SMCia/05
2005 IEEE Mid-Summer Workshop on Soft Computing in Industrial Applications
Helsinki University of Technology, Espoo, Finland, June 28-30, 2005, pp. 1-5, 2005

Using Neural Networks and the Rank Permutation Transformation to Detect Abnormal Conditions in Aircraft Engines

Neil H. Eklund and Kai F. Goebel
Computing and Decision Sciences
General Electric Global Research
One Research Circle, Niskayuna, NY 12309 USA
Phone: 518.387.5499, Fax: 518.387.6104, E-mail: {eklund, goebelk}@research.ge.com

Abstract - Real world aircraft engine (gas turbine) data are contaminated with substantial noise and outliers. The rank permutation transformation (RPT), founded in some early ideas in statistics, is proposed as a way to both diminish the effect of noise and outliers, and to facilitate classification by making statistically unlikely events more pronounced. The RPT is also found to improve the performance of neural networks used for fault detection and classification considerably. Results from both real engine monitoring data for abnormal condition detection and high-fidelity simulation data for on-wing fault detection and diagnosis are presented.

I. Introduction

The problem of detecting safety critical abnormal conditions and faults in aircraft engines (in this paper, gas turbines) for practical fault accommodation or health prognosis applications is challenging because real world aircraft engine data are contaminated with noise from several sources in addition to the normally expected sensor noise. Slight variations in the manufacturing process result in engine to engine variation in performance. Engine components deteriorate at different rates and are serviced at different intervals, producing additional variation in performance. Finally, engine performance is dramatically different in different regions of the flight envelope; while an attempt to correct flight envelope effects is usually made, it is not perfect, introducing further variation. Hence, real world aircraft engine data typically contain outliers, are non-normally distributed, and are quite noisy.

There are application specific issues that further confound the detection problem. Implementing automatic fault accommodation necessitates both that certain faults are detected essentially in real time (e.g., within 106 ms of fault occurrence at a 15 ms sampling rate) and a near zero false alarm rate to avoid taking remedial action when no fault exists [1,2]. Alternatively, for some remote monitoring and prognostic applications, the data are much more sparse (typically one or two snapshots per flight), comparatively more noisy, and it is still necessary to detect the onset of the fault quickly (relative to the number of data) and localize the start of the problem to accurately estimate remaining engine life in time for safe and economic remediation.

This paper presents a novel data transformation that can be used on its own as a trigger for further investigation (e.g., in remote monitoring), or in conjunction with various classification techniques for on-wing diagnosis. We propose the use of the rank permutation transformation to move from raw and poorly behaved feature space to a feature space that more closely represents the probability of occurrence. In this feature space, the data are much better behaved, and faults are much easier to detect.

II. Rank Permutation Transformation

The raw feature space (i.e., using the measurements without further processing) is seldom the best (or even a good) place to attempt classification; even simple transformations of the feature space (e.g., scaling each feature by its standard deviation) can boost classification accuracy significantly [3]. The general approach for fault detection in a time series (for either accommodation or prognostic applications) used here is to compare a test statistic for the "current" data (a few recent data) to the "past" data (some typically larger number of points prior to "current"), possibly with a buffer of unused points in between to make differences more pronounced (Fig. 1).

Using ranks rather than absolute values solves a number of problems. First, the problematic effect of outliers is vastly diminished [4, 5, 6, 7]. For example, consider the two sets of data, {1 3 5 7 99} and {2 4 6 8 10}; in the first set, the 99 is an outlier. The means of the sets are 23 and 6, respectively; the mean ranks (combine the sets, sort least to greatest, assign the first value a rank of one, the second value a rank of two, etc., then separate the ranks according to the original sets and compute their means) are 5.2 and 5.8, respectively, a much smaller difference and in a different direction. The rank transformation is employed to great advantage in nonparametric statistics

[5, 7]. The rank distribution for a given number of data can be calculated in advance, making implementation at run time very fast (an important consideration for on-wing applications, which have little computational power to spare). The cost of using ranks is slight; e.g., the Mann-Whitney U-test requires only ~3.5% more data than the t-test for equivalent power with normally distributed data.

The idea of using random permutations of the data to develop an exact or near exact probability of occurrence was proposed by Fisher [8]. Unfortunately for Fisher the computing power required for any but the simplest of problems would not be available for many years. However today the computing power required to calculate near exact probabilities for an instance of data is cheap and common, available on almost every researcher's desktop. The principal is illustrated in Fig. 2. The top set of axes show the original data. The hypothesis to be tested is whether the last five data (diamonds) are drawn from the same distribution as the previous data (dots). The null hypothesis is that they are not. If the null hypothesis is true, then a statistic (say, the mean) calculated for the diamond data should be about the same as that statistic for any five points randomly selected from all of the data (both diamond and dot). If we keep the data the same, and randomly permute the labels (diamond or dot), and calculate the "mean of diamond samples" statistic many times, we get the distribution in Fig. 3. This result suggests that any five points randomly selected from all data (both diamond and dot) will have a mean as great or greater than the original five (represented by the dashed line in Fig. 3) only 7.2% of the time. Similarly, any set of data can be compared to another set (e.g., using the current vs. past approach outlined above), and an exact (to an arbitrary number of significant figures) probability that the sets differ on any test statistic can be calculated.

Putting both ideas (using rank rather than raw data and permutation distributions), the "rank permutation transformation" (RPT) can be used to transform raw, poorly behaved time series features into features that closely represent exact probabilities of occurrence (with the assumption that the error across features is uncorrelated). To calculate RPT, one must first define a small number of points to be the "current" set and a (typically larger) number of points to be the "past" set. The two sets of data are first concatenated and the ranks of the combined data are calculated. The sum of the ranks for the original ordering (the test statistic) for the "current" data is calculated. Next, the data labels (current/past) are randomly permuted, and the test statistic is calculated; this step is repeated many times (e.g., 5,000). The value of the test statistic for the original ordering of the data is compared to the values generated via the permutations, and the probability of the original ordering occurring by chance is calculated. This probability is normalized (by subtracting it from 1 if it is greater than 0.5) and the absolute value of the $\log_{10}$

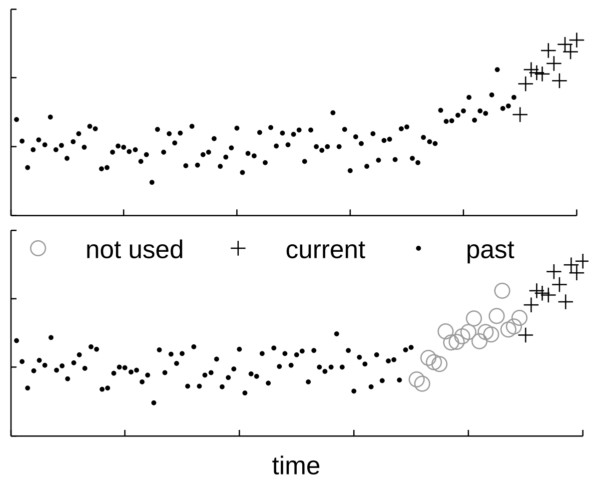

Fig. 1. The general approach to time series fault detection: compare "current" points to "past". On the top set of axes, all data is used; on the bottom, a small buffer is used to make differences more pronounced.

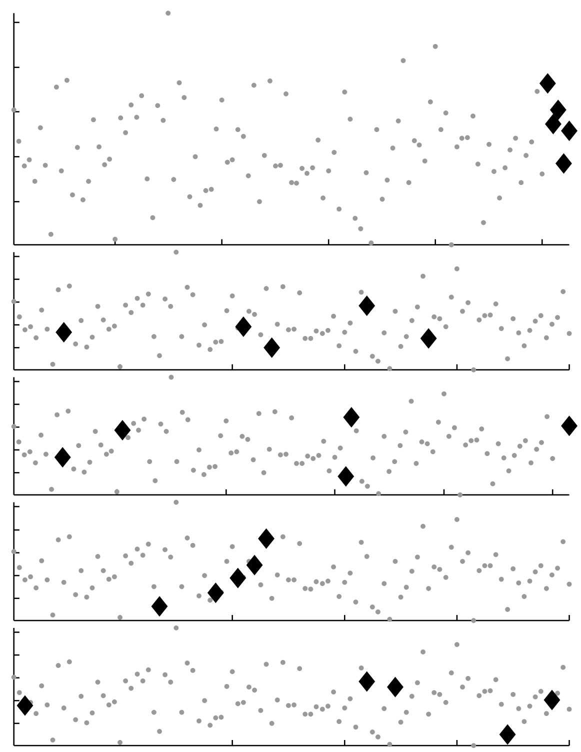

Fig. 2. The top set of axes shoes the original data; the lower four show different possible permutations.

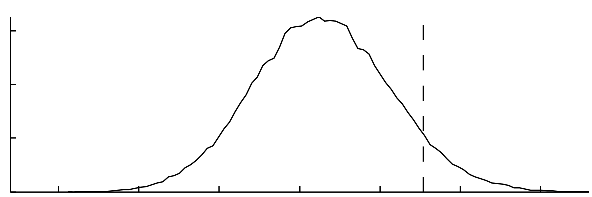

Fig. 3. The distribution of the test statistic for 100,000 permutations (solid line), and for the original ordering (dashed line).

probability is taken to emphasize rare events, and values greater than 10 (and $\log_{10}(0)$) are truncated to 10 (this value is arbitrary, depending on the application – for this particular application, the difference between one in 10 billion and one in a quadrillion is meaningless). Finally, events occurring nearer the lower tail are assigned a negative sign to make fault detection more straightforward (indicating not just rarity but also direction), the value of RPT is assigned to the most recent point of the current set and the counter for position in the time series in incremented by one. The algorithm for calculating RPT is outlined in Table I (using MATLAB® notation).

III. Results

A. Abnormal Condition Detection Application

In a remote monitoring and as a precursor for a prognostic application, one primary goal is to detect abnormal conditions; further diagnostics can then identify the root cause of a fault. In addition, a prognostic module can be kicked off at that time to estimate remaining component life. For aircraft engines, remote monitoring data may consist of snapshots at takeoff and at cruise for each flight. Because many faults affect engine exhaust gas temperature (EGT), it is closely monitored as a prime parameter indicative of engine health.

Figure 4 is an example of actual aircraft EGT data used for remote monitoring representing 415 flights over 91 days, with lower values being better. The upper set of axes is the raw data, and the lower set of axes is the RPT value calculated by comparing the latest five samples to the previous t-16 to t-35 samples (i.e., the previous 30 samples, with a 10 sample offset). An alarm is triggered whenever the probability of occurrence by chance drops below 10^{-5}. At around days 35 and 51 obvious maintenance events occur (EGT does not usually improve dramatically on its own), which are detected. At around day 89 a fault occurs, which is also detected (in this case, the fault was detected six flights sooner than the old rule-based fault detection mechanism). In this application, the RPT values are relatively straightforward to interpret manually, or can be easily automated, e.g., using fuzzy logic to quickly develop a rule set for interpreting alarms.

B. On-Wing Diagnostics Application

In the multivariate case, we have the advantage of distinct patterns that may signal not only the presence of an abnormal condition, but also the root cause of the fault. To that end, we transform all of the variables using the RPT, and perform both detection and classification in the transformed feature space.

Fault detection typically assumes a detectable sensor signature and sufficient time to come up with a reliable decision. However, faults that affect aircraft safety must

TABLE I
RPT IMPLEMENTATION DETAILS

```
a = "current" set;
b = "past" set;
na = length(a);
% concatenate the two sets
c = [a b];
% calculate value of the test statistic
% (sum of ranks of sample a) for the
% original labeling of the data
r = rank(c);
test_stat(1)=sum(r(1:na));
% calculate test statistic for
% many permutations (nperm) of the data
for i=2:nperm
    % randomly permute data
    perm = permute(c);
    % convert to rank
    rp = rank(perm);
    % calculate value of the test
    % statistic for the permuted data
    test_stat(i) = sum(rp(1:na));
end
% calculate probability that value of
% the test statistic for the original
% ordering of the data will occur
% by chance
p=sum(test_stat(1)<=test_stat)/neprm
% normalize
sign = 1;
if p >.5
    p=1-p;
    sign=-1;
end
RPT = sign * min(abs(log10(p+10^-30)), 10);
```

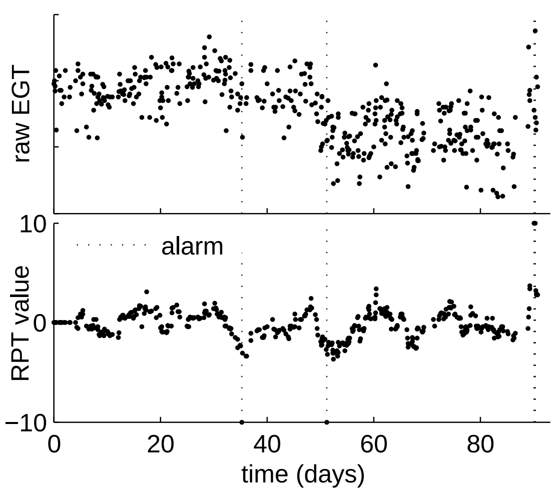

Fig. 4. RPT value for EGT data, and alarm points.

be detected quickly. Otherwise, undesired consequences may arise such as engine surge/stall events, severe vibrations, power loss, or thrust response inconsistent with commanded power. These events in addition to inappropriate crew response may lead to accidents [9]. Because air traffic is projected to increase in the long term, and at the current level of reliability, the number of accidents will increase as well. To reduce this projected number of accidents NASA established the Aviation Safety Program [10]. This section employs data developed at General Electric under the Aviation Safety Program. A brief description is given here, but details of how the data were generated can be found in [1] and [2]. However, in contrast to this earlier work, the results presented here do *not* use or require an extended Kalman filter (EKF) or component level model (CLM), resulting in *substantially* reduced computing power requirement, which is an important criterion for on-wing applications.

For this study a high fidelity physics-based nonlinear model of an advanced commercial high-bypass twin-spool turbofan engine was used. Data were generated at 2047 points distributed throughout the flight envelope. In order to represent more realistically engine behavior sources of variation are included as inputs to the simulations (a unique random combination for each simulation). Specifically, we considered manufacturing variability between engines (modeled by adding variation to the efficiency adder, and flow scalar parameters on the rotating components), deterioration as engines age (modeled by varying rotating component efficiencies and flow parameters, as well as leakage and clearances), and variability in the measurement processes (sensor noise, sensor bias).

The simulated fault types were chosen based on an investigation of an engine events database of the past 20 years digitally controlled engines. Faults from actuators, the engine, and sensors were used; at each point in the flight envelope, data were modeled for four faults, plus the non-faulted engine. The output of the simulation output was the nine sensed values available for diagnosis representing temperatures, pressures, and rotating component speeds. Of these nine, seven were independent of flight envelope position (two were ambient temperature and pressure).

As an example, we demonstrate results on the variable stator vane (VSV, an actuator) fault. In order to allow time for accommodation of a VSV fault, it must be detected within 106 ms (seven data samples after onset of the fault). The seven sensors used for fault detection were transformed to RPT space. Fig. 5 shows a random sample (in the flight envelope) of data. Flight envelope effects and noise obscure the fault in the raw data. Meanwhile, in the RPT transformed data, the fault becomes distinguishable within a few time steps.

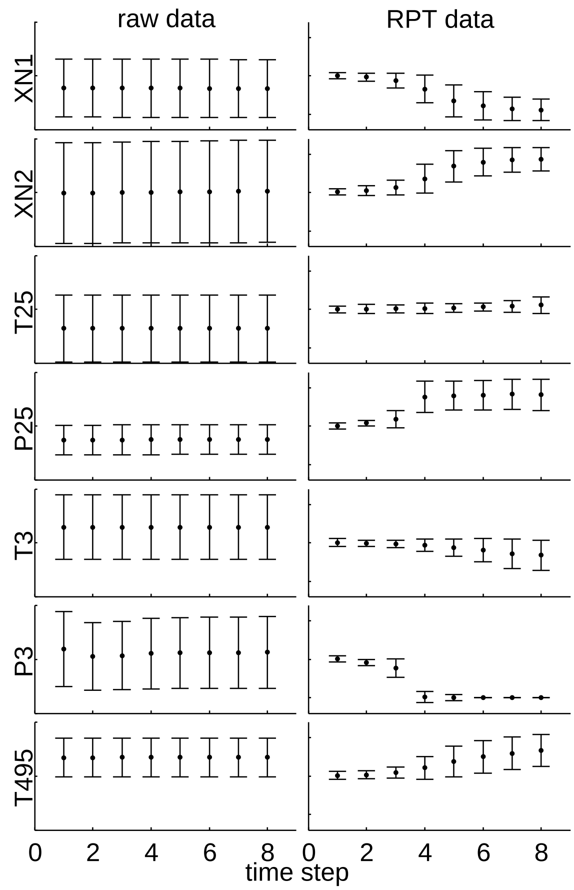

Fig. 5. Mean (± 1 standard deviation) of a 243 randomly sampled cases of VSV fault for the raw (left column) and RPT transformed (right column) data for seven sensed variables. Fault occurs at the first time step, and must be detected by the 7th time step.

The RPT data were rescaled to range from –1 to 1 and a seven input, one hidden layer with seven nodes, and one output neural network (NN) was trained (in a "all other classes and nominal vs. VSV fault" manner) using distinct training, test, and validation sets and five-fold cross validation. The neural network classifier trained on the RPT features performs better than the various transformation (z-transformation, elbow transformation, and others) and classification schemes (neural network, support vector machine, classification and regression trees, and others) reported in [11] and [2]. RPT-NN classifier results are presented in Fig. 6 along with results from a hand tuned (a very time consuming process) rule-based classifier that uses data that has the advantage of using the EKF and CLM described in [2] and a different transformation of the features. The rule-based classifier was previously the most accurate classification scheme tested on these data. The RPT classifier is able to classify 100% of the VSV faults in time for accommodation with no false positives. This increase in accuracy comes at a substantial reduction in on-wing computational power requirements: the permutation distribution can be precomputed, so the RPT transformation is calculated

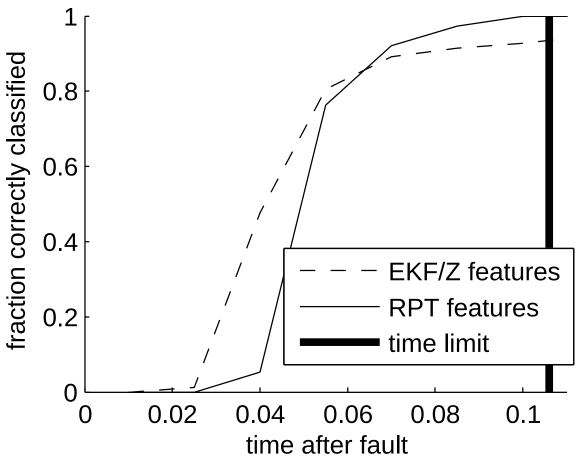

Fig. 6. Fraction VSV faults correctly classified for a neural network classifier trained using RPT transformed features, and the best previous classifier, a hand tuned rule-based classifier that required computationally intensive additional data processing.

quite quickly, and the approach outlined here does not require the enormous computational overhead of a CLM and EKF [2], just raw sensed data.

IV. Summary

The problem of detecting subtle changes in a time series that is noisy, non-normally distributed, and outlier ridden is quite difficult. A variety of methods for dealing with these problems has been tried for both fault detection and engine trend monitoring. The rank permutation transformation is one of the most promising techniques, which combines some simple statistical ideas and modern computing power to overcome many of the worst features of real world data. In addition to diminishing the effect of noise and outliers, the RPT can be used to facilitate classification by making events that are statistically improbable more pronounced. In the on-wing diagnostics application, we show how using the RPT in conjunction with neural networks can result in both a considerable detection and classification performance improvement, along with a substantial reduction in run-time computational overhead.

Acknowledgment

This paper uses data that were generated under NASA grant NAS3-01135 Task # 3. The authors greatly acknowledge the support of Don Simon and Tak Kobayashi (both with NASA).

References

[1] R. Rausch, D. Viassolo, A. Kumar, K. Goebel, N. Eklund, B. Brunell, and P. Bonanni. “Towards in-flight detection and accommodation of faults in aircraft engines.” in *AIAA 1st Intelligent Systems Technical Conference*; pp. 1-11., Chicago, IL; USA; 20-22 Sept. 2004.

[2] K. Goebel, N. Eklund, and B. Brunell. “Rapid detection of faults for safety critical aircraft operation.” in *Proceedings of the 2004 IEEE Aerospace Conference*, pp 3372-3383, Big Sky, Montana, March 6-14, 2004.

[3] R. Duda, P. Hart, and D. Stork. *Pattern Classification*, 2nd ed., John Wiley & Sons, Inc.: New York, 2001.

[4] P. Good. *Permutation Tests: A Practical Guide to Resampling Methods for Testing Hypotheses*, Springer-Verlag, New York, 1993.

[5] H. Mann, and D. Whitney. “*On a test of whether one of two random variables is stochastically larger than the other.*”, The Annals of Mathematical Statistics, 18(1), pp. 50-60, 1947.

[6] F. Mosteller, and J. Tukey. *Data Analysis and Regression: A Second Course in Statistics*. AddisonWesley, Reading, MA., 1977.

[7] F. Wilcoxon. “Individual comparisons by ranking methods.” *Biometrics*,1, pp. 80-83, 1945.

[8] R. Fisher. “The logic of inductive inference.” *Journal of the Royal Statistical Society*, 98, pp.39-54, 1935.

[9] G .Sallee, and D. Gibbons. *AIA/AECMA Project Report on Propulsion System Malfunction Plus Inappropriate Crew Response (PSM+IC), Volume 1*, Aerospace Industries Association and the European Association of Aerospace Industries, 1998.

[10] C. Belcastro, and C. Belcastro. “Application of failure detection, identification, and accommodation methods for improved aircraft safety.” *American Control Conference Proceedings*,. 4, pp. 2623 – 2624, 2001.

[11] B. Brunell, P. Bonanni, N. Eklund, K. Goebel, A. Kumar, R. Rausch, and D. Viassolo. *AvSP - MBFTC Quarterly Report January 20, 2004*, GE Global Research, Nisakyuna, NY, 2004.