The Information Content of Turbine Engine Data - A Chance for Recording-Based Life Usage Monitoring[1 2]

Hugo Pfoertner
MTU Aero Engines GmbH
P.O. Box 500640
80976 Munich, Germany
(++49)-89-1489-6185
hugo.pfoertner@muc.mtu.de

Abstract—The most important input signals for calculating the life consumption of fracture critical parts of turbine engines were analyzed for their information content. This information content determines the data volume to be transmitted and stored, if instead of calculating life usage in real-time in the on-board monitoring system this calculation is postponed to later processing in a ground-based support system. Signals from various sources are shown and processes contributing to an increase in information-theoretic entropy are identified. Some methods to avoid unnecessary components of the composition of signals are proposed.

It is shown that a considerable part of the entropy determining the storage requirements can be either avoided or removed, if only that part of the information is retained that has a deterministic influence on the results of a variety of algorithms for life usage monitoring. If proper rate conversion, quantization, low pass filtering and noise suppression is applied, highly efficient methods for data compression based on delta coding, statistical adaptive prediction models and arithmetic coders can be used to reduce the data volume to astonishingly low figures. On-board storage of 100 hours of engine operation and archiving the complete running history in the ground support system could remove many of the shortcomings of existing LUM systems.

TABLE OF CONTENTS

1. INTRODUCTION

Principles of LUM

Life usage monitoring (LUM) of fracture critical parts is considered to have a great potential to save costs without compromising flight safety. The general ideas of LUM are now well established and have been documented in comprehensive reports [1, 2]. Although some of the existing systems have proven their cost effectiveness and are well accepted by their users, the need for continued adaptations of LUM algorithms to the experience gained during fleet operation and to changes of engine hardware requires an unexpectedly high amount of system maintenance.

Current systems for monitoring the life consumption of the fracture critical engine parts rely on increasingly complex algorithms to determine the transient thermal and mechanical behavior of rotor structures as a function of the engine and aircraft data that are acquired in real time.

Those algorithms have regularly to be adapted to changes of the engine, e.g. introduction of redesigned components or to in service experience indicating that rotor areas not covered by the existing algorithms might become life-limiting [3]. If those algorithms are implemented in an onboard engine monitoring system the associated update process of the monitoring software in an aircraft fleet needs careful logistic planning and may cause considerable costs sometimes exceeding the savings of individual monitoring of the parts.

On-board Monitoring or Recording

Life usage monitoring does not necessarily require the presence of a fully featured on-board monitoring system, e.g. the OLMOS system [4] for the Tornado fleet of the German Air Force (GAF), but can also be performed based on recorded engine and aircraft data.

Whereas the GAF has decided to perform LUM for their whole Tornado fleet on-board, the Italian Air Force (IAF) pursues a different approach. All Italian Tornado aircraft are

[2] IEEEAC paper #408, Updated Dec 12, 2001

equipped with a tape based maintenance recorder. The data recorded by this device are downloaded after each flight, i.e. the tape cassette is exchanged and the data are transferred to a ground station computer that is part of the MaRe (Maintenance Recording) system. This data download has now been performed since several years, even though only a preliminary version of the ground station was available. The data were read from the tape and stored in one large database. The engine related data were transferred to the engine manufacturer and are stored for later processing. The current size of this database is several 10000 flights.

All aircraft in the Italian Tornado fleet are equipped with the MaRe system. The final target is to achieve individual monitoring of every single engine. Whereas the German and Italian air forces have their systems configured for monitoring individual engines, including a tracking of single life limited components, the UK Royal Air Force has a few Tornado aircraft equipped with a tape recorder for engine data (EUMS). The applied sampling technique tries to estimate the life usage of the RB199 engines using statistical reasoning based on the correlations between the life usage calculated for the sample data and the (unknown) usage seen by the not monitored engines. The high degree of uncertainty in this approach requires large safety margins.

Past experience indicates that only a fleet-wide fit of LUM is able to remove the uncertainty introduced by statistical predictions to the level required to exploit the full life potential of fracture critical parts. To improve the statistical basis a retrofit of modernized flight data recorders is currently planned to replace the obsolete EUMS recorders in the UK.

Other examples of installed data recording systems are the monitoring systems for the French Rafale [5] and recording systems used by the US Navy [6].

Advantages of Data Recording

As already mentioned, the algorithms used to compute the life usage of rotating components are highly dependent on the engine configuration. To avoid logistic limitations, the on-board LUM software must therefore be able to cope with any existing engine standard and combination of components. If new variants of components are introduced, this requires an adaptation of the on-board software and the corresponding data handling software in the ground support system. To avoid the cost of frequent fleet-wide retrofits of the monitoring software, long delays between the availability of updated algorithms and their introduction into the on-board software have to be accepted.

This limitation is not present, if the LUM calculation is performed for recorded data. With proper configuration control an update of algorithms can be performed within very short times thus reducing the logistic complications during the introduction of improved components into the fleet.

The greatest advantage of having recorded data is the ability to reassess the running history of those parts, for which it turns out after years of usage that the assumptions made on life limiting processes were inaccurate. This may lead to the appearance of cracks at areas not covered by the installed LUM algorithms. With a classical monitoring system without data storage, assumptions have to be made on the correlation between the life consumption at the newly detected life limiting area and the calculated life consumption at the actually monitored area [7]. In most cases this will require conservative assumptions thus loosing a significant portion of the useable life. The availability of a recorded running history would completely remove the necessity to determine the unknown life consumption by correlations. The stored data can be input into an updated or newly developed algorithm, using the best available temperature, stress and damage models. With highly optimized implementations of the LUM algorithms the recalculation of several 1000 flights for all critical parts of one engine can be accomplished in less than one hour CPU-time on a 1GHz PC. Dependent on the complexity of the applied algorithms the computer time for the recalculation of the complete running history for one critical part could drop to a few minutes.

The benefits of having access to the full running history are so significant, that it seems worthwhile seriously considering the option of replacing or at least enhancing obsolete on-board LUM systems by a suitably designed data recording process. It will be shown, that some of the objections thought to be inhibiting the use of recorded data for LUM are either not substantial or may be overcome by an appropriate design of the data acquisition and recording process.

The main part of this paper will be organized as follows:
First a typical result of an existing on-board monitoring system will be given. Some consequences for the required accuracies and timescales are discussed.

The next paragraph will introduce the types of signals used in typical LUM calculations. Some examples are given showing the non-perfect behavior of signals. The application of low pass filters to counteract noise or imperfections of the data acquisition process is demonstrated. It is shown, that proper preprocessing of the signals may dramatically reduce the storage space requirements.

An outline of the method for conditioning and compressing the data is presented, including analysis of signal properties (e.g. power spectrum of signal autocorrelation), filter selection, optimal quantization, selection of statistical predictors, update of a statistical signal model and finally arithmetic coding.

2. EXAMPLE OF A LIFE USAGE MONITORING RESULT

Life usage cannot be measured exactly. The basic idea is the dependence of the failure probability of a part on the load history of this part. A LUM system tries to calculate the load history for some highly loaded areas (critical areas) of the part using a mathematical thermal and mechanical model. The time-dependent boundary conditions for this model have to be calculated from the time-dependent measurements of engine and aircraft signals. The most important signals are the rotation speeds of the engine spools, the engine inlet conditions (e.g. inlet temperature, pressure), gas temperatures from the compressors and turbines. Typically also flight conditions (speed, altitude), the throttle position and the position of actuators and bleed valves may be needed for thermodynamic calculations.

The result of the life usage calculation at a certain critical area for one flight is finally cast into a single number, the “life consumption”, expressed in units of a reference life consumption of a defined design mission. The life consumptions of all flights seen by a part are added and compared against a predefined limit, corresponding to a given accepted failure probability (the “released life”, again expressed in units of the life usage of a design reference mission).

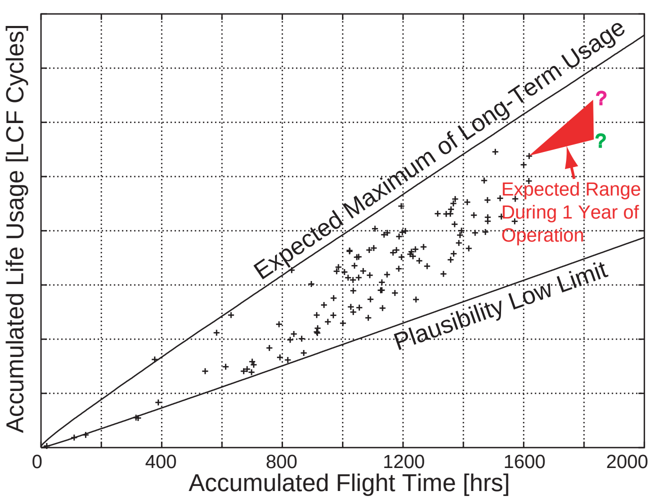

Figure 1: Observed Scatter of LCF Life Usage

The results of this process, applied to a population of parts are shown in Figure 1. The data show an example for a compressor disk. These parts were individually monitored from their introduction into service. After some years of operation the parts have experienced their individual histories leading to some sort of distribution “cloud” in the usage versus flight time diagram.

In the past one of the arguments for the necessity of real time monitoring in on-board systems was the immediate availability of current life consumption figures at the end of each single flight. To assess the technical justification of this requirement, it is useful to look at a selected part in the upper right corner of the cloud. Due to the accumulating nature of the damage process, one further year of usage will bring the part somewhere between the two question marks at the right side of the dark triangle. Without monitoring, the upper value would have to be assumed. Even with this worst-case assumption, there is no technically justified need for an immediate availability of detailed results of a LUM calculation. The decision, where the part eventually will be located in the diagram, can therefore be easily postponed to the end of the year or even to the time, when the worst-case extrapolation intersects the horizontal limit line indicating the released life. In the example shown, this line lies far above the upper boundary of the diagram.

The consequence for a planned replacement of detailed on-board monitoring by a ground based LUM calculation is rather encouraging: If statistical data on the expected scatter bands have been determined [7], than there is no additional risk by postponing the actual determination of life usage for a few month or even a year. It is therefore possible to optimize the download frequency of stored flight data without the need for meeting tight time constraints.

This is further justified by the nature of the underlying fatigue processes that result in scatter bands of life potential that are at least one order of magnitude larger than the difference between life consumption calculated by the most sophisticated available algorithms and a correlation with a non-specific usage parameter during an engine operating period of a few month.

3. EXAMPLES OF MEASURED SIGNALS

Total Inlet Temperature T1

The first example shows the engine total inlet temperature for a mission flown with a jet trainer.

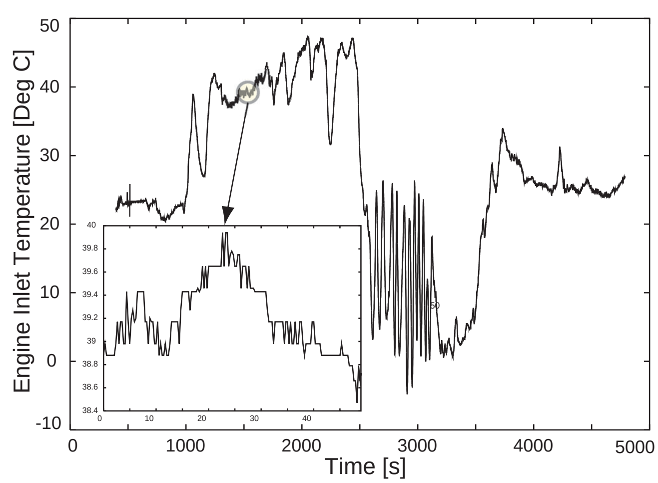

Figure 2 Total Inlet Temperatures for Trainer Mission

The inlet temperature determines the “base level” for the gas temperature calculation in the compressors and the cooling air system. It is influenced by the temperature of the ambient atmosphere and by the speed of the aircraft. Rapid changes of this signal may occur during hot gas ingestion of exhaust plumes or during thrust reverse. The normally expected behavior is quite smooth. Looking at the enlarged part of Figure 2 reveals the presence of superimposed fluctuations with amplitudes of 0.1-0.2K. This fluctuation that is probably caused by noise in the measurement system will have absolutely no influence on the temperature development of disks or even blades, since the time constants involved in heat conduction will be at least a few seconds for blades and minutes for the rotor structures.

The appropriate treatment of this signal to remove the unwanted noise is to apply a low pass filter. There is a vast amount of literature on the topic of filter design. The influence of input signal filtering on the results of LUM calculations has been investigated in [8]. To determine which filter has to be applied to remove noise without unacceptable deformation of the underlying deterministic signal, a spectral analysis of the signal autocorrelation has to be performed. Processes with long-term memory tend to produce spectra with continuously decreasing amplitudes, known as (1/f)-behavior. White noise creates equal amplitudes for all frequencies whereas periodic processes can be easily identified by the associated peaks in the autocorrelation spectrum.

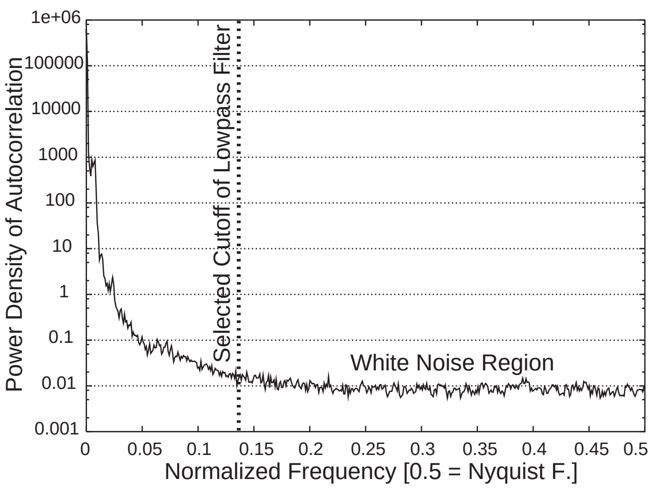

Figure 3 Autocorrelation Spectrum of T1 Signal

The method how to extract meaningful autocorrelation spectra is described in many books on signal processing. The spectra shown here were produced with a program from [9], which uses averaging of overlapping segments together with window functions (Segments of length 512 were used together with Hanning windows).

The spectrum in Figure 3 has two different regions. The low frequency part exhibits the characteristic falling tendency of processes with long memory, whereas the high frequency part has the typical white noise behavior. No periodic components are visible.

A well proven rule for the selection of a de-noising low pass filter is to select its cutoff frequency somewhere in the transition region before the spectrum becomes horizontal, indicating a pure random process. As random data are unpredictable, the random component is not accessible to any data compression technique and will therefore need additional storage space without contributing any useful information.

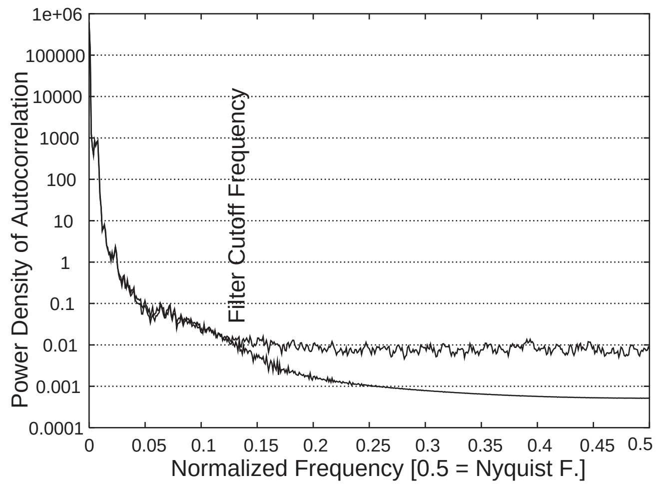

Figure 4 Influence of Butterworth Low pass Filter

Using the filter family discussed in [8], a recursive 4th order Butterworth low pass filter with normalized cutoff frequency 0.13 was applied to the stored raw T1 signal. Figure 4 shows the damping of the white noise part of the spectrum, whereas the low frequency part remains unaffected. Figure 5 below shows the influence in the time domain for the time interval also shown in Figure 2.

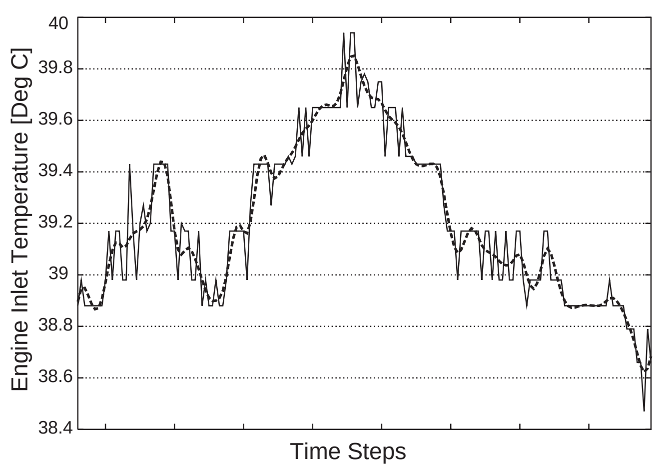

Figure 5 Order 4 Butterworth Filter Applied to T1 Signal

To compensate for the phase shift introduced by the filter the group delay for the frequency 0 was used for a re-synchronization of the raw signal and the filtered signal. The filtered signal now looks very "predictable", thus enabling a much better compression rate for the data storage process.

Spool Speed Signals

Rotational spool speeds are either measured by counting the number of impulses created by a phonic wheel over a short fixed time interval (typically 100 ms) or by directly measuring times for a certain number of impulses. Dependent on the engine design (2 or 3 spools) and on the need for multiplexing counters in the data acquisition system the update rate is limited to 2 or 4 Hz with the first method, whereas higher sampling rates, as needed by a digital engine control unit, are possible with the time measurement technique. The Tornado Data Acquisition Unit (DAU) provides spool speed signals with 2 Hz. This frequency is also used in the LUM algorithms as update rate for the thermal transient models. Unless there are erroneous pulse counts caused by electronic interference, this sort of measurements usually produces very reliable and noise-free signals, with decreasing autocorrelation power density up to the Nyquist limit. The possible change rates are limited by the large mass of the rotors. Any sort of filters will have an influence on the signal dynamics. It has been shown in [8] that the possible gain in data compressibility is inevitably accompanied by a decrease in the accuracy of LUM results with filtered spool speeds. It is therefore advisable to avoid signal filtering for spool speeds measured with low update rates.

The most decisive feature for the high sensitivity of LUM results to signal modifications of rotational spool speeds is their immediate influence on centrifugal stresses that may enter the damage calculation with high exponents. It has been found nearly impossible [8] to find filters that do not have some biased effect on overshoot peaks as shown in the following figure:

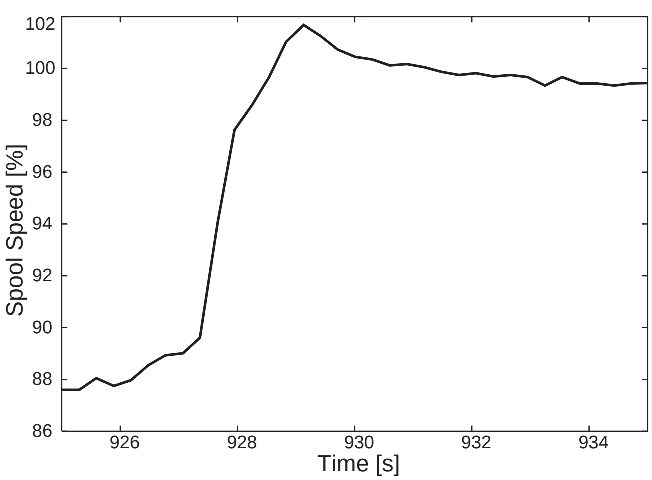

Figure 6 Overshoot of Spool Speed

On the other hand there exists usually no or only negligible noise in counted spool speed signals. The required data volumes to store spool speed data shown later in this presentation were therefore determined without any filtering of the spool speeds.

Problems Arising from Sampling Rate Conversions

If spool speeds or other engine signals are available with higher data rates, as they are required by the engine control system to counteract flow instabilities and to avoid speed and temperature exceedances, there is often no plausible reason to use this high data rate also in the LUM calculations. The appropriate method for a reduction of data rates is to limit the frequency content of the signal sampled at the high rate according to the Nyquist criterion with a suitably designed anti-alias filter before using a low frequency partial sample. If it is possible to define a sampling frequency for the acquisition of data for LUM, a frequency compatible with the update rate of the data source should be selected. To give an example, it is probably better to use a 2.5Hz recording frequency instead of 2Hz, if the control system runs with a cycle of 40ms. The rate conversion techniques, as they are applied in digital audio processing, usually require sophisticated digital filters considered to be too complicated or expensive in a flight data acquisition or monitoring system.

Ignoring these requirements by a simplistic approach –just record the last available sample, irrespective of phase considerations, produces artifacts that render most of the acquired information useless. Sometimes data have already been recorded without consideration of these constraints, as illustrated in the following example from the recorded engine data of flight tests of the Eurofighter. Those data were passed to our life usage monitoring group to assess the accuracy of the results of the on-board engine monitoring box. A spectral analysis of the autocorrelation of one of the spool speeds gave the following result:

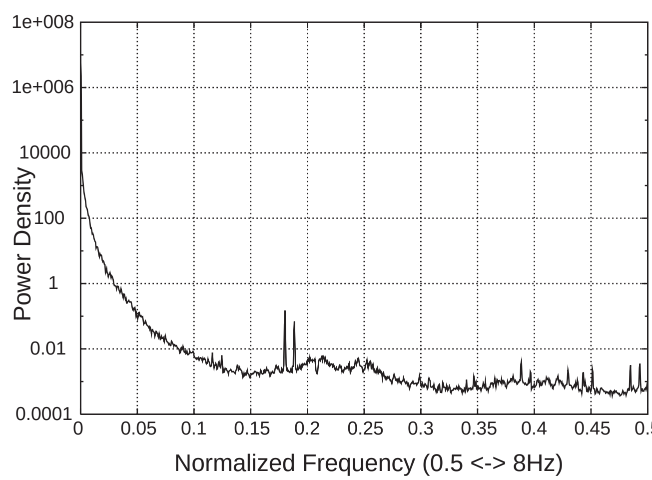

Figure 7 Autocorrelation Spectrum of Spool Speed

Without going into details, the peaks and increased power density level in the normalized frequency range around 0.2 were identified as mixing harmonics of the incompatible sampling rates of recording and data acquisition by the control system. To remove the resulting periodic signal content, a low pass filter with a normalized cutoff frequency near 0.15 was applied to the recorded data.

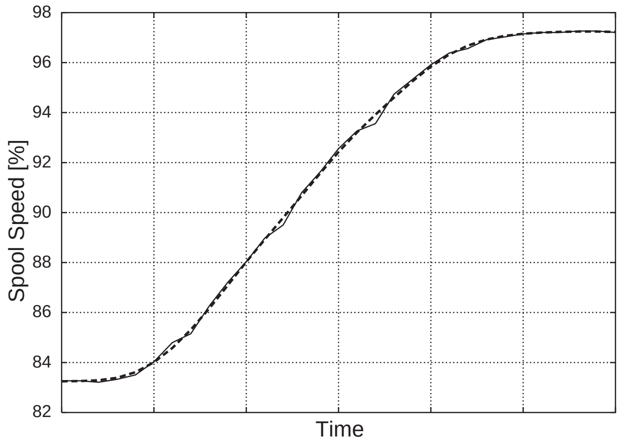

Figure 8 Comparison of Raw and Filtered Spool Speed

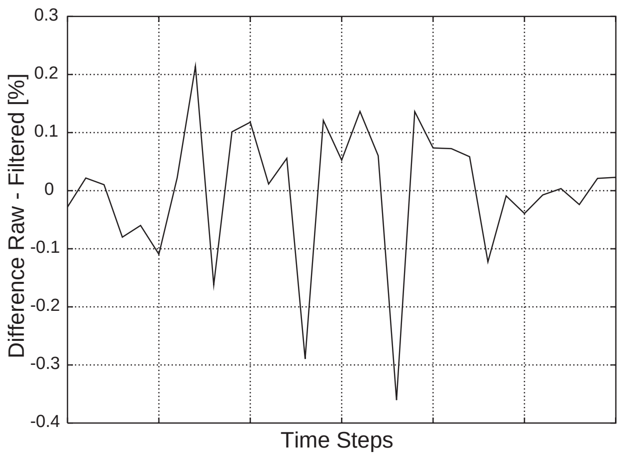

Figure 9 Difference between Raw and Filtered Signal after Re-Synchronization

A comparison of the original and filtered signal after compensation of the phase shift is shown in the previous two figures .The structure of the periodic deviations weakly visible in Figure 8 is better illustrated by the difference plot of Figure 9. Both pictures are also a good example, that a well-designed filter can do a good job in limiting the information content to the physical meaningful portion of a signal. The filtered signal seems to be perfectly sufficient to represent the physical behavior of the rotor, at least in the absence of heavy flow disturbances like compressor surge or of severe mechanical failures. It might therefore be desirable, to store only that amount of information corresponding to the properties of the filtered signal.

Signal with Quantization Noise (Throttle Position)

As a final example a signal is presented, for which the accuracy needs for LUM are rather low. The throttle position itself usually is not an input parameter for the mathematical thermal or mechanical models, but it is sometimes needed to identify certain phases of the engine run. It helps to detect engine start or shutdown, to identify the selection of take-off or combat power or to count idle times before the shut down of the engine.

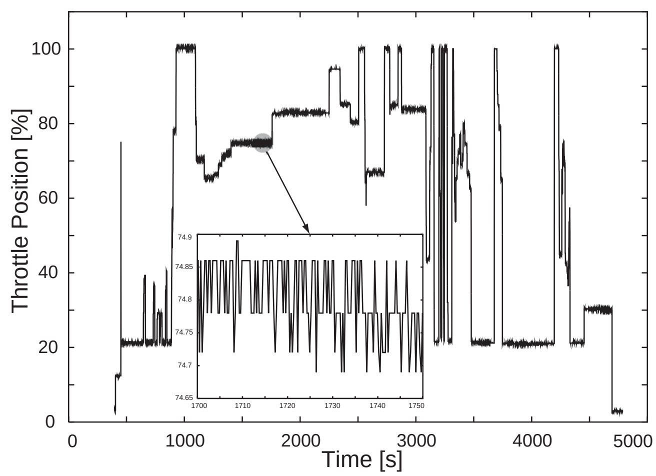

Figure 10 Throttle Position of Trainer Mission

It is therefore acceptable to use a very coarse resolution and to suppress the recording of small changes. The whole useful information in the selective enlargement insert in the following figure can be represented by "Throttle Position = 75% +-1%", ignoring the details obviously only showing the influence of the airframe vibration on the potentiometer pick-up. One has to be careful when applying low pass filters to this sort of signals, because of their smoothing effect on step changes. The detection of transitions in the engine operating mode may be complicated with an aggressively filtered signal.

4. DATA RECORDING

General Purpose Recorders

The current trend in the development of aviation recorders [10] 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 The rapidly decreasing cost of memory does not favor attempts to reduce the storage requirements, as it has been tried in the past [11]. In contrast the application to LUM discussed here only needs a few parameters with relatively low sampling rates.

The data recording process as it is now applied in most aviation data recorders captures a lot of information never needed in a reasonably defined life usage algorithm. The limitation of the recording process to a few parameters with well-known behavior allows using specifically tailored data conditioning to each signal type. The removal of those signal components not carrying useful information and the exploitation of the specific autocorrelation properties of each signal would enable the construction of a highly efficient recording function for the input data for LUM.

To give an outline, each signal would have its own dedicated processing, consisting of:

- o Rate conversion from the sampling rate of the data source with application of anti-alias filters
- o Sampling with the minimum required rate
- o Removing noise with a low pass filter or discarding low amplitude changes
- o Application of a suitable data compression technique

The following paragraph will discuss some aspects of data compression applied to the input data of LUM.

Data Compression Applied to Engine and Aircraft Signals

The field of data compression is one of the biggest in information processing science. Virtually every computer has some compression utilities stored (try to install any operating system without unpacking compressed archives). There are hundreds of compression programs and every square foot in the field of methods is mined with some patents. The majority of programs have been designed for text compression, but most of the methods are applicable to other data as well. Currently the most popular methods are the so-called dictionary coders that look for repeated sequences of text in the uncompressed file. They have their roots in the classical works of Ziv and Lempel in 1977 and 1978 [12]. Nearly everybody involved in the handling of flight data has probably already used one of the popular compression utilities to squeeze a large file with recorded data onto a 1.44MB diskette.

A typical result for an ASCII file with 16316 lines of 15 Integer data per line (converted data for one Tornado flight from the Italian MaRe system) looks like this:

zip –9 –v flight.zip recorded.txt
total bytes 1029232, compressed=147192 -> 86% savings

Even this totally unspecific method is able to pack the data for more than 2 hours of engine running time with two engines into a surprisingly small file. This simple experiment shows, that there obviously a lot of redundancy is present in the data that can be removed without any loss of information. In conjunction with data conditioning there exists a potential for a further reduction of the data volume.

One of the classical methods is coding signal values according to their probability of occurrence. The underlying idea is to use shorter codes for highly probable signals and longer codes for rarely occurring signal values. The best-known technique implementing this concept is Huffman coding [12]. Due to the high variability of military flight profiles most signals will not allow an efficient allocation of shorter codes to certain signal ranges. This is illustrated by the example in the following figure:

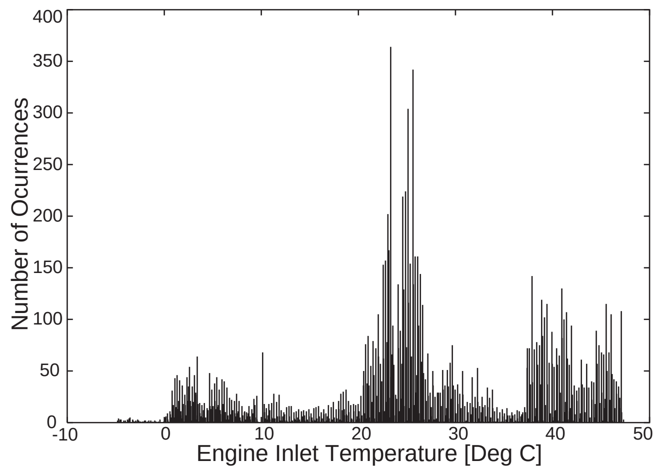

Figure 11 Histogram of T1 Distribution

The distribution is a consequence of the flight profile shown in Figure 2 and may be completely different for another mission.

Delta Coding

Signals with a high degree of short time autocorrelation are candidates for the application of delta coding.

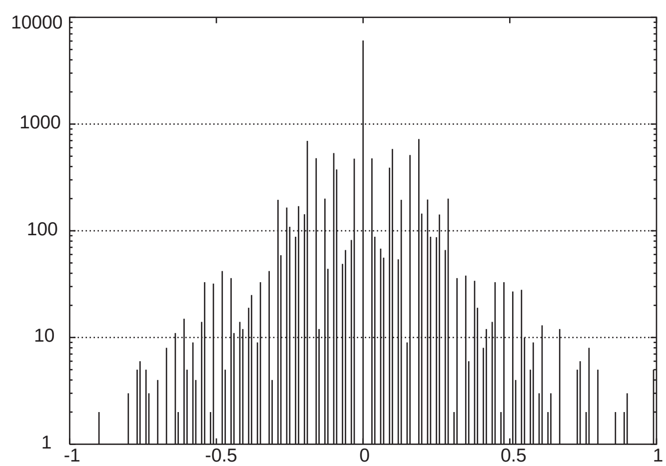

Figure 12 Histogram of Delta Values for Unfiltered T1

Although there is the expected peak at delta=0, the occurrence of gaps and many discontinuities is a consequence of the noise content of this signal. However this distribution is much less dependent on specific mission profiles than the one previously shown.

The situation changes if the noise is removed by the low pass filter and the reduction of the quantization step to 0.02K (which is still far below the resolution needed for the LUM) is performed.

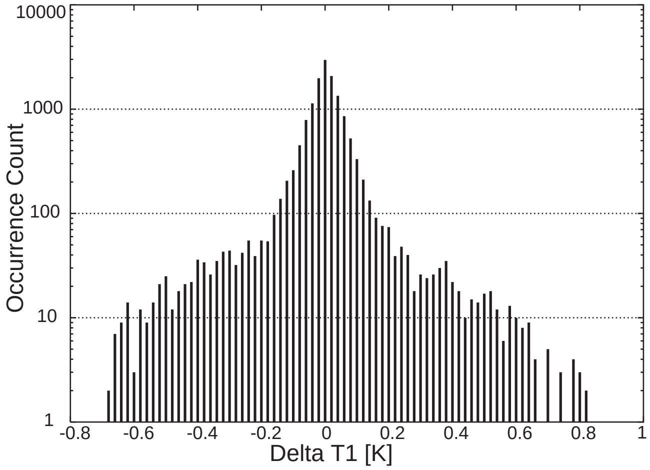

Figure 13 Histogram of delta T1 After Filtering

The information on the unequal probability distribution of occurrences of a change increment "delta" from the preceding signal value can now be used to statically assign code lengths according to its probability p. The optimal code length would be $-\log_2(p)$ where $\log_2$ is the logarithm for the basis 2. Because Huffman codes can only have an integer number of bits, only probabilities 2^{-n} can be exactly represented.

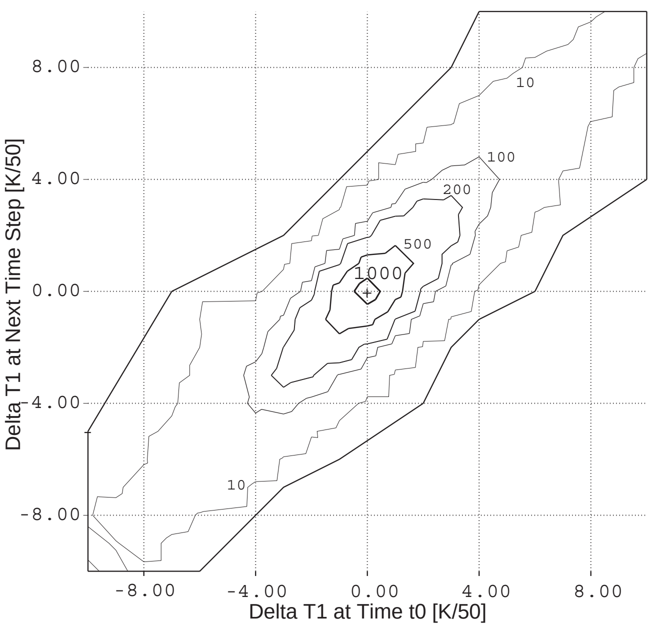

Figure 14 Occurrence Counts of Successive Combinations of Delta

The nature of the physical process creating the data and the smoothing effect of the low pass filter do not only create a statistically predictable distribution of changes "delta" from the preceding measurement (Order-0 statistics), but also correlations between successive delta values (Order-1 correlation), as shown in Figure 14. The alignment of the contour plot of occurrence counts along the diagonal simply represents the fact that a continuation of the signal with the current change rate has the highest probability.

It is even possible to consider higher order models. To give an example, Order-2 models represent the information: "What is the probability distribution of the occurrence of a certain delta_3 after a combination (delta_1, delta_2) has been encountered?" In the investigation of engine signals only models up to order 2 were found to be optimal.

Whereas Huffman codes can only use the fixed values ½, ¼, 1/8, 1/16, …of the probability, a method to represent arbitrary probabilities, known as "Arithmetic Coding", first mentioned in a conference presentation in 1979 [13] is available. With arithmetic coding, it is possible to use exactly the actual probability to create a reconstructable encoding. Arithmetic coding is now used in most of the current state-of-the-art data compression methods [14, 12]. The required storage space is then given by the sum of the negative binary logarithms of the probability decisions over all time steps.

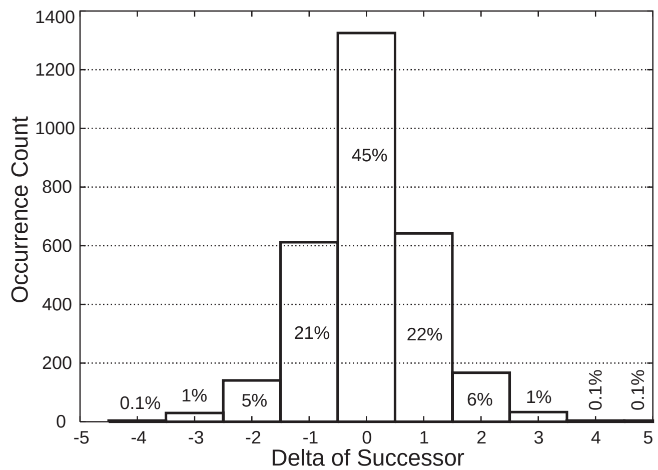

Figure 15 Histogram of Successor Counts for Delta=0

An example helps to understand this process. Figure 15 shows the accumulated counts of delta values that have occurred in the time step following a time step with delta=0. The physical delta value was 0.02K. The figure can be regarded as a cut through the distribution shown in Figure 14 at deltaT1 (t=t0). For simplicity the distribution at the end of the flight is shown. Due the application of the low pass filter no other successors than those shown in this figure occurred during the whole flight. The actual method has to build this distribution by gathering the histogram from the data seen before the current time step. Nearly half of the successors (45%) had again the value delta=0, 1/5 had abs(delta)=1 etc. With Huffman coding the 45% value would be represented by ½, the 20% and 21% values by ¼, the 5% and 6% values by 1/32, 1% by 1/128 etc. Coding a successor value of 3 would therefore require 7 bits, (ignoring the necessity for additional prefix information to guarantee unambiguous decoding).

With arithmetic coding a successor 0 uses exactly $-\log_2(0.45) = 1.152$ bits, a successor 1 would require $-\log_2(0.22) = 2.184$ bits, etc.

The good predictability of data during stable flight conditions and engine operation is a pre-requisite for efficient data compression. During transient operation the uncertainty of predictions is much higher. As a consequence the average required storage space over a whole flight lies somewhere between the extremely low values for stable operation and the maximum that occurs when no prediction is available. For the example just discussed the storage space for the whole flight was 2.6 bits per time step.

Compression rates near the limit posed by information theory may be attained only, if each signal has its own specific processing. The arithmetic coding process internally accumulates information in nearly arbitrary small increments that can be much smaller than one bit, dependent on how well the statistical signal prediction model and the actual signal coincide.

Example: Data Rates for Spool Speed Signals.

To gain some experience the process outlined above has been applied to a lot of available recordings of flight data. To avoid the expensive tailoring of the coding process to each specific signal type, a general purpose implementation of statistical modeling plus arithmetic coding ("ARITH-N"), taken from the book [12] was used. The main difference between an implementation specific for a certain signal type and the general purpose program is the replacement of dynamically allocated context tables in the general purpose program by fixed size tables in the tailored program (as proposed in [15]) This replacement removes a lot of software complexity and gives static figures for the memory allocation in a potential airborne recording system, which is also a requirement for a formal software qualification. The differences in observed compression rates between both implementations have been found to be negligible.

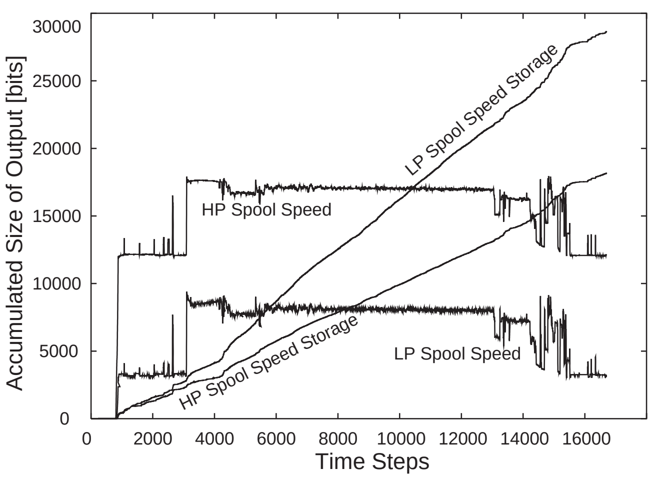

Figure 16 Data Storage for Spool Speed Signals (Ferry Flight)

Figure 16 and Figure 17 show the development of accumulated storage sizes required to store the spool speed signals of two realistic Tornado missions (a ferry flight and a high speed patrol flight).

In contrast to an intuitive assumption, there is no significant difference in the data volume produced by the ferry flight and the patrol mission containing a considerably higher amount of manual throttle movement. The reason for the sustained rate of information flow during the visually near constant spool speed phase of the ferry flight is the engagement of the automatic thrust control function ("auto-throttle") that tries to maintain a constant cruising speed by small modulations of the engine thrust. The main need for thrust adjustment is caused by atmospheric turbulence, thus requiring random-looking changes of the engine thrust settings. The recording of the corresponding time history of the spool speeds during this phase produces approximately 2 bits of data for the LP spool and 1 bit per time step for the HP spool. Both signals have a resolution of 10 bits.

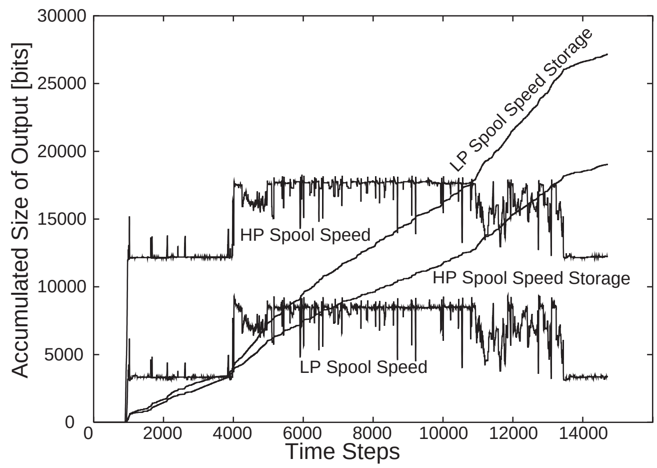

Figure 17 Data Storage for Patrol Flight

Although the variability in data accumulation rates shown in Figure 17 is greater than in the previous example, the finally achieved compression ratios are quite similar. The LP spool speed signal requires an average 1.75 bits per time step; the HP spool speed signal can be stored in 1.15 bits per time step.

Estimation of Storage Requirements for Other Signals

A simple technique has been found to perform a rough estimate of the storage requirements for arbitrary signals without the need to apply the statistical modeling and arithmetic coding. The main steps of the method are:

- Determine a scaling of the signal that limits the maximum delta per time step to the range +-127
- Apply low-pass filters to cut off noise or periodic components, if necessary
- Write sequence of delta values to a binary file. Due to the range limitation one byte per time step will be sufficient
- Apply the standard "ZIP" compression with the "maximum compression" switch to this file
- The expected size with an optimal statistical adaptive arithmetic coder will be around 75% of the size of the ZIP file.

Example: Data Rates for the RB199 Engine

To estimate the storage requirements for retaining the full running history of all engines of an aircraft fleet, the input data of the engine monitoring part of OLMOS [3] were analyzed using available recorded flights from the GAF and IAF. The following average data rates (Bytes/hour) per engine running time were found, based on the 10bit resolution of the Italian MaRe system.

Airspeed:	732
Altitude:	344
Inlet Temperature:	195
Pilot's Lever:	1543
LP Spool Speed:	2633
HP Spool Speed:	2165
Turbine Blade Temp.:	2169

The total amount of data per hour is less than 10kByte. One MByte of on-board non-volatile memory could store as much as 100 hours of engine operation. A fleet of 300 aircraft each flying 200 hours per year would produce 1.2 GByte of recorded compressed engine data.

Logistic Requirements

The introduction of a fleet-wide recording process would of course require careful housekeeping of data. Housekeeping may be simplified if data downloads are synchronized with configuration changes of the engine. The recording part of the on-board system should not need engine specific data or data that are dependent on the engine configuration.

Expected memory requirements for other engine types are expected to be in the order of 10 to 40kByte per flight hour. This will allow integrating the recording function into the control system or into existing monitoring systems without the need for external storage devices. Download frequencies could be as low as once every 100 hours of flight time, thus providing many workable options for the logistic handling of the data. An optimal solution would be to store the full running history of all engines in the fleet in a centralized database.

5. CONCLUSIONS

It has been shown, that a combination of data conditioning and data compression of flight and engine data can be used to reduce the required storage space in an on-board system to figures that allow storing all data required for individual life usage monitoring over operation times of at least several months. The logistic burden for data downloads can thus be minimized. The downloaded data have to be stored over the whole running history of an engine, with appropriate data bases to maintain the links between individual fracture critical engine components and their associated flight and configuration data. The introduction of such a system will require considerable investments into hardware, data links and data storage.

The greatest advantage of the proposed approach over classical on-board LUM systems is the ability to reassess component lives with the best available algorithms instead of being forced to make conservative assumptions, when new information on life limitations of components invalidates the usage figures accumulated with models not covering the newly detected damage mechanisms or critical areas.

The proposed method is a candidate to be used in monitoring systems for new engine projects. Plans for a replacement of obsolete on-board LUM systems for existing old engines have to consider the problem how to treat the missing running history of those engines.

6. REFERENCES

[1] SAE: Guide to life usage monitoring and parts management for aircraft gas turbine engines.
Aerospace Information Report AIR1872, Rev. A, 1998

[2] RTO AVT Panel Task Group AVT-017:
Recommended Practices for Monitoring Gas Turbine Engine Life Consumption.
RTO-TR-28 AC/323(AVT)TP/22, RTO/NATO April 2000.

[AVT01] NATO RTO Applied Vehicle Technology Panel (AVT): Symposium on Aging Mechanisms and Control. *Part B - Monitoring and Management of Gas Turbine Fleets for Extended Life and Reduced Costs*. Manchester, UK, 8-11 October 2001.

[3] H.Pfoertner: Long Term Operation and Maintenance of Engine Monitoring Systems - Recommendations Derived from 15 Years of OLMOS Use by the GAF. In [AVT01].

[4] 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, WA, 1997

[5] F. Genot: Cost Reduction and Engine Life Extension through Engine Life Monitoring at SNECMA.
In: [AVT01].

[6] R.L.Kiang: Critical Part Life Extension Efforts in a Military Engine. In [AVT01]

[7] H.Pfoertner, J.Broede: Statistical correlations for analysis and prediction of turbine engine component life usage. *37th AIAA/ASME/SAE/ASEE Joint Propulsion Conference,* Salt Lake City, Utah, 8-11 July 2001. Paper AIAA 2001-3754.

[8] H. Pfoertner, S. Graf: Data Conditioning Methods for Aero Engine Signals and their Influence on the Accuracy of

Fatigue Life Usage Monitoring Results. In: *Proceedings of the 20th Symposium Aircraft Integrated Monitoring Systems* (AIMS2000), Garmisch-Partenkirchen, 2000, pp. 157-176.

[9] S.D.Stearns, R.A.David: *Signal Processing Algorithms using Fortran and C.* Prentice Hall, Englewood Cliffs, N.J., 1993.

[10] D.R.Grossi: Aviation Recorder Overview. *International Symposium On Transportation Recorders,* Arlington, Virginia, May 3-5, 1999.

[11] M.E.Sigrist, A.N.Muggli, SWISSAIR Studies on the Compressibility of Aircraft Data.
15th Symposium AIMS, Aachen 1989

[12] M.Nelson, J.-L.Gailly: *The Data Compression Book*, 2nd Edition. M&T Books, New York, 1996.

[13] G.N.N. Martin: Range Encoding: An Algorithm for Removing Redundancy from a Digitised Message. *Video & Data Recording Conference*, IBM Scientific Center, Southampton, July 24-27, 1979

[14] T.C.Bell, J.G.Cleary, I.H.Witten: *Text Compression*. Prentice Hall, Englewood Cliffs, N.J., 1990.

[15] D.A. Lelewer, D.S. Hirschberg: Streamlining context models for data compression. *Proc. IEEE Data Compression Conference*, Snowbird UT, IEEE (1991) 313-322.

Hugo Pfoertner is a R&D engineer in the mechanical engineering department of MTU Aero Engines GmbH. He studied physics at the University of Technology Munich from 1968-1973. He received the degree of a "Diplom-Ingenieur". From 1975-1982 he was research assistant at the Laboratory for Hydraulic Machinery at the same university. Research work in this time was focused on fluid dynamics and the development of 3D Finite Element codes for the flow fields in hydraulic pump and turbine impellers. He was also engaged in the design of methods for the analysis of turbomachinery test bed data. In 1983 he joined MTU, where he first worked in the department for computational methods. One of his tasks was the design of robust numerical optimization methods. He is now expert adviser for engine monitoring and software in MTU's mechanical engineering department.

He was team leader of the development of the engine monitoring system for the RB199 in the OLMOS system of the Tornado and is also responsible for the development of the software for the life usage monitoring system of the MTR390 helicopter engine for the new German / French Tiger helicopter. He also worked in the development team of the EJ200 engine monitoring unit.

Main working fields are design of systems for life usage monitoring, software design for on-board systems and development of tools for flight data analysis.

In 1998 Hugo has started research into alternative monitoring methods that avoid the configuration problems of the existing on-board systems. He is now working on a prototype of a combined monitoring / recording system using highly efficient data compression methods for the engine and aircraft data.