



Inteligencia Artificial. Revista Iberoamericana
de Inteligencia Artificial

ISSN: 1137-3601

revista@aepia.org

Asociación Española para la Inteligencia
Artificial
España

Pous, Carles; Colomer, Joan; Meléndez, Joaquim; de la Rosa, Josep Lluís
Información cualitativa en el razonamiento basado en casos para el diagnóstico de circuitos
analógicos
Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial, vol. 8, núm. 23, verano, 2004, p.
0
Asociación Española para la Inteligencia Artificial
Valencia, España

Disponible en: <http://www.redalyc.org/articulo.oa?id=92502311>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica
Red de Revistas Científicas de América Latina, el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Información Cualitativa en el Razonamiento Basado en Casos para el Diagnóstico de Circuitos Analógicos

Carles Pous, Joan Colomer, Joaquim Meléndez, Josep Lluís de la Rosa

*

Institut d'Informàtica i Aplicacions. Universitat de Girona
Campus de Montilivi, Edifici p-4
Girona, 17071
{carles,colomer,quimmel,pepluis}@eia.udg.es

Resumen

Actualmente se dispone de numerosas técnicas de Inteligencia Artificial desarrolladas para diagnosticar circuitos electrónicos. Pero en el campo de los circuitos electrónicos analógicos todavía queda mucho por resolver. El propósito de este artículo es el de contribuir con una nueva metodología. La intención es desarrollar una metodología para circuitos analógicos que se fundamenta en la mejora de los métodos basados en diccionarios de fallas. Se dotará al diccionario de la capacidad de aprendizaje ya sea, añadiendo casos a la base, o bien adaptando los existentes. De esta forma se acaba construyendo un sistema de Razonamiento Basado en Casos (CBR). A modo de ejemplo se estudia un método de diccionario de fallas en concreto, como punto de partida para construir la base de casos inicial. La metodología se aplica a un circuito electrónico analógico real. Las fallas consideradas son paramétricas, permanentes, independientes y simples.

Palabras clave: Razonamiento basado en casos, diagnóstico de circuitos, inteligencia artificial.

1. Introducción

El diagnóstico de circuitos se basa en el análisis de la respuesta del sistema a ciertos estímulos. Por ello es necesario generar señales que sirvan de estímulo y adquirir medidas en nodos particulares del circuito. Hay múltiples métodos para diagnosticar circuitos tal y como se describen en [2]. Pero en estas últimas décadas el diagnóstico de fallos mediante técnicas de inteligencia artificial está siendo uno de los campos de investi-

gación más importantes. En [6] se da una posible clasificación de las diferentes técnicas de Inteligencia Artificial utilizadas para el diagnóstico de circuitos electrónicos. La Figura 1 resume dicha clasificación.

Las técnicas de Inteligencia Artificial pueden ser aplicadas tanto por sí mismas o como de refuerzo a otras técnicas. Entre ellas se pueden citar Fuzzy, Redes Neuronales, Sistemas Expertos y Razonamiento Basado en Casos. Según [6], estas técnicas pueden clasificarse en:

* datos de la edición del artículo.

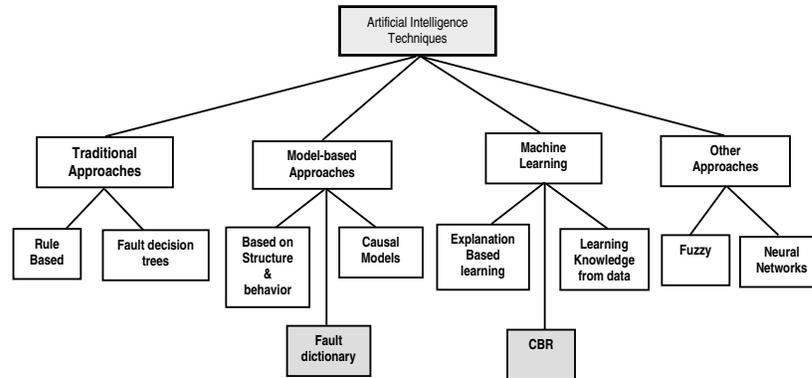


Figura 1: Clasificación de los métodos de IA

Tradicionales: Se basa en reglas IF-THEN y son muy intuitivas. Su principal inconveniente radica en obtener la información de los expertos y cómo tratar las fallas nuevas, ya que no incluyen la posibilidad de aprender por sí solas.

Basadas en el modelo: Utilizan el modelo para hacer las predicciones de las fallas. Presentan los mismos inconvenientes que las aproximaciones tradicionales ya que no pueden tratar con fallas no simuladas previamente, y al mismo tiempo, hace falta el conocimiento del experto para la construcción del modelo.

Machine Learning: Aprovechan las conclusiones obtenidas en diagnósticos anteriores para aprender de los errores o aciertos, y mejorar las prestaciones del sistema de diagnóstico. En general, requieren de bases de datos grandes y fiables para obtener buenos resultados.

Otras: Entre estas pueden citarse, por ejemplo, el fuzzy o las redes neuronales. La primera proporciona una forma intuitiva y sencilla de representar el conocimiento. Las redes neuronales tienen la capacidad de poder modelar situaciones no previstas resintonizando sus parámetros con un entrenamiento adecuado.

La tendencia futura es una solución híbrida constituida por diferentes técnicas de la clasificación anterior. Este es uno de los campos de investigación que ha ganado más adeptos en estos últimos años. Éste es el caso del presente artículo. Aquí se propone una nueva metodología para el diagnóstico de circuitos electrónicos analógicos que parte de un modelo de diccionario de fallos. Para completar las deficiencias que estos presentan, se toma el diccionario como punto de partida para la construcción de un sistema de Razon-

amiento Basado en Casos.

El artículo está organizado como sigue: La siguiente sección da una breve introducción a los diccionarios de fallos y sus limitaciones. La sección 3 describe la nueva metodología propuesta para la construcción del sistema CBR a partir del diccionario de fallos. Seguidamente se aplica al diagnóstico de un circuito electrónico, mostrando los resultados obtenidos. Finalmente se dan las conclusiones y el trabajo futuro a realizar.

2. Diccionarios de Fallos y sus Limitaciones

Los diccionarios de fallos son técnicas basadas en cálculos cuantitativos. Se obtiene una selección de medidas determinadas del circuito que se almacenan en una tabla. Estos valores son comparados con los medidos en el circuito bajo test. La comparación usualmente se hace mediante criterios de vecindad, bien obteniendo distancias o minimizando ciertos índices. La clase a la que corresponde el vecino extraído más cercano es la tomada como posible diagnóstico. Una clasificación de los principales diccionarios de fallos se puede hallar en [2].

Las principales dificultades con las que se enfrentan los sistemas de diccionarios de fallos son las tolerancias y las fallas no previstas previamente, ya que las medidas tomadas del circuito bajo test, no tienen porque coincidir con las guardadas en la tabla, dando muchas veces un diagnóstico erróneo. El objetivo principal

Case Num	Meas. 1	Meas. 2	...	Meas. n	Class	Compo	Desvi.
Case i	M1i	M2i	...	Mni	Class i	Compo i	X %

Cuadro 1: Estructura del caso

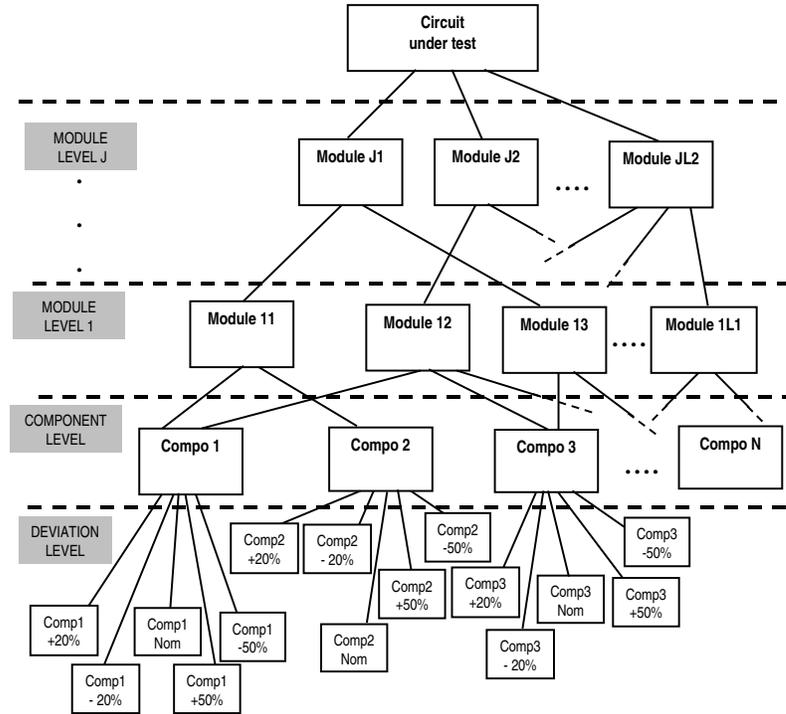


Figura 2: General Case hierarchy

es obtener un diccionario lo más reducido posible pero, al mismo tiempo, robusto a las tolerancias.

A menudo sucede que ciertas fallas producen los mismos valores en las medidas. Es decir, tienen exactamente los mismos síntomas, lo cual, hará que sean indistinguibles entre ellas. Estas fallas forman lo que se denominan *conjuntos ambiguos*. Dependerán de los puntos de medida escogidos (cuántos puntos y dónde), de la topología del circuito y de la precisión de las medidas tomadas. Es importante pues, detectar estos conjuntos cuando se construye el diccionario, existiendo varios métodos para hacerlo, como los propuestos en [10] o en [11].

3. Metodología Propuesta

Los diccionarios de fallas pueden verse como una aproximación muy simplificada de un sistema CBR. La propuesta pues, es extender el dic-

cionario de fallos hacia un sistema CBR capaz de aprender y de subsanar las limitaciones de los diccionarios de fallas. Seguidamente se describe en sistema propuesto en detalle.

3.1. Construcción de la Base de Casos y Memoria de Casos

La estructura del caso se escoge de forma muy similar a la del diccionario de fallas, incluyendo un campo que identifica la clase de la falla, otro con el componente dañado, y otro con la desviación exacta producida en el valor de dicho componente. La estructura típica se muestra en la Tabla 1.

El caso está claramente dividido en dos partes: una numérica correspondiente a las medidas tomadas y utilizada básicamente para la extracción de los casos, y otra cualitativa con información relativa al tipo de falla correspondiente que

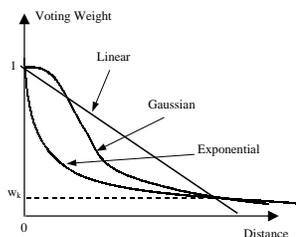


Figura 3: Distance-Weighted kernels

será usada principalmente para proporcionar el diagnóstico.

Entonces, se construye una base con un caso correspondiente a cada falla considerada. Como se suele hacer en diccionario de fallas, se toman para empezar desviaciones de $\pm 20\%$ y $\pm 50\%$ respecto del caso nominal. Estas fallas están suficientemente separadas para cubrir un amplio conjunto de fallas típicas. Pero esta base no puede diagnosticar fallas con una desviación, por ejemplo, de $\pm 39,5\%$ o con tolerancias. Para prever el efecto de estas últimas se simula la falla con el valor de los demás componentes variando aleatoriamente dentro de su margen de tolerancia. Las resistencias y condensadores tienen un valor con una distribución gaussiana alrededor del valor nominal ([3]).

Ya que los casos con tolerancia se generan de forma aleatoria, hay casos redundantes, se aplica alguna técnica de reducción de la base, disminuyendo su dimensión al tiempo que aumenta su eficiencia en los aciertos. Las técnicas IB3 (Instance Based Learning Algorithm) y DROP (Incremental Reduction Optimization Procedure) propuestas en [13] son las utilizadas para reducir la base obtenida mediante Monte-Carlo. Estas técnicas se basan en el algoritmo del vecino más cercano.

Se define también una jerarquía, de forma similar a la propuesta en [12]. La figura 2 muestra la idea.

El último nivel corresponde a la desviación del componente ($\pm 20\%$ o $\pm 50\%$). Algunas veces no será posible diagnosticar con tal precisión por lo que la decisión será tomada por un nivel jerárquico superior correspondiente al nombre del componente. Entonces en esta situación, el sistema no es capaz de proporcionar la desviación de la falla pero sí en que componente se halla. También es posible que para determinadas fallas sólo sea factible localizarlas en determinados módulos o partes del circuito. Por ello el circuito

se divide en J niveles de módulos, dependiendo de la complejidad que tenga dicho sistema. En este caso hace falta tener cierto conocimiento de la topología del circuito para generar la jerarquía de módulos de forma adecuada. En [8] o en [4] se proporcionan métodos para hacer la descomposición jerárquica de forma más o menos automática.

3.2. Extracción de Casos

Es necesario definir una métrica y un número de casos a extraer de la base. Entre todas las posibles distancias se ha escogido la euclídea, siendo necesario normalizar sus atributos debido al diferente orden de magnitud que pueden tener los parámetros. Con esto, la distancia entre dos instancias es la mostrada en la ecuación 1

$$E(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^m \left(\frac{x_i - y_i}{range_i} \right)^2} \quad (1)$$

donde \vec{x} y \vec{y} son los vectores correspondientes a las instancias a comparar, x_i y y_i son cada uno de los i atributos correspondientes, y m el número de atributos. $range_i$ es la diferencia entre el máximo y el mínimo valor que toma el atributo i .

El número k de casos extraídos se elige de forma que de el mejor resultado. Normalmente suele ser un número impar y no muy grande, ya que extraer muchos casos puede producir confusión en el diagnóstico.

3.3. Reutilización de Casos

Una vez extraídos los k vecinos más cercanos se hace un examen de la clase de los casos extraídos.

	Vecino	Clase	Compo	Desvi
<i>situation1</i>				
	Nuevo caso	5	<i>Compo2</i>	14,82 %
	Vecino 1	7	<i>Compo2</i>	50 %
	Vecino 2	7	<i>Compo2</i>	50 %
	Vecino 3	5	<i>Compo2</i>	20 %
<i>situation2</i>				
	Nuevo caso	6	<i>Compo3</i>	14,82 %
	Vecino 1	19	<i>Compo5</i>	50
	Vecino 2	19	<i>Compo5</i>	50
	Vecino 3	12	<i>Compo3</i>	50
<i>situation3</i>				
	Nuevo caso	30	<i>Compo8</i>	-34,74 %
	Vecino 1	30	<i>Compo8</i>	-30,50
	Vecino 2	32	<i>Compo8</i>	-50
	Vecino 3	32	<i>Compo8</i>	-50

Cuadro 2: Diversas situaciones de reutilización

Lo más simple es hacer un proceso de votación y contar los casos que hay de cada clase. El diagnóstico final será la clase que haya obtenido más votos. Pero tal y como se propone en [13], es mejor asignar un peso a cada caso extraído en función de la distancia a la que se encuentra del nuevo caso a diagnosticar. Se proponen tres tipos de distancia, que se muestran en la Figura 3.

La distancia ponderada que mejor funciona para los casos que hemos podido probar, ha sido la exponencial, que viene dada por la ecuación 2

$$w_j = w_k \frac{D_j}{D_k} \quad (2)$$

donde w_k es el peso dado al k -ésimo vecino y, D_k y D_j son las distancias al k -ésimo y j -ésimo vecinos respectivamente.

Utilizando pesos se ha comprobado que el porcentaje de aciertos en el diagnóstico mejora pero sólo ligeramente. Por este motivo se decidió analizar la información de los casos extraídos de forma cualitativa. Para ello se tiene en cuenta la información proporcionada por los campos *Compo* y *Desvi* en lugar del *Clase*. Deberán considerarse diferentes situaciones que pueden darse

algunas de las cuales se reproducen en la Tabla 2.

En la situación 1 el nuevo caso corresponde a una falla en el Componente Compo 2 con una desviación del 14,82 %. Si se utilizara sólo información de la clase y con votación, el diagnóstico final sería "La falla es Compo 2 + 50 %" (hay dos casos extraídos de la clase 7 y uno de la 5), lo cual es un diagnóstico erróneo. En cambio, si se utiliza la información cualitativa contenida en los campos *Compo* y *Desvi* se observa que hay dos posibilidades: la falla es Compo 2 + 20 % o bien Compo 2 + 50 %. Por ello la decisión tomada será "La falla es Compo 2", ya que es el primer nivel superior común en la jerarquía. Este diagnóstico pierde precisión, pero al menos no es equivocado.

Para la situación 2, si se utilizara solamente la información proporcionada por la clase, el diagnóstico sería: La falla es Compo 5 + 50 %, cuando realmente es Compo 3 + 14.82 %. Pero si se utiliza la información proporcionada por los otros dos campos, la conclusión sería "La falla es Compo 5 + 50 % o Compo 3 + 50 %", lo cual es un diagnóstico menos preciso pero por lo menos cierto. Una conclusión similar podría obtenerse para la Situación 3.

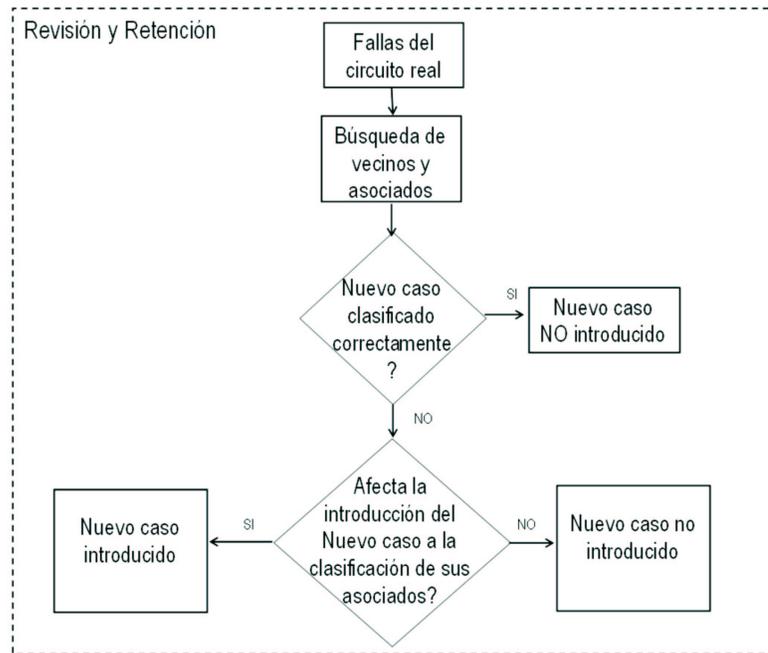


Figura 4: Proceso de Revisión y Retención detallado

3.4. Revisión y Retención de Casos

Una vez se propone un diagnóstico, se debe producir una revisión de los casos extraídos, y cuando corresponda, decidir si el nuevo caso va a ser introducido en la base, o bien si el contenido de la base ya es suficiente para diagnosticar la nueva situación. Al mismo tiempo deberá verse si los casos que van interviniendo en las decisiones lo hacen de forma apropiada o por el contrario inducen a error. Entonces se deberá también considerar la retención o eliminación de los mismos de la base de casos.

Para ello se utiliza la misma filosofía que los métodos DROP4 e IB3 usados para reducir la dimensión de la base inicial generada de forma aleatoria. La decisión a cerca de si el nuevo caso va a ser introducido en la base se hace utilizando los conceptos de *vecino* y *asociado* aplicados en DROP4. Es decir, primero se comprueba que la base actual no da un diagnóstico correcto del nuevo caso. Seguidamente se determina si la introducción de éste estropea la clasificación que ya se tenía de los demás casos de la base analizando si los asociados al nuevo caso no empeoran su clasificación por la introducción de éste último. El diagrama de flujo de la Figura 4 resume este proceso.

Pueden darse ocho situaciones diferentes a la hora de diagnosticar con los casos extraídos:

- Situación 1. El campo *Compo* de los extraídos es idéntico entre todos e igual al del caso nuevo. Además alguna de las desviaciones de los extraídos difiere en menos de un 10% de la del caso nuevo. El diagnóstico es correcto, por lo que no hace falta introducir el nuevo caso a la base
- Situación 2. Igual que el anterior, pero las desviaciones caen fuera del margen del 10% respecto el caso nuevo. Será necesario introducir el nuevo caso a la base, siempre que este no moleste a la clasificación de los demás.
- Situación 3. Los campo *Compo* de los casos extraídos son iguales entre si, pero distintos al del caso nuevo y distintos al caso nominal. Este caso está rodeado completamente por situaciones correspondientes a otro diagnóstico que muy probablemente no le corresponde. Por lo tanto no será introducido en la base.

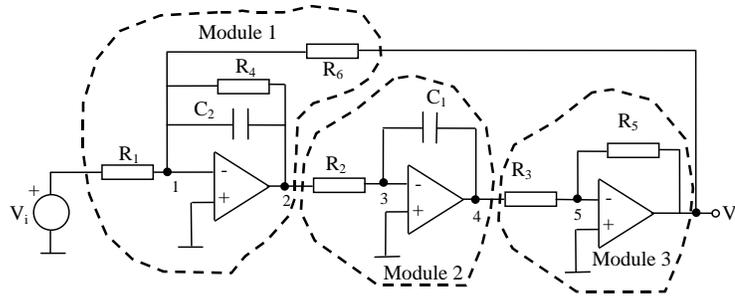


Figura 5: Circuito bajo test

- Situación 4. Igual que la situación 3, pero los extraídos se corresponden con el caso nominal. Este caso será introducido en la base, ya que es preferible una falsa alarma que una falla no detectada
- Situación 5. Alguno de los casos extraídos tiene el campo *Compo* igual al del caso nuevo, y las desviaciones con un error superior al 10%. En este caso debe estudiarse su introducción a la base.
- Situación 6. Es como la situación 5 pero las desviaciones de los casos extraídos con el mismo componente y misma desviación tienen están más cerca al nuevo caso, y por lo tanto tienen un peso superior. En este caso el diagnóstico será correcto y no será necesario introducir el caso a la base.
- Situación 7. Se repiten las condiciones de la situación 6 pero ahora el peso de los casos extraídos correspondientes al mismo componente y desviación que el caso nuevo, es inferior al de los demás. En este caso el diagnóstico será erróneo, por lo que deberá estudiarse la introducción del nuevo caso a la base.
- Situación 8. Está se da cuando ningún caso de los extraídos tiene un campo *Compo* igual al del nuevo caso. En esta situación el diagnóstico también es erróneo y deberá estudiarse la introducción del nuevo caso a la base.

Por otro lado, a medida que los casos van interviniendo en las decisiones, se lleva un control estadístico de si éstos influyen correctamente en el diagnóstico o si por el contrario inducen a error. Con estos datos, cada vez que un caso interviene en un diagnóstico, se actualiza su índice,

haciéndolo aumentar si su contribución es favorable y disminuir en caso contrario. Cuando el valor del índice esté por debajo de cierto umbral, el caso será eliminado de la base por considerar que su contribución es negativa. Éste es el concepto utilizado por el algoritmo IB3 y que se detalla en [13].

Para ver como funciona la metodología propuesta, en los siguientes apartados puede verse el resultado de aplicarla a un circuito real, a modo de ejemplo.

4. Ejemplo en un Circuito

4.1. Circuito a diagnosticar

El circuito seleccionado es el mostrado en la Figura 5. Este circuito es utilizado como benchmark en la bibliografía [1] [7] [9].

Los valores nominales de los componentes del ejemplo son: $R_1=2.7K$, $R_2=1K$, $R_3=10K$, $R_4=1.5K$, $R_5=12K$, $R_6=2.7K$ y $C_1=C_2=10nF$. Un componente es considerado como defectuoso cuando su valor difiera en más de un 10% de su valor nominal.

Debe tenerse en cuenta que este artículo se centra en la detección y localización de fallas paramétricas, permanentes, independientes y simples ([5])

4.2. Selección de las medidas

Las medidas que se van a tomar se basan en el método temporal descrito en [1]. En este caso se sugiere que las medidas que se utilizan son ciertas características de la respuesta del circuito a una

Fault	SP%	$T_d(\mu S)$	$t_r(\mu S)$	V_{est} (V)	Fault	SP%	$T_d(\mu S)$	$t_r(\mu S)$	V_{est} (V)
R1+20	4.4029	15	76	-0.8332	R5+20	4.6189	12	75	-0.9999
R1-20	4.4029	15	76	-1.2498	R5-20	3.9447	20	77	-1.0002
R1+50	4.4029	15	76	-0.6665	R5+50	4.8682	9	76	-1.0002
R1-50	4.4029	15	76	-1.9996	R5-50	2.1315	31	86	-0.9996
R2+20	4.0473	19	77	-1.0001	R6+20	4.0473	19	77	-1.2001
R2-20	4.6614	11	75	-1.0000	R6-20	4.6614	11	75	-0.8000
R2+50	3.3711	24	80	-1.0003	R6+50	3.3711	24	80	-1.5004
R2-50	5.2359	5	75	-0.9999	R6-50	5.2359	5	75	-0.4999
R3+20	4.0473	19	77	-1.0001	C1+20	4.0473	19	77	-1.0001
R3-20	4.6614	11	75	-1.0000	C1-20	4.6614	11	75	-1.0000
R3+50	3.3711	24	80	-1.0003	C1+50	3.3711	24	80	-1.0003
R3-50	5.2359	5	75	-0.9999	C1-50	5.2359	5	75	-0.9999
R4+20	5.7311	12	73	-0.9994	C2+20	5.7085	16	73	-0.9997
R4-20	2.4917	19	80	-1.0000	C2-20	3.0781	15	77	-1.0000
R4+50	6.9145	10	71	-0.9990	C2+50	7.6834	17	72	-1.0005
R4-50	0	29	92	-1.0000	C2-50	1.0031	15	80	-1.0000
					Nom	4.4029	15	76	-0.9998

Cuadro 3: Diccionario clásico para una entrada rampa saturada

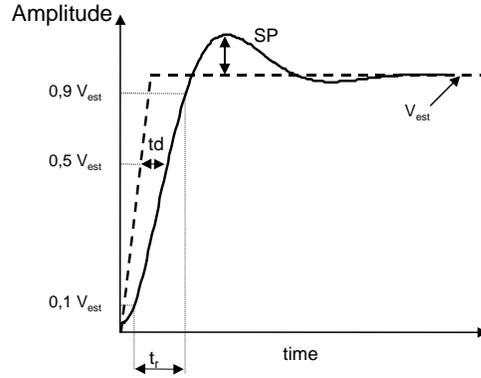


Figura 6: Respuesta a una entrada rampa

entrada rampa saturada. Estas son las siguientes:

- Regimen permanente (V_{est}): Valor final al que se establece la salida.
- Sobrepico (SP). Se define como

$$SP = \frac{V_{m\acute{a}x} - V_{est}}{V_{est}} 100 \quad (3)$$

- Tiempo de subida (T_r). El tiempo que emplea la salida del sistema para ir del 10 % al 90 % de su valor en regimen permanente.
- Tiempo de Retardo (T_d). Intervalo de tiempo que transcurre entre el momento en que la entrada y la salida llegan al 50 % del valor de establecimiento.

En [1] se resalta que escoger una rampa con un tiempo de subida menor, no implica tener posteriormente mejores resultados en el diagnóstico, no proporcionándose ningún método para hallar dicho parámetro.

Las fallas que se proponen inicialmente detectar son desviaciones del $\pm 20\%$ y $\pm 50\%$ del valor nominal de cada uno de los componentes pasivos que forman el circuito.

Se ha tomado como entrada una rampa con un valor de saturación de 1 V y un tiempo de subida de $100\mu s$. Así por ejemplo la falla R2+20 %, cuando todos los demás componentes están exactamente en su valor nominal, proporciona las siguientes medidas a la salida:

$$SP = 4,51\%, td = 19\mu s, tr = 76\mu s \text{ y } V_{est} = -0,99V.$$

La Figura 6, muestra de forma gráfica el significado de dichos parámetros.

4.3. El Diccionario de Fallas

Por lo visto anteriormente se tendrá un diccionario de 33 filas (8 componentes x 4 fallas/componente = 32 fallas más el caso sin falla = 33).

Para todas las fallas consideradas inicialmente, y sin tener en cuenta las tolerancias, el diccionario de fallas clásico tiene 33 filas y 4 columnas tal y como se muestra en la Tabla 3

Observando la Tabla 3, las fallas R2+20, R3+20 y C1+20 tienen unas medidas idénticas, y lo mismo puede decirse de las desviaciones -20% y $\pm 50\%$ de los mismos casos. Entonces los componentes R2, R3 and C1 forman claramente un grupo ambiguo, siendo imposible distinguir entre ellos si se toman las medidas en el nodo V_0 únicamente.

Por lo tanto el diccionario podría reducirse a 25 filas, donde [R2+20 R3+20 C1+20] forman un grupo ambiguo, al igual que [R2+20 R3+20 C1+20], [R2+50 R3+50 C1+50] y [R2-50 R3-50 C1-50].

Para incluir el efecto de las tolerancias se han generado aleatoriamente mediante Monte-Carlo, 501 casos para cada falla considerada. El diccionario entonces pasa a tener $33 \times 501 = 16533$ casos almacenados. Si se tienen en cuenta los grupos ambiguos, éste se reduce a $25 \times 501 = 12525$ casos.

Se ha procedido a reducir el número de casos del diccionario utilizando los métodos DROP4 e IB3 propuestos en [13]. Se han tomado tres vecinos ($k=3$) para proporcionar el diagnóstico final, con una ponderación exponencial en función de la distancia. Para el método IB3 los 12525 casos del diccionario se reducen a 2457 (19.61% de los retenidos originalmente), mientras que con el DROP4 la reducción es a 1112 (8.8% de los retenidos al principio). Se genera una nueva base de 100 casos por falla considerada, generados aleatoriamente (un total de 2500 casos) como conjunto de test para estudiar la eficacia del diccionario. Entonces, usando la distancia euclídea como métrica, se realiza la comparación entre los aciertos en el diagnóstico del diccionario clásico, el diccionario ampliado y los diccionarios reducidos. Estos resultados se dan en la Tabla 4

Observando la Tabla 4, puede verse que DROP4 es el método que da mejores resultados y con una reducción mayor que el IB3.

Cuando las nuevas fallas presentan desviaciones diferentes de las previstas, por ejemplo una desviación del 39.5%, los aciertos disminuyen. Se ha generado una base que contiene 100 fallas de cada componente que puede tener una desviación aleatoria entre el 0% y un $\pm 70\%$. En este caso, se considerará que se acierta si el diagnóstico propuesto es el más cercano que se puede dar. En el caso de una desviación del 39.5%, si la conclusión es 50% se considera correcta, al ser esta la más cercana de las posibles. Con esto, el porcentaje de aciertos para el diccionario clásico baja a un 71.97% y a un 77.77% para el diccionario reducido con DROP4.

Puede observarse que al tratar con fallas no previstas el diccionario de fallas disminuye su efectividad. Para ellos se desarrolla el sistema de Razonamiento Basado en Casos que se propone en el apartado siguiente.

4.4. Sistema de Razonamiento Basado en Casos Propuesto

La estructura del caso para estas medidas concretas será

Case i	SP1	Td1	Tr1	V_{est1}	Class i	Ri	X%
--------	-----	-----	-----	------------	---------	----	----

La jerarquía aplicada al circuito bicuadrático en cuestión se da en la Figura 7. Los módulos están representados en la propia figura.

Como conjunto de partida se toma el diccionario clásico. Es decir, un caso para cada una de las fallas previstas inicialmente de $\pm 20\%$ y de $\pm 50\%$ para cada componente. Por lo tanto la base de casos inicialmente tiene 25 casos, teniendo en cuenta los grupos ambiguos. Para entrenarla se dispone de conjuntos de nuevos casos que están constituidos por 10 casos de cada componente fallando entre 0% y $\pm 70\%$ de forma aleatoria, entorno a su valor nominal. Con el diccionario reducido se ha comprobado que tomando 3 vecinos para hacer el diagnóstico es el número que da mejor resultado.

Fault	Classic dictionary	Spread dictionary	DROP4	IB3
R1+20	84	83	85	77
R1-20	87	94	90	90
R1+50	99	98	99	97
R1-50	100	100	100	100
R2+20,R3+20,C1+20	41	30	41	35
R2-20,R3-20,C1-20	36	35	35	31
R2+50,R3+50,C1+50	79	78	83	72
R2-50,R3-50,C1-50	94	99	96	98
R4+20	85	89	87	78
R4-20	88	84	88	80
R4+50	98	98	98	98
R4-50	100	100	100	100
R5+20	47	33	46	36
R5-20	38	40	38	43
R5+50	82	78	83	75
R5-50	91	93	94	93
R6+20	79	78	79	78
R6-20	86	83	82	74
R6+50	98	99	100	100
R6-50	100	100	100	100
C2+20	82	75	74	76
C2-20	89	90	90	83
C2+50	99	99	99	95
C2-50	100	100	100	100
NOM	69	61	72	57
Average	82.04	80.68	82.36	78.64

Cuadro 4: Comparación de los métodos de reducción

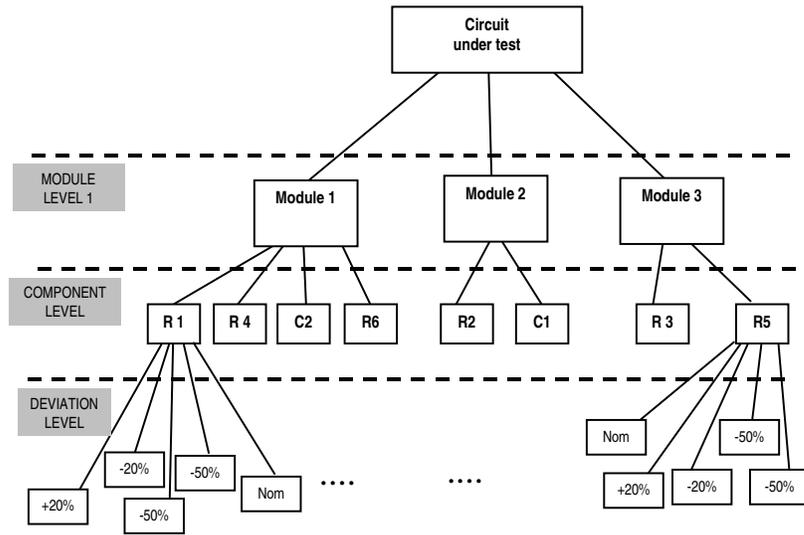


Figura 7: Jerarquía para el filtro bicuadrático

Por ello, cuando el diagnóstico de un nuevo caso sea erróneo, se introducirá en la base sin tener en cuenta otro criterio hasta que el número de casos de su clase no sea como mínimo de 3. A partir de este momento, además entrará en juego el algoritmo DROP4 para decidir la introducción del caso en la base i el criterio IB3 para la eliminación de los que proceda. Si el número de casos de una clase es 3 o inferior, el caso no será eliminado de la base.

Veamos con ejemplos como procede el sistema de Razonamiento Basado en Casos para diagnosticar y mantener la base de casos. Los ejemplos que aquí se muestran se han tomado de considerar como base original el diccionario ya entrenado

Caso 1.

	Class	Compo	Desvi.
Nuevo Caso	1	R1	16.71 %
Vecino1	1	R1	20 %
Vecino2	1	R1	20 %
Vecino3	1	R1	20 %

El diagnóstico proporcionado por el sistema CBR será "La falla es R1+20". En este caso es correcto ya que el componente es acertado y hay un error inferior al 10 % en la desviación. Por lo tanto el contenido de la base es suficiente para diagnosticar este caso y no será considerado para ser introducido.

Caso 2

	Class	Compo	Desvi.
Nuevo Caso	14	R4	-13.07 %
Vecino1	33	Nominal	0 %
Vecino2	30	C2	-20 %
Vecino3	30	C2	-20 %

En esta situación el diagnóstico sería "La falla es C2-20", ya que entre no detectarlo (caso nominal) o predecir una falla es preferible esta última decisión. En este caso, pero, el sistema se equivoca, ya que la falla es realmente R4-13.07%. Por lo tanto debe decidirse si es necesario entrar el nuevo caso en la base. Buscando los asociados que este tendría se obtiene que son los casos 33 y 34, de los cuales el nuevo caso está a una distancia de 0.0748 y 0.0359 respectivamente.

Asoc.	vecinos	distancia
33	34	0.0014
	36	0.0919
	35	0.1081
34	33	0.0014
	36	0.0917
	35	0.1083

Observando los resultados obtenidos en este caso, puede verse que los asociados al nuevo caso, no se ven muy seriamente afectados por su introducción. El nuevo caso ocuparía la posición del segundo vecino, el segundo la del tercero y el tercer vecino se perdería. Con ello, aun así, la distancia del primer vecino es mucho menor que la del nuevo caso, por lo que la suma de pesos final acertará en el diagnóstico. Por lo tanto el nuevo caso será introducido a en la base. Puede verse que este caso corresponde a la situación 5 anteriormente comentada

Aplicando el método al mismo conjunto de test anterior se obtiene un resultado promedio de un 80 % de aciertos. La diferencia respecto al resultado obtenido con la base reducida por el método DROP4 (77.77 %) no es muy grande. Pero debe tenerse en cuenta que la nueva base presenta una dimensión mucho menor.

Conclusiones

Después de comentarse y verse las características principales de los diccionarios de fallas, se han podido constatar sus limitaciones en cuanto al efecto que producen en ellos las tolerancias y la imposibilidad de tratar nuevas situaciones. Por otro lado, se ha mostrado como puede incluirse el efecto de las tolerancias generando múltiples casos mediante la técnica de Monte-Carlo y después reduciendo la base obtenida mediante diversos métodos. Se han probado los métodos IB3 y DROP4, siendo este último el que arroja mejores resultados.

Aunque la base obtenida con DROP4 presenta alguna mejora respecto el diccionario clásico, esta tiene una dimension mucho mayor. Por ello se ha diseñado un sistema CBR que partiendo del diccionario de fallas clásico, consigue construir una base de casos con un porcentaje de aciertos parecido, pero con una dimensión más reducida.

Aunque el método propuesto proporciona algunas mejoras, deben tenerse en cuenta varios factores a estudiar en un futuro. Uno de ellos sería si es mejor partir del diccionario clásico propuesto, o bien sería mejor partir de una base de casos generada aleatoriamente con tolerancias comprendidas entre 0 % y 70 %. También debe estudiarse en detalle el índice a partir del cual se decidirá que un caso debe ser eliminado de la base y si éste, depende de la topología del circuito a analizar.

Por otro lado, debería probarse el método en circuitos mayores, y contrastarlo con el diccionario de fallas clásico, para comprobar si sigue comportando mejor el método CBR propuesto, sin implicar un gran esfuerzo en su elaboración.

Referencias

- [1] A. Balivada, J. Chen, and J.A. Abraham. Analog testing with time response parameters. *IEEE Design and Test of computers*, pages 18–25, Summer 1996.
- [2] J.W. Bandler and A.E. Salama. Fault diagnosis of analog circuits. *Proceedings of the IEEE*, 73(8):1279–1325, August 1985.
- [3] R.R. Boyd. *Tolerance Analysis of Electronic Circuits Using Matlab*. Electronics Engineering. CRC Press, 1999. ISBN: 0-8493-2276-6.
- [4] Y. Chen. Experiment on fault location in large-scale analog circuits. *IEEE Transactions on Instrumentation and Measurement*, 42(1):30–34, February 1993.
- [5] P. Duhamel and J.C. Rault. Automatic test generation techniques for analog circuits and systems: A review. *IEEE Transactions on Circuits and Systems*, Cas-26(7):411–440, July 1979.
- [6] W. G. Fenton, T. M. McGinnity, and L. P. Maguire. Fault diagnosis of electronic systems using intelligent techniques: A review. *IEEE Transactions on Systems, Man and Cybernetics. Part C: Applications and Reviews*, 31(3):269–281, August 2001.
- [7] B. Kaminska, K. Arabi, P. Goteti, J.L. Huertas, B. Kim, A. Rueda, and M. Soma. Analog and mixedsignal benchmark circuits. first release. *IEEE MixedSignal Testing Technical Activity Committee*.
- [8] A. Sangiovanni-Vicentelli, L. Chen, and L.O. Chua. An efficient heuristic cluster algorithm for tearing large-scale networks. *IEEE Transactions on Circuits and Systems*, cas-24(12):709–717, December 1977.
- [9] M. Soma. A desing for test methodology for active analog filters. pages 183–192. IEEE, 1990.
- [10] J. Starzik, J. Pang, S. Manetti, M. Piccirilli, and G. Fedi. Finding ambiguity groups in low testability analog circuits. *IEEE Transactions on Circuits and Systems. Fundamental Theory and Applications*, 47(8):1125–1137, 2000.
- [11] G.N. Stenbakken, T.M. Souders, and G.W. Stewart. Ambiguity groups and testability. *IEEE Transactions on Instrumentation and Measurement*, 38(5):941–947, october 1989.
- [12] R. Voorakaranam, S. Chakrabarti, J. Hou, A. Gomes, S. Cherubal, and A. Chatterjee. Hierarchical specification-driven analog fault modeling for efficient fault simulation and diagnosis. *International Test Conference*, pages 903–912, 1997.
- [13] D. Wilson and T. Martinez. Reduction techniques for instance-based learning algorithms. *Machine Learning*, 38(3):257–286, 2000.