

APLICACIÓN DE LOS ALGORITMOS EVOLUCIÓN DIFERENCIAL Y COLISIÓN DE PARTÍCULAS AL DIAGNÓSTICO DE FALLOS EN SISTEMAS INDUSTRIALES

Lídice Camps Echevarría¹, Orestes Llanes Santiago** y Antônio José da Silva Neto***

*Centro de Estudios Matemáticos. Instituto Superior Politécnico José Antonio Echeverría (ISPJAE). Calle 114, No. 11901 entre 119 y 127, Marianao, C. Habana, Cuba.

** Departamento de Automática y Computación. Facultad de Eléctrica, ISPJAE.

*** Departamento de Ing. Mecánica y Energía. IPRJ-UERJ. Nova Friburgo, Brasil.

RESUMEN

El diagnóstico de fallos en sistemas industriales tiene como objetivo la detección, localización y determinación de los fallos que afectan a un sistema industrial de producción. Un correcto diagnóstico permite gestionar adecuadamente la información conocida del sistema, de manera que se puedan tomar decisiones en cuanto a reparación y mantenimiento, lo que permite aumentar los niveles de seguridad y eficiencia. Este artículo explora las bondades de los algoritmos Evolución Diferencial y Colisión de Partículas para el desarrollo de métodos de diagnóstico de fallos basados en modelos. Como consecuencia de este estudio se ha desarrollado un algoritmo híbrido DE-PCA. Para la experimentación se han tomado datos con diferentes niveles de ruido simulados a partir del sistema de prueba Dos Tanques. Los resultados indican ser una línea efectiva para la obtención de métodos de diagnóstico de fallos robustos a perturbaciones externas y sensibles a fallos a partir de la cooperación entre ambas estrategias.

ABSTRACT

The fault diagnosis implicates the automatic early detection, isolation and localization of faults that have an effect on industrial systems. This topic is of high interest in order to improve the reliability and safety. This article explores the development of new model based fault diagnosis method. Our proposal is based on the application of the algorithms Differential Evolution and Particle Collision. This approach is illustrated using simulation data of the Two tanks benchmark. With the purpose of analyze the advantage of this approach, principally in the topic of robustness and sensibility, some experiments with noisy data and incipient faults are considered. The results indicate the suitability of this approach.

KEY WORDS: Differential Evolution, Heuristic algorithms, Fault diagnosis, Particle Collision Algorithm, Robustness.

MSC: 68W25

1. INTRODUCCIÓN

El diagnóstico de fallos (FDI de sus siglas en inglés) es un proceso que envuelve la detección y localización de los fallos que afectan un sistema industrial de producción. También incluye la determinación de las causas que provocaron estas desviaciones que impiden al sistema funcionar según el propósito con el que fue diseñado [6]. El fallo debe ser detectado rápidamente de manera que permita tomar decisiones que eviten consecuencias lamentables.

En la actualidad el diagnóstico está orientado a la predicción de futuros fallos con el fin de incrementar las posibilidades de reparación, mantenimiento y protección de las máquinas, procesos, capital humano y el medio ambiente. Esto, unido al incremento en la complejidad de los sistemas industriales, indica la importancia de un supervisor automático que permita diagnosticar el sistema [6, 14]

Varios han sido los métodos de FDI desarrollados. Estos se dividen en dos grandes grupos: aquellos que usan para el diagnóstico un modelo matemático del proceso [14] y los que se basan en otro tipo de conocimiento [15, 16].

¹lidice@ceamat.cujae.edu.cu

Los sistemas a diagnosticar están bajo la acción de perturbaciones que se pueden manifestar como ruido en las mediciones obtenidas a partir de los sensores ubicados en el mismo. Esta presencia influye negativamente en el desempeño del diagnóstico provocando falsas alarmas, detección tardía del fallo o poca facilidad de predicción de los mismos, lo que empaña la toma de decisiones. Los métodos de FDI que manejen adecuadamente estas dificultades se denominan robustos a perturbaciones externas [11]. Los métodos de FDI basados en modelos parten del conocimiento de un modelo matemático que describa el sistema y están recibiendo mayor atención en las últimas dos décadas desde el punto de vista teórico y desde las aplicaciones reales [11]. La idea general de estos consiste en la generación y evaluación de residuos [6]. Dicha generación se basa en las mediciones del sistema y valores estimados a partir del modelo. Esta etapa determina la robustez del método y establece diferencias entre ellos, destacándose los basados en observadores de estado, en ecuaciones de paridad y en estimación de parámetros [11, 14]. Tomando las ideas de estas tres variantes fundamentales se han desarrollado métodos de FDI robustos para el caso de modelos lineales o linealizables [1, 11]. Las aplicaciones prácticas de los FDI a partir de los métodos conocidos se encuentran limitadas debido al exceso de simplificaciones en la modelación, la no observabilidad del sistema y la existencia de relaciones dinámicas no lineales desconocidas, entre otras causas. Es por esto que el desarrollo de nuevos métodos de diagnóstico robustos y sensibles se considera, en la actualidad, como un problema de investigación abierto [12].

La estructura básica de los problemas de diagnóstico es similar a la de los problemas inversos (PI): a partir del conocimiento del comportamiento del sistema se desea establecer la causa que lo provocó [10]. En particular los problemas inversos de estimación de parámetros en ingeniería son aquellos donde a partir de un modelo matemático del fenómeno a estudiar, y conocida la solución del mismo, se desea estimar los parámetros del modelo que garantizan el comportamiento que se manifiesta en la solución conocida [10]. Los métodos de diagnóstico basados en la estimación de parámetros del sistema poseen la misma estructura que este tipo de problemas inversos.

Considerando la necesidad de desarrollar nuevos métodos de diagnóstico de fallos y la semejanza con la estructura de los problemas inversos; teniendo en cuenta las crecientes aplicaciones que encuentran los algoritmos basados en estrategias heurísticas en la solución de PI [5, 9, 17], y las recientes aplicaciones de algunos de estos algoritmos al diagnóstico de fallos [18, 19, 20], donde han demostrado poseer algunas cualidades deseables: rapidez, simple estructura, eficiencia, eficacia y robustez; se ha decidido iniciar un estudio de la aplicación de estos al desarrollo de métodos de FDI basados en modelos, en sistemas industriales dinámicos. En este trabajo se han considerado las estrategias Evolución Diferencial (DE) y Colisión de Partículas (PCA).

La estructura del artículo es como sigue, en la sección 2 se presentan los aspectos generales de los métodos de diagnóstico basados en estimación de parámetros. En la sección 3 se introducen las dos estrategias heurísticas aplicadas a dicha estimación y la estrategia híbrida propuesta. En la sección 4 se describe el sistema de prueba Dos tanques y la metodología que se propone para el diagnóstico de los fallos. La experimentación numérica es mostrada en la sección 5 y finalmente se presentan las conclusiones.

2. DIAGNÓSTICO MEDIANTE ESTIMACIÓN DE PARÁMETROS

Frecuentemente los fallos provocan cambios en los parámetros físicos del sistema. Esto ha llevado a desarrollar métodos de diagnóstico que se basan en la verificación de sus valores. De manera general estos parámetros no son directamente medibles, requiriendo de la aplicación de técnicas de estimación las que precisan del conocimiento de un modelo matemático del sistema bajo supervisión, que permita la generación de residuos.

Sea

$$\begin{aligned}\dot{\mathbf{x}}(t) &= f(\mathbf{x}(t), \mathbf{u}(t), \theta) \\ \mathbf{y}(t) &= g(\mathbf{x}(t))\end{aligned}\tag{1}$$

el modelo que representa adecuadamente las leyes físicas que gobiernan el comportamiento del sistema donde $\mathbf{x}(t) \in \mathbb{R}^n$ es el vector de variables de estado, $\mathbf{u}(t) \in \mathbb{R}^m$ es el vector de variables de entrada, $\mathbf{y}(t) \in \mathbb{R}^p$ es el vector de variables de salida y $\theta \in \mathbb{R}^l$ representa el vector de parámetros.

Consideraremos que las entradas y salidas del sistema son medidas directamente mediante el uso de sensores. El vector de parámetros del modelo $\theta = [\theta_1 \dots \theta_l]$ está relacionado con el vector de coeficientes del proceso físico $\rho \in \mathbb{R}^j$, donde en general se cumple $l \neq j$. La estimación de los parámetros θ_l y la comparación con los valores de estos en el estado de funcionamiento normal permitirá detectar los fallos. El diagnóstico se producirá una vez que se establezca la relación entre θ_l y ρ_j .

La estimación de θ puede obtenerse mediante la minimización de la suma de los residuos obtenidos de la diferencia entre los valores de las mediciones de sistema y las estimadas a partir del modelo del mismo. Este problema de optimización queda planteado como sigue:

$$\begin{aligned} \min \quad & F(\hat{\theta}) = \sum_{t=1}^S [\mathbf{y}(t) - \hat{\mathbf{y}}(t)]^2 \\ \text{s. a} \quad & \theta_{\min} \leq \hat{\theta} \leq \theta_{\max} \end{aligned} \quad (2)$$

donde S es la cantidad de mediciones que se obtienen de las variables de entrada $\mathbf{u}(t)$ y de salida $\mathbf{y}(t)$; $\hat{\mathbf{y}}(t)$ denota las estimaciones de las variables de salida calculadas a partir de los valores del vector de entrada $\mathbf{u}(t)$ y del modelo (1). La solución del problema de optimización (1) incluye implícitamente la solución del sistema de ecuaciones (1). La estructura de estos métodos tiene semejanza con la de los problemas inversos de estimación de parámetros en ingeniería [10] descritos en la introducción. Si la relación entre θ_l y ρ_j es biunívoca entonces el diagnóstico de fallos basado en la estimación de parámetros es factible.

Varios métodos de estimación de parámetros han sido aplicados al diagnóstico de fallos reconociéndose la necesidad de realizar mayores esfuerzos en el desarrollo de métodos de identificación de parámetros y modelación de sistemas no lineales. A pesar de los trabajos propuestos en esta dirección en [11, 12] se continúa reconociendo esta línea, como una de las más prometedoras en el tema del desarrollo de métodos de diagnóstico de fallos robustos y que se basan en modelos.

Si se considerara el diagnóstico de fallos a través de la estimación de parámetros, mediante el problema de minimización (2), entonces pueden ser aplicadas las técnicas heurísticas que han sido desarrolladas para solucionar problemas de optimización, en particular aquellas que se basan en algún proceso físico o biológico, independientemente de la no linealidad del modelo que describa al sistema y considerando diferentes niveles de ruido.

En este trabajo se aplican estrategias heurísticas al diagnóstico de fallos basado en estimación de parámetros. En este caso se han aplicado las técnicas DE y PCA. A continuación se describen las características fundamentales de DE y PCA.

3. EVOLUCIÓN DIFERENCIAL Y COLISIÓN DE PARTÍCULAS

En esta sección se presentan las características fundamentales de las estrategias heurísticas Evolución Diferencial y Colisión de Partículas. Además se presenta la estrategia híbrida desarrollada.

3.1. Evolución Diferencial

DE fue propuesto alrededor del año 1995 para problemas de optimización [13]. En sus inicios el algoritmo surgió en un intento de mejorar otro algoritmo que se basaba en Algoritmos Genéticos de Goldberg [3] y Recocido Simulado (Simulated Annealing, SA) [4].

DE es un algoritmo evolutivo basado en poblaciones. El algoritmo DE se basa en tres operadores, mutación, cruzamiento y selección que a diferencia de los operadores con igual nombre de Algoritmo Genéticos [3], están basados en operaciones con vectores.

La selección de DE se basa en las ventajas que presenta sobre los algoritmos genéticos, de los cuales se reporta una aplicación al diagnóstico de fallos en [20]: estructura simple, mayor velocidad y robustez [13]. Estas características resultan adecuadas al enfocar problemas donde se requiera una detección del fallo en tiempo real. Otro criterio para seleccionar DE es el éxito que ha tenido su aplicación a la solución de varios problemas inversos de estimación de parámetros en ingeniería [5].

La idea detrás de DE es generar una población de nuevas soluciones factibles a partir de perturbar soluciones pertenecientes a la población de soluciones obtenidas hasta ese momento. Este esquema de generación se basa en tres operadores: mutación, cruzamiento y selección. El esquema de generación puede expresarse mediante la notación:

$$DE/\mathbb{X}^j / \gamma/\lambda$$

donde \mathbb{X}^j denota la solución a perturbar en la iteración j ésima, γ el número de pares de soluciones de la población que serán usadas en la perturbación de \mathbb{X}^j y λ indica la función de distribución que será usada durante el cruzamiento. En este caso se ha considerado el esquema $DE/\mathbb{X}^{j(Best)} / 1/Bin$, donde $\mathbb{X}^{j(Best)}$ indica al mejor individuo de la población Γ_Z y Bin la función de distribución binomial. Este operador mutación expresado en detalles queda

$$\mathbb{X}^{j+1} = \mathbb{X}^{j(Best)} + F_S(\mathbb{X}^{j(\alpha)} - \mathbb{X}^{j(\beta)})$$

donde \mathbb{X}^{j+1} , $\mathbb{X}^{j(Best)}$, $\mathbb{X}^{j(\alpha)}$, $\mathbb{X}^{j(\beta)} \in \mathbb{R}^n$, $\mathbb{X}^{j(\alpha)}$ y $\mathbb{X}^{j(\beta)}$ son elementos de la población Γ_Z y F_S es el factor de escalamiento. Con el objetivo de complementar el operador de mutación se define el operador de cruzamiento para cada componente x_n del vector solución:

$$x_n^{j+1} = \begin{cases} x_n^{j+1} & \text{si } R < C_R \\ x_n^{j(Best)} & \text{en otro caso} \end{cases}$$

donde $0 \leq C_R \leq 1$, es la constante de cruzamiento que como se conoce es otro parámetro de control en DE. R es un número aleatorio que es generado mediante la distribución λ que en este caso es la distribución binomial.

Finalmente el operador de selección

$$\mathbb{X}^{j+1} = \begin{cases} \mathbb{X}^{j+1} & \text{si } F(\mathbb{X}^{j+1}) \leq F(\mathbb{X}^{j(Best)}) \\ \mathbb{X}^{j(Best)} & \text{en otro caso} \end{cases}$$

Los parámetros de control en DE son el tamaño de la población Z , la constante de cruzamiento C_R y el factor de escalamiento F_S . En [13] se brindan algunas reglas simples para la selección de los parámetros en dependencia del tipo de aplicación.

El esquema general de DE se muestra en el Algoritmo 1 y hace referencia a la versión originalmente propuesta en [13].

Algoritmo 1: Algoritmo DE

1. Generar población inicial de soluciones Γ_Z con cardinalidad Z .
2. Determinar el mejor individuo de la población $\mathbb{X}^{0(Best)}$.
3. **for** $j = 0$ to $j = J_{max} - 1$ **do**
4. Generar Z nuevos candidatos según operador Mutación.
5. Aplicar operador Cruzamiento a los Z candidatos.
6. Seleccionar la nueva población mediante el operador Selección.
7. Seleccionar el mejor individuo $\mathbb{X}^{(Best)}$.
8. Verificar criterio de parada.
9. **end for**
10. Mejor solución: $\mathbb{X}^{(Best)}$

3.2. Algoritmo de Colisión de Partículas

El algoritmo de Colisión de Partículas es de tipo Metrópolis [8] y está inspirado en la colisión de partículas nucleares. Es de reciente creación [9]. PCA, al igual que DE, ha sido aplicado a problemas inversos de estimación de parámetros en ingeniería [17].

El algoritmo simula la interacción entre neutrones en un reactor nuclear: una partícula al interactuar con un núcleo puede ser absorbida o repelida. Estas interacciones se simulan

identificando una solución factible con una partícula \mathbb{X}^j en cada iteración j -ésima. Considerando un problema de minimización, a partir de la perturbación estocástica de la partícula \mathbb{X}^j , se genera otra $\mathbb{X}^{j'}$. Las cualidades de ambas partículas son comparadas considerando como criterio el valor de la función $F(\mathbb{X})$.

- Si $F(\mathbb{X}^{j'}) \leq F(\mathbb{X}^j)$ entonces se *absorbe* $\mathbb{X}^{j'}$ como nueva solución y se realiza un búsqueda local (intensificación) alrededor de $\mathbb{X}^{j'}$.
- Si $F(\mathbb{X}^{j'}) > F(\mathbb{X}^j)$ entonces o se *repele* $\mathbb{X}^{j'}$ con probabilidad de repulsión $p_r = 1 - \frac{F(\mathbb{X}^{Best})}{F(\mathbb{X}^{j'})}$, a otra región del espacio de búsqueda favoreciendo la diversificación de \mathbb{X}^j mediante la intensificación de $\mathbb{X}^{j'}$ o se desecha y se pasa a la próxima iteración.

Al finalizar cada iteración se actualiza \mathbb{X}^{Best} . De esta forma el espacio de búsqueda es recorrido mediante eventos sucesivos de absorción y repulsión. La intensificación y la diversificación se logran mediante un función que se ha llamado *Diversificación* y que se representa en el Algoritmo 2. Esta función consta de un ciclo iterativo de J_c iteraciones, en cada una de las cuales se realizan perturbaciones estocásticas de igual orden en $\mathbb{X}^{j'}$. Estas perturbaciones se pueden representar para cada iteración del ciclo interior j_c mediante:

$$x_n^{j(j_c)} = x_n^{j(j_c-1)} + (x_n^{Uj_c} - x_n^{j(j_c-1)})rd_1 - (x_n^{Lj_c} - x_n^{j(j_c-1)})(1 - rd_1)$$

siendo rd_1 un número aleatorio entre 0 y 1 y siendo además:

$$x_n^{Lj_c} = \begin{cases} x_n^{j(j_c-1)}rd_2 & \text{si } x_n^{j(j_c-1)}rd_2 \geq L_n \\ L_n & \text{en otro caso} \end{cases}$$

y

$$x_n^{Uj_c} = \begin{cases} x_n^{j(j_c-1)}rd_3 & \text{si } x_n^{j(j_c-1)}rd_3 \leq U_n \\ U_n & \text{en otro caso} \end{cases}$$

donde L_n y U_n son el mínimo y el máximo, respectivamente, para la variable x_n ; rd_2 es un número aleatorio que pertenece al intervalo $[0.8; 1]$ y rd_3 es otro número aleatorio que pertenece al intervalo $[1; 1.2]$.

Algoritmo 2: Función Diversificación $\mathbb{X}^{j'}$

1. Hacer $\mathbb{X}^{j(1)} = \mathbb{X}^{j'}$.
2. **for** $j_c = 1$ to $j_c = J_c$ **do**
3. Hacer una perturbación en $\mathbb{X}^{j(j_c)}$ y guardarla en $\mathbb{X}^{j(j'_c)}$.
4. **if** $F(\mathbb{X}^{j(j'_c)}) \leq F(\mathbb{X}^{j(j_c)})$ **then**
5. $\mathbb{X}^{j(j_c+1)} = \mathbb{X}^{j(j'_c)}$
6. **else** $\mathbb{X}^{j(j_c+1)} = \mathbb{X}^{j(j_c)}$
7. **end if**
8. **end for**
9. $\mathbb{X}^j = \mathbb{X}^{j(j_c)}$

Los parámetros de control del algoritmo son el número máximo de iteraciones J_{max} del ciclo exterior y el número de iteraciones J_c del ciclo de la intensificación. La estructura general de PCA se muestra en el Algoritmo 3. Esta es la versión original presentada en [9] y no se han hecho modificaciones en ella.

Algoritmo 3: Algoritmo PCA

1. Generar una solución factible inicial \mathbb{X}^0 .
2. **for** $j = 0$ to $j = J_{max} - 1$ **do**

3. Obtener otra solución factible $\mathbb{X}^{j'}$ a partir de la perturbación de \mathbb{X}^j .
4. **if** $F(\mathbb{X}^{j'}) \leq F(\mathbb{X}^j)$ **then**
5. Hacer $\mathbb{X}^j = \text{Diversificación}(\mathbb{X}^{j'})$.
6. **else**
7. Generar un número aleatorio rd entre 0 y 1.
8. Calcular p_r .
9. **If** $rd > p_r$ **then**
10. Hacer $\mathbb{X}^j = \text{Diversificación}(\mathbb{X}^{j'})$.
11. **end if.**
12. **end if.**
13. Actualizar \mathbb{X}^{Best} .
14. Verificar criterio de parada.
15. **end for**

3.3 Algoritmo Híbrido DE-PCA

La idea detrás del algoritmo híbrido es manejar adecuadamente las bondades de DE y PCA, a la vez que pretende superar el comportamiento que independientemente manifiestan DE y PCA cuando son aplicados al diagnóstico.

Luego de estudiar ambos algoritmos se ha considerado que una estrategia híbrida adecuada podría lograrse manteniendo las características fundamentales del mecanismo de mutación y cruzamiento de DE, que le proporcionan una estructura sencilla y de fácil implementación, y realizar cambios en el operador de selección a partir del criterio de absorción y repulsión de PCA.

Desde el punto de vista de DE, el algoritmo híbrido DE-PCA mejora su operador de selección mediante la idea de absorción y repulsión de PCA y solamente le incorpora un nuevo parámetro, J_c . Desde el punto de vista de PCA el algoritmo híbrido DE-PCA lo extiende para el caso de poblaciones y supera la generación estocástica del individuo a perturbar en cada iteración mediante el esquema de mutación y cruzamiento de DE, lo que permite al algoritmo mantener una *memoria superior*. Otras versiones de PCA al caso de poblaciones han sido propuestas recientemente [2].

El algoritmo híbrido usa como esquema de generación $DE/\mathbb{X}^j / 1/Bin$, ahora no se toma el mejor individuo en el esquema de generación, esta es una variación con respecto al algoritmo 1 y que proporciona mayor diversificación en la búsqueda. A este se le aplican los operadores de mutación y cruzamiento que se describieron para DE. Es en la selección de este nuevo individuo, como miembro o no de la población, que se hacen los cambios según la idea de absorción y repulsión de PCA: si el individuo supera al mejor individuo de la población, es absorbido, lo que significa que sustituye en la población al que lo había generado; si no supera al mejor individuo de la población se repele con probabilidad p_r , descrita para PCA, lo que implica que se hará una búsqueda estocástica alrededor de este y el resultado de esta será quien sustituya al que le dio lugar. Esta parte del algoritmo hace un llamado al algoritmo de PCA descrito en el algoritmo 3 sin hacer transformaciones. La estructura se muestra en el Algoritmo 4.

Algoritmo 4: Algoritmo DE-PCA

1. Generar población inicial de soluciones Γ_Z con cardinalidad Z .
2. Determinar el mejor individuo de la población $\mathbb{X}^{0(Best)}$.
3. **for** $j = 0$ to $j = J_{max} - 1$ **do**
4. Generar Z nuevos candidatos $\mathbb{X}_1^{j+1}, \mathbb{X}_2^{j+1} \dots \mathbb{X}_Z^{j+1}$. según operador Mutación de DE.
5. Aplicar operador Cruzamiento de DE a los candidatos.
6. **for** $i = 1$ to $i = Z$ **do**
7. **if** $F(\mathbb{X}_i^{j+1}) \leq F(\mathbb{X}^{j(Best)})$ **then**
8. Hacer $\mathbb{X}_Z^j = \mathbb{X}_Z^{j+1}$.
9. **else**
10. Generar un número aleatorio rd entre 0 y 1.
11. Calcular p_r .
12. **If** $rd > p_r$ **then**
13. Hacer $\mathbb{X}_Z^j = \text{Diversificación}(\mathbb{X}_Z^{j+1})$.
14. **end if.**
15. **end if.**

16. end for

17. Actualizar individuo $\mathbb{X}^{(Best)}$.

18. Verificar criterio de parada.

19. end for

20. Mejor solución: $\mathbb{X}^{(Best)}$

4. SISTEMA DE PRUEBA DOS TANQUES

La Figura 1 muestra el sistema de prueba *Dos Tanques*. Esta es una versión simplificada del sistema de los tres tanques que fue adoptado como problema de prueba para FDI y control reconfigurable a partir de la década de los noventa [7].

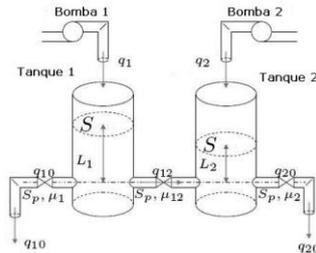


Figura 1. Sistema de prueba *Dos Tanques*.

El sistema consiste en dos tanques, tanque 1 y tanque 2, de igual sección transversal $S_1 = S_2$ sobre los que actúan dos bombas independientes 1 y 2 que controlan la entrada del líquido con flujo q_1 y q_2 sobre los tanques 1 y 2, respectivamente, de manera que se mantenga constante e igual a cierto valor nominal la altura del líquido en cada uno de ellos ($\bar{L}_1 = 4.0 \text{ m}$, $\bar{L}_2 = 3.0 \text{ m}$).

Todas las tuberías tienen la misma sección transversal S_p y los tanques se conectan mediante una de ellas. El nivel del líquido en los tanques L_1 y L_2 son las variables controladas y son medidas con sensores. Las variables q_1 y q_2 permiten controlar el nivel sobre los tanques 1 y 2, respectivamente.

El sistema puede presentar dos fallos: Fallo 1 (Salidero en el tanque 1 con un flujo saliente de valor q_{f1}) y Fallo 2 (Salidero en el tanque 2 con un flujo saliente de valor q_{f2}).

Bajo la presencia de los fallos descritos, a partir de la ley de conservación de la masa, de la aplicación de la Ley de Torricelli y considerando $C_i = \mu_i S_p \sqrt{2g}$, se obtienen las ecuaciones diferenciales que describen el sistema:

$$\begin{cases} \dot{L}_1 = \frac{q_1}{s_1} - \frac{c_1}{s_1} \sqrt{L_1} - \frac{c_3}{s_1} \sqrt{|L_1 - L_2|} \text{sig}(L_1 - L_2) - \frac{q_{f1}}{s_1} \sqrt{L_1} \\ \dot{L}_2 = \frac{q_2}{s_2} - \frac{c_2}{s_2} \sqrt{L_2} + \frac{c_3}{s_2} \sqrt{|L_1 - L_2|} \text{sig}(L_1 - L_2) - \frac{q_{f2}}{s_2} \sqrt{L_2} \\ y_1 = L_1 \\ y_2 = L_2 \end{cases} \quad (3)$$

El objetivo es diagnosticar la presencia de los fallos 1 ó 2. Como una primera aproximación se supone que el tamaño de los salideros en los tanques no cambia en el tiempo y que estos no permiten un flujo mayor que 1 l/s. Estas suposiciones se establecen mediante restricciones para los parámetros q_{fi} , $i = 1, 2$:

$$q_{f1}, q_{f2} \in \mathbb{R}: 0 \leq q_{f1}, q_{f2} \leq 1 \text{ l/s}$$

Las estimaciones de q_{f1} y q_{f2} permiten diagnosticar el sistema. Con el objetivo de estimar estos parámetros se formula el siguiente problema:

$$\min F(\hat{\theta}) = \sum_{t=1}^s (q(t, \theta) - \hat{q}(t, \hat{\theta}))^2$$

donde $\hat{\theta} = (\hat{q}_{f1}, \hat{q}_{f2})^t$, $q(t, \theta) = (q_1(t), q_2(t))^t$ son las mediciones de los valores de las variables que controlan la altura del líquido en los tanques, $\hat{q}(t, \hat{\theta}) = (\hat{q}_1(t, \hat{\theta}), \hat{q}_2(t, \hat{\theta}))^t$ son los valores de las variables de control estimados mediante el modelo (3) y por tanto $F: \mathbb{R}^2 \rightarrow \mathbb{R}^2$.

Los valores de las constantes consideradas en la simulación del sistema de prueba *Dos Tanques* son $C_1 = C_2 = 0,3028 \text{ m}^2$, $S_1 = S_2 = 2,54 \text{ m}^2$, $S_p = 0,1 \text{ m}^2$, $g = 9,8 \text{ m/s}^2$.

5. EXPERIMENTACIÓN NUMÉRICA

5.1. Simulación del sistema de prueba *Dos Tanques*

El comportamiento del sistema fue simulado en varias situaciones con fallos y sin fallos. La simulación se basó en el modelo (3) cuya solución se realizó mediante Runge Kutta 4. A los valores que arroja la solución numérica del modelo les fue añadido ruido de un 2 %, un 5 % y en otros un 8 %, de esta forma se simula el efecto de las perturbaciones externas al sistema y que influyen en el diagnóstico. Todas las implementaciones fueron hechas en MATLAB 7.0.1. La Figura 2 muestra el comportamiento del sistema en funcionamiento normal y los datos poseen un ruido de 2%.

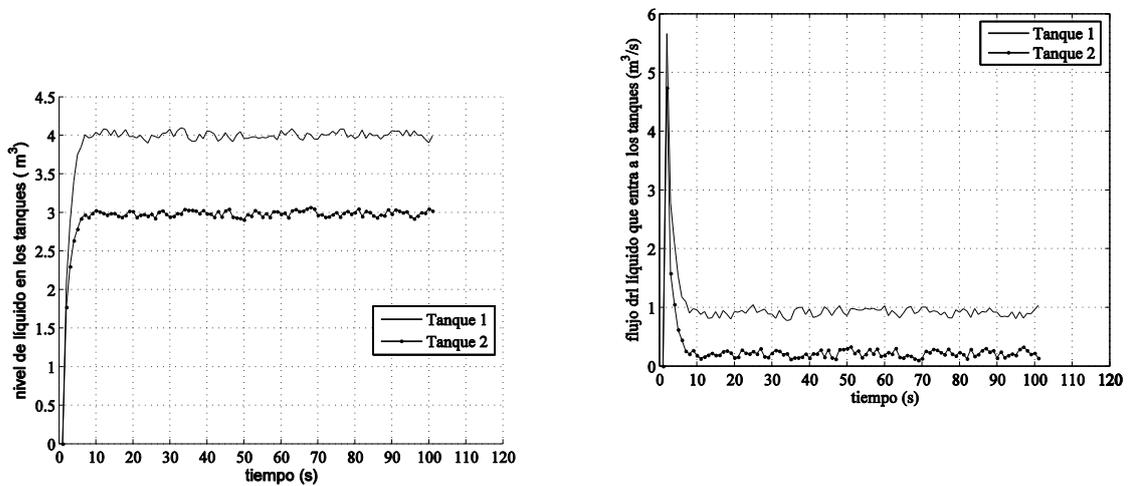


Figura 2. Simulación del comportamiento del sistema cuando no es afectado por fallos y las mediciones presentan un ruido de un 2 %.

De igual forma la Figura 3 reproduce el comportamiento cuando a partir de los 10 s, en el tanque 1 se presenta un salidero de 80 ml/s y en el tanque 2 uno de 250 ml/s; adicionalmente los datos poseen un ruido de un 5 %.

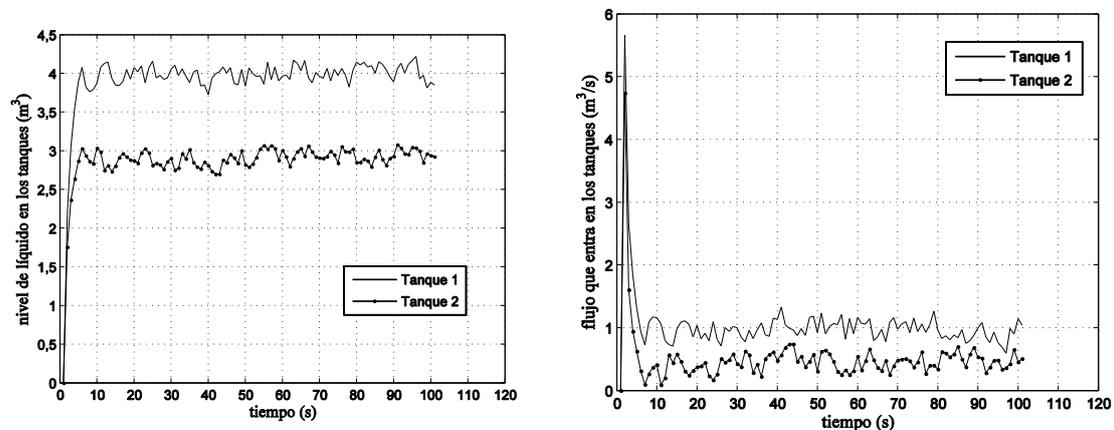


Figura 3. Simulación del comportamiento del sistema cuando a partir de los 10 s de iniciado el proceso en el tanque 1 se presenta un salidero de 80 ml/s y en el tanque 2 uno de 250 ml/s. Las mediciones presentan un ruido de un 5 %.

5.2. Diagnóstico de fallos mediante DE y PCA

Para diagnosticar el sistema se propone la solución del problema de minimización (4) mediante la implementación de DE, PCA y el algoritmo híbrido DE-PCA.

El criterio de parada en los tres algoritmos consideró el número de iteraciones $J_{max} = 100$, y una medida del valor de la función objetivo $\varepsilon_F < 0.1$ siendo $\varepsilon_F = \|F(\hat{\theta})\|_{\infty}$. De esta forma se tiene que la función a minimizar es $\varepsilon_F: \mathbb{R}^2 \rightarrow \mathbb{R}$ que permite realizar comparaciones de una manera más sencilla.

Cada iteración de DE y DE-PCA proporciona una población de las mejores Z soluciones obtenidas mientras PCA devuelve un individuo que representa la mejor solución obtenida hasta dicha iteración. Todas las implementaciones fueron realizadas en MATLAB 7.0.1.

5.2.1, Implementación de DE

La implementación de DE se basó en la descripción del Algoritmo 1. Se han considerado los siguientes valores para los parámetros: tamaño de la población $Z = 10$, $C_R = 0.7$ y $F_S = 0.6$. Estos valores son recomendados en [13].

5.2.2 Implementación de PCA

La implementación de PCA se basó en el Algoritmo 3. El parámetro del ciclo de la función Diversificación, mostrada en el algoritmo 2, se tomó $J_c = 10$. Estos valores son recomendados en [9].

5.2.3 Implementación del algoritmo híbrido DE-PCA

La implementación de DE-PCA se basó en la descripción hecha en el Algoritmo 4. Los parámetros de DE se fijaron según: tamaño de la población $Z = 10$, $C_R = 0.7$ y $F_S = 0.4$. En este caso se aumentó la probabilidad de mantener en las componentes del vector solución generado aquellos valores de las componentes que pertenecen al individuo que se perturba X^j . Este incremento se justifica en la incorporación de una búsqueda local según el esquema de repulsión de PCA, en caso de que la nueva solución no supera la mejor de la población. El valor de J_c se redujo a 5.

5.3 Resultados de la experimentación

La Tabla 1 muestra los resultados del diagnóstico para la misma situación de fallo pero bajo diferentes niveles de ruido afectando el sistema (2% y 5% respectivamente) cuando se consideraron las estrategias puras solamente. La notación usada es consecuente con la introducida anteriormente y se añade $Eval F$ para el número de evaluaciones de la función objetivo y $ruido$ para indicar el nivel de este que se introdujo en las mediciones. Ambos algoritmos detectan la presencia del fallo y con bastante precisión la magnitud de los mismos. En ambos experimentos el diagnóstico de DE es más preciso pero es mayor el número de evaluaciones de la función objetivo.

En la Tabla 2 se muestra la comparación entre los resultados de ambos algoritmos, en este caso se ha introducido la notación $prom Eval F$ para el promedio de las evaluaciones de la función objetivo. La mayor variación entre las dos situaciones de ruido la tuvo DE al estimar el parámetro q_{f2} . Con el objetivo de comparar los algoritmos en cuanto a la sensibilidad del diagnóstico, se realizaron comparaciones cuando un fallo incipiente se presenta en el tanque 1 (de 80 ml/s) y las mediciones presentan un ruido de un 2%. Las comparaciones para la media y la varianza de las estimaciones de los parámetros se muestran en las dos primeras filas de la Tabla 2. Se observa que ambos algoritmos diagnosticaron adecuadamente el fallo y los resultados no difieren considerablemente, aunque siguen siendo los de PCA menos precisos.

Además se compararon los diagnósticos tomando como criterio el balance que establecen entre robustez a perturbaciones y sensibilidad a fallos incipientes. Para eso se consideró la situación anterior pero el ruido se incrementó hasta un 8%. Los resultados se muestran en las dos últimas filas de la Tabla 3 donde se observa que DE maneja aceptablemente esta situación. El diagnóstico mediante DE no se aleja del mostrado en las primeras filas de la Tabla 3, cuyas simulaciones presentaban un ruido de un 2%,

excepto en la estimación del salidero del segundo tanque. Sin embargo se evidencia un incremento considerable en el promedio de evaluaciones de la función objetivo para PCA.

Tabla 1. Diagnóstico obtenido bajo un salidero de 250 ml/s en cada tanque

algoritmo	fallo: $q_{f1} = q_{f2} = 0.25$, ruido 2%			fallo: $q_{f1} = q_{f2} = 0.25$, ruido 5%		
	\hat{q}_{f1}	\hat{q}_{f2}	Eval F	\hat{q}_{f1}	\hat{q}_{f2}	Eval F
DE	0.2461	0.2508	580	0.2511	0.2493	580
DE	0.2476	0.2475	570	0.2525	0.2504	580
DE	0.2512	0.2501	450	0.2470	0.2516	600
DE	0.2083	0.2501	720	0.2475	0.2482	630
DE	0.2517	0.2528	510	0.2705	0.2628	1000
DE	0.2554	0.2597	560	0.2744	0.2744	1000
DE	0.2393	0.2505	660	0.2520	0.2497	620
DE	0.2604	0.2711	1000	0.2372	0.2604	730
DE	0.2520	0.2502	630	0.2547	0.2576	720
DE	0.2509	0.2496	600	0.2137	0.2001	100
PCA	0.2503	0.2120	110	0.2652	0.2733	99
PCA	0.2488	0.2501	54	0.2591	0.2214	99
PCA	0.2505	0.2498	60	0.2093	0.2577	110
PCA	0.2391	0.2502	88	0.2502	0.2585	44
PCA	0.2388	0.2490	110	0.2401	0.2379	66
PCA	0.2503	0.2143	110	0.2504	0.2143	99
PCA	0.2500	0.2504	44	0.2235	0.2512	880
PCA	0.2496	0.2507	55	0.2712	0.2466	110
PCA	0.2002	0.2714	1100	0.2635	0.2013	550
PCA	0.2515	0.2805	770	0.2490	0.2510	44

Tabla 2. Comparación del diagnóstico mostrado en la Tabla 1.

alg	ruido	μ_{qf1}	μ_{qf2}	σ_{qf1}	σ_{qf2}	$\frac{\sigma_{qf1}}{\mu_{qf1}} \cdot 10$	$\frac{\sigma_{qf2}}{\mu_{qf2}} \cdot 10$	Prom Eval F
DE	2%	0.2463	0.2532	2.092e-004	5.05e-005	5.87	2.8	630
PCA	2%	0.2429	0.2478	2.473e-004	4.513e-004	6.47	8.57	250
DE	5%	0.2454	0.2480	2.468e-004	0.96e-004	6.44	7.31	720
PCA	5%	0.2482	0.2458	3.764e-004	4.359e-004	7.82	8.49	309

Con el objetivo de analizar la robustez de ambos algoritmos, se simuló el sistema cuando no está afectado por fallos y está expuesto a perturbaciones que se manifiestan en un ruido de 8% en las mediciones. Los resultados se muestran en la Tabla 4. Como se puede observar, PCA diagnóstica de manera similar a DE, no se notan las diferencias de la Tabla 3.

Tabla 3. Resultados del diagnóstico obtenido bajo un salidero de 80 ml/s en el tanque 1.

alg	ruido	μ_{qf1}	μ_{qf2}	σ_{qf1}	σ_{qf2}	Prom Eval F
DE	2%	0.0794	0.0004	2.21e-005	1.5e-007	650
PCA	2%	0.0794	0.0005	2.778e-005	2.3e-007	246
DE	5%	0.0760	0.0010	4.847e-005	4.2e-007	940
PCA	5%	0.0688	0.0022	1.417e-004	1.9e-006	911

Luego de analizar el diagnóstico mediante DE y PCA de manera independiente, se analizó el obtenido mediante el algoritmo híbrido DE-PCA. En este sentido es interesante analizar si este supera el balance entre sensibilidad a fallos incipientes y robustez que mostraron los experimentos anteriores. Las seis primeras columnas de la Tabla 5 muestran los resultados de las comparaciones cuando se simula un salidero incipiente de 20 ml/s en ambos tanques y un ruido de 8%. La notación usada es algoritmo* para el peor comportamiento y algoritmo para el mejor. En el peor de los casos de DE y PCA se detecta el

fallo pero se sobreestima su dimensión y esto puede conducir a tomar decisiones erróneas. El peor de los casos de DE-PCA se comporta de manera semejante al mejor de DE y PCA respectivamente.

Tabla 4. Resultados del diagnóstico obtenido cuando el sistema no se encuentra bajo la acción de fallos.

alg	ruido	μ_{qf1}	μ_{qf2}	σ_{qf1}	σ_{qf2}	Prom Eval F
DE	8%	0.00072	0.00050	1.818e-007	1.489e-007	890
PCA	8%	0.00060	0.00059	1.689e-007	1.966e-007	659

Las seis últimas columnas de la Tabla 5 muestran otros resultados de las comparaciones en cuanto a robustez. En este caso no se consideran salideros y se mantienen perturbaciones que se manifiestan en un ruido en las mediciones de un 8%. Nuevamente el diagnóstico de la estrategia híbrida DE-PCA supera los de las puras, quienes darían lugar a falsas alarmas.

6. CONCLUSIONES

En este trabajo se plantea el estudio de la aplicación de estrategias heurísticas al diagnóstico de fallos en sistemas industriales. Esta forma de diagnóstico cae dentro de la clasificación de diagnóstico basado en modelos con la ventaja que no requiere la linealización de los mismos. En este caso se han aplicado dos algoritmos, DE y PCA, al sistema de estudio *Dos Tanques*.

Los resultados muestran las ventajas generales de la aplicación de estas técnicas: correcto diagnóstico y fácil implementación. Además han mostrado robustez a las perturbaciones externas. En particular DE ha logrado un adecuado balance entre sensibilidad a fallos incipientes y robustez a perturbaciones, lo que facilitaría el proceso de toma de decisiones a partir de los resultados del diagnóstico.

Tabla 5. Comparación del diagnóstico obtenido bajo diferentes situaciones

<i>fallo: $q_{f1} = q_{f2} = 0,02$, ruido 8%</i>					<i>sin fallo: $q_{f1} = q_{f2} = 0$, ruido 8%</i>				
No	alg	\hat{q}_{f1}	\hat{q}_{f2}	Eval F	No.	alg	\hat{q}_{f1}	\hat{q}_{f2}	Eval F
1	DE-PCA*	0.0214	0.0201	362	1	DE-PCA*	0.0000	0.0000	440
2	DE*	0.0221	0.0250	570	2	DE*	0.0003	0.0080	620
3	PCA*	0.0253	0.0285	880	3	PCA*	0.0009	0.0085	770
4	DE-PCA*	0.0262	0.0238	456	4	DE-PCA*	0.0262	0.0238	475
5	DE*	0.0517	0.0421	1000	5	DE*	0.1191	0.0050	1000
6	PCA*	0.0626	0.0455	1100	6	PCA*	0.0183	0.0486	1100

A partir del estudio de DE y PCA se ha desarrollado la estrategia híbrida DE-PCA. Dicho algoritmo mostró mejores resultados que los obtenidos de manera independiente por DE y PCA en el diagnóstico de fallos incipientes en ambientes ruidosos. Esto favorece la detección temprana permitiendo establecer estrategias de mantenimiento.

La aplicación de la estrategia híbrida también mostró un diagnóstico más robusto y disminuyó el número de evaluaciones de la función objetivo con respecto a DE y PCA. Esto indica sus posibilidades de aplicación en procesos en tiempo real. Además ha mostrado una capacidad de disminución de las falsas alarmas.

AGRADECIMIENTOS: Los autores agradecen el apoyo ofrecido por FAPERJ, Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro, C NPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico, y CAPES, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.

RECEIVED SEPTEMBER, 2011
REVISED APRIL, 2012

REFERENCIAS

[1] CHEN, J. y PATTON, R. J. (1999): **Robust model-based fault diagnosis for dynamic systems**. Dordrecht: Kluwer Academic Publishers, Amsterdam

- [2] DA LUZ, E. F. P., BECCENERI, J. C. , y DE CAMPOS VELHO, H. F. (2008): A new multi particle collision algorithm for optimization in a high-performance environments. **Journal of Computational Interdisciplinary Sciences**, 1, 1–7.
- [3] GOLDBERG, D. E. (1989): **Genetic Algorithms in search, optimization, and machine learning**. Reading, MA: Addison-Wesley.
- [4] KIRKPATRICK, S., GELATT, C. D. y VECCHI, M. P.(1983): Optimization by simulated annealing. **Science**, 220, 671–680.
- [5] LOBATO, F. S, STEFFEN, V. y SILVA-NETO, A. J.(2009): Solution of inverse radiative transfer problems in two-layer participating media with differential evolution. **Inverse Problems in Science and Engineering**, 00, 1–12.
- [6] ISERMANN, R.(1997): Supervision, fault-detection and fault-diagnosis methods- an introduction. **Control Eng. Practice**, 5, 639–652.
- [7] LUNZE, J. (1998): Laboratory three tanks system -benchmark for the reconfiguration problem. **Technical report, Tech. Univ. of Hamburg-Harburg, Inst. of Control. Eng., Germany**.
- [8] METROPOLIS, N., ROSENBLUTH, A.W. y TELLER, E. (1953): Equations of state calculations by fast computing machines. **Control Eng. Practice**, 21, 1087–1092.
- [9] SACCO, W. F. y OLIVEIRA, C. R. E. (2005): A new stochastic optimization algorithm based on particle collisions. En **2005 ANS Annual Meeting, Transactions of the American Nuclear Society**.
- [10] SILVA-NETO, A. J. y MOURANETO, F. D. (2005): **Problemas Inversos - Conceitos Fundamentais e Aplicaciones**. EdUERJ.
- [11] SIMANI, S., FANTUZZI, C. y PATTON, R. J. (2002): **Model-Based Fault Diagnosis in Dynamic systems Using Identification Techniques**. Springer-Verlag, Heidelberg.
- [12] SIMANI, S. y PATTON, R. J. (2008): Fault diagnosis of an industrial gas turbine prototype using a system identification approach. **Control Engineering Practice**, 16, 769–786.
- [13] STORN, R. y PRICE, K. (1995): Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces. **International Computer Science Institute**
- [14] VENKATASUBRAMANIAN, V., RENGASWAMY, R., YIN, K. y KAVURI, S. N. (2003): A review of process fault detection and diagnosis part I: Quantitative model-based methods. **Computers and Chemical Engineering**, 27, 293–311.
- [15] VENKATASUBRAMANIAN, V., RENGASWAMY, R., YIN, K. y KAVURI, S. N. (2003): A review of process fault detection and diagnosis part II: Quantitative model-based methods and search strategies. **Computers and Chemical Engineering**, 27, 313–326.
- [16] VENKATASUBRAMANIAN, V., RENGASWAMY, R., YIN, K. y KAVURI, S. N. (2003): A review of process fault detection and diagnosis part III: Process history based methods. **Computers and Chemical Engineering**, 27, 327–346.
- [17] FIGUEREIDO-SACCO. W., OLIVEIRA, C.R.E. y PEREIRA, C.M.N.A. (2006): Two stochastic optimization algorithms applied to nuclear reactor core design. **Progress in Nuclear Energy**, XX, 1–15.
- [18] WANG, L., NIU, Q. y FEI, M. (2008): A novel quantum ant colony optimization algorithm and its application to fault diagnosis. **Transactions of the Institute of Measurement and Control**, 30, 313–329.
- [19] WITCZAK, M. (2006): Advances in model based fault diagnosis with evolutionary algorithms and neural networks. **Int. J. Appl. Math. Comput. Sci.**, 16, 85–99.

[20] YANG, E., XIANG, H., GU, D. y ZHANG, Z. (2007): A comparative study of genetic algorithm parameters for the inverse problem-based fault diagnosis of liquid rocket propulsion systems. **International Journal of Automatic.**