ELSEVIER

ISA Transactions 39 (2000) 15–27

ISA TRANSACTIONS®

www.elsevier.com/locate/isatrans

Modeling of an intelligent pressure sensor using functional link artificial neural networks

Jagdish C. Patra*[,1], Adriaan van den Bos

Department of Applied Physics, Delft University of Technology, PO Box 5046, 2600 GA Delft, The Netherlands

Abstract

A capacitor pressure sensor (CPS) is modeled for accurate readout of applied pressure using a novel artificial neural network (ANN). The proposed functional link ANN (FLANN) is a computationally efficient nonlinear network and is capable of complex nonlinear mapping between its input and output pattern space. The nonlinearity is introduced into the FLANN by passing the input pattern through a functional expansion unit. Three different polynomials such as, Chebyschev, Legendre and power series have been employed in the FLANN. The FLANN offers computational advantage over a multilayer perceptron (MLP) for similar performance in modeling of the CPS. The prime aim of the present paper is to develop an intelligent model of the CPS involving less computational complexity, so that its implementation can be economical and robust. It is shown that, over a wide temperature variation ranging from −50 to 150°C, the maximum error of estimation of pressure remains within ±3%. With the help of computer simulation, the performance of the three types of FLANN models has been compared to that of an MLP based model. © 2000 Elsevier Science Ltd. All rights reserved.

Keywords: Intelligent pressure sensor; Functional link artificial neural networks; Temperature compensation; Computational complexity

1. Introduction

The capacitive pressure sensors (CPSs) have been used extensively in various engineering, industrial, scientific and medical applications. A CPS offers many advantages compared to other pressure sensors in terms of sensitivity and power dissipation. However, its response characteristics are highly nonlinear and capacitance change is small compared to its offset capacitance. Advanced signal processing techniques and learning algorithms can be employed for making the measuring system intelligent in the sense of automatic calibration, linearization, and error compensation [1,2].

Further, the response characteristics of a CPS are temperature dependent. In a dynamic environment, where the ambient temperature undergoes wide variations, the CPS needs appropriate temperature compensation, besides corrections to its nonlinear characteristics. To obtain correct readout of applied pressure, complex signal processing techniques have been proposed for such

* Corresponding author. Tel.: +31-15-278-6589; fax: +31-15-278-4263.

E-mail addresses: jcpatra@hotmail.com (J.C. Patra), a.vandenbos@tn.tudelft.nl (A. van den Bos).

[1] On leave from the Department of Applied Electronics & Instrumentation Engineering, Regional Engineering College, Rourkela, Orissa 769 008, India.

PII: S0019-0578(99)00035-X

Nomenclature

2-D	two-dimensional
ADC	analogue to digital converter
ANN	artificial neural network
BP	backpropogation
CPS	capacitive pressure sensor
EEROM	electrically erasable read only memory
FE	functional expansion
FLANN	functional link artificial neural network
FS	full scale
MCU	micro-controller unit
MLP	multilayer perceptron
PC	personal computer
RAM	random access memory
ROM	read only memory
SCI	switched capacitor interface

situations. For example, a Δ-Σ modulation technique has been reported in which the sensor model obtained after complex signal processing is stored in a ROM [3]. A higher order polynomial fit to the data obtained from measurement is used in which the EEROM capacity needed is small, however, at the cost of large computational time [4].

To reduce computational time, a noniterative 2-D calibration method has been reported in which simple formulas have been used to obtain good performance of the sensor model [5]. An MCU-based self calibration technique for a smart capacitor angular position sensor with relatively simple piecewise-linear fit formula has been reported [6].

However, none of the above mentioned techniques have learning or adaptive properties. Recently, artificial neural networks (ANNs) have emerged as a powerful learning technique to perform complex tasks in highly nonlinear dynamic environments. These networks are endowed with certain unique characteristics: the capability of universal approximation, the ability to learn from and to adapt to their environment, and the ability to invoke weak assumptions about the underlying physical phenomenon responsible for the generation of input data. There have been numerous successful applications of ANNs in various fields of science, engineering and industry [7].

In the field of instrumentation and measurement there have been several applications of ANNs [8,9]. To estimate the nonlinearity and for direct digital readout of a CPS, a simple functional link ANN (FLANN) has been employed [10]. Here, the FLANN has been used to estimate the power series coefficients of the sensor characteristics, and using these coefficients the applied pressure can be estimated accurately. Recently, a multilayer perceptron (MLP) has been proposed to model a CPS for estimation of applied pressure with a maximum of 1% full scale (FS) error over a temperature range from −20 to 70°C [11]. Noise tolerance aspects of an ANN-based CPS model have been reported in which it has been shown that with noisy input also, this model can estimate the applied pressure with good accuracy [12]. Modeling and development of an ANN-based smart CPS operated in a dynamic environment has been proposed [13]. Here, a 2-layer MLP has been proposed to compensate the nonlinear response characteristics, and its temperature dependence over a wide range of temperature variation from −50 to 150°C.

Compared to an MLP, a FLANN offers substantial computational advantage. The FLANN is basically a flat single layer network, in which the need of hidden layers has been removed by incorporating functional expansion of the input pattern. The functional expansion effectively increases the dimensionality of the input vector, and hence the hyperplanes generated by the FLANN provide greater discrimination capability in the input pattern space [14]. There have been several applications of FLANN including pattern classification and recognition, system identification and control, functional approximation, and digital communications channel equalization.

In the present paper, we propose an intelligent FLANN-based model of a CPS to obtain direct readout of the applied pressure in a dynamic environment. The prime purpose is to develop a computationally efficient ANN model of the CPS, because of which less hardware will be needed for its implementation providing reliability, economy, and accuracy. A switched capacitor interface

(SCI) converts the change in capacitance of the CPS due to applied pressure into a voltage signal. The FLANN learns the sensor characteristics during the training phase from the knowledge of the SCI output for few known values of applied pressure and the value of temperature fed into it. For expansion of the input pattern, we have used three types of polynomial functions such as, Chebyschev, Legendre and power series, and the corresponding ANNs are named C-FLANN, L-FLANN, and P-FLANN, respectively.

The performance of the three types of FLANN has been studied by means of computer simulations. Its performance has been compared with that of a 2-layer MLP-based CPS model. It is seen that over a wide variation of temperature starting from -50 to 150°C, the maximum estimation error remains within $\pm 3\%$ (FS) in the case of C-FLANN and L-FLANN, where as it is $\pm 4\%$ in the case of P-FLANN. Next, we have compared the computational complexity involved during training, and at the time of implementation of the MLP- and FLANN-based models. It is shown that substantial computational gain can be achieved with the FLANN model compared to an MLP model by sacrificing the accuracy a little.

2. Capacitive pressure sensor and switched capacitor interface

A CPS senses the applied pressure in the form of elastic deflection of its diaphragm. The normalized capacitance of a CPS resulting from the applied pressure P at the ambient temperature T is given by:

$$C_N = C(P, T)/C_0 \tag{1}$$

where

$$\begin{aligned} C(P, T) &= C_0(T) + \Delta C(P, T), \\ &= C_0 f_1(T) + \Delta C(P, T_0) f_2(T). \end{aligned} \tag{2}$$

The offset capacitance at the reference temperature T_0 is denoted by C_0. When pressure is applied to the CPS, its change in capacitance at the reference temperature is given by:

$$\Delta C(P, T_0) = C_0 P_N \frac{1-\tau}{1-P_N} \tag{3}$$

where τ is the sensitivity parameter of the CPS which depends on its geometrical structure, P_N is the normalized applied pressure given by $P_N = P/P_{\max}$, and the maximum permissible applied pressure is represented by $P_{\max}$. Assuming that the change in capacitance due to variation of temperature is linear, the functions $f_i(T)$, $i = 1$ and 2, are given by:

$$f_i(T) = 1 + g_i(T - T_0), \tag{4}$$

where the coefficients g_i, $i = 1, 2$, may have different values depending on the CPS chosen. The normalized capacitance in Eq.(1) may be expressed as:

$$C_N = f_1(T) + \gamma f_2(T), \tag{5}$$

where $\gamma = P_N(1-\tau)/(1-P_N)$. Because of requirement of the ANN modeling, the C_N in Eq. (5) is divided by a factor of 2, so as to keep its maximum value within 1.

When the applied pressure is zero, γ becomes zero. Therefore, the normalized capacitance at zero applied pressure is given by:

$$C_{N0} = f_1(T) = 1 + g_1(T - T_0) \tag{6}$$

From Eq. (6) it may be seen that C_{N0} bears a linear relationship with temperature. Let the normalized temperature T_N be given by $T_N = T/T_{\max}$, where $T_{\max}$ is the maximum permissible operating temperature at which the sensor may be operated. Hence, T_N can be computed from the knowledge of C_{N0} as follows:

$$T_N = \frac{[T_0 + (C_{N0} - 1)/g_1]}{T_{\max}}. \tag{7}$$

Therefore, to estimate the applied pressure by the ANN model, the temperature information, instead of collecting by some other means, maybe by another temperature sensor, can be obtained by using Eq. (7), and fed to the ANN model automatically.

A switched capacitor interface (SCI) for the CPS is shown in Fig. 1. The CPS is represented by $C(P)$. The SCI output provides a voltage signal proportional to the capacitance change in the CPS due to applied pressure. The SCI operation can be controlled by a reset signal θ. When $\bar{\theta} = 1$(logic 1), $C(P)$ charges to the reference voltage V_R while the capacitor C_S is discharged to ground. Whereas, when $\theta = 1$, the total charge $C(P)V_R$ stored in the $C(P)$ is transferred to C_S producing an output voltage given by:

$$V_o = K{\cdot}C(P), \tag{8}$$

where $K = V_R/C_S$. It may be noted that if room temperature changes, then the SCI output also changes, although the applied pressure remains same. By choosing proper values of C_S and V_R the normalized SCI output V_N may be obtained in such a way that

$$V_N = C_N \tag{9}$$

When the applied pressure is zero, the SCI output, denoted by V_{N0}, becomes equal to C_{N0}. Thus, the normalized temperature can be computed by using Eq. (7), and V_{N0}, and fed to the ANN model.

3. Functional link artificial neural network

The learning of an ANN may be considered as approximating or interpolating a continuous, multivariate function $f(X)$ by an approximation function $f_W(X)$. In the FLANN, a set of basis functions Φ and a fixed number of weight parameters W are used to represent $f_W(X)$. The set of basis functions is used to enhance the input pattern by increasing its dimension so as to provide better discrimination capability in the input pattern space. With a specific choice of a set of basis functions, the problem is then to find the weight parameters W that provide the best possible approximation of f on the set of input–output examples.

3.1. Learning with FLANN

A general structure of a FLANN is shown in Fig. 2. Let K be the number of input–output pattern pairs to be learned by the FLANN. Let the input pattern vector X be of dimension n, and the output Y be a scalar. Each of the input patterns passes through a functional expansion block producing a corresponding N-dimensional ($N \geqslant n$) expanded vector.

Let the kth input pattern vector be represented by $X_k = [x_1(k)x_2(k)\ldots x_n(k)]$, and the weight vector of the FLANN be represented by $W = [w_1w_2\ldots w_N]$. Using N basis functions, ϕ_i, $i = 1, 2, \ldots, N$, the FE block produces an expanded input vector given by $\phi(X_k) = [\phi_1(X_k)\phi_2(X_k)\ldots\phi_N(X_k)]$ for the kth pattern. Next, the linear weighted sum S_k is computed, and then it passes through a $\tan h(\cdot)$ nonlinear function to produce the output $\hat{Y}_k$. The input and output relationship for the kth pattern is given by:

$$S_k = \phi(X_k)W^T, \tag{10}$$

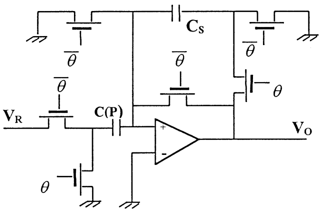

Fig. 1. The switched capacitor interface circuit for a capacitive pressure sensor.

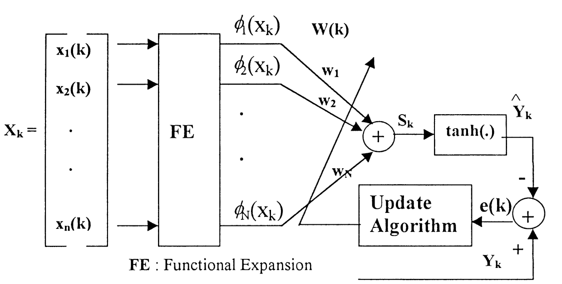

Fig. 2. The structure of a functional link artificial neural network.

$$\hat{Y}_k = \tanh(S_k), \quad \text{or, equivalently}\, S_k = \frac{1}{2}\log_e\left(\frac{1+\hat{Y}_k}{1-\hat{Y}_k}\right). \tag{11}$$

The FLANN, with a set of basis functions ϕ, attempts to approximate the nonlinear function f: $A \in \mathcal{R}^n \rightarrow \mathcal{R}$ using a set of K training patterns. Considering all the K patterns, from Eqs. (10) and (11), the input–output relation of the FLANN can be expressed as:

$$\Phi W^T = S, \tag{12}$$

where Φ is a $K \times N$ matrix given by $\Phi = [\phi(X_1)\phi(X_2)\ldots\phi(X_K)]^T$, and S is a K-dimensional vector given by $S = [S_1 S_2 \ldots S_k]^T$. Thus, from Eq. (12), it is evident that finding the weights of the FLANN requires the solution of K number of simultaneous equations in N unknowns. Now, depending on the values of N and K, three cases may arise:

3.1.1. Case I

$N = K$. If the determinant of Φ is not zero, the weight solution is given by

$$W^T = \Phi^{-1}S. \tag{13}$$

3.1.2. Case II

$N > K$. By choosing any arbitrary K columns of Φ, it may be partitioned to obtain a matrix Φ_F of dimension $K \times K$. Let W be modified to W_F by setting $w_j = 0$ for all $j > K$. If the determinant of Φ_F is not zero, then the weight solution is given by

$$W_F^T = \Phi_F^{-1}S. \tag{14}$$

3.1.3. Case III

$N < K$. In this case the weight solution may be obtained by using conventional pseudo-inversion technique, given by

$$W^T = (\Phi^T\Phi)^{-1}\Phi^T S. \tag{15}$$

From the above analysis it may be seen that the functional expansion technique always yields a flat net solution. The FLANN obtains the weight solution iteratively by using the backpropagation (BP) learning algorithm [7,14]. It may be questioned as to why to use the BP learning algorithm iteratively to obtain the weight solution when closed form solutions are available [Eqs. (13)–(15)]. However, these solutions are based on the presumption that precise values of S_k in terms of $\hat{Y}_k$ are available for all K patterns [Eq. (11)]. But, in practice, precise values of S_k may not be always available. Especially, when $\hat{Y}_k$ approaches ±1, S_k tends to infinity. Matrix manipulation of such large values may lead to numerical instability. Further, the solutions obtained by pseudo-inversion technique do not always provide an accurate solution. Whereas, a feasible and accurate solution is always possible by the FLANN by using BP algorithm iteratively.

Let a number of K patterns be applied to the FLANN in a sequence repeatedly. Let the training sequence be denoted by $\{X_k, Y_k\}$, and the weight of the FLANN by $W(k)$, where k is the discrete time index given by $k = P + \lambda K$, for $\lambda = 0, 1\ldots$ and $P = 1, 2, \ldots K$. Using the BP algorithm for a single layer, the update rule for all the weights of the FLANN is given by

$$\begin{aligned} W(k+1) &= W(k) + \Delta(k) \\ \Delta(k) &= \alpha\delta(k)\phi(X_k) + \beta\Delta(k-1) \end{aligned} \tag{16}$$

where $\delta(k) = (1 - \hat{Y}(k)^2)e(k)$, α and β are the learning parameter and momentum factor, respectively, and $e(k)$ is the error at the kth instant given by $Y(k) - \hat{Y}(k)$.

3.2. Computational complexity

Here, we present a comparison of computational complexity between an MLP and a FLANN structure trained by BP algorithm. For the details of MLP and BP algorithm, one may refer to [7]. Let us consider an L-layer MLP with n_l number of nodes (excluding the threshold unit) in layer l, $l = 0, 1, \ldots, L$, where n_0 and n_L are the number of nodes in the input and output layer, respectively. An L-layer ANN architecture may be represented by $n_0 - n_1 - \ldots - n_{L-1} - n_L$. Three basic computations, i.e. the addition, the multiplication, and the computation of $\tanh(\cdot)$ are involved for updating of the weights of the ANN. The computations in the network are due to the following requirements:

1. forward calculations to find the activation value of all the nodes of the entire network;
2. back-error propagation for calculation of square error derivatives;
3. updating of the weights of the entire network.

The total number of weights to be updated in one iteration in an MLP is given by $(\Sigma_{l=0}^{L-1}(n_l+1)n_{l+1})$, whereas in the case of a FLANN it is only (n_0+1). Since there is no hidden layer in a FLANN, the computational requirement is drastically reduced compared to that of an MLP. A comparison of the computational requirements in one iteration of training using the BP algorithm is provided in Table 1.

4. The three types of FLANN

In this study, we have used three different polynomials for functional expansion of the input pattern in the FLANN. The polynomials are Chebyschev, Legendre and power series, and the respective networks are named as C-FLANN, L-FLANN, and P-FLANN. The polynomials are described briefly below.

4.1. Chebyschev polynomials $-1 \leqslant x \leqslant 1$

$$\begin{aligned} T_0(x) &= 1\\ T_1(x) &= x\\ T_2(x) &= 2x^2 - 1\\ T_3(x) &= 4x^3 - 3x\\ T_4(x) &= 8x^4 - 8x^2 + 1 \end{aligned} \tag{17}$$

Recursion formula: $T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$, $n \geqslant 2$.

4.2. Legendre polynomials $-1 \leqslant x \leqslant 1$

$$\begin{aligned} L_0(x) &= 1\\ L_1(x) &= x\\ L_2(x) &= (3x^2 - 1)/2\\ L_3(x) &= (5x^3 - 3x)/2\\ L_4(x) &= (35x^4 - 30x^2 + 3)/8 \end{aligned} \tag{18}$$

Recursion formula $L_n(x) = [(2n-1)xL_{n-1}(x) - (n-1)L_{n-2}(x)]/n$, $n \geqslant 2$.

4.3. Power series x =*any value*

$$P_n(x) = x^n \quad n \geqslant 0 \tag{19}$$

5. The scheme of FLANN-based CPS modeling

In certain systems, such as missiles, aircraft, spaceships, and chemical and process industries, the CPS may be operated in a dynamic environment in which the temperature variation may be quite large. In consideration to such situations, we have proposed this ANN-based model to take care of a large variation of temperature ranging from −50 to 150°C.

A scheme of an ANN-based CPS model using FLANN for the estimation of applied pressure is shown in Fig. 3. This is analogous to the channel equalization scheme used in a digital communication receiver to cancel the adverse effects of the channel on the data being transmitted. To obtain direct digital readout of the applied pressure, an inverse model of the CPS may be used to compensate adverse effects of the nonlinear characteristics, and its variations due to change in temperature on the CPS performance.

In this sensor modeling, the inputs to the FLANN consist of normalized temperature (T_N), and normalized SCI output (V_N), and the desired (or the target) output for the network is the normalized applied pressure (P_N). One data-set for a specific temperature is generated by noting the

Table 1
Comparison of the computational complexity between and L-layer MLP and a FLANN in one iteration trained by the BP algorithm

Number of	MLP	FLANN
Weights	$\sum_{k=0}^{L-1}(n_k+1)n_{k+1}$	n_0+1
Additions	$3\sum_{k=0}^{l-1}n_k n_{k+1}+3n_L-n_{0-}n_1$	$2n_1n_0+2n_1+n_0/2$
Multiplications	$4\sum_{k=0}^{L-1}n_k n_{k+1}+3\sum_{k=1}^{L}n_k-n_0n_1+2n_L$	$2n_1n_0+4n_1+n_0$
tanh(·)	$\sum_{k=1}^{L}n_k$	n_1

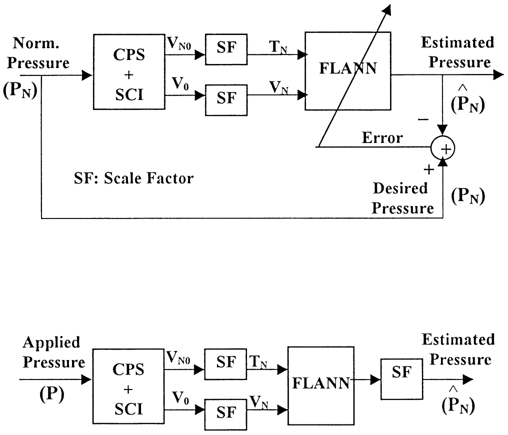

Fig. 3. The scheme of a FLANN-based modeling for a capacitive pressure sensor: (a) *training-mode*; (b) *test-mode*.

SCI output (V_N) for different values of applied pressure covering the operating range of the sensor at that temperature. Next, at different temperature values, covering the full operating range of temperature, its corresponding data-set is generated. Then, the whole of data-sets so generated are segregated into two parts; one as *training-set*, and the other as *test-set*.

The FLANN-based scheme works in two modes: *training-mode* and *test-mode*. During *training-mode* (Fig. 3a), the data patterns from the *training-set* are applied to the FLANN and subsequently its output is computed. Next, the weights of the FLANN are updated using the BP learning algorithm [Eq. (16)]. The training procedure continues considering all patterns of the *training-set* till the error reduces to a pre-set minimum value. Next, the weights of the ANN are frozen and stored in an EEROM for testing of the model to verify its performance, and for actual use.

During *test-mode* (Fig. 3b), these stored weights are loaded into the FLANN model. The ANN has learned the inverse characteristics of the CPS at different values of temperature, and this knowledge has been stored in its weights in a distributive manner. Next, the input patterns from the *test-set* are applied, and the model output $(\hat{P}_N)$ is computed. If the model output matches closely with the actual applied pressure (P_N), then it may be said that the ANN has learned the CPS characteristics correctly.

6. Simulation studies

We have carried out extensive simulation studies to evaluate the performance of the proposed FLANN-based CPS model. In the following we describe the details of the simulation study using the three types of FLANN.

6.1. *Preparation of data-sets*

All the parameters of the CPS, such as, ambient temperature, applied pressure, and SCI output voltage, used in the simulation study were suitably normalized so as to keep their values within ± 1. Several data-sets were needed for training as well as testing of the FLANN model. These data-sets are generated as follows. The SCI voltage (V_N) was recorded at the reference temperature (25°C) at different known values of normalized pressure (P_N) chosen between 0.0 and 0.6 at an interval of 0.05. These 13 pairs of data (P_N and V_N) constitute one data-set at the reference temperature. Next, with the knowledge of this data-set, with the chosen values of g_1 and g_2 as -2×10^{-3} and 7×10^{-3}, respectively, and using Eq. (5), the response characteristics of the CPS for a specific ambient temperature were generated. Each of these response characteristics consists of 13 pairs of data (P_N and V_N), and corresponds to a data-set at that temperature. The response characteristics of the CPS for different values of temperature is shown in Fig. 4. It may be observed from this figure that wide variation in the sensor characteristics occurs when the ambient temperature changes from −50 to 150°C.

For a temperature range from −50 to 150°C, at an increment of 10°C, 21 data-sets, each containing 13 data pairs were generated. Next, these data-sets were divided into two groups as *training-set* and *test-set*. The *training set*, used for training of the FLANN, consists of five data-sets corresponding to −50, 0, 50, 100, and 150°C, and the rest 16 data-sets constitute the *test-set*.

6.2. *Training of the FLANN*

In a FLANN, it is important to choose suitable functional expansion of the input pattern using an appropriate polynomial function. In this modeling problem, we have two inputs, the normalized temperature (T_N) and the normalized SCI output voltage (V_N), applied to the FLANN. The number of functional expansions was selected as 14, after several experiments to give the best performance. This functional expansion is carried out by the FE block as shown in Fig. 3. The different terms of functional expansion are provided in the Appendix.

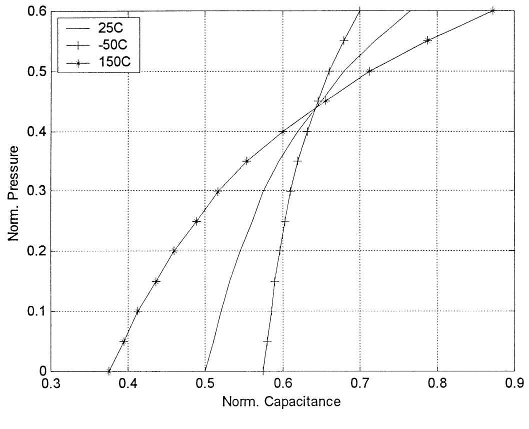

Fig. 4. The response characteristics of the capacitive pressure sensor at −50, 25 and 150°C.

Initially, all the weights of the FLANN were set to some random values within ±0.5. The target or the desired output of the FLANN is the normalized pressure (P_N). During training, the five data-sets were chosen randomly, and further, the individual patterns of each set were also selected in a random manner. After application of one input pattern, the FLANN produces an output $\hat{P}_N$, which was compared with the target pattern to obtain an error value. This error was used to update the weights of the FLANN by the BP algorithm. The values of the learning parameter α, and the momentum factor β, were chosen as 0.5 and 0.7, respectively [16]. Next, another pattern was applied, and this process continues till the mean square error between the desired and the FLANN output reaches a preset minimum value. Completion of weight adaptation for the 13 data-pairs of all the five *training-sets* constitute one iteration. For effective learning, 30,000 iterations were made to train the FLANN model. At the end, the final weights of the FLANN were stored for later use for performance evaluation and actual use of the model. The final weights of the three types of the FLANN model are provided in Table 2.

To compare the performance of the FLANN-based models with an MLP based model, similar simulation studies were carried out using a 2-layer MLP of 2-5-1 architecture [13]. The MLP was trained with BP algorithm taking the values of α, and β as 0.5 and 0.7, respectively. The MLP was trained with 30,000 iterations using the same data-sets. The final weights of the MLP are tabulated in Table 3.

6.3. *Testing of the FLANN model*

The performance evaluation of the model was carried out by loading the final stored weights into the FLANN. It may be noted that, during testing, and actual use of the CPS model, there is no updating of the weights. The inputs are applied to the model, and the FLANN estimates the applied pressure from the knowledge of the stored weights loaded into it. For testing purpose, the SCI output voltage was simulated with a range starting from 0.35 to 0.80 with an increment of 0.01, and then applied to the model along with the temperature

Table 2
Final weights of the three FLANN models (13-1 architecture). $w_i, i=0,1,\ldots,13$

i	C-FLANN	L-FLANN	P-FLANN
0	−1.997	−3.282	−3.262
1	1.265	2.226	2.971
2	0.232	−0.119	−1.405
3	−0.288	−2.17	0.546
4	1.700	3.726	7.175
5	4.317	4.188	−0.112
6	−3.029	−4.145	−2.933
7	0.602	0.955	−0.330
8	−0.512	−1.384	−3.023
9	−3.797	−4.765	−3.838
10	0.990	1.724	0.794
11	1.688	2.306	2.333
12	0.179	0.542	4.023
13	−0.374	−0.809	−2.239

Table 3
Final weights of the MLP model (2-5-1 architecture). First layer, $w_{ji}, j=1,2,\ldots5$, and $I=0,1$ and 2. Second layer, w_k, $k=0,1,\ldots5$

First layer		*j*			
i	1	2	3	4	5
0	1.604	0.070	−3.678	−1.009	1.731
1	−0.469	0.012	2.237	0.171	0.947
2	−3.643	−0.126	9.668	1.316	−1.423
Second layer		*k*			
0	1	2	3	4	5
−1.405	−0.450	−0.051	2.170	0.398	−0.702

information. To evaluate the effectiveness of the model, the FLANN output ($\hat{P}_N$) was computed and then compared with the true values of applied pressure (P_N).

7. Results and discussions

True and estimated pressures at different values of temperature taken from *training-set* for the three FLANN models and the MLP are plotted in Fig. 5. Here, different symbols represent true normalized pressure, whereas the dotted lines denote the estimated pressure. Similar plots for temperature values taken from *test-set* are shown in Fig. 6. It may be noted that the ANN models have not seen the sensor characteristics for the values of temperature taken from the *training-set*. From these two figures, it may be observed that, the MLP is capable of estimating the applied pressure quite accurately for the full range of applied pressure from 0.0 to 0.7. In the case of three FLANN models, for both *training-* and *test-sets*, the estimation of pressure is accurate enough up to the P_N value of 0.6. However, beyond this value of P_N, the FLANN models are not capable of accurate estimation, probably because during training, these models have not seen the CPS characteristics for this range of P_N.

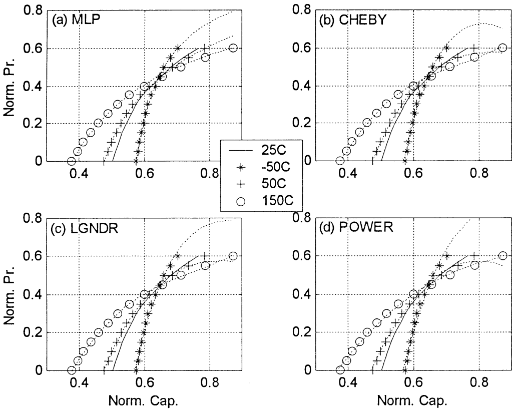

Fig. 5. True and estimated response characteristics of the CPS at different values of temperature (*training-set*): (a) MLP; (b) C-FLANN; (c) L-FLANN; (d) P-FLANN.

Plots between the true and estimated pressure at different values of temperature taken from *test-set* are shown in Fig. 7. The linearity of estimation in the case of MLP model is quite good. Among the three FLANN models, the performance of C-FLANN and L-FLANN is similar, but inferior to the MLP model. The performance of P-FLANN is worse than other two FLANN models.

Full scale percent error in estimation of applied pressure at different temperatures taken from *test-set* for the MLP and the three FLANN models are plotted in Fig. 8. The FS percentage error was computed as 100 times the difference between the true and estimated pressure. For the whole range

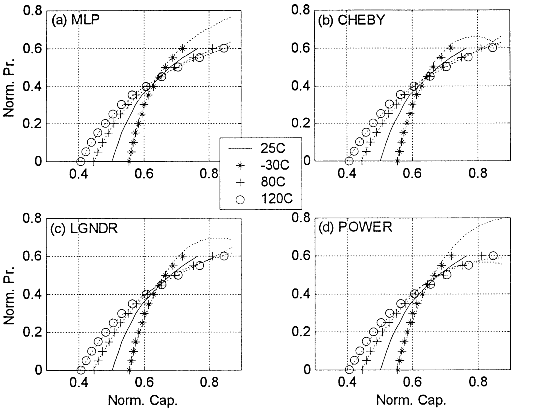

Fig. 6. True and estimated response characteristics of the CPS at different values of temperature (*test-set*): (a) MLP; (b) C-FLANN; (c) L-FLANN; (d) P-FLANN.

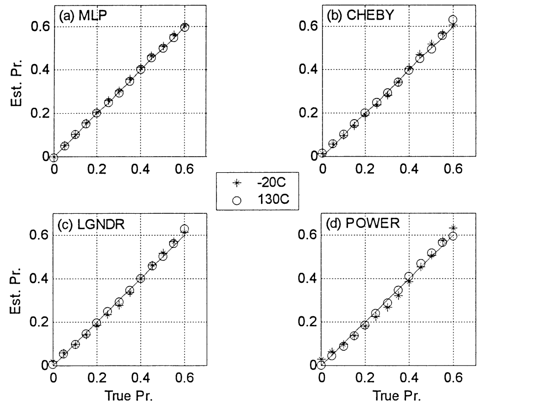

Fig. 7. The plots of the true versus the estimated pressure at different values of temperature: (a) MLP; (b) C-FLANN; (c) L-FLANN; (d) P-FLANN.

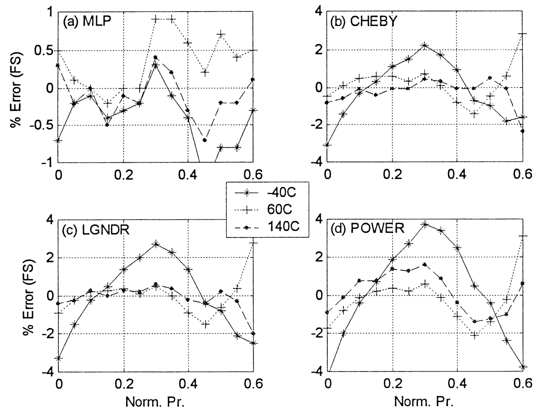

Fig 8. Full scale percent error in estimation of applied pressure at different values of temperature (*test-set*): (a) MLP; (b) C-FLANN; (c) L-FLANN; (d) P-FLANN.

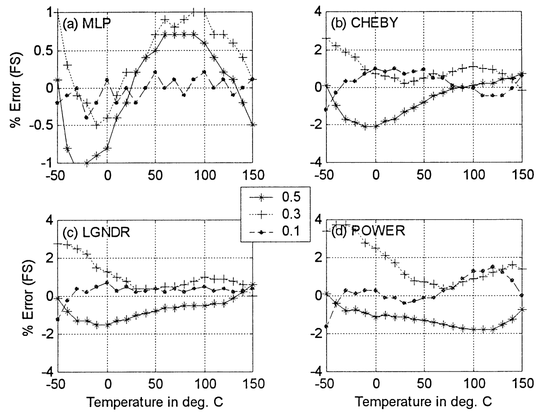

Fig. 9. Full scale error at specific values of applied pressure ($P_N = 0.5$, 0.3 and 0.1) for the full range of variation of temperature: (a) MLP; (b) C-FLANN; (c) L-FLANN; (d) P-FLANN.

of applied pressure from 0.0 to 0.6, the maximum FS error for the MLP, C-FLANN, L-FLANN, and P-FLANN remains within ±1, ±3, ±3, and ±4%, respectively.

For the whole range of temperature starting from −50 to 150°C, at specific values of applied pressure (i.e. $P_N = 0.5$, 0.3, 0.1), the FS error was plotted for the four ANN models and is shown in Fig. 9. Finally, the performance comparison between the four ANN models in terms of FS error is depicted in Fig. 10 at specific values of temperature.

From the above observations, it is clear that the performance of the MLP model is superior to the FLANN models. Between the FLANN models, the performance of the C-FLANN and L-FLANN is similar and is slightly inferior to that of the MLP model. It is observed that the performance of the P-FLANN is the worst, however, it may be noted that it has the simplest architecture, and has the minimum computational requirements between the four ANN models studied.

The amount of computational complexity of the learning algorithm determines its efficiency during training phase and cost of hardware at the time of implementation of the model. Numerical results of

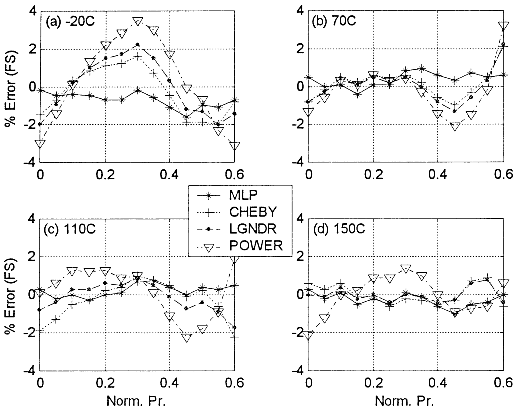

Fig. 10. The comparison of percent of error (FS) between the MLP and three FLANN models: (a) –20°C; (b) 70°C; (c) 110°C; (d) 150°C.

comparison of computational complexity between a 2-layer MLP (2-5-1 architecture) and a FLANN (13-1 architecture) using BP algorithm obtained from Table 1 is shown in Table 4. Computation of multiplication and tanh(·) involves more hardware and complexity. The number of multiplications and tanh( ) needed for the FLANN is about 60 and 20%, respectively, compared to that of the MLP. Further, the number of weights in the FLANN is about 70% of the number of weights of the MLP structure. The simulation studies were carried out using a Pentium-MMX, 233 MHz machine. The times required for training the MLP and the FLANN for 30,000 iterations are 111 and 71 s, respectively. Thus, there is a substantial amount of computational saving in the FLANN-based model. This results in a cost-effective and reliable hardware implementation of the model. By sacrificing a slight degradation in performance, substantial gain in computational load can be achieved by using FLANN-based CPS model.

Table 4
Numerical comparison of the computational complexity between a 2-layer MLP (2-5-1 architecture) and a FLANN (13-1 architecture) and a FLANN (13-1 architecture)-based CPS models

Number	NLP	FLANN
Weights	21	14
Additions	38	35
Multiplications	76	4
tan*h*(·)	6	1
Time in seconds	111	71

8. An implementation scheme

Recently the MCUs have been found quite suitable for use in various intelligent systems due to rapid decrease in unit cost and fast increase in on-chip capabilities. Currently available MCUs can be configured with all the required RAM/ROM/EEROM as well as serial interface and multiple channel ADC support chips. A scheme of implementation of the FLANN-based CPS model using an MCU is depicted in Fig. 11 [5]. The SCI converts the change in capacitance of the CPS due to applied pressure into an equivalent voltage level. This analog SCI voltage is passed through an ADC. The digital temperature information is also similarly obtained from the knowledge of C_{N0}.

During the *training-phase*, the CPS is operated under controlled temperature and the data pairs so collected can be stored in the memories of the MCU. These training data can be fed to a PC connected to the MCU during training of the FLANN-based model. After completion of training, the weights of the FLANN are stored in the EEROM of the MCU. With the available hardware, such as adders and multipliers of the MCU, the FLANN-based model can be implemented. The digital readout of the applied pressure can be displayed through the data bus.

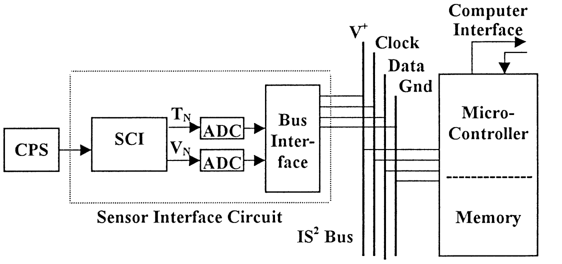

Fig. 11. An MCU-based implementation scheme of the CPS model.

9. Conclusions

We have proposed a novel computationally efficient ANN for modeling of a CPS operated in a dynamic environment. The FLANN is capable of estimation of pressure quite accurately irrespective of nonlinear characteristics of the CPS and its temperature dependence. Out of the three types of the proposed FLANN structures, the C-FLANN and L-FLANN perform similarly, and the performance of the P-FLANN is slightly inferior to the former two structures. However, between the three FLANN models, the computational requirement is the minimum in the case of P-FLANN. The maximum estimation error in the case of MLP and FLANN models is ±1 and ±3%, respectively, over a wide range of temperature variation starting from −50 to 150°C. The FLANN offers substantial advantage over the MLP in terms of computational gain. Because of these characteristics of the FLANN, it can offer large saving of hardware in the implementation of the CPS model by sacrificing a little degradation in performance.

The FLANN architecture, because of its simple architecture and computational efficiency, may be conveniently employed in other fields of instrumentation and measurement systems to compensate the adverse effects of the environment and nonidealities of the devices used in the system.

Appendix

Functional expansion used in simulation

T_N and V_N represent the normalized temperature and the normalized SCI output voltage, respectively. The functional expansion (FE) block performs the following expansion. First, three arrays, $Av[\,]$, $At[\,]$ and $Ap[\,]$ are generated. From these arrays, the $\phi(k)$ are computed.

```
for($n = 1; n \leqslant 4p; n++$)
{
    $Av[n]$        $= \mathcal{P}_n(V_n)$
    $At[n]$        $= \mathcal{P}_n(T_N)$
}
$Ap[0]$        $= Av[1]^* At[1]$
$Ap[1]$        $= Av[2]^* At[1]$
$Ap[2]$        $= Av[2]^* At[2]$
$Ap[3]$        $= Av[3]^* At[1]$
$Ap[4]$        $= Av[3]^* At[2]$
$Ap[5]$        $= Av[3]^* At[3]$
$\phi(0)$      $= 1$
$\phi(1)$      $= At[1]$
$\phi(2)$      $= At[2]$
$\phi(3)$      $= At[3]$
$\phi(4)$      $= Av[1]$
$\phi(5)$      $= Av[2]$
$\phi(6)$      $= Av[3]$
$\phi(7)$      $= Av[4]$
$\phi(8)$      $= Ap[0]$
$\phi(9)$      $= Ap[1]$
$\phi(10)$     $= Ap[2]$
$\phi(11)$     $= Ap[3]$
$\phi(12)$     $= Ap[4]$
$\phi(13)$     $= Ap[5]$
```

$\mathcal{P}_n(z) = T_n(z)$, $L_n(z)$, or $P_n(z)$ (see Section 4).

References

[1] L. Finkelstein, Intelligent and knowledge based instrumentation — an examination of basic concepts, Measurement 14 (1994) 23–29.

[2] M. Yamada, T. Takebayashi, S-I. Notoyama, K. Watanabe, A switched-capacitor interface for capacitive pressure sensors, IEEE Transactions Instrumentation and Measurement 41 (1) (1992) 81–86.

[3] M. Yamada, K. Watanabe, A capacitive pressure sensor interface using oversampling Δ-Σ demodulation techniques, IEEE Transactions on Instrumentation and Measurement 46 (1) (1997) 3–7.

[4] P. Hille, R. Hohler, H. Strack, A linearization and compensation method for integrated sensors, Sensors and Actuators B 44 (1994) 95–102.

[5] K.F. Lyahou, G. van der Horn, J.H. Huijsing, A noniterative polynomial 2-D calibration method implemented in a microcontroller, IEEE Transactions on Instrumentation and Measurement 46 (4) (1997) 752–757.

[6] X. Li, G.C. Meijer, G.W. de Jong, A microcontroller-based self-calibration technique for a smart capacitance angular-position sensor, IEEE Transactions on Instrumentation and Measurement 46 (4) (1997) 888–892.

[7] S. Haykin, Neural Networks, Maxwell Macmillan, Ontario, Canada, 1994.

[8] L.F. Pau, P. Johansen, Neural network signal understanding for instrumentation, IEEE Transactions on Instrumentation and Measurement 39 (4) (1990) 588–594.

[9] P. Daponte, D. Grimaldi, Artificial neural networks in measurements, Measurement 23 (1998) 93–115.

[10] J.C. Patra, G. Panda, R. Baliarsingh, Artificial neural network-based nonlinearity estimation of pressure sensors, IEEE Transactions on Instrumentation and Measurement 43 (6) (1994) 874–881.

[11] J.C. Patra, An artificial neural network-based smart capacitive pressure sensor, Measurement 22 (3-4)) (1997) 113–121.

[12] J.C. Patra, G. Panda, ANN-based intelligent pressure sensor in noisy environment, Measurement 23 (4) (1998) 229–238.

[13] J.C. Patra, A. van den Bos, Modeling and development of an ANN-based smart pressure sensor in a dynamic environment, Measurement, accepted for publication.

[14] Y.H. Pao, Adaptive Pattern Recognition and Neural Networks, Addison-Wesley, Reading, MA, 1989.