IMPROVING THE ANALYSIS OF OPERATING DATA ON ROTATING AUTOMOTIVE COMPONENTS.

A Dissertation submitted to the

Division of Graduate Studies and Research
of the University of Cincinnati

in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in the Department of Mechanical, Industrial,
and Nuclear Engineering of the
College of Engineering

December 1998

By

Jason R. Blough

B.S. Mechanical Engineering, Michigan Technological University, 1990.
M.S. Mechanical Engineering, Michigan Technological University, 1991.

Committee Chair: Professor David L. Brown

Acknowledgements.

How to thank all those that deserve it? **I can't!!** But I can try!! Here goes…

First, I need to thank my parents and brother for the never-ending support, encouragement, love, and advice over the last 30, make that 31 years or so. Everybody says you finish your Ph.D. when you are 30, whew, made that by 5 weeks! Though my brother beat me!!

I thank my many friends at Michigan Tech that helped give me the drive and background necessary to pursue this endeavor, both the scholastic background and the desire to get back to Michigan Tech to enjoy my family, the university, and the beautiful area. The most prominent of these people is Chuck VanKarsen….thanks Chuck for the support and encouragement over the last 9 years.

I thank my committee members who have worked with me over the last 5 years, has it been that long? I have had the best graduate student experience of any student I have met or talked to, SDRL has been truly GREAT. SDRL and its members have always provided me with whatever I needed to efficiently and happily pursue my degree and enjoy graduate student life! Thanks to Rhonda Christman who makes life easy by taking care of many of the details of life at UC for us. Special thanks to Dave and Pat Brown who truly took me into their family and always made me feel welcome to call or stop over no matter what time of the day or day of the week to discuss anything. And by the way….the only thing about Dave that I won't miss is his never ending list of ideas that *we ought to look at*…..you mean me or Bill or some other student Dave…not we! Though discussing many of the ideas has been fun and interesting! Randy Allemang gets thanks for being the steady constant and great friend over the last 5 years…needed to somebody to be organized and watch the details…thanks Randy! Thanks to Håvard Vold for giving me a good start into this rotating equipment stuff. And thanks to Bob Rost, another constant who has always been there for the details when I needed him. The lab itself deserves special thanks for being a great place with a great support system for one to develop their professional, academic, and personal life…the number of special opportunities for students to gain experience and enjoyment is truly unmatched.

Special thanks to Susan Dumbacher for being a great lifelong friend and always fixing all of my English mistakes, especially the run-on sentences…couldn't have done it without you. And I am truly grateful for the lifelong friends that I have made in the lab including, in no specific order, Bill Fladung, Allyn Phillips, Cindy Adelman, Jeremy Williams, Matt Witter, Mark Stebbins and too many others to list. I am gonna miss you guys!

Special thanks to Jim DeClerck, Tim Roggenkamp, Kevin Wittrup, and Gary Penn among many others at the General Motors Noise and Vibration Center for the personal friendship and support, the facility support, and helping secure financial support. Special thanks for that last one, it made graduate student life easy!

And thanks to the many others who have contributed to this endeavor and made it possible.

Abstract.

Order tracking is a very common method to analyze the response characteristics of rotating machinery to their rotating inputs. Unfortunately, many of the order tracking algorithms that are commercially available are considered proprietary by their developers. In an effort to produce a mostly complete reference and understanding of the characteristics of these methods many of them are documented in this dissertation. Two new order tracking methods are developed and documented which have the ability to separate both close and crossing orders. The Time Variant Discrete Fourier Transform (TVDFT) is developed and shown to be a very powerful and versatile order tracking method as well as a very computationally efficient algorithm. With the ever increasing speed of computers, post-processing of time domain data is becoming popular. A post-processing application which is very computationally demanding in its current commercial implementations is adaptive resampling, commonly used to resample data from the time to the angle domain. A new adaptive resampling method is developed that is based on an upsampled interpolation filter that is very computationally efficient. It should be noted that all order tracking and adaptive resampling methods rely very heavily on an accurate tachometer signal. For this reason, the processing of tachometer signals is included in this dissertation. Finally, a new set of analysis tools formulated around the singular value decomposition (SVD) and the Complex Mode Indicator Function (CMIF) algorithms are developed to compute virtual measurements from order tracks. These tools also provide the ability to estimate linearly independent operating shapes from a set of operating shapes based on order tracks. A virtual measurement called a Mode Enhanced Order Track (MEOT) is developed which should prove useful in estimating natural frequencies and damping from order track measurements. All new methods and several of the traditional methods are evaluated using both analytical and experimental datasets. The experimental datasets include the analysis of data acquired on an automobile operating on a chassis dynamometer, including the separation of the inputs from the left and right wheels. The final result of this dissertation is a complete reference and suite of tools for analyzing many rotating equipment noise or vibration problems.

Table of Contents.

Table of Figures.

List of Tables.

Nomenclature.

Greek Characters:

Δf	Frequency resolution of time domain FFT.
Δo	Order resolution of angle domain FFT.
Δ_{rpm}	Difference between the maximum and minimum rpm in the integration time, T.
Δt	Time sample interval for discrete data.
$\Delta\theta$	Angular sample rate of the angle domain resampled samples.
$\varepsilon(n)$	Nonhomogeneity term of Kalman filter.
$\eta(n)$	Nuisance component of Kalman filter.
$\phi(t)$	Angle of rotation of a shaft.
ϕ_1, ϕ_2, ϕ_3	Rotational positions of three consecutive tachometer pulses.
ϕ_k	Desired angular position for a new data sample, k – HP tachometer analysis.
ϕ_k	Phase angle of order k – Order analysis.
ω	Instantaneous frequency of the sine wave.

Roman Alphabet:

a_m	Estimated Fourier transform cosine coefficient.
b_m	Estimated Fourier transform sine coefficient.
b_0, b_1, b_2	Polynomial coefficients in HP tachometer analysis.
A(k,t)	Amplitude of order k as a function of time.
e_{ij}	Cross orthogonality contribution of order *i* in the estimate of order *j*.
F_i	Synthesized order input for order o_i.
F_{max}	Maximum frequency that can be analyzed of discretely sampled time domain data.
$F_{nyquist}$	Nyquist frequency of discretely sampled data.
f_m	Frequency of sine/cosine in Fourier transform, $m\Delta f$.
F_{sample}	Sample frequency of discretely sampled data.
$[G_{XF}(\omega)]$	Crosspower matrix between the response dofs and the tachometer signals.
$[G_{FF}(\omega)]$	Input crosspower matrix between the tachometer signals.
$[H(\omega)]$	Estimated FRF matrix (order tracks).
k	Order of interest.
$MEOT_{l,r}$	Mode enhanced order track for singular vector l at rpm r.
N	Total number of time points over which Fourier transform is performed.
N	Number of measured degrees of freedom in virtual measurement formulations.
o_i	Orthogonality compensated value of order *i*.
$\tilde{o}_i$	Estimated value of order *i* obtained using the TVDFT.
o_m	Order of interest, $m\Delta o$.
{O}	Operating shape vector for each order of interest at rpm r in no particular order.
$\{O_i\}$	Operating shape vector of order i at the analysis rpm.
$\{O_j\}$	Operating shape vector of order j at any rpm value.
$OA_{i,r}$	Order track autopower of order i at rpm r.
O_{ik}	Value of order i at degree of freedom k at rpm r.

O_{max} Maximum order that can be analyzed of discretely sampled angle domain data.
$O_{nyquist}$ Nyquist order of angle domain FFT of discretely sampled angle domain data.
O_{sample} Angular sample rate of discretely sampled angle domain data.
p Period of primary order.
R Total number of revolutions that are analyzed.
RPM_{sweep} Frequency which the primary order sweeps over in the integration time, T.
[S] Diagonal matrix of singular values resulting from an SVD.
$SVA_{l,r}$ Singular vector autopower of singular vector l at rpm r.
t_1, t_2, t_3 Three consecutive tachometer pulse arrival times in HP tachometer analysis.
t_k Corresponding time of a desired angular position to sample k in HP tachometer analysis.
T Integration time of FFT.
[U] Matrix of left singular vectors resulting from an SVD.
$\{U_l\}^H$ Hermitian of the left singular vector l.
$[V]^H$ Matrix of right singular vectors resulting from an SVD.
$\{V_l\}$ Right singular vector l that the MEOT is calculated with respect to.
W_{energy} Correction for the energy bandwidth of the applied window.
$x(n\Delta t)$ nth discrete data sample of x.
$\tilde{x}(t)$ Re-modulated order function.
$y_{DC}(t)$ Frequency shifted time history.

Abbreviations:

Binsband Number of bins that the energy of the maximum order of interest is smeared over.
CMIF Complex mode indicator function.
dof Measurement degree of freedom.
FFT Fast Fourier transform.
FRF Frequency response function.
HP Hewlett Packard Company.
LMS Leuven Measurement Systems.
MaxOrder Maximum order of interest.
MEOT Mode enhanced order track.
OCM Orthogonality compensation matrix.
rpm Instantaneous rpm of the machine.
SVD Singular Value Decomposition.
TMON LMS Time Data Processing Monitor.
TVDFT Time Variant Discrete Fourier transform.

Chapter 1

1 Introduction.

1.1 Rotating Machinery Problem Solving.

Experimental problem solving approaches commonly used to solve noise and vibration problems on automobiles and rotating machinery generally fit into two distinctly different classifications. The first of these classifications is defined by techniques that attempt to characterize the dynamic properties of the machine under study. Modal analysis is one of these techniques. The second of these classifications is defined by techniques that attempt to characterize the response signals measured while the machine is in operation. Order tracking is one of these techniques.

Techniques that attempt to characterize the dynamic properties of a machine are usually preferred because the processed results of these techniques have true physical meaning with respect to the dynamic properties of the structure. Characterizing physical properties of a structure allows an analytical model of the structure to be constructed for design analysis. Modal analysis is the most commonly utilized method for characterizing the dynamic properties of a structure. Over the last thirty years there has been much research time and energy devoted to understanding and further developing methods and algorithms to more easily determine the modal parameters of structures. These modal parameters are determined under the assumptions of time invariance and system linearity. These assumptions can be invalid when characterizing rotating machinery since oftentimes bearings are not linear or time invariant in nature.

The techniques that attempt to characterize the structure of response signals measured while a machine is in operation have the disadvantage of not being based on physical models of the system. The parameters estimated by these techniques may or may not have significance in actually attempting to characterize the dynamic properties of a structure. However, an advantage these techniques possess is that there is no assumption made on the linearity of the structure.

The most commonly used methods for characterizing the response signals acquired on operating rotating machinery fall under the classification of *order tracking*. Order tracking is the estimation of the amplitude and phase of a response of a machine to a rotating input that is allowed to vary in both amplitude and frequency over time. There are several basic methods that have been developed to perform digital order tracking. Nearly all of these methods have been in use for less than 25 years and several for under a decade. Since these techniques have only recently been developed, there has not been a large amount of literature published that clearly explains and compares the details, performance, and implementation of these algorithms.

1.2 Background.

Digital order tracking techniques range in complexity from simple FFT based techniques to much more complicated adaptive filtering techniques. Generally, the more theoretically complicated the technique, the more computationally demanding the algorithm. This leads to the more complicated algorithms being implemented in special hardware or software which is expensive and therefore in limited use.

1.2.1 Fourier Transform Based Order Tracking.

The simplest of the digital order tracking techniques are based on performing Fourier transforms on time domain data. The transformed data is then displayed in either a waterfall or colormap format to obtain a global view of the frequency domain response. Orders of interest are estimated through determining the average frequency of each order over which the FFTs were performed and extracting the corresponding frequency lines from the FFTs. Limitations of Fourier transform based techniques are many and can be significant. The two largest limitations of these techniques are limited order resolution at low rpms and slow sweep rates. Both of these limitations have been thoroughly explored and documented [ref. 1-4,24].

The FFT based techniques can be used to obtain either constant frequency or constant order bandwidth results. Regardless of which bandwidth option is chosen, the Fourier transforms are performed over the time determined by the blocksize of the transform. The blocksize of the transforms is not related to the rpm of the machine. This independence of the blocksize and rpm is what leads to many of the limitations of the technique. There is also no special processing performed to minimize leakage, other than the application of standard windows. The theory behind many of the limitations of time based sampling and Fourier transforms for analyzing operating data is presented in Appendix A.

1.2.2 Computed Order Tracking.

A method developed to overcome the limitations of the Fourier transform based techniques is the computed order tracking technique based on synchronous resampling. This method was developed and patented by Potter at Hewlett Packard in 1989 [ref. 5-11]. The method is the digital equivalent of using an encoder attached to the rotating shaft of interest, or a frequency synthesizer with a low count pulse per revolution tachometer pick-up, to drive the sample clock of the ADC and an analog tracking anti-alias filter.

Computed order tracking, as developed by Potter, is based on digitally resampling data acquired with an equal time interval, constant Δt, to data obtained with an equal angular interval, constant $\Delta\theta$. The resampling is computationally demanding. The result of the process is the transformation of time/frequency data to angle/order data. Orders sweep across multiple frequency bins in the frequency domain, while in the order domain these same orders are stationary with respect to frequency. This allows leakage free estimates of orders by using the standard Fourier transform and an appropriate blocksize. The theory of angle domain sampling is presented in Appendix A for completeness.

A Fourier transform performed on angle domain data calculates estimates of order domain amplitudes and phases. Any order that falls on a spectral line in this transform is estimated as leakage free, if the transform is performed over an integral number of revolutions of the machine. A computational advantage of this transform can be realized if a discrete Fourier transform is calculated for each order of interest. Not computing the full spectrum implies that an overall observation of the data cannot be obtained from a waterfall or colormap plot, but can result in a large computational savings.

A study of the limitations and behavior of the computed order tracking techniques was performed by Fyfe and Munck [ref. 12]. This study of errors and limitations is satisfactory for many of the errors that are related to tachometer signal processing and acquisition. The tachometer processing which is evaluated is the method that Potter published. This method is based on fitting a local polynomial to every three tachometer pulses. The local estimation technique is required for the real time implementation that HP has implemented in their dynamic signal analyzer systems. Many of the limitations beyond the tachometer signal processing which Fyfe and Munck have evaluated are for various digital resampling techniques, none of which are believed to be what Potter has implemented, or very practical in a computational sense. The exact interpolation method that Potter uses in his patented technique is not published and is considered to be proprietary by Hewlett Packard.

Recently LMS has implemented an adaptive resampling algorithm for performing the transformation from the time domain to the angle domain, or any other domain, for which there exists the necessary reference signal, as a post processing procedure. This implementation of adaptive resampling first upsamples the Δt data by a given amount, typically sixteen. A linear interpolation is then performed on the upsampled data to obtain the desired $\Delta\theta$ data [ref. 13,14]. This method is very computationally demanding since the length of the original data set is first made sixteen times longer before the linear interpolation is performed. The signal to noise ratio of this procedure is determined by the amount of upsampling performed prior to the linear interpolation.

Implementing the adaptive resampling procedure as a post processing option allows other analysis in addition to order tracking. Once the data is resampled to constant $\Delta\theta$, many standard digital signal processing procedures may be used for analysis flexibility. For example, standard digital filters become constant order bandwidth order filters when applied to angle domain data. This allows “what if games” to be evaluated for sound quality or fatigue analysis. The data may also be transformed from the angle domain back to the time domain for further analysis after modification in the angle domain.

Another recently implemented technique of computed order tracking has been developed by Bruel & Kjaer [ref. 15,16]. This technique is again considered to be proprietary and many details of the actual implementation are not published. The implementation allows a type of zoom processing in the order domain that can give better order resolution while analyzing higher orders [ref. 15]. This implementation is similar to Hewlett Packard’s in the fact that it is implemented in hardware.

1.2.3 Kalman Filter Based Order Tracking.

Vold and Leuridan developed an approach for order tracking based upon the Kalman filter from controls theory [ref. 17]. The method, while being computationally demanding, is much more flexible in use than any previously developed order tracking technique. The main advantages of Kalman order tracking are the abilities to separate very close orders and to extract the actual time domain representation of the orders [ref. 18]. The time domain representation of the orders is extracted without phase bias and therefore can be used for sound quality analysis and *what if games*.

The disadvantages of the Kalman order tracking are the computational load and the harmonic confidence factor determination. The harmonic confidence factor is a weighting factor that must be defined for the least squares solution process. The weighting factor controls the performance of the filter. It determines the bandwidth and convergence speed of the filter. Unfortunately, no straight forward technique exists for determining the value of this weighting factor to obtain consistent results for all datasets.

The original Kalman order tracking filter has been recently compared to other signal processing techniques in a series of papers [ref. 28-30]. These papers compare both the Kalman order tracking filter and the resampling based order tracking to various other digital signal processing techniques for the analysis of different types of data.

Vold has extended the concept of Kalman order tracking with the Vold-Kalman tracking filter [ref 19]. This filter is a slightly different formulation of the original Kalman order tracking filter and extends the capabilities to include the separation of crossing orders as well as better separations of close orders [ref. 20,21]. The filter shape may be varied through the use of a higher order filter implementation. The filter uses a more user friendly approach to determine the harmonic confidence factor. However, experience is still required to obtain accurate results.

1.2.4 Non-Commercial Order Tracking Techniques.

There are other order tracking techniques that are not implemented in commercial software or hardware. These methods include using a time variant zoom transform to shift the frequency of the order of interest to DC, followed by applying a low pass digital filter or performing an FFT and zeroing frequency lines above a frequency near DC. These methods result in time domain order extraction similar to the original Kalman order tracking filter. These techniques were developed by Vold and Leuridan as part of the development of the Kalman order tracking filter [ref. 22,23].

The complex exponential algorithm that is commonly used in modal analysis has also been studied as a method for order tracking [ref. 26]. The disadvantage with this algorithm is the necessity to determine the number of orders present in the data before an accurate result can be obtained. Typically there is no easy or automated way of determining this model order.

The maximum entropy technique has also been evaluated as an order tracking technique. This method proved to have the same basic limitation as that of the complex exponential, the determination of the order of the optimum FIR filter [ref. 27].

1.2.5 Frequency Domain Order Tracking.

Weber explored a frequency domain order tracking technique that used multiple input frequency response function estimation theory to estimate the response of an automobile to rotating unbalance forces [ref. 25]. This technique Fourier transformed tachometer signals from different rotating components and response channels. The Fourier transforms of the tachometer signals were treated as inputs to the structure and the appropriate input crosspower matrix calculated. The crosspowers between each assumed input, or tachometer signal, and each response were then calculated and a multiple input H1 calculation performed. This procedure estimates the portion of each response caused by each rotating input monitored by a tachometer signal. This method should accurately separate the contributions of each rotating input under arbitrary operating conditions. This method is not restricted to operating on simple run-ups or run-downs, but can operate on any arbitrary combination of run-ups and run-downs.

1.2.6 Tachometer Signal Processing.

Tachometer signal processing is a very important portion of order tracking analysis. Any order tracking results can be thought of as only being as accurate as the tachometer signal that was used to estimate the instantaneous frequency of the order in the analysis process. The most important acquisition channel in any test is the tachometer channel. If the quality of the tachometer channel is poor, the results from all other channels will be poor or in the extreme case completely unreliable. The higher fidelity the order tracking method the more important the tachometer channel.

Historically, tachometer channels were conditioned with a tracking ratio tuner. The use of the tracking ratio tuner or phase locked loop to condition the tachometer signal resulted in a stable clean tachometer signal for input into the data acquisition or tracking filter hardware. The disadvantage of this type of tachometer signal conditioning is the limited sweep rate which results from the use of a phase locked loop. To overcome these sweep rate limitations digital tachometer processing methods have been developed.

Potter developed the concept of fitting a local polynomial to consecutive tachometer pulse arrival times to enable the real time computed order tracking which Hewlett Packard patented and implemented in their hardware [ref. 5-11]. This method of processing tachometer signals was thoroughly investigated for errors by Fyfe and Munck [ref. 12]. The error analysis is both quite in depth and at the same time not completely accurate due to the interpolation methods that were evaluated. While the investigation accurately shows trends in errors, it is not evident whether the magnitudes of the errors are realistic.

Vold and Leuridan developed a tachometer processing technique that fits multiple cubic splines to estimated tachometer pulse arrival times of an entire time history simultaneously. The method divides the entire time block into multiple blocks. Splines are then fit to each block with continuity enforced across the blocks. This methodology works very well and is used in both the Kalman order tracking and the LMS implementation of adaptive resampling. Fitting the splines to the tachometer pulse arrival times averages out many of the small inconsistencies in rotational speed and a smooth rpm curve is estimated. Errors can still exist in the rpm estimate if there is a drop-out in the tachometer signal or if an extra tachometer pulse is recorded.

Vold extended the concept of spline fitting the tachometer signal with the implementation of the Vold-Kalman order tracking filter. The advances in this new spline fit algorithm include the ability to relax the slope continuity between blocks to allow for more realistic rpm estimates near gear shifts where the rpm changes very drastically. The ability to reject rpm estimates from either missed tachometer pulses or extra tachometer pulses is also present in the second generation spline fitting algorithm. This is done through obtaining a first estimate of the splines for each block. Rpm estimates that are outside a pre-defined band relative to this original spline fit are then discarded. Having discarded outliers, the spline fit is repeated for an improved rpm estimate. This implementation is currently considered to be the state of the art and works very well.

1.3 GOALS OF DISSERTATION.

This dissertation seeks to extend current order tracking capabilities and to determine more computationally efficient methods of implementing current methodologies.

The first objective of this dissertation is to explain the theory and implementation of the currently employed order tracking techniques. These methods will also be contrasted and compared using identical datasets for all methods. The end result of this study will be a concise and complete reference for understanding current order tracking methodologies.

It has been shown that the resampling based order tracking techniques are superior in most cases to the FFT based techniques but suffer from a large computational penalty. This dissertation seeks to address this issue along two separate avenues. The first thrust of research will be in identifying the most efficient and accurate means of digitally resampling data from the time domain to the angle domain. Current post processing implementations of adaptive resampling are very computationally demanding. It is believed that an efficient means of adaptive resampling will lead to more widespread use of the signal processing and analysis advantages of this technique.

The second avenue that will be pursued in obtaining resampling based order tracking results efficiently will be the investigation of an alternative method of processing the data. It is shown that the kernels of the Fourier transform can be resampled, as opposed to response channel resampling. This method gives amplitude and phase estimates nearly identical to the current resampling methods but does not allow for “what if games” with angle domain data.

A deficiency in current tachometer signal processing techniques is evident if a tachometer signal generated by a magnetic inductance probe is analyzed. Oftentimes, a signal generated by a magnetic inductance probe varies in amplitude as a function of the rotational speed of the machine. The varying amplitude is not processed accurately if a constant trigger level is used to determine when each tachometer pulse occurs. One solution to this problem that will be evaluated in this dissertation is the development of an algorithm that adaptively varies the trigger level as a function of total signal amplitude.

Currently the only method that can separate crossing or very close orders is the Vold-Kalman order tracking filter. This filter is computationally demanding but provides time domain outputs for *what if games*. This dissertation will evaluate other methods of separating both close and crossing orders that are more computationally efficient than the Vold-Kalman order tracking filter. Both frequency and order domain approaches will be evaluated to separate close or crossing orders.

The frequency domain approach will be an extension of the technique that Weber developed. The order domain approach will be based on an order domain implementation of the multiple input H1 frequency response function estimator.

Finally, the estimation of linearly independent operating shapes will be evaluated using SVD techniques similar to those used in the CMIF modal parameter estimation procedure [ref. 54-56]. This estimation of linearly independent operating shapes is a new contribution to the field and should provide more information into the dynamic properties of a structure than is commonly achieved from order tracking analysis. Currently, operating shapes can be obtained from order tracking many response channels and animating the amplitude and phase of the results at each rpm. These estimated shapes are not currently processed in any manner to attempt to obtain a set of linearly independent shapes.

1.4 CONCLUSIONS.

This dissertation will result in a complete reference for current order tracking techniques, as well as the development of several new methods for processing rotating machinery response data. These new methods will include new order domain and frequency domain methods of analysis capable of separating close and crossing orders. Finally, an evaluation of methods to obtain linearly independent operating shapes from the processed order tracked data will be undertaken. All methods are evaluated with analytical and experimental datasets.

For completeness, appendices are included to provide a short introduction to the digital resampling processes necessary for the adaptive resampling procedures and for the theory and significance of angle domain sampling and processing versus time domain sampling and processing.

Chapter 2

2 Current Order Tracking Techniques.

Currently used order tracking techniques range from very simple and basic in nature to very complicated and computationally demanding. A brief summary of the basic analog order tracking technique is presented for completeness. All other commercially available order tracking techniques are also presented in this chapter with an explanation of the basic techniques as well as a summary of each technique's advantages, limitations, and disadvantages.

2.1 Introduction to Order Tracking Analysis.

The analysis of non-stationary noise and vibration signals on rotating machinery is commonly performed through the use of digital order tracking techniques. Order tracking is the analysis of frequency components whose frequency is related to the rotational frequency of the operating machine. The frequency components will be time varying if the machine is not operating in a stationary condition. The analysis of non-stationary components requires additional information for accurate results to be obtained. This additional information is usually presented in the form of a tachometer signal measured on a reference shaft of the machine.

An order is a time varying phasor that rotates with an instantaneous frequency related to the rotational frequency of the reference shaft. A graphical presentation of this time varying phasor is given in Figure 2.1. It can be seen that the rotating phasor will produce a sinusoidal function with varying frequency. The different rotational positions of the phasor are meant to show how the phasor produces a sinusoid they are not at consecutive sample points.

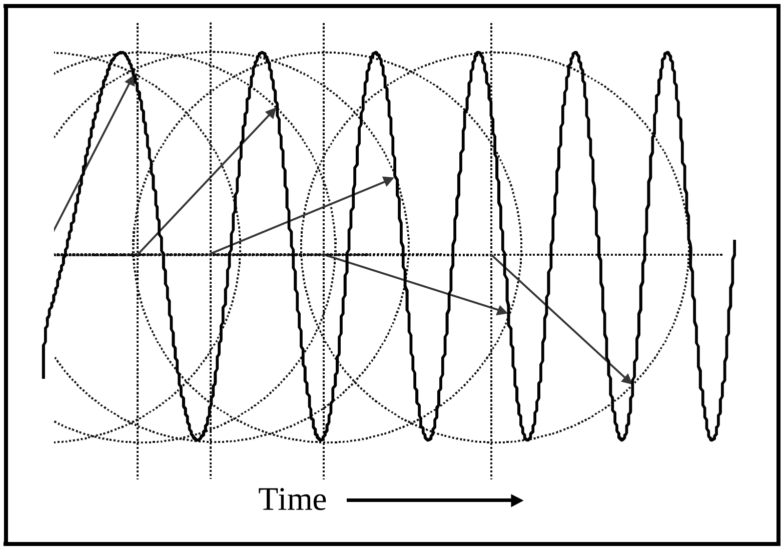

Figure 2.1 Graphical representation of time varying phasor.

A single order may be mathematically defined by the time varying phasor described by Equation 2.1.

$$X(t) = A(k,t)\sin(2\pi i(k/p)t + \phi_k) \qquad (2.1)$$

where: A(k,t) is the amplitude of order k as a function of time.
ϕ_k is the phase angle of order k.
p is the period of primary order in seconds.
t is time.
k is the order being tracked.
k = 0 DC offset.
k < 0 Negative frequencies.

Multiple orders are normally present in a dataset acquired from an operating machine. The combination of orders may be described mathematically by a summation of time varying phasors. This combination is expressed in Equation 2.2.

$$X(t) = \sum_{k=-\infty}^{\infty} A(k,t)\sin(2\pi(k/p)t + \phi_k) \qquad (2.2)$$

An order commonly possesses frequencies that are integer multiples of the primary frequency of the reference shaft. However, non-integer multiples of the primary frequency may be generated if gearboxes or pulley systems are present in the system. These systems can have non-integer rotational speed relationships that can cause fractional orders that are close to one another in frequency. Fractional orders are also generated by bearing or gear defects. Oftentimes, it is this situation that makes the analysis of orders difficult. Another source of non-integer multiples of the primary reference signal are from other input sources. Oftentimes, a machine may have multiple uncoupled rotating components that generate forces. Examples of multiple inputs are the 4 wheels on an automobile, engine and transmission, when they are not rigidly coupled, and many other combinations of components.

Order functions can be generated by any rotating input on an operating machine and may vary in amplitude and/or frequency as a function of time. This amplitude varying property causes considerable problems in any type of order tracking analysis. Until recently, all order tracking techniques considered the amplitude of an order to be semi-constant over the analysis period used to estimate the amplitude and phase of the order. This assumption can cause considerable errors in the analysis.

The most common method to visualize order estimates is through the use of both 3-dimensional waterfall and colormap plots and 2-dimensional x-y plots. The 3-dimensional plots are typically plotted with frequency along the x-axis, time or rpm along the y-axis, and amplitude along the z-axis. These 3-dimensional plots are typically called rpm spectral maps or Campbell diagrams. The 2-dimensional x-y plots are often plotted with the x-axis as either time or rpm and the y-axis as amplitude. Oftentimes, a 2-dimensional plot will contain the amplitude profiles of multiple orders and an overall energy level in the frequency range analyzed. These 2-dimensional plots are typically called order slice plots. The advantage of plotting either the 3-dimensional or 2-dimensional plots as a function of time instead of rpm is that both run ups and run downs can be visualized simultaneously without confusion.

2.2 Analog Order Tracking.

Prior to Potter's development of computed order tracking based on digital resampling in 1990, purely digital order tracking methods were not often based on synchronous sampling and therefore not considered order tracking in the strictest sense. Non-synchronous sampling based digital order tracking was accomplished through standard Δt based Fourier transforms. Prior to the widespread adoption of the computed order tracking methods, analog order tracking was performed in many cases where the Fourier transform was limited by sweep rates. Computed order tracking in one form or another is now very widely used since the cost of the necessary DSP chips for the computationally demanding algorithms has drastically dropped in price over the last few years. This section explains the basic analog order tracking method and is primarily included for completeness of documentation of order tracking methods.

Originally analog order tracking methods required no digital equipment whatsoever. These original analog order tracking methods used a tracking bandpass filter whose output was input into an x-y plotter to plot the amplitude of an order as a function of rpm. The frequency of the tracking filter and the x axis of the plotter were driven by a tachometer or encoder signal. The tracking bandpass filters were very expensive and could process only limited sweep rates. These purely analog order tracking methods are no longer used in industry except possibly in very special cases.

A derivative of the original analog order tracking, while still requiring expensive equipment, is still used today by many diesel engine manufacturers, and in some cases where the DSP chips and sampling rates of the current DSAs are not sufficiently fast to provide the desired results. The diesel engine manufacturers continue to use this pseudo-analog order tracking because they already possess the required equipment and very often are only interested in a few channels of data.

While called analog order tracking, these methods are not completely analog in nature, however from hereon they will be referred to as analog order tracking to be consistent with industry terminology. Analog order tracking methods use a tracking low pass filter to prevent aliasing. This tracking filter is inserted into the acquisition system before the ADC. The ADC clock is driven by an external sample clock signal of some type to attempt to sample the signal synchronously with the rotation of the machine. Once the data is anti-alias filtered and synchronously sampled, a discrete Fourier transform is performed over an integer number of revolutions. This technique originated shortly after the FFT became widely used, the earliest reference which was found was 1972, however it is believed to have originated long before this time [ref. 48-53]. A schematic of the hardware necessary to perform this type of order tracking is shown below in Figure 2.2.

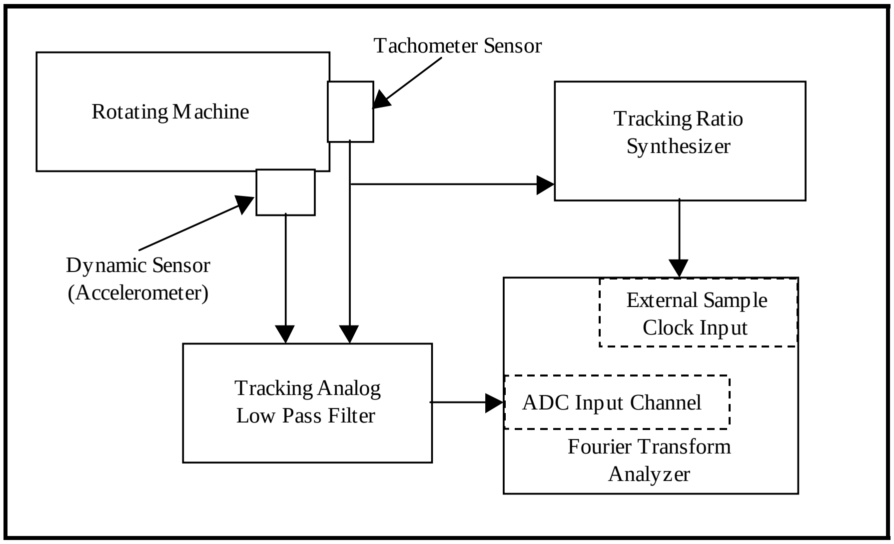

Figure 2.2: Analog order tracking hardware schematic.

The expensive component in an analog order tracking system is the tracking analog lowpass filter required to prevent aliasing in the sampled data. The frequency of the tracking filter is required to vary based on the rotational speed of the machine being analyzed. The cutoff frequency of the filter is varied such that the same number of orders are allowed to pass through the filter regardless of the rotational speed of the machine. Another limitation of the tracking filter is the transient characteristic produced by varying the cutoff frequency of the filter. Oftentimes, the cutoff frequency cannot be varied at high rates because the filtered data will contain transient phenomena from the filter settling at each cutoff frequency. This limitation of the filter can limit the sweep rate that a machine may undergo and still result in valid data.

To build a tracking filter whose amplitude and phase frequency response characteristics are constant with an arbitrary cutoff frequency is both complicated and expensive. Considering that in many cases it would be desirable to order track more than one channel of data simultaneously, the cost of the required filters increases dramatically. In tracking multiple channels simultaneously, the filters used must be amplitude and phase matched with one another. This amplitude and phase matching becomes very difficult and therefore very expensive with tracking filters. To reduce the cost of this system, data is usually recorded on a tape recorder and played back one channel at a time through the same tracking filter. This procedure guarantees amplitude and phase matching between channels, since only a single filter is used for all channels. However, it linearly increases the analysis time with the number of channels processed.

To drive the frequency of the tracking filter, a reliable tachometer signal is required. The tachometer signal is also required to allow the data to be acquired synchronous with the rotation of the operating machine. Just as in all order tracking methods, the higher quality the tachometer signal, the higher quality the resulting data will be. The tachometer signal may be generated by an encoder mounted on the rotating shaft of the machine that generates a pulse count the same as the desired synchronous sample rate. Another common method of generating a tachometer signal is with the use of a tachometer signal and a tracking ratio synthesizer. The tracking ratio synthesizer is an expensive piece of hardware, however there is only one needed for each rotating shaft and not one per general acquisition channel. The hardware schematic for the latter type of system is shown in Figure 2.2.

While the encoder method requires less actual hardware, it is often difficult to mount an encoder on a rotating shaft. The encoder also does not allow the engineer the ability to change the sample rate or cutoff frequency of the tracking filter, since it has a fixed number of output pulses per revolution. The advantage of an encoder is that there is no intermediate estimate of the rotational speed of the rotating shaft and therefore no errors in the synchronous sampling of the data.

While the encoder does not require any intermediate speed estimation, this is not true with a tracking ratio synthesizer. A tracking ratio synthesizer estimates the period between two consecutive tachometer pulses. This period is then used to generate the external sample frequency used to trigger the ADC in the acquisition system until the arrival of the next tachometer pulse. Inherent is this sampling procedure is a single tachometer pulse lag of the sample frequency with respect to the actual rotational frequency of the machine. This characteristic can limit the sweep rate that the machine may undergo and still obtain accurately sampled data. Figure 2.3 shows a plot of the actual rpm vs. the assumed rpm of a tracking ratio synthesizer and gives a graphical representation of the difference between the actual and assumed rpm values [ref.7]. Where the two curves are close to one another, the lag error will be small.

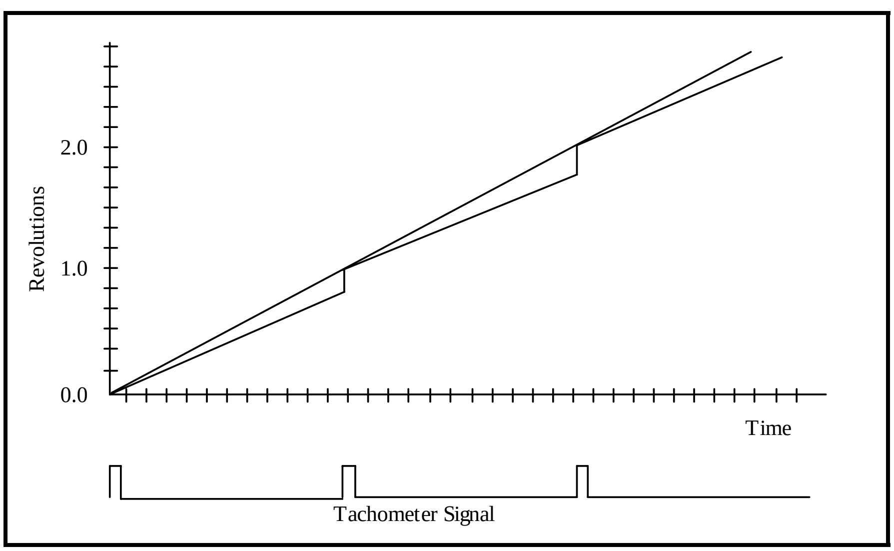

Figure 2.3: Actual versus estimated revolutions from tracking ratio synthesizer.

The error introduced by the sample frequency estimation lag will be small if the sweep is linear and/or slowly varying. This results in long sweep times and can exclude a sweep condition that may be necessary for troubleshooting. Oftentimes a faster sweep rate may be necessary to simulate an actual operating condition, such as a speed sweep in the low gear of a transmission. These low gear sweeps can be very rapid and must be simulated with rapid sweeps if powertrain mounts are to be loaded in a realistic manner.

If the tachometer signal is not accurate, the anti-alias filter will not be operating with the proper cutoff frequency and the sample rate of the data will be incorrect. An incorrect sample rate will lead to many of the errors that order tracking analysis was developed to minimize. The largest of these errors is leakage. Aliasing may also occur if the anti-alias filter is not operating with the proper cutoff frequency.

2.3 Time Domain Sampling Based Order Analysis.

The simplest and oldest method of digital order analysis is through the use of the standard Fourier transform. This method came into use early in the FFTs history. Again, the earliest reference found was 1972 but it is known that the method was in use before that time [ref. 48-53]. The Fourier transform is typically applied as a fast Fourier transform (FFT) because of its computational speed, therefore from this point on it will be referred to as an FFT based technique.

The advantage of the FFT based order analysis is its simplicity and computational efficiency. FFT based order tracking does not require any special hardware and can easily be implemented in either a real time or post-processing calculation. However, there are severe disadvantages in using the FFT to analyze time varying signals. These disadvantages are due to the constant frequency sampling approach and constant frequency kernel which are used to acquire and analyze the data, respectively.

2.3.1 Fourier Transform Based Order Analysis Theory.

The sampling theorem that the FFT is based on is the well understood and documented Shannon's sampling theorem presented in Equation 2.3.

$$
\begin{aligned}
&\Delta f = \frac{1}{T} = \frac{1}{N * \Delta t} \\
&T = N * \Delta t \\
&F_{nyquist} = F_{\max} = \frac{F_{sample}}{2} \\
&F_{sample} = \frac{1}{\Delta t}
\end{aligned}
\tag{2.3}
$$

where: Δf is the frequency resolution of the resulting frequency spectrum.
T is the total sample time analyzed.
N is the total number of time points over which the transform is performed.
Δt is the time spacing of the time samples.
F_{sample} is the sample frequency of the data.
$F_{nyquist}$ is the Nyquist frequency.
F_{max} is the maximum frequency which can be analyzed.

As can be seen in Equation 2.3, the sampling theorem is based on acquiring data samples with a uniform time spacing, Δt. The equation also shows that the sampling theorem is not related in any way to the behavior of the rotating machine. This implies that the result from the transform is based on frequencies and not orders. This property is further shown by the transform itself that is presented in Equation 2.4.

$$
\begin{aligned}
a_m &= \frac{1}{N} \sum_{n=1}^{N} x(n\Delta t) \cos(2\pi f_m n\Delta t) \\
b_m &= \frac{1}{N} \sum_{n=1}^{N} x(n\Delta t) \sin(2\pi f_m n\Delta t)
\end{aligned}
\tag{2.4}
$$

where: $x(n\Delta t)$ is the nth discrete data sample.
f_m is the frequency of the sine/cosine terms, $m\Delta f$.
a_m, b_m are the estimated Fourier coefficients.

The properties of the sampling, kernels, and the type of data that the FFT is well suited to analyzing are shown graphically in Figure 2.4. The kernels of the FFT are the constant frequency constant amplitude sine and cosine functions for each Δf. This figure also shows that the kernel has the same form as the constant frequency data that it was developed to analyze. Mathematically, a transform works well if the form of the kernel matches that of the data. Clearly, the form of the FFT kernel does not match time varying data.

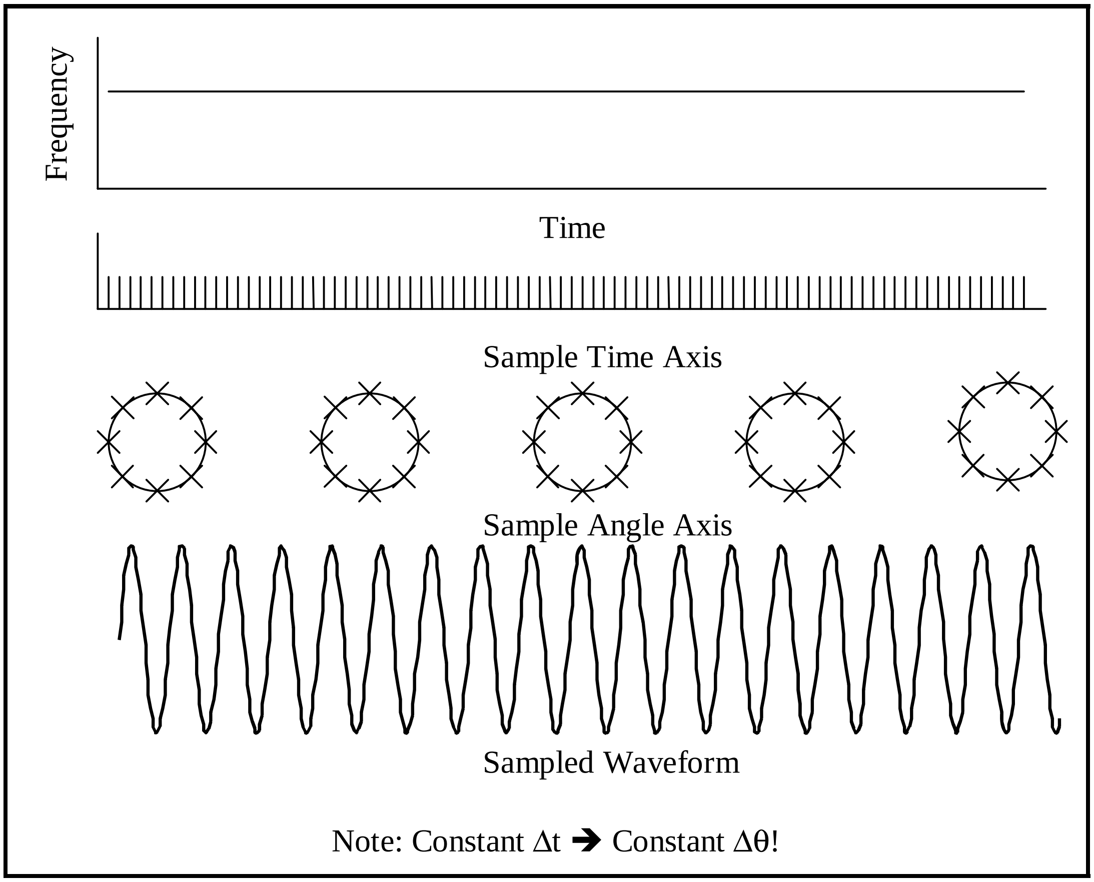

Figure 2.4: Graphical representation of Fourier transform, its properties, and sampled constant frequency data.

FFT based order tracking performs sliding FFTs on time domain data. The average rpm over which the transform is performed is also calculated. This average rpm is then used to estimate the frequency, and therefore the frequency bin, of the orders of interest for each estimated spectra. Once the frequency bin of an order is known, the amplitude and phase of the order can be extracted from the FFT spectra. Since the order is not in any way locked to the Fourier transform, the order will not necessarily fall on one spectral line. For this reason, oftentimes multiple spectral lines are summed around the center bin determined from the average rpm. Since multiple spectral lines are being summed to arrive at an amplitude/phase estimate, the correction factor for the window applied to the data, usually a Hanning, is typically the energy correction factor.

Three different strategies are normally used to determine the number of bins to sum on each side of the estimated order spectral line to calculate the amplitude and phase of an order. The first sums a constant number of spectral lines, regardless of the rpm or the frequency of the order, resulting in a constant frequency bandwidth order slice. While this strategy of extracting an order is a commonly used method, its results do not usually correlate well with other more advanced order bandwidth types of order tracking.

The second strategy that can be used to extract the order information from a series of FFT spectra is to sum the frequency bins around a given order to obtain a constant order bandwidth result. A constant order bandwidth is determined from the frequency of the primary or reference order that the tachometer measured directly. The 1st order frequency can be multiplied by a factor that results in a constant order bandwidth. This then results in a different number of frequency bins being summed for different rpm values. This method usually produces a result that is close to the result obtained with other constant order bandwidth order tracking methods.

The last strategy that may used to extract order information from a series of FFT spectra provides a constant percentage bandwidth result. Like the constant order bandwidth procedure, this method sums a different number of frequency bins for each rpm value. In this case however, the number of bins not proportional to the frequency of 1st order, but instead proportional to the frequency of the order being extracted. A graphical representation of all three of these bandwidth strategies is shown in Figure 2.5.

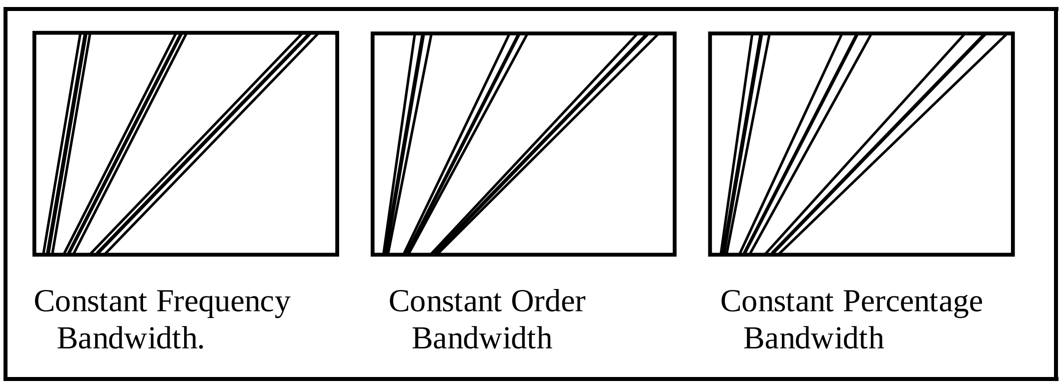

Figure 2.5: Graphical representation of different bandwidth strategies for FFT based order tracking.

Regardless of which strategy is employed to obtain the order estimates, the phase estimate must still be correlated to the tachometer signal if all of the data is not simultaneously acquired. The phase must be estimated by performing exactly the same analysis procedure on the tachometer signal that is performed on the data channels. The resulting phase of the tachometer channel is then subtracted from the order estimates of the response channels. Any channel may be used as the phase reference channel. Typically the phase reference is the tachometer channel since it has physical meaning. Using the tachometer channel as the phase reference also means that an additional channel does not need to be acquired with every acquisition for phase matching the results, the tachometer channel must be acquired for other purposes as well.

2.3.2 Fourier Transform Based Order Tracking Limitations and Errors.

The FFT based order tracking technique has limitations based on the constant time over which the transform is performed, regardless of the rpm of the machine. This constant integration time of the FFT leads to many disadvantages when the FFT is used to analyze time varying signals.

The constant frequency bandwidth of the FFT results in a varying number of orders present in each FFT spectra if the speed of the machine is changing. There will be a higher number of orders present in the spectra than at high rpm values at low rpm values. This implies that the maximum frequency of the highest order of interest must be calculated at the highest rpm of a sweep and that bandwidth used for the entire sweep.

One limitation that all Fourier transform based techniques possess is based on the assumption that all sinusoidal functions are constant in amplitude over the time that the transform is performed. The assumption of constant amplitude leads to underestimates of the amplitudes of the order if the order's amplitude varies over the integration time. The transform will give an average amplitude over the integration time. This can clearly be seen from the kernel presented in Equation 2.4. Underestimate of the amplitude can be severe if a lightly damped resonance is excited by the order. The resonance may first increase in amplitude, peak, and then decay in amplitude, during the integration time. An example of the differences in the estimated amplitude of a varying amplitude sine wave with different analysis blocksizes is shown in Table 2.1 for the sine wave presented in Figure 2.6.

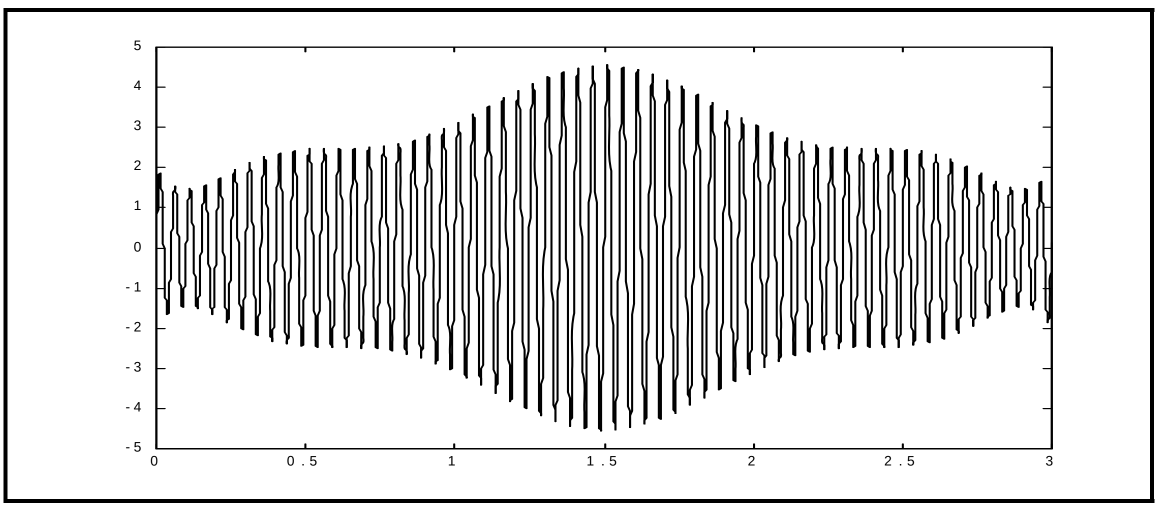

Figure 2.6: Varying amplitude sine wave.

Table 2.1 was generated by using 4 different blocksizes, all centered around the center of the block of data. It can be seen that the difference in the amplitude estimates varies from

3.8099 to 4.5476. All estimates were obtained using a Hanning window to minimize leakage.

BLOCKSIZE	AMPLITUDE ESTIMATE
512	4.5476
1024	4.4850
2048	4.2923
4096	3.8099

Table 2.1: Amplitude estimate vs. Blocksize for time function shown in Figure 2.6.

A leakage error is also present in all orders that are estimated with an FFT based technique. Since the orders can vary in frequency during the integration time, the orders will not be stationary on one spectral line of the transform. There is also no provision in the FFT techniques to analyze an integer number of revolutions to minimize the leakage error. For this reason a Hanning window is normally applied to reduce leakage. The sampled waveform that results when a constant Δt sample rate is used and the frequency is allowed to vary is shown in Figure 2.7. Obviously, this sampled waveform does not resemble the form of the kernel of the Fourier transform. This leads to both the leakage and smearing problems that hinder the use of the FFT as an order tracking method.

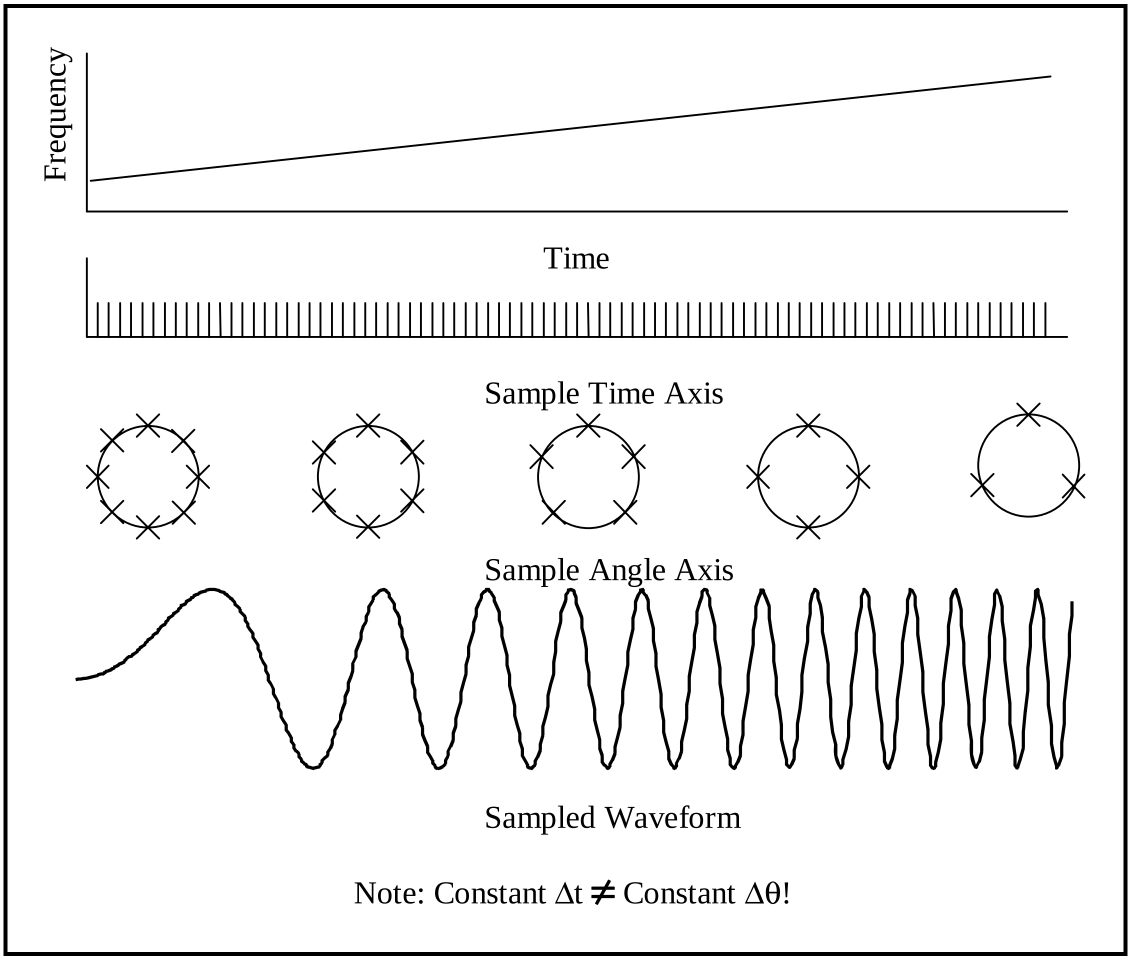

Figure 2.7: Varying frequency sine wave sampled with a constant Δt.

The varying frequency of the order relative to the constant frequency kernels of the Fourier transform can limit the sweep rate. The sweep rate and the order being analyzed control the rate at which the frequency of the order varies. If the order is to be analyzed accurately, this variation in frequency must fall within the bandwidth which is extracted from the FFT spectra. This limitation on the sweep rate can be estimated with Equation 2.5 [ref 2].

$$RPM_{sweep} = \frac{\Delta_{rpm}}{60} * T \tag{2.5a}$$

$$Binsband = \frac{RPM_{sweep} * MaxOrder * N}{F_{sample}} + W_{energy} \tag{2.5b}$$

where: RPM_{sweep} is the frequency which the primary order sweeps over in the total sample time.

Δ_{rpm} is the difference between the maximum and minimum rpm in the total sample time.
Binsband is the number of bins that the energy of the maximum order of interest is smeared over.
MaxOrder is the maximum order of interest.
W_{energy} is a correction for the energy bandwidth of the applied window (Hanning window = 1.5).

The relationships given in Equation 2.5 can then be used in conjunction with Equation 2.6 to determine the actual order resolution obtained with a defined set of sampling parameters and a given sweep rate with the Fourier transform.

$$\Delta o = \frac{(MaxOrder * 2 * Binsband)}{N} \tag{2.6}$$

where: Δo is the actual order resolution.

After determining the sweep rates and order bandwidths that can be analyzed with the Fourier transform, it becomes obvious that the Fourier transform is not ideally suited to order tracking. At low rpm values, the orders will be close together in frequency. The Fourier transform does not account for this and therefore close orders may not be easily separated. At the high rpm values, the orders are very well separated in frequency. This implies that a shorter integration time, T, may be analyzed and the orders still separated. Analyzing the data with a shorter integration time would result in a better estimate of the peak amplitude that an order contains because the averaging time of the FFT would be shorter.

2.4 Angle Domain Sampling Based Order Tracking.

The second most common order tracking methods in use in commercial software and dynamic signal analyzers are the digital resampling based order tracking methods. These methods are the digital equivalent of the analog order tracking methods that use a tracking anti-alias filter and a frequency ratio synthesizer as an external sample clock. The digital resampling based methods are implemented in a completely digital manner with only a single frequency analog anti-alias filter, and are frequently referred to as computed order tracking methods.

The first published material on this type of digital order tracking method was by Potter, et al from Hewlett Packard [ref. 5-11]. Hewlett Packard considers the exact implementation of the technique to be proprietary and as such has not published many of the details. Both the small and large channel count dynamic signal analyzers that HP manufactures have this type of order tracking available either as a standard feature or as an option. Recently many other dynamic signal analyzer manufacturers have begun to offer a type of resampling based order tracking. Again, these manufacturers consider their exact implementation to be proprietary and have not published their methods of implementation.

2.4.1 Auto-variate Discrete Fourier Transform.

From a historical perspective, the first primarily digital based angle domain based order tracking method was the Stepped Blocksize Auto-variate Discrete Fourier Transform. This method was developed to overcome many of the leakage and sweep rate limitations of the basic FFT order tracking technique.

To improve the results of analog order tracking with a frequency ratio synthesizer, Van der Auweraer et al. developed a method called the auto-variate discrete Fourier transform with stepped sampling [ref. 1]. The main principle behind the auto-variate DFT is to vary the blocksize, of the DFT, of oversampled angle domain data to ensure that the DFT transform is always performed over an integer number of revolutions.

The data acquired for the auto-variate DFT is oversampled by a large factor. A frequency ratio synthesizer is used to attempt to acquire evenly spaced angle domain sampled data. However, the sample clock that is generated from the frequency ratio synthesizer is much higher than necessary based on the highest order to be analyzed. The use of the very high sample rate is three fold. The first reason for employing a very high sample rate is to allow the use of one fixed frequency analog anti-alias filter. By oversampling low rpm data by a large factor, no aliasing will occur even if the anti-alias filter's frequency is set to allow the acquisition of an order at a much higher rpm. The second reason for oversampling the data by a large factor is to accurately estimate the arrival time of each tachometer pulse, this is necessary for the auto-variate DFT computation. The last reason the data is acquired with a high sample rate is to improve the odds of having an integer number of revolutions occur at an actual data sample instead of between samples, which will minimize the leakage problem.

The auto-variate DFT is accomplished by pre-computing the DFT kernels for various numbers of samples both above and below the target blocksize and storing them away for efficiency. Upon acquiring the number of revolutions of data required to obtain the desired order resolution, the tachometer signal pulse arrival times are re-analyzed to determine what the actual rotational speed of the machine was. This re-analysis allows the exact number of data samples to be determined which were acquired over an integer number of revolutions. The DFT is then computed over this number of data samples from the pre-computed DFT kernels. Hence, the blocksize of the DFT is adaptively changed to account for the lag inherent in the frequency ratio synthesizer. Adaptively varying the DFT blocksize minimizes leakage at the cost of the size of the required buffers to store the oversampled data prior to the DFT and the pre-computed DFT kernels.

If the blocksize of the DFT must be varied from the target blocksize, there will be a leakage error. The leakage error occurs because the data is not sampled with an equal angular spacing. Obtaining equal angular spaced samples was the goal of using a frequency ratio synthesizer. The data samples are not evenly spaced because the speed of the machine is changing in the acquisition of each block of data. If the speed had not changed, the DFT size would not need to be varied. Since the data is not sampled with an equal angular interval, the DFT will not be computed correctly. Remember that the kernel of the DFT is generated with equal angular spacing between samples. This implies that at each sample instance the phase of the order of interest and the phase of the DFT kernel will vary with respect to each other by a small amount. The sample by sample phase variation between the order and the kernel causes a leakage error. The leakage error will be much smaller than if the auto-variable DFT is not employed.

This approach will especially increase the accuracy of order estimates that are near the Nyquist order and all order estimates with high sweep rates where the lag of the frequency ratio synthesizer is most pronounced.

2.4.2 Digital Resampling Based Order Tracking Theory.

The resampling based order tracking methods acquire digital data with a uniform Δt after anti-alias filtering with a single frequency analog anti-alias filter. The uniformly sampled time data is then digitally resampled to the angle domain through the use of an adaptive digital resampling algorithm. These adaptive digital resampling algorithms are what is considered proprietary by the dynamic signal analyzer manufacturers. A summary of the steps required to do this angle domain transform and processing are shown in Figure 2.8.

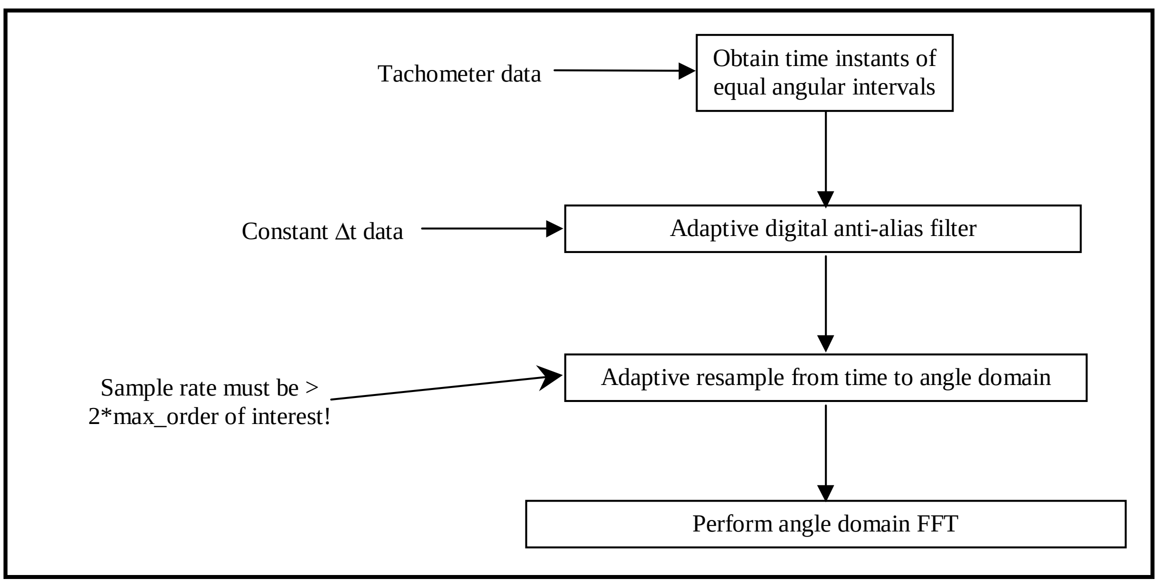

Figure 2.8: Flow chart of adaptive resampling process.

The philosophy behind adaptively resampling the uniform Δt data to a uniform angular interval, $\Delta\theta$, is given in Appendix A. The uniformly spaced angle data is then processed using the Fourier transform to obtain amplitude and phase estimates of the orders of interest. The significance of angle domain sampling is that this data has the same properties as a stationary sine wave sampled with uniform time intervals. The adaptive resampling from the time domain to the angle domain transforms non-stationary time domain data into stationary angle domain data that can then be analyzed with standard digital signal processing methods. The result of resampling a varying frequency sine wave is shown in Figure 2.9. This figure shows that the chirp function, which is shown in Figure 2.7 sampled with a uniform Δt, appears to be a sine wave after angle domain resampling. This function now matches the form of the Fourier transform kernel, which implies a more accurate analysis of the signal is possible.

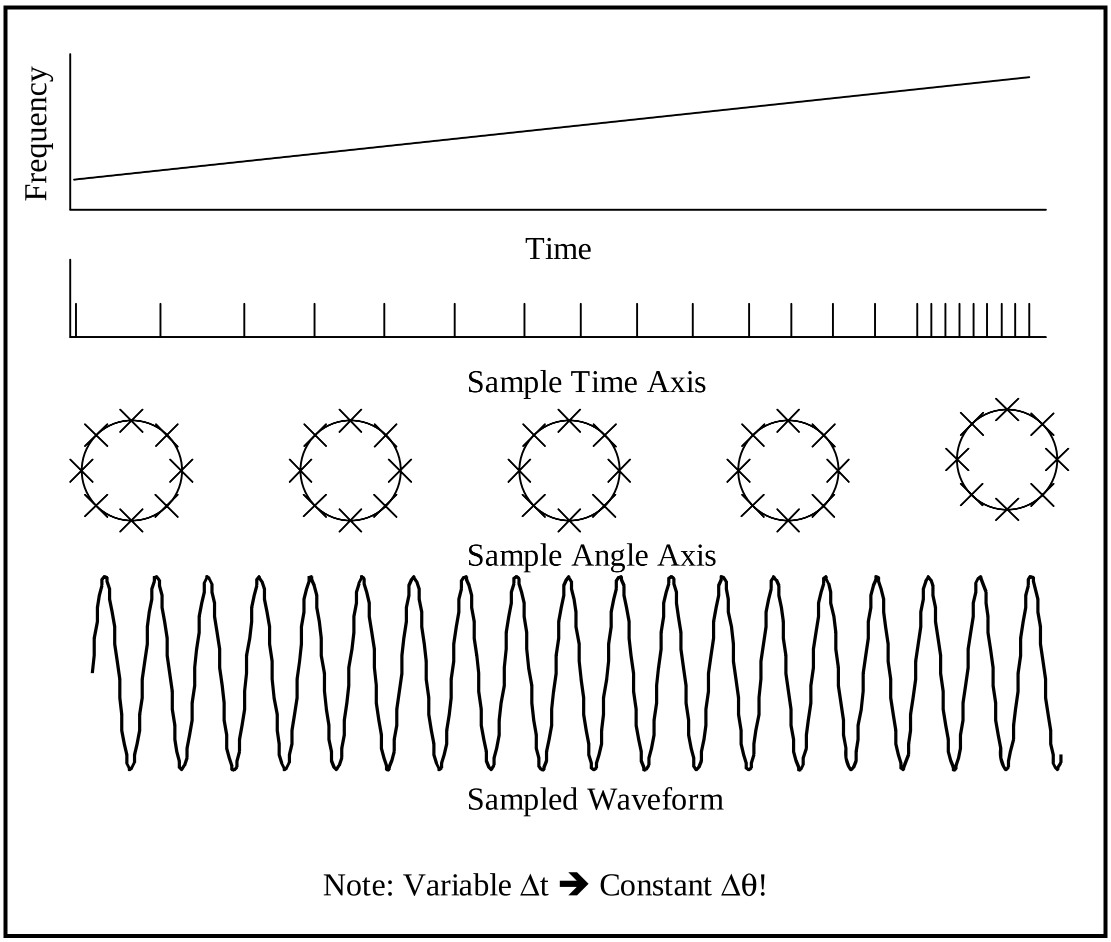

Figure 2.9: Chirp function resampled to angle domain.

To enable the transformation from the time domain to the angle domain, a reference signal of some type must be measured to allow the determination of the times of the uniform angular intervals. This reference signal is typically a tachometer signal measured on a reference shaft of the operating machine. Through the use of numerical integration and an interpolation algorithm, this tachometer signal can be processed to obtain the time instants of the uniformly spaced angular points. This tachometer processing is explained in detail in Section 2.7.

The time instants of the angular intervals are then used with a digital interpolation algorithm to obtain the estimates of the response channels at these points in time. Several different interpolation algorithms may be used to estimate these new data samples. Several of these techniques are summarized in Section 2.8.

The angle domain data is then analyzed with the use of angle domain Fourier transforms to obtain amplitude and phase estimates of the orders. The time domain sampling relationships must be reformulated into the angle domain to understand the sampling and resolution of these Fourier transforms. The reformulated sampling equations are given in Equation 2.7. Further discussion of the development of this relationship is given in Appendix A.

$$\begin{aligned}
&\Delta o = \frac{1}{R} = \frac{1}{N * \Delta\theta} \\
&R = N * \Delta\theta \\
&O_{nyquist} = O_{\max} = \frac{O_{sample}}{2} \\
&O_{sample} = \frac{1}{\Delta\theta}
\end{aligned} \tag{2.7}$$

where: Δo is the order spacing of the resulting order spectrum.
R is the total number of revolutions that are analyzed.
N is the total number of time points over which the transform is performed.
$\Delta\theta$ is the angular spacing of the resampled samples.
O_{sample} is the angular sample rate at which the data is sampled.
$O_{nyquist}$ is the Nyquist order.
O_{max} is the maximum order that can be analyzed.

As can be seen in Equation 2.7, there are analogous quantities for each of the time domain sampling parameters. In the angle domain, the order resolution is related to the number of revolutions that the machine turns through over the transform period. It should also be noted that the minimum sampling rate needed to avoid aliasing is two samples per cycle of the highest order of interest, the same as that required with time domain data.

The kernels of the Fourier transform are also reformulated in terms of the uniform angular intervals. These kernels are presented in Equation 2.8.

$$
\begin{aligned}
a_m &= \frac{1}{N}\sum_{n=1}^{N} x(n\Delta\theta)\cos(2\pi o_m n\Delta\theta) \\
b_m &= \frac{1}{N}\sum_{n=1}^{N} x(n\Delta\theta)\sin(2\pi o_m n\Delta\theta)
\end{aligned}
\tag{2.8}
$$

where: o_m is the order which is being analyzed, mΔo.
a_m is the Fourier coefficient of the cosine term for o_m.
b_m is the Fourier coefficient of the sine term for o_m.

The result from this angle domain Fourier transform is that the orders fall on spectral lines, regardless of the speed variations over which the transform is applied. The transform is usually applied as a discrete Fourier transform (DFT) instead of an FFT for computational speed and flexibility. The DFT allows the transform to be performed for only the orders of interest. Another advantage of using the DFT is the fact that the transform can be applied over the number of points which corresponds to an integer number of revolutions of the machine's rotation, as opposed to being restricted to a power of two samples. Since the data is sampled in the angle domain there is a constant number of orders present in the data regardless of the rotational speed of the machine.

The advantages of the resampling based order tracking are leakage free estimates of orders which fall on spectral lines as well as an order resolution which is of constant order width. The constant order resolution is beneficial because the integration time is shorter at higher rpm values. This property is desired since orders will change frequency more rapidly at higher rpm values and therefore are more likely to change amplitude more rapidly as well. In addition, the order resolution is now independent of rpm. This allows closer orders to be analyzed at low rpm values where they are close together in frequency.

2.4.3 Digital Resampling Based Order Tracking Limitations and Errors.

Resampling based order tracking, while being more accurate than the FFT based order tracking, still has limitations and errors associated with it.

An obvious limitation with any FFT is the finite defined order resolution. This presents a problem if there are orders present that do not fall on spectral lines. These orders are difficult to analyze with an FFT approach. Practically, the Fourier transform for a spectral line on each side of the actual order can be calculated and these results summed. This approach compromises the order resolution. If the DFT is used to analyze an order that does not fall on a spectral line, the transform will not conserve energy! Energy is not conserved because the kernel of this order will not be orthogonal with the kernels of the orders that fall on spectral lines.

The limitation of the FFT that was discussed in Section 2.3.2, the analysis of non-stationary amplitude data, is still present with the angle domain Fourier transform. The resampling process does nothing to overcome this limitation other than allow the transform to be applied over a shorter period of time at the higher rpm values. The transform is applied over a shorter time at higher rpm values because a shaft requires less time to rotate through the desired number of revolutions.

The finest order resolution that may be analyzed in real time is typically limited by the amount of memory that is present in the acquisition system. To implement the digital resampling in real time the acquisition system must buffer not only the raw time sampled data but also the resampled angle domain data prior to performing the desired DFTs. This implies that the memory buffer must be large since it must also contain the time data which has not been digitally anti-alias filtered. In some cases the limitation may be the speed of the DSP processors in the acquisition cards. This limitation is fading quickly with the increased speed of the latest DSP processors.

Another restriction with resampling based order tracking is that orders may only be tracked relative to one rotating shaft. This restriction also implies that orders that cross one another can not be analyzed accurately. Orders which cross another may be generated by two different rotating components which are not rigidly coupled to one another. This may occur across a torque converter in an automatic transmission or in a constant variable transmission. To separate the crossing orders or orders that are not relative to the same shaft requires the use of multiple tachometer signals, one for each independent input to the system. Resampling based order tracking can not use multiple tachometer signals to estimate the response of the system due to the multiple inputs present.

It will be seen that several of the limitations and errors associated with the resampling based order tracking can be overcome by other more recently developed order tracking methods.

2.5 Kalman Filter Based Order Tracking.

An order tracking method that overcomes many of the limitations of order resolution is the Kalman filter based order tracking. The Kalman filter methods allow the extraction of the time history of the order as well as the estimate of the amplitude and phase of an order. The Kalman filter was first adapted to order tracking by Vold and Leuridan [ref. 17,18]. Since this original implementation, Vold has continued to develop more advanced filters with additional capabilities [ref. 19-21].

2.5.1 Original Kalman Order Tracking Filter.

The Kalman filter approach to estimation requires that a priori information of some type be known [ref. 31,32]. To use the Kalman filter to extract order information from data requires information about the order to be extracted. In this case, the frequency is known. The frequency of the order is estimated from processing the tachometer signal and a knowledge of what order is to be extracted. The tachometer data is processed by estimating a set of splines through the tachometer periods. These splines provide an estimate of the instantaneous frequency for each time sample. This procedure is presented in detail in Section 2.7.

The a priori information is used to formulate the structural equation of the Kalman filter. The structural equation is an equation that describes the mathematical characteristics of the order to be extracted. An equation that can be used to mathematically describe a sampled sine wave is given in Equation 2.9. This equation is the structural equation used in the original second order Kalman order tracking filter.

$$x(n\Delta t) - 2\cos(\omega\Delta t)x((n-1)\Delta t) + x((n-2)\Delta t) = 0 \tag{2.9}$$

where: $x(n\Delta t)$ is the n^{th} discrete time sample.
ω is the instantaneous frequency of the sine wave.

Equation 2.9 describes a sine wave whose frequency and amplitude is constant over three consecutive time points. The frequency of an order is allowed to vary with time, which implies that the frequency of the sine wave is not constant. The structure equation is then re-written to account for this, shown in Equation 2.10.

$$x(n\Delta t) - 2\cos(\omega\Delta t)x((n-1)\Delta t) + x((n-2)\Delta t) = \varepsilon(n) \tag{2.10}$$

where: $\varepsilon(n)$ is the nonhomogeneity term.

The nonhomogeneity term is used to describe the amplitude and frequency variations from a perfect sine wave. Mathematically, if a sine wave is amplitude modulated there must be other frequency components present in the data. These additional frequencies are sidebands which allow the amplitude of the sine wave to change with time. If the amplitude is to change quickly, then more frequency information must be allowed to pass through the filter and the nonhomogeneity term must be larger. The standard deviation of this nonhomogeneity term is defined as $s_\varepsilon(n)$.

The second equation that the Kalman filter is based on is the data equation. The data equation describes the relationship between the order, x(n), and the measured data, y(n).

The measured data contains not only the order of interest but all orders generated by the machine and random background noise. This equation is written in Equation 2.11.

$$y(n) = x(n) + \eta(n) \tag{2.11}$$

where: $\eta(n)$ is the nuisance component.

The nuisance component, $\eta(n)$, is the portion of the signal containing the non-tracked orders and random noise. The standard deviation of this term is denoted $s_\eta(n)$. If the nuisance term is large it indicates that a significant portion of the measured signal, y(n), is attributable to non-tracked orders and random noise.

The structure and data equations are combined into a set of linear equations to solve for the amplitudes of the order of interest. A minimum of three data values are required to satisfy the structure equation and therefore to solve for the order. The structure and data equations are written in matrix form in Equation 2.12.

$$\begin{bmatrix} 1 & -2\cos(\omega\Delta t) & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x(n-2) \\ x(n-1) \\ x(n) \end{Bmatrix} = \begin{Bmatrix} \varepsilon(n) \\ y(n) - \eta(n) \end{Bmatrix} \tag{2.12}$$

A weighted solution to this problem is formed by ratioing the standard deviations of the structure and data equations. This ratio is shown in Equation 2.13.

$$r(n) = \frac{s_\varepsilon(n)}{s_\eta(n)} \tag{2.13}$$

Applying the ratio as a weighting function to the solution of the matrix problem results in Equation 2.14.

$$\begin{bmatrix} 1 & -2\cos(\omega\Delta t) & 1 \\ 0 & 0 & r(n) \end{bmatrix} \begin{Bmatrix} x(n-2) \\ x(n-1) \\ x(n) \end{Bmatrix} = \begin{Bmatrix} \varepsilon(n) \\ r(n)(y(n) - \eta(n)) \end{Bmatrix} \tag{2.14}$$

Assuming an isotropic error that is consistent with a minimum variance unbiased solution with locally zero mean error terms gives Equation 2.15. Assuming a different error model will alter the tracking characteristics of the filter.

$$\begin{bmatrix} 1 & -2\cos(\omega\Delta t) & 1 \\ 0 & 0 & r(n) \end{bmatrix} \begin{Bmatrix} x(n-2) \\ x(n-1) \\ x(n) \end{Bmatrix} = \begin{Bmatrix} 0 \\ r(n)y(n) \end{Bmatrix} \tag{2.15}$$

Equation 2.15 can be expanded into a least squares solution if multiple estimates of the order amplitude and phase are of interest, this formulation is used to extract the time history of an order. The least squares formulation shown in Equation 2.16 will solve for m points of the order time history.

$$\begin{bmatrix} 1 & -2\cos(\omega_n\Delta t) & 1 & & & & & \\ & 1 & -2\cos(\omega_{n+1}\Delta t) & 1 & & & & \\ & & 1 & -2\cos(\omega_{n+2}\Delta t) & 1 & & & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & 1 & -2\cos(\omega_{n+m}\Delta t) & 1 \\ & & r(n) & & & & & \\ & & & r(n+1) & & & & \\ & & & & r(n+2) & & & \\ & & & & & \ddots & & \\ & & & & & & r(n+m) \end{bmatrix} \begin{Bmatrix} x(n-2) \\ x(n-1) \\ x(n) \\ x(n+1) \\ x(n+2) \\ \vdots \\ x(n+m) \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ r(n)y(n) \\ r(n+1)y(n+1) \\ r(n+2)y(n+2) \\ \vdots \\ r(n+m)y(n+m) \end{Bmatrix} \tag{2.16}$$

Normally, the least squares formulation is formulated to solve for all points of a time history simultaneously. This formulation is technically a Kalman smoothing algorithm, as opposed to a filtering algorithm, because it can use time points both before and after the desired time point to obtain the order estimate [ref 31,32].

Equation 2.16 can be solved through a normal equations approach, which will result in a tightly banded diagonal matrix that must be inverted. This inversion can be done with either a forward/backward substitution approach or a direct matrix inversion.

The choice of the weighting factor, r(n), which is the inverse of the Harmonic Confidence Factor(HCF) in the LMS implementation of this Kalman filter, defines the tracking characteristics of the filter.

Choosing a relatively high value for the HCF weights the structure equation more heavily in the solution process. Weighting the structure equation heavily results in a filter shape which is very narrow and allows very little sideband information to pass through the filter. Without sideband information, the amplitude of the filtered sine wave must change very slowly. A relatively high HCF then gives a very sharp filter with very good frequency discrimination. The tradeoff for this performance characteristic is a filter that does not allow the filtered order to change amplitude quickly.

Choosing a relatively low value for the HCF has the effect of weighting the data equation more heavily in the solution process. Weighting the data equation more heavily results in a filter that does not possess as sharp a rolloff and therefore is not as frequency discriminating as the high HCF filter. Using a low HCF allows the amplitude of the filtered order to vary much more quickly than the high HCF filter. This behavior may be necessary around lightly damped resonances or in fast speed sweeps where the frequency and amplitude of the order must be allowed to vary quickly.

Figure 2.10 shows the filter shape when tracking different normalized frequencies. While the filter shape looks very similar from one frequency to another, the same HCF was used for all frequencies, the filter's peak amplitude changes as a function of frequency. A Flattop window was used to generate this plot in an attempt to eliminate any leakage from the results. It can be seen that the filter's amplitude is lowest at the center of the block and highest at both ends.

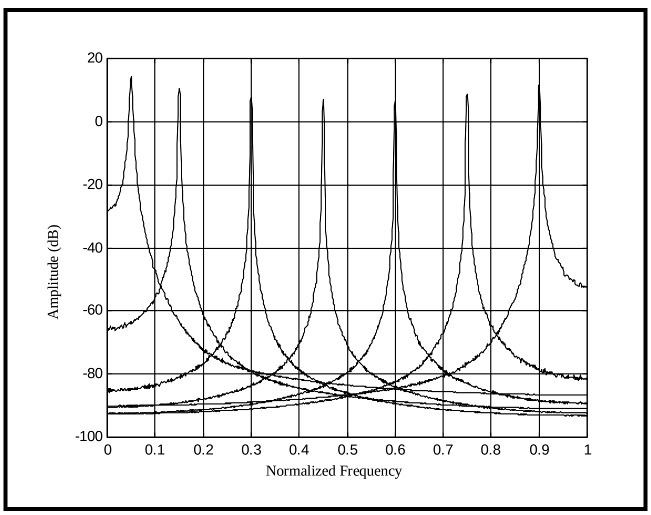

Figure 2.10: Effect of tracked frequency on original Kalman filter's shape.

The amplitude convergence time tradeoff with the HCF is shown in Figure 2.11. A sine wave was generated that was set to 0 amplitude near the center of the time block to generate a signal with a severe amplitude change. This waveform was tracked with two different HCF values, 100 and 1600, to show the effect of HCF on tracking amplitude changes. The higher the HCF the slower the filter is to converge to the correct amplitude of the waveform. This is consistent with the necessity of requiring sideband energy to allow the amplitude of the waveform to change quickly.

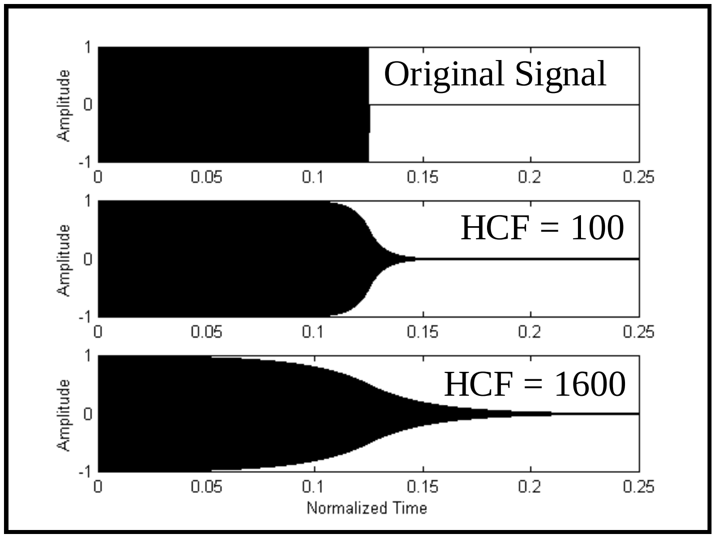

Figure 2.11: Effect of HCF on ability to follow rapid amplitude change, HCF=100, 1600.

2.5.2 Vold-Kalman Order Tracking Filter.

Vold both simplified and extended the original Kalman order tracking filter into the Vold-Kalman order tracking filter [ref. 19]. This extended filter can be formulated with different numbers of poles to alter its bandpass characteristics. The filter may also be applied in either an iterative or direct solution to separate the contributions of very close or crossing orders.

Vold realized that the second order formulation for the structure equation, presented in Equations 2.9 and 2.10, could be written in an equivalent complex first order form. The complex formulation for the structure equation is shown in Equation 2.17.

$$x(n+1) - x(n)\exp(i\omega\Delta t) = \varepsilon(n) \tag{2.17}$$

The exponential term in this equation is the angle that the order rotates through in one time sample, Δt. If the amplitude of the order does not change in the sample period, $\varepsilon(n)$, the nonhomogeneity term, will be 0. This form of the structure equation has several advantages over the original second order formulation. The data may be pre-multiplied by the exponential function, or phasor, which varies with time. The phasor's frequency exactly matches the frequency of the order at all time values. This pre-multiplication is shown in Equation 2.18.

$$y_{DC}(t) = y(t)\exp\left(-ik\int_0^t \omega(u)du\right) \tag{2.18}$$

where: $y_{DC}(t)$ is the frequency shifted time history.
k is the order of interest.

Equation 2.18 shows the phasor in time-varying form. Multiplying the data by the time-varying phasor centers the order of interest about DC. This operation simplifies the structure equation to the form shown in Equation 2.19.

$$x(n+1) - x(n) = \varepsilon(n) \tag{2.19}$$

This simplified structure equation for the order is simply a relationship where ε(n) represents the amplitude change of the order from one time sample to the next. The order is then simply a DC amplitude profile, in other words, the extracted order at this phase is the demodulated complex amplitude profile.

One key advantage to this form of the structure equation is that frequency is not present in the equation and therefore it is obvious that there is absolutely no frequency or slew rate limitations.

The Vold-Kalman filter uses the same data equation as the original Kalman filter, except in this case the data, y(n), is actually the phasor shifted data, $y_{DC}(n)$. This filter also uses the same definition for the weighting function and error terms. The matrix formulation of the structure and data Equations is shown in Equation 2.20.

$$\begin{bmatrix} 1 & -1 & \\ & 1 & -1 \\ r(n) & & \\ & r(n+1) & \end{bmatrix} \begin{Bmatrix} x(n) \\ x(n+1) \\ x(n+2) \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ r(n)y_{DC}(n) \\ r(n+1)y_{DC}(n+1) \end{Bmatrix} \tag{2.20}$$

The normal equations form of the matrix equation is a tri-diagonal set of equations which may be solved very efficiently using a forward/backward substitution method. If a forward/backward substitution method is used for solution, further computational advantages can be gained if the same weighting factor is used for either multiple orders or multiple channels of data [ref. 33,34]. The forward substitution portion of the solution is independent of the data and relies only on the normal equations matrix of the left hand side of the equation. The forward substitution may then be performed and stored for repeated application of the backward substitution to solve for the multiple orders or channels of data. This computational advantage means that for additional orders or channels after the initial order or channel, the number of calculations necessary to solve for the orders is reduced by a factor of 2!

The order amplitude/phase profile that is estimated by Equation 2.20 is not a time-varying phasor that represents the order, as was estimated by the original Kalman order tracking filter. This first order formulation will estimate the amplitude/phase envelope that must be re-modulated to remove the tracked order from the original data. Re-modulation is performed by multiplying the estimated amplitude/phase profile by the negative time-varying phasor in Equation 2.18, used to de-modulate the original data. The estimate is only the positive frequency component of the order. The negative frequency component must also be accounted for. The complete re-modulation equation is shown in Equation 2.21 for both the positive and negative frequency order components.

$$\tilde{x}(t) = 2 * \mathrm{Real}\left(x(t) \exp\left(ik \int_0^t \omega(u) du \right) \right) \tag{2.21}$$

where: $\tilde{x}(t)$ is the re-modulated order function.

This re-modulated order function can be subtracted from the original time history of the data to analyze the data without the presence of the extracted order.

The effect of tracked frequency with the Vold-Kalman filter is shown in Figure 2.12. It can be seen that tracked frequency has no effect of the filter shape or amplitude, this is opposed to the original filter formulation that possessed a different amplitude at different frequencies. Independence of tracked frequency relative to the sample rate is a large improvement in the usefulness of the filter, this allows a weighting factor to be chosen independent of the frequency tracked.

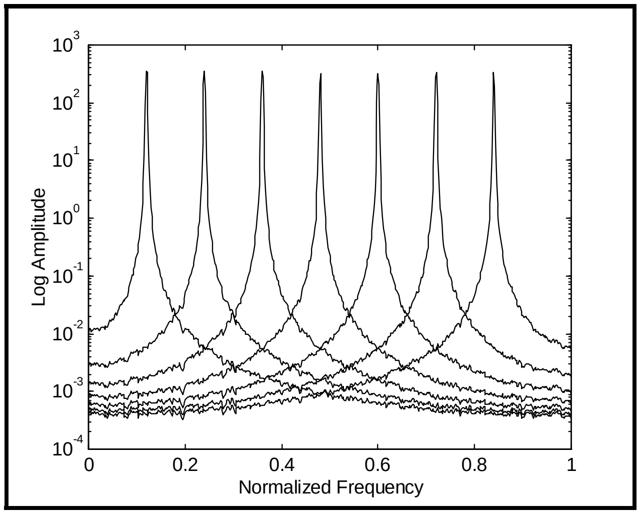

Figure 2.12: Effect of frequency on Vold-Kalman filter.

Figure 2.13 shows the effect of different weighting factors on the shape of the Vold-Kalman filter. The higher the weighing factor the better the sideband suppression and the slower the amplitude of the tracked order will vary. Weighting factors of 100, 400, 700, 1000, 1300, 1600, and 1900 were used to create this figure. Note that regardless of the weighting factor the amplitude at the center frequency is identical.

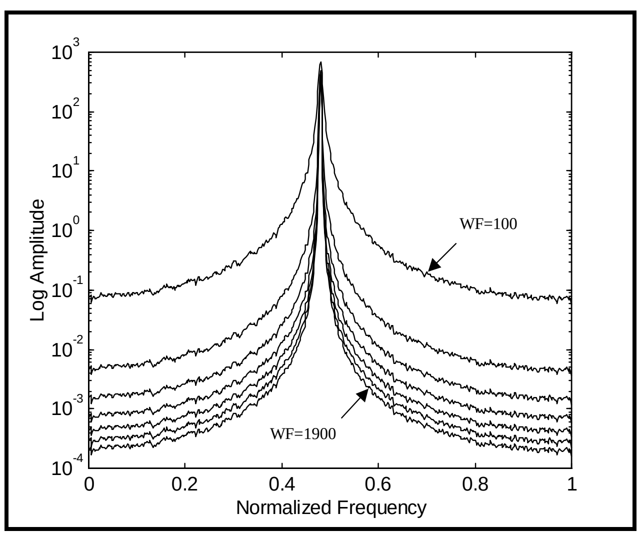

Figure 2.13: Effect of weighting factor on Vold-Kalman filter shape, wf=100, 400, 700, 1000, 1300, 1600, 1900.

An additional capability of the Vold-Kalman filter is the ability to reformulate the filter as a higher order filter in order to have a broader passband region while improving the sideband rejection. This filter has a flatter top relative to the single point peak which the first order filter possesses. The penalty paid for these improved filter characteristics is computational complexity. A higher order formulation requires the solution of a more heavily banded matrix as opposed to the first order's tri-diagonal matrix. For example, a 2^{nd} order formulation has a banded matrix with 5 fully populated diagonals. The actual formulation of the higher order filters is considered to be proprietary by Vold.

While the higher order filter formulations can possess much sharper passbands, they may not be the filter of choice for very close orders because of their broader top. They are more forgiving if an imperfect tachometer signal was measured. The higher order filters should also be used if there are high amplitude orders that are close to the tracked order but outside of its passband. The sharper filter skirts will allow these other orders to be more effectively removed from the tracked order.

Probably, the most important characteristic which has been added to the order tracking analysis capability by the Vold-Kalman filter is the ability to separate either very close or crossing orders. The ability to separate interacting orders may be implemented through either an iterative solution or a direct solution. The formulation and solution of these filters are considered to be proprietary by Vold. This ability is commercially available in multiple software packages.

2.5.3 Kalman/Vold-Kalman Applications and Realizations.

The formulations and discussions of both the original and the Vold-Kalman filters are centered about a weighting factor that is constant as a function of time. This is not a requirement of either filter's formulation. In fact, all commercial implementations of either filter allow the weighting factor to vary as a function of time or rpm. If the weighting factor is varied as a function of rpm, a pseudo-constant order bandwidth filter may be obtained. Another strategy to vary the weighting factor of the filter is based on instants of known transient activity in the data. Examples of this transient activity are gear shifts or clutch engagements. In these areas of transient activity, it is assumed that the amplitude of a tracked order may change very quickly, thus the weighing factor is reduced in these regions to allow more sideband energy to pass through the filter. The additional sideband energy allows the amplitude of the order to change very quickly in these regions, allowing a realistic amplitude profile to be estimated.

The Vold-Kalman order tracking filter, if implemented with the steps discussed above, allows *what-if games* to be played that are more difficult with the original filter implementation. The most powerful of these capabilities is that the estimated amplitude/phase profile at DC may be modulated to its original frequency. This order can then be subtracted from the original time history. The next step that can be taken is to modulate the amplitude/phase profile at DC to a different order than it was originally. The modulated order can then be added into the time history without the original order. This process has applications, for example, in accessory drive evaluations. An order relative to an accessory can be extracted, modulated to a different order, and added back into the data. Modulating the order to a different order in this case would approximate using a different pulley diameter to drive the accessory. This process can then be used to assess potential order interactions between accessories. This process, while not perfect, can significantly reduce the hardware testing time to assess these interactions. The resulting time histories can be played back through a set of headphones for sound quality analysis.

Another application that these filters may be used for is the extraction of either constant frequency or order components that have been frequency shifted by the Doppler effect. This situation is very common in passby noise applications. The Doppler effect is compensated for by including its frequency shifting effect in the estimate of the instantaneous frequency used in the structure equations. Realizing this capability opens up many analysis possibilities that have historically been avoided due to the complexity and difficulty in accounting for the Doppler shift.

Obviously, these filters may also be used to filter constant frequencies out of data. This capability can be applied to remove 60 cycle electrical noise from time domain data. The very sharp frequency characteristics of these filters coupled with the steady state amplitude nature of electrical noise makes these filters ideal for this type of application as the filters can be made very sharp and the frequency of the electrical noise is very well controlled. Any application of these filters requires that the frequency of the filtered component be very well estimated. It is very easy to miss the frequency component of interest if the frequency is at all in error.

The best frequency discrimination that can be expected from the Kalman filters is the same as that of the Fourier transform. Frequencies closer together than the inverse of the total length of time of the data cannot be effectively separated, as defined by Rayleigh's criteria.

While the Kalman order tracking methods have many advantages over the traditional order tracking methods, including better dynamic range and the time domain order extraction, they do have some disadvantages. Computational complexity is one disadvantage of the adaptive filters. The largest disadvantage, however, is the experience required to get valid accurate results for each order extraction. The experience is required in choosing the appropriate weighting factor to extract the order with a minimal bandwidth while tracking the amplitude profile accurately. The weighting factor that is necessary for an accurate extraction is a function of the other orders present in the data, the sweep rate, and the properties of excited resonances. All of these items, which can vary from channel to channel, change the rate at which the amplitude of an order changes and therefore affect the width of the filter necessary for an accurate extraction.

2.5.4 Kalman Filter Weighting Factor Determination.

In any formulation of the Kalman or Vold-Kalman filter the weighting factor must be determined. Choosing a weighting factor with either filter that heavily emphasizes the structure equation results in a very frequency selective filter. The question that remains to be answered then, is what weighting factor weights the structure equation enough. This question is not easily answered, through experience with the filters it has been learned that there is no one right answer. No mathematical relationship between a filter weighting factor and some function of the data has been derived which will always result in the desired filter characteristic. It has been observed that the weighting factor is a function of many parameters.

Parameters that effect the filter performance include the sample rate relative to the frequency of the order of interest, the amplitude of the order of interest relative to the amplitudes of the other orders present in the data. This amplitude characteristic depends on whether the other orders have relatively large/small amplitudes close in frequency to the order of interest or well away from the frequency of the order of interest. Amplitude effects are also present from the order of interest exciting resonances. In essence, the weighting factor appears to control the convergence speed of the filter in terms of time samples as much as it controls the bandwidth of the filter. Due to the very large number of parameters which effect the proper choice of the weighting factor it is not possible to determine a reliable method of choosing a weighting factor which will work for every dataset. The weighting factor very often varies from channel to channel of data, even if it was acquired simultaneously, due to the presence of resonances and orders that are different from between measurement dofs. In short, the only reliable way to choose the proper weighting factor is through experience. This ultimately is the major disadvantage of using the Kalman based tracking filters.

2.6 Frequency Response Function Based Order Tracking.

A very simple yet powerful order tracking technique which is not commercially implemented was used by Weber to track orders on a vehicle traveling down a road [ref. 25]. This method is based upon the single/multiple input H1 frequency response function estimator where the tachometer signals are assumed to be the inputs to the system. The computed FRFs are considered to be order tracks.

Using the tachometer signals as the inputs to the system allows the phase relationship between any rotating components with measured tachometer signals and response dofs to be estimated. The phase relationship between an input and response is constant for a linear system. This phase relationship then allows cross-spectral averaging to be performed between the tachometer signals and the response dofs. The equation that is used to compute the FRFs, or in this case order tracks, from the averaged cross-spectra is shown in Equation 2.22 and is the multiple input H1 frequency response function estimator.

$$[H(\omega)] = [G_{XF}(\omega)][G_{FF}(\omega)]^{-1} \tag{2.22}$$

where: $[H(\omega)]$ is the estimated FRF matrix (order tracks).
$[G_{XF}(\omega)]$ is the crosspower matrix between the response dofs and the tachometer signals.
$[G_{FF}(\omega)]$ is the input crosspower matrix between the tachometer signals.

To allow the input crosspower matrix to be inverted, the data must be acquired such that the different inputs' phase relationships are varied from average to average. Data acquisition is done by constantly changing the speeds of the inputs relative to one another to give the necessary phase variations. The crosspower matrices are only estimated at frequencies that correspond to the orders of each tachometer instrumented component. If the tachometer signal does not contain energy at an order of interest that order cannot be estimated. Using techniques presented in Chapter 3 this limitation can be overcome and all possible orders can be estimated.

Each resulting order track must be scaled by the amplitude of its respective tachometer signal to obtain a physically meaningful amplitude. This amplitude scaling is necessary to correct for the amplitude of the tachometer signal itself. The scaling is computed as shown in Equation 2.23.

$$X(\omega) \cong H(\omega) mag(T(\omega)) \tag{2.23}$$

where: $mag(T(\omega))$ is the magnitude of the tachometer signal at frequency ω.

The magnitude of the tachometer signal is calculated from the Fourier transform of one block of the tachometer signal or from an analytical Fourier transform of a signal with the same characteristics as the tachometer signal. If the amplitude of the tachometer signal varies over the acquisition time then the amplitude of the estimated order will not be correct. The tachometer amplitude invariance is an additional restriction that this order tracking method contains that none of the other order tracking methods possess.

2.6.1 Single Input/Tachometer Formulation, Multiple Input Configuration.

Each tachometer signal may be treated as a single input to the system using Equation 2.22. Treating each tachometer signal as a single input to the system imposes specific restrictions on the data acquisition. The data must be acquired such that the frequencies of the orders of interest from each tachometer do not overlap. If the different orders of the same tachometer excite the same frequencies it is not possible to estimate the order tracks accurately.

The order tracks are estimated by performing some type of tachometer analysis to determine the approximate frequency of the order of interest for each average. This frequency information is then used to zero out all frequency bins except those around the orders of interest for each tachometer signal in the respective Fourier transforms of the tachometer signals. This zeroing is performed for each average. The zeroing is done before the necessary autopowers and crosspowers are computed. If two different rotating components excite the same frequency in a speed sweep there must be at least two averages of the order tracks at that frequency to uncouple the contributions due to each component. The two averages must have different phase relationships between the two rotating components to allow uncoupling.

Using only the first order of each rotating component can result in a scaled order track estimate relative to a known force input. The dominant force input at 1^{st} order is typically rotating unbalance. A rotating unbalance input that sweeps in speed is no different than a swept sine input to a structure. If the magnitude of the rotating unbalance is quantified, the estimated order track can be scaled such that a scaled FRF is estimated.

2.6.2 Multiple Input/Tachometer Formulation, Multiple Input Configuration.

The multiple input FRF formulation can also be solved where there are multiple inputs from each tachometer signal. This formulation allows multiple orders of one rotating component to be tracked from the same tachometer signal. Using a multiple input/tachometer allows the contributions of orders to be separated if the different orders excite the same frequencies in an acquisition period.

To formulate multiple inputs from each tachometer signal, the tachometer signal is again processed to obtain an estimate of the rotational frequency of each component. This information is then used to zero out frequency bins that do not correspond to orders of interest. The difference in this formulation is that all frequency bins except those corresponding to one order are zeroed out and the resulting Fourier transform is treated as one input. All frequency bins except those of the next order of interest are zeroed for the next input. This procedure is repeated for each order of interest. The number of resulting inputs is the total number of orders tracked for all components.

These methods have the ability to separate both close and/or crossing orders if enough averages are acquired with a suitable frequency resolution. The separation is possible because the cross-spectral averaging technique is very effective at separating responses due to independent inputs if there is a large enough phase difference between the inputs.

The major advantage of an FRF based order tracking method is the inclusion of the transient response of a system due to the rotating inputs. Other order tracking techniques often discard the transient response because they behave like or are tracking filters. A tracking filter follows the frequency of the order, therefore in a rapid speed sweep the frequency of an order may change rapidly and the transient ringing of a system may be filtered out. Filtering out the transient response often leads to an order estimate near a resonance that is too low, the energy of the ringing is not included.

The major disadvantage to the FRF based methods is the computational complexity of the multiple input formulations. The number of inputs to the system varies for each spectral line. The multiple input problem must be re-formulated for each spectral line to accommodate the varying number of inputs. The number of inputs changes as a function of frequency because the number of orders which pass through a frequency changes, low frequencies are typically excited by larger numbers of orders than high frequencies. The implementation of the FRF based methods, while not difficult theoretically requires a large amount of bookkeeping.

2.7 Tachometer Signal Processing.

Tachometer signal processing is very important in any order tracking method. In most cases, the results of the order tracking analysis are only as accurate as the tachometer signal measured and the processing used in the analysis. The resampling based order tracking, Kalman filtering, and the frequency domain order tracking previously discussed are all based directly on the tachometer signal analysis.

If a good stable tachometer signal is not acquired then the desired order tracking can not be performed on any of the response channels. Failing to acquire a valid tachometer signal then invalidates all of the channels if order tracking is to performed, for this reason a great amount of care should be taken in acquiring the tachometer signal.

2.7.1 Hewlett Packard’s Polynomial Based Tachometer Analysis.

The resampling process of the resampling based order tracking is directly linked to the tachometer signal. For this reason, a very integral part of the computed order tracking algorithm which Potter patented for HP is the tachometer processing algorithm which is included in the patent [ref. 8].

The tachometer analysis that is used for HP’s resampling method must be a real time algorithm since the resampling is done in real time on the dynamic signal analyzers (DSAs). This method fits a local quadratic polynomial to tachometer pulse arrival times. The polynomial used is shown in Equation 2.24.

$$\phi(t) = b_0 + b_1 t + b_2 t^2 \tag{2.24}$$

where: $\phi(t)$ is the angle of the shaft.
b_0, b_1, b_2 are the polynomial coefficients.
t is time.

Formulating this polynomial in matrix form to allow for the solution of the polynomial coefficients results in Equation 2.25.

$$\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ 1 & t_3 & t_3^2 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix} \tag{2.25}$$

where: t_1, t_2, t_3 are three consecutive tachometer pulse arrival times.
ϕ_1, ϕ_2, ϕ_3 are the rotational positions of the three tachometer pulses.

Solving for the polynomial coefficients involves a simple inversion of the t matrix. Once the polynomial coefficients have been solved for, an equation to solve for the time of a desired angular position may be formulated. This equation is shown in Equation 2.26. This equation is used because the resampling process requires obtaining new samples at know angular intervals.

$$t_k = \frac{1}{2b_2}\left[\sqrt{b_1^2 + 4b_2\left(\phi_k - b_0\right)} - b_1\right] \tag{2.26}$$

where: ϕ_k is the desired angular position.
t_k is the corresponding time of the desired angular position.

This equation is then solved for the time instants of the each of the resampled data points based on the desired angular spacing. Since three consecutive tachometer pulses are used to formulate and solve this equation, the equation is only used to estimate times of angular positions which fall in the center of the angular interval spanned by the three pulses. This interval is shown in Equation 2.27. Equation 2.26 is reformulated and solved for each new tachometer pulse arrival.

$$\frac{\phi_1}{2} \le \phi_k \le \frac{\phi_3}{2} \tag{2.27}$$

This assumed model for the tachometer behavior is valid for any speed sweep profile that can be approximated with a linear shaft acceleration. If the shaft acceleration cannot be approximated as linear, the model is not valid and errors will result. Errors associated with assuming a linear shaft acceleration when the shaft acceleration profile is a higher order function were evaluated by Munck [ref. 12]. The analysis presented by Munck is valid for this error. Other portions of this reference are not based on a practical implementation and should be neglected.

While this method works very for HP in their real time implementation in a DSA it is not very well suited for use in a post-processing application. HP uses a dedicated tachometer channel in the DSAs, typically an analog triggering channel that has a very high frequency clock to record the tachometer pulse arrival times. This clock is very often on the order of 1-4 MHz. Sampling the tachometer channel at a sample rate commensurate with measured response channels can oftentimes give significant errors in the tachometer pulse arrival times. These errors exist because the tachometer pulse train is sampled and not continuous. The sampling of the signal means that the occurrence of a tachometer pulse transition is not sensed until after the actual pulse transition has occurred. This error can be on the order of an entire Δt and has a random distribution. Other errors associated with digitally triggering events are discussed in reference 35.

2.7.2 Original Kalman Filter Spline Based Tachometer Analysis.

Realizing the shortcomings of the local polynomial fit for post-processed data, Vold and Leuridan developed a spline fit technique as part of the original Kalman order tracking filter [ref.17].

The spline fit method was developed to smooth the errors associated with estimating the tachometer pulse arrival times from sampled data. The method determines the tachometer pulse arrival times as best as possible from the sampled data. These pulse arrival times are then used to calculate the rotational frequency of the machine. This is straightforward since the frequency is simply the inverse of the difference between two consecutive pulse arrival times. After estimating the pulse arrival times, there exists an rpm estimate for each pair of tachometer pulse arrivals. This spline fit method can be used with any number of pulses per revolution from the tachometer signal. Very often only a single tachometer pulse per revolution is required.

The next step in the procedure is to divide the entire time history into a number of sections. A least squares spline is fit to each section of rpm estimates, with boundary conditions enforced between the sections to ensure a smooth curve.

The boundary conditions that are enforced between the sections to ensure a smooth rpm profile are based on end conditions and derivatives of the spline equation. The end conditions imposed are that the value of the spline at the end of each section is equal to the value of the spline at the start of the next section. This ensures that there are no discontinuities in the curve. The first derivatives are forced to be equal at the end of each section and the start of the next section. Forcing the first derivatives to be equal ensures that the curve will be smooth. A complete discussion of the actual spline fitting procedure can be found in any numerical methods text book including Reference 33.

Once the splines have been determined for each section, they are used to obtain an estimate of the instantaneous rpm of the machine at every sample point. These instantaneous rpm values are then used in the structure equation of the Kalman filters.

Figure 2.14 gives an example of how large an error can result from estimating tachometer periods from sampled data. Figure 2.15 gives an example of the effectiveness of the spline fitting process.

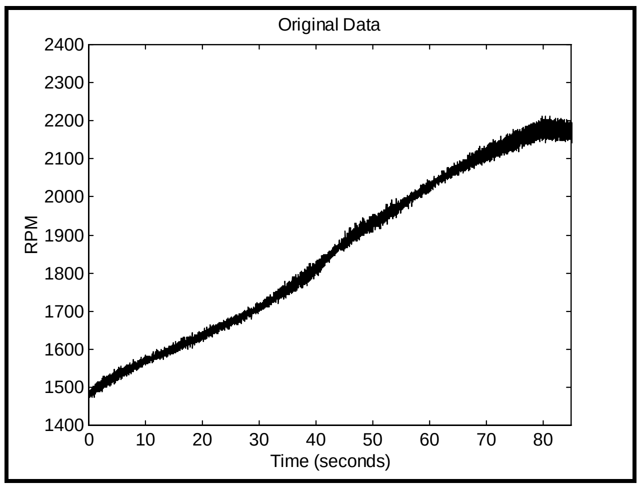

Figure 2.14: Estimated periods from sampled tachometer pulse train.

Figure 2.14 shows the estimated rpm, based on the period estimate, for each revolution of a machine based on an acquisition sample rate that had approximately 10 samples/cycle for each tachometer pulse at the highest rpm of the sweep. Obviously, there is a large variance in the rpm from one revolution to the next, which is not physically realistic. Every machine has some amount of inertia that ensures that there will not be drastic instantaneous speed changes from one revolution to the next. The effectiveness of fitting 10 spline segments to this data is shown in Figure 2.15.

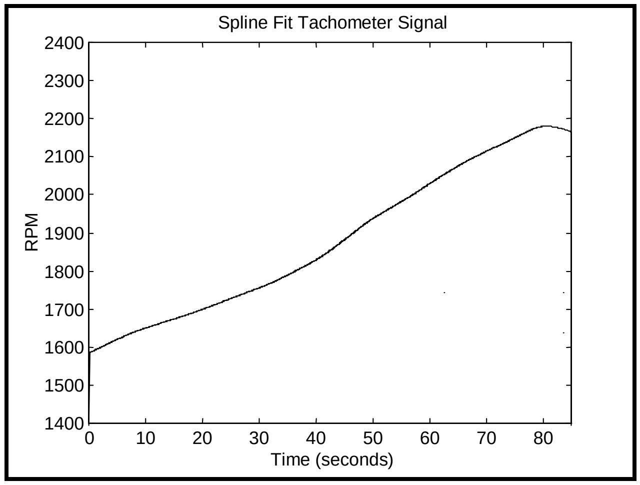

Figure 2.15: Spline fit of estimated tachometer periods.

Even with the effectiveness of the spline fit, sometimes it is necessary to do further tachometer signal processing. If the initial period estimates contain too much variance, the final spline fit may not be as smooth as desired if many spline segments are used to follow the general sweep profile. In this case, it may be possible to upsample the raw tachometer signal prior to the period estimation. Upsampling may better define the pulse transitions that are used for period estimation. Upsampling cannot increase the amount of high frequency content in the signal, but it can provide samples that are closer to the transitions and may therefore reduce the variance.

A major advantage to fitting a limited number of spline segments to an entire data block is the stiffness of the splines. If a tachometer pulse is missed or an extra pulse occurs in the tachometer signal acquisition, it may be possible to fit a spline through the signal which negates the error from the incorrect tachometer signal. However, if there are multiple missed tachometer pulses, the spline fit may not be stiff enough to effectively span the bad data. Missed or extra tachometer pulses cause severe errors in HP's polynomial fitting algorithm. There is no possibility to account for this error because all processing is done in real time and only 3 tachometer pulses are used.

A disadvantage of the stiff spline fit occurs when an aggressively changing speed sweep is measured. If the rpm of the machine is changing very rapidly, such as near gear shifts, the splines may be too stiff to accurately follow the sweep profile. To overcome this limitation, Vold expanded the spline fit concept in the Vold-Kalman filter implementation [ref. 19].

2.7.3 Vold-Kalman Filter Enhanced Spline Based Tachometer Analysis.

Two major advances are present in the tachometer processing which Vold developed/extended as part of the Vold-Kalman filter [ref. 19]. These developments more accurately fit the spline functions of the original Kalman filter tachometer processing algorithm to obtain estimated rpm values from the tachometer period estimation process.

The first of these developments allows the effects of missed or extra tachometer pulses to be effectively eliminated in the spline fit. This method, called *shaving*, fits the original spline to the estimated periods as a first step. The second step compares each originally estimated rpm value to the rpm value of the spline. If the difference between the estimated rpm and the spline rpm is too great, the estimated rpm is removed from the dataset. After comparing all estimated rpms to the spline rpms, the splines are again fitted to the reduced set of estimated rpm values. This multi-step iterative spline fit very effectively removes all effects of the missed or extra tachometer pulses from the instantaneous rpm estimates.

The second development which Vold included in the second generation spline fitting algorithm was the ability to relax the first derivative constraints. If an event channel is measured, for instance on the clutch pedal, it is possible to enforce a boundary between spline sections to occur at the time that the state of the event channel changes. For a clutch, this would allow a spline to be fit up to the shift point, through the shift, and immediately following the shift. If the first derivative constraints are relaxed at these points then the spline fit rpm profile can very accurately follow the extremely fast slew rates that occur during the shift. This capability to relax slew rates allows rapidly changing rpm profiles to be fit very well if the user accurately picks the points to relax the derivatives or if there is an event channel to identify these points automatically.

With the addition of these two enhancements to the spline fit algorithm there are very few, if any, sweep scenarios that cannot be processed accurately in the post-processing phase of analysis. However, situations still exist in real time tachometer processing implementations that cannot be handled accurately, the most obvious being the compensation for missed or extra tachometer pulses.

2.8 Adaptive Resampling Procedure.

Adaptive resampling is performed when it is desired to resample data relative to a time varying reference signal. The reference signal may be locked to the rotational speed of a non-stationary rotating machine to allow synchronous sampling. The reference signal may also be locked to the instantaneous velocity between two test objects. In this case adaptive resampling is performed to remove the Doppler shift between the test object and a stationary transducer. Many other applications exist for adaptive resampling, but these two examples are the two most common applications in noise and vibrations.

Since the reference signal can be time varying, the resulting data samples may not be evenly spaced in time but instead evenly spaced relative to the reference signal. These unevenly spaced data points are the output from adaptively resampled data samples that were evenly spaced in time when they were acquired. Obviously, since these new samples are not evenly spaced in time, the decimation and interpolation methods presented in Appendix B cannot be applied directly to obtain the new data values. However, through the creative application of one or all of these techniques, the data can be accurately adaptively resampled.

A preliminary step in implementing any adaptive sampling rate interpolation method is to determine at what points in time the desired sample points must occur from the reference signal. This may involve considerable processing and may be more computationally intensive than the actual adaptive interpolation process. The processing of tachometer signals acquired on rotating machinery is presented in Section 2.7. The tachometer processing estimates the instants in time at which new data values are required for the synchronous sampling process with respect to the rotation of a machine's rotating reference shaft.

2.8.1 Adaptive Sampling Rate Interpolation Theory Based on Upsampling and Linear Interpolation.

The adaptive sampling rate interpolation method is implemented in multiple commercially available software packages. The method is based on a combination of upsampling the original signal and linearly interpolating between the upsampled data values to obtain the desired data values.

The first step in this adaptive resampling approach, after the new time locations are determined from the reference signal, is to upsample the original data. The amount which the original data is upsampled has a large effect on the signal to noise ratio (SNR) of the final result. The more the data is upsampled, the higher the SNR of the final result. In determining how much to upsample the data, the SNR of the original measurement should be considered. Obviously, it is not necessary to upsample the data by an amount that will result in 80 dB of SNR if the original data only possesses a SNR of 60 dB. A typical upsample factor to obtain a SNR of approximately 60 dB is 16. The effects of different upsampling factors is discussed in example Section 2.8.2.

Once the data is upsampled, a linear interpolation is performed to obtain a new data value at an arbitrary instant in time as dictated by the reference signal. The upsampled data is evenly spaced in time therefore the linear interpolation is necessary to obtain values which are not evenly spaced in time. The results of the linear interpolation are the adaptively resampled data values.

Finally, if the data values are not evenly spaced in time, the lowest sample rate in the adaptively resampled data must be greater than the original sample rate of the data. If the largest Δt between new data values is greater than the original Δt, then the data will contain aliasing.

The adaptive resampling procedure is graphically represented in Figure 2.16. In this figure, an example is presented of a procedure that is used to obtain data values that are evenly spaced in the angle domain relative to the rotation of the shaft on which the tachometer signal was measured. In this case the tachometer signal is processed to generate the reference signal and the new instants in time at which it is desired to obtain samples.

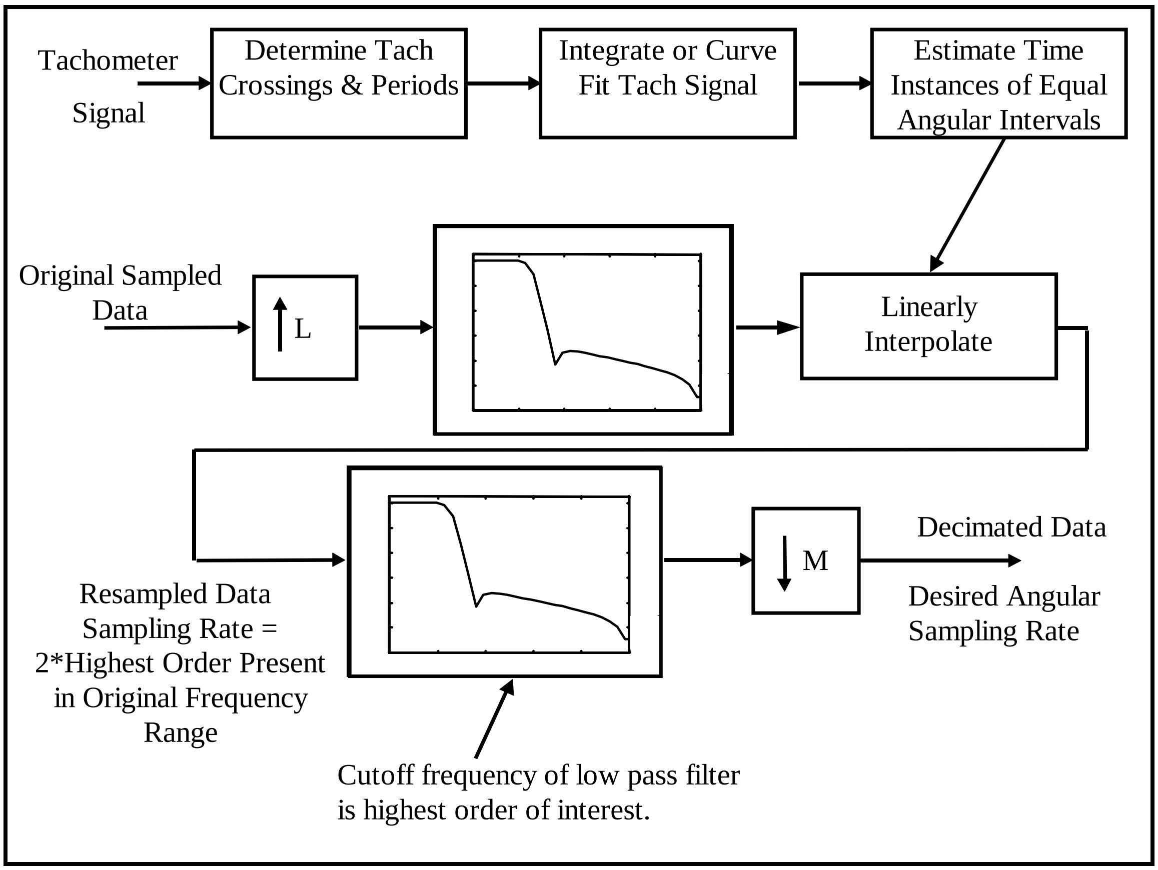

Figure 2.16: Graphical representation of adaptive sampling rate interpolation, transformation from time domain to angle domain.

The last step presented in this example that was not discussed above is the final lowpass filtering operation and decimation. This step is often performed on data that is resampled from a rotating machine if the data is acquired during a speed sweep. Acquiring data during a speed sweep implies that if the data is to be evenly spaced in the angle domain, the original sample rate was higher than necessary at the low rpm values. To prevent aliasing the data must be resampled with a sample rate that prevents aliasing at these low rpm values. This necessary sample rate may result in more samples per revolution then is desired in the end result, hence a decimation operation may be performed to obtain the final desired sample rate. It should be remembered that if a certain angular sample rate is desired which will require decimation as the final step, the data should be resampled at a sample rate which is an integer multiple of this final desired sample rate. This allows a standard decimation procedure to be used on the resampled data.

2.8.2 Numerical Example of Adaptive Sampling Rate Interpolation by Upsampling/Linear Interpolation.

This section presents a numerical example of the resampling of a chirp function whose frequency is constantly increasing across the data block. This situation is representative of data acquired on a rotating machine undergoing a speed sweep from a low rotational speed to a higher rotational speed. The data is resampled with differing amounts of upsampling prior to the linear interpolation process to show its effects.

Figure 2.17 shows both the time and frequency domain plots of the original signal. Note how the frequency increases across the time domain trace from left to right and that the frequency domain plot shows no distinct frequency component. The minimum frequency of the chirp is approximately 3 Hz, while the maximum frequency is approximately 12 Hz. A sample rate of 50 Hz was used and a Flattop window is used for all frequency domain spectra since a sine wave type function is being analyzed. The flattop window also allows the dynamic range of the algorithms to be estimated much better since the height of the window's sidelobes is down ~90 dB. Other windows that do not have sidelobes with such a large amount of rejection will limit the true signal to noise ratio analysis of the frequency domain signal. The sidelobe rejection of these other windows will dictate the amount of signal to noise rejection that can be observed in the frequency domain. Essentially, the signal to noise rejection of the window may be less than that of the algorithm under scrutiny.

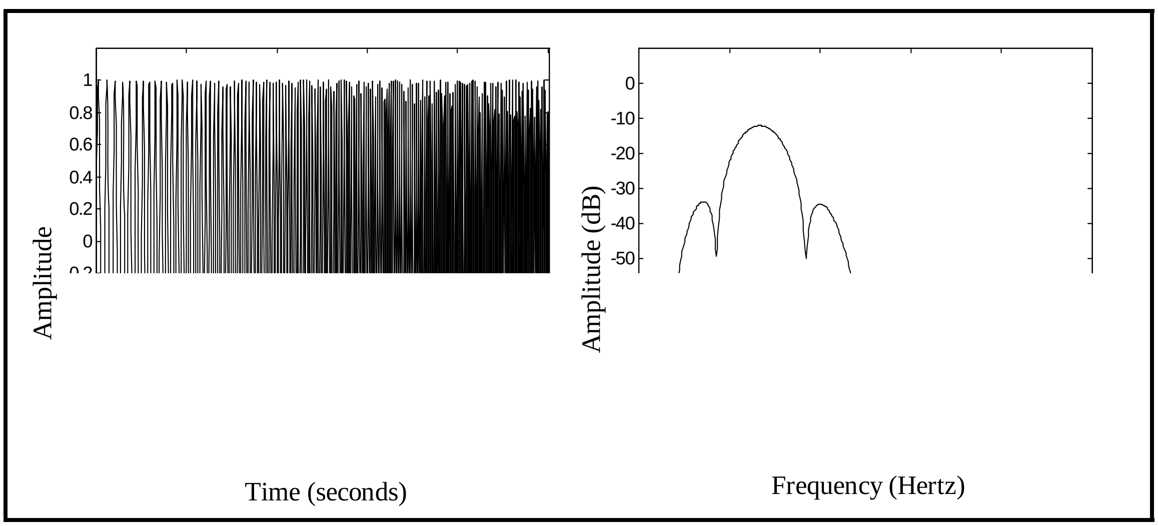

Figure 2.17: Original chirp signal, time and frequency domain representations.

Figure 2.18 shows the time domain representations of both the original and resampled signals. The signal was adaptively resampled with a sample rate of 5 samples/cycle of the chirp signal. The left hand plot shows the time signals at the beginning of the time block, the original signal marked with the symbol "*o*" and the resampled signal marked with the symbol "+". Note that the original signal contains many more samples per cycle at the beginning of the time block than the resampled signal and that the original samples are not evenly spaced across a cycle of the waveform. Later in the time block, shown in the right hand plot, the resampled signal contains nearly the same number of samples per cycle as the original signal. The sample rate of the resampled signal is constant across the entire time block with respect to samples/cycle while the sample rate of the original signal is constant with respect to samples/second. The consistent samples/cycle sample rate is also apparent from the fact that in all cycles of the resampled waveform, the samples fall at the same location of each cycle. This property can be verified by comparing the minimum and maximum values of the resampled waveform at both the beginning of the time block and further into the time block. These values are the same in all cycles.

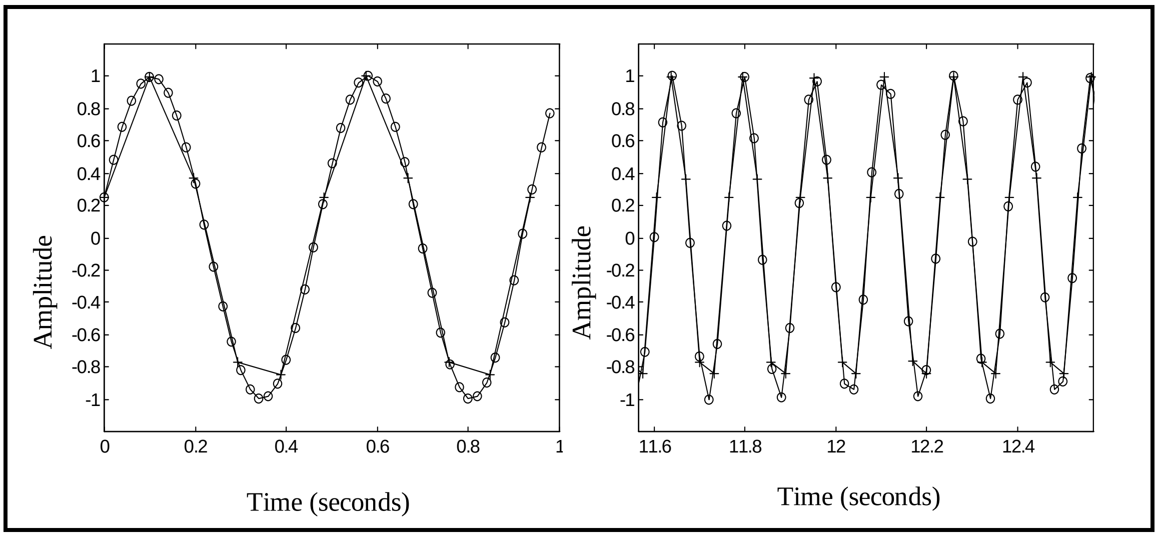

Figure 2.18: Original and resampled time histories, beginning and middle of time block.

The frequency domain representation of the adaptively resampled signal is shown in Figure 2.19. The left hand figure was adaptively resampled after upsampling the original signal by a factor of 2 prior to the linear interpolation. The right hand plot was adaptively resampled after upsampling the original signal by a factor of 16. Since the original chirp did not have frequency content greater than approximately ½ of the Nyquist frequency, this upsampling factor gives approximately 64 samples/cycle of the highest frequency of the chirp. Observe that the additional upsampling improves the SNR of the final result by approximately 10 dB. This improvement in SNR comes at the expense of considerable increased computational load.

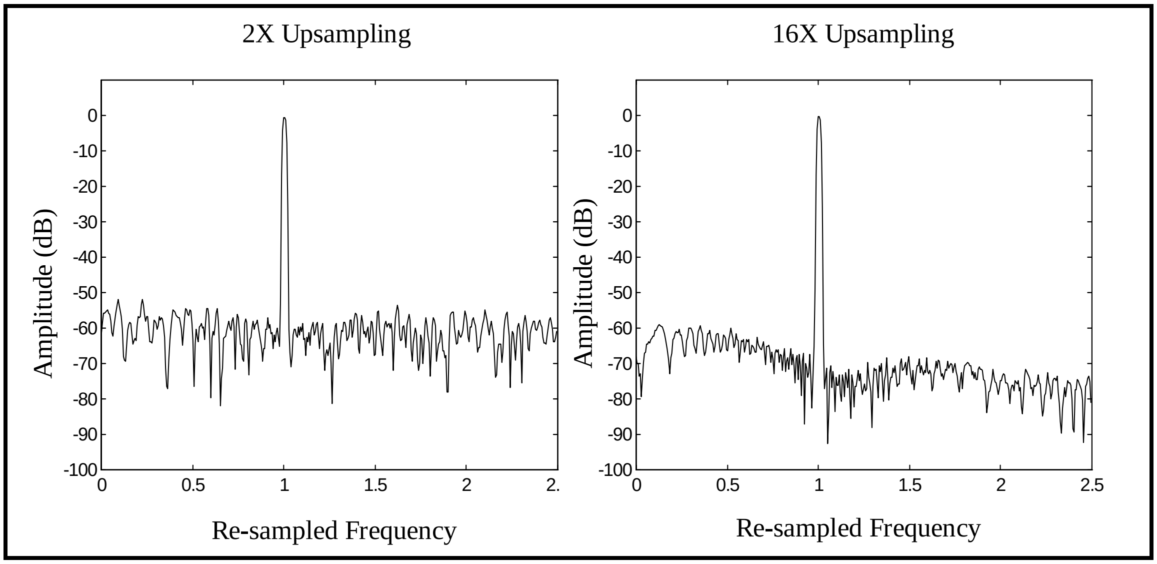

Figure 2.19: Adaptively resampled signals showing effect of amount of upsampling prior to linear interpolation.

This adaptive resampling method works well given a powerful computer to perform the operations and a maximum desired SNR of approximately 60 dB. A greater dynamic range is possible, however, the computational expense to improve this result increases in a non-linear fashion.

To reduce the computational load and the required memory in the implementation of this method, only the upsampled data values that are to be used in the linear interpolation process are calculated. This realization together with the use of a polyphase interpolation filter can greatly reduce the number of calculations required to implement this algorithm. However, the SNR of this method is still somewhat restricted to approximately 60 dB. The method also requires that a minimum of two upsampled points be calculated for each new data value. The linear interpolation requires one data point before and after the desired sample instant.

Further gains in the signal to noise ratio can be gained through the use of a higher order interpolation technique such as a spline or Lagrangian interpolation algorithm. However, all higher order interpolation algorithms require that more upsampled data points be calculated and require more calculations than the simple linear interpolation, again leading to a very computationally demanding implementation.

Chapter 3

3 Theory of Original and Improved Order Tracking Methods.

The previous chapter presented the currently implemented order tracking and tachometer analysis methods in detail. The discussions of each technique presented both the advantages and disadvantages of each method. Upon analyzing the shortcomings of each method, it was determined that advances could be made in computational efficiency and/or the accuracy of several of the methods. This chapter presents the original or improved methods that were developed for analyzing orders as part of this dissertation.

While it is known that several of these methods are new in nature, the adaptive resampling method may not be. It is uncertain whether the adaptive resampling is new because of the proprietary nature of the algorithms that are commercially implemented. The hardware/software companies are not willing to divulge the methods that they employ, hence it is impossible to determine the originality of this technique as applied to order tracking.

A method for estimating linearly independent operating shapes from order tracking based on the Complex Mode Indicator Function (CMIF) algorithm used for parameter estimation in modal analysis is also developed in this chapter. Methods are derived from the CMIF procedure to estimate operating shapes, track operating shapes, and condense large amounts of data. The data condensation algorithm estimates a small number of virtual measurements from a potentially very large set of real measurements, rendering the analysis of the data much more efficient.

3.1 Amplitude Varying Tachometer Analysis.

If an optical or TTL type tachometer sensor is used, the resulting pulse trains can be easily analyzed using the standard tachometer processing algorithms. The standard tachometer algorithms require the user to choose a trigger level and slope to determine the time instants of the arrival of each tachometer pulse. This method works well with a clean tachometer signal as shown in Figure 3.1.

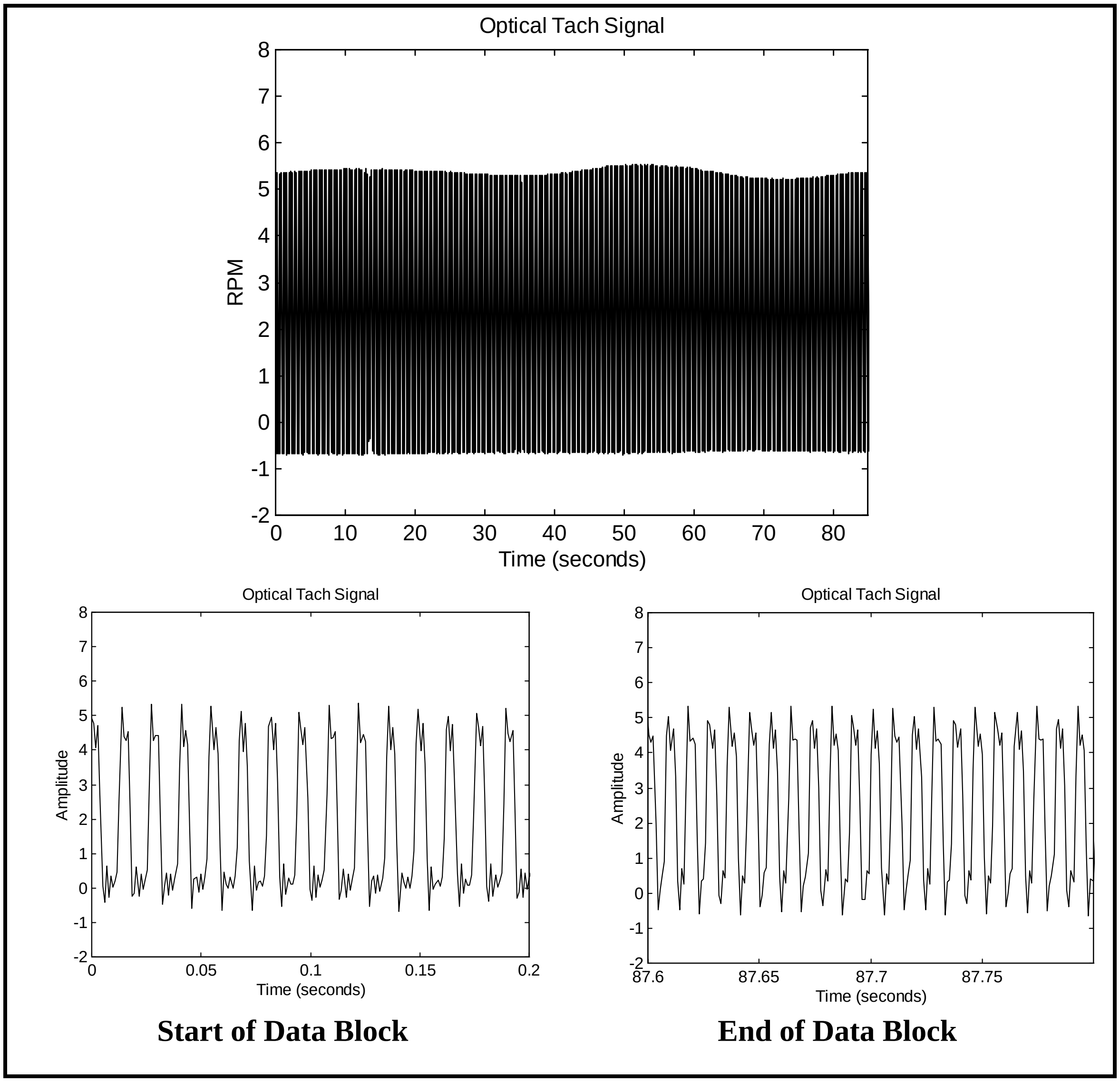

Figure 3.1: Typical optical tachometer pulse train.

It is easily seen in this figure that one trigger level may be set for all time since the amplitude of the tachometer pulses do not vary as a function of time or speed. The small variation seen in the amplitude profile is due to sampling the data not the sensor output.

When a magnetic inductance probe is used as a tachometer sensor, the resulting pulse train often varies in amplitude as a function of rpm. A typical magnetic inductance probe pulse train is shown in Figure 3.2. Obviously, this pulse train is much more difficult, if not impossible, to analyze with a trigger level that does not vary as a function of time or rpm.

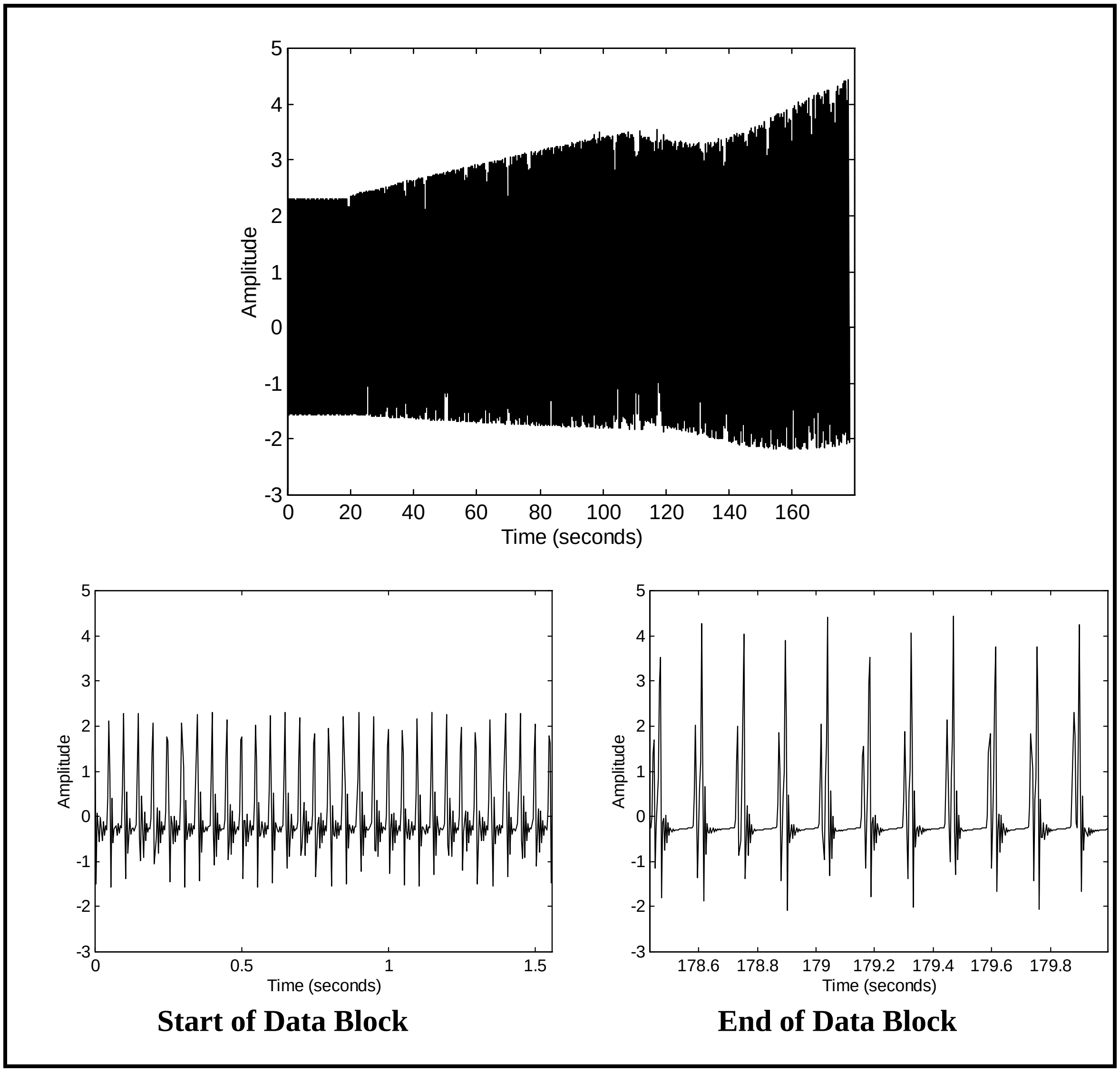

Figure 3.2: Typical magnetic inductance probe tachometer pulse train.

At the beginning of the data block, the amplitude of the peaks in the pulse train shown in Figure 3.2 is near 2. A series of secondary peaks with an approximate amplitude of 0.5 is also present in this portion of the data. Near the end of the same pulse train, the primary peaks have an amplitude of 3 while the secondary peaks have an amplitude of approximately 2. It is therefore impossible to choose one trigger value that will work across the entire data block.

A method that can be used to overcome this problem is to use a trigger level that is a percentage of the peak amplitude of the signal. This trigger amplitude is allowed to vary with time and therefore has an adaptive behavior that allows tachometer signals like that presented in Figure 3.2 to be analyzed effectively.

The trigger level is initially set at a level that will only sense the presence of the tachometer pulse and not give a false indication due to a secondary peak. The first period of the tachometer signal is then determined from the 1st and 2nd crossings. The first period of the tachometer signal is then scanned for its maximum. The trigger level is divided by this maximum to determine the percentage of maximum that the trigger level is set to. The next tachometer period is then determined and the trigger level updated to the same percentage of the maximum in this period as was determined by the initial period. This procedure is performed for each tachometer period. The trigger level tracks any peak amplitude changes in the tachometer signal, allowing any noise in the tachometer signal that maintains a constant amplitude relationship to the peaks to be neglected.

This adaptive trigger level algorithm is much more computationally intensive then a non-adaptive algorithm. However, this method can effectively process tachometer signals that are otherwise impossible to analyze.

3.2 Time Variant Discrete Fourier Transform Order Tracking.

The resampling based order tracking methods provide accurate leakage free estimates of orders at the expense of computational complexity. With the acceptance of large channel count acquisition systems and test methods, it is desirable to obtain similar order estimates with a reduced amount of computational complexity. Oftentimes the computational complexity limits the sweep rate or number of channels which can be analyzed.

A new order tracking method has been developed that gives results very similar to the resampling based order tracking with considerably less computational demand. This new order tracking method is based on a Fourier transform kernel whose frequency is allowed to vary with time [ref. 36,37]. It is therefore called *Time Variant Discrete Fourier Transform Order Tracking* (TVDFT). The TVDFT does not require the resampling of each channel from the time domain to the angle domain to achieve these results, hence the large reduction in computational load.

The TVDFT may be extended to effectively separate close or crossing orders through a secondary calculation. The secondary calculation is the use of an orthogonality compensation matrix (OCM) which compensates for the non-orthogonality of the orders when they are either very close in frequency or cross one another [ref. 37].

3.2.1 Theory of Time Variant Discrete Fourier Transform Order Tracking.

TVDFT order tracking is a special case of the chirp-z transform. The chirp-z transform is defined as a type of Fourier transform with a kernel whose frequency and damping vary as a function of time [ref. 38,39]. The TVDFT is defined as a discrete Fourier transform whose kernel frequency varies as a function of time defined by the rpm of the machine. The damping of the kernel does not vary as a function of time.

The TVDFT method is based on constant Δt sampled data. Therefore, Shannon's sampling theorem for the FFT is valid to determine both F_{max} and Δf. However, since the kernel of the transform is varying in frequency as a function of the rpm, the sampling theorems presented for the resampling based order tracking methods are also applicable.

The TVDFT is based on the transforms shown in Equation 3.1. It should be noted that the kernel of this transform appears as a portion of the structure equation used in the original Kalman order tracking filter presented in Section 2.5. This kernel is a cosine or sine function of unity amplitude with an instantaneous frequency matching that of the tracked order at each instant in time. Alternatively, the kernel may be formulated in a complex exponential format similar to the corresponding exponential Fourier transform.

$$
\begin{aligned}
a_m &= \frac{1}{N}\sum_{n=1}^{N} x(n\Delta t)\cos\left(2\pi\int_0^{n\Delta t}\left(o_m * \Delta t * rpm / 60\right)dt\right) \\
b_m &= \frac{1}{N}\sum_{n=1}^{N} x(n\Delta t)\sin\left(2\pi\int_0^{n\Delta t}\left(o_m * \Delta t * rpm / 60\right)dt\right)
\end{aligned}
\qquad (3.1)
$$

where: o_m is the order that is being analyzed, mΔo.
a_m is the Fourier coefficient of the cosine term for o_m.
b_m is the Fourier coefficient of the sine term for o_m.
rpm is the instantaneous rpm of the machine.

The kernels of the transform shown in Equation 3.1 appear the same as the sampled waveform of Figure 2.7. The reason the angle domain resampling is so effective at analyzing orders is because after resampling, the kernel of the transform and the data have the same form. Both are constant frequency sine waves. The same characteristic has been achieved in the case of the TVDFT by essentially resampling the kernel of the transform, as opposed to resampling each individual data channel. The TVDFT still does not have a kernel whose amplitude varies as a function of time like that of an order. This is not possible since the purpose of the TVDFT is to estimate the amplitude and phase of an order.

This transform is best suited to estimate an order with a constant order bandwidth. The constant order bandwidth estimate is obtained by integrating the instantaneous rpm of the machine to obtain the number of revolutions the shaft has rotated through at each instant in time. A constant order bandwidth estimate may be obtained by performing the transform over the number of time points required to achieve the desired order resolution, as defined by the angle domain sampling relationships presented in Equation 2.7. This implies that as the rpm increases, the transform will be applied over a shorter time, giving a wider Δf but achieving a constant Δo. This integration time behavior was exhibited by the resampling methods and determined to be advantageous for order tracking. The transform is normally only performed for the orders that are desired and not for a full spectrum of orders as was done in the FFT order tracking methods. Two relationships that are applied with the sampling theorem established in Equation 2.7 are given in Equation 3.2. These relationships are necessary to minimize leakage effects in this type of analysis.

$$\begin{gathered} \frac{1}{\Delta o} = \text{integer} \\ \frac{\textit{order tracked}}{\Delta o} = \text{integer} \end{gathered} \tag{3.2}$$

The relationships in Equation 3.2 impose restrictions on the actual order bandwidth that may be applied in the application of the TVDFT. The second of these relationships further imposes a restriction on which orders may be tracked with minimal leakage errors using the TVDFT. It should be noted, however, that even with these restrictions, the user may track most orders that are typically of interest in many applications.

Since the frequency of the kernel of this transform matches the frequency of the order of interest at each instant in time, there is no leakage due to the order not falling on a spectral line. There will, however, be leakage effects from other orders that are present in the data. These orders can *leak* into the frequency band of analysis around the order. Any of the windows used for conventional FFT analyses can be used with the transform to minimize this effect. Since all windows have a frequency resolution/amplitude estimate tradeoff, the window chosen can have a significant effect on the results. Which window to use depends on the order content of the data and the aspect of the order estimate the user feels are most important.

There can also be a leakage error using this transform with constant Δt sampled data because it is not guaranteed that the integer revolution values required for a constant order bandwidth analysis will fall on a Δt. If the integer revolution value does not fall on a Δt, the transform is performed over a non-integer number of cycles, leading to a leakage error. This error is due to the transform kernel of other orders, which fall on delta order lines, not being exactly orthogonal with the order being analyzed. The magnitude of the error is a function of the number of cycles over which the data is analyzed. The error can be corrected through the use of an orthogonality compensation matrix, as described in Section 3.2.2. In most cases, this error will be minimal and can be neglected for trouble shooting.

A second method of reducing the leakage error is over-sampling the data, which provides a finer Δt. This allows the method to analyze sections of data that are closer to having an integer number of revolutions than were possible with lower sampling rates. The over-sampling can be performed when the data is acquired, or as a post-processing phase of the data analysis using an interpolation algorithm. This type of upsampling is much less computationally demanding than the adaptive resampling procedure because it is not necessary to upsample by a large amount to improve accuracy or to do any linear interpolation.

The TVDFT transform can also be applied to obtain order estimates with a constant frequency bandwidth. These estimates are more susceptible to the leakage error described above at low rpm values where the orders are not well separated. The constant frequency formulation has the same practical problem of the FFT methods. They both analyze the same length of time for all rpm values, regardless of the rate of change of the frequency or amplitude of the order. The TVDFT estimate, however, will not be smeared across multiple spectral lines and will be more accurate than an FFT method.

The TVDFT order tracking method is a very practical order tracking method that can be implemented in a very efficient manner. The method contains many of the advantages of the resampling based algorithms without much of the computational load and complexity, since interpolation from the time domain to the angle domain is not required. Computational efficiency is gained for large numbers of channels by computing the transform kernel once, storing it, then applying it to each channel. Any window used in the analysis should be applied to the pre-computed kernel for computational efficiency. The window only has to be applied once if it is applied to the kernel instead of once for each channel.

3.2.2 Theory of Orthogonality Compensation Matrix.

To enhance the capabilities of the TVDFT for order tracking and to reduce the errors due to non-orthogonality of the kernels, an orthogonality compensation matrix (OCM) may be applied. The application of the OCM allows faster sweep rates to be analyzed, as well as closely spaced and crossing orders to be analyzed more accurately. The OCM may be applied in a post-processing step to the order estimates from a TVDFT analysis.

To apply the OCM, all orders of interest are first tracked using the TVDFT with either a constant frequency or constant order bandwidth. The tracking should be done intelligently, as the quality of the compensation is related to the quality of the original order estimates. This implies that the user may want to apply a Hanning window to increase out of band rejection. The bandwidth used may be somewhat wider than is minimally necessary to separate closely spaced orders. The amount of relaxation of the minimum bandwidth depends on the window used in the analysis. The relaxation of the bandwidth allows fewer revolutions to be analyzed at a time, allowing faster sweep rates to be analyzed.

All orders present in the data that have significant energy should be tracked using the TVDFT before applying the OCM. Any non-tracked orders that have significant energy will lead to errors in the final results because they add noise to the linear equations. The non-tracked orders appear as noise because they can not be represented by the orders used in the calculation. The energy from these untracked orders can lead to major errors, it is actually a form of leakage and can have significant amplitude depending on the window used and their frequency relationship to the tracked orders.

The application of the OCM is a linear equations formulation that is shown in Equation 3.3.

$$\begin{bmatrix} e_{11} & e_{12} & e_{13} & \cdots & e_{1m} \\ e_{21} & e_{22} & e_{23} & & \\ e_{31} & e_{32} & e_{33} & & \vdots \\ \vdots & & & \ddots & \\ e_{m1} & & \cdots & & e_{mm} \end{bmatrix} \begin{Bmatrix} o_1 \\ o_2 \\ o_3 \\ \vdots \\ o_m \end{Bmatrix} = \begin{Bmatrix} \tilde{o}_1 \\ \tilde{o}_2 \\ \tilde{o}_3 \\ \vdots \\ \tilde{o}_m \end{Bmatrix} \tag{3.3}$$

where: e_{ij} is the cross orthogonality contribution of order i in the estimate of order j.
o_i is the compensated value of order i.
$\tilde{o}_i$ is the estimated value of order i obtained using the TVDFT.

The cross orthogonality terms, e_{ij}, are calculated by applying the kernel of order i to the kernel of order j, as shown in Equation 3.4.

$$e_{ij} = \frac{1}{N} \sum_{n=1}^{N} \left\{ \exp\left(2\pi \int_0^{n\Delta t} (o_i * \Delta t * rpm / 60) dt \right) \times Window \right\} \times \exp\left(2\pi \int_0^{n\Delta t} (o_j * \Delta t * rpm / 60) dt \right)^* \tag{3.4}$$

The window used in the original order estimate is applied to order i to compensate for any correction factor that may need to be applied to scale the data correctly. It also includes the effects of the shape of the window in the compensation. Each term in the matrix represents the amount that the orders' kernels interact with one another in the transform estimation. If the orders included in the calculation are orthogonal, the off diagonal terms of this matrix will be zero, as is the case for the standard Fourier transform kernels with a Uniform window.

The compensated order estimates are obtained by pre-multiplying both sides of Equation 3.3 by the inverse of the OCM. This results in the solution shown in Equation 3.5.

$$\begin{Bmatrix} o_1 \\ o_2 \\ o_3 \\ \vdots \\ o_m \end{Bmatrix} = \begin{bmatrix} e_{11} & e_{12} & e_{13} & \cdots & e_{1m} \\ e_{21} & e_{22} & e_{23} & & \\ e_{31} & e_{32} & e_{33} & & \vdots \\ \vdots & & & \ddots & \\ e_{m1} & & \cdots & & e_{mm} \end{bmatrix}^{-1} \begin{Bmatrix} \tilde{o}_1 \\ \tilde{o}_2 \\ \tilde{o}_3 \\ \vdots \\ \tilde{o}_m \end{Bmatrix} \tag{3.5}$$

Since the effects of any orders not included in this calculation are not compensated, it is recommended that all significant orders be included in the compensation calculation.

Very closely coupled orders are normally very difficult to separate using standard FFT or resampling techniques because the orders may beat with one another. The TVDFT without compensation also has difficulty separating very closely spaced orders. However, with compensation the TVDFT can separate the contributions of the orders effectively. Initially, the orders should be tracked with a bandwidth that is at its largest approximately equal to the spacing of the closely coupled orders. If the orders are tracked with this bandwidth using a Hanning window, oftentimes the order estimates will contain beating of the two orders. This beating effect can be removed by applying OCM.

Crossing orders pose a similar problem to that of closely spaced orders. Oftentimes, if two orders cross one another, the order estimates are incorrect at the crossing rpm due to the interaction of the orders. Tracking the orders and then applying the OCM allows the separation of the contributions from each order.

Since a matrix inversion is required to apply the OCM, all numerical concerns with matrix inversion must be considered. The OCM will be invertible if the transform is applied over a sufficient number of revolutions for there to be a measurable phase difference between the different orders of analysis.

A common quantity evaluated to obtain an estimate of the accuracy of a matrix inversion is the condition number of the matrix to be inverted. The condition number of the OCM can be shown to be purely a function of the difference in degrees of the rotation between the two closest orders in the analysis. Since the degree of rotation difference between orders is the only criteria which affects the condition number, the condition number does not have to be directly evaluated for each measurement case. Figure 3.3 shows the relationship between the degree of rotation difference of the two closest orders of analysis and the condition number. This plot can be generated independent of sweep rate and the actual orders analyzed.

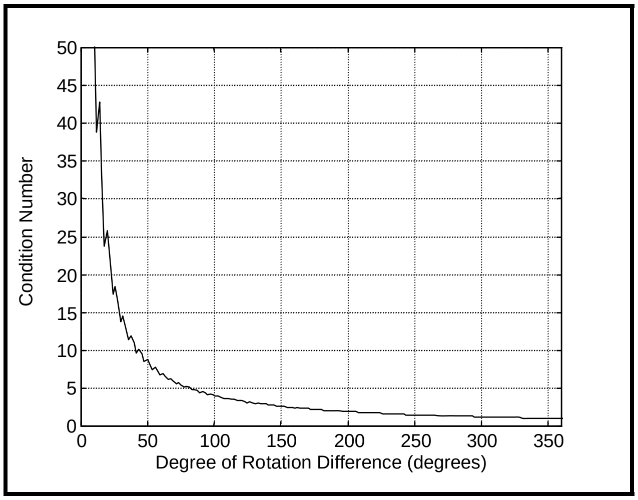

Figure 3.3: Degree of Rotation Difference vs. Condition Number.

The phase difference can be calculated from integrated tachometer signals and the orders to be analyzed for different desired order resolutions. Alternatively, a condition number can be chosen which the user believes will result in a valid matrix inversion, the phase determined from the plot shown in Figure 3.3, and the minimum order resolution which gives this phase difference determined.

3.3 Variations in Frequency Response Function Based Order Tracking.

The FRF based order tracking method presented in Section 2.6 has several limitations. The largest limitation of the method is due to the use of the tachometer signals as inputs.

Using the tachometer signals as assumed inputs to the system does not guarantee that all orders of interest can be tracked. For example, if the tachometer pulse train is on ½ of a revolution and off ½ of a revolution the pulse train is a classical 50% duty cycle square wave. Fourier analysis of a 50% duty cycle square wave shows that only the odd harmonics will be present. Only having odd harmonics present in the frequency domain means that only odd orders may be tracked. If the assumed input contains no energy at a frequency, a response to that input can not be calculated.

A simple method of overcoming this limitation would be to determine which orders were desired at the start of a test set up and use a tachometer sensor setup such that those orders would be present in the Fourier transform of the tachometer pulse train. Realistically, in most cases the tachometer pulse train is not a perfect square wave and this analysis is not required for most desired orders.

A robust method of ensuring that the orders of interest exist in the Fourier transform of an assumed input is to synthesize an input that contains these orders from the tachometer signal information. The tachometer signal can be processed using any of the methods of Section 2.7. On obtaining the rpm at each time instant, a signal can be synthesized that will have the desired orders present.

To synthesize the time domain input orders from the tachometer signals, the complex exponential form of Equation 3.1 is used. Equation 3.6 shows this form of the equation.

$$F_i = \frac{1}{N}\sum_{n=1}^{N} \exp\left(2\pi \int_0^{n\Delta t} \left(o_i * \Delta t * rpm / 60\right) dt \right) \tag{3.6}$$

where: F_i is the synthesized order input for order o_i.
o_i is the order of interest..

Having synthesized each order with a unity amplitude chirp function there are two options on how to process the data. The first option is to combine all of the order inputs from each shaft into one composite input for each shaft. These composite inputs are then treated as the inputs for their respective shafts. The composite inputs are then Fourier transformed and the necessary crosspower matrices calculated. This analysis has the same limitations as the FRF based method presented in Section 2.6.1. The main limitation to this approach was that frequencies of orders of the same tachometer signal could not overlap.

The other option is to treat each independent order as an input to the system. This second option is more computationally intensive but results in a separate order estimate for each input order regardless of whether frequencies of orders cross. This is essentially the same analysis as the FRF based method presented in Section 2.6.2 with the added ability to track any order of interest.

Using synthesized inputs in the estimation of the FRFs has the advantage of not requiring the amplitude of the tachometer signal to be estimated and compensated for. The disadvantage of using synthesized inputs is the added computational load due to the tachometer signal processing and the computations required to synthesize the inputs. Mathematically, whether synthesized inputs are used or the raw tachometer signal is used the ability to separate close and/or crossing orders is identical.

3.4 Adaptive Resampling Based on an Upsampled Interpolation Filter.

While the decimation, interpolation, and resampling methods presented in Appendix A can be understood without a thorough understanding of the digital sampling process, the adaptive resampling interpolation based on an upsampled filter cannot. The non-adaptive decimation and interpolation methods are based simply on lowpass digital filters that must possess a specified cutoff frequency. The adaptive sampling rate interpolation based on an upsampled filter is based on a specific filter that must possess very distinct properties. This section will explain the requirements of this filter as well as its implementation. The section will conclude with an analysis of the performance of this method.

3.4.1 Digital Sampling and Interpolation Theory.

The analog sampling process can be approximated digitally through the use of the sinc (sin x/x) function [ref. 40]. The digital resampling process is accomplished by positioning the center of the sinc function at each point in time a new data sample is desired. The analytical sinc function exists for all time and acts as an allpass digital filter. Since the digitally sampled data does not exist for all time, the sinc function must be truncated in time. The frequency bandwidth of this truncated function must also be limited to eliminate aliasing errors.

One method of producing an FIR filter that approximates the ideal sinc function is through the generation of a sinc function in the time domain which is truncated in time by the application a window function. The window that is applied to truncate the function has an effect on the frequency domain characteristics of the filter. The transition bandwidth and stopband rejection, as well as the passband ripple, are affected.

Time and frequency domain examples of a truncated sinc function are shown in Figure 3.4. This sinc function was generated directly in the time domain and truncated with a Hanning window. The zero crossings of the sinc function are spaced exactly a Δt apart. The zero crossing spacing is what determines the cutoff frequency of an FIR filter based on the sinc function. This property along with the unity amplitude at the center of the function allows any data value which falls at an original Δt to pass the through the filter unaltered.

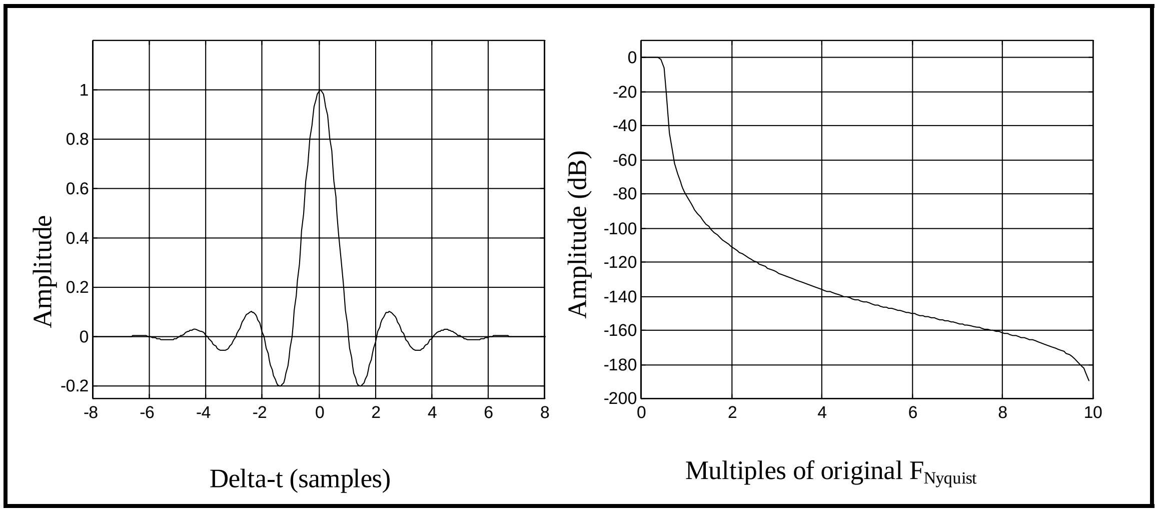

Figure 3.4: Time and frequency domain plots of truncated sinc function (interpolation filter).

The cutoff frequency of this truncated sinc function is the original Nyquist frequency. The filter uses 8 points prior and after the point in time at which the new data point is calculated. The filter possesses 20 values between each zero crossing and therefore could be used to upsample a signal by a factor of 20, giving approximately 80 dB of dynamic range.

To implement this filter in a computationally efficient manner only 16 (2*8) multiplications/adds per new data point are required. The first step in the application of the filter is to center the filter at the point in time that a new data value is to be estimated. Only the filter coefficients that are aligned in time with original sampled data values are used to calculate the new data sample. All other filter coefficients are multiplied by zero. This implies that only every 20th filter point is needed to calculate the new interpolated data value. This computationally efficient implementation is shown in Figure 3.5.

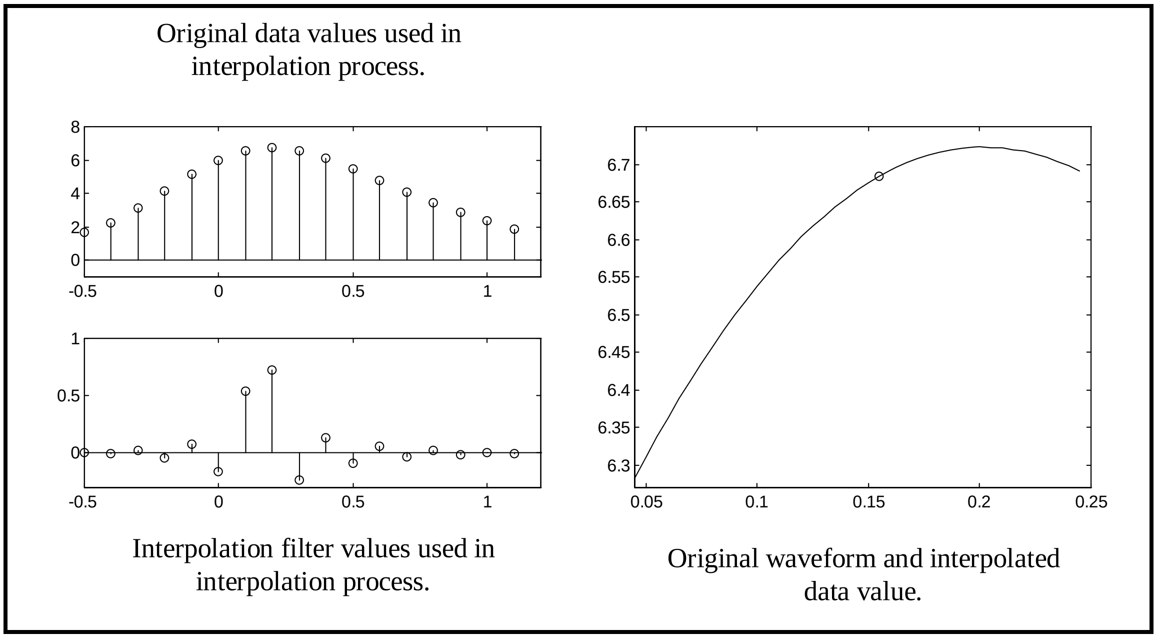

Figure 3.5: Example of efficient implementation and result of digital interpolation process.

As can be seen in Figure 3.5, this interpolation method provides very accurate results. To implement this method for adaptive resampling requires considerably more overhead as described below. Even with the necessary overhead this method is very efficient method for calculating a new data sample at an arbitrary time.

The adaptive implementation of this interpolation method requires that a filter be pre-calculated with a very large number of filter coefficients. The procedure is based on approximating the analog sampling process, in that a new data value can be calculated at any point in time. This requires that the filter be created with, for example, 8192 values between each original Δt. This upsampled filter can then be applied in the same manner as the filter described above. Note there are still only 16 (8*2) multiply/adds per new interpolated data point. By upsampling the filter by such a large amount, an approximation of the analog sampling process is obtained by using the set of filter coefficients which are closest to the time at which it is desired to have a new data value. This is in effect applying a digital sample and hold circuit, where the original signal is upsampled by a very large amount and the new data values extracted from the upsampled data by assuming the closest point is the correct point.

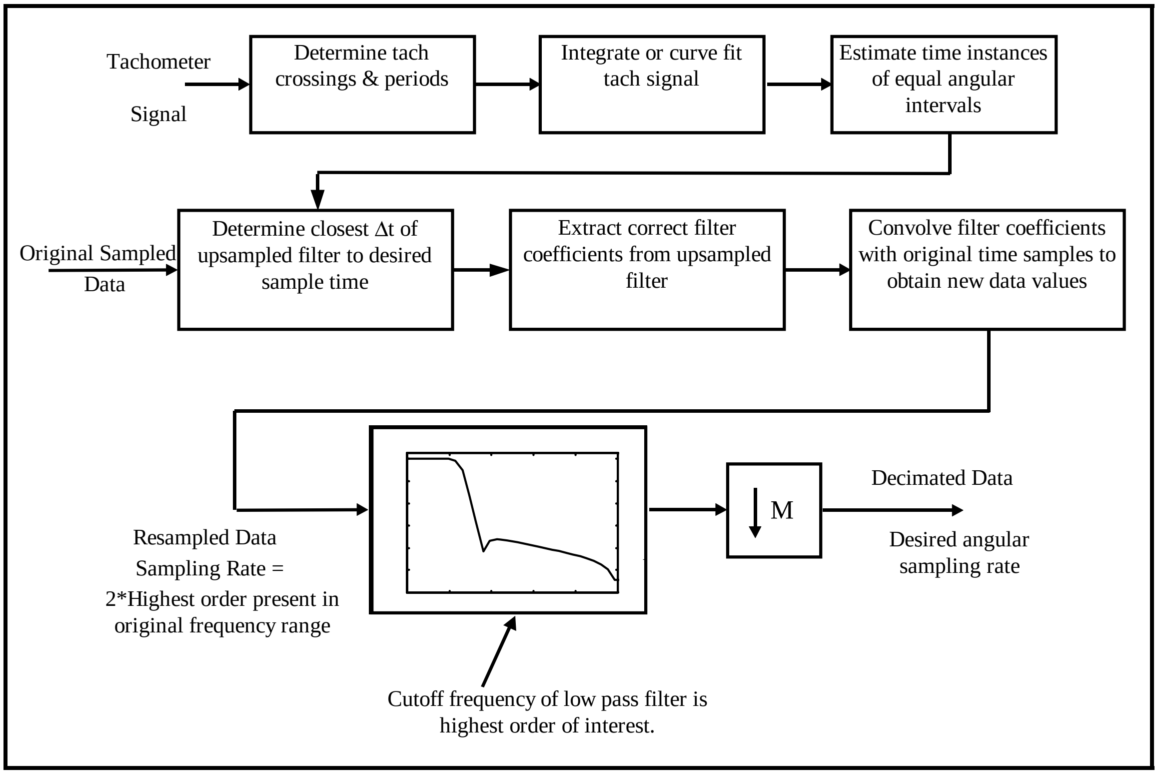

Figure 3.6: Graphical representation of adaptive resampling with an upsampled interpolation filter.

Figure 3.6 shows a graphical representation of an adaptive implementation of the upsampled interpolation filter resampling process for synchronous sampling with respect to a reference tachometer signal. This procedure is equivalent to the procedure presented in Figure 2.15 for the upsampling/linear interpolation adaptive resampling process.

3.4.2 Numerical Example of Upsampled Interpolation Filter Adaptive Resampling.

The same test case used to exhibit the performance of the adaptive resampling by upsampling and linear interpolation is again used to present the performance of the adaptive resampling based on an upsampled interpolation filter. Using the same example for both algorithms allows the broadband noise levels to be directly compared. The example is a chirp function of a linear sweep. The chirp is resampled such that it should appear as a sine wave with a resampled frequency of 1.

Figure 3.7 shows the frequency domain representation of both the original signal and the adaptively resampled result.

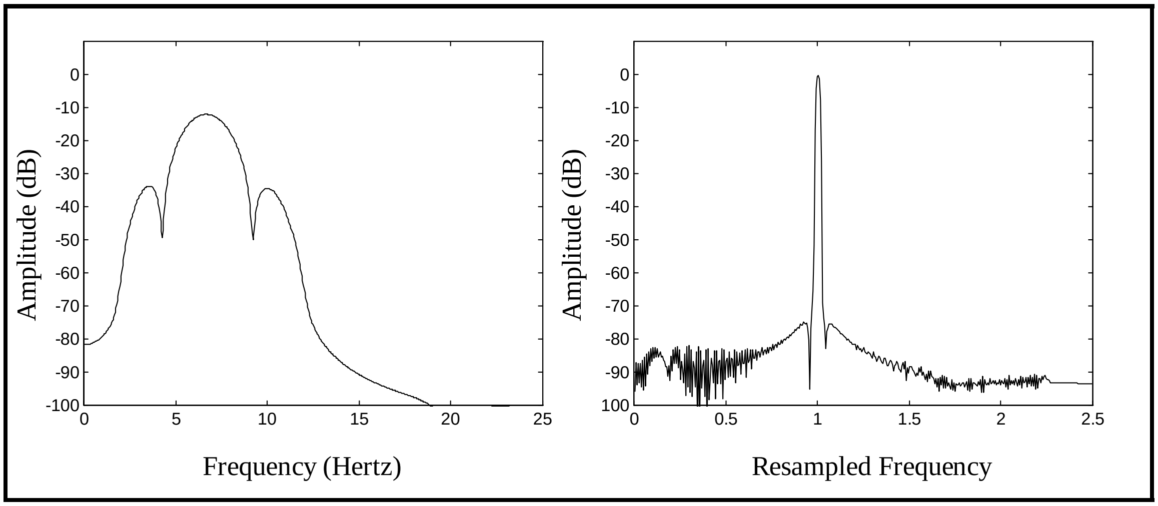

Figure 3.7 Frequency domain representations of original and resampled chirp function.

The upsampled interpolation filter used in this example was generated by truncating an analytical sinc function with a Hanning window. The filter used 9 points before and after each resampled data sample and contained 8192 steps between each original Δt. Note that while the adaptively resampled results from using the upsampling/linear interpolation approach had approximately 60 dB of dynamic range, the result from using the upsampled linear interpolation filter has a dynamic range of approximately 75-80 dB.

The effect of using fewer steps between each original Δt is shown in Figure 3.8. The left hand plot shows the result using 4096 steps with 9 original data values prior to and after each new data sample used in the interpolation process. The right hand plot shows the result using 4096 steps with 6 original data values prior to and after each new data sample used in the interpolation process. Note that the effect of using 18 data samples in the interpolation process compared with using 12 data samples results in a greater signal to noise ratio in the final result. Both plots show approximately the same SNR near the frequency of the signal, however, the plot generated by using 18 data samples has a higher SNR away from this frequency. Using either number of data samples results in a SNR of at least approximately 75 dB everywhere, surpassing that of the previously presented upsampling/linear interpolation based adaptive resampling method.

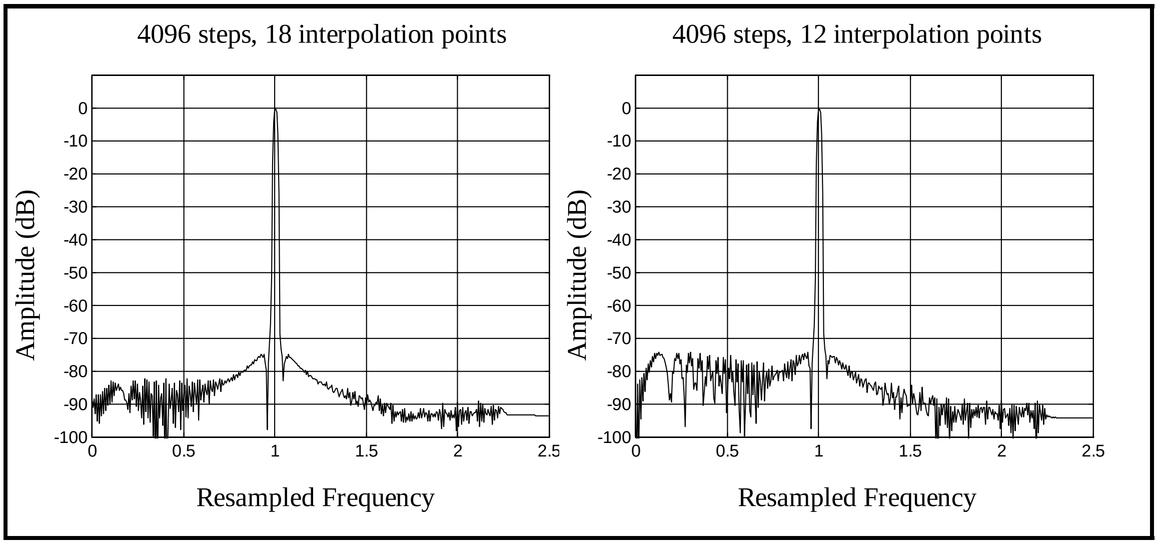

Figure 3.8: Frequency domain representations of resampled chirp function showing the effect of the number of interpolation points.

Figure 3.9 further shows the effect of the number of steps that the original Δt is subdivided into. The left hand plot uses an upsampled interpolation filter with 512 steps. The right hand plot uses an upsampled interpolation filter with 1024 steps. Both of these filters use 9 original data samples prior to and after the new data sample. Again, note that both of these filters give approximately 75 dB of signal to noise ratio. It can clearly be seen that as the number of steps in the filter is decreased the broad band noise becomes larger. The SNR is still dominated by the behavior of the filter near the large frequency component, this frequency range is controlled by the window used in generating the filter.

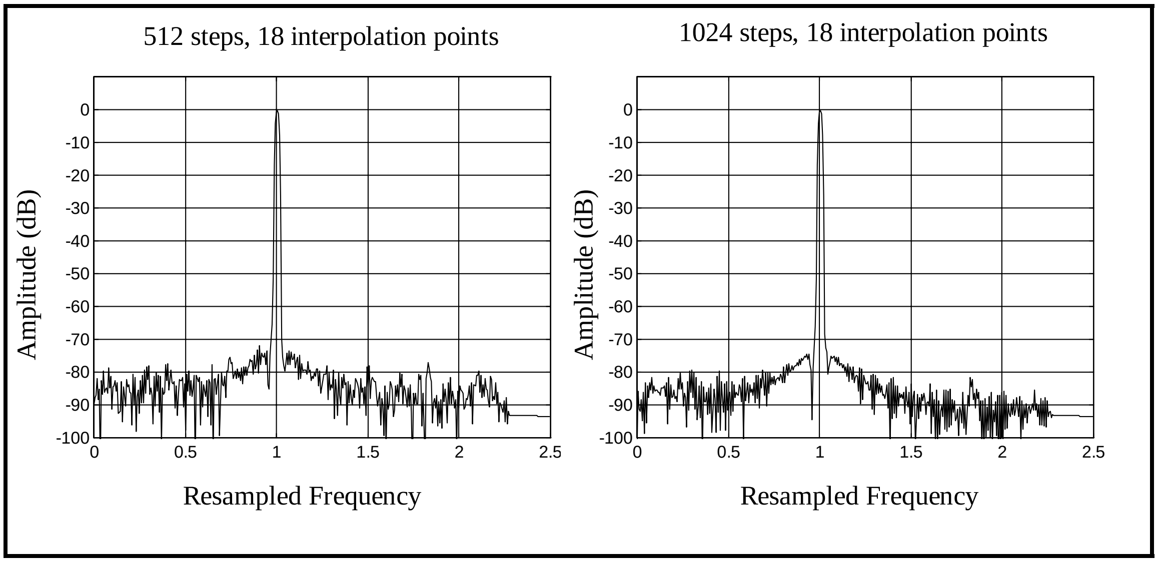

Figure 3.9: Frequency domain representation of resampled chirp functions showing the effect of the number of steps.

3.4.3 Numerical Example to Determine the Effects of Effective Clock Jitter in the Upsampled Interpolation Filter Adaptive Resampling Procedure.

To further analyze the effect of the number of steps that the original Δt is subdivided into, a second example was formulated. This example again uses a chirp function, however the maximum frequency of this chirp is much closer to the Nyquist frequency. This situation allows a better evaluation of the effect of the effective clock jitter that is introduced by the adaptive resampling filter. Clock jitter is introduced if the timing of the sampling intervals is not accurate and the data is sampled at non-uniform intervals. All of the filters used in this example were generated by truncating an analytical sinc function with a Hanning window. This is essentially the same filter that was used in the previous example. These filters also use an identical 9 points prior to and after each resampled data sample in the interpolation process, regardless of the number of Δt subdivisions.

Figure 3.10 shows the frequency domain representations of both the original chirp and the adaptively resampled chirp calculated using 8192 steps.

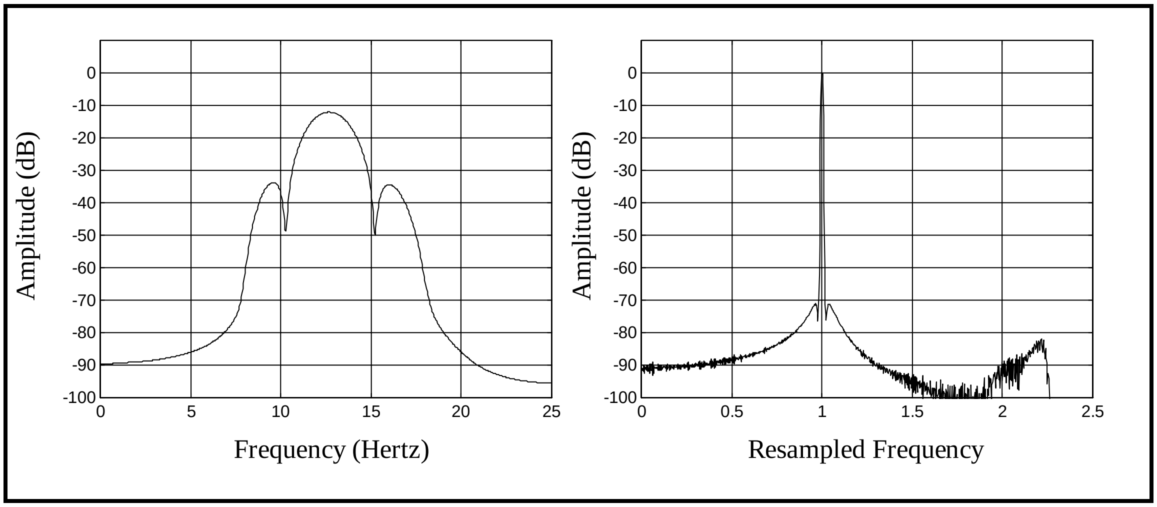

Figure 3.10: Frequency domain representations of original and resampled chirp.

The SNR of this resampled chirp is greater than 70 dB across the entire spectrum showing that the performance of the algorithm provides an acceptable SNR near the original Nyquist frequency.

Figure 3.11 shows the same chirp function resampled with interpolation filters using 512 and 1024 steps respectively. Clearly, it can be observed in this figure that neither of these results has an acceptable SNR. An acceptable SNR is determined by the measurement system and process. Typically experimental measurements have a SNR of approximately 65 dB. The amount of jitter introduced by using too few steps results in a high amount of broadband noise regardless of the number of interpolation points employed.

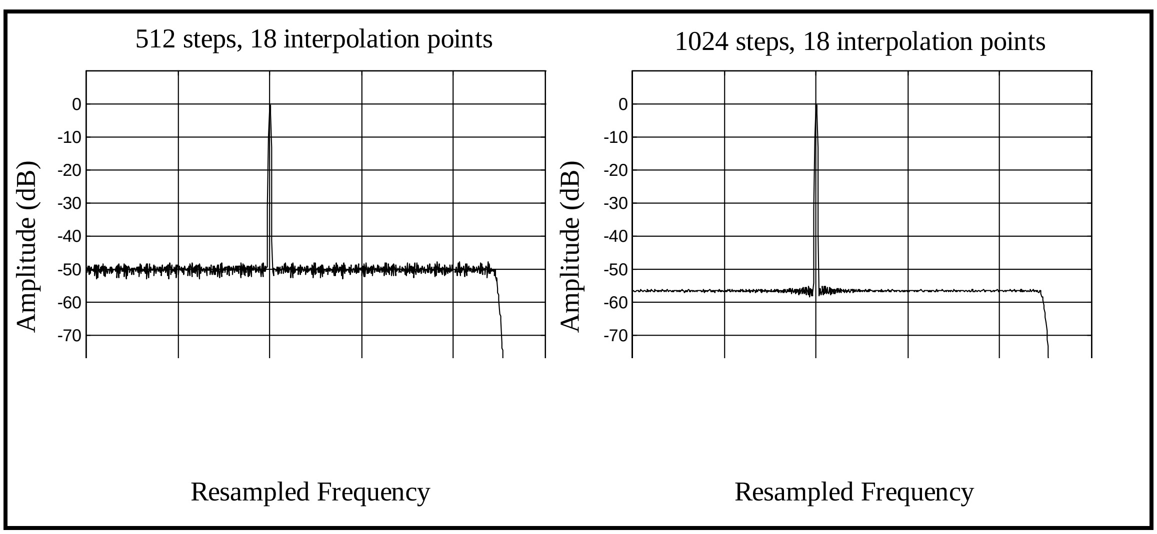

Figure 3.11: Frequency domain representation of adaptively resampled chirps showing the effect of the number of steps.

Figure 3.12 shows the same adaptively resampled chirp function. To generate this figure, 2048 and 4096 steps were used in the upsampled interpolation filter. Both of these plots show that the SNR has improved considerably over the previous two filters. Each of these results is acceptable and has over 70 dB of dynamic range. The difference between the two results is exhibited in the amount of noise away from the large frequency component at resampled frequency 1.

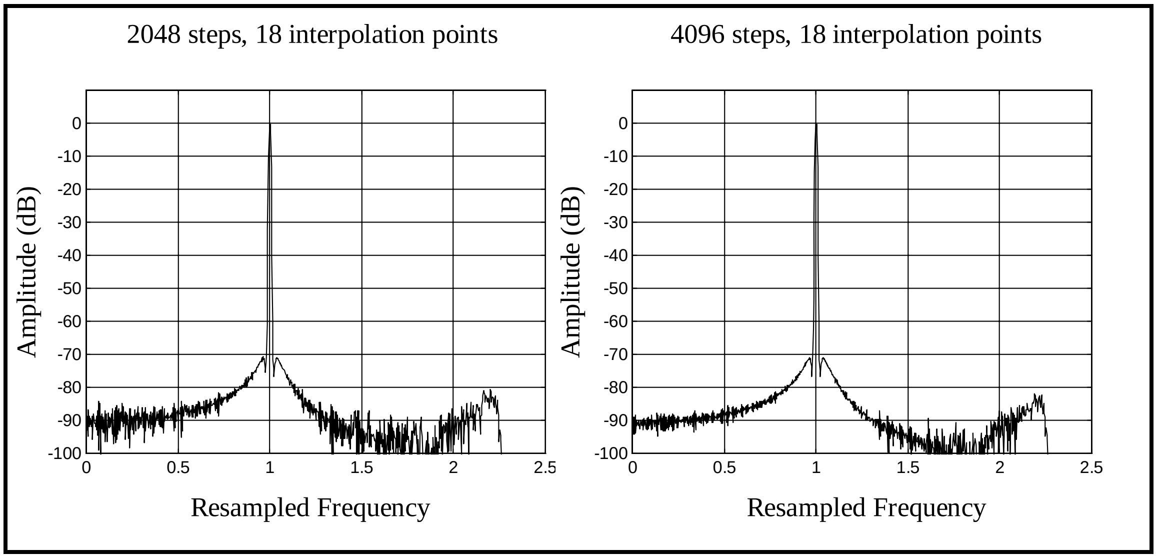

Figure 3.12: Frequency domain representations of adaptively resampled chirps showing the effect of the number of steps.

Finally, Figure 3.13 shows the same chirp function adaptively resampled with the upsampling/linear interpolation approach. These plots are included for comparison and completeness to show that this method also functions well for adaptively resampling components near the Nyquist frequency. The left hand plot was upsampled by 16 prior to the linear interpolation. The SNR of this result is approximately 60 dB. While 60 dB is acceptable in many instances with experimental data, it is not as good as the 70+ dB exhibited in the upsampled interpolation filter results. To obtain a higher SNR the same chirp was upsampled by a factor of 32 prior to linear interpolation. This result is shown in the right hand plot. Note that the SNR has improved in this case to the 70 dB exhibited by the upsampled interpolation filter results. This amount of upsampling prior to the linear interpolation process requires many more calculations than the upsampled interpolation filter.

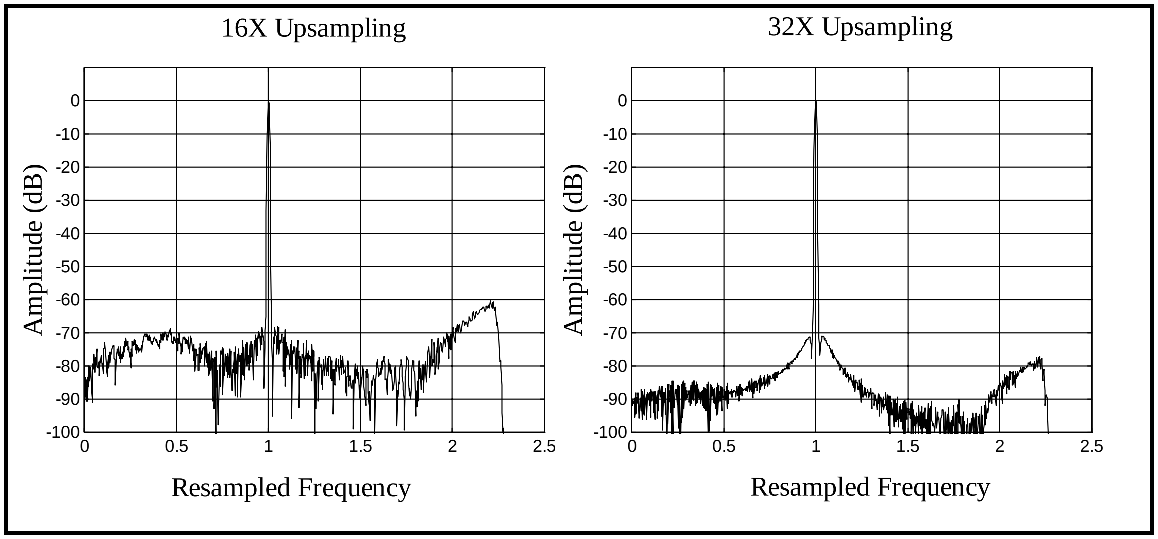

Figure 3.13: Frequency domain representation of adaptively resampled chirps showing the effect of the amount of upsampling prior to linear interpolation.

3.4.4 Additional Adaptive Resampling Errors.

Both of the adaptive resampling procedures have errors inherent in them. Both methods presented have some amount of approximation present in them. To implement a truly perfect adaptive resampling algorithm would require a separate digital filter to be designed for each new data sample. This filter would possess the exact amount of phase delay between the original data samples and the new data sample. This type of procedure would be very computationally demanding and is not feasible even with today's powerful computers.

The upsampling/linear interpolation method has an error that is associated with the linear interpolation process. When linearly interpolating a sinusoidal function, an error that is similar to a slight phase delay at each sample is introduced into the resampling process. The magnitude of this error is not predictable and varies as a function of the distance from the closest original time sample to the newly interpolated sample. This error is reduced through a higher upsampling factor prior to the linear interpolation as shown in the example in Sections 2.8.2 and 3.4.3.

The upsampled interpolation filter adaptive resampling procedure introduces an error that can be described exactly by sample clock jitter. The error that sample clock jitter introduces into a signal is a broadband type of noise. This error is also not predictable in the general case and is minimized through the use of a filter that is created with a larger upsampling factor. The magnitude of the possible jitter error can be analyzed if the characteristics of the original data acquisition system that was used to acquire the data are understood. The magnitude of the error that is introduced by sample clock jitter is a function of both the sample clock instability and the number of bits which this ADC uses. An analysis of this error is presented by Watkinson in reference 41.

Both adaptive resampling procedures require that the tachometer signal used as the reference be very accurate. If, for example, the tachometer signal that is measured has substantial inaccuracies present, or is processed in such a manner as to have errors, the new values of time at which data values are to be estimated will be in error. This error can manifest itself as either a bias error or as a random error.

Bias errors are introduced if the sweep rate of a test sweep is estimated incorrectly. For example, a bias error will occur if the sweep rate is estimated to be too slow. The new data values will be estimated to be further apart than they actually are, resulting in a non-integer number of samples/revolution and/or a non-uniform angular spacing. This invalidates the reason for performing the resampling process. If a Fourier transform were used to analyze re-sampled data with this error present, leakage would be present in the results which otherwise should have been leakage free!

A random error could occur if the machine's rpm were estimated to be slowly varying when in actuality it was constant. This error in rpm would introduce a Δt spacing error that is constantly varying from too small a Δt to too large a Δt. This type of error is similar to sample clock jitter and would therefore introduce broadband random noise into the resulting data. This type of error would also occur if it were desired to analyze the torsional vibration of a system and the number of tachometer pulses/revolution was less than the highest order that was present in the data.

3.5 Applying the Singular Value Decomposition to Order Track Results - An Innovative CMIF Application.

The use of operating mode shapes to aid in the solution of noise and vibration problems has been increasing in recent years. The primary measurement that has been made to obtain these operating shapes has been transmissibility [ref. 42]. The necessary transmissibility measurements are typically made under steady state operating conditions. The forcing functions that typically excite the structure under these conditions are deterministic in nature and are often from a rotating component of a machine.

Recently there has been research into the decomposition of operating shapes measured under steady state conditions with multiple random source inputs [ref. 43,44]. Multiple uncorrelated random sources allows the decomposition of the spectral matrices into linearly independent operating shapes. There are as many linearly independent operating shapes estimated at each frequency as there are uncorrelated random forcing functions. The singular values of the spectral matrices reach peak values at frequencies corresponding to either natural frequencies of the system or peaks in the forcing function. At frequencies that correspond to natural frequencies of the system, the linearly independent operating shapes will approximate the actual mode shapes of the system.

Another approach to obtaining operating shapes is through the use of order tracking techniques. Order tracking allows an operating shape to be estimated for each order that is tracked at each rpm value that is evaluated. Estimating operating shapes with order tracking techniques can provide valuable insight into the forced response of a structure over a range of operating conditions. However, until recently, valid order tracking results could not be obtained when orders were either very close to one another in frequency or crossed one another.

New order tracking techniques have been developed which allow the separation of the contributions from either close or crossing orders [ref. 19-21,36,37]. Each of these techniques was presented earlier in this dissertation, in Sections 2.5.2, 2.6, and 3.2. Effectively separating contributions from different orders allows multiple operating shapes to be estimated at each rpm value. The operating shapes are each due to an orthogonal input. Each order is orthogonal to every other order. Since the inputs to the system are orthogonal, the shapes obtained at each rpm value can be decomposed into a set of linearly independent operating shapes. It is possible to estimate as many independent operating shapes as there are modes in the system which are excited at each given rpm, or orders evaluated, the minimum of the two.

Estimating operating shapes from many orders at many different rpm values results in a very large number of operating shapes. This section presents several methods to attempt to determine how many different operating shapes are excited over the entire rpm range. Further analysis can be performed to determine at which rpm values similar shapes are excited by different orders. Finally, the sets of operating shapes can be decomposed at each rpm value into a set of linearly independent operating shapes through the use of the singular value decomposition, SVD [ref. 46,47]. This decomposition is done in a manner similar to that of the CMIF modal parameter estimation algorithm [ref. 45,54-56]. The results of the SVD are linearly independent operating shapes that in most cases will approximate the mode shapes of the system.

The SVD may be performed at each rpm value to obtain an estimate of the number of linearly independent operating shapes excited at each rpm or point in time. Mode tracking may then be applied to the left singular vectors to obtain an estimate of the rpm values where the same mode may be excited by different orders. Finally, the right and left singular vectors may be used to calculate a *Mode Enhanced Order Track*, MEOT [ref. 46]. The estimated MEOT will approximate a single degree of freedom system and can be used to estimate the frequency and damping of the mode if the mode is due to the excitation of a natural frequency of the system, and not a peak in the forcing function. This MEOT is very similar to an enhanced FRF and has similar properties. The relationship to an eFRF is clear if an order track is thought of as an unscaled FRF. An order track can be thought of as simply a swept sine FRF as long as there are no peaks in the forcing function. This will be true, for example, if the order is due to an unbalance force.

The SVD may also be performed on the set of operating shapes that are similar in frequency to one another, which involves decomposing the set of shapes from different orders at different rpm values. The result of this frequency based decomposition is an estimate of the number of linearly independent operating shapes excited at each frequency. This may give an estimate of any repeated modes of the structure.

3.5.1 Estimation of Linearly Independent Operating Shapes.

Once a suitable order tracking analysis has been performed and a set of operating shapes estimated for each order at each rpm value, the set of operating shapes may be decomposed into a linearly independent set of operating shapes. These linearly independent operating shapes will approximate the mode shapes of the system in most cases but are not guaranteed to do so.

The decomposition can be performed in two different ways, both of which are described in the following sections.

3.5.1.1 Operating Shape Decomposition by RPM Value.

The first manner in which the operating shapes can be decomposed is on an rpm by rpm basis. This method will estimate as many different linearly independent operating shapes as there are orders included in the analysis or modes excited at each rpm value, the minimum of the two. This operating shape decomposition is accomplished by performing a singular value decomposition of the operating shapes estimated by each order at each rpm. This decomposition is shown in Equation 3.7 and is performed for each rpm estimate.

$$[U][S][V]^H = SVD(\{O_1\} \quad \{O_2\} \quad \cdots \quad \{O_3\}) \tag{3.7}$$

where: [U] is the matrix of left singular vectors.
[S] is the diagonal matrix of singular values.
$[V]^H$ is the matrix of right singular vectors.
$\{O_i\}$ is the operating shape vector of order i at the analysis rpm.

This analysis is essentially the same as the CMIF parameter estimation algorithm used in modal analysis. It has been shown that the left singular vectors represent the set of linearly independent operating shapes corresponding to the singular values. These shapes will approximate actual mode shapes of the system in many instances. Since the matrices of the left and right singular vectors are of unity scaling, the singular values, S, identify how well excited each linearly independent shape is at each rpm. The right singular vectors, V, represent a type of participation factor between each order and the linearly independent operating shapes.

To obtain an estimate of how many different linearly independent shapes are excited at each rpm value, the singular values may be plotted as a function of rpm. This plot is interpreted in the same fashion as a CMIF plot is interpreted in modal analysis. There will be as many singular value curves as there are orders included in the analysis. The number of well excited linearly independent shapes is then estimated by observing how many singular value curves peak at each rpm value. If the lowest singular value curve peaks up at a given rpm value this is an indication that there are as many independent shapes excited as there are orders being analyzed. The highest singular value curve represents the best excited operating shape. This will be clearly shown in the examples of Chapter 4.

The left singular vectors from the dominant singular values may be animated to give an understanding of the linearly independent shapes which combine to describe the forced response of the system for the orders analyzed.

3.5.1.2 Operating Shape Decomposition by Frequency.

Section 3.5.1.1 described the decomposition of estimated operating shapes on an rpm by rpm basis. This section describes performing the same decomposition except in this case the matrix of operating shapes is filled with shapes that occur at the same frequency regardless of the rpm or order. Different orders can pass through the same frequency at different rpm values in a run up/down acquisition. The operating shapes estimated from these orders, corresponding to a given frequency, can be grouped together for analysis.

Grouping the operating shapes by frequency allows an estimate of whether there are two or more excited modes of the system that fall within one frequency bin. The number of operating shapes excited at each frequency will vary on a frequency by frequency basis. At low frequencies, there are usually more orders that pass through a given frequency. As the frequency increases, the lower orders may not pass through a given frequency in a run up/down condition. This characteristic can lead to the ability to estimate a large number of linearly independent operating shapes at low frequencies and possibly only one at higher frequencies. If there is only one operating shape present in a frequency bin, no decomposition can be performed and no estimate of whether there are repeated modes of the structure can be obtained.

The frequency based decomposition calculations and their interpretations are the same as those listed in Section 3.5.1.1. The left and right singular vectors can once again be used to calculate enhanced functions and autopowers with respect to a given singular vector. These functions are then plotted as a function of frequency instead of rpm and are thus interpreted with respect to frequency instead of rpm or time. The enhanced frequency based order track should be nearly the same as the MEOT estimated for the same singular vector and frequency. The singular vector autopowers should only peak at the frequency which the corresponding singular vector is calculated. Where a singular vector autopower might have peaked at multiple rpm values because different orders excited the same mode, in the frequency based calculation of the singular vector autopower it should only peak at the frequency of the shape. The frequency based decomposition provides no new useful information for trouble shooting other than a possible indication of repeated modes of a structure.

3.5.2 Operating Shape and Singular Vector Tracking by RPM Value.

An estimate of where a given operating shape is excited across the rpm range can be obtained by tracking the desired shape as a function of rpm. This mode tracking is performed by calculating a *Modal Assurance Criterion*, MAC, between the desired operating shape and all other estimated operating shapes at each rpm value. Where the operating shapes are similar, the MAC value between the tracked shape and the operating shape will be high, above for instance 0.9. This mode tracking will indicate if similar shapes are excited by different orders at different rpm values. The equation used to perform the mode tracking is shown in Equation 3.8.

$$MAC_{ij} = \frac{\left| \{O_i\}^H \{O_j\} \right|^2}{\{O_i\}^H \{O_i\} \{O_j\}^H \{O_j\}} \tag{3.8}$$

where: $\{O_i\}$ is the operating shape vector of order i at the analysis rpm which is being tracked.
$\{O_j\}$ is the operating shape vector of order j at any rpm value.

Just as mode tracking can be used to determine whether a similar operating shape is excited at different rpm values by different orders, it can also be used to track the linearly independent operating shapes obtained from the SVD. The tracking of the singular vectors gives an estimate of at what rpm values the same mode of the system is excited, if the independent operating shape being tracked represents a mode of the system. The tracking calculation with the singular vectors is performed by substituting the U vectors from the SVD for the operating shape vectors, $\{O_i\}$, in Equation 3.8.

3.5.3 Calculation of Order Track Autopower.

The autopower of an order may be calculated and compared with the autopower of all other tracked orders to obtain an estimate of the total energy of an order across all measurement dofs. The order that has the highest amplitude at a given rpm is the most likely candidate to attack to solve a noise or vibration problem at that rpm value. The autopower is one method of reducing a very large number of response dofs down to one composite measurement per order. The order track autopower is calculated by using Equation 3.9 shown below.

$$OA_{i,r} = \left(\sum_{k=1}^{n} (O_{ik})^2 \right)_r \tag{3.9}$$

where: $OA_{i,r}$ is the order track autopower of order i at rpm r.
O_{ik} is the value of order i at degree of freedom k at rpm r.
n is the number of measured degrees of freedom.

3.5.4 Calculation of Singular Vector Autopower.

The autopower of a singular vector may be calculated across all orders and rpm values. The singular vector autopower indicates the rpm values where the singular vector of interest is excited. Again, if the singular vector is representative of a mode of the system then this quantity is an estimate of how well excited this mode is at each rpm value. The calculation of the singular vector autopower, or mode autopower, is calculated by applying the principle of a modal filter. The basis of this calculation is the weighting of each degree of freedom by the corresponding element in the left singular vector. The participation factors for each order are assumed to be unity. The equation used to calculate the singular vector autopower is shown in Equation 3.10.

$$SVA_{l,r} = \{U_l\}_{1xn}^{H} [\{O\} \quad \{O\} \quad \cdots \quad \{O\}]_{nxm} \begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix}_{mx1} \tag{3.10}$$

where: $SVA_{l,r}$ is the singular vector autopower of singular vector l at rpm r.
$\{U_l\}^H$ is the hermitian of the left singular vector l that the autopower is calculated with respect to.

{O} is the operating shape vector for each order of interest at rpm r in no particular order.

3.5.5 Calculation of Mode Enhanced Order Track.

The left and right singular vectors of a singular value from any rpm may be used to calculate an enhanced order track relative to the singular vector, or mode, of interest. This quantity is called a *Mode Enhanced Order Track*, MEOT, and can be useful in determining the exact frequency and damping of a specific excited mode. The MEOT is very similar to an enhanced FRF that is calculated to estimate the frequency and damping of a mode of vibration in the CMIF parameter estimation method.

Since the order tracks are estimated from orthogonal functions, the system can be considered to have linearly independent inputs, one for each order. This property implies that the MEOT should appear very similar to a single degree of freedom FRF in most cases. The MEOT should not peak at each rpm that the mode of interest is excited. The singular vector autopower peaks at every rpm that the mode of interest is excited at, regardless of which order provides the excitation. Instead, the MEOT should peak only at the rpm values of the estimated singular vectors used in the calculation, unless the phase of the excitation orders' aligns itself to the same condition elsewhere in the speed sweep. This latter condition will be illustrated in an example in Chapter 4. The calculation that is performed to estimate this MEOT is shown in Equation 3.11.

$$MEOT_{l,r} = \{U_l\}^H_{1xn} \left[\{O\} \quad \{O\} \quad \cdots \quad \{O\} \right]_{nxm} \{V_l\} \tag{3.11}$$

where: $MEOT_{l,r}$ is the mode enhanced order track for singular vector l at rpm r.

$\{U_l\}^H$ is the hermitian of the left singular vector l that the MEOT is calculated with respect to.

$\{O\}$ is the operating shape vector for each order of interest at rpm r in no particular order.

$\{V_l\}$ is the right singular vector l that the MEOT is calculated with respect to.

If the autopower of a singular vector approaches the maximum singular value curve at more than one rpm value, multiple MEOTs can be calculated for the same left singular vector. These MEOTs can then be used together to obtain an estimate of the frequency and damping of the mode of interest. If the inputs to the system are not smooth functions, the MEOT may not have a clean SDOF FRF appearance due to additional peaks from the shape of the input forcing function. An example of this final case is shown in Chapter 4.

Chapter 4

4 Performance Evaluation of Order Tracking Methods and Virtual Measurements.

This chapter presents several test cases to evaluate and compare the performance of five of the order tracking methods presented in Chapters 2 and 3. The first set of examples compares performance of a digital resampling based method and the TVDFT. The last examples evaluate the ability of the Vold-Kalman filter and the TVDFT based methods to separate close and/or crossing orders. The final example is experimental data acquired on a two wheel chassis dynamometer. In this final example the inputs from the two front wheels of a vehicle are separated using the TVDFT.

The last section of this chapter computes the virtual measurements discussed in Chapter 3 from both a set of analytical data and an experimental set of data. These examples provide insight into the application of the virtual measurements. The experimental data uses a set of automobile operating data collected on a chassis dynamometer.

4.1 Slow Sweep, Rich Order Content Analytical Test Case.

An analytical dataset was generated with a fairly large number of orders, orders 1-8 and orders 5.5 and 6.5. The rpm in this example ramps from approximately 800 rpm to 4400 rpm in 30 seconds as a squared function of time. This type of rpm profile may be typical of data acquired on a chassis dynamometer. This dataset was generated to have 3 resonances to evaluate the performance of the order tracking methods when the amplitudes of the orders change very rapidly. A waterfall plot of this dataset is shown in Figure 4.1.

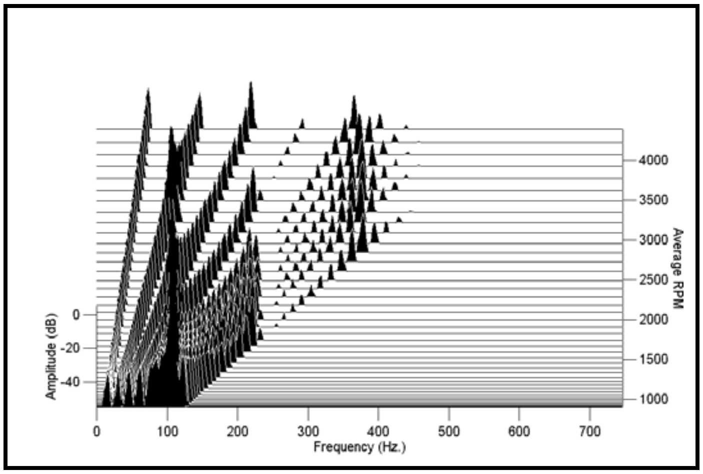

Figure 4.1: Waterfall plot fast slew rate data.

As can be seen in Figure 4.1, the orders are very close together at low rpm values. The amplitudes of the orders also appear to change very rapidly as the orders cross through a resonant frequency. Both of these phenomena can pose problems when tracking orders. Figure 4.2 shows this same dataset after it has been resampled from the time domain to the angle domain. It can clearly be seen in this figure that the orders have been straightened out and now fall on spectral lines. The orders have also been separated at the low rpm values. Note how the resonances do not fall on a spectral line in the angle/order domain.

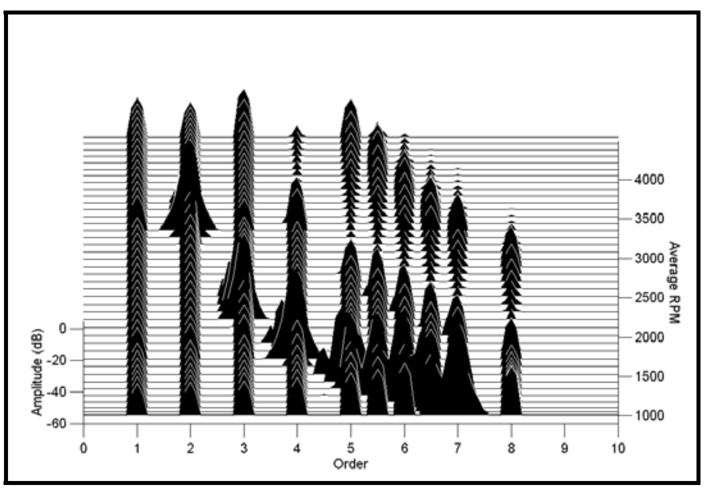

Figure 4.2: Waterfall plot of re-sampled angle/order domain data.

The order estimates of order 5.5 are shown in Figure 4.3. It can be seen in this figure that the three order tracking methods used, the TVDFT with both constant frequency and constant order bandwidths and the resampling method, all agree very well except at the resonances. The differences in the estimates at the resonances are due to the fact that the amplitude of the order changes very rapidly at this point. As was discussed in Section 2.3, all of the Fourier transform methods estimate the average amplitude of a signal over the total sample time. This property results in a larger estimate of the low frequency resonance by the TVDFT frequency bandwidth method than was obtained by either of the other two methods, due to the fact that the TVDFT frequency bandwidth method is averaging over a shorter time at the low rpm values. The TVDFT order bandwidth and the resampling methods both result in a higher amplitude estimate at the high frequency resonance because they are averaging over a shorter time at the high rpm values. This example clearly shows the time/frequency tradeoff that the different methods exhibit. Note that the TVDFT order bandwidth and the resampling methods give nearly identical order estimates.

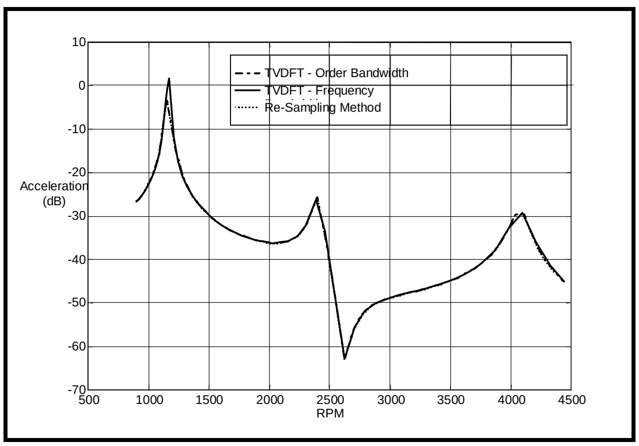

Figure 4.3: Order estimates of order 5.5.

The FFT based order tracking methods would not work well on this dataset due to the closeness of the orders shown in Figure 4.1. The waterfall plot clearly shows that at the lowest rpm values the orders are smeared into overlapping spectral lines. This smearing effect is the main limitation of the FFT based order tracking methods.

4.2 Analytical Crossing Order Test Case.

An analytical dataset was generated with both closely spaced and crossing orders. This dataset contains orders 3 and 3.1, which might be typical of a drive axle gearset. The dataset also has constant frequency orders that would be consistent with an electric fan in an automobile. The rpm was swept from approximately 900 rpm to 4500 rpm in a period of 16 seconds. This might be a common sweep rate for a vehicle on the road. The orders of the sweeping shaft as well as the orders of the constant frequency shaft were tracked both with and without OCM compensation. A Hanning window was used with a constant order bandwidth of 0.05 orders as defined by the sweeping frequency shaft. This implies that the 3 and 3.1 orders should be barely separable with the TVDFT method without OCM compensation.

A waterfall plot of the data is shown in Figure 4.4. Note that the closely spaced orders cannot be visually separated and that all but one of the sweeping orders interact with the constant frequency orders at the crossing rpms.

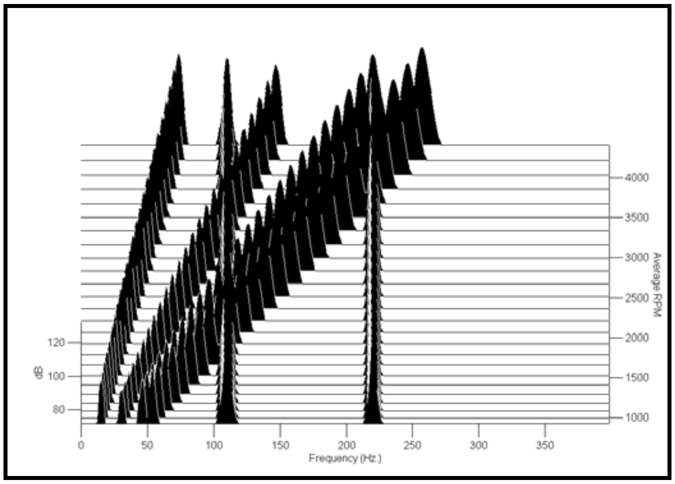

Figure 4.4: Waterfall plot of dataset with closely coupled and crossing orders.

All of the orders present were tracked using the TVDFT constant order bandwidth method without any compensation and with OCM compensation. The results of the order tracking without compensation are shown in Figure 4.5.

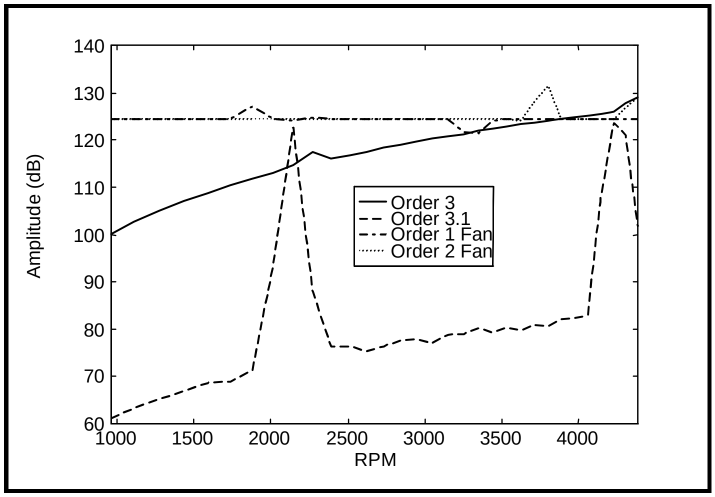

Figure 4.5: TVDFT constant order bandwidth order estimates without compensation.

Note the inability of the TVDFT to separate completely the close orders. The bottom curve is order 3.1 that should be 40 dB below order 3. The TVDFT cannot account for any of the interaction between the crossing orders. The resampling and FFT based methods would give similar results.

The results of the TVDFT order tracking with compensation are shown in Figure 4.6. Note the ability of the OCM to separate the close orders and to compensate for the order crossings almost perfectly for all orders. This shows that the compensation effectively increases the ability of the TVDFT method to both separate closely spaced orders and to simultaneously compensate for interactions between crossing orders.

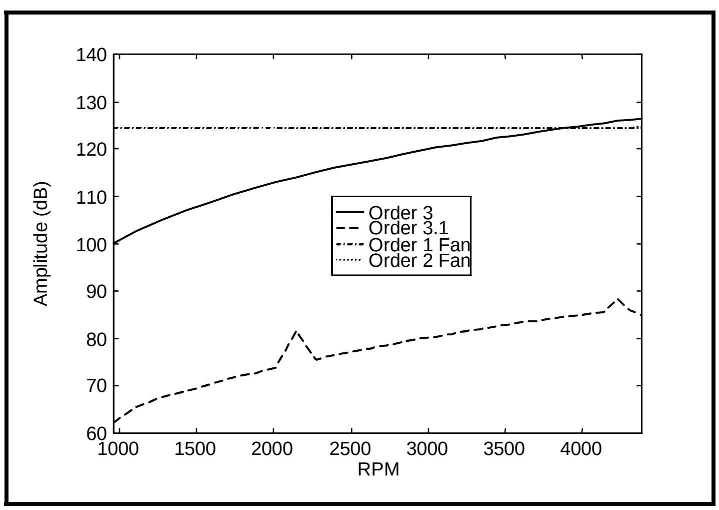

Figure 4.6: TVDFT constant order bandwidth estimates with OCM compensation.

The effectiveness of the OCM compensation is somewhat determined by how rapidly the amplitude of the orders change and how much interaction there is between orders. This limits the sweep rate that may be used to process data accurately. If the orders were constant in amplitude instead of amplitude varying, orders as close as 3 and 3.02 can be separated perfectly, even if order 3.02 is 60 dB below order 3 in amplitude. There is also perfect compensation for the crossing order interaction. These results are shown in Figure 4.7.

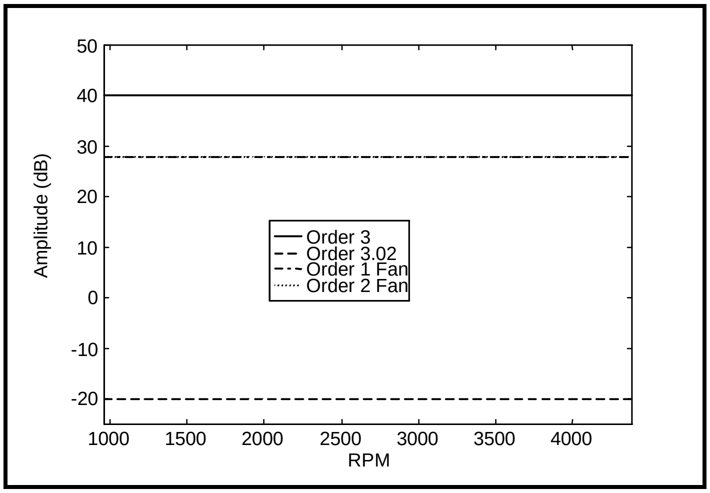

Figure 4.7: TVDFT constant order bandwidth estimates with OCM compensation.

Nearly the same sweeping orders as shown in Figure 4.6 were also tracked without the presence of the crossing orders. In this case, orders as close as 3 and 3.05 could be separated accurately. Remember, a Hanning window with a constant order bandwidth of 0.05 was used. This should result in a minimum separation of close orders of approximately 0.1 due to the window shape. This example thus shows that the ability to separate close orders with the OCM compensation can exceed what at first glance seems to be possible. The OCM has this ability because it compensates for the window characteristics, the shapes of the window's lobes are accounted for.

4.3 Separation of Driving Wheel Orders on Two Wheel Chassis Dynamometer.

This example uses actual experimental data acquired on a two wheel chassis dynamometer. A car was instrumented with a tachometer channel on each of the two front wheels. The car was a front wheel drive vehicle. Since the two wheel chassis dynamometer did not have independent control of the two wheels, the inflation pressure of the two tires varied. The inflation pressure between the two tires had a difference of 4 psi. Data was acquired over approximately 500 seconds with the speed swept from approximately 55 to 80 mph. The speed sweep had an rpm sweep of approximately 770 to 1050 rpm. The sweep was a computer controlled linear sweep.

The tachometer signals were processed by dividing the entire time block into 5 sections and fitting spline segments to the sections. Figure 4.8 shows the resulting rpm profiles from processing the two tachometer signals.

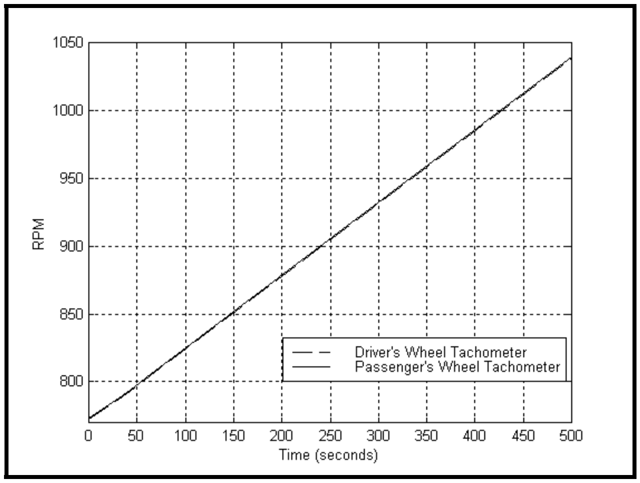

Figure 4.8: RPM profiles from driver's and passenger's tachometer signals.

Figure 4.9 shows a zoomed in portion of Figure 4.8 to show that there are indeed two separate tachometer signals present.

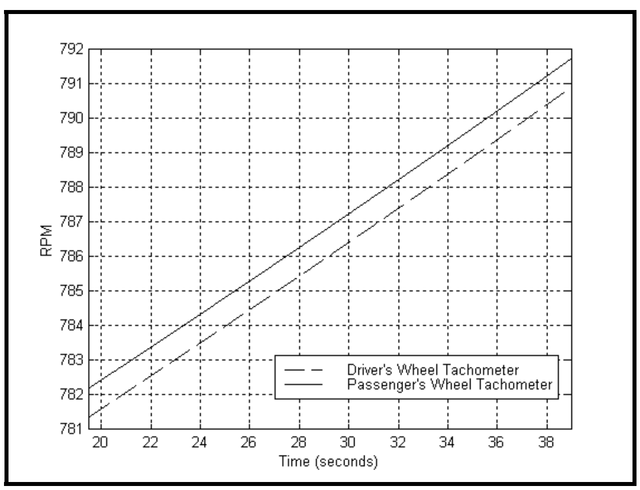

Figure 4.9: Zoom in of wheel RPM profiles.

The frequency difference between the rotational speeds of the two wheels is ~0.01 Hz. This leads to a difference of ~0.001 orders. Obviously, the separation of the 1st orders from the two wheels is a very difficult test condition.

4.3.1 Analytical Validation Test Data.

To evaluate whether the 1st orders are separable in this dataset an analytical dataset was created. This dataset was created using the processed rpm profiles to assure that there would be no errors due to inaccurate processing of the tachometer signals. A degree of freedom was chosen from the measured 45 dofs of the vehicle and processed to evaluate what shape the order profiles contained. This information was then used to generate the analytical dataset with a similar amplitude profile. This analytical example then gave a known answer that was very similar to the experimental data. The only orders present in the analytical dataset were the 1st orders of each tachometer signal.

Three different methods were used to attempt to extract the two 1st orders from the analytical dataset. The three methods were the Vold-Kalman 1st order filter, a time domain complex exponential based residue estimation coupled with a tracking order filter, and the TVDFT with OCM.

4.3.1.1 Vold-Kalman 1st Order Filter Order Separation.

The Vold-Kalman 1st order filter was used in an attempt to separate the two wheel orders. While the literature states that the Vold-Kalman filter has the ability to distinguish between two frequencies that are only 1/500 Hz apart if 500 seconds of data are acquired, in this case it did not have the ability to separate the two orders.

To determine just how close together the two orders could be and still get separation, a study was performed. In the study one of the rpm profiles was multiplied by a constant to shift the entire profile up or down. The two orders could be separated when one of the rpm profiles was 0.997 of the other rpm profile. In this case a weighting factor of 30,000 was used. The resulting order profiles were accurate for approximately the first 85% of the time block. The last 15% of the time block was in error because of the filter's settling time. The order estimates showed very little beating between the orders. Even though the order estimates contained a small amount of beating, they would be considered acceptable.

The experimental data has approximately a ratio of 0.9987 between the two rpm profiles. The Vold-Kalman 1st order filter cannot separate the two orders in this case, even with weighting factors in excess of 40,000. The beating effect between the two orders cannot be eliminated. The beating effect is a result of the incomplete separation of the orders.

The ultra-high weighting factors require the perfect tachometer signal that this analytically generated data possessed. The filter settling time at the end of the time history becomes unacceptably long with these ultra-high weighting factors as well. The final disadvantage of using the ultra-high weighting factors is the loss of the high dynamic range of the Kalman filters. The ultra-high weighting factors require a long time to converge on the amplitude of the filter and thus cannot follow rapid amplitude changes. The Kalman filters behavior is on a par with other order tracking techniques in this case with respect to the dynamic range of the order tracks.

4.3.1.2 Time Domain Residue/Tracking Filter Order Separation.

A combination of a tracking bandpass filter and a time domain residue estimation was used to attempt to separate the two wheel orders. This method was described in Section 1.2.4. The method uses a tracking bandpass filter to limit the number of orders present in the data to a known number, in this case the two wheel orders. A time domain residue estimation is then performed on the filtered time history. The rpm profiles are used to define the frequency of the complex exponential terms in the residue matrix. Both the positive and negative frequency orders must be included in this estimation. Overlap processing is used to obtain a fairly high number of rpm estimates. In this example approximately 100 seconds of data were used to obtain each estimate.

Using 100 seconds of data again limits the speed at which the order estimates may change. This leads to a relatively low dynamic range in the final results. Unfortunately, in this example the beating effect could not be entirely eliminated from the order estimates.

4.3.1.3 TVDFT with OCM Order Separation.

The last method that was used to attempt to separate the two wheel orders from one another was the TVDFT with OCM. This method was implemented with an order resolution of 0.0005 orders. This appears to be a narrower resolution than would be required based on the order separation, however a Hanning window was used for improved sideband rejection. The use of the Hanning window gives a wider effective resolution due to its sidelobe shape.

Figure 4.10 shows the analytical amplitude profiles and the estimated amplitude profiles for both wheel orders. Notice the differences between the estimated amplitudes and analytical profiles. The estimated profiles do not follow the exact shape of the analytical profiles because of the long integration time required to separate the orders. The long integration time means that the order amplitudes cannot change rapidly. It also limits the effective dynamic range.

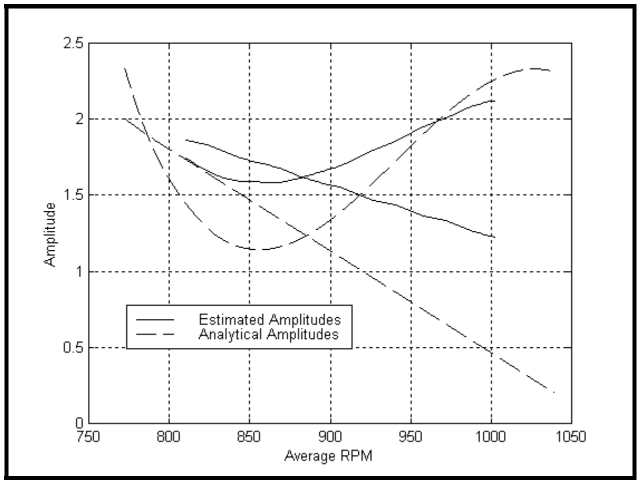

Figure 4.10: Analytical vs. Estimated Order Amplitudes with TVDFT w/OCM.

While the order estimates cannot follow the exact analytical amplitudes, they still follow the same general shape and do not exhibit any beating effects. This order resolution and window combination was then chosen to estimate the orders in the experimental data.

4.3.2 Experimental Data Evaluation.

Having verified that the TVDFT with OCM could separate and estimate the wheel orders with analytical dataset, the experimental dataset was then evaluated. All dofs were processed to obtain the data necessary for the generation of the virtual measurements in Section 4.5.

The order estimates from two different dofs are shown in Figure 4.11. It can be seen in these plots that there is significant information to be gained from having the ability to separate the left and right wheels of an automobile.

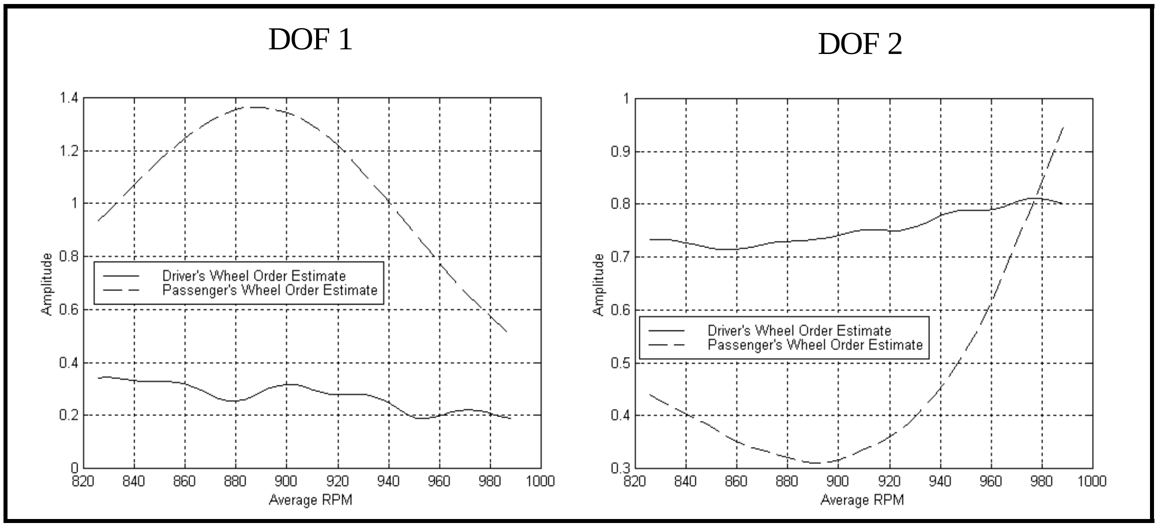

Figure 4.11: Order estimates for different dofs with order separation.

The left hand plot shows a dof where the driver's side wheel is the dominant contributor across the entire RPM range. The right hand plot, on the other hand, shows a dof where the passenger's side wheel is the dominant contributor across most of the RPM range. In fact, it appears in the left hand plot that there may be a resonance of some type in the 890 RPM area, while in the right hand plot it appears that there may be an anti-resonance in this region! Obviously, this information could prove to be invaluable in solving a noise or vibration problem.

To obtain an idea of what the results from a conventional order tracking method would provide, the wheel orders were tracked for the same two dofs. The result of this tracking is shown in Figure 4.12. Note that there is only one order track for each dof because the orders from the two wheels could not be separated.

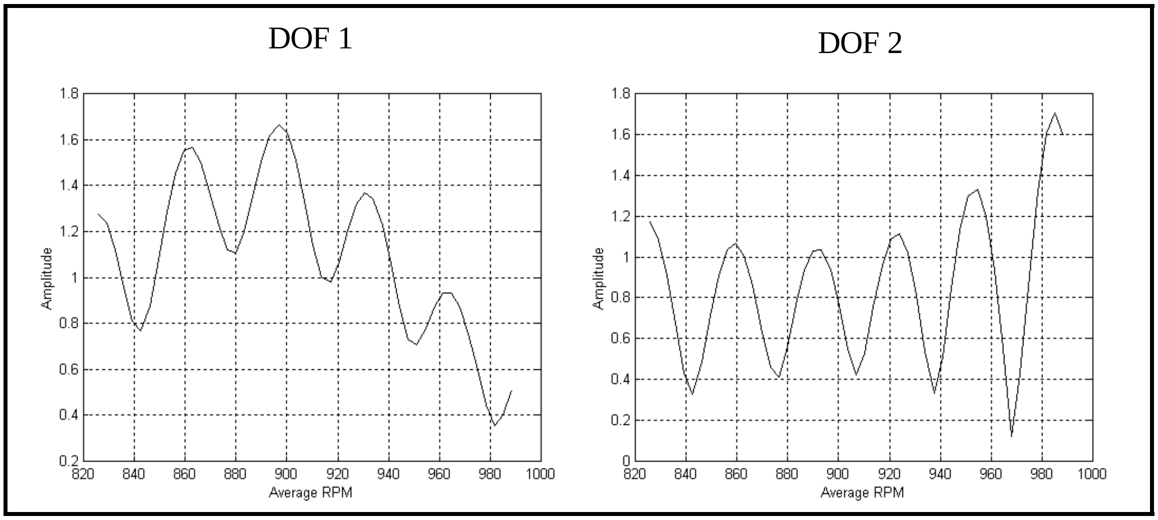

Figure 4.12: Order estimates for different dofs without order separation.

The order estimates shown in Figure 4.12 do not provide any information about which wheel may be the major excitation since there is only one order estimate for each dof. These estimates clearly show the beating interaction between the two wheels. It can be seen in these order tracks that the wheels come into and out of phase approximately 6 times in the time history.

This example explored the abilities of several different order tracking methods to separate very close orders on real experimental data. The only method that was able to effectively separate the contributions from the two wheels was the TVDFT with OCM. The results presented clearly show the additional valuable information that is gained from being able to separate the contributions from different wheels.

4.4 Virtual Measurement Test Cases.

Two different examples are presented which show the value of the virtual measurements formulated in Section 3.5. The first example is an analytical dataset while the second example uses the same experimental dataset used in Section 4.3.

4.4.1 Analytical Virtual Measurement Test Case.

An analytical example was created using experimentally determined modal parameters of a test structure in the SDRL laboratory, the H-frame. 26 modes are included. The analytical data was created with three simulated inputs attached to the structure. Each input was a combination of three sweeping sine functions which approximate the input of a rotating machine over a speed sweep. The first two inputs were driven by the same tachometer signal and are simply different combinations of orders from this signal. The third input was constructed such that in some frequency regions its orders would cross those of the first two inputs. This type of input scenario is typical of what might be observed on a structure with multiple uncoupled rotating components. A plot of the instantaneous frequencies of each order is shown in Figure 4.13.

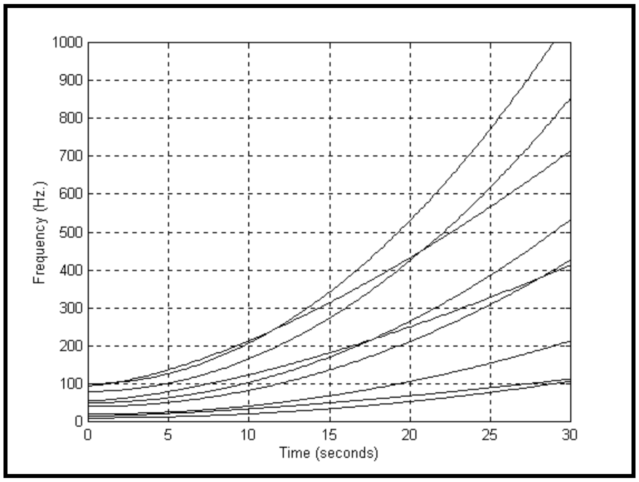

Figure 4.13: Instantaneous frequencies of inputs.

The TVDFT with OCM was used to track the nine input orders at 372 response degrees of freedom. These order tracks are estimated at 126 different rpm values. A combination of the nine orders tracked at 126 different rpm values gives 9*126=1134 different operating shapes to evaluate!

This test case has inputs with uniform amplitudes. The order tracks for this test case only peak in amplitude at rpm values where they excite a resonance of the system.

The first analysis performed was the calculation of the order track autopowers to obtain an estimate of which orders were dominant at each rpm value. A plot of these autopowers is shown in Figure 4.14.

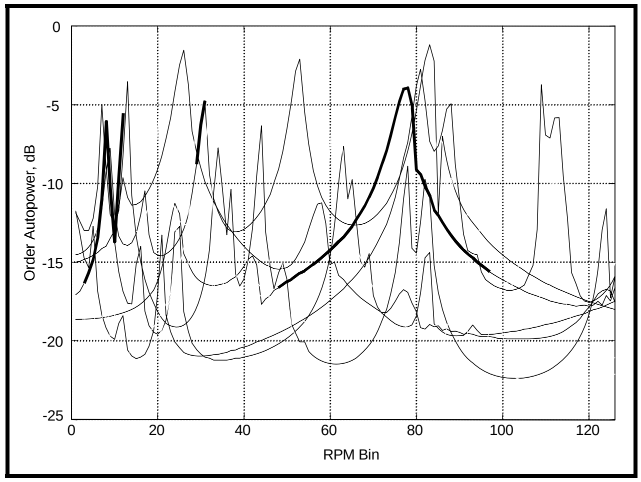

Figure 4.14: Order track autopower with tracked operating shape.

The next phase of analysis was to determine at what rpm values the same operating shapes were excited by tracking the operating shape from one order at one rpm value across all orders and rpm values. The orders and rpm values where the shapes are similar are distinguished by the thick line overlays at these rpms/orders in Figure 4.14. Note that the same operating shape is excited by different orders at different rpm values. This is typical of a case where multiple orders excite the same resonance.

To get a better understanding of whether the tracked operating shape is a mode shape or a combination of modes at each particular rpm, an SVD is performed on the entire set of operating shapes at each rpm value. This analysis yields the linearly independent set of shapes that combine to describe the previously observed operating shapes. These linearly independent shapes should more closely approximate the mode shapes of the system than the original operating shapes. A plot of the resulting singular values obtained at each rpm is shown in Figure 4.15. Note that there are nine singular value curves at every rpm.

From Figure 4.15, the number of singular value curves which peak at each rpm can be counted and an estimate of the number of significant linearly independent shapes excited at each rpm can be obtained. This is usually an estimate of the number of modes excited at each rpm value or point in time.

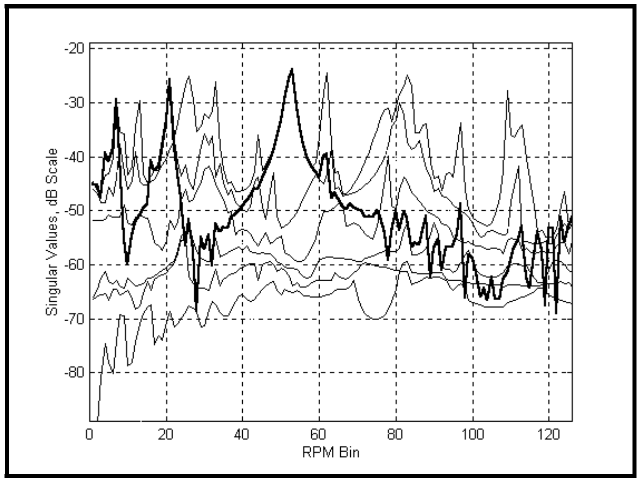

Figure 4.15: SVD as function of rpm with overlay of autopower of singular vector.

Using the left singular vector from a specific singular value and rpm, an autopower of the singular vector can be calculated across all rpm values. This result can be plotted as a function of rpm to obtain an understanding of at what rpm values the analyzed singular vector, or mode, is excited and how strongly it is excited. An autopower of a singular vector is computed and shown in Figure 4.15 by the heavy line. Note that this specific singular vector is excited at multiple rpm values.

Using both the left and right singular vectors from a specific singular value and rpm, a MEOT may be calculated. The MEOT should only peak to the level of its corresponding singular value at the rpm that it corresponds to. At all other rpm values, the MEOT should fall below the singular value curves. The MEOT should look very similar to a single degree of freedom system, similar to an enhanced FRF. The MEOT is not usually a perfect SDOF FRF in shape because at other rpm values where the same left singular vector is excited, it may peak up a small amount. A typical MEOT for this dataset is shown in Figure 4.16.

The operating shapes from each order and rpm can be organized according to frequency. Organizing the shapes in this manner results in a potentially different number of operating shapes at each frequency. Typically, there will be fewer operating shapes at the higher frequencies than at the lower frequencies. The higher frequencies are only excited by the higher orders whereas the lower frequencies are excited by both the lower and higher orders, resulting in many more estimated shapes at the lower frequencies.

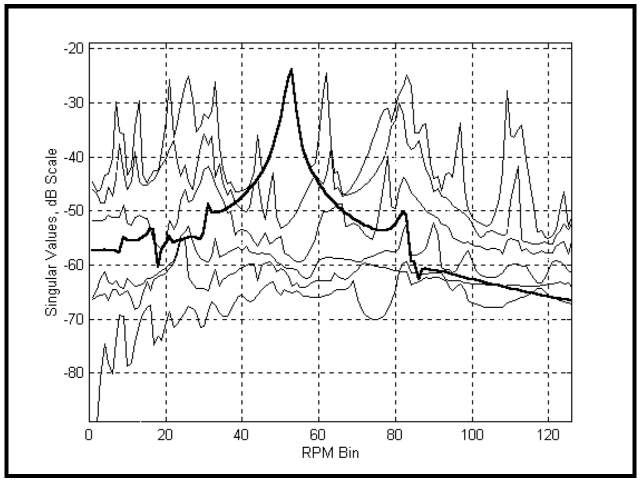

Figure 4.16: Example MEOT.

The operating shapes for this example were also organized according to frequency and an SVD performed at each spectral line, as shown in Figure 4.17. It can be seen that at many frequencies there is more than one mode excited in a frequency band. The frequency resolution, Δf, is 8 Hz in this example. There are actually many more singular value curves than are shown at the lower frequencies, they are not shown because of the dynamic range of the plot. These other singular value curves are all of very low amplitude and do not provide any information as to the number of independent shapes excited. They indicate that there are fewer independent shapes excited than there are order track estimated shapes at each of the lower frequencies, hence their low amplitudes.

In the high frequency region of this plot, there is only one singular value curve. This means that the estimated operating shapes may be a combination of modes and cannot be decomposed into its underlying linearly independent shapes.

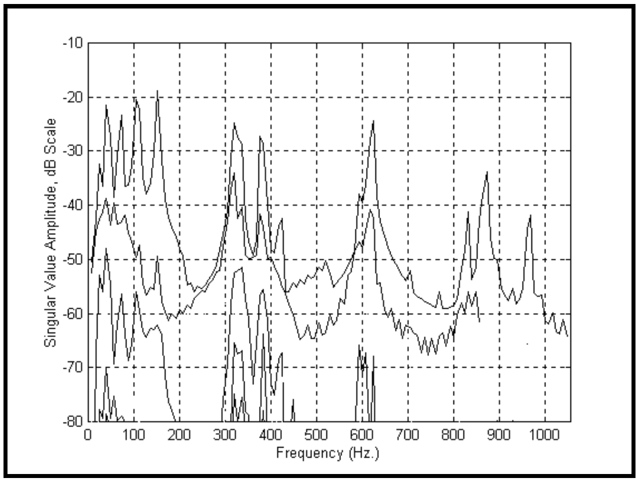

Figure 4.17: Singular values of order tracks as a function of frequency.

If the input to the system is not flat, but instead has peaks due to resonances in the path from the actual source to the structure being analyzed, the results may not be as clear or as easily interpreted [ref. 46].

4.4.2 Experimental Automotive Virtual Measurement Test Case.

As a second example to explore the behavior of the various virtual measurements that may be calculated, the dataset used in Section 4.3 was analyzed. The dataset was acquired on an automobile on a two wheel chassis dynamometer and the contributions from the two front wheels separated using the TVDFT with OCM.

The first virtual measurements that were calculated were the order autopowers to evaluate whether the right or left wheel was the dominant overall contributor to the measured dofs at each rpm. The order autopowers are shown in Figure 4.18.

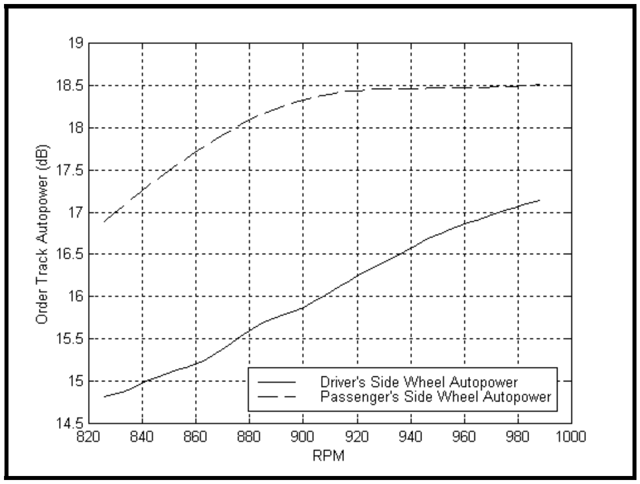

Figure 4.18: Order track autopowers for two wheels.

Figure 4.18 shows that there are no resonances present in the rpm range of the data. It can clearly be seen that the passenger's side wheel is the dominant overall contributor to the measured dofs at all rpm values. It is also interesting to note that while the driver's side autopower increases in amplitude with increasing rpm, as would be expected with an unbalance force, the passenger's side autopower does not.

An SVD was performed on the orders as a function of rpm to allow the computation of linearly independent operating shapes and a MEOT. A MEOT is shown in Figure 4.19.

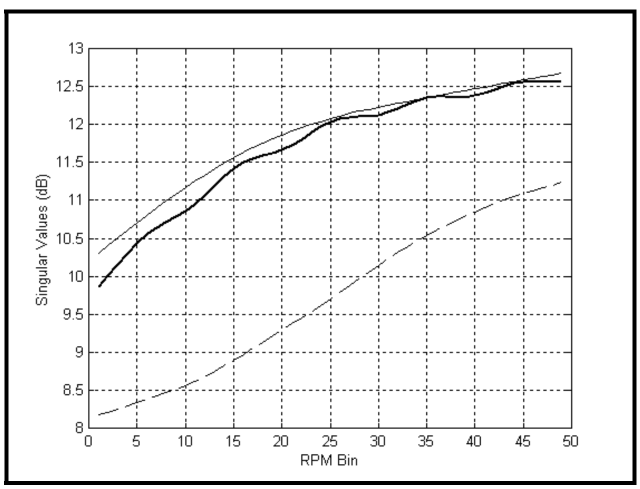

Figure 4.19: Singular value curves with MEOT overlay for two wheels.

Notice the difference between the MEOT computed in the analytical example and this example. In this example the MEOT does not appear like an SDOF system because the forcing functions of the wheels come back into phase approximately 6 times in the sweep. Each time that the wheels are in phase with one another one mode is excited while when the wheels are out of phase a different mode is excited. Also realize that the frequency of the orders only sweeps from approximately 15 to 17.5 Hz. The same two modes are excited at all rpm values. The dominantly excited mode remains the dominant mode at all rpm values, it just decreases in amplitude slightly as the tires go into phase with one another. The MEOT in this case only serves to show when the same excitation is input into the vehicle as far as the unbalance phasing of the two wheels, this is determined from observing when the MEOT approaches the top singular value. Anytime the orders that excite a system can come into and go out of phase with one another there is a chance that the MEOT will appear as it does in this example and not like an SDOF system as seen in the previous example.

Chapter 5

5 Conclusions and Recommendations.

5.1 Summary and Conclusions.

One of the goals of this dissertation was to document many of the order tracking techniques that have been developed and used over time, both commercial and non-commercial. Many of the commercially available order tracking methods were presented in Chapter 2. The commercially available methods presented include time sample based FFT, digital resampling based angle domain, and Kalman/Vold-Kalman filter based methods. The theory of each method was presented, followed by practical issues related to each method. Therefore, Chapter 2 should serve as a single source for an explanation on the details of each of the currently available order tracking methods.

Chapter 2 also presented some order tracking methods that are not commercially available. These non-commercial algorithms originated either in published papers or through discussions on how order tracking has been done in the past. The FRF based methods which were discussed have not been thoroughly documented previously but have been used in the past.

No order tracking reference could be complete without documentation and explanation of tachometer processing methods since all methods intimately rely on an rpm estimate. Several commercially available tachometer processing methods were presented in Chapter 2. These methods include algorithms that are best suited for both real time and post-processing applications. The limitations of each method were presented and documented. It should be emphasized that the tachometer channel is the most important analysis channel in order tracking analysis. Without an accurate tachometer channel no order tracking can be done on any of the response channels.

Obviously, a dissertation includes new work along with a summary of historical work. To satisfy this requirement, Chapter 3 included the development of several new order tracking approaches. A tachometer processing algorithm that can be used to efficiently process induction based tachometer probes is presented. Chapter 3 also contains the development of a completely new order tracking method, the TVDFT, and a new variation on the FRF based order tracking method presented in Chapter 2.

The TVDFT has the capability to separate both close and crossing orders if an orthogonality compensation matrix is used (OCM). This ability to separate close and crossing orders was documented through both analytical and experimental examples in Chapter 4. The TVDFT was used to separate the two order inputs of the front wheels of an automobile in the experimental example where the orders were only ~0.001 orders apart! For most order tracking cases where there are no close or crossing orders present, the TVDFT was shown to provide results that are very similar to those of the angle domain resampling based methods. This is to be expected since the TVDFT essentially resamples the kernel of the Fourier transform as opposed to resampling the response. The TVDFT is therefore much more computationally efficient than the resampling based methods.

Finally, several new virtual measurements were developed which are based on the SVD and the CMIF algorithm used in modal analysis. These virtual measurements are useful for condensing large amounts of actual data down to a single virtual measurement. The SVD was also used to decompose sets of order tracking based operating shapes into sets of linearly independent operating shapes. In many cases these linearly independent shapes will approximate the mode shapes of a system. A virtual measurement was developed based on the left and right singular vectors from the SVD to condense order tracking measurements into a single measurement which in many cases has the same properties as an enhanced FRF. This measurement was called a Mode Enhanced Order Track (MEOT).

As support for the digital resampling based order tracking algorithms, two adaptive resampling algorithms were presented. One of the algorithms is commercially implemented and has been documented. The other algorithm has not been documented. It is not known whether this method is new, due to the proprietary nature of the commercially implemented resampling algorithms. The potentially new adaptive resampling method is based on an upsampled interpolation filter and is shown to be much more computationally efficient and memory efficient than the known commercial method. To further the understanding of the adaptive resampling procedures, Appendix B provides background information on both digital interpolation and decimation.

To summarize then, this dissertation includes a fairly complete reference of digital order tracking techniques that should be a valuable one-stop reference. This dissertation also includes the development and application of the TVDFT order tracking method which has the ability to estimate orders which are very close together or even cross one another. Finally, the computational efficiency of both the TVDFT and the adaptive resampling based on the upsampled interpolation filter provide a set of order tracking tools that can be implemented on the increasing popular personal computer.

5.2 Recommendations for Future Work.

The FRF based order tracking method presented in Chapter 3 has not been fully implemented or evaluated. This method should be evaluated in the future as it should be able to separate both close and crossing orders, and therefore add another tool a growing suite of powerful tools for these types of datasets. The major advantage this method should possess over tracking filter type order tracking methods is the ability to estimate the transient response of a system.

Further work should also be done to evaluate the usefulness and applications of the virtual measurements developed in Chapter 3. The MEOT should prove to be a valuable virtual measurement which may allow the estimate of the natural frequency and damping of a mode which is excited by sweeping orders. The ability to estimate both frequency and damping of a mode from operating data would be very valuable in cases where there are non-linearities present in the system. The frequency and damping would be estimated under actual operating conditions.

Finally, all of the new methods must be used on many more real world experimental datasets to evaluate and validate their advantages. It is impossible to evaluate these methods under every possible condition with analytical simulations because there are an infinite number of operating data scenerios that may arise. There is no substitute for using new algorithms in the field to fully understand and exploit their strengths and weaknesses!

References.

1. Van de Ponseele, P., Van der Auweraer, H., and Mergeay, M., *"A Global Approach to the Acquisition and Analysis of Harmonic Waveforms"*, Proceedings of International Modal Analysis Conference 7, Las Vegas, NV., pp.1290-1299, 1989.

2. Van de Ponseele, P., Van der Auweraer, H., and Mergeay, M., *"Performance Evaluation of Advanced Signature Analysis Techniques"*, Proceedings of International Modal Analysis Conference 7, Las Vegas, NV., pp.154-158, 1989.

3. Bandhopadhyay, D.K. and Griffiths, D., *"Methods for Analyzing Order Spectra"*, Proceedings of Society of Automotive Engineers 1995 Noise and Vibration Conference, SAE paper number 951273.

4. Harris, C.M., *"Shock & Vibration Handbook"*, McGraw-Hill, 3rd Edition, pp. 13:42-13:43, New York, New York, 1987.

5. Potter, R. and Gribler, M., *"Computed Order Tracking Obsoletes Older Methods"*, Proceedings of the Society of Automotive Engineers Noise and Vibration Conference, Traverse City, MI., 1989, SAE paper no. 891131.

6. Potter, R., *"A New Order Tracking Method for Rotating Machinery"*, Sound and Vibration Magazine, Vol. 24, No. 9., September 1990.

7. Hewlett Packard, *"Appendix A – Computed Synchronous Resampling and Order Tracking"*, Effective Machinery Measurements using Dynamic Signal Analyzers, Application Note 243-1, pp. 64-69.

8. Potter, R.W., *"Tracking and Resampling Method and Apparatus for Monitoring the Performance of Rotating Machines"*, United States Patent Number 4912661, March 27, 1990.

9. McDonald, D. and Gribler, M., *"Digital Resampling – A Viable Alternative for Order Domain Measurements of Rotating Machinery"*, Proceedings of International Modal Analysis Conference 9, Italy, pp.1270-1275, 1991.

10. Kirtley, N., *"The Fine Art of Order Tracking"*, Hewlett Packard Realtime Update, Spring 1994, pp. 1-4.

11. Hewlett Packard, *"Realtime Basics: Order Analysis"*, Hewlett Packard Realtime Update, Fall 1996-Winter 1997, pp. 8-9.

12. Fyfe, K.R. and Munck E.D.S., *"Analysis of Computed Order Tracking"*, Mechanical Systems and Signal Processing, Volume 11, No. 2, 1997, pp. 187-205.

13. Lembregts, F., Top, J., and Neyrinck, F., *"Adaptive Resampling for Off-line Signal Processing"*, Proceedings of the International Seminar of Modal Analysis - 21, Leuven, Belgium, 1996, pp. 1171-1182.

14. Vancauter, R., *"A Doppler Correction Procedure for Exterior Pass-By Noise Testing"*, Proceedings of Society of Automotive Engineers 1997 Noise and Vibration Conference, SAE paper number 971987.

15. Gade, S, Herlufson, H., and Konstantin-Hansen, H., *"Diagnostics in Powerplant Station Using Zoom Order Tracking"*, Proceedings of the International Seminar of Modal Analysis - 21, Leuven, Belgium, 1996, pp. 1527-1534.

16. Gade, S, Herlufson, H., and Konstantin-Hansen, H., *"New PC and MS-Windows Based Multichannel Digital Order Tracking Analysis System"*, Proceedings of the International Seminar of Modal Analysis - 21, Leuven, Belgium, 1996, pp. 1535-1539.

17. Vold, H. and Leuridan, J., *"High Resolution Order Tracking at Extreme Slew Rates, Using Kalman Tracking Filters"*, Proceedings of the SAE Noise and Vibration Conference, Traverse City, MI., 1993, SAE paper no. 931288.

18. Leuridan, J., Vold, H., Kopp, G., and Moshrefi, N., *"High Resolution Order Tracking Using Kalman Tracking Filters - Theory and Applications"*, Proceedings of the Society of Automotive Engineers Noise and Vibration Conference, Traverse City, MI., 1995, SAE paper no. 951332.

19. Vold, H., Mains, M., and Blough, J.R., *"Theoretical Foundations for High Performance Order Tracking with the Vold-Kalman Filter"*, Proceedings of Society of Automotive Engineers 1997 Noise and Vibration Conference, SAE paper number 972007.

20. Vold, H. and Deel, J., *"Vold-Kalman Order Tracking: New Methods for Vehicle Sound Quality and Drive-Train NVH Applications"*, Proceedings of Society of

Automotive Engineers 1997 Noise and Vibration Conference, SAE paper number 972033.

21. Vold, H., Herlufson, H., Mains, M., and Corwin-Renner, D., *"Multi Axle Order Tracking with the Vold-Kalman Tracking Filter"*, Sound and Vibration Magazine, May 1997, pp.30-34.

22. LMS International, *"Evolving the Approach to Order Tracking, A review of recent LMS developments"*, LMS International News Letter, Volume 12, Number 2, pp. 5-7.

23. Vold, H., *"Multichannel Signal Processing - Course Notes"*, 1990.

24. Randall, R.B. and Tech, B., *"Frequency Analysis"*, Bruel & Kjaer, 3rd Edition, Denmark, September 1987.

25. Weber, P., *"Correlating Operating Data with Modal Testing: An Automotive Application"*, Master of Science Thesis, University of Cincinnati, Cincinnati, Ohio, 1988.

26. Vold, H., Crowley, J., and Nessler, J., *"Tracking Sine Waves in Systems with High Slew Rates"*, Proceeding of 6th International Modal Analysis Conference, Kissimmee, FL., 1988, pp.189-193.

27. Vold, H. and Crowley, J., *"Time Variant Spectral Analysis using the Maximum Entropy Method"*, Proceeding of 6th International Modal Analysis Conference, Kissimmee, FL., 1988, pp.1403-1406.

28. Van der Auweraer, H., Leuridan J, and Vold, H., *"The Analysis of Nonstationary Dynamic Signals"*, Sound and Vibration Magazine, August 1994, pp. 14-26.

29. Van der Auweraer, H., Leuridan J, and Vold, H., *"Analysis of Nonstationary Noise and Vibration Signals"*, Proceedings of the International Seminar of Modal Analysis - 19, Leuven, Belgium, 1994, pp. 385-405.

30. Leuridan, J., Van der Auweraer, H., and Vold, H., *"The Analysis of Nonstationary Dynamic Signals"*, Sound and Vibration Magazine, Vol. 28, No. 8., August 1994, pp. 14-26.

31. Grewal, M.S. and Andrews, A.P., *"Kalman Filtering - Theory and Practice"*, Prentice Hall Inc., Englewood Cliffs, New Jersey, 1993.

32. Haykin, S., *"Adaptive Filter Theory"*, Second Edition, Prentice Hall Inc, Englewood Cliffs, New Jersey, 1991.

33. James, M.L., Smith, G.M., and Wolford, J.C., *"Applied Numerical Methods For Digital Computation"*, Third Edition, Harper & Row Publishers Inc., New York, New York, 1985.

34. Flannery, B.P., Press, W.H., Teukolsky, S.A., and Vetterling, W.T., *"Numerical Recipes in Fortran – The Art of Scientific Computing"*, Second Edition, Cambridge University Press, New York, New York, 1992.

35. Blough, J.R., Dumbacher, S.M., and Brown, D.L., *"Time Scale Resampling to Improve Transient Event Averaging"*, Proceedings of Society of Automotive Engineers 1997 Noise and Vibration Conference, SAE paper number 972005.

36. Blough, J.R., Brown, D.L., and Vold, H., *"Order Tracking with the Time Variant Discrete Fourier Transform"*, Proceedings of the International Seminar of Modal Analysis - 21, Leuven, Belgium, 1996.

37. Blough, J.R., Brown, D.L., and Vold, H., *"The Time Variant Discrete Fourier Transform as an Order Tracking Method"*, Proceedings of Society of Automotive Engineers 1997 Noise and Vibration Conference, SAE paper number 972006.

38. Rabiner, L. and Gold, B., *"Theory and Application of Digital Signal Processing"*, Prentice Hall International, London, pp.390-399, 1975.

39. IEEE Acoustics, Speech, and Signal Processing Society, *"Programs for Digital Signal Processing"*, IEEE Press, New York, New York, 1979.

40. Crochiere, R.E. and Rabiner, L.R., *"Multirate Digital Signal Processing"*, Prentice Hall, Englewood Cliffs, New Jersey, 1983.

41. Watkinson, J., *"The Art of Digital Audio"*, Focal Press, Oxford, England, 1991.

42. Dossing, O., *"Structural Stroboscopy-Measurement of Operational Deflection Shapes"*, Sound and Vibration Magazine, Vol. 22, No. 8, pp.18-26, 1988.

43. Martin, M.W., *"Modal Decomposition of Operating Data in a Multi-Source Environment"*, Master of Science Thesis, Michigan Technological University, Houghton, Michigan, 1995.

44. Otte, D., Van de Ponseele, P., and Leuridan, J., *"Operational Deflection Shapes in Multisource Environments"*, Proceedings of Eighth International Modal Analysis Conference, 1990, pp 413-421.

45. Allemang, R., *"Vibrations: Experimental Modal Analysis"*, University of Cincinnati, Cincinnati, Ohio, April 1998. (UC-SDRL CN-20-263-663/664)

46. Blough, J.R., Brown, D.L., and VanKarsen, C., *"Independent Operating Shape Determination of Rotating Machinery, Based on Order Track Measurements"*, Proceedings of Sixteenth International Modal Analysis Conference, 1998.

47. Blough, J.R. and Brown, D.L., *"Estimation of Independent Operating Shapes, Based on Order Track Measurements"*, Proceedings of International Session on Modal Analysis –23, Leuven, Belgium, 1998.

48. Sparks, C. and Wachel, J.C., *"Quantitative Signature Analysis for On-Stream Diagnosis of Machine Responses"*, Symposium on On-stream Nondestructive Examination of Rotating Machinery, 1972 Fall ASNT Conference, Cleveland, Ohio, 1972.

49. Anderson, J.J. and Babkin, A.S., *"Mechanical Signature Analysis of Ball Bearings by Real Time Spectral Analysis"*, Application Note 3, Federal Scientific Corporation, New York, New York, 1972.

50. Hewlett Packard, *"Signature Analysis Software Reference Manual 5451B (Option 450)"*, Hewlett Packard, 1974.

51. Hewlett Packard, *"Dynamic Signal Analyzer Applications – Application Note 243-1"*, Hewlett Packard, 1983.

52. Litz, J., *"High-Powered Analysis of Vibration Spectra"*, Machine Design, January 26, 1984.

53. Hewlett Packard, *"Using the 3561A Dynamic Signal Analyzer – Product Note 3561A-1"*, Hewlett Packard.

54. Fladung, W.A, *"The Development and Implementation of Multiple Reference Impact Testing"*, Master of Science Thesis, University of Cincinnati, 1994.

55. Shih, C.Y, Tsuei, Y.G., Allemang, R.J., and Brown, D.L., *"A Frequency Domain Global Parameter Estimation Method for Multiple Reference Frequency Response Measurements"*, Proceedings of Sixth International Modal Analysis Conference, 1988, pp 389-396.

56. Phillips, A.W., Allemang, R.J., and Fladung, W.A., "The Complex Mode Indicator Function (CMIF) as a Parameter Estimation Method", Proceedings of Sixteenth International Modal Analysis Conference, 1998, pp 705-710.

Appendix A

Time Domain to Angle Domain, Why Do We Go There?

A Why Sample in the Time Domain?

Surprisingly enough, all digital signal processing (DSP) theory is developed in the angle domain and not the time domain. This statement is contrary to popular opinion but this is the reason for this appendix. General DSP theory is seldom presented in terms of the angle domain but is in fact developed around the angle domain. It will be shown in this appendix that the time domain is simply a special case of the angle domain for steady state signals.

DSP theory is typically presented in terms of the time and frequency domains because these domains have physical significance that almost everyone can relate to. These domains may have physical significance but they have no theoretical significance in DSP theory.

From an educational perspective, it is easier to develop the theory in a domain that most people are somewhat familiar with. The disadvantage to developing theory based on a special case, the time domain, instead of the general case, the angle domain, is the lack of realization and understanding that time/frequency domain algorithms are applicable to other types of data. The same algorithms and methods of the time/frequency domains may be applied to data in the angle domain, or any other domain developed from the angle domain.

From a practical perspective, much of the DSP theory is presented based on the time and frequency domains because these are the domains in which the commercially available dynamic signal analyzers operate. Commercial dynamic signal analyzers (DSAs) sample data relative to the time axis because it is an independent axis that samples can always be acquired relative to. Since all DSAs have a built-in sample clock, this reference is always available and is very accurate. Many commercial DSAs have an external sample clock input to allow sampling relative to an independent axis which may be generated for a specific application. Another advantage of sampling relative to the time axis is that it is easy to implement the necessary analog anti-alias filter relative to a constant frequency. In this case a tracking filter is not necessary because the sample rate does not change as a function of time.

A.1 Time and Frequency Domains.

As a first step in explaining the angle domain, several of the basic sampling criteria will be presented relative to time and frequency. The time and frequency domain sampling theorems are presented to attempt to familiarize the reader with the terminology that will be used to explain the angle domain concepts.

The sampling equations that are normally presented and which most readers are familiar with are shown below in Equation A.1. Notice that these equations have been written with respect to the sampling interval Δt.

$$\begin{aligned} \Delta f &= \frac{1}{T} = \frac{1}{N * \Delta t} \\ T &= N * \Delta t \\ F_{nyquist} &= F_{\max} = \frac{F_{sample}}{2} \\ F_{sample} &= \frac{1}{\Delta t} \end{aligned} \tag{A.1}$$

where: Δf is the frequency resolution of the resulting frequency spectrum.
T is the total sample time analyzed.
N is the total number of time points over which the transform is performed.
Δt is the time spacing of the discrete time samples.
F_{sample} is the sample frequency of the data.
$F_{nyquist}$ is the Nyquist frequency.
F_{max} is the maximum frequency which can be analyzed.

The equations shown in A.1 must be enforced in the acquisition of all data that will be Fourier transformed. The Nyquist frequency, $F_{nyquist}$, and frequency resolution, Δf, of the Fourier transform are shown to be related to the sample frequency, F_{sample}, or the spacing between the acquired time samples, Δt.

Since the Nyquist frequency is the highest frequency that can be analyzed with a Fourier transform, it can be seen that the data must be acquired with a sample frequency of at least two times the Nyquist frequency to avoid aliasing. Simply acquiring data with a sample frequency that is twice the Nyquist frequency however does not eliminate aliasing. All dynamic signal analyzers contain an analog low pass filter before the ADC of each acquisition channel. The cutoff frequency of the analog anti-alias filter is at a maximum equal to the Nyquist frequency. Since a perfect filter cannot be constructed, the cutoff frequency is often in the neighborhood of 20% below Nyquist. This analog low pass filter is necessary to prevent aliasing. For this reason, every digital data acquisition system must contain an analog anti-alias filter.

The most common analysis that is done on discretely sampled data in the noise and vibrations industry uses the Fourier transform. The Fourier transform defined in terms of time and frequency is shown in Equation A.2.

$$
\begin{aligned}
a_m &= \frac{1}{N}\sum_{n=1}^{N} x(n\Delta t)\cos(2\pi f_m n\Delta t) \\
b_m &= \frac{1}{N}\sum_{n=1}^{N} x(n\Delta t)\sin(2\pi f_m n\Delta t)
\end{aligned}
\qquad \text{(A.2)}
$$

where: $x(n\Delta t)$ is the nth discrete data sample.
a_m, b_m are the estimated Fourier coefficients.
f_m is frequency $m\Delta f$.

The only frequencies that can be analyzed without error using a discrete Fourier transform are those that fall on a spectral line or Δf. The special property that these frequencies possess is that they all contain an integer number of cycles in the integration time, T. All frequencies that fall between spectral lines will be estimated with a leakage error. While leakage can be reduced with a window, it can never be eliminated.

One point that should be made is that up to this point all of the signals that have been acquired and analyzed are assumed to be steady state in both amplitude and frequency. If the Fourier transform is defined in terms of constant Δt's and therefore Δf's, it is very well suited to this type of analysis. The time/frequency domain Fourier transform performs very well with steady state frequency data because the kernel of the transform looks the same as the frequency components which are being analyzed. This property is shown in Figure A.1.

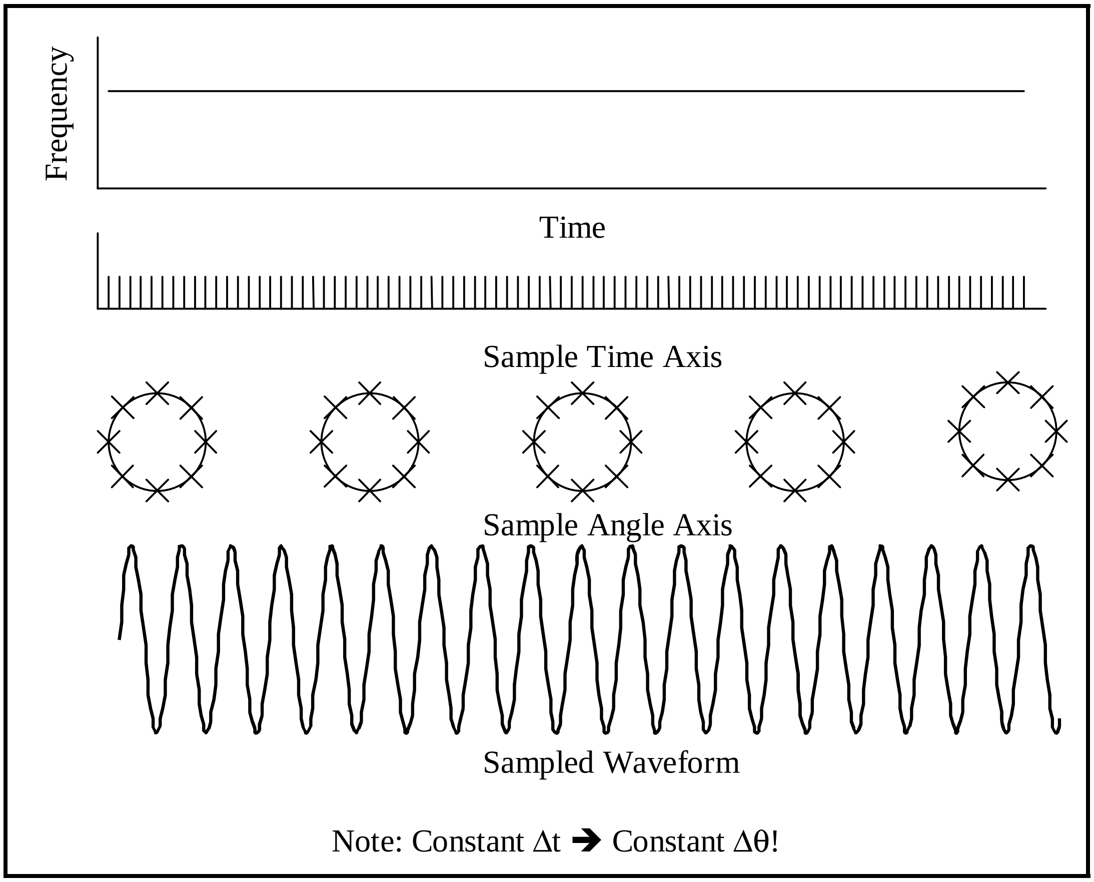

Figure A.1: Constant frequency time domain sampled data.

The key to understanding Figure A.1 is that the sampled waveform, the bottom plot, looks like a sine or cosine wave. Notice also that this figure shows that for the case of constant frequency data, a constant Δt is also exactly constant $\Delta\theta$ data, where $\Delta\theta$ is the angle domain sample spacing. The relationship between the Δf and $\Delta\theta$ sample rates is shown in the middle two plots of Figure A.1. In other words, the derivations that are usually presented that are based on constant Δt are actually based on constant $\Delta\theta$! This implies that the Fourier transform and sampling relationships that usually presented are just a special case of the general angle domain case. One last thing to note is that the sampled waveform in this case is of constant amplitude. The kernel of the transform is also of constant amplitude. In many cases the sampled waveform will not be of constant amplitude, this will result in an approximation of the amplitude of the signal since the average value of the amplitude over the integration time will be estimated by the Fourier transform.

A similar presentation could also be made to show that digital filters are actually based on a constant $\Delta\theta$ spacing. Digital filters are nearly always presented and developed in the z-domain where the coefficients of a filter are applied to consecutively delayed data. The mathematical derivation often does not state that these delays are delays in time, just delays in the sample stream. The theoretical development of digital filters is also always done with frequency normalized parameters, normalized with respect to the Nyquist frequency, such as bandstop/bandpass frequencies. It is only in actual examples that the jump is made to time domain sampled data and non-normalized frequency parameters. Obviously then these same filters may be and are applied to angle domain data.

Since the digital decimation and interpolation procedures of Appendix B are based on simple digital filters, they may also be used on angle domain data. The Kalman and Vold-Kalman filters of Chapter 2 may also be formulated to operate in the angle domain very easily. All of these filters will of course contain a constant order bandwidth in the angle domain and not a constant frequency bandwidth.

A.2 Angle and Order Domains.

Understanding that Fourier transforms and digital filters can be applied to angle domain data, next an understanding must be gained into why data is analyzed in the angle domain. The major reason to work in the angle domain is to accurately analyze data that contains time variant frequency components. Time variant frequencies can be evaluated very accurately if the sample rate of the discrete data can be locked to the frequency variation of these components.

To understand why locking the sample rate of data to a reference signal is beneficial in obtaining accurate estimates of time varying frequency components in the data, an understanding must be gained of what this type of sampling does to the data. Figure A.2 shows the same information as Figure A.1, except in this case the frequency of the data varies with time as shown by the top plot in the figure.

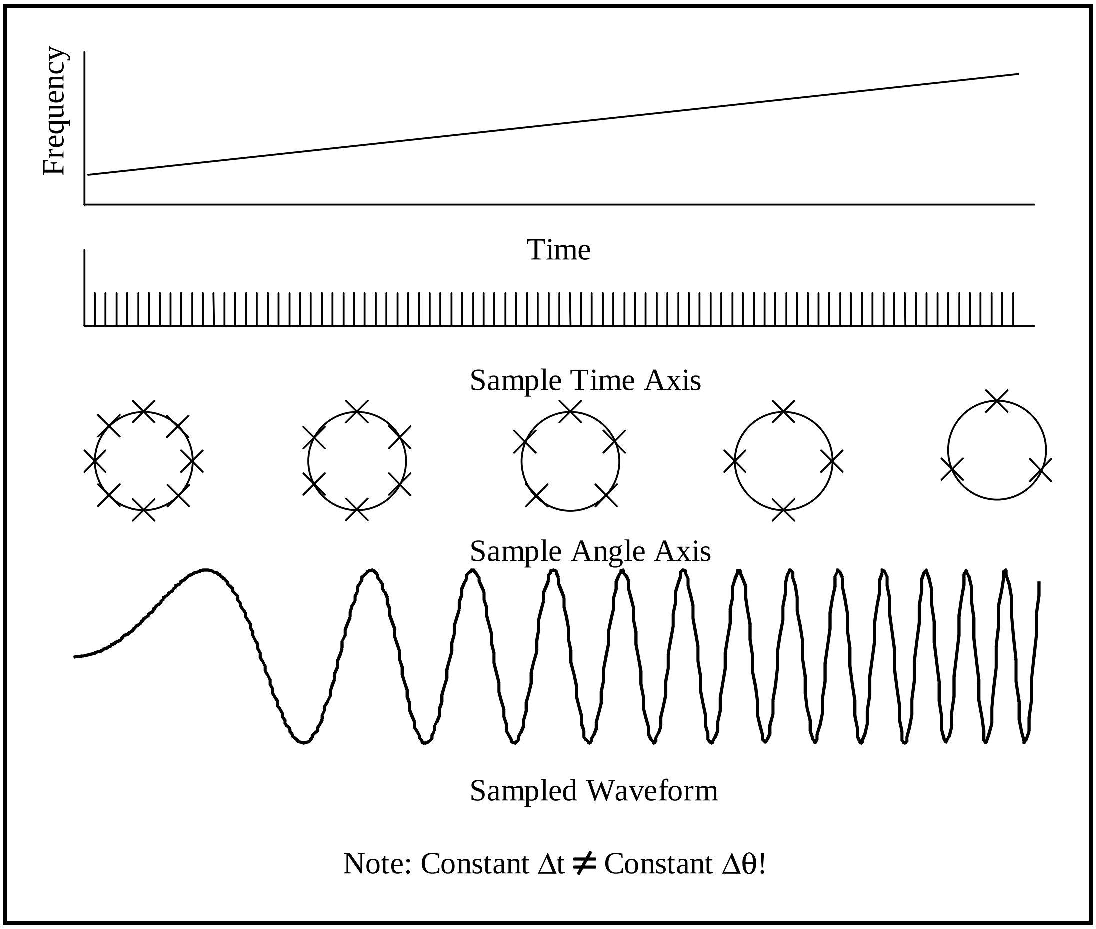

Figure A.2: Varying frequency sine wave sampled with a constant Δt.

The data shown in the bottom of the figure shows that if a constant Δt is used to sample this data, the sampled data does not look like the kernel of the time based Fourier transform. Remembering that accurate results will be obtained from a transform if the data looks similar to the kernel of the transform, clearly it can seen that this is not an optimal case. It is also clear from the time and angle sample rate plots shown in the middle of Figure A.2 that the constant Δt sample rate is not a constant Δθ. This implies that the relationship between the time and angle domain sample rates is different than it was in Figure A.1.

With a reference signal, this same varying frequency sine wave can be sampled with a constant angular sample rate. The result is shown in Figure A.3.

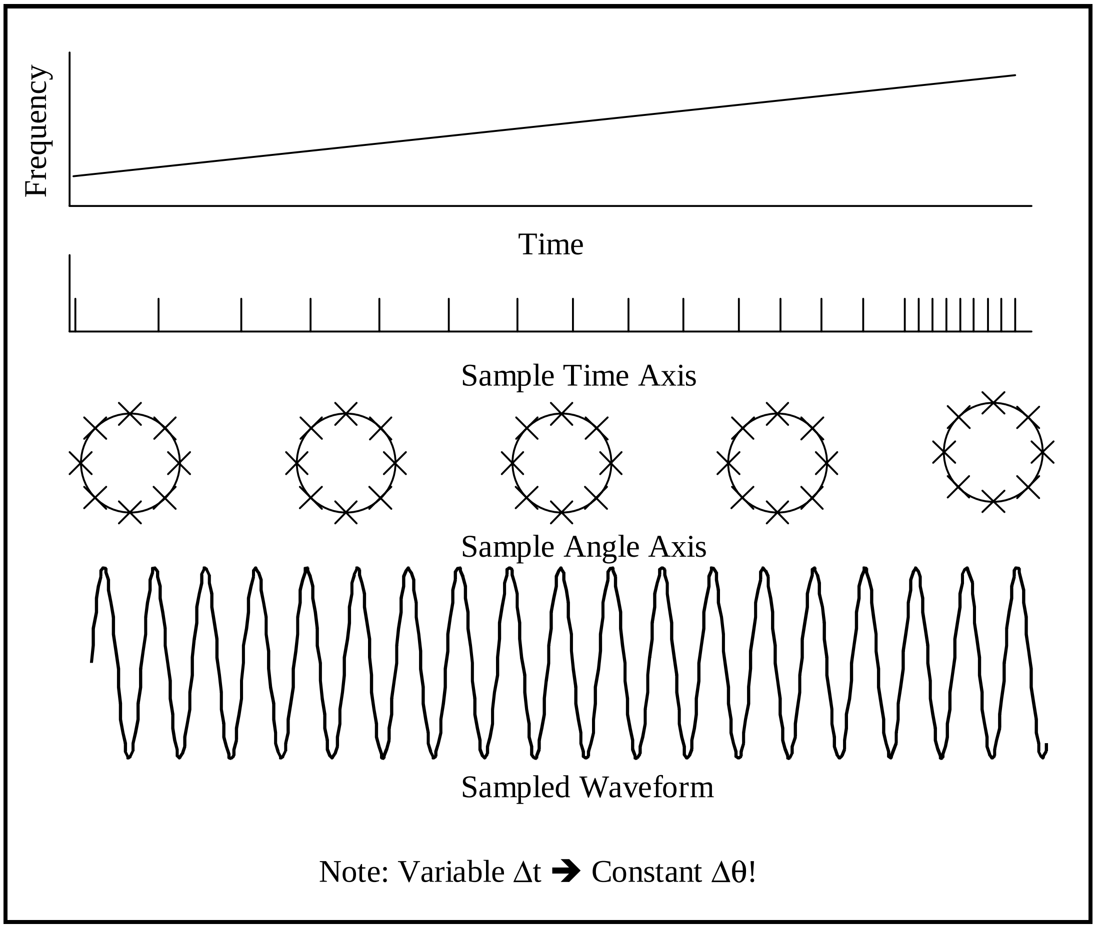

Figure A.3: Varying frequency sine wave sampled with a constant $\Delta\theta$.

In this case it is clearly seen by the sample rate plots that the signal is not sampled with a constant Δt. The important result of this figure is the bottom plot. The sampled waveform once again appears similar to a sine or cosine wave. Now that the sampled waveform and kernel appear similar, the results of the Fourier transform should be accurate.

Analyzing the constant $\Delta\theta$ sampled data with the sampling relationships and Fourier transforms presented in Equations A.1 and A.2 respectively is not possible. These equations are defined in terms of a constant Δt. This data has a constant $\Delta\theta$! The sampling relationships are then re-written in terms of a constant $\Delta\theta$. These new sampling relationships are shown in Equation A.3.

$$
\begin{aligned}
&\Delta o = \frac{1}{R} = \frac{1}{N * \Delta\theta} \\
&R = N * \Delta\theta \\
&O_{nyquist} = O_{\max} = \frac{O_{sample}}{2} \\
&O_{sample} = \frac{1}{\Delta\theta}
\end{aligned}
\tag{A.3}
$$

where: Δo is the order spacing of the resulting order spectrum.
R is the total number of revolutions that are analyzed.
N is the total number of time points over which the transform is performed.
$\Delta\theta$ is the angular spacing of the resampled samples.
O_{sample} is the angular sample rate at which the data is sampled.
$O_{nyquist}$ is the Nyquist order.
O_{max} is the maximum order that can be analyzed.

Note that the equations in A.3 are exactly the same as those of Equation A.1 if a substitution of $\Delta\theta$ for Δt, Δo for Δf, and R for T is made. The substitution of R for T is made because to achieve the spectral line spacing, Δf, in the frequency domain, a specified period of time was used in the Fourier transform integration. In this case a specified number of revolutions is used in the integration. Over a specified period of time, T, an integer number of cycles of the all of the Δf frequency sine waves was observed. It can be seen that if the same number of integer cycles, or revolutions, is observed in the angle domain, the Δo resolution will be the same as the Δf resolution.

It can also be seen in Equation A.3 that the relationship between the Nyquist order and the angular sample rate is the same as that seen between the Nyquist frequency and the time based sample rate shown in Equation A.1. This implies that there must be at least 2 samples/cycle for the maximum order that is to be analyzed to avoid aliasing. The other implication from this sample rate relationship is that the cutoff frequency of the required analog anti-alias filter must always be, at a maximum, equal to the frequency of the Nyquist order. In other words, the cutoff frequency of the analog anti-alias filter must vary with the sample rate to avoid aliasing!

Now that the sampling relationships have been reformulated and evaluated in the angle domain, the Fourier transform kernels of Equation A.2 must be reformulated in the angle domain. These reformulated kernels are shown in Equation A.4.

$$a_m = \frac{1}{N}\sum_{n=1}^{N} x(n\Delta\theta)\cos(2\pi o_m n\Delta\theta)$$
$$b_m = \frac{1}{N}\sum_{n=1}^{N} x(n\Delta\theta)\sin(2\pi o_m n\Delta\theta) \tag{A.4}$$

where: o_m is the order which is being analyzed, mΔo.
a_m is the Fourier coefficient of the cosine term for o_m.
b_m is the Fourier coefficient of the sine term for o_m.

These equations were derived from Equation A.2 by making the same substitutions as those made to derive Equation A.3 from Equation A.1. The results of the angle domain Fourier transform will have a constant Δo bandwidth instead of the constant Δf bandwidth achieved with Equation A.2. Any order that has an integer number of cycles in the integration period R will fall on a spectral line Δo and will be estimated leakage free.

To obtain a leakage free estimate of an order, the number of revolutions of the transform must be chosen such that the order, or all orders, of interest fall on spectral lines, Δo's. For this reason, oftentimes a discrete Fourier transform (DFT) is done to estimate orders and not a fast Fourier transform (FFT). The DFT allows any number of samples to be used in the transform and does not contain the power of 2 restriction of the FFT.

Locking the sample rate to a time varying reference requires that information about that reference is known or measured. On rotating components, the necessary sampling rate information is related to the rotational speed of the component of interest. To obtain this rotational speed information, a tachometer signal is measured. Realize then that the accuracy of the sampling process is directly proportional to the accuracy of the measurement and processing of the tachometer signal.

A.2.1 Using a Frequency Ratio Synthesizer to Drive the External Sample Clock Input.

One way to use the tachometer signal is to input it into a frequency ratio synthesizer (FRS). The FRS estimates the time between tachometer pulses and generates a new pulse train with a different number of pulses per revolution than the tachometer sensor generates. This pulse train is then used as the input to the external sample clock input on a DSA. The DSA samples the acquisition channels each instant that a pulse is received from the FRS. There are errors associated with this hardware configuration because the output pulse train of the FRS is always one tachometer pulse in frequency behind that of the rotating shaft. This error is addressed in more detail in Chapter 2.

Because the analog anti-alias filter on most DSA's is a fixed frequency filter, an external tracking anti-alias filter must be used to avoid aliasing in this configuration. The external filter is necessary because the frequency of the pulse train from the FRS, the rotational speed of the shaft, varies as a function of time and therefore the sample rate of the DSA varies as a function of time. Since the cutoff frequency of the analog anti-alias filter must always be less than the Nyquist frequency of the data, the cutoff frequency of the analog anti-alias filter must change with time, rpm in this case. To eliminate aliasing in the sampled data then the same signal from the FRS that is used to drive the DSA sample clock is used to drive the cutoff frequency of a tracking lowpass analog anti-alias filter. The frequency of the tracking filter is always less than the Nyquist frequency of the sampled data and therefore aliasing is prevented.

Producing tracking analog anti-alias filters is both expensive and difficult. For this reason they are not used very often, especially in large channel count systems. The cost for large channel count systems rises exponentially because all of the filters, one for each channel, must be amplitude and phase matched. Another limitation of the tracking analog filter is that usually it will introduce filter transients into the data if its frequency is changed too rapidly. This last limitation can severely limit the sweep rate of data.

A.2.2 Adaptive Resampling From the Time Domain to the Angle Domain.

The second way the tachometer signal can be used as a reference signal to drive the sample clock is through digital adaptive resampling. In this process, the data is originally acquired with constant sample time intervals, Δt's. The data is acquired using the sampling relationships presented in Equation A.1. The Nyquist frequency must be chosen such that the highest order of interest will have a lower frequency than the Nyquist frequency at the highest rpm of the acquisition period. The Δt sampled data is alias protected by a fixed frequency analog anti-alias filter. This process does not require the expensive analog tracking anti-alias filter.

Once the data has been acquired with constant Δt's, it is adaptively resampled to the angle domain. To accomplish the adaptive resampling, the tachometer signal is curve fit using one of the algorithms described in Chapter 2. The curve fit tachometer signal is then integrated to obtain an estimate of angle vs. time of the rotation of the machine. The integrated curve fit tachometer signal is then used to determine at what instants in time new data samples are desired.

The angular sample rate that the original data must be resampled to is determined either by the highest order present at the lowest rpm value or the highest order the user is interested in. The larger of these two order values must be used in the resampling process. A sample rate of at least 2 samples/cycle of the largest order must be used. The larger of these two values must be used to prevent aliasing in the adaptive resampling process. This strategy for determining the new angle domain sample rate results in the new adaptively resampled data always having a higher sample rate than the original data.

Aliasing is always prevented in any resampling process by either resampling to a higher sample rate than the data was acquired with or by lowpass filtering the data to an appropriate frequency before reducing the sample rate. These concepts are discussed in greater detail in Appendix B. In the adaptive resampling from the time domain to the angle domain the data is always resampled to a higher sample rate to prevent aliasing.

Having determined the new angle domain sample rate, the integrated tachometer signal is interpolated to determine the instants in time that new samples are required. The time domain data is then adaptively resampled at the new time instants to obtain the angle domain data. The adaptive resampling is performed using either the adaptive resampling method presented in Chapter 2 or the method presented in Chapter 3.

If the angle domain sample rate was determined by the highest order of interest, then the resampling process is complete and the angle domain data may be analyzed using Equation A.4, or digitally filtered, or otherwise processed using an angle domain DSP algorithm.

If the angle domain sample rate was determined by the highest order present in the data at the lowest rpm value, the resampling process may not be finished. If the user desires a lower angle domain sample rate than the adaptively resampled data contains, a decimation procedure must be performed. Knowing in advance how the angle domain sample rate was determined can make the decimation process much more computationally efficient. If the user desires the highest order in the resampled data to be order 20, for example, then the data must be resampled to contain at least 40 samples/cycle. If the data on the other hand contains order 65 at the lowest rpm, the necessary angle domain sample rate is 130 samples/cycle. It is not computationally efficient to decimate from 130 samples/cycle to 40 samples/cycle because the division of 130 by 40 does not result in an integer decimation factor. If however the original data had been resampled to an angular sample rate of 160 samples/cycle, the adaptively resampled angle domain data could be decimated using a standard decimation procedure down to 40 samples/cycle by using a decimation factor of 160/40 or 4. The adaptive resampling process is now complete and the angle domain data may be processed however the user desires with an angle domain DSP algorithm.

A.3 Time to Angle Domain Summary.

This appendix has provided a complete explanation of the ramifications and sample rate implications that must be considered in the transformation of data from the time domain to the angle domain. Realizing that the same DSP algorithms can be applied to both time and angle domain data however is just a start in completely understanding the power of DSP algorithms. Many of the same DSP algorithms used in the time and angle domains are used in domains such as the spatial domain. Oftentimes, spatial domain processing is 2 dimensional. Many of the 1 dimensional algorithms that are commonly used in NVH work are easily expanded to be 2 dimensional. Spatial domain processing is done in Nearfield Acoustic Holography (NAH) and image processing for radar or medical applications, as well as many other fields.

An example of a 2 dimensional image processing DSP application is the JPEG compression algorithm. This algorithm is based on doing 8X8 cosine transforms and storing only the largest coefficients to obtain the compression. Wavelet compression is also becoming common in the image processing field, yet another application of an algorithm which is used on time domain data in the NVH field. Understanding the DSP algorithms that are used by engineers and scientists in other fields and in other domains may lead to new developments in the processing of NVH data. Remember all DSP algorithms are angle based and can be re-derived into any domain of interest.

Appendix B

Digital Resampling Techniques and Procedures.

B Introduction to Digital Resampling.

Digital resampling can be thought of as two separate procedures which when combined together allow the effective sample rate of a signal to be changed to any value the user desires. The new sample rate may contain samples that are evenly spaced in time or evenly spaced relative to another known variable. This appendix will systematically explain the algorithms employed in these two separate procedures. The two procedures are decimation and interpolation. Decimation and interpolation are well documented in the literature and are presented only for completeness in this dissertation.

B.1 Decimation Procedures.

In this section the basic decimation procedure is presented, followed by some observations to greatly reduce the computational demands of the decimation process.

B.1.1 Decimation Theory.

Decimation is the process of reducing the sample rate of digitally sampled data. The first step in the decimation process is the digital filtering of sampled data that was passed through an analog anti-alias filter prior to the sampling process. The data is digitally filtered with a lowpass filter that possesses a maximal cutoff frequency of the Nyquist frequency of the new sample rate. Decimation performed in this manner reduces the effective sample rate of the data by an integer amount, M. This implies that the digital filter has a maximal cutoff frequency of $F_{Nyquist}/M$. Once the data has been digitally filtered to reduce its effective sample rate, every Mth data point is recorded while the remaining data points are discarded.

The basic decimation procedure is presented in Figure B.1. In this figure, a hypothetical signal is sampled at 100 Hz and decimated to an effective sample rate of 25 Hz, thus using a decimation factor of 4.

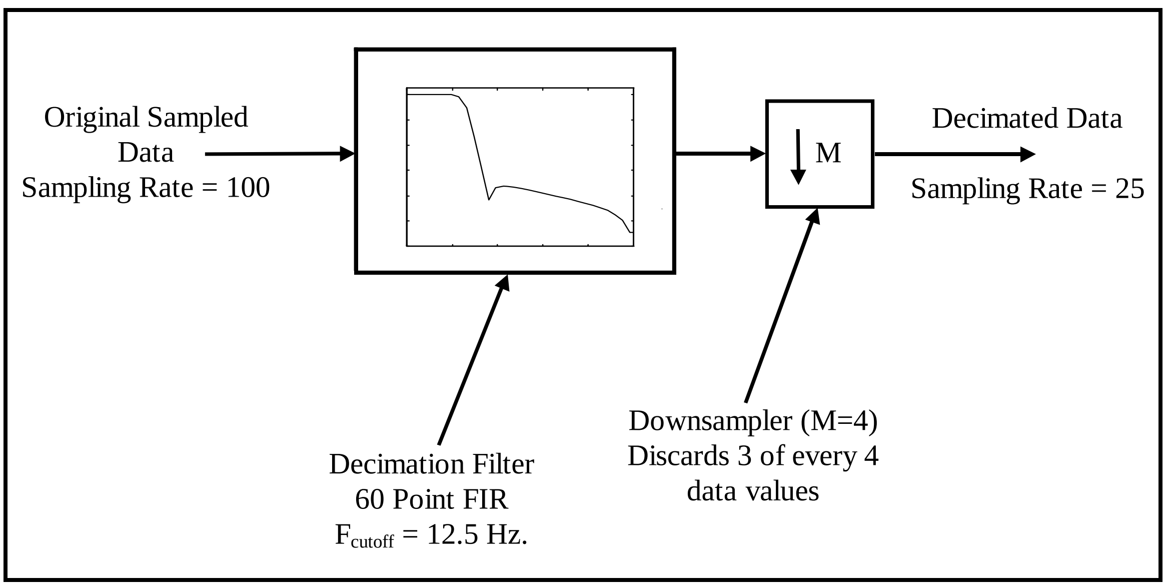

Figure B.1 Graphical representation of decimation process.

B.1.2 Decimation Application.

Decimation is a very integral part of the digital data acquisition systems commonly employed to acquire data. Decimation is performed in real time in many of these acquisition systems to reduce the sample rate of the data down to the desired frequency bandwidth. Many analyzers originally sample the data with a fixed maximal rate. Decimation is also employed as a post-processing procedure in many analysis packages commonly used in the laboratory.

The digital filter applied in the decimation process may be either a FIR or an IIR filter. An IIR filter will usually be more efficient if the data is decimated by a relatively small amount. The use of an IIR filter also allows a filter to be used which emulates an analog anti-alias filter.

An FIR filter can be applied more efficiently if the data is to be decimated by a large amount. While the FIR filter usually will contain more filter coefficients than an IIR filter that gives a similar out of band suppression, the FIR filter does not depend on previous values of the output of the filter. This latter property of the FIR filter is where the computational advantage can be realized for decimation by large factors. The FIR filter can be applied only to the data values that are to be kept after the filtering process, thus discarding the undesired data values in the filtering process, since it does not depend on past outputs of the filter. This procedure will reduce the number of computations required to filter the data by the decimation factor, M. Obviously, for large decimation factors this procedure will then save considerable computational effort even if the filter order of the FIR filter is considerably higher than that of an equivalent IIR filter.

Regardless of which filter is used in the decimation procedure, the user must design the filter such that all frequency content above the new Nyquist frequency is attenuated to prevent aliasing. Since a digital filter with a zero width transition band cannot be realized, this oftentimes leads to a cutoff frequency of the lowpass filter which is lower than the new Nyquist frequency. For example, Hewlett Packard uses a decimation filter which possesses a cutoff frequency of approximately 80% of the desired Nyquist frequency. This is the reason that if a 1024 Hz sample rate is chosen, frequency domain results are only presented out to 400 Hz. The frequency content between 400 and 512 Hz is not presented because it may contain aliased information.

In either filter case it may also be more computationally efficient to decimate the data in multiple stages. A digital filter which has a desired set of passband and stopband properties and a cutoff frequency of ¼ to ½ of the original Nyquist frequency oftentimes can be designed with much fewer coefficients than the same filter with a cutoff frequency of 1/32 of the original Nyquist frequency. This implies that it may be more efficient to decimate the original data by a factor of 4 in multiple stages than to decimate the data by a factor of 16 in one stage.

B.2 Digital Interpolation Procedures.

This section presents the theory behind digital interpolation, followed by a numerical example of the interpolation process, and finally a discussion of interpolation applications.

B.2.1 Interpolation Theory.

Interpolation is the opposite of decimation and is used to effectively upsample digitally sampled data. Interpolation is the process of inserting L-1 new data values between the original data values. Usually the new data values will be evenly spaced between the original data values. The upsampled signal will then possess a Δt that is 1/L of the original Δt. However, the new signal will not contain any frequency content above the original signal's frequency content.

One method of performing the upsampling process is simply to insert L zeros evenly spaced between the original data values. This zero inserted time history is then filtered with a lowpass filter to suppress the spectral images introduced into the signal through the zero insertion process. Oftentimes a specific interpolation filter may be designed which requires fewer filter coefficients than a generic lowpass filter. This special interpolation filter contains *don't care bands* which are systematically positioned in spectral regions where there will not be high frequency content introduced by the zero insertion process. These regions occur around the original Nyquist frequency and its multiples.

Figure B.2 graphically presents the steps employed in the upsampling process. In this figure the original data is assumed to have a sample rate of 25 Hz while the upsampled data has a sample rate of 100 Hz. This process therefore uses an interpolation factor of L=4.

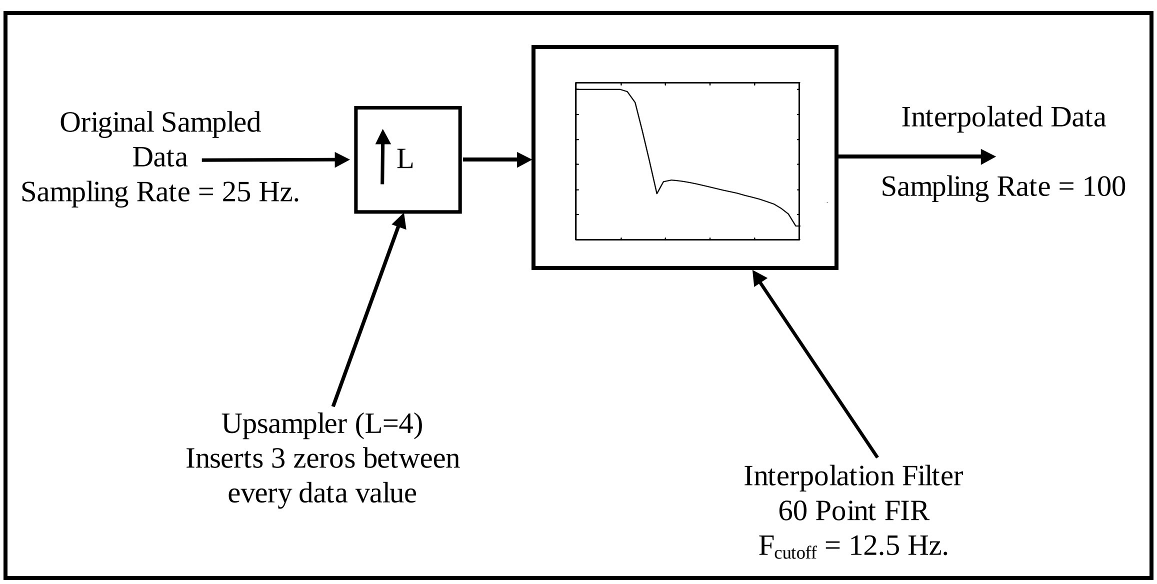

Figure B.2 Graphical presentation of interpolation process.

B.2.2 Numerical Interpolation Example.

An example of the processing and intermediate steps of spectral interpolation is presented here to illustrate the effects of the zero insertion process and its subsequent filtering. The original sampled data is presented in Figure B.3. The original data is a sine wave with a frequency of 1.5 Hz that has been sampled at 10 Hz.

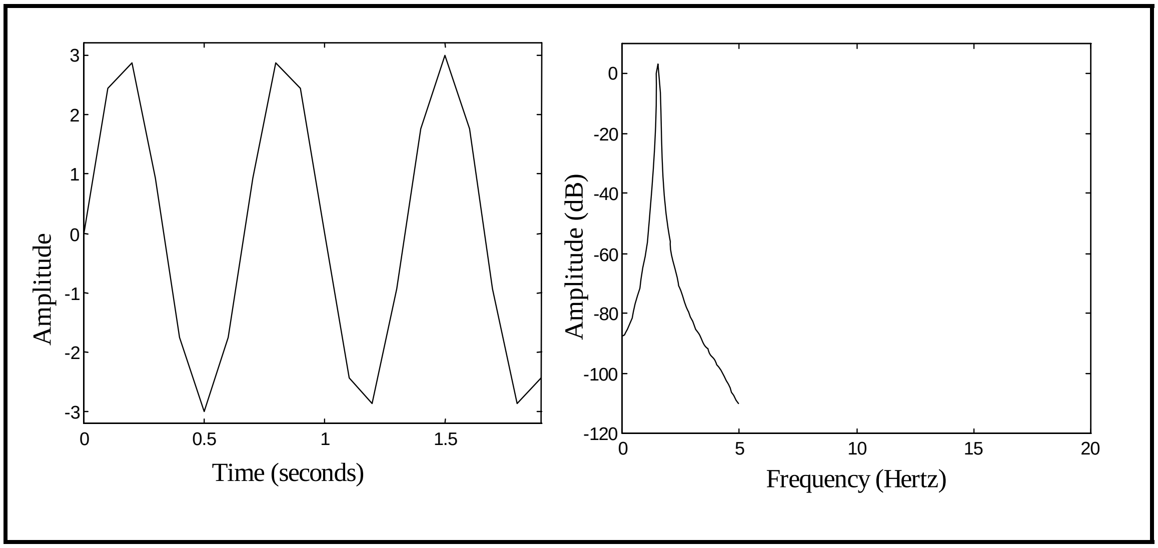

Figure B.3 Original time and frequency representations of signal.

The first step in the interpolation process is to insert L-1 zeros between the original data values. In this case, L=4 and 3 zeros are inserted between each original data value. The zero inserted data is presented below in Figure B.4. Note the spectral images that are produced in the frequency domain plot of the zero inserted time history. The zero inserted time history now has a sample rate of 40 Hz.

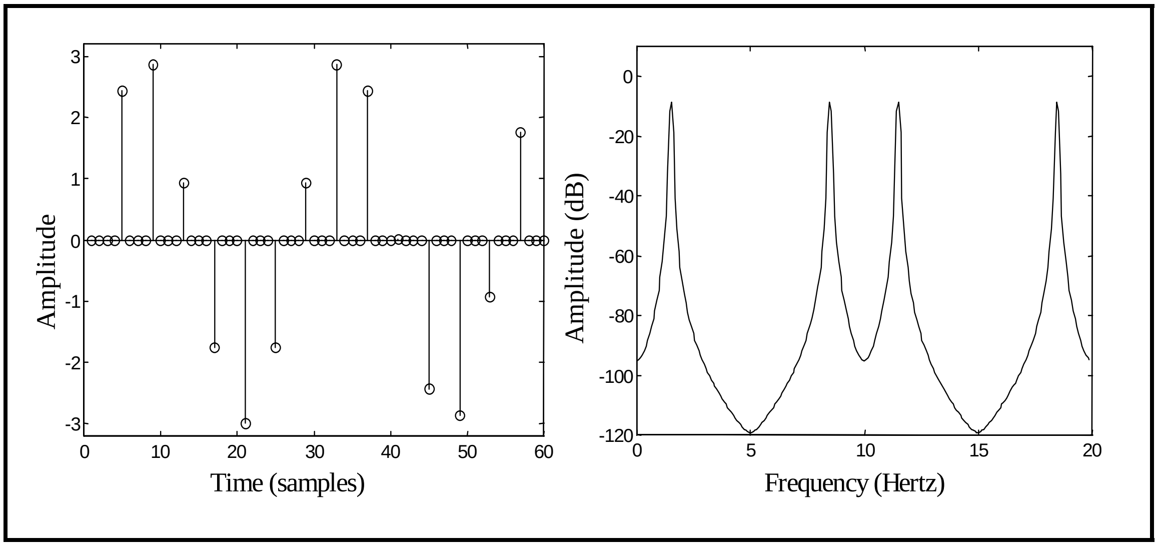

Figure B.4 Time and frequency characteristics of zero inserted signal.

The FIR interpolation filter that is used to remove the spectral images is presented in Figure B.5. Note the frequency domain filter shape. The filter contains a *don't care band* in the 15 Hz region which is where the spectral images have a minimum amplitude and do not require a significant amount of suppression. It can be seen in the frequency domain characteristics of the filter that the spectral images will be suppressed by approximately 80 dB with this filter.

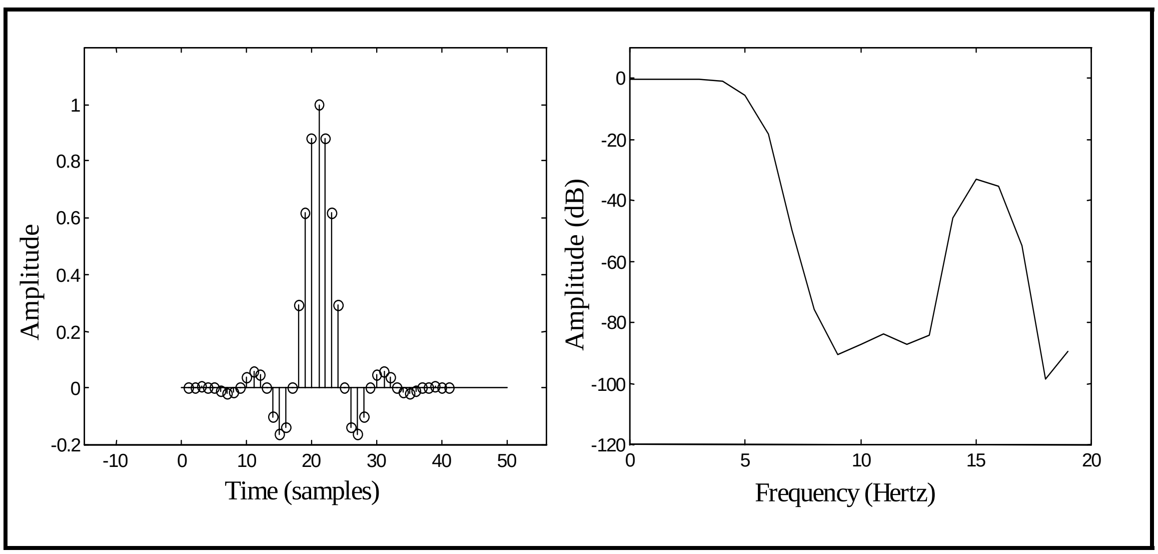

Figure B.5 Time and frequency characteristics of interpolation filter.

Finally, the filtered zero inserted time and frequency domain plots are presented in Figure B.6. The time domain plot shows the improvement in the amplitude resolution of the signal. The frequency domain plot shows that the spectral content of the interpolated signal has not been altered by the upsampling process if 80 dB of dynamic range is considered sufficient.

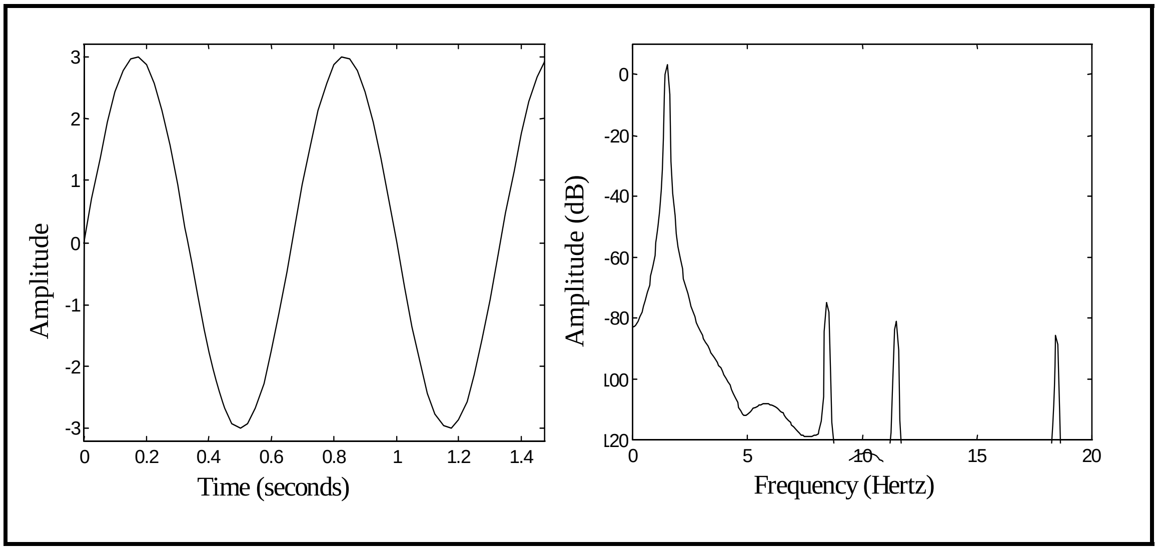

Figure B.6 Time and frequency domain characteristics of upsampled signal.

B.2.3 Interpolation Application.

The interpolation process is not normally performed in data acquisition systems as most acquisition systems initially sample the data at a high rate. Interpolation is however widely used in the post processing of acoustic data if the data is to be played back for sound quality evaluation. Interpolation is also a necessary step in changing the effective sample rate of a signal by a non-integer value. An example of this would be changing the sample rate of data acquired on a DAT tape with a sample rate of 48 kHz to the sample rate of an audio CD sampled at 44.1 kHz. This type of sample rate conversion is referred to as rational fraction interpolation and uses a combination of decimation and interpolation stages.

Upsampling is also necessary if the time domain amplitude of a signal is desired. Oftentimes data may be acquired with a minimal sample rate as dictated by Shannon's sampling theorem. This sample rate will give accurate frequency domain amplitude estimates but will not give accurate time domain amplitude estimates. Upsampling may also be applied to a time captured tachometer signal to improve the estimates of the zero crossings and thereby improve the rotational period estimates. Obviously, the same benefits exist in upsampling time captures of trigger or event channels. It should be remembered that while upsampling may improve the analysis results of these signals, upsampling does not add any new information to the signal as the frequency content of the upsampled signal is the same as that of the original signal.

Applying a standard lowpass filter to perform digital interpolation, as presented in the above numerical example, requires the filter have enough coefficients so that 5 to 8 of the original data values on each side of the desired new data value are included in the filtering process. If fewer than 5 to 8 original data values are used on each side of the new data value the filter will not accurately predict the new data value. This implies that the number of filter coefficients must be (5-8)*L*2+1. The filter is then applied by performing the convolution of the filter with the zero padded signal. This procedure multiplies many of the filter coefficients by zero and then adds these zeros together along with the non-zero results to obtain the new data values. The use of a polyphase filter allows a much more efficient implementation of the digital interpolation process. A polyphase filter is an FIR filter which is applied by only performing the multiplications/adds of the filter coefficients which are aligned with an original data value. This eliminates the operations of multiplying/adding all of the known zero results. This requires more computational overhead in the filtering algorithm but reduces the number of computations by a factor of L, which can be significant for a large upsampling factor, L.

B.3 Rational Fraction Sampling Rate Interpolation.

The procedure of changing sampling rates by a non-integer amount, either up or down, often requires the use of a combination of decimators and interpolators. If the desired sampling rate can be obtained by using integer decimation factors, M, and integer interpolation factors, L, the procedure is called rational fraction sampling rate interpolation. This section will present the theory and application of this procedure.

B.3.1 Rational Fraction Sampling Rate Interpolation Theory.

The sampling rate of a signal may be increased or decreased by an integer amount through the use of only an interpolator or decimator respectively. When the desired sample rate does not have an integer relationship to the original sample rate, a combination of interpolators and decimators may be applied to realize the new sample rate. The first step in this process is to determine what interpolation and decimation factors, L and M, are required to arrive at the new sample rate. This involves determining the smallest common denominator between the original and desired sample rates. This then allows the determination of the rational fraction L/M which provides the necessary interpolation and decimation factors.

The first step once the interpolation factor is determined is to upsample the original data by a factor of L. The upsampled data is then decimated using a decimation factor M. This procedure will preserve the frequency content of the original signal either to the original Nyquist frequency or to the Nyquist frequency of the new re-sampled signal. Aliasing will not be introduced into the re-sampled signal as long as the interpolation is carried out before the decimation. A graphical representation of this process is presented in Figure B.7.

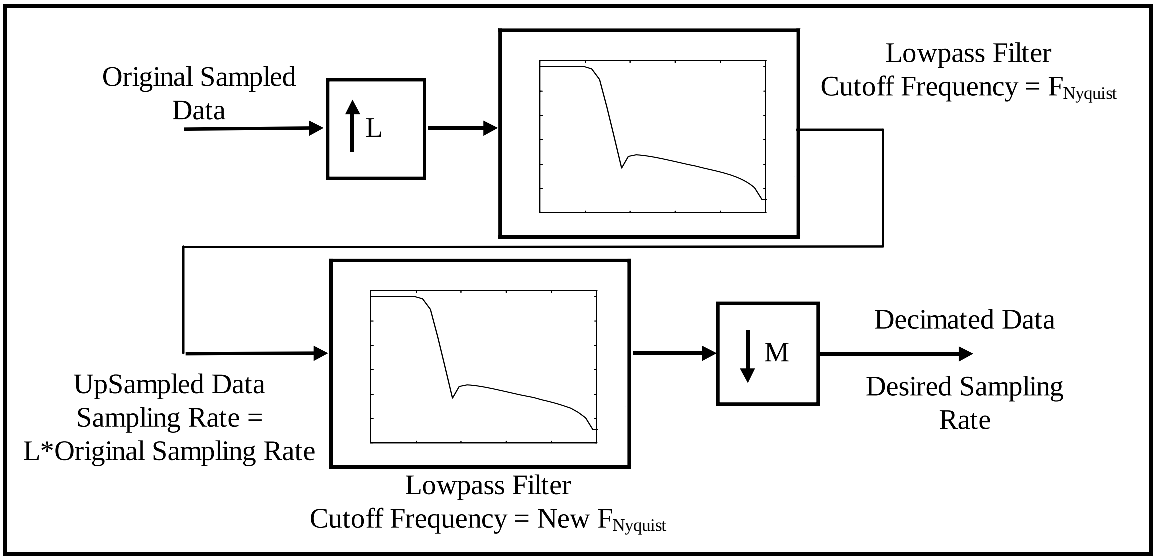

Figure B.7: Graphical representation of rational fraction sampling rate interpolation.

B.3.2 Rational Fraction Sampling Rate Interpolation Application.

Figure B.7 shows a straightforward way of performing the rational fraction sampling rate interpolation, but is not a numerically efficient implementation. Large gains in computational efficiency can be achieved through applying only one digital lowpass filter. The single lowpass filter replaces the two filters that are presented in series. The cutoff frequency of the single lowpass filter must be the lower of the cutoff frequency of the two filters presented. If the single filter is a polyphase FIR filter, further computational savings can be obtained by only applying the filter to the data points that will be kept after the decimation process. While applying the filter in this manner does require more computational complexity in the filtering algorithm, it can result in large speed improvements.

An alternative method of performing this sampling rate interpolation is to use multiple stages in both the interpolation and decimation processes. While this approach is not as computationally efficient as the method described above, it may still be much more computationally efficient than the straightforward approach presented in Figure B.7. The savings are gained using multiple stages because of the lower order filters required, as discussed in the decimation Section B.1.

This multistage approach is particularly beneficial in instances where the interpolation and decimation factors needed to obtain the desired sample rate are large. For example, to transform audio CD data which is sampled at 44.1 kHz to the 48 kHz sample rate of a DAT tape requires L=160 and M=147. This implies that to use the straightforward approach, the sample rate is first increased to (160*44.1 kHz) 7.056 MHz, which will result in the original time history becoming 160 times longer. This zero inserted data is then filtered with an interpolation filter with, for example, (5*160*2+1) 1601 coefficients! The 1601 point filter is then applied to every data value. Once this data has been filtered, a decimation filter is applied to the long time history and 146 out of every 147 samples is discarded.

If the interpolation is performed in multiple stages the computational load is significantly reduced. For example, factoring L=160 into stages of 2,2,2,2,2, and 5 will result in an interpolation filter with only (5*2*2+1) 21 coefficients for each of the first 5 stages and (5*5*2+1) 51 coefficients for the last stage. The result of this approach is having to perform approximately 27 times fewer multiply and adds in the interpolation procedure.

B.3.3 Arbitrary Sampling Rate Interpolation.

The arbitrary sampling rate interpolation algorithms are based on the same methodology as adaptive sampling rate interpolation algorithms, explained in Chapters 2 and 3, without the necessity of a reference signal to drive the adaptive portion of the resampling.

The adaptive resampling algorithms can be effectively used to resample a dataset to any arbitrary time axis. This implies that they may be used to resample a dataset to a new constant Δt, different than the original Δt that the data was acquired with. This is done much more efficiently than the adaptive resampling since much of the computational effort involved in the adaptive resampling procedures can be the determination of the new instants in time which new data samples are desired. Using adaptive resampling procedures to resample data to a new constant Δt can be more computationally efficient than using the rational fraction resampling approach if a large amount of interpolation and decimation must be used to obtain the new desired sample rate.