Analysis, Design, and Implementation of an AC Bridge for Impedance Measurements

Selim S. Awad, *Member, IEEE*, Natarajan Narasimhamurthi, *Member, IEEE*, and William H. Ward

***Abstract*—The paper reviews the digital AC bridge. A more accurate and thorough analysis of the bridge is presented. Based on the analysis a simple adaptive algorithm is proposed in order to ensure fast conversion. In addition, a new design of the bridge, based on the TMS 320C25, is given. The system is implemented with the Dalanco Spry model 25 DSP board, on a software system designed whereby an IBM PC host computer provides control of the bridge. Experimental results are presented.**

I. Introduction

DUTTA *et al.* [1] proposed the idea of controlling an AC bridge by means of a microprocessor. They utilized the least mean square (LMS) adaptive algorithm to balance the bridge. However, their analysis is not accurate enough as will be discussed in the following section. In addition, their implementation was based on the Intel 8085 microprocessor (an 8-bit machine) which has very limited capabilities. In this paper we will reanalyze the bridge to obtain the appropriate equations for the controlling parameters of the bridge for fast convergence. To this end an adaptive algorithm is proposed to ensure fast convergence of the bridge. In addition, a new design and implementation of the bridge, based on the TMS 320C25 DSP (also referred to as the 'C25) using a commercial evaluation board [2], containing a 'C25, a 12-bit A/D, a 12-bit D/A, and an interface circuit are presented.

The board, a Dalanco-Spry model 25 data acquisition and signal processing board, is designed to function as a coprocessor in an IBM PC compatible computer. An additional module containing two matched 12-bit DACs and the bridge circuit was constructed. In order to control the bridge, two software programs were developed: one for the 'C25 board, and the other for the PC computer. The performance of the bridge in real time was evaluated by measuring a wide range of known impedances.

II. Bridge Analysis

A popular method to measure an unknown impedance is based on the AC bridge shown in Fig. 1. In this bridge V_r and V_x are two sinusoidal voltage sources with the same frequency ω_0, but different amplitudes and phase shifts. The reference voltage source V_r is of constant amplitude A and zero phase shift. However, V_x has a variable amplitude and phase shift.

Manuscript received June 28, 1993; revised May 16, 1994.

S. S. Awad and N. Narasimhamurthi are with the Department of Electrical and Computer Engineering, University of Michigan, Dearborn, MI 48128-1491 USA.

W. H. Ward is with the Ford Motor Company, Dearborn, MI 48126 USA.

IEEE Log Number 9406571

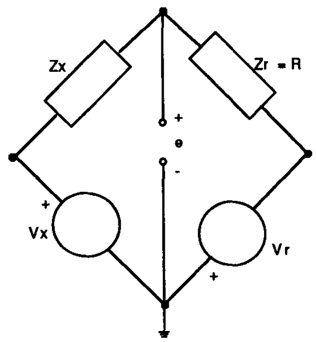


Fig. 1. AC bridge. The reference impedance Z_r is shown as R, a pure resistor.

Thus V_r and V_x can be written as follows

$$V_r = A\sin(\omega_0 t)$$
$$V_x = B\sin(\omega_0 t + \phi) \tag{1}$$

where B and ϕ can be controlled to balance the bridge. The other two elements of the bridge are the unknown impedance Z_x and the reference impedance Z_r. For simplicity, Z_r is chosen to be resistive, i.e., $Z_r = R$. When the bridge is balanced (the voltage $e = 0$), the unknown impedance is given by

$$Z_x = R\frac{B}{A}\angle\phi \quad \text{at the frequency } \omega_0. \tag{2}$$

V_x can be expressed in terms of the in-phase and quadrature components (the quadrature Fourier series) [1] as follows

$$V_x = W_1 A\sin(\omega_0 t) + W_2 A\cos(\omega_0 t) \tag{3}$$

where W_1 and W_2 are the weights of the in-phase and quadrature components respectively. With B and ϕ (1) expressed in terms of W_1 and W_2 (2) can be rewritten as

$$Z_x = W_1 R + jW_2 R \quad \text{at balance.} \tag{4}$$

$W_1 R$ and $W_2 R$ are the real and imaginary components of Z_x, respectively.

III. Balancing the Bridge

To balance the bridge, one starts with arbitrary W_1 and W_2, and iteratively modifies these to force $e(t)$ to zero. It can be

0018–9456/94$04.00 © 1994 IEEE

shown that the instantaneous error signal is given by:

$$\begin{aligned} e(t) &= |\beta_1|A[W_1\sin(\omega_0 t+\theta_1)+W_2\cos(\omega_0 t+\theta_1)] \\ &\quad -|\beta_2|A\sin(\omega_0 t+\theta_2) \\ &= |\beta_1|A\big[(W_1-W_1^b)\sin(\omega_0 t+\theta_1) \\ &\quad +(W_2-W_2^b)\cos(\omega_0 t+\theta_1)\big] \end{aligned} \tag{5}$$

where

$$\begin{aligned} \beta_1 &= |\beta_1|\angle\theta_1 = \frac{Z_r}{Z_x+Z_r} \\ \beta_2 &= |\beta_2|\angle\theta_2 = \frac{Z_x}{Z_x+Z_r} \\ W_1^b &= \frac{|\beta_2|}{|\beta_1|}\cos(\theta_2-\theta_1) \\ W_2^b &= \frac{|\beta_2|}{|\beta_1|}\sin(\theta_2-\theta_1). \end{aligned} \tag{6}$$

It is interesting to note the presence of θ_1 in (5). It represents the phase shift between the two sinusoidal inputs and the error signal. The final form of the error in (5) can also be obtained by using superposition. The section on convergence analysis will show that the speed of convergence depends on this phase shift and becomes progressively worse as this angle deviates from zero. Note that if both the unknown impedance and the reference impedance are passive, this angle is strictly between $-\pi/2$ and $+\pi/2$. In what follows we shall assume that this angle is between these two limits.

The approach to balancing the bridge is as follows: The error voltage $e(t)$ is sampled periodically with period T. At each sampling interval, the weights W_1 and W_2 are adjusted using the steepest descent direction to minimize $e^2(t)$, i.e., at the k-th sampling time, W_1 and W_2 are changed as (LMS algorithm):

$$\begin{aligned} W_1(kT+T) &= W_1(kT)-\mu\frac{\partial e^2(t)}{\partial W_1}\bigg|_{t=kT} \\ W_2(kT+T) &= W_2(kT)-\mu\frac{\partial e^2(t)}{\partial W_2}\bigg|_{t=kT} \end{aligned} \tag{7}$$

where μ is a small constant. From (5) and (7) we get

$$\begin{aligned} W_1((k+1)T) &= W_1(kT)-2u\frac{e(kT)}{A}\sin(\omega_0 kT+\theta_1) \\ W_2((k+1)T) &= W_2(kT)-2u\frac{e(kT)}{A}\cos(\omega_0 kT+\theta_1) \end{aligned} \tag{8}$$

where u is a small dimensionless quantity. Note that (8) contains the unknown angle θ_1 which was not considered in [1].

IV. Convergence Analysis

In principle θ_1 can be treated as an additional parameter to be estimated and the LMS algorithm suitably modified. The disadvantage with such an approach is that the number of computations between samples is increased, which in turn decreases the operating frequency of the bridge. See Section VI for further details. However, the following theorem shows that if u is sufficiently small, a crude estimate of θ_1 can be used in (8). In what follows we assume that there are N samples per period or

$$T=\frac{2\pi}{N\omega_0}. \tag{9}$$

Theorem 1: Suppose that θ_1^e is an estimate of θ_1 satisfying $|\theta_1^e-\theta_1|<\pi/2$. Then for sufficiently small u, the iterative scheme

$$\begin{aligned} W_1((k+1)T) &= W_1(kT)-2u\frac{e(kT)}{A}\sin(\omega_0 kT+\theta_1^e) \\ W_2((k+1)T) &= W_2(kT)-2u\frac{e(kT)}{A}\cos(\omega_0 kT+\theta_1^e) \end{aligned} \tag{10}$$

(viz. (8) with θ_1 replaced by θ_1^e) converges to $W_1\to W_1^b$ and $W_2\to W_2^b$.

Furthermore, for u sufficiently small, $E^2(n)$ (sum of the error squared over the n-th cycle) defined as

$$E^2(n)=\sum_{k=nN}^{nN+N-1} e^2(kT) \tag{11}$$

satisfies

$$E^2(n+1)\le\mu^2E^2(n),\qquad \mu^2<1 \tag{12}$$

i.e. $E^2(n)$ converges geometrically to zero.

Proof: Let $\Delta(k)$ be defined as

$$\Delta(k)=\begin{bmatrix}W_1(kT)-W_1^b\\ W_2(kT)-W_2^b\end{bmatrix}. \tag{13}$$

$e(kT)$ and $\Delta(k+1)$ can be written as

$$\frac{e(kT)}{A}=|\beta_1|[\sin(\omega_0 kT+\theta_1),\ \cos(\omega_0 kT+\theta_1)]\Delta(k) \tag{14}$$

$$\Delta(k+1)=[\mathbf{I}-uM(k)]\Delta(k), \tag{15}$$

where (16), (see the bottom of the page) and $\mathbf{I}$ is the 2×2 identity matrix. Similarly

$$\begin{aligned} \Delta(k+2) &= [\mathbf{I}-uM(k+1)]\Delta(k+1) \\ &= [\mathbf{I}-uM(k+1)][\mathbf{I}-uM(k)]\Delta(k) \\ &= \big[\mathbf{I}-uM(k+1)-uM(k)+O(u^2)\big]\Delta(k), \end{aligned} \tag{17}$$

where $O(u^2)$ represents terms involving u^2 and possibly higher powers of u. In general,

$$\Delta(k+n)=\left[\mathbf{I}-u\sum_{i=k}^{k+n-1}M(i)+O(u^2)\right]\Delta(k). \tag{18}$$

When $n=N$, the summation in (18) is over one complete cycle and (18) simplifies to

$$\Delta(k+N)=\Phi\Delta(k)=\big[\Phi_0+O(u^2)\big]\Delta(k) \tag{19}$$

$$M(k)=2|\beta_1|\begin{bmatrix}\sin(\omega_0 kT+\theta_1)\sin(\omega_0 kT+\theta_1^e) & \cos(\omega_0 kT+\theta_1)\sin(\omega_0 kT+\theta_1^e)\\ \sin(\omega_0 kT+\theta_1)\cos(\omega_0 kT+\theta_1^e) & \cos(\omega_0 kT+\theta_1)\cos(\omega_0 kT+\theta_1^e)\end{bmatrix} \tag{16}$$

where

$$\Phi_0 = \begin{bmatrix} 1 - Nu|\beta_1|\cos(\theta_1^e - \theta_1) & 1 - Nu|\beta_1|\sin(\theta_1^e - \theta_1) \\ Nu|\beta_1|\sin(\theta_1^e - \theta_1) & -Nu|\beta_1|\cos(\theta_1^e - \theta_1) \end{bmatrix}. \quad (20)$$

Hence

$$\|\Delta(k+N)\|^2 \leq \|\Phi\|^2 \|\Delta(k)\|^2. \quad (21)$$

$\|\Phi_0\|^2$ is easily seen to be $1 + N^2u^2|\beta_1|^2 - 2Nu|\beta_1|\cos(\theta_1^e - \theta_1)$ and is strictly less than unity for small u. Hence, in (19), the linear term in u dominates $O(u^2)$, and Δ converges to zero.

The proof of the second part of the theorem is similar. By similar computations which are omitted, we get

$$E^2(n) = K(u)\|\Delta(nN)\|^2 \quad (22)$$

where $K(u)$ is a function of u and θ_1^e. Hence

$$E^2(n+1) \leq \|\Phi\|^2 E^2(n) \quad (23)$$

and for sufficiently small u, $\|\Phi\|^2 < 1$. Q.E.D.

Remarks:

1) In the above theorem, the speed of convergence is governed by the norm of the matrix Φ in (19). It can be shown that although Φ depends on k, its norm does not. This can be seen from the observation that Φ depends on when in the cycle we compare $\Delta(k+N)$ and $\Delta(k)$. However, independent of where we start the comparisons, Δ should either converge to zero or diverge at the same geometric rate. Hence the norm of Φ can not depend on k. The plot of the norm of this matrix as a function of $\theta_1 - \theta_1^e$ is given in Fig. 2, which shows the importance of having a good estimate of θ_1.
2) If it is known that the unknown impedance is passive, then the choice of $\theta_1^e = 0$ ensures convergence for small enough u. If, in addition, the sign of the reactive component of the unknown impedance is known, then we can choose $\theta_1^e = \pm\pi/4$ with the sign chosen appropriately.
3) The above analysis also shows the importance of choosing the reference impedance to be of the same order of magnitude as the impedance to be measured. If, for example, $|Z_r| \ll |Z_x|$, then $|\beta_1| \ll 1$, and θ_1 is the negative of the angle of Z_x, and could potentially be close to $\pm\frac{\pi}{2}$. This will result in very slow convergence. Conversely, if $|Z_r| \gg |Z_x|$, both W_1^b and W_2^b will be small, and will result in loss of accuracy in the measurement.

V. An Implementation Using an Adaptive Algorithm

In the previous section, we showed that the algorithm converges for sufficiently small values of u. However, the rate of convergence deteriorates when u is very small and/or the estimate of θ_1 is not accurate enough. An adaptive scheme that overcomes both these problems is to select a u so that the rate of convergence is acceptable if $\theta_1 - \theta_1^e$ is small enough and to adaptively change θ_1^e. The steps of the algorithm are as follows:

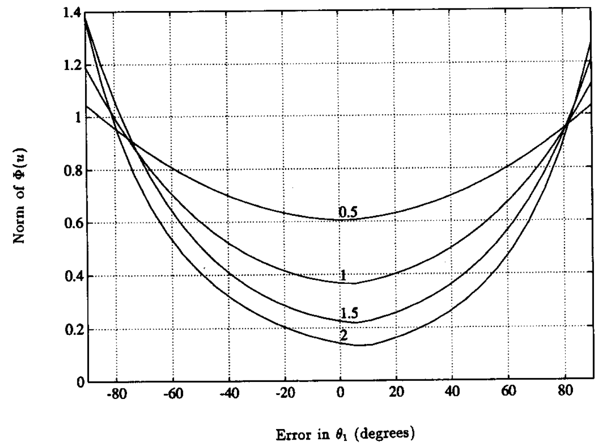


Fig. 2. Norm of Φ as a function of the error in θ_1 for different values of $Nu\beta_1$.

Step 0: Select $Z_r = R$ to be of same order of magnitude as Z_x. This ensures that $|\beta_1|$ is at least 0.25. Pick u so that the algorithm will converge rapidly whenever $|\theta_1 - \theta_1^e|$ is small. An acceptable choice is $u = 2/N$. From Fig. 2 the norm of Φ would be less than 0.75 for a wide range of $|\theta_1 - \theta_1^e|$.

Step 1: At the beginning of each cycle, save the values of W_1 and W_2. Perform the iterations (10) over one complete cycle and also compute E^2, the sum of e^2 over the cycle.

Step 2: If E^2 is smaller than some threshold, stop; the bridge is balanced; else continue with step 3.

Step 3: See if the iterations over one complete cycle provide sufficient improvement by comparing the sum of the squares of $e(k)$ in that cycle with that of the previous cycle. If the improvement is acceptable, e.g., a decrease by 0.75, then go back to step 1.

Step 4: If E^2 is larger than some threshold (so as not to be sensitive to noise), increase θ_1^e by $2\pi/N$, reset W_1 and W_2 to the values at the beginning of the cycle and go back to step 1.

In the above algorithm, θ_1^e will get sufficiently close to θ_1 in at most N cycles. This is the "hunting phase" of the algorithm. Once the error $|\theta_1 - \theta_1^e|$ is small enough, the error decreases by at least 0.75 each cycle, so that after k cycles, the error would decrease by at least 0.75^k. For example, for $N = 64$ and an operating frequency of 200 Hz, the error will drop by $0.75^{(200-64)} \approx 10^{-17}$ in one second. Even with a more conservative factor of 0.9, the error will drop by a factor of $0.9^{(200-64)} \approx 10^{-6}$ in the first second, and by a factor of $0.9^{200} \approx 10^{-9}$ every second after the hunting phase. These numbers are used for illustrative purposes only. In practice, there will be a small residual error due to round-offs, finite precision arithmetic and measurement errors. Thus, the algorithm will terminate in step 2 well before such small errors are encountered.

VI. System Description

A. Hardware Description

The block diagram of the realized AC bridge is shown in Fig. 1, where the bridge is directly controlled by means of

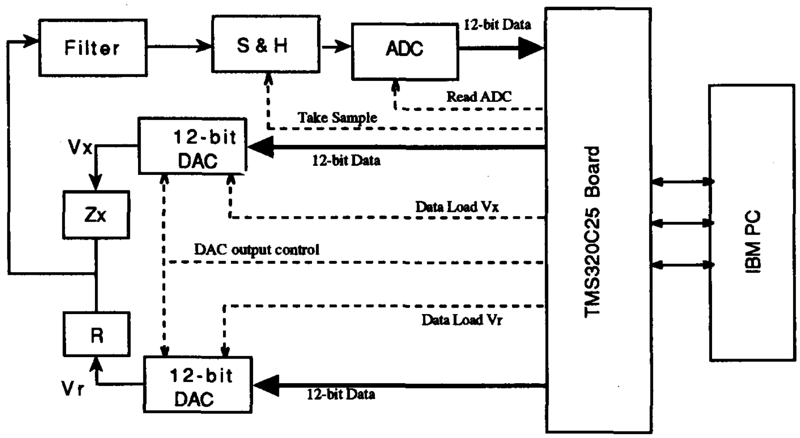


Fig. 3. Schematic block diagram of the hardware.

the 'C25-based board. In addition, an IBM PC compatible computer is used to control the 'C25 and to display the results in a convenient format.

The sinusoidal voltage sources V_r and V_x are produced, and fed to two 12-bit DACs (Analog Devices AD667). The DACs are controlled by a Texas Instruments TMS 320C25 DSP which runs the LMS algorithm. A simple method to synthesize the needed sinusoidal waveforms has been adopted. This method, which is known as the "look-up table" method, is based on the idea of storing n samples of one cycle of a sine wave [3]. The synthesized frequency is given by

$$\omega_0 = \frac{2\pi}{NT}. \tag{24}$$

Thus, it is possible to change ω_0 by either changing N or T. Moreover, it is possible to simultaneously synthesize the sine and cosine waveforms by retrieving their corresponding values from the look-up table (stored in the DSP RAM). The error signal $e(t)$ is first filtered by an anti-aliasing low-pass filter to remove all frequency components above the Nyquist frequency and then is read by a 12-bit A/D converter (Harris HT-774A). The updated weights W_1 and W_2, which are needed to synthesize V_x, are calculated. The IBM PC compatible computer (shown in Fig. 3) is used to control the 'C25 and to continuously interpret and display the results that the LMS routine produces. The results are merely the coefficients W_1 and W_2. Due to the interface between the 'C25 and the PC it takes several cycles of the LMS routine to transfer W_1 and W_2 to the PC. The values of W_1 and W_2 will continuously change as the system balances the bridge. Therefore, copies of these values are stored in the 'C25's RAM, and are not updated until the transfer is complete. Once the AC bridge reaches balance (i.e., $e(t) \approx 0$), the final values of W_1 and W_2 are determined. The value of Z_x is calculated using (4). Knowing the frequency of the sinusoidal sources allows the calculation of the equivalent resistive and reactive parts of Z_x (which can be modeled as two ideal impedances in series).

B. Software Description

Two software packages have been developed to operate the system. The first package is written for the TMS 320C25 processor in assembly language to execute the LMS routine. Floating-point arithmetic was chosen in order to preserve the accuracy of the sums and products of the intermediate steps of the LMS routine. While the 'C25 has basic instructions that make the use of floating-point numbers easy, it does not directly support them. Therefore, addition and multiplication are performed by software routines. Even though the TMS 320C25 can process 10 million instructions per second (10 MIPS), these arithmetic routines consume most of the time required between each sample. This results in the need to compromise between the number of samples per cycle of the sinusoidal voltage sources, and the frequency of these sources. Since the number of machine cycles required to complete the LMS routine varies at random (depending on the floating-point operations that are required), a counter/timer internal to the 'C25 is used to create uniform sampling. Thus the LMS routine is executed when the 'C25 is interrupted by the timer. When the 'C25 is not processing the LMS routine, it is in an idle state awaiting the next interrupt. Since the number of machine cycles required to complete the LMS routine depends on the floating-point operations that are required, a counter/timer internal to the 'C25 is used to create uniform sampling. The sampling is done after the count-down interrupt of the timer, in order to avoid jitter in the generated signals. Thus, the LMS routine is executed when the 'C25 is interrupted by the timer and the processor performs the required computations and is ready for the next sample. The processor then enters a wait state till the next timer interrupt. Since the times between the updates of the weights and generated signals are controlled by the timer interrupt, variations in the computation time due to conditional execution of subroutines are avoided.

The second software package is needed for the IBM compatible computer. It is written in Microsoft Quick Basic language version 4.5. The program is used to control the 'C25, interpret and display the results and provide the means to load the LMS routine into the 'C25 memory from the PC's disk drive. Also, the program calculates the various constants such as the number of machine cycles between samples and the sine table for the sinusoidal waveform. These are downloaded to the 'C25's RAM when the processor is halted. This allows the user to adjust and control the operation of the LMS routine prior to running it.

C. Experimental Results

The performance of a prototype AC bridge was tested by measuring a set of impedances formed using resistors and capacitors. The impedances were measured using the Hewlett-Packard 4192LF impedance analyzer at 203.45 Hz. The instrument has an accuracy of $\pm 1\%$. The results obtained by using the HP 4192LF and the digital bridge are shown in Table I. We list below some of the factors that affect the accuracy of the measurements.

1) Excessive electronic noise is generated in the system. This noise can be reduced by careful shielding of the impedances and by a complete decoupling of the analog and digital grounds.
2) Mismatch in the anti-aliasing analog filters introduces phase errors in the two signals. Thus, the use of matched analog filters and compensations for the phase errors would improve the accuracy.
3) Timing jitter introduced during the sampling process is another source of errors. Averaging techniques may be employed to reduce the effect of the jitter.
4) The number of samples per cycle (and hence the size of the table) used to generate the sine waves affect the accuracy of the magnitude and phase of the generated sine wave. The experimental setup used a relatively small value of 64. Increasing the number of samples will result in increased accuracy. However, the number of samples is limited by the minimum time the LMS needs to process the data and by the frequency of the generated waveform.
5) Another factor that affects the accuracy of the bridge is the "headroom" available in the D/A voltage source, i.e., its amplitude range. Balancing the bridge may require V_x to be several times larger or smaller than V_r depending on the ratio of Z_x to Z_r. At the high end of the ratio, the D/A convertor may saturate and not provide the required voltage, while at the low end the lack of resolution will not permit the balancing of the bridge. Thus, it is important to make sure that Z_x and Z_r are of approximately the same order of magnitude and select the magnitude of V_r to ensure that the full range of the D/A convertor is utilized in generating V_x. In the experimental setup the choice of V_r and Z_r was done manually and shown in the last two columns of Table I. In our next implementation of the bridge we will let the choice of these two parameters be controlled by the bridge.

TABLE I
MEASURED IMPEDANCES ANDPERCENT DIFFERENCE FROM REFERENCE VALUES. REFERENCE VALUES MEASURED USING HEWLETT-PACKARD 4192 LF IMPEDANCE ANALYZER OPERATING AT 203.4505 HZ. THE INSTRUMENT HAS AN ACCURACY OF $\pm 1\%$. OPERATING FREQUENCY OF THE BRIDGE WAS 203.45 HZ AT 64 SAMPLES PER CYCLE

Reference Value		Measured Impedance		$\frac{\text{Reference}-\text{Measured}}{\text{Reference}}$		Parameters	
Magnitude	Phase	Magnitude	Phase	Magnitude	Phase	V_r	Z_r
100.6 Ω	0°	100.4 Ω	0°	0.199%	0%	4.5V	1 k Ω
1000.0 Ω	0°	998.6 Ω	0°	0.140%	0%	2.0V	1 k Ω
10.08k Ω	0°	10.03k Ω	0°	0.496%	0%	1.0V	10 k Ω
64.30 k Ω	0°	63.00k Ω	0°	2.02 %	0%	0.75V	10 k Ω
5.027 Ω	−84.52°	5.00 Ω	−84.24°	0.537%	0.331%	4.0V	10 k Ω
8.377 Ω	−81.00°	8.32 Ω	−81.70°	0.680%	-0.864%	4.0V	10 k Ω
78.8 Ω	−84.7°	78 Ω	−85.6°	1.015%	-1.063%	2.0V	1 k Ω
85.0 Ω	−85.01°	83.9 Ω	−85.23°	1.294%	-0.259%	4.0V	10 k Ω
877.0 Ω	−83.9°	859.0 Ω	−85.34°	2.052%	-1.716%	4.0V	1 k Ω
7.97k Ω	−89.9°	7.99k Ω	−89.86°	-0.251%	0.044%	2.0V	10 k Ω
485 Ω	−47.5°	480 Ω	−47.5°	1.031%	0%	3.0V	10 k Ω
558 Ω	−4.5°	555.8 Ω	−4.35°	0.394%	3.333%	4.0V	10 k Ω
625 Ω	−47.5°	480 Ω	−47.5°	1.031%	0%	2.0V	10 k Ω
3.54 k Ω	−77.8°	3.54k Ω	−77.7°	0%	0.130%	4.0V	10 k Ω
4.3k Ω	−32.5°	4.3k Ω	−32.5°	-0%	0%	2.0V	10 k Ω
27.2k Ω	−7.1°	26.89k Ω	−7.2°	1.14%	-1.14%	1.0V	10 k Ω

VII. CONCLUSION

This paper is an extension of the digital AC bridge proposed in [1]. We have shown that the LMS algorithm for updating the parameters depends on the unknown impedance, a result not covered in [1]. We have theoretically shown that an approximation to the true LMS algorithm will converge for sufficiently small "step length". A new design of the bridge based on a DSP processor, namely the TMS 320C25, is presented. Preliminary experimental results show that the system provides acceptable accuracy and speed. However, the maximum frequency of operation is limited by the processor speed and the need for floating-point computations. We plan to investigate convergent algorithms that avoid floating-point computations and the use of more advanced TMS 320 processors.

In the section on convergence, we have provided a theoretical basis for the speed of convergence and for the choice of the step size for the iterative process. We have shown that the speed of convergence and the step size depend on the quantity θ_1 which is a measure of how close the unknown impedance Z_x and the known impedance Z_r are. If the unknown impedance is highly reactive and the known impedance is a resistor, this angle θ_1 could approach $\pm\frac{\pi}{2}$ if the magnitudes of the impedances are orders of magnitude apart. As this angle deviates from zero, the algorithm would require smaller step sizes and result in a slower convergence rate, and in extreme cases to outright divergence. This observation, we feel, is an important contribution of the paper.

Finally, one may question the justification of using the LMS algorithm when simpler methods exist such as using the A/D convertor as a "vector" voltmeter when two (conceptually complex) measurements are taken to compute the unknown impedance. However, since the LMS algorithm uses several measurements over a complete cycle, the effects of measurement noise are minimized. Also, the computations used in the LMS algorithm are simple and can be implemented easily in a signal-processing unit.

ACKNOWLEDGMENT

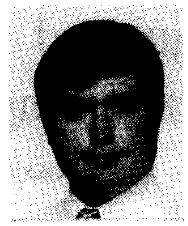
The authors would like to thank the reviewers of this paper for their valuable input which has strengthened the revised version of the paper.

REFERENCES

[1] M. Dutta, A. Rakshid, S. N. Battacharyya and J. K. Choudhury, "An application of the LMS adaptive algorithm for a digital AC bridge," *IEEE Trans. Instrum. Meas.*, vol. 36, pp. 894–897, 1987.
[2] Dalanco-Spry, 89 Westland Ave, Rochester, NY 14618.
[3] M. Schanenberger and S. Awad, "The implementation of a digital sine wave oscillator using TMS 320C25: distortion reduction and applications," *IEEE Trans. Instrum. Meas.*, vol. 39, pp. 870–873, 1990.

Selim Saad Awad (M'86) was born in Cairo Egypt, on August 7, 1951. He received the B.Sc. degree in Electronics and Communications from Cairo University, in 1973, the B.Sc. degree in Applied Mathematics from Ain-Shams University, and the Ph.D. degree from the Polytechnic Institute of Grenoble, France, in 1983.

He has been an associate professor at the University of Michigan since 1989. His current research includes A/D converter testing, effect of timing jitter on signal measurements, speech application in instrumentation, and digital signal processing applications.



William Ward was born in Livonia, Michigan. He received the B.S.E.E. degree from Lawrence Technological University in 1986, and the M.S.E. degree in Electrical Engineering from the University of Michigan, in 1991.

Since 1987, he has been with the Ford Audio Systems unit of Ford Motor Company. His work has involved vehicle-level electromagnetic compatibility for audio systems, and the design of radios.

He is a member of Eta Kappa Nu and Tau Beta Pi, holds an amateur radio license KB8NT, and is registered as a Professional Engineer in the State of Michigan.

Natarajan Narasimhamurthi (S'75, M'93) was born September 21, 1952, in Bangalore, India. He received the B.Tech. degree in electronics from the Indian Institute of Technology, Madras, India, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the University of California, Berkeley, in 1974, 1978, and 1979, respectively.

Since 1987, he has been with the University of Michigan-Dearborn. His research interests are in applied artificial intelligence, modern control theory and signal processing.