

DATA CONDITIONING METHODS FOR AERO ENGINE SIGNALS AND THEIR INFLUENCE ON THE ACCURACY OF FATIGUE LIFE USAGE MONITORING RESULTS

Hugo Pfoertner*
Simone Graf**

***MTU Motoren- und Turbinen-Union Muenchen GmbH,
P.O.Box 500640, 80976 Munich, Germany**

****Universitaet Passau, Fakultaeat Mathematik und Informatik,
Innstrasse 33, 94032 Passau, Germany**

Abstract

During the design of monitoring systems for aircraft engines it is common to include algorithmic provisions to reduce the anticipated noise content of the input signals by applying some sort of filters at a suitable stage of processing. A tacit assumption is often made that such filtering will make the system more robust with respect to noise or even data errors.

The elimination of uncorrelated noise in a signal emanating from a physical process may have a large effect on the predictability of this signal, thus enabling high data compression rates in a dedicated or embedded flight data recording system. However, little is known about the influence of filtering the input into engine fatigue life usage calculations on the outcome of various models used in present monitoring systems.

A simplified, yet realistic mathematical model is used to describe the thermal response, stress and fatigue behavior of fracture critical parts in an aero engine compressor. Using this model, the consequences of applying digital recursive filters to recorded engine data are investigated. The analysis concentrates on statistical methods to assess the accuracy. From the results some guidelines are derived that allow a more systematic selection of filter parameters when a predefined accuracy of the fatigue life usage results is required.

Introduction

Existing engine monitoring systems for military aircraft include methods to calculate the fatigue life usage of rotating engine components known to have the potential to destroy the aircraft or to cause at least high financial loss in case of an in-flight failure. At first sight, on-board fatigue life usage monitoring (LUM) seems to have reached maturity over the years and will be applied to the engines of the Tiger helicopter and of the Typhoon Eurofighter.

These systems use on-board processing based on a few (less than 10) engine or aircraft signals to calculate the life consumption for all fracture critical parts of the engines. Figure 1 shows a typical spool speed signal of the HP spool speed of a military engine for one flight. The calculation uses mathematical models of the thermal, mechanical and material properties of the engine and its components. A simplified example of such a model will be presented later in this paper.

The algorithms are based on existing knowledge of failure mechanisms and take into account the experience (e.g. test results, inspection findings) available at the time when the on-board software is specified. Although some flexibility can be built into the monitoring software (e.g. by using loadable parameter sets), the software is not able to cope with newly detected damage mechanisms or with unanticipated configuration changes of the engine. It is therefore necessary to perform regular updates of the on-board software. Even if the system architecture is carefully designed, a fleetwide software update together with the accompanying adaptations in the logistic system is at least very expensive and carries the risk of introducing inconsistencies into the fatigue life usage data [BP97, PR95].

The most challenging situation for existing LUM systems occurs, if damage (e.g. cracks) is detected during inspection of components with long in-service times without any proper coverage of the corresponding areas in the on-board LUM algorithms. Current practice is to use statistical correlations between the computed damage at monitored areas and model calculations of loads at the newly detected area to get an idea, which parts would have to be removed from service or to be inspected. The inevitable consequence is a considerable uncertainty and loss of usable life for the

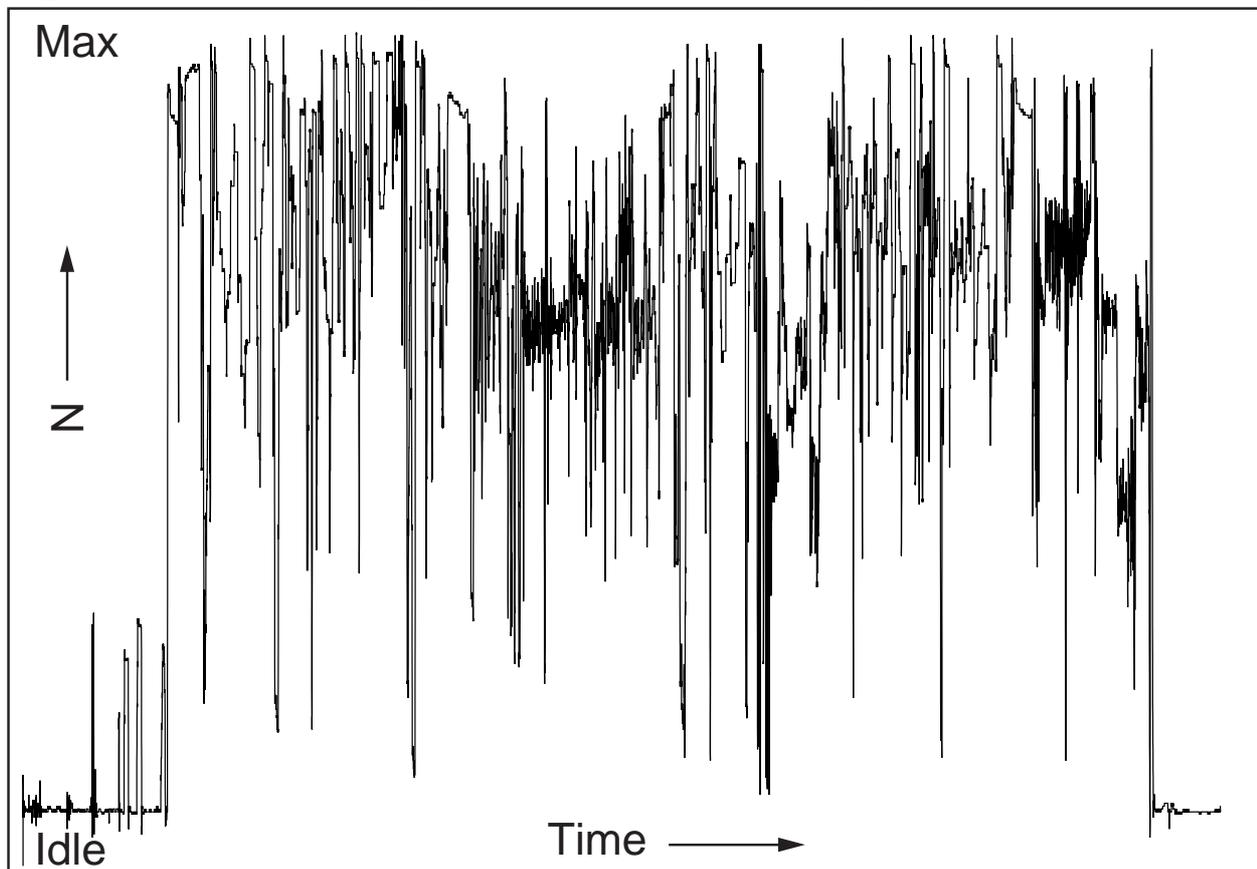


Figure 1: Spool speed signal of a military jet engine for one flight

components in question. The only way out of this dilemma would be the availability of a complete running history of the part, including recordings of all flight data necessary to recalculate the load history and life consumption at newly detected critical areas. None of the existing systems currently provides such long time history data.

Aviation recorders, either in the form of crash protected devices for accident investigations or as very large capacity airborne quick access data recorders for maintenance support are now profiting from the availability of storage media (e.g. solid state memory devices) with high reliability and enormous capacity. Many of the existing tape based devices are currently being replaced by solid state FDRs, both in the military [SS95] and in the civil transport [Gro99] environment. However, the direction of development seems to be aiming either at an ever increasing number of parameters or at higher sampling rates, which are needed for incident and accident investigations [HK99]. In contrast LUM applications only need a few parameters with relatively low sampling rates.

Long Term Recording of Engine Data

The monitoring function of the RB199 engines, which is part of the German Tornado OLMOS system [BP97], uses a data rate of 2Hz and only 8 input parameters plus some logical signals. All those data are of course available on the general purpose FDR [SS95], which only stores a few hours of data, however. No regular readout of those data is performed, unless other problems require an analysis of the FDR data, and it would be completely impractical to require a data readout of all FDR parameters just to get the engine data.

Although some previous work on flight data compression has been reported [SM89], no reliable figures were available on the information content of the signals entering a LUM calculation. The current practice in most existing recording systems is to use some sort of Run Length Encoding (RLE), which is known to be rather sensitive to small disturbances in the signals.

Some preliminary investigations were performed to get an idea on the benefits of various data compression methods, if applied to the signals needed for a LUM calculation. It turned out that different strategies have to be applied to the different signal types and that it is absolutely necessary to gain detailed insight in the signal properties to design a recording strategy yielding practicable data volumes when the full running history of an engine shall be stored. A strategy using delta coding together with statistical coders was found to be most promising. Those techniques depend heavily on the statistical properties of a signal. Examples for a spool speed signal are shown in Figs. 2 and 3.

The determination of the power density spectrum of the autocorrelation of a signal, which is the most important tool to detect noise, needs some precautions to produce meaningful results

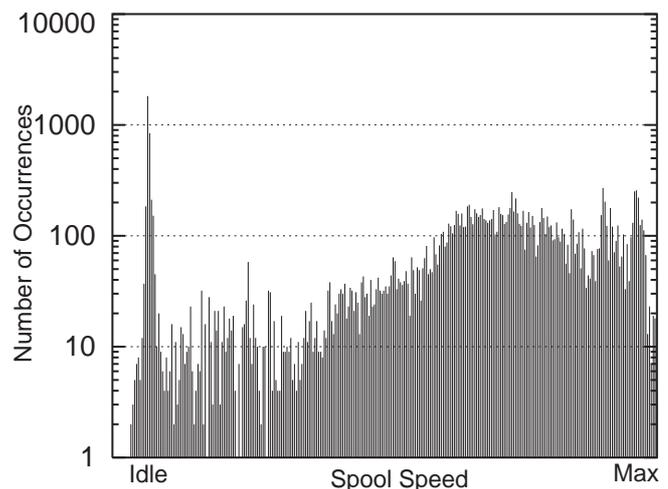


Figure 2: Histogram of spool speed values for one flight

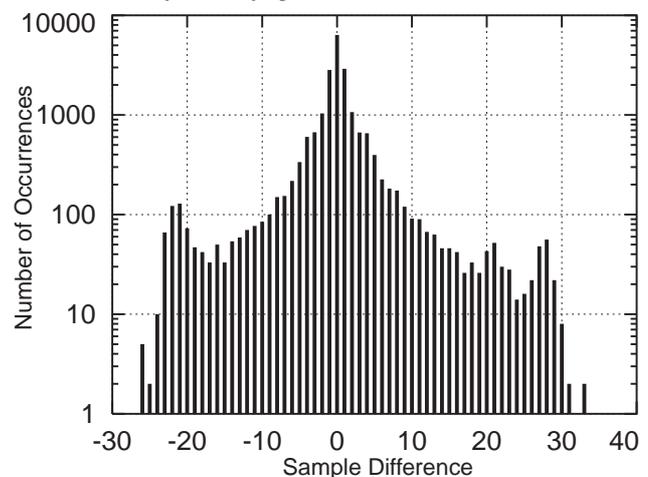


Figure 3: Histogram of spool speed delta values for one flight

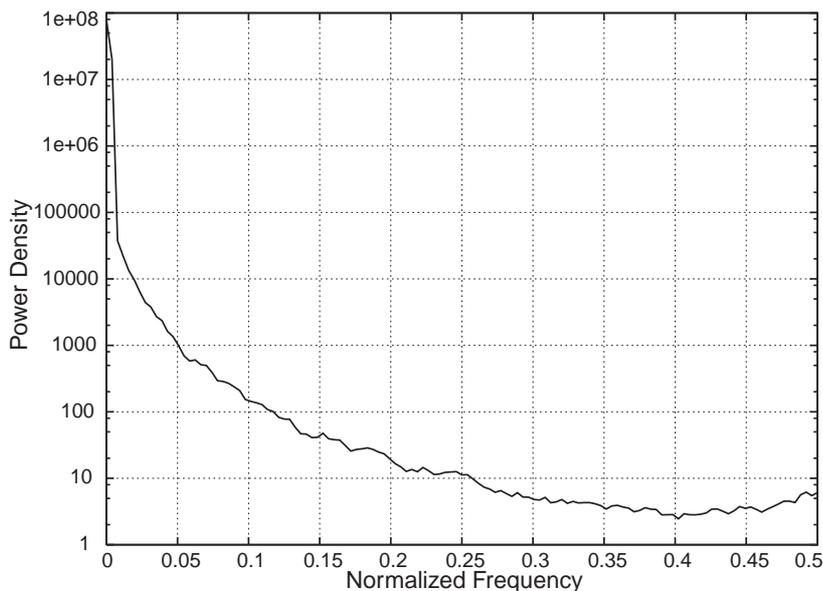


Figure 4: Power density spectrum of spool speed autocorrelation

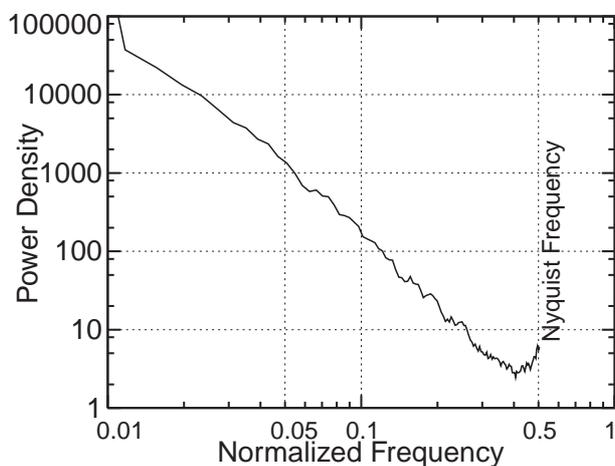


Figure 5: $\sim f^{-\beta}$ behavior of spool speed signal

[Smi97]. The method is usually applied to periodic signals, where distinct frequencies with some physical meaning (resonances) are sought for. If applied to other signal types, adapted analysis methods are required. A computer program “SPPOWER” in [SD93] was used to produce the spectrum data shown in this presentation. The program computes the average periodogram of a real data sequence by averaging the Fourier transforms of overlapping segments in the data. A typical segment length used, that has to be a power of

2, was 128. To mitigate boundary effects, a Hanning data window is applied to the single segments.

Figures 4 and 5 reveal a signal behavior that is found in many systems with interacting processes with different time scales. An enormous literature exists on this so called $1/f$ noise [Ber94]

An understanding of the signal properties is necessary to find preprocessing techniques with minimum influence on the information content. Digital filters are the most powerful tools to accomplish certain intended signal modifications [OS89, PB87].

Digital Lowpass Filters

Systems that smooth input signals, thus removing noise are usually called filters. They can be treated as a black box, that modifies a discrete input signal x into an output signal y . Certain systems (discrete, linear, time-invariant, continuous) are fully characterized by the so-called impulse response. That is the output of the system, if a delta impulse ($=1$ for time step $n=0$, 0 otherwise) is supplied as input. An arbitrary input signal $x(n)$ will then produce the output $y(n)$, where $y(n)$ is the convolution of the impulse response with $x(n)$.

For the present investigation only the special case of Butterworth filters was considered. These filters are easily implemented in the time domain and have a smooth amplitude response. To speak of amplitudes we have to look at the representation of the impulse response in the frequency domain. The convolution theorem states, that the Fourier transform of the output signal is equal to the product of the Fourier transforms of the input signal and that of the impulse response. If represented in the usual way as complex number, the convolution theorem yields the result, that the magnitude of the Fourier transform of the output signal is equal to the product of the Fourier transforms of the input signal and of the system's impulse response.

For Butterworth filters this magnitude of the Fourier transform of the impulse response is a very smooth function of frequency (Fig.6). This amplitude response approaches that of an ideal filter, if the filter order increases. An ideal low pass filter suppresses frequencies outside of its pass

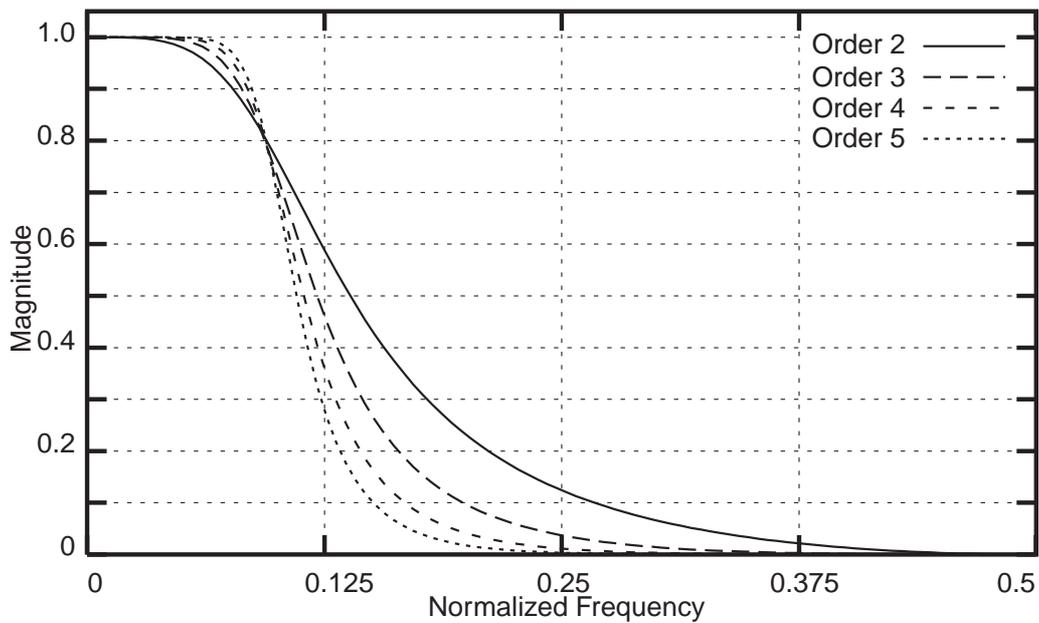


Figure 6: Amplitude response of 4 digital Butterworth filters with nearly identical cutoff frequency

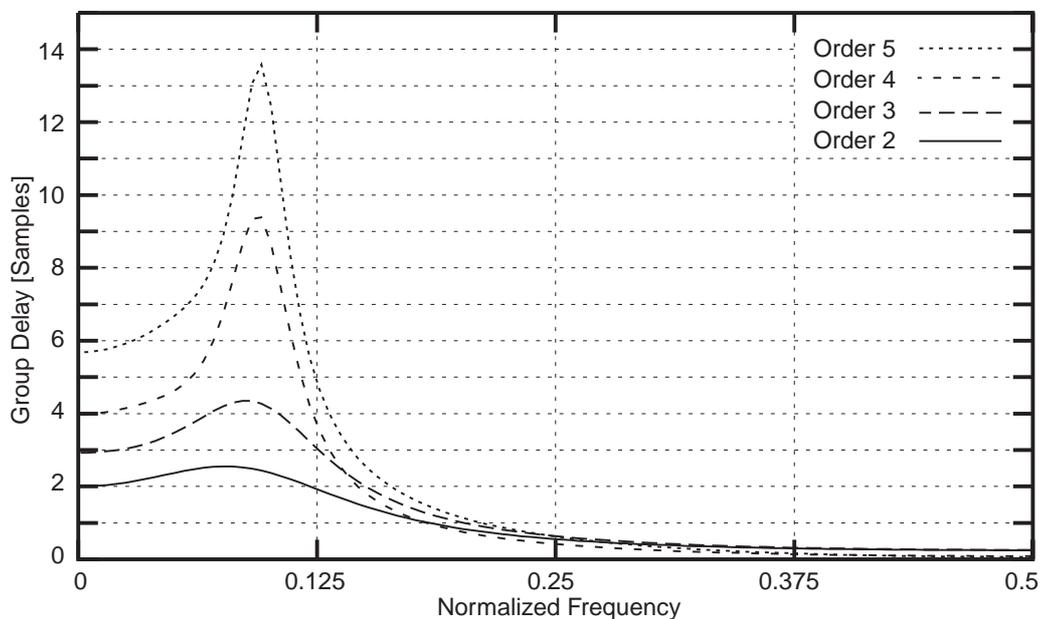


Figure 7: Group delay (negative phase derivative) for the 4 filters of Figure 6

band completely and has no influence on frequencies inside its pass band. Unfortunately such filters can not be obtained with practicable effort and one has to stay with approximations of the desired ideal behavior. Butterworth filters put emphasis on the quality of the amplitude response at the expense of the phase response. A linear phase response is tantamount to the filter exerting a simple time delay on the signal. Since both goals are interrelated, one has to accept a nonlinear phase response, that means the frequency contents of the signals are variably delayed (Fig. 7). A ripple in the amplitude response, which is quite common for linear-phase filters, would induce the risk of creating spurious new extreme values in the output. As a consequence new cycles without physical meaning could be created, thus distorting subsequent calculations of fatigue damage.

A short sketch of how to design and implement digital Butterworth filters follows. The available literature concentrates on analog Butterworth filters and on the Fourier transform of their impulse response. To design a digital Butterworth filter from the given Fourier transform of its analog impulse response, the following steps have to be performed:

- 1) Determine the analog impulse response
- 2) Compute the Laplace transform of the analog impulse response

- 3) Adjust the frequency representation with a bilinear transform
- 4) Compute the Z-transform of the digital impulse response
- 5) Compute the Fourier transform of the digital impulse response.

The result is the digital impulse response. (See example in Fig. 9). Applying the outlined procedure, a digital Butterworth filter is determined by its order N (the number of samples used to compute the actual output), the sampling interval of the discrete input signal T and the cutoff frequency ω_g . By selection of these three parameters the filter is fully determined.

1) Poles of analog filter: For $k \in \{0, \dots, N-1\}$ let $z_k := \omega_g e^{i\pi \frac{2k+1+N}{2N}}$.

2) Determine coefficients b_k for $k \in \{0, \dots, N\}$: $b_k := \binom{N}{k} \frac{\omega_g^N T^N}{\prod_{l=0}^{N-1} (2 - Tz_l)}$

3) Determine coefficients a_k for $k \in \{0, \dots, N\}$ by expanding the expression: Let

$$v \in \mathbb{C} \text{ arbitrary: } \sum_{k=0}^N a_k v^k = \prod_{l=0}^{N-1} \left(1 - \left(\frac{2 + Tz_l}{2 - Tz_l} \right) \cdot v \right).$$

4) Now let $x : \mathbb{N} \rightarrow \mathbb{C}$ be a discrete input signal, then the filtered output signal

$y : \mathbb{N} \rightarrow \mathbb{C}$ is obtained by: Let $n \in \mathbb{N}$:

$$y(n) := \sum_{k=0}^N b_k x(n-k) - \sum_{k=1}^N a_k y(n-k).$$

Input and output shall be 0 for negative n : $\forall n \in \mathbb{Z} : n < 0 \Rightarrow x(n) = y(n) = 0$. One remark on the cutoff frequency ω_g : The magnitude of the Fourier transform at the point: $2 \arctan\left(\frac{T\omega_g}{2}\right)$ is $\frac{1}{\sqrt{2}}$ for Butterworth filters of arbitrary order. At the corresponding frequency (transition or cutoff frequency) the amplitude has a turning point. Higher frequencies will be suppressed increasingly. The phase response is rather nonlinear in this transition area, with an extremum of the group delay (derivative of the phase response) near the cutoff frequency. (Figure 7).

For our investigation 25 Butterworth filters were used with a maximum order of 5. This choice was based on the idea to avoid longer lags between data acquisition and data storage, which becomes difficult to handle at engine shut down. Besides from that, the implementation of higher order filters poses increasing difficulties in accuracy and stability. The cutoff frequencies were chosen to match the values 1.0, 2.0, 3.0, 4.0 and 5.0 for the phase derivative (group delay) at zero frequency. This results in a linearly varying output signal delayed by whole multiples of the sampling interval for a linearly varying input signal after the overshoot oscillations (only present for filter orders greater than 1) have decayed. A derivation is given in [Gra00].

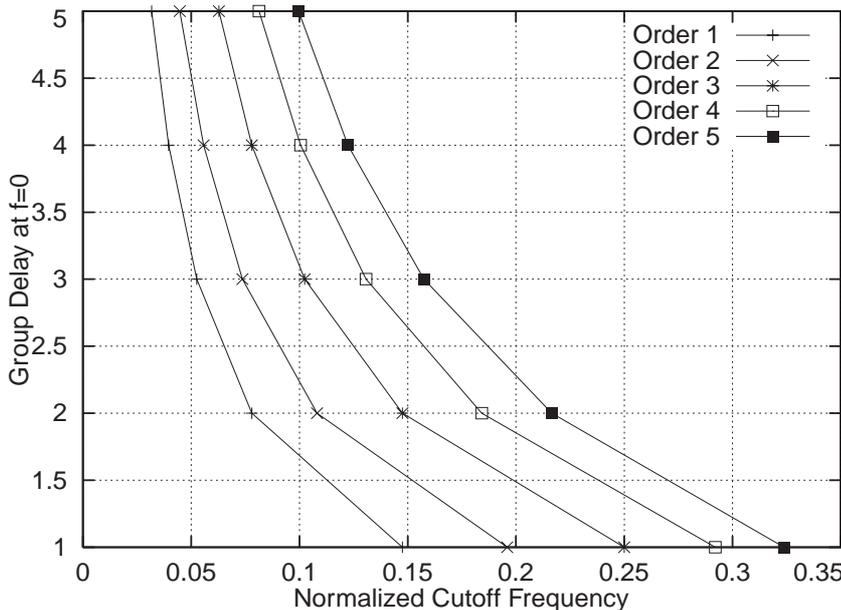


Figure 8: Correspondence between group delay at $f=0$ and cutoff frequency for Butterworth filter family

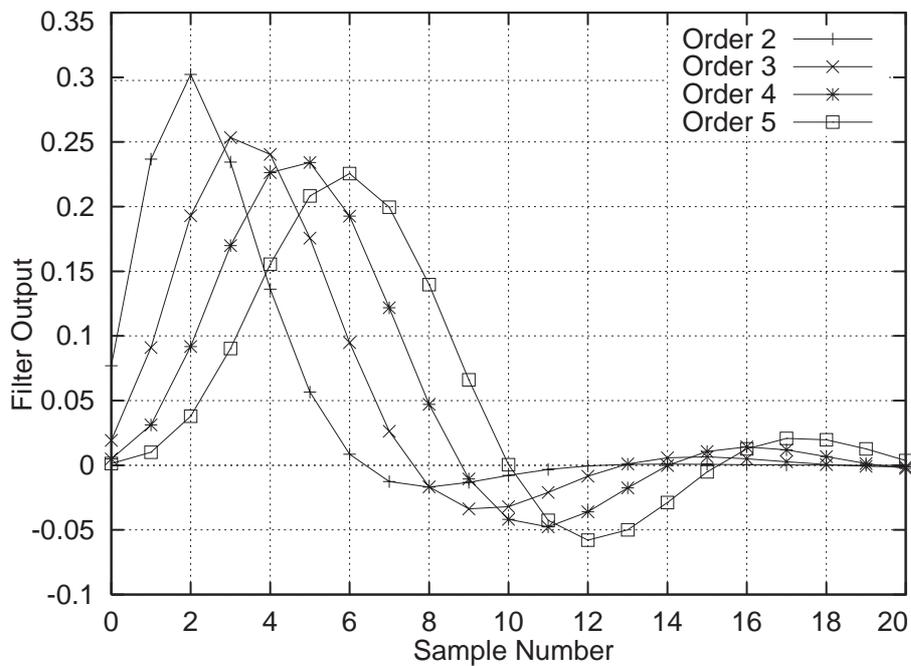


Figure 9: Impulse response of 4 Butterworth filters with equal cutoff frequency

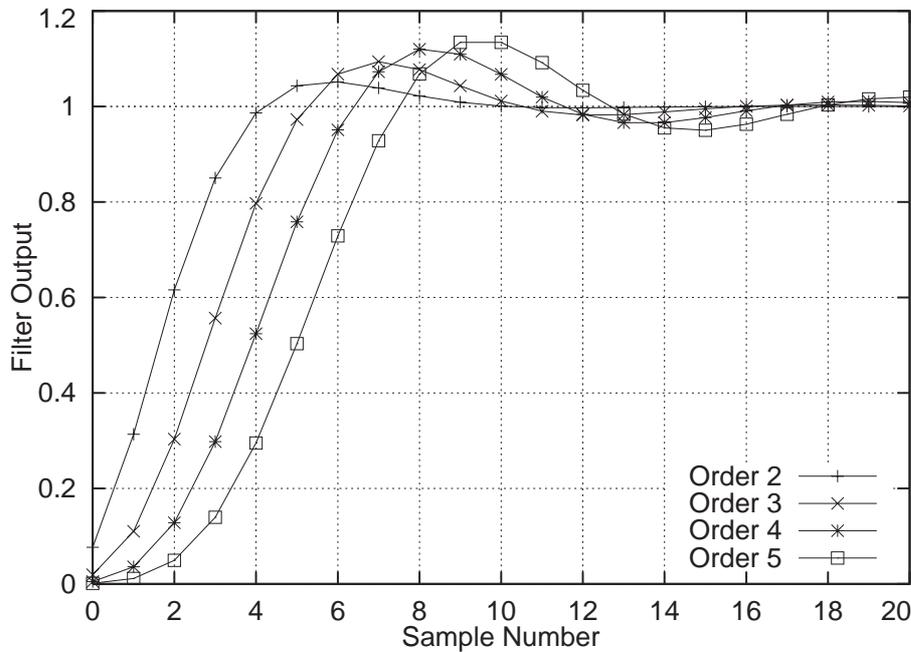


Figure 10: Step response of 4 Butterworth filters with equal cutoff frequency

filter order. Order-1 filters approach the input signal in an asymptotic manner without overshoot. The overshoot properties of filters are best visible in their step response. (Fig.10). The overshoot properties will turn out to be crucial for the accuracy of computed fatigue life usage from spool speed (N) signals. On the other hand, the impulse response is immediately linked with noise reduction. A single spike in an otherwise smooth signal is reduced by the amount shown in Figure 9. E.g. the filter (2,2) will transform a single noise pulse into an output with 30% the height of the raw pulse with a subsequently decaying oscillation.

To give an idea on the coefficients of the Butterworth filters, 2 formulas are given:

The output $y(n)$ of filter (2,2) at time step n is determined by:

$$y(n) = (x(n) + 2x(n-1) + x(n-2) + 14y(n-1) - 5y(n-2)) / 13.$$

The corresponding formula for filter (4,4) is: $y(n) =$

$$0.0049061458 (x(n) + x(n-4)) + 0.019624583 (x(n-1) + x(n-3)) + 0.029436875 x(n-2) + 2.3615368686 y(n-1) - 2.301335 y(n-2) + 1.0470692 y(n-3) - 0.18576948 y(n-4).$$

The correspondence between group delay at zero frequency and the cutoff frequency is shown in Fig.8. The usual frequency scaling with the Nyquist frequency set to 0.5 is used in this and all figures with a frequency abscissa. For convenience the 25 investigated filters are addressed by their filter order and their delay time for linear input in the following discussion. Filter (5,2) means order 5 and 2 time steps delay at $f=0$. The conversion into cutoff frequency can be done using Figure 8. For example, the filter (5,2) has an approximate cutoff frequency of 0.22. In the Figures 9 and 10 impulse and step responses are shown for the filters (2,2), (3,3), (4,4) and (5,5), which all have a cutoff frequency near 0.1. This makes them suitable candidates for the demonstration of the effects of filter order.

Overshoot is only present for filter orders >1 , and its magnitude increases with increasing

Filter Application Examples

In this chapter results of the application of some of the proposed filters to measured engine signals are shown, together with the influence of filtering on the statistical properties of the modified signals. For the intended compressed coding, which capitalizes on statistical predictability, the advantages of applying filters will become visible in an example with heavy noise content (Figure 13).

As visible in Figs. 4 and 5, properly acquired spool speed (N) signals usually have a negligible noise content. That means that nearly every filter will have an influence on the nonrandom information contained in the signal. This may have further consequences, when the signal is used as input in a subsequent fatigue life usage calculation. Figures 11 and 12 demonstrate the effects of filter order on filtered signals, if the Butterworth filters (2,2) and (4,4) are applied to a high pressure spool

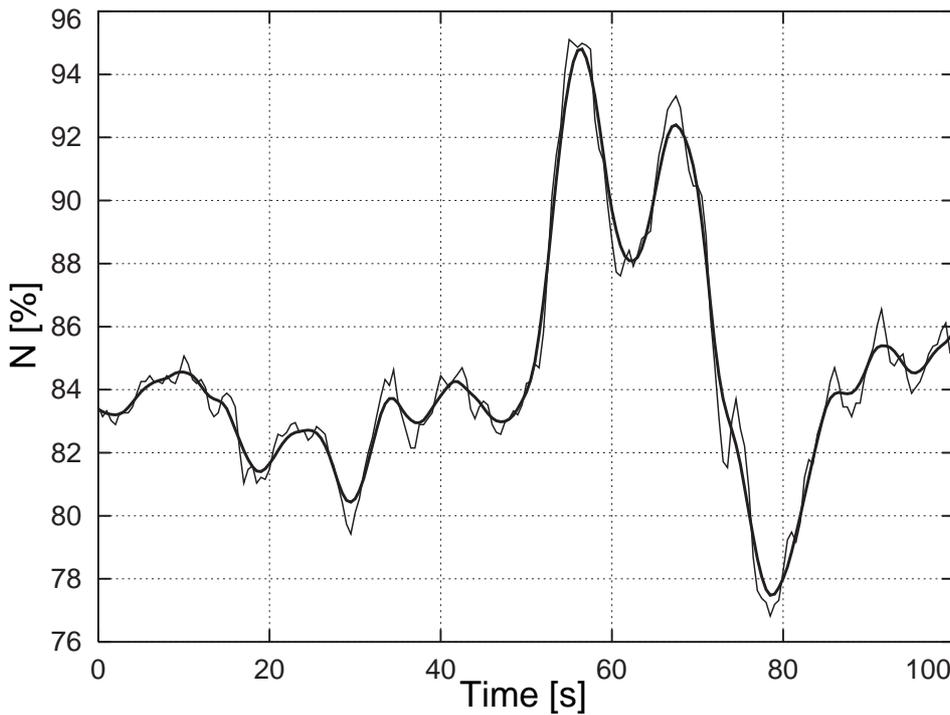


Figure 11: Application of Butterworth filter (2,2) on spool speed signal (corrected delay)

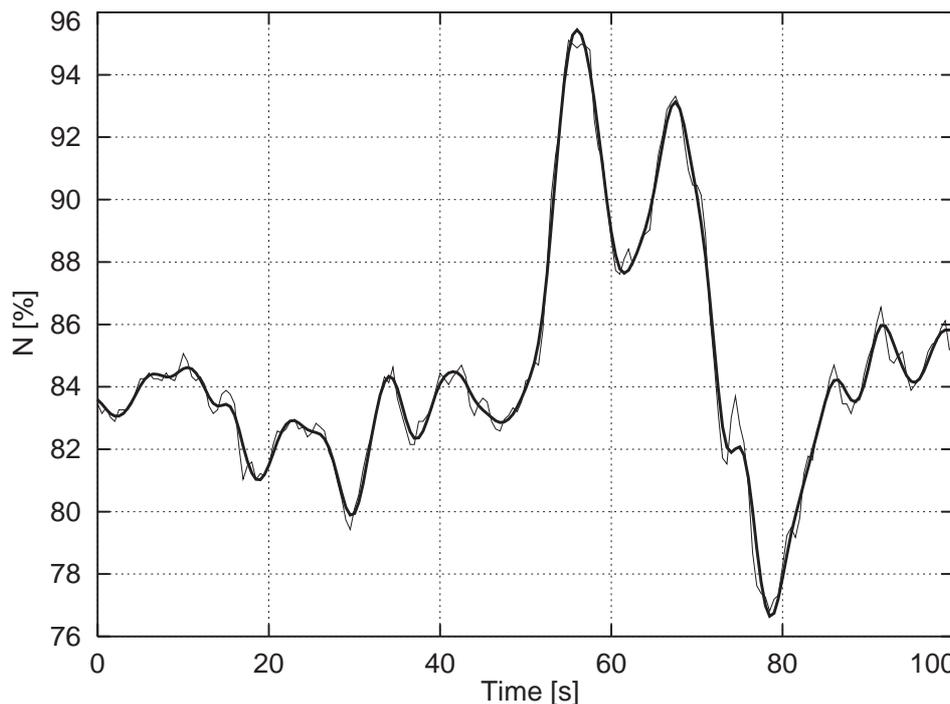


Figure 12: Application of Butterworth filter (4,4) on spool speed signal (corrected delay)

speed signal of a military jet engine, sampled at 2Hz. It is clearly visible that the filter (2,2) underestimates most peaks, whereas filter (4,4) follows the input signal more closely and produces an overshoot at the absolute maximum at Time \approx 56s. In both figures the time delay for $f=0$ is used to synchronize the output with the raw input. The same method would have to be applied when recovering data from a recording system that applies filters. Although theoretically only valid for slowly changing signals – note the strong frequency dependence in Fig.7 – shifting the filtered signal by an integer amount is computationally very simple and in most cases sufficiently accurate to bring into phase signals that have been processed by filters of different delay.

The second example deals with a turbine blade temperature (TBT) signal mea-

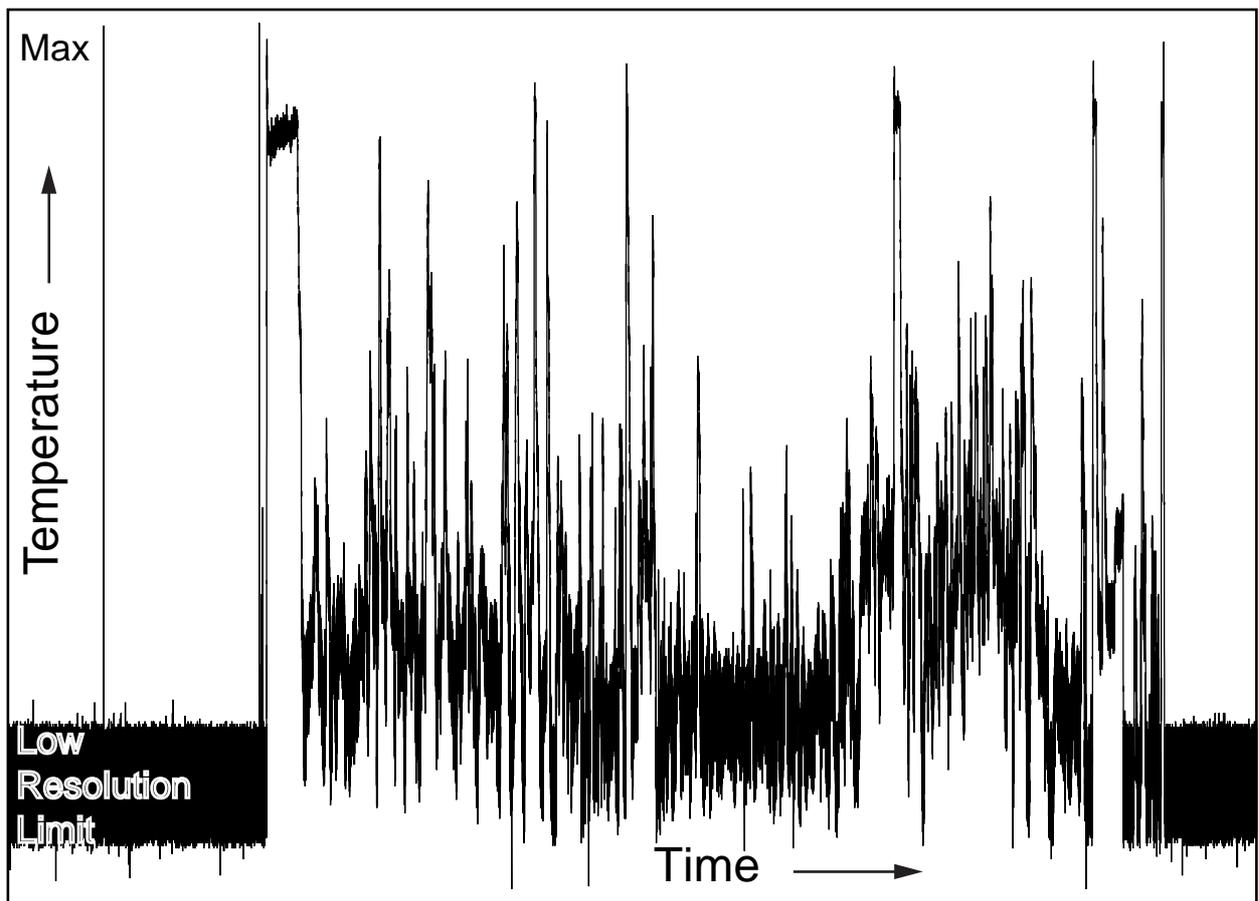


Figure 13: Turbine blade temperature raw signal with high noise content

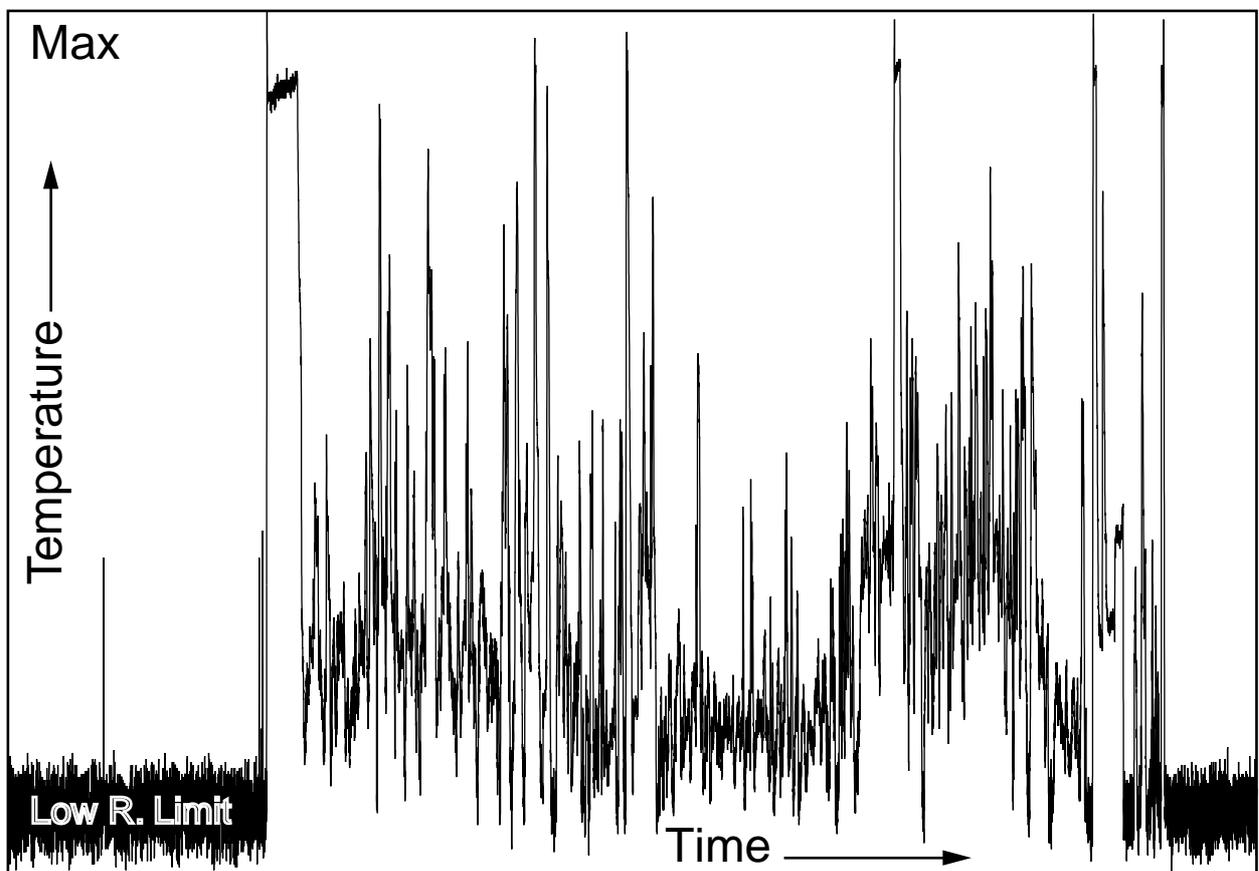


Figure 14: Turbine blade temperature signal after application of filter (2,2)

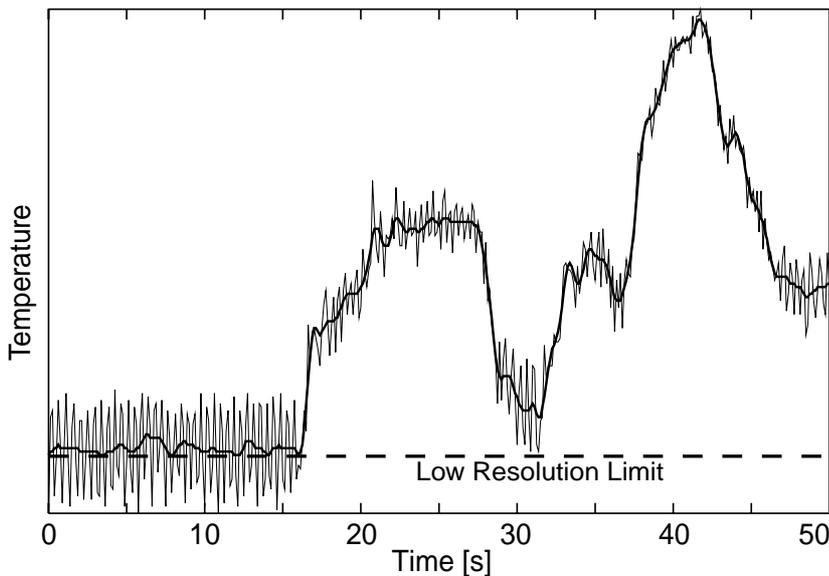


Figure 15: Turbine blade temperature signal with periodic noise, result of application of filter (4,4)

The measured TBT as input to some heat conduction equation. The temperature at the blade cross section considered to be at the highest creep risk therefore follows the TBT with some time delay and considerably smoothed.

The pyrometer output is fed into a linearizing amplifier, where it is converted into a voltage signal with sufficient signal strength to be input into a standard A/D converter for further processing or recording. Due to the $\sim T^4$ law of heat radiation no usable output is produced by the amplifier at idle or moderate power setting of the engine. This is acceptable for the engine control system, because only high values will have the chance to activate the limiter function. Therefore the function checks performed by the maintenance personnel often ignore the behavior of this signal at low power settings. The signal shown in Figure 13 has very high noise content, especially in the mentioned low resolution range. If such a signal is fed into a recording system that tries to reduce storage space using statistical prediction, the compression efficiency will be rather poor (it would, of course, be poor with any other compression technique, e.g. RLE, too).

A closer look at the signal details in Figure 15 reveals that there is a nearly periodic background noise of considerable amplitude superimposed to that part of the signal following the power setting of the engine. Without going into details this is a known phenomenon for certain combinations of amplifiers and control units for a particular engine type. For use in a creep life calculation, the “high” frequency behavior of the TBT signal is clearly of negligible influence. A suitably chosen filter can be used to cut off the superimposed fluctuations, with the side effect of greatly improving the predictability of the signal.

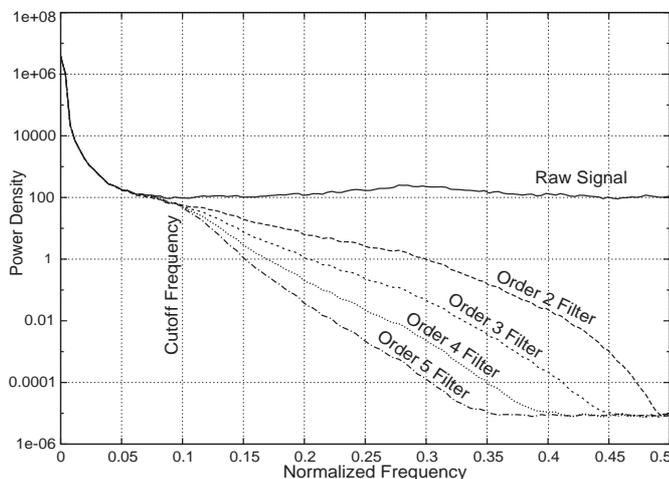


Figure 16: Influence of filter order on stop band attenuation

sured by an optical pyrometer. The main reason to measure this signal is for the engine control system, where it is used to prevent the engine from running to hot. Besides from that, the TBT signal is the most important parameter for monitoring creep life consumption of turbine blades. For some blade types creep may be the life limiting factor. Creep is usually only present, if high spool speeds and very high blade temperatures occur simultaneously. The metal temperature calculation in the creep monitoring algorithm usually takes the

To select appropriate filter parameters for a recording application it is advisable to look at the periodogram (power density spectrum of the autocorrelation function). Figure 16 shows a periodogram for the data of Fig. 13. As already noted for the N signal spectrum in Figs. 4 and 5, the spectrum starts with a $\sim 1/f^3$ part characteristic for the long-term correlations present in the signal. At $f=0.08$ the spectrum becomes horizontal, thus indicating “white noise”. There is an additional maximum around $f=0.28$, that coincides with the periodic content of the signal with a physical fre-

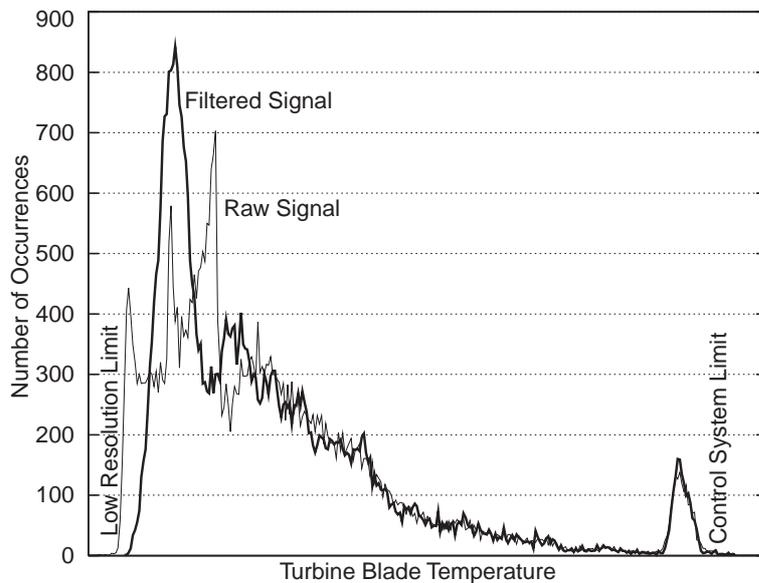


Figure 17: Histogram of raw and filtered temperature signal

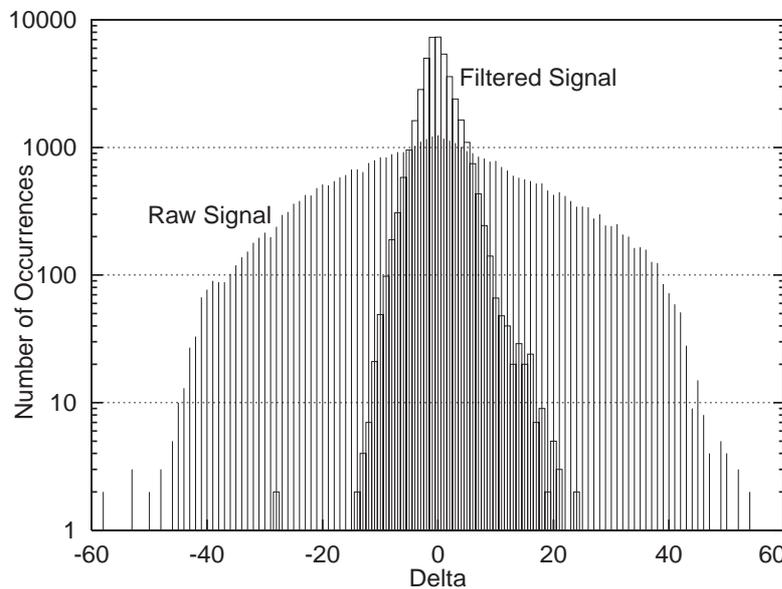


Figure 18: Influence of filter application on histogram of temperature delta values

quency $\approx 2.2\text{Hz}$ (sampling interval: $\Delta t=0.125\text{s}$), visible in Fig.15. The four filters already used as examples in Figs. 6 and 7 have been applied to the raw data of Fig.13. As expected the power density of the filtered signals starts to drop around $f=0.08$ and decreases at a rate dependent on the order of the applied filter (all 4 filters have approximately the same cutoff frequency $f_{\text{cut}}=0.1$). The effect of filter application is clearly visible also in the time domain, either in the overall plot of Fig.14, where filter (2,2) was used, or in the detail plot of Fig.15, with filter (4,4). Comparing Figs.13 and 14, the reductions of noise amplitudes in the low range and of sharp data spikes to $\approx 30\%$ of their raw value is to be noted. This is to be expected from the impulse response of the filter (2,2) shown in Fig.9.

Influence of Filters on Statistical Signal Properties

Using the TBT data of the previous chapter, some consequences of filter application will be shown. The histogram of signal occurrence counts in Figure 17 reveals an important result for this signal type.

The filter mainly influences the low temperature range, whereas the high temperatures are nearly unaffected. The 3 peaks in the raw signal histogram are a consequence of the periodic noise and are completely removed in the filtered signal. The filter output of the four different filters was fed into a creep calculation for a turbine blade, leaving other input parameters (e.g. spool speed) unchanged. The relative difference between all results was less than 0.08%, including the result with the raw TBT. The reason is that only temperatures in the peak at the upper end of the range can significantly contribute to creep damage. Other parameters indicating blade creep, as time spent above certain temperature limits will also remain unaffected. The use of such simple counts for inspection planning is discussed in [Bra00], showing only a poor blade failure prediction capability.

The most important signal properties for the application of delta encoders are of course the bandwidth and distribution of differences between successive samples (delta). A small bandwidth with most data centered around zero will improve the prediction success of a coder, thus improving compression rates. Fig. 18 (note the logarithmic scale) shows a dramatic reduction of the delta scatter. Even a very simple coder only exploiting the number of bits necessary to code the delta values could save nearly two bits/sample due to the reduced range. More sophisticated coders, whose description is beyond the scope of this presentation, use statistical information gathered

Figure 19:
 Occurrence counts
 for order-2 contexts
 of raw blade
 temperature delta
 signal

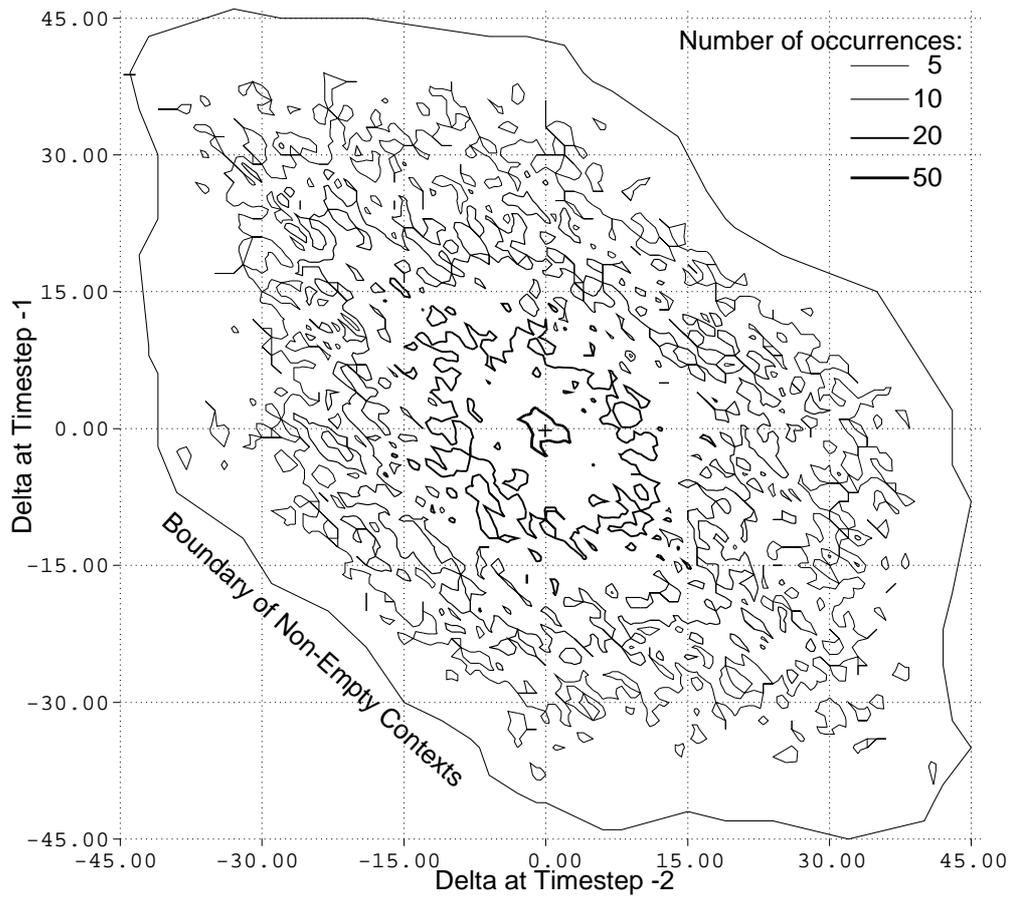
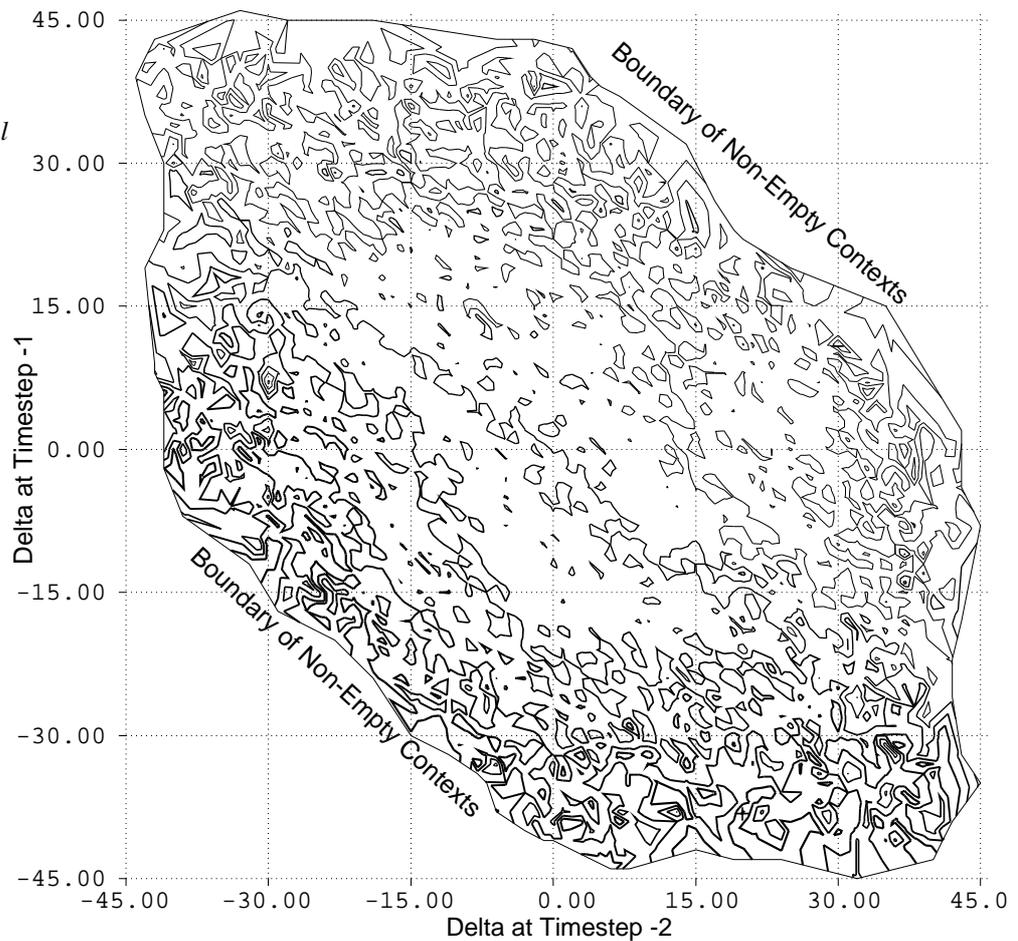


Figure 20: Mean
 successor delta of
 order-2 contexts for
 raw temperature signal



Average Successor Delta:

<u>-3.000E+01</u>	<u>-2.000E+01</u>	<u>-1.000E+01</u>	<u>0.000E+00</u>	<u>1.000E+01</u>
<u>2.000E+01</u>	<u>3.000E+01</u>	<u>4.000E+01</u>		

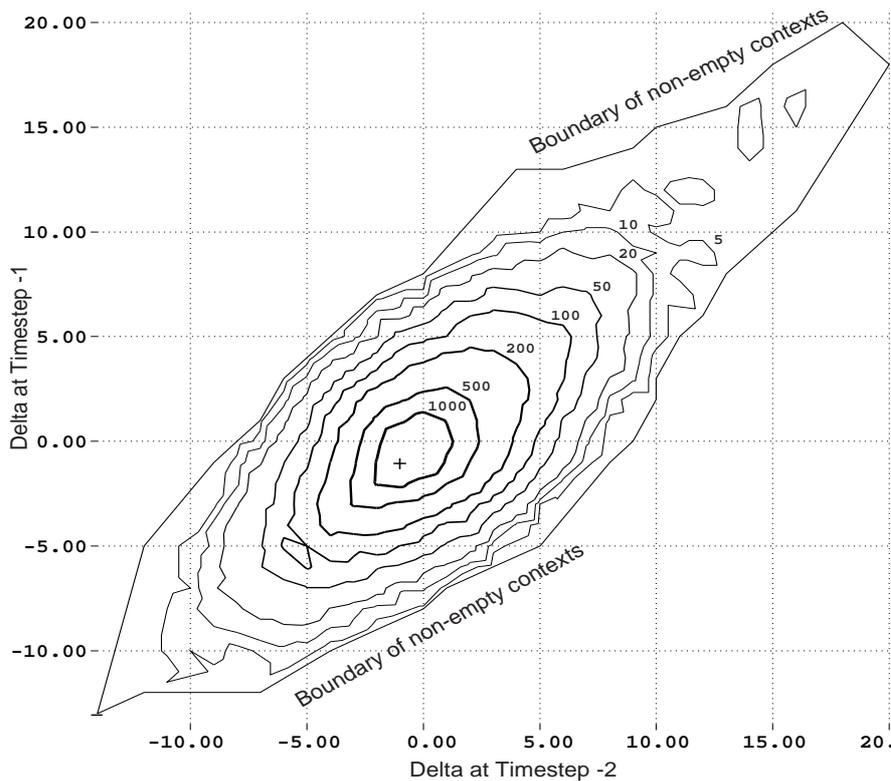


Figure 21: Occurrence counts of order-2 contexts after application of filter (2,2)

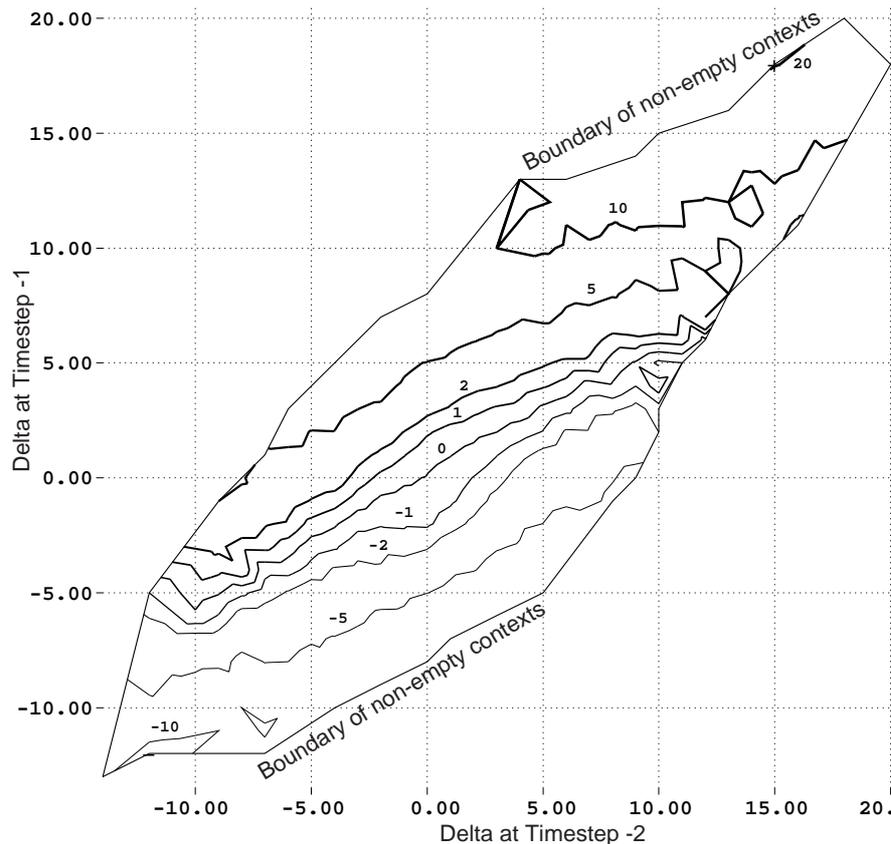


Figure 22: Mean successor delta of order-2 contexts after application of filter (2,2)

average next delta of 2). A coder can benefit from that type of information by adjusting its predictions accordingly.

There is an enormous literature on efficient data compression methods. Beating the compression efficiency of the best available general purpose coders requires very sophisticated tailoring with respect to a special data type. If the delta values for engine or flight data are limited to the 8-bit

during the collection of the data to code each new value exactly according to the accumulated knowledge about its probability at the time when the parameter is arriving. To give an idea about the sort of information used for statistical coding, two pairs of figures, Figs.19 and 20 for the raw signal (Fig.13) and Figs. 21 and 22 for the same data after filtering (Fig.14) are shown. Figs.19 and 21 show, how many times successive combinations of delta values occur during the whole flight (the total number of data points is ≈ 42000). For the raw signal the picture is rather discouraging, showing only a marginal increase of counts around the origin and the boundary of nonzero contexts covering a large area. The contexts of the filtered signal are well centered and there are no outliers.

Figs. 20 and 22 show the average successor delta value after the occurrence of two preceding deltas. Whereas the picture for the raw signal is again not easy to read (contexts below the $\approx 140^\circ$ meandering line through the origin are followed by positive successors on the average), the contour plot for the filtered signal gives a well structured picture (e.g. a pair of two deltas of 5 predicts an

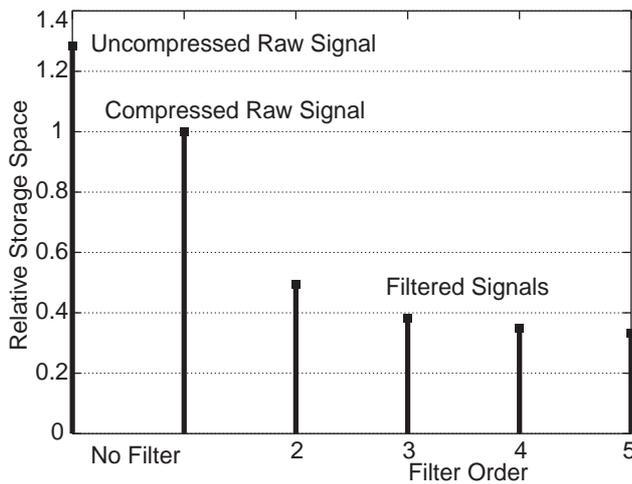


Figure 23: Influence of filters on the compressibility of delta coded turbine blade signals

range, arbitrary text coders [BCW90] can be used for compressing those data. Because a comparison of the performance of some of the best coders applied to delta coded engine signals showed only marginal differences in compression rates, the well documented and freely available “ARITH-N” program in [NG96] was used to determine the compression rates in the present investigation. Figure 23 gives some results for the TBT signal example, demonstrating the reduction of required storage space that can be accomplished without any adverse influence on the useful information content of the stored signal. A comparison was made with the memory occupied by the TBT signal on the new flight data

recorder of the German Tornado aircraft [SS95], that uses a RLE method with an adjustable threshold for ignoring small parameter changes. The 42000 data in Fig. 13 consume 67.6k byte of storage space, whereas the application of filter (5,5) plus statistical coding of the delta signal squeeze the data volume down to 10.7 k byte, a factor > 6.

A Simplified Model for LUM

To assess the fatigue life usage of rotating engine parts, a mathematical model has to be developed, that is simple enough to be either executed in an on-board LUM system or to process large amounts of stored flight data after they have been downloaded and stored. A short outline of the method is presented here. A more rigorous treatment is given in [Gra00]. During the design and development testing of a new or modified engine component finite element calculations of the thermal and mechanical behavior are performed to verify that the requirements on safety and durability are fulfilled. During this process some high stress areas are identified on the component, that are considered to be candidates for cracking. Because the stress at those areas also comprises thermal stresses a model for the temperature development within the component is also needed. This model calculates metal temperatures at selected points, some of them coinciding with the critically stressed areas. Fig. 24 shows a compressor disk with 6 temperature points, 4 of them critical areas.

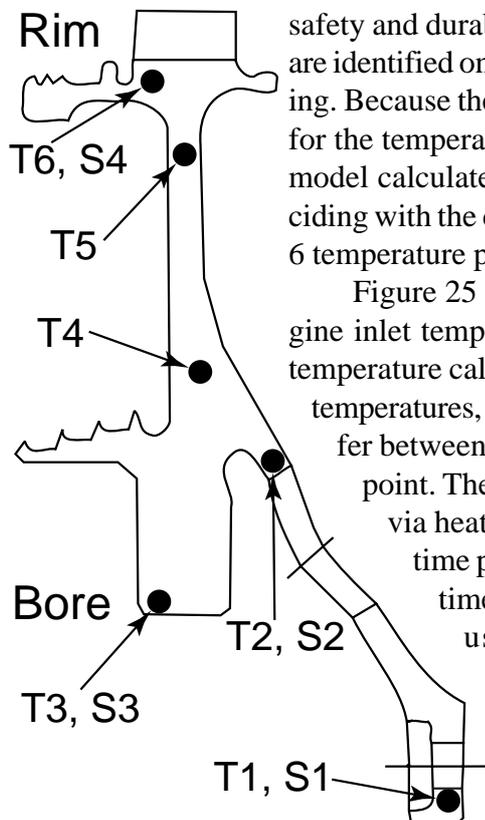


Figure 25 outlines the most important steps of the calculation: The engine inlet temperature T_{in} and the spool speed N are used as input into the temperature calculation. Both parameters are used to determine the local gas temperatures, that influence heating or cooling of the component. Heat transfer between gas and metal is a nonlinear function of the engine operating point. There is a mutual influence of the temperatures in the component via heat conduction, indicated in Figure 26. The temperatures at one time point n are influenced by all other temperatures at the previous time point $n-1$. The updated temperatures and the spool speed are used to compute the stresses at the critical areas:

Figure 25 outlines the most important steps of the calculation: The engine inlet temperature T_{in} and the spool speed N are used as input into the temperature calculation. Both parameters are used to determine the local gas temperatures, that influence heating or cooling of the component. Heat transfer between gas and metal is a nonlinear function of the engine operating point. There is a mutual influence of the temperatures in the component via heat conduction, indicated in Figure 26. The temperatures at one time point n are influenced by all other temperatures at the previous time point $n-1$. The updated temperatures and the spool speed are used to compute the stresses at the critical areas:

$$S(n) = a + b \cdot (N(n))^2 + \sum_{i=1}^l g_i T_i(n)$$

with a, b, g_i coefficients specific for each area.

Figure 24: Temperature and stress areas for compressor disk

The process is similar for all critical areas. The metal temperature at the critical area is then used to compute a temperature dependent material strength (i.e. $UTS(T)$) which usually drops with increasing T .

$S_{Norm}(n) = \frac{S(n)}{UTS(T(n))}$ is then used as input into a cycle extraction process, which sorts the stress extremes into closed hysteresis loops. By using S_{Norm} instead of S the well known fact is considered, that for two equal stress cycles the one with higher T will contribute a higher damage. The cycle extraction produces cycles (n_1, n_2) , (n_3, n_4) , ... Each cycle is then converted into an equivalent 0-max tension cycle with assumed equal damage. The well known Goodman correction is used:

$$S_{0-max}(n_{max}, n_{min}) = \frac{UTS(T(n_{max})) \cdot (S(n_{max}) - S(n_{min}))}{UTS(T(n_{max})) - S(n_{min})}$$

The Goodman formula usually has to be amended by further corrections (e.g. precautions for the denominator becoming 0 or negative). Next the specific stress concentration factor k_t for the local geometry of the critical area and the material's infinite cyclic life threshold $FCUT$ are used to calculate a damage parameter:

$$S_{aux}(n_{max}, n_{min}) = \frac{S_{0-max}(n_{max}, n_{min}) \cdot k_t}{S_{cut}(n_{max})} - FCUT$$

S_{cut} is a temperature dependent threshold stress value. In the present investigation

$S_{cut}(n_{max}) = FCUT \cdot UTS(T(n_{max}))$ was used. To normalize all damages at a critical area to the damage of the largest stress cycle of the design mission, the Goodman-corrected S_{0-max} of this cycle is used to calculate

$$S_{ref} = \frac{S_{0-max, design} \cdot k_t}{S_{cut}} - FCUT$$

Here $S_{cut} = FCUT \cdot UTS(T_{ref})$ is assumed, where T_{ref} is a characteristic high temperature at this area, usually that corresponding to the maximum stress of the design mission. If S_{aux} is calculated from S_{0-max} of an arbitrary cycle, only positive values will contribute a damage increment:

$$D(n_{max}, n_{min}) = \left(\frac{S_{aux}(n_{max}, n_{min})}{S_{ref}} \right)^{ESN}$$

Other cycles are said to fall below the fatigue cutoff. By virtue of the definition of S_{ref} , $D=1$ will be computed for the largest cycle of the design mission. Life releases for a particular component type of an engine usually are expressed as multiples of this value.

ESN is the exponent in the so-called S/N curve, which expresses the number of survived stress cycles as a function of the cycle's magnitude for a certain material. This parameter determines, how

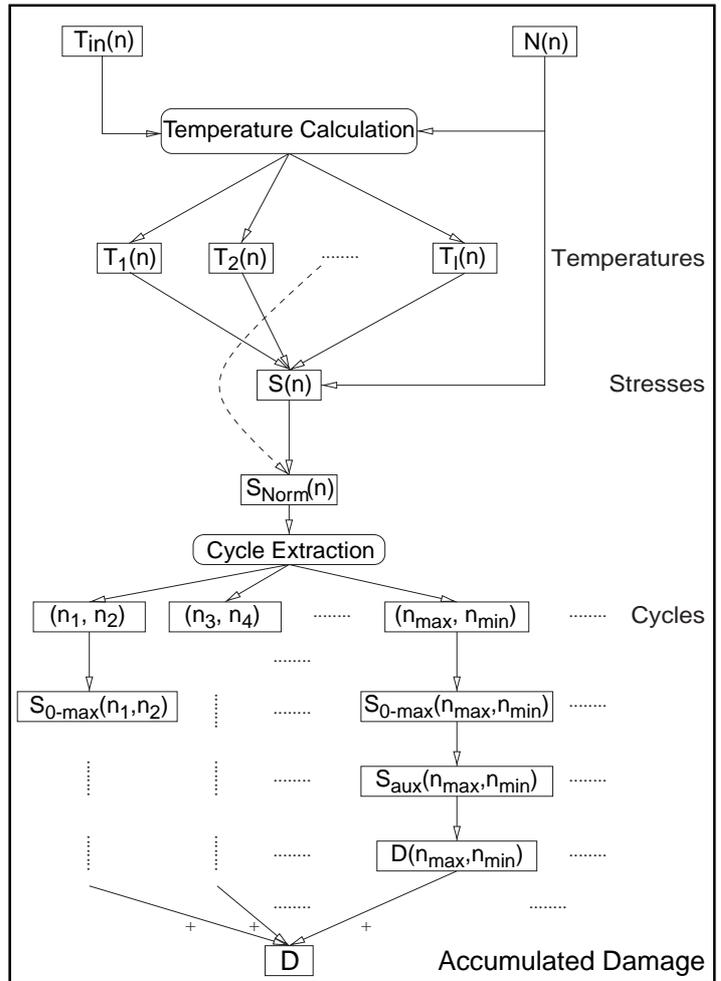


Figure 25: Overview of cyclic fatigue life usage calculation

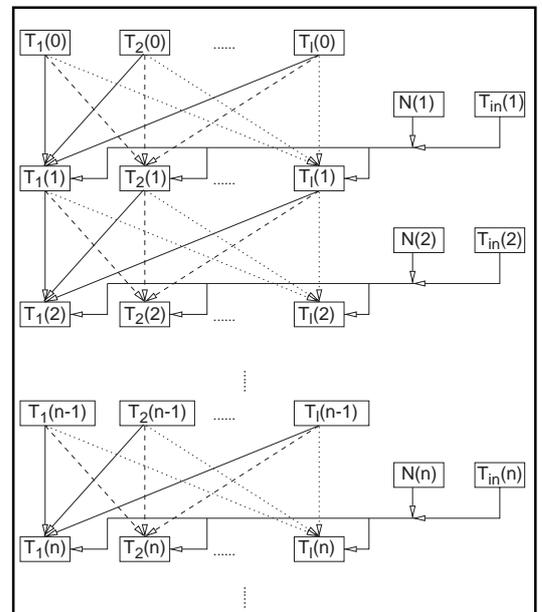


Figure 26: Calculation of metal temperatures

fast the damage grows with increasing stress level. For large values of ESN even small changes in the computed stresses will have a considerable influence on the computed damage results. This parameter plays a decisive role in the determination of the required accuracy for the storage of the spool speed signals.

This is easily seen for a critical area with low thermal stresses, where the total stress varies $\sim N^2$. If additionally no fatigue cutoff would be present, the damage would vary $\sim N^{2 \cdot \text{ESN}}$. Since ESN may assume values as high as 4, a small change in the spool speed signal might be amplified by a factor of 8 in the damage calculation. If fatigue cutoff is present, this effect becomes even worse. For the life limiting critical area of the compressor disk in Fig.24 (Area 2), which has $\text{ESN}=2.5$, the presence of fatigue cutoff would amplify a spool speed increment of 1% at design conditions into an increase in damage of 6.9%, assuming that the total stress is only dependent on N^2 .

Another effect immediately influencing the sensitivity of calculated damage to small variations of the spool speed signal is the ratio of thermal stresses and centrifugal stresses in the stress law. Stresses at areas with a high portion of thermal stresses tend to react to spool speed changes with some delay thus de-coupling the highest stress events from “natural” occurrences of the highest spool speeds (e.g. at the start of the aircraft).

Influences of Filtering

As preliminary tests with various types of flight data indicated some potential benefit of applying lowpass filters before trying to store the data, and no generally applicable rule was found, how to select the filters and their parameters, a decision was made to perform some systematic tests with the maybe most important single parameter entering the LUM calculation, the spool speed signal of the HP spool. An available calculation model for the compressor disk shown in Fig. 24 was used, because of its representative nature and its relative simplicity involving various types of critical areas.

Because of the plan, to track the propagation of data modifications throughout the whole calculation process outlined in the previous chapter, it was necessary to limit the number of flights to 24, also due to limitations in available data storage capacity for a student’s project. The data were carefully selected from an existing pool of recorded flights from all 3 owner nations of the Tornado aircraft.

Investigation Method

The data from both engines in an aircraft were used. An in-house program of the MTU stress department for the life consumption analysis of rotating components (Mission Analysis Program) was used to compute all relevant data with the two input parameters T_{in} and N taken from the flight data recordings. Temperatures and stresses and the normalized stresses were computed and stored for every time point in the flights for all areas indicated in Fig.24. All cycles found by the cycle extraction, together with the damage at each area were also stored. The results of the unfiltered data were used as a reference. The same computation was repeated with the spool speed signal replaced by the delay-corrected output of each of the 25 lowpass filters. The inlet temperature was not filtered. 1200 sets of files (24 recorded flights · 2 engines · 25 filters) were stored to enable a statistical analysis of the deviations between the reference using the raw data and the results with the filtered data. The differences between computed temperatures, stresses, normalized stresses and damages with respect to the values based on the original flight data were computed. Histograms of the deviations were used to get an idea about the resulting distributions. Basic statistical parameters, as expected values, medians, variances, ranges and quantiles were computed for all difference data. A detailed description and comprehensive results are given in [Gra00].

The original and filtered spool speed data were converted into delta values and were fed into the „ARITH-N“ file compression program [NG96] with model orders set to 1,2,3. The resulting output sizes were recorded. Note the difference between filter order and coding order (the maximum number of predecessor values used in the coder’s internal statistics tables). Summarizing the

coding results, it turned out, that coding with order-1 prediction was always optimal for the raw spool speed values and in most cases also for the data with filters (1,1),(1,2) and sometimes (2,1). For all other filtered data order-2 coding resulted in best compression, whereas order-3 was nearly never better than order-2, with single exceptions for filter (5,5).

The median value of storage space reduction by the statistical arithmetic coder was 3.7 for the raw N data relative to the storage of uncompressed 8 bit delta values. This means, that the average data volume for the 2Hz spool speed signal for 1 hour of flight time was 1944 bytes. These data permit a lossless reconstruction of the original signal, which had a 11 bit accuracy. The additional gain, that can be reached by filtering the data is shown in Fig. 27. In contrast to the previously shown results for the noisy TBT signal (Fig.23), the compression gain is now only a function of the cutoff frequency and nearly independent of the filter order with the exception of order 1 filters, which

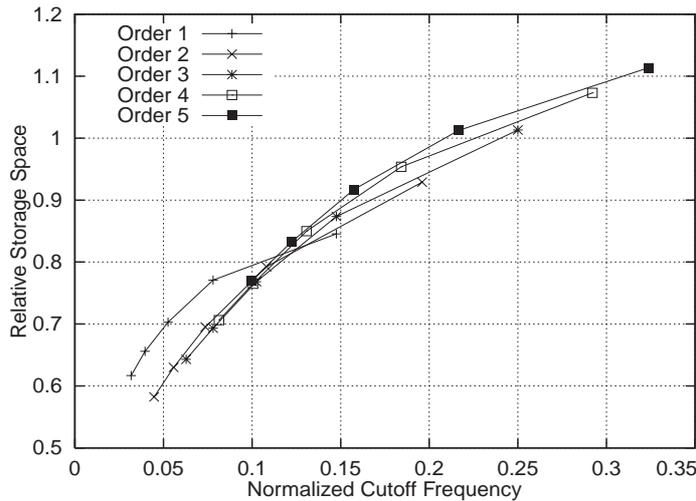


Figure 27: Size reduction factors for filtered N signals

behave different due the lack of overshoot in their step response. The 4 filters with cutoff frequency > 0.2 even cause a deterioration of compression rates due to their tendency to amplify overshoots in the signal.

Accuracy Loss of Computed LUM Results

Omitting a detailed discussion about the various effects of the input signal filtering on temperature and stress development, a summary of the effects on the final result, i.e. the computed fatigue life consumption

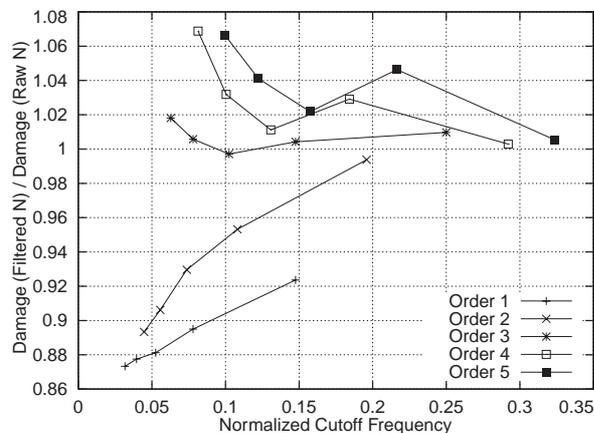


Figure 28: Influence on damage results for area 1

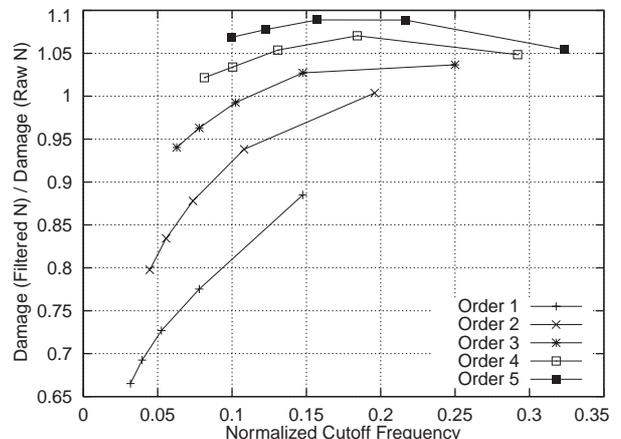


Figure 29: Influence on damage results for area 2

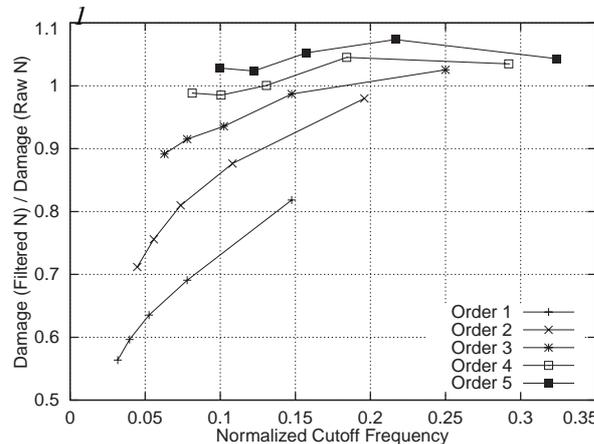


Figure 30: Influence on damage results for area 3

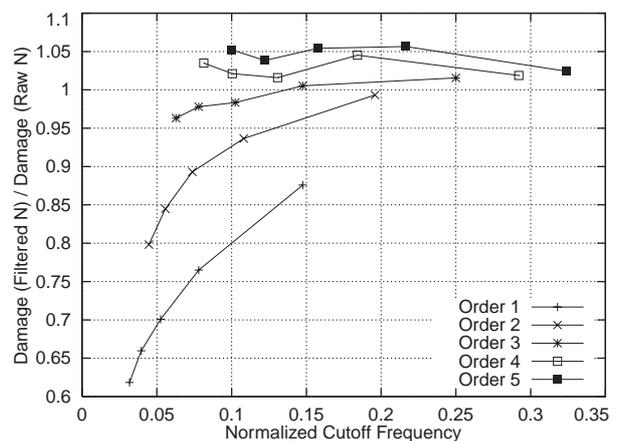


Figure 31: Influence on damage results for area 4

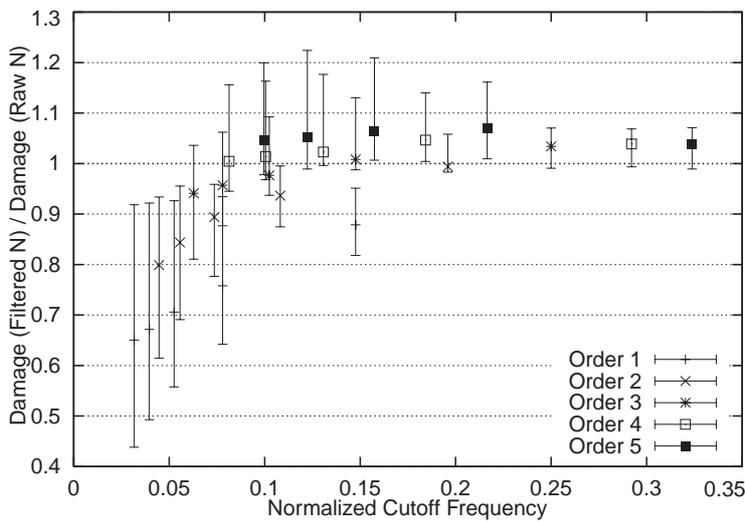


Figure 32: Scatter bands of computed damage for area 2

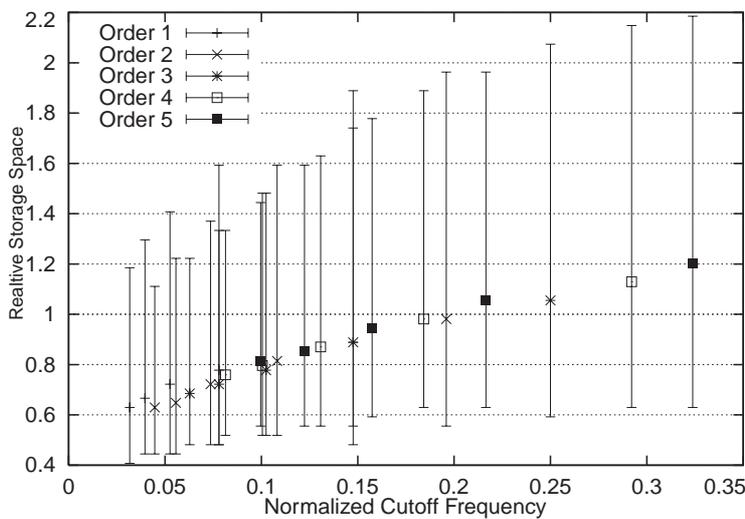


Figure 33: Scatter bands of storage requirements after application of filter and compression

result from either end of the scatter band in the plot. The scatter is lowest for filters with high cutoff frequencies, because their output tends to follow closely the (physically correct) overshoots of the spool speed signal. With decreasing filter cutoff there is not only an average under-prediction of damage, but also an increase in the scatter. One remarkable effect is the clustering of scatter bandwidths for filters with equal group delay, i.e. the results for filters (5,1), (4,1), (3,1), ..., form one group, the filters (5,2), (4,2), ... the next one, and so on.

Figure 33 shows the ranges of required storage space. Again one result from each the lower and upper end of the scatter band has been omitted from the plot for reasons of consistency. The symbols inside the range bars are at the same locations and the same scaling is used as in Fig.27, that means the storage space values are divided by the median of the storage space required by the compressed raw signals. The relative scatter varies only slightly with a minor increase towards the „less-intrusively“ filtered signals (filters (5,1),(4,1),...with cutoff frequencies > 0.2). As a rule of thumb, a factor of 2 may be assumed for the maximum deviation of the storage space from the median for a single flight for an arbitrary choice of filters. The largest scatter [0.52,2.26] occurs, as expected, for the raw signals, whose scatter band is thus exceeding slightly the worst case range of the filtered data. The variability in compression ratio would have to be taken into account for the design of a recording system, because it determines the extra memory to be allocated to avoid memory overflows and corresponding data loss, if the readouts are to be performed after a fixed number of flights or after a fixed engine operating time.

at the critical areas is now given. The results are shown in Figures 28 - 31, each corresponding to one critical area. Some common features are easily visible: The application of order 1 filters always leads to a severe underestimation of computed damage results. Most of the order 2 filters also produced results with too low damage. Filter (2,1) performed remarkably well with damage ratios close to 1.0 for all areas. The damage ratios of the 5 filters with order 3 were grouped around the desired value 1.0 with best results for the filters (3,2) and (3,3). The filters with order 4 and 5 generally yielded an overestimation of damage results. For area 3, the filters (4,3), (4,4), (4,5) and (3,2) yielded better results than all other filters.

These results are average results over all investigated flights. They do not preclude much higher deviations of the damage results for a particular flight. Figure 32 shows the scatter bands of the damage results at area 2 for all combinations of flights and filters. Since there was at least one result, that was considered as an outlier due to problems in the recording process, it was decided to omit one result

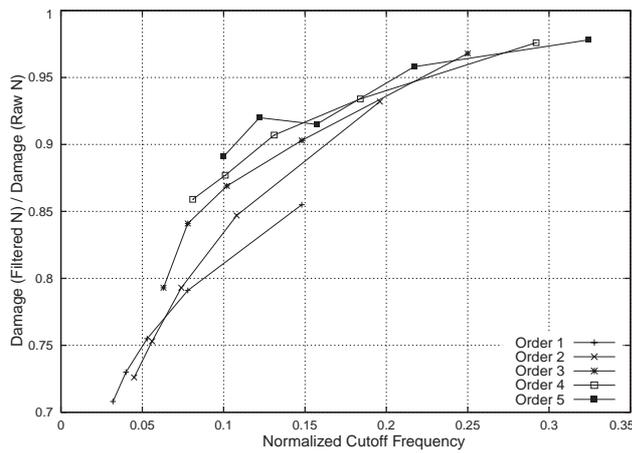


Figure 34: Influence of filter application on computed damage for critical area in HP compressor

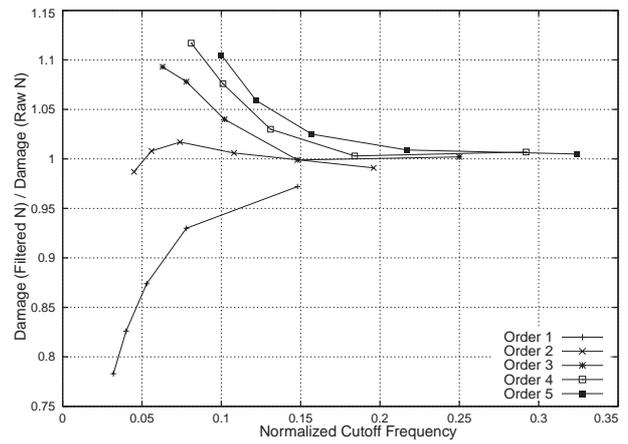


Figure 35: Influence of filter application on computed damage for critical area in HP turbine

The dependence of accuracy on filter parameters shown in Figures 28-31 does not allow to derive a simple strategy for the filter selection. Some selected results of an independent study with the same filter family applied to the same signal type (HP spool speed), but only to a single flight are shown in figures 34 and 35. In this study all 34 areas influenced by the HP spool speed of the RB199 were included and checked for deviation from the damage results calculated with the original signals. Some of the obtained dependencies were found to be very similar to those shown in the figures 28-31, but also new types of dependencies occurred. In Fig.34 the damage results almost entirely depended on filter cutoff frequency, with only weak influence of filter order, whereas Fig. 35 shows a completely different behavior, with an increased over-prediction of damage with decreasing cutoff frequency for all filter orders > 1 .

Conclusions

The influence of filter application on the results of a fatigue life usage calculation is strongly dependent on the type and the noise content of its input signals. For temperature signals and other signals contributing to damage only via integrating algorithms, suitable filter parameters can be derived from a spectral analysis of the signal's autocorrelation function. Filter cutoff frequencies can be selected to remove all noise components found in the spectrum, leaving a smoothed filter output with great potential for efficient data compression.

Properly acquired spool speed signals have a very low noise content. Even small changes to this signal may have an immediate influence on the computed life consumption results. The dependence of the results on the filter parameters is very complex and it is neither possible to provide a generally applicable law nor to predict the magnitude of the deviations, if more than one critical area or even different engine components are affected.

Only modest additional gains in data compression rates can be accomplished by filtering signals with low noise content, unless considerable information loss is accepted. By applying delta coding together with available statistical data compression methods impressive compression rates can be achieved already for the original signals. Therefore it is recommended to avoid filter application to spool speed signals.

Using appropriate filtering and compression methods to each signal type would result in very low storage requirements. To give an order of magnitude: 10k bytes of memory will be sufficient to store one hour of engine operation without any adverse influence on the accuracy. A recording system entirely dedicated to the data needed for engine LUM calculation could overcome some of the disadvantages of existing on-board engine monitoring systems. It could be integrated as a separate task in an existing monitoring or engine control system.

References

- [BCW90] T.C.Bell, J.G.Cleary, I.H.Witten: Text Compression. Prentice Hall, Englewood Cliffs, N.J., 1990
- [Ber94] J.Beran: Statistics for Long-Memory Processes. Chapman & Hall, New York 1994
- [BP97] J.Broede, H.Pfoertner: OLMOS in GAF MRCA Tornado - 10 Years of Experience with On-Board Life Usage Monitoring. 33rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, Seattle 1997
- [Bra00] B.Bradford, S.Rose, B.Bloomfield, R.Greenwood: Engine Review. FY99 Engine-Related Mishaps. United States Air Force Flying Safety Magazine, January/February 2000
- [Gra00] S.Graf: Filterung und Datenkompression von Flugdaten zur Unterstützung der Zustands- und Lebensdauerüberwachung von Triebwerken. Diplomarbeit im Fachgebiet Mathematik, Universität Passau, Fakultät für Mathematik und Informatik, 2000
- [Gro99] D.R.Grossi: Aviation Recorder Overview. International Symposium On Transportation Recorders, Arlington, Virginia, May 3-5, 1999
- [HK99] R.A'Harrah, G.Kaseote: A Case for Higher Data Rates. International Symposium On Transportation Recorders, Arlington, Virginia, May 3-5, 1999
- [NG96] M.Nelson, J.-L.Gailly: The Data Compression Book, 2nd Edition. M&T Books, New York, 1996
- [PB87] T.W.Parks, C.S.Burrus: Digital Filter Design. John Wiley & Sons, New York, 1987
- [PR95] H. Pfoertner, C.Ross: Preparing Life Usage Monitoring for the Next Decade. 18th International Symposium AIMS, Stuttgart 1995
- [OS89] A.V.Oppenheim, R.W.Schafer: Discrete-Time Signal Processing. Prentice Hall, Englewood Cliffs, N.J., 1989
- [SS95] A.Schick, U.Schulz: Technical Logistic Support with Modern Solid State Flight Data Recording System, 18th International Symposium AIMS, Stuttgart 1995
- [SM89] M.E.Sigrist, A.N.Muggli, SWISSAIR Studies on the Compressibility of Aircraft Data. 15th Symposium AIMS, Aachen 1989
- [Smi97] S.W.Smith: The Scientist and Engineer's Guide to Digital Signal Processing. California Technical Publishing, 1997
- [SD93] S.D.Stearns, R.A.David: Signal Processing Algorithms using Fortran and C. Prentice Hall, Englewood Cliffs, N.J., 1993

Acknowledgments

Part of this work is based on research performed by Simone Graf at the structural mechanics department of MTU Munich within the context of her Master's Thesis in Mathematics. Supervisor was Professor Siegfried Graf of the Department of Mathematics and Computer Science at the University of Passau, whose main focus is on measure theory and integration. His help and continuous support is gratefully acknowledged. Many thanks also to Jürgen Broede, who provided many ideas and his excellent mission analysis program, to Guido Dhondt and Manfred Köhl for help and advice in many questions on mathematical modelling and last but not least to Ottomar Gradl, who helped us manage all the problems related to multi-Gigabyte data storage and handling, that were necessary for the statistical analysis of the flight data.

List of Acronyms

ESN ...	Exponent in S/N Curve	n ...	Number of Time Point
FCUT...	Fatigue Cutoff	N ...	Rotational Spool Speed
FDR ...	Flight Data Recorder	OLMOS	On-board Life Monitoring System
HP ...	High Pressure	RLE ...	Run Length Encoding
LUM ...	Fatigue Life Usage Monitoring	TBT ...	Turbine Blade Temperature
MAP ...	Mission Analysis Program	UTS ...	Ultimate Tensile Strength