# A Signal-Based Fault Detection and Classification Strategy with Application to an Internal Combustion Engine

R. Ahmed[1], S. A. Gadsden[1], M. El Sayed[1], S. R. Habibi[1], and J. Tjong[2]
[1]McMaster University, Hamilton, Ontario, Canada and [2]Ford Canada
E-mail: ryan.ahmed85@gmail.com

*Abstract*-**Fault detection strategies are important for ensuring the safe and reliable operation of mechanical and electrical systems. Recently, a new signal-based fault detection and classification strategy has been proposed, which makes use of artificial neural networks (NNs) and the smooth variable structure filter (SVSF). The strategy, referred to as the NN-SVSF, has shown promising results with applications to benchmark classification problems. New developments of the SVSF have resulted in improved performance in terms of state and parameter estimation. These developments are used to enhance the NN-SVSF in an effort to further advance the signal-based strategy. This paper studies and compares the results of applying other popular strategies on an internal combustion engine (ICE), for the purposes of fault detection and classification.**

## INTRODUCTION

The Kalman filter (KF) is the most popular state estimation tool. It provides optimal estimates for known linear systems in the presence of Gaussian white noise. In the case of nonlinear systems, the extended Kalman filter (EKF) is applied by linearizing the system around the latest state estimate at each time interval. An EKF-based neural network training technique was first introduced by Singhal and Wu in 1989 [1]. The EKF provides a powerful neural network training capability compared to conventional first-order gradient algorithms [2]. In literature, the EKF has been extensively applied for training of both feed-forward [3] and recurrent networks [4,5] in both a global form (GEKF) or in a decoupled form (DEKF). Although the EKF demonstrates a close performance compared to second-order derivative, batch-based methods, it can outperform them by avoiding local minima problems [2]. Accordingly, the EKF represents an efficient and practical alternative to second-order training methods.

Through the last decade, various enhanced artificial neural networks (ANNs) training techniques have been proposed in several studies. A new hybrid learning algorithm that combines the EKF and particle filter (PF) has been presented in [6]. The new training scheme provides faster speed of convergence than the stand-alone EKF. An advanced EKF training technique has been proposed in [7]. The advanced form of Kalman filter-based parameter estimation method obtains a more accurate estimate of how a Gaussian distribution evolves under a nonlinear transformation. It has proven to offer performance advantages over standard EKF training. Reference [8] provides suggestions on how to initialize the EKF parameters in addition to presenting a new decoupling strategy that reduces the update rate of the error covariance matrix for faster training. Recently, Wan et al. [9] reported the effective use of the unscented Kalman filter (UKF) [10] for feed-forward neural networks training.

The recently proposed smooth variable structure filter (SVSF) provides a robust dynamic adaptation, high-rate of convergence, and can guarantee estimation stability for bounded uncertainties and noise levels [11]. The SVSF has been successfully used for parameter and state estimation [12,13]. The SVSF has demonstrated some advantages over the EKF in target tracking applications with respect to computational complexity, robustness, and tracking accuracy [14]. This is due to the sensitivity of the EKF to model uncertainties when used as a parameter estimator. Therefore, a combined variable structure and Kalman filtering approach for parameter estimation has been proposed in [15]. A comparison between the EKF, SVSF, PF, and UKF on a bearing-only target tracking problem was demonstrated in [16]. The results demonstrated that the SVSF yielded accurate state estimates while maintaining robustness to uncertainties.

Recently, a new form of the SVSF has been introduced which offers a more accurate estimation method [17]. This strategy minimizes the SVSF state error covariance with respect to the smoothing boundary layer term. The strategy is referred to as the SVSF with a time-varying (or variable) smoothing boundary layer, or SVSF-VBL. The resulting algorithm is combined with the neural network training method to create a new signal-based fault detection and clasification strategy.

The purpose of this paper is to compare the performances of the EKF-based, SVSF-based, and SVSF-VBL-based neural network training techniques. These three strategies are applied and tested on accelerometer measurements obtained from an internal combustion engine setup. The following section provides an overview of the estimation methods. The SVSF-based neural network training strategy is shown in Section III. The main simulation results are shown in Section IV, followed by a brief discussion. Section V conclueldes the main findings of this paper.

978-1-4673-1408-4/12/$31.00 ©2012 IEEE

ESTIMATION STRATEGIES

### A. Kalman Filter (KF)

In 1960, Rudolph Kalman presented a new approach to linear filtering and prediction problems, which would later become known as the Kalman filter (KF) [18]. The KF yields a statistically optimal solution for linear estimation problems in the presence of Gaussian noise. The KF is a model based method, derived in the time domain and a discrete-time setting. A continuous-time version was developed by Kalman and Bucy, and is consequently referred to as the Kalman-Bucy filter [19]. Like many other filters, the KF is formulated in a predictor-corrector manner. The states are first estimated using the system model and input, termed as a priori estimates, meaning 'prior to' knowledge of the observations. A correction term is then added based on the innovation (also called residuals or measurement errors), thus forming the updated or a posteriori (meaning 'subsequent to' the observations) state estimates. The following five equations form the core of the KF algorithm, and are used in an iterative fashion. Equations (2.1) and (2.2) define the a priori state estimate $\hat{x}_{k+1|k}$ based on knowledge of the system A and previous state estimate $\hat{x}_{k|k}$, and the corresponding state error covariance matrix $P_{k+1|k}$, respectively.

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + Bu_k \quad (2.1)$$

$$P_{k+1|k} = AP_{k|k}A^T + Q_k \quad (2.2)$$

The Kalman gain $K_{k+1}$ is defined by (2.3), and is used to update the state estimate $\hat{x}_{k+1|k+1}$ as shown in (2.4). The gain makes use of an innovation covariance $S_{k+1}$, which is defined as the inverse term found in (2.3).

$$K_{k+1} = P_{k+1|k}C^T\left(CP_{k+1|k}C^T + R_{k+1}\right)^{-1} \quad (3.3)$$

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\left(z_{k+1} - C\hat{x}_{k+1|k}\right) \quad (3.4)$$

The a posteriori state error covariance matrix $P_{k+1|k+1}$ is then calculated by (2.5), and is used iteratively, as per (2.2).

$$P_{k+1|k+1} = (I - K_{k+1}C)P_{k+1|k} \quad (2.5)$$

The derivation of the KF is well documented, with details available in [18,20,21]. The optimality of the KF comes at a price of stability and robustness. The KF assumes that the system model is known and linear, the system and measurement noises are white, and the states have initial conditions with known means and variances [22]. However, the previous assumptions do not always hold in real applications. If these assumptions are violated, the KF yields suboptimal results and can become unstable [23]. Furthermore, the KF is sensitive to computer precision and the complexity of computations involving matrix inversions [24]. For nonlinear systems and measurements, the KF may be used to formulate the extended Kalman filter (EKF). In this case, the nonlinear system $f$ or measurement $h$ is linearized according to its Jacobian. Partial derivatives are used to compute linearized system and measurement matrices F and H, respectively found as follows [25]:

$$F_k = \left.\frac{\partial f}{\partial x}\right|_{\hat{x}_{k|k},u_k} \quad (2.6)$$

$$H_{k+1} = \left.\frac{\partial h}{\partial x}\right|_{\hat{x}_{k+1|k}} \quad (2.7)$$

Equations (2.6) and (2.7) essentially linearize the nonlinear system or measurement functions around the current state estimate [20]. This comes at a loss of optimality, as the KF gain is no longer considered to be the best solution to the estimation problem [21]. Note that the EKF process is the same as the KF process, except that (2.6) and (2.7) replace $A$ and $C$, respectively.

### B. Smooth Variable Structure Filter (SVSF)

The smooth variable structure filter (SVSF) was presented in 2007 [11]. The SVSF strategy is also a predictor-corrector estimator based on sliding mode concepts, and can be applied on both linear or nonlinear systems and measurements. As shown in Fig. 1, it utilizes a switching gain to converge the estimates to within a boundary of the true state values (i.e., existence subspace) [11]. The SVSF has been shown to be stable and robust to modeling uncertainties and noise, when given an upper bound on the level of un-modeled dynamics and noise [11,26].

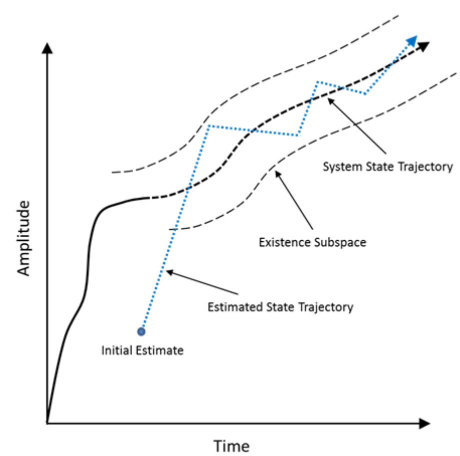


Fig. 1. The above figure shows the SVSF estimation strategy [17]. Starting from some initial value, the state estimate is forced by a switching gain to within a region referred to as the existence subspace.

The origin of the SVSF name comes from the requirement that the system is differentiable (or 'smooth') [11,27]. Furthermore, it is assumed that the system under consideration is observable [11]. Consider the following process for the SVSF estimation strategy, as applied to a nonlinear system with a linear measurement equation. The predicted state estimates $\hat{x}_{k+1|k}$ are first calculated as follows:

$$\hat{x}_{k+1|k} = \hat{f}\left(\hat{x}_{k|k}, u_k\right) \quad (2.8)$$

Utilizing the predicted state estimates $\hat{x}_{k+1|k}$, the corresponding predicted measurements $\hat{z}_{k+1|k}$ and measurement errors $e_{z,k+1|k}$ may be calculated:

$$\hat{z}_{k+1|k} = C\hat{x}_{k+1|k} \quad (2.9)$$

$$e_{z,k+1|k} = z_{k+1} - \hat{z}_{k+1|k} \quad (2.10)$$

Next, the SVSF gain is calculated as follows [11]:

$$K_{k+1}^{SVSF} = C^{+}\left(\left|e_{z,k+1|k}\right| + \gamma\left|e_{z,k|k}\right|\right) \circ sat\left(\frac{e_{z,k+1|k}}{\psi}\right) \quad (2.11)$$

The SVSF gain is a function of: the a priori and a posteriori measurement errors $e_{z,k+1|k}$ and $e_{z,k|k}$; the smoothing boundary layer widths $\psi$; the 'SVSF' memory or convergence rate $\gamma$ with elements $0 < \gamma_{ii} \leq 1$; and the linear measurement matrix $C$. The SVSF gain is used to refine the state estimates as follows:

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}^{SVSF} \quad (2.12)$$

Next, the updated measurement estimates $\hat{z}_{k+1|k+1}$ and corresponding errors $e_{z,k+1|k+1}$ are calculated:

$$\hat{z}_{k+1|k+1} = C\hat{x}_{k+1|k+1} \quad (2.13)$$

$$e_{z,k+1|k+1} = z_{k+1} - \hat{z}_{k+1|k+1} \quad (2.14)$$

The SVSF process may be summarized by (2.8) through (3.14), and is repeated iteratively. According to [11], the estimation process is stable and convergent if the following condition is satisfied:

$$\left|e_{k|k}\right| > \left|e_{k+1|k+1}\right| \quad (2.15)$$

The proof, as described in [11] and [27], yields the derivation of the SVSF gain from (2.15). The SVSF results in the state estimates converging to the state trajectory. Thereafter, it switches back and forth across the state trajectory within a region referred to as the existence subspace. The existence subspace represents the amount of uncertainties present in the estimation process, in terms of modeling errors or the presence of noise. The width of the existence space $\beta$ is a function of the uncertain dynamics associated with the inaccuracy of the internal model of the filter as well as the measurement model, and varies with time [11]. Typically this value is not exactly known but an upper bound may be selected based on a priori knowledge.

### C. *Smooth Variable Structure Filter with a Time-Varying Smoothing Boundary Layer (SVSF-VBL)*

The partial derivative of the a posteriori covariance (trace) with respect to the smoothing boundary layer term $\psi$ is the basis for obtaining a strategy for the specification of $\psi$. The approach taken is similar to determining an optimal gain for the KF. Previous forms of the SVSF included a vector form of $\psi$, which had a single smoothing boundary layer term for each corresponding measurement error. Essentially, the boundary layer terms were independent of each other such that the measurement errors would only directly be used for calculating its corresponding gain. The coupling effects are not explicitly considered thus preventing an optimal derivation. A 'near-optimal' formulation of the SVSF could be created using a vector form of $\psi$, however this would lead to a minimization of only the diagonal elements of the state error covariance matrix. In an effort to obtain a smoothing boundary layer equation that yields optimal state estimates for linear systems (like the KF), a full smoothing boundary layer matrix was proposed in [17]. Hence, consider the full matrix form of the smoothing boundary layer:

$$\psi = \begin{bmatrix} \psi_{11} & \psi_{12} & \cdots & \psi_{1m} \\ \psi_{12} & \psi_{22} & \cdots & \psi_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \psi_{m1} & \psi_{m2} & \cdots & \psi_{mm} \end{bmatrix} \quad (2.16)$$

Note that the off-diagonal terms of (2.16) are zero for the standard SVSF, whereas this is not the case for the algorithm presented in [17]. This definition includes terms that relate one smoothing boundary layer to another (i.e., off-diagonal terms). To solve for a time-varying smoothing boundary layer (VBL) based on (2.16), consider:

$$\frac{\partial\left(trace\left[P_{k+1|k+1}\right]\right)}{\partial\psi} = 0 \quad (2.17)$$

The complete proof and derivation for the SVSF-VBL is provided in [17]. For the purposes of this paper, only the algorithm will be summarized here. Consider the prediction stage for a linear system and measurement (as an example), where the state estimates and covariance are first calculated as per (2.18) and (2.19), respectively. The following 12 equations summarize the SVSF-VBL strategy.

$$\hat{x}_{k+1|k} = \hat{A}\hat{x}_{k|k} + \hat{B}u_k \quad (2.18)$$

$$P_{k+1|k} = \hat{A}P_{k|k}\hat{A}^T + Q_k \quad (2.19)$$

The a priori measurement estimate (2.20) and errors (2.21) are then calculated.

$$\hat{z}_{k+1|k} = C\hat{x}_{k+1|k} \quad (2.20)$$

$$e_{z,k+1|k} = z_{k+1} - \hat{z}_{k+1|k} \quad (2.21)$$

The update stage is then defined by the following sets of equations. The innovation covariance (2.22) and combined error vector (2.23) are calculated, and then used in (2.24) to determine the smoothing boundary layer matrix. Note that a 'divide by zero' check should be performed on (2.23) to avoid inversion of zero in (2.24). This can be accomplished using a simple if statement with a very small threshold (i.e., $1 \times 10^{-12}$).

$$S_{k+1} = CP_{k+1|k}C^T + R_{k+1} \quad (2.22)$$

$$A_{k+1} = \left|e_{z,k+1|k}\right|_{Abs} + \gamma\left|e_{z,k|k}\right|_{Abs} \quad (2.23)$$

$$\psi_{k+1} = \left(\bar{A}_{k+1}^{-1}CP_{k+1|k}C^T S_{k+1}^{-1}\right)^{-1} \quad (2.24)$$

The SVSF gain is then calculated (2.25), and then used to update the state estimates (2.26).

$$K_{k+1} = C^{-1}\bar{A}_{k+1}\psi_{k+1}^{-1} \quad (2.25)$$

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}e_{z,k+1|k} \quad (2.26)$$

Finally, the a posteriori state error covariance (2.27), updated measurement estimate (2.28), and a posteriori errors (2.29) are calculated.

$$P_{k+1|k+1} = (I - K_{k+1}C)P_{k+1|k}(I - K_{k+1}C)^T + K_{k+1}R_{k+1}K_{k+1}^T \quad (2.28)$$

$$\hat{z}_{k+1|k+1} = C\hat{x}_{k+1|k+1} \quad (2.29)$$

$$e_{z,k+1|k+1} = z_{k+1} - \hat{z}_{k+1|k+1} \quad (2.30)$$

The SVSF-VBL estimation strategy is summarized by (2.18) through (2.30).

## SVSF-Based Neural Network Training

The SVSF can be applied for training nonlinear feed-forward neural networks by estimating network weights. In the same fashion as the Kalman filter, the SVSF has been adapted to train feed-forward neural networks by visualizing the network as a filtering problem where $F$, $G$, and $C_k$ are the system, input, and output matrices, respectively follows:

$$\widehat{w}_{k+1|k} = F\widehat{w}_{k|k} + Gu_k \tag{3.1}$$

$$y_k = C_k(w_{k|k}, u_k) \tag{3.2}$$

The global SVSF training algorithm is iterative and is summarized by the following steps, assuming training data set $\{x_k, z_k\}$:

Step 1: Network weights initialization.

A-priori state estimates (network weights) $\widehat{w}_{k|k}$ are randomly initialized ranging from -1, 1.

Step 2: Calculation of the predicted (a-posteriori) weight estimates $\widehat{w}_{k+1|k}$ from (3.1).

For neural networks training, the system matrix $F$ is an identity matrix and the system input $u_k$ is set to zero. Consequently, when the algorithm is initialized, the a-posteriori weight matrix is the same as the a-priori one and thus (3.1) is rewritten as follows:

$$\widehat{w}_{k+1|k} = \widehat{w}_{k|k} \tag{3.3}$$

Step 3: Jacobian Matrix calculation (Linearization) of the measurement matrix $C_k$.

The Jacobian matrix is calculated here. After applying the algorithm, $C_{k|linearized}$ is obtained.

Step 4: Network's actual output (measurements) $\hat{z}_{k+1|k}$ calculation.

By multiplying the linearized Jacobian measurement matrix $C_{k|linearized}$ with the a-priori network weights $\widehat{w}_{k+1|k}$ one has the following:

$$\hat{z}_{k+1|k} = C_{k|linearized}\widehat{w}_{k+1|k} \tag{3.4}$$

Step 5: measurement error $e_{z_{k+1|k}}$ calculation.

Using the output $\hat{z}_{k+1|k}$ and the corresponding target (from the neural network training data set) $z_k$, measurement errors $e_{z_{k+1|k}}$ may be calculated as follows:

$$e_{z_{k+1|k}} = z_k - \hat{z}_{k+1|k} \tag{3.5}$$

Step 6: SVSF gain calculation.

The SVSF gain is a function of the a-priori and the a-posteriori measurement errors $e_{z_{k+1|k}}$ and $e_{z_{k|k}}$, the smoothing boundary layer widths $\psi$, the 'SVSF' memory or convergence rate $\gamma$, as well as the linear measurement matrix $C_{k|linearized}$. For the derivation of the SVSF gain $K_{k+1}$, refer to [11,28]. The SVSF gain is defined as a diagonal matrix such that:

$$K_{k+1} = C_{k|linearized}{}^{+} diag\left[\left(\left|e_{z_{k+1|k}}\right| + \gamma\left|e_{z_{k|k}}\right|\right) \circ sat\left(\frac{e_{z_{k+1|k}}}{\psi}\right)\right] diag\left(e_{z_{k+1|k}}\right)^{-1} \tag{3.6}$$

Step 7: Calculation of the updated state estimates $\widehat{w}_{k+1|k+1}$.

$$\widehat{w}_{k+1|k+1} = \widehat{w}_{k+1|k} + K_{k+1}e_{z_{k+1|k}} \tag{3.7}$$

Step 8: Calculation of a-posteriori output estimate $\hat{z}_{k+1|k+1}$ and measurement errors $e_{z_{k+1|k+1}}$ to be used in later iterations:

$$\hat{z}_{k+1|k+1} = C_{k|linearized}\widehat{w}_{k+1|k+1} \tag{3.8}$$

$$e_{z_{k+1|k+1}} = z_{k+1} - \hat{z}_{k+1|k+1} \tag{3.9}$$

Steps 3 to 8 are iteratively repeated while shuffling (randomly shifting) the training data set each epoch.

Training proceeds until one of the stopping conditions occurs. Note that the SVSF-based NN steps may also be applied on the standard EKF and SVSF-VBL equations, but have been omitted due to page constraints.

## EXPERIMENTAL SETUP AND RESULTS

The experimental setup as shown in Fig. 2 involves a four stroke, 5.0 L, eight cylinder engine. The test is performed at FORD's Powertrain Engineering Research and Development Centre (PERDC). Vibration data has been recorded over four seconds using a charge-type piezoelectric accelerometer. The accelerometer has been attached to the engine lug in a premeditated position in order to detect faults of interest. Vibration data has been acquired using a PROSIG 5600 data acquisition system with built-in 16-bit analog-to-digital convertor card set at a sampling frequency of 32,768 Hz.

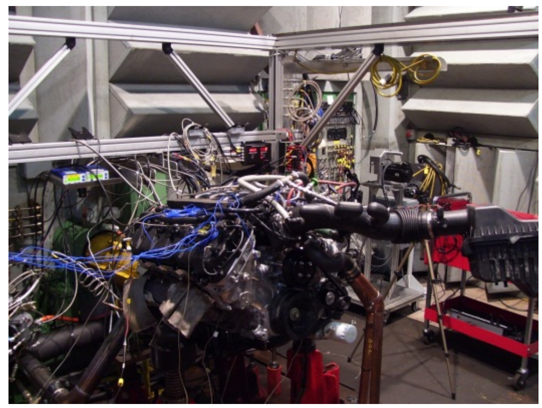

Fig. 2. The above figure shows the experimental setup in a semi-anechoic chamber. This setup was used to generate the data used in this paper.

After data acquisition, the time domain data has been converted offline to the crank angle domain using the cam identification (CID) sensor signal. The CID sensor is used to detect camshaft angle position. It is a non-contact sensor mounted on the engine and generates sinusoidal pulses at specific angles of $90^{\circ} - 120^{\circ} - 60^{\circ} - 120^{\circ} - 60^{\circ} - 180^{\circ} - 90^{\circ}$ per engine cycle. The first sinusoidal pulse zero-crossing indicates that the first cylinder is 10° away from the top-dead-center (TDC). After transformation to the crank angle domain, data resampling is performed so that each engine cycle has the same number of points.

Two faults have been induced in the engine. They involve missing bearing fault (MB) and piston chirp (PC) fault. PC faults occur due to dislocation of the engine's piston ring which leads to excessive wear and high engine noise. MB faults occur due to assembly problem throughout the manufacturing process. MB faults cause severe vibration spike as shown in Fig. 3. Vibration signals recorded from these two fault cases as well as the baseline fault-free engine case are used as a training data set for neural networks training. A piston chirp signal reading is provided in Fig. 4.

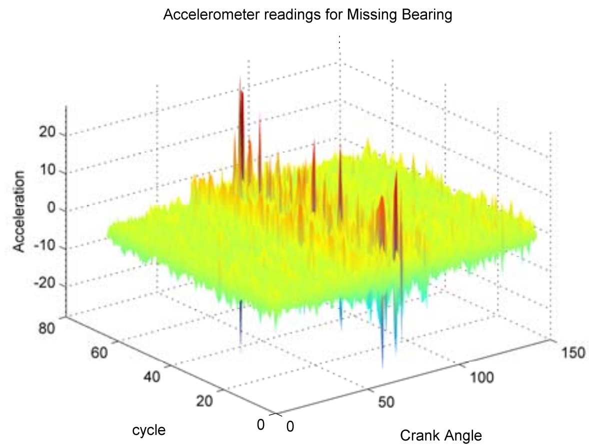


Fig. 3. The above figure shows the accelerometer readings for the missing bear fault case.

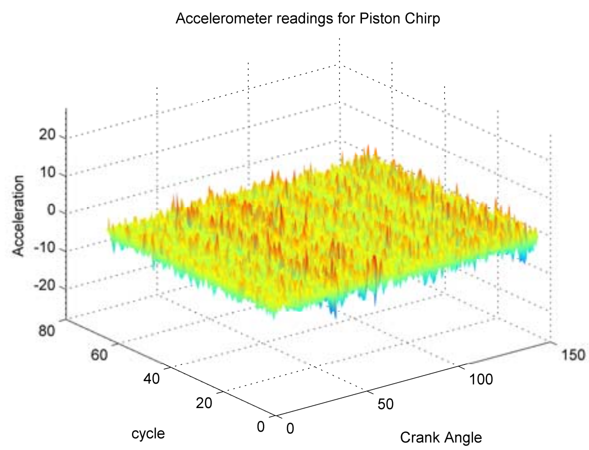


Fig. 4. The above figure shows the accelerometer readings for the piston chirp fault case.

In this paper, fully connected feed-forward multilayer perceptron with a number of input neurons representing sampled vibration data input in the crank angle domain, two hidden layers with four neurons each, and three output units is used. Trained network should be able to classify engines to either one of the two induced faults or to a baseline (fault-free) case as follows: (1, 0, 0: Baseline engine), (0, 1, 0: Piston Chirp fault detected), (0, 0, 1: Missing Bearing fault detected). The test has been conducted through several runs, 30 engine cycles from each case stated before (i.e.: Baseline, MB, and PC) resulting in 90 training sets. Trained neural networks have been tested using 10 engine cycles from each case resulting in 30 testing set.

As mentioned earlier, networks were trained using the EKF, SVSF, and SVSF-VBL algorithms. Figure 5 shows the root mean square error (RMSE) results for 30 epochs. Notice that the SVSF-VBL method converged faster than the standard SVSF. However, after about $4-5$ epochs both methods yield the same RMSE.

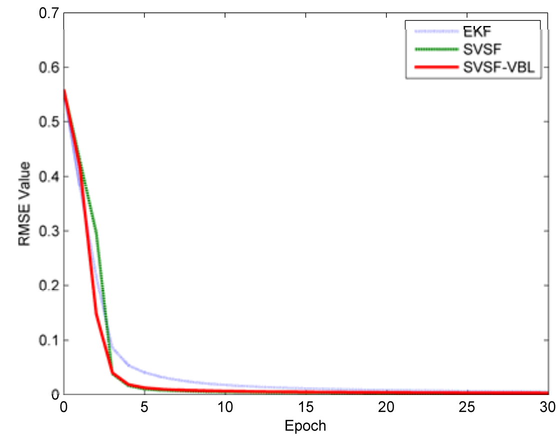


Fig. 5. The above figure illustrates the RMSE value at each epoch for the EKF, SVSF, and the SVSF-VBL. Note how after about epoch 20, all three filters provide roughly the same estimation result in terms of accuracy.

Note that for fault detection and classification, the RMSE values are not as valuable as the actual classification percentages. Trained networks have been tested using 30 data sets. Training and testing results are summarized into confusion matrices as shown below.

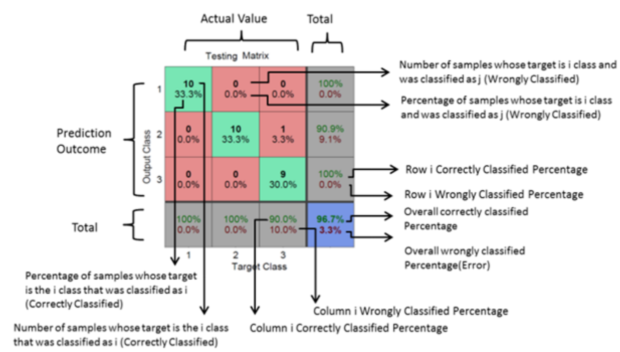


Fig. 6. Explanation of the 'confusion' matrix (appropriately named).

The following three fiures are the corresponding confusion matrices for the EKF, SVSF, and SVSF-VBL estimation strategies.

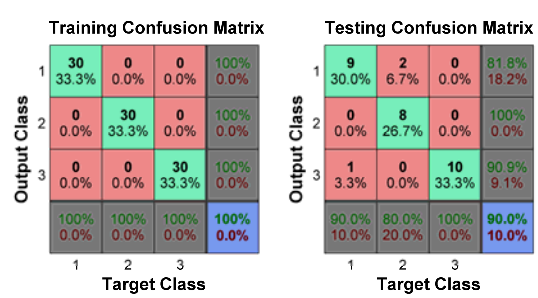


Fig. 7. The above illustration represents the training and confusion matrices for the EKF estimation strategy.

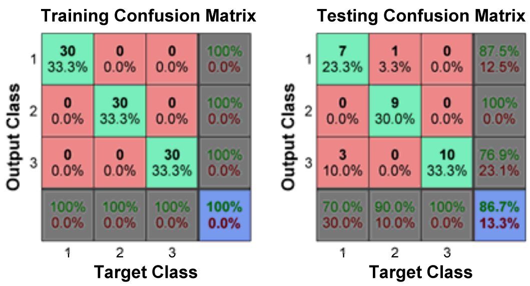


Fig. 8. The above illustration represents the training and confusion matrices for the SVSF estimation strategy.

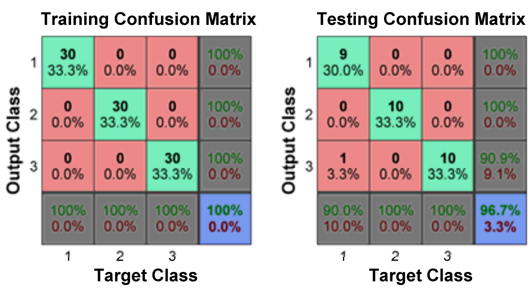


Fig. 9. The above illustration represents the training and confusion matrices for the SVSF-VBL estimation strategy.

Training and testing results are summarized in the following table. The SVSF-VBL achieved the highest testing (generalization) percentage (least mean squared error) in both training and testing, followed by the EKF, and the standard SVSF.

TABLE I
OVERALL TRAINING AND TESTING CLASSIFICATION RESULTS

| Training Technique | Training (%) | Testing (%) |
|---|---|---|
| SVSF-VBL | 100 | 96.7 |
| EKF | 100 | 90.0 |
| SVSF | 100 | 86.7 |

Interestingly, the SVSF-VBL method only misclassified one fault, compared with the standard SVSF which misclassified four faults. The improvement is most likely due to the introduction of the time-varying boundary layer.

## CONCLUSIONS

The purpose of this paper was to introduce a new form of the SVSF with a time-varying boundary layer as a method for training neural networks. The EKF, SVSF, and SVSF-VBL strategies were applied for fault detection and classification on an internal combustion engine. As demonstrated in the results, the SVSF-VBL yielded the highest testing classification percentage, with only one misclassified fault. Future work will look at expanding this method and applying it to a number of other fault conditions.

## REFERENCES

[1] S. Singhal and L. Wu, "Training multilayer perceptrons with the extended Kalman algorithm," *Advances in Neural Information Processing Systems 1*, pp. 133–140, 1989.

[2] Simon Haykin, *Kalman filtering and neural networks*, 3rd ed.: Prentice Hall, 2001.

[3] G.V. Puskorius and L.A. Feldkamp, "Decoupled extended Kalman filter training of feedforward layered networks," *Proc. IJCNNÕ91 I, Seattle*, pp. 771-777, 1991.

[4] R.J. Williams, "Training recurrent networks using the extended Kalman filter," *Proc. IJCNN'92 IV*, pp. 241-246, 1992.

[5] Pu Sun and Marko Kenneth, "Training recurrent neural networks for very high performance with the extended Kalman algorithm," *Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 8, pp. 121-126, 1998.

[6] Xiaolong Deng, Xie Jianying, Weizhong Guo, and Jun Liu, "A new learning algorithm for diagonal recurrent neural network," in *First International Conference on Natural Computation*, 2005, pp. 44-50.

[7] L.A. Feldkamp, T.M. Feldkamp, and D.V. Prokhorov, "Neural network training with the nprKF ," in *Proceedings of the International Joint Conference on Neural Networks*, 2001 , pp. 109-114.

[8] Felix Heimes, "Extended Kalman Filter Neural Network Training: Experimental Results and Algorithm Improvements," *IEEE*, vol. 0-7803-4778-1/98, 1998.

[9] E. A. Wan and R. van der Merwe, "The Unscented Kalman Filter for Nonlinear Estimation," in *Proceedings of the IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium.*, 2000, pp. 153-158.

[10] S. J. Julier, J.K. Uhlmann, and H. F. Durrant-Whyte, "A New Approach for Filtering Nonlinear Systems," in *Proceedings of the American Control Conference*, 1995, pp. 1628-1632.

[11] S. R. Habibi, "The Smooth Variable Structure Filter," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 1026-1059, 2007.

[12] S.R. Habibi and Burton R., "Parameter identification for a high-performance hydrostatic actuation system using the variable structure filter concept," *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME*, vol. v 129, n 2, pp. 229-235.

[13] S. Wang, Burton R., and S. R. Habibi, "A smooth variable structure filter for state estimation," *Control and Intelligent Systems*, vol. v 35, n 4, pp. 386-393, 2007.

[14] S. A. Gadsden and S. R. Habibi, "Target tracking using the smooth variable structure filter," in *ASME Dynamic Systems and Control Conference*, 2009, DSCC2009.

[15] S. R. Habibi, "A Combined Variable Structure and Kalman Filtering Approach," *American Control Conference*, 2008.

[16] S.A. Gadsden, D. Dunne, S.R. Habibi, and T. Kirubarajan, "Comparison of extended and unscented Kalman, particle, and smooth variable structure filters on a bearing-only target tracking problem," *Signal and data processing of small targets* , August 2009.

[17] S. A. Gadsden, "Smooth Variable Structure Filtering: Theory and Applications," Department of Mechanical Engineering, McMaster University, Hamilton, Ontario, Ph.D. Thesis 2011.

[18] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering, Transactions of ASME*, vol. 82, pp. 35-45, 1960.

[19] R. Kalman and R. Bucy, "New Results in Linear Filtering and Prediction Theory," *ASME Journal of Basic Engineering*, vol. 83, pp. 95-108, March 1961.

[20] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.

[21] A. Gelb, *Applied Optimal Estimation*. Cambridge, MA: MIT Press, 1974.

[22] D. Simon, *Optimal State Estimation: Kalman, H-Infinity, and Nonlinear Approaches*.: Wiley-Interscience, 2006.

[23] S. J. Julier, J. K. Ulhmann, and H. F. Durrant-Whyte, "A New Method for Nonlinear Transformation of Means and Covariances in Filters and Estimators," *IEEE Transactions on Automatic Control*, vol. 45, pp. 472-482, March 2000.

[24] M. S. Grewal and A. P. Andrews, *Kalman Filtering: Theory and Practice Using MATLAB*, 3rd ed. New York: John Wiley and Sons, Inc., 2008.

[25] S. Haykin, *Kalman Filtering and Neural Networks*. New York, U.S.A.: John Wiley and Sons, Inc., 2001.

[26] S. R. Habibi and R. Burton, "The Variable Structure Filter," *Journal of Dynamic Systems, Measurement, and Control (ASME)*, vol. 125, pp. 287-293, September 2003.

[27] M. Al-Shabi, "The General Toeplitz/Observability SVSF," Department of Mechanical Engineering, McMaster University, Hamilton, Ontario, Ph.D. Thesis 2011.

[28] S. A. Gadsden and S. R. Habibi, "A New Form of the Smooth Variable Structure Filter with a Covariance Derivation," in *IEEE Conference on Decision and Control*, Atlanta, Georgia, 2010.