



Aplicación de algoritmos bioinspirados en la monitorización de vigas rotatorias

B. Muñoz-Abella¹, L. Rubio², P. Rubio³

Departamento de Ingeniería Mecánica, Grupo de Investigación SiOMec, Universidad Carlos III de Madrid

¹mmunoz@ing.uc3m.es ²lrubio @ing.uc3m.es ³prubio@ing.uc3m.es

Actualmente, el uso de las diferentes herramientas de Inteligencia Artificial (IA) cada vez está más extendido en diferentes ámbitos de la Ingeniería Mecánica. Entre las aplicaciones más frecuentes se encuentra su utilización para monitorización de fallos de elementos mecánicos.

Los algoritmos bioinspirados, encuadrados dentro de la IA, son métodos de optimización que, para su formulación, se inspiran en procesos biológicos o de la naturaleza. Entre ellos se encuentran, por ejemplo, los algoritmos genéticos, algoritmos de enjambre de hormigas o abejas, o aquellos basados en el comportamiento de otros animales, como conejos, cebras, ballenas, pelícanos, focas, etc.

En este trabajo se presenta la aplicación de este tipo de algoritmos para la monitorización de vigas rotatorias. El estudio de estos elementos tiene especial interés ya que, a menudo, se utilizan para estudiar de forma simplificada componentes muy habituales en la industria, tales como palas de aerogeneradores, palas de helicópteros o álabes de turbinas.

En definitiva, se han utilizado y comparado diferentes 6 tipos de algoritmos bioinspirados para detectar e identificar defectos en palas rotatorias, estudiando cuáles son los que ofrecen mejores resultados y analizando su viabilidad en la monitorización de este tipo de elementos. En concreto, se han estudiado algoritmos inspirados en el comportamiento de conejos, ballenas, águilas pescadoras, cebras y ranas, así como algoritmos genéticos.

A partir de los resultados obtenidos, para el problema analizado, se puede concluir que los algoritmos más adecuados son los basados en el comportamiento de los conejos y las cebras, así como los algoritmos genéticos. Sin embargo, en una primera aproximación, los inspirados en la conducta de ranas y ballenas no son adecuados para este tipo de problemas, este último, especialmente en el caso de fisuras incipientes.

1. Introducción

En numerosos sectores clave como el transporte y las industrias energéticas, existen componentes mecánicos que, en su forma simplificada, se pueden modelar como vigas rotatorias. Un ejemplo típico de ellos son las palas de los aerogeneradores o los rotores de los helicópteros. Estos componentes operan habitualmente en condiciones extremas de temperatura, entornos agresivos y cargas cíclicas, lo que, junto con pequeños defectos en su fabricación, puede provocar la propagación de fisuras y, en última instancia, la fractura catastrófica de los mismos. Uno de los métodos utilizados habitualmente para la detección e identificación de fisuras es la evaluación de las variaciones de las frecuencias naturales. La presencia de fisuras en componentes mecánicos se refleja, principalmente, en la disminución de las frecuencias naturales [1,2]. Sin embargo, en el caso de las vigas rotatorias fisuradas, este efecto se ve contrarrestado por otro fenómeno, el aumento de las frecuencias, que es consecuencia de la fuerza centrífuga generada por el giro [3-7]. La combinación de estos dos efectos complica la identificación y localización de los defectos [3,8,9].

Por otro lado, el planteamiento del problema inverso es fundamental para identificar los defectos y poder establecer adecuados planes de mantenimiento que eviten sobrecostos y daños personales. Este consiste en estimar profundidad y posición de las fisuras a partir variables características de su comportamiento dinámico, como, por ejemplo, el valor de las frecuencias naturales. En las dos últimas décadas se han empleado cada vez con más frecuencia las técnicas de Inteligencia Artificial (IA). Dentro de estas herramientas se encuentran los algoritmos bioinspirados, que para su formulación se inspiran en el comportamiento de la naturaleza [10]. Aunque estos algoritmos no necesitan obligatoriamente un modelo matemático del sistema para funcionar, si se dispone de él se mejora considerablemente la precisión de los resultados.

En este trabajo se presenta el estudio comparativo de diferentes algoritmos bioinspirados aplicados a la identificación de fisuras en vigas rotatorias esbeltas a baja velocidad de giro (hasta 10 rad/s). En primer lugar, se presenta la obtención de las expresiones que relacionan los parámetros de viga y fisura con las frecuencias naturales del sistema (problema directo), la cual permitirá abordar el problema inverso mediante la aplicación de los citados algoritmos. Esta contribución permitirá decidir si este tipo de algoritmos es conveniente para el problema de optimización planteado y, en su caso, determinar cuál de ellos es el más adecuado.

2. Formulación del problema directo

2.1. Modelo de viga fisurada rotatoria de Euler Bernoulli

Se analiza una viga rotatoria fisurada uniforme basada en el modelo de Euler-Bernoulli. La sección transversal de la viga, que presenta doble simetría, tiene forma rectangular, con una altura H y una anchura B . Su longitud total es L , y gira a una velocidad angular constante Ω alrededor del eje de revolución Z' . La fisura, de profundidad a , está ubicada a una distancia Lc desde la unión del cubo con la viga, punto que corresponde al origen del sistema de referencia XYZ , como se muestra en la Figura 1. En dicha figura también se señala el diámetro del cubo (D) acoplado a la viga. Cabe destacar que en este estudio únicamente se ha considerado el desplazamiento $W(X,t)$ en la dirección Z (*flapwise*), donde t representa el tiempo.

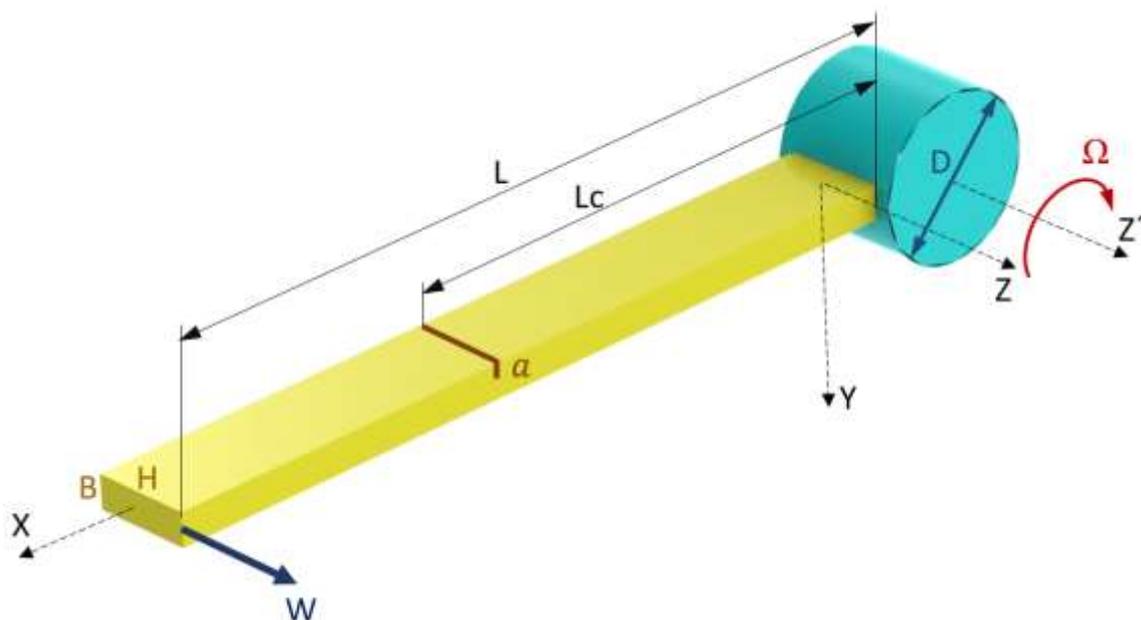


Figura 1: Esquema de la viga rotatoria fisurada.

La ecuación que gobierna el movimiento de una viga intacta similar a la descrita es dada por la expresión (1) [9]:

$$\rho A \frac{\partial^2 W(X,t)}{\partial t^2} + \frac{\partial^2}{\partial X^2} \left(EI \frac{\partial^2 W(X,t)}{\partial X^2} \right) - \frac{\partial}{\partial X} \left(P(X) \frac{\partial W(X,t)}{\partial X} \right) = 0 \quad (1)$$

donde ρ es la densidad del material, E es el módulo de Young, A es el área de la sección rectangular e I es el momento de inercia de la misma. $P(X)$ es la fuerza centrífuga que aparece en dirección axial (2):

$$P(X) = \int_X^L \rho A \Omega^2 (X + R) dX \quad (2)$$

La solución de la ecuación se aborda utilizando el método de separación de variables, de forma que:

$$W(X, t) = Lw(X)e^{i\omega t} \quad (3)$$

donde ω es la frecuencia natural.

Para la resolución del problema se han utilizado las variables adimensionalizadas:

$$\xi = \frac{X}{L} \quad r = \frac{D}{2L} \quad \alpha = \frac{a}{H} \quad \mu = \sqrt{\frac{\rho AL^4}{EI}} \omega \quad M = \sqrt{\frac{\rho AL^4}{EI}} \Omega \quad S_L = \sqrt{\frac{AL^2}{I}} \quad (4)$$

En el caso de la viga fisurada, esta se ha modelado mediante la unión 2 vigas intactas unidas entre sí mediante un muelle sin masa (5), cuya rigidez K es igual a la que introduce la fisura, según (6).

$$\begin{aligned} \frac{d^4 w_1}{d\xi^4} - M^2 \frac{d}{d\xi} \left(\left(r - 1 - \xi + \frac{1}{2} (1 - \xi^2) \right) \frac{dw_1}{d\xi} \right) - \mu^2 w_1 &= 0 \quad 0 \leq \xi \leq \xi_c \\ \frac{d^4 w_2}{d\xi^4} - M^2 \frac{d}{d\xi} \left(\left(r - 1 - \xi + \frac{1}{2} (1 - \xi^2) \right) \frac{dw_2}{d\xi} \right) - \mu^2 w_2 &= 0 \quad \xi_c \leq \xi \leq 1 \\ \frac{1}{K} &= \frac{6\pi(1-\nu^2)h}{EI} \Psi(\alpha) \\ \Psi(\alpha) &= \left[\begin{array}{l} 0.6272\alpha^2 - 0.04533\alpha^3 + 4.5948\alpha^4 - 9.9736\alpha^5 + 20.2948\alpha^6 - 33.031\alpha^7 + 47.1063\alpha^8 - \\ 40.7556\alpha^9 + 19.6\alpha^{10} \end{array} \right] \quad (6) \end{aligned}$$

Las condiciones de contorno de la viga (7) y las de compatibilidad en la sección fisurada (8), necesarias para resolver la ecuación diferencial son:

$$w_1(0) = 0 \quad \frac{dw_1(0)}{d\xi} = 0 \quad \frac{d^2 w_2(1)}{d\xi^2} = 0 \quad \frac{d^3 w_2(1)}{d\xi^3} = 0 \quad (7)$$

$$\begin{aligned} w_1(\xi_c) = w_2(\xi_c) \quad \frac{dw_2(\xi_c)}{d\xi} - \frac{dw_1(\xi_c)}{d\xi} &= \frac{1}{k_c} \frac{d^2 w_2(\xi_c)}{d\xi^2} \\ \frac{d^2 w_1(\xi_c)}{d\xi^2} = \frac{d^2 w_2(\xi_c)}{d\xi^2} \quad \frac{d^3 w_1(\xi_c)}{d\xi^3} &= \frac{d^3 w_2(\xi_c)}{d\xi^3} \quad (8) \end{aligned}$$

Donde k_c es:

$$k_c = \frac{K}{EI} \quad (9)$$

2.2. Solución al problema planteado. Expresiones cerradas de las frecuencias

La solución de las ecuaciones diferenciales se ha obtenido mediante la utilización del método de Frobenius, expresándola en forma de series de potencias [9,11], obteniendo las siguientes expresiones (10):

$$\begin{aligned} F_1 \xi &= C_1 w_1 \xi,0 + C_2 w_1 \xi,1 + C_3 w_1 \xi,2 + C_4 w_1 \xi,3 \quad 0 \leq \xi \leq \xi_c \\ F_2 \xi &= C_5 w_2 \xi,0 + C_6 w_2 \xi,1 + C_7 w_2 \xi,2 + C_8 w_2 \xi,3 \quad \xi_c \leq \xi \leq 1 \quad (10) \end{aligned}$$

donde C_n son los coeficientes de la serie de Frobenius calculados con las condiciones (7) y (8) y ξ_c es la posición de la fisura adimensionalizada.

El estudio se ha llevado a cabo con una viga fabricada en acero con un módulo de Young $E=210$ GPa, coeficiente de Poisson $\nu=0.3$ y densidad $\rho=7850$ kg/m³. En cuanto a las dimensiones, $L=0.7$ m y $B=0.01$ m. Además, para

obtener las expresiones cerradas de las 2 primeras frecuencias naturales adimensionalizadas, μ_1 y μ_2 , se han considerado la combinación de casos de 5 variables de entrada que se muestran en la Tabla 1.

Tabla 1: Valores de las variables de entrada para la obtención de las expresiones cerradas.

Variable	Rango de estudio
Esbeltez, S_L	70, 120, 170 y 220
Radio del cubo, r	0, 0.1, 0.2 y 0.3
Velocidad de rotación, Ω (rad/s)	0, 2, 4, 6, 8 y 10
Posición de la fisura, ξ_c	0.1 to 0.9 cada 0.1
Profundidad de la fisura, α	0, 0.1, 0.2, 0.3, 0.4 y 0.5

Teniendo en cuenta la combinación de todos los valores recogidos en la Tabla 2 se consiguen 4416 casos, 96 correspondientes a vigas intactas y 4320 a vigas fisuradas. Para los que se han calculado las 2 primeras frecuencias naturales adimensionales mediante la resolución de las ecuaciones diferenciales. A partir de esta colección de datos, se han obtenido las expresiones cerradas según el esquema mostrado en la Figura 2. Donde q representa 1 y 2, para la primera y segunda frecuencias, respectivamente.

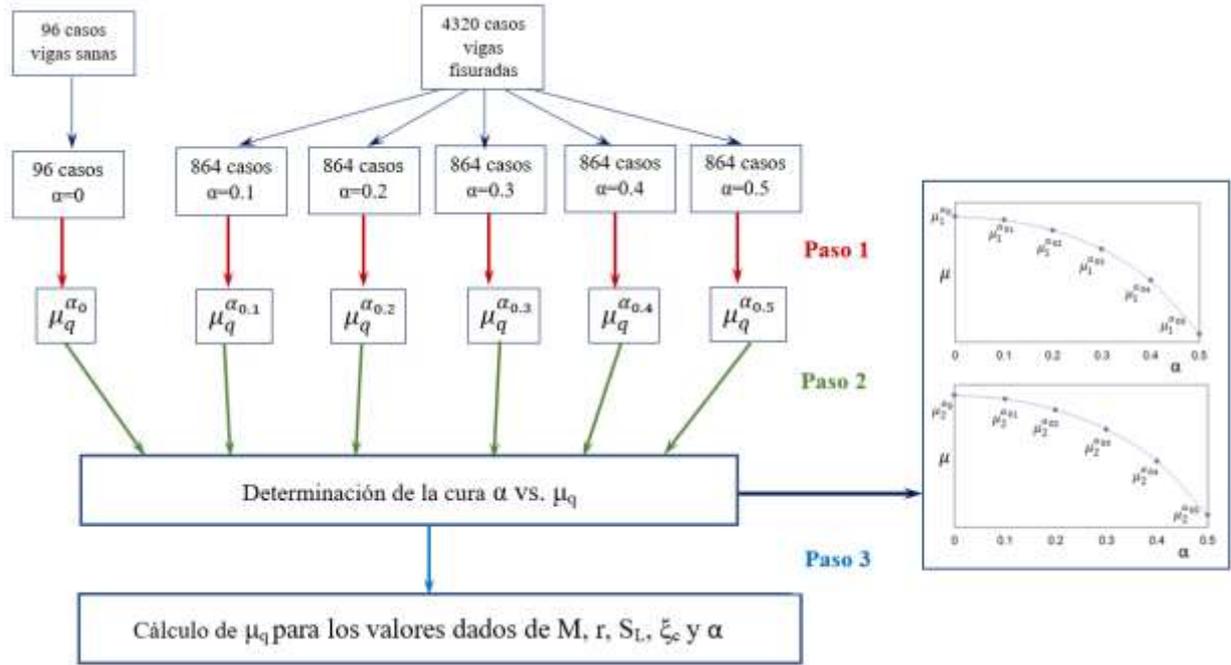


Figura 2: Procedimiento de obtención de las expresiones cerradas de μ_1 y μ_2 .

Las expresiones obtenidas en el paso 1 tienen una forma similar para las 2 frecuencias, de acuerdo con (11).

$$\mu_q^\alpha(M, r, S_L, \xi_c) = \sum_{i=0}^{i^*} \sum_{j=0}^{j^*} \sum_{k=0}^{k^*} \sum_{l=0}^{l^*} C_{ijkl}^\alpha \cdot M^i \cdot r^j \cdot S_L^k \cdot \xi_c^l \quad (11)$$

Donde i, j, k y l son los grados de los polinomios en M, r, S_L y ξ_c , respectivamente. Los coeficientes C_{ijkl} están recogidos en un repositorio público [12], mientras i^*, j^*, k^*, l^* son los valores de los mejores ajustes para cada caso [9].

Las expresiones de las curvas α vs. μ_q , calculadas en el paso 2 tienen la forma de (12).

$$\mu_q = \Phi_1^q \alpha^5 + \Phi_2^q \alpha^4 + \Phi_3^q \alpha^3 + \Phi_4^q \alpha^2 + \Phi_5^q \alpha^1 + \Phi_6^q \quad (12)$$

Donde Φ_1^q a Φ_6^q son 12 coeficientes desconocidos, 6 para cada una de las frecuencias, calculadas según la expresión (13). En definitiva, con el conjunto de expresiones mostradas, para cualquier conjunto de variables (M, r, S_L, ξ_c, α) se pueden calcular μ_1 y μ_2 .

3. Método inverso. Problema de optimización

Una vez resuelto el problema directo y obtenidas las expresiones que permiten obtener las frecuencias naturales en función de los parámetros del sistema y la fisura, se procede a la resolución del problema inverso que consiste en determinar las características de la fisura contenida en una viga (si existe) a partir del conocimiento de las

frecuencias naturales de la viga. Para lograr el objetivo se utilizarán 6 algoritmos de optimización bioinspirados diferentes: algoritmos genéticos (GA), algoritmo de conejos artificiales (ARO), algoritmo de ballenas (WOA), algoritmo de águilas pescadoras (OOA), algoritmo de cebras (ZOA) y algoritmo de ranas (SFLA), lo que permitirá compararlos y determinar cuál de ellos es el más adecuado para su aplicación en este tipo de problemas.

$$\begin{pmatrix} \Phi_1^q \\ \Phi_2^q \\ \Phi_3^q \\ \Phi_4^q \\ \Phi_5^q \\ \Phi_6^q \end{pmatrix} = inv \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0.10.1^2 & 0.1^3 & 0.1^4 & 0.1^5 & 0.1^6 \\ 0.20.2^2 & 0.2^3 & 0.2^4 & 0.2^5 & 0.2^6 \\ 0.30.3^2 & 0.3^3 & 0.3^4 & 0.3^5 & 0.3^6 \\ 0.40.4^2 & 0.4^3 & 0.4^4 & 0.4^5 & 0.4^6 \\ 0.50.5^2 & 0.5^3 & 0.5^4 & 0.5^5 & 0.5^6 \end{pmatrix} \begin{pmatrix} \mu_q^{\alpha_0} \\ \mu_q^{\alpha_{01}} \\ \mu_q^{\alpha_{02}} \\ \mu_q^{\alpha_{03}} \\ \mu_q^{\alpha_{04}} \\ \mu_q^{\alpha_{05}} \end{pmatrix} \quad (13)$$

Los algoritmos meta-heurísticos inspirados en la naturaleza resuelven problemas de optimización mediante la imitación de fenómenos biológicos o físicos. Estos algoritmos se pueden clasificar en tres categorías principales: métodos basados en la evolución, métodos basados en la física y métodos basados en enjambres. Los métodos basados en la evolución se fundamentan en los principios de la selección natural y los procesos evolutivos que ocurren en la naturaleza. El proceso de búsqueda comienza con una población de soluciones generada aleatoriamente, que representa las posibles soluciones al problema de optimización en cuestión. En cada generación, los individuos más "aptos" (aquellos que presentan mejores soluciones) son seleccionados para reproducirse. Este proceso de selección y reproducción permite que las soluciones más eficientes se combinen, generando una nueva generación de individuos. El ciclo de selección, reproducción y mutación se repite a lo largo de varias generaciones, lo que lleva a la mejora continua de la población de soluciones. A medida que el proceso evoluciona, la población se aproxima cada vez más a una solución óptima. La ventaja de los métodos basados en la evolución radica en su capacidad para explorar el espacio de soluciones de manera efectiva, aprovechando la combinación de los mejores individuos y la introducción de pequeñas variaciones aleatorias (mutaciones), lo que permite encontrar soluciones globalmente óptimas. Dentro de estos los más conocidos y utilizados en para resolver problemas en ingeniería son los algoritmos genéticos (GA) [13].

Otros métodos que se inspiran en la naturaleza son aquellos que se basan en el comportamiento de enjambres, imitando cómo interactúan socialmente ciertos grupos de animales. En estos métodos, se emplea un grupo de individuos (soluciones candidatas) que exploran un espacio de búsqueda en busca de la mejor solución posible (la posición óptima). A medida que avanzan, cada uno de ellos busca el mejor lugar (la mejor solución) entre todas las opciones disponibles. En la búsqueda, cada individuo tiene en cuenta tanto sus propias mejores soluciones como la mejor solución que ha encontrado el grupo hasta ese momento. Los algoritmos de optimización meta-heurísticos que se fundamentan en poblaciones presentan una característica común, independientemente de su especificidad. Este proceso de búsqueda se articula en dos fases fundamentales: exploración y explotación. Durante la fase de exploración, el algoritmo incorpora formulaciones que permiten una búsqueda global en el espacio de soluciones. En esta etapa, los movimientos, que implican la perturbación de las variables de diseño, deben ser lo más aleatorios posible para garantizar una cobertura amplia del espacio de búsqueda. Posteriormente, se inicia la fase de explotación, que se define como el proceso de análisis detallado de las áreas prometedoras identificadas durante la fase de exploración. En la fase de explotación se realiza una búsqueda local en las regiones del espacio de diseño que han demostrado ser más prometedoras. Para este estudio se han escogido, además de los algoritmos genéticos, cinco algoritmos basados en el comportamiento de animales y que se detallan a continuación.

El algoritmo de conejos artificiales (ARO: Artificial Rabbit Optimization) se basa en el comportamiento de los conejos durante la búsqueda de comida y durante la búsqueda de escondite. El objetivo es encontrar el individuo que sobrevive y mejor se adapta. En el caso del algoritmo ARO, las etapas de funcionamiento son dos: exploración (búsqueda del alimento) y explotación (búsqueda de la madriguera). La formulación matemática del algoritmo se puede encontrar, entre otros, en el trabajo de Wang y colaboradores [14] y su aplicación en un trabajo previo de las autoras [15]. Por su parte, el algoritmo de las ballenas (WOA: Whale Optimization Algorithm), se basa en el comportamiento de las ballenas para su alimentación. Las etapas en las que se divide el algoritmo son tres: la identificación y cerco de las presas, las maniobras de alimentación con redes de burbujas y, por último, la búsqueda de las presas. El objetivo en este caso es encontrar el mejor espacio. El desarrollo matemático del algoritmo se encuentra, por ejemplo, en el trabajo pionero de Mirjalili y colaborador [16]. Otro algoritmo que se puede utilizar en problemas de optimización en ingeniería es el de las águilas o halcones pescadores (OOA: Osprey Optimization Algorithm). Este algoritmo, muy reciente, se basa en el comportamiento de estas aves para su alimentación. En este caso, el algoritmo tiene dos etapas fundamentales: la localización en vuelo de los peces y su caza, y la posterior búsqueda de un lugar seguro para su alimentación [17]. Por su parte, el algoritmo de las cebras (ZOA: Zebra Optimization Algorithm), se basa en el comportamiento de estos mamíferos africanos para su supervivencia mientras se alimentan. Así, una vez localizada la zona de alimentación, según el depredador, establecen una estrategia de huida, si el depredador es grande (por ejemplo un león), o bien, se enfrentan como manada a él, si es pequeño (por ejemplo si son hienas). El objetivo es alimentarse y sobrevivir. El desarrollo de este algoritmo,

también bastante reciente, se encuentra en el trabajo de Trojovská y colaboradores [18]. En el caso del algoritmo de la rana saltarina (SFLA: Shuffle Frog Leaping Algorithm), el objetivo es encontrar individuos que se acerquen a la piedra en la que hay mayor cantidad de alimento saltando de piedra en piedra de la manera más óptima posible. Para ello las ranas se agrupan y aprenden de la estrategia de los distintos grupos saltando de unas posiciones a otras mejores. Este algoritmo fue desarrollado por primera vez por Eusuff y colaboradores [19]. En la Figura 3 se pueden ver los esquemas gráficos de 3 de los algoritmos utilizados.

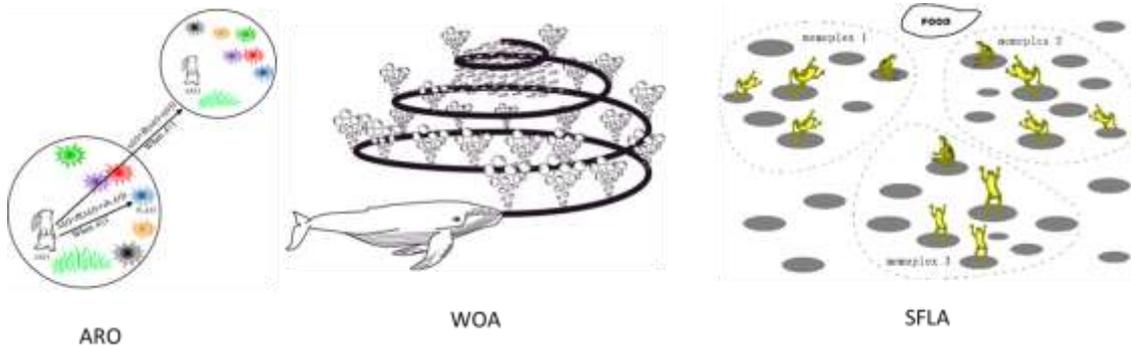


Figura 3: Ejemplos de algoritmos de optimización bioinspirados. [14,16,20]

Como se ha explicado anteriormente, el problema inverso se ha abordado con la aplicación de 6 algoritmos diferentes para poder realizar la comparación entre ellos. En todos los casos se han elegido los mismos 10 casos aleatorios, con valores de las variables de entrada diferentes de los utilizados en la determinación de las expresiones cerradas descritas en el apartado 2. En la Tabla 2 se muestran los valores de las variables de entrada y salida de estos 10 casos.

Tabla 2: Valores de las variables de entrada y salida para los casos utilizados en la comparación de los algoritmos

Caso	ENTRADAS PROBLEMA INVERSO					SALIDAS REALES PROBLEMA INVERSO	
	Ω (rad/s)	r	S_L	μ_1	μ_2	ξ_c	α
1	5.4	0.12	85	3.478	22.035	0.21	0.17
2	9.6	0.23	210	3.541	21.661	0.58	0.34
3	5.8	0.08	132	3.512	21.344	0.68	0.42
4	4.6	0.27	113	3.517	21.810	0.78	0.37
5	2.1	0.12	75	3.476	21.479	0.43	0.27
6	3.4	0.01	216	3.519	22.035	0.89	0.16
7	8.5	0.17	98	3.222	22.036	0.22	0.48
8	1.5	0.25	106	3.515	21.899	0.77	0.27
9	6.8	0.16	121	3.518	22.016	0.38	0.08
10	7.3	0.07	89	3.237	21.747	0.14	0.41

Aunque cada algoritmo tiene su propia formulación, ha habido parámetros que se han mantenido iguales en todos los casos. En concreto, el número de individuos de la población es de 50 y el número máximo de iteraciones 200. Por otra parte, la función objetivo f , común en todos los casos, tiene la forma de la expresión (14).

$$f = \sqrt{(\mu_1^{real} - \mu_1^{estimado})^2 + (\mu_2^{real} - \mu_2^{estimado})^2} \quad (14)$$

Donde μ_1^{real} y μ_2^{real} son las frecuencias naturales adimensionalizadas 1 y 2, respectivamente, los cuales son datos de entrada conocidos, medidos, por ejemplo, si se dispusiera de un sistema real. Por otra parte, $\mu_1^{estimado}$ y $\mu_2^{estimado}$ son las mismas variables, pero estimadas en cada iteración del algoritmo. Estas frecuencias estimadas son función de la posición (ξ_c) y el tamaño (α) de la fisura, parámetros que se obtienen al final del proceso de optimización, cuyo objetivo es minimizar la función objetivo f .

4. Resultados

En este apartado se explica la comparación de los algoritmos bioinspirados utilizados para abordar el problema inverso. Para ello se han comparado diferentes parámetros obtenidos a partir de su utilización en el problema planteado, estos son la evolución de la función objetivo a lo largo de las iteraciones, la bondad del ajuste entre los valores reales y los estimados y, por último, los valores de los errores cuadráticos medios.

Como ejemplo ilustrativo del resto de casos, en la Figura 4 se muestra la evolución de la función objetivo a lo largo de las 200 iteraciones para el caso 5 y todos los algoritmos empleados. En este ejemplo se observa que los

algoritmos que peor comportamiento tienen en cuanto a minimización de la función objetivo son los que imitan a ballenas y ranas, mientras que el resto consiguen valores de la función objetivo próximos a cero.

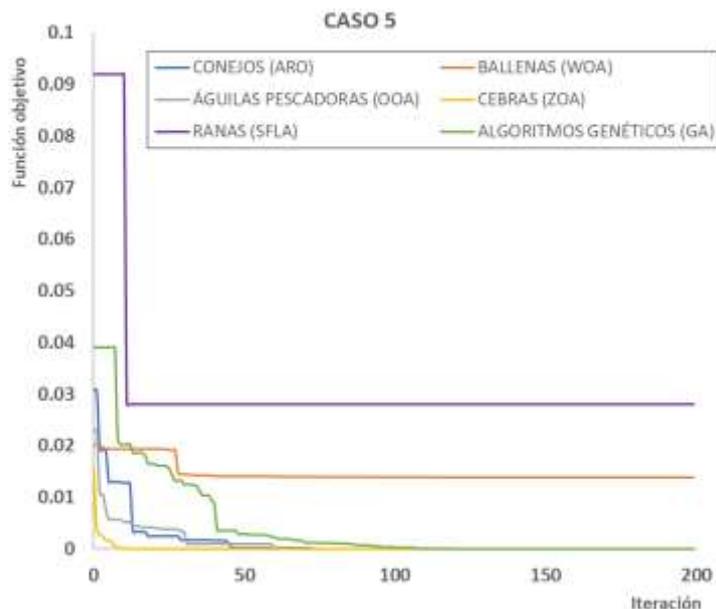


Figura 4: Evolución de la función objetivo para el caso 5 y todos los algoritmos comparados

En la Figura 5 se pueden ver las gráficas, para los 6 algoritmos, de los valores reales de la posición de la fisura (ξ_c) frente a los estimados. En la Figura 6 se pueden ver figuras similares para la profundidad del defecto (α). En todas ellas se muestra la línea que representaría la estimación perfecta, y el valor de R^2 del ajuste entre los valores reales y los estimados.

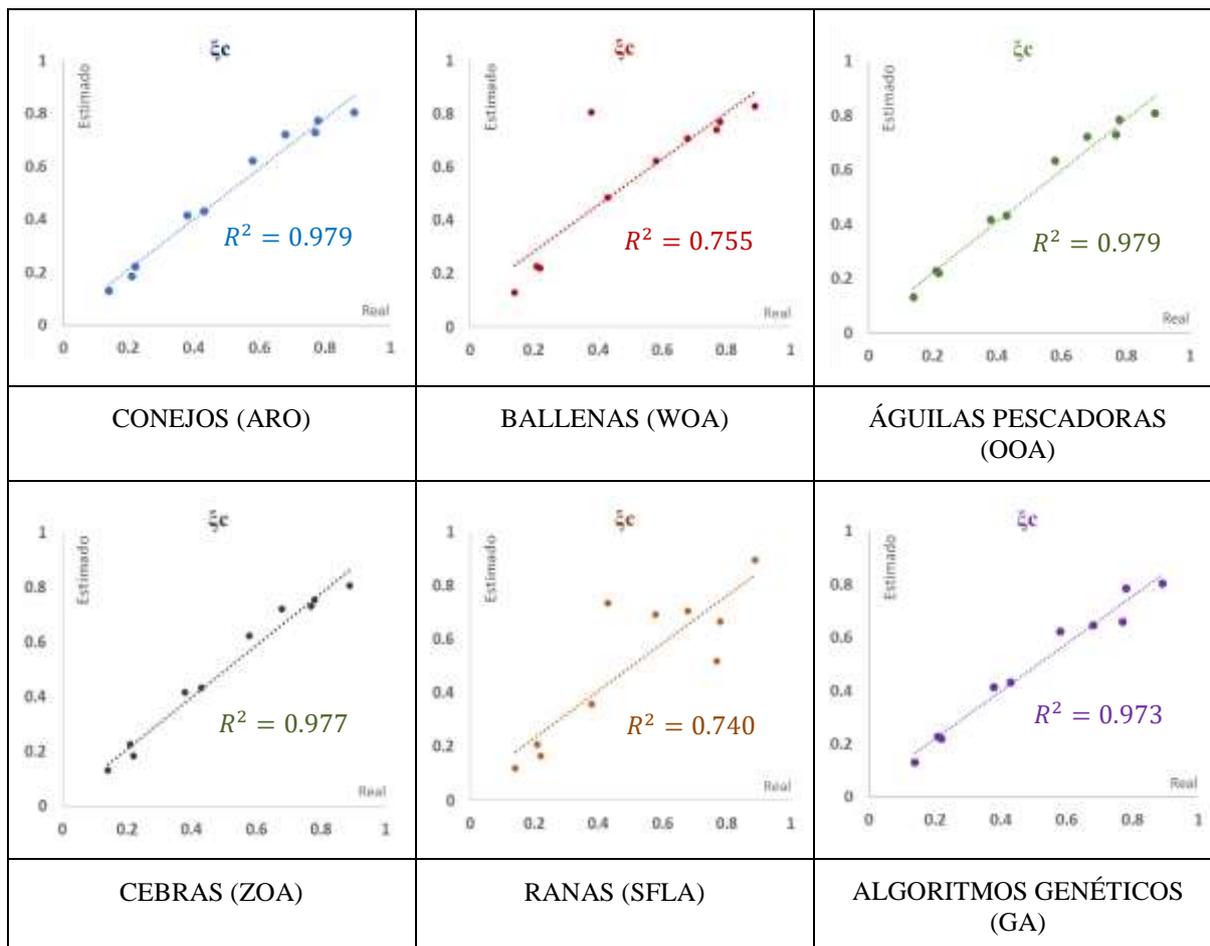


Figura 5: Valores reales de la posición de la fisura (ξ_c) frente a los estimados

Los resultados de estimación de la posición de la fisura muestran que hay un grupo de algoritmos formados por los que imitan la conducta de conejos, águilas pescadoras, cebras y algoritmos genéticos que, para los 10 casos utilizados en la comparación, tienen un comportamiento en el ajuste similar, con valores de R^2 muy aceptables, en torno a 0.97. Para el algoritmo basado en el comportamiento de las ballenas existe un dato en el que la estimación no es buena (caso 9 que se corresponde con una fisura incipiente) que hace que R^2 sea de 0.74. Sin embargo, sin tener en cuenta este caso, el valor de R^2 sube a 0.9845. Por último, el algoritmo basado en la conducta de las ranas, en general, no ofrece buenos resultados en el ajuste, con $R^2=0.74$.

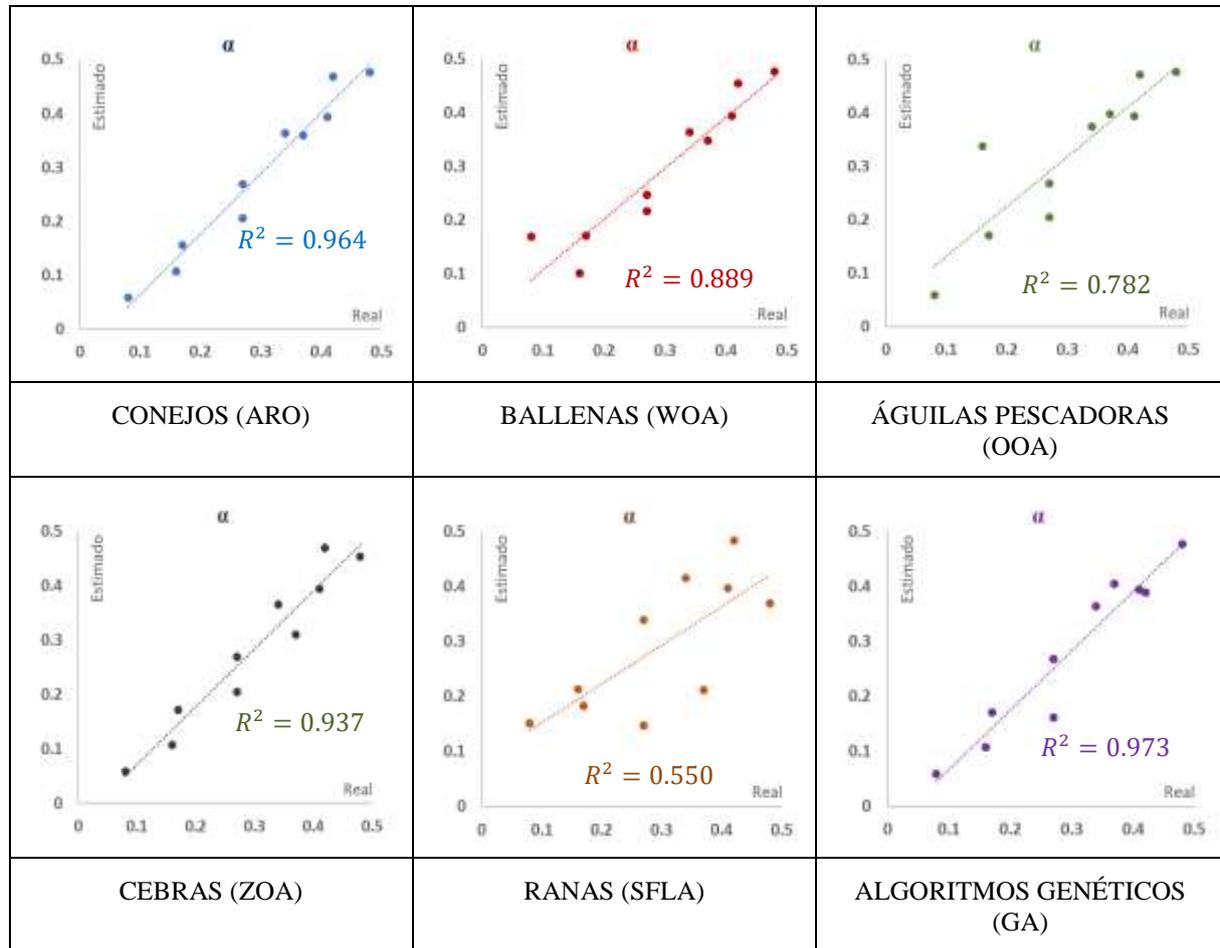


Figura 6: Valores reales del tamaño de la fisura (α) frente a los estimados

Según se observa en las gráficas, se puede decir que, en general, la estimación del tamaño de la fisura es peor que la de su posición. Los algoritmos que consiguen mejor ajuste para los casos analizados son los que se basan en el comportamiento de conejos, cebras y algoritmos genéticos. Todos ellos tienen valores de R^2 superiores a 0.9. El basado en el comportamiento de ballenas es ligeramente peor, con $R^2=0.889$ e, igual que sucedía en la estimación de la posición, el algoritmo inspirado en ranas no ofrece resultados aceptables, con $R^2=0.55$.

Por último, en la Tabla 3 se puede ver el valor del Error Cuadrático Medio (ECM) para cada uno de los ajustes, calculados según la expresión (15).

$$ECM = \frac{1}{n} \sqrt{\sum_{i=1}^{i=n} (X_i^{real} - X_i^{estimado})^2} \quad (15)$$

Donde X es ξ_c ó α y $n=10$ (número de casos).

Tabla 3: Error Cuadrático Medio para cada algoritmo y variable de salida

Algoritmo	ECM(ξ_c)	ECM(α)
CONEJOS (ARO)	0.0014	0.0011
BALLENAS (WOA)	0.0193	0.0017
ÁGUILAS PESCADORAS (OOA)	0.0015	0.0041
CEBRAS (ZOA)	0.0016	0.0015
RANAS (SFLA)	0.0187	0.0075
ALGORITMOS GENÉTICOS (GA)	0.0024	0.0018

Para la estimación de la posición de la fisura de los casos analizados, los menores errores se producen en el caso de algoritmos inspirados en conejos, águilas pescadoras y cebras, en el caso de los algoritmos genéticos el error es aproximadamente el doble de los anteriores, aunque es aceptable. Sin embargo, los algoritmos basados en el comportamiento de las ballenas y las ranas ofrecen errores 10 veces más grandes que los primeros. Hay que tener en cuenta, que, como se explicó anteriormente, en el caso del algoritmo basado en ballenas hay un solo dato, de una fisura muy incipiente, que hace que el error aumente de forma considerable. En cuanto al error en la estimación de la profundidad de la fisura, los que tienen errores menores son los algoritmos genéticos y los inspirados en conejos y ballenas, el algoritmo basado en águilas pescadoras ofrece un error en torno a 3 veces más grande, mientras que el que imita la conducta de las ranas es el que peor se comporta, con un error en torno a 7 veces el de los primeros. En definitiva, atendiendo al ECM de la estimación de profundidad y posición, el algoritmo que mejor se comporta para los casos estudiados es el basado en el comportamiento de los conejos.

5. Conclusiones

En este trabajo se presenta la comparación de diversos algoritmos bioinspirados en la aplicación al problema inverso de detección e identificación de fisuras en vigas rotatorias a baja velocidad.

En concreto se han comparado los algoritmos inspirados en el comportamiento de conejos, ballenas, águilas pescadoras, cebras y ranas, más algoritmos genéticos, en total 6 tipos de algoritmos diferentes. Para conseguir este objetivo, en primer lugar, se han obtenido las expresiones que relacionan los parámetros geométricos y de funcionamiento de la viga (velocidad de giro, esbeltez de la viga y radio del cubo) junto con las características de la fisura (posición y profundidad) con las 2 primeras frecuencias naturales del elemento. En segundo lugar, a partir de estas expresiones, se ha abordado, mediante la utilización de los 6 algoritmos bioinspirados, el problema de optimización que permite obtener las características de la fisura.

Aplicando cada uno de ellos a 10 casos elegidos aleatoriamente se puede concluir que, para el problema planteado y los casos estudiados, el algoritmo que se inspira en el comportamiento de los conejos parece el más adecuado, junto con el inspirado en cebras y los algoritmos genéticos. Aunque hay que tener en cuenta que el número de casos estudiado es limitado, el algoritmo basado en la conducta de las ballenas también ofrece buenos resultados, aunque para casos en que la fisura es incipiente no se comporta de manera adecuada. Por último, los algoritmos inspirados en el comportamiento de águilas pescadoras y ranas, para el problema y los casos estudiados, no son adecuados, ya que no se obtienen estimaciones suficientemente buenas.

En definitiva, se puede concluir que, independientemente del algoritmo bioinspirado que se utilice, ya sean los basados en el comportamiento de animales o en otros procesos de la naturaleza, cada vez están alcanzando más importancia, convirtiéndose en una poderosa herramienta para abordar problemas de optimización en el campo de la detección e identificación de defectos y, en general, en el ámbito de la Ingeniería Mecánica.

6. Referencias

- [1] Fernández-Sáez J., Rubio L., Navarro C., “Approximate calculation of the fundamental frequency for bending vibrations of cracked beams”, *Journal of Sound and Vibration* **225**(2), 345-352 (1999)
- [2] Loya JA., Rubio L., Fernández-Sáez J., “Natural frequencies for bending vibrations of Timoshenko cracked beams”, *Journal of Sound and Vibration*, **290**(3-5), 640-653 (2006)
- [3] Lee J.W., Lee J.Y., “In-plane bending vibration analysis of a rotating beam with multiple edge cracks by using the transfer matrix method”, *Meccanica*, **52**, 1143–1157 (2017)
- [4] Wauer J., “Dynamics of cracked rotating blades”, *Applied Mechanics Reviews*, **44**, 273-278 (1991)
- [5] Chen J L.W., Chen C.L., “Vibration and stability of cracked thick rotating blades”, *Computers & Structures* **28**, 67-74 (1988)
- [6] Yashar A., Ferguson N., Ghandchi-Tehrani M., “Simplified modelling and analysis of a rotating Euler-Bernoulli beam with a single cracked edge”, *Journal of Sound and Vibration*, **420**, 346-356 (2018)
- [7] Banerjee A., Pohit G., “Crack detection in rotating cantilever beam by continuous wavelet transform”, *Applied Mechanics and Materials* **594**, 2021-2025 (2014)
- [8] Valverde-Marcos B., Muñoz-Abella B., Rubio P., Rubio L., “Influence of the rotation speed on the dynamic behaviour of a cracked rotating beam”, *Theoretical and Applied Fracture Mechanics*, **117**, 103209 (2022)
- [9] Muñoz-Abella, B., Rubio, L., Rubio, P., “Closed-Form Solution for the Natural Frequencies of Low-Speed Cracked Euler–Bernoulli Rotating”, *Mathematics*, **10**:4742 (2022)

- [10] Abualigah, L. et al., “Meta-heuristic optimization algorithms for solving real-world mechanical engineering design problems: a comprehensive survey, applications, comparative analysis, and results”, *Neural Computing and Applications* **34**:4081–4110 (2022)
- [11] Muñoz-Abella, B., Rubio, L., Rubio, P., “Detección e identificación de fisuras en vigas rotatorias esbeltas a baja velocidad mediante parámetros modales y redes neuronales artificiales”, *Revista Iberoamericana de Ingeniería Mecánica* **27**(2): 61-73 (2023)
- [12] Muñoz-Abella, B., Rubio, L., Rubio, P., “Coefficients low-speed EB cracked rotating beam” <https://zenodo.org/records/7404726> (2022)
- [13] Muñoz-Abella, B., Rubio, L., Rubio, P., “Two-stage crack identification in an Euler-Bernoulli rotating beam using modal parameters and Genetic Algorithm”, *Smart Structures and Systems* **33**(2): 165-175 (2024)
- [14] Wang L., Cao Q., Zhang Z., Mirjalili S., Zhao W., “Artificial rabbits optimization: A new bio-inspired meta-heuristic algorithm for solving engineering optimization problems”, *Engineering Applications of Artificial Intelligence* **114**, 105082 (2022)
- [15] Muñoz-Abella B., Rubio L., Rubio P., “Identification of cracks in low-speed rotating slender cracked beams using frequencies and artificial rabbit algorithm”, *Latin American Journal of Solids and Structures* **21**(2), e530 (2024)
- [16] Mirjalili S., Lewis A., “The Whale Optimization Algorithm”, *Advances in Engineering Software*, **95**, 57-65 (2016)
- [17] Dehghani M., Trojovský P., “Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems”, *Frontiers in Mechanical Engineering*, **8**, 1126450 (2023)
- [18] Trojovska, E., Dehghani M., Trojovský P., “Zebra Optimization Algorithm: A New Bio-Inspired Optimization Algorithm for Solving Optimization Algorithm”, *IEEE Access*, **10**, 49445-49473 (2022)
- [19] Eusuff, MM., Lansey, KE. “Optimization of water distribution network design using the Shuffled Frog Leaping Algorithm”, *Journal of Water Resources Planning and Management*, **129** (3) , 210-225, (2003)
- [20] Zhao Z., Wang M., Liu Y., Chen Y., He, K., Liu Z., “A modified shuffled frog leaping algorithm with inertia weight”, *Scientific Reports*, **14**, 4146 (2024)