On Analog Signature Analysis

Franc Novak
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
franc.novak@ijs.si

Bojan Hvala, Sandi Klavžar
Department of Mathematics, PeF
University of Maribor, PF
Koroška 160, 2000 Maribor, Slovenia

Abstract

We formalize the problem of analog data compression and analyze the existence of a polynomial data compression function. Under relaxed conditions we explore the existence of a solution employing digital signature analysis in the analog domain.

1 Introduction

Comparison of responses of a circuit-under-test to some previously determined reference values, obtained under the same stimuli, is a common approach in automatic testing techniques in both digital and analog domain. A compact measurement data characterizing the behavior of a circuit is often referred to as a *signature*.

Signature analysis proved to be an effective fault detection and localization technique for digital circuits, [2]. Conventional signature analysis uses a pseudorandom binary sequence (PRBS) generator provided with an external input. Feeding data into a PRBS generator has the same effect as dividing the data by the characteristic polynomial of the PRBS generator. When the process is stopped, the remaining contents of the PRBS generator represent the signature of the input data stream. Successful application of the above technique in practice is due to the fact that even for PRBS generators of modest length the probability of two different input data streams of equal length having the same signature is close to zero.

Simulation before test techniques in analog circuit diagnosis also deal with "signatures", but they are rather used for characterization of the effects of a selected set of faults than for the description of correct circuit operation. There are no standard or widely accepted rules for generating analog signatures. Since analog signatures are stored and processed by a computer they are discrete. Although limited data storage imposed significant restriction to the earlier implementations of fault dictionaries the reported approaches do not employ data compression techniques to reduce the need of memory resources. Rather they optimize the number of stored values by selecting only those corresponding to the selected significant measurements [1].

As technology advances, memory restrictions are getting less stringent, but the problem of managing large amount of measurement data in practice still remains. An open question is if it is possible to define also in the analog domain some common way of compressing measurement data in a unique signature. In that case, exhausive tables of measurement results which are used for plotting a response of a circuit as a function of input stimuli could be compressed into a single signature. Such signature could be a suitable basis for automatic testing and diagnosing of analog circuits. Furthermore, it would become possible to characterize the behavior of a mixed signal circuit by a *mixed signal signature* composed of a sequence of digital and analog signatures corresponding to given parts of the circuit during a complex functional test (e.g. built-in mixed signal circuit test, or digital/analog core test in MCM applications).

The approaches to the analog signature analysis proposed so far [3], [5], [6], diverge in methodology and application domain. In practice they have not converged to a widely accepted solution as it is the case with digital signature analysis technique. One of possible reasons may be the fact that an explicit definition of the problem has not been stated.

In this paper we formalize the problem of analog data compression and analyze the existence of a polynomial data compression function. Under relaxed conditions we explore the existence of a solution employing the conventional digital signature analysis in the analog domain. Analytical expressions are derived which may serve for further analysis and assessment of aliasing of this approach.

2 Data compression of analog measurements

2.1 Description of the problem

We analyze the possibility of data compression of a series of n real numbers characterizing the response of a circuit for the defined stimuli. In particular, we are looking for some data compression function f that would enable us to determine for any two given responses $Y = y_1, y_2, \dots, y_n$ and $Z = z_1, z_2, \dots, z_n$ whether $z_i \leq y_i$ holds for all i merely on the basis of their signatures $f(Y)$ and $f(Z)$, as depicted in Figure 2. If such data compression function existed, one could describe regions that characterize the response of a circuit simply by the signatures of their margins. Furthermore, from the signature of a response of a circuit-under-test it would be possible to determine if the response lies in the given region or not.

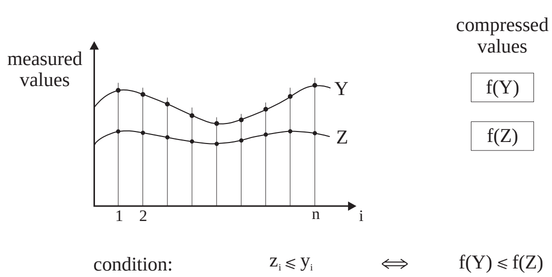

Figure 1. Data compression of analog measurements

2.2 Problem formalization

Suppose that we want to characterize the behavior of a circuit for a given type of fault by the values $y_1, y_2, \dots, y_n$, denoted as *reference values*, measured or obtained by simulation at some chosen test point. We will briefly denote this values by Y. Clearly, $Y \in \mathbb{R}^n$. Y will always mean an arbitrary but fixed vector. Furthermore, assume that the fault is characterized by the reference values Y such that any set of measured values $z_1, z_2, \dots, z_n$, Z for short, for which $z_i \leq y_i$ holds for all i, corresponds to the same fault. This will be denoted $Z \leq Y$. Since n is in general large, we would like to compress $Y \in \mathbb{R}^n$ to some $Y' \in \mathbb{R}^k$, where $k << n$, by standard arithmetic operations - addition and multiplication. In fact, these are just the operations (essentially) available in computers. This compression means that we compute Y' from Y using standard arithmetic operations, i.e. we are looking for a polynomial function (of n variables) $f : \mathbb{R}^n \to \mathbb{R}^k$. By the above we want such a polynomial that for any $Z \in \mathbb{R}^n$:

$$Z \leq Y \text{ if and only if } f(Z) \leq f(Y). \qquad (1)$$

Let us call such a function Y-*compatible*.

2.3 Existence of solution

It is not difficult to find an Y-compatible function. Consider, for example, the following function $g : \mathbb{R}^n \to \mathbb{R}$:

$$g(X) = \begin{cases} (-1)^{n+1}(x_1 - y_1) \cdots (x_n - y_n); X \leq Y, \\ \max\{x_1 - y_1, \dots, x_n - y_n\}; \text{otherwise.} \end{cases}$$

It is easy to see that g is continuous and Y-compatible. However, the definition of g involves Y, which is clearly in contradiction with our intentions and furthermore, g is not a polynomial function. But for such functions we have:

Theorem 2.1 *For any $Y \in \mathbb{R}^n$, $n \geq 2$, there is no Y-compatible polynomial function $f : \mathbb{R}^n \to \mathbb{R}^k$, $1 \leq k < n$.*

Proof. We will give only a sketch of the proof, which can be found in [4]. For $m < n$ let $Y_m = (y_m, y_{m+1}, \dots, y_n)$ and let $S = \{Z \in \mathbb{R}^{n-1};\ Z < Y_2\}$. Suppose on the contrary that $f : \mathbb{R}^n \to \mathbb{R}^k$ is Y-compatible, $f = (f_1, f_2, \dots, f_k)$. Then it is not difficult to show that at least one of the following holds:

(i) there exist $j \in \{1, 2, \dots, k\}$, $Z \in S$, and a neighbourhood $U \subset S$ of Z in $\mathbb{R}^{n-1}$ such that

$$f_j(y_1, X) - f_j(Y) = 0, \quad \text{for every } X \in U.$$

(ii) there exists $X \in \mathbb{R}^{n-1}$, $X < Y_2$, such that for all $i = 1, 2, \dots, k$,

$$f_i(y_1, X) - f_i(Y) < 0.$$

If (ii) would hold then $f(y_1+\delta,X) \leq f(Y)$ for $\delta > 0$ small enough, therefore f is not Y-compatible. Therefore (i) holds. But in this case one can deduce that Y-compatibility of $f : \mathbb{R}^n \rightarrow \mathbb{R}^k$ implies an existence of $Y_2 = (y_2, y_3, \dots, y_n)$-compatible function $g : \mathbb{R}^{n-1} \rightarrow \mathbb{R}^{k-1}$. We continue this procedure and finally end up with a $Y_k = (y_k, y_{k+1}, \dots, y_n)$-compatible real mapping h. Again, at least one of (i) and (ii) holds for h. If (ii) holds then as above h is not Y_k-compatible. And if $h(y_k, X) - h(Y_k) = 0$ holds for $X \in \mathbb{R}^{n-k}$, it follows that we can easily choose X, which violates the compatibility. Q.E.D.

Although a negative result, presented proof of non-existence of Y- compatible polynomial function for data compression actually states the limits one should be aware of when searching for a satisfactory practical solution.

By analogy with digital signature analysis, a solution may become feasible if we relax condition (1). For example, there may exist an analog data compression function which does not 100% satisfy condition (1) but gives still acceptable results in practice.

3 Possible directions toward practical solutions

One possible approach, employed in [7], is to divide the amplitude range of the response into 2^m quantisation bands, denoted by the corresponding m- bit binary numbers. The response of a circuit-under-test can be represented by a bit stream composed of a sequence of binary values of the measured response. For example, response *a* in Figure 2 can be represented by 1010 1010 1001 0101 0001 0001. The resulting stream can be compressed into a signature by a PRBS generator. In this way one could make use of the advantages of digital signature analysis in the analog domain.

Yet this approach faces some problems. When samples are taken close to the edge of the quantisation band outliers, different input streams may result for two (nearly) equal responses. The second and more serious problem concerns aliasing error (i.e., the probability of some faulty response having the same signature as the reference fault-free characteristic). For a w-bit input stream and a z-bit PRBS generator, $2^{w-z}-1$ different responses will result in an equal signature. Since the response of an analog circuit may in principle result in an arbitrary bit stream it is imperative to keep the aliasing error as low as possible. This can be done either by increasing z or by applying some additional selectivity criterion for further selection among the candidate bit streams. For example, some quantitative description of the shape of the reference response curve can be used for this purpose. Since reduced aliasing error is achieved at the expense of decreased efficiency of data compression, the final solution will be a compromise in each specific case.

In the following we explore the possibility of reducing the aliasing error by keeping track of the number of increasing/decreasing samples in the input sequence. We describe the idea on some illustrative examples and derive the expressions which may serve as the basis for further study in this direction.

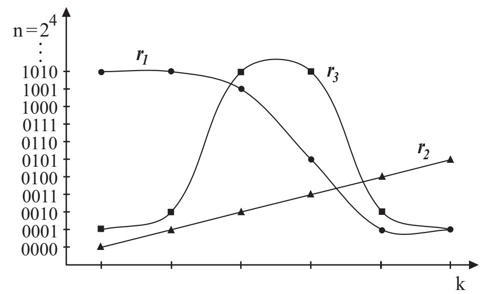

Figure 2. Examples of sampled responses

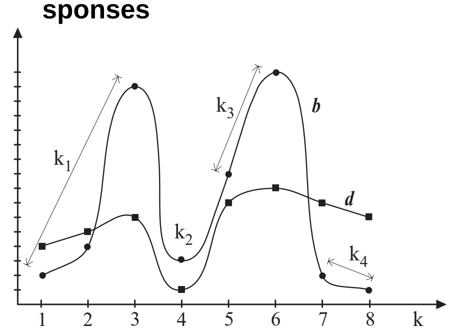

Figure 3. Two compatible sequences

3.1 Counting compatible sequences

Let $\mathbb{N}_n = \{1, 2, \dots, n\}$ be the set corresponding to the range of a sampled response (n=2^m quantisation bands) and let k be the number of samples of a response.

Let $b = (b_1, b_2, \ldots, b_k)$ and $d = (d_1, d_2, \ldots, d_k)$ be sequences of a series of samples. For our purposes we will without loss of generality assume that for $i = 1, 2, \ldots, k-1$ we have $b_i \neq b_{i+1}$ as well as $d_i \neq d_{i+1}$. Indeed, if $b_i = b_{i+1}$ would hold for some i, then the number of samples which have the same increasing/decreasing pattern as b can be obtained by considering the sequence $(b_1, \ldots, b_{i-1}, b_{i+1}, \ldots b_n)$. Let

$$\begin{aligned} b = \quad & (b_1, \ldots, b_{k_1}), (b_{k_1+1}, \ldots, b_{k_1+k_2}), \ldots, \\ & (b_{k_1+\cdots+k_{t-1}+1}, \ldots, b_{k_1+\cdots+k_t}) \end{aligned}$$

and

$$\begin{aligned} d = \quad & (d_1, \ldots, d_{s_1}), (d_{s_1+1}, \ldots, d_{s_1+s_2}), \ldots, \\ & (d_{s_1+\cdots+s_{r-1}+1}, \ldots, d_{s_1+\cdots+s_r}) \end{aligned}$$

be the partitions of the sequences b and d into maximal monotone subsequences which we obtain from left to right. Clearly, $k_1 + \ldots + k_t = k$ and $s_1 + \cdots + s_r = k$. We say that b is *compatible with* d, $b \sim d$, if $t = r$ and $k_i = s_i$ for $i = 1, 2, \ldots t$. In the sequel we may without loss of generality assume that $k_1 \geq 2$.

Roughly speaking, $b \sim d$ if b and d coincide in their increasing/decreasing intervals. For example, if $n = 16$, $k = 8$ and $b = (2, 4, 15, 3, 9, 16, 2, 1)$ then the corresponding partition of b is

$$b = (2, 4, 15), (3), (9, 16), (2, 1).$$

We have $t = 4$, $k_1 = 3$, $k_2 = 1$, $k_3 = 2$ and $k_4 = 2$, and, for instance, b is compatible with $d = (4, 5, 6, 1, 7, 8, 7, 6)$ depicted in Figure 3.

Note that (for fixed n and k) the relation $\sim$ defines an equivalence relation in the set of all sequences of length k over the alphabet $\mathbb{N}_n$. We are interested in the size of its equivalence classes. Thus, let $N(b)$ denote the number of elements of the equivalence class containing b.

Consider first the case $t = 1$, i.e., the sequence $b = (b_1, b_2, \cdots, b_k)$ is strictly increasing. In this case the corresponding increasing sequences bijectively correspond to the k-subsets of the set $\mathbb{N}_n$, hence

$$N(b) = \binom{n}{k}.$$

Note also, that there are

$$\binom{n-1}{d-1}$$

such sequences which end with the element d. For the general case let us write the partition

$$\begin{aligned} d = \quad & (d_1, \ldots, d_{k_1}), (d_{k_1+1}, \ldots, d_{k_1+k_2}), \ldots, \\ & (d_{k_1+\cdots+k_{t-1}+1}, \ldots, d_{k_1+\cdots+k_t}) \end{aligned}$$

of a sequence d which is compatible with b briefly as

$$\begin{aligned} d = \quad & (d_1, \ldots, \delta_1), (d_{k_1+1}, \ldots, \delta_2), \ldots, \\ & (d_{k_1+\cdots+k_{t-1}+1}, \ldots, \delta_t). \end{aligned}$$

Let $N_{t-1}(\delta_{t-1})$ be the number of sequences of length $p = k_1 + \cdots + k_{t-1}$ which are compatible with the first p terms of the sequence b and end with a fixed element δ_{t-1}. We now distinguish two cases.

Case 1: t is even.
In this case the sequence d must be completed after the element δ_{t-1} with a decreasing sequence of length k_t. There are $\binom{\delta_{t-1}-1}{k_t}$ such sequences and therefore we have:

$$N(b) = \sum_{\delta_{t-1}=k_{t-1}+1}^{n} \binom{\delta_{t-1}-1}{k_t} \cdot N_{t-1}(\delta_{t-1}).$$

Case 2: t is odd.
Now the sequence d must be completed after δ_{t-1} via an increasing sequence of length k_t. There are $\binom{n-\delta_{t-1}}{k_t}$ such sequences hence

$$N(b) = \sum_{\delta_{t-1}=1}^{n-k_{t-1}} \binom{n-\delta_{t-1}}{k_t} \cdot N_{t-1}(\delta_{t-1}).$$

The above recursion could be solved by induction, but it would be rather time and space consuming. Instead, we present the formula for the case $t = 4$ which essentially reflects the general case. Thus, if b is a sequence of length k over $\mathbb{N}_n$ and $t = 4$, then $N(b)$ is equal to

$$\sum_{\delta_3=k_3+1}^{n} \binom{\delta_3-1}{k_4} \cdot \left[\sum_{\delta_2=1}^{\substack{\min\{n-k_2,\\ \delta_3-1\}}} \binom{\delta_3-\delta_2-1}{k_3-1} \cdot \left(\sum_{\substack{\delta_1=\max\{k_1,\\ \delta_2+1\}}}^{n} \binom{\delta_1-\delta_2-1}{k_2-1} \binom{\delta_1-1}{k_1-1} \right) \right].$$

To the last formula we add that the functions max and min are present to assure that the upper values of the binomial coefficients are nonnegative.

We use the above formula to compute the number of compatible sequences N(b) for the examples in Figure 2. Table 1 gives the computed N(b) for the responses r_1, r_2 and r_3 for three different quantisation bands: 16, 32 and 64. Efficiency of the selectivity criterion is best characterized if we divide N(b) by the total number of possible responses of

	$N(b)$			$N(b)/N_{total}$		
	n=16	n=32	n=64	n=16	n=32	n=64
r_1	1820	35960	635376	$1.08\ 10^{-4}$	$3.35\ 10^{-5}$	$9.24\ 10^{-5}$
r_2	8008	906192	74974368	$4.77\ 10^{-4}$	$8.43\ 10^{-4}$	$1.09\ 10^{-2}$
r_3	37688	1428976	49600992	$2.24\ 10^{-3}$	$1.33\ 10^{-3}$	$7.22\ 10^{-4}$

Table 1: Computed N(b) for the responses r_1, r_2 and r_3

	equal signatures			aliasing error probability		
	z=4	z=6	z=8	z=4	z=6	z=8
r_1	111	34	9	$6.6\ 10^{-6}$	$2.0\ 10^{-6}$	$5.3\ 10^{-7}$
r_2	499	131	33	$2.9\ 10^{-5}$	$7.8\ 10^{-6}$	$1.9\ 10^{-6}$
r_3	2353	589	147	$1.4\ 10^{-4}$	$3.5\ 10^{-5}$	$8.7\ 10^{-6}$

Table 2: Generated signatures for sequences compatible with r_1, r_2 and r_3

consisting of k samples (in our case 6). The results are given in the right part of Table 1.

Table 2 gives the number of sequences (compatible with r_1, r_2 and r_3) generating 4, 6 and 8 bit signatures that are equal to the signatures of the responses r_1, r_2 and r_3 . Associated aliasing error probabilities are also given.

4 Conclusion

We have formalized the problem of analog data compression and presented proof of non-existence of polynomial data compression function. The proof actually states the limits one should be aware of when searching for a satisfactory practical solution. We have concentrated our discussion on polynomial functions because they can be easily implemented by microprocessor-based measurement instrumentation. Further work can be directed toward searching for some other function possibly satisfying condition (1), or toward defining some practical solution for a relaxed condition (1).

Under relaxed conditions we explored the existence of a solution employing digital signature analysis in the analog domain. We analyzed the possibility of reducing the aliasing error by keeping track of the number of increasing/decreasing samples in the input sequence. Some illustrative examples are given. Derived expressions may serve as the basis for further study in this direction. Assessment of efficiency of the proposed approach on more complex examples is subject of our current research. Further work can also be directed toward defining other selectivity criteria.

References

[1] J.W. Bandler, A.E. Salama, ”Fault diagnosis of analog circuits“, *Proc. IEEE*, Vol. 73, No. 8, August 1985, pp. 1279-1327.

[2] R.A. Frohwerk, ”Signature analysis: A new digital field service method“, *Hewlett Packard J.*, May 1977, pp. 2-8.

[3] D.R. Glancy, ”Analog Signature Analysis - Part 1“, Electronics Test, Vol.4, No.9, 1981, pp.82-101, ”Part 2“, Electronics Test Vol4, No.10, 1981, pp.38-50.

[4] B. Hvala, S. Klavžar, F. Novak, “On Y-compatible and strict Y-compatible functions“, *Appl. Math. Letters*, Vol.10, No.1, 1997, pp. 79-82.

[5] M.Lubaszewski, S.Mir, L.Pulz, ”A multifunctional test structure for analog BIST“, *Proc. IEEE Int. Mixes-Signal Testing Workshop*, pp. 239-244, 1996.

[6] N.Nagi, A.Chatterjee, H.Yoon, J.A.Abraham, ”Signature analysis for analog test response compaction“, *IEEE Trans. CAD of Int. Circuits and Systems*, Vol.17, No.6. pp.540-546, 1998.

[7] M.Renovell, F.Azais, Y.Bertrand, ”Analog signature analyzer for analog circuits: BIST implementations“, *Proc. IEEE Int. Mixes-Signal Testing Workshop*, pp. 233-238, 1996.

[8] G.E.Taylor, J.Webster, P.Skladel, P.Simek, ”Digital data compression technique for analogue blocks in mixed signal devices“, Proc. IEE Colloq. on Testing Mixed Signal Circuits and Systems, pp.5/1-5/6, 1997.