Vibration-based Damage Assessment for Controller Reconfiguration: Application to an Oilpan

Bart Peeters\textsuperscript{1,}\textsuperscript{a}, Stoyan Kanev\textsuperscript{2,}\textsuperscript{b}, Michel Verhaegen\textsuperscript{2,}\textsuperscript{c} and Herman Van der Auweraer\textsuperscript{1,}\textsuperscript{d}

\textsuperscript{1}LMS International, Interleuvenlaan 68, B-3001 Leuven, Belgium
\textsuperscript{2}Delft Center for Systems and Control, TU Delft, Mekelweg 2, 2628 CD Delft, The Netherlands
\textsuperscript{a}bart.peeters@lms.be, \textsuperscript{b}kanev@ecn.nl, \textsuperscript{c}M.Verhaegen@dcsc.tudelft.nl, \textsuperscript{d}herman.vanderauweraer@lms.be

**Keywords:** Damage detection, Fault detection and diagnosis, Modal analysis, Multiple-model estimation, oilpan.

**Abstract.** One of the objectives of the EU research project InMAR (“Intelligent Materials for Active Noise Reduction”) is to reduce car engine noise by active control. An oilpan of a passenger car serves as a demonstrator. A concern in the application of active control is that the controlled system may change during service life (e.g. due to damage), and hence, may degrade the control performance. This paper presents two vibration-based methods that are able to autonomously detect damage and yield updated experimental models of the structure. A first approach is based on (operational) modal analysis. Based on vibration measurements, the modal parameters of the structure are estimated. The idea is now to automate this process so that, without human intervention, a representative dynamic model of the structure is always available. A second approach uses multiple-model estimation in the case when the state-space models have different state dimensions. To this end, an existing non-interacting multiple-model estimator has been extended to make it alert to jumps from one model to another. Both techniques (“Automatic Modal Analysis” and “Alert Autonomous Multiple Model Estimator”) will be applied to experimental vibration data from an oilpan of a passenger car subjected to damage (loosening of bolts).

**Introduction**

An oilpan in free-free boundary conditions was used to verify the proposed damage assessment procedures (Fig. 1). A single shaker was exciting the oilpan with continuous random excitation in a frequency band of $0 – 1600$ Hz. The shaker force was measured by a force sensor and a single accelerometer was recording the response. Two blocks of 4096 samples were recorded; the sampling frequency being 4096 Hz. A first experiment was conducted using a “healthy” oilpan. This experiment is labeled as S0. Afterwards, 10 damage scenarios were investigated, S1 – S10, with a pair of bolts loosened in each scenario (Fig. 1). Only S5 had just one bolt loosened. For each damage scenario the original vibration experiment was repeated.

**Automatic Modal Analysis**

The first approach to identify a representative model of a possibly changing oilpan relies on the PolyMAX frequency-domain method \cite{1} that is complemented by an automatic procedure \cite{2} to find the stable poles from a stabilization diagram. As it is a frequency-domain method, the Frequency Response Function (FRF) needs to be estimated from the data first. As the data were generated using pure random excitation, the classical FRF estimates will suffer from leakage. Even the application of a Hanning window will not solve this completely and results in biased damping estimates. Therefore, the new so-called Taylor method was used to estimate the FRF. The benefits of this method become clear in Fig. 2. The Taylor FRF peaks are much more reliable. Details on the Taylor method can be found in \cite{3}.
Fig. 3 shows the PolyMAX stabilization diagram for the reference data set S0. The clear stabilization diagram is a typical feature of PolyMAX that makes it rather easy to automate the mode selection process. Fig. 3 also shows the comparison between the measured and synthesized FRF indicating that the identified modal parameters do well represent the data.

Finally, Fig. 4 represents the identified eigenfrequencies for all 1+10 damage scenarios. Apart from mode 6, all other modes were identified in all scenarios. Mode 5 has in some scenarios a higher frequency than in the undamaged reference case. It is speculated that the re-tightening of the loosened bolts may have caused this. The conclusion here is that automatic modal analysis is able to identify a representative model of an oilpan subjected to structural changes.

Fig. 1. Oilpan free-free set-up and indication of the loosened bolts in the 10 damage scenarios.

Fig. 2. FRF estimation: $H_1$ estimate with uniform window, $H_1$ estimate with Hanning window and the new Taylor method estimate [3]. Full band and zoom.

Fig. 3. PolyMAX for automatic modal analysis: stabilization diagram and FRF synthesis.
Fig. 4. Modal-based health monitoring: frequency changes in the different damage scenarios.

**Alert Autonomous Multiple Model Estimator**

Rather than automatically identifying models, the *Alert Autonomous Multiple Model (AMM) Estimator* seeks to autonomously detect which of the previously identified models is “active”. The Alert AMM algorithm updates on a sample-by-sample basis the weights of the different models. The model having the largest weight is assumed to be representative for the current configuration.

**Model set identification.** The available data for each damage scenario are split in two equal parts. The first part is the identification data which are used to identify a model for each damage scenario, resulting in a so-called model set. The second part is the validation data set which will be used afterwards to validate the developed damage detection and diagnosis method. Subspace identification [5] was used to identify state-space models. The quality of the models is assessed by their variance accounted for (VAF) which is defined as:

\[
\text{VAF}(y, \hat{y}) = \left(1 - \frac{\text{var}(y - \hat{y})}{\text{var}(y)}\right) \times 100\%.
\]

where \(y\) is the measured output and \(\hat{y}\) the simulated model output. The variances in Eq. 1 are estimated empirically using the available data. The VAF indices have been computed for both the identification and validation data set and for all damage scenarios. They are represented in Table 1. The state-space model orders (2\textsuperscript{nd} row of Table 1) have been chosen as the smallest order that gives a high VAF (above 98%).

<table>
<thead>
<tr>
<th>Scenario</th>
<th>S0</th>
<th>S1</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
<th>S5</th>
<th>S6</th>
<th>S7</th>
<th>S8</th>
<th>S9</th>
<th>S10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model order</td>
<td>10</td>
<td>8</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>18</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td>VAF id. data [%]</td>
<td>98.9</td>
<td>99.1</td>
<td>99.5</td>
<td>99.4</td>
<td>99.5</td>
<td>99.6</td>
<td>99.5</td>
<td>99.5</td>
<td>98.9</td>
<td>97.8</td>
<td>99</td>
</tr>
<tr>
<td>VAF val. data [%]</td>
<td>99</td>
<td>99.1</td>
<td>98.4</td>
<td>99.2</td>
<td>99.1</td>
<td>99.4</td>
<td>99.3</td>
<td>99.3</td>
<td>99</td>
<td>98.5</td>
<td>98.2</td>
</tr>
</tbody>
</table>

**The Alert AMM Estimator.** Generally speaking, there are two major classes of multiple model (MM) estimators: non-interacting and interacting. The non-interacting MM estimators are based on a bank of filters, running in parallel with no mutual interaction. Each filter is based on one local model of the system, and has its own state that is independent on the states of the other filters. This setup will, hence, also be applicable when the local models can have different state dimensions, or when they have the same state dimensions but lie in different state bases. As discussed in [4], the disadvantage of the non-interacting MM method is that it cannot track changes. In other words, when the model does not change with time, the method is capable of detecting the active model, but
will fail to work properly when the active model (abruptly) changes. The reason for that is the lack of interaction between the filters; the filter states of the filters corresponding to models other than the actual model in effect will therefore be inaccurate and not representative for the true system state. The basic non-interacting MM method is the so-called autonomous MM estimator (AMM).

To improve on that, interacting multiple model (IMM) algorithms have been developed. The difference here is that the filters in the bank are used to compute the global system state as a convex combination of the individual states. This global state is, subsequently, also used in the next time instant to initialize the filters. This makes it possible to track modal changes fast. However, due to the underlying state fusion, the method requires that all local models are of the same order and in the same state-basis. This assumption makes the algorithm, in its basic form, inapplicable to the present situation (see local model orders in Table 1).

In order to use the advantages of the AMM method (models of different order) and those of the IMM methods (alertness to changes), a new scheme is proposed, called alert AMM. It is basically an extension of the conventional AMM approach in that (a) the filter states are reinitialized once a possible change is detected, and (b) instead of the filter outputs, a vector of features based on the model VAFs is used for classification. It should be pointed out that, even though there is a filter re-initialization here, that the new method remains an non-interacting MM method as the filters remain running independently of each other and are not sharing any information. The new algorithm is schematically given in Fig. 5, and is described in detail in [7]. In this paper, the main ideas behind the different algorithm blocks in Fig. 5 will be given.

Fig. 5. Block-schematic representation of the Alert AMM method.

For local model state estimation, a bank of Kalman filters is used. Each filter in the bank runs separately from the rest, and exchanges also no information with the rest. For numerical robustness, the square-root covariance implementation of the Kalman filter is used (see [6] for details). After running the bank of filters, the filter outputs and the measured outputs are used to estimated the VAFs for each model. In order to have reliable VAF estimates, a certain number of samples need to be available, before the online sample-by-sample update of the VAFs can be computed.

The classification block in Fig. 5 has the purpose to classify the VAFs on the basis of some pre-defined values. For that purpose, a matrix of reference VAFs is pre-computed off-line (hence, using the identification data). The so-called cross-VAFs for all model-data combinations are computed and represented in Table 2. In addition to the empirical estimates of the cross-VAFs, their variances are also estimated. Notice, that the VAF and VAF-variance matrices are being precomputed off-line, using the identification data. Online (thus, using validation data), at each sample, the current VAFs are computed. These two are then compared, and the residual is formed, that acts as a feature for classification. Based on these residuals and the pre-computed VAF-variances, the maximum likelihood estimates can be computed. Using these likelihood estimates, the model weights can be
updated at each data sample. All model weights $\mu^{(i)}_k$ ($k$ is the sample number and $i$ the model number) are initialized at $1/N$ ($N$ being the number of models) and also during the sample-by-sample updates, it is ensured that the sum of all weights equals 1.

Table 2. Reference VAF [%] matrix.

<table>
<thead>
<tr>
<th>Model \ scenario</th>
<th>S0</th>
<th>S1</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
<th>S5</th>
<th>S6</th>
<th>S7</th>
<th>S8</th>
<th>S9</th>
<th>S10</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>100</td>
<td>66</td>
<td>-12</td>
<td>38</td>
<td>-20</td>
<td>12</td>
<td>64</td>
<td>68</td>
<td>83</td>
<td>30</td>
<td>27</td>
</tr>
<tr>
<td>1</td>
<td>73</td>
<td>100</td>
<td>30</td>
<td>69</td>
<td>13</td>
<td>9</td>
<td>94</td>
<td>85</td>
<td>82</td>
<td>27</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>24</td>
<td>44</td>
<td>100</td>
<td>36</td>
<td>63</td>
<td>29</td>
<td>35</td>
<td>25</td>
<td>25</td>
<td>27</td>
<td>41</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>46</td>
<td>-38</td>
<td>100</td>
<td>-38</td>
<td>18</td>
<td>48</td>
<td>48</td>
<td>42</td>
<td>52</td>
<td>-31</td>
</tr>
<tr>
<td>4</td>
<td>-3</td>
<td>10</td>
<td>53</td>
<td>19</td>
<td>100</td>
<td>9</td>
<td>18</td>
<td>12</td>
<td>10</td>
<td>2</td>
<td>24</td>
</tr>
<tr>
<td>5</td>
<td>-82</td>
<td>-115</td>
<td>-110</td>
<td>-18</td>
<td>-106</td>
<td>100</td>
<td>-116</td>
<td>-94</td>
<td>-68</td>
<td>80</td>
<td>42</td>
</tr>
<tr>
<td>6</td>
<td>72</td>
<td>94</td>
<td>20</td>
<td>71</td>
<td>20</td>
<td>10</td>
<td>99</td>
<td>82</td>
<td>26</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>70</td>
<td>82</td>
<td>-8</td>
<td>66</td>
<td>1</td>
<td>8</td>
<td>90</td>
<td>100</td>
<td>89</td>
<td>24</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>82</td>
<td>76</td>
<td>-18</td>
<td>57</td>
<td>-22</td>
<td>13</td>
<td>77</td>
<td>88</td>
<td>100</td>
<td>34</td>
<td>17</td>
</tr>
<tr>
<td>9</td>
<td>-78</td>
<td>-112</td>
<td>-170</td>
<td>17</td>
<td>-169</td>
<td>75</td>
<td>-118</td>
<td>-96</td>
<td>-63</td>
<td>100</td>
<td>-23</td>
</tr>
<tr>
<td>10</td>
<td>9</td>
<td>-32</td>
<td>-5</td>
<td>-15</td>
<td>10</td>
<td>66</td>
<td>-39</td>
<td>-23</td>
<td>5</td>
<td>42</td>
<td>100</td>
</tr>
</tbody>
</table>

Finally, the block “change detection” in Fig. 5 has the purpose to make the filter bank alert to changes. To this end, the filters are re-initialized with their initial states and covariances once a change in the active model is detected. Such a model change can easily be detected since it results in decrease of the VAFs of all filters; i.e. with their current states, all filters are incapable of providing correct state estimation. This leads to all online computed VAFs dropping below some threshold value $VAF_{\text{min}}$. If such a drop is detected, then a change detection signal is triggered leading to re-initialization of all filters, and also a reset of the model weights: $\mu^{(i)}_k = 1/N$. Once a change is detected and re-initialization is triggered, the detection scheme is switched off for a number of samples which corresponds to the window length for the computation of the online VAFs. Before this interval elapses, the VAFs will be computed using data from the filters both before and after the re-initialization, which would be inaccurate and should not be used for change detection.

**Experimental results.** The Alert AMM algorithm, presented in the previous section, was applied to the oilpan data, described earlier. To this end, the validation data sets of the different damage scenarios are stacked. This represents a total of 45056 samples (11 scenarios times 4096 samples). Hence, each model is active within some interval of time, which should be detected by the Alert AMM algorithm. This is indeed the case, as can be seen from Fig. 6. The algorithm results in a changing model weight sequence accurately and timely detecting the model in effect. Note that since the model weights at each time instant sum up to one, once one of them is close to one the remaining should be clustered at zero. Note also that the detection delay is never less than about 75 ms (i.e. 300 samples), which corresponds to the waiting time after the re-initialization of the filters triggered by a detected change.

Only the change from damage scenario S6 to S7 has not been detected by the algorithm ($\mu_7$ remains 1, whereas $\mu_8$ should become 1). The reason for that is that the corresponding models are too close to each other in terms of input/output behavior. This can be observed from comparing columns with entry S6 and S7 of the cross-VAF matrix (Table 2).

**Conclusions**

This paper presented two vibration-based methods that are able to autonomously detect damage and yield updated experimental models of the structure. The first approach is based on automatic modal analysis and requires no a priori knowledge of the possibly occurring damage scenarios. In this case, 2 s of data were used to identify a new model. So this time (2 s) can be considered as the detection delay. The second approach is based on the so-called Alert Autonomous Multiple Model Estimator. It requires a priori representative models for the possible damage scenarios, but is afterwards able to detect in a very short time (75 ms) which of these models is currently active.
Acknowledgements

The authors would like to thank Dr. Dirk Mayer from the Fraunhofer Institute for Structural Durability LBF, who organized the oilpan experiments. The work was conducted in the framework of the EC 6-FWP research project NMP2-CT-2003-501084 “INMAR” (Intelligent Materials for Active Noise Reduction, www.lbf.fhg.de/inmar). The support of the EC is gratefully acknowledged.

References


