and one SBS path. Additionally, a falsely identified ABS source is also included in the model – that is to say, an ABS source that does not radiate sound is included, simulating, for example, the tire reference MPs during a neutral run-up measurement.

Two primary reference MPs are used for the identified paths – one for the SBS path and one for the actual ABS path. An additional ABS reference MP is included in the model at varying distance from the airborne source. This additional reference MP represents the falsely identified ABS source. Thus, a total of three reference MPs (two for ABS and one for SBS) and one response MP are included in the analysis. The VTS layout is displayed along with the sound field in Figure 5.11.

The actual and calculated source contributions at the response position are displayed in Figure 5.12. Referring to Figure 5.12a, observe that there is an over-prediction of the ABS, and under-prediction of SBS, while the calculated total contribution agrees with the actual contribution. This result coincides with the OTPA results for the tractor OTPA case study.

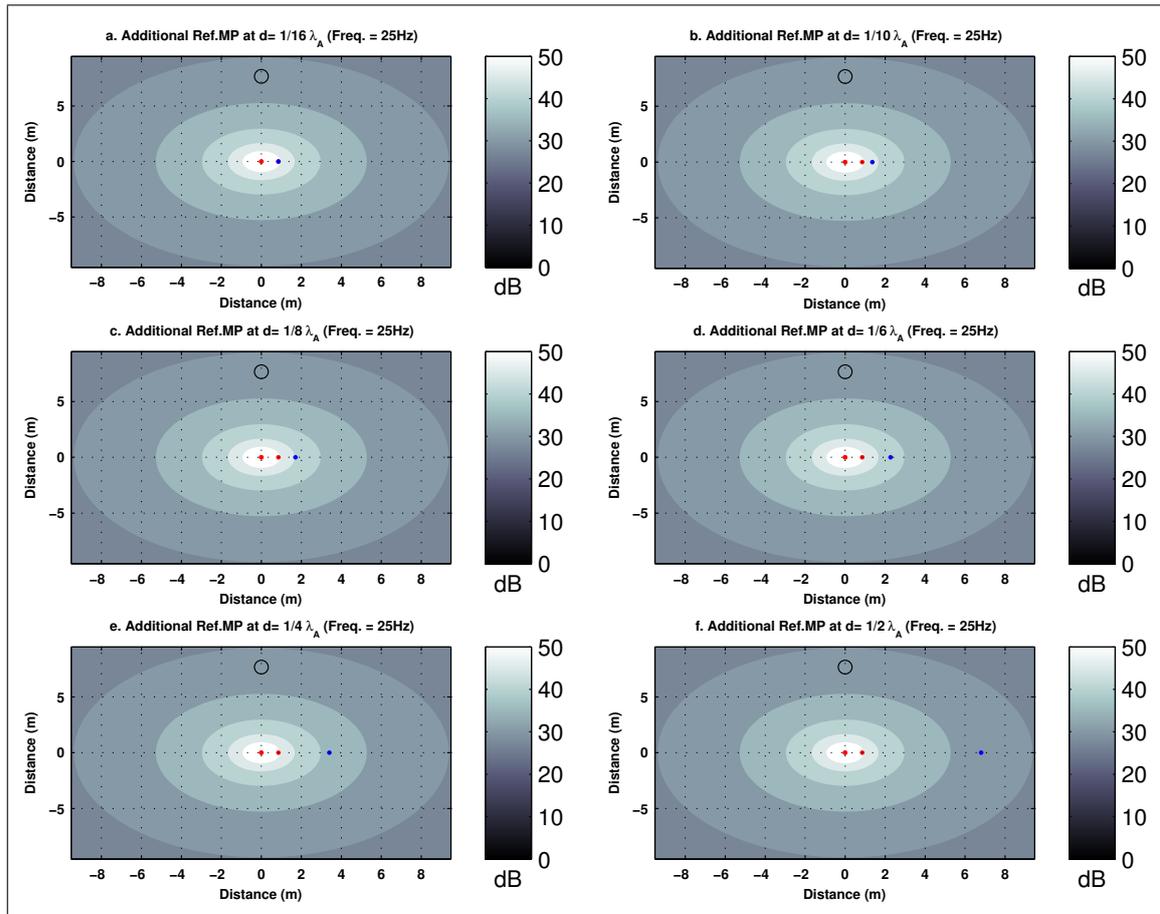


Figure 5.11.: VTS layout – varying additional ABS MP distance

Referring to Figures 5.12b and c, we see that there is an under-prediction of the ABS and SBS contributions, as well as the total response at the receiver. This is due to destructive addition of the ABS contributions caused by the phase relationship between the the two ABS reference MPs and the response position. The difference in distance between the reference MPs and the response position is $\frac{\lambda_A}{6}$. This difference in distance yields a phase difference of approximately $\frac{\pi}{3}$, leading to a slight reduction in the calculated response at the receiver¹.

Once the additional ABS reference MP is at a certain distance from the actual ABS source (e.g. at a distance greater than $\frac{\lambda_A}{6}$, refer to Figures 5.12d-f), the sound pressure at this point is lower than 10dB below the sound pressure at the primary ABS reference MP, and therefore the overall results are no longer influenced by including the additional reference MP in the analysis.

¹For the OTPA algorithm, it would appear as though there are two correlated sources, with only slightly different amplitudes and a phase offset of approximately $\frac{\pi}{3}$.

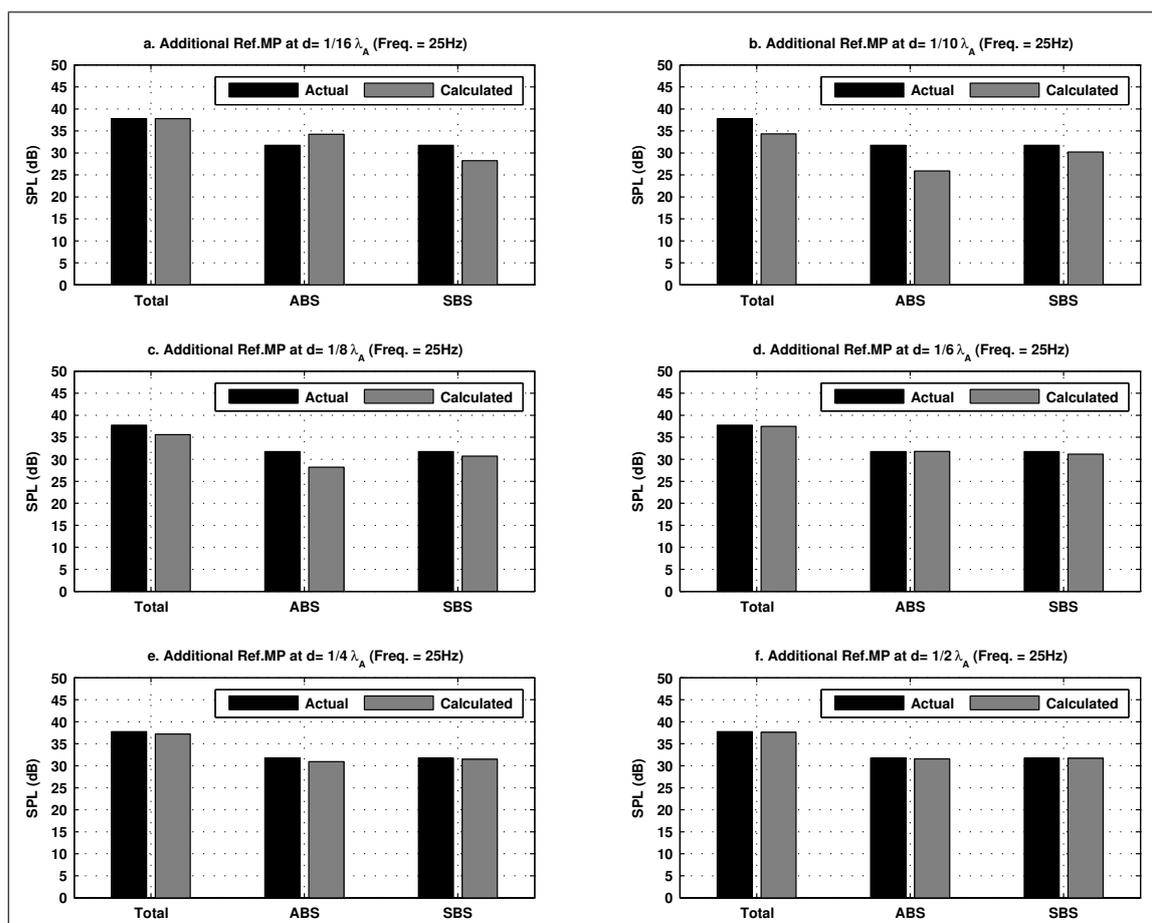


Figure 5.12.: Source contribution – varying additional ABS MP distance

In summary, the results of this study indicate that if an ABS reference MP does not measure what it is intended to, for example, if the ABS source is not actually a strong source compared to other nearby sources, the cross-talk from the other sources will influence the results. The calculated ABS contribution will either be over-predicted, thus taking energy away from the calculated SBS contribution (refer to Figure 5.12a), or the phase offset between the reference MP and the additional reference MP will cause destructive summation, leading to an under-prediction of the calculated ABS and total contribution results.

For the latter case, an error in the OTPA will be clearly identifiable upon comparison of the response synthesis to the measured response at the receiver – the two will not match. For the former case, the error in the contribution predictions will not be so clear since the overall response synthesis will match the measured response.

5.4.2. Varying The Number of Reference MPs

The tractor OTPA case study presented in Section 1.1, displayed an over-prediction of the airborne sound by approximately 5 dB. When six of the twelve ABS reference MPs were discarded from the analysis, the SBS contribution was increased and the ABS contribution was decreased, making the results appear more reasonable.

The purpose of this study is to test the effects of change in the number of ABS reference MPs used for the analysis.

The VTS in this study includes one ABS path, and one SBS path. The ABS and SBS are fully correlated. The SBS includes one reference MP, while the number of reference MPs for the ABS varies from one to six. The VTS layout is presented in Figure 5.13.

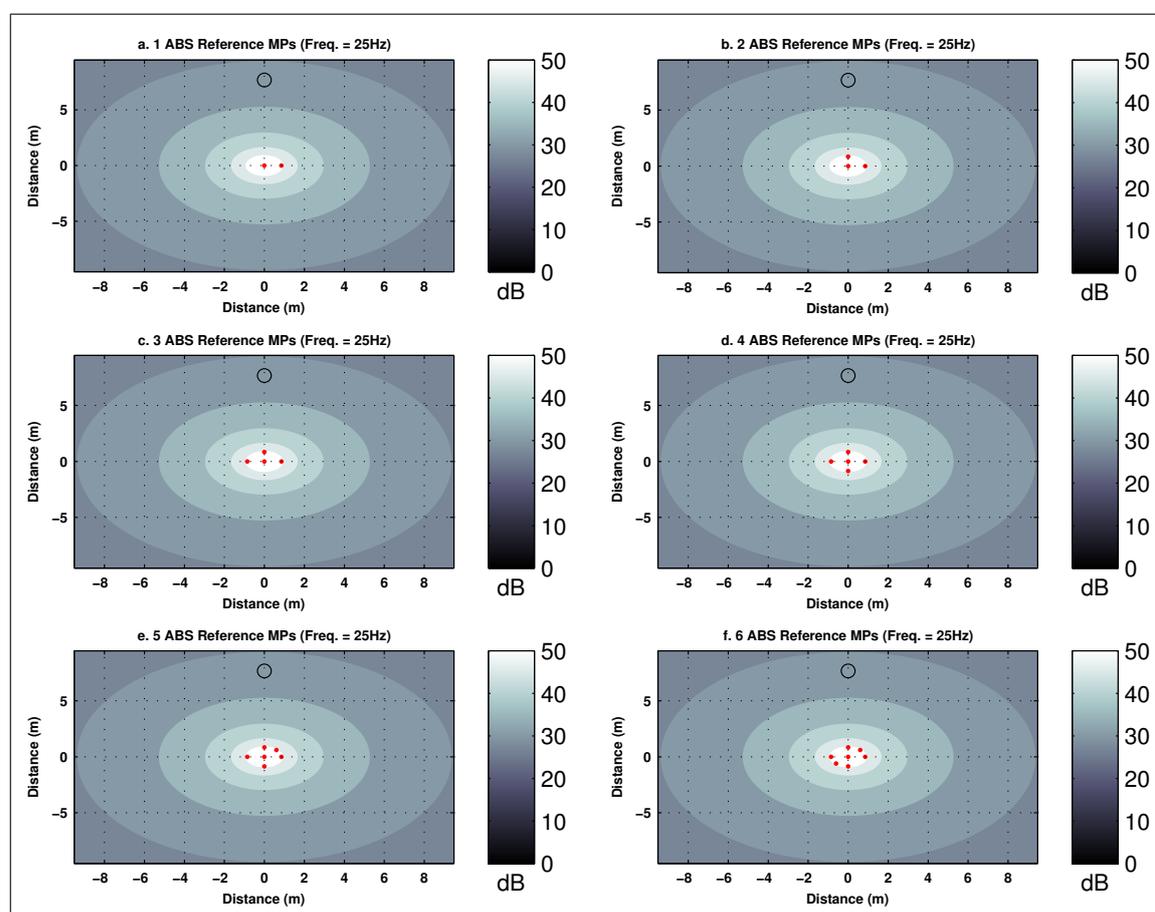


Figure 5.13.: VTS layout – varying number of ABS reference MPs

The results of the contribution analysis are presented in Figure 5.14. It is clear that as the number of reference MPs for the airborne source increases, the calculated contribution is biased towards the airborne source. This bias seems to occur because of the way the

OTPA is conducted – The SVD predicts that the contribution of the correlated source is equally distributed throughout all of the reference MPs. However, to characterize the contribution of the airborne source, the response at the reference MPs are summed, thus yielding a higher calculated contribution for the ABS (which has more reference MPs).

It can be seen that the contribution for the ABS and SBS is proportional to the number of reference MPs used – That is for a case where the correlated paths are represented by two ABS MPs, and one SBS MP, two-thirds of the energy would be predicted to be coming from the ABS, and one-third of the energy from the SBS.

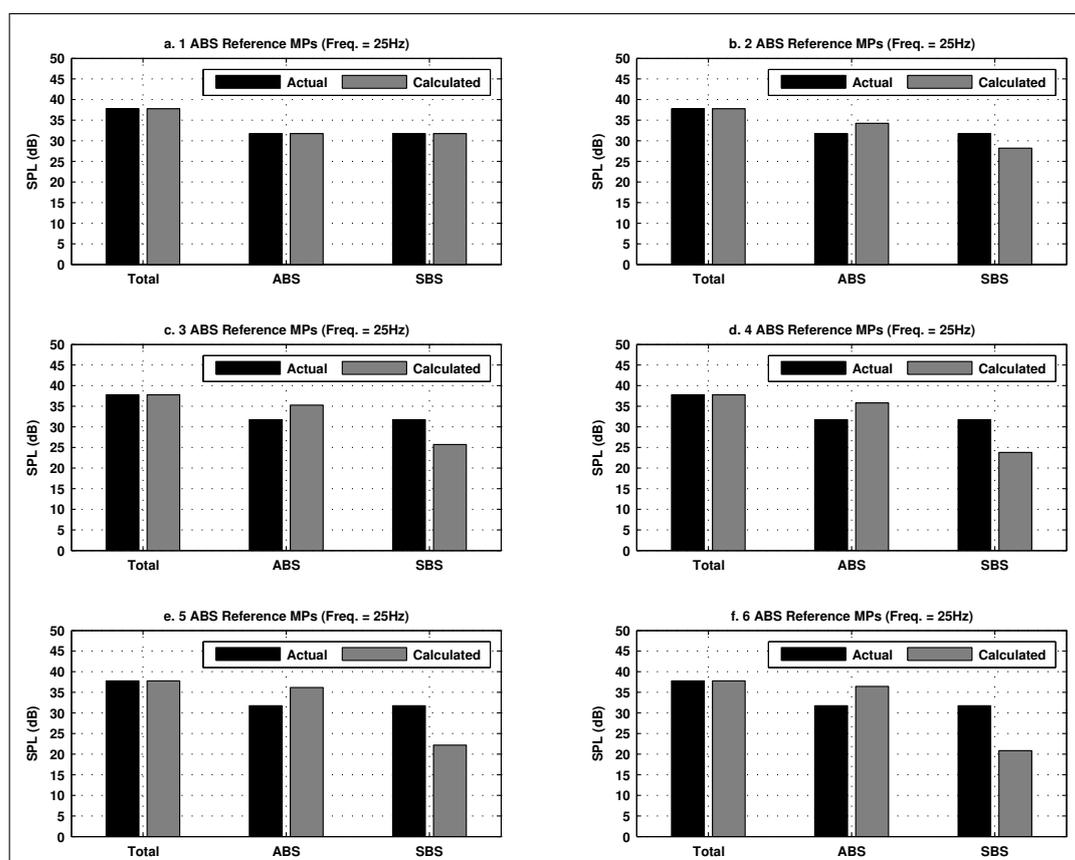


Figure 5.14.: Source contribution – varying number of ABS reference MPs ($L_{SBS} = L_{ABS}$)

To further demonstrate this theory, the analysis is repeated, however this time the SBS contribution is increased so that it is twice that of the ABS (i.e. an increase of 6 dB). The source contribution results are presented in Figure 5.15.

The accurate prediction for this case now occurs when there are two ABS reference MPs (Referring to Figure 5.15b). In fact, there is an over-prediction of the SBS when only one ABS MP is included (refer to Figure 5.15a). As soon as more than two ABS reference

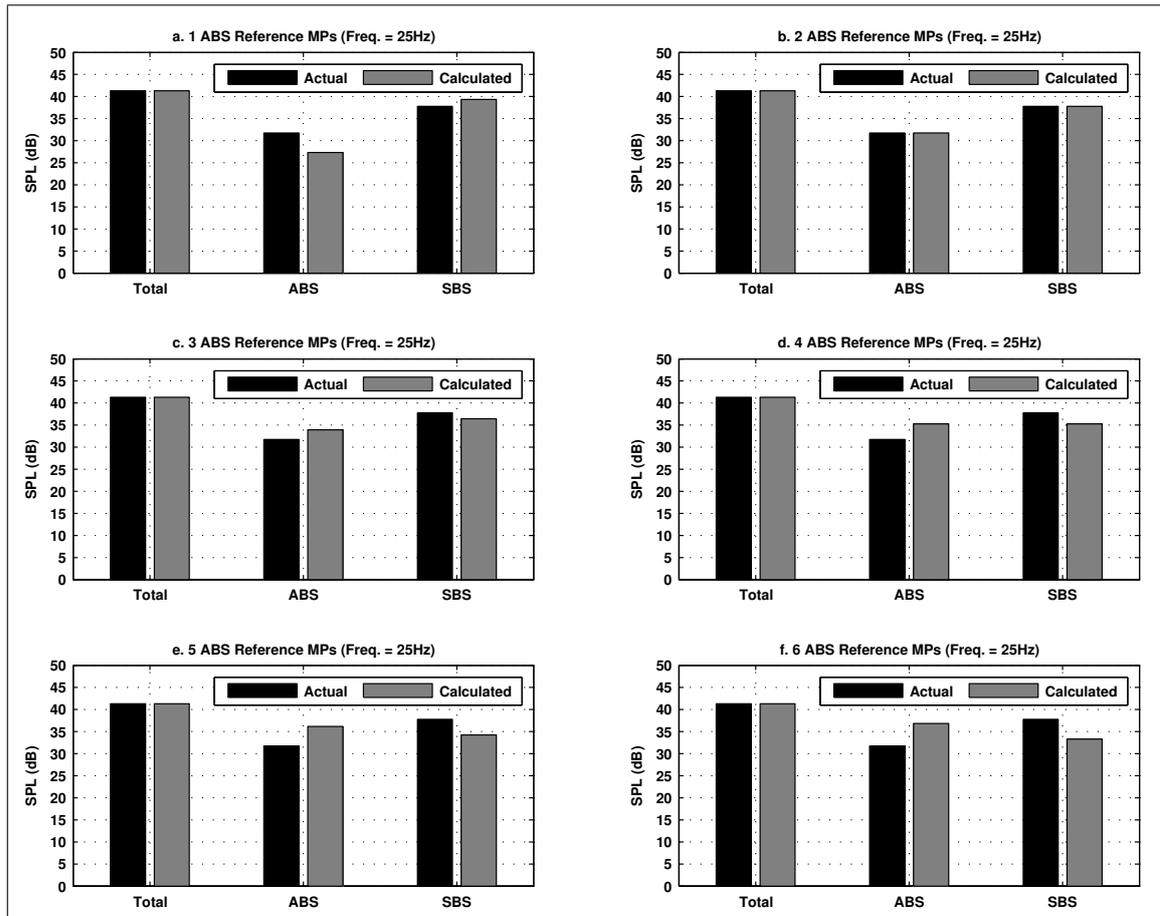


Figure 5.15.: Source contribution – varying number of ABS reference MPs ($L_{SBS} = 2L_{ABS}$)

MPs are included, an over-prediction of the ABS contribution (and under-predictions of SBS contribution) occurs again.

These results are an indication that the number of ABS reference MPs used for a source influence the results if there is a strong contribution from a correlated SBS – to avoid this error, the ABS reference MPs for a distinct source that are measuring the same amplitude and phase could be combined and included in the input matrix, $\mathbf{X}(j\omega)$, as one vector of data prior to conducting the SVD computations (thus the results would appear as shown in Figure 5.14a) . Note that the results still may not be exact, as seen in the case with a higher SBS contribution, refer to Figure 5.15a.

Further, observe that results in Figure 5.15a also agree with the results in Section 5.3.1 (Figure 5.6), where it was concluded that for correlated sources, the OTPA method yields an over-prediction of the stronger contribution, and under-prediction of the weaker contribution.

The results also concur with the conclusion from Section 5.4 where the ABS reference MPs included in the analysis that was not actually measuring a strong source picked up cross-talk from nearby sources (which is a correlated signal to the reference MP for the other source). This, in turn, leads to a bias in the calculated source contribution. The results indicate that the more input data from invalid ABS reference MPs included in the analysis, the greater the error in the calculated source contribution. Note that the total source contribution remains accurate for all the analysis cases (refer to Figure 5.14), and it is only the contribution predictions of the ABS and SBS that shift.

In conclusion, it is clear that the OTPA method is sensitive to the measurement setup – the results may be biased towards sources that include more reference MPs, whether the reference MPs were intended for that source or not.

5.4.3. Discussion and Summary of Results

The findings derived from the VTS including an airborne source and a structure-borne source are summarized as follows:

1. The over-prediction of the airborne source seen in the actual measurements has been successfully recreated in the VTS.
2. The analyses indicate that the number of ABS reference MPs used to measure a source may bias the results of contribution analysis towards the ABS if a strong and highly correlated contribution from the SBS is also present.
3. Cross-talk from strong sources that is not properly cancelled in the analysis will lead to erroneous results. To improve the signal-to-noise ratio at the reference MPs, and to decrease the effects of cross-talk, the reference MPs should be placed as close as possible to the sources.
4. The ABS reference MPs for a particular source that are measuring the same amplitude and phase should be combined and included in the OTPA computation as a single data vector for each source to avoid error in the contribution prediction results.
5. The OTPA contribution analysis results may not be completely accurate, however, if the analysis is set up correctly, the results will lead to the correct conclusions.

In the VTS, the airborne sources are small compared to the wavelength of the sound that is being radiated to the surrounding. In this low frequency range, the size of the source is much smaller than one-sixth wavelength ($k_A \frac{\lambda_A}{40} = k_A a \ll 1$), and therefore the source radiates as a simple point source. For this case, the radiation from the source will be uniform regardless of the shape of the source. At a certain frequency, where $k_A a > 1$, the size of the source is larger than one-sixth of the wavelength, and the radia-

tion from the source will no longer be uniform. In this frequency range, the source can be seen as a combination of multiple sources – thus, the source will exhibit directivity in the radiation pattern [2].

Because of this physical characteristic, to properly measure an airborne sound source (e.g. a vehicle engine) in the high frequency range ($ka > 1$), several reference MPs would be required, whereas in the low frequency range ($ka \ll 1$), a single reference MP would be sufficient.

Based on conclusion number 4 (see above), and the physics behind sound radiation in the two frequency ranges, it's clear that a methodology is required to be able to characterize the source, while not influencing any bias in the OTPA. Two possible methods come to mind:

The first, is a simple average of all the ABS reference MPs for each particular airborne source, and including the data as a single vector in the input matrix. However, using this method the phase information may be lost.

The second method involves dividing the ABS into a low frequency component and a high frequency component. This way, the filtered data from certain reference MPs can be excluded or included from the OTPA, depending on the frequency range that they are assigned to measure. The data could then be combined and included in the OTPA as a single input vector (or two input vectors – one for high frequency, one for low frequency).

6. Summary and Conclusions

The objective of this thesis work was to study the OTPA method and investigate the potential causes of the over-prediction of airborne sound contribution that has occurred in some MBBMS projects. The OTPA investigation was conducted using a simplified *virtual test setup* (VTS), consisting of two sources and one receiver, programmed using MATLAB.

The VTS and OTPA programs were validated using a simple test setup consisting of two airborne sources. Further, the CTC algorithm, which uses SVD and PCA, was validated by removing noise introduced to the input signals.

The analysis using the VTS commenced with a study of two airborne sources – first as fully correlated sources, then as fully uncorrelated sources. Next, a study of the effects of including a "falsely identified source" and a correlated structure-borne sound path in the VTS was conducted. The study also included the effects of distance between the additional MPs (i.e. associated with the falsely identified source) and the number of MPs for a particular source.

Subsequently, the result of the tractor OTPA case study presented in Section 1.1 was successfully recreated using the VTS. By recreating the real-life problem in the VTS, it is assumed that the conclusions drawn from the VTS also apply to actual OTPA.

Several conclusions have been drawn from the OTPA study using the VTS. These conclusions, which are relevant mainly to OTPA applied to airborne sound, are summarized in the following:

1. For correlated airborne sources, in order for the source contribution prediction to be accurate, the dominant sound pressure level at the reference MPs must be from the source for which they are assigned (i.e. a good signal to noise ratio is required). To improve the signal-to-noise ratio, reference MPs should be placed as close as allowable to the source which they are assigned to represent.
2. Reference MPs associated with non-critical sources (i.e. falsely identified sources) will lead to incorrect source contribution results. The error is sometimes apparent when comparing the overall source synthesis to the actual response measurement, however, in some cases it may not be so clear.

3. The number of reference MPs associated with a source has an effect on the source contribution prediction results if the ABS and SBS sources/paths are highly correlated and at similar amplitudes – the source contribution prediction results are biased towards the source that has more reference MPs (in this case the ABS), because of the way the energy is distributed among the reference MPs of the correlated sources/paths. To avoid this bias, the reference MPs for an individual source/path that measures the same amplitude and phase should be combined and included in the OTPA input matrix as a single data vector. Two methods have been suggested to handle this: The first, involves simply averaging the data of all the reference MPs (however loss of phase information may occur). The second proposed method involves assigning the reference MPs according to frequency range.
4. Even for an OTPA using a correct setup, the source/path contribution results may not necessarily be completely accurate, however, the results will still lead to the correct practical conclusions.

6.1. Suggestions for Future Work

A few aspects of the OTPA were briefly touched upon in this thesis work that would be interesting to study further. These topics are discussed in the following.

This thesis work included a study of very simple correlated and uncorrelated sources radiating a single tone. The SVD is only able to separate uncorrelated, or partly correlated data. It would be interesting to study what the upper limit of correlation between data sets is for the SVD and PCA algorithm to effectively separate sources/paths. It would also be interesting to take this study further towards a more realistic VTS by using geometry and source signals that better represent an actual vehicle.

Further, it could be beneficial to develop an algorithm to recognize if there is significant cross-talk between the ABS reference MPs that will not be separable by using SVD (i.e. highly correlated sources/paths). This could perhaps be done, for example, by examining the coherence, relative magnitude and phase between the data for reference MPs. Certain reference MPs could then be excluded from the analysis computations, particularly in the low frequency range.

Once an ABS reference MP selection algorithm has successfully implemented in the VTS, it would, of course, be interesting to take the study further and use a real test setup.

Finally, there are indications of advantages in conducting the TPA and OTPA of vehicle pass-by in the time domain [25]. A study of using the OTPA in the time domain, particularly for airborne sources, may yield improved results.

References

- [1] J.S. Bendat. Solutions for the multiple input/output problem. *Journal of Sound and Vibration* 44(3):311-325, 1976.
- [2] L.L. Beranek. *Acoustics*. Acoustical Society of America, New York, 1993.
- [3] H. Brick. *Application of the boundary element method to combustion noise and half-space problems*. Ph.D. Thesis, Chalmers University of Technology, 2009.
- [4] L. Cremer et al. *Structure-Borne Sound*. Springer, Berlin, 2005.
- [5] D. de Klerk, et al. Application of operational transfer path analysis on a classic car. *Müller-BBM VAS Special Report for DAGA 2009*. Accessed from www.muellerbbm-vas.com on 2012.02.01.
- [6] D. de Klerk. *Dynamic response characterization of complex systems through operational identification and dynamic substructuring*. Ph.D. Thesis, Delft University of Technology, 2009.
- [7] D. de Klerk and A. Ossipov. Operational transfer path analysis: Theory, guidelines and tire noise application. *Journal of Mechanical Systems and Signal Processing* 24:1950-1962, 2010.
- [8] G. de Sitter, et al. Operational transfer path analysis. *Journal of Mechanical Systems and Signal Processing* 24:416-431, 2010.
- [9] P. Gajdatsy, et al. Critical assessment of operational path analysis: Effect of coupling between input paths. *Proceedings from Acoustics '08 Paris* 5821-5826, 2008.
- [10] P. Gajdatsy, et al. Critical assessment of operational path analysis: Mathematical problems of transmissibility estimation. *Proceedings from Acoustics '08 Paris* 5463-5468, 2008.
- [11] P. Gajdatsy, et al. Application of the transmissibility concept in transfer path analysis. *Journal of Mechanical Systems and Signal Processing* 24:1963-1976, 2010.

- [12] J.E. Jackson. *A User's Guide to Principal Components*. John Wiley & Sons Inc., New York, 1991.
- [13] I.T. Jolliffe. *Principal Component Analysis, 2nd ed.* Springer-Verlag, New York, 2002.
- [14] K. Janssens et al. Operational path analysis: A critical review. *LMS International NV Technical Report*, 2008. Accessed from www.lmsintl.com on 2012.02.01.
- [15] Müller-BBM Vibro-Acoustik Systeme. PAK v5.6. computer software. Müller-BBM VAS, 2012.
- [16] K. Noumura and J. Yoshida. Method of transfer path analysis for interior vehicle sound by actual measurement. *Japan Society of Automotive Engineers, Proceedings of The Annual Congress 55(06):7-12*, 2006.
- [17] P.E.F. Padhila and J.R. de França Arruda. Comparison of estimation techniques for vibro-acoustic transfer path analysis. *Journal of Shock and Vibration* 13(4):459-467, 2006.
- [18] J. Plunt. Finding and fixing vehicle NVH problems with transfer path analysis. *Sound and Vibration* 39:12-17, 2005.
- [19] W. H. Press, et al. *Numerical Recipes: The Art of Scientific Computing, 3rd Ed.* 65-75, Cambridge University Press, New York, 2007.
- [20] A. Schuhmacher. Techniques for measuring the vibro-acoustic transfer function. *Sound and Vibration*, March, 2010.
- [21] A. Sontacchi, et al. Transfer path analysis experimental investigations and modelling. *Proceedings of the 3rd congress of the Alps Adria Acoustics Association*, 2007.
- [22] G. Strang. *Linear Algebra and its Applications*. Harcourt Brace Jovanovich, Inc. Orlando, 1988.
- [23] D. Tcherniak and A.P. Schuhmacher. Application of decomposition-based technique in NVH source contribution analysis. *Brüel & Kjær Technical Report for ISMA 2008*. Accessed from www.bksv.com on 2012.02.01.
- [24] D. Tcherniak and A.P. Schuhmacher. Application of transmissibility matrix method to NVH source contribution analyses. *Proceedings of the IMAC-XXVII, Society for Experimental Mechanics*, 2009.
- [25] S. Zheng, et al. Time-domain transfer path analysis of multiple moving noise sources. *Journal of Noise Control Engineering* 59(5):541-548, 2011.

A. MATLAB Script

A.1. MATLAB Script - Two Airborne Noise Sources

A.1.1. OTPA Script (*otpa.m*)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Operational Transfer Path Analysis – Virtual Test Setup           %
%   %                                                                    %
%   Mihkel Toome                                                       %
%   %                                                                    %
%   Description:                                                       %
%   Two sources , distance between sources varied                     %
%   Sources include near field                                       %
%   One reference MP. One response MP.                                 %
%   Results are processed via the OTPA CIC Function.                  %
%   %                                                                    %
%   Functions associated with code (x3):                               %
%   1. MPs and Src/resp positions defined by function "dist.m".      %
%   2. Sources defined in INPUT. Calcs done in function "source.m". %
%   3. Countour plot is setup by function "contourmesh.m".          %
%   %                                                                    %
%   Version: 4.0                                                       %
%   Date: 28 May 2012                                                 %
%   %                                                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tic;
clear all; close all; clc;
global rA rB N Nb f r_AtoB mps beta numMPs
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                    %
%                               INPUT                                  %
%                                                                    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% INPUT: SrcA and SrcB properties , distances between Src's.

FreqA = 25;
sizeA = 1/40;    % Size (radius) of spherical SrcA rel. to lambda A.
Va = 60;
NAfact = 0.2;   % Amplitude of noise (relative to Va)
rA = 1/16;      % distance between SrcA and ref MPs (rel. to lambda A)

FreqB = 50;
beta = 0*pi;    % Assign phase shift for Source B [radians]
sizeB = sizeA;  % Size of spherical SrcB rel. to lambda A.
Vb = (FreqA/FreqB)*Va;
```

A.1. MATLAB Script - Two Airborne Noise Sources

```
NBfact = 0.2;      % Amplitude of noise (relative to Vb)
rB = rA;          % distance between SrcB and ref MPs (rel. to lambda A)

% vector to vary distance between SrcA and SrcB (rel. to lambda A)
r_Atob = [0 1/8 1/6 1/4 1/2 1];

numMPs = 1;      % Define number of MPs for the sources

%% INPUT: PCA Contribution Threshold

% Select PCA method
PCA_METHOD = 'thres'; % 'thres' or 'cumcon' or 'PCs' or 'minPC'

% Input value according to selected PCA method
thres = 5; %Threshold: Minimum PCs % contribution towards PCs score (e.g. %5, thres←=5)
cumcon = 85; % Cumulative Contribution of PCs contributing to % of signal
PCs = 2; % # of principal components to be use in analysis
minPC = 8000; % Minimum magnitude of the PCs to be included

%% INPUT: Turn Figures on/off; print figures on/off.
plotspectrum = 'off'; % Plot spectrum figures 'on'/'off'
plotbar = 'on'; % Contribution bar graph 'on/off'
plotcontour = 'off'; % Plot contour plot of SPL 'on/off'

printfigures = 'off'; % print EPS of figures 'on'/'off'

block = 2; % Which block of data is to be plotted [1<block<Nb]

%% CALCULATIONS

% Function calculates source strength at MPs
tic;
[prefa,prefb,presp,pacont,pbcont] = source(FreqA,FreqB,Va,Vb,NAfact,NBfact,sizeA,←sizeB);
toc;

% FREQUENCY DOMAIN
% Organize ref & resp measurements into matrices X and Y
% Dimensions of X: Rows=Meas.Blocks; Columns=MPs (for "N" Freq. lines and
% "length(rA)" distances between SrcA and MPs).
% X = (Meas.Blocks, MPs, Freq.Components, Src.Distances)
X = [prefa prefb]; % freq domain
% Y = (Meas.Blocks, Freq.Components, Src.Distances)
Y = presp; % freq domain

%% Cross Talk Cancellation & Transfer Function Calculation (Frequency Domain)

for ii=1:length(r_Atob) % Loop for various SrcA-MP distances
    for kk=1:N % Loop for N FFT frequency components
```

```

% Singular Value Decomposition (SVD). N.B. "S" is the matrix of PCs.
[U,S,V] = svd(X(:,:,kk,ii), 'econ'); % use "econ" to ensure S is a square ←
      Matrix.

% Principal Component Analysis (PCA)
% COEFF – The coefficients corresponding to the PCs
% PC – The Principal Components (i.e. the Singular Values)
% CONT – The contribution to the overall signal (%)

% [COEFF, PC, CONT] = pcacov(S); % Matlab function for PCA
% Equivalent calc. for the PCA matlab function above
PC = diag(S);
COEFF = U*S;
CONT = diag(S)/sum(diag(S))*100;

% Find PCs that correspond to PCA method input
if strcmp(PCA_METHOD, 'thres') == 1
    numPC = find(CONT>thres,1,'last');
elseif strcmp(PCA_METHOD, 'cumcon')==1
    numPC = find(sum(CONT)>cumcon,1,'last');
elseif strcmp(PCA_METHOD, 'PCs')==1
    numPC = PCs;
elseif strcmp(PCA_METHOD, 'minPC')==1
    numPC = find(PC>minPC,1,'last');
end

% Calculate "invSr" – The noise reduced inverse of S.
invSr = zeros(size(S)); % Allocate zeros for size of the invSr matrix
S1 = diag(diag(inv(S(1:numPC,1:numPC))));
invSr(1:numPC, 1:numPC) = S1(1:numPC, 1:numPC);

%     Ssave(:,:,kk)=S1;
%     Usave(:,:,kk)=U;
%     Vsave(:,:,kk)=V;

% Calculate noise reduced inverse of X

X_inv(:,:,kk,ii) = V*invSr*U'; % Noise reduced inverse of X

% Transfer function estimate
H(:,:,kk,ii) = X_inv(:,:,kk,ii)*Y(:,:,kk,ii);

end
end
toc; tic;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Calculate Synthesized Response at Receiver (Freq. Domain)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Re-run source: Random noise is redefined and a random amplitude and phase
% shift is introduced to the tones (same for srcA and srcB)
% This simulates a separate measurement for transfer function calculation
% and synthesis calculation.
clear X Y prefA prefB presp pacont pbcont
[prefA,prefB,presp,pacont,pbcont] = source (FreqA,FreqB,Va,Vb,NAfact,NBfact,sizeA,←
      sizeB);
% X = (Meas.Blocks, MPs, Freq.Components, Src.Distances)
X = [prefA prefB]; % freq domain
% Y = (Meas.Blocks, Freq.Components, Src.Distances)

```

```

Y = presp; % freq domain

for ii=1:length(r_AtoB)
    for jj=1:Nb
        for kk=1:N

            % Take transpose of X to rearrange variables for Ys
            Ys(:,jj,kk,ii) = X(jj,:,kk,ii).*H(:,kk,ii)';

            % Sum for total synthesis and source contributions
            % Sum is as if for continuation of matrix multiplication
            % (Ys=XH)
            Ystot(kk,jj,ii) = sum(Ys(:,jj,kk,ii));
            Ys_srcA(kk,jj,ii) = sum(Ys(1:numMPs,jj,kk,ii));
            Ys_srcB(kk,jj,ii) = sum(Ys(numMPs+1:2*numMPs,jj,kk,ii));

        end
    end
end
[prepmeshA,prepmeshB,x,y] = contourmesh (FreqA, FreqB);
toc;

```

A.1.2. Function to Define The Virtual Test Setup Layout (*dist.m*)

```

% Define source & receiver (Reference and Response) locations as
% coordinates (x,y). Distances are multiples of wavelength (lambda=d).
% $$$$ SrcA to SrcB distance is varied $$$$
% r_AtoB - Distance between SrcA and SrcB
% r_AA - Distance between SrcA and SrcA ref MPs
% r_BB - Distance between SrcB and SrcB ref MPs
% r_AB - Distance between SrcA and SrcB ref MPs
% r_BA - Distance between SrcB and SrcA ref MPs
% mps - Matrix of all Src's, resp, and MPs (i.e. for a plot)
% a - Radius of SrcA (1/40*lamda_SrcA)
% b - Radius of SrcB (1/40*lamda_SrcA)
% c - Speed of sound in air (340m/s)

function [r_A,r_B,r_AA,r_BB,r_AB,r_BA] = dist (FreqA,FreqB)
global rA rB c r_AtoB mps d srcA srcB numMPs

c = 340;
lambda = c/FreqA;
d=lambda;

% refB_dist = rB*c/FreqA; % Define distance for Source B reference ↔
% positions
coords = [-d, 0; d,0;0,d;0,-d];

% srcA = [0,0];
% refA = rA*coords; % Ref MPs for SrcA

resp = [0, d/sqrt(pi)]; % Response measurement position (receiver)

```

```

% r_AtoB = sqrt((srcA(1)-srcB(1))^2+(srcA(2)-srcB(2))^2); % SrcA-SrcB distance is d←
/16

for ii=1:length(r_AtoB) % Iteration for each SrcA to SrcB distance

    % Define SrcB position relative to SrcA (centre to centre distance)
    srcB(ii,:) = [d/2*r_AtoB(ii),0];
    srcA(ii,:) = [-d/2*r_AtoB(ii), 0];

    % Define reference MP's for SrcA and SrcB (centre to centre distance)
    for i = 1:numMPs
        refB(i,:,ii) = -rB*coords(i,:) + srcB(ii,:);
        refA(i,:,ii) = rA*coords(i,:) + srcA(ii,:);
    end

    % Assemble src's and response and reference positions into a matrix
    mps(:, :, ii) = [srcA(ii, :); srcB(ii, :); resp; refA(:, :, ii); refB(:, :, ii)];

    % Distance b/w sources and reference MPs
    for i=1:numMPs
        r_AA(i,ii) = sqrt((refA(i,1,ii)-srcA(ii,1))^2 + (refA(i,2,ii)-srcA(ii,2))^2)←
        ;
        r_BB(i,ii) = sqrt((refB(i,1,ii)-srcB(ii,1))^2 + (refB(i,2,ii)-srcB(ii,2))^2)←
        ;
        r_AB(i,ii) = sqrt((refB(i,1,ii)-srcA(ii,1))^2 + (refB(i,2,ii)-srcA(ii,2))^2)←
        ;
        r_BA(i,ii) = sqrt((refA(i,1,ii)-srcB(ii,1))^2 + (refA(i,2,ii)-srcB(ii,2))^2)←
        ;
    end

    % Distance b/w sources and response MPs
    r_A(:,ii) = sqrt((resp(1)-srcA(ii,1))^2 + (resp(2)-srcA(ii,2))^2);
    r_B(:,ii) = sqrt((resp(1)-srcB(ii,1))^2 + (resp(2)-srcB(ii,2))^2);
end

end

```

A.1.3. Function to Define The Source Signal (*source.m*)

```

% Create source signals for otpa_vto.m analysis.
% Two sources: srcA & srcB.
% This version creates a single tone plus noise for each source.

function [prefa,prefb,preisp,pacont,pbcont] = source (FreqA,FreqB,Va,Vb,NAfact,NBfact←
, sizeA, sizeB)

global r_AtoB c a b f Nb N A B noiseA noiseB beta numMPs
% Compute distances between src's and MPs. See function "dist.m"

[r_A,r_B,r_AA,r_BB,r_AB,r_BA] = dist (FreqA, FreqB);

% Random additional amplitude and phase shift for source (srcA=srcB)

```

A.1. MATLAB Script - Two Airborne Noise Sources

```

rndamp = 10*rand(1);
rndphase = pi*rand(1);

rho = 1.2;
a = sizeA*(c/FreqA/2); % Radius of Source A spherical source
b = sizeB*(c/FreqA/2); % Radius of Source B spherical source

N = 200; % Number of frequency lines per sample
f=1:N; % Frequency vector (Hz)
Nb = 10; % number of measurement blocks

% Build Input Signals
omegaA = 2*pi*FreqA; % Frequency (rad/s)
kA = omegaA/c; % wavenumber (rad/m)

omegaB = 2*pi*FreqB; % Frequency (rad/s)
kB = omegaB/c; % wavenumber (rad/m)

A = (Va+rndamp)*(1j*omegaA*rho*a^2)/(1+1j*kA)*exp(1j*kA*a); % Amplitude for Source A↵
.
B = (Vb+rndamp)*(1j*omegaB*rho*b^2)/(1+1j*kB)*exp(1j*kB*b); % Amplitude for Source B↵
.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% "Measured" Sound Pressure at MPs

% Sound pressure at reference MPs (FREQUENCY DOMAIN)

for n = 1:Nb
    % Amplitude for noise
    noiseA = Nfact*Va*rand(size(f));
    noiseB = Nfact*Vb*rand(size(f));
    omega = 2*pi.*f;

    for ii=1:length(r_AtoB)
        for i = 1:numMPs

            % Define signal as noise (rnd amplitude, rnd phase assign)
            paan(:,i,ii) = noiseA./r_AA(i,ii).*exp(-1j*(omega*r_AA(i,ii)+rand*2*pi))↵
            ;
            pbbn(:,i,ii) = noiseB./r_BB(i,ii).*exp(-1j*(omega*r_BB(i,ii)+rand*2*pi))↵
            ;
            pabn(:,i,ii) = noiseA./r_AB(i,ii).*exp(-1j*(omega*r_AB(i,ii)+rand*2*pi))↵
            ;
            pban(:,i,ii) = noiseB./r_BA(i,ii).*exp(-1j*(omega*r_BA(i,ii)+rand*2*pi))↵
            ;

            paa(:,i,ii) = paan(:,i,ii);
            pbb(:,i,ii) = pbbn(:,i,ii);
            pab(:,i,ii) = pabn(:,i,ii);
            pba(:,i,ii) = pban(:,i,ii);

            % Add tone at specified frequencies (includes noise
            % contribution)
            paa(FreqA,i,ii) = paan(FreqA,i,ii) + A/r_AA(i,ii)*exp(-1j*(omegaA*r_AA(↵
            i,ii)+rndphase));

```

```

pbb (FreqB,i,ii) = pbbn(FreqB,i,ii) + B/r_BB(i,ii)*exp(-1j*(omegaB*r_BB(←
i,ii)-beta+rndphase));
pab (FreqA,i,ii) = pabn(FreqA,i,ii) + A/r_AB(i,ii)*exp(-1j*(omegaA*r_AB(←
i,ii)+rndphase));
pba (FreqB,i,ii) = pban(FreqB,i,ii) + B/r_BA(i,ii)*exp(-1j*(omegaB*r_BA(←
i,ii)-beta+rndphase));

prefa(n,i,:,ii) = paa(:,i,ii) + pba(:,i,ii);
prefb(n,i,:,ii) = pbb(:,i,ii) + pab(:,i,ii);
end

% Actual contribution from Source A and Source B at receiver

pacontn(n,:,ii) = noiseA./r_A(:,ii).*exp(-1j*omega*r_A(:,ii));
pbcontn(n,:,ii) = noiseB./r_B(:,ii).*exp(-1j*(omega*r_B(:,ii)-beta));

pacont(n,:,ii) = pacontn(n,:,ii);
pbcont(n,:,ii) = pbcontn(n,:,ii);

pacont(n,FreqA,ii) = pacont(n,FreqA,ii) + A/r_A(:,ii)*exp(-1j*(omegaA*r_A(:,←
ii)+rndphase));
pbcont(n,FreqB,ii) = pbcont(n,FreqB,ii) + B/r_B(:,ii)*exp(-1j*(omegaB*r_B(:,←
ii)-beta+rndphase));

% Sound pressure at response MP (receiver)
presp(n,:,ii) = pacont(n,:,ii) + pbcont(n,:,ii);

end
end
end

```

A.2. MATLAB Script - Airborne and Structure-Borne Sources

A.2.1. OTPA Script (*otpa_sbs.m*)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Operational Transfer Path Analysis – Virtual Test Setup           %
%                                                                    %
% Mihkel Toome                                                    %
%                                                                    %
% Description:                                                    %
% Two sources. Measurement positions at various distances from source.%
% Sources include near field                                     %
% Several reference MPs. One response MP.                        %
% Results are processed via the OTPA CTC Function.              %
%                                                                    %
% Functions associated with code (x3):                             %
% 1. MPs and Src/resp positions defined by function "dist_sbs.m". %
% 2. Sources defined in INPUT. Calcs done in function "source_sbs.m". %
% 3. Countour plot is setup by function "contourmesh.m".        %
%                                                                    %
% Version: 4.0                                                    %
% Date: 23 April 2012                                           %

```

A.2. MATLAB Script - Airborne and Structure-Borne Sources

```
%
%
%%
tic;
clear all; close all; clc;
global rA rB N Nb mps beta
%
% INPUT
%
% INPUT: SrcA and SrcB properties , distances between Src's.

FreqA = 25;
sizeA = 1/40; % Size (radius) of spherical SrcA rel. to lambda A.
Va = 60;
NAfact = 0*0.01; % Amplitude of noise (relative to Va)
rA = 1/16; % distance between SrcA and ref MPs (rel. to lambda A)

FreqB = 25;
beta = 0; % Assign phase shift for Source B [radians]
sizeB = sizeA; % Size of spherical SrcB rel. to lambda A.
Vb = 0;
NBfact = 0; % Amplitude of noise (relative to Va)

rB = [1/16 1/10 1/8 1/6 1/4 1/2]; % distance between SrcB and ref MPs (rel. to lambda A)

numA = 4; % Number of reference MPs for Source A (max. 4)
numB = 4; % Number of additional reference MPs for Source A (max. 4)

Asbs = 1; % Amplitude of structure-borne sound source (rel to SrcA ref MP SPL)

%%
% INPUT: PCA Contribution Threshold

% Select PCA method
PCA_METHOD = 'thres'; % 'thres' or 'cumcon' or 'PCs' or 'minPC'

% Input value according to selected PCA method
thres = 5; %Threshold: Minimum PCs % contribution towards PCs score (e.g. %5, thres=5)
cumcon = 85; % Cumulative Contribution of PCs contributing to % of signal
PCs = 2; % # of principal components to be use in analysis
minPC = 100; % Minimum magnitude of the PCs to be included

%%
% INPUT: Turn Figures on/off; print figures on/off.
plotbar = 'on'; % Contribution bar graph 'on/off'
plotcontour = 'on'; % Plot contour plot of SPL 'on/off'

printfigures = 'off'; % print EPS of figures 'on'/'off'

block = 2; % Which block of data is to be plotted [1<block<Nb]

%%
%
% CALCULATIONS
%
```

```

%
%
% Function calculates source strength at MPs
tic;
[prefa,prefb,preSp,pacont, sbs] = source_sbs(FreqA,FreqB,Va,Vb,NAfact,NBfact,sizeA,←
sizeB, Asbs);
toc;

% FREQUENCY DOMAIN
% Organize ref & resp measurements into matrices X and Y
% Dimensions of X: Rows=Meas.Blocks; Columns=MPs (for "N" Freq. lines and
% "length(rA)" distances between SrcA and MPs).
if numA == 0

    X = [prefb(:,1:numB, :, :) sbs];
elseif numB == 0
    X = [prefa(:,1:numA, :, :) sbs];
else
% X = (Meas.Blocks, MPs, Freq.Components, Src.Distances)
X = [prefa(:,1:numA, :, :) prefb(:,1:numB, :, :) sbs]; % freq domain
end
% Y = (Meas.Blocks, Freq.Components, Src.Distances)
Y = preSp; % freq domain

%% Cross Talk Cancellation & Transfer Function Calculation (Frequency Domain)

for ii=1:length(rB) % Loop for various SrcA-MP distances
    for kk=1:N % Loop for N FFT frequency components

        % Singular Value Decomposition (SVD). N.B. "S" is the matrix of PCs.
        [U,S,V] = svd(X(:, :, kk, ii), 'econ'); % use "econ" to ensure S is a square ←
        Matrix.

        % Principal Component Analysis (PCA)
        % COEFF – The coefficients corresponding to the PCs
        % PC – The Principal Components (i.e. the Singular Values)
        % CONT – The contribution to the overall signal (%)

        % [COEFF, PC, CONT] = pcacov(S); % Matlab function for PCA
        % Equivalent calc. for the PCA matlab function above
        PC = diag(S);
        COEFF = U*S;
        CONT = diag(S)/sum(diag(S))*100;
        Ssave(:, :, kk)=S;
        % Find PCs that corresponds to PCA method input
        if strcmp(PCA_METHOD, 'thres') == 1
            numPC = find(CONT>thres,1,'last');
        elseif strcmp(PCA_METHOD, 'cumcon')==1
            numPC = find(sum(CONT)>cumcon,1,'last');
        elseif strcmp(PCA_METHOD, 'PCs')==1
            numPC = PCs;
        elseif strcmp(PCA_METHOD, 'minPC')==1
            numPC = find(PC>minPC,1,'last');
        end
    end
end

```

```

% Calculate "invSr" – The noise reduced inverse of S.
invSr = zeros(size(S)); % Allocate zeros for size of the invSr matrix
S1 = diag(diag(inv(S(1:numPC,1:numPC))));
invSr(1:numPC, 1:numPC) = S1(1:numPC, 1:numPC);

% Calculate noise reduced inverse of X

X_inv(:, :, kk, ii) = V*invSr*U'; % Noise reduced inverse of X

% Transfer function estimate
H(:, kk, ii) = X_inv(:, :, kk, ii)*Y(:, kk, ii);

end
end
toc; tic;
%% Calculate Synthesized Response at Receiver (Freq. Domain)
%% Re-run source: Random noise is redefined, simulate separate measurements
% for transfer function calculation and synthesis calculation.
clear X Y prefa prefb presp pacont pbcont
[prefa, prefb, presp, pacont, sbs] = source_sbs(FreqA, FreqB, Va, Vb, NAfact, NBfact, sizeA, ←
    sizeB, Asbs);
if numA == 0
    X = [prefb(:, 1:numB, :, :) sbs];
elseif numB == 0
    X = [prefa(:, 1:numA, :, :) sbs];
else
% X = (Meas.Blocks, MPs, Freq.Components, Src.Distances)
X = [prefa(:, 1:numA, :, :) prefb(:, 1:numB, :, :) sbs]; % freq domain
end
% Y = (Meas.Blocks, Freq.Components, Src.Distances)
Y = presp; % freq domain

for ii=1:length(rB)
    for jj=1:Nb
        for kk=1:N

            % Take transpose of X to rearrange variables for Ys
            Ys(:, jj, kk, ii) = X(jj, :, kk, ii).*H(:, kk, ii)';

            % Sum for total synthesis and source contributions
            % Sum is as if for continuation of matrix multiplication
            % (Ys=XH)

            if numA == 0
                Ystot(kk, jj, ii) = sum(Ys(:, jj, kk, ii));
                Ys_srcA(kk, jj, ii) = sum(Ys(1:numB, jj, kk, ii));
                Ys_abs(kk, jj, ii) = Ys_srcA(kk, jj, ii);
                Ys_sbs(kk, jj, ii) = Ys((1+numB), jj, kk, ii);

            elseif numB == 0
                Ystot(kk, jj, ii) = sum(Ys(:, jj, kk, ii));
                Ys_srcA(kk, jj, ii) = sum(Ys(1:numA, jj, kk, ii));
                Ys_abs(kk, jj, ii) = Ys_srcA(kk, jj, ii);
                Ys_sbs(kk, jj, ii) = Ys((1+numA), jj, kk, ii);

            else

```

```

Ystot(kk,jj,ii) = sum((Ys(:,jj,kk,ii)));
Ys_srcA(kk,jj,ii) = sum((Ys(1:numA+numB,jj,kk,ii)));
Ys_abs(kk,jj,ii) = Ys_srcA(kk,jj,ii);
Ys_sbs(kk,jj,ii) = Ys((numA+numB+1),jj,kk,ii);

    end
end
end
[prepmeshA,prepmeshB,x,y] = contourmesh (FreqA, FreqB);
toc;

```

A.2.2. Function to Define The Virtual Test Setup Layout (*dist_sbs.m*)

```

% Define source & receiver (Reference and Response) locations as
% coordinates (x,y). Distances are multiples of wavelength (lambda=d).
% $$$$ SrcB MPs distance is varied $$$$
% r_AtoB – Distance between SrcA and SrcB
% r_AA – Distance between SrcA and SrcA ref MPs
% r_BB – Distance between SrcB and SrcB ref MPs
% r_AB – Distance between SrcA and SrcB ref MPs
% r_BA – Distance between SrcB and SrcA ref MPs
% mps – Matrix of all Src's, resp, and MPs (i.e. for a plot)
% a – Radius of SrcA (1/40*lamda_SrcA)
% b – Radius of SrcB (1/40*lamda_SrcA)
% c – Speed of sound in air (340m/s)

function [r_A,r_B,r_AA,r_BB,r_AB,r_BA] = dist_sbs (FreqA,FreqB)
global rA rB c mps d srcA srcB

c = 340;
lambda = c/FreqA;
d=lambda;

% refB_dist = rB*c/FreqA;           % Define distance for Source B reference ↔
% positions
coords = [d, 0; 0, d; -d, 0; 0, -d];

% srcA = [0,0];
% refA = rA*coords; % Ref MPs for SrcA

resp = [0, d/sqrt(pi)];           % Response measurement position (receiver)

% r_AtoB = sqrt((srcA(1)-srcB(1))^2+(srcA(2)-srcB(2))^2); % SrcA-SrcB distance is d↔
% /16

for ii=1:length(rB) % Iteration for each SrcA to SrcB distance

    % define SrcB position relative to SrcA (centre to centre distance)
    srcB(ii,:) = [0,0];
    srcA(ii,:) = [0,0];

    % Define reference MP's for SrcA and SrcB (centre to centre distance)

```

```

for i = 1:4
    refA(i,:,ii) = rA*coords(i,:);
    refB(i,:,ii) = rB(ii)*coords(i,:);
end

% Assemble src 's and response and reference positions into a matrix
mps(:, :, ii) = [srcA(ii, :); srcB(ii, :); resp; refA(:, :, ii); refB(:, :, ii)];

% Distance b/w sources and reference MPs
for i=1:4
    r_AA(i,ii) = sqrt((refA(i,1,ii)-srcA(ii,1))^2 + (refA(i,2,ii)-srcA(ii,2))^2)↵
    ;
    r_BB(i,ii) = sqrt((refB(i,1,ii)-srcB(ii,1))^2 + (refB(i,2,ii)-srcB(ii,2))^2)↵
    ;
    r_AB(i,ii) = sqrt((refB(i,1,ii)-srcA(ii,1))^2 + (refB(i,2,ii)-srcA(ii,2))^2)↵
    ;
    r_BA(i,ii) = sqrt((refA(i,1,ii)-srcB(ii,1))^2 + (refA(i,2,ii)-srcB(ii,2))^2)↵
    ;
end

% Distance b/w sources and response MPs
r_A(:,ii) = sqrt((resp(1)-srcA(ii,1))^2 + (resp(2)-srcA(ii,2))^2);
r_B(:,ii) = sqrt((resp(1)-srcB(ii,1))^2 + (resp(2)-srcB(ii,2))^2);
end

end

```

A.2.3. Function to Define The Source Signal (*source_sbs.m*)

```

% Create source signals for otpa_vto.m analysis.
% Two sources: srcA (airborne) and sbs (structure-borne).
% sbs is equal to one ref MP of airborne (multiplied by a factor)

function [prefa,prefb,presep,pacont, sbs] = source_sbs (FreqA,FreqB,Va,Vb,NAfact,↵
    NBfact,sizeA,sizeB, Asbs)

global rB c a b f Nb N A B noiseA noiseB beta
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Compute distances between src 's and MPs. See function "dist.m"

[r_A,r_B,r_AA,r_BB,r_AB,r_BA] = dist_sbs (FreqA, FreqB);

% Random additional amplitude and phase shift for source (srcA=srcB)

rndamp = 10*rand(1);
rndphase = pi*rand(1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
rho = 1.2;
a = sizeA*(c/FreqA/2); % Radius of Source A spherical source
b = sizeB*(c/FreqA/2); % Radius of Source B spherical source

N = 200; % Number of frequency lines per sample

```

```

f=1:N; % Frequency vector (Hz)
Nb = 10; % number of measurement blocks

% Build Input Signals
omegaA = 2*pi*FreqA; % Frequency (rad/s)
kA = omegaA/c; % wavenumber (rad/m)

omegaB = 2*pi*FreqB; % Frequency (rad/s)
kB = omegaB/c; % wavenumber (rad/m)

A = (Va+randamp)*(1j*omegaA*rho*a^2)/(1+1j*kA)*exp(1j*kA*a); % Amplitude for Source A↔
.
B = (Vb+randamp)*(1j*omegaB*rho*b^2)/(1+1j*kB)*exp(1j*kB*b); % Amplitude for Source B↔
.

%% "Measured" Sound Pressure at MPs

% Sound pressure at reference MPs (FREQUENCY DOMAIN)
for n = 1:Nb
    % Amplitude for noise
    noiseA = MAfact*Va*rand(size(f));
    noiseB = NBfact*Vb*rand(size(f));
    omega = 2*pi.*f;

    for ii=1:length(rB)
        for i = 1:4

            % Define signal as noise
            paan(:,i,ii) = noiseA./r_AA(i,ii).*exp(-1j*omega*r_AA(i,ii)+rand*2*pi);
            pabn(:,i,ii) = noiseB./r_AB(i,ii).*exp(-1j*omega*r_AB(i,ii)+rand*2*pi);

            paa(:,i,ii) = paan(:,i,ii);
            pab(:,i,ii) = pabn(:,i,ii);

            % Add tone at specified frequencies (includes noise
            % contribution)
            paa(FreqA,i,ii) = paan(FreqA,i,ii) + A/r_AA(i,ii)*exp(-1j*(omegaA*r_AA(↔
            i,ii)+randphase));
            pab(FreqA,i,ii) = pabn(FreqA,i,ii) + A/r_AB(i,ii)*exp(-1j*(omegaA*r_AB(↔
            i,ii)+randphase));

            prefa(n,i,:,ii) = paa(:,i,ii);
            prefb(n,i,:,ii) = pab(:,i,ii);
        end

        % Actual contribution from Source A and Source B at receiver
        pacontn(n,:,ii) = noiseA./r_A(:,ii).*exp(-1j*omega*r_A(:,ii));
        pacont(n,:,ii) = pacontn(n,:,ii);
        pacont(n,FreqA,ii) = pacont(n,FreqA,ii) + A/r_A(:,ii)*exp(-1j*(omegaA*r_A(↔
        ii)+randphase));
    end
end

```

A.3. Function to Plot Sound Field (*contourmesh.m*)

```
    % Simulate Structure-Borne sound contribution

    sbs (n,1,:,ii) = Asbs*prefa(n,1,:,ii);

    % Sound pressure at response MP (receiver)
    presp(n,:,ii) = pacont(n,:,ii) + Asbs*pacont(n,:,ii);

end
end
end
```

A.3. Function to Plot Sound Field (*contourmesh.m*)

```
% Function to create data for contour plots of the SPL over
% the Virtual Test Setup area. Two plots at FreqA and FreqB.

function [prespmeshA,prespmeshB,x,y] = contourmesh (FreqA, FreqB)
global A B noiseA noiseB Nb d srcA srcB beta

omegaA = 2*pi*FreqA;
omegaB = 2*pi*FreqB;

[x,y] = meshgrid(-0.7*d:0.1:0.7*d,-0.7*d:0.1:0.7*d); % Dims chosen to make plot look ←
    nice

for n = 1:Nb
    for ii=1:6

        rA = sqrt((x-srcA(ii,1)).^2+(y-srcA(ii,2)).^2);
        rB = sqrt((x-srcB(ii,1)).^2+(y-srcB(ii,2)).^2);

        pan (:,:,ii,n) = noiseA(FreqA)./rA.*exp(-1j*omegaA.*rA);
        pbn (:,:,ii,n) = noiseB(FreqB)./rB.*exp(-1j*(omegaB.*rB-beta));

        % Redefine tone at specified frequencies
        pa (:,:,ii,n) = A./rA.*exp(-1j*omegaA.*rA);
        pb (:,:,ii,n) = B./rB.*exp(-1j*(omegaB.*rB-beta));

        % Add total SPL

        if FreqA==FreqB
            prespmeshA (:,:,ii,n)= pa(:,:,ii,n) + pb(:,:,ii,n);
            prespmeshB (:,:,ii,n) = pa(:,:,ii,n) + pb(:,:,ii,n);
        else
            prespmeshA (:,:,ii,n)= pa(:,:,ii,n) + pbn(:,:,ii,n);
            prespmeshB (:,:,ii,n)= pb(:,:,ii,n) + pan(:,:,ii,n);

        end
    end
end
end
end
```

A.4. MATLAB Script - Airborne and Structure-Borne Sources, Varied Number of Reference MPs

A.4.1. OTPA Script (*otpa_sbs_num.m*)

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   Operational Transfer Path Analysis – Virtual Test Setup           %
%   %                                                                 %
%   Mihkel Toome                                                    %
%   %                                                                 %
%   Description:                                                    %
%   Two sources: ABS and SBS.                                       %
%   Measurement positions at various distances from source.        %
%   Sources include near field                                     %
%   Several reference MPs. One response MP.                         %
%   Results are processed via the OTPA CTC Function.                %
%   %                                                                 %
%   Functions associated with code (x3):                             %
%   1. MPs and Src/resp positions defined by function "dist_sbs.m". %
%   2. Sources defined in INPUT. Calcs done in function "source_sbs.m". %
%   3. Countour plot is setup by function "contourmesh.m".         %
%   %                                                                 %
%   %                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tic;
clear all; close all; clc;
global rA N Nb mps numA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                 %
%                               INPUT                               %
%                                                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% INPUT: SrcA and SrcB properties , distances between Src's.

FreqA = 25;
sizeA = 1/40;    % Size (radius) of spherical SrcA rel. to lambda A.
Va = 60;
NAfact = 0*0.01; % Amplitude of noise (relative to Va)
rA = 1/16;      % distance between SrcA and ref MPs (rel. to lambda A)

numA = [1 2 3 4 5 6]; % Number of reference MPs for Source A

Asbs = 2; % Amplitude of structure-borne sound source (rel to SrcA ref MP SPL)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% INPUT: PCA Contribution Threshold

% Select PCA method
PCA_METHOD = 'thres'; % 'thres' or 'cumcon' or 'PCs' or 'minPC'

% Input value according to selected PCA method

```

A.4. MATLAB Script - Airborne and Structure-Borne Sources, Varied Number of Reference MPs

```

thres = 5; %Threshold: Minimum PCs % contribution towards PCs score (e.g. %5, thres←
    =5)
cumcon = 85; % Cumulative Contribution of PCs contributing to % of signal
PCs = 2; % # of principal components to be use in analysis
minPC = 100; % Minimum magnitude of the PCs to be included

%% INPUT: Turn Figures on/off; print figures on/off.
plotbar = 'on'; % Contribution bar graph 'on/off'
plotcontour = 'on'; % Plot contour plot of SPL 'on/off'

printfigures = 'off'; % print EPS of figures 'on'/'off'

block = 2; % Which block of data is to be plotted [1<block<Nb]

%% CALCULATIONS

%% Cross Talk Cancellation & Transfer Function Calculation (Frequeny Domain)

for ii=1:length(numA) % Loop for various SrcA-MP distances

    % Function calculates source strength at MPs
    tic;
    [prefa,presp,pacont, sbs] = source_sbs_num(FreqA,Va,Nafact,sizeA, Asbs);
    toc;
    % FREQUENCY DOMAIN
    % Organize ref & resp measurements into matrices X and Y
    % Dimensions of X: Rows=Meas.Blocks; Columns=MPs (for "N" Freq. lines and
    % "length(rA)" distances between SrcA and MPs).

    % X = (Meas.Blocks, MPs, Freq.Components)
    X = [prefa(:,1:numA(ii),:) sbs]; % freq domain
    % Y = (Meas.Blocks, Freq.Components, Src.Distances)
    Y = presp; % freq domain

    for kk=1:N % Loop for N FFT frequency components

        % Singular Value Decomposition (SVD). N.B. "S" is the matrix of PCs.
        [U,S,V] = svd(X(:,:,kk), 'econ'); % use "econ" to ensure S is a square ←
            Matrix.

        % Principal Component Analysis (PCA)
        % COEFF – The coeffecients corresponding to the PCs
        % PC – The Principal Components (i.e. the Singular Values)
        % CONT – The contribution to the overall signal (%)

        % [COEFF, PC, CONT] = pcacov(S); % Matlab function for PCA
        % Equivalent calc. for the PCA matlab function above
        PC = diag(S);
        COEFF = U*S;

```

```

CONT = diag(S)/sum(diag(S))*100;
% Ssave(:, :, kk)=S;
% Find PCs that correspond to PCA method input
if strcmp(PCA_METHOD, 'thres') == 1
    numPC = find(CONT>thres,1, 'last');
elseif strcmp(PCA_METHOD, 'cumcon')==1
    numPC = find(sum(CONT)>cumcon,1, 'last');
elseif strcmp(PCA_METHOD, 'PCs')==1
    numPC = PCs;
elseif strcmp(PCA_METHOD, 'minPC')==1
    numPC = find(PC>minPC,1, 'last');
end

% Calculate "invSr" – The noise reduced inverse of S.
invSr = zeros(size(S)); % Allocate zeros for size of the invSr matrix
S1 = diag(diag(inv(S(1:numPC,1:numPC))));
invSr(1:numPC, 1:numPC) = S1(1:numPC, 1:numPC);

% Calculate noise reduced inverse of X

X_inv = V*invSr*U'; % Noise reduced inverse of X

% Transfer function estimate
H(:,kk) = X_inv*Y(:,kk);

end

toc; tic;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Calculate Synthesized Response at Receiver (Freq. Domain)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Re-run source: Random noise is redefined, simulate separate measurements
% for transfer function calculation and synthesis calculation.
clear X Y prefa presp pacont sbs
[prefa, presp, pacont, sbs] = source_sbs_num(FreqA, Va, Nafact, sizeA, Asbs);
%% X = (Meas.Blocks, MPs, Freq.Components, Src.Distances)
X = [prefa(:,1:numA(ii),:) sbs]; % freq domain
% Y = (Meas.Blocks, Freq.Components, Src.Distances)
Y = presp; % freq domain

for jj=1:Nb
    for kk=1:N

        % Take transpose of X to rearrange variables for Ys
        Ys(:,jj,kk) = X(jj,:,kk)'.*H(:,kk);

        % Sum for total synthesis and source contributions
        % Sum is as if for continuation of matrix multiplication
        % (Ys=XH)

        Ystot(kk,jj,ii) = sum((Ys(:,jj,kk)));
        Ys_srcA(kk,jj,ii) = sum(Ys(1:numA(ii),jj,kk));
        Ys_abs(kk,jj,ii) = Ys_srcA(kk,jj,ii);
        Ys_sbs(kk,jj,ii) = Ys(end,jj,kk);

    end
end

```

```

end

clear H X Ys

end
[premeshA,x,y] = contourmesh_num (FreqA);
toc;

```

A.4.2. Function to Define The Virtual Test Setup Layout (*dist_sbs_num.m*)

```

% Define source & receiver (Reference and Response) locations as
% coordinates (x,y). Distances are multiples of wavelength (lambda=d).
% $$$$ SrcB MPs distance is varied $$$$
% r_AtoB – Distance between SrcA and SrcB
% r_AA – Distance between SrcA and SrcA ref MPs
% r_BB – Distance between SrcB and SrcB ref MPs
% r_AB – Distance between SrcA and SrcB ref MPs
% r_BA – Distance between SrcB and SrcA ref MPs
% mps – Matrix of all Src's, resp, and MPs (i.e. for a plot)
% a – Radius of SrcA (1/40*lamda_SrcA)
% b – Radius of SrcB (1/40*lamda_SrcA)
% c – Speed of sound in air (340m/s)

function [r_A,r_AA] = dist_sbs_num(FreqA)
global rA c mps d srcA numA

c = 340;
lambda = c/FreqA;
d=lambda;

% refB_dist = rB*c/FreqA;           % Define distance for Source B reference ↔
% positions
coords = [d, 0; 0, d; -d, 0; 0, -d; d/sqrt(2), d/sqrt(2); -d/sqrt(2), -d/sqrt(2)];

srcA = [0,0];
% refA = rA*coords; % Ref MPs for SrcA

resp = [0, d/sqrt(pi)];           % Response measurement position (receiver)

% Define reference MP's for SrcA and SrcB (centre to centre distance)
for i = 1:length(numA)
    refA(i,:) = rA*coords(i,:);
end

% Assemble src's and response and reference positions into a matrix
mps = [srcA; resp; refA];

% Distance b/w sources and reference MPs
for i=1:length(numA)
    r_AA(i) = sqrt((refA(i,1)-srcA(1))^2 + (refA(i,2)-srcA(2))^2);
end

% Distance b/w sources and response MPs

```

```
r_A = sqrt((resp(1)-srcA(1))^2 + (resp(2)-srcA(2))^2);
```

```
end
```

A.4.3. Function to Define The Source Signal (*source_sbs_num.m*)

```
% Create source signals for otpa_vto.m analysis.
% Two sources: srcA (airborne) and sbs (structure-borne).
% sbs is equal to one ref MP of airborne (multiplied by a factor)

function [prefa,presp,pacont, sbs] = source_sbs_num (FreqA,Va,NAfact,sizeA, Asbs)

global c a f Nb N A noiseA numA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Compute distances between src's and MPs. See function "dist.m"

[r_A,r_AA] = dist_sbs_num (FreqA);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Random additional amplitude and phase shift for source (srcA=srcB)
rndamp = 10*rand(1);
rndphase = pi*rand(1);

rho = 1.2;
a = sizeA*(c/FreqA/2); % Radius of Source A spherical source

N = 200; % Number of frequency lines per sample
f=1:N; % Frequency vector (Hz)
Nb = 10; % number of measurement blocks

% Build Input Signals
omegaA = 2*pi*FreqA; % Frequency (rad/s)
kA = omegaA/c; % wavenumber (rad/m)

A = (Va+rndamp)*(1j*omegaA*rho*a^2)/(1+1j*kA)*exp(1j*kA*a); % Amplitude for Source A←
.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% "Measured" Sound Pressure at MPs

% Sound pressure at reference MPs (FREQUENCY DOMAIN)

for n = 1:Nb
% Amplitude for noise
noiseA = NAfact*Va*rand(size(f));
omega = 2*pi.*f;

for i = 1:length(numA)

% Define signal as noise
paan(:,i) = noiseA./r_AA(i).*exp(-1j*omega*r_AA(i)+rand*2*pi);
```

```

    paa (:,i) = paan(:,i);

    % Add tone at specified frequencies (includes noise
    % contribution)
    paa (FreqA,i) = paan(FreqA,i) + A/r_AA(i)*exp(-1j*(omegaA*r_AA(i)+
    rndphase));
    prefa(n,i,:) = paa(:,i);
end

% Actual contribution from Source A and Source B at receiver

pacontn(n,:) = noiseA./r_A(:).*exp(-1j*omega*r_A);
pacont(n,:) = pacontn(n,:);
pacont(n,FreqA) = pacont(n,FreqA) + A/r_A*exp(-1j*(omegaA*r_A+rndphase));

% Simulate Structure-Borne sound contribution

sbs (n,1,:) = Asbs*prefa(n,1,:);

% Sound pressure at response MP (receiver)
presp(n,:) = pacont(n,:) + Asbs*pacont(n,:);

end
end

```

A.4.4. Function to Plot Sound Field (*contourmesh_num.m*)

```

% Function to create data for contour plots of the SPL over
% the Virtual Test Setup area. Two plots at FreqA and FreqB.

function [prespmeshA,x,y] = contourmesh_num (FreqA)
global A noiseA Nb d srcA

omegaA = 2*pi*FreqA;

[x,y] = meshgrid(-0.7*d:0.05:0.7*d,-0.7*d:0.05:0.7*d); % Dims chosen to make plot ←
look nice

for n = 1:Nb

    rA = sqrt((x-srcA(1)).^2+(y-srcA(2)).^2);
    pan (:,:,n) = noiseA(FreqA)./rA.*exp(-1j*omegaA.*rA);

    % Redefine tone at specified frequencies
    pa (:,:,n) = A./rA.*exp(-1j*omegaA.*rA);

    % Add total SPL

    prespmeshA (:,:,n) = pa(:,:,n) + pan(:,:,n);

end
end

```