Panel Damping Loss Factor Estimation

Using The

Random Decrement Technique

By

Himanshu Amol Dande

Submitted to the graduate degree program in Aerospace Engineering and the

Graduate faculty of the University of Kansas in partial fulfillment of the requirements

for the degree of Master of Science

_____

Chairperson: Dr. Mark Ewing

_____

Dr. Richard Hale

_____

Dr. Saeed Farokhi

_____

Dr. Adolfo Matamoros

The Thesis Committee for Himanshu Amol Dande

certifies this is the approved version of the following thesis:

Panel Damping Loss Factor Estimation

Using The

Random Decrement Technique

_____

Chairperson: Dr. Mark Ewing

_____

Dr. Richard Hale

_____

Dr. Saeed Farokhi

_____

Dr. Adolfo Matamoros

Date Approved: 10th December 2010

Abstract

The use of the Random Decrement Technique (RDT) for estimating panel damping loss factors ranging from 1% to 10% is examined in a systematic way, with a focus on establishing the various parameters one must specify to use the technique to the best advantage.

Throughout, loss factors are estimated in full or 1/3rd octave frequency bands with standard 1/3rd octave center frequencies. The full octave filters, which are more computationally efficient than the 1/3rd octave filters, are chosen in the experimental analysis of the damped plates with varied loss factor levels.

Two computational models are examined: a single degree of freedom oscillator and a computational model of a uniform rectangular panel. The panel computational model is a finite element model of a rectangular plate mechanically exited at a single point. These models are used to establish a systematic process for evaluating: the appropriate narrow band filter selection; trigger conditions; record length required as a function of frequency and damping level; the averaging scheme; and, the curve–fitting scheme for assigning loss factors in narrow frequency bands.

Loss factor estimates for three damped plates are computed using the "optimized" Random Decrement estimation algorithm and compared with estimates from the Impulse Response Decay Method. For the highly damped plates, RDT out–performed IRDM because the loss factors from IRDM are underestimated. For lightly damped plates, RDT and IRDM are consistent in most frequency bands.

Acknowledgements

Table of Contents

List of Figures

List of Tables

List of Symbols

$1DOF$	Single or One Degree Of Freedom
$Apass$	Passband attenuation (dB)
$Astop1$	First stopband attenuation (dB)
$Astop2$	Second stopband attenuation (dB)
$B_\%$	Surplus width of passband
C	Vector of damping coefficients
CLD	Constrained Layer Damping
c	Damping coefficient (kg/sec)
DAQ	Data AcQuisition
dB	Decibels (dB)
DR	Decay Rate (dB/sec)
$D_{stinger}$	Diameter of stinger (m)
D_X	Randomdec signature
E	Young's modulus of elasticity (Pascal)
E_D	Dissipated energy (Joule)
E_{Tot}	Total mechanical energy (Joule)
F	Vector of forcing functions
$FDATool$	Filter Design and Analysis Tool
FDF	Frequency Domain Filtering
FFT	Fast Fourier Transformation

FIR	Finite–duration Impulse Response
F_{Octave}	Octave Factor, $2^{1/2}$ for full octave and $2^{1/6}$ for 1/3rd octave bands
*Fpass*1	Beginning of the passband (Hz)
*Fpass*2	End of the passband (Hz)
FRF	Frequency Response Function
Fs	Sampling frequency (Hz)
*Fstop*1	Band of the first stopband (Hz)
*Fstop*2	Beginning of the second stopband (Hz)
$F(\omega)$	Force in frequency domain
f_c	Band's central frequency (Hz)
f_n	Natural frequency (Hz)
Δf	Frequency resolution (Hz)
$f(t)$	Forcing function
H	Hilbert transformation
HPBWM	Half Power BandWidth Method
$h(t)$	Impulse response function, time domain
$H(\omega)$	Frequency response function, frequency domain
IFFT	Inverse Fast Fourier Transformation
IIR	Infinite–duration Impulse Response
IR	Impulse Response
IRDM	Impulse Response Decay Method
K	Stiffness matrix

$KUAE$	The University of Kansas, Department of Aerospace Engineering
k	Stiffness (N/m)
LF	Loss Factor
$LF1$	Loss Factor from dB decay estimation scheme#1
$LF2$	Loss Factor from dB decay estimation scheme#2
$LF3$	Loss Factor from dB decay estimation scheme#3
M	Mass matrix
$MDOF$	Multiple Degrees Of Freedom
m	Mass (kg)
N	Length of sample, in terms of cycles
N_{lines}	Lines in the Frequency Response Function
Order	Filter order
P_D	Dissipated power (Watt)
PIM	Power Input Method
P_{IN}	Input mechanical power (Watt)
TLF	Trigger Level Factor
$T_{\%}$	Width of trigger band as percentage of x_0
t	Time (seconds)
t_{plate}	Thickness of plate (m)
Q	Quality factor
RDM	Reverberation Decay Method
RDT	Random Decrement Technique

RDS	RandomDec Signature
R_X	Autocorrelation function
TDF	Time Domain Filtering
VRD	Vector Random Decrement
$X(\omega)$	Response in frequency domain
x	Displacement of a system
$\dot{x}$	Velocity of a system
$\ddot{x}$	Acceleration of a system
x_0	Amplitude of trigger condition or initial displacement
$\dot{x}_0$	Slope of triggering condition or initial velocity

Greek

η_{est}	Initial guess of loss factor
τ	Dummy time variable (seconds)
ζ	Damping Ratio
η	Loss Factor
ω	Radial frequency (radians/second)
ω_n	Radial natural frequency (radians/second)
ω_d	Radial damped natural frequency (radians/second)
ρ	Density (kg/m^3)
σ	Standard deviation of the measured response
ϕ	Phase angle (radians)

1   Introduction

The purpose of this chapter is to introduce some of the basic concepts and terms associated with the identification of vibration characteristics of a system, in particular the "damping loss factor". This chapter also includes discussion of the contemporary experimental loss factor estimation techniques, their application and limitations. Finally, the chapter is concluded with an outline of the thesis.

A vibration, in its general sense, is a periodic motion about an equilibrium position due to a disturbing force. Undesired vibrations can cause premature fatigue and unpleasant noise. The vibratory response of a system is usually a function of three variables: mass, stiffness and damping. Most of the contemporary damping estimation techniques are restricted to lower damping levels; hence, the challenge here lies in being able to accurately estimate the higher damping.

1.1   Damping

The vibrations induced in a structure due to a finite–duration excitation do not last forever and eventually "damp–out". This ability of a structure to dissipate vibrational energy is associated with an important characteristic of the system, known as damping which can be quantified as damping ratio(ζ) or loss factor(η).When subjected to persistent excitation, a structure has the highest amplitude of vibratory response at its natural frequency, which depends on the structure's mass and stiffness. The overall response level of a structure is inversely proportional to its damping.[1] Thus, the level of damping is an important design variable in noise reduction, earthquake resistant structures, instrument safety, and definition of flutter boundaries for airfoils. Therefore, accurate estimation of damping, usually experimentally, is an integral and critical part of all types of dynamic analyses.

For an inadequately damped structure, damping can be added explicitly. Additionally damping can be broadly classified as either active or passive. Active damping, generally in a narrow frequency band, is enforced by use of external devices such as actuators. On the contrary, passive damping is a method used to increase damping by application of layer(s) of energy dissipating materials. Passive damping treatments are simple to apply, cost effective and rather well established.

Constrained Layer Damping (CLD) is one of the most commonly used passive damping treatments. In CLD, a base structure is coated with a thin layer of damping material (usually viscoelastic material) which is then covered by a constraining layer (cover sheet). Under dynamic loading, this layer of damping material undergoes shear and dissipates energy, as represented in Figure 1.[2] This process, for addition of damping, is simple and cost effective as compared to other methods.
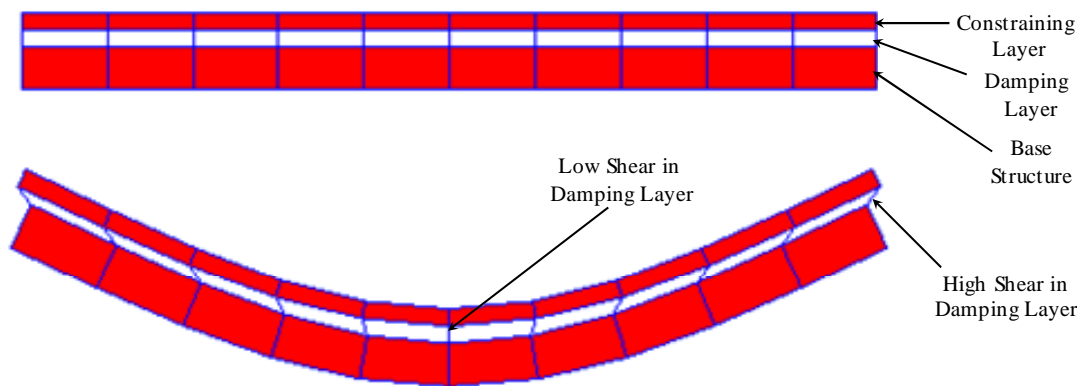
Figure 1: Schematic of constrained layer damping treatment. Reprinted with permission from Reference 2.

1.2   Damping Loss Factor

The equation of motion for a single Degree Of Freedom (1DOF) system, acted upon by force, is given by

$$m\ddot{x} + c\dot{x} + k x = f(t) \qquad \text{(eqn. 1.1)}$$

where,

$f(t)$ is the externally applied force as a function of time

m is the mass of the system

c is the damping coefficient of the system

k is the stiffness of the system

x is the displacement of the system

$\dot{x}$ is the velocity of the system

$\ddot{x}$ is the acceleration of the system.

It is not uncommon to express damping as "damping loss factor" (or simply loss factor), the latter being the ratio of dissipated and total energy. The relation between the damping ratio and loss factor is given by following equation

$$\eta = 2 \cdot \zeta = \frac{1}{Q} \qquad \text{(eqn. 1.2)}$$

where,

$\zeta = \dfrac{c}{2\sqrt{k \cdot m}}$ is the damping ratio

$2\sqrt{k \cdot m}$ is the critical damping

η is the loss factor

Q is the quality factor.

The above relation is accurate, within 5%, for the values of loss factor that are less than 0.3.[1,2] However, the loss factor can also be represented in terms of: ratio of energies, decay rate and half power bandwidth frequencies. A few of these relations are discussed in the following section.

1.3   Experimental Loss Factor Estimation

A real structure can have multiple natural frequencies in a frequency band of interest. For each mode of vibration a unique loss factor exists, i.e. damping in real structures is frequency dependent. For a system's reliable design, accurate damping estimation is very important. Unlike other system properties, such as mass and stiffness, estimation of loss factor is rarely straightforward and it is most often evaluated experimentally. Most of the experimental techniques for loss factor estimation rely on the assumption of a linear system with viscous damping.[3] On one hand, underestimation of loss factor may result in an unacceptable design, requiring damping to added, thereby increasing weight; on the other, overestimation of loss factor may lead to failure before the predicted life and/or discomfort because of unwanted and/or under–suppressed vibrations.[4]

The equation of motion for a Multiple Degree Of Freedom (MDOF) system is:

$$[M]\{\ddot{X}\}+[C]\{\dot{X}\}+[K]\{X\}=\{F\} \qquad \text{(eqn. 1.3)}$$

where,

$[\ ]$ represents a square matrix

$\{\ \}$ represents a column vector

M is the mass matrix, for a linear system

C is the damping matrix

K is the stiffness matrix

F is the vector of externally applied force

X is the vector of displacement

$\dot{X}$ is the vector of velocity

$\overset{\bullet\bullet}{X}$ is the vector of acceleration.

A linear MDOF system can be decoupled to identify system characteristics for each mode of vibration. With such an approach, the frequency dependent loss factor can be estimated at each of the system's natural frequencies.

The techniques discussed in the following section can be used to compute the loss factor of structures excited by persistent excitation and these techniques are based on the measure of total energy loss from the structure. In general, these loss factor estimation techniques can be classified as time domain or frequency domain. For further simplification, in this report, the techniques are segregated as decay–rate–based techniques and Frequency Response Function (FRF)–based (or alternatively from relations of Fast Fourier Transformation (FFT) of input and output) techniques.

1.3.1   Loss Factor Estimation Techniques Based on Transient Decay

This section describes some of the methodologies involved with computing loss factor from the Decay Rate (DR) of free decay. The decaying (i.e. transient) response of the structure can be generated by either impulse or interrupted steady–state excitation. The following three loss factor estimation techniques use different approaches to calculate the time domain Impulse Response (IR) or free decay:

1.   Random Decrement Technique (RDT)

2.   Impulse Response Decay Method (IRDM)

3.   Reverberation Decay Method (RDM).

For the above three techniques the procedure to evaluate loss factor – from the decay rate of an impulse response or free decay – is the same. To evaluate the decay rate, of exponentially decaying signal, Hilbert transform is often employed. This transform

essentially yields the envelope of free decay. In fact, the magnitude of Hilbert transformation of the impulse response is nearly equal to the exponential decay, as shown in Figure 2, that is:

$$e^{-\omega_n \cdot \zeta \cdot t} \approx \left| H(h(t)) \right| \qquad \text{(eqn. 1.4)}$$

where,

$H()$ is the Hilbert transformation of a function

$h(t)$ is the impulse response function.
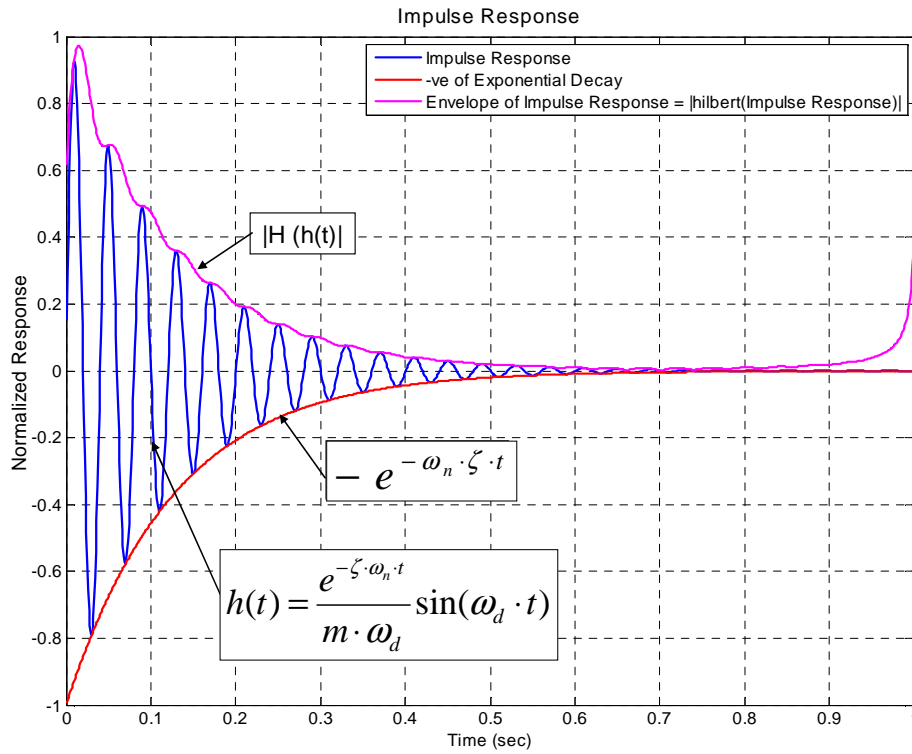
Figure 2: Impulse response and exponential decay of 1DOF system

    Lyon [1], in his book titled "Theory and Application of Statistical Energy Analysis", suggests that the decay rate is easily determined when the response signal is plotted as the log–rms amplitude versus time. The decay rate obtained by the log–rms amplitude is equivalent to decay rate estimated using the log of the exponential decay versus time. In either approach, the decay rate of exponentially decaying envelope – in dB/sec – is measured.

For a single mode and constant loss factor, the dB decay curve is linear with the slope of this curve being equal to the decay rate. (Refer to Figure 2 and 3). The dB decay, of an impulse response, can be calculated by the following equation

$$dB\_Decay = 20 \cdot \log_{10} |H(h(t))| \qquad \text{(eqn.1.5)}$$
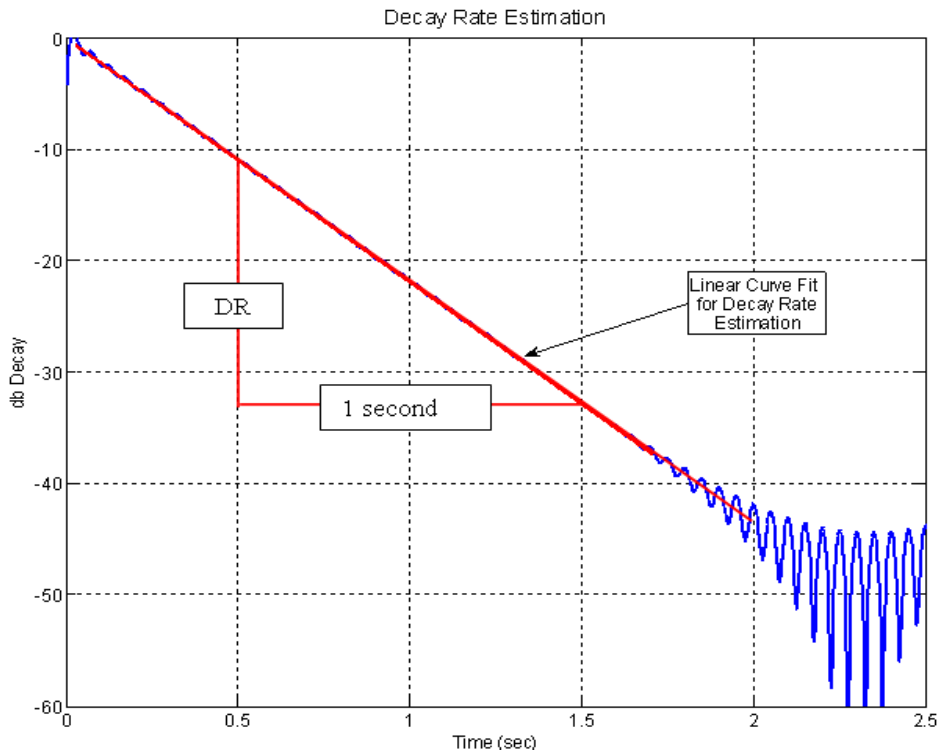
Figure 3: dB decay curve of 1DOF system

The following equation (derived in Chapter 2, section 2.3) provides a relation between decay rate and loss factor:

$$\eta = \frac{DR}{27.3 \cdot f_n} \qquad \text{(eqn. 1.6)}$$

where,

DR is the decay rate (dB/sec)

f_n is the natural frequency or central frequency of a band (Hz).

1.3.1.1 Random Decrement Technique (RDT)

The Random Decrement Technique (RDT) was developed by Henry Cole[5] in the late 1960's. Driven by the motive of "on–line failure detection" by monitoring the damping of a structure (excited by white noise, which is difficult to measure); Cole intuitively came up with a process of averaging the time samples with the same initial conditions to obtain the free decay. Later, Vandiver[6] proved that the averaged response is a free decay only when the excitation is Gaussian, zero–mean, white noise.

The process of evaluating the free decay from the random response, by means of RDT, is entirely in the time domain. RDT, unlike most other techniques, is solely based on response measurements. The underlying principle of RDT lies in the extraction of free decay from random response by averaging a large number of samples with identical initial conditions (or "trigger conditions"). In one such trigger setting, as shown in Figure 4, the local peak in the measured response inside the trigger band is called a "trigger". The trigger sets up the initial condition for the free decay.
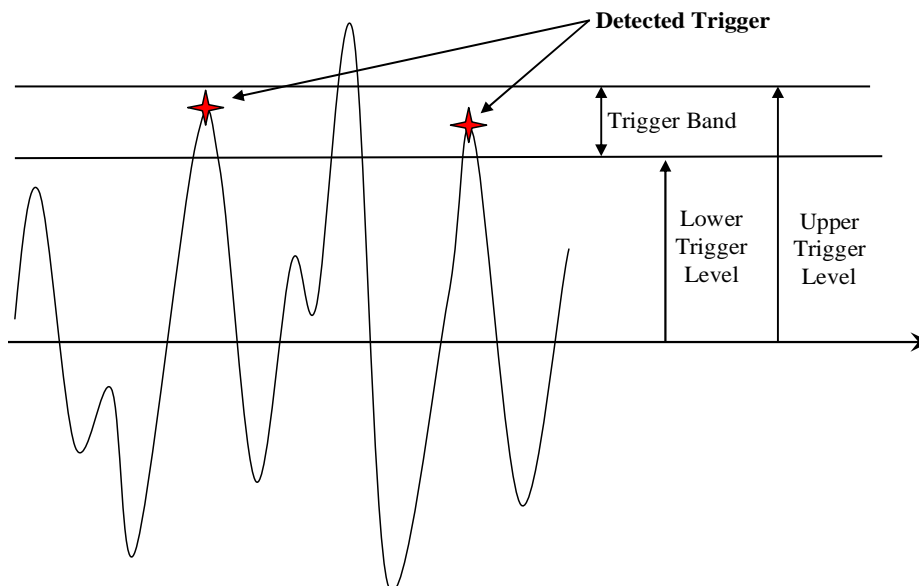
Figure 4: Schematic of a triggering event

In RDT, the time history of response – whether displacement, velocity or acceleration – is monitored and once a triggering event, i.e. pre–specified slope and amplitude, is detected a sample of appropriate length is recorded. When a large numbers of these samples with the same triggering condition are averaged, the random component of the zero–mean response "vanishes" and only the free decay "survives". In Figure 5, first three subplots illustrate the triggered samples from a response filtered in a narrow frequency band. The final subplot is an average of 1000 such samples. Although, some "beating" is observed, the averaged response clearly decays. The beating indicates there are two or more resonances within the narrow frequency band.
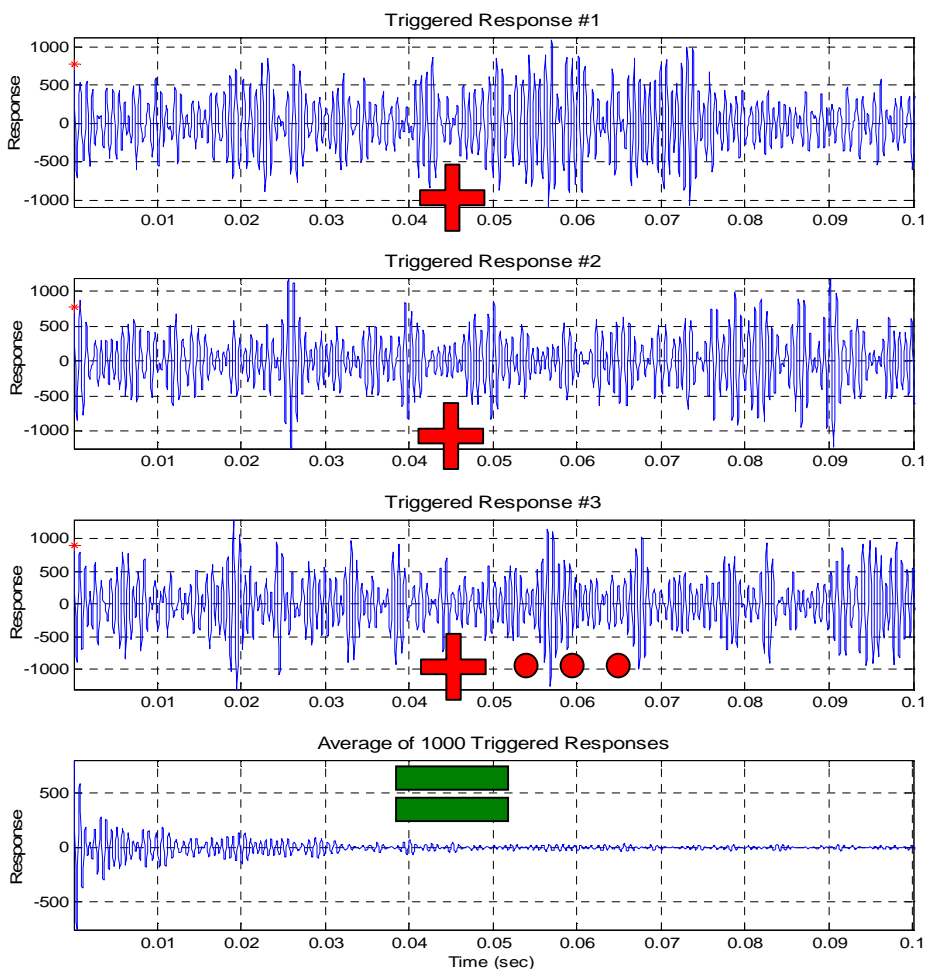
Figure 5: Schematic of the Random Decrement Technique

As RDT is not very popular among structural dynamists, the literature pertaining to the same is very limited with respect to other known techniques, such as IRDM. Mathematical explanations of RDT have already been published, but the limitations and optimum variable settings for experimental analysis are still a matter for study.

Ewing et al [7] have shown that RDT out–performed IRDM when applied to thin plates, especially in the case of high damping loss factor. Theoretically, RDT can be used for any damping levels, but is not recommended for lightly damped structures. The reasons will become clear through discussion presented in subsequent chapters.

1.3.1.2   Impulse Response Decay Method (IRDM)

Impulse Response Decay Method (IRDM) is based on evaluating the narrow–band impulse responses of a structure from the FRF, using inverse Fourier transformations. The excitation can be either steady–state using a mechanical shaker, or transient using an impulse hammer. The FRF is retrieved from the time domain data and/or directly from many turn–key Data AcQuisition (DAQ) systems. Frequency domain filtering, as shown in Figure 6 and explained in detail in section 2.4.3.1.2, is implemented to extract band–limited FRFs which when inverted, in the Fourier sense, will yield a narrow–band decaying impulse response. The loss factor is estimated based on the decay rate of the impulse response.

IRDM is sensitive to the type of hammer tip used for impulsive excitation at a point. The hard tip hammer can be used to excite a wide range of frequencies while the soft tip hammer will excite only lower frequency modes. On the other hand, if a hard tip is used on a very lightly damped structure, exciting the structure with a single "hit" is nearly impossible – double–hits result. If the excitation point is located on a node line, the loss factors will be over estimated.[8,9] With some engineering judgment and precaution taken during experimentation, these errors can be overcome.
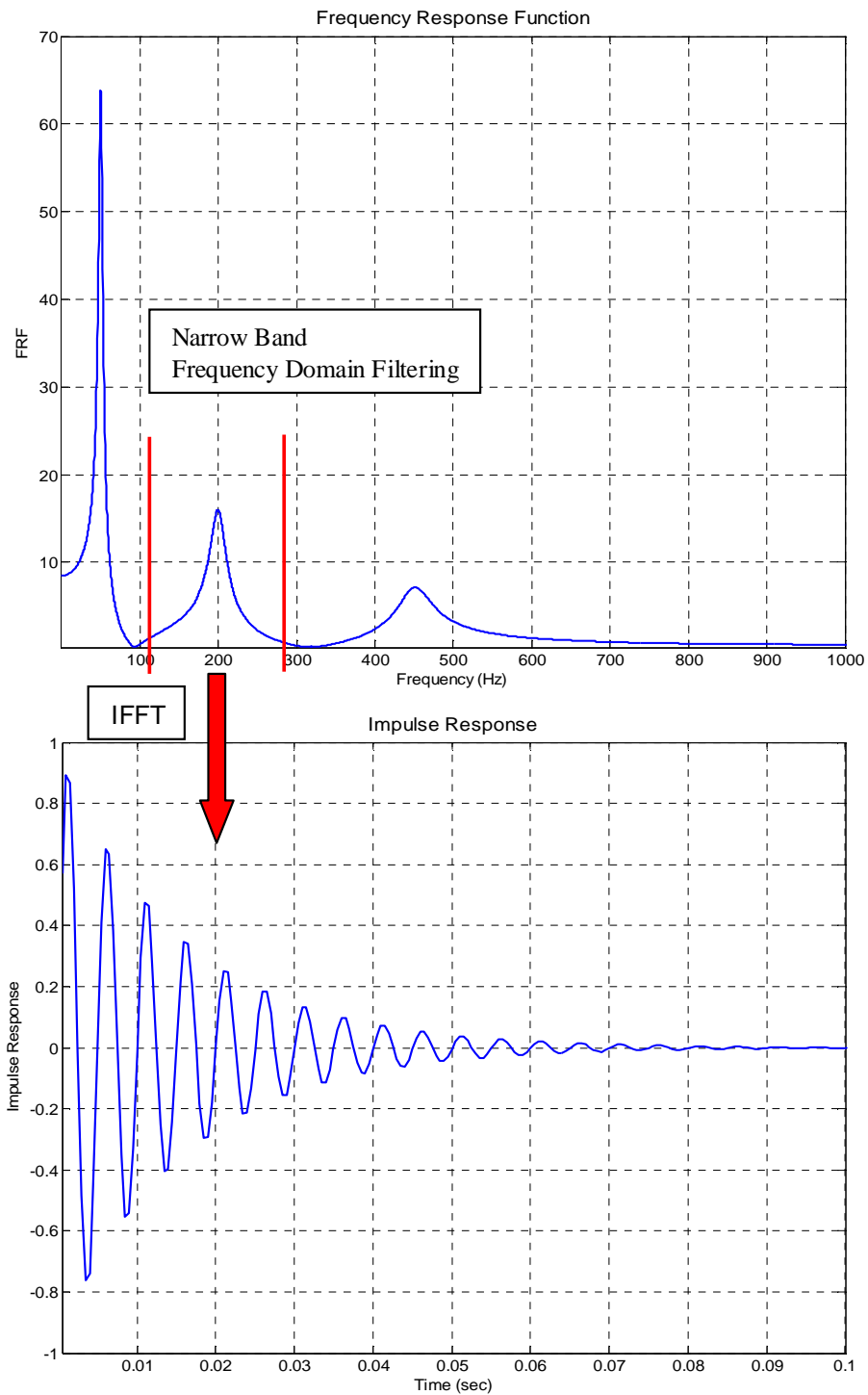
Figure 6: Schematic of Impulse Response Decay Method

1.3.1.3   Reverberation Decay Method (RDM)

Loss factor estimation from the Reverberation Decay Method (RDM) is based on the calculation of the decay rate of free vibrations after the steady state excitation is suddenly interrupted. The structure is allowed to reach a steady state before interrupting the force. When the vibration energy reaches steady state, the energy density at any point comprises two components: one from the direct field (forced vibration) and the other from the reverberant field (free vibration). When the excitation source is "switched–off", the steady state energy density starts to decay.[10] Based on the decay characteristics of the free vibration, the loss factor is computed. Figure 7 represents the theoretical response of a 1DOF system excited by persistent force which is interrupted at the fifth second. The response beyond the fifth second is referred as free vibration (or decay) and its decay characteristics are identical to that of an impulse response.
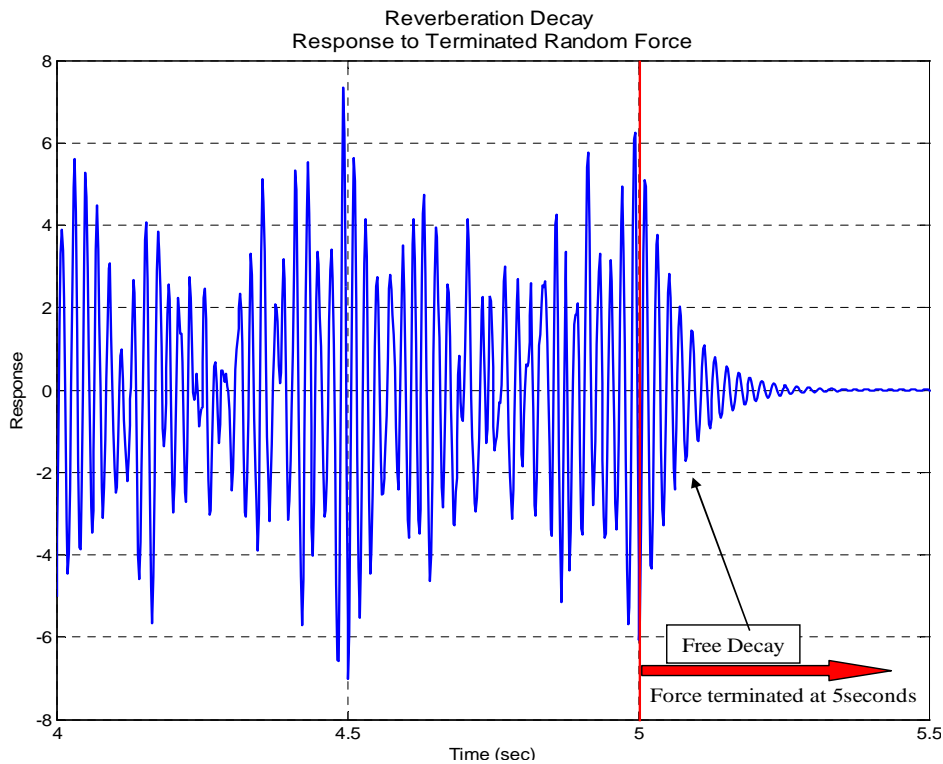
Figure 7: Response to interrupted steady state force – Reverberation Decay Method

1.3.2 Half Power Bandwidth Method (HPBWM)

The Half Power BandWidth Method (HPBWM) is a frequency domain technique, in which the loss factor is estimated from the FRF. The performance of this technique is optimum only for isolated modes for which the half power points can be reliably calculated. In practice this is often the case only for the fundamental (lowest) natural frequency for a structure or structural element. The half power points are 3dB below the maximum magnitude of the isolated mode's FRF. The schematic of HPBWM is presented in Figure 8. The loss factor can be measured from the observed natural frequency and the half power frequencies using the following equation:

$$2 \cdot \zeta = \eta = \frac{f_2 - f_1}{f_n} \qquad \text{(eqn. 1.7)}$$

where,

f_1 and f_2 are the half power frequencies

f_n is the natural frequency.

The FRF is defined as:

$$H(\omega) = \frac{X(\omega)}{F(\omega)} \qquad \text{(eqn. 1.8)}$$

where,

$H(\omega)$ is the Frequency Response Function (FRF)

$X(\omega)$ is the Fourier transformation of response

$F(\omega)$ is the Fourier transformation of force.
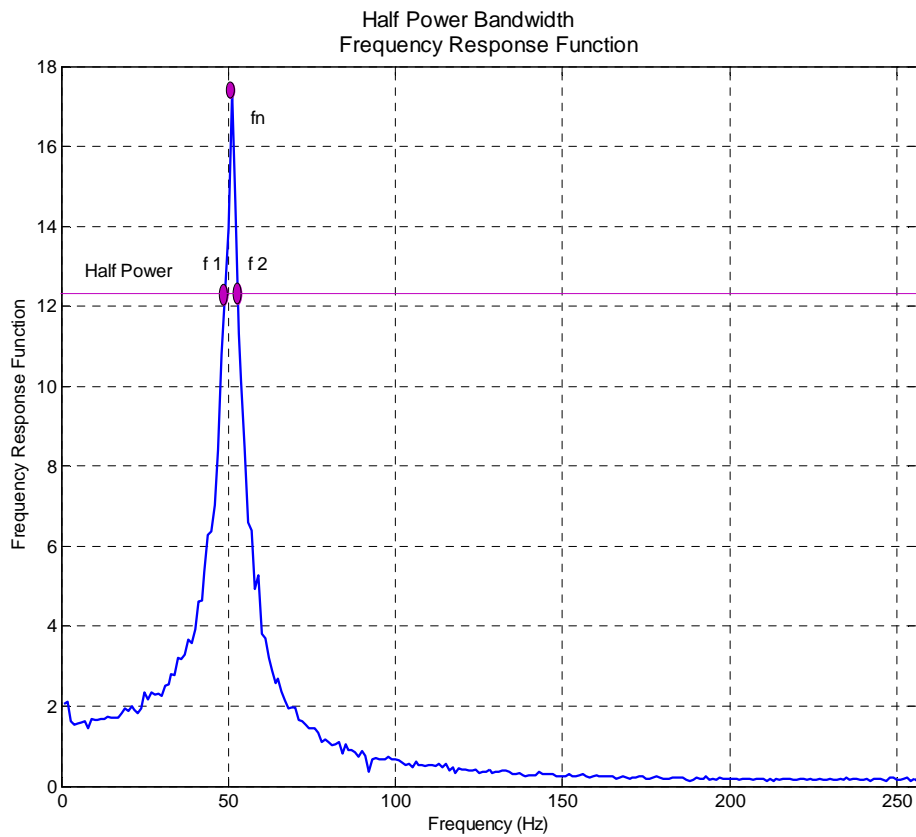
Figure 8: Schematic of Half Power BandWidth Method

The HPBWM is relatively simple and relies only on the measurement of FRF, which can be obtained as an output from most commercially–available data acquisition software. Experimental measurements, in general, do not yield a smooth and noise–free frequency response function. In order to obtain a smooth FRF, the response signal needs to be filtered and FRFs have to be averaged. For accurate estimation of loss factor, especially for lightly damped structures, the frequency resolution of the measured FRF should be high enough to capture the peak in the FRF and the half power frequencies. To reduce experimental error, the value of loss factor should be averaged over multiple excitation and response points on the structure.

If a structure is symmetric or close to being symmetric, there is a high probability of multiple peaks existing in a small frequency range, and separation of these "close modes" might not be simple. Two close modes will result in a beating phenomenon, which is recognizable in the time domain record. This technique is not recommended when either the FRF of the structure is modally dense in the desired frequency band, i.e. when it is difficult to separate the modes, or if the structure has a high loss factor.

1.3.3 Power Input Method (PIM)

Principles involved with Power Input Method (PIM) are based on the very definition of the loss factor, and hence there is no theoretical limitation to this technique. The loss factor, using PIM, is defined in terms of dissipated power and total energy through the following relations:

$$\eta = \frac{P_D}{\omega \cdot E_{Tot}} = \frac{E_D}{2 \cdot \pi \cdot E_{Tot}} \qquad \text{(eqn. 1.9)}$$

where,

η is the loss factor

ω is the frequency in radians/sec

P_D is the dissipated power

E_D is the dissipated energy

E_{Tot} is the total mechanical energy.

If a structure is excited to a steady state condition, the input power <u>equals</u> the dissipated power, so:

$$\eta = \frac{P_{IN}}{2 \cdot \pi \cdot f_c \cdot E_{Tot}} \qquad \text{(eqn. 1.10)}$$

15

where,

P_D is the dissipated power

P_{IN} is the input power and $P_D = P_{IN}$

f_c is the band's central frequency (Hz).

Since the PIM is based on steady state response measurements, i.e. after the transients have died out, a continuous source of power, or excitation, is required for the process. Frequency domain filtering is typically implemented to estimate loss factors in narrow frequency bands, thereby establishing the frequency–dependency of the loss factor. The accuracy of loss factor estimates along with the number of required measurement points has been a topic of discussion in many papers. With appropriate care taken during experimentation, PIM can be used for all frequency ranges and damping levels.[2]

1.3.4   Common Issues Associated with Experimental Loss Factor Estimation

It is important to understand the requirements and limitations of methods involved with loss factor estimation as no one method is applicable to all situations. Most of the techniques fail to accurately estimate high loss factors, particularly when insufficient or limited information is available.[11]

The positioning of sensors and excitation points plays a critical role when taking measurements from multiple input and output locations on the structure. In general, it is important not to excite a structure at a node line, which may result in rigid body response or to measure response from a point on node line. To efficiently simulate a continuous system and remove spatial effect in measurements, it is suggested to record and analyze long sets of time history of responses from multiple locations, preferably randomly chosen, on the test specimen.

For analysis involving multiple input–output locations on the structure, slope fitting can be another tricky and sensitive issue. If a large number of slope fits are to be executed by an operator, human error is inevitable and the results may not be reproducible. Hence, it is recommended to use an automated slope fitting algorithm. All the slope fits on the decay curves should be at least visually inspected to avoid possibility of a biased error.

For the experimental evaluation of the IRDM and RDT, LabVIEW "virtual instruments" are used. Further experiment–related variables and processing issues are discussed in detail in Chapter 2.

1.4   Scope of Work

The foremost purpose of this research is to establish RDT as a main–stream method for loss factor estimation, by understanding the effect of variables associated with the technique and recommending process variables for them. This is accomplished by simulating the response of a spring–mass–damper (1DOF) and a plate, as a simple representation of Multiple Degree of Freedom (MDOF) systems.

To establish RDT for MDOF systems, a model of a thin plate is created in MSC/ NASTRAN and used to study the parameters affecting the loss factor estimation. Frequency domain and time domain filtering approaches have been implemented and are compared on equal grounds. In the experimental analysis of plates the loss factor estimates from the RDT are compared to estimates from IRDM. Methods and results published herein and the other papers published by KUAE authors on RDT will give structural dynamists a better insight into the effect of the process variables associated with experimental loss factor estimation. During this research, two versatile and powerful data acquisition and analysis codes were written. These software codes, which incorporate the results and conclusions drawn from this research, can perform the analysis in real time**.**

RDT has been successfully implemented to estimate loss factors, and the technique performed equally well or better than IRDM for high damping loss factor estimation. Though the objective of better understanding of RDT has been achieved by providing closed–end solutions to some of the problems associated with it, additional improvements are possible.

1.5   Thesis Layout

RDT and the theory associated with it is examined and discussed in Chapter 2. This chapter begins with a literature review of previously published work on RDT and is followed by the mathematical foundations related to the technique. Chapter 2 includes the discussion of possible process variables, such as triggering and filters settings. These variables could significantly affect the accuracy of the loss factor estimated from RDT and hence, these variables are analyzed for implementation of RDT to computational and experimental models of MDOF systems. Finally, Chapter 2 is concluded with sections on possible averaging schemes and experimental set up.

Chapter 3 is aimed at the evaluation of loss factor estimates using RDT. The results are divided in three parts: first, is the study of a simulated 1DOF system; second, is the study of MDOF computational plate model; and finally, the experimental loss factor estimation and comparison of loss factor estimates from RDT with IRDM.

In Chapter 4 conclusions, from the results, are discussed. Chapter 5 very briefly provides the recommendations for future studies. This is followed by list of references and appendices. The appendices include MATLAB codes and a detailed comparison of loss factor estimates from frequency and time domain filtering approaches.

2 Theoretical Background of Random Decrement Technique

The purpose of this chapter is to provide a detailed description of the principles and the variables associated with the Random Decrement Technique (RDT) for experimental loss factor estimation. A literature review of the previously published work and the mathematical foundation of RDT are also included in this chapter. The studies of parameters important to RDT are presented for both SDOF and MDOF sytems. Latter sections contain the description of experimental setup required for loss factor estimation using RDT.

From the analysis of free decay, presented and discussed in Section 2.4.2.2, it is noted that the normalized acceleration and displacement have the same decay rate. Thus, the loss factor computed from these decay rates will also be equal. Henceforth, the term "response" is generalized to apply to acceleration, velocity or displacement measurements.

2.1 Random Decrement Technique

Response of a system to Gaussian white noise excitation is theoretically equal to the time domain convolution of force and impulse response function. When a large number of selected samples from a very long time history of random response measurement with the same initial conditions are averaged, the random component of response averages out to zero and only the free decay (or, "Randomdec Signature") survives. The procedure of extracting the free decay from the random response is entirely in the time domain (Refer to Figures 4 and 5). The choice of the "trigger" or initial conditions, length of samples, filter settings and averaging schemes is a central part of this research.

RDT, unlike IRDM and modal curve–fitting techniques, is based solely on the response of a system to random excitation. As such, one's expectations for loss factor estimation using RDT might be somewhat low. In fact, Ewing et al [7] have shown that RDT

out–performs other techniques based on input–output relationships for panels with very high damping loss factors (≈ 0.1). For such highly– damped structure, there can be a significant portion of the structure which is not experiencing "reverberant field" conditions, especially in higher frequency bands. That is, localized disturbances "die–out" before they can reflect off boundaries, thereby establishing a reverberant field and "vibration". Therefore, the usual input–output relationships of vibration are not in force. Specifically, the response is higher than what would be expected in a reverberant field where the energy is, more or less, uniformly distributed. Instead, energy is flowing away from the excitation point, so the energy level in the "near field" (near the excitation) is higher than elsewhere, resulting in a higher response level than when vibration is occurring. Since response is inversely proportional to damping, the loss factor predicted by vibration theory is therefore underestimated.

The advantage of RDT stems from the fact that it does not depend on an input–output relationship. Instead, the temporal decay rate characteristic of a point on a structure with a given loss factor is also the decay rate of that point's response autocorrelation function. Actually, most implementations of RDT do not estimate the autocorrelation function, but do estimate what amounts to the positive portion of it, which possesses the same decay rate. It simply does not matter if a response is measured inside the near field or not. Processes like IRDM and modal curve–fitting are, presumably, best implemented by not including response measurements in the near field.

For the computational studies herein, the analytical FRF is convolved with a random force, producing an arbitrarily–long decaying response signal. This signal is then used for the decay signature extraction. The filtered response, in specified (typically octave–wide) frequency bands, is then decomposed to an ensemble of responses with same triggering event.

In the discussed analyses the trigger setting refers to attainment of response in a pre–selected amplitude band with zero slope. The average of this set of triggered responses, theoretically, removes the effect of the randomness of excitation.

2.2 Literature Review

RDT is simple and easy to implement for identification of vibration characteristics of structures acted upon by stochastic forces. The technique has not been widely used in damping loss factor estimation and only a handful of people have published papers on the theoretical description and application of RDT. Hence, the following brief literature review is categorized based on authors and their contributions to the technique.

Cole H. [5, 12]

Henry Cole, in the late 1960s, published the first paper on RDT. He was looking for "a simple and direct method for translating the response time history into a form meaningful to the observer".[22] Cole was trying to develop a method for on–line failure detection in structures by measuring the change in vibration characteristics, such as the decay rate of a structure. For example, if a bolt comes loose, the decay characteristics will change. If this change in damping can be measured with respect to the unflawed structure, the defect can be detected well in advance of failure. Cole found it difficult to measure ambient loads acting on the structure, inspiring him to develop a method based on the measurement of response only. Three of the main conclusions from Cole's NASA contractor report [5] and paper [12] are:

1. The randomdec signature is equivalent to the free vibration decay, with initial conditions equal to the trigger settings.

2. The randomdec signature is stable in form and scale, under ambient loading; thus, it can be used for on–line failure detection and damping measurement.

3. The randomdec signature from narrow–band time histories is a more efficient way of estimating damping or failure than the input–output based approach.

Cole was also successful in estimating damping ratio for a lightly damped non–linear systems.

Vandiver J. K. et al [6]

The development of RDT by Cole was primarily intuitive. In 1982, Vandiver et al [6] provided a mathematical basis for RDT. They proved that the randomdec signature, in general, is not equivalent to the free decay. In fact, the randomdec signature is proportional to the autocorrelation function. The randomdec signature is equal to free decay only for the 1DOF system, excited by Gaussian white noise. These mathematical relations from Vandiver et al's work are included in the subsequent subsections.

Ibrahim S. R., Brincker R. and Asmussen J. C. [11, 13–22]

As individuals and together, Ibrahim, Brincker and Asmussen have published a substantial number of papers on RDT. Asmussen's dissertation [22], on the application of RDT for modal analysis, has the best mathematical description of RDT and the parameters associated with it. Brincker et al [13] have provided an excellent explanation of the mathematics and methodologies for estimation of autocorrelation and cross–correlation using RDT. Ibrahim also came up with a more general vibration analysis technique known as the "Ibrahim Time Domain Modal Testing Technique". Asmussen, Brinker and Ibrahim also proposed a vector triggering technique as an alternative, viable option to RDT. The Vector Random Decrement (VRD) functions are equivalent to free decays. The study of VRD technique was considered to be beyond the scope of this research and is not studied as a part of this thesis.

Kijewski T. and Kareem A. [23, 24]

Kijewski and Kareem published their work on the reliability of RDT for the estimation of structural damping when some of the trigger conditions, associated with RDT, are relaxed. Allowing overlapping is another such condition in which two or more consecutive triggered samples share some part of the time history. They concluded that the effect of overlapping on loss factor estimation is marginal. In this research only non–overlapping samples, i.e. 0% overlap or uncorrelated triggered samples, were employed to evaluate the randomdec signature. In their research, they used the triggering conditions in its "strictest sense", i.e. selecting only the response peaks in a particular amplitude window. Their triggering approach is adapted for the research on RDT provided herein.

2.3 Theoretical Background

The randomdec signature, D_{x0}, is proportional to the autocorrelation function, $R_x(\tau)$, of the system according to following equation [6]:

$$D_{x0}(\tau) = x_0 R_x(\tau) / R_x(0) \qquad \text{(eqn. 2.1)}$$

where,

$\tau = t_2 - t_1$ is a dummy (or secondary) time variable

t_2 and t_1 corresponds to values of time 't' on linear time scale

x_0 is the initial displacement, or trigger's amplitude level

$\dot{x}_0$ is the initial velocity, or trigger's slope; if peaks in the response are selected as trigger then, $\dot{x}_0 = 0$.

For a single degree of freedom system, excited with Gaussian, zero–mean, white noise, the randomdec signature is equivalent to the autocorrelation function, as given in equation 2.2, below.

$$D_{x0}(\tau) \equiv E[x(t_2) \mid x(t_1) = x_0 \, and \, \dot{x}(t_1) = \dot{x}_0] \qquad \text{(eqn. 2.2)}$$

The autocorrelation function can be normalized to represent the following form:

$$R_x(\tau) = Ce^{-\zeta\omega_n\tau}\cos(\omega_d\tau) \qquad \text{(eqn. 2.3)}$$

where,

$C = x_0 / R_x(0)$

ω_n is the natural frequency

ω_d is the damped natural frequency

ζ is the damping ratio

$x(t)$ is the measured response.

Comparing equation 2.1 and 2.3, it can be concluded that the randomdec signature is equivalent to free decay for a 1DOF system excited with Gaussian, zero–mean, white noise. [24] Figure 9 represents a response record satisfying a trigger event (top figure), the resulting autocorrelation function and the magnitude of the Hilbert transformation of the positive portion of the autocorrelation function (bottom figure).

The Gaussian white noise has a constant power spectral density but random phase. The Figure 10 illustrates the frequency domain characteristics of a simulated band–limited random force. In this figure, the power spectral density of force is constant, non–zero, in the frequency range of 10 Hz to 200 Hz. The random response generated from this force will also be random and in a narrow–band of 10 Hz to 200 Hz.
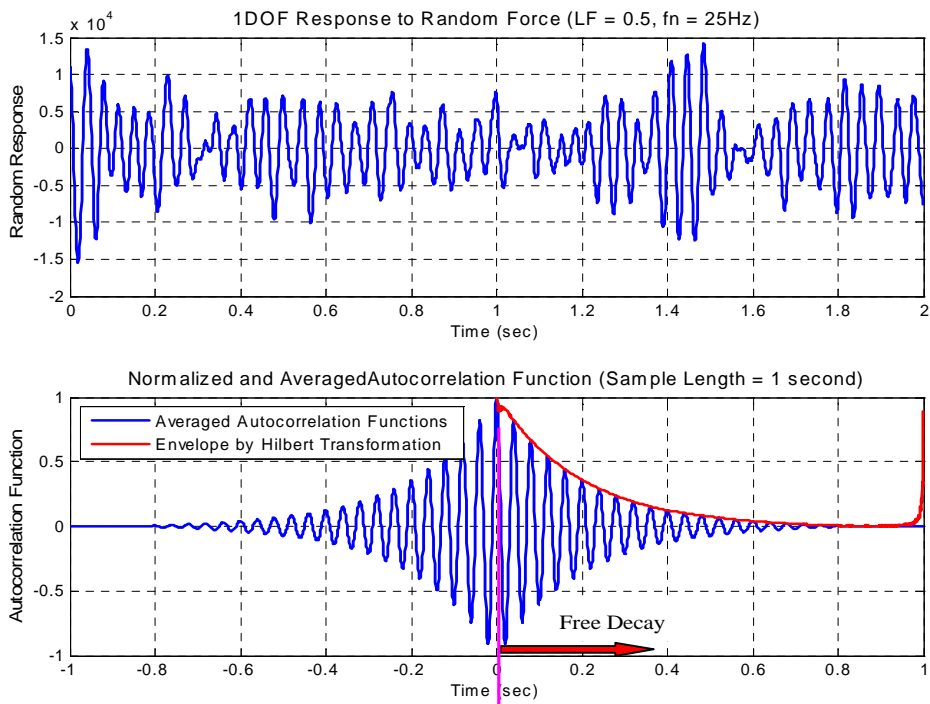
Figure 9: Schematic of process of loss factor estimation using autocorrelation functions
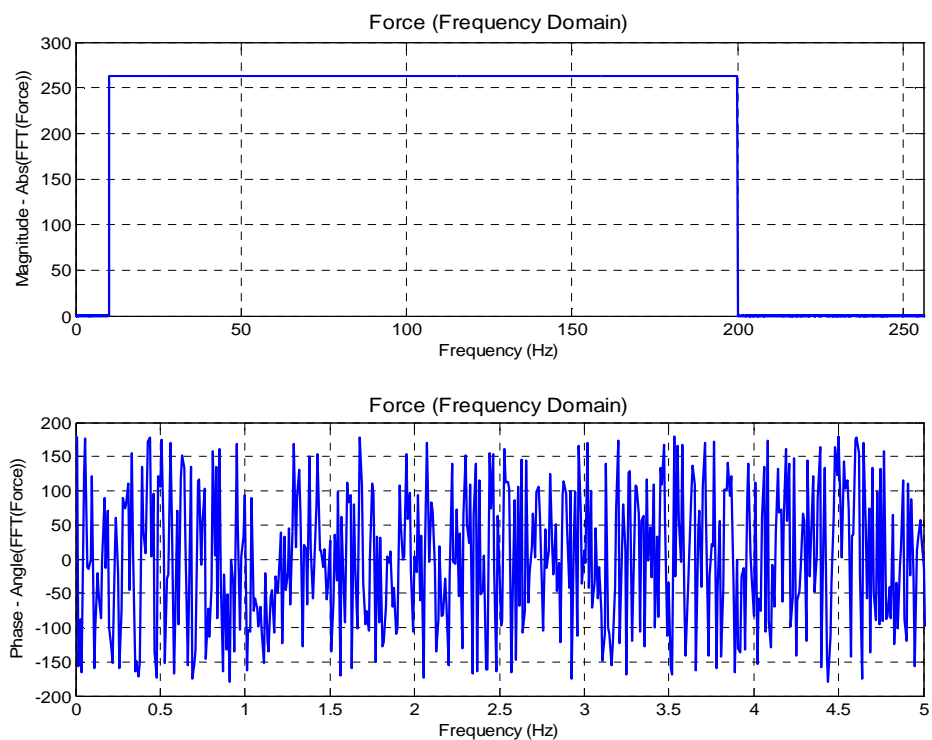
Figure 10: Gaussian white noise – Frequency Domain

A random response, as shown in Figure 11, can be generated by convolving a random force with the impulse response of a 1DOF system. To estimate the free decay characteristics of the system, using the RDT, narrow–band filtering needs to be implemented. If the random response is filtered using a narrow–band filter the out–of–band frequencies in the random response are eliminated. Filtering options are discussed in Section 2.4.2.3. The filtered response of the response given in Figure 11 by using a filter centered at 200 Hz, is plotted in Figure 12. Note the resulting nearly uniform rate of zero–crossing indicative of narrow–band response.
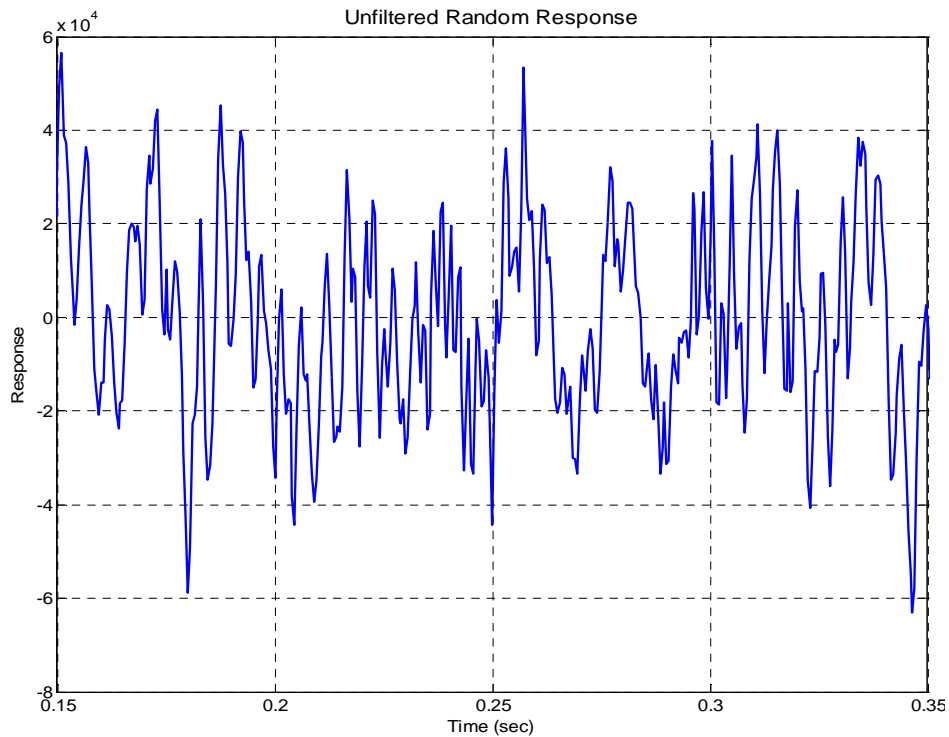
Figure 11: Unfiltered random response of a 1DOF system with f_n = 200 Hz
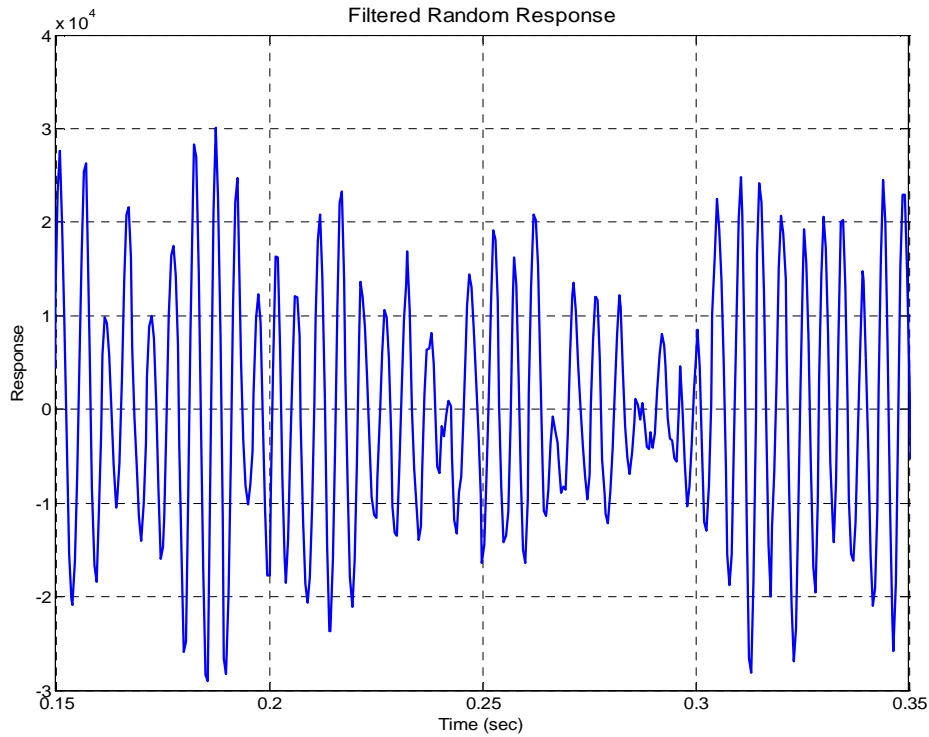
Figure 12: Filtered random response using bandpass filter with center frequency of 200 Hz (for a 1DOF system with f_n = 200Hz)

The impulse response function for a 1DOF system is mathematically described as:

$$h(t) = \frac{e^{-\zeta \cdot \omega_n \cdot t}}{m \cdot \omega_d} \sin(\omega_d \cdot t) \qquad \text{(eqn. 2.4)}$$

where,

$h(t)$ is the impulse response function

m is the mass of the system.

The exponential decay, $e^{-\zeta \cdot \omega_n \cdot t}$, for a 1DOF system subjected to non–zero initial displacement and zero velocity, is identical to the exponential decay of the impulse response function. From the Hilbert transformation of decaying response, the exponential decay or envelope can be computed. The loss factor can be then calculated by measuring the decay rate of this envelope and using the following relation [1,8]:

$$\eta = \frac{DR}{27.3 \cdot f_n} \qquad \text{(eqn. 2.5)}$$

where,

f_n is the 1DOF system's natural frequency; when dealing with a MDOF system, f_n is

replaced by f_c – the band's central frequency

DR is the decay rate of the response and Hilbert transformation of the response in dB/sec.

Equation 2.5 can be derived as follows:

$$DR = \frac{\Delta H}{\Delta t}$$

where,

H is measured in decibels, that is:

$$H_1 = 20 \cdot \log_{10}(e^{-\zeta \cdot \omega_n \cdot t_1})$$

$$H_2 = 20 \cdot \log_{10}(e^{-\zeta \cdot \omega_n \cdot t_2})$$

$$\Rightarrow \Delta H = H_2 - H_1 = -\zeta \cdot \omega_n \cdot t_2 \cdot 20 \cdot \log_{10} e - (-\zeta \cdot \omega_n \cdot t_1 \cdot 20 \cdot \log_{10} e)$$

$$\Delta t = t_2 - t_1$$

$$\omega_n = 2\pi \cdot f_n$$

$$\Rightarrow \zeta = \frac{\Delta H}{\Delta t \cdot 2\pi \cdot f_n \cdot 20 \cdot \log_{10} e}$$

$$20 \cdot 2\pi \cdot \log_{10} e = 54.5757$$

$$\Rightarrow 2\zeta = \eta = \frac{DR}{27.288 \cdot f_n}$$

It is evident from equation 2.5 that the decay rate is proportional to the loss factor and is inversely proportional to the band's central frequency. This implies that the free vibrations

for lightly damped structure will last longer as compared to free vibration of a highly damped structure. The process of computing loss factor from the decay rate of impulse response or free decay is discussed in section 1.3.1 and is presented in the Figure 13, below.

Impulse Response

$$h(t) = \frac{e^{-\zeta \cdot \omega_n \cdot t}}{m \cdot \omega_d} \sin(\omega_d \cdot t)$$

Exponential Decay = Absolute of Hilbert Transform of Impulse Response

$$e^{-\zeta \cdot \omega_n \cdot t}$$

dB Decay

$$\eta = \frac{DR}{27.3 \cdot f_n}$$

Time (sec)
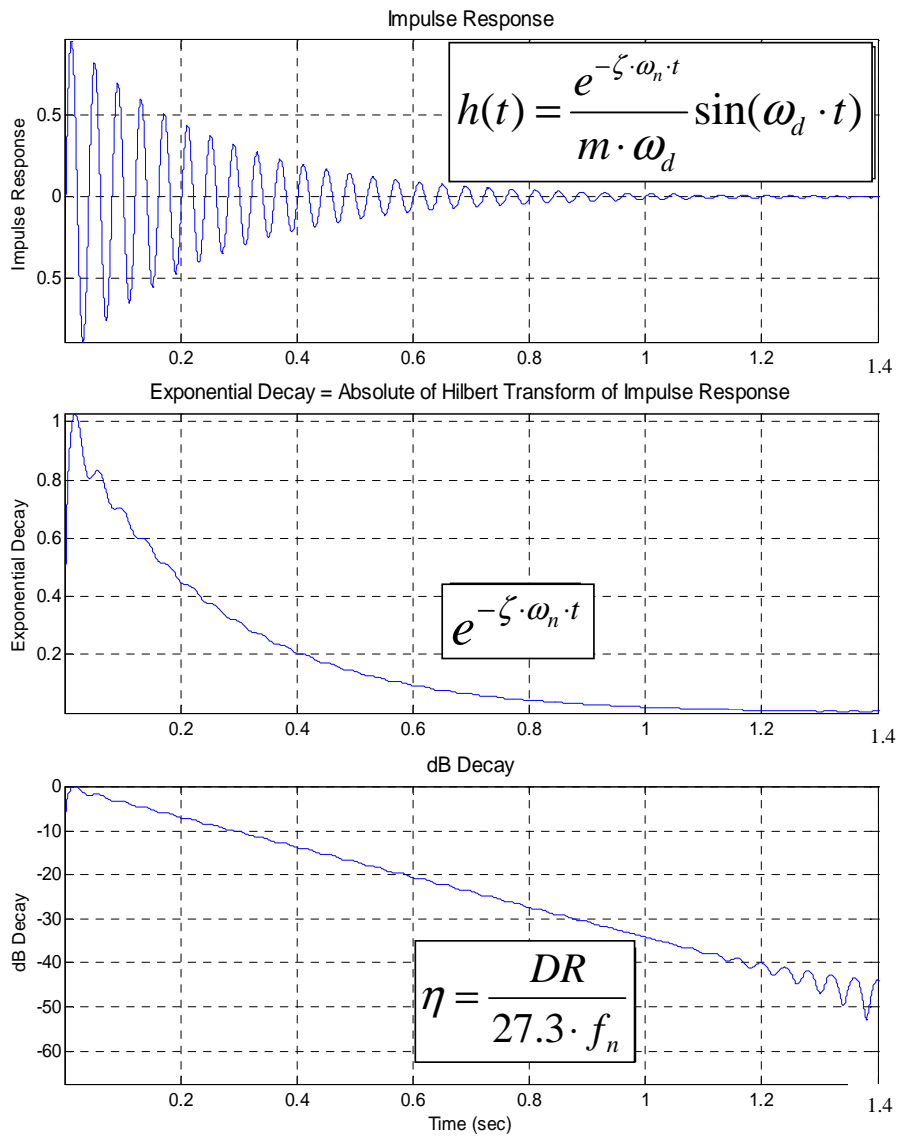
Figure 13: Schematic of process of damping loss factor estimation

2.4   Processing Parameter Study

For the efficient use of RDT, the data processing parameters associated with the technique need to be understood. Some of these parameters play a significant role in accurate estimation of loss factor. The parameter settings are judged on two criteria: first, their effect on the accuracy of estimated loss factor; and second, their effect on the data processing efficiency. Based on the above two criteria, the optimum parameter settings are chosen for implementation in the experimental validation of RDT.

2.4.1   Research Methodology

The process parameters and their impact on accuracy of loss factor estimate using RDT are evaluated on three different models:

1. Spring–Mass–Damper (1DOF) simulations

2. Finite Element Analysis (FEA) of a thin rectangular plate model – using MSC/ NASTRAN to compute the FRFs

3. Experimental validation by comparison with IRDM estimates of loss factor using three damped plates.

For a 1DOF system, the effect of sample length, trigger settings and type of averaging (direct averaging or averaging of the autocorrelation functions) are examined. The optimized settings from analysis of a 1DOF system are used in the study of the computational plate model. If a particular value of a variable leads to a poor loss factor estimate in the analysis of the 1DOF system, that particular variable setting was not used in the analysis of the MDOF system.

The computational plate used in this study has 16 output locations at which the FRFs are computed. The critical findings, regarding the process parameters, of this computational

study form the basis of experimental analysis of a real structure. The experimental evaluations of RDT were accomplished using 3 DAMPED plates, with different damping levels. Finally the loss factor estimates from RDT and IRDM are compared.

2.4.2   Single Degree of Freedom System

In the following sections, the first study of parameter settings for efficient implementation of RDT in loss factor estimation is discussed.

2.4.2.1   Damping Loss Factor Range

Most of the experimental loss factor estimation techniques rely on the assumption of viscous damping. Typically, for structures treated with constrained layer damping, the loss factor will range between 0.01 and 0.1.[1] These empirical damping loss factors for thin steel and aluminum plates are represented in Figure 14 (adapted from Figure 9.4 of Reference 1). Thus, computational plate models with loss factors of 0.01 and 0.1 were simulated. The plates analyzed experimentally also have estimated loss factors in the same range.
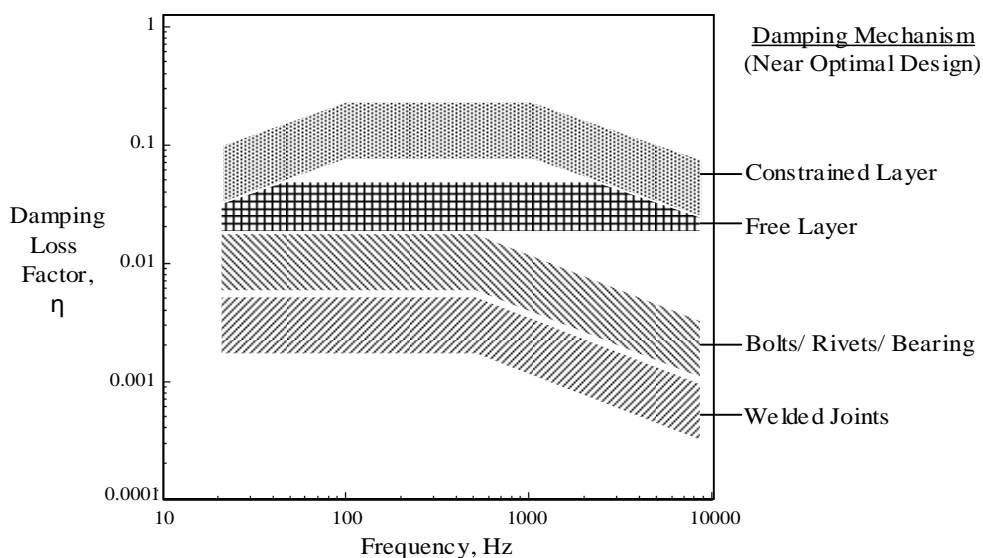
Figure 14: Empirical damping loss factors for steel and aluminum thin plates and shells in the size range of 0.1 and 10m.

2.4.2.2 Length of Sample

In an experimental analysis, the length of sample should be long enough to capture the complete decay of the randomdec signature, i.e. until the measured response is at the noise floor. The length of triggered sample and the number of triggers will dictate the length of response measurement to be recorded for data analysis. To optimize the memory usage and reduce computation time, a target sample length needs to be established. As discussed in earlier sections, the decay rate is proportional to loss factor and hence the length of sample can be expressed as a function of loss factor.

To determine the target length of sample, free decays for different damping levels and frequencies are plotted. In experimental analyses, the response of the plate will be measured by an accelerometer. Therefore, the normalized acceleration's free decay for the same frequency and damping levels are plotted, along with normalized displacement, in Figure 15.

For a 1DOF system, the transient displacement response is given by:

$$x(t) = e^{-\zeta \cdot \omega_n \cdot t} \cdot \sin(\omega_d \cdot t + \phi) \qquad \text{(eqn. 2.6)}$$

and transient acceleration response is given by the following relation:

$$\ddot{x}(t) = e^{-\zeta \cdot \omega_n \cdot t} \cdot \left\{ (\zeta^2 \cdot \omega_n{}^2 - \omega_d{}^2) \cdot \sin(\omega_d \cdot t + \phi) - 2 \cdot \zeta \cdot \omega_n \cdot \omega_d \cdot \cos(\omega_d \cdot t + \phi) \right\}$$

$$\text{(eqn. 2.7)}$$

where,

$$\omega_d = \omega_n \sqrt{1 - \zeta^2}$$

It can be noted that the exponential decay of displacement and acceleration, from equation 2.6 and 2.7, are equal to $e^{-\zeta \cdot \omega_n \cdot t}$. Therefore, for a constant damping level, the

numbers of cycles for vibrations to damp–out will remain the same in all the frequency bands.

For one such case, when the damping loss factor is 0.1 and the frequency range is 100 Hz to 6300 Hz, the transient response of displacement and normalized acceleration are plotted in Figure 15.
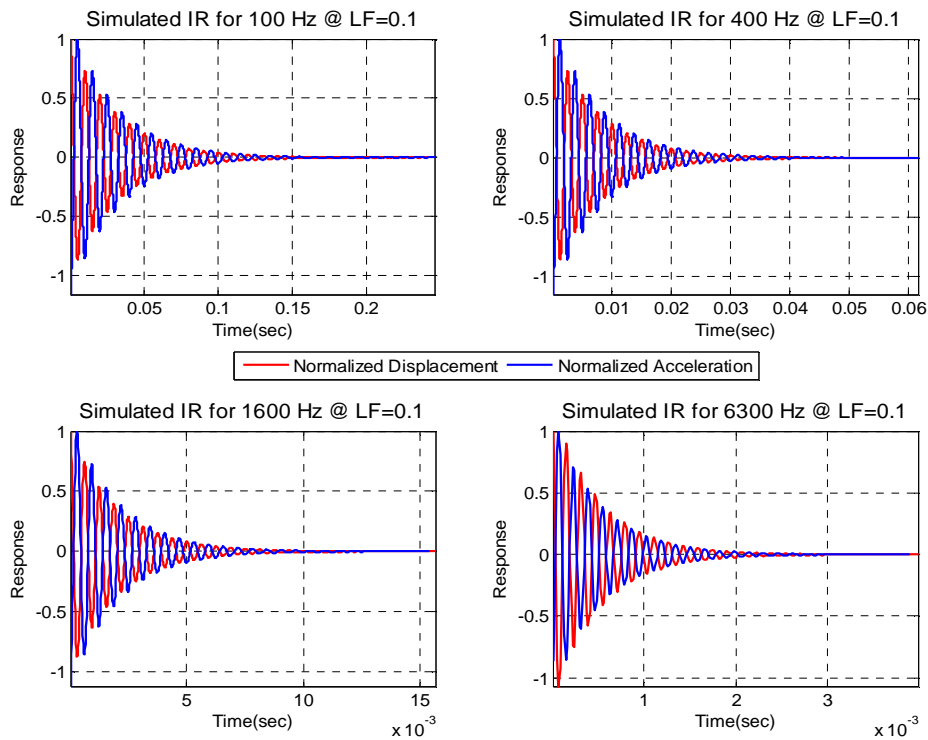
Figure 15: Comparison of normalized impulse response of displacement and acceleration

To verify that the acceleration and response will yield the same decay rate, and in turn the same loss factor, the dB decay curves for acceleration and displacement are plotted in Figure 16. For a defined loss factor and central frequency, the decay rate for displacement and acceleration appear identical. Further, this study was performed for multiple damping loss factor values ranging between 0.0025 and 0.5.
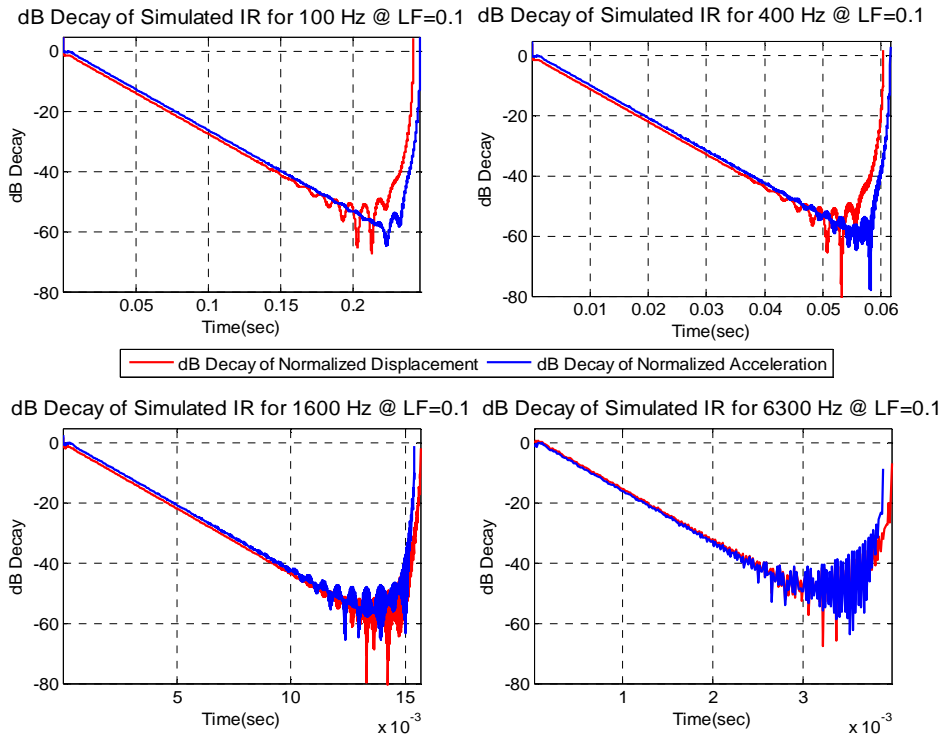
Figure 16: Comparison of dB decay of displacement and acceleration

It is concluded that for a particular level of damping, the length of sample in terms of number of cycles, can be established as a fixed parameter for the entire frequency range. The number of cycles to damp–out versus the damping loss factor is plotted, on a log–log plot, in Figure 40 of section 3.1.1. The empirical relation, equation 2.8, is derived from the slope of the curves presented in Figure 40. The optimum length of sample (or triggered sample) to record the complete decay of the free response is given by:

$$N \cong \frac{2.2}{\eta_{est}}$$

(eqn. 2.8)

where,

N Length of sample in terms of cycles, for narrow–band time history

η_{est} Initial "guess" of the loss factor.

Equation 2.8 indicates that the length of sample in terms of cycles is independent of the frequency and is only dependent on the loss factor. That is, for a lightly damped structure, the length of sample will be longer than the length of sample for a highly damped structure. The length of measured response should be at least equal to the cumulative length of all the triggered responses required to be averaged to compute the randomdec signature (because the triggered samples have no overlap). This is an important observation, especially for lower damping levels or central frequencies, for which very long time records are needed to be recorded and analyzed.

2.4.2.3   Filter Selection

If the structure is excited by random white noise, the response will also be random in nature. To extract a response in a narrow frequency band, or for a single mode, band pass filtering needs to be implemented. Computation of a randomdec signature from narrow–band time histories is statistically more efficient.[12] By filtering the measured response, noise and the other out–of–band information can be removed and the resulting randomdec signature will be "cleaner".[15,16]

Filtering is the first step to condition the data for analysis; therefore, it should be understood in detail. During the analysis of RDT, it is observed that filtering is one of the most time consuming process, especially when the random response is required to be filtered in multiple frequency bands.

In general, filters can be classified as Infinite–duration Impulse Response (IIR) filter and Finite–duration Impulse Response (FIR) filters. Unlike the IIR filters, the FIR filters are always stable in form. FIR filters can have exactly linear phase and the design methods are generally linear.[25] Kaiser window filters (FIR) were chosen over the other FIR filters because of the relatively better control over the filter design parameters and ease in

implementation. All the filters used in this study are designed using MATLAB's Filter Design and Analysis Tool (FDA Tool), which is represented in Figure 17.
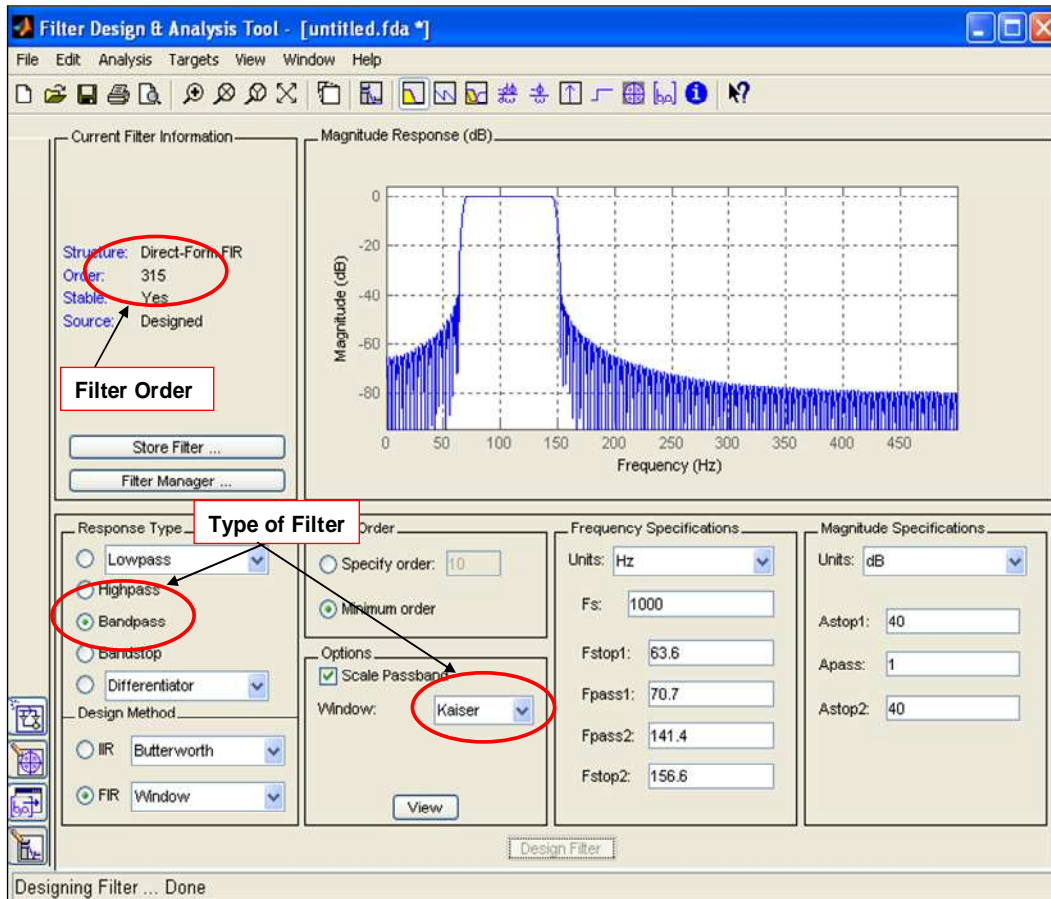
Figure 17: FDA Tool design window for The Mathworks MATLAB R2009a

In the field of structural dynamics and acoustics it is common to analyze a signal in frequency bands with constant percentage bandwidths, for example full and one–third octave bandwidths. In the computational and experimental studies, reported herein, both $1/3^{rd}$ and full octave bandpass filters are studied. The filter in Figure 17 is designed for central frequency of 100 Hz for a full octave bandwidth, that is, 70.7 to 141.4 Hz. The parameters associated with this filter design, to be implemented using the MATLAB's FDA Tool, are explained through the following Figures 18 and 19.
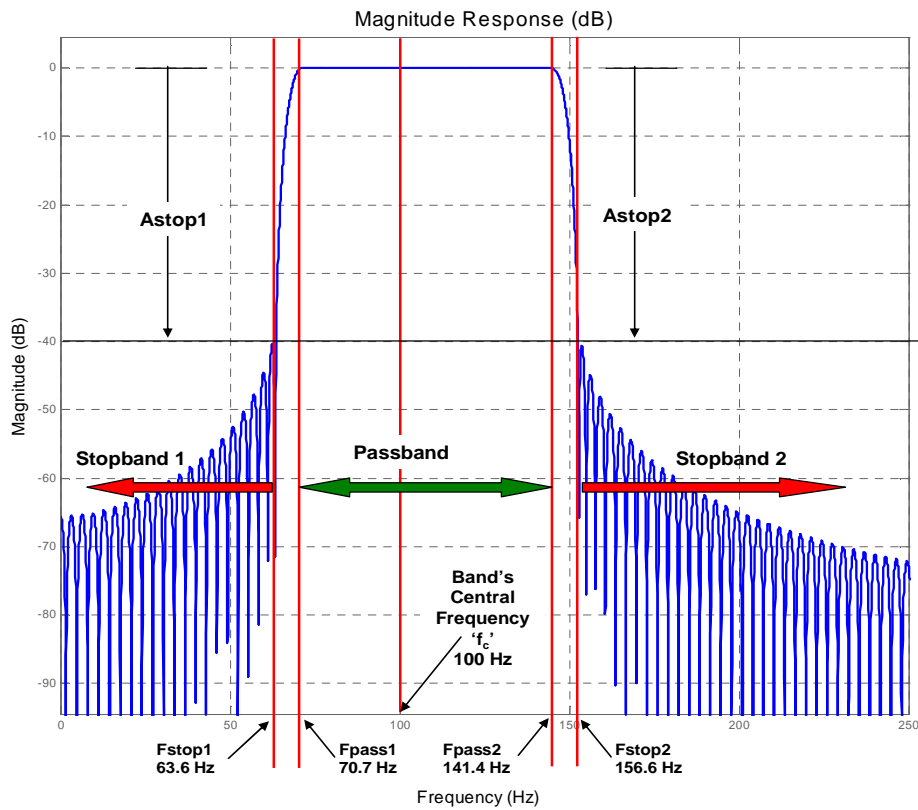
Figure 18: Filter design parameters – MATLAB FDA Tool

For the Figure 18:

Fstop1   is the end of the first stopband (Hz)

Fstop2   is the beginning of the second stopband (Hz)

Fpass1   is the beginning of the passband (Hz)

Fpass2   is the end of the passband (Hz)

Fs   is the sampling frequency (in Hz)

f_c is the band's central frequency (Hz)

Astop1   is the first stopband attenuation (dB)

Astop2   is the second stopband attenuation (dB)

Apass   is the passband attenuation (dB)

Order   is the filter order of the designed filter

For 1/3rd octave bandpass filters "octave factor" or F_{Octave} is $2^{1/6}$.[1]

For full octave bandpass filters F_{Octave} is $2^{1/2}$.[1]

To define the filter boundaries, the "roll–off" frequencies are calculated in terms of the percentage, or $B_{\%}$,of the width of the passband.

$$Fstop1 = (1 - B_{\%}) \cdot f_c / F_{Octave}$$

$$Fpass1 = f_c / F_{Octave}$$

$$Fpass2 = f_c \cdot F_{Octave}$$

$$Fstop2 = (1 + B_{\%}) \cdot f_c \cdot F_{Octave}.$$

The central frequencies, f$_c$, for which the performance of RDT is analyzed, are: 100 Hz, 125 Hz, 160 Hz, 200 Hz, 250 Hz, 315 Hz, 400 Hz, 500 Hz, 630 Hz, 800 Hz, 1000 Hz, 1250 Hz, 1600 Hz, 2000 Hz, 2500 Hz, 3150 Hz, 4000 Hz, and 5000 Hz.

The filter design parameters for 1/3rd octave filters and full octave filters are presented in Figure 19.
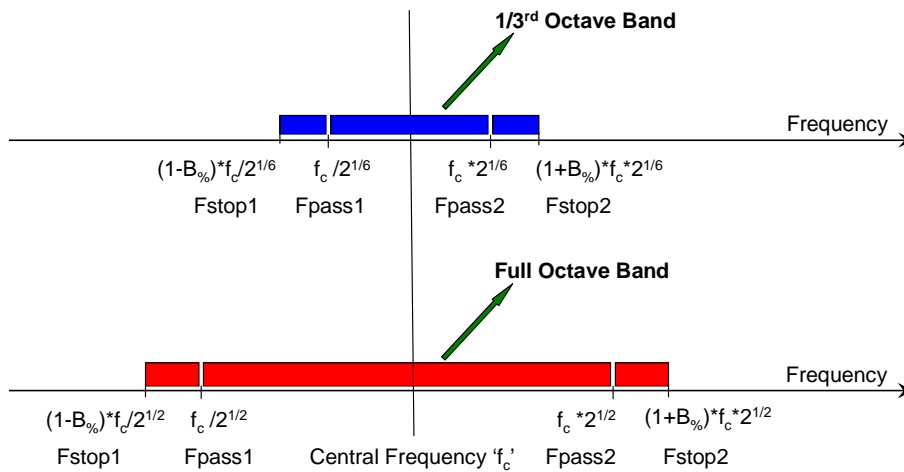
Figure 19: Filter design parameters for 1/3rd and full octave bands (Representation only – not to scale)

"Tight filter bands" correspond to higher stop band attenuations and/or a narrow passband, i.e. higher magnitude of Astop1 and Astop2 and/or smaller $B_\%$. As the F_{Octave} for $1/3^{rd}$ octave is smaller than the F_{Octave} of full octave, the $1/3^{rd}$ octave bands will be tighter than the full octave filter bands.

To test the effect of filters on decay rate, i.e. loss factor, a simple computational experiment was conducted. In this experiment, the impulse response of a 1DOF system with a natural frequency of 100 Hz and loss factor of 0.01 is "contaminated" with impulse responses of 10Hz, 50Hz and 170Hz. Filters of different attenuations and bandwidth were created and implemented to filter out the unwanted out–of–band frequencies. The following four filters with the same central frequency of 100 Hz and full octave bandwidth were created to study the effect of filters:

1. 10th Order Filter – 6dB stopband attenuation

2. 100th Order Filter – 6dB stopband attenuation

3. 166th Order Filter – 20dB stopband attenuation

4. 315th Order Filter – 40dB stopband attenuation.

All filters cause a "group delay" in the filtered response; however the group delay does not alter the decay characteristics. For a Kaiser Window (FIR filter), the group delay is half of the filter order. Time delay is equal to the product of the group delay and time increment 'dt'. Hence, for a filter with higher order, the associated group delay and time delay will be also higher. In Figure 20, the filtered response and the time delay associated with different filters are plotted. It can be observed that for this 1DOF computational experiment, the response filtered with 315th order filter has the highest group delay and the response with 10th order filter has the least group delay. This observation is consistent with theory which states that higher filter order causes higher group delay.

The group delay introduced in the filtered response, to be analyzed using RDT, will only delay the detection of the first trigger. Therefore, there is no impact of group delay on the process of loss factor estimation using RDT. When time domain filters are to be used for IRDM, the length of measured response should be long enough to record the complete decay of the impulse response. Failing to follow this may result in inaccurate damping estimation.
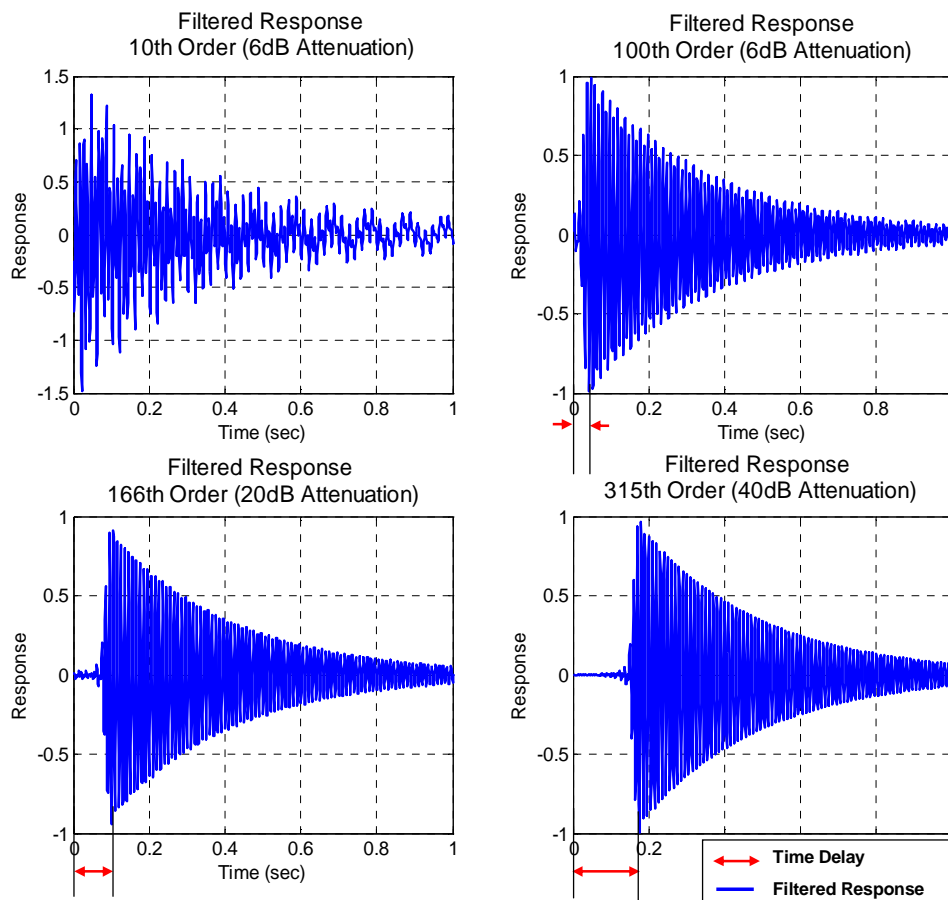
Figure 20: Comparison of group delay for different filter order

2.4.2.4 Quality of the Filtered Response

Filters are evaluated based on their ability to filter out the unwanted out–of–band frequencies. The Fourier transformation of a signal is often used to represent the signal in the frequency domain. In MATLAB, Fast Fourier Transformation (FFT) algorithms are

implemented to analyze the filtered signal in the frequency domain. The peaks in the magnitude of transformed signal (in the frequency domain) correspond to the dominating modes (or frequencies) and for a 1DOF system only one such peak is expected.

For the previously–discussed 1DOF system, in section 2.4.2.3, the filtered responses in the frequency domain are plotted in Figure 21. The 315th order filter has removed all the out–of–band frequencies as the only peak noted corresponds to the desired 100 Hz mode. The 10th order filter failed to eliminate all "out–of–band" frequencies: 10 Hz, 50 Hz and 170 Hz, and this is concluded from the prominent peaks, corresponding to the out–of–band frequencies, in the FFT of the filtered response. A further, detailed simulated 1DOF system study on filters is included in section 3.1.4.
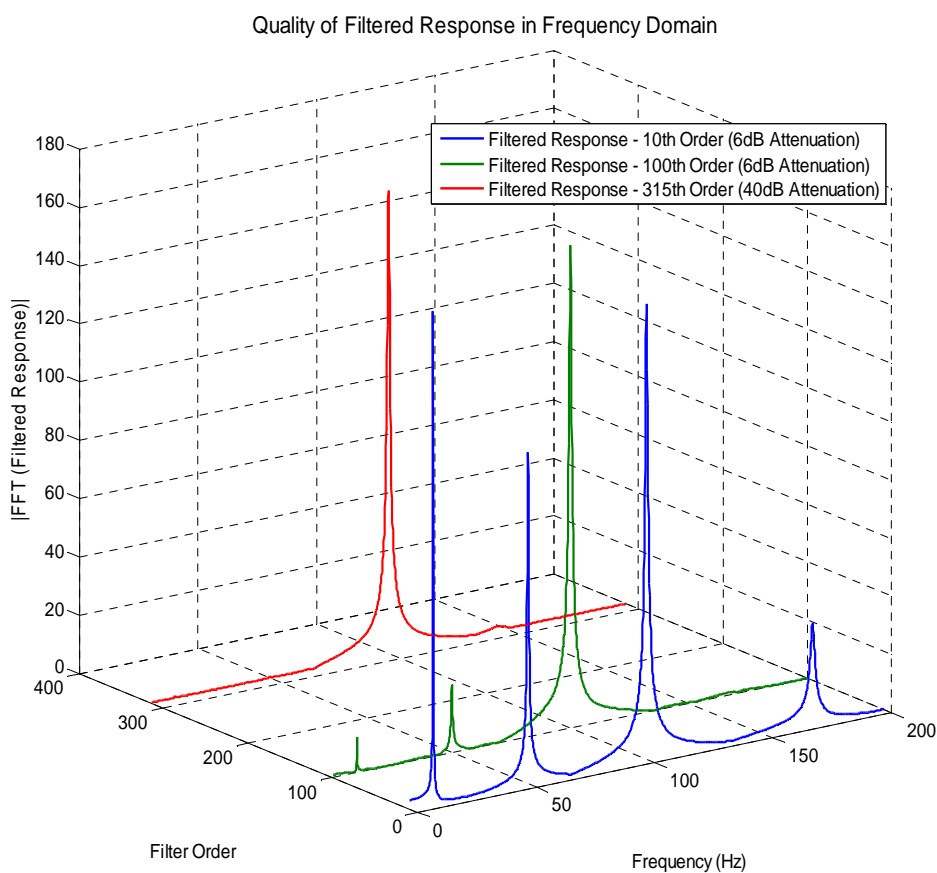
Figure 21: Comparison of quality of filtered response

For the MDOF computational plate model, created in MSC/ NASTRAN, there are two possible filtering options: time domain and frequency domain. Both of these options are discussed in later parts of this chapter.

2.4.2.5 Effect of Filters on dB Decay Curve

As discussed earlier in previous sections, 2.4.2.3 and 2.4.2.4, lower order filters fail to clean the contaminated signal. Therefore, the dB decay of the response filtered using the lower order filters will correspond to the decay of long lasting, lower frequency modes, in this case the 10 Hz mode. The dB decay curves of the filtered responses are plotted in Figure 22. The very initial decay rates of filtered impulse response from all the filters are consistent with the decay rate of the targeted 100 Hz mode with a loss factor of 0.01. From the dB decay curve of response filtered using the 10th order filter (red curve) it is very difficult to identify the region in which the decay rate can be accurately estimated. On the other hand, the filtered response from the 315th order filter (magenta curve) has an identifiable region – 0 second to 2.5 seconds – in which the decay rate is almost equal to the decay rate of the 100 Hz mode with a loss factor of 0.01.

Clearly, the output from the lower order filters that are contaminated with the out–of–band frequencies makes it difficult to accurately measure the targeted decay rate of simulated 1DOF system. The secondary decay curve of filtered response is of the lower frequency mode, which has a lower decay rate than the decay rate of the 100 Hz mode. Therefore, the loss factor computed from the secondary dB decay curve will underestimate the decay rate of the desired 100 Hz mode.
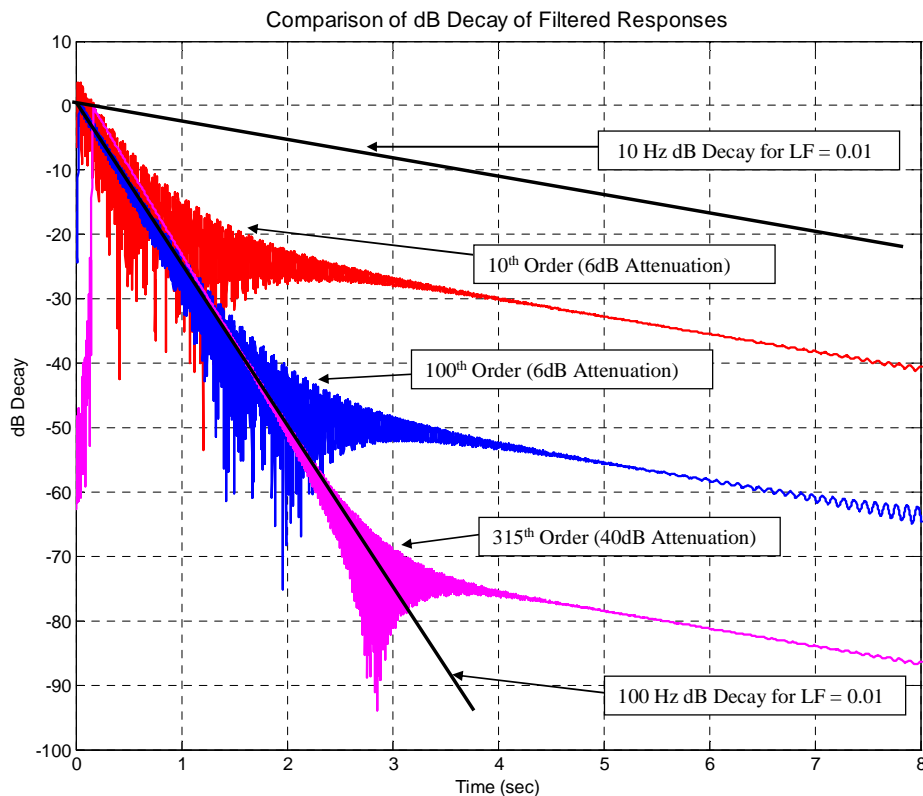
Figure 22: Comparison of dB decay for response filtered from different order filters

2.4.2.6 Effect of Triggering

The purpose of triggering in RDT is to extract response records with the same initial conditions. When a large number of these triggered responses are averaged, the random elements of the random response, which are presumed to have zero–mean, will "disappear", leaving only the decaying response with the trigger setting as the initial condition. The initial conditions used here are a reasonably high value of positive displacement with essentially zero slope – characteristic of a "twang" excitation (hold and release, like a guitar string). Such initial conditions are recommended by Kijewski and Kareem.[23,24] Figure 23, represents the schematic of a triggering scheme based on the definition of "lower trigger level", which the response must exceed to be a candidate trigger point. If the response

"peaks" above this level and decreases below this level before exceeding an "upper trigger level", the peak constitutes a triggering event.
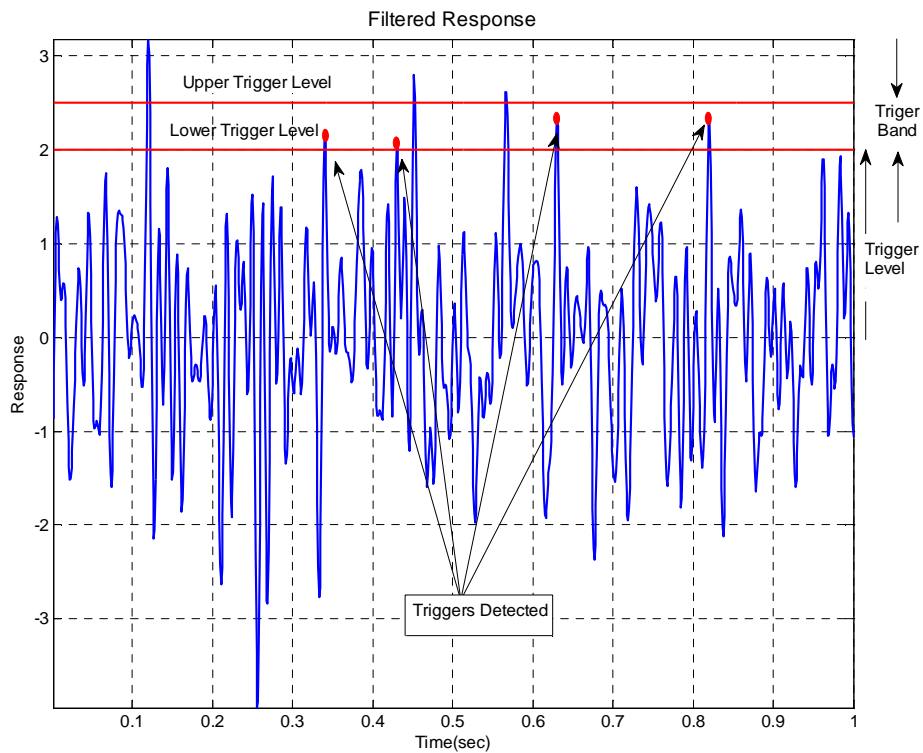
Figure 23: Schematic of triggering

A study has been performed in which the lower triggering level and the width of the triggering band are varied. The number of triggered samples available for averaging, i.e. to compute the randomdec signature, will depend on the definition of the lower and upper trigger levels. For a finite length time history of measured response, the following are found to be true:

1. Increasing the lower triggering level beyond a certain value results in detection of a fewer number of triggers

2. Increasing the width of the triggering band (or spread) results in detection of more triggers.

In Chapter 3, the outcome of the study of the effect of changing the trigger level and bands is discussed in detail. Based on the observations for a simulated 1DOF study, reported in Section 3.1.2, a lower trigger level of twice the standard deviation of random response ($2 \cdot \sigma$) and a trigger band of width equal to one–fifth of the lower trigger level is recommended for the MDOF computational and experimental analysis.

2.4.3 Computational Plate – NASTRAN Model

The computational model is a thin plate, used in a previous studies [7,26], with in–plane dimensions a by b, an aspect ratio, a/b, of 1.5, a thickness ratio, b/t, of 385 and a specific stiffness, E/ρ, of 2.59 * 10^7 Nm/kg. The damping loss factor, η, is assigned values of 0.01 and 0.1. The plate is considered to be free of external supports and is loaded by a concentrated mechanical force.

Although an analytical model is undoubtedly available, a computational model has been developed using MSC/ NASTRAN. QUAD4 elements (of dimensions, A by B) with a quadrilateral element fineness of approximately 80 (b/B or a/A) have been used to create a regular, rectangular mesh. As such, the spatial Nyquist frequency–at which modal half–wavelengths approach the element width (A or B)–has been shown to be approximately 10 kHz. The mechanical loads are applied in the center of one quadrant of the plate to a single node, modeling physical loading via the bottom surface of a small force gage through which the load would be applied in a physical experiment. Responses to excitation at the single forcing point were computed at approximately 17 points as indicated in Figure 24, wherein each square region represents 100 elements. There are a total of 9,600 square–shaped QUAD4 elements in the computational plate model. In testing completed for the physical

model, the external force was exerted by an electrodynamic shaker via a thin stinger ($t_{plate}/D_{stinger} = 3$), with a force gage attached to the plate.
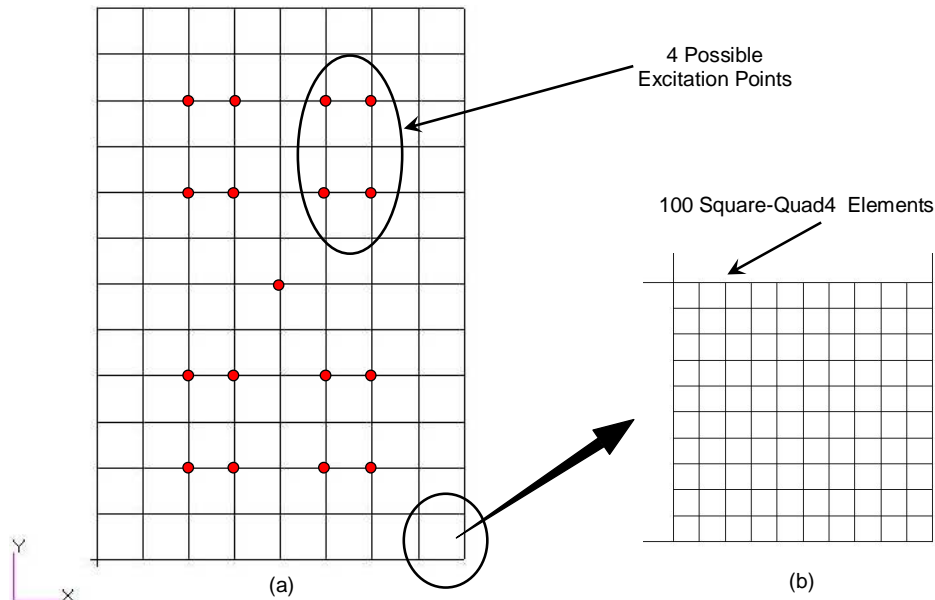
Figure 24: Computational plate model – MSC/ NASTRAN

For the computational studies, the best frequency resolution for a frequency response function (FRF) attempted is 1 Hz over a 10 kHz frequency range. These FRFs were inverted using the inverse Fourier transform, resulting in the analogous impulse response functions. Based on the frequency resolution, $\Delta f = 1$ Hz, and the number of 'lines' in the FRF, $N_{lines} = 10,000$, the length of the IRF is $1/ \Delta f = 1$ second. Since there were 'N_{lines}' lines in the FRF, the number of response points in the impulse response function is $2.N_{lines} = 20,000$, which corresponds to the temporal resolution of $1/20,000 = 50$ microseconds. As such, the temporal resolution in the computational studies allows resolution up to 10 kHz, but only useful responses up to about 5 kHz.

2.4.3.1 Filtering

As discussed earlier, filtering is required to obtain narrow–band time histories, ideally to result in only one mode per frequency band. From the MSC/ NASTRAN computational plate model, only the FRFs were available. However, RDT requires time–domain responses. Therefore, to get a random response, the impulse response is first computed by the inverse Fourier transformation of the FRFs. This impulse response is then convolved with a simulated time domain random force to generate the time history of a random response. Therefore, filtering can either be applied to the resulting response, that is,

Time Domain Filtering (or TDF), or to the FRF <u>before</u> convolution, that is, Frequency Domain Filtering (or FDF).

In the conventional, i.e. experimental approach, RDT loss factor estimation process, only the time domain filtering is possible because only output measurements from accelerometers or strain gages are recorded. In Chapter 3 and Appendix A, the loss factor estimates from FDF and TDF approaches are compared. This comparative study of TDF and FDF can be used as a reference when dealing with filtering options to make a better choice.

2.4.3.1.1 Time Domain Filtering (TDF)

The TDF approach relies on the filters created using the MATLAB's FDA Tool for filtering the long time histories of the response in finite width frequency bands. The steps involved in time–domain filtering, also graphically represented in Figure 25, for a MDOF computational plate model are:

1. FRFs from a desired output location on the plate are extracted from the MSC/ NASTRAN output files. For the simulated plates, these FRFs are for a frequency scale of 0 Hz to 10 kHz.

2. The impulse response for the system, which comprises of all possible modes below the Nyquist frequency, 10 kHz, is calculated by inverting the complete FRF in the Fourier sense.

3. The time domain response is generated by convolving the impulse response with a long time domain random force, thereby allowing for many triggered time records to be extracted later.

4. The random response is then filtered by using filters created in MATLAB to produce the narrow–band time history.

5. The randomdec signature is calculated from each narrow–band response. The loss factor can be estimated from the decay rate of randomdec signature, using equation 2.5.

2.4.3.1.2 Frequency Domain Filtering (FDF)

As the process's name suggests, the narrow–band response is calculated using filters in the frequency domain. The steps involved, as represented in Figure 26, with FDF are:

1. The complete FRF is evaluated in a similar way as discussed in TDF approach.

2. The FRFs are filtered in frequency domain by "zero–padding" the FRF elements corresponding to out–of–band frequencies. By computing the inverse, in Fourier sense, of this "chunked" FRF the narrow–band impulse response is estimated.

3. The convolution of this band–limited impulse response with a long random force yields a narrow–band response.

4. The process to calculate the loss factor from the free decay is the same as discussed in the TDF section.
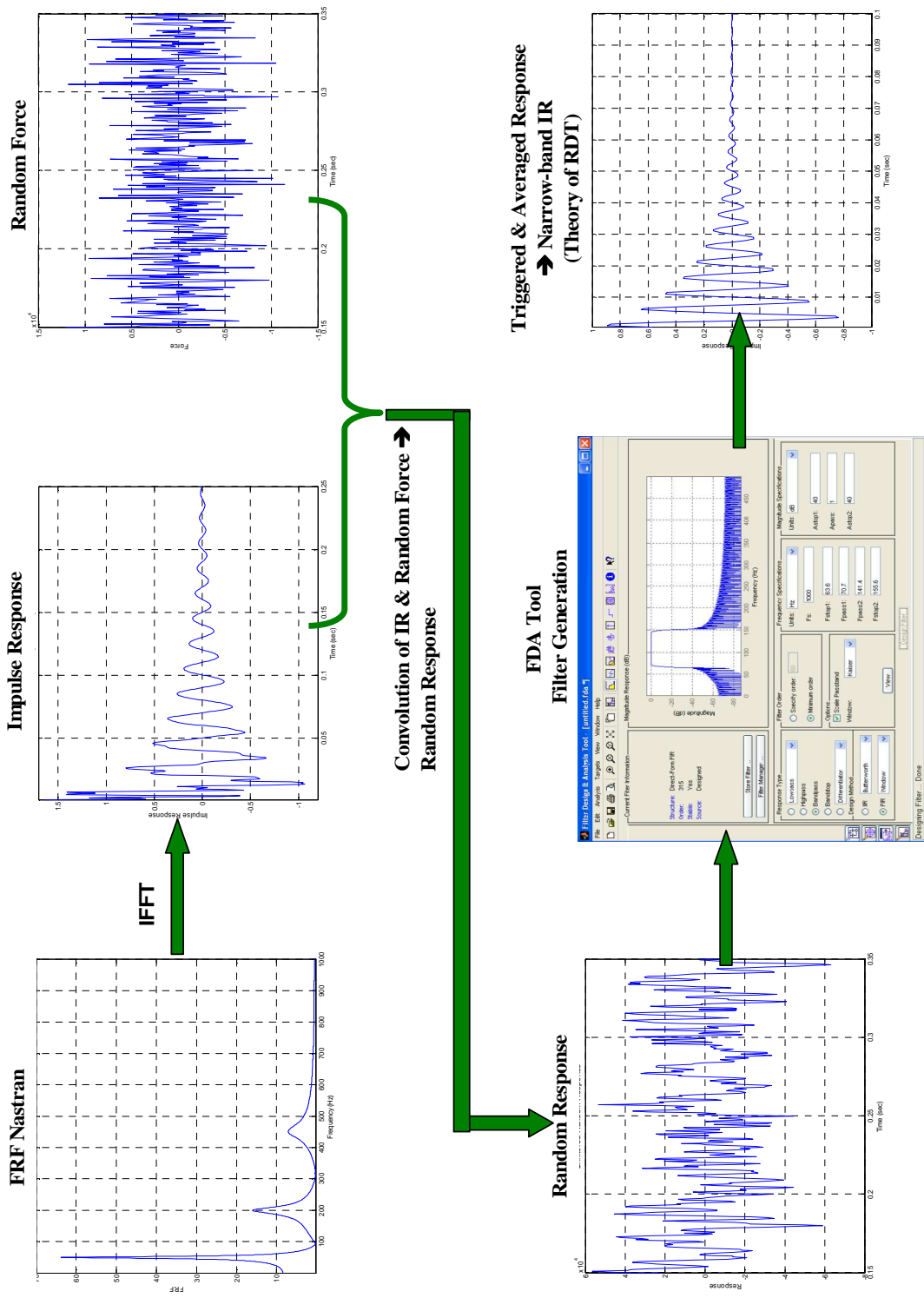
Figure 25: Schematic of Time Domain Filtering
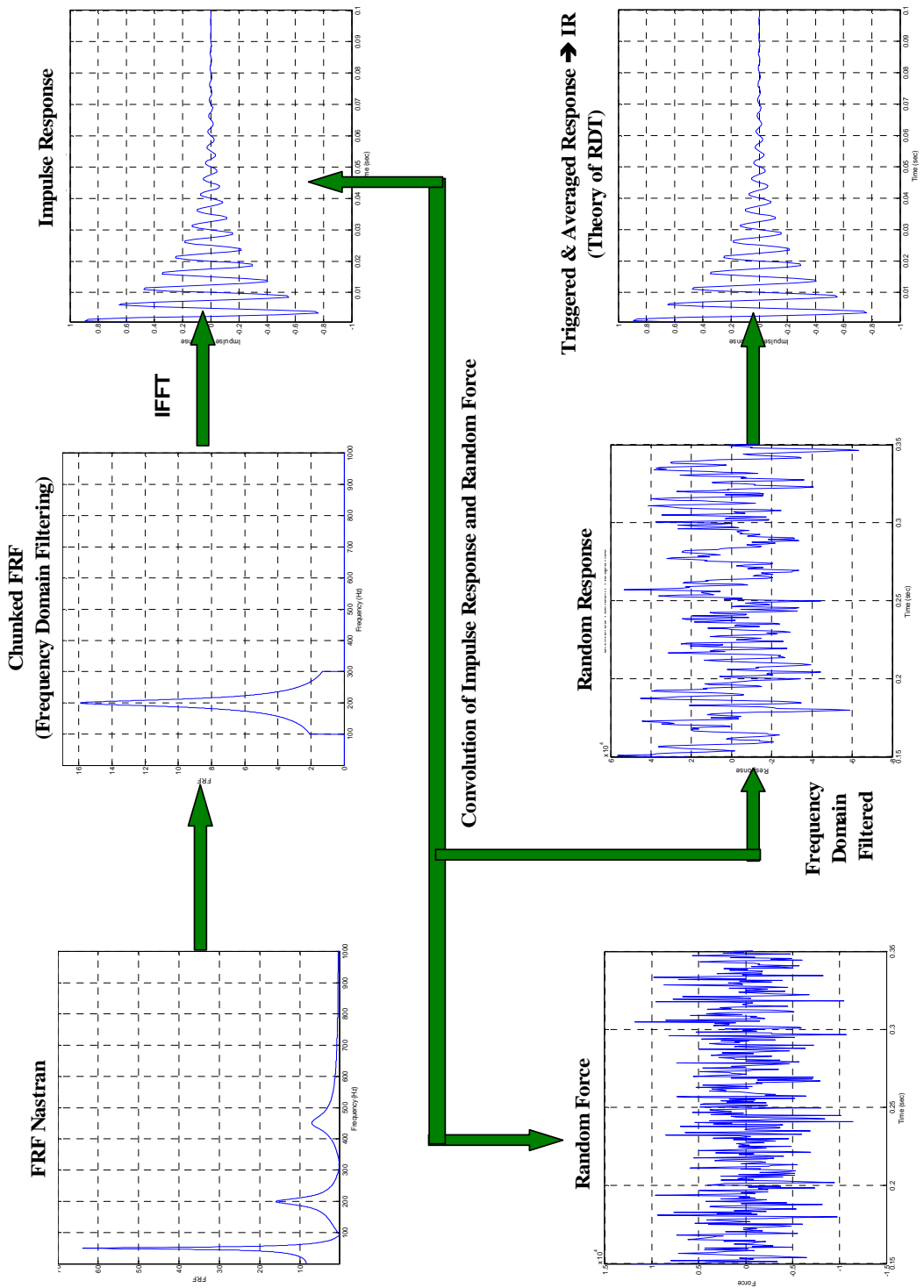
Figure 26: Schematic of Frequency Domain Filtering

2.4.3.2   Averaging Schemes – Extracting the Free Decay

For the case of a single degree of freedom system excited with white noise, (a stationary, Gaussian process), the autocorrelation function is equal to the free decay.[5,6] The fact that the decay of the autocorrelation function of random response is equivalent to the free decay of response offers an option to select one of two possible averaging schemes, discussed in following section, to extract the free decay rate. The direct method is to average the triggered responses; the alternate way is to average the autocorrelation functions of triggered response samples.

2.4.3.2.1   Averaging the Triggered Responses – Direct Averaging

When a system is excited by Gaussian, zero–mean, white noise, the response is also zero–mean random response. If a large numbers of samples with the same initial conditions from a long record are averaged, the random component of the response "averages out" and only the transient decay survives. In Figure 27, the first three subplots represent the three triggered samples, with similar initial conditions, from a long time history of measured response. The last subplot in this figure is the average of 1000 such triggered samples which is basically the randomdec signature.

2.4.3.2.2   Averaging the Autocorrelation Functions

The randomdec signature is proportional to the autocorrelation function which when normalized to initial conditions, is equal to the free decay of the system. In this averaging scheme, the positive autocorrelation functions from the triggered responses are averaged. In Figure 28, the first three subplots correspond to the positive part of the autocorrelation functions of triggered samples. The final subplot is the average of 1000 such autocorrelation

functions. The average of autocorrelation functions and triggered responses should be equivalent to the normalized free decay. These relations are discussed in the theoretical description and literature review, presented earlier in this chapter.
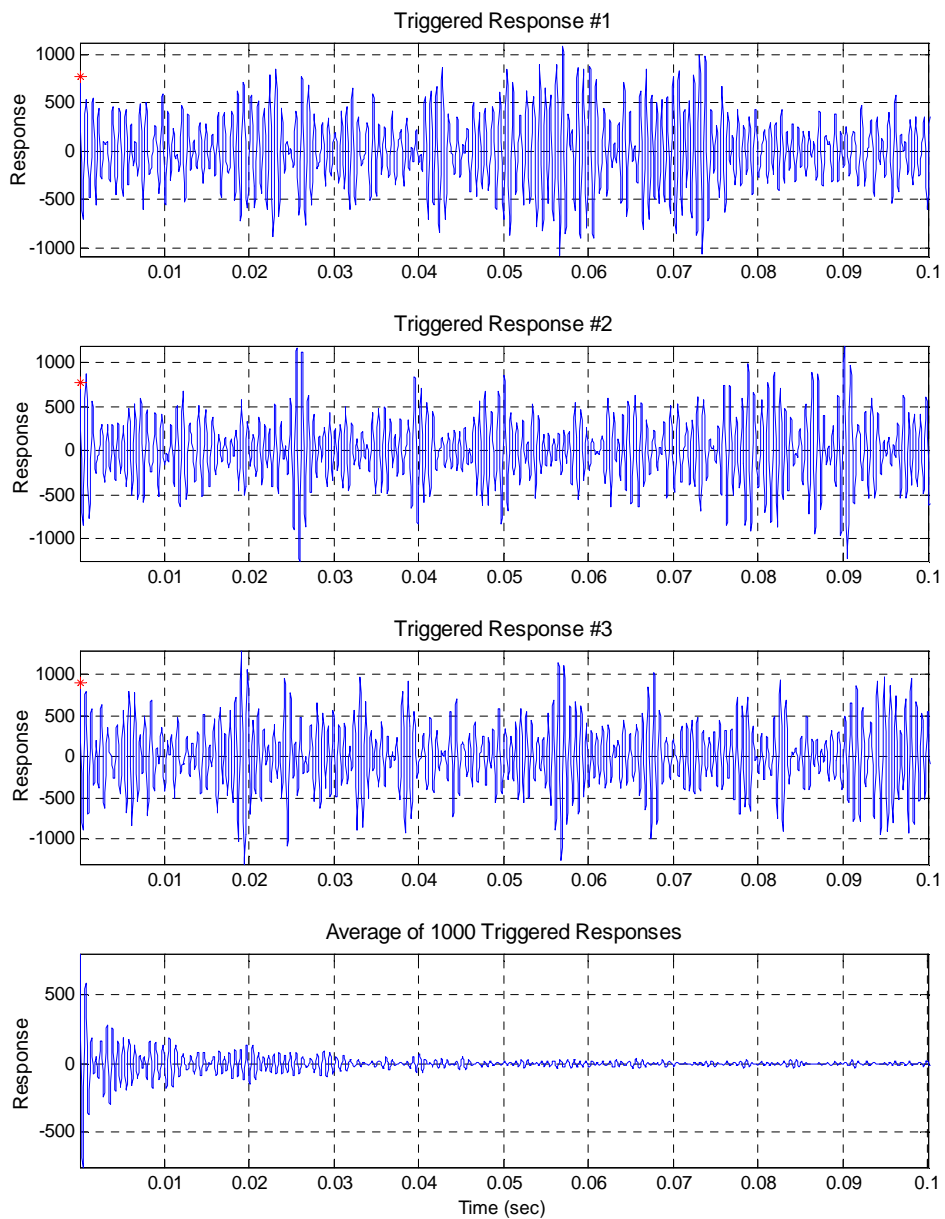
Figure 27: Averaging the triggered responses nullify the random component from the response leaving only the free decay
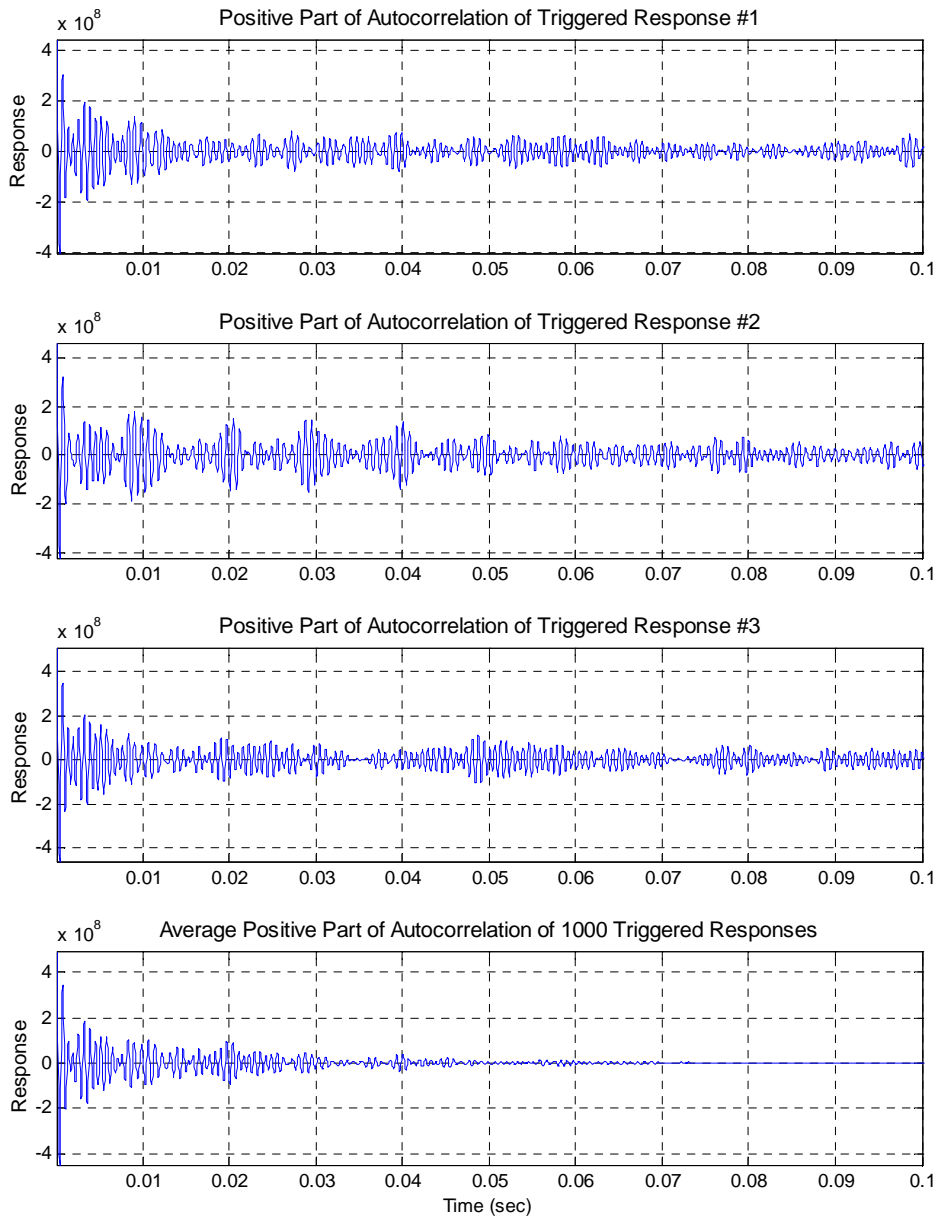
Figure 28: Averaging the autocorrelation function of triggered responses yields the free decay

In Figure 29, the normalized free decays of response from the above–mentioned averaging schemes, are plotted. The dB decays, for both averaging approaches, are plotted in Figure 30. The decay rate and the free decay obtained from both techniques seem to be in agreement for "early" times (< 0.05 sec). The autocorrelation averaging, however, better shows the expected decay rate for 'long' times (> 0.05 sec).
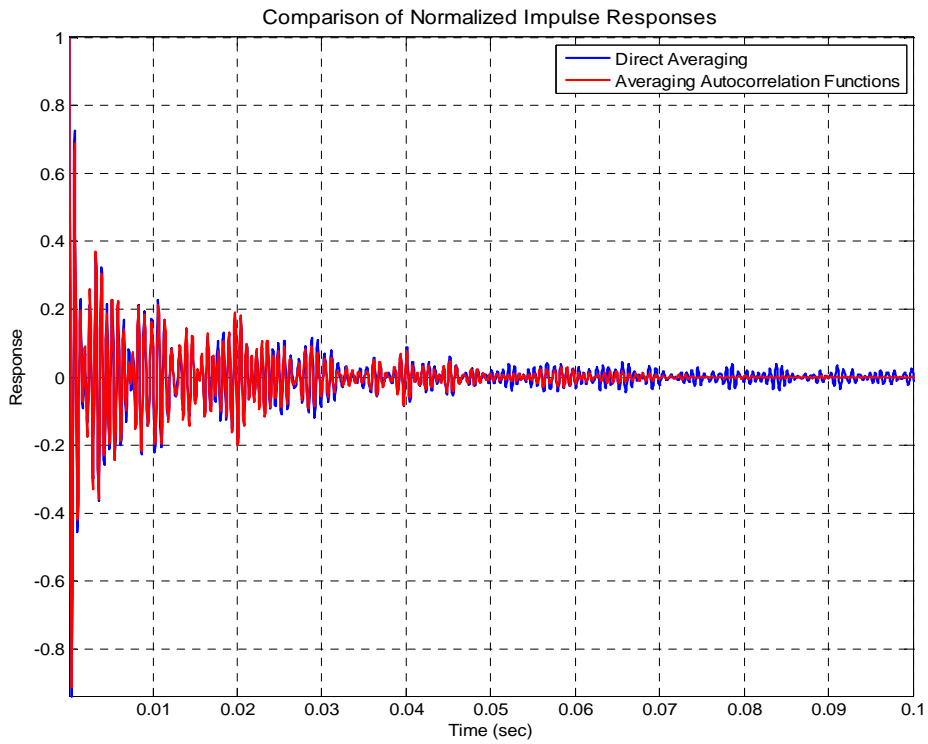
Figure 29: Comparison of free decays for different averaging schemes
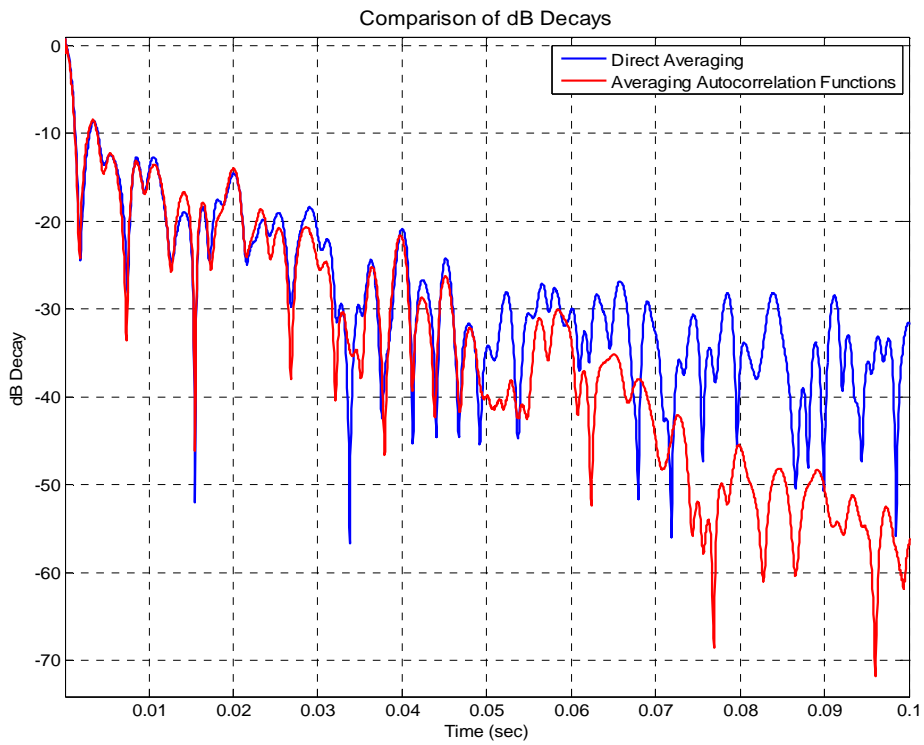
Figure 30: Comparison of dB decays for different free decay averaging schemes

2.4.3.3   Averaging Schemes – Estimating the dB Decay

There are 16 available output locations for the computational plate model. From each output location, one randomdec signature can be processed. Loss factors can be calculated from one of three potential approaches to estimate the final dB decay. The difference in these possible methodologies is the step in the process of loss factor estimation at which the averaging is done, i.e., before or after taking the $\log_{10}$ or the Hilbert transformation.   The possible averaging schemes are:

1. LF1 approach – The free decays from all the output locations are averaged. This scheme is graphically presented in Figure 31

2. LF2 approach – The exponential decays or envelopes of free decays from all the output locations are averaged. This scheme is graphically presented in Figure 32

3. LF3 approach – The dB decay curves from all the output locations are averaged. This scheme is graphically presented in Figure 33. This approach is consistent to the averaging scheme used in conventional IRDM loss factor estimation when response is measured from multiple locations on a panel.[7]

Once the dB decay curves from which the loss factor is to be estimated are calculated, an automated slope–fitting algorithm, discussed in the following section, is used to measure the decay rate. For the computational plate model, a comparative study of loss factor estimates, averaged over the entire frequency range, is presented in Appendix A and is summarized in Section 3.2. LF2 and LF3 are most promising approaches, while, and LF1 is the least promising.

The potential benefit of the LF3 scheme over LF2 schemes is that in the former approach, the dB decay curves from all output locations are available and if one of the decay rates has poor characteristics it is relatively simple to reject that particular decay curve. In

experimental analysis, the LF3 approach was chosen over LF2 because the averaging scheme associated with LF3 is easier to implement in practice. The LF3 approach is analogous to averaging of the estimated loss factors from multiple response locations.
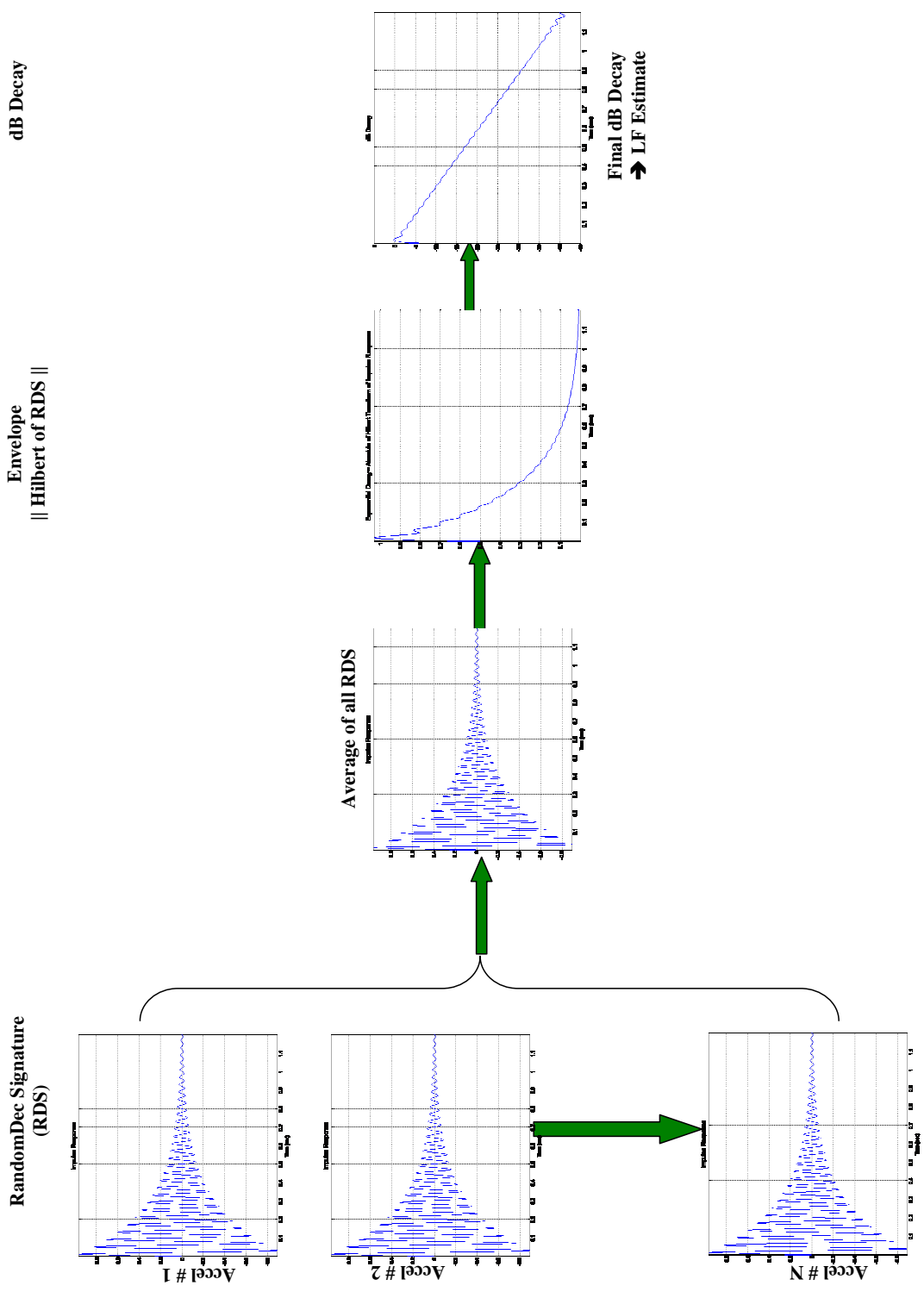
Figure 31: Schematic of LF1 averaging approach
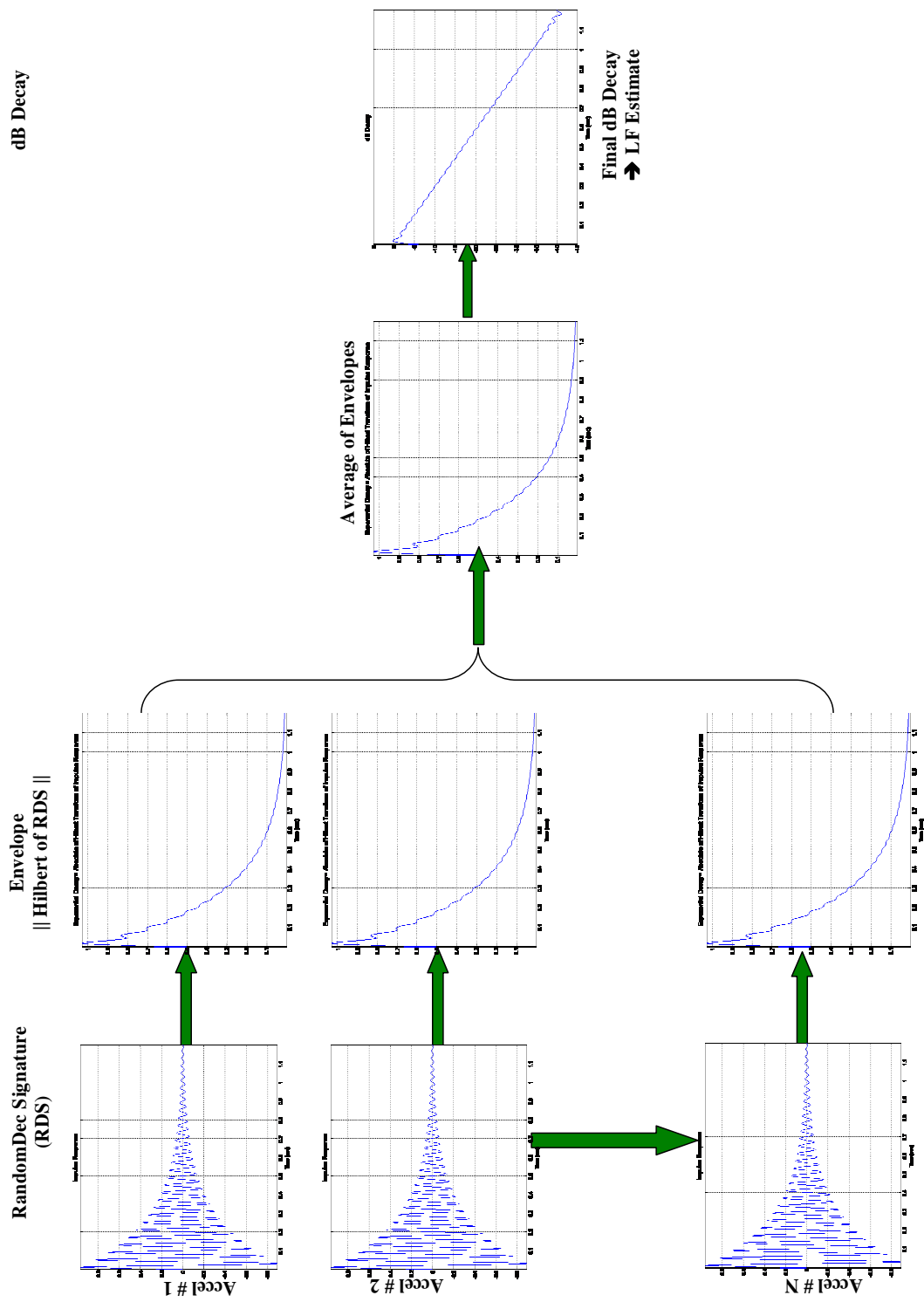
Figure 32: Schematic of LF2 averaging approach

58
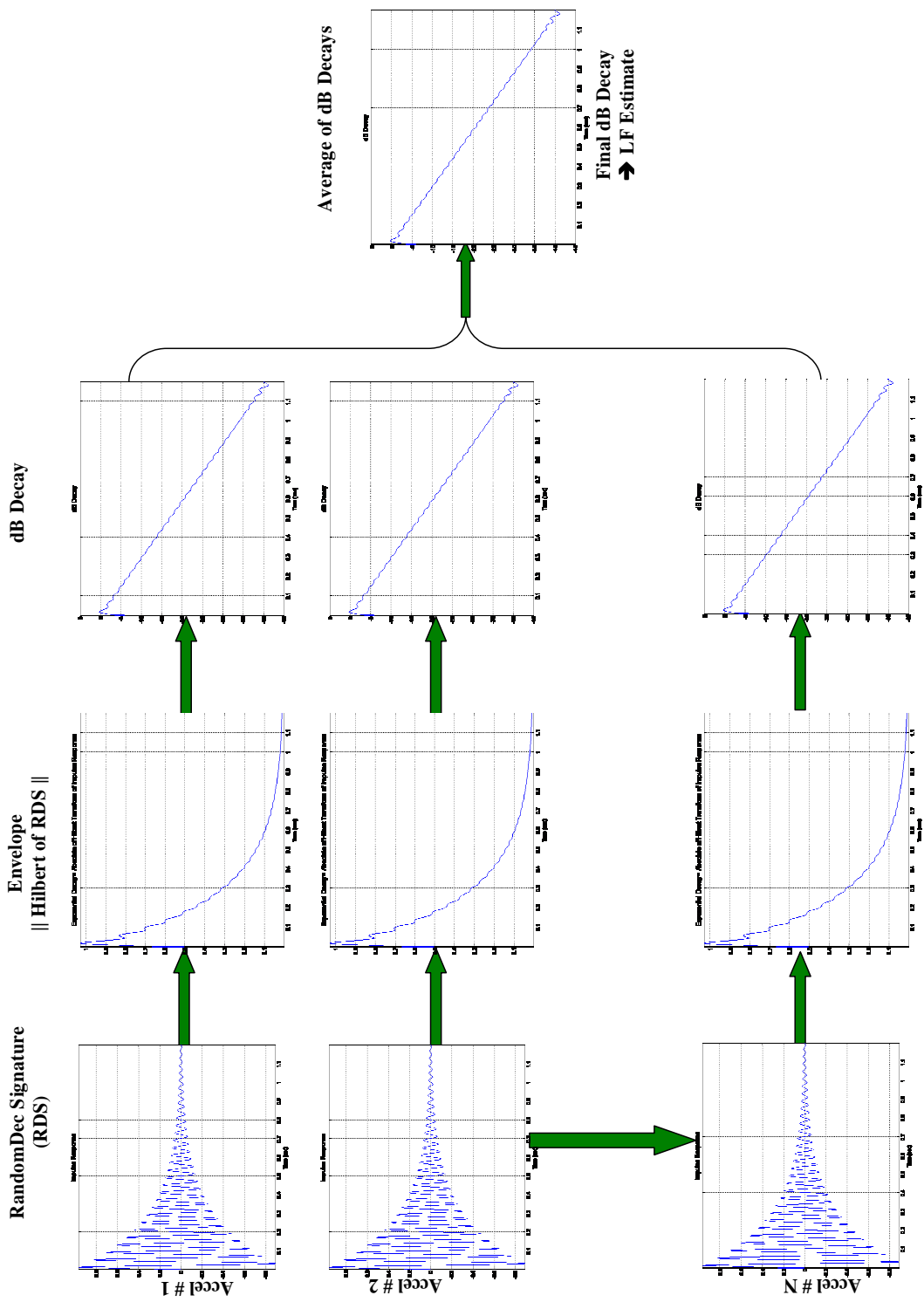
Figure 33: Schematic of LF3 averaging approach

2.4.3.4 Slope Fitting – Estimating the Decay Rate

Loss factors are estimated from the decay rate in dB/sec. The dB decay curves from experimental or computation analyses are rarely linear. Therefore, to estimate decay rate, a least squares linear curve fit can be implemented. The curve fitting can be user defined (manual) or it can be automated. For multiple input–output locations and multiple frequency bands, automated slope fitting is strongly recommended.

Manual slope–fitting may turn out to be a better option for a small number of dB decay curves, but when a large number of curves are to be fitted, human error in fitting a curve is inevitable and the results are difficult to reproduce. On the other hand, automated slope fitting is faster and efficient, but must be studied extensively before general use. It is suggested that the slope fitting should be automated with some supervision or control over the process. In the analysis presented, all the slope fits from which the loss factors are estimated are visually inspected. For decay curves on which the slope fitting does not seem to work accurately, the manual slope fitting approach is enforced.

The proposed automated slope–fitting algorithm identifies the local 'peaks' in the dB decay curves and then fits a least squares linear curve through them. For the slope fitting, the user specifies the start and end points on the dB decay curve. The slope of the least squares linear curve fit is the decay rate. If less than two peaks are detected, manual slope fitting is chosen over the automated slope fitting approach to avoid bias.

Equation 2.5, as derived in section 2.3, provides a relation between the decay rate and the loss factor:

$$\eta = \frac{DR}{27.3 \cdot f_c} \qquad \text{(eqn. 2.5)}$$

where,

DR   is the decay rate (dB/sec) = Negative of the slope of the least squares linear curve fit

f_c is equal to frequency band's central frequency (Hz).

Figure 34 presents the dB decay curves from different averaging schemes, as explained in section 2.4.3.3.
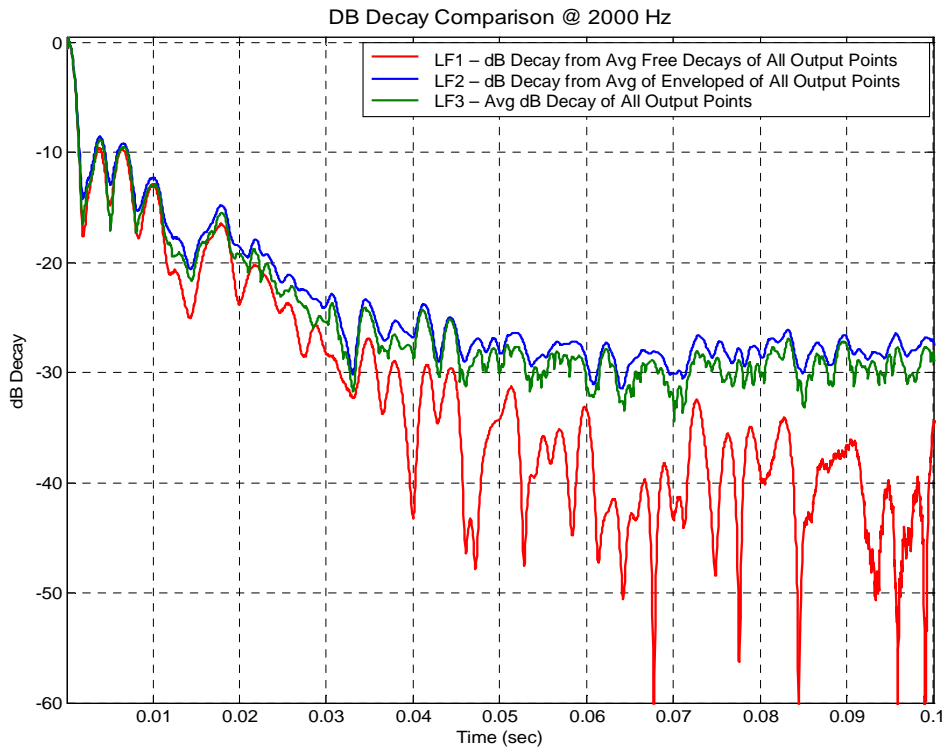
Figure 34: Comparison of dB decays for different dB decay estimation schemes

Figures 35 to 37 correspond to slope fitting on the dB decay curves from different averaging schemes. In these figures, the black markers '*' are the local peaks in the dB decay curve and the black line is the linear slope fit through the indicated local peaks. MATLAB's least squares linear curve fitting function is used for implementation of the automated slope fitting.
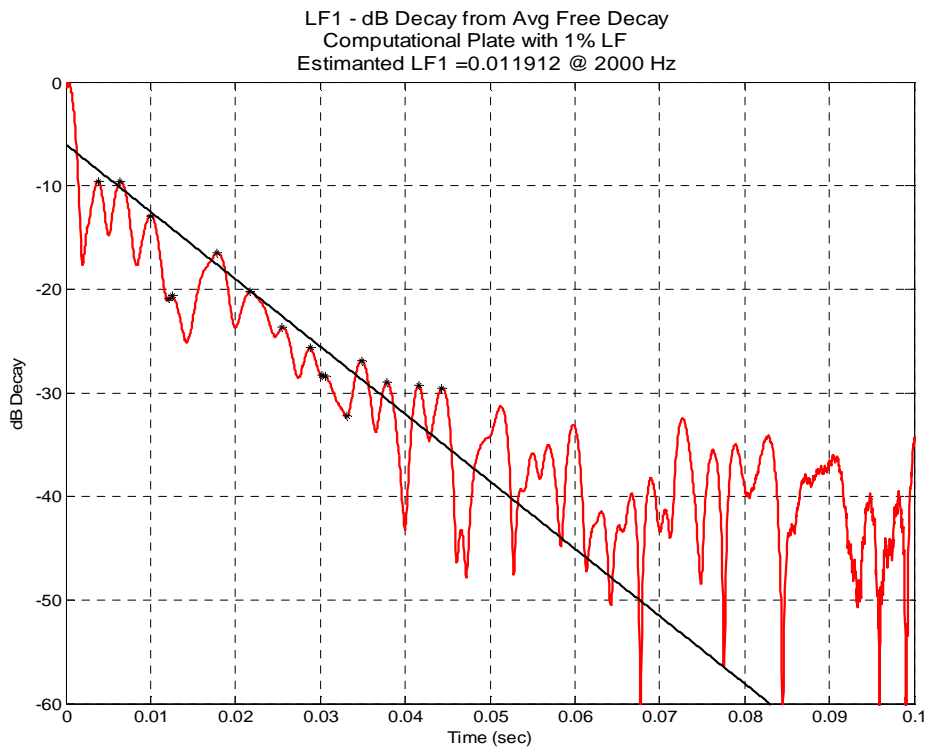
Figure 35: Slope fit and loss factor estimation from LF1 approach
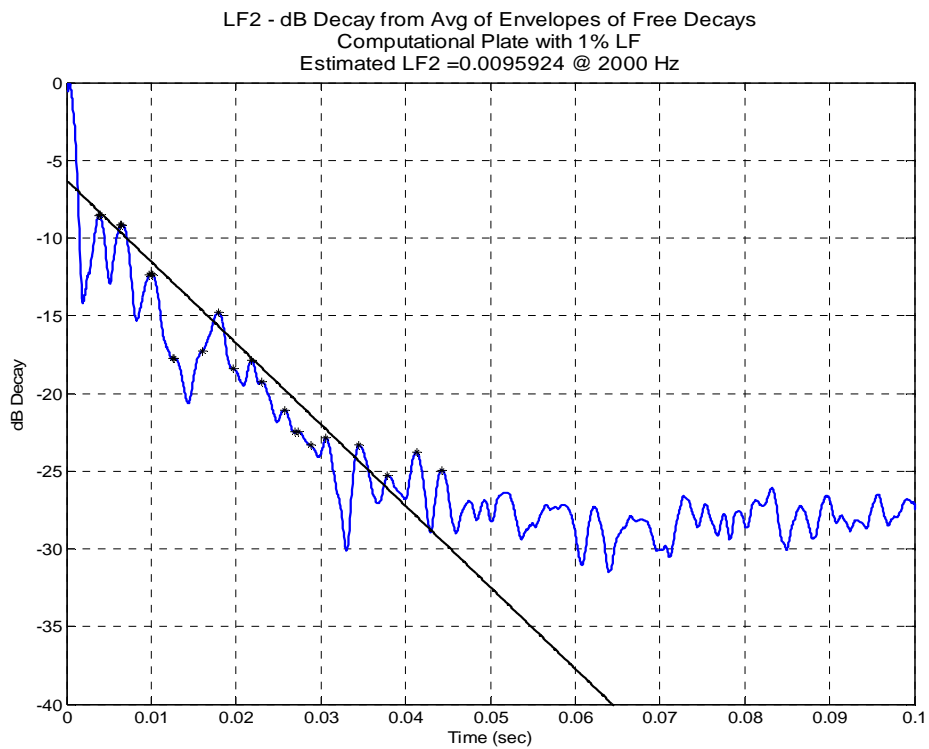
Figure 36: Slope fit and loss factor estimation from LF2 approach
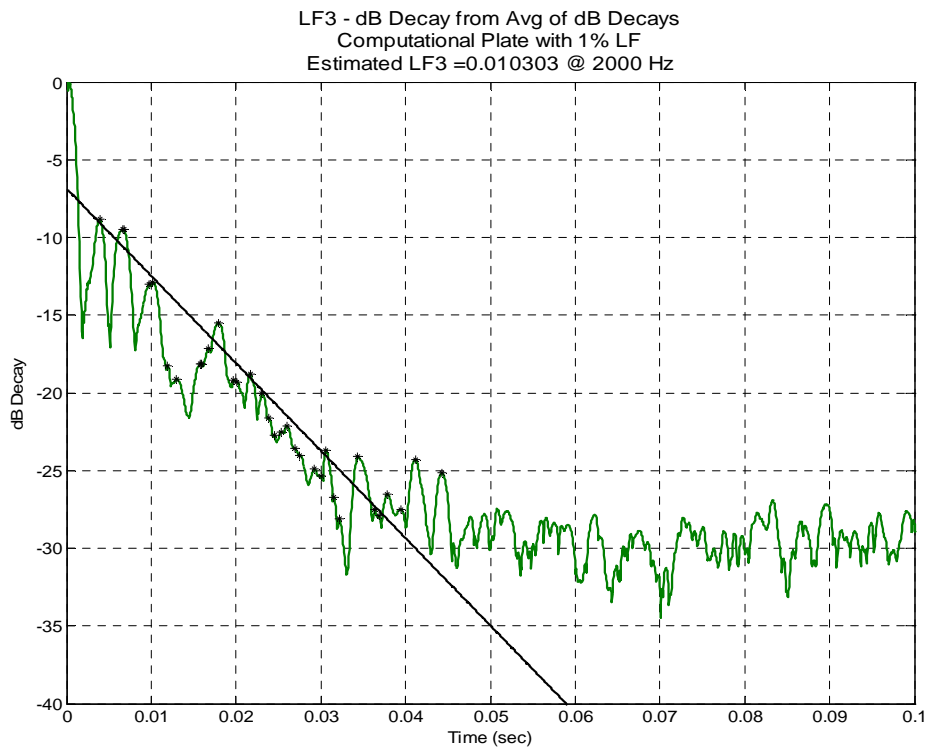
Figure 37: Slope fit and loss factor estimation from LF3 approach

2.4.4 Experimental Validation

The final objective of this research is to experimentally validate RDT for loss factor estimation. This is achieved by evaluating the loss factor of three different plates with damping treatments applied to them. The plate's dimensions and expected damping levels are provided in Table 1. Plate 1 is the most damped and Plate 3 is the least damped amongst the three plates. The estimated damping loss factor of all the three plates is in the range of 0.1 and 0.01, which is the loss factor range studied for the simulated MDOF computational plate model. Hence the conclusions drawn for the computational models have been directly adapted in the experimental validation studies.

Filters were created using the MATLAB's FDA Tool and implemented using MATLAB codes; i.e. the analysis is mainly off–line. The RDT loss factor estimates for the

plates are compared to loss factors estimates by IRDM. An in–house LabVIEW code for real–time analysis has been used to estimate the damping loss factor using IRDM. In IRDM, both hammer and shaker excitations were used to compute FRFs. Frequency domain filtering is implemented to extract the band–limited impulse response functions from the accelerometer data. The excitation and measurement location were the same in RDT– and IRDM–based loss factor analysis.

The accelerometers used, PCB Model # 352A71, were small and light weight (0.64 gms) which is negligible as compared to the mass of plate. The plates were excited at four different locations and the response was measured by eight accelerometers. In Figure 39 the excitation locations are indicated by red squares and the response measurement locations are represented by green circles. These excitation and response locations were randomly chosen so as not to excite the structure at expected node lines. The sampling frequency for the experimentation was 20 kHz, which is sufficiently high to analyze the system's response up to a band with central frequency of 5 kHz.

All plates were hung from two thin steel wires to simulate free boundary conditions. For persistent random excitation, a mechanical shaker is used. The power delivered by the shaker can be amplified from the gain setting on the power amplifier or through the virtual 'knob' on the LabVIEW front panel. A force gage is used to measure the input force, though in RDT the force information is not used at all. Through a thin, flexible stinger the mechanical excitation from the shaker is transferred to the plate thereby avoiding moment–loading. The force gage is positioned between the stinger and the point of application on the plate. Response of the plate from eight different locations is measured by accelerometers positioned on the opposite side of the test specimen from the force gage and shaker.

The measured signals are "conditioned" using a 16–channel signal conditioner. The excitation to the mechanical shaker and the measured signals (acceleration and force) are converted from analog to digital form using a NI PXI–1033 chassis. A LabVIEW code is used to record these processed signals in either '.txt' or '.lvm' format. The schematic of the experimental setup is shown in Figure 38.
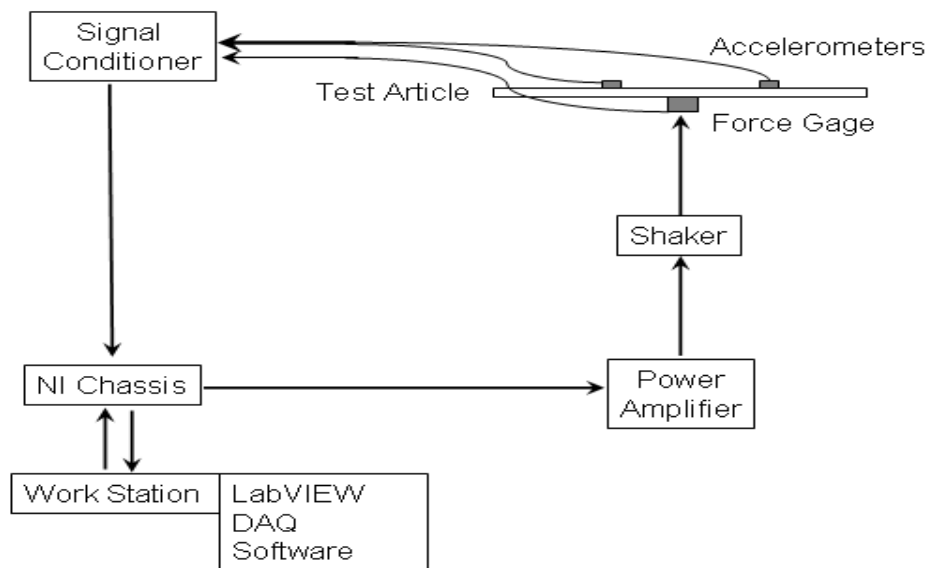
Figure 38: Schematic of the experimental setup

The sensitivities and description of the transducers and instruments used are provided below:

Hardware

1. Chassis – NI PXI 1033

2. Signal Conditioner – ICP 16 channel

3. Amplifier – LDS PA25E Power Amplifier

4. Shaker – LDS V203

5. Transducers

    1. Impulse Hammer – Modally Tuned

PCB Model # 086C03 with a sensitivity of 10 mV/lb for plastic vinyl tip

2. Accelerometers (8 used)

PCB Model # 352A71 with a sensitivity of between 9.13 mV/g to 10.16 mV/g

3. Force Gage

PCB Model # 708 A50 with a sensitivity of 49.83 mV/lbf
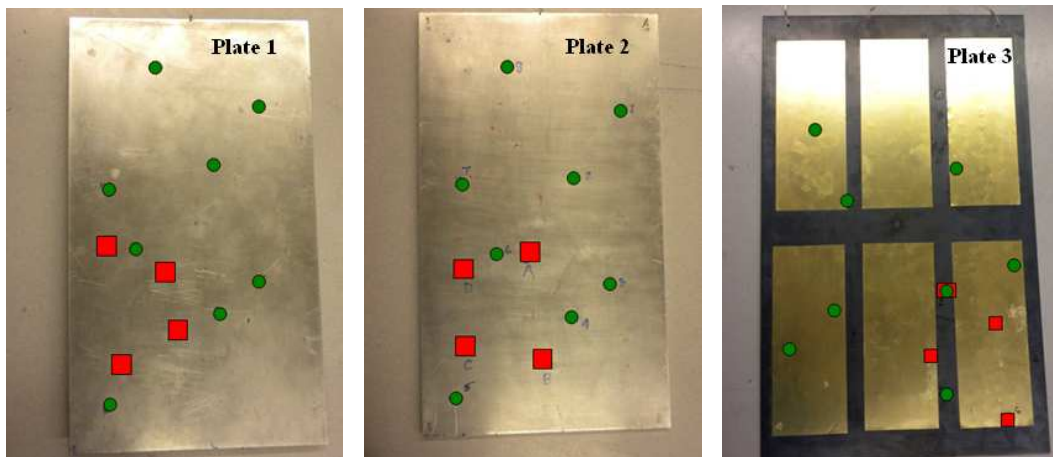
Figure 39: Plates with damping treatments for experimental analysis

	Plate 1	**Plate 2**	**Plate 3**
Base Plate	Al CLAD 2024–T3 34.9 x 20.29 x 0.16002 cm^3	5052–H34 34.7 x 20.10 x 0. 3055 cm^3	Mild Steel 39.37x63.50x0.6350 cm^3
Viscoelastic Material	3M F9469PC 0.0127 cm thick	3M F9469PC 0.0127 cm thick	3M F9469PC 0.0127 cm thick
Constraining Layer	Al CLAD 2024–T3 34.9x20.29x0.0508 cm^3	Al CLAD 2024–T3 34.7x20.1×0.0508 cm^3	Brass Sheet 25.4x10.16x0.0127 cm^3
Coverage	Full	Full	Partial
Estimated Loss Factor	≈ 0.1	≈ 0.07	≈ 0.01

Table 1: Description of plates used in experimentation

In the estimation of loss factor for Plates 1 and 2 using RDT, 500 triggered samples are averaged. For the lightly damped plate, 200 triggered samples are averaged to estimate the randomdec signature, from which the decay rate is measured.

A LabVIEW code, which incorporates all the conclusions and recommendations from this research, has been developed for real time analysis. It is recommended that in the future, all the RDT or IRDM analysis should be done in real time using this LabVIEW code.

3 Results and Discussion

This chapter is aimed at the discussion of the accuracy and efficiency of RDT for loss factor estimation. Section 3.1 and 3.2 provide the qualitative analysis of parameters affecting loss factor estimation for the 1DOF and computational plate models. The chapter concludes with a comparison of experimentally–determined loss factors from RDT and IRDM.

3.1 1DOF Simulation

3.1.1 Length of Sample

It is recognized in section 2.4.2.2 that the number of cycles provides a better representation of the length of a triggered sample needed for loss factor estimation. For a simulated 1DOF system with constant damping, the number of cycles to damp out the vibration remains nearly constant over the entire frequency range. In Figure 40, the number of cycles to damp–out, N , is plotted versus the simulated loss factor, η_{est} , on a log–log plot. The curves in Figure 40, which are almost linear on the log–log plot, can be algebraically described as:

$$N \cong \frac{2.2}{\eta_{est}}$$

(eqn. 2.6)

where,

N is length of sample in terms of cycles, for a narrow–band time history

η_{est} is the initial "guess" of the loss factor during experimentation.

From equation 2.6, it is noted that the length of triggered sample is inversely proportional to the loss factor. Therefore, a lightly damped structure will be required to have a longer randomdec signature than that of highly damped structure. Though the numbers of

cycles to damp–out over the entire frequency range are constant for a specified damping level, the randomdec signature in a lower frequency band will last longer – in terms of time to damp–out (in seconds) – than the length of randomdec signature corresponding to a higher frequency band. If, non–overlapping triggered samples are used to calculate the randomdec signature, the length of time history to be measured will depend on the loss factor and the lowest frequency under consideration for loss factor analysis.

In RDT the loss factor is calculated from the decay rate of the randomdec signature. For a better estimate of the damping characteristics, the ensemble of triggered samples should be long enough to record the complete decay of vibrations. As the decay rate of a lightly damped structure is small, it will vibrate for a longer time. As the triggered sample's length increases, computation or processing time also increases. Thus, RDT is not recommended for loss factor estimation in very lightly damped structures and/or in lower frequency bands.
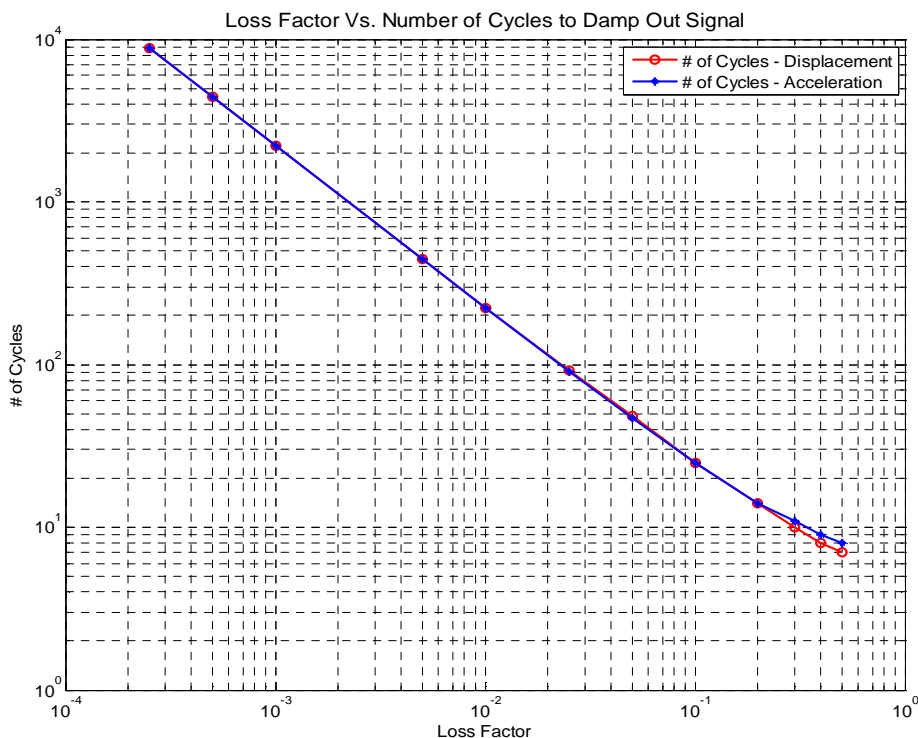
Figure 40: Loss factor Vs. Sample length, in terms of cycles

3.1.2 Trigger Level and Trigger Band

For the simulated 1DOF system, to study the trigger level and trigger band, the randomdec signature is estimated from multiple triggering conditions. The observations for this simulated experiment are included in Figures 41 and 42. Ideally, an infinite number of triggered samples are required to be averaged to completely remove the random component of the measured response.[5] But because of practical limitations on data acquisition, only a finite number of triggered samples can be recorded. If the triggering condition is very strict, such as very high trigger levels and/or very narrow trigger bands, fewer triggers are detected.

Asmussen[22] has noted that for a level–crossing triggering scheme a trigger level between σ and $2 \cdot \sigma$ is a good choice, where σ is the standard deviation of the random response. From the results herein for loss factor estimation, a lower trigger level of $2 \cdot \sigma$ and trigger band of 20% of the lower trigger level were considered as a better choice of trigger settings when compared to other possible combinations. Since the variation in estimated loss factor is under 15%, it cannot be neglected that the possible reason for this deviation is the automated slope fitting algorithm.

It is evident from Figure 41, for a simulated 1DOF system with loss factor of 0.1 and natural frequency of 1000 Hz, the 15 to 20% triggered band performed better than other possible trigger settings. However, from results presented in Figure 42, for a simulated 1DOF system with loss factor of 0.01 and natural frequency of 1000 Hz, it is clear that 20% triggered band performed successfully. In both cases the length of triggered sample is based on equation 2.6 and a total of 250 triggered samples are averaged to compute the free decay. Therefore, for computational and experimental analysis, a trigger level of $2 \cdot \sigma$ and trigger band of 20% is used. For Figures 41 and 42, the following parameters associated with trigger settings are defined as:

$$TLF = \frac{x_0}{\sigma} \qquad \text{(eqn. 3.1)}$$

where,

TLF is the trigger level factor and

x_0 is the lower trigger level

σ is the standard deviation of the measured response

$$Spread = \left(1 + \frac{T_{\%}}{100}\right) \qquad \text{(eqn. 3.2)}$$

where,

Spread   is the trigger band in terms of percentage of lower trigger level amplitude

$T_{\%}$ is the width of trigger band as percentage of x_0.
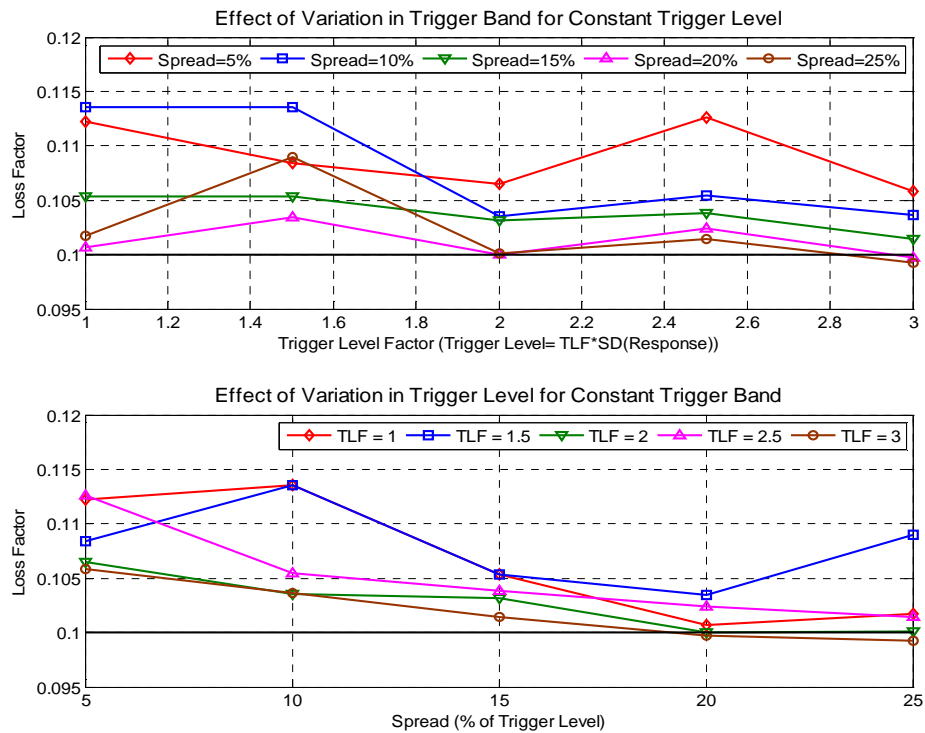
Figure 41: Study of trigger level and trigger band (for 1DOF system with simulated LF = 0.1)
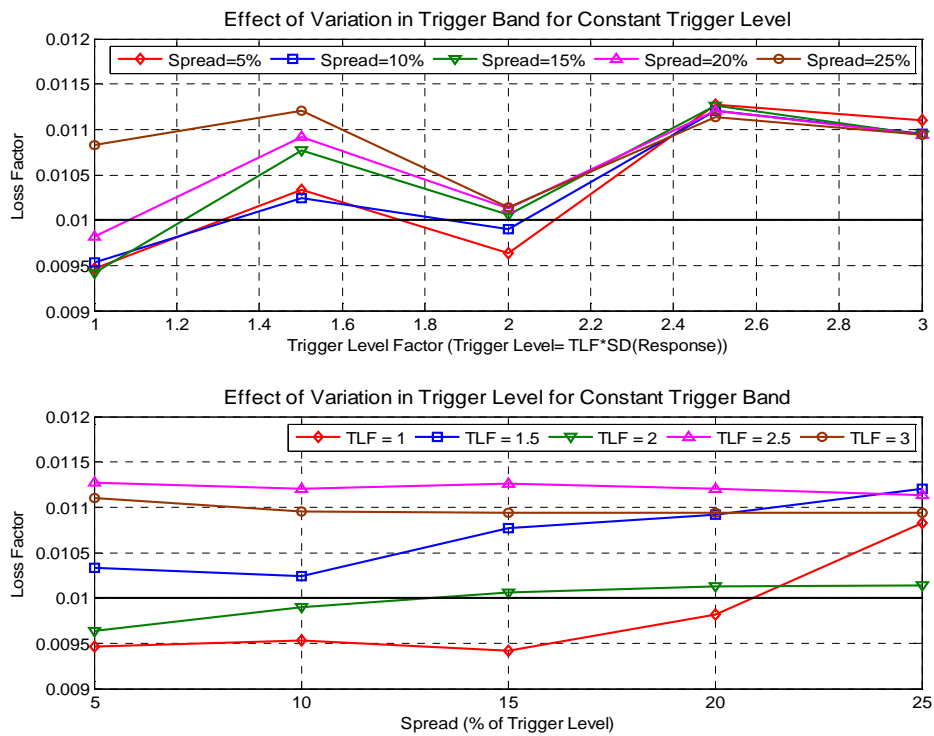
Figure 42: Study of trigger level and trigger band (for 1DOF system with simulated LF = 0.01)

3.1.3   Effect of Number of Triggers

The theory of RDT is based on extraction of decaying randomdec signature by averaging samples with the same initial conditions from a narrow–band response measurement. Although, an infinite number of triggered samples are required to retrieve the true free decay (or randomdec signature), only a finite number of triggers can be detected from the finite length of the response measurement. Therefore, a realistic approach would be to average as many triggered responses as possible. To determine the effect of the number of triggers used for loss factor estimation, the following study is performed.

3.1.3.1   Effect of Number of Triggers on dB Decay

The number of triggered samples averaged to compute the free decay controls the quality of dB decay rate, from which the loss factor is to be estimated. In Figures 43 and 44,

the dB decay curves of normalized randomdec signatures corresponding to different numbers of triggered samples averaged are plotted.

The red line in Figures 43 and 44 corresponds to the expected dB decay of a simulated 1DOF system. It is evident that if larger numbers of triggered samples are averaged, a smooth and almost linear dB decay is observed. Alternatively, it can be concluded that the initial dB decay converges to the analytical (or expected) dB decay curve with increase in number of triggered samples that have been averaged. The initial dB decay from averaging the autocorrelation function seems to have a higher slope than expected, and is relatively more linear when compared to the dB decay computed using the direct averaging approach. These observations are consistent for loss factor levels between 1% and 10%.
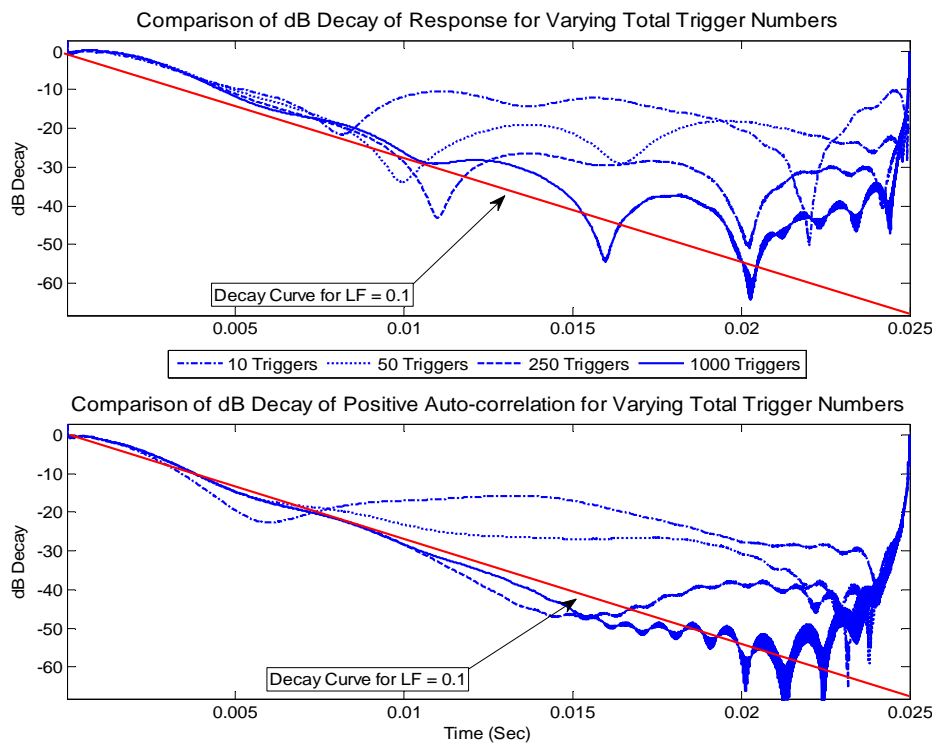
Figure 43: Study of the effect of number of triggers on dB decay (for a 1DOF system with simulated LF = 0.1 and f_n=1000 Hz).
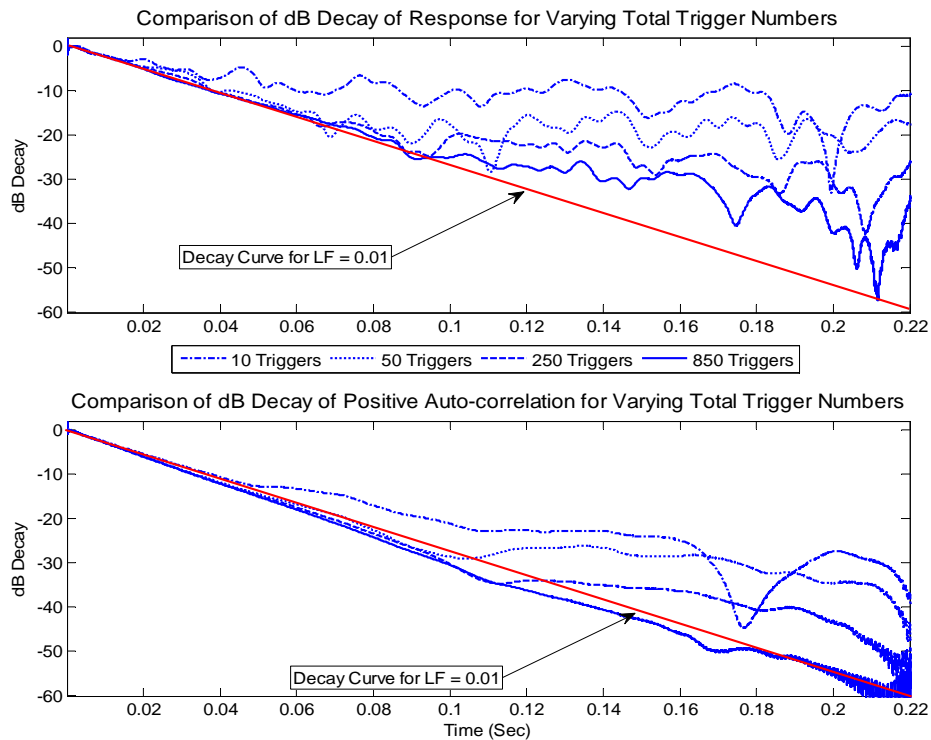
Figure 44: Study of the effect of number of triggers on dB decay (for a 1DOF system with simulated LF = 0.01 and f_n=1000 Hz)

3.1.3.2 Direct Averaging or Averaging Autocorrelation Functions

In Chapter 2, it was established that the decay of autocorrelation function of random response is equivalent to the free decay of response. Thus, the randomdec signature can be measured by averaging either the triggered responses directly or the autocorrelation functions of triggered responses.

The number of samples required to be averaged for accurate estimation of the loss factors are calculated for both averaging schemes (direct averaging and averaging autocorrelation functions). In Figures 45 and 46, for a simulated 1DOF system with natural frequency of 1000 Hz and loss factor of 0.1 and 0.01, respectively, the results of this study are reported. The following observations can be drawn from Figures 45 and 46:

1. Accuracy of loss factor estimation increases with the number of triggered samples being averaged.

2. Averaging autocorrelation functions converges quickly, i.e. by averaging fewer triggered samples, to a slightly overestimated loss factor.

   From Figure 45 it can be noted that for a simulated 1DOF system with loss factor of 0.1, the averaging of autocorrelation functions converges to a slightly overestimated loss factor in about 50 triggered samples. A similar trend is observed for the system with a loss factor of 0.01.

3. Directly averaging the triggered responses converges accurately to the simulated loss factor but with a relatively large number of triggered samples.

   It is noted, from Figures 45 and 46, for a loss factor of 0.1 and 0.01, the direct averaging approach converged to simulated the loss factor at about 500 triggered samples.

It is recommended to use the direct averaging approach if sufficiently long time histories are available to extract the required number of samples for averaging. If fewer numbers of triggers are detected, the alternative approach, i.e. averaging the autocorrelation functions, can be implemented with the knowledge that the estimates will be slightly biased (high).

The averaging autocorrelation functions approach has a potential advantage over the direct averaging approach, particularly in the analysis of lightly damped structure and/or in lower frequency bands. In the experimental and computational studies that are included in later sections of this chapter, loss factor estimates from both approaches will be analyzed.
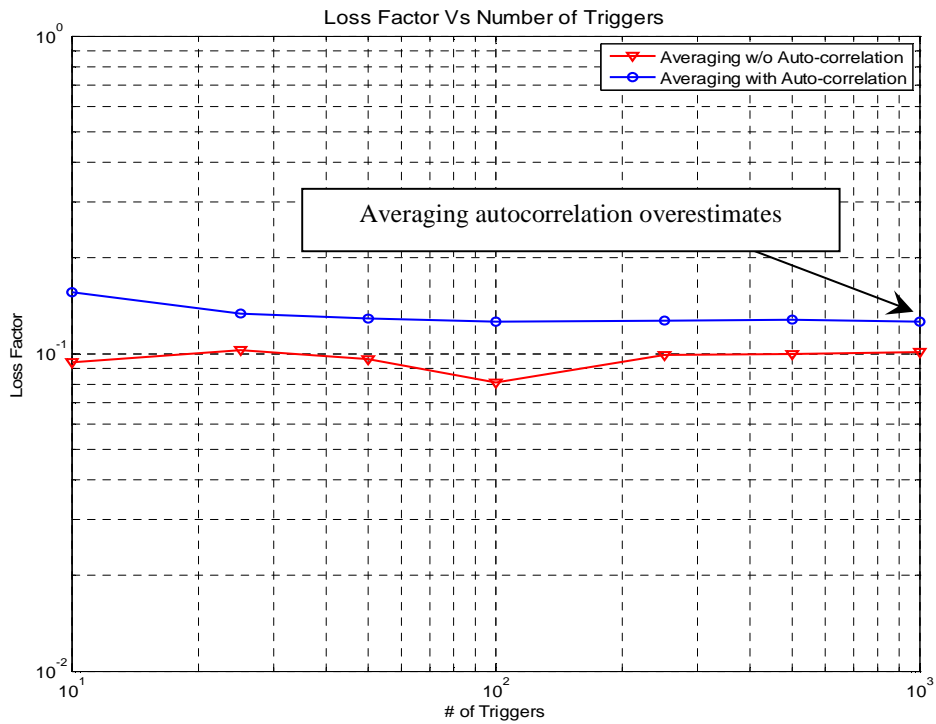
Figure 45: Study of effect of number of triggers on loss factor estimation (for a 1DOF system with simulated LF = 0.1)
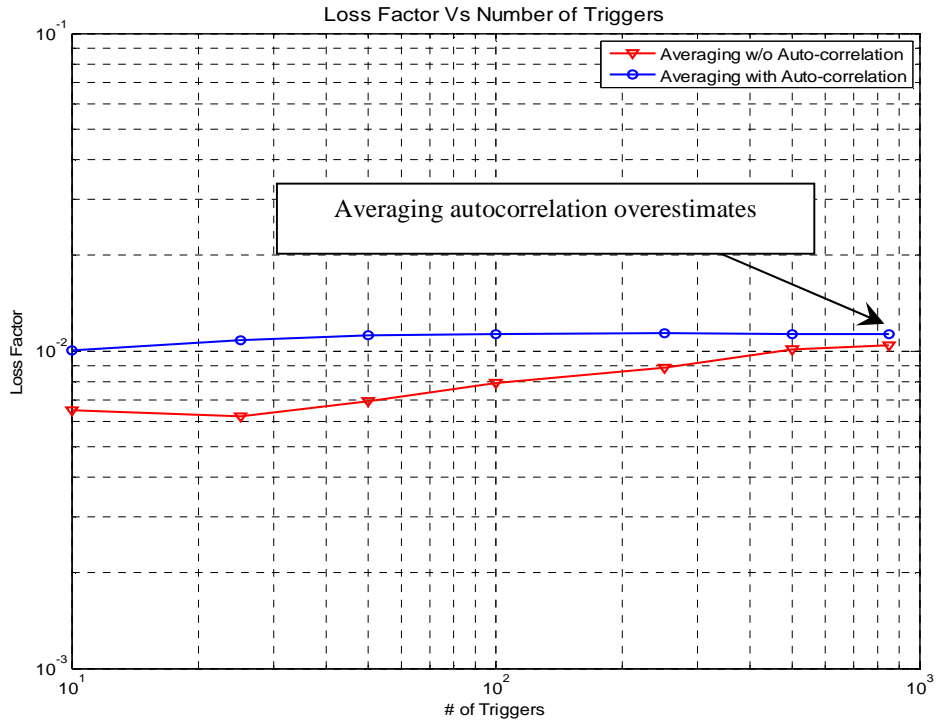
Figure 46: Study of effect of number of triggers on loss factor estimation (for a 1DOF system with simulated LF = 0.01)

3.1.4 Filtering

Since structures have multiple natural frequencies, bandpass filtering is implemented to analyze the structure's damping in multiple frequency ranges. To study the effect of filters on loss factor estimation, a simulated experiment has been devised. A single impulse response is generated by summing up impulse responses with natural frequencies of 400 Hz, 500 Hz, 600 Hz, 700 Hz, 800 Hz, 1000 Hz, 1250 Hz, 1350 Hz, 1450 Hz, 1550 Hz and 1650 Hz. All these modes have a simulated loss factor of 0.05, which is the intermediate value of loss factor range under consideration in this document. Different filters are used to extract the 1000 Hz mode and analyze the effect of filters on group delay, filter order and loss factor estimation. Results of this analysis are tabulated in Table 2.

It is discussed in section 2.4.2.4 that a tighter and highly attenuated filter will yield a better narrow–band data. Both of these parameters, i.e. stopband attenuation and passband width, have a direct effect on filter order. In general, the computation time is higher if the filter order is higher. In terms of processing time, usually the analysis with implementation of lower order filters is faster. Another important result of the filter study, reported in Table 2, is that if the stopband attenuation is high, the wider filter will yield a better result and if the filter band is narrow, lower stopband attenuation should be adopted. The higher order filters, with very high attenuation and a tighter band, tend to overestimate the loss factor.

For experimental and computational work related to RDT reported herein, filters with stopband attenuation of 20 or 40dB and roll off frequencies of +/− 10% (i.e. $B_\%$) of the extremes of filtering bands are used as filter settings.

Attenuation (dB) →		20 dB	40 dB	60 dB	80 dB
Band Width (or B%) with respect to Band Edge Frequencies ↓					
40 %	Filter Order	216	411	667	923
	Group Delay	108	205.5	333.5	461.5
	Computation Time (sec)	0.014771	0.014872	0.018113	0.021568
	Measured Loss Factor	**0.03774**	**0.0393**	**0.0397**	**0.0393**
30 %	Filter Order	288	548	889	1231
	Group Delay	144	174	444.5	615.5
	Computation Time (sec)	0.012590	0.015923	0.020883	0.023389
	Measured Loss Factor	**0.0385**	**0.0408**	**0.0413**	**0.0450**
20 %	Filter Order	432	821	1334	1846
	Group Delay	216	410.5	667	923
	Computation Time (sec)	0.014031	0.017488	0.023286	0.029481
	Measured Loss Factor	**0.0459**	**0.0427**	**0.0481**	**0.0480**
10 %	Filter Order	864	1642	2667	3692
	Group Delay	432	821	1333.5	1846
	Computation Time (sec)	0.136432	0.022761	0.033606	0.053053
	Measured Loss Factor	**0.0495**	**0.0516**	**0.0521**	**0.0542**

Table 2: Effect of filter settings on loss factor estimation (0.05 is the simulated loss factor for the model)

3.2   Computational Plate Model (MDOF)

To evaluate the performance of RDT, two simulated plate models with loss factors of 0.01 and 0.1 were created. For these models, parameters like filter order and averaging scheme were standardized based on the results obtained from a 1DOF system. The automatic slope selection algorithm was chosen over manual slope fitting in this analysis.

For the computational plate model, the effect of the following parameters on the accuracy of loss factor estimated using RDT are evaluated:

1. Simulated loss factor

    a. 10% or 0.1

    b. 1% or 0.01

2. Filtering

    a. Filtering bands

        i.   Full octave

        ii. $1/3^{rd}$ octave

    b. Filtering scheme

        i.   Time domain filtering: TDF

        ii. Frequency domain filtering: FDF

3. Averaging schemes – to estimate the randomdec signature

    a. Direct averaging that is without (w/o) averaging autocorrelation functions

    b. Averaging autocorrelation functions

4. Number of triggered samples averaged to retrieve the randomdec signature

    a. For simulated loss factor of 0.1: 50, 100, 250, 500

    b. For simulated loss factor of 0.01: 50, 100, 150, 200.

Appendix A provides a detailed review of the above parameters through Figures A.1 to A.16. Also in Appendix A:

Figures A.17 and A.18 – compares loss factor in full and 1/3rd octave frequency bands,

Figures A.19 and A.20 – compares loss factor for TDF and FDF,

Figures A.21 and A.22 – compares loss factor computed from direct averaging and averaging the autocorrelation functions.

Tables 5 through 8, in Appendix A, summarize the loss factor averaged over the entire frequency range, 100 Hz to 5 kHz.

The Figures 47 and 48 represent the estimated loss factor, using RDT, in 1/3rd and full octave frequency bands with LF3 approach and both possible averaging schemes (to compute the randomdec signature).

3.2.1 Filtering Scheme – FDF or TDF

Both, Time Domain Filtering (TDF) and Frequency Domain Filtering (FDF) approaches have successfully predicted the simulated loss factors for the computational plate models. For practical applications of RDT, only time domain filtering is used as it is based on absence of knowledge of the input force.

3.2.2 Choice of Filtering Bands – 1/3rd Octave or Full Octave Bands

Full octave filters are wider in frequency band than the 1/3rd octave filters. For this reason, the full octave filters with the same central frequency as that of 1/3rd octave band have lower filter order. It is observed from Figures A.17 and A.18 that the loss factor estimated in 1/3rd octave filtered response are essentially the same as the estimates from full octave filtered response. The filter order for full octave bands are lower than the filter order of 1/3rd octave

band, hence the full octave filters will be faster. The marginally enhanced performance of $1/3^{rd}$ octave filters can be spared for computationally faster full octave filters.

3.2.3 Averaging Schemes

Direct averaging and averaging of autocorrelation functions provided consistent results for damping levels of 0.1 and 0.01. Unlike, the 1DOF system, the latter approach did not overestimate the loss factor of 0.1. For the loss factor of 0.01 almost all the averaging schemes have slightly over–estimated the loss factor. This may be due to the possibility of the excitation point being positioned on multiple node lines.[8,9]

The averaging of autocorrelation function converges quickly with fewer triggers to the simulated loss factor, hence it is recommended for experimental analysis. Also in these simulations, modal damping was constant for all the modes. For a simulated loss factor of 0.1 (highly damped plate) averaging of autocorrelation functions performed better than the direct averaging approach, in both full and $1/3^{rd}$ octave bands.

LF2 and LF3 averaging schemes have performed better than the LF1 averaging approach. LF3 is finally chosen over LF2 because of higher flexibility in experimental application. Results included herein are for the LF3 approach only.
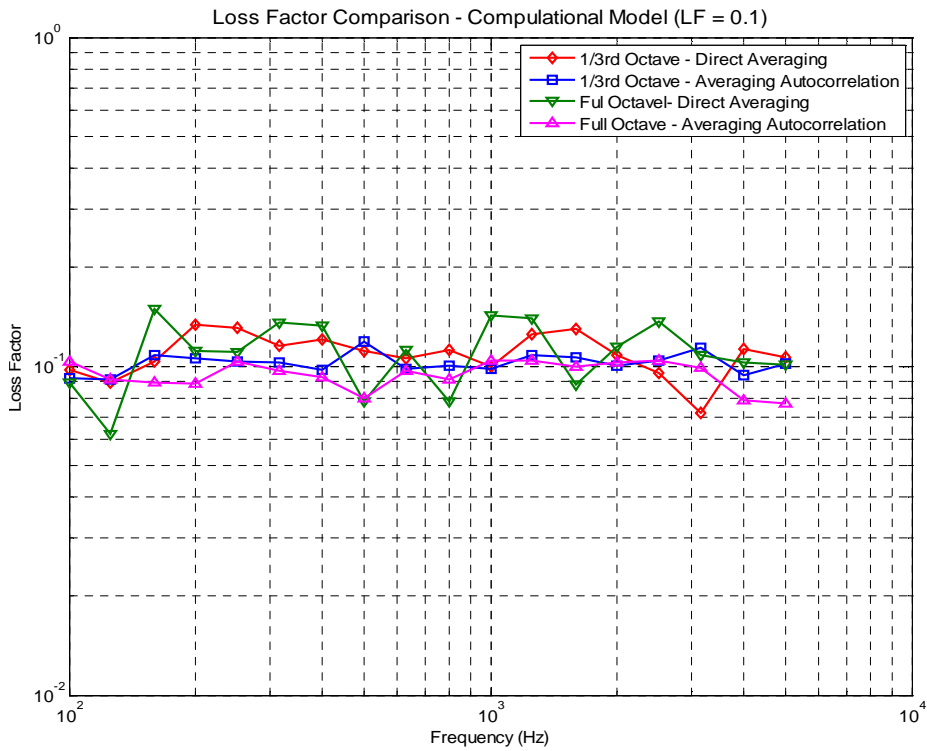
Figure 47: RDT – Loss factor estimates for computational model with simulated LF = 0.1
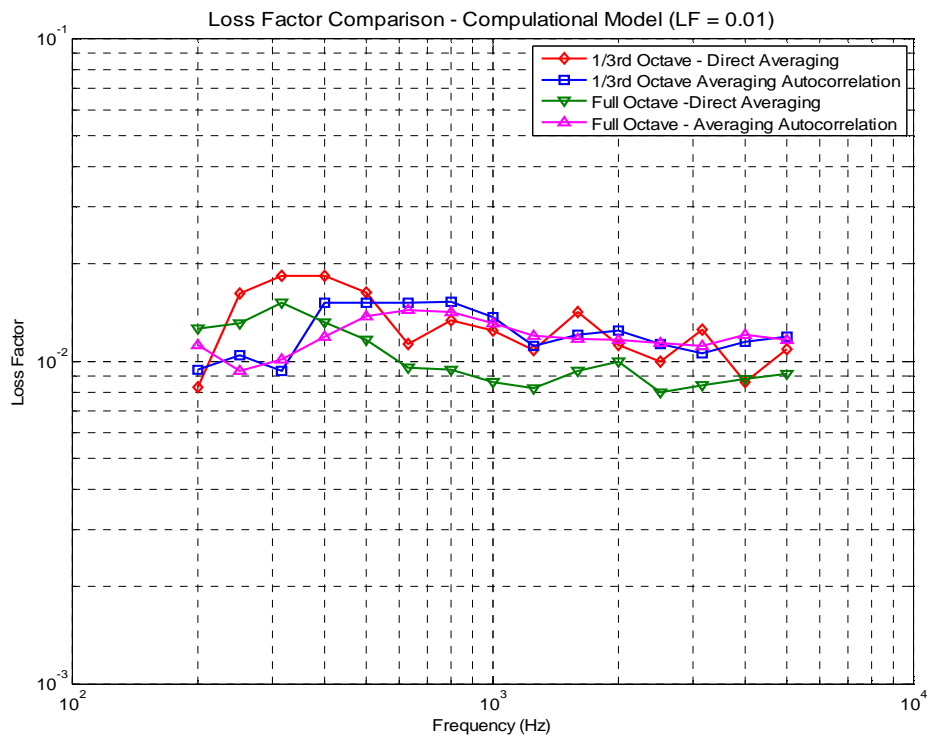
Figure 48: RDT – Loss factor estimates for computational model with simulated LF = 0.01

Based on the study of the computational plate model and the results obtained, we can recognize that loss factor estimation using RDT is promising, in particular for higher loss factors (comparing Figures 47 and 48). It is concluded that the optimum setting for experimental loss factor estimation would include:

1. Time Domain Filtering in full octave bands centered at $1/3^{rd}$ octave central frequencies,

2. Averaging the dB decay curves (LF3 scheme) for loss factor estimation is chosen over other the two approaches (LF1 and LF2),

3. Automated slope fitting works and is highly recommended,

4. Length of triggered sample can be calculated based on eqn. 2.6,

5. Kaiser Window (FIR Filter), with 40dB stopband attenuation at +/–10% of the passband frequencies have successfully performed as filter parameters.

Table 3 and 4 represent the estimated loss factor for MDOF system with simulated loss factor of 0.1 and 0.01, respectively. For the computational model with a simulated loss factor of 0.1, a total of 500 triggered samples are averaged. For the case of computational model with a loss factor of 0.01, a total of 200 triggered samples are averaged.

The loss factor estimates for the highly damped computational models are more accurate if $1/3^{rd}$ octave filtering and autocorrelation averaging are implemented. On the other hand, the loss factor estimates for the lightly damped computational plate are most accurate when direct averaging and full octave bands are used.

Averaging Scheme	Frequency Band	
	Full Octave	$1/3^{rd}$ Octave
Direct Averaging	0.1107	0.1095
Averaging Autocorrelation	0.0946	0.1025

Table 3: Averaged LF, over the entire frequency range, for computational plate model (MDOF with simulated LF = 0.1) – Summary of Table 5

	Frequency Band	
Averaging Scheme	Full Octave	1/3rd Octave
Direct Averaging	0.0103	0.0129
Averaging Autocorrelation	0.0120	0.0123

Table 4: Averaged LF, over the entire frequency range, for computational plate model (MDOF with simulated LF = 0.01) – Summary of Table 6

3.3   Experimental Results

Figures 49 through 51 show the loss factor estimated by RDT and IRDM for three damped plates in full octave bands. These plates all had constrained layer damping treatments which produced damping loss factors in the range of 1% to 10%. These tests were conducted simply to assess the level of agreement between RDT and IRDM.

For the RDT results, the data processing also included averaging the dB decay from the autocorrelation of the triggered response in full octave bands with the largest number of triggers available. The IRDM results were obtained both using persistent random excitation and using an impulse hammer.

It can be noted, for all three plates tested, that hammer tests have not yielded the best loss factor estimates. Also there is considerable underestimation beyond 2500 Hz, which possibly is due to the impulse hammer tip selected for which the performance decreases with frequency. For mechanical excitation, the loss factor estimated using IRDM and RDT have a similar trend.

RDT's computational efficiency (processing time requirements) is lower for the lightly damped plates evaluated because of the longer record required, from which the triggered samples are to be detected. In cases like these, only high frequency bands should be selected or averaging based on autocorrelation should be used, as it converges in a smaller number of triggered samples (but slightly overestimates).

From previous work by Dande and Ewing[26], it is expected that IRDM underestimates loss factor of highly damped plates. This underestimation is also observed in the experimental study reported herein. By way of explanation, the "direct field" will dominate the response near the point of excitation and the "reverberant field" will dominate at large distances, i.e. away from the excitation point. The radius of the direct field is directly proportional to the loss factor and frequency band under analysis.[1] Thus, for a plate which is highly damped, there is higher probability of having more accelerometers positioned inside the direct field. The kinetic energy inside the direct field is higher than that in the reverberant field. For a highly damped and small plate, loss factor estimation based on randomly positioned accelerometers – which have a higher probability of being <u>inside</u> the direct field will underestimate damping. In the case of a lightly damped plate, the direct field is small and hence more of accelerometers, randomly placed, will be positioned outside the direct field. Therefore, for a lightly damped plate, the loss factor is more likely to be estimated from the accelerometers in the reverberant field – which is a key assumption of any method based on an input–output relationship.

Hence, it is hypothesized that the loss factor calculated from the accelerometers inside the direct field will be underestimated, especially for the highly damped plates which have relatively larger direct field. Loss factor estimated using RDT, which is entirely a time domain technique, does not seem to be affected by positioning of accelerometers inside or outside the direct field.

For impulsive excitation, a modally tuned impulse hammer with a plastic vinyl tip is used. In general, the loss factor estimated using this plastic vinyl hammer tip is acceptable up to 2500 Hz. In experimental analysis provided in following sections, it is observed that for highly damped plates, the hammer excitation does not generate promising results. It is

possible that the impulsive excitation provided lacks sufficient energy to effectively excite the highly damped structure.

In the presented experimental analysis, the measured random response is filtered in full octave bands. The trigger settings considered are: trigger level of twice the standard deviation of response and a trigger band is 20% of the lower trigger level. Only LF3 averaging scheme is considered in the loss factor estimation of following three plates.

3.3.1   Plate 1 – Results

Plate 1 has the highest damping level amongst the three plates analyzed for loss factor estimation. Based on discussion presented before this section, it is expected that the frequency domain techniques, such as IRDM, will tend to underestimate the loss factor. A total of 500 triggered samples were averaged to generate the randomdec signature. Following observations can be made based on estimated loss factors plotted in Figure 49:

1. RDT and IRDM (with shaker excitation) have a good agreement in the 1000Hz and above frequency range. Unlike RDT, IRDM (with shaker excitation) underestimated the loss factors in lower frequency ranges (i.e. below 500 Hz).

2. The IRDM hammer test results are not consistent with the loss factors estimated from RDT or IRDM for persistent mechanical excitation.
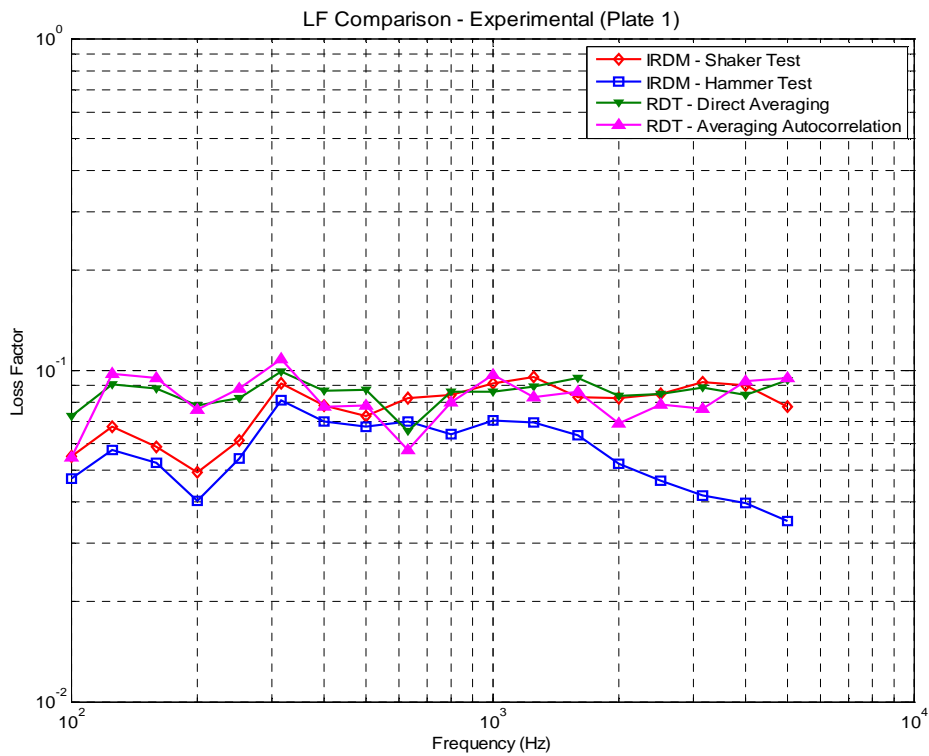
Figure 49: RDT – Experimental loss factor estimates for Plate 1

3.3.2 Plate 2 – Results

Plate 2 is the intermediately–damped plate amongst the three plates analyzed. A total of 500 triggered samples were averaged to generate the randomdec signature. IRDM has performed relatively better for this plate. The following observations can be made based on estimated loss factors plotted in the Figure 50:

1. RDT loss factor estimates have moderately good agreement with IRDM (with persistent mechanical excitation).

2. IRDM with impulse excitation has relatively better loss factor estimation for this plate. The problem with loss factor estimation in the higher frequency bands persists and loss factor is underestimated.

3. IRDM, with either shaker or impulse excitation, has a "dip" in loss factor estimation curve in the 200Hz to 400Hz frequency range. In IRDM frequency domain filtering is implemented, hence, it is possible that the dominating mode may have had a natural frequency at the "edge" of the frequency band, and may have not been adequately sampled.
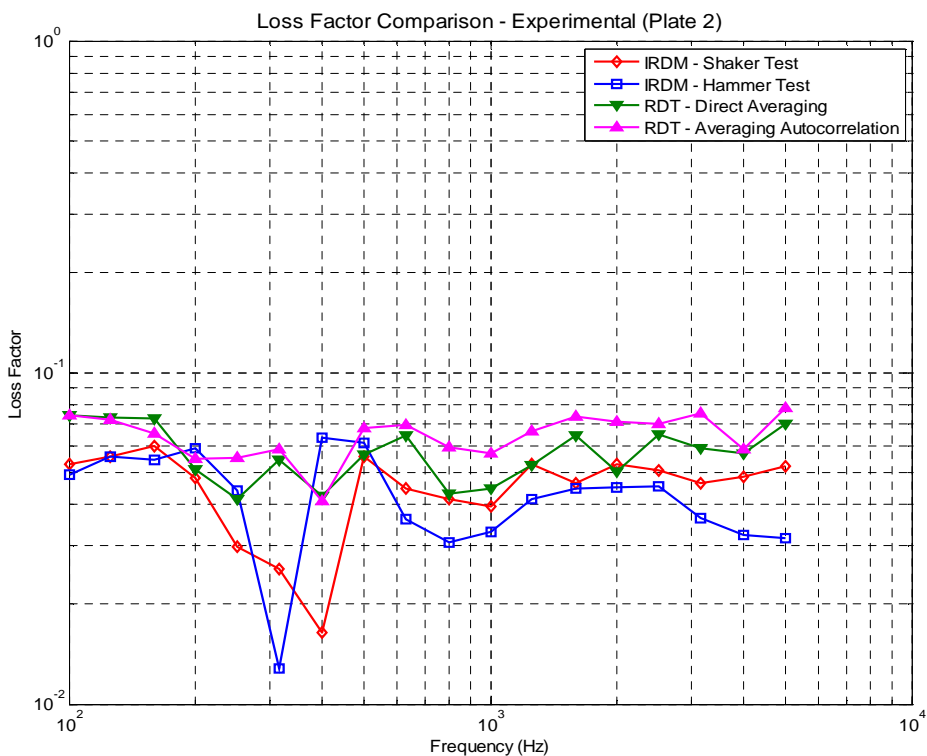
Figure 50: RDT – Experimental loss factor estimates for Plate 2

3.3.3   Plate 3 – Results

Plate 3 is the least damped plate amongst the three plates being analyzed and has partial constrained layer damping treatment. The lightly damped plate required a longer time history to extract triggers. But due to practical limitations on the DAQ, a finite length time history is recorded which yielded 200 triggered samples in the 100 Hz frequency band. Hence, 200 triggered samples were averaged to generate the randomdec signature. Averaging

autocorrelation functions underestimated the loss factor as compared to direct averaging. This is an atypical trend based on the study of Plate 2 and the computational simulations. The loss factor estimated for this plate, using IRDM and RDT have a very good agreement. These results are presented in Figure 51.
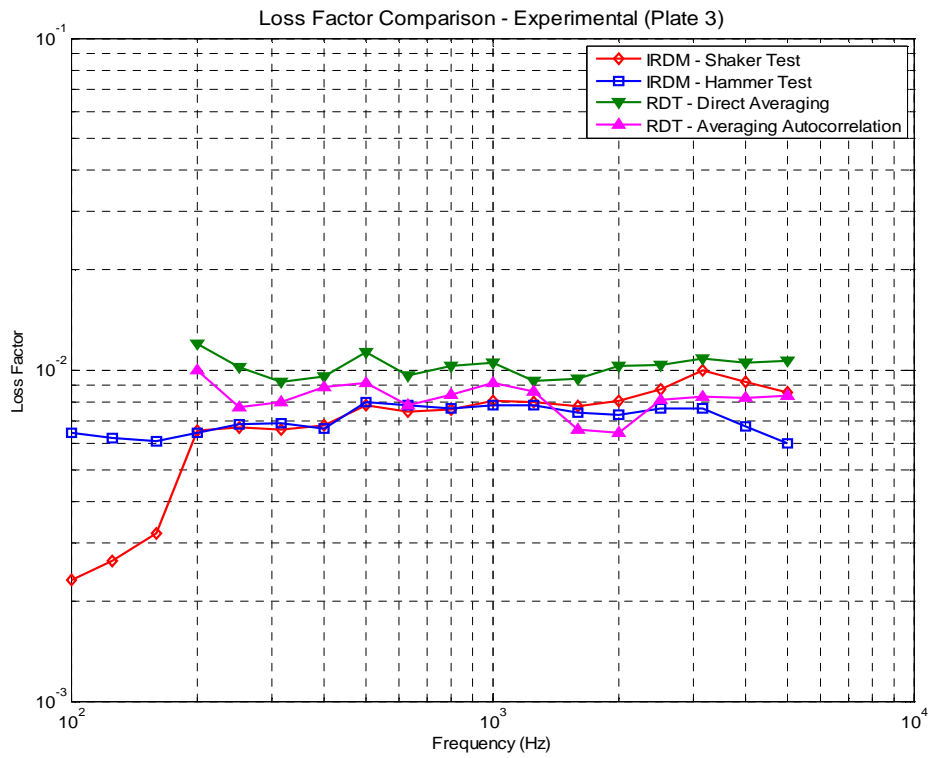
Figure 51: RDT – Experimental loss factor estimates for Plate 3

4   Conclusions

This chapter is aimed at the discussion of conclusions for efficient implementation of RDT in loss factor estimation. This discussion is guided by the experience gained in experimental studies using the "optimum" parameter settings from the computational studies.

4.1   Length of Sample

The length of triggered sample (or randomdec signature), <u>in terms of cycles</u> to damp–out, is a function of damping loss factor. The following relation is derived empirically and has been implemented to optimize the length of time history of response to be measured.

$$N \cong \frac{2.2}{\eta_{est}}$$
(eqn. 2.8)

The triggered samples have no overlap; therefore, the length of measured response should be at least equal to the cumulative length of all the triggered samples. From equation 2.8, it is noted that the length of triggered samples – in terms of cycles – is independent of frequency. But the length of randomdec signature or triggered sample in terms of seconds is inversely proportional to frequency and will be longer for lower frequencies. Therefore, the length of time history to be recorded in an experiment should be based on the lowest frequency under consideration.

4.2   Filtering

Filtering is first step in the RDT to "condition" the measured response in narrow frequency bands for loss factor analysis. In the computational and experimental loss factor estimation, filters with a stopband attenuation of 40 dB and a surplus width of passband equal to 10% yielded the best results.

4.2.1 Group Delay

The group delay or time delay due to filtering is inevitable but it does not affect the decay characteristics. The group delay only causes a delay in the detection of the first trigger. Group delay is directly proportional to the filter order, i.e. for higher filter order the group delay will be higher.

4.2.2 Filter Order

It is recognized that higher order filters can successfully eliminate the out–of–band frequencies. The dB decay of response, filtered with a higher order filter, is noise–free and the initial decay – from which the loss factor is to be estimated – is more prominent with higher order filters.

The higher order filters, with very high attenuation and/or a tighter band, tend to overestimate the loss factor. If the stopband attenuation is high, a wider filter will yield better results. If the filter is too narrow, lower stopband attenuation is preferred.

4.2.3 TDF vs. FDF

For the computational plate models, both filtering schemes – TDF and FDF – were successfully used in the overall process to accurately estimate the simulated loss factor. For RDT–based loss factor analysis, the force causing the excitation is not measured. Thus, the usual input–output based FDF can not be implemented. But the equivalency of TDF and FDF has been established.

4.2.4 1/3rd Octave vs. Full Octave Filters

Full octave and 1/3rd octave filters performed essentially the same, but 1/3 octave filters are more computationally–intensive making them less attractive. In the experimental

analysis, only full octave wide filters, centered at $1/3^{rd}$ octave central frequencies, are selected.

4.3   Triggering

If the triggering conditions are very strict, such as very high trigger levels and/or very narrow trigger bands, fewer triggers are detected. A trigger setting with a lower trigger level of twice the standard deviation of response and a trigger band of 20% has been shown to typically produce the best results in the computational studies. Therefore these trigger conditions were also used for the experimental loss factor estimations.

If a large number of triggered samples are averaged, the quality of the randomdec signature and dB decay improves. That is, by increasing the numbers of triggered samples in the averaging process, the performance of RDT for loss factor estimation can be enhanced.

4.4   Averaging Schemes to Estimate the Decaying Response

Although averaging the autocorrelation function approach converges quickly, i.e. with only a few triggered samples averaged to retrieve the randomdec signature, it tends to slightly overestimate the loss factor. On the contrary, the direct averaging approach displays slow convergence but accurately predicts the loss factor.

For a time history in which few triggers are detected, the averaging autocorrelation function approach is recommended if the amount of data available is limited.

In the following section, the results for the above–mentioned averaging schemes are discussed with respect to LF1 and LF3 averaging approaches.

4.5 Averaging Schemes to Estimate the dB Decay

If the response is measured from multiple locations on a plate, the loss factor can be estimated using either the LF1, LF2, or LF3 averaging scheme. For the computational plate model, the LF1 averaging scheme's performance is inferior with respect to LF2 and LF3 averaging schemes. Averaging the dB decay curves, i.e. the LF3 averaging scheme, is recommended for the experimental loss factor estimation process.

For the computational plate models with simulated loss factor of 0.1 and 0.01, the estimated loss factor – in full octave band and utilizing time domain filtering – is averaged over the entire frequency range.

From Table 7, estimated loss factors have a variance of:

1. 5.4% using the LF3 approach and 8.5% using the LF1 approach for direct averaging

2. 10.7% using the LF3 approach and 21.2% using the LF1 approach for the averaged autocorrelation function approach.

From Table 8, estimated loss factors have a variance of:

1. 20.0% using the LF3 approach and 31.0% using the LF1 approach for the direct averaging approach

2. 3.0% using the LF3 approach and 47.0% using the LF1 approach for averaging autocorrelation functions.

Clearly the LF3 averaging approach provides a lower variance of loss factor estimates.

4.6   Automatic Slope Fitting

An automated slope fitting algorithm – based on fitting a linear curve through the local maxima/ peaks in the dB decay curve – performed efficiently and is implemented in the experimental and computational studies.

Slope fitting is a repetitive, at times monotonous, process when multiple frequency bands are under analysis. Therefore, automated slope fitting is strongly recommended.

4.7   Experimental Performance of RDT

Based on the computational 1DOF and MDOF plate model, the following parameters are selected for experimental loss factor estimation using RDT:

1. Triggering level – Twice the standard deviation

2. Triggering band – 20% of lower trigger level

3. Filtering scheme – time domain filtering using full octave filters with 40 dB attenuation and 10% surplus passband width

4. Averaging scheme (to estimate the dB decay) – LF 3, i.e. averaging the dB decay curves

5. Slope fitting – Automated.

Based on results presented in Figures 49 and 50, for Plates 1 and 2, it is observed that RDT out–performed IRDM. For these relatively highly damped plates, IRDM (with impulsive excitation) rarely predicted loss factors that are consistent with loss factor estimates from either RDT or IRDM with persistent random excitation. It is hypothesized that this underestimation of measured loss factor, for the impulsive excitation–based IRDM, is due to the larger radius of the direct field associated with high damping. Apparently,

impulsive excitation is less effective for establishing a reverberant field than the persistent excitation.

For the lightly damped plate, Plate 3, the loss factor estimates from RDT and IRDM are consistent. For Plate 3, the direct averaging approach overestimated loss factor as compared to the averaging autocorrelation approach. This is an anomalous trend, as it is expected that using the averaged autocorrelation functions will result in over–estimated loss factor, as noted in the 1DOF and MDOF computational models.

5   Recommendations for Further Study

RDT has successfully performed for the 1DOF computational model, a computational plate model and plates treated with constrained layer damping. However, further studies are indicated.

Panel damping loss factors of most interest in the aerospace industry typically range between 0.01 and 0.1. It is suggested that if RDT is to be used on lightly damped structures (loss factors below 0.01) preference should be given to analysis in higher frequency bands.

The conclusion of experiments on three damped plates in the laboratory is that the RDT and IRDM provide similar estimates of damping for plates with loss factors between 1% and 10%. Testing on more complicated structures is now needed.

One topic for further study is to evaluate loss factors using acoustic excitation. Since the applied force need not be measured, this makes acoustic excitations appealing.

The loss factor estimates by IRDM, for a highly damped plate, under–predicted when using impulse excitation. It is recommended that the cause of the underestimation of loss factor when using an impulse hammer should be investigated.

Filters add a finite damping to the filtered response. This additional damping will cause overestimation of the loss factor, which is more evident in higher order filters and for systems with lower loss factors. It is recommended that the effect of finite damping added because of filtering should be evaluated in future studies.

6 References

6.1 Primary References

[1]   Lyon, R. H. and Dejong, R. G., *Theory and Application of Statistical Energy Analysis*, 2nd edition, RH Lyon Corp 1998

[2]   Liu, W. B., *Experimental and Analytical Estimation of Damping in Beams and Plates with Damping Treatments*, PhD Dissertation, The University of Kansas, Aerospace Engineering, 2009

[3]   Adhikari, S. and Woodhouse, J., *Identification of Damping: Part1, Viscous Damping*, Journal of Sound and Vibration, 2001, Vol. 243(1), pp. 43–61

[4]   Chu, F. H. and Wang, B. P., *Experimental Determination of Damping in Materials and Structures*, American Society of Mechanical Engineers, Applied Mechanics Division, AMD, Vol. 38, pp. 113–122,1980

[5]   Cole, H. A., Jr., *On–The–Line Analysis of Random Vibrations*, AIAA paper No. 68–288, AIAA/ASME 9th Structures, Structural Dynamics and Materials Conference, Palm Springs, CA, 1968

[6]   Vandiver, J. K., Dunwoody, A. B., Campbell, R. B. and Cook, M. F., *A Mathematical Basis for the Random Decrement Vibration Signature Analysis Technique*, Journal of Mechanical Design, Vol. 144, pp. 307–313, 1982

[7]   Ewing, M. S., Dande, H. and Vatti, K., Validation of Panel Damping Loss Factor Estimation Algorithms Using a Computational Model, Proceedings of 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Palm Springs, CA, 2009

[8]   Bloss, B. C. and Rao, M. D., *Estimation of Frequency–Averaged Loss Factors by the Power Injection and the Impulse Response Decay Methods*, Journal of the Acoustical Society of America, Vol. 117(1), pp. 240–249, 2005

[9]   Liu, W. B. and Ewing, M. S., *Experimental and Analytical Estimation of Loss Factors by the Power Input Method*, AIAA Journal, Vol. 45(2), pp. 477–484, 2007

[10]   Wu, L., Agren, A. and Sundback, U, *A Study of Initial Decay Rate of Two–Dimensional Vibrating Structures in Relation to Estimates of Loss Factor*, Journal of Sound and Vibration, Vol. 206 (5), pp. 663–684, 1997

[11]   Brincker, R., Kirkegaard, P. H., and Rytter, A., *Identification of System Parameters by the Random Decrement Technique,* Proceedings of 16th International Seminar on Modal Analysis, Florence, Italy, 1991

[12]   Cole, H. A., Jr., *On–Line Failure Detection and Damping Measurement of Aerospace Structures by Random Decrement Signatures*, NASA CR–2205, 1973

[13]   Brincker, R., Krenk, S., and Jensen, J. L., *Estimation of Correlation Functions by the Random Decrement Technique*, Skandinavisk Forum for Stokastic Mekani, Sweden, 1990

[14]   Asmussen, J.C., Ibrahim, S. R. and Brincker, R., *Random Decrement: Identification of Structures Subjected to Ambient Excitation*, Proceedings of 16th International Modal Analysis Conference, Santa Barbara, CA, 1998

[15]   Asmussen, J. C., Ibrahim, S. R., and Brincker, R., *Application of Vector Triggering Random Decrement*, Proceedings of 15th International modal Analysis Conference, Orlando, FL, 1997, Vol. II, pp. 502–509

[16]   Asmussen, J. C., Ibrahim, S. R., and Brincker, R., *Statistical Theory of the Vector Random Decrement Technique*, Journal of Sound and Vibration, Vol. 2(2), pp, 329–344, 1999

[17]   Ibrahim, S. R., *Incipient Failure Detection from Random Decrement Time Functions*, American Society of Mechanical Engineers, Applied Mechanics Devision, AMD, Vol. 65, pp. 69–81, 1984

[18]   Ibrahim, S. R., *Efficient Random Decrement Computation for Identification of Ambient Responses*, Proceedings of International Modal Analysis Conference and Exhibit, Vol. 1, pp. 1–6, 2001

[19]   Ibrahim, S. R., *Random Decrement Technique for Modal Identification of Structures,* Journal of Spacecraft and Rockets, Vol. 14 (11), pp. 696–700, 1977

[20]   Brincker, R. and Asmussen, J. C., *Random Decrement Based FRF Estimation*, Proceedings of International Modal Analysis Conference and Exhibit, Vol. 2, pp. 1571–1576, 1997

[21]   Asmussen, J. C. and Brincker, R., *A New Approach of Predicting the Variance of Random Decrement Technique*, Proceedings of International Modal Analysis Conference and Exhibit, Vol. 1, pp. 542–549, 1998

[22]   Asmussen, J. C., *Modal Analysis Based on The Random Decrement Technique—Application to Civil Engineering Structures*, PhD Dissertation, University of Aalborg, Department of Building Technology and Structural Engineering, 1997

[23]   Kijewski, T., and Kareem, A., *On the Reliability of a Class of System Identification Techniques: Insights from Bootstrap Theory*, Proceedings of 8th International Conference on Structural Safety and Reliability, Structural Safety Vol. 24(2–4), pp. 261–280, 2002

[24] Kijewski, T., and Kareem, A., *Reliability of Random Decrement Technique for Estimates of Structural Damping*, 8th ASCE Specialty Conference on Probabilistic Mechanics and Structural Reliability, PMC2000–294, 2000

[25] MATLAB Product Help, *FIR Filter Design :: Filter Design and Implementation (Signal Processing Toolbox[TM])*, MATLAB Version 7.8.0.347 (R2009a)

[26] Dande, H. and Ewing, M. S., *Panel Damping Loss Factor Estimation Using the Random Decrement Technique*, Proceedings of 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Orlando, FL, 2010

6.2   Secondary References

1.   Brown, D. L. Allemang, R. J., Zimmerman, R., and Mergeay, M., *Parameter Estimation Techniques for Modal Analysis*, SAE Preprints, n 790221,1979

2.   Ewins, D. J., *Modal Testing: Theory and Practice*, Research Studies Press, 1984

3.   Hartog, D., *Mechanical Vibrations*, Dover Publications Inc, New York, 1985

4.   Rao, S. S., *Mechanical Vibrations*, 3rd Edition, Addison–Wesley Publishing Company, 1995

5.   Inman, D. J. , *Engineering Vibrations*, 3rd Edition, Pearson Prentice Hall, New Jersey, 2007

Appendix A: Computational Plate Model – Extended Study

This appendix is aimed at providing a detailed review of loss factor estimation process and effect of various parameters on its accuracy. The study herein is only for the computational plate model. These results were considered as a basis for establishing parameters for the experimental analysis.

The results are categorized in the following chart to facilitate the reader to identify a particular study of interest. These results compare the LF1, LF2 and LF3, the averaging schemes to obtain the dB decay, for various damping levels, filtering and averaging schemes.

LF = 10%

	Frequency Domain Filtering		**Time Domain Filtering**	
	w/ autocorrelation	w/o autocorrelation	w/ autocorrelation	w/o autocorrelation
1/3rd Octave	Figure A. 3	Figure A. 1	Figure A. 4	Figure A. 2
Full Octave	Figure A. 7	Figure A. 5	Figure A. 8	Figure A. 6

LF = 1%

	Frequency Domain Filtering		**Time Domain Filtering**	
	w/ autocorrelation	w/o autocorrelation	w/ autocorrelation	w/o autocorrelation
1/3rd Octave	Figure A. 11	Figure A. 9	Figure A. 12	Figure A. 10
Full Octave	Figure A. 15	Figure A. 13	Figure A. 16	Figure A. 14

In Appendix A, the averaging schemes to estimate the free decay are defined as:

1. w/ autocorrelation   – Averaging the autocorrelation functions

2. w/o autocorrelation – Direct averaging

The Figures A.1 to A.16, have three subplots and these subplots correspond to the loss factor estimates for averaging schemes: LF1, LF2 and LF3. The four curves, in these subplots, represent the loss factors estimated from the randomdec signature computed by

averaging different numbers of triggered samples. The title in following figures is in the format – 'LFi Comparison (LF = X%; Filtering Type; Filtering Band; Averaging Scheme)'. The terms in the title are defined as:

LFi Comparison   – i is the index, 1 to 3, to define the type of averaging scheme utilized for

computing dB decay is compared

first subplot – LF1 approach

second subplot – LF2 approach and

third subplot – LF 3 approach

LF = X%   – simulated loss factor – X% is equal to either 1% or 10%

Filtering Type   – type of filtering scheme – FDF or TDF

Filtering Band   – width of filtering band – 1/3rd octave or full octave

Averaging Scheme – w/ Autocorrelation or w/o Autocorrelation.

The legend, common for all three subplots in following figures, indicate the respective loss factor curves for different number of triggered samples been averaged for computation of the randomdec signature.

The x–axis defines the frequency range – 100 Hz to 10 kHz – for analysis of loss factor in central frequency bands from 100 Hz to 5 kHz.

Figures A.17 and A.18 – compares loss factor in full and 1/3rd frequency octave bands,

Figures A.19 and A.20 – compares loss factor for TDF and FDF, and

Figures A.21 and A.22 – compares loss factor for direct averaging and averaging autocorrelation function approach.

Tables 5 through 8 summarize the results by tabulating estimated loss factors averaged over the entire frequency range – 100 Hz to 5 kHz.

102

Figure A. 1: Computational Model (LF = 0.01, 1/3rd Octave, Direct Averaging, FDF)
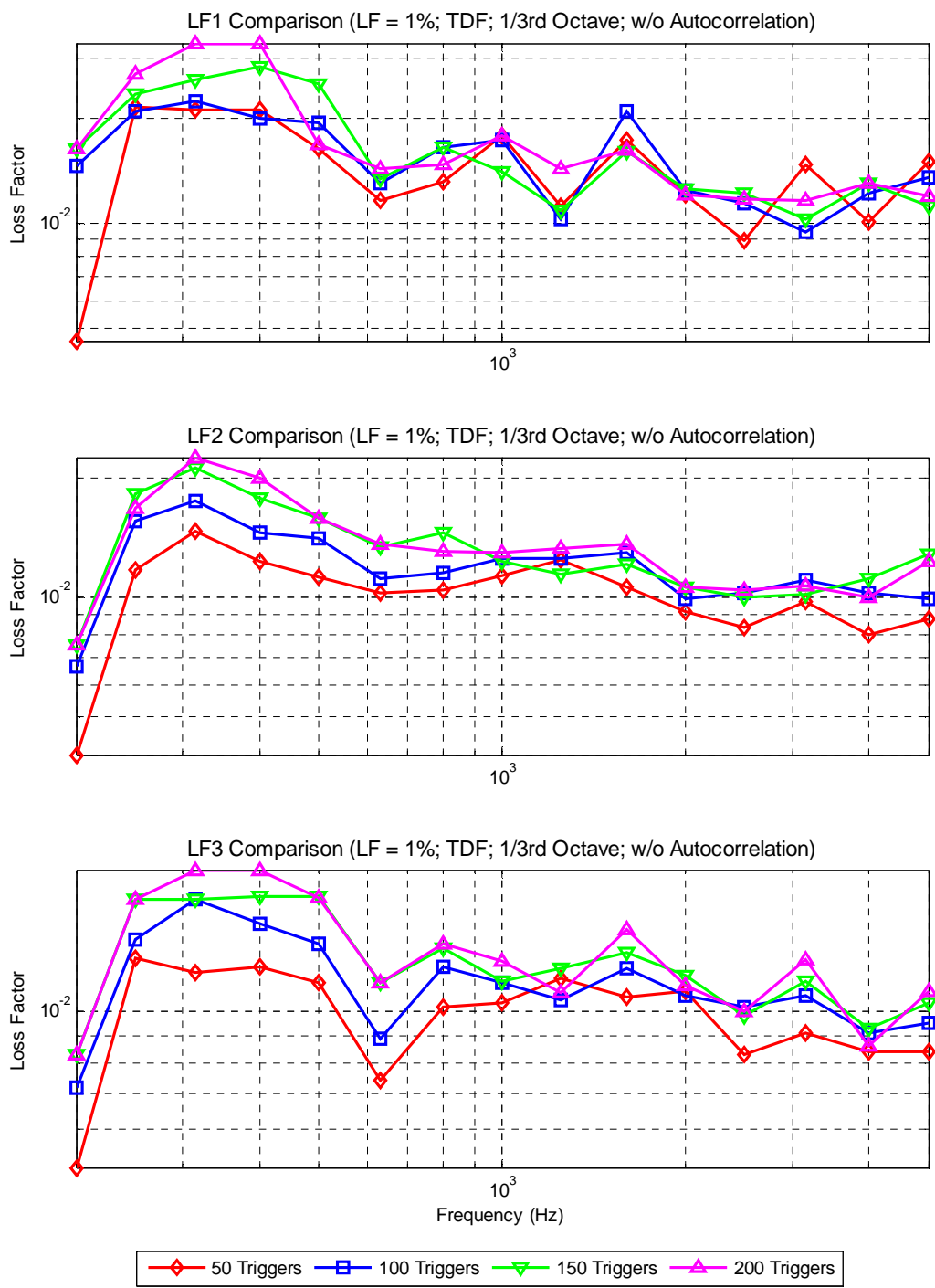
Figure A. 2: Computational Model (LF = 0.01, 1/3rd Octave, Direct Averaging, TDF)
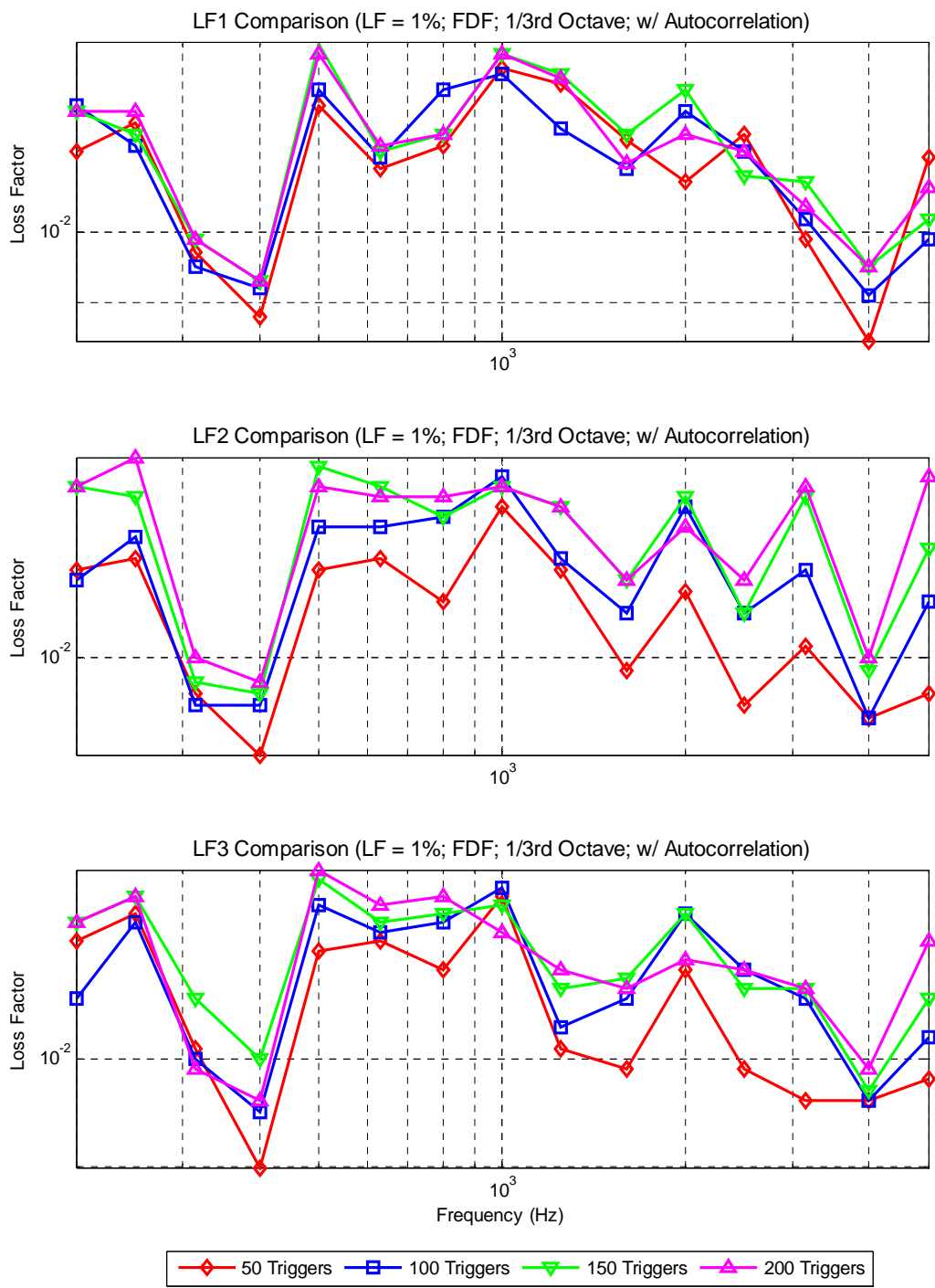
Figure A. 3: Computational Model (LF = 0.01, 1/3rd Octave, Averaging Autocorrelation, FDF)
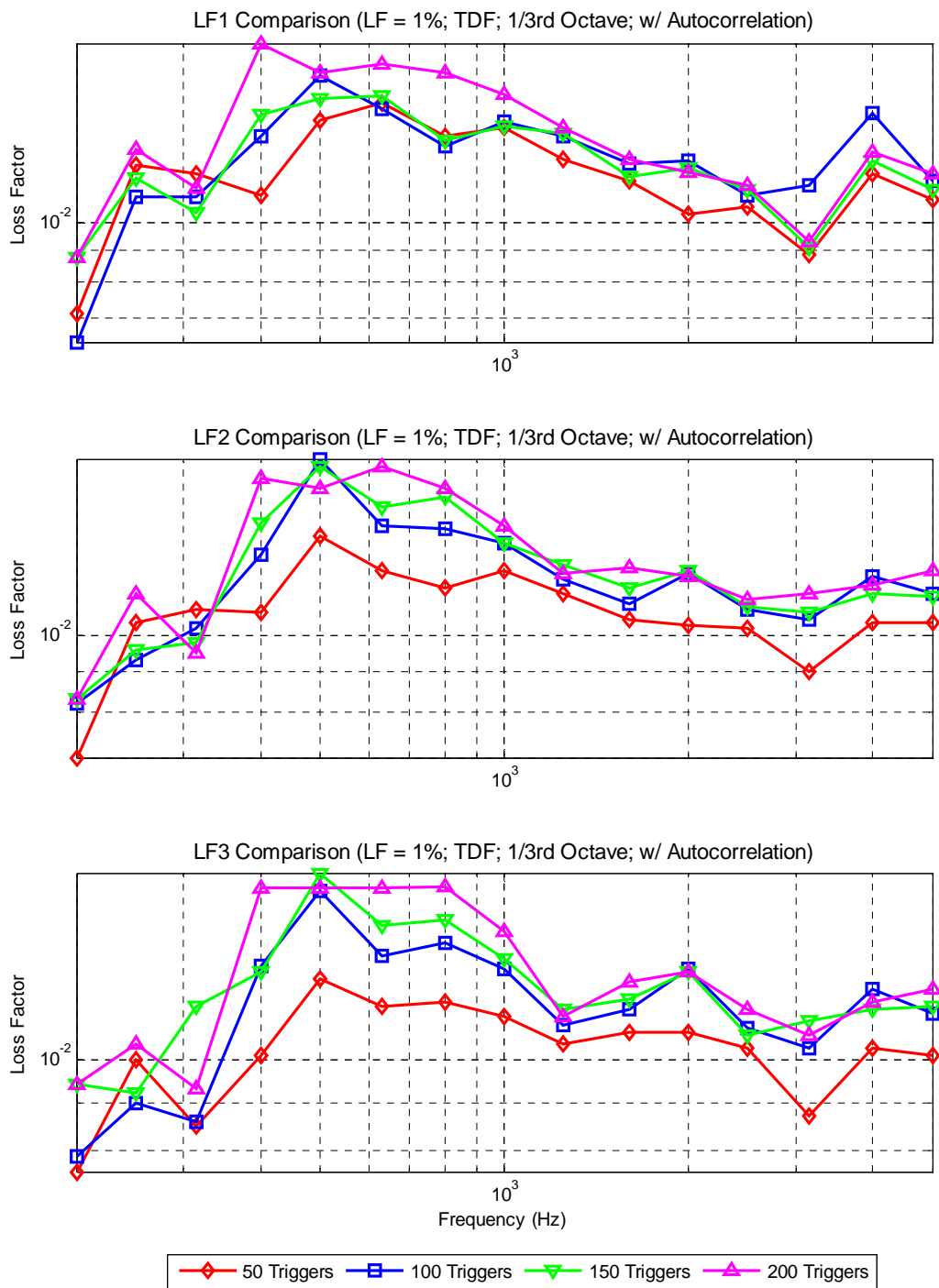
Figure A. 4: Computational Model (LF = 0.01, 1/3rd Octave, Averaging Autocorrelation, TDF)
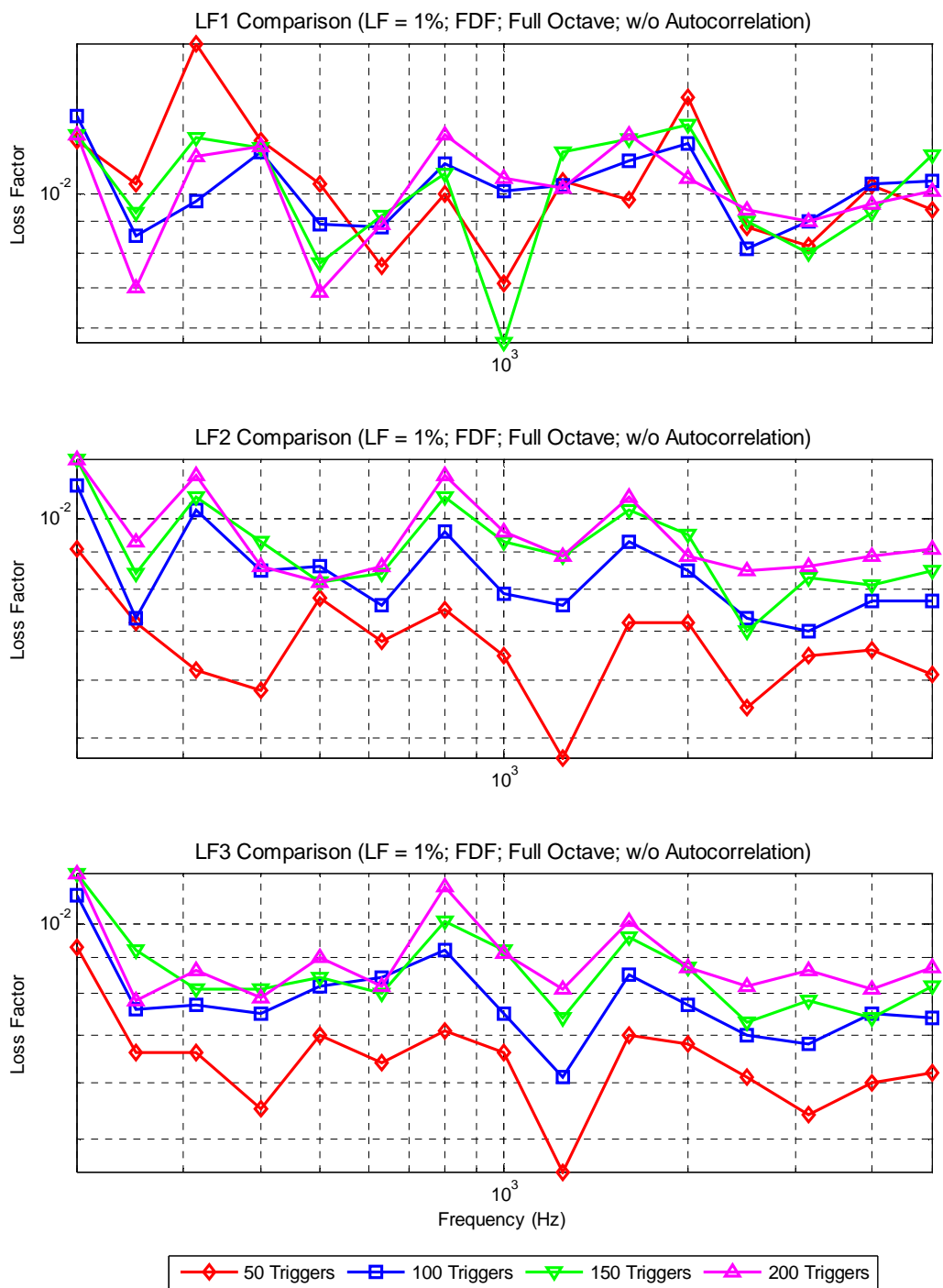
Figure A. 5: Computational Model (LF = 0.01, Full Octave, Direct Averaging, FDF)
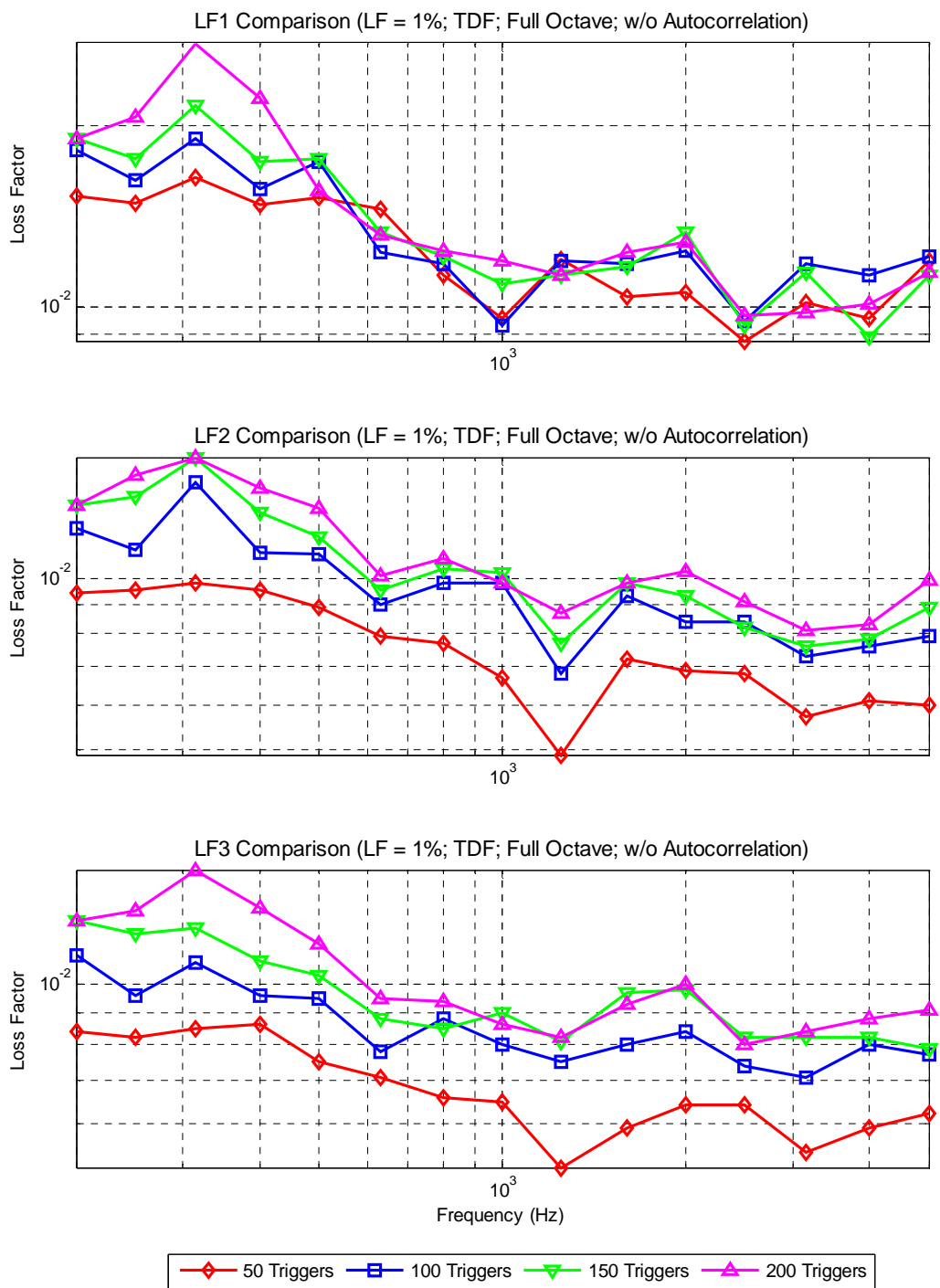
Figure A. 6: Computational Model (LF = 0.01, Full Octave, Direct Averaging, TDF)
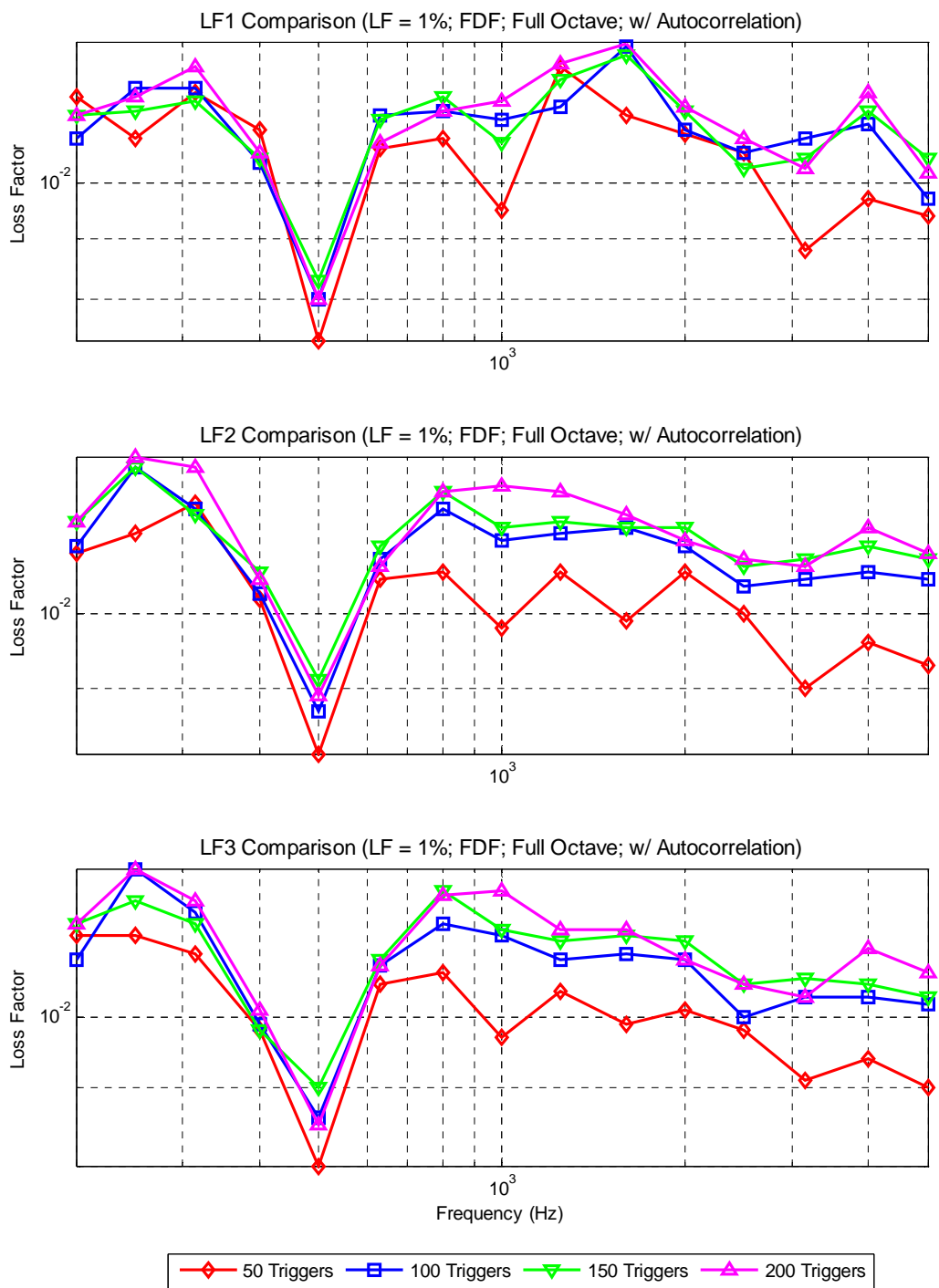
Figure A. 7: Computational Model (LF = 0.01, Full Octave, Averaging Autocorrelation, FDF)
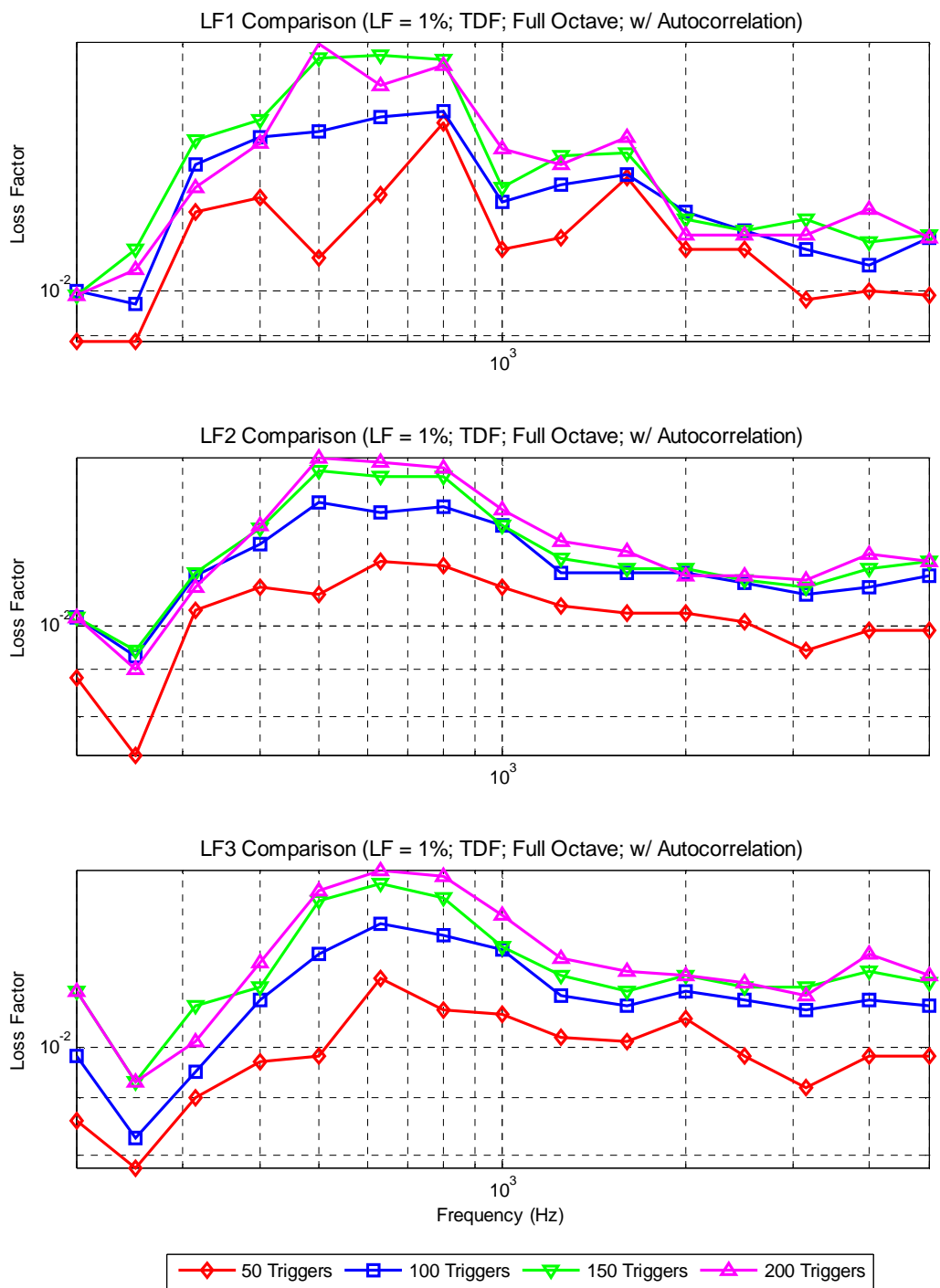
Figure A. 8: Computational Model (LF = 0.01, Full Octave, Averaging Autocorrelation, TDF)
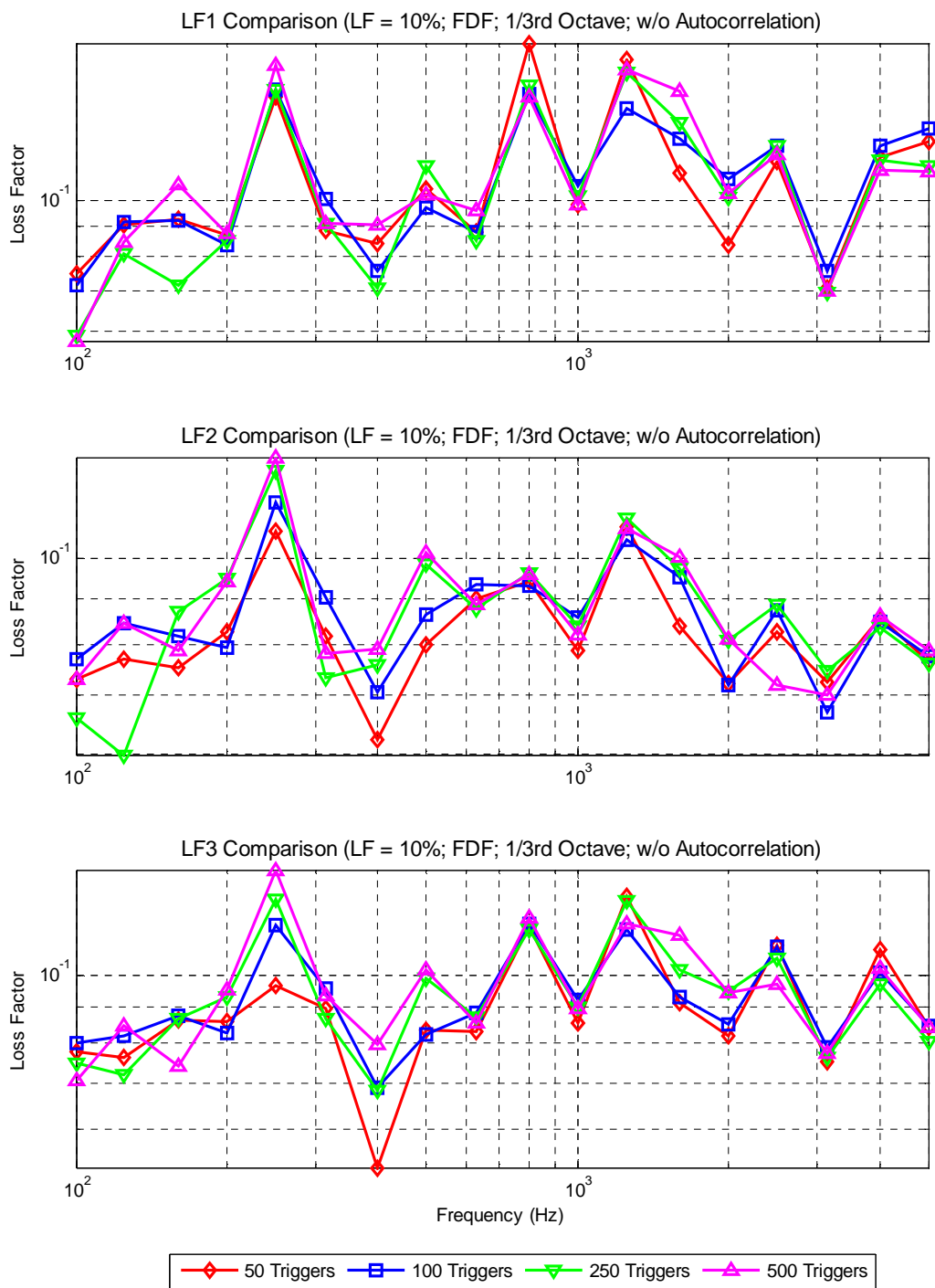
Figure A. 9: Computational Model (LF = 0.1, 1/3rd Octave, Direct Averaging, FDF)
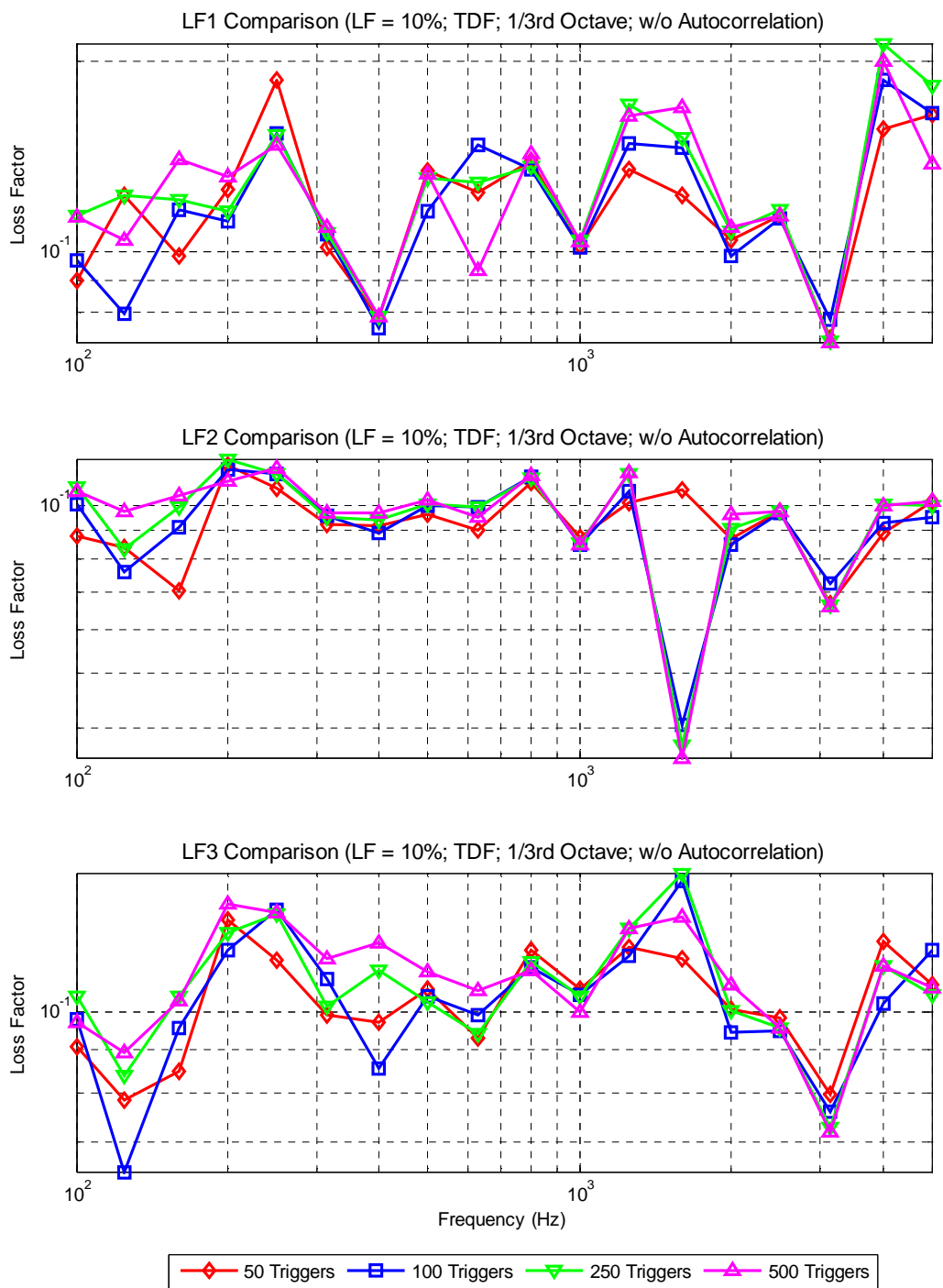
Figure A. 10: Computational Model (LF = 0.1, 1/3rd Octave, Direct Averaging, TDF)
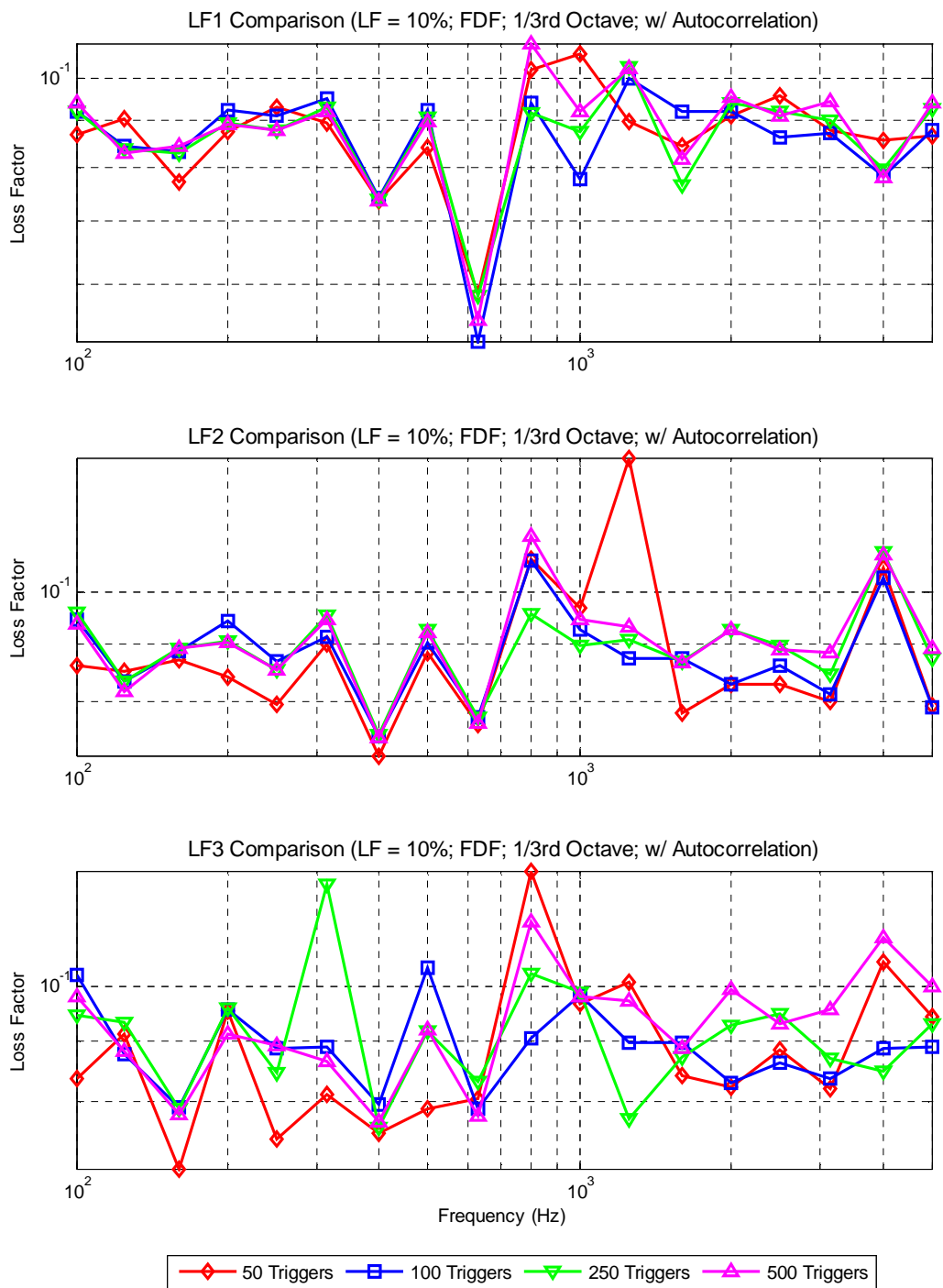
Figure A. 11: Computational Model (LF = 0.1, 1/3rd Octave, Averaging Autocorrelation, FDF)
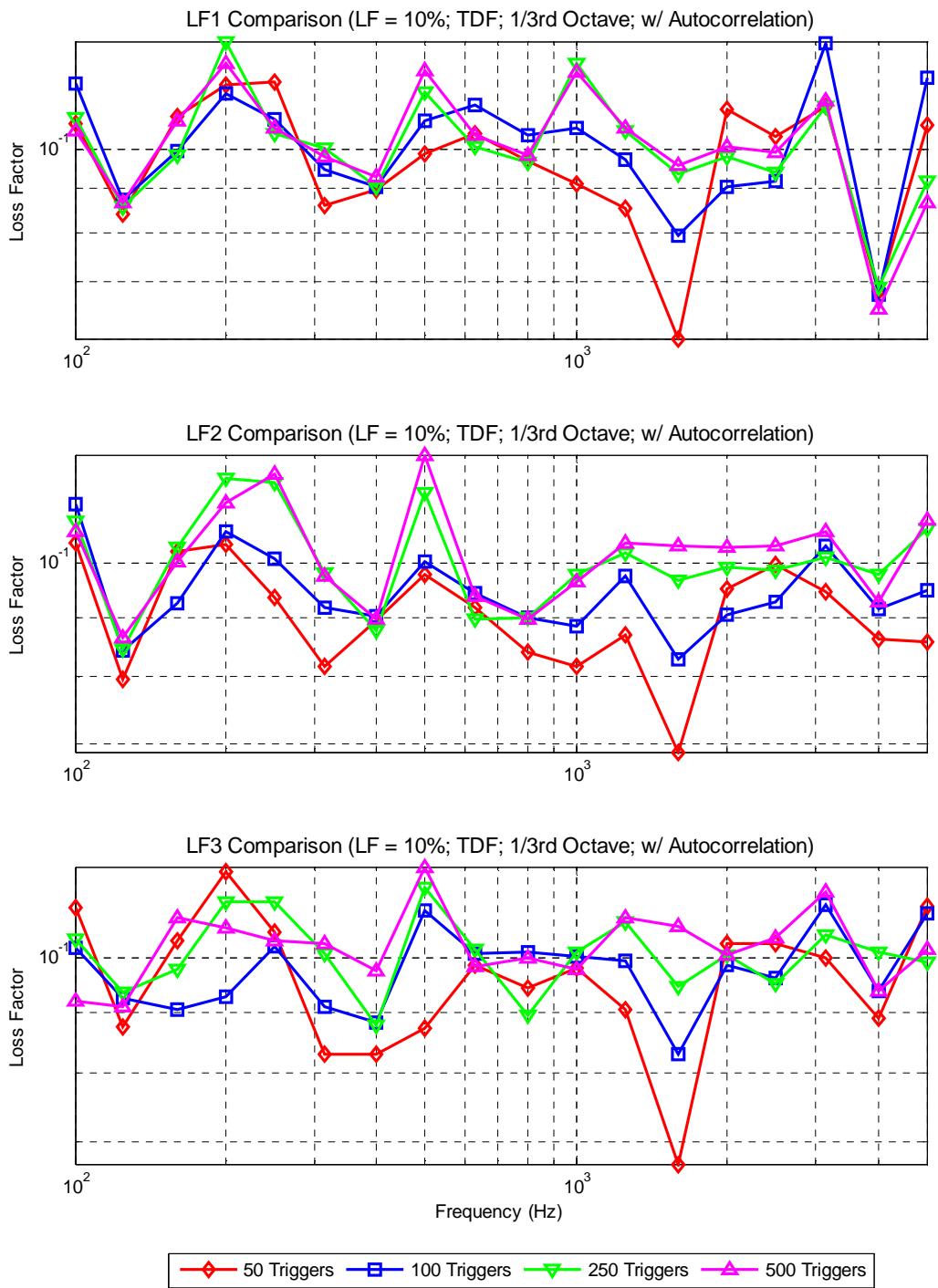
Figure A. 12: Computational Model (LF = 0.1, 1/3rd Octave, Averaging Autocorrelation, TDF)
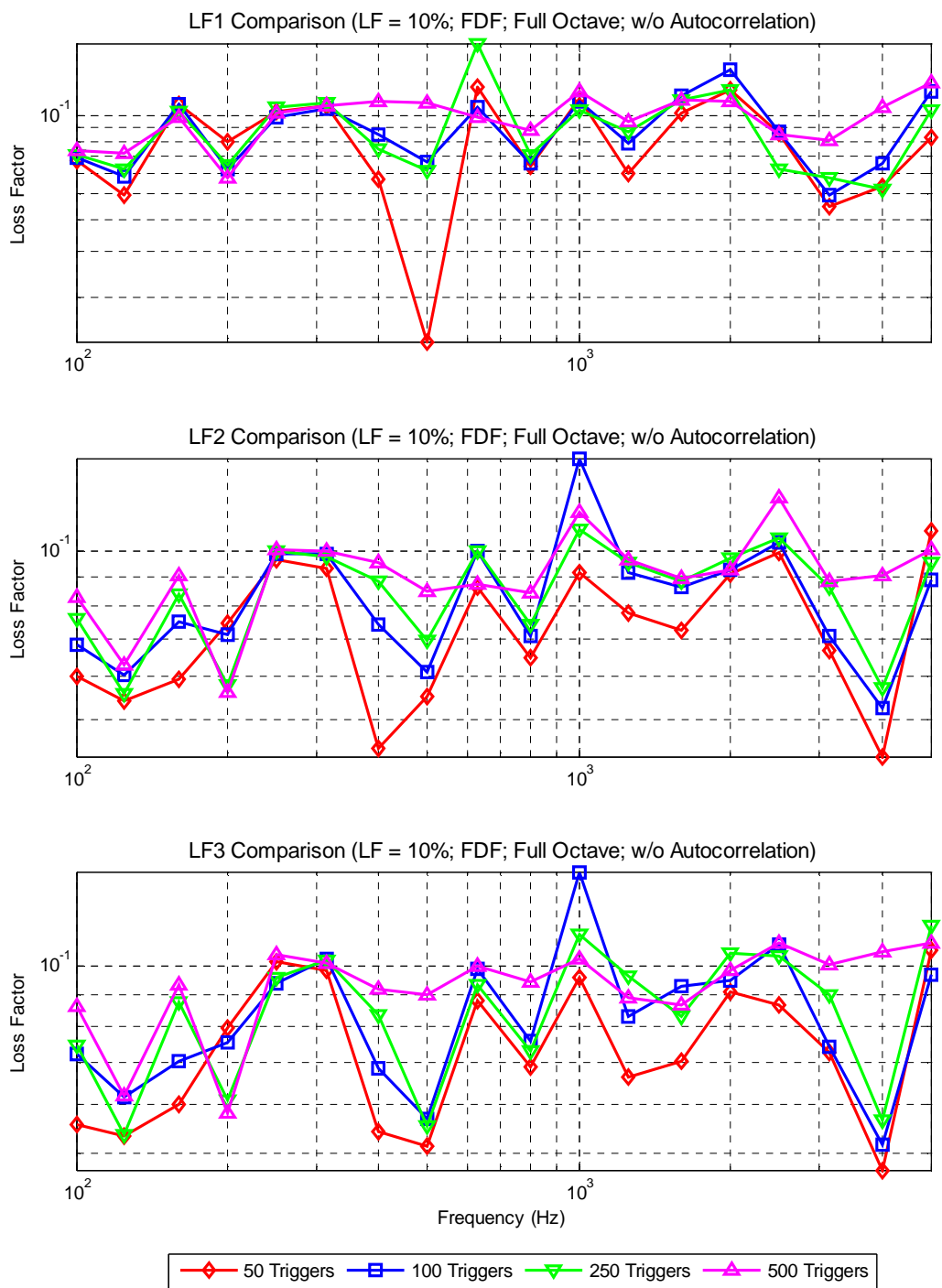
Figure A. 13: Computational Model (LF = 0.1, Full Octave, Direct Averaging, FDF)

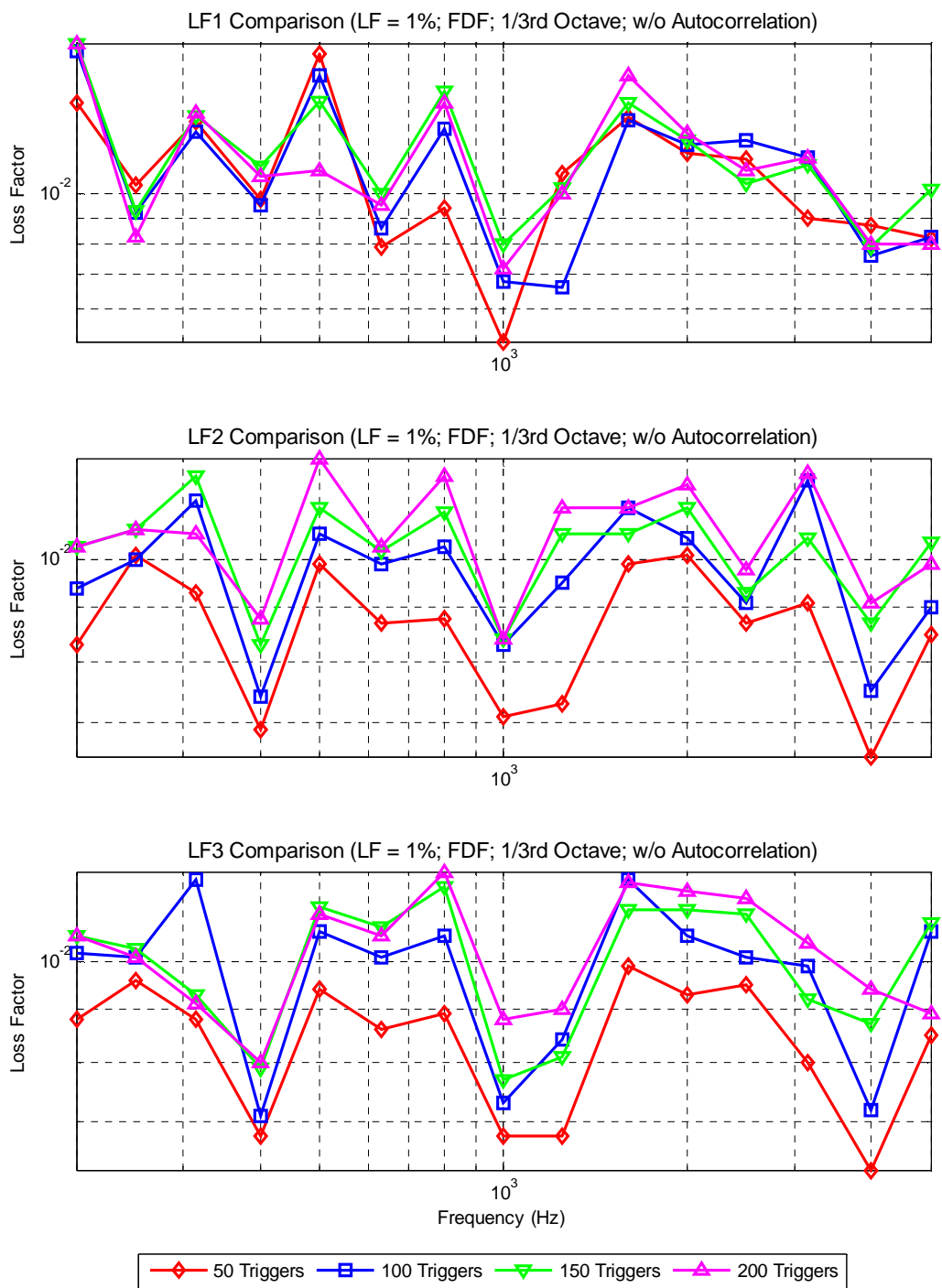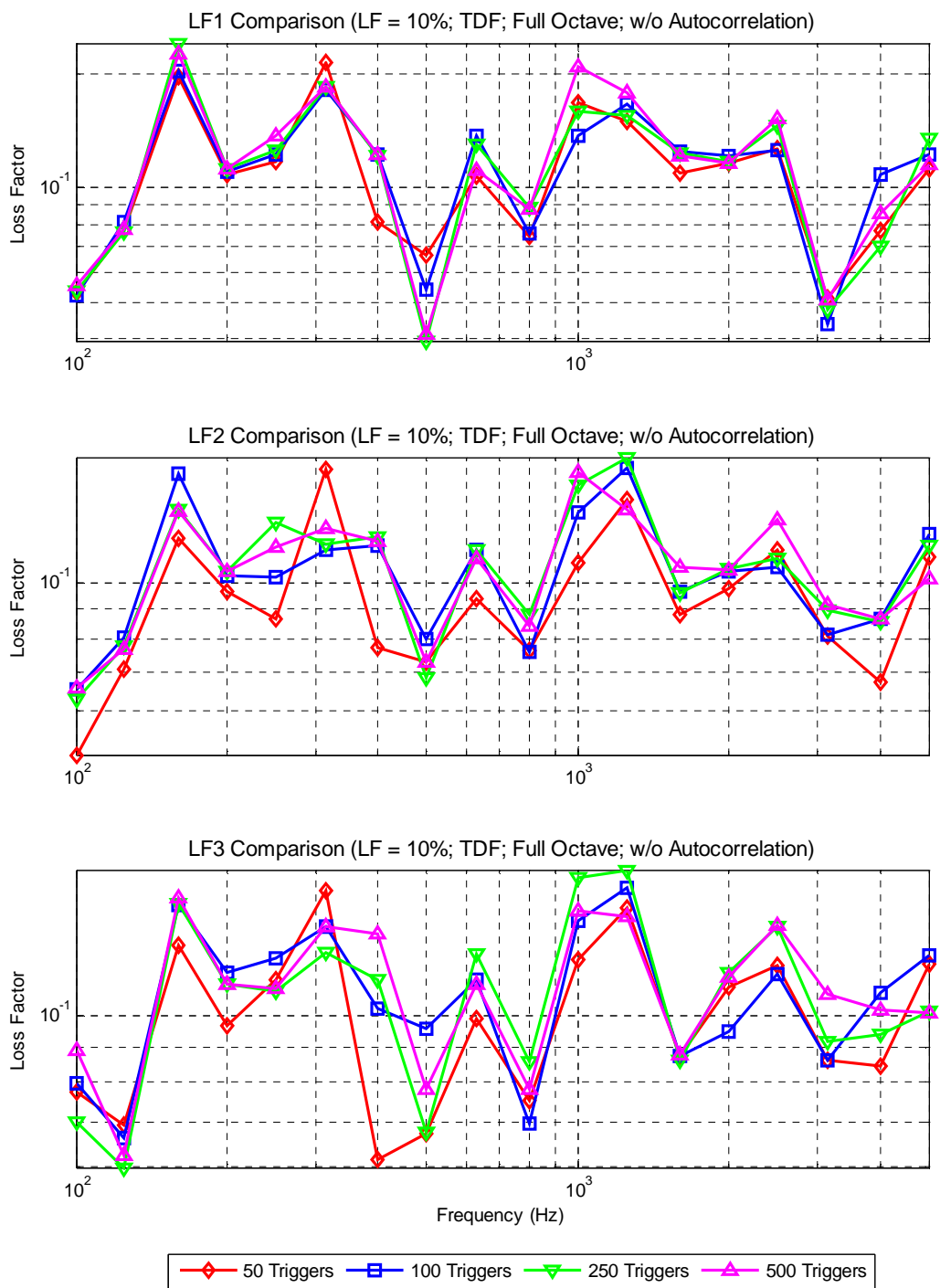**Figure A. 14: Computational Model (LF = 0.1, Full Octave, Direct Averaging, TDF)**
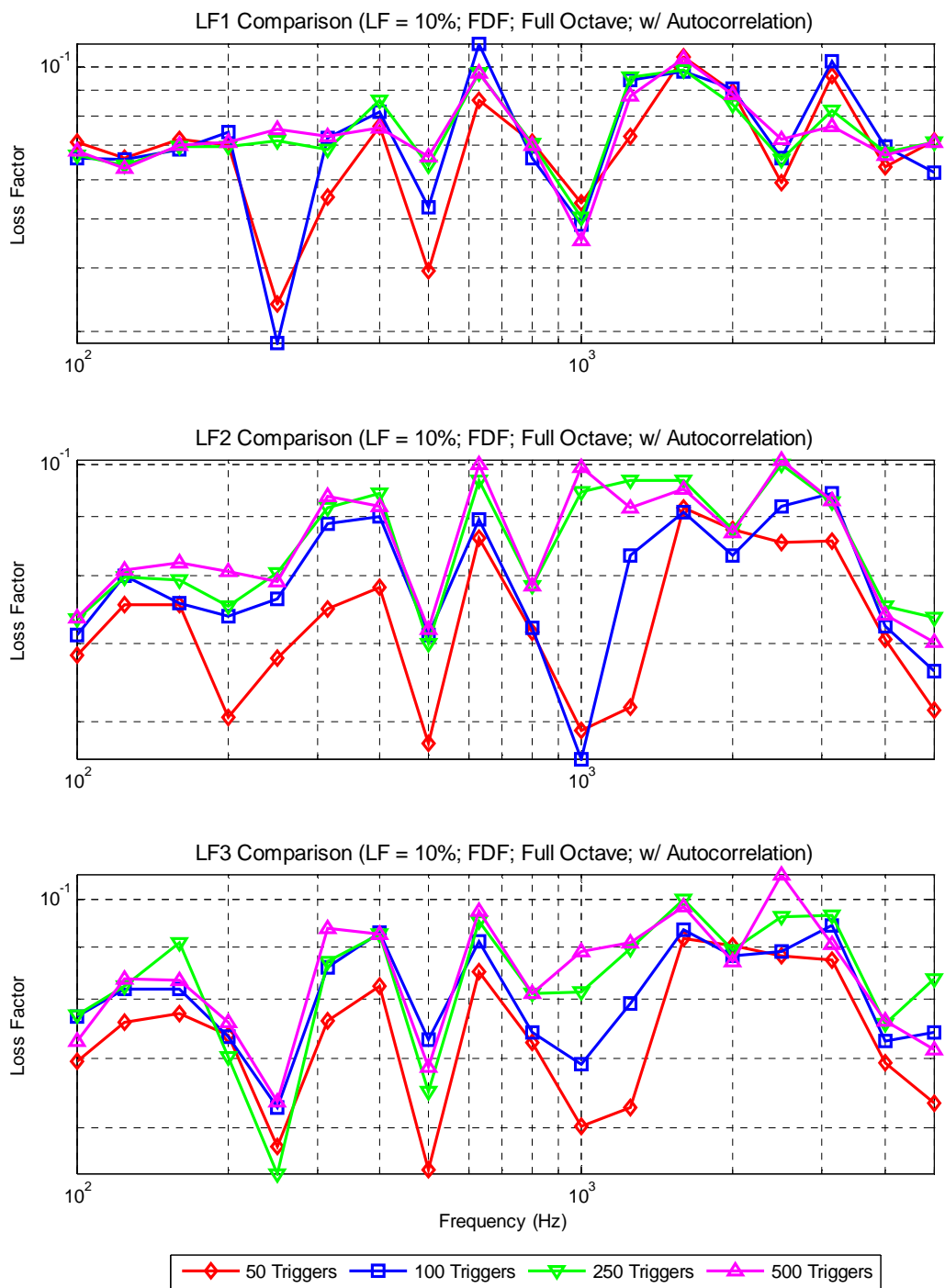
Figure A. 15: Computational Model (LF = 0.1, Full Octave, Averaging Autocorrelation, FDF)
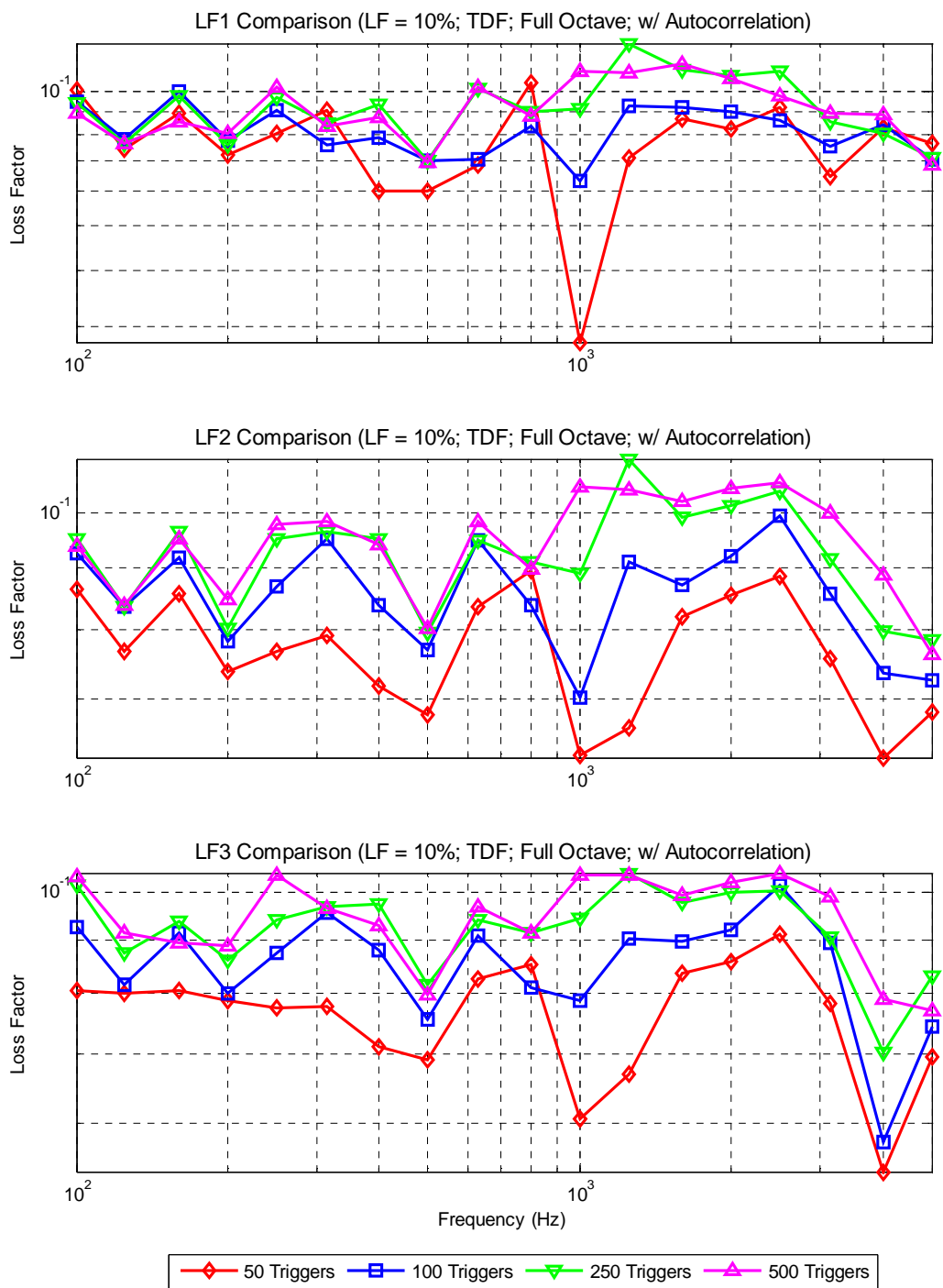
Figure A. 16: Computational Model (LF = 0.1, Full Octave, Averaging Autocorrelation, TDF)
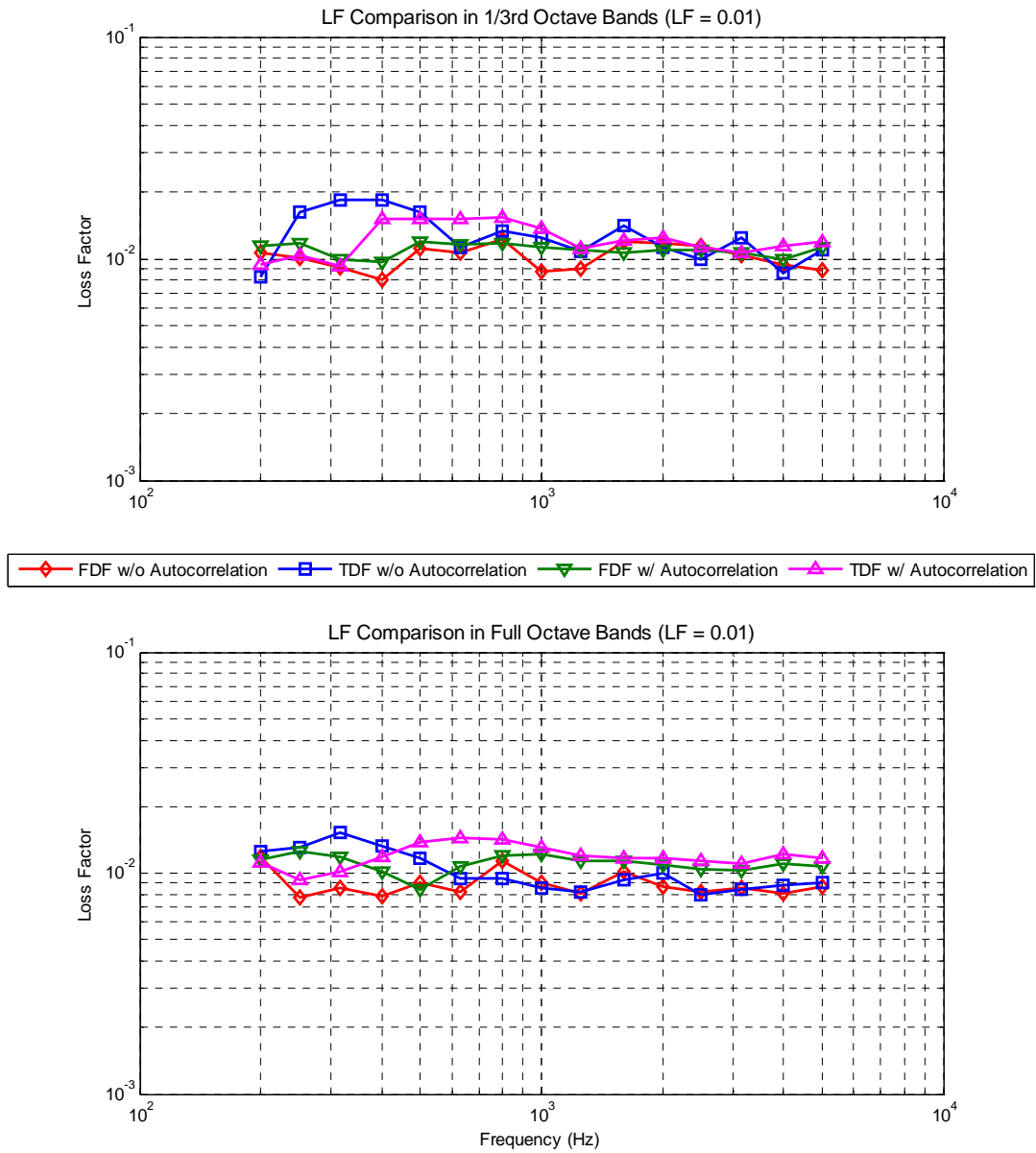
Figure A. 17: Computational Plate (LF=0.01) Summary – Loss Factor Comparison – Full or 1/3rd Octave Filter Bands
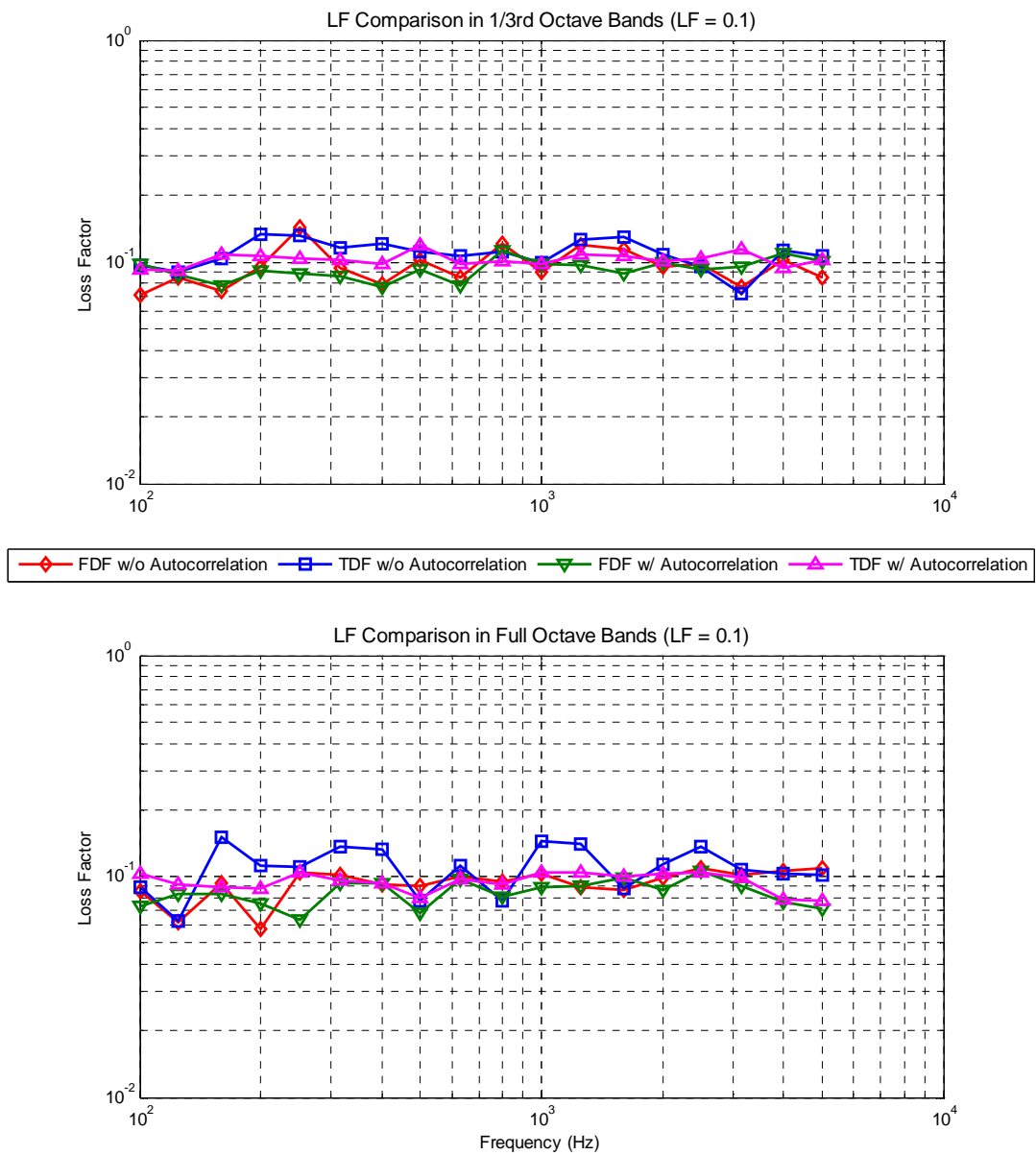
Figure A. 18: Computational Plate (LF=0.1) Summary – Loss Factor Comparison – Full or 1/3rd Octave Filter Bands
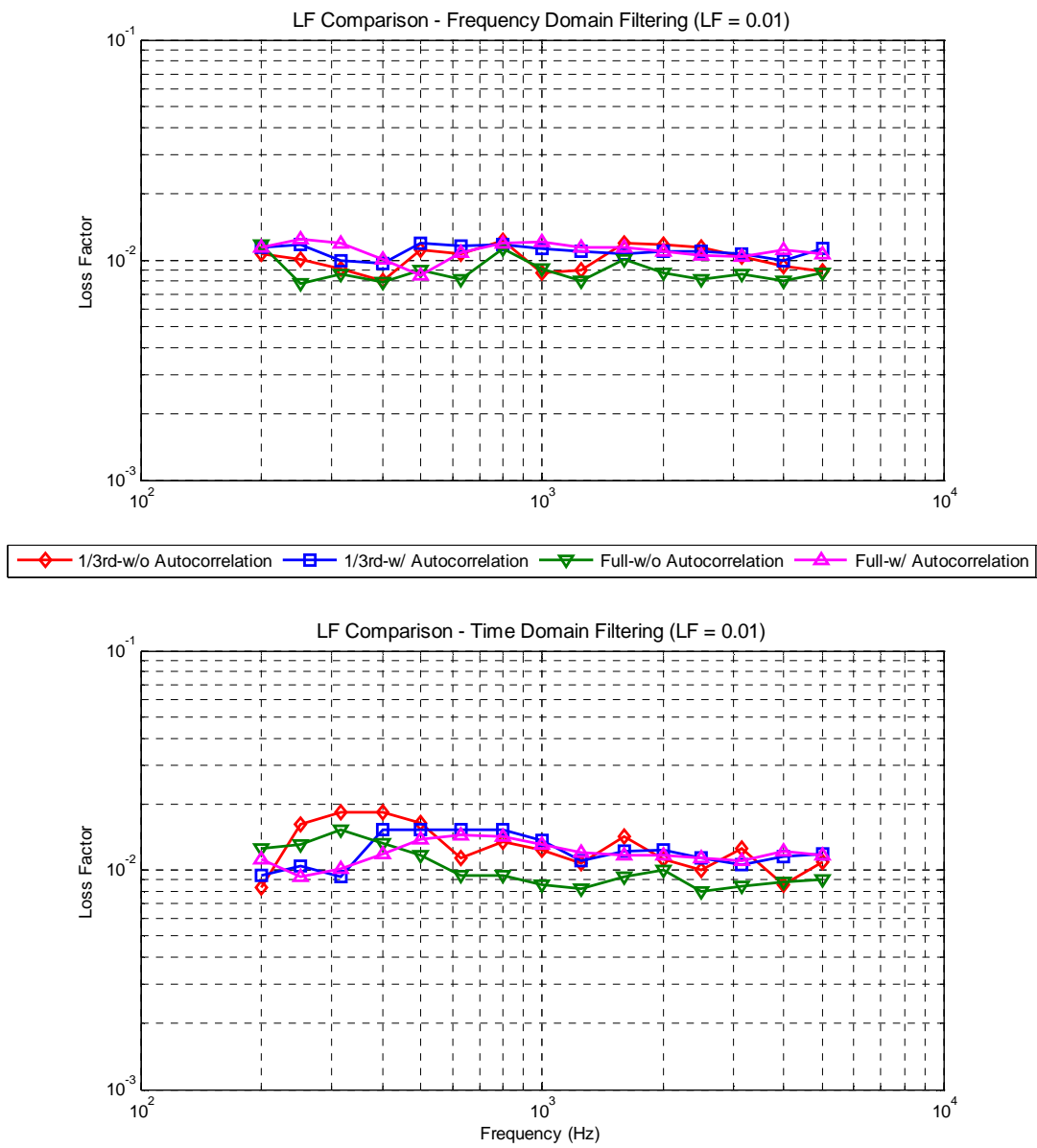
Figure A. 19: Computational Plate (LF=0.01) Summary – Loss Factor Comparison – TDF and FDF
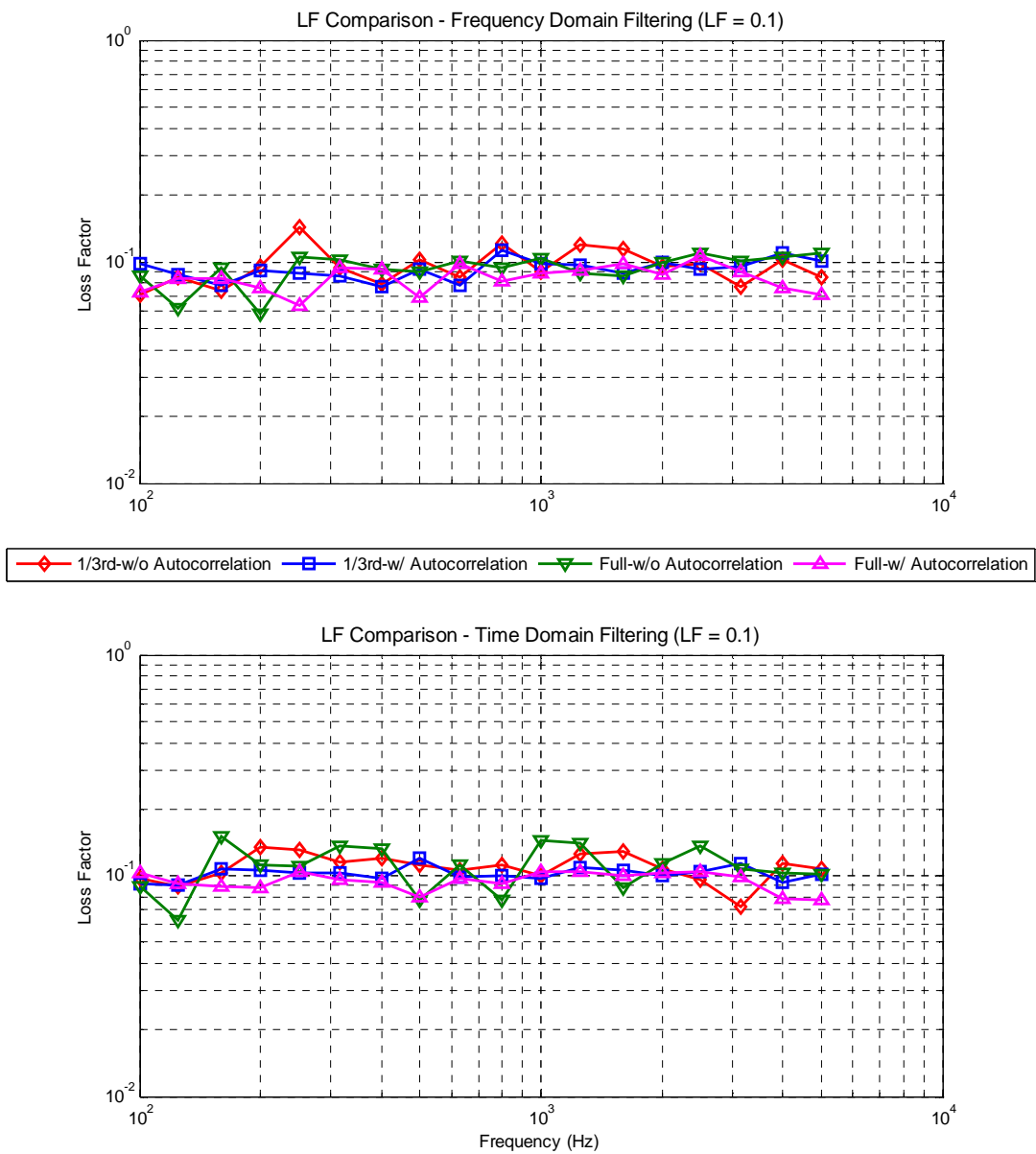
Figure A. 20: Computational Plate (LF=0.1) Summary – Loss Factor Comparison – TDF and FDF
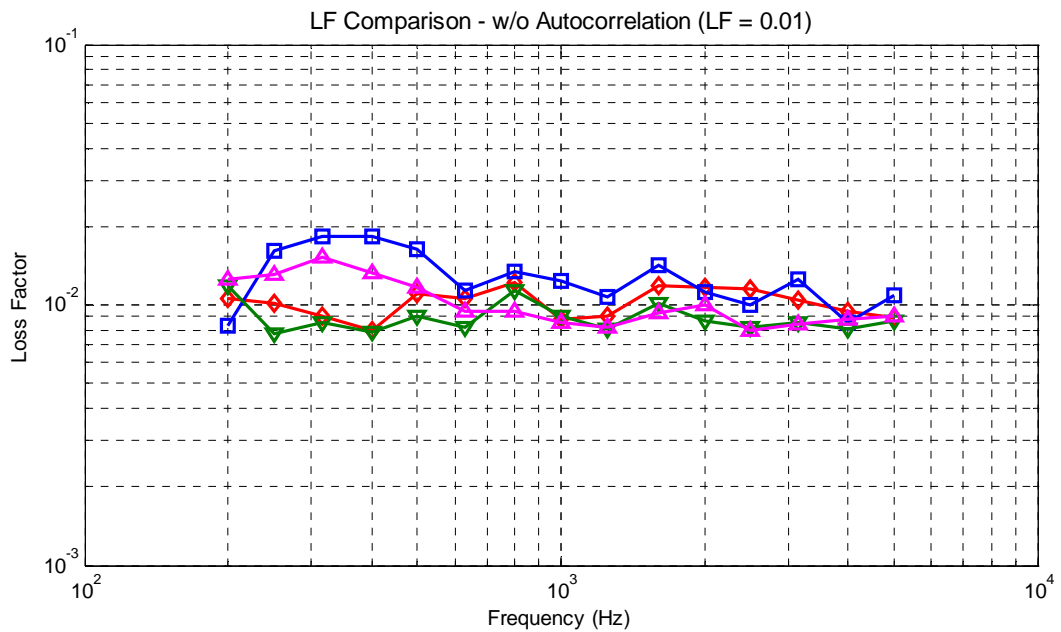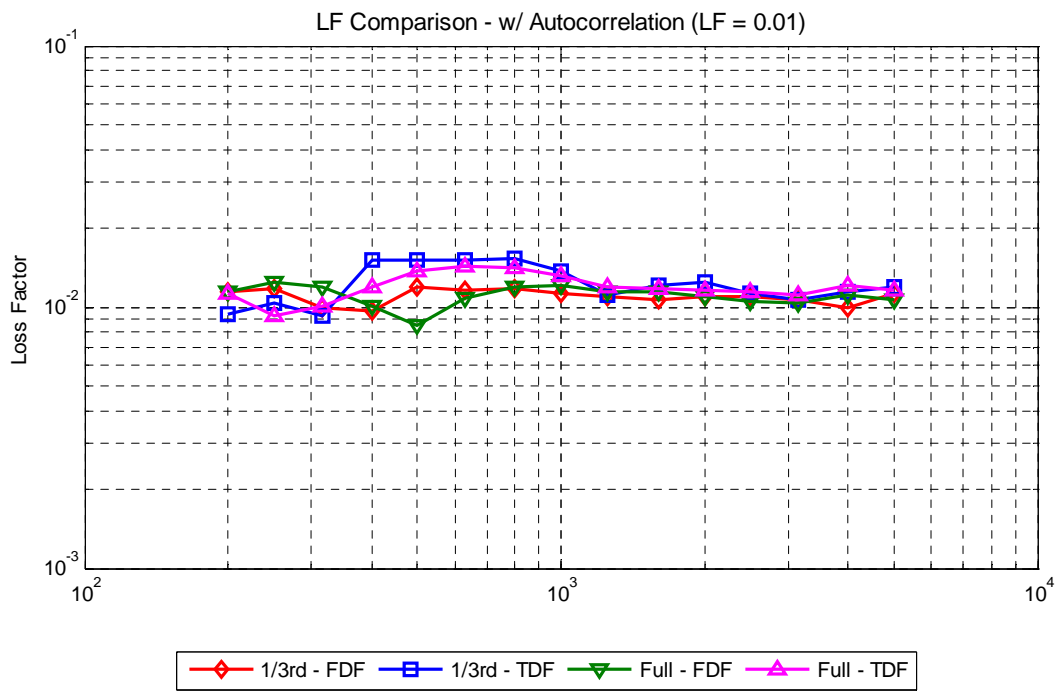
Figure A. 21: Computational Plate (LF=0.01) Summary – Loss Factor Comparison – Direct Averaging or Averaging Autocorrelation
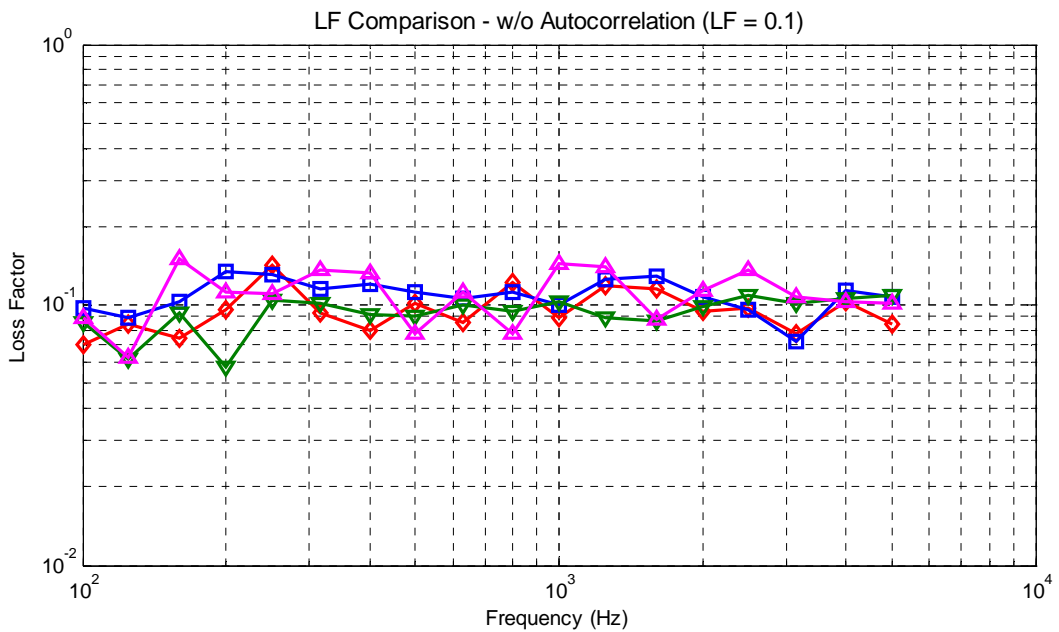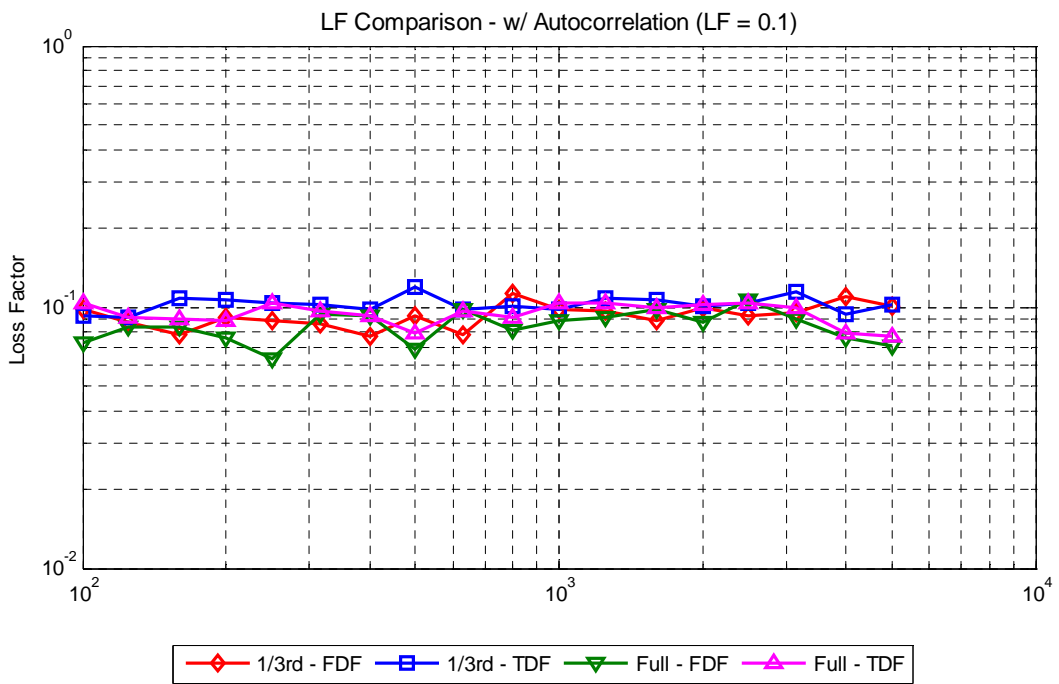
Figure A. 22: Computational Plate (LF=0.1) Summary – Loss Factor Comparison – Direct Averaging or Averaging Autocorrelation

A.1 Summary of Computational Plate Model – Tabular Form

		Simulated Loss Factor = 0.1			
		FDF		TDF	
		1/3rd	Full	1/3rd	Full
LF1	Direct Averaging	0.0882	0.0745	0.1021	0.0915
	Averaging Autocorrelation	0.1095	0.0991	0.1258	0.1212
LF2	Direct Averaging	0.0906	0.0858	0.1022	0.0947
	Averaging Autocorrelation	0.0880	0.0915	0.0963	0.1065
LF3	Direct Averaging	0.0930	0.0846	0.1025	0.0946
	Averaging Autocorrelation	0.0961	0.0936	0.1095	0.1107

Table 5: Averaged loss factors over entire frequency range – MDOF Computational plate model with simulated LF=0.1

		Simulated Loss Factor = 0.01			
		FDF		TDF	
		1/3rd	Full	1/3rd	Full
LF1	Direct Averaging	0.0113	0.0113	0.0138	0.0131
	Averaging Autocorrelation	0.0117	0.0102	0.0176	0.0147
LF2	Direct Averaging	0.0111	0.0112	0.0126	0.0122
	Averaging Autocorrelation	0.0106	0.0095	0.0135	0.0112
LF3	Direct Averaging	0.0110	0.0110	0.0123	0.0120
	Averaging Autocorrelation	0.0102	0.0089	0.0129	0.0103

Table 6: Averaged loss factors over entire frequency range – MDOF Computational plate model with simulated LF=0.01

		Simulated Loss Factor = 0.1			
		FDF		TDF	
		1/3rd	Full	1/3rd	Full
LF1	Direct Averaging	11.8	25.5	2.1	8.5
	Averaging Autocorrelation	9.5	0.9	25.8	21.2
LF2	Direct Averaging	9.4	14.2	2.2	5.3
	Averaging Autocorrelation	12.0	8.5	3.7	6.5
LF3	Direct Averaging	7.0	15.4	2.5	5.4
	Averaging Autocorrelation	3.9	6.4	9.5	10.7

Table 7: Deviation of averaged loss factor from simulated loss factor (%) – MDOF Computational Plate Model with Simulated LF=0.1

		Simulated Loss Factor = 0.01			
		FDF		TDF	
		1/3rd	Full	1/3rd	Full
LF1	Direct Averaging	13.0	13.0	38.0	31.0
	Averaging Autocorrelation	17.0	2.0	76.0	47.0
LF2	Direct Averaging	11.0	12.0	26.0	22.0
	Averaging Autocorrelation	6.0	5.0	35.0	12.0
LF3	Direct Averaging	10.0	10.0	23.0	20.0
	Averaging Autocorrelation	2.0	11.0	29.0	3.0

Table 8: Deviation of averaged loss factor from simulated loss factor (%) – MDOF Computational Plate Model with Simulated LF=0.01

Appendix B: MATLAB Codes

B.1 Random Force Generation

```matlab
%%% Random Force Generator Code
clc
clear all
close all

Fms=1; % Force Amplitude
fs=20000; % Sampling Frequency
npts=fs*10.5; % Number of Points, 10.5 is number of seconds of data
deltaf=fs/npts; % Frequency Increment
f_low=64; % Lowest Frequecy
f_high=8000; % Highest Freqeuency
band=f_high-f_low; % Force Band
f_cutoff_low=round(f_low/deltaf)+1;
f_cutoff_high=round(f_high/deltaf)+1;
So=Fms/band;
P=zeros(npts/2+1,1);
P(f_cutoff_low:f_cutoff_high,1)=So*ones(f_cutoff_high-
f_cutoff_low+1,1);
N=length(P);
level=P/2;
level=npts*(npts*deltaf)*level;
level=sqrt(level);
phase=2*pi*rand(N,1);
Fw=level.*(cos(phase)+i*sin(phase));
Fw(N+1:2*(N-1))=conj(flipud(Fw(2:N-1)));
f=real(ifft(Fw)); % Random Force
force=f;
%%% Saving the generated force
save force_10pt5_Seconds_fs_20000_set_1.mat force
```

B.2 NASTRAN FRFs to Full FRFs

```matlab
%%% For converting NASTRAN FRFs to Full FRFs
clc
clear all
close all

for quad=1:4
    for position=1:4
        for input=1:4
            eval(['load
input_',num2str(input),'_quadrant_',num2str(quad),'_',num2str(positi
on),';'])
            half_FRF=zeros(1,10000);

eval(['raw_FRF=input_',num2str(input),'_quadrant_',num2str(quad),'_'
,num2str(position),'(2:10001);'])
            conj_FRF=conj(raw_FRF);
            other_half_raw_FRF=fliplr(conj_FRF);
            FRF=[raw_FRF other_half_raw_FRF];
            eval(['save
FRF_input_',num2str(input),'_quad_',num2str(quad),'_position_',num2s
tr(position),'.mat FRF;'])
            clear half_FRF raw_FRF conj_FRF other_half_raw_FRF FRF
        end
    end
end
clear

for quad=1:4
    for position=1:4
        avgFRF=zeros(1,20000);
        for input=1:4
            eval(['load
FRF_input_',num2str(input),'_quad_',num2str(quad),'_position_',num2s
tr(position),'.mat;'])
            avgFRF=avgFRF+FRF/4;
            clear FRF
        end
        position_number=position+4*(quad-1)
        FRF=avgFRF;
        eval(['save
10perc_LF_FRF_position_',num2str(position_number),'.mat FRF'])
        clear avgFRF FRF position_number
    end
end
clear
```

B.3 Filtered Response Generation – Time Domain Filtering

```matlab
%%% Filtered Response Generation
%%% Step1 IR generation
%%% Step2 Convolution of IR and Random Force
%%% Step3 Full Octave Filtering
%%% Step4 One-Third Octave Filtering
clc
clear all
close all

%%% Converting the FRF to impulse responses, 16 are total o/p
locations
for i=1:16
    eval(['load 10perc_LF_FRF_position_',num2str(i),'.mat'])
    IR_long=real(ifft(FRF));
    IR=IR_long(1:2000);
    figure(i)
    time=1/20000:1/20000:.1;
    plot(time,IR)
    eval(['save 10perc_LF_IR_position_',num2str(i),'.mat IR'])
    clear
end
clear
%%% Generating random force and impulse responses to get random
force
for i=1:16
    for ii=1:25
        eval(['load 1perc_LF_IR_position_',num2str(i),'.mat'])
        eval(['load
force_10pt5_Seconds_fs_20000_set_',num2str(ii),'.mat'])

eval(['non_refined_convolved_response_LF_1perc_fs_20000_set_',num2st
r(ii),'=conv(IR,force);'])

eval(['response_LF_1perc_fs_20000_set_',num2str(ii),'=non_refined_co
nvolved_response_LF_1perc_fs_20000_set_',num2str(ii),'(1:10.5*20000)
;'])

eval(['response=response_LF_1perc_fs_20000_set_',num2str(ii),';'])
        eval(['save
response_LF_1perc_fs_20000_position_',num2str(i),'_set_',num2str(ii)
,'.mat response;'])
        eval(['clear response IR force
response_LF_1perc_fs_20000_set_',num2str(ii),'
non_refined_convolved_response_LF_1perc_fs_20000_set_',num2str(ii)])
        data=[i ii]
    end
end
clear
%%% Full Octave Filters are created using FDATool, stored and
recalled
%%% Full octave filtering
```

```matlab
load Full_Octave_Filters.mat
for i=1:16
    for ii=1:25
        fc=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600
2000 2500 3150 4000 5000];
        for iii=1:length(fc)
            eval(['load
response_LF_1perc_fs_20000_position_',num2str(i),'_set_',num2str(ii)
,'.mat'])
            frequency=fc(iii);

eval(['filtered_response=filter(filter',num2str(frequency),',1,respo
nse);'])
            eval(['save
full_octave_filtered_',num2str(frequency),'Hz_response_LF_1perc_fs_2
0000_position_',num2str(i),'_set_',num2str(ii),'.mat
filtered_response'])
            data=[i ii frequency]
            clear filtered_response response frequency
        end
    end
end
clear
%%% ONe-Third Octave Filters are created using FDATool, stored and
recalled
%%% One-Third octave filtering
load One_Third_Octave_Filters.mat
for i=1:16
    for ii=1:25
        fc=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600
2000 2500 3150 4000 5000 6300];
        for iii=1:length(fc)
            eval(['load
response_LF_1perc_fs_20000_position_',num2str(i),'_set_',num2str(ii)
,'.mat'])
            fc=[100 125 160 200 250 315 400 500 630 800 1000 1250
1600 2000 2500 3150 4000 5000 6300];
            frequency=fc(iii);

eval(['filtered_response=filter(filter',num2str(frequency),',1,respo
nse);'])
            eval(['save
one_third_octave_filtered_',num2str(frequency),'Hz_response_LF_1perc
_fs_20000_position_',num2str(i),'_set_',num2str(ii),'.mat
filtered_response'])
            data=[i ii frequency]
            clear fc filtered_response response frequency
        end
    end
end
clear
```

B.4 Filtered Response Generation – Frequency Domain Filtering

```matlab
%%% Filtered Response Generation
%%% Step1 Full FRF to Chunked FRF
%%% Step2 IR from Chunked FRF
%%% Step3 Convolution of IR and Random Force
%%% Step4 Full Octave Chunk Filtering
%%% Step5 One-Third Octave Chunk Filtering
clc
clear all
close all

%%% Full octave CHUNK filtering
for i=1:16
    for ii=1:25
        fc=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600
2000 2500 3150 4000 5000];
        for iii=1:length(fc)
            frequency=fc(iii)
            eval(['load 10perc_LF_FRF_position_',num2str(i),'.mat'])
            startfrequency=round((1/sqrt(2))*frequency);
            endfrequency=round((1*sqrt(2))*frequency);
            FRF_for_IR=[zeros(1,startfrequency-1)
FRF(startfrequency:endfrequency) zeros(1,10000-endfrequency)
zeros(1,10000-endfrequency) FRF(20000-endfrequency:20000-
startfrequency) zeros(1,startfrequency-1)];
            IR=real(ifft(FRF_for_IR));
            eval(['load
force_10pt5_Seconds_fs_20000_set_',num2str(ii),'.mat'])

eval(['non_refined_convolved_response_LF_10perc_fs_20000_set_',num2s
tr(ii),'=conv(IR,force);'])
            % using only first 10.5 seconds of response

eval(['response_LF_10perc_fs_20000_set_',num2str(ii),'=non_refined_c
onvolved_response_LF_10perc_fs_20000_set_',num2str(ii),'(1:10.5*2000
0);'])

eval(['filtered_response=response_LF_10perc_fs_20000_set_',num2str(i
i),';'])
            eval(['save
full_octave_chunk_filtered_',num2str(frequency),'Hz_response_LF_10pe
rc_fs_20000_position_',num2str(i),'_set_',num2str(ii),'.mat
filtered_response'])
            i
            ii
            eval(['clear filtered_response
response_LF_10perc_fs_20000_set_',num2str(ii),'
non_refined_convolved_response_LF_10perc_fs_20000_set_',num2str(ii),
' force IR startfrequency endfrequency FRF_for_IR'])
        end
    end
end
```

```matlab
clear
%%% One-Third octave CHUNK filtering
for i=1:16
    for ii=1:25
        fc=[100 125 160 200 250 315 400 500 630 800 1000 1250 1600
2000 2500 3150 4000 5000 6300];
        for iii=1:length(fc)
            frequency=fc(iii)
            eval(['load 10perc_LF_FRF_position_',num2str(i),'.mat'])
            startfrequency=round((1/(2^(1/6)))*frequency);
            endfrequency=round((1*(2^(1/6)))*frequency);
            FRF_for_IR=[zeros(1,startfrequency-1)
FRF(startfrequency:endfrequency) zeros(1,10000-endfrequency)
zeros(1,10000-endfrequency) FRF(20000-endfrequency:20000-
startfrequency) zeros(1,startfrequency-1)];
            IR=real(ifft(FRF_for_IR));
            eval(['load
force_10pt5_Seconds_fs_20000_set_',num2str(ii),'.mat'])

eval(['non_refined_convolved_response_LF_10perc_fs_20000_set_',num2s
tr(ii),'=conv(IR,force);'])
            % using only first 10.5 seconds of response

eval(['response_LF_10perc_fs_20000_set_',num2str(ii),'=non_refined_c
onvolved_response_LF_10perc_fs_20000_set_',num2str(ii),'(1:10.5*2000
0);'])

eval(['filtered_response=response_LF_10perc_fs_20000_set_',num2str(i
i),';'])
            eval(['save
one_third_octave_chunk_filtered_',num2str(frequency),'Hz_response_LF
_10perc_fs_20000_position_',num2str(i),'_set_',num2str(ii),'.mat
filtered_response'])
            i
            ii
            eval(['clear filtered_response
response_LF_10perc_fs_20000_set_',num2str(ii),'
non_refined_convolved_response_LF_10perc_fs_20000_set_',num2str(ii),
' force IR startfrequency endfrequency FRF_for_IR'])
        end
    end
end
clear
```

B.5 Triggered Response Generation

```
%%% Triggered Data Generation
%%% This code generates triggered response from the filtered, time
domain
%%% or frequency domain. jjj is total number of triggers
clc
clear all
close all

    for position=1:16
        position_123=position
        for freq=1:19
            fc=[100 125 160 200 250 315 400 500 630 800 1000 1250
1600 2000 2500 3150 4000 5000 6300];
            frequency=fc(freq)
            fs=20000; %%% Sampling Frequency
            cycles=222; %%% Length of sample

            ncorr=round(cycles*fs/frequency);
            trigger_level_factor=2; %%% Trigger Level
            spread=1.2; %%% Trigger Band
            overlap_factor=0; %%% Overlap in consequitive triggered
samples

            setindex=1;
            eval(['load
one_third_octave_filtered_',num2str(frequency),'Hz_response_LF_1perc
_fs_20000_position_',num2str(position),'_set_',num2str(setindex),'.m
at filtered_response'])
            respc=filtered_response;
            clear filtered_response
            trigger_level=std(respc)*trigger_level_factor;

            i=2;
            j=0;
            while i<(length(respc)-ncorr+2)
                  if respc(i)>=trigger_level &&
respc(i)<trigger_level*spread && respc(i-1)<trigger_level
                      trig1=i;
                      while respc(i)>=trigger_level &&
respc(i)<trigger_level*spread && i<(length(respc)-ncorr+2)
                          i=i+1;
                          if respc(i+1)<trigger_level
                              trig2=i;
                              [aa bb]=max(respc(trig1:trig2));
                              if respc(trig1+bb-
1)<trigger_level*spread && respc(trig1+bb-1)>trigger_level &&
(trig1+bb-1)~=0
                                      j=j+1;
                                      triggerindex(j)=trig1+bb-1;
                                      triggerdata(:,j)=respc(trig1+bb-
1:trig1+bb-1+ncorr-1);
```

```matlab
                                        end
                                            i=i+round((1-overlap_factor)*ncorr);
% for uncorrelated
                                            break
                                    end
                                end
                            else
                                i=i+1;
                            end
                            pqr=j;

                            if i>=(length(respc)-ncorr)
                                i=2;
                                setindex=setindex+1;
                                if setindex>25
                                    break
                                end
                                clear filtered_response respc
                                eval(['load
one_third_octave_filtered_',num2str(frequency),'Hz_response_LF_1perc
_fs_20000_position_',num2str(position),'_set_',num2str(setindex),'.m
at filtered_response'])
                                respc=filtered_response;
                                clear filtered_response
                            end

                            if j>1000
                                break
                            end
                        end
                        setindex=setindex-1;
                        eval(['save
triggered_one_third_octave_filtered_',num2str(frequency),'Hz_respons
e_LF_1perc_fs_20000_position_',num2str(position),'.mat triggerdata
j'])
                        j
                        clear i j aa bb trig1 trig2 respc ncorr setindex pqr
frequency trigger_level triggerindex triggerdata frequency
                    end
                    clear position
                end
```

B.6 Loss Factor Estimation Code – Averaging Autocorrelation

```matlab
%%% Loss Factor Comparison - With Autocorrelation
%%% For this code the triggered samples are loaded
clc
clear all
close all

for iiii=1:4
    for freq=1:18
        accelnum=16
        for position=1:accelnum
            fc=[100 125 160 200 250 315 400 500 630 800 1000 1250
1600 2000 2500 3150 4000 5000 6300];
            frequency=fc(freq);
            eval(['load
triggered_one_third_octave_filtered_',num2str(frequency),'Hz_respons
e_LF_10perc_fs_20000_position_',num2str(position),'.mat'])
            trignumbers=[50 100 250 500];
            jjj=trignumbers(iiii)

            avgdata1=mean(transpose(triggerdata(:,1:jjj)));
            avg_xcorr_response=zeros(2*length(avgdata1)-1,jjj);
            for ii=1:jjj
                data=triggerdata(:,ii);
                avg_xcorr_response(:,ii)=xcorr(data);
            end
            avgdata2=mean(transpose(avg_xcorr_response(:,1:jjj)));
            avgdata=avgdata2(length(avgdata1):2*length(avgdata1)-1);

            if position==1
                avg_response=zeros(1,length(avgdata));
                avg_hilbert=zeros(1,length(avgdata));
                avg_logdecay=zeros(1,length(avgdata));
            end


avg_response=avg_response+(avgdata/max(avgdata))/accelnum;

avg_hilbert=avg_hilbert+abs(hilbert(avgdata/max(avgdata)))/accelnum;

avg_logdecay=avg_logdecay+20*log10(abs(hilbert(avgdata/max(avgdata))
))/accelnum;
            position

            clear triggerdata
        end

        dB_decay_avg_response=20*log10(abs(hilbert(avg_response)));
        dB_decay_avg_hilbert=20*log10(abs(avg_hilbert));
        dB_decay_avg_logdecay=avg_logdecay;
```

```matlab
        ncorr=length(avgdata);
        dt=1/20000;
        timescale=dt:dt:ncorr*dt;

        startperc=round(0e-2*ncorr+1) %%% Start percent for slope
fitting
        endperc=round(50e-2*ncorr+1) %%% End percent for slope
fitting

        figure(frequency)
        subplot(2,2,1);
        plot(timescale,dB_decay_avg_response,'r');hold on
        plot(timescale,dB_decay_avg_hilbert);hold on
        plot(timescale,dB_decay_avg_logdecay,'g');hold on
        axis tight
        title(['DB Decay Comparison',' @ ',num2str(frequency),'
Hz'])
        legend('DB Decay from Avg Free Decay','DB Decay for Avg
Envelope of All Output Points','Avg DB Decay of All Output Points')

        %%%
        for iii=1:3
            if iii==1
                data=dB_decay_avg_response;
            elseif iii==2
                data=dB_decay_avg_hilbert;
            elseif iii==3
                data=dB_decay_avg_logdecay;
            end

            figure(frequency)
            subplot(2,2,1+iii)
            plot(timescale,data);hold on;
            axis([0 ncorr*dt min(data) max(data)]);
            ylabel('DB Decay')
            xlabel('Time')

            k=0;
            if frequency<400 %%% Between 100 & 315 HZ
                i=5;
                while i<endperc
                    if data(i)>data(i-1) && data(i)>data(i-2) &&
data(i)>data(i-3)
                        if data(i+1)<data(i) && data(i+2)<data(i) &&
data(i+3)<data(i)
                            pt=data(i);
                            pt_i=i;
                            k=k+1;
                            ptarray(k)=data(i);
                            pt_iarray(k)=i;
                        end
                    end
                    i=i+1;
```

```matlab
                end
            end

            if frequency>315 && frequency<1600 %%% Between 400 &
1250 Hz
                i=4;
                while i<endperc
                    if data(i)>data(i-1) && data(i)>data(i-2)
                        if data(i+1)<data(i) && data(i+2)<data(i)
                            pt=data(i);
                            pt_i=i;
                            k=k+1;
                            ptarray(k)=data(i);
                            pt_iarray(k)=i;
                        end
                    end
                    i=i+1;
                end
            end

            if frequency>1250 && frequency<6300 %%% Between 1600 &
5000 Hz
                i=3;
                while i<endperc
                    if data(i)>data(i-1)
                        if data(i+1)<data(i)
                            pt=data(i);
                            pt_i=i;
                            k=k+1;
                            ptarray(k)=data(i);
                            pt_iarray(k)=i;
                        end
                    end
                    i=i+1;
                end
            end
            if k>=2
                ptarray1=ptarray;
                pt_iarray1=dt*(pt_iarray);
                plot(pt_iarray1,ptarray1,'g*');hold on
                p=polyfit(pt_iarray1,ptarray1,1);
                x=0:dt:ncorr*dt;
                y=polyval(p,x);
                plot(x,y,'g');
                slope=p(1);
                lossfactor_slopefit(iii)=-slope/(27.3*frequency)
%           title(['DB Decay from Avg Free Decay
LF:',num2str(lossfactor_slopefit(iii)),' @ ',num2str(frequency),'
Hz'])
                clear pt pt_i k ptarray pt_iarray ptarray1
pt_iarray1 p x y slope data
            else
```

```matlab
                    lossfactor_slopefit(iii)=-(data(startperc)-
data(endperc))/(timescale(startperc)-
timescale(endperc))/(27.3*frequency)
%                   title(['DB Decay from Avg Free Decay
LF:',num2str(lossfactor_slopefit(iii)),' @ ',num2str(frequency),'
Hz'])
                clear pt pt_i k ptarray pt_iarray ptarray1
pt_iarray1 p x y slope data
            end
        end

eval(['loss_factor_',num2str(jjj),'(freq,:)=lossfactor_slopefit'])
    end
end
eval(['lossfactormatrix=[loss_factor_',num2str(trignumbers(1)),'
loss_factor_',num2str(trignumbers(2)),'
loss_factor_',num2str(trignumbers(3)),'
loss_factor_',num2str(trignumbers(4)),']'])

figure(12345)
loglog(fc(1:18),loss_factor(:,1),'-r*');hold on
loglog(fc(1:18),loss_factor(:,2),'-v');hold on
loglog(fc(1:18),loss_factor(:,3),'-g^'); hold on; grid on
title('Loss Factor Estimation (i/p LF=0.1, One-Third Octave & FDF)')
legend('Avg Response','Avg Hilbert','Avg dBDecay')
xlabel('Frequency (Hz)')
```

B.7 Loss Factor Estimation Code – Direct Averaging

```matlab
%%% Loss Factor Comparison - Without Autocorrelation
%%% Direct Averaging
%%% For this code the triggered samples are loaded
clc
clear all
close all

for iiii=1:4
    for freq=1:18
        accelnum=16
        for position=1:accelnum
            fc=[100 125 160 200 250 315 400 500 630 800 1000 1250
1600 2000 2500 3150 4000 5000 6300];
            frequency=fc(freq);
            eval(['load
triggered_one_third_octave_filtered_',num2str(frequency),'Hz_respons
e_LF_10perc_fs_20000_position_',num2str(position),'.mat'])
            trignumbers=[50 100 250 500];
            jjj=trignumbers(iiii)
            avgdata=mean(transpose(triggerdata(:,1:jjj)));

            if position==1
                avg_response=zeros(1,length(avgdata));
                avg_hilbert=zeros(1,length(avgdata));
                avg_logdecay=zeros(1,length(avgdata));
            end


avg_response=avg_response+(avgdata/max(avgdata))/accelnum;

avg_hilbert=avg_hilbert+abs(hilbert(avgdata/max(avgdata)))/accelnum;

avg_logdecay=avg_logdecay+20*log10(abs(hilbert(avgdata/max(avgdata))
))/accelnum;
            position

            clear triggerdata
        end

        dB_decay_avg_response=20*log10(abs(hilbert(avg_response)));
        dB_decay_avg_hilbert=20*log10(abs(avg_hilbert));
        dB_decay_avg_logdecay=avg_logdecay;

        ncorr=length(avgdata);
        dt=1/20000;
        timescale=dt:dt:ncorr*dt;

        startperc=round(0e-2*ncorr+1) %%% Start percent for slope
fitting
```

```matlab
        endperc=round(25e-2*ncorr+1) %%% End percent for slope
fitting

        figure(frequency)
        subplot(2,2,1);
        plot(timescale,dB_decay_avg_response,'r');hold on
        plot(timescale,dB_decay_avg_hilbert);hold on
        plot(timescale,dB_decay_avg_logdecay,'g');hold on
        axis tight
        title(['DB Decay Comparison',' @ ',num2str(frequency),'
Hz'])
        legend('DB Decay from Avg Free Decay','DB Decay for Avg
Envelope of All Output Points','Avg DB Decay of All Output Points')

        %%%
        for iii=1:3
            if iii==1
                data=dB_decay_avg_response;
            elseif iii==2
                data=dB_decay_avg_hilbert;
            elseif iii==3
                data=dB_decay_avg_logdecay;
            end

            figure(frequency)
            subplot(2,2,1+iii)
            plot(timescale,data);hold on;
            axis([0 ncorr*dt min(data) max(data)]);
            ylabel('DB Decay')
            xlabel('Time')

            k=0;
            if frequency<400 %%% Between 100 & 315 HZ
                i=5;
                while i<endperc
                    if data(i)>data(i-1) && data(i)>data(i-2) &&
data(i)>data(i-3)
                        if data(i+1)<data(i) && data(i+2)<data(i) &&
data(i+3)<data(i)
                            pt=data(i);
                            pt_i=i;
                            k=k+1;
                            ptarray(k)=data(i);
                            pt_iarray(k)=i;
                        end
                    end
                    i=i+1;
                end
            end

            if frequency>315 && frequency<1600 %%% Between 400 &
1250 Hz
                i=4;
```

```matlab
                while i<endperc
                    if data(i)>data(i-1) && data(i)>data(i-2)
                        if data(i+1)<data(i) && data(i+2)<data(i)
                            pt=data(i);
                            pt_i=i;
                            k=k+1;
                            ptarray(k)=data(i);
                            pt_iarray(k)=i;
                        end
                    end
                    i=i+1;
                end
            end

            if frequency>1250 && frequency<6300 %%% Between 1600 &
5000 Hz
                i=3;
                while i<endperc
                    if data(i)>data(i-1)
                        if data(i+1)<data(i)
                            pt=data(i);
                            pt_i=i;
                            k=k+1;
                            ptarray(k)=data(i);
                            pt_iarray(k)=i;
                        end
                    end
                    i=i+1;
                end
            end
            if k>=2
                ptarray1=ptarray;
                pt_iarray1=dt*(pt_iarray);
                plot(pt_iarray1,ptarray1,'g*');hold on
                p=polyfit(pt_iarray1,ptarray1,1);
                x=0:dt:ncorr*dt;
                y=polyval(p,x);
                plot(x,y,'g');
                slope=p(1);
                lossfactor_slopefit(iii)=-slope/(27.3*frequency)
%                 title(['DB Decay from Avg Free Decay
LF:',num2str(lossfactor_slopefit(iii)),' @ ',num2str(frequency),'
Hz'])
                clear pt pt_i k ptarray pt_iarray ptarray1
pt_iarray1 p x y slope data
            else
                lossfactor_slopefit(iii)=-(data(startperc)-
data(endperc))/(timescale(startperc)-
timescale(endperc))/(27.3*frequency)
%                 title(['DB Decay from Avg Free Decay
LF:',num2str(lossfactor_slopefit(iii)),' @ ',num2str(frequency),'
Hz'])
```

```matlab
                    clear pt pt_i k ptarray pt_iarray ptarray1
pt_iarray1 p x y slope data
            end
        end

eval(['loss_factor_',num2str(jjj),'(freq,:)=lossfactor_slopefit'])
    end
end
eval(['lossfactormatrix=[loss_factor_',num2str(trignumbers(1)),'
loss_factor_',num2str(trignumbers(2)),'
loss_factor_',num2str(trignumbers(3)),'
loss_factor_',num2str(trignumbers(4)),']'])

figure(12345)
loglog(fc(1:18),loss_factor(:,1),'-r*');hold on
loglog(fc(1:18),loss_factor(:,2),'-v');hold on
loglog(fc(1:18),loss_factor(:,3),'-g^'); hold on; grid on
title('Loss Factor Estimation (i/p LF=0.1, One-Third Octave & FDF)')
legend('Avg Response','Avg Hilbert','Avg dBDecay')
xlabel('Frequency (Hz)')
```