

Análisis de Diagnosticabilidad y Localización de Sensores en un Pozo de Extracción de Petróleo por Inyección de Gas

Leal R.*; Camargo E.*; Aguilar J.**; Ríos A.***; Travé-Massuyès L.****

*Departamento de Automatización, PDVSA, Cabimas, Venezuela
e-mail: lealrs@pdvsa.com, camargoea@pdvsa.com

** CEMISID, Universidad de Los Andes, Mérida, Venezuela
Investigador Prometeo, Universidad Técnica Particular de Loja, Ecuador
e-mail: aguilar@ula.ve

*** CEMISID, Universidad de Los Andes, Mérida, Venezuela
e-mail: ilich@ula.ve

**** DISCO Department, LAAS-CNRS, Toulouse, France
e-mail: louise@laas.fr

Resumen: En este trabajo se propone el uso de un enfoque basado en algoritmos genéticos para obtener relaciones de redundancia de análisis para estudiar la propiedad de diagnosticabilidad en procesos petroleros, y si esta no cumple, nuestro enfoque permite estudiar el problema de localización de sensores con el fin de cumplir con ella. Las relaciones de redundancia se basan en un análisis estructural sobre un grafo bipartito. El análisis de fallas es estudiado usando una función multi-objetivo en varios algoritmos genéticos que describen los diferentes criterios que se deben tratar con el fin de llegar a la propiedad diagnosticabilidad en el sistema. Además, nuestro enfoque permite estudiar el problema de localización de sensores en los sistemas que no cumplen las propiedades de detectabilidad o aislabilidad, usando otro algoritmo genético.

Palabras clave: Algoritmo Genético, Diagnosticabilidad, Análisis Estructural, Localización de Sensores, Extracción de Petróleo por Inyección de Gas.

Abstract: In this work we propose to use an approach based on genetic algorithms to obtain analytical redundancy relations to study the diagnosability property on oil processes, and if this not fulfill, our approach allows studying the sensor placement problem in order to fulfill it. The redundancy relations are based on a structural analysis over a bipartite graph. The faults analysis is studied using a multi-objective fitness function in several genetic algorithms which describe the different constraints to be covered in order to reach the diagnosability property on the system. Additionally, our approach allows studying the sensors placement problem on systems that do not fulfill the detectability or isolability properties, using another genetic algorithm.

Keywords: Genetic Algorithm, Diagnosability, Structural Analysis, Location of Sensors, Oil Production Gas Injection.

1. INTRODUCCIÓN

En las últimas décadas, el crecimiento de la demanda en la confiabilidad, seguridad y operación de los procesos industriales, es cada vez mayor. Se requiere de grandes inversiones en recursos físicos y humanos que ayuden a minimizar las fallas y las consecuencias que estas puedan ocasionar en los sistemas de automatización y a disminuir las pérdidas de oportunidad. Por este motivo, existe la necesidad de realizar análisis de diagnóstico completo en los procesos de producción, estudiando sus componentes, realizando modelos estructurales de los mismos, determinando la propiedad de diagnosticabilidad en ellos, entre otras cosas. Básicamente, una falla significa cualquier cambio en el comportamiento de cualquiera de los componentes del

sistema, por lo que ya no puede cumplir la función para la que fue diseñado [1]. Un sistema de diagnóstico consiste en la detección y el aislamiento de un conjunto de fallos. Un diagnóstico determina un comportamiento específico del sistema, dado el modelo y un conjunto de variables medidas desde el sistema.

El diagnóstico de fallas se puede abordar desde múltiples metodologías, algunas provenientes de la comunidad de control conocidas, como "Fault Diagnosis and Isolation" (FDI), la cual basa sus aproximaciones en la teoría de control y en modelos matemáticos de los procesos; otras basadas en la inteligencia artificial conocidas como "Diagnosis Community" (DX), que se fundamentan en modelos de la inteligencia artificial. Cada comunidad ha desarrollado su

propia terminología, herramientas, técnicas y enfoques para resolver los problemas de diagnóstico [2]. Nuestro enfoque es una mezcla entre las dos teorías, mediante un modelo matemático y el uso de técnicas inteligentes para resolver los problemas planteados [3, 4]

Por otro lado, con la ayuda de esas técnicas inteligentes, nosotros también proponemos un modelo que permite determinar la localización de sensores en el caso de que los procesos de producción no cumplan con las propiedades de detección y aislamiento de fallos. Esto se debe a que no se cuentan con suficientes variables observadas (medidas) en el sistema.

La técnica de inteligencia artificial usada son los Algoritmos Genéticos (AG), unos AG para determinar la propiedad de diagnosticabilidad en el proceso, y otro AG para determinar la localización de los sensores en aquellas variables no observadas, que permitan incrementar la redundancia del proceso, y así poder cumplir con la propiedad de diagnosticabilidad en él. Existen otros trabajos que realizan parte de estos estudios [5, 6, 7], pero nunca un análisis de todo el contexto (propiedad de diagnosticabilidad y localización de sensores).

En específico, este trabajo propone un enfoque para el análisis de diagnosticabilidad basado en un híbrido entre los modelos estructurales y AG. Para ello, se usan modelos estructurales de procesos continuos, y AG para encontrar el conjunto de las relaciones de redundancia analítica. Además, proponemos usarlo en un proceso real continuo de extracción de petróleo por inyección de gas. En este estudio se considera uno de los métodos más utilizados para la extracción artificial de petróleo, que se basa en pozos con inyección de gas [15].

Este artículo está organizado de la siguiente manera. La siguiente sección presenta algunos fundamentos básicos acerca de diagnosticabilidad basado en análisis de residuos, modelos estructurales y localización de sensores. La sección 3 presenta nuestro enfoque basado en AG, y finalmente, la última sección presenta el experimento en un proceso de producción real, como es la extracción de petróleo por inyección de gas

2. ANÁLISIS DE DIAGNOSTICABILIDAD BASADOS EN MODELOS ESTRUCTURALES

El concepto fundamental en nuestro trabajo es la diagnosticabilidad. Comenzamos por su definición, seguido de otros conceptos que nos ayudan a explicar nuestro enfoque.

Definición 1: Diagnosticabilidad. Dado un determinado sistema, tiene esta propiedad si es capaz de detectar y aislar todas sus fallas consideradas [4, 8].

Una metodología para el análisis de diagnosticabilidad es el enfoque basado en modelos (por sus siglas en inglés, MBD).

El principio básico consiste en comparar el comportamiento actual del sistema con un modelo matemático de referencia, el cual representa el comportamiento normal. Al momento que ocurre una falla en el sistema, la comparación de ambos produce un resultado diferente de cero, este resultado se conoce como residuo [9].

Una forma de generación de residuos es por medio de las relaciones de redundancia analítica [2]. Para ello se utiliza un subconjunto de las ecuaciones que caracterizan el sistema, las cuales estarán compuestas únicamente por variables medidas o conocidas.

Dicho subconjunto de ecuaciones son usadas en el momento que existe una perturbación (falla), tal que la evaluación final de los residuos de las que ellas forman parte, es en esos casos diferente al valor nulo. La generación de un residuo estará compuesta por una secuencia finita de cálculos que termina con la evaluación de una ecuación analíticamente redundante (ARR por sus siglas en inglés). En la literatura existen algoritmos para calcular ARR como los propuestos en [8, 10], los cuales propone esquemas de análisis estructurales de procesos continuos.

2.1 Análisis estructural para la diagnosticabilidad

El análisis estructural es un conjunto de herramientas para explorar las propiedades fundamentales de un sistema, ya sea en la forma de una matriz de incidencia o en un grafo bipartito [3]. En nuestro trabajo, el análisis estructural del sistema se utiliza para la detección y el aislamiento fallas, siguiendo los métodos utilizados por la comunidad FDI [1].

Trabajos previos han modelado la diagnosticabilidad en procesos continuos basados en la comunidad FDI [9, 11]. Un modelo estructural es una representación de un sistema en el que sólo utiliza las conexiones entre las variables y ecuaciones, y no las ecuaciones analíticas reales [12]. El modelo estructural contiene sólo la información de qué variable pertenece a cada ecuación, independientemente del valor de los parámetros y la forma detallada de la expresión matemática [8].

Considere un modelo $M(X, Y, E, F)$, donde E es un conjunto de ecuaciones, $E = \{e_1, \dots, e_m\}$, X es un conjunto de variables desconocidas, $Y = \{x_1, \dots, x_n\}$, Y es un conjunto de variables conocidas, $Y = \{y_1, \dots, y_p, u_1, \dots, u_g\}$, y F es un conjunto de fallas en el sistema $F = \{f_1, \dots, f_o\}$. Adicionalmente, $Z = X \cup Y$. En el caso donde existan variables diferenciales o integrales en el modelo, es necesario un cuarto conjunto, $D = \{\dot{x}_1, \dots, \dot{x}_n\}$, el cual contenga estas variables.

Definición 2: ARR para $M(X, Y, E, F)$. En un modelo la ecuación $r_i: h(y, \dot{y}, \ddot{y}, \dots) = 0$ es un ARR para $M(E, X, Z, F)$, si cada y en la ecuación r_i es consistente con M [2].

Estas relaciones pueden ser derivadas sólo si el modelo tiene más ecuaciones que incógnitas, es decir, si el sistema es estructuralmente sobre-determinado (SO structurally over-determined) [2, 13].

Cada residuo deben ser evaluados con el fin de decidir si se puede utilizar o no. La evaluación consiste si pueden ser usados para detectar y aislar fallas, para lo cual se usa un vector de firmas de cada falla ($S = \{S_1, \dots, S_n\}$), que es un conjunto de vectores de fallas que debe permitir detectarlas y aislarlas, donde cada S_i es el conjunto de residuos que se activan con esa falla.

Dado un conjunto de vectores ($S = \{S_1, \dots, S_n\}$), y un conjunto de fallas $F = \{f_1, \dots, f_o\}$, se puede definir la matriz de firma de fallas para codificar el efecto de cada falla en cada residuo [12].

Definición 3: Matriz de firma de fallas de M. Es una tabla obtenida por la concatenación de todas las posibles firmas de fallas. Cada fila corresponde a una ARR y cada columna a cada falla. Un "1" en la posición (ij), indica que la falla j es detectada por el ARR_i [1].

Definición 4: Detectabilidad de $M(E, X, Z, F)$. Una falla F_o , donde $o = 1, \dots, n$, la cual pertenece a F en M, es detectable si existe al menos un residuo diferente a cero cuando la falla ocurre, es decir, $r_i \neq 0$.

Definición 5: Aislabilidad de $M(E, X, Y, F)$. Es cuando todas las firmas de las fallas consideradas en el sistema en la matriz de firma de fallas son diferentes, y se puede discernir una de las otras [14]. Por lo tanto, $S(f_o) \neq S(f_t), \forall o, t \in \{1, \dots, n\}, o \neq t$.

Ejemplo 1: Considere un sistema que contenga un conjunto de residuos $\{Arr_1, Arr_2, Arr_3, Arr_4, Arr_5, Arr_6\}$, construido para detectar y aislar cinco fallas $\{f_1, f_2, f_3, f_4, f_5\}$. La matriz de firma de fallas muestra la sensibilidad de cada falla en cada ARR, incluyendo el Sistema bajo un comportamiento normal (N). Por ejemplo, Arr_1 es sensible a las fallas f_1, f_2 , y f_5 , y así consecutivamente, como se muestra en la Tabla 1.

Tabla 1. Matriz de firma de fallas

Arr	Conjunto de Fallas					
	N	f_1	f_2	f_3	f_4	f_5
Arr_1	0	1	1	0	0	1
Arr_2	0	1	1	0	1	0
Arr_3	0	0	0	0	1	1
Arr_4	0	0	0	1	0	1

Nosotros adoptamos el método de los mínimos conjuntos sobre-determinado (MSO, minimal structurally over determined) para calcular ARRs, como en [10], donde proponen un algoritmo que identifica los MSO, posibilitando la construcción de ARR más eficientes. Cada ARR corresponde a un MSO.

En [8] introducen la noción de Ecuaciones de Soporte (TES, Test Equation Support), que son conjuntos de ecuaciones que expresan la redundancia específica en un conjunto de fallas consideradas. Las cantidades correspondientes que expresan las mínimas redundancias se denotan mínima TES (MTES), y el conjunto de MTES pueden ser vistos como un subconjunto de los MSO. Dado a que existe una relación uno a uno entre

MTES y ARR, en este trabajo nos enfocaremos en los MTES para generar residuos basados en aproximaciones FDI, en orden de estudiar la propiedad de diagnosticabilidad en un sistema [11].

2.2 Localización de Sensores

La eficiencia de un sistema de diagnóstico depende de la información que se puede recuperar del proceso diagnosticado. Obviamente, si la información es insuficiente, el sistema de diagnóstico no es capaz de realizar su tarea. Por lo tanto, la eficiencia de un sistema de sensor se puede medir por el grado de diagnosticabilidad que proporciona. También, un mismo nivel de diagnosticabilidad puede obtenerse para diferentes configuraciones de sensores.

Continuando con el método de análisis estructural, el problema de localización de sensores se puede resolver usando una representación estructural del modelo, tal como se utiliza en [16]. La representación estructural de un conjunto E de ecuaciones con X incógnitas es un gráfico bipartito, tal como se ha descrito a lo largo de la investigación (ver ejemplo 1).

Sea E y F que indican un conjunto de ecuaciones y un conjunto de fallas individuales, respectivamente. Sin pérdida de generalidad, es posible suponer que un fallo sólo puede violar una ecuación. Si una señal de fallo f aparece en más de una ecuación, simplemente la reemplazamos en las ecuaciones con una variable nueva f_x y añadimos la ecuación $f = f_x$, que será entonces la única ecuación violada por la falla. Sea $e_f \in E$ la ecuación que podría ser violada por una falla $f_x \in F$.

Una falla f es detectable si existe una observación que sea consistente con el sistema en modo de falla f e inconsistente con el sistema en modo normal. Esto significa que un fallo detectable puede violar una ecuación controlable en el modelo. Una ecuación, es controlable si está contenido en la parte estructuralmente sobre-determinado de M [17].

En un modelo estructural del sistema donde hay fallos que no pueden ser detectados, podemos añadir sensores para aumentar la observabilidad del sistema. Suponemos que un sensor y_{p+1} se ha añadido para la medición de x_n , y se incluyen en el modelo como $e_{m+1}: y_{p+1} = x_n$.

De esta manera, se aspira que las fallas no detectadas puedan ser detectables, a través de un residuo que sea capaz de detectarla porque la redundancia del sistema se incrementa. Una suposición general del enfoque es que el modelo no contiene ninguna parte indeterminada.

Este es el enfoque que será usado por nuestro enfoque para incorporar nuevos sensores en el sistema bajo estudio.

3. ENFOQUE INTELIGENTE PARA ANÁLISIS DE DIAGNOSTICABILIDAD Y LOCALIZACIÓN DE SENSORES

Varios investigadores han aplicado el enfoque de AG con éxito para resolver el problema de localización de sensores [5, 6, 7]. En este trabajo proponemos usar las características de optimización de los AG para resolver el problema de diagnosticabilidad, y a partir de allí, si es necesario, resolver también el de localización de sensores. En previos trabajos hemos propuesto resolver el problema de diagnosticabilidad usando un modelo estructural (basado en análisis de redundancia) y AGs [11]. En este trabajo proponemos un modelo híbrido, donde combinamos varios AGs y el modelo estructural, para resolver ambos problemas: el de diagnosticabilidad y el de localización de sensores, en procesos del ámbito petrolero. En específico, usamos dos AGs con el modelo estructural para analizar la propiedad diagnosticabilidad, y si no se cumple, otro AG que se apoya en los otros AGs para resolver el problema de localización de sensores, que permita hacer al proceso petrolero bajo estudio diagnosticable.

3.1. Modelo algorítmico de base para resolver los problemas de Diagnosticabilidad y de Localización de Sensores

Para cumplir con esto se han desarrollado tres algoritmos que durante sus ejecuciones invocan varios AGs. El primer algoritmo (llamado *Principal*) simplemente llama a los otros dos algoritmos (*Detección* y *Localización*). El objetivo del algoritmo de detección es comprobar si se detectan todas las fallas en estudio (ver las líneas 1-2, del algoritmo *Principal*).

En la línea 2, si la propiedad de detectabilidad no se satisface ($MTES(f) = \emptyset \forall f = 1, \#fallas$), o la propiedad de aislabilidad no es satisfecha ($ind_opt \neq \emptyset$, donde $ind_opt \neq \emptyset$ es una variable de control del segundo algoritmo), el tercer algoritmo (*Localización*) se invoca en la línea 3, con el fin de asignar nuevos sensores a las variables desconocidas para aumentar así la redundancia del sistema.

Algoritmo 1 *Principal*(G)

```

1  MTES=Detección(G);
   % Determina la detectabilidad y la aislabilidad % del
   % sistema
2  si  $MTES(f) = \emptyset \forall f = 1, \#fallas$  ó  $ind\_opt \neq \emptyset$ 
3  Localización (G);
   % Incremente la redundancia en el sistema
   % para cumplir con la propiedad de
   % diagnosticabilidad, localizando sensores
   % en variables desconocidas
4  fin
5  fin

```

El segundo algoritmo (llamado *Detección*) busca los emparejamientos en el modelo (un emparejamiento representa una redundancia), basado en la teoría de MTES. El algoritmo encuentra las posibles conexiones entre las variables y las ecuaciones, con el fin de eliminar las variables

no observadas, para cumplir con los requisitos específicos de una ARR. En particular, se encuentran poblaciones de MTES invocando el primer AG, llamado AG1. Este algoritmo también determina la detectabilidad y la aislabilidad (para el último caso utiliza un segundo AG, llamado AG2, que utiliza los mejores MTES generados por AG1 para construir la matriz de firma de falla).

Con el fin de verificar la aislabilidad del sistema, se llama a un segundo AG (línea 10, el algoritmo 3.2). Se invoca a AG2 con la matriz de firma de fallas como parámetro. Si el sistema es aislable, el mejor individuo de AG2 cumple la propiedad de aislabilidad (el sistema tiene la propiedad diagnosticabilidad). Si esta propiedad no se puede alcanzar, entonces se define $ind_opt \neq \emptyset$, para invocar en el algoritmo principal al tercer algoritmo (algoritmo *Localización*), que se utiliza cuando hay fallas no detectadas o no aisladas.

Algoritmo 2 *Detección* (G)

```

1  repita para  $k = 1, k = \#fallas \mid f \in F$ 
2   $MTES(f_k) = AG1(f_k, G)$ ;
   % Determina los MTES's para cada falla
3  fin
4  si al menos  $\exists MTES(f) \neq \emptyset \mid \forall f \in F$ 
5  repita para  $i = 1, i = \#fallas \mid f \in F$ 
6   $S(i) = det(i, MTES(f_i))$ ;
   % Determina el conjunto de fallas cubiertas
   % por los MTES de cada falla
7  fin
8   $siverificacion(S) = verdad$ 
   % Determina la detectabilidad en el modelo
9   $S_m = iso(f_i, S)$ ;
   % Construye la matriz de firma de fallas
10  $ind\_opt = AG2(S_m)$ 
   % Determina la aislabilidad de las fallas
   % ( $ind\_opt = 0$ )
11 fin
12 fin

```

El algoritmo de *Localización* utiliza otro AG, llamado AG3, para encontrar un conjunto de posibles localizaciones de sensores (cada individuo es una localización posible de un sensor) en base a las fallas a detectar (variables desconocidas), con el costo de implantación de dichos sensores, con el fin de elegir el más bajo valor para ahorrar dinero en el momento de la implantación (ver línea 6).

El nuevo grafo bipartito propuesto por AG3, que representa el modelo estructural del sistema es G' , posteriormente es evaluado en el algoritmo *Detección*, como se muestra en la línea 2. Según las líneas 3 y 5, si AG3 converge debido a la cantidad de generación, entonces significa que no hay individuos (G') que cumplen con la propiedad de diagnosticabilidad (véase la siguiente sección, donde se explica AG3), es decir, es una propiedad que no puede alcanzar el sistema real (véase la línea 4). Si no, el individuo G' es el nuevo grafo bipartito del sistema, y la localización de los nuevos sensores tiene un costo de acuerdo a la función de aptitud del individuo (línea 6).

Algoritmo 3 **Localización (G)**

```
% Propone sensores en variables desconocidas
1  cost = AG3(G, G', cost)
2  Detección (G')
3  si AG3 converge debido a Nb_generaciones entonces
4    imprimir "El Sistema no cumple con la Diagnosticabilidad"
5  sino
6    imprimir "El costo del sensor es" cost, "y la nueva configuración es" G'
fin
```

3.2. Algoritmos genéticos usados para el análisis de diagnosticabilidad

AG es una de las más poderosas herramientas heurísticas para resolver problemas de optimización, basado en física selección y el proceso de la evolución biológica [5]. Un AG modifica repetidamente una población de soluciones individuales.

En cada paso, el GA selecciona individuos de la población actual para ser padres (a veces solo a los mejores), y los utiliza para producirla próxima generación. A lo largo de sucesivas generaciones, la población "evoluciona" hacia una solución óptima. En la siguiente sección vamos a describir los diferentes AGs definidos para nuestro problema.

3.3. AG1

Dado un modelo M , su propiedad de diagnosticabilidad puede ser analizada utilizando un modelo estructural del sistema como grafo bipartito G , y es representado por la unión del conjunto de las fallas y el conjunto de las variables desconocidas $G = G_X \cup G_F$. De G_F se extrae la falla a estudiar f_k , y AG1 genera una población de individuos con combinaciones aleatorias de arcos activos en el modelo con esa falla (un arco activo es cuando se selecciona la relación entre una ecuación y una variable, para componer los posibles MTES definidos por este individuo de la falla f_k , ver más abajo para más detalles). Es decir, cada individuo va a definir un posible MTES. La función objetivo permite determinar la propiedad de detectabilidad de los posibles MTES, y los mejores individuos serán la entrada del AG2, con el fin de estudiar la propiedad de aislabilidad.

Un individuo C en AG1 (cromosoma) es representado por una cadena de bits, donde cada bit (gen) representa los arcos que conectan los vértices en el grafo bipartito, tal que a_{ij} es el arco entre los vértices i (representa una variable desconocida x o la falla f en el gráfico bipartito) y j (representa la ecuación e en el grafo bipartito).

Si se selecciona un arco $a_{ij} = 1$ (lo llamaremos arco activo), de lo contrario $a_{ij} = 0$ (representa que este arco no existe en el grafo bipartito o no es un arco activo). Dado el grafo bipartito en la Fig. 1, el tamaño del individuo es la multiplicación de la cantidad de variables (X, F) por la cantidad de ecuaciones

(E), una posible representación del individuo (cromosoma) sería como la mostrada en la Fig. 1.

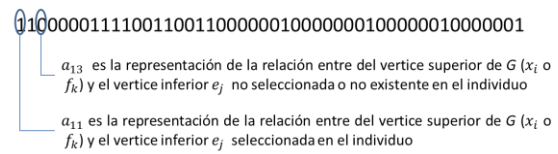


Figura 1. Estructura del individuo del AG1

Por lo tanto, para cada posible MTES para la falla f_k existe una secuencia única de bits. El MTES (individuo) se evalúa utilizando la función objetivo que incluye todos los criterios de diseño importantes de un MTES.

Este proceso evolutivo se detiene cuando se alcanza un número de generaciones, o cuando una persona alcanza un valor de $FF_1 = 0$ (este MTES representa un residuo de la falla f_k). En general, AG1 es:

Algoritmo Genético 1 ($f_k G$)

```
1  Inicio AG1
2    Pop(C)=genera_pop(G);
   % Genera la población inicial
3    Evaluación( $FF_1$ ,pop);
   % Calcula el valor de la función objetivo en
   % cada individuo
4    repita mientras no  $FF_1 = \emptyset$  o Nb_generaciones
   hacer
5      inicio
   % Inicia ciclo reproductivo
6      Reproducción (pop)
7      Evaluación ( $FF_1$ , pop)
8      Selección (pop)
9      fin
10   fin
fin
```

FF_1 es una función objetivo multi-objetivo, con el fin de encontrar los MTES para la falla f . El enfoque es de suma ponderada, el cual es utilizado para agregar todos los criterios de optimización (ver ecuación 1). El valor numérico determina la calidad de cada solución posible (MTES).

$$FF_1 = \text{Min} \left\{ \sum_{n=1}^3 w_n P_n \right\} \quad (1)$$

Donde, w_n es el peso correspondiente y P_n es el criterio a ser optimizado. Los diferentes criterios a ser optimizados se encuentran en la Tabla 2.

Tabla 2. Criterio de Optimización

Objetivo	Optimización
P_1	Número de fallas
P_2	Falla en estudio
P_3	Cardinalidad del sistema

P_1 asegura que sólo se estudia una falla f_k en un momento específico (o la menor cantidad de fallas, ver ecuación 2): Se

describe por la suma de los arcos activos desde las fallas que se encuentran en la parte G_F del individuo diferentes a f_k , (la falla en estudio). Si esta cantidad es diferente de cero, entonces se penaliza la ecuación con un peso w_1 .

$$P_1 = \sum_{\substack{f_i \in F \\ f_i \neq f_k}} \sum_{j=1}^m a_{kj} \quad (2)$$

Donde,

a_{kj} , son los arcos activos entre las fallas y las ecuaciones

f_k , falla en estudio

m , número de ecuaciones

P_2 asegura que la falla en estudio f_k es considerada por el individuo (ver ecuación 3): se hará una suma de los arcos activos de la falla estudiada presentes en el individuo. Normalmente debe existir sólo un arco activo de la falla estudiada para garantizar un TES, de lo contrario este valor es distinto de cero, y el individuo debe ser penalizado con un peso w_2

$$P_2 = \left| \sum_{j=1}^m a_{f_k j} - 1 \right| \quad (3)$$

donde

Donde, $a_{f_k j}$ son los arcos activos desde la falla en estudio

P_3 verifica que el grado de redundancia del modelo sea igual a uno, es decir, hay una ecuación más que variables desconocidas (ver ecuación 4). Para ello, comprueba que la diferencia de la cardinalidad de las ecuaciones j menos la cardinalidad de las variables x del grafo G que están enlazadas por los arcos activos en el individuo sea uno, para garantizar la propiedad que caracteriza a un MTES. El valor absoluto de esta función debe arrojar como resultado el valor de cero, de lo contrario el individuo debe ser penalizado con un peso w_3

$$P_3 = |(Card(J^*) - Card(I^*)) - 1|_{\substack{I^* = \{i | \exists a_{ij} = 1 \forall j = 1, m\} \\ J^* = \{j | \exists a_{ij} = 1 \forall i = 1, n\}}} \quad (4)$$

Donde,

I^* son los arcos activos de las variables desconocidas que pertenecen a J^*

J^* son los arcos activos provenientes de las ecuaciones

Una vez descritos los criterios de diseño, formalizamos nuestra función objetivo (ver ecuación 5).

$$FF_1 = w_1 P_1 + w_2 P_2 + w_3 P_3 \quad (5)$$

En esta función objetivo la importancia de cada criterio se define por el peso w_n ($n = 1, 2, 3$). Los valores se determinan de acuerdo con los requisitos de diseño y experimentación (ver Tabla 3). Los coeficientes de los pesos son determinados por la importancia de los resultados deseados. El criterio más importante es la redundancia del sistema, tal que el objetivo principal es obtener una redundancia igual a uno. El siguiente criterio más importante es asegurar que la falla en estudio

aparezca en los mejores individuos, y el último criterio es la cantidad de fallas en el individuo, para asegurar que se estudie a una sola.

Particularmente, en AG1 los individuos élite (los individuos con los mejores valores de aptitud) de cada generación se eligen, con el fin de asegurar que los mejores individuos actuales siempre sobreviven en la próxima generación. Los operadores genéticos usados son el cruce básico y la mutación multigen [19]. Se usaron esos operadores genéticos básico, por ser cada cromosoma una cadena de bits (para ese tipo de cromosoma son muy fáciles de implementar y muy eficientes [19]). El cruce básico consiste en seleccionar de los individuos élites dos individuos como padres, para luego seleccionar un mismo punto al azar en cada cadena. Luego se generan dos nuevos individuos, tal que la parte anterior del punto es copiada de uno de los padres y la posterior del otro.

En cuanto al operador de mutación multigen, consiste en seleccionar un individuo del conjunto de individuos élites, seleccionar de ese individuo un grupo de bits (genes), y cambiarlos aleatoriamente. Al aplicar este operador se requiere un segundo paso, ya que los nuevos individuos no pueden generar arcos activos que no estén contenidos entre los posibles arcos del grafo bipartito original. En ese sentido, el operador de mutación propone cambios aleatorios para generar nuevos individuos, pero se corrigen los nuevos individuos para garantizar que los nuevos arcos activos correspondan a arcos existentes en el grafo bipartito original.

Siguiendo las recomendaciones de [19], el operador genético de cruce es usando con una probabilidad de 0.9, y el de mutación con una probabilidad de 0.3. La población inicial de AG1 es de 50 individuos, estos individuos son submodelos estructurales del modelo original, lo cual es suficiente para buscar MTES. Los individuos son evaluados usando la función objetivo FF_1 , y cada generación AG1 selecciona los mejores individuos como padres, para ejecutar sobre ellos los operadores genéticos de cruce y de mutación. El criterio de parada es: si el individuo con mejor valor de aptitud no mejora en 50 iteraciones (generaciones) el AG1 se detiene. Para definir el tiempo de ejecución teórico de AG1 nos basamos en la complejidad computacional, que se refiere a la cantidad de recursos computacionales (en nuestro caso solo consideramos tiempo de CPU) que requiere un algoritmo para su ejecución.

A eso llamaremos el orden de complejidad (O). Para AG1, O viene definido por el O (tiempo requerido) de un AG clásico básico y el O para calcular FF_1 . $O(AG)$ es igual al número de iteraciones por el número de individuos, y $O(FF_1)$ es igual a m por el número de fallas. $O(AG1)$ viene determinado por el máximo valor de esas cuatro variables (número de iteraciones, número de individuos, m o número de fallas) para la instanciación que se esté haciendo en un momento dado, por lo cual depende de la complejidad del problema (m o número de fallas) o de un AG clásico. Los detalles de este algoritmo están en [11], su prueba formal es también hecha en [11].

Tabla 3. Pesos de los coeficientes

Objetivo	Criterio de optimización	Valor
w ₁	Número de Fallas	100
w ₂	Falla en estudio	1000
w ₃	Redundancia del sistema	10000

3.4. AG2

La población resultante de AG1 implica tal vez un gran número de MTES. Dado que existe una relación uno a uno entre los ARR y MTESs, y si la propiedad detectabilidad se alcanza en el sistema, la línea 10 del algoritmo de *Detección* construye la matriz de firmas de fallas y llama AG2, con el fin de determinar un conjunto de ARRs que garantiza la propiedad de aislabilidad del sistema.

El individuo (cromosoma) en AG2 también está representado por una cadena de bits. En este caso, cada bit (gen) representa cada MTES (ARR) encontrado por AG1 para todas las fallas.

AG2 define aleatoriamente individuos para ser posteriormente evaluadas por la función objetivo, donde un bit (gen) con el valor de "1" significa que un MTES se elige para conformar la matriz de firmas de fallas, de otro modo su valor es "0". AG2 estudia la sensibilidad de las fallas para los MTES encontrados por AG1 seleccionados en un individuo (cromosoma) dado, y observa si en este individuo se alcanza la propiedad de aislabilidad.

La función objetivo de AG2 asegura que su valor es cero cuando el individuo tiene la propiedad aislabilidad. De lo contrario, este individuo no puede aislar las fallas (no cumple la propiedad de aislabilidad). Hay dos condiciones en la función objetivo (ver ecuación 6).

$$FF_2 = \sum_{i=1}^{\#f} \sum_{k=i+1}^{\#f} O_{ij}^1 + \sum_{i=1}^{\#f} O_i^2 \quad (6)$$

Donde

$$O_{ik}^1 = \begin{cases} 0 & \text{if } \forall j = 1, \#ARRs \exists V_{ij} \neq V_{kj} \\ 10 & \text{de lo contrario} \end{cases} \quad (7)$$

$$O_i^2 = \begin{cases} 0 & \sum_{j=1}^{\#ARRs} V_{ik} \neq \emptyset \\ 10000 & \text{de lo contrario} \end{cases} \quad (8)$$

La primera condición O^1 asegura que el vector de firma de la falla f_i en la matriz construida por el individuo es diferente que el vector de firma de la falla f_j , de lo contrario se penaliza con un valor bajo (ver ecuación 7).

La interpretación es que f_i no puede ser aislado de f_j . O^2 es la condición que garantiza que la suma de cada vector en la matriz de firmas de fallas es distinto de cero, de lo contrario no puede distinguirse entre el comportamiento de un sistema con falla de un comportamiento normal, penalizando el individuo con un valor alto (ver ecuación 8).

Algoritmo Genético 2 (S_m)

```

1  Inicio GA2
2  pop=genera_pop( $S_m$ );
   % Genera la población inicial con los ARR
   % de  $S_m$ 
3  Evaluación( $FF_2$ , pop);
   % Calcula la función objetivo de cada
   % individuo
4  mientras  $FF_2 \neq \emptyset$  or Nb_generaciones hacer
5     inicio
6     % comienza el ciclo de reproducción
7     Nueva Reproducción(pop)
8     Evaluación ( $FF_2$ , pop)
9     selección (pop)
10    fin
11 fin

```

Este AG genera los individuos iniciales usando S_m (la matriz de firmas de fallas), y el resto del procedimiento es una AG clásico. La población inicial de AG2 es de 20 individuos, suficiente para analizar cuales ARRs (MTES) deben conformar la matriz de firmas de fallas, evaluados por la función objetivo FF_2 .

Por ser sus cromosomas también cadenas de bits, los operadores genéticos usados y la probabilidad de uso de cada uno es igual a como los usa el AG1. AG2 asegura que si hay un individuo con $FF_2 = 0$ representa una matriz de firmas de fallas que cumple con las propiedades de aislabilidad y detectabilidad. En este caso, el tiempo de ejecución teórico de AG2 $O(FF_2)$ es igual al número de fallas al cuadrado. Así, $O(AG2)$ es el máximo valor de: número de iteraciones, número de individuos, o número de fallas al cuadrado, para el problema que se esté resolviendo.

3.5. AG3

En el caso de que el sistema no cumpla con la propiedad de diagnosticabilidad (detectabilidad o aislabilidad), en el algoritmo 3 (*Localización*) se realiza la invocación a un nuevo AG, llamado AG3, el cual propone un nuevo grafo bipartito G' . Para ello, se seleccionan aleatoriamente variables desconocidas, y AG3 las asume como variables conocidas del sistema, lo cual significa incorporar sensores en el sistema para que puedan ser conocidas (se puedan medir).

Algoritmo Genético 3 ($G, G', cost$)

```

1  Inicio AG3
2  repita hasta tamaño de la población
3   $G' = generacion\_aleatoria(G)$ 
   % propone nuevos sensores en variables
   % desconocidas
4  pob = pob +  $G'$ 
5  fin

```

```

6  repita mientras no (Detect ( $G'_i$ )  $\neq true \forall G'_i \in pob$ ) o
   Nb_generaciones hacer
7  inicio
   % inicia el ciclo reproductivo
8   Reproducción (pob)
9  fin
10 si  $\exists$  Detect ( $G'_i$ ) =  $true \forall G'_i \in pob$  entonces
11 inicio
12   repita para  $\forall G'_i \in pob$  donde Deteccion( $G'_i$ ) =
   verdad
inicio
13  $cost_i = \sum_j^{#vd} (ind. Cost) * (ind. valor)$ 
   % Para cada sensor (ind.valor) existe un costo
   % (ind.costo). Cada valor de 1 de ind.valor
   % en el gen del individuo  $i$  significa un
   % sensor  $j$  escogido, donde  $vd$  es el número
   % de variables desconocidas
14 fin
15  $FF_3 = \min_i(cost_i)$ 
   % selecciona la configuración con menor
   % costo
16  $G' = G'_i$  donde  $\min_i(cost_i)$ 
17 fin
18 sino
19  $cost = \emptyset$ 
20 fin

```

En AG3 los grafos G' se generan aleatoriamente (cada G' es un individuo (cromosoma) que tiene las mismas características que el definido para AG1, ver línea 3 del algoritmo. Los G' se generan aleatoriamente desde G , al incluirse nuevos arcos (genes) como activos). AG3 revisa si con esa nueva configuración de sensores (arcos activos) propuesta en cada individuo se cumple con las propiedades de detectabilidad y aislabilidad (para eso, AG3 invoca el algoritmo *Detección* con G' como parámetro, ver la línea 6).

Mientras no se consigan individuos que cumplan con esas propiedades se generan nuevos G' (individuos) durante un determinado número de generaciones (ver líneas 6 al 9). Si AG3 logra varios individuos con esta propiedad, a continuación elige al individuo con el menor costo de implementación de la nueva configuración de sensores (ver línea 16), que se determina en función del costo de implantación de los sensores que se define en cada individuo (ver línea 14).

AG3 resuelve el problema de localización de sensores, y su resultado será el individuo con el menor costo para obtener a la propiedad diagnosticabilidad. Como dijimos antes, los individuos de AG3 son estructuralmente similares a los individuos de AG1, es decir, una cadena de bits. Por esa razón, los operadores genéticos usados, y sus probabilidades de uso, son los mismos que AG1.

El algoritmo que genera aleatoriamente la población de individuos G' (lleva a cabo una variación de G) se muestra a continuación.

Algoritmo **generacion_aleatoria(G)**

```

1  repita para  $k, #VD$ 
   % ajusta G añadiendo aleatoriamente
   % sensores en variables desconocidas
2   $G' = \text{Modifica}(G)$ 
   %  $G'$  con la nueva configuración de
   % sensores
3  fin

```

AG3 define una población de 5 individuos, que es suficiente para explorar los posibles sensores a incorporar para observar las fallas en estudio. La función de evaluación de AG3 considera el mínimo costo al momento de la evaluación de los individuos propuestos (ver línea 16 del algoritmo), solo si el resultado de la *Detección* es positivo (ver líneas 10 y 12 del algoritmo), y se caracteriza por la ecuación 9.

$$FF_3 = \min_i(cost_i) \quad (9)$$

En este caso, para definir el tiempo de ejecución teórico de AG3 se debe considerar el máximo entre $O(AG1)$ y $O(FF_3)$, ya que en este caso el número de iteraciones y de individuos de AG3 es muy pequeño.

4. EXPERIMENTOS

La mayoría de los procesos de producción de petróleo y gas requieren supervisión constante y el control de las instalaciones asociado. En particular, es necesario el diagnóstico y la detección temprana de fallas, con el fin de tener tiempo suficiente para contrarrestar las consecuencias que podrían ocasionar estas fallas. Es por esto que en esta sección se realiza un análisis de diagnosticabilidad en un pozo de levantamiento artificial.

Para ello, se usa el modelo matemático definido en [18].

Posteriormente, se realiza un análisis de diagnosticabilidad utilizando el método expuesto en la sección anterior. Este análisis revisa las propiedades de detectabilidad y aislabilidad del proceso. Si dichas propiedades no se cumplen, se proponen nuevos sensores en el proceso productivo.

4.1 Modelo de Extracción de petróleo por inyección de gas

Según [15], el Método de Extracción de crudo por Inyección de Gas (LAG) es un método que utiliza gas comprimido como fuente de energía para llevar los fluidos del yacimiento desde el fondo del pozo hasta la superficie; por lo tanto, su principal consideración a la hora de su selección en grupo de pozos petroleros es la disponibilidad de una fuente rentable de gas a alta presión.

En [19] realizan un estudio para observar el comportamiento de un pozo de inyección de gas caracterizado por las ecuaciones físicas que rigen el proceso, basada en tres variables de estado: x_1 es el flujo de masa de gas en el anular, x_2 es el flujo de masa de gas en la tubería y

x_3 es el flujo masa de petróleo en el tubing. Con ellas se pueden definir las ecuaciones del Modelo del pozo por extracción de crudo por inyección de gas (descrito por las ecuaciones del 10 al 22).

$$\dot{x}_2 = w_{iv} - w_{pg} \quad (10)$$

$$\dot{x}_3 = w_r - w_{po} \quad (11)$$

$$w_{gc} = C_{iv} \sqrt{(P_{plp} + 14.7) \cdot P_{gldp}} + f_1 \quad (12)$$

$$w_{iv} = C \sqrt{p_{a,inj} \cdot (P_a - P_{t,inj})} \quad (13)$$

$$w_{pc} = C \sqrt{p_m \cdot (P_t - P_s)} + f_2 \quad (14)$$

$$w_{pc} = \frac{x_2}{x_2 + x_3} \cdot w_{pc} \quad (15)$$

$$w_{pc} = C_r \cdot (P_r - P_{t,b}) + f_3 \quad (16)$$

$$p_{a,inj} = \frac{M}{RT_a} \cdot P_a \quad (17)$$

$$p_m = \frac{x_2 + x_3 - \rho_o L_r A_r}{L_t A_t} \quad (18)$$

$$P_a = \left(\frac{RT_a}{M_o} + \frac{g L_a}{V_a} \right) \quad (19)$$

$$P_t = \frac{M}{RT_a} \cdot \frac{x_2}{L_r A_r + L_t A_t - x_3 v_o} \quad (20)$$

$$P_{t,inj} = P_t + \frac{g}{A_r} \cdot (x_2 + x_3) \quad (21)$$

$$P_{t,b} = P_{t,inj} + \rho_o g L_r \quad (22)$$

Donde w_{gc} es la tasa de flujo másico de gas inyectado en el anular, w_{iv} es la tasa de flujo másico de gas desde el anular hacia el tubing, w_{pg} es el flujo másico de gas a través de la válvula de choke de producción, w_r es el flujo másico de petróleo proveniente del reservorio por el tubing, w_{po} es el flujo másico de petróleo a través de la válvula de choke y w_{pc} es el flujo másico mezclado en la tubería de producción.

C_{iv} , C_o y C_r son constantes, $p_{a,inj}$ es la densidad del gas en el anular en el punto de inyección, p_m es la densidad de la mezcla de petróleo/gas en la parte superior de la tubería, $gldp$ es la presión del sistema proveniente de plantas compresoras, $gldp$ es la presión diferencial medida a través de una placa orificio, P_a es la presión en el anular, $P_{t,inj}$ es la presión en la tubería en el punto de inyección de gas, P_t es la presión en la parte superior de la tubería, P_s es la presión en el separador, P_r es la presión en el reservorio, y $P_{t,b}$ es la presión en la parte inferior de la tubería.

Además, M_o es el peso molecular del gas, R es la constante del gas, T_a es la temperatura en el anular, T_t es la temperatura en la tubería, V_a es el volumen del anular, L_a es la longitud del anular, L_t es el longitud de la tubería, A_t es el área transversal de la tubería por encima del punto de inyección, L_r es la longitud desde el fondo del pozo hasta el punto de inyección de gas, A_r es el área transversal de la tubería por debajo del punto de inyección, g es la constante de gravedad, ρ_o es la densidad del petróleo, y v_o es el volumen específico

del petróleo. Finalmente, como el petróleo se considera incompresible, la densidad ρ_o es constante.

4.2. Análisis de diagnosticabilidad en un pozo por extracción de petróleo por inyección de gas

El análisis de diagnosticabilidad se basa en el modelo de relaciones de redundancia analíticas y en generación de residuos comparando el comportamiento del proceso con el modelo a través de las variables observadas.

En particular, se determinará la detectabilidad y aislabilidad de cinco fallas consideradas en el estudio: f_1 (falla en el flujo de gas inyectado en el anular), f_2 (falla en el flujo mixto en la línea de producción), f_3 (falla en el fluido mezclado en el tubing), f_4 (insuficiencia de presión en la parte inferior del tubing) y f_5 (falla en el tubing, en el punto de inyección de gas).

En nuestro enfoque, la capacidad de diagnóstico se desarrolla en base a esquemas de generación de residuos derivados de ARR. En primer lugar, con el análisis matemático descrito en la sección anterior, se identifican las ecuaciones que rigen el proceso, y se extiende con las ecuaciones definidas más abajo (ecuaciones 23 al 26).

$$y_1 = x_1 \quad (23)$$

$$y_2 = P_{ai} \quad (24)$$

$$y_3 = P_t \quad (25)$$

$$y_4 = P_r \quad (26)$$

En función de validar el funcionamiento del algoritmo de localización de sensores solo se realizan las mediciones de las variables descritas anteriormente (x_1, P_{ai}, P_t, P_r), y se dejan de medir el resto de variables. La matriz de relaciones entre ecuaciones y variables (también conocida como matriz de incidencia) para el caso de procesos LAG es mostrado en la Fig. 2

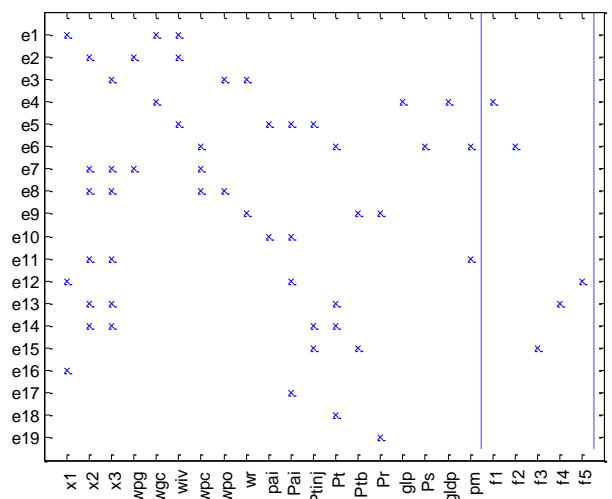


Figura 2. Matriz de relaciones entre ecuaciones y variables para procesos LAG

El resultado del análisis de diagnosticabilidad para el proceso LAG con cinco fallas, arroja que el mismo no cumple con esta propiedad al no poder aislar las fallas f_3 , f_4 y f_5 . Esto es debido a que no hay suficiente redundancia en el sistema porque son pocas las variables medidas. La Tabla 4 muestra la matriz de firma de fallas resultante de la evaluación de los AGs inmersos en el algoritmo de detección.

Por lo antes descrito, el análisis invoca al algoritmo de localización de sensores para determinar la posible ubicación de nuevos sensores que permitan incrementar la redundancia del sistema, y así poder cumplir con la propiedad de diagnosticabilidad

Tabla 4. Matriz de firma de fallas

Arr	Conjunto de Fallas					
	N	f_1	f_2	f_3	f_4	f_5
Arr ₁	0	1	1	1	1	1
Arr ₂	0	1	0	1	1	1
Arr ₃	0	0	0	1	1	1

El algoritmo de Localización realiza un llamado a al AG3, el cual propone una población de posibles sensores (G'), adicionando ecuaciones que corresponden a la inclusión de posibles sensores (para lo cual modifica el modelo original G).

Los nuevos sensores propuestos incrementan la redundancia del sistema LAG. La población de G' es evaluada por el algoritmo Detección, de manera de verificar que se cumpla con la propiedad de diagnosticabilidad en el sistema.

En general, el algoritmo de Localización va adicionando aleatoriamente una ecuación más en el modelo (G' , propuesta de un nuevo sensor), y va evaluando G' según el sensor propuesto. En las Tablas 5, 6 y 7 se muestran las matrices de firmas de fallas resultante del análisis proveniente de la propuesta de un nuevo sensor para medir/observar w_{gc} , o glp , o $gldp$, respectivamente.

Posteriormente, AG3 evalúa estos resultados con la función de aptitud descrita en la ecuación 11, que no es más que el mínimo costo de implantación de los sensores propuestos. En particular, en los sistemas LAG las variables de superficie tienen menor costo que las variables de fondo, y a su vez, en las variables de superficie los sensores de presión son de menor costo que los sensores de flujo másico. Por tal motivo, según los datos incluidos en la función de costo del AG3, él elige la variable X18, que equivale a $gldp$, la cual es la presión diferencial medida a través de una placa orificio.

Por otro lado, podemos corroborar el cumplimiento de la propiedad de diagnosticabilidad analizando la matriz de firma de fallas mostrada en la Tabla 7, donde se aprecia que todas las fallas son sensibles a un residuo.

Además, se pueden identificar todas las fallas consideradas, ya que sus firmas en la matriz son distintas unas de las otras (la propiedad de diagnosticabilidad se cumple en el sistema, al añadir cualquiera de los tres sensores propuesto por AG3, ver Tablas 5 al 7).

Tabla 5. Matriz de firma de fallas resultado del AG3 con un sensor en w_{gc}

Arr	Conjunto de Fallas					
	N	f_1	f_2	f_3	f_4	f_5
Arr ₁	0	1	1	1	1	1
Arr ₂	0	0	1	1	1	0
Arr ₃	0	0	0	0	1	1
Arr ₄	0	0	0	1	0	0

La Fig. 3 muestra la matriz de incidencia con la inclusión del nuevo sensor propuesto por AG3, y no es más que la inclusión de la ecuación propuesta por AG3, por ser el individuo con menor costo de implantación

Tabla 6. Matriz de firma de fallas resultado del AG3 con un sensor en glp

Arr	Conjunto de Fallas					
	N	f_1	f_2	f_3	f_4	f_5
Arr ₁	0	1	1	1	0	0
Arr ₂	0	0	1	1	1	1
Arr ₃	0	1	1	1	1	0
Arr ₄	0	0	0	1	0	0

Tabla 7. Matriz de firma de fallas resultado del AG3 con un sensor en $gldp$

Arr	Conjunto de Fallas					
	N	f_1	f_2	f_3	f_4	f_5
Arr ₁	0	0	1	1	1	1
Arr ₂	0	0	0	1	1	1
Arr ₃	0	0	1	1	1	0
Arr ₄	0	0	0	0	1	0

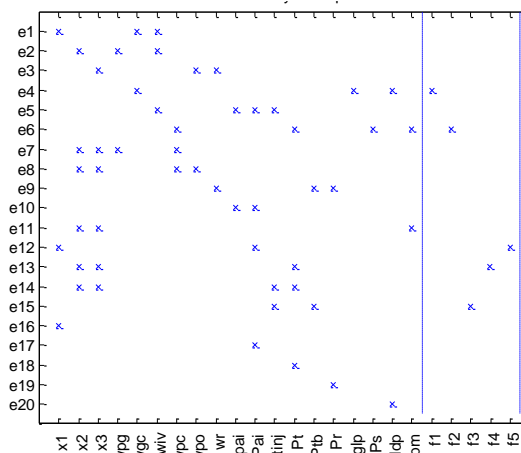


Figura 3. Matriz de incidencia del Sistema LAG resultado de la ejecución del AG3

5. CONCLUSIONES

La capacidad de detectar un fallo a tiempo en un sistema proporciona seguridad, disponibilidad y fiabilidad. En ese sentido, los mecanismos de diagnóstico de fallos cobran una especial relevancia. En este trabajo se propone un enfoque híbrido basado en las técnicas de análisis estructural y de AGs, con el fin de estudiar la diagnosticabilidad y localización de sensores en procesos petroleros.

Los resultados de la sección anterior verifican el comportamiento de nuestro enfoque, identificando tres diferentes sensores para medir variables que permiten detectar y aislar las fallas consideradas (de esta manera, el sistema es diagnosticable). Adicionalmente, este algoritmo evalúa el costo de implantación de los sensores,

REFERENCIAS

seleccionando aquellos cuyos costos sean menores. En general, el tiempo de ejecución teórico requerido de nuestra propuesta depende de la complejidad del problema a estudiar (número de ecuaciones y de fallas a analizar), ya que los valores de las variables de AG (número de iteraciones, número de individuos) no son grandes. Para el caso que se presenta en este trabajo es del orden de los segundos, siendo el mayor el de AG3.

Nuestro enfoque es diferente a cualquiera de los trabajos previos, porque permite estudiar ambos problemas simultáneamente [1, 9, 11]: analizar la diagnosticabilidad de un proceso, y localizar nuevos sensores si el proceso lo requiere para ser diagnosticable. Además, el análisis de localización de nuevos sensores es otra diferencia (ventaja) importante de nuestro algoritmo con respecto a [18], el cual no considera dichos costos.

Nuestro enfoque es una herramienta eficaz para resolver la problemática combinatoria detrás de la determinación del MTES, dado un modelo estructural como grafo bipartito.

Además, propone donde deben colocarse nuevos sensores, con el fin de incrementar la redundancia del sistema y cumplir con la propiedad de diagnosticabilidad (cuando el sistema no cumpla con ella). Así, nuestro enfoque lleva a cabo varios estudios en forma automática: la determinación de la detectabilidad y aislabilidad de las fallas consideradas en un sistema (propiedad diagnosticabilidad), y la posibilidad de definir nuevos sensores en el sistema estudiado. Hemos demostrado que el AG es una herramienta de gran alcance para proporcionarlas respuestas a estas preguntas. El tiempo de ejecución de los AGs puede ser controlado con sus parámetros, a fin de obtener buenas soluciones en un corto tiempo de cálculo.

Este es el primer trabajo de este tipo en el ámbito petrolero, Un trabajo futuro es extender el análisis de diagnosticabilidad a otros procesos de producción petrolera involucrados en la extracción y manejo del petróleo, tales como estaciones de flujo, levantamientos artificiales con métodos de bombas electrosumergibles, entre otras. Esto permitirá tener sistemas robustos, que tomen decisiones certeras sobre posibles fallas, reduciendo las consecuencias que estas puedan ocasionar en los mismos.

RECONOCIMIENTO

Este trabajo fue soportado por Programa de Cooperación de Postgrado (PCP) entre Venezuela y Francia llamado "Supervision and maintenance task in a shared organizational environment", contrato No. 2010000307, entre la Universidad de Los Andes, Venezuela en cooperación con LAAS- CNRS, Universidad de Toulouse, France. Dr. Aguilar ha sido parcialmente financiado por el Proyecto Prometeo del Ministerio de Educación Superior, Ciencia, Tecnología e Innovación de la República de Ecuador.

- [1] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and fault tolerant control*. Berlin: Springer, 2003.
- [2] J. Armengol, A. Bregon, T. Escobet, E. Gelso, M. Krysander, M. Nyberg, X. Olive, B. Pulido, and L. Travé-Massuyès, *Minimal structurally overdetermined sets for residual generation, A comparison of alternative approaches*, In Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, pp 1480–1485, 2009.
- [3] L. Travé-Massuyès, T. Escobet, and X. Olive. *Diagnosability analysis based on component supported analytical redundancy*. IEEE Trans. on Systems, Man, and Cybernetics – Part A: Systems and Humans: pp 1146–1160, 2006.
- [4] C. Svard and M. Nyberg. *Residual generators for fault diagnosis using computation sequences with mixed causality applied to automotive systems*, IEEE Transactions on Systems, Man, and Cybernetics – Part A, 40(6), pp 1310–1328, 2010
- [5] S. Sen, S. Narasimhan, K. Deb. *Sensor network design of linear processes using genetic algorithms*, Comput. Chem. Eng. 22(3), pp. 385–390, 1998
- [6] S. Jin, M. Zhou, A.S. Wu. *Sensor network optimization using a genetic algorithm*, In Proceedings 7th World Multiconference on Systemics, Cybernetics, 2003
- [7] A. Bhondekar, R. Vig, M. LalSingla, C. Ghanshyam, P. Kapur. *Genetic Algorithm Based Node Placement Methodology For Wireless Sensor Networks*, In Proceedings of the International Multi Conference of Engineers and Computer Scientists Vol I IMECS, 2009.
- [8] M. Krysander, J. Åslund, and E. Frisk, *A structural algorithm for finding testable sub-models and multiple fault isolability analysis*, In Proceedings of the 21st International Workshop on Principles of Diagnosis (DX-10), 2010.
- [9] L. Travé-Massuyès, T. Escobet, and R. Milne. *Model-based diagnosability and sensor placement. application to a frame 6 gas turbine subsystem*. In Proceedings DX01 twelfth international workshop on principles of diagnosis, pp 205–212, 2001.
- [10] M. Krysander, J. Åslund, and M. Nyberg, *An efficient algorithm for finding minimal over-constrained subsystems for model-based diagnosis*, IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, 38(1), pp 197–206, 2008
- [11] R. Leal, J. Aguilar, L. Trave-Massuyes, A. Ríos, E. Camargo. *A Genetic Algorithm Approach for Diagnosability Analysis*, International Journal of Engineering Development and Research, 2(4), pp. 3786–3799, 2014
- [12] M. Krysander, M. Nyberg. *Structural analysis utilizing MSS sets with application to a paper plant*. In Proceedings of the Thirteenth International Workshop on Principles of Diagnosis, 2002
- [13] M. Nyberg. *Automatic design of diagnosis systems with application to an automotive engine*. Control Engineering Practice, 87(8), pp 993–1005, 1999
- [14] V Puig, J. Quevedo, T. Escobet, B. Pulido. *On the integration of detection and isolation in model based fault diagnosis*. In Proceeding In DX04, 15th International Workshop on Principles of Diagnosis, 2004
- [15] G. Hernández, E. Alí and T. Sharon. *Manual del curso de levantamiento artificial por gas avanzado*. Segunda edición. PDVSA Intevep, Venezuela. 2001
- [16] M. Krysander, E. Frisk. *Sensor Placement for Fault Diagnosis* IEEE Transactions on Systems, Man, and Cybernetics part A: Systems and Humans, 38(6), 2008
- [17] E. Frisk, M. Krysander. *Sensor placement for maximum fault isolability*. In Proceedings of 18th international workshop on principles of diagnosis, pp 106–113, 2007
- [18] R. Leal, E. Camargo, J. Aguilar, A. Ríos and L. Travé. *Analysis of diagnosability for gas lift Wells*, In Ingeniería y Ciencias Aplicadas: Modelos Matemáticos y Computacionales (Ed. E. Dávila, J. Del Río, M. Cerrolaza, R. Chacón), Sociedad Venezolana de Métodos Numéricos en Ingeniería, pp. EC91-EC97, 2014.
- [19] J. Aguilar, F. Rivas (Eds), *Introducción a las Técnicas de Computación Inteligente*, Meritec, Venezuela. 2001.