Signal Preprocessing Techniques for an Electronic Tongue

F. Javier Acevedo, Arantzazu Narvaez, Saturnino Maldonado, Phillip Siegmann, Sergio Lafuente, Hilario Gómez
Department of Signal Theory and Communications
Universidad de Alcalá de Henares
Escuela Politécnica Superior. Campus Universitario. 28805 Alcalá de Henares
SPAIN

Abstract: - Electronic tongues are systems composed by an array of sensors plus a pattern recognition for the evaluation of liquids. In our case, voltametric sensors are used due to their selectivity and advantages. However, as is explained in this paper, this kind of sensors presents several problems that can be partially solved by adequate signal processing techniques. The system is completed with a neural network as pattern recognition system plus a scheme for taking account all the information obtained from the sensors

Key-Words: - Electronic tongue, electronic nose, signal processing, pattern recognition, enzyme sensors.

1 Introduction

Rapid and low-cost methods enabling quality assessment of food products are of great interest for the industry. Traditionally used analytical techniques such as HPLC are selective and reliable but they are quite expensive, require experienced operators and are difficult to be automated to develop on-site applications.

During last few years a great deal of work has been done in the development of systems based on an array of sensors plus a pattern recognition system. Specially interesting are those systems that try to imitate human senses such as those that detect volatile components, known as "electronic noses" [1] and those developed for the evaluation of liquids known as "electronic tongues".

Basically two approaches in the design of electronic tongues devices are followed by using either potentiometric or voltametric sensors [2]. Voltametric sensors have been reported to show several advantages related to robustness and versatility. However, the major drawback of both kind of sensors is the baseline drift and lack of stability [3]. This is a major problem because it makes the response non stationary and the performance of the system under a real situation is reduced in such a way that results can be out of sense.

The purpose of this paper is to show how preprocessing the signals obtained from the sensor array can be used to improve the performance of these systems. In particular, we have applied several preprocessing techniques to solve problems derived from an array of sensors that try to classify Madrid wines into their different classes: red, white and rose. Signal processing is applied to discriminate useful sensors signals, to detect peaks that are going to be of maximum relevance in the information of the signal

and to avoid drift effects in the measurement of the kinetic of the signal.

Finally, a classifier is used to test the performance in the identification. In this work we have used a MLP neural network [4].

Although the application studied in this work is the classification of wines, the method here proposed is also valid for a wide range of applications.

2 Sensors Signals

In the system under study, responses of an array of six biosensors are measured. These sensors are made of carbon paste which several enzymes have been added to [5]. The main advantage of this kind of sensors is that they can be made very easily with very low cost of manufacturing.

The system works as follows: There is a continuous flow of a tampon substance with baseline responses from the sensors. When the analyte, in our case wine, is injected there is a reaction from the sensors that is due to an oxidation or reduction process. This reaction makes a signal peak that can be positive or negative depending on if there is a reduction or an oxidation. The sensors signals recovers baseline depending on the analyte, giving all the process, peak and baseline recovery, useful information about the substance that has been injected. The process is repeated, giving a pseudo-periodic signal over the time.

However, if an automated system is required, some issues must be taken account and corrected previous to the measurement process.

2.1 Signal Measurements

For every cycle of each sensor we are going to have an oxidation/reduction peak signal that contains rich

information to be evaluated. Following the references of [6] we can separate them in those related to the level of response and those that are related to the kinetic of the substance under study. In figure 1 it is shown the different measures that under a theoretical point of view should be extracted from every cycle of each sensor.
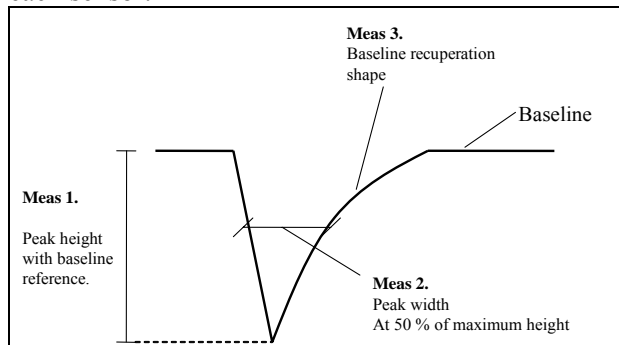

Fig 1. Measures to be taken per cycle.

2.2 Measurement Difficulties

In figure 2 is represented a real experiment done with the array of sensors. One can appreciate in this realization several issues to be corrected.
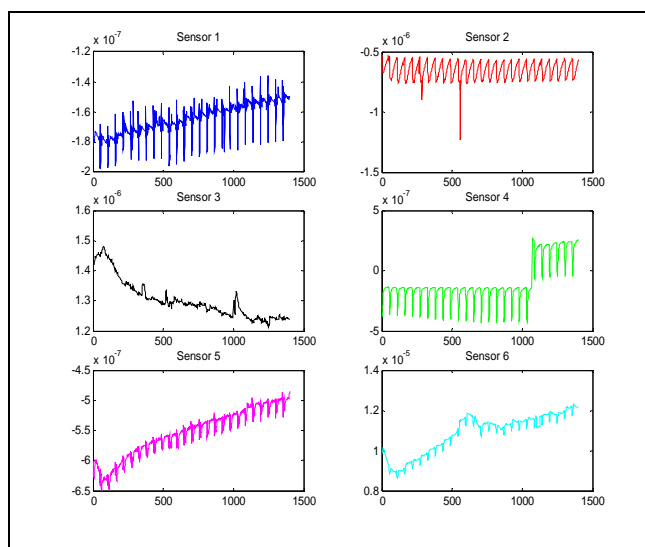

Fig 2. Real representation of the six sensors

2.1.1 Sensor Outage

In figure 1 it can be seen how the aforementioned process give us so many peaks as injections have been carried out.

However, sensor 3 is clearly not responding to this process and only noise is reflected. As this experiment has been repeated under the same study several times, we can discard that the enzyme related to sensor 3 does not respond to the analyte under study. In fact, what it really happens is that due to the continuous use of the sensors, the carbon paste gets a lot of impurities that causes the sensor do not respond to the analyte injection.

Under this situation, an automated system must detect this kind of fault and taking it account, in order to give and alarm to change this sensor and also to not consider the information contained in the sensor for pattern recognition purposes.

2.1.2 Baseline Drift

In figure 1 can be also appreciated the baseline drift. Only sensor 2 keeps and stable baseline whereas the rest of the valid sensors, as explained sensor 3 is out of service in this case, have a clear baseline deviation.

Unfortunately, the drift behavior is unpredictable and can not be modeled by a mathematical function. This lack of stability causes a serious problem. However, it should be noted that we are interested in the baseline per cycle rather than in the global baseline. That is, if we appreciate sensor 4 response we can realize that the baseline changes abruptly, but the measurements described in figure 1 keep stable. So, the important point here is to determinate what is the baseline per cycle.

2.1.3 Buffer Substances

Sometimes a buffer substance is used to control the live activity of the enzymes. In such a case, instead of the analyte, wine in our case, the substance used makes a huge response in all the sensors. In figure 2 there was no buffer used, but as can be appreciated in figure 3, at the end of the process, it has been used a control substance in the two last cycles, giving a bigger height of the peaks compared to the rest, where the analyte was wine. These cycles must be removed because they do not contain any useful information about the substance under study.
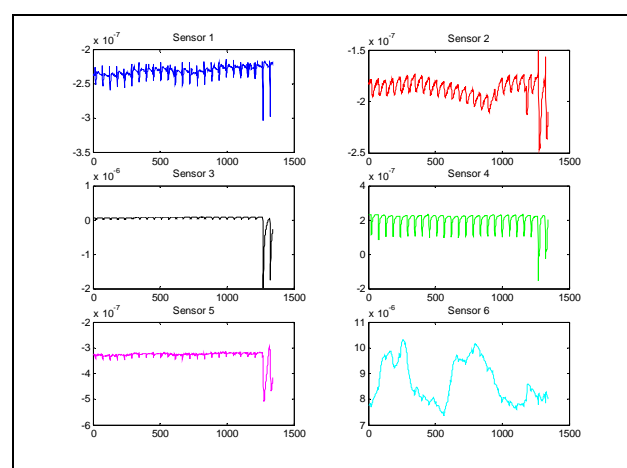

Fig 3. Real representation using buffer substance at the end

3 Signal Processing and Pattern Recognition

As have been described in the previous section there are some problems that need to be solved to guarantee the functionality of the system. In this section the solutions in the field of the signal processing are exposed. Finally, results of a classification experiment are shown using a MLP-NN.

3.1 Sensor Outage Detection

The main feature of the signals selected as useful information is the periodicity of the peaks as a response to the analyte substance. As the system is fully automated the injection is done in equal time intervals. On the other hand, we have to think here also about drift and control substances presence that makes the signal to be no so periodical. Following these ideas and in order to discriminate automatically between those signals with useful information and those signals with a strange behaviour, two methods are proposed.

3.1.1 Method Based on the FFT

This method is straightforward once that we realize that the useful signals are pseudo-periodical due to the constant time between injections. The DFT of a useful signal should give us peaks around the related point in the spectrum and also in its harmonics. In figure 4 it can be appreciated a valid signal and its spectrum.
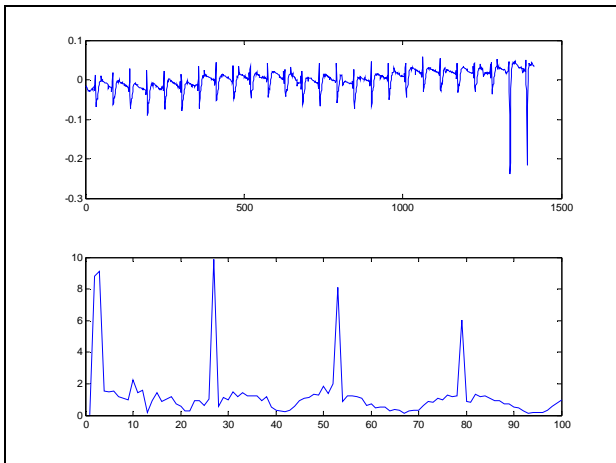

Fig 4. A useful signal and its spectrum.

On the other hand, in figure 5 a non useful signal and its spectrum are shown. As can be appreciated there is no need to calculate the whole spectrum but only those coefficients that are expected to be maximum when the signal is useful.
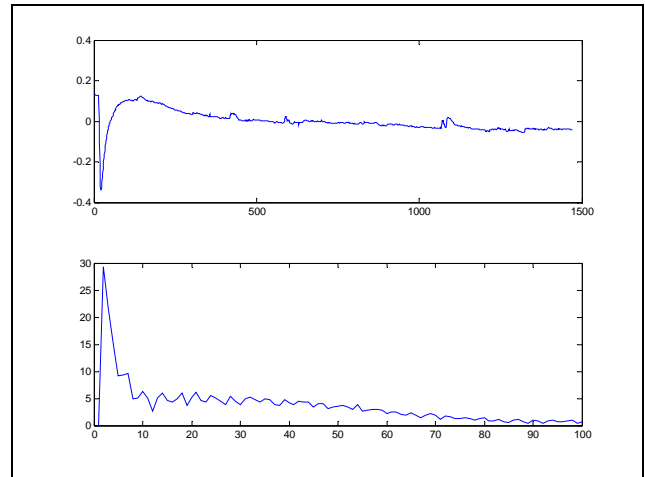

Fig 5. A non useful signal and its spectrum.

The decision about if a signal is useful or not is taken depending on the parameter α calculated as:

$$\alpha = E\left[\sum_{i=1}^{N}\left|X\left[i\Omega_0\right]\right|^2\right] \ i = 1,2,...N \quad (1)$$

Where N is the number of harmonics considered, X[k] is the kth coefficient of the DFT applied over a samples window and Ω_0 is the principal harmonic associated to the analyte injection period. Note that we are making several FFTs over the same signal and calculating the mean in order to avoid those control peaks and noise samples.

If the parameter α is greater than a threshold, then the signal is considered as useful information, whereas if this parameter is lower the system should give an alarm because that means that the sensor is out of service.

3.1.2 Method Based on Correlation

Although the method based on the DFT seems to be optimum it has some problems regarding the drift. Another method to know if our signal shows oxidation/reduction processes is to compare it with a template. The covariance is defined as:

$$Cov(\boldsymbol{x_1},\boldsymbol{x_2}) = E\left[(\boldsymbol{x_1} - \mu_1)(\boldsymbol{x_2} - \mu_2)^T\right] (2)$$

Where $\boldsymbol{x_1}$ and $\boldsymbol{x_2}$ are the vectors associated to the signals $x_1[n]$, $x_2[n]$ and μ_i is the mean of the signal $x_i[n]$. Then the correlation coefficient is:

$$r = \frac{cov(\boldsymbol{x},\boldsymbol{p})}{\sqrt{cov(\boldsymbol{x},\boldsymbol{x})cov(\boldsymbol{p},\boldsymbol{p})}} \quad (3)$$

Where **x** is the signal under study and **p** is the template used to calculate the correlation coefficient. It should be mentioned that opposite to the FFT method, where the module is invariant to signals shift, with this method we have to be aware of the possible signal shifts. Thus, we have not only to use a template, but also shifted versions of this template. The coefficient is calculated then as:

$$r = \underset{m\,max}{arg} \frac{cov(\boldsymbol{x}, \boldsymbol{p}_m)}{\sqrt{cov(\boldsymbol{x}, \boldsymbol{x})cov(\boldsymbol{p}_m, \boldsymbol{p}_m)}} \quad (4)$$

Where $\boldsymbol{p}_m$ is the template shifted m samples. As we are interested in detecting a signal with peaks, templates are as simple as peaks separated by T samples. In this method we have also considered to take windowed signals and templates of a length of the window.

So, the whole signal is divided in windowed signals and the correlation coefficient as defined in eq. 4 is calculated for each of them. Then, the parameter α is defined as:

$$\alpha = E[r_i], \; i = 1, 2, 3 ... l \quad (5)$$

Where r_i is the coefficient calculated over the window i and l is the number of windows.

3.1.3 Results for Sensor Outage Detection

In table 1 it is shown the results obtained with the two methods described. Both methods were tested on 420 signals where it was known if they belong to useful signals or not. Results show us the probability of success in detecting a useful signal.

	White	Red	Rose
FFT Method	80 %	84 %	84 %
Correlation Method	92 %	93 %	87 %

Table 1. Comparing methods for sensor outage detection

Results show a better behaviour of the correlation method. However, this method is very expensive under a computational point of view.

3.2 Peak Detection.

Peak detection has a double function. On one side it is useful to detect those samples considered as peaks, as they could indicate that there is a control cycle. On the other hand, it is necessary to have this information to take the measurements described in section 2.1.

The method used to detect peaks is a correlation method. The process is quite similar to the one exposed in the correlation coefficient method, but now templates are built with the number of samples per cycle, and the search is done only around ± 5 samples. The pseudo-algorithm is as follows:

1. Detect first peak.
2. Average peak ← first peak value
3. Search for next peak around T samples later.
   If (abs(peak value)> 1.4 average peak)
       Control cycle detected
       Remove control cycle
   Else
       Update average peak
       Store the peak value and index.
4. Until the end of process repeat step number 3.

Once the peak is found and control cycles have been removed, we have to proceed to take the measures described on section 2.1. Note that for all of them it is necessary to know where the peak is located and, for measures 1 and 2, it is very important to know the baseline reference. In figure 6 are represented 3 cycles of a signal, that has been determined as useful, and the peaks (negative in this case as this is a reduction) detected. It can be seen that the baseline must be also estimated. The baseline associated to the kth peak is calculated as:

$$B_k = \frac{BS_{k-1} + BS_k}{2} \quad (6)$$

Being BS_k the mean of the last samples previous to the following peak. For the first estimation we have considered it as BS_1.
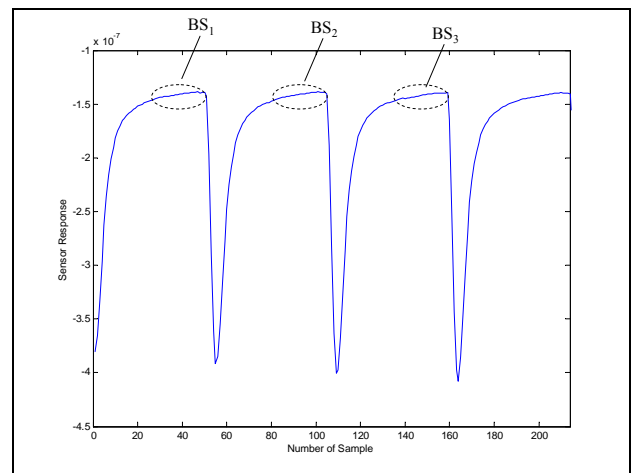

Fig 6. Peaks detected on a real signal and base line estimation.

Measures 1 and 2 are then easily extracted for every cycle. Measure 3 is related to the shape of the recovery signal. As we are going to compare in this case shapes and not values a normalization of the signal is needed, as described in [7]. In figure 7 is shown several shapes taken from the sensor 1 for white wines.
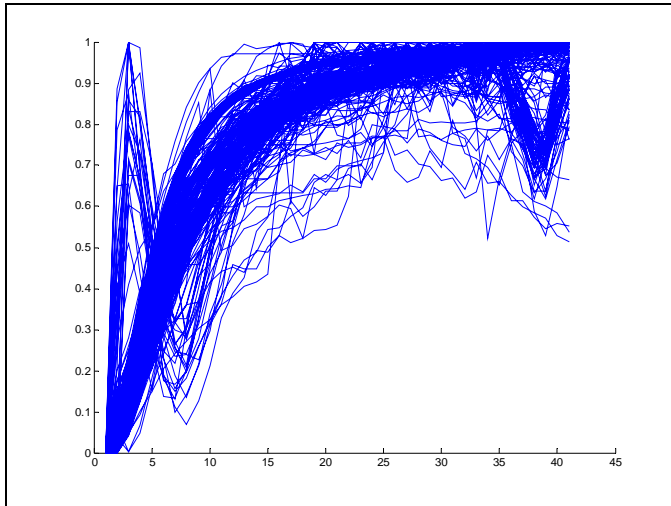

Fig 7. Shape recovery for white wine.

3.3 Pattern Recognition

In this stage and with the measures taken, the system proceeds to recognize the information in order to classify it into the different kinds of wine.

3.3.1 Candidate per Cycle

Every single cycle proposes a candidate based on the information that has been obtained on the processing stage. This candidate is proposed according to the scheme presented in figure 8, where I_1 is a six feature vector composed by the 6 values, one per sensor, obtained as measure number 1. If a signal sensor was discarded due to sensor outage, then we have to be aware of it, since the feature associated to that sensor is null. In such a case, the choice taken is to assign a mean value in order to minimize the influence of that feature. I_2 is also a vector composed by 6 features, but in this case instead of the measure 1 is composed by the 6 values obtained from measure 2. I_3 to I_8 are composed by the recovery signals of every sensor. Each vector is composed by 40 samples taken just after the peak detection. In these cases, if there is a sensor outage, the neural network output is not considered for the voting.

Tables 2 to 9 show the confusion matrixes obtained for every neural network. Different elements of the tables are $P(D_i/H_j)$, that is, the probability of getting an output D_i under the hypothesis H_j. A ideal matrix confusion is the identity one.
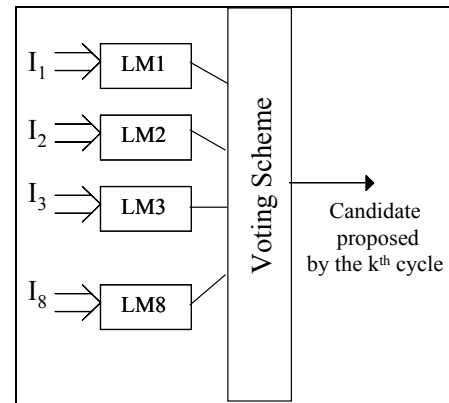

Fig 8. Scheme proposed per cycle.

I_1	White	Rose	Red
White	0,889	0,000	0,111
Rose	0,371	0,577	0,052
Red	0,000	0,214	0,786

Table 2. Confusion matrix for the measure 1.

I_2	White	Rose	Red
White	0,789	0,053	0,158
Rose	0,253	0,672	0,075
Red	0,111	0,365	0,524

Table 3. Confusion matrix for the measure 2.

I_3	White	Rose	Red
White	0,545	0,131	0,323
Rose	0,021	0,979	0,000
Red	0,276	0,133	0,592

Table 4. Confusion matrix for the sensor shape recovery in sensor 1.

I_4	White	Rose	Red
White	0,687	0,212	0,101
Rose	0,000	1,000	0,000
Red	0,162	0,027	0,811

Table 5. Confusion matrix for the sensor shape recovery in sensor 2.

I_5	White	Rose	Red
White	0,755	0,122	0,122
Rose	0,080	0,800	0,120
Red	0,900	0,040	0,060

Table 6. Confusion matrix for the sensor shape recovery in sensor 3.

I_6	White	Rose	Red
White	0,990	0,010	0,000
Rose	0,170	0,766	0,064
Red	0,261	0,065	0,674

Table 7. Confusion matrix for the sensor shape recovery in sensor 4.

I_7	White	Rose	Red
White	0,778	0,030	0,192
Rose	0,191	0,596	0,213
Red	0,408	0,061	0,531

Table 8. Confusion matrix for the sensor shape recovery in sensor 5.

I_8	White	Rose	Red
White	0,778	0,030	0,192
Rose	0,191	0,596	0,213
Red	0,408	0,061	0,531

Table 9. Confusion matrix for the sensor shape recovery in sensor 6.

3.3.2 Global Classification.

Once that every single cycle has proposed a candidate it can be determined what kind of wine is under the experiment. The output decision is also made by a voting scheme where every circle represents a candidate per cycle obtained as has been described in the previous section.

The output given will have a confidence depending on the number of votes that has obtained the winner. Low confidences mean that, although there was a winner, the candidates are scattered. In figure 9 is represented the confidence of the true output. Confidences bellow 33 % in this graphic mean that the system is wrong, since other candidate have been proposed. Confidences above the 85 % are considered as good results because most of the cycles have voted for the right candidate.
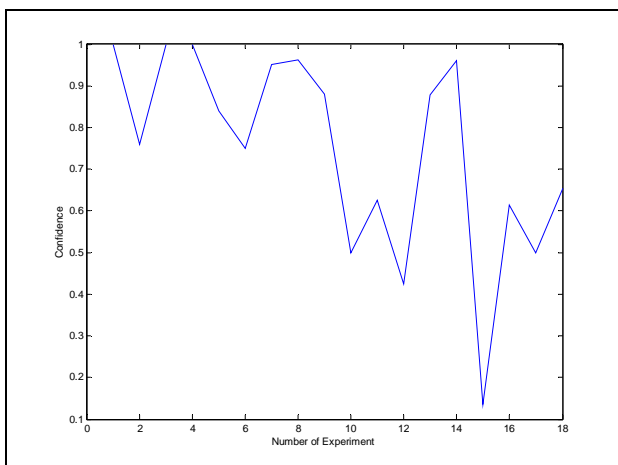

Fig 9. Confidences of the test experiments.

4   Conclusion

In this work we have presented several advances in the signal processing field for an electronic tongue, from the sensor outage detection to the measurement automatization. Not only peak responses have been considered but also measures related to the kinetic.

This work is completed with the classification of the wines using a MLP NN. Results show us a good global performance but in order to develop a commercial system they must be improved to ensure the reliability of the system.

Further work will try to obtain better results by improving the detection of sensor outage, and specially testing other learning systems and voting schemes. It is needed also to test the behavior of the system when testing with other wines with different taste.

There is also a good field to exploit in recognizing, not only the kind of wine, but the origin of the wine that would be useful to determine if a wine has the properties expected to a selected region. Other industrial applications such as residual waters analysis, olive oil quality or marketing improvements for drinks open a huge field to research.

References:
[1] J. W. Gardner, P.N. Barlett, *Electronic noses: Principles and applications*, Oxford University Press, 1999.
[2] R. Martinez et al, An "electronic tongue" design for the qualitative analysis of natural waters, *Sensors and Actuators B,* 104, 2005, pp. 302-307.
[3] S. Holmin et al, Compression of electronic tongue data based on voltammetry-a comparative study, *Sensors and Actuators B,* 76, 2001, pp. 455-464.
[4] Richard Duda, Peter Hart, David Stork, *Pattern Classification 2nd Edition*. Wiley Interscience. 2001.
[5] M. Hedenmo et al, Improved mediated tyrosinase amperometric enzyme electrodes, *Journal Electroanalysis Chemical*, 425, 1997 pp. 1-11.
[6] J. Parellada et al, Amperometric immunosensors and enzyme electrodes for environmental applications, *Journal Analysis Chemical Acta*, 362, 1998, pp. 47-57.
[7] E.L. Hines, E. Llobet J.W. Gardner, Electronic noses: a review of signal processing techniques, *IEE Proc.-Circuit Devices Syst.,* 146, 1999, pp. 297-310.