1st International Conference on Sensing Technology
November 21-23, 2005 Palmerston North, New Zealand

Linearising Calibration Methods for a Generic Embedded Sensor Interface (GESI)

Amra Pašić
Work done in: PEI Technologies, School of Electronic Engineering,
Current address: National Centre for Sensor Research,
Dublin City University, Glasnevin, Dublin 9, Ireland
amra.pasic@dcu.ie

Jim Dowling
School of Electronic Engineering,
Dublin City University, Glasnevin, Dublin 9, Ireland
jim.dowling@dcu.ie

Abstract

Various linearisation and calibration techniques for sensor characteristic are investigated and the most promising techniques, curve fitting and the progressive polynomial calibration method, are then examined in greater detail. Results of simulations are presented to compare their performance in order to determine the methods most suitable for the implementation of a generic embedded sensor interface (GESI). The fundamental limitations and trade offs encountered in the design and implementation of these techniques using a microprocessor based systems are studied.

Keywords: linearization, calibration, sensors, characteristic, generic, embedded, interface, interpolation, curve, fitting, progressive, polynomial, GESI, PPC

1 Introduction

The relationship between a sensor's output and input is, in general, non-linear. For many reasons (ready availability of instrumentation and use in control systems, for example) it is desirable that the signal representing the sensor input is linearly related to that input. To achieve this, the actual sensor output must in general be linearised. It is usual to combine the linearisation and calibration processes as shown in Figure 1.

If sensor data are to be subsequently processed digitally and a sensor output is nonlinear the simplest way to perform linearisation is in the digital domain. In a digital signal processing environment, an analogue calibration of the sensors and compensation for temperature effects or nonlinearities often make sense before the signal is applied to an analogue to digital converter (ADC), it is better to perform the linearisation digitally. Apart from errors in the transfer characteristic of the sensor, errors from the other function blocks, such as the ADC or some analogue interface, are corrected automatically in the calibration process.

Since a microcontroller affects the cost of the system and performing linearisation and calibration is power consuming, it was necessary to find the trade off between the linearisation and calibration techniques and processing power.

These techniques are chosen so that they can serve a vast variety of sensors rather than particular sensors (generic), hence they can be a part of a GESI.

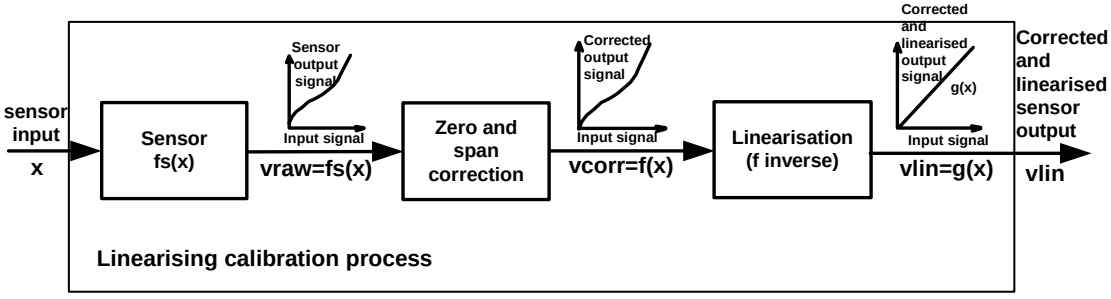

Figure 1: Linearising calibration process

2 Background

2.1 Linearising Calibration Methods

A sensor transfer characteristic can have one or more of the following error types [1]: offset; gain, range or full-scale error; nonlinearity; cross-sensitivity; hysteresis; drift.

A number of measurements must be taken to reduce/eliminate these errors including the determination and correction of a sensor's nonlinearity. The number of measurements necessary to reduce the linearity error depends on the linearising calibration (LC) method used and, to reduce the costs of calibration, it is important to minimize the number of measurements. Costs refer to the expense of processing power and time. This is an important criterion in the selection of an appropriate linearising calibration method for sensor calibration.

All of the following LC methods [1] are based on the use of calibration measurements:

- Look-up table,
- Piecewise linear interpolation,
- Piecewise polynomial or spline interpolation,
- Error minimization,
- Sensor characteristic linearisation,
- Curve fitting,
- Progressive polynomial calibration (PPC).

Look-up table, error minimization and piecewise polynomial or spline linearization require many calibration measurements, large memory for storage, and advanced computations. Error minimization and sensor characteristic linearisation also require *a priori* knowledge of the transfer characteristic and as such are not suitable for implementation on a GESI. Curve-fitting and progressive polynomial calibration offer the possibility to linearise a sensor transfer characteristic using a small number of calibration measurements. Hence, curve fitting and progressive polynomial calibration are chosen for further investigations due to their advantages, and discussed in the following sections.

2.2 Curve Fitting

Curve fitting can be achieved by either polynomial approximation or interpolation. In calibration more than 5 points will rarely be used so an interpolation would produce better results. The only advantage is if noise is present (errors in measurement). In that case approximation might be better than matching the data exactly.

Interpolation methods are based on polynomials. A polynomial of degree n is uniquely determined by n+1 points. The number of points used determines the degree of the polynomial used. The general form of a polynomial of degree n is

$$P(x) = c_n x^n + c_{n-1} x^{n-1} + \ldots + c_1 x + c_0 \quad (1)$$

When a sensor output is measured for a small set of known input signals an interpolation algorithm is then used to compute a sensor transfer function $v = f_{est}(x)$ using curve fitting techniques (Figure 2). The inverse transfer function is then calculated so the input signal can be derived from the sensor output $(f_{est})^{-1}$. The other possibility is to find a curve fit function for the inverse sensor transfer function straight away $x = (f^{-1})_{est}(v)$.

The advantages of curve fitting are that an inverse transfer function is obtained for the complete signal range and memory requirements are low due to a small number of coefficients. The disadvantage can be that sometimes higher order polynomials are necessary to get the desired accuracy. This involves more complex calculations and may also require high accuracy computations (floating point arithmetic). In general, the degree of the polynomial is kept low.

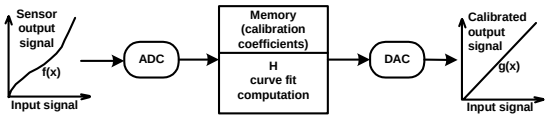

Figure 2: Inverse transfer function method for linearising calibration based on curve fitting.

2.3 Progressive Polynomial Calibration

Progressive polynomial calibration [1] operates on the principle that each calibration measurement is used directly to calculate one calibration coefficient in the correction function. This correction is then applied to the sensor output. Each step is independent of the previous one. The first measurement is used to correct the offset; the second corrects the gain and all the rest of the measurements are used for nonlinearity correction. The advantages are that it gives good linearisation for a minimum number of points, has low memory requirements due to a small number of coefficients and uses a repetitive algorithm (step by step calibration). The disadvantages are that some knowledge is needed when choosing calibration measurements.

The input variable to the sensor is indicated as x, the sensor output is indicated by y; the sensor transfer characteristic is denoted as $y = f(x)$; the desired sensor transfer characteristic is $y = g(x)$ and it is assumed to be linear function of the input signal, $g(x) = Kx$ (Figure 3).

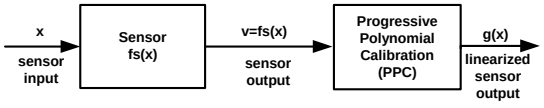

Figure 3: Block diagram of PPC method.

PPC is performed by taking calibration measurements for a set of a well-known input signals x_n, and comparing the measured sensor output $f(x_n)$ to the desired output. The method [1] calculates a corrected transfer curve $h_n(x)$ after each calibration measurement. Previously calculated transfer functions $h_1(x)$ to $h_{n-1}(x)$, the previous calibrated output values y_1 to y_n and the n^{th} calibration measurement $f(x_n)$ are used in calculations. The calibration can be presented as a series of nested formulae:

$$h_n(a_n, h_1(x), ..., h_{n-1}(x), y_1, ..., y_{n-1}) \tag{2}$$

where a_n is referred to the nth calibration coefficient.

At each calibration step the calibration coefficient a_n has to be calculated in order to obtain the linearised sensor output $h_n(x_n) = g(x_n) = y_n$.

The equations can be generalized [1]. The calibration coefficient a_n is calculated as:

$$a_n = \frac{y_n - h_{n-1}(x)}{y_{ref}} \prod_{i=1}^{n-1} \frac{y_{ref}}{h_i(x_n) - y_i} \tag{3}$$

while the n^{th} calibration function is given by:

$$h_n(x) = h_{n-1}(x) + a_n y_{ref} \prod_{i=1}^{n-1} \frac{h_i(x) - y_i}{y_{ref}} \tag{4}$$

3 Simulation and Test on Sensor Characteristic Functions

Linearisation methods using function interpolation based on Taylor polynomial interpolation (3rd order), Gaussian elimination, Lagrange interpolation and progressive polynomial calibration (PPC) were tested and simulated, firstly on chosen functions and then on sensor characteristic functions.

3.1 Simulations on Chosen Functions

To determine the coefficients of the third order Taylor polynomial i.e. the first three derivatives, the numerical finite difference formulae were applied. Estimation of derivatives at some point (x_k, y_k) can be made to the left (forward differences), to the right (backward differences) or about the middle point (central differences). Central differences use the slope of the secant through the points (x_{k-1}, y_{k-1}) and (x_{k+1}, y_{k+1}) instead of the slope of the tangent at the point (x_k, y_k), because it is generally more accurate, due to the symmetrical position of two points about the point of interest (Figure 4). The step h should be as small as possible, $h \to 0$.

The disadvantage with using central differences is that they cannot be applied at the end points because there are no "known" points from both sides.

These derivatives are coefficients of the Taylor series and are used in further calculations for any other values of output from the sensor (Eq. 5). In order to make this method work it is necessary to have data points equally spaced ($x-nh$, $x+nh$) for the calibration process.

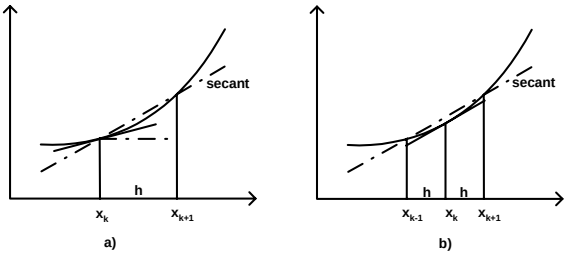

Figure 4: a) Backward differences, b) Central differences.

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

$$f''(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} \tag{5}$$

$$f'''(x) \approx \frac{f(x+2h) - 2f(x+h) + 2f(x-h) - f(x-2h)}{2h^3}$$

The error originates in choosing a step h; for 5 calibration measurements on a wide range, a large step h is inevitable. Thus the approximation is only good around the approximating point while for the rest of the range it is very poor.

Curve fitting using the piecewise Taylor polynomial interpolation was also simulated. However, calculating the piecewise Taylor series is not very practical. It also requires a lot of memory to store the coefficients of each subrange.

With Gaussian elimination finding the inverse matrix is an inefficient approach in computer algorithms. Basic elimination without pivoting consists of $n-1$ eliminations steps for a system of n equations and gives a transformed linear system equivalent to the original system that is then solved by backward substitution. For example, interpolation with a third order polynomial (Eq. 1) is simply obtained solving the system of 4 equations with 4 unknowns (4 coefficients) with Gaussian elimination.

Lagrange interpolation in general showed that equally spaced y_k points (sensor output) give good results when interpolating the inverse sensor transfer characteristic. The proposed number of measurements is 5; more than 6 is not advisable as it often causes deviation leading to large errors associated with the Runge phenomenon.

The choice of calibration measurements in PPC is important but so is the order in which they are subsequently used. In terms of choosing calibration measurements for the functions that change rapidly it is better to "squeeze" calibration measurements on the interval of the full range where the error is expected to be large. In most of the cases this should be done by placing the third and rest of the measurements in that interval, keeping the first and the second

measurement at both ends of the full range for offset and gain correction. It is also discovered that when taking the rest of the measurements one of them should be at mid-range. However, this mid-range measurement gives better results when it is taken as a fourth measurement, with the third measurement point between it and the last measurement on the sensitive interval.

Further improvement can be achieved taking additional measurements and/or further analysis of the sensor transfer function in order to adjust the calibration measurements accordingly. The larger the range the more points are needed in order to minimize the nonlinearity error.

3.2 Test on Sensor Characteristic Functions

Taylor series, Gaussian elimination, Lagrange interpolation and progressive polynomial calibration are implemented for two functions that represent actual sensor responses. Function f_1 describes the Humirel HS1100/HS1101 capacitive RH sensor [2] whereas function f_2 describes the Figaro TG832 semiconductor gas sensor [3]. These responses are plotted in Figure 5.

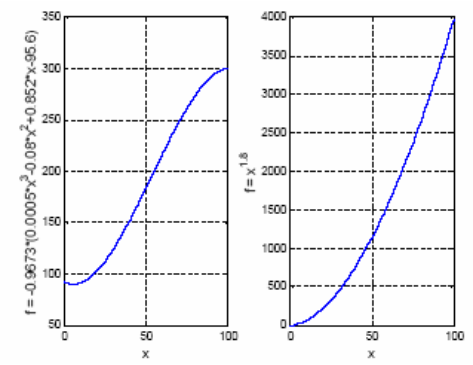

Figure 5: Sensor responses used as test functions described with $f_1 = -0.9673(0.0005x^3 - 0.08x^2 + 0.852x - 95.6)$ and $f_2 = x^{1.8}$.

The piecewise Taylor series polynomial for both of the functions f_1 and f_2 (as shown in Figure 5) were evaluated by the rule derived in [4] (equation 5.7). The results obtained demonstrate very good approximation of the original functions with the error less than 1.5% of full scale (FS), where FS is the difference between maximum and minimum value of the output range and it changes with respect to the function evaluated (Figure 6).

To construct the third order polynomial using Gaussian elimination, 4 points are needed to solve the system of 4 equations (4 unknown calibration coefficients). Points are taken in such a way to get the better fit for the inverse function. The error for the functions f_1 and f_2 is about 11% of the FS (Figure 7).

Lagrange interpolation for the same functions was restricted to 5 points to avoid common errors for higher degree Lagrange polynomials. Points are taken equally spaced on the output range as this was proven to give less error for the lower order polynomials. The error obtained (Figure 8) is less than 6% of FS.

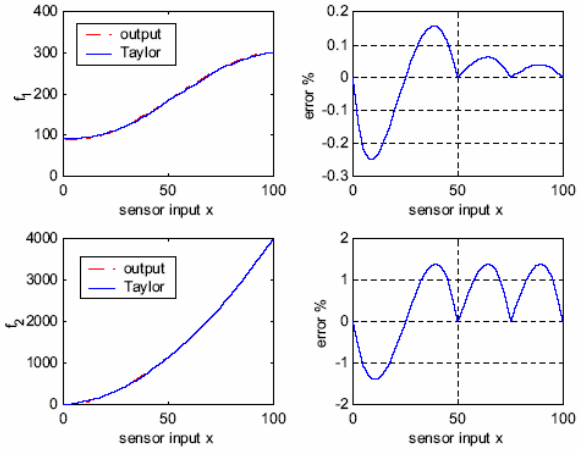

Figure 6: Taylor polynomial interpolation and error percentage for the functions *f1* and *f2*.

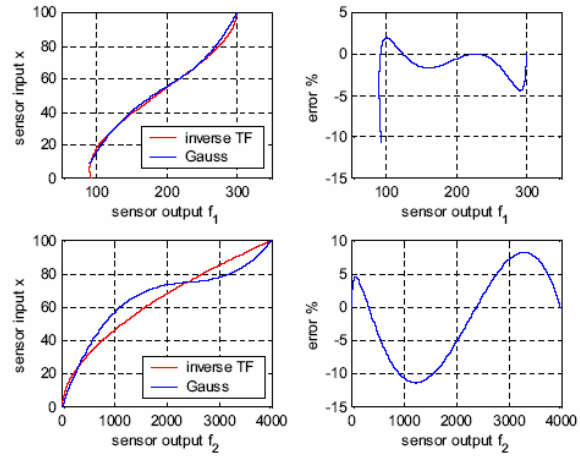

Figure 7: Interpolation with the third order polynomial using Gaussian elimination for calculating the coefficients and error percentage for the inverse transfer functions of the functions f_1 and f_2.

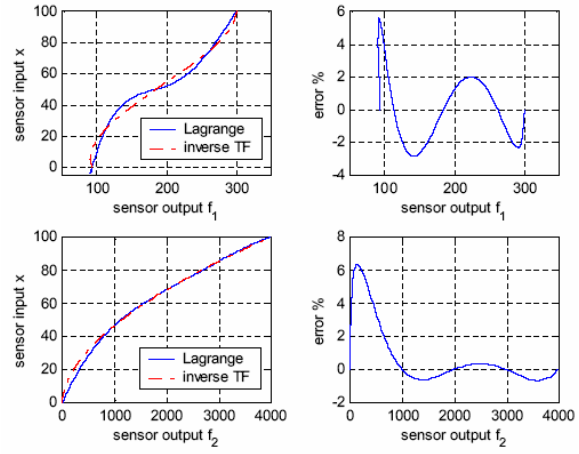

Figure 8: Lagrange polynomial interpolation and error percentage for the inverse transfer functions of the functions f_1 and f_2.

The measurements for PPC were chosen equally spaced on input range. It can be seen that the error is as large as 12% of FS (Figure 9). In order to reduce the error an additional measurement is taken where the deviation was the most pronounced. It was expected that this would reduce the error but to the contrary it produced even greater deviation and larger error (i.e. larger than 12% of FS, Figure 10).

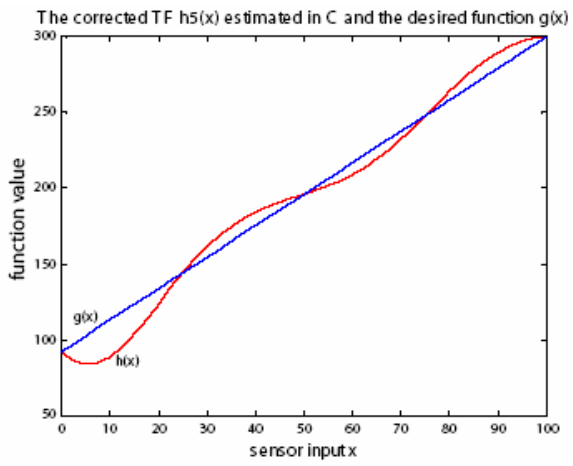

Figure 9: Corrected sensor TF $h_5(x)$ of the sensor TF f_1 using the PPC method with 5 points and the desired sensor TF $g(x)$ plotted in MATLAB (x = {0, 100, 50, 75, 25}).

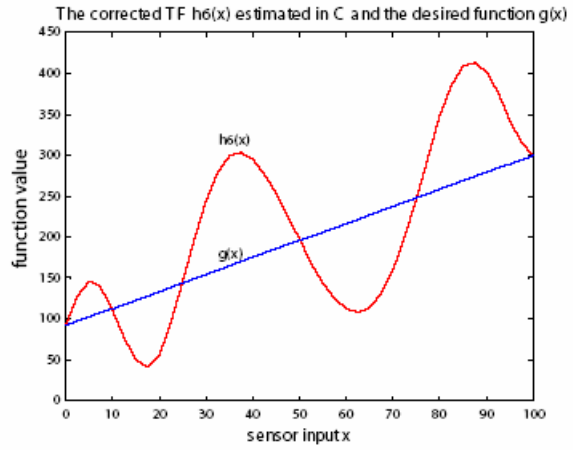

Figure 10: Corrected sensor TF $h_6(x)$ of the sensor TF f_1 using the PPC method with 6 points and the desired sensor TF $g(x)$ plotted in MATLAB (x = {0, 100, 50, 75, 25, 10}).

Two tests were run for the function f_2. One test was when calibration measurements were taken as equally spaced points on the input range and the other test was for equally spaced points on the output range. The results are presented in Figure 11 (equally on x axis) and Figure 12 (equally on y axis).

Progressive polynomial calibration produced an error less than 6% of full scale. This amount of error is associated with the deviation caused by adding the extra points.

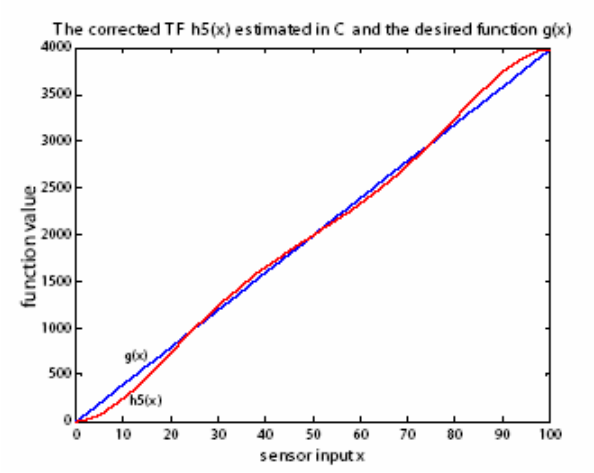

Figure 11: Corrected sensor TF $h_5(x)$ of the sensor TF f_2 using the PPC method with 5 points and the desired sensor TF $g(x)$ plotted in MATLAB (x = {0, 100, 50, 75, 25} - equally spaced points on x axis).

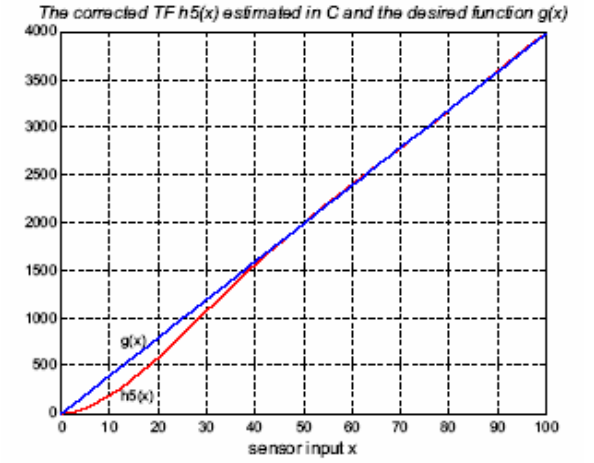

Figure 12: Corrected sensor TF $h_5(x)$ of the sensor TF f_2 using the PPC method with 5 points and the desired sensor TF $g(x)$ plotted in MATLAB (x = {0, 100, 68.04, 85.23, 46.294} - equally spaced points on y axis).

When extreme cases are inspected (functions $x^{1/2}$, x^{-1}, log x and e^x) all methods experienced difficulties and the errors were much larger. (This will be addressed in future publications).

In addition to these tests, some other functions and methods were tested and the results are summarized in Table 1. The methods that are used for testing and comparison and are not explained are piecewise linear, piecewise quadratic and least squares quadratic and cubic approximation [5][6].

4 Conclusions

The Taylor series was abandoned because of the nature of the method and its need for equally spaced points which in practice is almost impossible to achieve. Even a small digression causes a large error.

Table 1: Summary of different methods.

FUNCTION	METHOD (ERROR [%])								
	GAUSS 2nd order of inv	*PW-linear of inv*	*PPC*	*GAUSS 3rd order of inv*	*LS-quadratic of inv*	*PPC*	*PW-quadratic * of inv*	*LS-cubic of inv*	*PPC*
f1 = -0.9673(0.0005x^3-0.08^2+0.852x-95.6)	12%	15.00%	14.40%	12.00%	>12%	12.00%	12.00%	12%	12%
f2 = x^(2.5)	20.60%	15.80%	30.00%	15.84%	14.30%	14.00%	15.85%	11.10%	8.00%
f3 = x^(1.8)	11%	14.80%	10%	7.75%	8.50%	5.50%	7.75%	5.42%	4%
f4 = x^(1/2.5)	2.50%	6%	4.80%	0.20%	2%	1.67%	0.40%	0.18%	0.67%
f5 = x^(1/1.8)	1%	7.70%	1.23%	0.20%	0.62%	0.33%	0.25%	0.20%	0.25%
NUMBER OF POINTS	3	3	3	4	4	4	5	5	5
NUMBER OF COEFFICIENTS	3	4	3	4	3	4	6	4	5

Note: * Gaussian elimination is used for calculating the coefficients in PW-quadratic method.
All the functions are tested on x = 0 − 100 scale.
Colour indicate which method yielded the smallest error for the certain number of the calibration points.

Gaussian elimination, Lagrange interpolation and PPC are more flexible. However, they require some knowledge of the sensor transfer function and adjusting the measurements accordingly during the calibration process which can be a disadvantage since the sensor transfer function is usually unknown. They also have the same drawbacks: selection of the calibration measurements in order to get the best approximation and avoid the global dependence on local properties. Lagrange interpolation and progressive polynomial calibration share the common problem of deviation. Adding extra calibration points after a certain point is to no avail as it results in a more complicated model and even larger deviation. However, the progressive polynomial method demonstrated good results for certain functions and it was tested further together with the Gaussian elimination.

Out of the numerous mathematical methods available, Gaussian elimination and PPC gave the best linearisation results and were deemed the most suitable for the design of the linearisation and calibration module.

Since none of the linearisation methods is good enough for all types of sensor characteristics it is desirable to have two or more linearisation methods implemented on the microcontroller or the DSP chip. Alternative methods can be then selected through the Graphical User Interface (GUI), as well as the number of the calibration points, by the person performing the calibration process. Such a person would need some knowledge of the sensor output in order to choose the appropriate method.

5 Acknowledgements

I would like to thank Jim Dowling, my supervisor, for his suggestions and support during this research.

I am also thankful to PEI Technologies staff for their guidance and help during the research. They shared their knowledge and gave me a friendly encouragement.

6 References

[1] Van der Horn, G. and Huijsing J.L., *Integrated Smart Sensors: Design and Calibration*, Kluwer Academic Publishers, Netherlands (1998).

[2] Relative humidity sensor hs1100/hs1101 datasheet. Humirel "Home Page", http://www-humirel.com/product/fichier/HS1101-HS1100.pdf, visited on 20/4/2005.

[3] TGS 832 datasheet. Figaro Sensor "Home Page", http://www.figarosensor.com/products/832pdf.pdf, visited on 20/4/2005.

[4] Pasic, A., *M.Eng.: Generilised Sensor Linearisation and Calibration*, Dublin City University (2004).

[5] Maron, M.J. and Lopez, R.J., *Numerical Analysis: a Practical Approach*, Wadsworth Publishing, California USA (1991).

[6] Press, W.H, Teukolosky, S.A., Vetterling, W.T., Flannery, B.P., *Numerical Recipes in C*, 2nd Edition, Cambridge University Press, Cambridge (1992).