# A Machine Learning Approach for Big Data in Oil and Gas Pipelines

Abduljalil Mohamed, Mohamed Salah Hamdi
Information Systems Department
ABMMC
Doha, Qatar
{ajmaoham, mshamdi}@abmmc.edu.qa

Sofiène Tahar
Electrical and Computer Engineering Department line
Concordia University
Montreal, Quebec, Canada
tahar@ece.concordia.ca

***Abstract*— Experienced pipeline operators utilize Magnetic Flux Leakage (MFL) sensors to probe oil and gas pipelines for the purpose of localizing and sizing different defect types. A large number of sensors is usually used to cover the targeted pipelines. The sensors are equally distributed around the circumference of the pipeline; and every three millimeters the sensors measure MFL signals. Thus, the collected raw data is so big that it makes the pipeline probing process difficult, exhausting and error-prone. Machine learning approaches such as neural networks have made it possible to effectively manage the complexity pertaining to big data and learn their intrinsic properties. We concentrate, in this work, on the applicability of artificial neural networks in defect depth estimation and present a detailed study of various network architectures. Discriminant features, which characterize different defect depth patterns, are first obtained from the raw data. Neural networks are then trained using these features. The Levenberg-Marquardt back-propagation learning algorithm is adopted in the training process, during which the weight and bias parameters of the networks are tuned to optimize their performances. Compared with the performance of pipeline inspection techniques reported by service providers such as GE and ROSEN, the results obtained using the method we proposed are promising. For instance, within ±10% error-tolerance range, the proposed approach yields an estimation accuracy at 86%, compared to only 80% reported by GE; and within ±15% error- tolerance range, it yields an estimation accuracy at 89% compared to 80% reported by ROSEN.**

***Keywords-big data; neural networks; machine learning; pipeline inspection; magnetic flux leakage***

## I. INTRODUCTION

Natural gas and crude oil production is usually carried through long-distance transmission metallic pipelines. Due to the nature of environment and extreme temperature, metallic pipelines are subjected to corrosion, which is considered as a leading cause of pipeline defects. In [1], it has been reported that nearly 30% of pipeline defects are due to external corrosion. These pipeline defects can result in huge financial losses, damage to the environment, and loss of life. Thus, pipeline operators are required to utilize effective and efficient intelligent tools to detect and locate pipeline defects. Efficient intelligent tools utilize Magnetic Flux Leakage (MFL) signals and ultrasonic waves and use them to detect and localize defect types (e.g., corrosion, cracks, dents, etc.). MFL recordings around the center of a metal-loss defect do have a distinct pattern of behavior. The sensor passing directly above the defect center has highest amplitude of the axial and radial components of the MFL signal. The amplitude of these components gets lower for sensors further away from the defect center. Using the MFL measurements of the neighborhood sensors, the type and size of a defect can be determined. In the literature, several techniques have been proposed for the purpose of detecting and localizing pipeline defects [2]. In [3], using MFL signals, Artificial Neural Networks (ANNs) are used to classify signal patterns of various kinds of defects. These defects were manufactured and deliberately implanted. The ANN was able to distinguish between defect and non-defect signals with great accuracy (94.2%). For a particular type of defect signals, the ANN recognized them 92.5% of the time. In [4], a fuzzy artificial neural network-based approach is proposed for reliability assessment of oil and gas pipelines. The actual condition of aging pipelines vulnerable to metal loss corrosion are characterized by eight pipe parameters as input variables obtained from MFL signals. The proposed method uses these parameters to estimate the probability of failure of aging pipelines vulnerable to corrosion. In [5], a recognition and classification of pipe cracks using images analysis and neuro-fuzzy algorithm is proposed. Crack-related features are first extracted. The combination of a fuzzy membership function, used to absorb variation of feature values, and a back-propagation network, with learning ability, shows good classification efficiency. In [6], a Radial Basis Function Neural Network (RBFNN) is deemed to be a suitable technique and a corrosion inspection tool to recognize and quantify the corrosion characteristics. An Immune RBFNN (IRBFNN) algorithm is proposed to process the MFL data to determine the location and size of the corrosion spots on the pipeline.

There is a relationship between the amplitude and the area under the curve of the MFL signal and the depth of the corresponding defect. However, this relationship is not well-understood and cannot be analytically described. To further compound the problem, numerous MFL sensors are usually used to scan the pipelines; this in reality results in large amount of data collected by these sensors. Hence, visual inspection may not be an adequate approach to analyze such big data. Therefore, neural networks seem an appealing alternative and are proven capable of learning this relationship. To the best of our knowledge, neural networks have only been used to detect and localize

978-1-4673-8103-1/15 $31.00 © 2015 IEEE
DOI 10.1109/FiCloud.2015.54

pipelines defects. In this work, the use of artificial neural networks (ANNs) for estimating pipeline defects is evaluated, using MFL signals obtained an intelligent pig.

## II. PIPELINE DEFECT DETECTION AND SIZING

### A. MFL-based Pipeline Inspection

Non-destructive pipeline inspection uses autonomous devices. Travelling through the pipelines propelled by product flow, these devices, called pigs, use special sensors that are capable of detecting any magnetic flux leakage. This method of pipeline inspection is known as MFL scanning, where the measured data are in the form of MFL signals. The resolution of the acquired data is determined by the number of sensors arranged around the pipeline circumference. Theoretically, MFL scanning can be simply explained as follows [7]. Using two magnets of opposite polarity, magnetic field is first introduced on the surface of a pipeline. The lines of the magnetic force (magnetic flux) normally traverse through the pipeline walls from the south pole to the north pole. However, in the existence of a crack or thinning, two new poles emerge around the edges of these defects and the air gap between the two new edges forces some flux to leak out. An increase in flux leakage indicates metal loss. The strength and shape of the leaking flux can help estimate defect properties such as depth, width, etc.

### B. MFL Data is Big Data

As shown in Figure 1, if we roll-out the wall of the targeted pipeline, the sensors, placed on the inspecting autonomous device, are equally spaced along the x-axis. To effectively cover the whole circumference of the pipeline, a large number of sensors are utilized. Moreover, to effectively cover the whole length of the pipeline (usually spreads out into hundreds of kilometers), the sensors measure the MFL signals every three millimeters along the axis of the pipeline. This, of course, results in a tremendous amount of raw data that makes the manual inspection process futile. To alleviate the difficulties accompanying the analysis process of such big data, the data dimensionality is first reduced through suitable feature extraction techniques. Furthermore, the obtained features are fed into machine learning tools such as neural networks to learn the intrinsic relationship between MFL signals and their corresponding defect depths.

### C. Metal-Loss Detection and Sizing Using Wavelet-based Techniques

Wavelets [8, 9, 10] techniques are successfully used in numerous applications such as data compression [11], classification [12], and de-noising [13, 14, 15]. Wavelets techniques can also be applied for metal-loss detection in pipelines [16 ]. Given a certain defect, corresponding MFL signals may have a specific form at the center of the defect. The MFL signal $B(x)$, shown in Figure 2, is sampled from a pipeline, and represents three cubical defects. It mainly consists of three components $B_x$, $B_y$ and $B_z$. As can be seen from Figure 2, the three defects have different extents

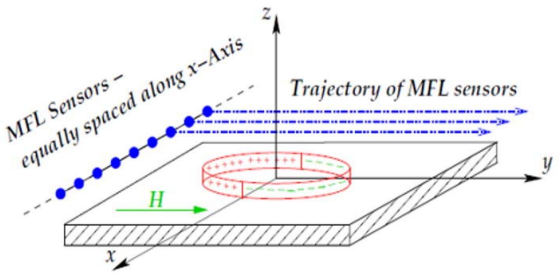


Figure 1. Rolled-out representation of pipeline wall

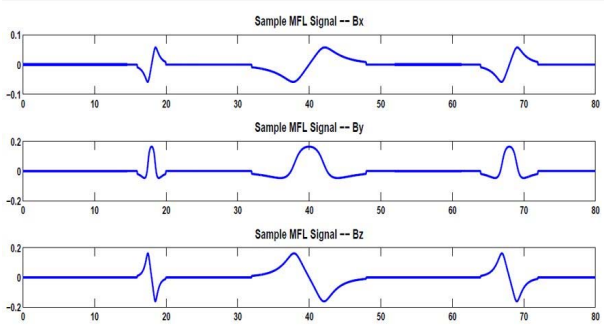


Figure 2. A sampled MFL signal representing three cubical defects

along the x-axis of the pipeline. Each component of the MFL signal consists of a sum of curves. Each curve is a translated and dilated copy of a reference shape. If a mother wavelet, $\psi(x)$ is assigned to this reference shape, an orthonormal wavelet basis $\langle \psi_{j,k}(x) \rangle$ can be derived from it. Then, the MFL signal can be expressed as follows:

$$B(x) = \sum_{j,k} c_{j,k} \psi_{j,k}(x)$$

The wavelet transform of the MFL is computed, with respect to the basis $\langle \psi_{j,k}(x) \rangle$, in order to determine a pipeline defect and estimate its extent. The set of non-zero coefficients, $c_{j,k}$ indicates the positions and factors of the reference shape, which are used to locate the defect on the pipeline and specify its extent.

### D. Machine Learning-based Techniques for Defect-Depth Estimation

Once defects have been detected and located using wavelet techniques, an equally important problem is estimating their depths. Considering the large amount of data obtained by hundreds of MFL sensors, obtaining meaningful defect depth-related properties is by no means an easy task and hard to accomplish using traditional approaches. Thus, machine learning techniques can play a significant role as they are capable of extracting such properties through iterative training from large data. While most machine learning-based techniques reported in the literature have

addressed the defect localization problem, little work has been dedicated for defect depth estimation. In [17], a support vector machine and principal component analysis are used in defect depth estimation. In this work, the neural networks are utilized to learn the intrinsic properties between certain defect depths and the shapes of their corresponding MFL signals.

## III. NEURAL NETWORK-BASED APPROACH FOR ESTIMATING DEFECT DEPTHS

Along with the length of the defect, its depth is a very important factor for determining its severity. According to industry standards, some defects, based on their depths, may be considered completely safe, while others are deemed too severe. It has been observed that the magnitude of MFL signals is much higher for defects with larger depths. The relationship, however, between defect depths and the magnitude of MFL signals is not well understood and cannot be analytically described. Therefore, machine learning techniques can be used to capture this relationship. In this paper, we investigate the application of Artificial Neural Networks (ANNs) as a learning tool and propose an ANN-based approach for estimating failure depths. The structure diagram of the proposed approach is shown in Figure 3.

As shown in Figure 3, a feature set is first extracted from the MFL signals obtained from a set of sensors. These features should distinctly characterize different patterns of different failure depths. These features are then used in the learning process of the neural network.

### *A. Feature Extraction*

Feeding raw MFL signals directly into the neural network may prolong the learning task and lead to unsatisfactory results. Instead, a number of features that characterize the MFL signals is computed. Hence, we calculate five statistical features for each component of the MFL signal, namely, maximum magnitude, peak-to-peak distance, integral of the normalized signal, mean average, and standard deviation. To obtain more features, we approximate the MFL signals by polynomial series of the form:

$$a_n X^n + \ldots + a_1 X + a_0$$

To find the best MFL approximation, the values of the polynomial coefficients $a_n + \ldots + a_0$ are optimized. These coefficients are then extracted and used, along with the statistical features, to train the neural network. It has been found that polynomials of degree three provide the best approximation for the $B_x$ component, whereas polynomials of degree six provide the best approximation for both the $B_y$ and $B_z$ components. In total, we have an input vector consisting of thirty-three features.

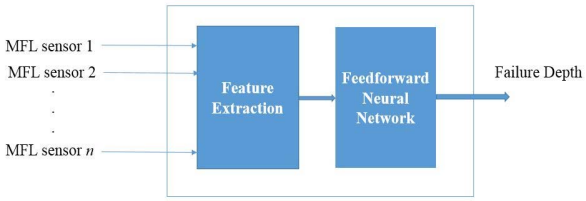


Figure 3. The main components of the proposed ANN technique

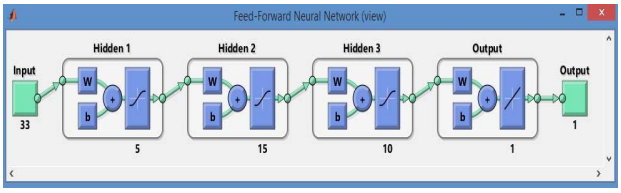


Figure 4. Architecture of static FFNN architecture

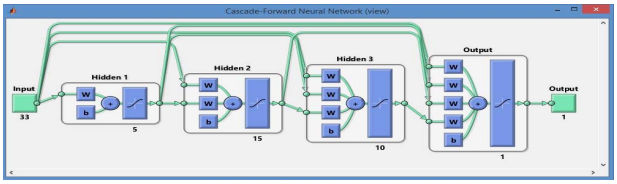


Figure 5. Architecture of the cascaded FFNN

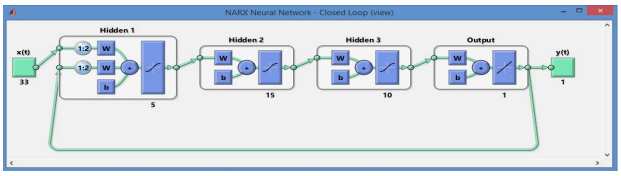


Figure 6. Architecture of the dynamic FFNN

### *B. Neural Network Architecture and Parameters*

We examine three architectures, namely static, cascaded, and dynamic Feed-Forward Neural Networks (FFNN).

*1) Static FFNN*

The architecture of the static FFNN is shown in Figure 4. The extracted feature vector is fed into the first hidden layer. Weight connections, based on the number of neurons in each layer, are assigned between every adjacent layers.

*2) Cascaded FFNN*

As shown in Figure 5, these networks are similar to feed-forward networks, but include a weight connection from the input layer to each other layer, and from each layer to the successive layers.

*3) Dynamic FFNN*

In dynamic networks as shown in Figure 6, the network outputs depend not only on the current input feature vector, but also on the previous inputs and outputs of the network.

*4) Neural Network Parameters*

In this study, different network parameters are examined including the number and size of hidden layers, the type of

the transfer functions, and the type of the performance functions. As for the opted learning algorithm, it has been shown that the Levenberg-Marquardt back-propagation algorithm provides the best performance for function approximation; and hence it is more suitable than other learning algorithms for defect depth estimation [18, 19].

## IV. PERFORMANCE EVALUATION

The main performance measure used to evaluate a given network structure and configuration is the estimation accuracy of the failure depth within a certain level of error-tolerance. The error-tolerance levels used in this study are ±1%, ±5%, ±10%, ±15%, ±20%, ±25%, ±30%, ±35%, and ±40%. For each network structure, the FFNN is experimented with different numbers of hidden layers, each varies in size from 10 neurons up to 100 neurons. The results of the experimental work are reported in the following subsections.

### *A. Performance and Transfer Functions*

The performance function is the first parameter examined as it plays a crucial role in the accuracy and speed of network learning. Two performance functions are selected:

(1) Mean Squared Error (MSE)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left|\hat{Y}_i - Y_i\right|^2$$

(2) Sum Squared Error (SSE)

$$SSE = \sum_{i=1}^{n}\left|\hat{Y}_i - Y_i\right|^2$$

The effect of the MSE performance function is first examined on a static FFNN with a single hidden layer and a different number of neurons. The network performance is tested on two types of transfer functions used in neurons of the hidden layer:

(1) Log sigmoid

$$\log sig = \frac{1}{(1-e^{-x})}$$

(2) Hyperbolic Tangent sigmoid

$$\log sig = \frac{2}{(1+e^{-2x})} - 1$$

Figure 7 shows the results obtained by the static FFNN with MSE as a performance function and Log-sigmoid as a transfer function. It can be noted from the figure that, for ±1% error-tolerance level, the defect depth estimation accuracy obtained by the network is approximately 10.29% with 50 neurons in the hidden layer. Naturally, as the error-tolerance increases, the network performance starts getting better. For instance, for the ±5% error-tolerance level, the defect depth estimation accuracy obtained by the network, with 70 neurons in the hidden layer, is around 47.55%. For the ±10%, ±15%, ±20%, ±25%, ±30%, ±35%, ±40% levels, the network best estimation performances are 64.71%, 79.41%, 83.33%, 88.24%, 90.69%, 90.69%, 91.18% with 60, 20, 20, 10, 10, 80, and 80 neurons in the hidden layer, respectively.

Figure 8 shows the results obtained by the static FFNN with SSE as a performance function and Log-sigmoid as a transfer function. For the ±1%, ±5%, ±10%, ±15%, ±20%, ±25%, ±30%, ±35%, ±40% levels, the network best estimation performances are 10.78%, 37.25%, 65.69%, 76.47%, 82.84%, 89.22%, 92.16%,93.63%, 94.61% with 10, 90, 90, 40, 40, 100, 100, 70, and 100 neurons in the hidden layer, respectively.

Tables I and II show the comparison of results obtained by the static FFNN using different transfer and performance functions. For both performance functions (MSE and SSE), both transfer functions show very close performance results. However, the difference in hidden neurons is notable. For example, at ±1% level error-tolerance, an FFNN with the Log-sig transfer and MSE performance functions needs only 10 neurons to obtain an accuracy of 10.29%, while with the Tan-sig transfer function, an FFNN needs 70 neurons to obtain 9.80% accuracy. At ±5% error-tolerance level, the network configuration is the opposite, an FFNN with Log-sig transfer and MSE performance functions needs 70 neurons to obtain 47.55% accuracy, while with the Tan-sig transfer function, an FFNN needs 70 neurons to obtain 45.59% accuracy. Similar observation can be made for the FFNN with SSE performance function.

### *B. Static, Cascaded, and Dynamic FFNN*

Based on the results and observations obtained in the previous section, the MSE performance function and Log-sigmoid transfer function are fixed for the next experiment settings. Table III shows the comparison of the results obtained by the static, cascaded, and dynamic FFNN structures with different numbers of hidden layers. It should be noted from Table III that dynamic networks with a single hidden layer yield the best performance results for error-

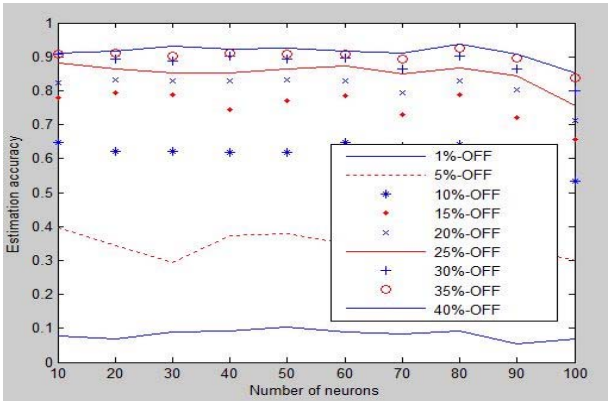


Figure 7. Performance of a static single-hidden layer FFNN with Log-sigmoid transfer and MSE performance functions

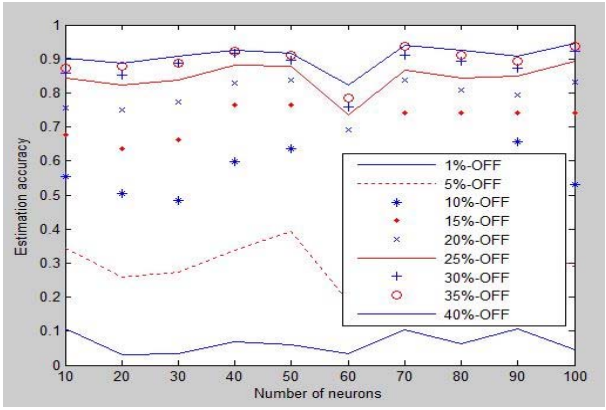


Figure 8. Performance of a static single-hidden layer FFNN with Log-sigmoid transfer and SSE performance functions

TABLE I.
BEST ESTIMATION ACCURACY OF STATIC FFNN WITH MSE PERFORMANCE

| | Tan-sig | | Log-sig | |
|---|---|---|---|---|
| | Number of neurons | Accuracy | Number of neurons | Accuracy |
| **±1%** | 70 | **0.0980** | 10 | **0.1029** |
| **±5%** | 10 | **0.4559** | 70 | **0.4755** |
| **±10%** | 10 | **0.6716** | 60 | **0.6471** |
| **±15%** | 40 | **0.7990** | 20 | **0.7941** |
| **±20%** | 80 | **0.8725** | 20 | **0.8333** |
| **±25%** | 40 | **0.9020** | 10 | **0.8824** |
| **±30%** | 90 | **0.9265** | 10 | **0.9069** |
| **±35%** | 90 | **0.9314** | 80 | **0.9265** |
| **±40%** | 20 | **0.9363** | 90 | **0.9363** |

TABLE II.
BEST ESTIMATION ACCURACY OF STATIC FFNN WITH SSE PERFORMANCE

| | Tan-sig | | Log-sig | |
|---|---|---|---|---|
| | Number of neurons | Accuracy | Number of neurons | Accuracy |
| **±1%** | 50 | **0.0882** | 10 | **0.1078** |
| **±5%** | 50 | **0.4216** | 90 | **0.3725** |
| **±10%** | 40 | **0.6275** | 90 | **0.6569** |
| **±15%** | 40 | **0.8137** | 40 | **0.7647** |
| **±20%** | 40 | **0.8725** | 40 | **0.8284** |
| **±25%** | 40 | **0.9216** | 100 | **0.8284** |
| **±30%** | 40 | **0.9412** | 100 | **0.9216** |
| **±35%** | 40 | **0.9510** | 70 | **0.9363** |
| **±40%** | 40 | **0.9559** | 100 | **0.9461** |

tolerance levels of ±1%, ±5%, ±10%, ±15%, and ±20% at 23%, 74%, 86%, 89%, and 90% estimation accuracies, respectively. Moreover, dynamic networks with 4 hidden layers yield the best performance for error-tolerance levels of ±25%, ±30%, ±35%, and ±40%, at 91%, 93%, 95%, 96%, and 96% estimation accuracies, respectively. Cascaded networks, however, have performed the worst for error-tolerance levels of ±1%, ±5%, ±10%, ±15%, and ±20%, at 7%, 4%, 60%, 72%, and 78% estimation accuracies, respectively. At other error-tolerance levels, they yield comparable results. Static networks performed better than cascaded networks but less than dynamic networks. It should be noted that, in this particular application, increasing the number of hidden layers has not necessarily

TABLE III.
BEST ESTIMATION ACCURACY OF STATIC, CASCADED, DYNAMIC FFNN

| Error-Tolerance | Feed Forward Network Architecture | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Static | | | | Cascaded | | | | Dynamic | | | |
| | Hidden Layers | | | | Hidden Layers | | | | Hidden Layers | | | |
| | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| **±1%** | **0.11** | **0.11** | **0.09** | **0.08** | **0.07** | **0.06** | **0.06** | **0.06** | **0.23** | **0.22** | **0.23** | **0.40** |
| **±5%** | **0.47** | **0.41** | **0.38** | **0.40** | **0.31** | **0.30** | **0.25** | **0.25** | **0.70** | **0.68** | **0.63** | **0.74** |
| **±10%** | **0.67** | **0.64** | **0.66** | **0.70** | **0.49** | **0.60** | **0.43** | **0.45** | **0.86** | **0.84** | **0.79** | **0.83** |
| **±15%** | **0.81** | **0.77** | **0.75** | **0.81** | **0.65** | **0.72** | **0.63** | **0.62** | **0.89** | **0.89** | **0.85** | **0.86** |
| **±20%** | **0.87** | **0.84** | **0.82** | **0.88** | **0.75** | **0.78** | **0.75** | **0.74** | **0.90** | **0.89** | **0.89** | **0.91** |
| **±25%** | **0.92** | **0.87** | **0.88** | **0.90** | **0.81** | **0.83** | **0.82** | **0.80** | **0.91** | **0.89** | **0.91** | **0.93** |
| **±30%** | **0.94** | **0.90** | **0.92** | **0.93** | **0.85** | **0.85** | **0.86** | **0.86** | **0.92** | **0.90** | **0.92** | **0.95** |
| **±35%** | **0.95** | **0.92** | **0.93** | **0.94** | **0.88** | **0.91** | **0.89** | **0.89** | **0.92** | **0.90** | **0.92** | **0.96** |
| **±40%** | **0.95** | **0.94** | **0.93** | **0.95** | **0.90** | **0.93** | **0.90** | **0.91** | **0.92** | **0.90** | **0.92** | **0.96** |

improved the performances of the networks.

With the exception of dynamic networks (with 4 hidden layers, and for the error-tolerance levels ±1% and ±5%), it has actually reduced the overall performance of the feed-forward neural networks.

### C. *Discussion*

In this paper, we suggested an approach consisting of combining pattern-adapted wavelets for locating metal-loss defects and determining their length and neural networks for estimating the defect depth. In the current work, dealing with big data resulted from using large number of MFL sensors, the focus was on studying the applicability and suitability of the neural network component. Compared with the performance of pipeline inspection techniques reported by service providers such as GE and ROSEN, the results obtained using the method we proposed are promising. For instance, within ±10% error-tolerance range, the obtained estimation accuracy is 86%, compared to only 80% reported by GE; and within ±15% error-tolerance range, the achieved estimation accuracy is 89% compared to 80% reported by ROSEN.

## V. CONCLUSIONS

The difficulties of identifying the relationship between defect depths and amplitudes of MFL signals are mainly attributed to the lack of a descriptive analytical model and to the inherent complexity pertaining to the huge amount of data collected from MFL sensors. To cope with such complexity, we proposed the use of artificial neural networks (ANNs) for estimating pipeline defect depths. The new approach is evaluated using different levels of error-tolerance. Extensive experimental work for different parameters and configurations, and various architectures including static, cascaded, and dynamic networks, has been conducted. The Levenberg-Marquardt back-propagation learning algorithm has been utilized. It has been shown that dynamic neural networks yield the best performance of 86% and 89% defect depth estimation accuracy within ±10% and ±15% of error-tolerance, respectively; while cascaded networks yield the worst performance. To increase the defect depth estimation accuracy at lower levels of error-tolerance, we intend in future work to obtain more sophisticated features and employ other learning algorithms.

## ACKNOWLEDGMENT

This work was made possible by NPRP grant # [5-813-1-134] from Qatar Research Fund (a member of Qatar Foundation). The findings achieved herein are solely the responsibility of the authors.

## REFERENCES

[1] P. Nicholson, "Combined CIPS and DCVG survey for more accurate ECDA data," Journal of World Pipeline, vol. 7, pp. 1-7, 2007.

[2] M. Layouni, S. Tahar, and M. Hamdi, "A survey on the application of neural networks in the safety assessment of oil and gas pipelines, " IEEE Symposium on Computational Intelligence for Engineering Solutions, pp. 95-102, 2014.

[3] A. A. Carvalho, et al, "MFL signals and artificial neural networks applied to detection and classification of pipe weld defects," Ndt & E International, vol. 39(8), pp. 661-667, 2006.

[4] S. K. Sinha and M. D. Pandey, "Probabilistic neural network for reliability assessment of oil and gas pipelines," Computer-Aided Civil and Infrastructure Engineering, vol. 17(5), pp. 320-329, 2002.

[5] S. K. Sinha and F. Karray, "Classification of underground pipe scanned images using feature extraction and neuro-fuzzy algorithm," IEEE Transactions on Neural Network, vol. 13(2), pp. 393-401, 2002.

[6] M. Zhongli and H. Liu, "Pipeline defect detection and sizing based on MFL data using immune RBF neural networks," IEEE Congress on Evolutionary Computation, pp. 3399-3403, 2007.

[7] K. Mandal and D. L. Atherton, "A study of magnetic flux-leakage signals," Journal of Physics D: Applied Physics, vol. 31(22), pp. 3211, 1998.

[8] I. Daubechies, Ten Lectures on Wavelets. SIAM: Society for Industrial and Applied Mathematics, 1992.

[9] S. Mallat, A Wavelet Tour of Signal Processing. Academic Press, 2008.

[10] M. Misiti, Y. Misiti, G. Oppenheim, and J. M. Poggi, Wavelets and their Applications. Wiley-ISTE, 2007.

[11] J. N. Bradley, C. M. Brislawn, and T. Hopper, "FBI wavelet/scalar quantization standard for gray-scale fingerprint image compression," In: SPIE Procs, Visual Information Processing II, vol. 1961, pp. 293–304, 1993.

[12] M. Unser and A. Aldroubi, "A review of wavelets in biomedical applications," Proceedings of the IEEE, vol. 84(4), pp. 626–638, 1996.

[13] S. Shou-peng and Q. Pei-wen, "Wavelet based noise suppression technique and its application to ultrasonic flaw detection," Ultrasonic, vol. 44(2), pp. 188-193, 2006.

[14] A. Muhammad and Satish Udpa, "Advanced signal processing of magnetic flux leakage data obtained from seamless gas pipeline," Ndt & E International, vol. 35(7), pp. 449-457, 2002.

[15] S. Mukhopadhyay and G. P. Srivastava, "Characterization of metal loss defects from magnetic flux leakage signals with discrete wavelet transform."Ndt & E International, vol. 33(1), pp. 57-65, 2000.

[16] K. Hwang, et al, "Characterization of gas pipeline inspection signals using wavelet basis function neural networks," NDT & E International, vol. 33(8), pp. 531-545, 2000.

[17] A. Khodayari-Rostamabad, et al, "Machine learning techniques for the analysis of magnetic flux leakage images in pipeline inspection," IEEE Transactions on Magnetics, vol. 45(8), pp. 3073-3084, 2009.

[18] K. Levenberg, "A method for solution of certain problems in least squares," Quart. Applied Math, vol. 2, pp. 164-168, 1944.

[19] D. W. Marquardt, "An algorithm for least squares estimation of nonlinear parameters," SIAM Journal on Applied Mathematics, vol. 11(2), pp. 431-441, 1963.