Classification of Aircraft Maneuvers for Fault Detection

Nikunj C. Oza, Irem Y. Tumer, Kagan Tumer, and Edward M. Huff

Computational Sciences Division
NASA Ames Research Center
Mail Stop 269-3
Moffett Field, CA 94035-1000
{oza,itumer,kagan,huff}@email.arc.nasa.gov

Abstract. Automated fault detection is an increasingly important problem in aircraft maintenance and operation. Standard methods of fault detection assume the availability of either data produced during all possible faulty operation modes or a clearly-defined means to determine whether the data provide a reasonable match to known examples of proper operation. In the domain of fault detection in aircraft, the first assumption is unreasonable and the second is difficult to determine. We envision a system for online fault detection in aircraft, one part of which is a classifier that predicts the maneuver being performed by the aircraft as a function of vibration data and other available data. To develop such a system, we use flight data collected under a controlled test environment, subject to many sources of variability. We explain where our classifier fits into the envisioned fault detection system as well as experiments showing the promise of this classification subsystem.

1 Introduction

A critical aspect of the operation and maintenance of aircraft is detecting problems in their operation when they occur in flight. This allows maintenance and flight crews to fix problems before they become severe and lead to significant aircraft damage or even a crash. Fault detection systems designed for this purpose are becoming a standard requirement in most aircraft [2, 7]. However, most systems produce too many false alarms, mainly due to an inability to compare real behavior with modeled behavior, making their reliability questionable in practice [6]. Other systems require a clearly-defined means to determine whether the data provide a reasonable match to known examples of proper operation or assume the availability of data produced during all possible faulty operation modes [2, 3, 7]. Because of the highly safety-critical nature of the aircraft domain application, most fault detection systems are faced with the task of functioning for systems for which fault data are non-existent. Models are typically used to predict the effect of damage and failures on otherwise healthy (baseline) data [4, 6]. However, while models are a necessary first start, the modeled system response often does not take operational variability into account, resulting in high

Table 1. Conceptual open loop model illustrating assumed causal relationships.

Flight $\longrightarrow$ Maneuver (M)	Aircraft $\longrightarrow$ Attitude (A)	Physical $\longrightarrow$ Input (I)	Internal $\longrightarrow$ Response (R)	Measured Output (O)
•Fwd. Flight •Side Flight •Fwd. Climb •Fwd. Descent •Hover •Hover Turn •Coord. Turn •*[other]*	•Radar Alt. •Airspeed •Climb Rate •Heading •Bank •Pitch •Side Slip •*[other]*	•Engine Torque •Engine Speed •*[Mast Lifting]* •*[Mast Bending]* •*[other]*	•*[Tooth Bending]* •*[Backlash]* •*[Friction]* •*[Heat]* •*[other]* •*[DAMAGE]*	•Vibration - x axis - y axis - z axis •*[Temp]* •*[Noise]* •*[other]*

false-alarm rates. Novelty detection is one approach to overcoming this problem, addressing the problem of modeling the proper operation of a system and detecting when its operation deviates significantly from normal operation [3, 5].

In this paper, we present an approach to novelty detection for in-flight aircraft data. The data were collected as part of a research effort to understand the sources of variability present in the actual flight environment, with the purpose of reducing the high rates of false alarms [4, 8]. In past work, we have described aircraft operation conceptually according to the open-loop causal model shown in Table 1. We assume that the maneuver being performed (M) influences the observable aircraft attitudes (A), which in turn influence the set of possibly observable physical inputs (I) to the transmission. The physical inputs influence the transmission in a variety of ways that are not typically observable (R); however, there are outputs that can be observed (O). Our approach to fault detection in aircraft depends fundamentally on the assumption that the nature of the relationships between the elements M, A, I, R, and O described above change when a fault materializes. Many approaches to fault detection attempt to model only the set of possible outputs (O) and indicate the presence of a fault when the actual outputs do not match the model. However, this approach is difficult because the output space is often too complicated to allow faithful modeling and measuring differences between the model and actual outputs. This latter difficulty remains even if one attempts to model the output as a function of something that influences it such as the physical inputs or the flight maneuver due to noise and other influences. Approaches to fault diagnosis (e.g., [9]) attempt to predict either normal operation or one of a designated set of faults. As stated earlier, this is not possible in the aircraft domain because the set of possible faults is unknown and fault data is non-existent. For this reason, we envision a fault detection system containing a classifier that models the flight maneuver (M) as a function of the outputs (O). This allows us to measure differences between modeled and actual operation in the discrete space of flight maneuvers, which is a much simpler space than the space of vibration signals (O). We would like to harness this fact in our system.

In order for our fault detection system to have a low false-alarm rate, we need a maneuver classifier with the highest performance possible. In addition

to using Multilayer Perceptrons (MLPs) and Radial Basis Function (RBF) networks, we use ensembles of MLPs and RBF networks. We have also identified sets of maneuvers (e.g., three different hover maneuvers) that are similar enough to one another that misclassifications within these groups is unlikely to imply the presence of a fault. Additionally, we smooth over the predictions for small windows of time in order to mitigate the effects of noise.

In the following, Section 2 discusses the aircraft under study and the data generated from them. We discuss the machine learning methods that we used and the associated data preparation that we performed in Section 3. We discuss the experimental results in Section 4. We summarize the results of this paper and discuss ongoing and future work in Section 5.

2 Aircraft Data

The data used in this work were collected from two helicopters: an AH1 Cobra and OH58c Kiowa [4]. The data were collected by having two pilots each fly two designated sequences of steady-state maneuvers according to a predetermined test matrix [4]. It uses a modified Latin-square design to counterbalance changes in wind conditions, ambient temperature, and fuel depletion. Each of the four flights consisted of an initial period on the ground with the helicopter blades at flat pitch, followed by a low hover, a sequence of maneuvers drawn from the 12 primary maneuvers, a low hover, and finally a return to ground. Each maneuver was scheduled to last 34 seconds in order to allow a sufficient number of cycles of the main rotor and planetary gear assembly to apply the signal decomposition techniques used in the previous studies [4].

Summary matrices were created from the raw data by averaging the data produced during each revolution of the planetary gear. The summarized data consists of 31475 revolutions of data for the AH1 and 34144 revolutions of data for the OH58c. Each row, representing one revolution, indicates the maneuver being performed during that revolution as well as the following 30 quantities: Revolutions per minute of the planetary gear, torque (mean, standard deviation, skew, and kurtosis), and vibration data from six accelerometers (root-mean-square, skew, kurtosis, and a binary variable indicating whether signal clipping occurred). For the AH1, the mean and standard deviation values were available for the following attitude data from a 1553 bus: altitude, speed, rate of climb, heading, bank angle, pitch, and slip.

3 Methods

Sample torque and RPM data from one maneuver separated by pilot and by flights are shown in Figures 1 and 2, respectively. The highly-variable nature of the data, as well as differences due to different pilots and different days when the aircraft were flown, are clearly visible and make this a challenging classification problem. To perform the necessary mapping for this problem, we chose multilayer perceptrons (MLPs) with one hidden layer and radial basis function (RBF)

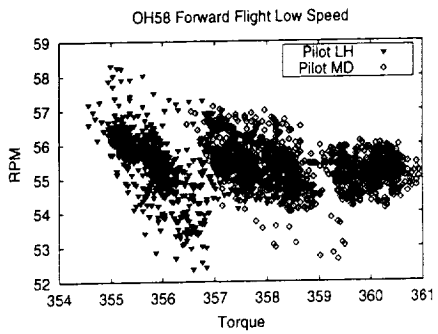

Fig. 1. OH58 Maneuver A (Forward Flight Low Speed) Pilot-Separated

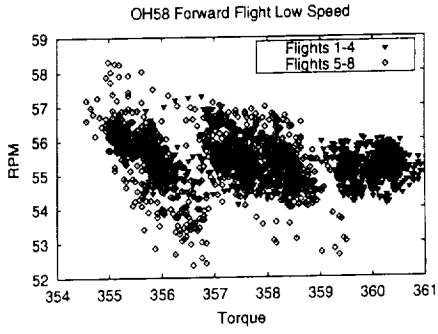

Fig. 2. OH58 Maneuver A (Forward Flight Low Speed) Flight-Separated

Table 2. Sample confusion matrix for OH58 (MLP).

True Class	Predicted Class													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	693	0	7	6	79	0	0	0	0	0	0	0	0	0
2	0	679	0	0	0	0	0	0	0	0	0	0	47	0
3	55	1	568	64	31	6	0	11	9	1	11	7	0	3
4	26	0	43	691	15	0	0	3	0	0	0	2	0	1
5	196	0	68	41	412	0	0	0	0	0	2	16	0	0
6	0	0	0	0	0	719	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	1079	0	0	0	0	0	0	0
8	0	9	22	16	0	0	0	**748**	**177**	**97**	11	6	3	0
9	0	1	1	6	0	0	0	**172**	**381**	**162**	4	7	6	0
10	0	4	1	6	0	0	0	**186**	**170**	**376**	0	8	13	0
11	4	0	15	4	3	0	0	2	1	0	**494**	**217**	0	0
12	3	0	7	6	4	0	0	2	1	0	**200**	**531**	0	0
13	0	63	0	0	0	0	0	4	1	0	0	0	712	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	685

networks as base classifiers. Furthermore, we constructed ensembles of each type of classifier, as well as ensembles consisting of half MLPs and half RBF networks, because ensembles have been shown to improve upon the performance of their constituent or base classifiers, particularly when the correlation among the base classifiers can be kept low [1, 10].

We created data sets for each of the two aircraft by combining its 176 summary matrices. This resulted in 31475 patterns (revolutions) for the AH1 and 34144 for the OH58. Both types of classifiers were trained using a randomly-selected two-thirds of the data (21000 examples for the AH1, 23000 for the OH58) and were tested on the remainder for the first set of experiments. For both aircraft, we used various subsets of the inputs.

In addition, we calculated the *confusion matrix* of every classifier we created. Entry (i, j) of the confusion matrix of a classifier states the number of times that

an example of class i is classified as class j. In examining the confusion matrices of the classifiers (see Table 2 for an example of a confusion matrix—entry $(1,1)$ is in the upper left corner), we noticed that particular maneuvers were continually being confused with one another. In particular, the three hover maneuvers (8-Hover, 9-Hover Turn Left, and 10-Hover Turn Right) were frequently confused with one another and the two coordinated turns (11-Coordinated Turn Left and 12-Coordinated Turn Right) were also frequently confused (the counts associated with these errors are shown in bold in Table 2.) These sets of maneuvers are similar enough to one another that misclassifications within these groups are unlikely to imply the presence of faults. Therefore, for the second set of experiments, we recalculated the classification accuracies allowing for these misclassifications. For our third set of experiments, we consolidated these two sets of maneuvers in the data before running the experiments. That is, we combined the hover maneuvers into one class and the coordinated turns into one class, yielding a total of 11 possible predictions instead of the original 14. We expected the performance to be best for this third set of experiments because, informally, the classifiers do not have to waste resources distinguishing among the two sets of similar maneuvers.

Finally, we used the knowledge that a helicopter needs some time to change maneuvers. That is, two sequentially close patterns are unlikely to come from different maneuvers. To obtain results that use this "prior" knowledge, we tested on sequences of revolutions by averaging the classifiers' outputs on a window of examples surrounding the current one. In one set of experiments, we averaged over windows of size 17 (8 revolutions before the current one, the current one, and 8 revolutions after the current one) which corresponds to about three seconds. Because the initial training and test sets were randomly chosen from this sequence, this averaging could not be performed on the test set alone. Instead it was performed on the full data set for both helicopters. To allow meaningful comparisons of these results, we also computed the errors of the single-revolution classifiers on this full dataset and present them in Tables 4 and 6.[1]

4 Results

In this section we describe the experimental results that we have obtained so far. We first discuss results on the OH58 helicopter. In Table 3, the column marked "Single Rev" shows the results of running individual networks and ensembles of various sizes on the summary matrices randomly split into training and test sets. We only present results for some of the ensembles we constructed due to space limitations and because the ensembles exhibited relatively small gains beyond $N = 10$ base models. MLPs and ensembles of MLPs outperform RBFs

[1] We performed this windowed averaging as though the entire data were collected over a single flight. However, it was in fact collected in stages, meaning that there are no transitions between maneuvers. We show these results to demonstrate the applicability of this method to sequential data obtained in actual flight after training the network on "static" single revolution patterns.

Table 3. OH58c Single Revolution Test Set Results.

Base Type	N	Single Rev	Post-Run Consolidated	Pre-Run Consolidated
MLP	1	79.789 ± 0.072	92.709 ± 0.055	93.566 ± 0.060
	4	81.997 ± 0.065	93.820 ± 0.044	94.422 ± 0.038
	10	82.441 ± 0.045	94.015 ± 0.028	94.672 ± 0.032
	100	82.771 ± 0.016	94.133 ± 0.011	94.672 ± 0.032
RBF	1	75.451 ± 0.103	89.305 ± 0.080	90.460 ± 0.169
	4	75.817 ± 0.048	89.485 ± 0.047	90.912 ± 0.056
	10	75.871 ± 0.040	89.498 ± 0.034	90.987 ± 0.032
	50	75.908 ± 0.016	89.506 ± 0.011	91.018 ± 0.014
MLP/ RBF	2	80.190 ± 0.079	92.834 ± 0.065	93.777 ± 0.046
	4	80.946 ± 0.059	93.189 ± 0.042	94.097 ± 0.048
	10	81.406 ± 0.043	93.403 ± 0.039	94.348 ± 0.025
	100	81.543 ± 0.020	93.463 ± 0.017	94.457 ± 0.011

and ensembles of RBFs consistently. The ensembles of MLPs improve upon single MLPs to a greater extent than ensembles of RBF networks do upon single networks, indicating that the MLPs are more diverse than the RBF networks. Mixed ensembles have performances superior to the pure-MLP ensembles for two base models, but have worse performances for larger numbers of models. Mixed ensembles perform better than pure-RBF ensembles for all numbers of base models. In the smaller ensembles, the diversity provided by including RBF networks helped relative to pure-MLP ensembles. However, in the larger ensembles, replacing half the MLPs with RBFs degrades performance—the RBFs are different from the MLPs but not different enough from each other to warrant having such a large number of them. The standard errors of the mean performances decrease with increasing numbers of base models as is normally the case with ensembles. The column marked "Post-Run Consolidated" shows the single revolution results after allowing for confusions among the hover maneuvers and among the coordinated turns, consolidating them into single classes (hover and coordinated turns). As expected, the performances improved dramatically. The column "Pre-Run Consolidated" shows the single revolution results on the summary matrices in which the hovers and coordinated turns were consolidated as described in section 3. The performances here were consistently the highest as we hypothesized.

The top half of table 4 shows the results of performing the windowed averaging described in the previous section in the column marked "Window of 17." The columns "Window 17 Post-Consolidated" and "Window 17 Pre-Consolidated" give the results allowing for the confusions mentioned earlier. The bottom half of the table gives the full set errors of the single-revolution classifiers. We can clearly see the benefits of windowed averaging, which serves to smooth out some of the noise in the data.

Table 5 shows the results with the AH1 summary matrices randomly split into training and test sets. Table 6 has the windowed averaging and single-

Table 4. OH58c Full Data Set Results.

Base Type	N	Window of 17	Window 17 Post-Consolidated	Window 17 Pre-Consolidated
MLP	1	89.905 ± 0.121	96.579 ± 0.066	97.586 ± 0.078
	4	90.922 ± 0.074	96.799 ± 0.026	97.635 ± 0.041
	10	91.128 ± 0.064	96.820 ± 0.018	97.729 ± 0.031
	100	91.307 ± 0.015	97.063 ± 0.140	97.695 ± 0.006
RBF	1	82.564 ± 0.154	92.831 ± 0.103	94.611 ± 0.124
	4	82.634 ± 0.059	92.882 ± 0.047	94.548 ± 0.063
	10	82.618 ± 0.055	92.895 ± 0.043	94.517 ± 0.029
	50	82.644 ± 0.019	92.901 ± 0.013	94.524 ± 0.012
MLP/ RBF	2	88.674 ± 0.108	95.910 ± 0.059	97.155 ± 0.045
	4	88.895 ± 0.078	95.902 ± 0.040	97.145 ± 0.067
	10	89.140 ± 0.057	95.980 ± 0.033	97.226 ± 0.032
	100	89.320 ± 0.025	96.003 ± 0.012	97.204 ± 0.009
Base Type	N	Single Rev	Single Rev Post Consolidated	Single Rev Pre-Consolidated
MLP	1	82.097 ± 0.072	93.539 ± 0.058	94.495 ± 0.064
	4	84.304 ± 0.049	94.622 ± 0.039	95.321 ± 0.035
	10	84.750 ± 0.043	94.805 ± 0.028	95.540 ± 0.029
	100	85.048 ± 0.012	94.922 ± 0.011	95.595 ± 0.008
RBF	1	76.406 ± 0.099	89.680 ± 0.077	90.788 ± 0.147
	4	76.799 ± 0.040	89.872 ± 0.039	91.187 ± 0.045
	10	76.836 ± 0.033	89.902 ± 0.027	91.244 ± 0.027
	50	76.910 ± 0.011	89.948 ± 0.007	91.271 ± 0.013
MLP/ RBF	2	82.146 ± 0.075	93.523 ± 0.061	94.587 ± 0.049
	4	82.877 ± 0.053	93.854 ± 0.041	94.876 ± 0.051
	10	83.332 ± 0.036	94.066 ± 0.029	95.089 ± 0.024
	100	83.505 ± 0.015	94.142 ± 0.015	95.163 ± 0.014

Table 5. AH1 Single Revolution Test Set Results.

Base Type	N	Single Rev	Post-Run Consolidated	Pre-Run Consolidated
MLP	1	96.752 ± 0.059	99.843 ± 0.032	99.990 ± 0.002
	4	97.284 ± 0.031	99.975 ± 0.010	99.997 ± 0.001
	10	97.448 ± 0.027	99.992 ± 0.001	99.994 ± 0.001
	100	97.542 ± 0.006	99.995 ± 0.001	99.992 ± 0.001
RBF	1	95.669 ± 0.059	99.626 ± 0.017	99.695 ± 0.011
	4	95.946 ± 0.029	99.706 ± 0.010	99.751 ± 0.009
	10	95.911 ± 0.023	99.711 ± 0.006	99.757 ± 0.005
	50	95.946 ± 0.009	99.716 ± 0.003	99.761 ± 0.002
MLP/ RBF	2	97.040 ± 0.054	99.980 ± 0.004	99.994 ± 0.002
	4	97.318 ± 0.025	99.986 ± 0.003	99.998 ± 0.001
	10	97.429 ± 0.018	99.990 ± 0.002	99.998 ± 0.001
	100	97.521 ± 0.011	99.998 ± 0.001	100.000 ± 0.000

Table 6. AH1 Full Data Set Results.

Base Type	N	Window of 17	Window 17 Post-Consolidated	Window 17 Pre-Consolidated
MLP	1	98.344 ± 0.059	99.737 ± 0.028	100.000 ± 0.000
	4	98.757 ± 0.031	99.811 ± 0.005	100.000 ± 0.000
	10	98.779 ± 0.021	99.815 ± 0.002	100.000 ± 0.000
	100	98.861 ± 0.006	99.816 ± 0.001	100.000 ± 0.000
RBF	1	96.662 ± 0.102	99.404 ± 0.013	99.653 ± 0.010
	4	96.988 ± 0.042	99.431 ± 0.012	99.659 ± 0.021
	10	96.968 ± 0.028	99.428 ± 0.008	99.676 ± 0.007
	50	97.003 ± 0.008	99.438 ± 0.003	99.696 ± 0.003
MLP/ RBF	2	98.256 ± 0.064	99.690 ± 0.006	99.908 ± 0.003
	4	98.482 ± 0.034	99.682 ± 0.004	99.901 ± 0.013
	10	98.475 ± 0.028	99.683 ± 0.003	99.918 ± 0.002
	100	98.553 ± 0.005	99.687 ± 0.001	99.920 ± 0.001
Base Type	N	Single Rev	Single Rev Post-Consolidated	Single Rev Pre-Consolidated
MLP	1	96.933 ± 0.060	99.826 ± 0.037	99.992 ± 0.009
	4	97.555 ± 0.025	99.975 ± 0.014	99.997 ± 0.007
	10	97.683 ± 0.013	99.994 ± 0.009	99.997 ± 0.005
	100	97.762 ± 0.008	99.996 ± 0.009	99.997 ± 0.001
RBF	1	95.743 ± 0.067	99.676 ± 0.014	99.726 ± 0.012
	4	96.063 ± 0.032	99.738 ± 0.005	99.767 ± 0.008
	10	96.042 ± 0.026	99.742 ± 0.009	99.773 ± 0.009
	50	96.067 ± 0.005	99.747 ± 0.000	99.781 ± 0.002
MLP/ RBF	2	97.231 ± 0.055	99.984 ± 0.000	99.997 ± 0.005
	4	97.502 ± 0.028	99.988 ± 0.005	99.998 ± 0.005
	10	97.570 ± 0.018	99.993 ± 0.005	99.999 ± 0.003
	100	97.659 ± 0.008	99.999 ± 0.005	100.000 ± 0.000

revolution classifier results, respectively, on the full AH1 dataset. These results are substantially better than the OH58 results. We expected this because the AH1 is a heavier helicopter, so it is less affected by conditions that tend to introduce noise such as high winds. Just as with the OH58, with the AH1 without consolidating maneuvers, the mixed ensembles outperform the pure ensembles for small numbers of base models but perform worse than the MLP ensembles for larger numbers of base models. With consolidation, the mixed ensembles outperform the pure ensembles more often; however, the performances are all very high. Once again, we can see that ensembles of MLPs outperform single MLPs to a greater extent than ensembles of RBFs outperform single RBFs, so the RBFs are not as different from one another. Because of this, it does not help to add large numbers of RBF networks to an MLP ensemble. The standard errors of the mean performances tend to decrease with increasing numbers of base models just as with the OH58.

On the AH1, the hover maneuvers were frequently confused just as they were on the OH58, but the coordinated turns were not confused. Taking this confusion into account boosted performance significantly. The windowed averaging

Table 7. AH1 Bus and Non-Bus Results

Inputs	Single Rev	Single Rev Consolidated	Window of 17	Window of 17 Consolidated
Bus	90.380 ± 0.110	95.871 ± 0.091	91.209 ± 0.126	96.027 ± 0.086
Non-Bus	87.884 ± 0.228	93.731 ± 0.171	92.913 ± 0.355	96.110 ± 0.236
$P(agree)$	79.523 ± 0.247	90.063 ± 0.202	85.609 ± 0.320	93.393 ± 0.247

approach did not always yield improvement when allowing for the maneuver confusions, but helped when classifying across the full set of maneuvers. However, in all cases when windowed averaging did not help, the classifier performance was at least 99.6%, so there was very little room for improvement.

5 Discussion

In this paper, we presented an approach to fault detection that contains a subsystem to classify an operating aircraft into one of several states. More specifically, the proposed subsystem determines the maneuver being performed by an aircraft as a function of vibration data and any other available data. Through experiments with two helicopters, we demonstrated that the subsystem is able to determine the maneuver being performed with good reliability. These results show great promise in classifying the correct maneuver with high certainty. Future work will involve applying this approach to "free-flight data", where the maneuvers are not static or steady-state, and transitions between maneuvers exist.

We are currently constructing classifiers using different subsets of the available data as inputs. For example, for the AH1, we have constructed some classifiers that use only the bus data as input and others that use only the vibration data. We hypothesize that disagreement among these classifiers that use different sources of information may indicate the presence of a fault. For example, if the vibration data-based classifier predicts that the aircraft is flying forward at high speed but the bus data-based classifier predicts that the aircraft is on the ground, then the probability of a fault is high. Table 7 shows the results of training 20 single MLPs on these data using the same network topology as for the other MLPs trained on all the AH1 data. They performed much worse than the single MLPs trained with all the inputs presented at once. The last line in the table indicates the percentage of maneuvers for which the two types of classifiers agreed. We would like these agreement probabilities to be much higher because none of our data contains faults. However, we hypothesize that we can use the bus data in a much simpler way to achieve better performance. For example, if a vibration data-based classifier predicts that the aircraft is performing a forward flight, but the bus data indicate that airspeed is near zero, then the probability of a fault is high. We do not necessarily need a classifier that returns the maneuver as a function of all the variables that constitute the bus data. In this example, we merely need to know that a near-zero airspeed is inconsistent

with a forward flight. We plan to perform a detailed study of the collected bus data so that we may construct simple classifiers representing knowledge of the type just mentioned and use them to find inconsistencies such as what we just described.

There is ongoing work within our research group to model aircraft engine operation from "first principles." In particular, models of the gear system are being prepared so that simulated data may be collected. We plan to use this simulation to insert cracks and other types of faults in the gear system in order to learn how the data changes as a function of these faults. This information can be used to mathematically insert faults into the real data. This gives us the fault data that we clearly cannot collect from the aircraft directly. We hope to generate such fault data and test whether our classification subsystems react to fault data in the way we expect.

References

1. L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.
2. Robert Campbell, Amulya Garga, Kathy McClintic, Mitchell Lebold, and Carl Byington. Pattern recognition for fault classification with helicopter vibration signals. In *American Helicopter Society 57th Annual Forum*, 2001.
3. Paul Hayton, Bernhard Schölkopf, Linel Tarassenko, and Paul Anusiz. Support vector novelty detection applied to jet engine vibration spectra. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in Neural Information Processing Systems-13*, pages 946–952. Morgan Kaufmann, 2001.
4. Edward M. Huff, Irem Y. Tumer, Eric Barszcz, Mark Dzwonczyk, and James McNames. Analysis of maneuvering effects on transmission vibration patterns in an AH-1 cobra helicopter. *Journal of the American Helicopter Society*, 2002.
5. Nathalie Japkowicz, Catherine Myers, and Mark Gluck. A novelty detection approach to classification. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 518–523, Montreal, Canada, 1995. Morgan Kaufmann.
6. D.A. McAdams and I.Y. Tumer. Towards failure modeling in complex dynamic systems: impact of design and manufacturing variations. In *ASME Design for Manufacturing Conference*, volume DETC2002/DFM-34161, September 2002.
7. Sunil Menon and Rida Hamza. Machine learning methods for helicopter hums. In *Proceedings of the 56th Meeting of the Society for Machinery Failure Prevention Technology*, pages 49–55, 2002.
8. I.Y. Tumer and E.M. Huff. On the effects of production and maintenance variations on machinery performance. *Journal of Quality in Maintenance Engineering*, To appear, 2002.
9. V. Venkatasubramanian, R. Vaidyanathan, and Y. Yamamoto. Process fault detection and diagnosis using neural networks—i. steady-state processes. *Computers and Chemical Engineering*, 14(7):699–712, 1990.
10. D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.