CHAPTER 5

TESTING THE SYSTEM

5.1 Introduction

Several parameters and modules are present in the change detection system described in chapter 4, each having their own influence on the whole change detection system. The meaning of performing tests is to evaluate every single parameter or module properties, and to reach a conclusion on which value is optimal for the individual parameter and which design is optimal for the individual design. Most often we are faced with the problem that parameter values are dependent on each other, i.e. a change in parameter A will cause a change in parameter B in order to fulfill the design criteria.

To assess every parameter and design in the system, one could apply the following two approaches,

1. *For every parameter P_i an interval is defined, which goes from the minimum value of P_i, which has to be assessed, to the maximum value of P_i, which has to be assessed, with a proper step value. Next, all combinations of parameter values are tested on the system. This can easily be done by use of for-loops. The same procedure is applied on module designs, i.e. for every module a specific number of designs are chosen, and all combinations of designs are then tested on the system.*

2. *Similar to approach 1, an interval is defined for every parameter, and designs are chosen for every module. But instead of testing all combinations of parameter values and module designs, only a single parameter or module is swept in its interval or design methods, respectively, and the remaining parameters and modules are fixed at a single value or design, respectively.*

It is obvious that approach no. 1 consumes a lot of time if several parameter values and module designs are to be tested. Consider the following example,

Six parameters are to be tested. For each parameter, five values must be tested, and this corresponds to 6^5 = 7,776 different combinations. Two modules with three designs each must also be tested, this gives 2^3 = 8 different design combinations. In the end, 7,776 8 = 62,208 different combinations have to be tested. If a test on a single combination lasts one minute, then the full test, i.e. all 62,208 combinations will last 43.2 day and nights.*

If approach no. 2 is used in the above example, only $6 * 5 * 2 * 3 = 180$ combinations must be tested. This will only take three hours, if a single test lasts one minute. Since dependent parameters and designs exist, approach no. 2 is not appropriate in its full form. On the other hand, approach no. 1 is not feasible due to the vast amount of test time. Therefore, a compromise has to be established.

At this point it is a good idea to make an overview over the modules and parameters included in this work. The modules are:

Modules:
- Acquiring data.
- Pre-processing and feature extraction.
- On-line change detection.
- Off-line hypothesis testing.
- Off-line change point estimation.

And the parameters in the modules are:

Acquiring data module parameters:
- The original data is applied from the four AE-RMS sensors
- Five re-sampling methods:
  - N: the size of the window used in some of the methods.
  - $\alpha(c)$: The mixer function applied in method no. 5.

Pre-processing and feature extraction module parameters:
- Mean value of cycles:
  - All samples in the cycles are used.
  - Hand tuned cycle intervals are used.
- Standard deviation of cycles:
  - All samples in the cycles are used.
  - Hand tuned cycle intervals are used.
- PCA:
  - c: the number of components.
  - N: The size of the training set.
  - PC's used as feature signals.
  - SE used as feature signal.
  - All samples in the cycles are used.
  - Hand tuned cycle intervals are used.

On-line change detection module parameters:
- Boost on or off.
- A: Boost factor.
- $h_{A/2}$: Boost factor.
- Choice of a_{sub} (where $y_{boo} = 1 \cdot y$):
  - A priori investigation on feature signals by "simulated technicians": $a_{sub} =$ (dynamic range / 20).
  - Different percentiles.
- Thresholding only $\sigma_{\mu(x)}$, $\sigma_{\sigma(x)}$ or both.

- *cal_wl*: Calibration window length.
- *mwl*: Main window length.
- *swl*: Small window length.
- *sw_dist*: Distance between small windows.

Off-line hypothesis test module parameters:
- Critical values for ratios:
  - *cri_mean*: Change in mean value.
  - *cri_devi*: Change in deviation.
  - *cri_both*: Change in both mean value and deviation.
- *N*: The number of cycles used to estimate the distribution in the beginning and the end of the feature signal.

Off-line change point estimation module parameters:
- Two maximum likelihood methods.
- *N*: The number of cycles used to estimate the distribution in the beginning and the end of the feature signal.

Three main tests will be executed. The first one applies the original data only, thus no re-sampling is present. The purpose of the test is to investigate whether or not the change detection system has a chance of detecting and estimating the engine changes adequately. In order to use statistics, so that the system is tested on several examples, and not only on the single original one, the second main test is executed. It uses a re-sampling method and is tested on a single experiment from a single sensor. The third and final test is a noise test, which is simply the second test with noise added. In the end of the chapter a proposal on how to implement a final automatic condition monitoring system is given, based on the results from tests.

5.2 Test on the original data

In the following a test specification is given for the test on the original data.

Acquiring data module parameters:
- The original data is applied from the four AE-RMS sensors.

Pre-processing and feature extraction module parameters:
- Mean value of cycles:
  - All samples in the cycles are used.
- Standard deviation of cycles:
  - All samples in the cycles are used.
- PCA:
  - *c*: 20.

- N: 50.
- SE used as feature signal.
- All samples in the cycles are used.

On-line change detection module parameters:
- Boost on.
- A: 30.
- $h_{A/2}$: $1.5 \cdot a_{sub.}$
- Choice of a_{sub}:
  - The 100th percentile.
- Thresholding both $\sigma_{\mu(x)}$ and $\sigma_{\sigma(x)}$.
- *cal_wl*: 30.
- *mwl*: 30.
- *swl*: 10.
- *sw_dist*: 30.

Off-line hypothesis test module parameters:
- Critical values for ratios:
  - *cri_mean*: 1.
  - *cri_devi*: 2.
  - *cri_both*: [2; 8].
- N: 50.

Off-line change point estimation module parameters:
- Two maximum likelihood methods.
- N: 50.

5.2.1 Pre-processing and feature extraction

In chapter 3, the mean value and the standard deviation as feature signals were given for every four AE sensors. All the five condition changes, but the unstable region, are observable in these feature signals. Fortunately, the PCA feature signals show this unstable region. However, they seemed to have trouble observing the changes related to the oil experiments. Appendix C shows all the feature signals, generated by the PCA approach, of the four sensors[1]. Also the test, where the training set has a smaller size (N = 20 examples) and fewer components (c = 10), is shown.

From the figures of the feature signals in chapter 3 and appendix B, it is obvious that sensor no. 1 and no. 3 senses the first oil experiment far better that sensor no. 2 and no. 4. However, the situation is the opposite in the last oil experiment, when the oil lubrication is turned on again. The feature signals from sensor no. 2 and no. 4 are also “cleaner”, i.e. the conditions

[1] NB: The change occurs at cycle 103 in the PCA cases when N = 50, and not at cycle 153. When N = 20, the change is at cycle 133.

are more steady, than the feature signals from sensor no. 1 and no. 3. Applying cleaner feature signals will probably make the change detection system more reliable with relation to the probability of false alarms.

Perhaps there is a problem with the oil experiments, because they seem to consist of only a normal condition and a drift. The problem is that if the feature signal does not drift in to a new steady condition, then the on-line algorithm fails in detecting the second change point. However, this might be a small problem, since the experiments have be cut down to a reasonable number of cycles around the first change point. Therefore, the feature signals will in fact drift in to a new steady condition.

5.2.2 On-line change detection

Appendix C consists of DMD-plots provided by applying the DMD algorithm on the feature signals of each experiment. All three pre-processing and feature extraction methods are investigated, thus for each sensor there are five DMD-plots of the mean value feature signals, the standard deviation feature signals, and the residual error feature signals (N = 50, c = 20). In figure 5.1 the mean value from sensor no. 1 is used as the feature signal from the first oil experiment. The change is very easy to observe, and the DMD-plot reveals that the two decision functions have very nice peaks. Thus the change is likely to be detected by the DMD algorithm. Notice that the boost function seems to work as expected, since the peaks corresponding to false alarms are diminished, and the true alarm peaks are enhanced.

Figure 5.2 shows another DMD-plot where the mean value is used as a feature signal. But in this case no change is observable. This is because the experiment is the unstable region experiment, for which it has been experienced that the feature signal generation approach is not capable of observing. The peaks in the decision functions are small compared to the peaks in the decision functions in figure 5.1. This means that the DMD algorithm might be able to

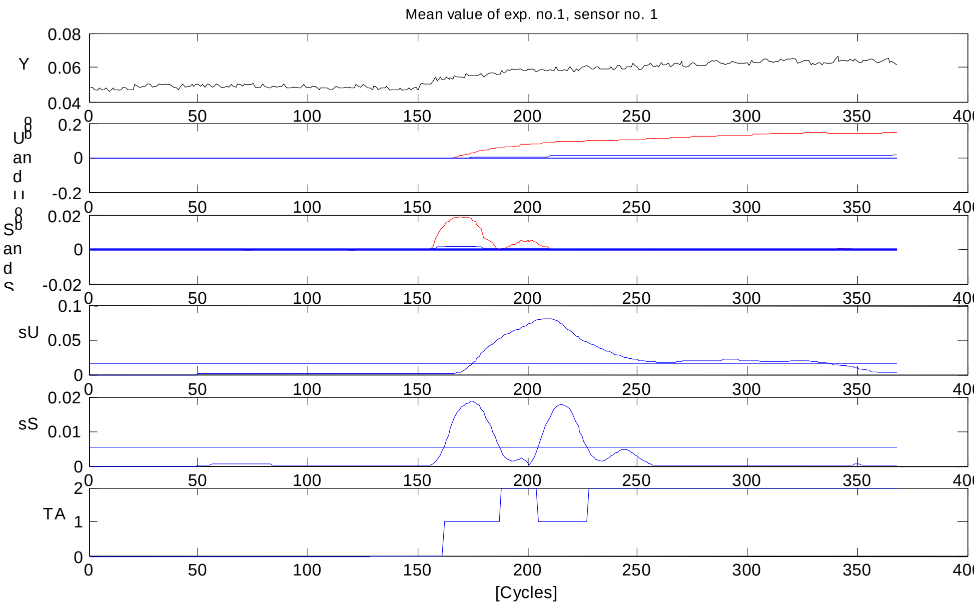

Figure 5.1: DMD-plot of experiment no. 1, sensor no. 1, applying the mean value as the feature signal. The DMD algorithm detects the change.

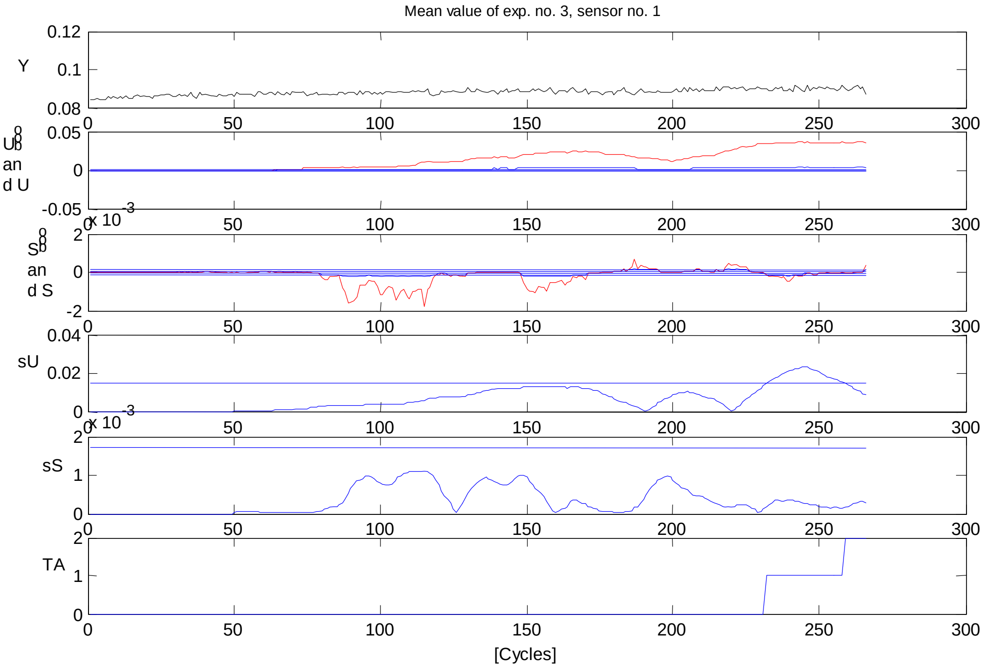

Figure 5.2: DMD-plot of experiment no. 3, sensor no. 1, applying the mean value as the feature signal. The DMD algorithm is not likely to detect a change, when there is no change.

detect changes, when they actually occur, and that it is less sensitive of the changes in the feature signal, which can not be related to a change.

To assess how sensitive the DMD algorithm is to changes in the feature signals, a threshold is inserted in the decision functions. The threshold is the same for all experiments and sensors being 20 times a_{sub}, which is the magnitude of the feature signal causing the feature signal to be multiplied by 1. Thus, if the ratio between the maximum value of the decision function and the threshold is small, then the DMD algorithm probably does not detect a change. On the other hand, if the ratio is high, then the algorithm probably detects the change.

In a previous section it was mentioned that the standard deviation perhaps should be left as feature signals, since they have far more “unstable” conditions than the mean value feature signals. However, experiment no. 4 from sensor no. 4 reveals that this is not true in all cases. Figure 5.3 shows the DMD-plot for this experiment when the feature signal is the mean value of the cycles. Figure 5.4 is the same but with the standard deviation feature signal. The thresholds are plotted “higher” in the decision functions in the mean value figure. Thus, the DMD algorithm will have more trouble detecting the change in this feature signal.

As expected, it is feasible to apply the residual error from the PCA approach as a feature signal to the DMD algorithm. This can be confirmed in figures 5.5-10. However, with sensors no. 1 and no. 3 the DMD algorithm will have a problem in detecting the second change point, since the drift from the normal condition to the new condition is relatively long. But again, sensor no. 2 and no. 4 do not have this long drift and the DMD algorithm attached to these feature signals detects the second change point in adequate time. Thus, this is not a major

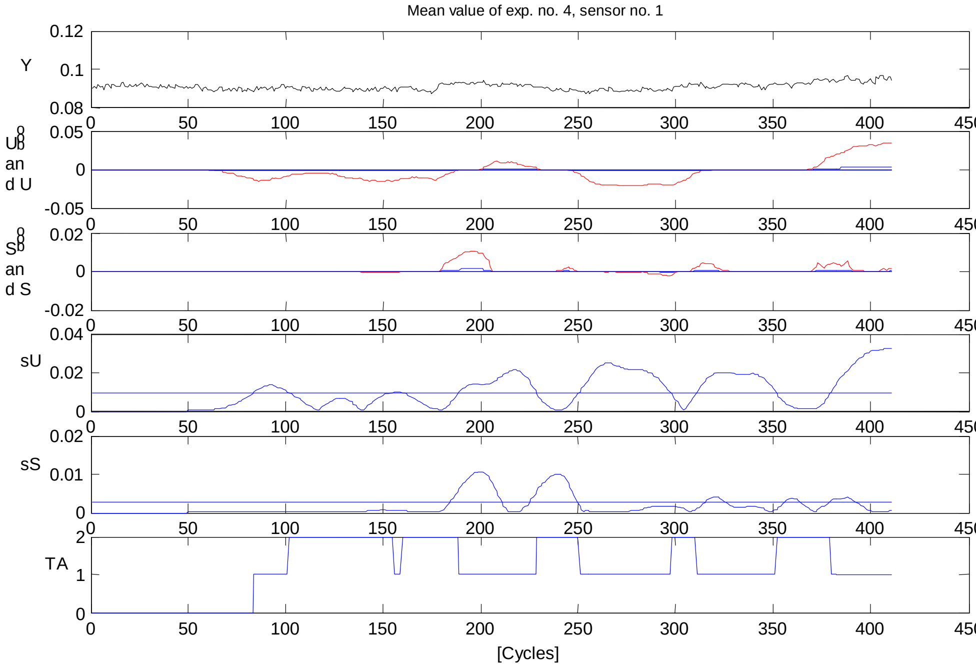

Figure 5.3: DMD-plot of experiment no. 4, sensor no. 1, applying the mean value as the feature signal. The DMD algorithm is less likely to detect the change than if the standard deviation feature signal is applied.

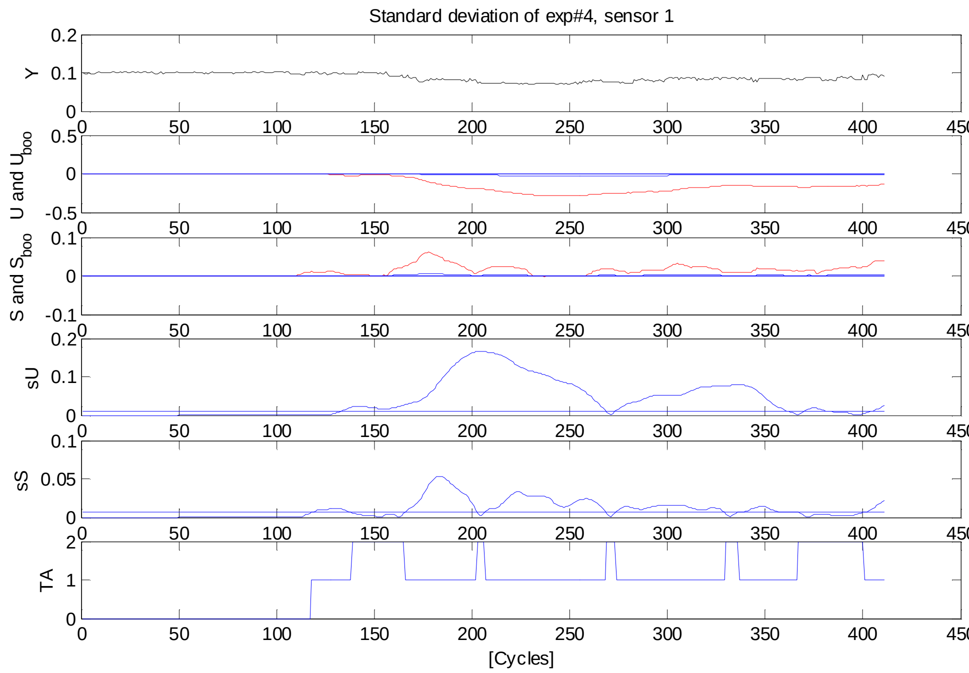

Figure 5.4: DMD-plot of experiment no. 4, sensor no. 1, applying the standard deviation as the feature signal. The DMD algorithm is more likely to detect the change than if mean value feature signal is applied.

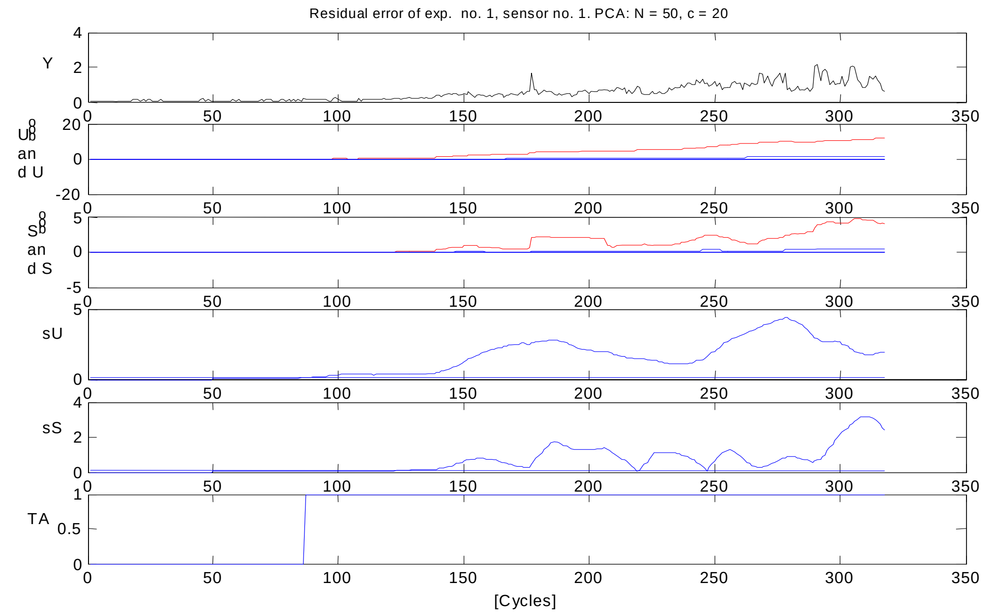

Figure 5.5: DMD-plot of experiment no. 1, sensor no. 1, applying the residual error as the feature signal. The DMD algorithm will have a problem detecting the second change point, because of the long drift.

problem if all the sensors are used, and if the statements from the different algorithms are combined properly. In chapter 3, it was stated that the oil changes, especially the first one, would have troubled being detected if only sensor no. 2 is applied, because of a poor ratio. However, figure 5.6 and 5.10 show that this is not the case.

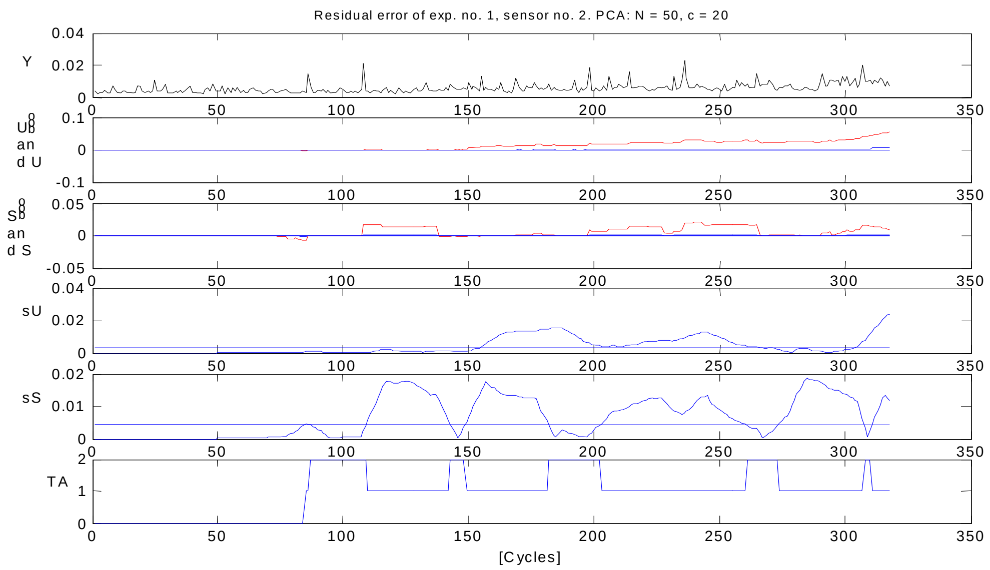

Figure 5.6: DMD-plot of experiment no. 1, sensor no. 2, applying the residual error as the feature signal. The DMD algorithm does not have a problem detecting the second change point.

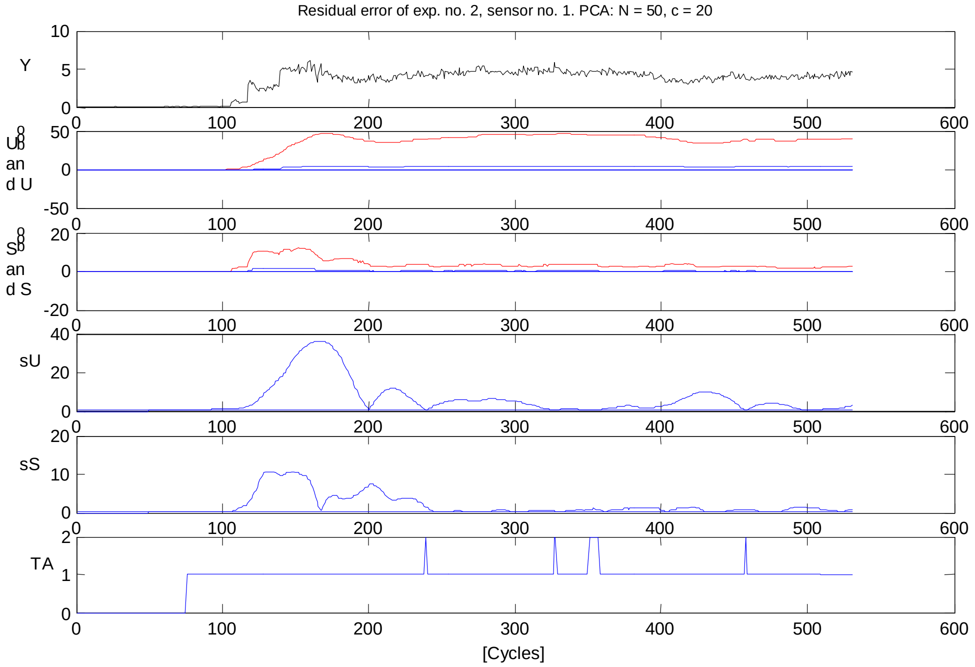

Figure 5.7: DMD-plot of experiment no. 2, sensor no. 1, applying the residual error as the feature signal. The DMD algorithm has no problem at all detecting both change points.

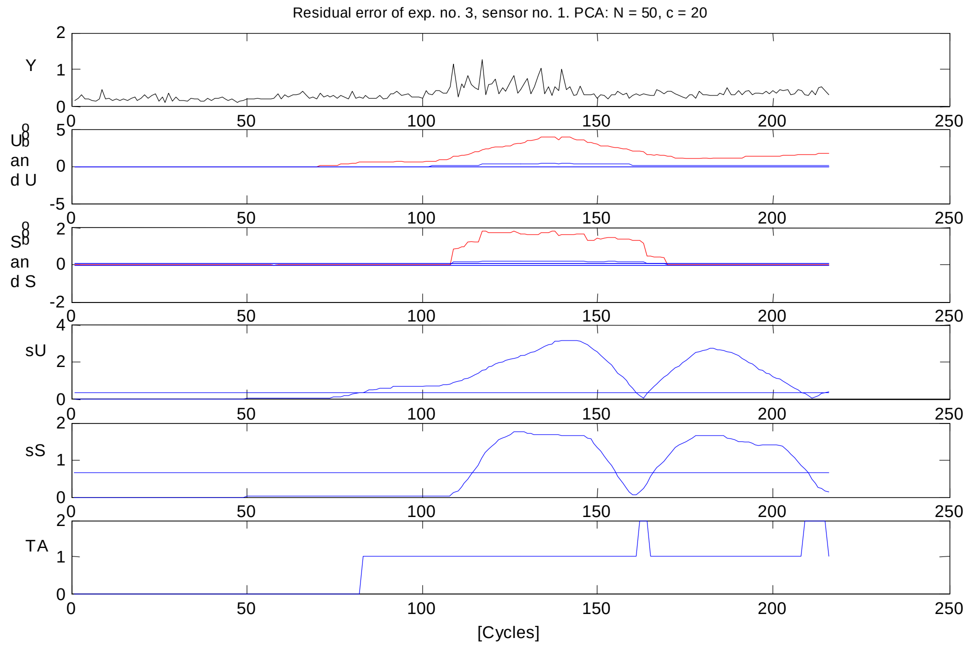

Figure 5.8: DMD-plot of experiment no. 3 (the unstable region), sensor no. 1, applying the residual error as the feature signal. The DMD algorithm succeeds in detecting both change points. This wont be a success if the other two feature signal generation approaches are applied in stead of the residual error approach.

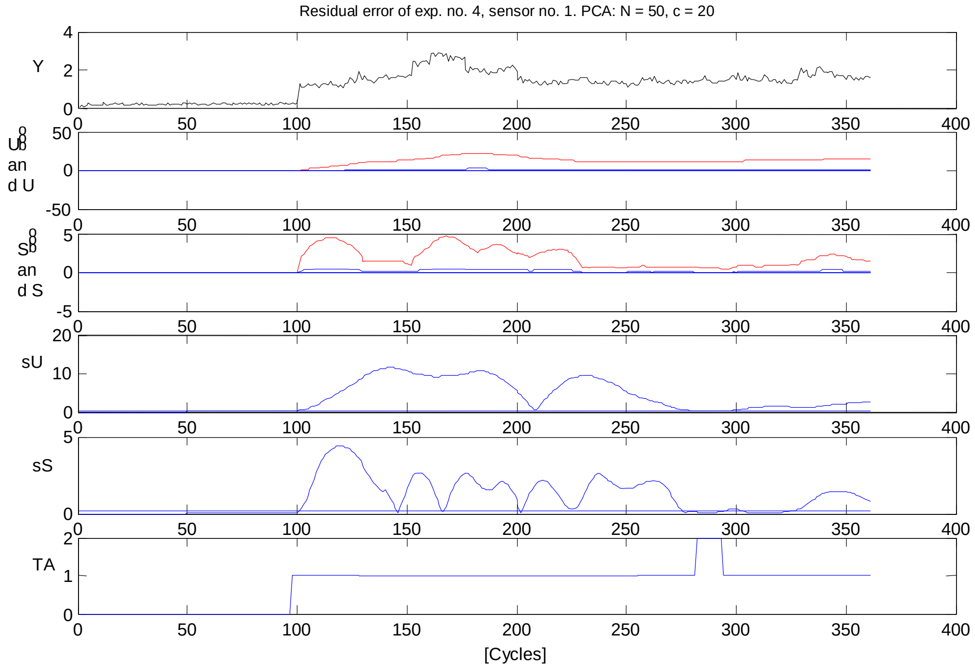

Figure 5.9: DMD-plot of experiment no. 4, sensor no. 1, applying the residual error as the feature signal. Again, no problem in detecting both change points.

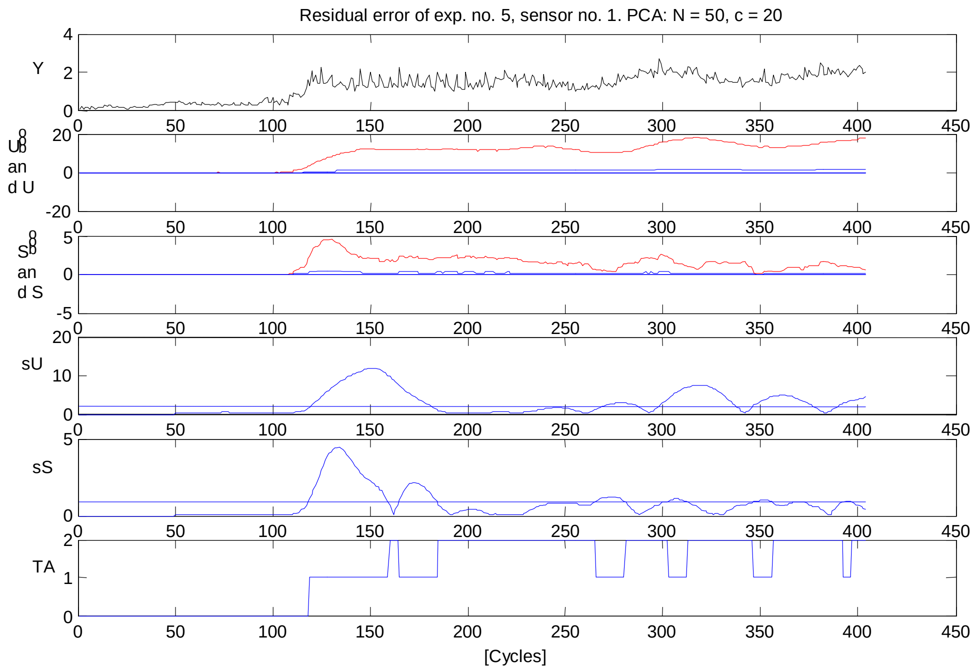

Figure 5.10: DMD-plot of experiment no. 5, sensor no. 1, applying the residual error as the feature signal. Again the DMD algorithm succeeds in detecting both change points.

5.2.3 Off-line hypothesis test

On the original AE-RMS data set, testing on the full experiment and on the normal condition only, performs the off-line hypothesis test. Figure 5.11 shows the off-line hypothesis test of the full experiment no. 1 from sensor no. 1. The first element in the estimation vector *est* is 1 and the remaining elements are 0. Thus, the off-line hypothesis test estimates that a change has occurred only in the mean value. Looking at the signal confirms this, which means that the test functions as wanted. The same test is also performed on the normal condition of the experiment, i.e. a no-change experiment. Figure 5.12 shows the result. The estimation vector is all-zero, thus the statement is that no change has occurred, which is true.

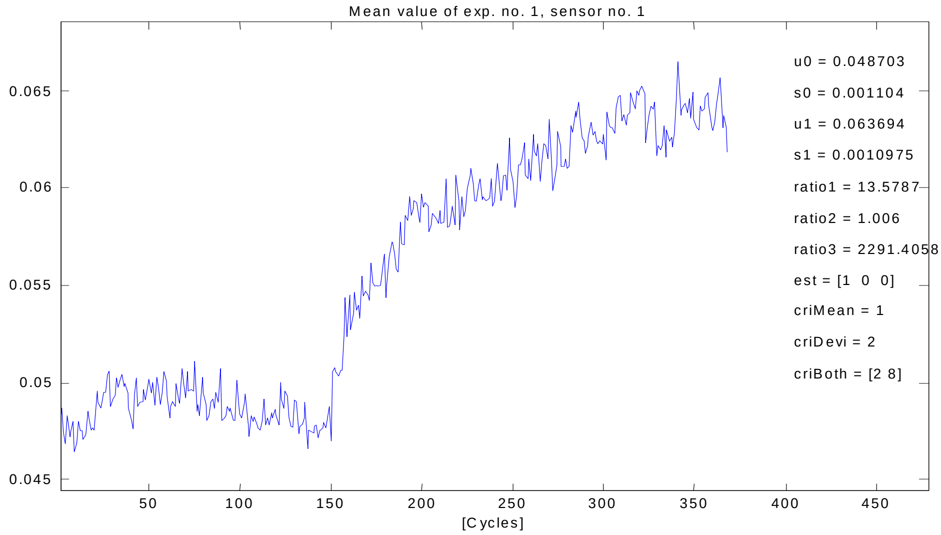

Figure 5.11: Off-line hypothesis plot of the full experiment no. 1, sensor no. 1, applying mean value as the feature signal. A change has occurred, and this is confirmed by the test.

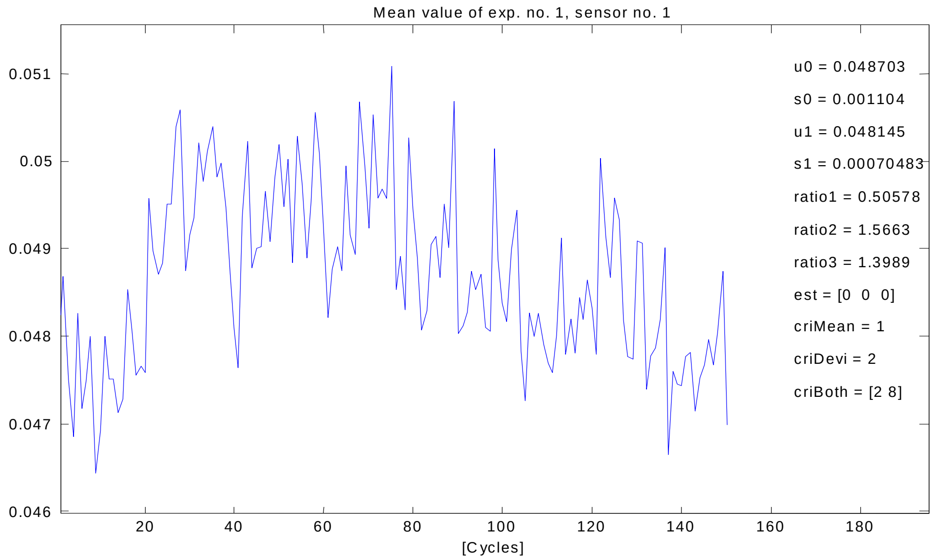

Figure 5.12: Off-line hypothesis plot of the same experiment as in figure 5.11, but only for the normal condition. No change has occurred, which is confirmed by the test.

From the appendix it is observed that:

- All tests on the mean value feature signals confirm a change in the full experiments, but for the unstable region experiment.
- The standard deviation applied as feature signals provides a less reliable off-line hypothesis test, since it in some cases, where a change has occurred, estimates that no change has occurred, and in other cases it estimates that a change has occurred when no change has not occurred – see figure 5.13-14.
- All residual error feature signals, which include a change, are also estimated to include changes.
- All residual error feature signals from sensor no. 2 and no. 4, which do not include a change, are also estimated to include no changes.

From these observations, a well-working change detection system that detects and confirms all engine condition changes has been obtained. Also, it is very probable that when false alarms occur in the on-line change detection, they will be ignored by the off-line hypothesis test. However, this is only true if the PCA approach is used to generate feature signals.

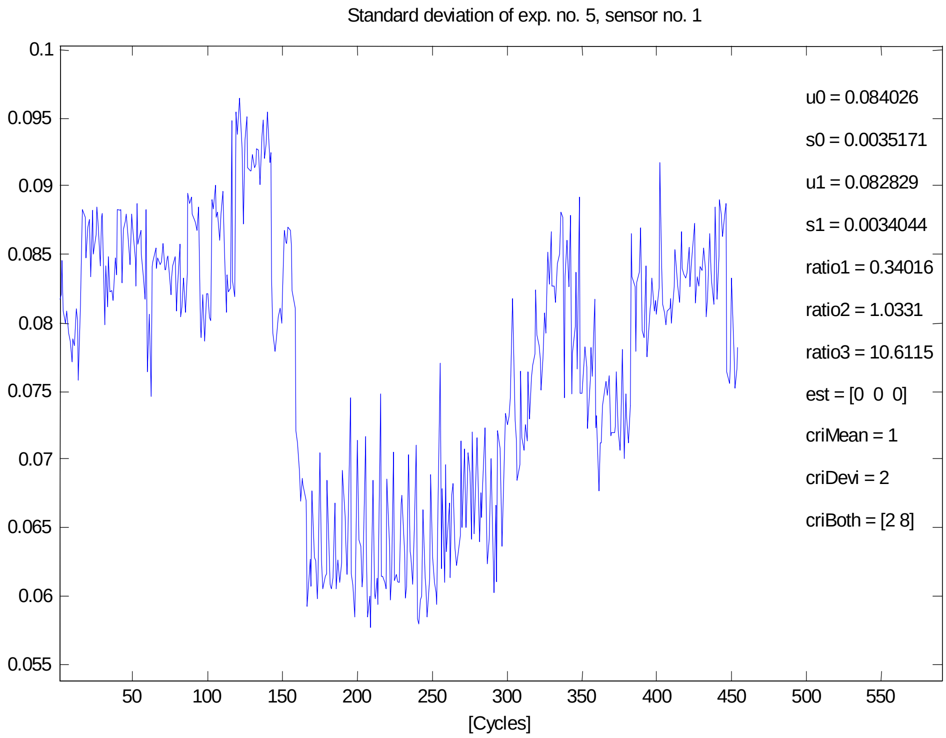

Figure 5.13: Off-line hypothesis plot of the full experiment no. 1, sensor no. 1. A change has occurred, but this is not confirmed by the test.

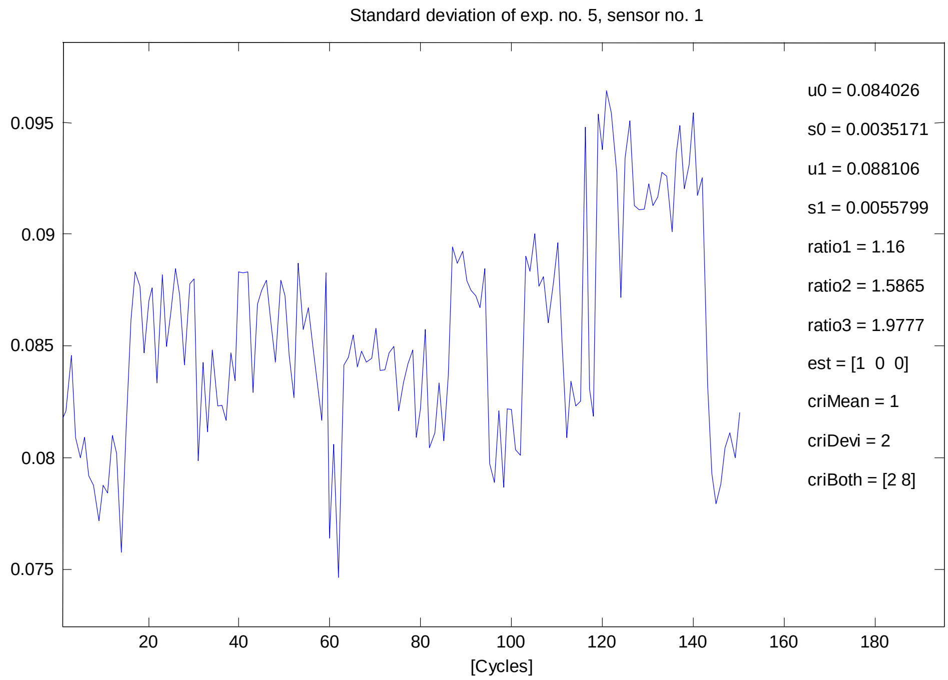

Figure 5.14: Off-line hypothesis plot of the normal condition in experiment no. 1, sensor no. 1. A change has not occurred, but this is not confirmed by the test.

5.2.4 Off-line change point estimation

Two maximum likelihood methods are tested on all the feature signals from the full experiments. The results are given in appendix E. The first method is shown in figures 5.16-17. From these figures it can be concluded that if the feature signal fulfill the assumption about a linear drift between the two conditions, then the method works adequately. But if the drift is not linear, the first change point is estimated too far from the true change point. Notice that an abrupt change is in fact a linear drift.

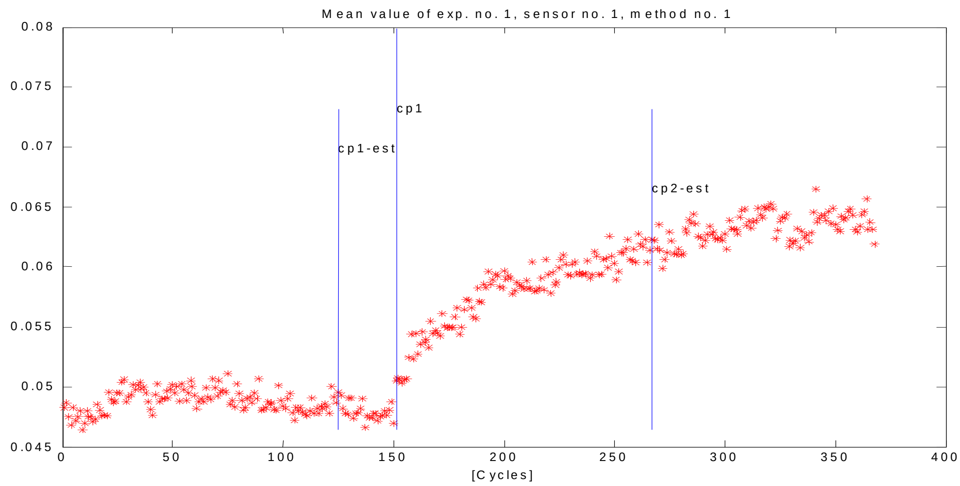

Figure 5.15: Off-line change point estimation of the mean value feature signal from experiment no. 1, sensor no. 1. The drift is not linear, thus cp_1 is too far from cp_{true}.

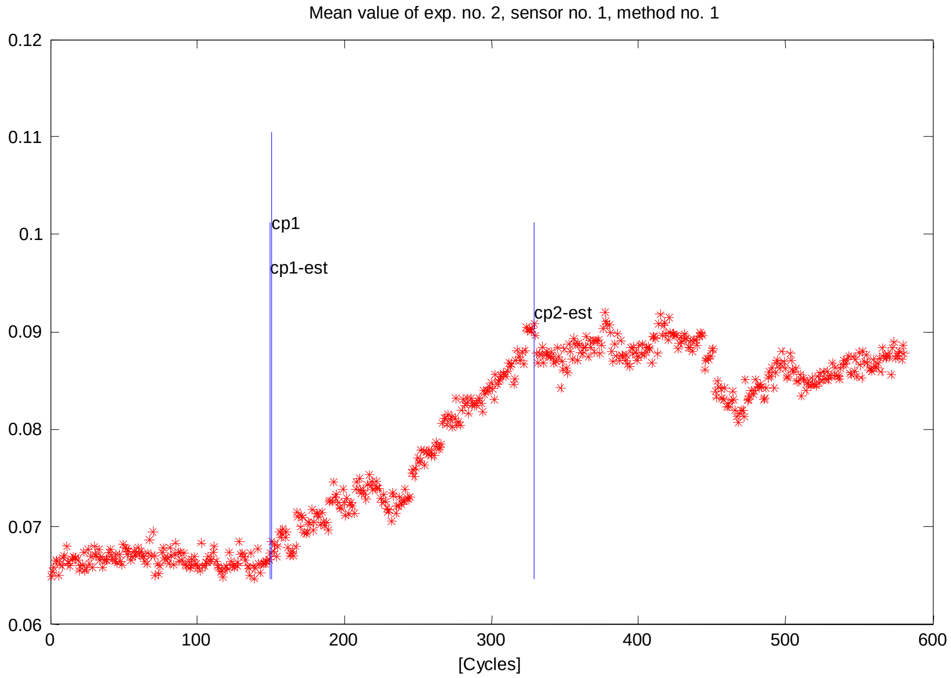

Figure 5.16: Off-line change point estimation of the mean value feature signal from experiment no. 2, sensor no. 1. The drift is linear, thus cp_1 is almost equal to cp_{true}.

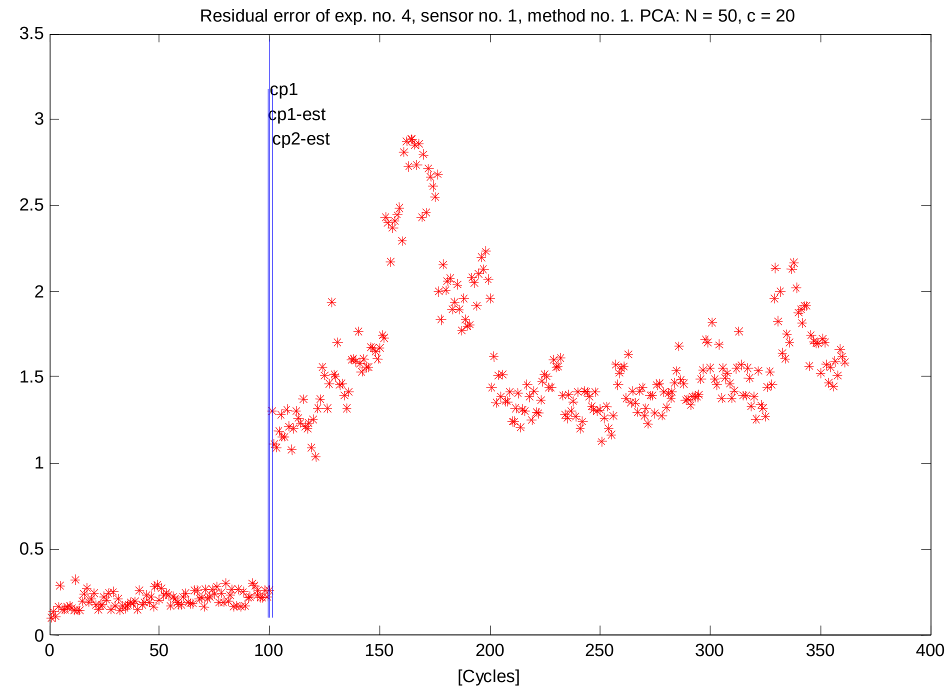

Figure 5.17: Off-line change point estimation of the residual error feature signal from experiment no. 4, sensor no. 1. The drift is abrupt, thus cp_1 and cp_2 are almost equal to cp_{true}.

To avoid the strong dependency on the drift, the second maximum likelihood method was developed and tested. The results are shown in figures 5.18-20, where the experiments are the same as in figures 5.15-17. The estimation of cp_1 has improved significantly for the case on a no linear drift. Unfortunately, something is very wrong with the estimation of cp_2. This change point seems to be in the end of the feature signals in nearly each case. The reason behind this fault estimation is perhaps that the new condition is likely to be unsteady. However, if method no. 2 is used to estimate cp_1, and method no. 1 is used to estimate cp_2, then there is a fair chance of estimating the true change points properly.

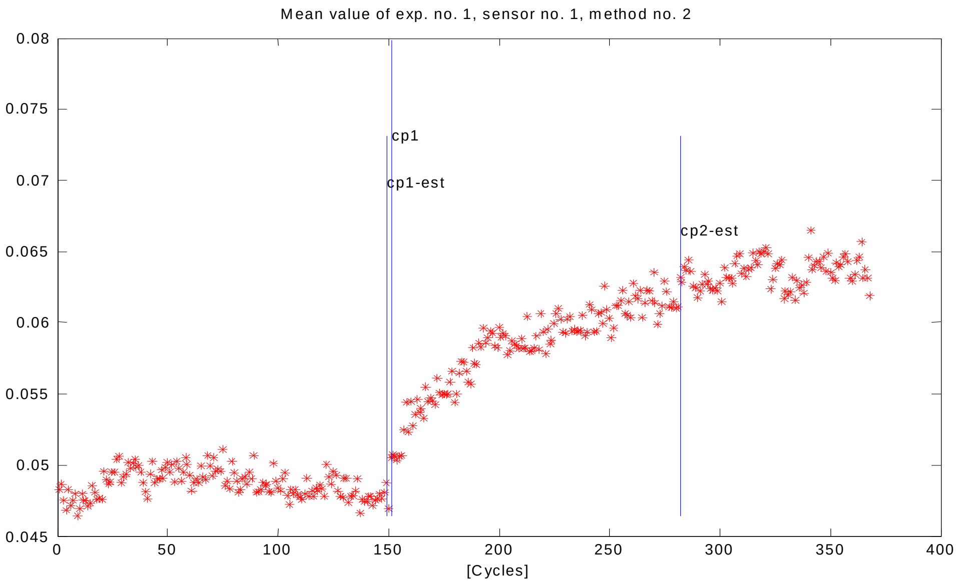

Figure 5.18: Off-line change point estimation of the mean value feature signal from experiment no. 1, sensor no. 1. cp_1 is now almost equal to cp_{true}. cp_2 is almost the same.

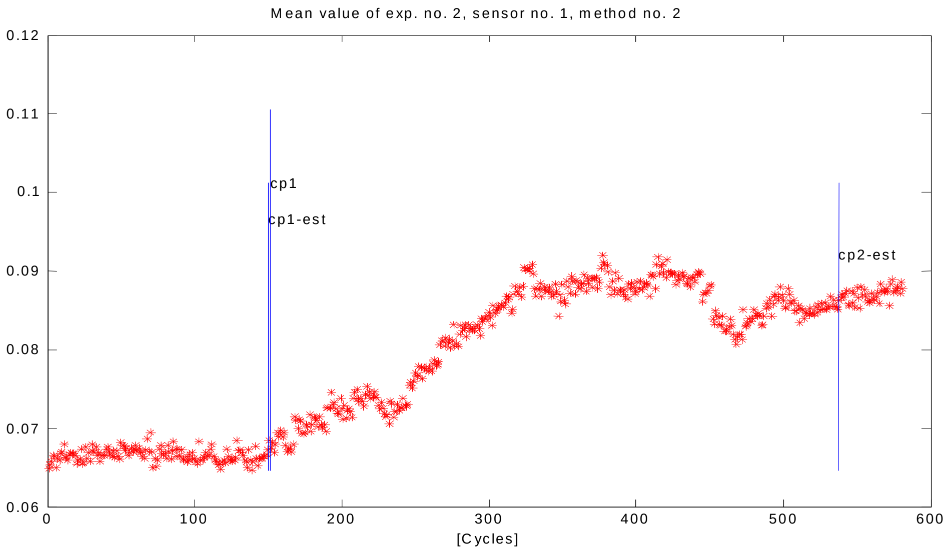

Figure 5.19: Off-line change point estimation of the mean value feature signal from experiment no. 2, sensor no. 1. cp_1 is again almost equal to cp_{true}, but cp_2 is at the very end of the experiment.

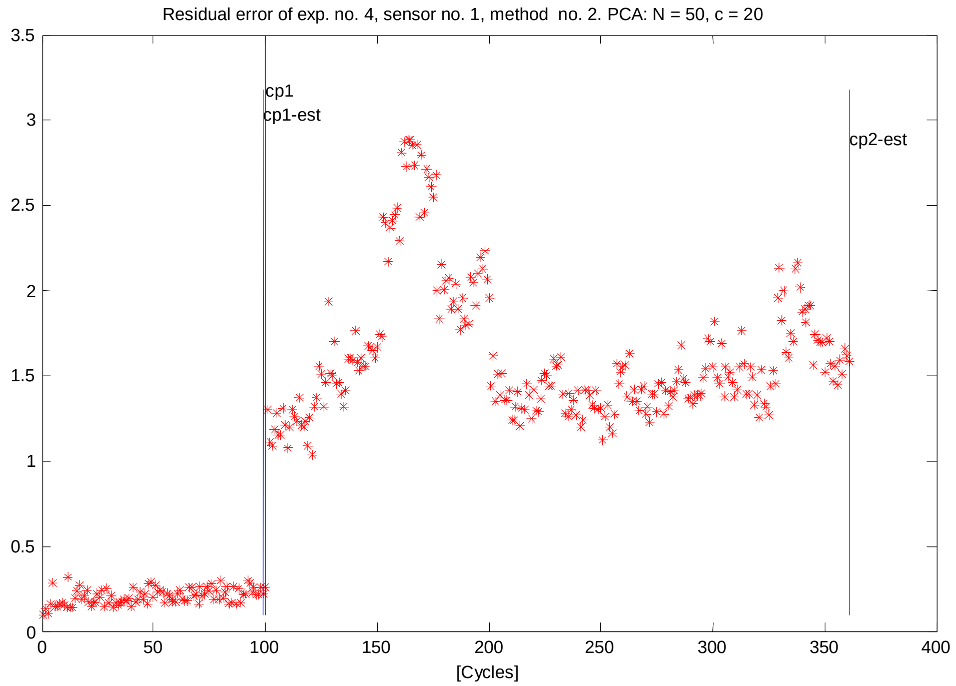

Figure 5.20: Off-line change point estimation of the residual error feature signal from experiment no. 4, sensor no. 1. cp_1 is again almost equal to cp_{true}, but cp_2 is at the very end of the experiment

5.3 Test on re-sampled data

The second main test is basically the same as the first main test, but now the original data is replaced by re-sampled data acquired by re-sampling method no. 5. Since a high number of re-sampled signals are used, the tests are very time consuming. Therefore, only experiment no. 1 from sensor no. 1 is tested. The test results are collected in appendix F.

5.3.1 On-line change detection

The DMD algorithm has two decision functions with a threshold. Thus, it is impossible to make a single ROC curve of the algorithm. However, the two decision functions work together in the way that if just one of them exceeds its threshold, then an alarm is set. Therefore, two ROC curves are created to describe the DMD algorithm performance, one where the mean value decision function is deactivated, and one where the standard deviation decision function is deactivated. The performance of the DMD algorithm, when both decision functions are activated, will then never be worse than when one of the decision functions is deactivated. The two ROC curves describing the DMD algorithm performance are shown in figure 5.21-22. Only cp_1 is tested. The false alarm rate is defined as the rate of samples in the normal condition, which are after cp_{true}, and the true alarm rate is defined in a similar manner. From the figures it can be concluded that the DMD algorithm succeeds in detecting the change.

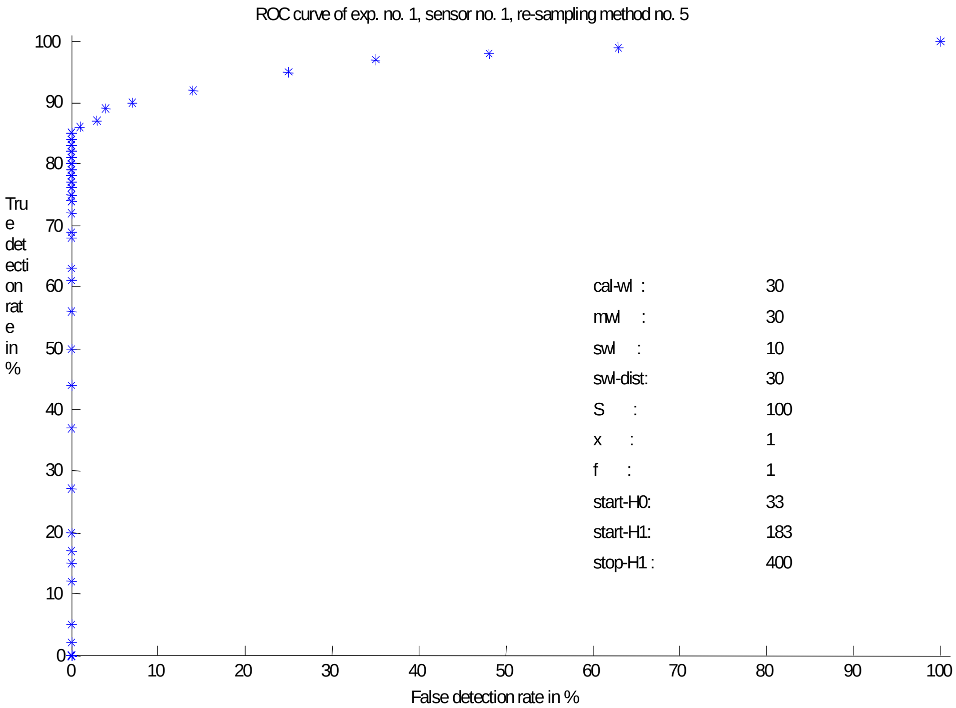

Figure 5.21: ROC curve of the DMD algorithm when the deviation decision function is deactivated.

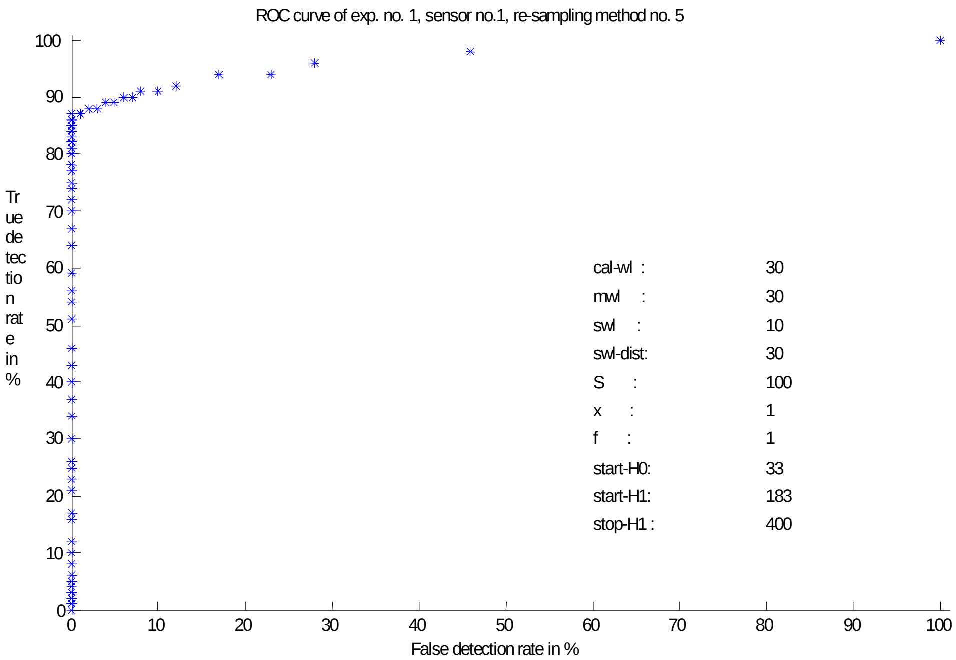

Figure 5.22: ROC curve of the DMD algorithm when the mean value decision function is deactivated.

5.3.2 Off-line hypothesis test

When the first main test was finished, it was experienced that it is only necessary to test for either a change in the mean value or a change in the deviation, not both. The off-line hypothesis test is evaluated on 1,000 re-sampled signals on the same experiment as before with the on-line change detection algorithm. Again two ROC curves are created, one is sweeping the critical value for the log-likelihood ratio when a change in the mean value is assumed, and the other is sweeping the critical value when a change in the deviation is assumed. Figure 5.23-24 show the result. Here the false detection rate is defined as the rate of re-sampled signals, which are estimated to include a change, when they do not include a change, and again the true detection rate is defined in a similar approach. The results are good, since the statements of the off-line hypothesis test at some chosen critical values is 100%. In appendix F, the same test is performed on sensor no. 2, because this feature signal only includes a change in the mean value. As expected, the ROC curve for the off-line hypothesis test is around (0%; 0%), when a change in the deviation is assumed. The off-line hypothesis test is binomial, which means that the deviation is,

$$\sigma = \sqrt{\frac{p(1-p)}{N}}\,, \tag{5.1}$$

where p is the probability and N the number of re-sampled signals. The deviation is plotted in appendix F.

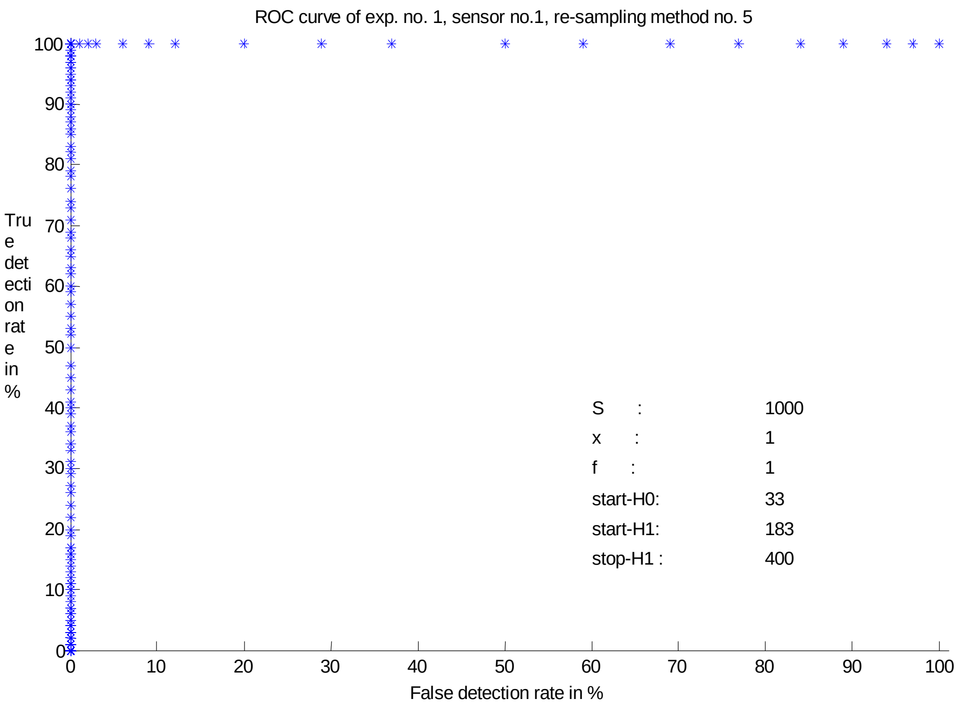

Figure 5.23: ROC curve of the off-line hypothesis test when a change in the mean value only is assumed.

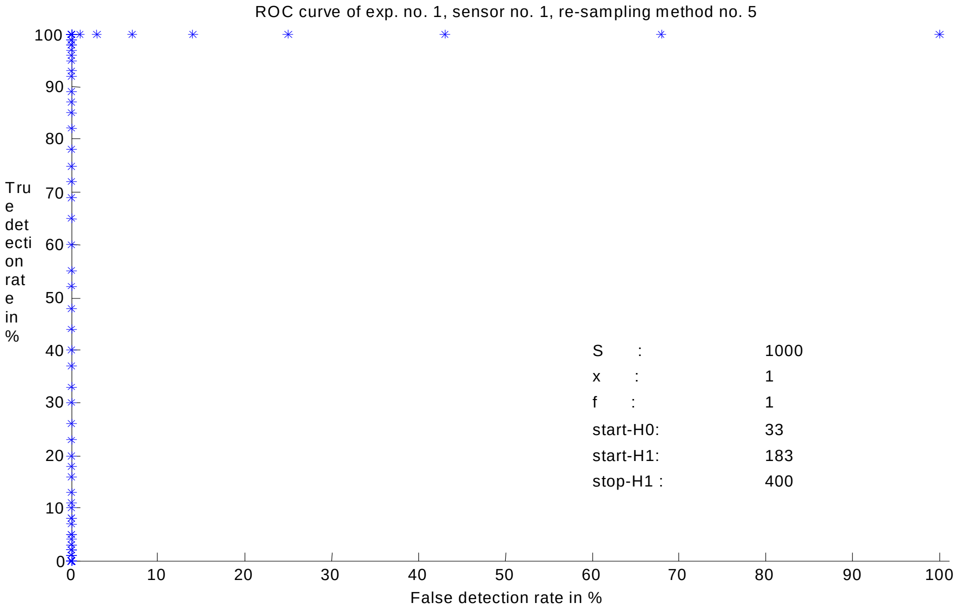

Figure 5.24: ROC curve of the off-line hypothesis test when a change in the standard deviation only is assumed.

5.3.3 Off-line change point estimation

For both methods no. 1 and no. 2, 1,000 re-sampled signals are used to test the off-line change point estimation. The mean value and the deviation of the two change points are calculated. From each method, three figures are created. The first is an example of a re-sampled feature signal, the second is the histogram of the first change point, and the third is the histogram of the second change point. The deviation of the estimated change points is small, but the first estimated change point is quite far from the true change point. This is however because of the re-sampling method, which forces the first cycles of the drift between the two conditions to be more likely to belong to the normal condition. Thus the true change point is shifted to the right. When looking at the re-sampled signal, it is reasonable to conclude that the methods estimate the first change point good.

5.4 Noise test

The last main test is the same as the second main test, but with noise added to the feature signal. Actually, the noise should be added to the raw AE signals, but since they are not present in this work, it was decided to add noise on the feature signal. Three Signal - Noise Ratios (*SNR*) are tested; 0, 20 and 40 dB. The noise is white additive Gaussian noise, which is Gaussian distributed with a zero mean value and the deviation σ_{noise}. From the first 50 cycles of the feature signal the deviation σ_{sig} is estimated. Then the *SNR* is defined as,

$$SNR = 20 \cdot \log \frac{\sigma_{sig}}{\sigma_{noise}}. \tag{5.2}$$

The results are shown in appendix G. The conclusion is that the noise in the tests only affects the change point estimation with a few samples when *SNR* = 0 dB.

5.5 Interfacing the main parts into a final system

From the tests described in this chapter it seems to be feasible to implement the following change detection system:

- Four AE-RMS signals are used as input to the system. The data is converted into the crank angle domain, and each engine cycle is represented by 2,048 samples. All samples are used.
- PCA is applied on the four AE-RMS signals. The first N engine cycles in the normal condition are used to train the PCA-system. Four feature signals are generated, one from each AE sensor. The feature signals begin at cycle $N+1$ and consist of the Squared Error (SE) of the cycles.
- An on-line change detection algorithm is applied on every feature signal. In the calibration period the threshold intervals for the sub-feature signals are determined by the 100th percentile method. The thresholds for the decision functions are set to α times the threshold intervals, where α is chosen by technicians.
- When an on-line change detection algorithm points out a window, in where it believes its feature signal has changed condition, the off-line hypothesis test is performed.
- If the outcome is a success, i.e. a change has occurred, the off-line change point estimation algorithm is performed. Then PCA is applied on the new condition, the on-line change detection algorithms are calibrated and reset, and the process repeats itself.
- If the outcome is a failure, no change point estimation is performed, and the threshold for the decision function causing the alarm is updated. The on-line change detection continues from t_{a1} in order to catch true changes immediately after t_{a2}.

The system proposal is shown as a flow diagram in figure 5.25.

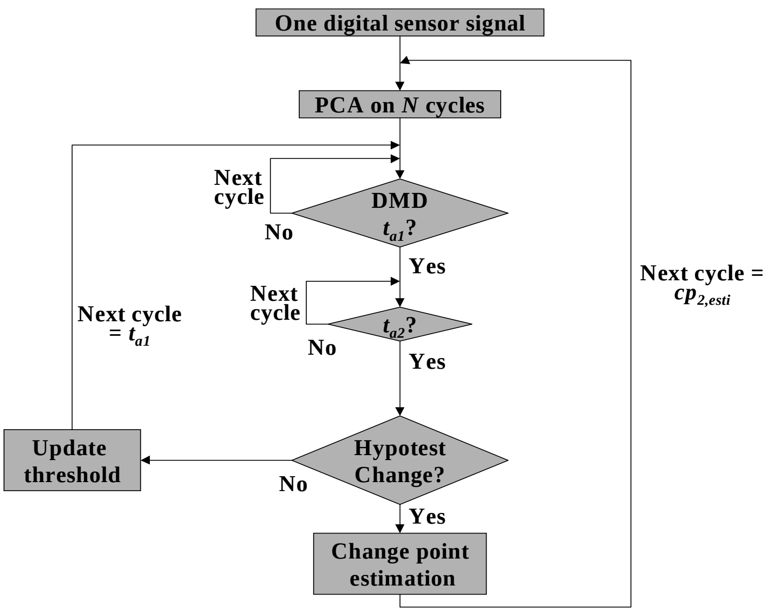

Figure 5.25: Flow diagram for the interfaced automatic condition monitoring system. Only a single sensor signal is applied, since no panel of experts exists in this work.