

Optimal Detector Design for On-line Testing of Linear Analog Systems

EMMANUEL SIMEU*

TIMA-Laboratory, 46, Av. Félix Viallet, F-38031 Grenoble, France

(Received 1 April 1999; In final form 5 October 1999)

The design of integrated fault detector for on-line testing of linear analog systems is discussed in this paper. The method consists in a concurrent processing of available the node voltage signals to provide a residual on-line, that carries information about the faults. Contrary to the few previous works dealing with the particular case of state variable analog systems, the method proposed here is useable without limitation for a larger class of linear analog systems, even when the state variables are not available as measurable voltages. For this purpose, an algorithm providing an extended state space model for any linear analog system from its netlist description is developed and implemented.

Keywords: Concurrent, testing, state space modeling, analog circuits, optimal residual

1. INTRODUCTION

A novel fault detection methodology based on the use of analytical rather than physical redundancy has emerged and is increasingly discussed. For this purpose, the inherent redundancies contained in the static and dynamic relationships among the system input and measured signals are exploited. In other words, a mathematical model of the system or a part of it is used. In this paper, the problem of concurrent model-based fault detection is discussed for linear analog systems. The basic principle of concurrent testing techniques consists

in exploiting the mathematical knowledge available on the monitored system to generate a fault indicating signal (called residual). The residual express the difference between the signals coming from the actual system and the information provided by the system nominal model in the normal operation. When the mathematical model is an accurate representation of the nominal system behavior, the residual characterizes the system operating mode: close to zero in normal operation, different from zero if there is a mismatch. The goal is to detect the faults of interest and their causes early enough so that a failing of the

*e-mail: Emmanuel.Simeu@imag.fr

overall system can be avoided. Residual is generated on-line by means of dedicated additional hardware into the system data flow graph.

The idea of designing additional hardware for error detection and fault tolerance was applied to FFT networks by Jou and Abraham in [1]. More recently in [2], Chatterjee used the checksum codes proposed in [1] to develop a concurrent error detection method usable for linear analog and switched-capacitor state variable systems. In contrast to the previous works using algorithm-based fault tolerance [4, 5] the error detection process is directly hard-wired into the system using additional detection circuitry. The method proposed in [2] is applicable only to state variable systems. Such systems consist of a serial interconnection of integrators and summers (realized with op-amps). The availability for physical connection of all the state variables facilitates redundancy generation and the design of concurrent error detection schemes. In the approach proposed in this paper, the use of extra circuitry with the objective of on-line fault detection is extended to a larger class of linear analog systems, with detection dedicated circuitry comparable to those used in the previous works, for only state variable systems. The scheme that is proposed here involves the elimination of unknown state variables by aggregation using linear combinations of input, output and available node signals. Test dedicated hardware is optimized by a reduction of the number of integration (or derivations) required for residual generation. To minimize the overhead for implementation, information contained on any circuit node available for measurement should be usable as input of the detector scheme. This requires a mathematical model of the circuit providing not only the classical behavioral relationships between the circuit input and its functional output, but also an appropriated modeling of the signal on any internal node.

The first part of this paper deals with a symbolic modelling method, deriving an appropriated state space model from the netlist description of any linear analog circuit. The algorithm presented in

Section 3 is implemented to transform the netlist description (Spice netlist for example) of linear analog circuit into an extended space state model usable for residual generation. A dead-beat observer method described in Section 4 uses the extended state model to generate the equations of optimal fault detector circuitry, for a large class of linear analog circuit.

2. CONCURRENT TESTING PROBLEM STATEMENT AND MODELING REQUIREMENTS

The first step of any model based fault detection method is the determination of an appropriate mathematical behavioural model of the system under test. The mathematical knowledge available on the model is then exploited to generate a residual that will be used as on-line fault indicating signal.

Contrary to the classical (off-line) testing strategies, the normal operation of the system under test is not suspended during on-line testing. That involves two main constraints governing concurrent testing:

- (i) The system nominal input/output behaviour should be held without disturbance:

The concurrent signal measurement and processing required for on-line residual generation should be made without any perturbation of the normal operating mode.

- (ii) Fault detection should be guaranteed for any input signal:

In off-line testing procedure, the test input signals are specially chosen and optimized for specific fault coverage. Contrary, for concurrent testing, the normal operating input signals are the only excitation exercising on the system. So, the input signal used in the detection process depends on the random operating mode. The operating input signal sequence can not be chosen for a particular testing objective. On-line fault detection will be

guaranteed for any system particular operating condition only if the detection process efficiency is independent to the input signal form.

Generally, the residual is obtained by a concurrent processing of available signals provided by sensors and actuators [3, 6].

Electronic circuits are driven by voltage and current sources. Hence, there are no actuators to the circuit other than these sources. Also, there are no additional sensors necessary to measure voltages.

The few publications that focus on on-line error detection for electronic circuits deal with fully available systems [2]. For this type of circuit, all state variables are voltages and can be easily tapped from the circuit under test. When the system state variables are not directly connectable, on-line fault detection can be performed using the classical input/output behavioral model that is generally used for functional output simulation. Unfortunately, the use of such a simulation model for concurrent testing of analog circuit would be too costly in terms of additional circuitry. Measuring currents is problematic because the corresponding signals have to be read with minimal disturbance to the primary function of the circuit under test. It is assumed in this paper that only circuit node voltages are available for measurement. The corresponding node voltages are assumed to be read through voltage followers without disturbance of initial functionality of the circuit under test. Figure 1 shows that the

residual is built from the functional outputs and the internal node voltages of the circuit.

Contrary to the models generally used for functional output simulation, the extended model required for fault detection must also give:

- an estimation of the signal on any internal node;
- a model for unknown inputs or noise effects;
- a model for fault effects.

In the extended model, the circuit unknown inputs modeling the noise sources are clearly differentiated to the faulty behavior. The required extended model for the circuit may have the following state equation form:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{E}\mathbf{d}(t) + \mathbf{K}\mathbf{f}; \quad (1)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) + \mathbf{F}\mathbf{d}(t) + \mathbf{G}\mathbf{f}. \quad (2)$$

In these equations, $\mathbf{x}(t)$ is the state vector of size \mathbf{n} , $\mathbf{u}(t)$ is the input vector, $\mathbf{y}(t)$ is the output vector of size \mathbf{p} . Each output component $y_i(t)$ represents the voltage signal on a circuit node. $\mathbf{d}(t)$ is a vector modeling the effects of unknown inputs and disturbances when vector \mathbf{f} models the fault effects. \mathbf{A} , \mathbf{B} , \mathbf{C} , \mathbf{D} , \mathbf{E} , \mathbf{F} , \mathbf{G} and \mathbf{K} are matrices with time invariant components and appropriate dimensions. A similar model has been used in the topic of automatic control for fault detection and identification [6, 7].

The first problem to be solved is: how to derive such extended state space model from the classical analog circuit description tools.

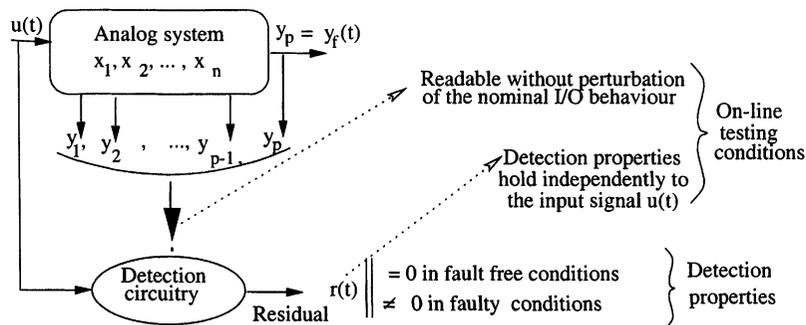


FIGURE 1 On-line testing scheme.

3. FROM NETLIST DESCRIPTION TO EXTENDED STATE SPACE MODEL

The choice of a model to represent a dynamical system depends on the situation and the type of analysis required. Contrary to the classical modeling techniques available for analog circuit representation (Circuit schematics, Block diagrams, Bond graphs, ...), the netlist description is not really usual for analog systems. It is generally used as an intermediary stage between the previous representation techniques and the mathematical behavioral models, usable for system simulation. Figure 2 shows the schematic and the corresponding netlist for a fourth order low pass filter.

3.1. Variable/Equation Dependency Matrix

The transformation of a netlist description of electrical circuits into admittance matrix form is generally simple. A circuit can be simulated with an admittance matrix \mathbf{P} and an excitation vector \mathbf{Q} . To obtain the voltages for the nodes of the circuit, it suffices to solve

$$\mathbf{P} \cdot \mathbf{v} = \mathbf{Q}. \quad (3)$$

This approach works very well for numerical simulations. There are efficient algorithms to factor \mathbf{P} numerically. However, the deduction of a symbolic transfer function or state space model using symbolic variables in \mathbf{P} and \mathbf{Q} is far from trivial: it requires the symbolic inversion of \mathbf{P} .

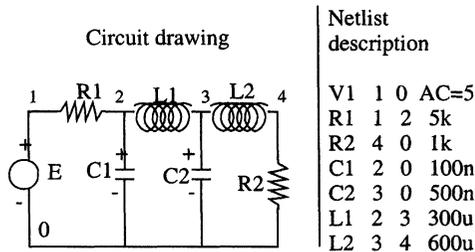


FIGURE 2 Circuit schematic and netlist for a fourth order low pass filter.

This section explains how the matrixes \mathbf{A} , \mathbf{B} , \mathbf{C} , \mathbf{D} , \mathbf{E} , \mathbf{F} , \mathbf{G} and \mathbf{K} of the extended model Eqs. (1) and (2) are generated from a circuit description SPICE netlist.

Any linear element in a SPICE circuit description defines a topologic equation. The connections of different elements in a circuit give possible additions of Kirchoff node equations.

For simplicity and without loss of generality, the noise caused by the effects of resistors heating is the only unknown inputs taken into account in the system modeling. Their influence can be modeled as a small voltage source E_R in series with the resistor. This voltage source is an unknown input because it cannot be measured or controlled. This modeling can easily be extended to any other type of noise disturbing the system nominal behavior.

From a netlist description, the following dependency relations hold for each circuit element:

Resistor defines a topology equation and adds its current to the nodes it is connected to.

$$-i_R \cdot R + (u_{+node} - u_{-node} - E_R) = 0; \quad (4)$$

$$\begin{aligned} u_{+node} &: + i_R; \\ u_{-node} &: - i_R. \end{aligned} \quad (5)$$

Conductor defines a topology equation and adds its current to the nodes it is connected to

$$-i_Y + (u_{+node} - u_{-node} - E_Y) \cdot Y = 0; \quad (6)$$

$$\begin{aligned} u_{+node} &: + i_Y; \\ u_{-node} &: - i_Y. \end{aligned} \quad (7)$$

Capacitor defines a state equation and a topology equation. Its current gets added to the nodes it is connected to

$$-s x_i \cdot C + i_C = 0; \quad (8)$$

$$-x_i + (u_{+node} - u_{-node}) = 0; \quad (9)$$

$$\begin{aligned} u_{+node} &: + i_C; \\ u_{-node} &: - i_C. \end{aligned} \quad (10)$$

Inductor defines a state equation and a topology equation. Its current gets added to the nodes it is connected to

$$-sx_j \cdot L + (u_{+node} - u_{-node}) = 0; \quad (11)$$

$$-x_j + i_L = 0; \quad (12)$$

$$\begin{aligned} u_{+node} &: +i_L; \\ u_{-node} &: -i_L. \end{aligned} \quad (13)$$

Voltage source defines a topology equation

$$-V + (u_{+node} - u_{-node}) = 0. \quad (14)$$

Printing of node equations for nodes u_{+node} and u_{-node} is inhibited. This is because the current generated by the voltage source automatically adjusts to maintain the voltage across the voltage source.

Current source doesn't define an equation but its current gets added to the nodes it is connected to

$$\begin{aligned} u_{+node} &: +I; \\ u_{-node} &: -I. \end{aligned} \quad (15)$$

Voltage Controlled Voltage Source (VCVS) defines a topology equation

$$\begin{aligned} -E \cdot (u_{+control\ node} - u_{-control\ node}) \\ + (u_{+node} - u_{-node}) = 0. \end{aligned} \quad (16)$$

Current Controlled Current Source (CCCS) adds to the node equations u_{+node} and u_{-node} .

$$\begin{aligned} u_{+node} &: +F \cdot i_{control\ element}; \\ u_{-node} &: -F \cdot i_{control\ element}. \end{aligned} \quad (17)$$

Voltage Controlled Current Source (VCCS) adds to the node equations u_{+node} and u_{-node} .

$$\begin{aligned} u_{+node} &: +G \cdot (u_{+node\ control\ element} \\ &\quad - u_{-node\ control\ element}); \\ u_{-node} &: -G \cdot (u_{+node\ control\ element} \\ &\quad - u_{-node\ control\ element}). \end{aligned} \quad (18)$$

Current Controlled Voltage Source (CCVS) adds a topology equation

$$-H \cdot i_{control\ element} + (u_{+node} - u_{-node}) = 0. \quad (19)$$

Earth node defines one equation

$$u_0 = 0. \quad (20)$$

Printing of the node equation associated with the earth node is inhibited. This equation is redundant as it is a linear combination of the other node equations.

A state equation is an equation containing one or more signal derivatives sx_i and the corresponding state variable is the signal x_i . Variables that are indexed are: derived state variables sx_1, \dots, sx_n , current variables i_{XX} and node variables u_0, \dots, u_j . Equations can be subdivided in:

- state dependency equations (SDE);
- topology node dependency equations (TND);
- node current dependency equations (NCD).

Table I shows the contribution of each element in dependency matrix.

For the fourth order low pass filter (Fig. 2) the following variables are identified:

- derived state variables: sx_0, sx_1, sx_2 and sx_3 ;
- current variables: $i_{R_1}, i_{R_2}, i_{C_1}, i_{C_2}, i_{L_1}$ and i_{L_2} ;
- voltage potential variables: u_0, u_1, u_2, u_3 and u_4 ;
- input variables: V_1 ;
- unknown input variable: E_{R_1} and E_{R_2} .

With the method outlined in this section, the following equations are obtained:

- State equations

$$\begin{aligned} -sx_0 \cdot C_1 + i_{C_1} &= 0; \\ -sx_1 \cdot C_2 + i_{C_2} &= 0; \\ -sx_2 \cdot L_1 + (u_2 - u_3) &= 0; \\ -sx_3 \cdot L_2 + (u_3 - u_4) &= 0. \end{aligned}$$

- Topology equations

$$\begin{aligned} -V_1 + (u_1 - u_0) &= 0; \\ -i_{R_1} \cdot R_1 + (u_1 - u_2 - E_{R_1}) &= 0; \\ -i_{R_2} \cdot R_2 + (u_4 - u_0 - E_{R_2}) &= 0; \end{aligned}$$

TABLE I Circuit element contribution in dependency sub-matrices

Circuit element		Contribution		
Element	Netlist symbol	SDE	TND	NCD
Components				
Resistor	R		X	X
Conductor	Y		X	X
Capacitor	C	X	X	X
Inductor	L	X	X	X
Sources				
Voltage source (VS)	V		X	
Current source (CS)	I			X
Voltage controlled voltage source (VCVS)	E		X	
Voltage controlled current source (VCCS)	F			X
Current controlled voltage source (CCVS)	G		X	
Current controlled current source (CCCS)	H			X

$$\begin{aligned}
-x_0 + (u_2 - u_0) &= 0; \\
-x_1 + (u_3 - u_0) &= 0; \\
-x_2 + i_{L_1} &= 0; \\
-x_3 + i_{L_2} &= 0; \\
u_0 &= 0.
\end{aligned}$$

- Kirchoff node equations

$$\begin{aligned}
-i_{R_1} + i_{C_1} + i_{L_1} &= 0; \\
+i_{R_2} - i_{L_2} &= 0; \\
+i_{C_2} - i_{L_1} + i_{L_2} &= 0.
\end{aligned}$$

A matrix that holds the variable/equation dependencies can be constructed. The rows of the matrix correspond to successive equations and a column of the matrix index a variable. The matrix is filled with zeros and ones. A one (black point) in position (r, c) indicates that equation r contains the variable c . A zero indicates that the equation does not contain this variable. Figure 3 (before causality, on the left side of the figure) shows how this translates into a variable/equation dependency matrix. Each row represents an equation in the order given above. Each column represents a variable also in the order given above.

3.2. Construction of a Causal Dependency Matrix

In this step, it's a question of arranging equations and variables in such a way that they can be

successively and easily solved. Causality is assigned to initial non-causal equations stored in the adjacency matrix. The degree of dependency is calculated for each of the following depending variables: derived state variables (sx_X); current variables: ($i_{R_X}, i_{C_X}, i_{L_X}$) and node voltage potential variables: (u_X).

The degree of dependency $nd(\sigma)$ assigned to each depending variable σ is equal to the number of unsolved equations that contain the variable σ . As the input signal, state variables x_X are considered as explicit variables. The calculation of the dependency degree of such explicit variable is not useful in this section. In the example of the low pass filter, the initial dependency degree for each variable is

- $sx_0, sx_1, sx_2, sx_3 \implies 1$;
- $i_{R_2}, i_{C_1}, i_{C_2}, u_1, u_4 \implies 2$;
- $i_{L_1}, i_{L_2}, i_{R_1}, u_2, u_3 \implies 3$;
- $u_0 \implies 5$.

The algorithm of Figure 4 is run in order to assign a causal order to equations. Each step of this algorithm solves a variable σ_k and an equation Eq_k . The weighting of unsolved variables must be adjusted by decreasing the degree of dependency of unsolved variables that are used in equation Eq_k .

Finding only multiple dependency equations means the existence of loops that have to be solved later on. In this case, the remaining

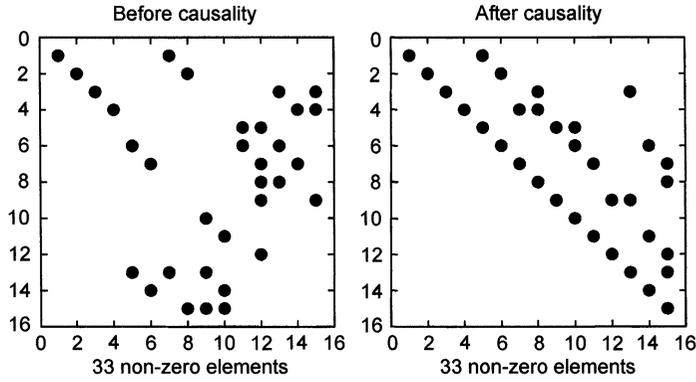


FIGURE 3 V/E dependency matrix for fourth order low pass filter.

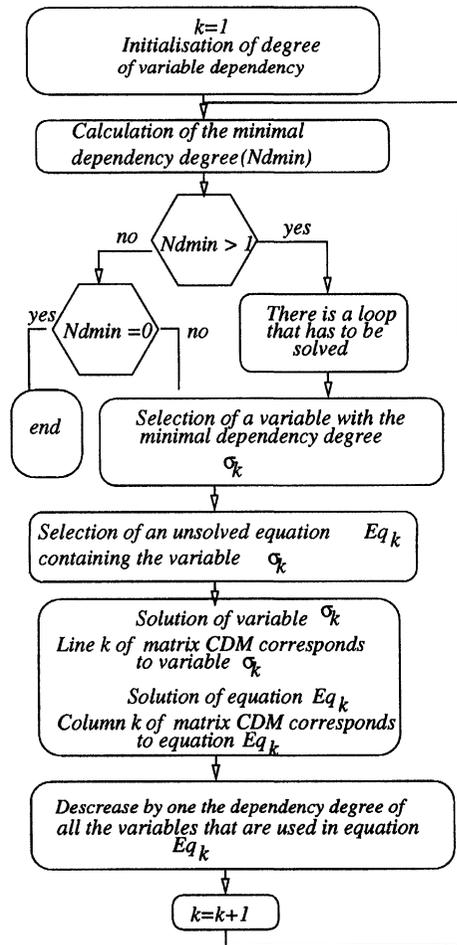


FIGURE 4 Causal dependency matrix algorithm.

variables are reordered on weighted dependency. Lowest dependency degree comes first; highest dependency degree last. If the number of equations/variables is n , it can be proved that the algorithm presented on Figure 4 assigns causality in worst-case time complexity of

$$O\left(\sum_{k=1}^n k \log k\right) \leq O(n^2 \log n).$$

Figure 3 (after causality, on the right) shows the causal variable/equation dependency matrix (CDM) for the fourth order low pass filter. The columns of the figure on the right hold the variables: $sx_0, sx_1, sx_2, sx_3, i_{C_1}, i_{C_2}, u_4, u_3, i_{R_1}, i_{L_1}, i_{R_2}, u_1, u_2, i_{L_2}, u_0$. The rows show the equations:

$$\begin{aligned} -sx_0 \cdot C_1 + i_{C_1} &= 0; \\ -sx_1 \cdot C_2 + i_{C_2} &= 0; \\ -sx_2 \cdot L_1 + (u_2 - u_3) &= 0; \\ -sx_3 \cdot L_2 + (u_3 - u_4) &= 0; \\ -i_{R_1} + i_{C_1} + i_{L_1} &= 0; \\ +i_{C_2} - i_{L_1} + i_{L_2} &= 0; \\ -i_{R_2} \cdot R_2 + (u_4 - u_0 - E_{R_2}) &= 0; \\ -x_1 + (u_3 - u_0) &= 0; \\ -i_{R_1} \cdot R_1 + (u_1 - u_2 - E_{R_1}) &= 0; \\ -x_2 + i_{L_1} &= 0; \\ +i_{R_2} - i_{L_2} &= 0; \\ -V_1 + (u_1 - u_0) &= 0; \\ -x_0 + (u_2 - u_0) &= 0; \\ -x_3 + i_{L_2} &= 0; \\ u_0 &= 0. \end{aligned}$$

Similar results were obtained for other types of circuits. For some circuit structures, it may be difficult to assign causality from the initial circuit schematic (or netlist). In this case, some preliminary classical transformations of the circuit net are required (delta-star or star-delta transformation for example). These net transformation retains the static and dynamical properties for the nodes and the currents but eliminate or create some algebraic equations.

The circuit obtained is equivalent in every respect to the original one, but in contrast with the original circuit, causality can be easily assigned.

3.3. Removing the Loops

After assigning causality to variables in equations, it is possible that some variables depend on other variables declared later in the list. A variable that is responsible of such a loop in the dependency equations has to be eliminated. A simple back-substitution of variables in the next step of the conversion process is applied to remove the loops. Consider that the variable x_i in equation j causes a dependency loop. This means that every equation between index i and index j must be reworked to eliminate this variable. Note that all other equations outside this range are not affected by this operation.

Consider dependency equations from x_i to x_j as $f_i(\cdot)$ to $f_j(\cdot)$. For simplicity, the variables outside this range are not shown in these equations.

$$\begin{aligned} x_i &= f_i(x_{i+1}, \dots, x_j); \\ x_{i+1} &= f_{i+1}(x_{i+2}, \dots, x_j); \\ &\vdots \\ x_{j-1} &= f_{j-1}(x_j); \\ x_j &= f_j(x_i). \end{aligned}$$

Substituting back from x_j to x_i gives the following equations:

$$\begin{aligned} x_j &= f_j(x_i) = g_j(x_i); \\ x_{j-1} &= f_{j-1}(g_j(x_i)) = g_{j-1}(x_i); \\ &\vdots \\ x_{i+1} &= f_{i+1}(g_{i+2}(x_i), \dots, g_j(x_i)) = g_{i+1}(x_i); \\ x_i &= f_i(g_{i+1}(x_i), \dots, g_j(x_i)) = g_i(x_i). \end{aligned}$$

The functions f are linear, the rewrite does not change this linear properties, *i.e.*, all the functions g are also linear. In fact, the last equation in x_i can be characterized as $x_i = g_i(x_i) = \alpha x_i + \beta$. Removing x_i from the right hand side of the equation is as easy as $x_i = \beta / (1 - \alpha)$.

The substitution of this new value for x_i in all the equations g removes one loop.

3.4. Deriving the Nominal Model

Using the causal variable/equation dependency matrix, the system can be solved by a simple back-substitution of variables starting at the last equation and working up to the first. The back-substitution gives a number of state equations of signal variables depending only on the explicit variables (inputs signal or state variables) and the component parameters. Any equation with a single dependency has the following form:

$$\gamma z + \delta_1 x_1 + \delta_2 x_2 + \dots + \delta_n x_n + \eta u.$$

where z is the corresponding depending variable, x_1, x_2, \dots, x_n are the state variables and u is the input signal. This equation can be written in the following explicative form, expressing the depending variable as function of the input signal and state variables:

$$z = \Gamma_1 \cdot x_1 + \Gamma_2 \cdot x_2 + \dots + \Gamma_n \cdot x_n + \Upsilon \cdot u.$$

Isolating the variables selected by the causality assignment process yields the following equations in the case of the low pass filter:

$$\begin{aligned} sx_0 &= i_{C_1}/C_1; \\ sx_1 &= i_{C_2}/C_2; \\ sx_2 &= (u_2 - u_3)/L_1; \\ sx_3 &= 1/L_2 \cdot u_3 - u_4/L_2; \\ i_{C_1} &= i_{R_1} - i_{L_1}; \\ i_{C_2} &= i_{L_1} - i_{L_2}; \\ u_4 &= R_2 \cdot i_{R_2} + (u_0 + E_{R_2}); \\ u_3 &= u_0 + x_1; \\ i_{R_1} &= -1/R_1 \cdot u_2 + (u_1 - E_{R_1})/R_1; \\ i_{L_1} &= x_2; \\ i_{R_2} &= i_{L_2}; \\ u_1 &= u_0 + V_1; \\ u_2 &= u_0 + x_0; \\ i_{L_2} &= x_3; \\ u_0 &= 0. \end{aligned}$$

Now, these equations have to be put into matrix form with the state variables split off from the element parameters: The equations expressing the state variable derivatives (sx_i) function of input signal and state variables constitute the state dynamic equations and the equations giving the node voltage signals (u_j) function of input signal and state variables form the system output equations. The nominal model has the following expressions.

$$\begin{aligned} \dot{\mathbf{x}} &= \begin{pmatrix} \frac{-1}{R_1 C_1} & 0 & \frac{-1}{C_1} & 0 \\ 0 & 0 & \frac{1}{C_2} & \frac{-1}{C_2} \\ \frac{1}{L_1} & \frac{-1}{L_1} & 0 & 0 \\ 0 & \frac{1}{L_2} & 0 & \frac{-R_2}{L_2} \end{pmatrix} \mathbf{x} + \begin{pmatrix} \frac{1}{R_1 C_1} \\ 0 \\ 0 \\ 0 \end{pmatrix} V_1; \\ \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} &= \begin{pmatrix} u_2 \\ u_4 \\ u_3 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & R_2 \\ 0 & 1 & 0 & 0 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} V_1. \end{aligned}$$

3.5. Unknown Input and Fault Model

Noise generated by the elements should *not* be considered as a fault; their effect is temporary and does not affect the functionality of the circuit. For simplicity and without loss of generality, it is assumed here that only resistors (and conductors) generate noise. Their influence is modeled as a small voltage source E_R in series with the resistor. This voltage source is an unknown input to the system because it cannot be measured or controlled. This modeling can easily be extended to any other type of noise disturbing the system nominal behavior. The machinery described in Subsections 3.1–3.4 derives the matrixes for the unknown inputs. For the fourth order low pass filter, the unknown input model is:

$$\mathbf{E} = \begin{pmatrix} \frac{1}{R_1 C_1} & \frac{1}{R_1 C_1} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}; \quad \mathbf{F} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Noise affects the state variables because \mathbf{E} is non-zero. But noise does not directly influence the outputs as \mathbf{F} is zero.

Fault models describe how fault processes interact with the function of the hardware. A linear approximation around the nominal model is used to extract a fault model. Matrices \mathbf{A} and \mathbf{B} can be concatenated as they are affected by the same element faults.

$$(\mathbf{A} + \Delta\mathbf{A}\mathbf{f}\mathbf{B} + \Delta\mathbf{B}\mathbf{f}) \quad (21)$$

The nominal symbolic matrix for each element is substituted by a linear approximation. Any faulty element X in the circuit is replaced by a power series as follows:

$$X \mapsto X \cdot \left(1 + \frac{dX}{X} + o\left(\frac{dX^2}{X}\right) \right).$$

This substitution is made for each element in the symbolic matrix. Then the fault matrix can be extracted without difficulty. The power series $O(\cdot)$ nullified and the fault matrix can then be subtracted from the nominal matrix to obtain the difference matrix. This gives the linear approximation for the difference variables for each element.

4. DESIGN OF ON-LINE DETECTION CIRCUIT

The extended state space model derived in the previous section for linear analog circuits is used in this section to design an optimal error detection circuit. The detection circuitry is built using the dead-beat observer principle. The observer is combined so that its output is zero under nominal conditions and non-zero in faulty conditions.

4.1. Dead-beat Observer Principle

It is assumed that the circuit model can be described with Eqs. (1) and (2). In general, state vector \mathbf{x} is not directly measurable from the circuit under test. The error detection circuitry must use

only available measurable signals: input vector \mathbf{u} , and output vector \mathbf{y} . The unknown state parameters of vector \mathbf{x} have to be eliminated from Eqs. (1) and (2). This goal is achieved by expressing the successive derivatives of the output vector \mathbf{y} explicitly in terms of \mathbf{x} , \mathbf{u} , \mathbf{d} and \mathbf{f} . For simplicity and without loss of generality, only single input is considered.

The k successive first derivatives of the output vector are given by the following equations:

$$\begin{aligned} \mathbf{y}^{(0)}(t) &= \mathbf{C}\mathbf{x}^{(0)}(t) + \mathbf{D}\mathbf{u}^{(0)}; \\ \mathbf{y}^{(1)}(t) &= \mathbf{C}\mathbf{A}\mathbf{x}^{(0)}(t) + \mathbf{C}\mathbf{B}\mathbf{u}^{(0)}(t) + \mathbf{D}\mathbf{u}^{(1)}(t); \\ &\vdots \\ \mathbf{y}^{(k)}(t) &= \mathbf{C}\mathbf{A}^k\mathbf{x}^{(0)}(t) + \mathbf{C}\mathbf{A}^{k-1}\mathbf{B}\mathbf{u}^{(0)}(t) \\ &\quad + \mathbf{C}\mathbf{A}^{k-2}\mathbf{B}\mathbf{u}^{(1)}(t) + \dots + \mathbf{C}\mathbf{B}\mathbf{u}^{(k-1)}(t) \\ &\quad + \mathbf{D}\mathbf{u}^{(k)}(t). \end{aligned}$$

To obtain the equations for \mathbf{d} , substitute in the above equations \mathbf{E} for \mathbf{B} and \mathbf{F} for \mathbf{D} ; likewise for \mathbf{f} , substitute \mathbf{K} for \mathbf{B} and \mathbf{G} for \mathbf{D} .

These equations can be grouped in the following matrix form:

$$\mathbf{Y}^{[k]} = \mathbf{O}\mathbf{b}_k\mathbf{x}(t) + \mathbf{H}_u^{[k]}\mathbf{U}^{[k]} + \mathbf{H}_d^{[k]}\mathbf{D}^{[k]} + \mathbf{H}_f^{[k]}\mathbf{F}^{[k]} \quad (22)$$

where

$$\begin{aligned} \mathbf{Y}^{[k]} &= \begin{pmatrix} \mathbf{y}^{(0)}(t) \\ \mathbf{y}^{(1)}(t) \\ \vdots \\ \mathbf{y}^{(k)}(t) \end{pmatrix}; \quad \mathbf{U}^{[k]} = \begin{pmatrix} \mathbf{u}^{(0)}(t) \\ \mathbf{u}^{(1)}(t) \\ \vdots \\ \mathbf{u}^{(k)}(t) \end{pmatrix}; \\ \mathbf{D}^{[k]} &= \begin{pmatrix} \mathbf{d}^{(0)}(t) \\ \mathbf{d}^{(1)}(t) \\ \vdots \\ \mathbf{d}^{(k)}(t) \end{pmatrix}; \quad \mathbf{F}^{[k]} = \begin{pmatrix} \mathbf{f}^{(0)}(t) \\ \mathbf{f}^{(1)}(t) \\ \vdots \\ \mathbf{f}^{(k)}(t) \end{pmatrix}; \\ \mathbf{O}\mathbf{b}_k &= \begin{pmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \\ \vdots \\ \mathbf{C}\mathbf{A}^k \end{pmatrix}. \end{aligned} \quad (23)$$

and

$$\mathbf{H}_u^{[k]} = \begin{pmatrix} \mathbf{D} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{CAB} & \mathbf{CB} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{CA}^{k-1}\mathbf{B} & \mathbf{CA}^{k-2}\mathbf{B} & \dots & \mathbf{CB} & \mathbf{D} \end{pmatrix}; \quad (24)$$

The matrix $\mathbf{H}_d^{[k]}$ (respectively $\mathbf{H}_f^{[k]}$) is obtained from the expression of matrix $\mathbf{H}_u^{[k]}$ by substituting \mathbf{B} by \mathbf{E} (respectively by \mathbf{K}) and \mathbf{D} by \mathbf{F} (respectively by \mathbf{G}). The integer k is the number of output derivatives required for residual generation. k will also define the order of the residual, *i.e.*, the number of integration needed in the residual generation scheme. Generally, in the case of electronic circuits, more than one integration is rarely necessary. If n is the order of the system, then $\mathbf{Ob}_k|_{k=n-1}$ is the observability matrix of the system. In the nominal conditions ($d(t)=f(t)=0$) Eq. (22) has the following simpler form:

$$Y^{[k]} = \mathbf{Ob}_k \mathbf{x}(t) + \mathbf{H}_u^{[k]} U^{[k]}. \quad (25)$$

In Eq. (25), it is clear that the signal vectors $Y^{[k]}$ (node signal vector) and $U^{[k]}$ (input vector) are assumed to be measurable, when the state vector $x(t)$ is not necessarily available for connection. The basic idea of the residual generation method presented in this paper consists of eliminating the unknown variable in Eq. (25). This elimination may be performed by combination of equations or by projection. The projection technique consists of determining the subspace P_s of all the vectors \mathbf{v} defined by Eq. (26).

$$P_s = \{\mathbf{v} | \mathbf{v}^T \mathbf{Ob}_k = 0\}. \quad (26)$$

Every vector \mathbf{v}_j of the subspace P_s can be used to generate a scalar residual $r_j(t)$ that can be used for a parity check. Equation (27) will just do that.

$$r_j(t) = \mathbf{v}_j^T \left(Y^{[k]} - \mathbf{H}_u^{[k]} U^{[k]} \right). \quad (27)$$

Using Eqs. (22) and (26), the expression of the residual r_j become

$$\begin{aligned} r_j(t) &= \mathbf{v}_j^T \left(\mathbf{Ob}_k \mathbf{x}(t) + \mathbf{H}_d^{[k]} \mathbf{D}^{[k]} + \mathbf{H}_f^{[k]} \mathbf{F}^{[k]} \right) \\ &= \mathbf{v}_j^T \left(\mathbf{H}_d^{[k]} \mathbf{D}^{[k]} + \mathbf{H}_f^{[k]} \mathbf{F}^{[k]} \right). \end{aligned}$$

From this explicative form, it is clear that the residual r_j is not affected neither by the initial condition $\mathbf{x}(0)$ nor by the input signal $\mathbf{u}(t)$ but is only function of disturbances \mathbf{d} and faults \mathbf{f} . Hence r_j can be used for the purpose of fault detection. When there is no fault ($\mathbf{f}=0$), then r_j is only affected by unknown inputs, \mathbf{d} . When however a fault occurs ($\mathbf{f} \neq 0$) r_j is influenced by both \mathbf{f} and \mathbf{d} and its value increases. Thus, a fault can be detected by checking the increment of r_j caused by \mathbf{f} .

The algorithm of Figure 5 summarizes the residual generation process.

For the low pass filter of Figure (2), an application of the relations defined in Eq. (23) (for $k=1$) gives

$$\begin{aligned} U^{[1]} &= \begin{pmatrix} u \\ \dot{u} \end{pmatrix}, \quad Y^{[1]} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix}, \\ \mathbf{Ob}_1 &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & R_2 \\ 0 & 1 & 0 & 0 \\ \frac{-1}{R_1 C_1} & 0 & \frac{-1}{C_1} & 0 \\ 0 & \frac{R_2}{L_2} & 0 & \frac{-R_2^2}{L_2} \\ 0 & 0 & \frac{1}{C_2} & \frac{-1}{C_2} \end{pmatrix}, \quad (28) \\ \mathbf{H}_u^{[1]} &= \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{R_1 C_1} & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}. \end{aligned}$$

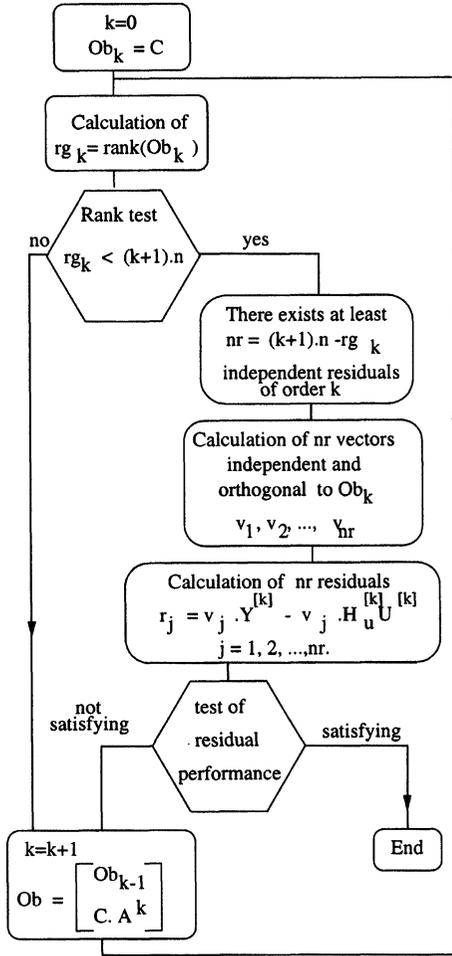


FIGURE 5 Residual generation algorithm.

Since \mathbf{Ob}_1 is not a full line-rank matrix, the subspace P_s orthogonal to \mathbf{Ob}_1 and defined by Eq. (26) is not zero. It is easy to verify that the vectors \mathbf{v}_1 and \mathbf{v}_2 defined by Eqs. (29) and (30) respectively belong to P_s

$$\mathbf{v}_1^T = (R_2 \quad R_1 \quad 0 \quad R_1 R_2 C_1 \quad 0 \quad R_1 R_2 C_2); \quad (29)$$

$$\mathbf{v}_2^T = (0 \quad R_2 \quad -R_2 \quad 0 \quad L_2 \quad 0). \quad (30)$$

According to Eq. (27), the vectors \mathbf{v}_1 and \mathbf{v}_2 can be used to provide two independent residuals $r_1(t)$ and $r_2(t)$ defined respectively by the following

equations:

$$r_1(t) = R_2 y_1 + R_1 y_2 + R_1 R_2 C_1 \dot{y}_1 + R_1 R_2 C_2 \dot{y}_3 - R_2 u; \quad (31)$$

$$r_2(t) = R_2 y_2 - R_2 y_3 + L_2 \dot{y}_2. \quad (32)$$

More generally, any vector \mathbf{v}_j in subspace P_s linear is a linear combination of form

$$\mathbf{v}_j = \delta_1 \mathbf{v}_1 + \delta_2 \mathbf{v}_2 = \vartheta_s \delta. \quad (33)$$

Where δ is a vector of real numbers of form $\delta^T = [\delta_1, \delta_2]$, and ϑ_s has the meaning of the base for P_s defined by $\vartheta_s = [\mathbf{v}_1, \mathbf{v}_2]$.

To any vector \mathbf{v}_j defined by Eq. (33) corresponds a scalar residual defined by Eq. (27). The following equation gives the general expression of such a scalar residual.

$$r_j(t) = \alpha_0 y_1 + \beta_0 y_2 + \gamma_0 y_3 + \alpha_1 \dot{y}_1 + \beta_1 \dot{y}_2 + \gamma_1 \dot{y}_3 + \lambda_0 u. \quad (34)$$

Where the design parameters $\alpha_0, \beta_0, \gamma_0, \alpha_1, \beta_1, \gamma_1$ and λ_0 are function of vector \mathbf{v}_j (i.e., function of δ_1 and δ_2).

4.2. Optimization of the Residual Equation

Ideally, the resulting detector should react strongly under the presence of a fault but should not react to unknown inputs to the system, like noise. The following equations put these robustness requirements into mathematical rigor:

$$\mathbf{v}_j^T \mathbf{H}_f^{[k]} \neq 0; \quad (35)$$

$$\mathbf{v}_j^T \mathbf{H}_d^{[k]} = 0. \quad (36)$$

Finding a solution \mathbf{v}_j^T that satisfies Eqs. (26)–(36) provides a robust residual that is unaffected by unknown inputs but is affected by a fault.

Unfortunately, these equations may not yield a solution. In this case, an optimal approximation due to a performance criterion can be found. The performance criterion to be used must take into account the sensitivity of the residual with respect

to unknown disturbance inputs \mathbf{d} as well as the sensitivity with respect to faults \mathbf{f} . It is then useful to define a performance measurement that shows in a way how much the detector is sensitive to faults and insensitive to unknown inputs. The following performance index satisfies these constraints:

$$\rho = \frac{\|\mathbf{v}_j^T \mathbf{H}_d^{[k]}\|}{\|\mathbf{v}_j^T \mathbf{H}_f^{[k]}\|}. \quad (37)$$

where $\|\cdot\|$ is the 2-norm; the largest singular value of the matrix. Minimizing the performance index ρ will yield the desired residual.

$$\rho_{\min} = \min_{\mathbf{v} \in P_s} \left(\frac{\|\mathbf{v}^T \mathbf{H}_d^{[k]}\|}{\|\mathbf{v}^T \mathbf{H}_f^{[k]}\|} \right). \quad (38)$$

This constrained optimization problem can be reduced to the following unconstrained optimization:

$$\rho_{\min} = \min_{\delta} \left(\frac{\|\delta^T \vartheta_s \mathbf{H}_d^{[k]}\|}{\|\delta^T \vartheta_s \mathbf{H}_f^{[k]}\|} \right). \quad (39)$$

Where ϑ_s has the meaning of a base for P_s . Hence, δ singles out the best vector \mathbf{v}^T amount all the possible solutions represented by ϑ_s . The solution to this optimisation problem is obtained by a differentiation of the performance index ρ . This leads to the relation

$$\delta^T (\vartheta_s \mathbf{H}_d^{[k]} \mathbf{H}_d^{[k]} \vartheta_s^T - \rho \vartheta_s \mathbf{H}_f^{[k]} \mathbf{H}_f^{[k]} \vartheta_s^T) = 0. \quad (40)$$

The solution of this of this equation gives the vector \mathbf{v}^T solution to optimal detector design problem.

4.3. Residual Implementation

Normally, the implementation of the residual r_j using Eq. (27) requires the differentiation of the output and input signals. This operation is prone to error. Integrating the Eq. (34) once (k times in the general case) replaces differentiation by integration. The drawback of this trick is a potentially unstable pole at $s=0$. The stabilization is obtained by using a negative feedback factor $P=-p_0$ of the residual output into the detection circuit. To provide a stable residual signal, pure integrator has to be replaced by a one a order filter. The division of the Laplace transform of relation (34) by $(s+p_0)$ gives the expression of filtered residual r_{ff} , usable for on-line fault detection.

$$\begin{aligned} r_{ff}(s) = & (\alpha_0 y_1(s) + \beta_0 y_2(s) + \gamma_0 y_3(s) + \lambda_0 u(s) \\ & + s\alpha_1 y_1(s) + s\beta_1 y_2(s) + s\gamma_1 y_3(s)) / (s + p_0) \end{aligned} \quad (41)$$

The use of a one order transfer function provides a mean to control the sensitivity of the residual signal to tolerances in component values. As in [2], the residual filter also allows an adjustment of the response of the test circuitry to sharp transients. The general scheme of the residual block diagram an the corresponding detection circuit schematic are given on Figures 6 and 7 respectively.

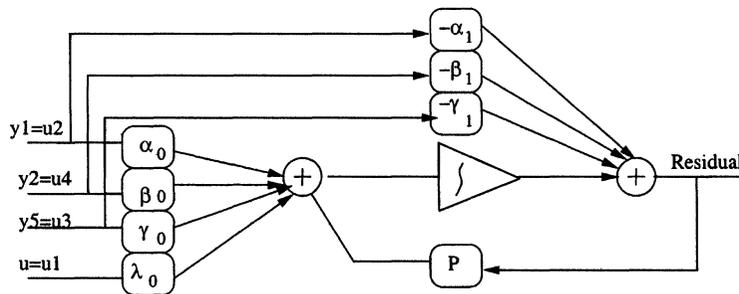


FIGURE 6 Block diagram of the error detection circuit.

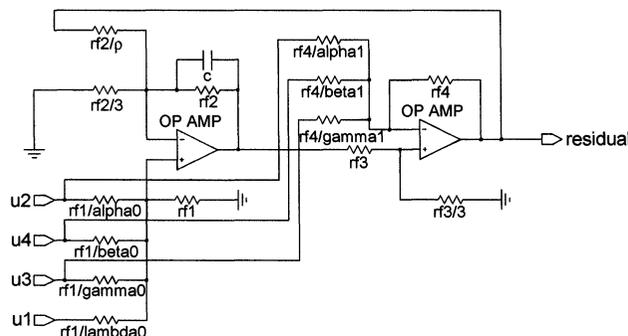


FIGURE 7 Schematic of one order detection circuit.

The size of the detection circuitry is only function of the number k of output derivatives required in the residual generation process. This number depends on the quality of information available on node voltages and not on the size of the circuit under test.

5. SIMULATION RESULTS

The method described in the previous sections has been applied to different type of analog circuits including biquadratic filter, high pass filter, Pierce oscillator and the low pass filter example of Figure 2. For simplicity, the results presented in this section correspond to the low pass filter circuit of Figure 2 only. The results obtained for the other circuits were very similar and are not presented here.

In each simulation, the following component values were used: $R_1 = 5 \text{ k}\Omega$, $R_2 = 1 \text{ k}\Omega$, $C_1 = 1 \mu\text{F}$, $C_2 = 5 \mu\text{F}$, $L_1 = 30 \text{ mH}$ and $L_2 = 60 \text{ mH}$.

It was first verified that each residual was very close to zero in the fault free condition. The single fault simulation was applied on the circuit model. Open, short as well as positive and negative parametric faults were simulated on each circuit element separately, when the other elements were held at their nominal value.

The Figures 8 and 9 show the two residual wave forms obtained for positive parametric deviation of $+50\%$. The overall shapes of the residual were generally very close for symmetric negative offsets.

The results represented on Figures 8 and 9 are obtained on two independent residuals for $+50\%$ parametric deviation faults. The simulation results for fault detection are summarized in the Tables II and III that hold the maximum gain values associated with each residual for each faulty element.

With a reasonable detection threshold corresponding to a gain of 0.01, it is clear on Tables II and III that the detection of all the catastrophic faults (*i.e.*, short and open on any circuit component) is ensured on each of the both residuals.

Figure 8 and Table II show that the parametric faults on resistors R_1 and R_2 are detected by the first residual up to about 5000 rad/sec. Parametric faults on capacitors C_1 and C_2 are detected only in the frequency range 500–5000 rad/sec when $\pm 50\%$. Parametric faults on inductors are undetected by the two detectors.

Each of the residual may probably cover a different spectrum of possible faults. To obtain larger coverage results, it may be judicious to combine different residuals with different spectrum and fault coverage. But we are still looking for a formal method. The choice for a residual or possibly a combination of residuals can depend on

- the allotted space for the error detection circuitry;
- the desired fault coverage;
- the operational frequency range;
- the criticality of faults covered (in terms of effects gravity and occurrence probability).

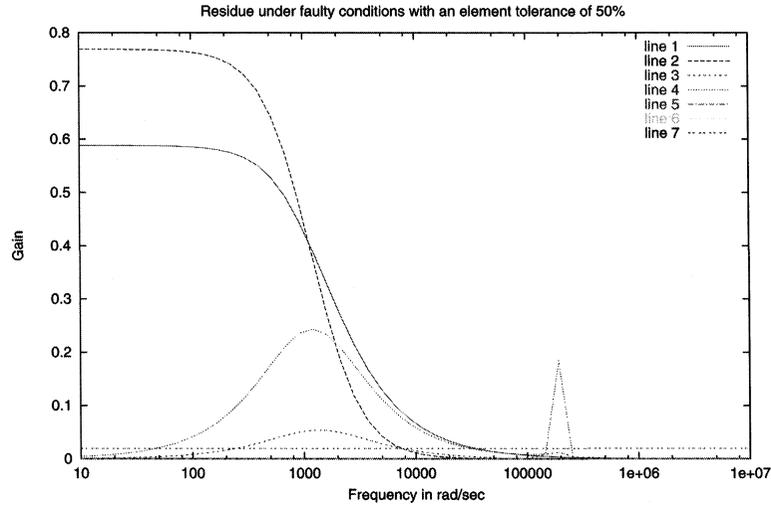


FIGURE 8 First residue under faulty conditions.

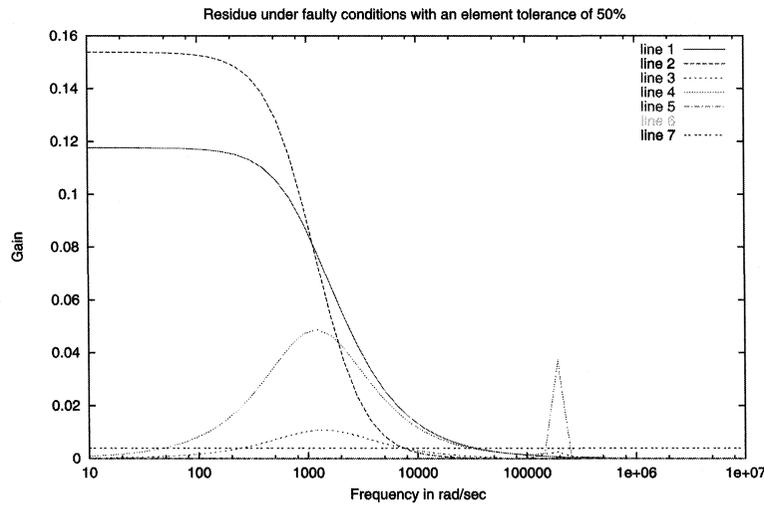


FIGURE 9 Second residue under faulty conditions.

TABLE II Maximum response for first residual

Element	Open	+ 50%	-50%	Short
R_1	2.062	0.588	1.428	3.322
R_2	1.325	0.769	0.909	2.041
C_1	1.071	0.0548	0.162	1.784
C_2	1.813	0.244	0.322	1.05
L_1	0.539	0	0	0.478
L_2	0.727	0	0	0.361

TABLE III Maximum response for second residual

Element	Open	+ 50%	-50%	Short
R_1	0.8022	0.118	0.286	2.661
R_2	1.430	0.154	0.182	1.784
C_1	0.4345	0.0110	0.0324	0.783
C_2	0.988	0.0488	0.0645	0.931
L_1	0.167	0	0	0.389
L_2	0.122	0	0	0.159

In the end, the designer has to choose between maximum fault coverage and minimum space.

6. CONCLUSION

The first part of this paper deals with the generation of an extended symbolic model for linear analog circuits from its netlist description. This extended model required for efficient on-line fault detection comprises state space matrices for the nominal model, an unknown noise input model and a fault model. The noise sources are modeled by the unknown inputs. Fault models are generated from a sensitivity analysis of the symbolic state space model. From the previous extended model of the circuit, an optimal concurrent error detection circuitry is derived in the second part of this paper. The detector optimality request here corresponds to maximal sensitivity to faults and minimal sensitivity to unknown inputs.

Contrary to previously proposed methods on concurrent fault detection of linear analog circuits, the methodology presented in this paper is usable for any linear analog circuit, state variable or non. Further development should focus on the extension to non-linear problems. As a first step, non-linearity can be captured in the unknown input vectors. Techniques on how this can be accomplished for electronic networks have yet to be developed.

References

- [1] Jou, J. Y. and Abraham, J. A., Fault tolerant FFT networks, *IEEE Transactions on Computers*, 37, May, 1988, 548–561.
- [2] Chatterjee, A. (1993). Concurrent Error Detection and Fault Tolerance in Linear Analog Integrated Circuits Using

- Continuous Checksums, *IEEE Transactions on VLSI Systems*, 1(2), 138–150.
- [3] Chow, E. Y. and Willisky, A. S. (1984). Analytical Redundancy and the Design of Robust Failure Detection System, *IEEE Transactions on Automatic Control*, 29, 603–614.
- [4] Nair, V. S. S. and Abraham, J. A. (1990). Real-Number Code for Fault Tolerance Matrix Operations on Processor Array, *IEEE Transactions on Computers*, 39(4), 426–435.
- [5] Reddy, L. N. and Banerjee, P. (1990). Algorithm-Based Fault Detection for Signal Processing Applications, *IEEE Transactions on Computers*, 39(10), 1304–1308.
- [6] Frank, P. M. (1990). Fault diagnosis in dynamical systems using analytical and knowledge based redundancy—A survey and some new results, *Automatica*, 26(3), 459–474.
- [7] Chen, J., Patton, R. J. and Zhang, H. Y. (1996). Design of Unknown Input Observers and Robust Fault Filters, *International Journal of Control*, 63(1), 85–105.

Author's Biography

Emmanuel Simeu received his electrical engineering degree from Casablanca institute of engineering sciences in 1987, the DEA in automatic control and signal processing and the Ph.D. in automatic control and system theory, all from national poly-technique institute of Grenoble in 1988 and 1992 respectively.

From 1989 to 1995 he was researcher in systems and control group in Automatic Control Laboratory of Grenoble. From 1992 to 1995, he also held the post of associate professor at Valence Institute of Automatic and Robotic.

Since September, 1995, he is an Associate Professor of dependability analysis and automatic control in the Department of industrial risk management at Science and Technique Institute of Joseph Fourier University of Grenoble. He has been researcher in Reliable Integrated Systems Group of TIMA laboratory since 1995.

His current research interests include on-line testing, synthesis of testable design, experimental design for fault detection, reliability analysis, robust monitoring, modeling and optimization.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

