Estimation of Sparse Memory Taps for RF Power Amplifier Behavioral Models

John Dooley, Bill O'Brien, *Member, IEEE*, Keith Finnerty, and Ronan Farrell, *Member, IEEE*

Abstract—When a larger than required dimension such as memory depth or order of nonlinearity, is specified during behavioral model extraction, redundant terms can be calculated when determining the weights of the model. Extraction of a behavioral model can therefore benefit from a priori knowledge of the system to be modeled. Conversely if there is a limitation in the hardware required to calculate model outputs a limit can be set for the maximum number of weights to be used. In this letter, an approach is proposed which allows the input delay vector to be reduced to a sparse vector including the delayed samples which are most important in the construction of the power amplifier model. Simulations of behavioral models for experimentally measured data of two different PAs demonstrates the sparse models extracted in this way are as accurate as a full model but have a more compact and as a result more computationally efficient structure.

Index Terms—Active antenna arrays, behavioral modeling, memory effect, power amplifier (PA), Volterra series.

I. INTRODUCTION

POWER amplifiers (PAs) are indispensable components in wireless communication systems and are inherently nonlinear. In situations where there are multiple distributed PAs in a single system, top-down design approaches can be used to allow designers to perform faster analyses of these highly functional products [1]. Extraction of compact PA behavioral models is extremely important in the case of simulation of active antenna arrays where an array of distributed nonlinear PAs must be modeled simultaneously. In this scenario any saving on one signal path is immediately scaled up by a factor of 16, 32, or 64 depending on the number of PAs used in the system.

It is not immediately apparent from the output signal spectrum of an RF PA what the required model memory depth should be. Not knowing the memory depth of a PA to be modeled or linearized means the chosen model or pre-distorter memory depth is only an estimate arrived at by conjecture. If the memory depth is chosen too large the PA will be adequately modeled but at the cost of efficiency. If the memory depth is set too small, the model may not be a good approximation of the PA.

In some cases a model is extracted initially using a large number of terms. Once the coefficients have been calculated, those which have the least overall effect on the output are identified and removed until a threshold level of accuracy is achieved [2]. Ensuring a compromise of efficiency and accuracy of the model can be obtained by employing a pruning or growing algorithm in this way. However, these techniques have been introduced only for certain models, and the computational overhead can increase by a large amount if the techniques are applied repetitively during model extraction. It is therefore desirable to determine the optimum size for a behavioral model before the model is extracted.

From [3], a broad categorization of memory depth calculation approaches can be summarized as physical mechanisms, empirical and systematic methods. The latter being the preferred solution as it is more reliable and less complex. Analysis of the physical mechanisms, such as non-constant frequency response of biasing and matching networks is based on solid theory, however it is complex to implement. Empirical approaches which have been proposed previously require multiple model training and validation runs in order to identify the optimum memory depth. An approach used in both [3] and [4] involves the calculation of false nearest neighbors. This systematic approach involves progressively increasing the number of delayed input signal samples of the model space until some metric is satisfied that indicates that the memory depth of the system is covered. Since most of the data points are not the next point in the time series, they are *false neighbors* of the current time point.

Another systematic approach proposed in [5] presents Lipschitz quotients to calculate the total model memory depth. This approach was applied to the case of PA memory depth calculation directly from the input and output signal data for a PA in [5]. In this letter this approach is extended to produce a sparse delay vector including only those delays which contribute significantly to the output signal even if they are sparsely distributed. A sparse Volterra series model is used to validate the proposed approach.

II. SPARSE MEMORY VECTOR IDENTIFICATION

System memory depth calculation is possible, but special care must be taken that the sampling rate of the time domain signal is first set so that the signal is effectively uncorrelated from the other integer sample delayed versions of the signal [4]. The objective is to choose a value for the sampling interval that makes each delayed version of the signal orthogonal to each other, which will allow the extraction of the most compact model. The

choice of a suitable signal sample interval in this way will avoid a model with stability or convergence problems. Next a behavioral model structure must be chosen in order to validate the extraction of the sparse time delay vector. The model chosen in this letter is the classical form of the discrete time Volterra series model of a nonlinear dynamic system. It is described in the discrete time domain as shown in (1). $\tilde{x}(n)$ and $\tilde{y}(n)$ are the input and output complex envelope signals, NL and M represents the nonlinear order and memory depth of the model and $\tilde{h}_p(i_1, \ldots, i_p)$ are discrete time Volterra kernels

$$\tilde{y}(n) = \sum_{\substack{p=1 \\ p=p+2}}^{NL} \sum_{i_1=0}^{M} \cdots \sum_{i_{p+1}/2=i_{p-1}/2}^{M} \sum_{i_{p+3}/2=0}^{M} \cdots \sum_{i_p=i_{p-1}}^{M} \tilde{h}_p(i_1, \ldots, i_p)$$

$$\prod_{j=1}^{(p+1)/2} \tilde{x}(n-i_j) \prod_{j=(p+3)/2}^{p} \tilde{x}^*(n-i_j). \qquad (1)$$

Volterra kernels can be readily calculated from time domain input and output signals. In this letter our objective is not to calculate the optimum delay length but the optimum sparse delay vector to be used on the input signal. The ultimate aim being, to minimize the number of weights required by a model for an accurate description of a PA.

In the case of an RF PA we can observe the PA output for a given input signal. An observation can be made of N input-output complex envelope sample pairs [6], [7]. Assuming the RF PA may be characterized using the Volterra series in (1). The functionality of sparse memory vector identification using Lipschitz function can be described as follows:

a) Acquire a set of N input-output data pairs for the system and choose the maximum number of delays to consider.
b) Set a threshold value either explicitly or as a percentage of the total of all Lipschitz numbers
c) Calculate the Lipschitz numbers for sample points up to a set maximum delay
   i) Compute Lipschitz quotients for input-output sample pairs

$$q_{ij} = \frac{|y(i) - y(j)|}{|x(i) - x(j)|}, (i \neq j) \qquad (2)$$

   where q_{ij} is the Lipschitz quotient.
   ii) Determine the p largest quotients
   iii) Calculate the corresponding Lipschitz numbers.

$$q^{(n)} = \left(\prod_{k=1}^{p} \sqrt{n} q^{(n)}(k) \right)^{1/p}. \qquad (3)$$

d) Calculate the first differences between the Lipschitz numbers.
e) Determine what first differences are greater than the threshold value.

Following steps (a), (b) and (c) as in [5] we define the Lipschitz quotient q_{ij} given by (2) where $|x(i) - x(j)|$ is the distance
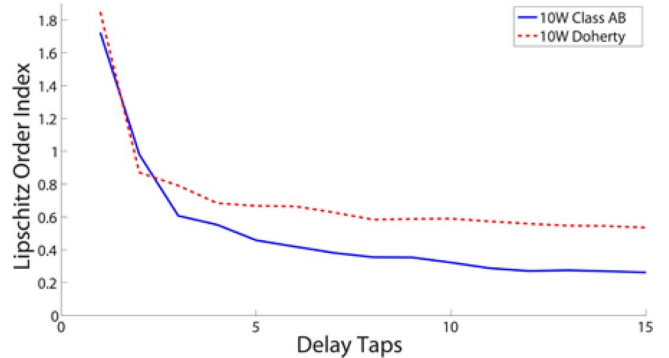

Fig. 1. Plot of Lipschitz order index versus sample delays.

between two input signal samples $x(i)$ and $x(j)$ in the input signal and $|y(i) - y(j)|$ the distance between two output signal samples, $y(i)$ and $y(j)$. The Lipschitz number is given by [5], $q^{(n)}(k)$ is the k-th largest Lipschitz quotient among all $q_{ij}^{(n)} (i \neq j; i, j = 1, 2, \ldots, N)$ and with n input signal sample values. Parameter p is a positive integer whose value is typically a small fraction of the number of input signal samples N.

The maximum number of delay taps required by the model can be visualized using the information of Lipschitz numbers computed directly from input and output data samples in the time domain as explained by steps (a) to (c). In Fig. 1 the Lipschitz number $q^{(n)}$ is plotted against the number of input variables. From [5]–[7] it is advised to truncate the number of input delayed taps at the knee point of this plot. With this representation it is not always obvious from visual inspection what point corresponds to the best point at which to truncate. In this work steps (d) and (e) document the extension of the method presented in [5]. These involve the subtraction of Lipschitz number magnitudes from one another to gauge the relative influence of one delay tap in relation to the previous delay tap. In doing this, the most influential input samples can be identified even if they are irregularly spaced.

To establish which delays to include or exclude in a sparse behavioral model, the first difference is taken of the Lipschitz numbers which shows proportionally how much each delay contributes to the output signal compared with its adjacent delays [8]. A threshold for inclusion of the sparse delays can then be set to determine which samples to use in the behavioral model. One methodology which can be employed is to set the threshold based on the hardware resources available to perform the computations of the model output. In this way the threshold is varied up or down until the maximum model size possible is reached. As indicated in Fig. 2.

III. VALIDATION

In order to validate the approach the complex envelopes of input and output signals for two different 10 W power amplifiers are measured for a quad carrier wideband code division multiple access (WCDMA) signal. Plotting the output signal in the frequency domain, an acceptable order of nonlinearity for the models to be extracted is estimated as third order. Equation (3) is used on the input and output time domain signal samples to calculate the dependency of the output on the delayed input
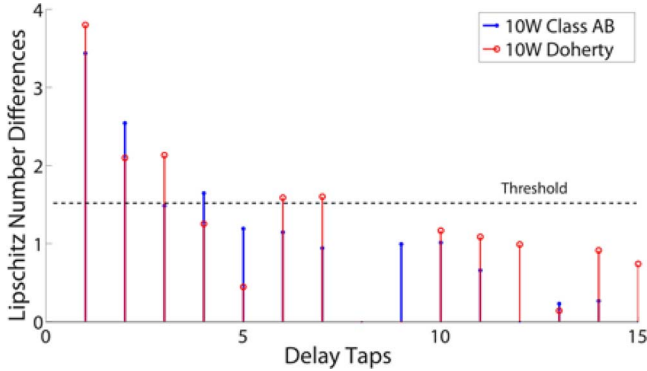
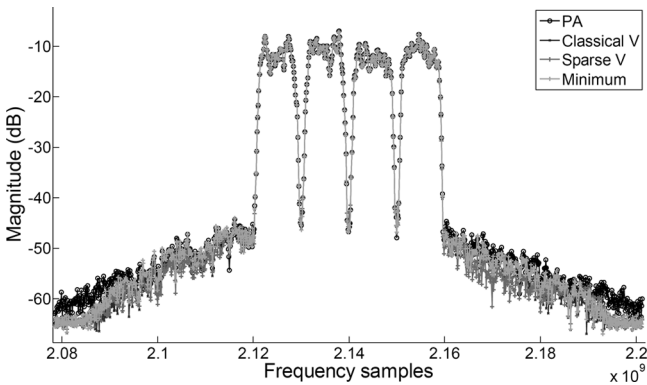Fig. 2. Plot of Sparse Lipschitz order index versus sample delays.

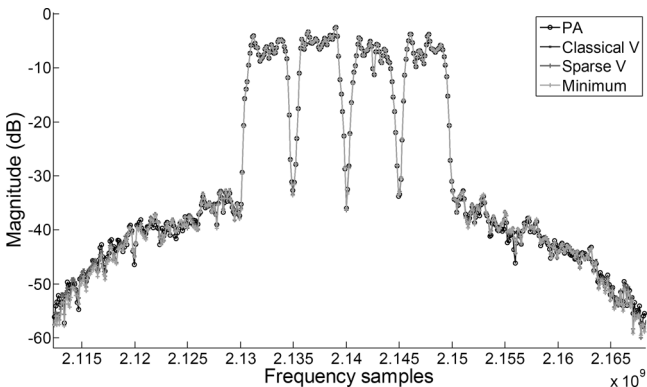Fig. 3. Plot of model outputs for Class AB PA in frequency domain.

Fig. 4. Frequency spectra of model outputs for Doherty PA.

TABLE I
COMPARISON OF NONLINEAR MODELS

PA	First Minimum	Classical	Sparse
Doherty NMSE(dB)	NL=3,M=3 -27.21	NL=3,M=7 -33.47	NL=3,M={0,1,2,3,6,7} -33.52
	NL=5,M=3 -27.79	NL=5,M=7 -33.88	NL=5,M={0,1,2,3,6,7} -35.94
Class AB NMSE(dB)	NL=3,M=2 -33.3	NL=3,M=4 -34.22	NL=3,M={0,1,2,4} -34.56
	NL=5,M=2 -32.53	NL=5,M=4 -38.46	NL=5,M={0,1,2,4} -38.86

signal samples. The corresponding dependency of each sample delay is plotted in Fig. 1. Beyond the point at which the values saturate the use of delay taps should not increase the accuracy

of the model by a large amount and therefore an upper limit to the number of sample delays can be set. Next the first difference between successive sample delays is calculated and a threshold set to identify the sparse sample delays. The sparse delays being those that hold a value above that of the threshold.

Fig. 2 shows a plot of the first differences for the Lipschitz numbers using a log scale. The threshold level used is indicated on the plot and can be set according to the total number of taps allowed for the sparse model. In the case of the two PAs used in this work sparse delay tap vectors of {0,1,2,3,6,7} and {0,1,2,4} were identified and used to extract behavioral models for the Doherty and Class AB PAs respectively. Where 0 indicates no delay, 1 indicates a delay of one sample, etc.

In order to compare the results for the sparse Lipschitz method used to identify the sparse delay vector, two other models are extracted. First an additional behavioral model was extracted using all the delay taps up to and including the total memory depth and this is denoted as the Classical Volterra model. Another model denoted as First minimum, uses the number of delay taps up to the first unnecessary tap equivalent to what would constitute the knee point of the Lipschitz plots in Fig. 1. In this way it is possible to show that further information is contained in the additional sparse taps.

It can be seen from the normalized mean square error (NMSE) performance metrics in Table I and Figs. 3 and 4, that the sparse vector model provides comparable or increased accuracy compared to the Classical model.

IV. CONCLUSION

In this letter, a novel method is presented, to calculate the optimum sparse delay tap vector for an RF PA behavioral model. A sparse tap Volterra series based model is used in the verification, and its improvement in performance is evident from the tabulated results for two different PAs. It therefore provides a means of removing less important delay taps from a behavioral model for use where there may be limited computational resources.

REFERENCES

[1] V. J. Mathews and G. L. Sicuranza, *Polynomial Signal Processing.* New York: Wiley, 2000.
[2] L. Yao, W. A. Sethares, and Y.-H Hu, "Identification of a nonlinear system modeled by a sparse Volterra series," in *Proc. IEEE Int. Conf. Syst. Eng.*, 1992, pp. 624–627.
[3] B. Fehri and S. Boumaiza, "Application of embedding dimension estimation to Volterra series-based behavioral modeling and predistortion of wideband RF power amplifier," *Int. J. Microw. Wireless Technol.*, pp. 115–122, Apr. 2013.
[4] J. Wood, M. Lefevre, D. Runton, J.-C. Nanan, B. H. Noori, and P. H. Aaen, "Envelope-domain time series (ET) behavioral model of a Doherty RF power amplifier for system design," *IEEE Trans. Microw. Theory Tech.*, vol. 54, no. 8, pp. 3163–3172, Aug. 2006.
[5] B. O'Brien, J. Dooley, A. Zhu, and T. J. Brazil, "Estimation of memory length for RF power amplifier behavioral models," in *Proc. 36th Eur. Microw. Conf.*, 2006, pp. 680–692.
[6] X. He and H. Asada, "A new method for identifying orders of input-output models for nonlinear dynamic systems," in *Proc. Amer. Control Conf. (ACC'93)*, 1993, pp. 2520–2523.
[7] M. Norgaard, L. K. Hansen, O. Ravn, and N. K. Poulsen, *Neural networks for modelling and control of dynamic systems.* New York: Springer, 2000.
[8] B. O'Brien, "Behavioural modelling of RF power amplifiers using neural networks," Ph.D. dissertation, University College Dublin, Dublin, Ireland, 2010.