



Halmstad University Post-Print

## **A self-organized approach for unsupervised fault detection in multiple systems**

Thorsteinn Rognvaldsson, Georg Panholzer, Stefan Byttner and Magnus  
Svensson

*N.B.: When citing this work, cite the original article.*

©2008 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Rognvaldsson T, Panholzer G, Byttner S, Svensson M. A self-organized approach for unsupervised fault detection in multiple systems. In: 19th International Conference on Pattern Recognition: (ICPR 2008) ; Tampa, Florida, USA 8-11 December 2008.. Piscataway, N.J.: IEEE; 2008. p. 1-4.

DOI: <http://dx.doi.org/10.1109/ICPR.2008.4761283>

Copyright: IEEE

Post-Print available at: Halmstad University DiVA  
<http://urn.kb.se/resolve?urn=urn:nbn:se:hh:diva-5022>

# A self-organized approach for unsupervised fault detection in multiple systems

Thorsteinn Rögnvaldsson

*Intelligent Systems Laboratory Halmstad University, Sweden, and  
Learning Systems lab., AASS Örebro University, Sweden  
thorsteinn.rognvaldsson@hh.se*

Georg Panholzer

*Salzburg Research Advanced Networking Center  
Jakob-Haringer-Str. 5/III 5020 Salzburg Austria  
georg.panholzer@salzburgresearch.at*

Stefan Byttner

*Intelligent Systems Laboratory, Halmstad University,  
Box 823, 301 18 Halmstad, Sweden  
stefan.byttner@hh.se*

Magnus Svensson

*Volvo Technology, 405 08 Göteborg, Sweden  
magnus.svensson@volvo.com*

## Abstract

*An approach is proposed for automatic fault detection in a population of mechatronic systems. The idea is to employ self-organizing algorithms that produce low-dimensional representations of sensor and actuator values on the vehicles, and compare these low-dimensional representations among the systems. If a representation in one vehicle is found to deviate from, or to be not so similar to, the representations for the majority of the vehicles, then the vehicle is labeled for diagnostics. The presented approach makes use of principal component coding and a measure of distance between linear subspaces. The method is successfully demonstrated using simulated data for a commercial vehicle's engine coolant system, and using real data for computer hard drives.*

## 1. Introduction

Operators of commercial vehicle fleets generally try to maximize the uptime of their vehicles while minimizing the cost. This means, among other things, that the

maintenance of a vehicle should be scheduled with as long intervals as possible while the risk for break-down is minimized. To achieve this, service intervals should be based on the health status of the vehicle rather than on static mileage intervals, which requires that the status of the vehicle is monitored continuously (or at least frequently) so that break-downs are avoided.

The traditional approach for doing health status based fault detection and diagnosis on vehicles is to design software for this prior to the production of the vehicle. This means expensive experiments to look for characteristics of possible faults, of which many will never occur on the actual vehicle. On-board computers on vehicles are not very powerful and it is impossible to provide on-board error detection for all possible errors on a vehicle even if software would exist for this; only the most important malfunctions are monitored. It therefore happens that unexpected parts break on the vehicles once they are in production. This leads to customer dissatisfaction and a depreciation of the brand value, why it is important that such problems are detected early, preferably before the customers detect them.

This paper presents a methodology for a fault detection system that does not require prior experiments to

define fault characteristics, and that is small enough to be run on current on-board computers in real time. It is a telematic based fault detection scheme that enables on-board fault detection by collecting data on a fleet of vehicles, doing data mining in real time and finding errors by looking for deviations from the norm, where the norm is defined dynamically from the population of vehicles, not in a static way from experiments in a lab. Self-organizing methods are used to create low-dimensional representations (models) of subsystems in a vehicle. These low-dimensional representations are compared between a large group (a fleet) of similar vehicles. If a model is found to deviate compared to the fleet group of models, then the vehicle is judged to need diagnostics for that subsystem. This diagnostics can be done, e.g., by downloading (via the telematics gateway) diagnostics software to the vehicle.

## 2 Method

The full method can be split into two separate steps:

- i. Look for interesting relationships among sensor values and control signals on the vehicle, and encode these relationships in a suitable model.
- ii. Compare the models from the vehicles and look for deviations. This is done in the back office with a global perspective, with knowledge of the models from every vehicle.

The first step requires a method for generating models and a measure of “interestingness” that does not require a global perspective (i.e. any knowledge of other vehicles). It must be done in an on-line fashion, where the model is built from a stream of data that is not saved, and it must not require large computing resources. The second step requires a metric for comparing models and a method for generating a statistical model that can be used to flag “abnormal” models.

### 2.1 Self-organized data encoding (i)

We use the principal components encoding for the first part. It is linear (hence fast), well known with a hoard of theoretical results, and there are several methods available for on-line PCA. We have in two previous papers suggested linear methods for the first part, i.e. fitting linear models with different variable subsets and evaluating them based on their generalization performance (estimated on the system) [1, 2]. We explore a different approach here where we instead of selecting individual variables select projections. We use the standard PCA method where we compute eigenvalues

$\lambda_i$  and eigenvectors  $\mathbf{v}_i$  for the data covariance matrix  $\mathbf{C}$ , i.e.

$$\mathbf{C}\mathbf{v}_i = \lambda_i\mathbf{v}_i. \quad (1)$$

All measured variables are standardized to zero mean and unit standard deviation before the PCA is done.

### 2.2 Comparison of linear subspaces (ii)

The PCA method yields a set of orthonormal eigenvectors  $\{\mathbf{v}_i\}$  for each system. These sets are compared using Krzanowski’s angle based similarity measure [3]

$$\begin{aligned} S(A, B) &= \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^M \cos^2 \theta_{ij}^{AB} \\ &= \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^M (\mathbf{v}_i^A \cdot \mathbf{v}_j^B)^2, \end{aligned} \quad (2)$$

where  $\mathbf{v}_i^A$  and  $\mathbf{v}_j^B$  are the eigenvectors for system  $A$  and system  $B$ , respectively, and  $M$  is the number of eigenvectors used in each set. The  $S(A, B)$  measure lies between zero and one; the value is one for completely overlapping subspaces and zero for completely orthogonal subspaces. The  $S(A, B)$  is related to the recently introduced “subspace distance” [6, 5] measure  $D(A, B)$  through

$$D^2(A, B) = M - MS(A, B). \quad (3)$$

The subspace distance  $D(A, B)$  is a proper distance, i.e. it fulfills the triangle equality etc. [6, 5].

We assume that we have access to data from a set  $\Omega_R$  of vehicles that are mostly fault-free, which we denote the “reference set”. We then pick bootstrap samples from this set, emulating that we score one system in the reference set against a set of size  $N$  of the other systems in the reference set (bootstrap sampling is used to reduce the effect of possibly faulty systems in the reference set). The mean  $\mu$  and standard deviation  $\sigma$  of the similarity is computed. This is repeated several (100) times so that we get a set  $\Omega_{BS}$  of mean and standard deviations for the similarities within the reference set. The covariance matrix  $\Sigma$  is then computed for the set  $\Omega_{BS}$  and the average Mahalanobis distance

$$d(A) = \frac{1}{|\Omega_R|} \sum_{k \in \Omega_R} (\mathbf{x}^A - \mathbf{x}^k)^T \Sigma^{-1} (\mathbf{x}^A - \mathbf{x}^k) \quad (4)$$

for each system from the reference set is used as a measure for its “strangeness”. The sum runs over all systems in the reference set, except system  $A$  if it is a member of the reference set.

The distribution of the “strangeness” measure  $d$  for all systems in the reference set is computed, assuming

that it is normally distributed. A one-sided test is then applied to the measure, flagging a system as deviating if the probability is less than 5% for having a strangeness measure above the observed value.

There are two free parameters: the number of principal components to use in the comparison ( $M$  in equation 2) and the cut-off for the “strangeness” deviation (which is 5% in the description above). Clearly, if all principal components are used, then there is no difference between the systems (since the subspaces overlap completely). However, if few principal components are used then there might not be enough information to detect deviations. One can, e.g., choose the number of principal components that explains 95% of the variance. This, however, can vary very much between systems. We use an alternative method where a large number of principal components are computed and sent to the back office. Comparisons are then done using  $M = 1, 2, 3, \dots$  principal components (ordered in magnitude of the eigenvalues) and it is observed how often a system is flagged as deviating. If a system is flagged as deviating in more than half the cases (e.g. in 2 out of 3 cases when  $M = 1, 2$  and 3) then it is finally flagged as deviating, otherwise it is flagged as normal.

The “strangeness” deviation cut-off is set to 5% in the engine coolant examples. For the hard drive data it varied between 0.0001% and 1% to show how the risk for false alarm behaves with the detection ability. The cut-off value controls the false alarm rate (it is essentially the false alarm rate).

### 3 Data

We have tested the proposed method on two simulated problems related to vehicle fault detection on buses. We have also tested the method on a previously published data set for detecting computer hard drives that are about to break down, to compare our methods performance to previously published results.

An engine cooling system was simulated with a leaking pipe and different levels of clogged radiator for a commercial vehicle. The vehicles are from one of three different weight classes (15 tons, 18 tons and 23 tons) and three different ambient temperatures (10°C, 20°C and 30°C). There are 12 normal cases (with different ambient temperatures and weights). Every simulation has 1,600 consecutive observations (corresponding to 1,600 seconds of data) with 51 variables such as fan speed, cooling water and oil temperature, engine speed and other sensor and actuator values that are considered possibly relevant for the cooling system. There are 27 faulty systems for the leakage case, with different severity levels. There are 9 faulty systems for the clogged

**Table 1. Number of systems flagged as strange (within brackets is the number of principal components used)**

	Reference set	Fault set
Sim. leakage (7)	0/12	18/27
Sim. leakage (9)	0/12	23/27
Sim. leakage (11)	1/12	27/27
Sim. clogged radiator (7)	0/12	0/9
Sim. clogged radiator (9)	0/12	2/9
Sim. clogged radiator (11)	1/12	3/9

radiator cases, with three different severity levels.

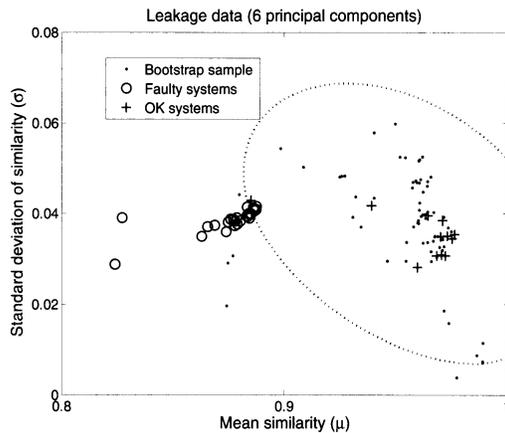
The SMART hard drive data was introduced by Murray et al. [4] in 2005 and is thoroughly described by them. The task is to flag a drive as faulty before it breaks down, without too many false alarms. We modified it somewhat and required 60 consecutive observations (equal to 120 hours of operation) from each hard drive. Covariances were computed for sets of 24 consecutive observations (equal to 48 hours of operation). The data set therefore has 178 normal drives and 86 faulty drives. The hard drive serial number and constant attributes were removed, which left 59 attributes. No further variable selection was done.

## 4 Results and discussion

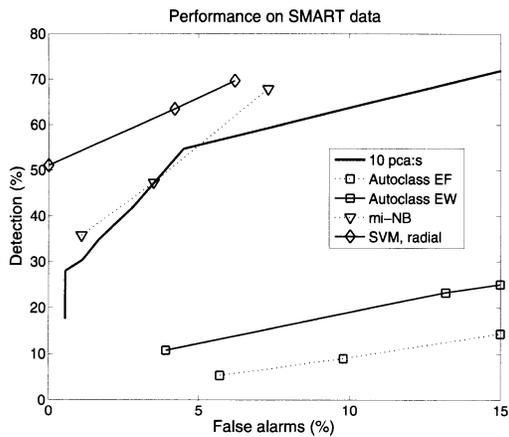
The detection and false alarm performance for the engine coolant systems data are shown in Table 1, for different number of principal components (using the voting scheme described earlier).

The false alarm rate for the coolant system is below 5%, which is to be expected since the deviation cut-off is 5%. The detection rate is almost 100% for the leakage data but less than 33% for the clogged radiator. The leakage faults produce a much clearer deviation. Fig. 1 illustrates how different the leakage systems are from the reference group, when plotted in the  $(\mu, \sigma)$  plane, and it is easy to detect the difference. The clogged radiator is difficult because there is an interaction between the weight of the vehicle and the clogged radiator. A heavy vehicle with non-clogged radiator behaves similar to a light vehicle with a clogged radiator.

The results on the hard drive failure data are shown in Fig. 2. Each drive is represented by multiple measurements (varying between 2 and 12) and we classify the drives using the multiple instance assumption: an individual drive is classified as faulty if at least one of these measurements is classified as faulty. This is the same procedure as used by Murray et al. [4]. Our un-



**Figure 1.** The  $\mu$  and  $\sigma$  for the normal systems (+) and the faulty systems (o). The dots (.) are the bootstrap sample values and the almost elliptical line shows the boundary between “normal” and “faulty” when a cut-off of 5% is used.



**Figure 2.** Our method’s performance on the hard drive failure data set (solid line without any markers). The reference results are from Fig. 4 in [4] and show the performance of a self-organized clustering approach (Autoclass) and two supervised approaches (Support Vector Machine and Multi Instance Naive Bayes).

supervised method with principal components and subspace distances performs better than earlier unsupervised approaches and on par with one supervised (or semi-supervised) approach, without any variable selection.

## 5 Conclusion

The results show that it is possible to use a fairly straightforward self-organized representation of sensor and actuator values on mechatronic systems and compare these, in an unsupervised and on-line fashion, to detect deviating (faulty) systems among a fleet (group) of systems, or changes in behavior for a system over time. The method only needs a reference group and a (possibly long) list of interesting variables, which is easy to get from the engineers that designed the system. The method we proposed is based on linear PCA on board the vehicles, which is sufficiently uncomplicated to be implemented on an embedded computer on-board a vehicle (e.g. a bus). It is easy to generalize to non-linear PCA with kernels but this is not at all straightforward to implement in an on-board computer (due to much larger memory requirements). The method is adaptive and can therefore learn and adapt during the lifetime of the systems. We believe that an adaptive method like this, which builds a health soft-sensor in a self-organized and unsupervised way, has the potential to be commonplace in the monitoring of tomorrow’s mechatronic systems.

## References

- [1] S. Byttner, T. Rögnavaldsson, and M. Svensson. Modeling for vehicle fleet remote diagnostics. Technical paper 2007-01-4154, Society of Automotive Engineers (SAE), 2007.
- [2] S. Byttner, T. Rögnavaldsson, and M. Svensson. Self-organized modeling for vehicle fleet based fault detection. Technical paper 2008-01-1297, Society of Automotive Engineers (SAE), 2008.
- [3] W. J. Krzanowski. Between-groups comparison of principal components. *Journal of the American Statistical Association*, 74(367):703–707, 1979.
- [4] J. F. Murray, G. F. Hughes, and K. Kreutz-Delgado. Machine learning methods for predicting failures in hard drives: A multiple-instance approach. *Journal of Machine Learning Research*, 6:783–816, 2005.
- [5] X. Sun, L. Wang, and J. Feng. Further results on the subspace distance. *Pattern Recognition*, 40:328–329, 2007.
- [6] L. Wang, X. Wang, and J. Feng. Intrapersonal subspace analysis with application to adaptive bayesian algorithm for face recognition. *Pattern Recognition*, 38(4):617–521, 2005.