

## Optimización topológica con el empleo de la metaheurística de enjambre de partículas

Topological optimization using particles  
swarm metaheuristic

Carlos Millán-Páramo\*  
Euriel Millán-Romero\*\*  
*Universidad de Sucre (Colombia)*

---

\* Magíster en Ingeniería Civil, Profesor Asociado, Universidad de Sucre (Sincelejo, Colombia), Facultad de Ingeniería. [carlos.millan@unisucreeu.co](mailto:carlos.millan@unisucreeu.co)

\*\* Magíster en Suelos, Profesor Titular, Universidad de Sucre (Sincelejo, Colombia), Facultad de Ingeniería. [euriel.millan@unisucreeu.co](mailto:euriel.millan@unisucreeu.co)

**Correspondencia:** Carlos Millán-Páramo. Carrera 28 N°5-267, Puerta Roja, Sincelejo, Sucre, (Colombia). Teléfono: +57 2771195

## Resumen

Este trabajo propone la implementación de la metaheurística PSO (optimización con enjambre de partículas) a fin de sustituir el criterio de optimalidad utilizado dentro del método de optimización topológica propuesto por Andreassen. Con el propósito de evaluar y validar el desempeño de la técnica planteada se analizaron tres problemas de elasticidad plana reportados en la literatura especializada. Cada problema se analizó mediante el empleo del método de elementos finitos con tres tipos de mallas diferentes, con el objeto de comparar los resultados obtenidos en cuanto a topologías, el valor de energía de deformación y los tiempos de ejecución. Se logró establecer que el procedimiento que involucra la PSO arroja menores tiempos computacionales a medida que se analizan los problemas con mallas más refinadas. Finalmente, las distribuciones de material en el dominio de diseño y los valores de energía obtenidos fueron similares a los reportados en el trabajo de Andreassen, lo que dio validez a la propuesta aquí presentada.

**Palabras claves:** distribución de material, optimización con enjambre de partículas, optimización topológica.

## Abstract

This work proposes the implementation of the PSO (*particle swarm optimization*) metaheuristic to replace the optimality criterion used in the topology optimization method proposed by Andreassen. To evaluate and validate the PSO performance we studied three plane elasticity problems reported in the literature. Each problem was analyzed with three different finite element mesh types in order to compare the results obtained in terms of topology, strain energy value and average runtimes. It was established that the procedure involving the PSO, yields lower computational times in problems with more refine meshes. Finally, the material distribution and the energy values obtained were similar to those reported in the work of Andreassen giving validity to the work presented here.

**Keywords:** material distribution, particle swarm optimization, topological optimization.

Fecha de recepción: 6 de julio de 2017  
Fecha de aceptación: 7 de abril de 2018

## I. INTRODUCCIÓN

La optimización topológica (OT) consiste en buscar una distribución óptima de material en un dominio de diseño que satisfaga las solicitaciones y las condiciones de borde establecidas. La OT es un campo de investigación de rápido crecimiento, en el cual intervienen distintas áreas como, por ejemplo, las matemáticas, la mecánica y las ciencias computacionales; cuenta, además, con importantes aplicaciones prácticas en la industria y en el sector de manufactura. En la actualidad, la OT se utiliza en las industrias de obras civiles, aeroespaciales y automotrices, entre otras.

Generalmente, la OT se ha formulado en términos de minimizar la energía de deformación de la estructura analizada. En la mayoría de trabajos sobre OT reportados se encuentra que los métodos de optimización comúnmente empleados son: el criterio de optimalidad estándar (*optimality criteria*) [1], [2], el método de la curva de nivel (*level set method*) [3], y la eficiencia de Pareto (pareto optimal tracing) [4], entre otros. Sin embargo, se debe recordar que la OT es un problema con múltiples mínimos locales susceptible de solución a través de métodos metaheurísticos diseñados para identificar mínimos globales. Por tanto, se propone en este estudio emplear el algoritmo de optimización con enjambre de partículas (PSO) en su solución. De forma general, todas estas técnicas exploran el espacio de búsqueda de una manera controlada y tienen la ventaja de no depender del cálculo de derivadas para llevar a cabo el proceso de optimización.

En su primera parte, este trabajo presenta la metodología y los ejemplos numéricos a analizar. Luego, se describe el PSO, sus fundamentos y los parámetros que lo controlan, así como su evaluación y el desempeño en funciones de pruebas (*benchmark problems*) con y sin restricciones. Finalmente, se muestran los resultados obtenidos con la implementación de PSO en la resolución de problemas de OT y se comparan con resultados de la literatura internacional.

## II. METODOLOGÍA

Con el fin de obtener la distribución óptima de material para cada una de las estructuras, se procedió a discretizar el dominio de diseño mediante elementos finitos tipo C4 (cuadrilátero bilineal), y a cada elemento (e) se le asignó una densidad  $x_e$ . Posteriormente, se empleó la ecuación (1) definida

en el método SIMP modificado [1] con el propósito de definir el módulo de elasticidad de cada elemento.

$$E_e \cdot E_e(x_e) = E_{\min} + x_e^p (E_0 - E_{\min}) \quad (1)$$

Donde  $x_e \in [0,1]$  es la densidad del material,  $E_{\min}$  es un valor de densidad muy pequeño asignado para anular elementos con el fin de que la matriz de rigidez no se convierta en singular, y  $p$  es un factor de penalización (generalmente  $p = 3$ ), introducido con el propósito de garantizar soluciones blanco y negro (*black-and-white*). La formulación matemática del problema de optimización se basa en una ley potencial, en la cual el objetivo es minimizar la energía de deformación y se describe así [1]:

$$\min_x c(x) = U^T K U = \sum_{e=1}^N E_e(x_e) u_e^T k_0 u_e \quad (2)$$

Sujeto a:

$$V_{(x)}/V_0 = f \quad (3)$$

$$K U = F \quad (4)$$

$$0 \leq x \leq 1 \quad (5)$$

Donde  $c$  es la energía de deformación a minimizar,  $U$  y  $F$  son los vectores de desplazamiento y fuerzas globales, respectivamente,  $K$  es la matriz de rigidez global,  $u_e$  es el vector desplazamiento del elemento,  $E_e$  está dado por la ecuación 1,  $k_0$  es la matriz de rigidez del elemento,  $x$  es el vector de variables de diseño (es decir, las densidades de los elementos),  $N$  es el número de elementos usados para discretizar el dominio de diseño,  $V_{(x)}$  y  $V_0$  y son el volumen del material y el volumen del dominio de diseño, respectivamente, y  $f$  es la fracción de volumen. La fracción de volumen determina la máxima cantidad de volumen permitido en la topología final.

La energía de deformación aumenta a medida que la estructura se deforma, por tanto, el proceso de optimización consiste en hallar el conjunto de valores  $x_e$  que la minimizan, lo que justifica el uso de las técnicas metaheurísticas

para su solución. En la solución del problema de optimización se usó el algoritmo PSO, el cual ofrece una forma simple, sencilla y precisa de resolver modelos complejos con un algoritmo básico de fácil implementación y alta robustez. El PSO es una técnica basada en poblaciones en las que los individuos, denominados "partículas", se mueven en el espacio solución de una función objetivo (más detalles en la siguiente sección).

Con el objeto de evitar problemas de inestabilidad numérica el algoritmo de OT emplea una técnica de filtrado de densidad, descrito por Andreassen [1], el cual produce un alisado de las energías de deformación de los elementos al tener en cuenta el valor de la energía de los elementos vecinos y así encontrar soluciones razonables. Con el fin de especificar los elementos vecinos (vecindario), se define el área de influencia con el empleo de un radio (equivale a un porcentaje del ancho del elemento 3-4 %), a fin de seleccionar el vecindario de cada elemento.

La estructura básica del algoritmo propuesto por Andreassen con la adición propuesta en este trabajo (véase el punto 6) se presenta a continuación:

1. Diseño inicial. Se realiza una distribución homogénea del material en la geometría de trabajo.
2. Se calcula la matriz de rigidez global teniendo en cuenta la actual distribución del material.
3. Se calcula mediante elementos finitos el campo de desplazamientos para cierto estado de cargas.
4. Se calcula la energía de deformación con respecto la variable de diseño (densidades).
5. Se aplica un filtro con el objeto de encontrar soluciones menos pixeladas y más continuas, y así evitar problemas de inestabilidad mencionados anteriormente.
6. Finalmente, mediante el PSO, se calculan las nuevas densidades.
7. Se vuelve al paso 2.

De esta forma, el proceso continúa iterando hasta un punto en que el valor de las densidades no cambia significativamente y el bucle termina.

## A. Problemas propuestos de OT

A fin de evaluar el comportamiento del PSO en problemas de OT, se analizaron tres problemas reportados en la literatura (Figura 1).

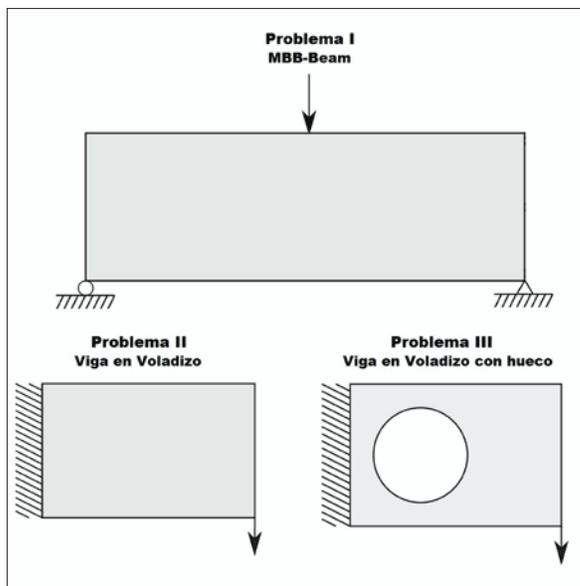


Figura 1. Problemas de optimización [1]

## III. OPTIMIZACIÓN CON ENJAMBRE DE PARTÍCULAS

La optimización con enjambre de partículas fue propuesta, inicialmente, por Kennedy y Eberhart [5] y nace, al igual que otras técnicas estocásticas de cálculo evolutivo, en un intento por imitar y mimetizar el comportamiento de procesos naturales.

En la terminología empleada por PSO se introduce el término general de partículas o agentes a fin de representar a cualquier tipo de individuos que exhiban un comportamiento social, de forma que interactúan entre sí. De acuerdo con los fundamentos teóricos del método, el movimiento de cada una de estas partículas hacia un objetivo común en un espacio de búsqueda está condicionado por dos factores básicos: la memoria autobiográfica de la partícula y la influencia social de todo el enjambre. A nivel computacional, como método de optimización esta filosofía puede extenderse a un

espacio N-dimensional, de acuerdo con el problema en análisis. La posición instantánea de cada una de las partículas de la población en el espacio N-dimensional representa una solución potencial, de manera que es N el número de incógnitas del problema original. Básicamente, el proceso evolutivo se reduce a mover cada partícula dentro del espacio de soluciones con una velocidad que variará de acuerdo con su velocidad actual, la memoria de la partícula y la información global que comparte el resto del enjambre.

### A. Parámetros del algoritmo

En la formulación del PSO, Kennedy y Eberhart [5] definen la velocidad de la partícula como el único operador disponible que permite controlar la evolución de la optimización, en el que cada partícula del enjambre se identifica con dos variables: un vector velocidad ( $V$ ), y un vector de posición ( $X$ ), en el cual este último corresponde a una solución potencial al problema de optimización. Los límites de los parámetros a optimizar conforman el espacio de búsqueda al cual debe restringirse el movimiento del enjambre; adicionalmente, cada partícula mantiene en la memoria información de la posición espacial asociada con la mejor solución encontrada por ésta ( $Pbest$ ) y la posición de la mejor solución encontrada por todos sus congéneres ( $Gbest$ ). De esta manera, la primera versión del algoritmo PSO está formulada como:

$$V_{i+1}^d = V_i^d + c_1 \times rand1_i^d \times (Pbest_i^d - X_i^d) + c_2 \times rand2_i^d \times (Gbest_i^d - X_i^d) \quad (6)$$

$$X_{i+1}^d = X_i^d + V_{i+1}^d \quad (7)$$

Donde  $V_i^d$  y  $X_i^d$  corresponden a la velocidad y a la posición de la partícula  $i$  en la dimensión  $d$  respectivamente, ( $Pbest$ ) representa la mejor posición dentro del espacio de búsqueda alguna vez visitada por la partícula  $i$ , ( $Gbest$ ) es la mejor posición dentro del enjambre encontrada hasta el momento, los coeficientes  $C_1$  y  $C_2$  indican el grado de confianza en la mejor posición encontrada por la partícula individual (parámetro cognitivo) y la de todo el enjambre (parámetro social), respectivamente; por su parte,  $rand1$  y  $rand2$  son dos números aleatorios distribuidos uniformemente en el intervalo  $[0,1]$ , cuyo objetivo es emular el comportamiento estocástico y un tanto impredecible que exhibe la población del enjambre. Después de calcular la nueva velocidad de la partícula, la nueva posición se actualiza de acuerdo con la ecuación (7).

Con el objeto de reducir efectos de convergencia temprana (óptimos locales) y de mejorar el control de búsqueda sobre el espacio de soluciones, surge la inclusión de un nuevo parámetro denominado peso inercial, introducido por Shi y Eberhart [6]. El concepto de peso de inercia se desarrolló con el fin de mejorar el control de la exploración y la explotación del algoritmo, y está definido como:

$$V_{i+1}^d = w \times V_i^d + c_1 \times rand1_i^d \times (Pbest_i^d - X_i^d) + c_2 \times rand2_i^d \times (Gbest_i^d - X_i^d) \quad (8)$$

Adicionalmente, Shi y Eberhart [6] sugieren usar  $0,8 \leq w \leq 1,4$ , comenzar con grandes valores (mejor búsqueda global), y realizar una reducción dinámicamente durante el proceso de optimización (mejor búsqueda local).

## B. Selección de parámetros

En el presente trabajo se empleó la versión del PSO que incluye el parámetro de inercia. El peso de inercia seleccionado fue de  $w = 1$ , y reducido dinámicamente al usar:

$$w_{k+1} = 0.98w_k \quad (9)$$

Donde  $w_{k+1}$  es el nuevo valor ajustado de  $w$ ,  $w_k$  que corresponde al valor previo de  $w$ .

La selección del peso inercial no se debe disociar de la elección de las constantes de aceleración cognitiva ( $c_1$ ) y social ( $c_2$ ) que controlan el flujo de información dentro del enjambre; si la partícula tendrá más confianza en los resultados de la búsqueda del enjambre, si  $c_2 > c_1$  de lo contrario, tendrá más confianza en los resultados obtenidos de su propia búsqueda. Con el fin de mantener un equilibrio entre las capacidades de búsqueda local y global, Shi [6] propone un valor de  $c_1 = c_2 = 2$ .

Otro parámetro que precisa ser seleccionado con extrema rigurosidad es el tamaño de la población. Poblaciones demasiado grandes exploran minuciosamente el espacio de búsqueda, pero el costo computacional asociado con el aumento del número de evaluaciones de la función objetivo puede resultar excesivamente elevado. Sin embargo, estudios paramétricos revelan que una población de alrededor de 30 agentes es suficiente para múltiples

problemas, y típicamente se utilizan poblaciones que oscilan entre 10 y 50 partículas, o entre 100 y 200 partículas a fin de abordar problemas más complejos [6].

### C. Evaluación del desempeño del PSO

A continuación, se describen los problemas (sin y con restricciones) para el análisis del desempeño y la validación del algoritmo PSO que posteriormente se va a implementar en el procedimiento de OT. La descripción incluye un conjunto importante de problemas bien conocidos de optimización combinatoria y numérica. Todos estos problemas se han estudiados mediante diferentes enfoques, desde métodos tradicionales hasta las modernas técnicas metaheurísticas. Se elaboró un código con el paquete computacional Matlab® y el equipo utilizado fue un Intel Core™ 2 Duo-1.33 GHz y 2.00 Gb de RAM.

**1. Problemas sin restricciones:** los problemas seleccionados para evaluar el potencial y las limitaciones de PSO se encuentran en [7], [8], en los que existe una gran variedad de funciones para validar algoritmos estocásticos de optimización global. En la Tabla 1 se encuentran resumidas dichas funciones.

**Tabla 1.** Funciones de prueba sin restricciones

Problema	Nombre de la función	Valor óptimo	Dimensión n
1	Función Rosenbrock (R)	0,000	50
2	Función Dixon (DX)	0,000	10
3	Función Ackley (AK)	0,000	50
4	Función Schwefek (SW)	-8379,672	20
5	Función Sphere (SP)	0,000	50
6	Función Neumaier (NM)	-210,000	10

Con base en que el algoritmo estudiado es de naturaleza estocástica, a continuación se describen los criterios para evaluar su desempeño. La desviación estándar de las funciones analíticas se utilizó con el fin de medir la precisión y la estabilidad del método. Se dice que un método estocástico de optimización es estable y preciso si su desviación estándar es baja. Se establece exactitud en el método cuando la diferencia entre la media de las pruebas y el valor óptimo analítico (conocido para la función de prueba) es

pequeño. El método resulta robusto si al aplicarse a diferentes problemas presenta precisión y exactitud. En este trabajo cada corrida del algoritmo se realizó 100 veces; el mejor valor de la función (MF), el peor valor de la función (PF), el promedio de los valores de la función (MEF), la desviación estándar de los valores de la función (DF) y el tiempo de ejecución promedio (TP) se presentan en la tabla 2.

**Tabla 2.** Desempeño de PSO en las funciones sin restricciones

Estadística	Funciones de prueba					
	R	DX	AK	SW	SP	NM
MF	0,031	0,000	0,000	-8379,672	0,000	-209,999
PF	0,040	0,000	0,000	-8379,678	0,000	-209,994
MEF	0,035	0,000	0,000	-8379,673	0,000	-209,996
DF	2,81E-04	5,24E-05	6,13E-04	812E-05	477E-05	6,16E-04
TP (seg)	350,1	20,5	58,4	30,4	6,1	55,2

Se puede considerar que PSO es un algoritmo de optimización robusto, debido a que se encontró un valor óptimo global en todas las funciones. Además, es un algoritmo muy rápido, lo cual se ve reflejado en los tiempos de ejecución promedio que presenta. Por otra parte, es un algoritmo muy estable, capaz de obtener resultados consistentes y razonables. Por último, se puede afirmar que el PSO muestra un gran potencial para manejar funciones unimodales difíciles de resolver (R, NM) y capacidad para escapar de óptimos locales cuando se trata de funciones multimodales (SW, AK).

**2. Problemas con restricciones:** con el objeto de probar la eficiencia del PSO se adoptó un conjunto estándar de funciones de prueba disponibles en la literatura [9]. Todas estas funciones se refieren a problemas de optimización numérica global resueltas previamente con técnicas evolutivas y clásicas. Este conjunto de funciones de prueba incluye problemas con espacios de búsqueda de alta y baja dimensionalidad, restricciones de igualdad y desigualdad, espacios convexos y no convexos, además de zonas factibles disjuntas o muy pequeñas. En la Tabla 3 se resumen las características de las 13 funciones de prueba adoptadas con el fin de evaluar el PSO.

**Tabla 3.** Funciones de prueba con restricciones

Problema	Valor óptimo	$n$	Función	$\rho$	LI	NI	LE	NE
g01	-15,000	13	cuadrática	0,00 %	9	0	0	0
g02	-0,803	20	no lineal	100,00 %	1	1	0	0
g03	-1,000	10	no lineal	0,00 %	0	0	0	1
g04	-30 665,530	5	cuadrática	27,01 %	0	6	0	0
g05	5126,490	4	no lineal	0,00 %	2	0	0	3
g06	-6961,814	2	no lineal	0,01 %	0	2	0	0
g07	24,306	10	cuadrática	0,00 %	3	5	0	0
g08	-0,096	2	no lineal	0,86 %	0	2	0	0
g09	680,630	7	no lineal	0,52 %	0	4	0	0
g10	7049,250	8	lineal	0,00 %	3	3	0	0
g11	0,750	2	cuadrática	0,10 %	0	0	0	1
g12	-1000	3	cuadrática	4,77 %	0	9	0	0
g13	0,054	5	no lineal	0,00 %	0	0	1	2

LI indica el número de desigualdades lineales, NI el número de desigualdades no lineales, LE el número de igualdades lineales, NE el número de igualdades no lineales,  $n$  el número de variables de decisión, y  $\rho$  representa una estimación del tamaño de la región factible con respecto a todo el espacio de búsqueda que se obtiene al generar un millón de soluciones aleatorias y evaluar qué porcentaje de ellas son factibles. Puede observarse que hay funciones que tienen la zona factible muy pequeña, como en el caso de g05, g07 y g13, en las que  $\rho=0,00\%$ , es decir, no se encuentra ningún individuo factible de manera aleatoria, lo que implica que en esos casos resultará desafiante el llegar siquiera a la zona factible. Los problemas g02, g03 y g12 son problemas de maximización y los casos restantes son de minimización, sin embargo, a fin de facilitar la presentación de resultados se manejarán como problemas de minimización. En el análisis de resultados se aplicó la misma estadística que para las funciones sin restricciones.

En la tabla 4 se observa que el PSO alcanza el valor óptimo en varios problemas y converge al óptimo en otros. Además, puede observarse que el algoritmo puede lidiar con problemas en los que el óptimo global se encuentra en la frontera de la región factible (g01, g04, g06, g07, g09); con regiones factibles grandes (g02), muy pequeñas (g05, g13) o disjuntas (g12); con problemas que contienen restricciones moderadas (g06); con alto número de restric-

ciones (g01, g04); con diferentes tipos de restricciones y sus combinaciones (lineales, no lineales, igualdad y desigualdad); con dimensionalidad baja (g06, g08), moderada (g09) y alta (g01, g02, g03, g07). Los problemas en los que el algoritmo presentó mayor dificultad fueron g05 y g13, debido a que tienen zonas factibles muy pequeñas y además restricción de igualdad, por lo que encontrar soluciones es una gran labor del algoritmo.

**Tabla 4.** Desempeño de PSO en las funciones con restricciones

Funciones de prueba						
Estadística	g01	g02	g03	g04	g05	g06
MF	-15,000	-0,803	-1,000	-30 665,500	5126,640	-6961,810
PF	-15,000	-0,750	-1,000	-30 665,500	5304,550	-6961,812
MEF	-15,000	-0,790	-1,000	-30 665,500	5161,180	-6961,811
DF	4,27E-04	1,49E-03	1,89E-05	3,01E-04	1,12E+01	1,57E-03
TP (seg)	102,2	238,9	315,1	42,6	41,5	53,9

Funciones de prueba							
Estadística	g07	g08	g09	g10	g11	g12	g13
MF	24,307	-0,096	680,630	7057,590	0,749	-1,000	0,058
PF	26,376	-0,096	681,550	8104,310	0,752	-1,000	0,527
MEF	25,355	-0,096	680,850	7560,040	0,750	-1,000	0,295
DF	2,04E-01	1,11E-04	2,93E-02	7,54E+02	4,43E-04	1,36E-04	5,89E-01
TP (seg)	254,7	67,2	47,4	101,7	44,4	397,3	95,8

Con base en los resultados presentados en las tablas 2 y 4 se procedió a la implementación del algoritmo PSO para la resolución de problemas de optimización topológica reportados en la literatura relevante. Los resultados que se obtuvieron se presentan en la siguiente sección.

#### IV. RESULTADOS Y DISCUSIÓN

Los resultados obtenidos se compararon con los reportados por Andreassen [1]. La implementación del algoritmo de OT también se realizó en Matlab® y el equipo que se utilizó fue un Intel Core™ 2 Duo-1.33 GHz y 2.00 Gb de RAM. Los resultados numéricos indicaron que con una población de cinco partículas, peso de inercia  $w = 1$ , y  $c_1 = c_2 = 2$ , son adecuados para proporcionar buenos resultados.

El primer problema es la denominada viga Messerschmitt-Bölkow-Blohm (MBB) [10] y las condiciones de contorno predeterminadas corresponden a la mitad de la viga (figura 2).

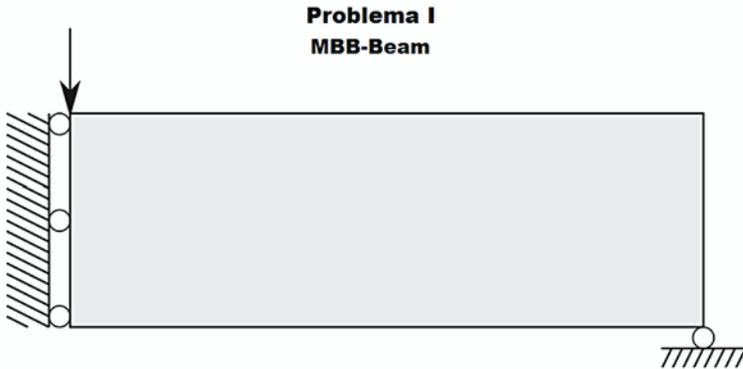


Figura 2. Viga MBB

La carga se aplica de forma vertical en la esquina superior izquierda; se encuentran condiciones de contorno simétricas a lo largo del borde izquierdo, y la estructura se apoya horizontalmente en la esquina inferior derecha. La estructura fue discretizada con tres tipos de malla, como se indica a continuación: (a)  $60 \times 20$  (60 elementos en dirección horizontal y 20 elementos en dirección vertical); (b)  $150 \times 50$ ; y (c)  $300 \times 100$ . El problema II es la viga en voladizo (Figura 1), conocida en la literatura internacional como el voladizo de Michell. Este problema se modeló con tres tipos de malla: (a)  $80 \times 50$ , (b)  $160 \times 100$ , y (c)  $240 \times 150$ . Por último, el problema III, denominado “viga en voladizo con hueco” (véase la Figura 1), se analizó con las mallas: (a)  $75 \times 50$ ; (b)  $150 \times 100$ ; y (c)  $225 \times 150$ .

Con el fin de evaluar el desempeño de la metodología aquí empleada se realizaron experimentos numéricos, y se obtuvo: las topologías, sus valores correspondientes de energía de deformación (c) y los tiempos de ejecución promedio para cada problema. Estos se encuentran resumidos en la Tabla 5.

Tabla 5. Valores de energía de deformación (c), tiempos de ejecución promedio y topologías.

P	Malla	Andreassen [1]			PSO		
		c	Tiempo (seg)	Topología	c	Tiempo (seg)	Topología
I	60 x 20	233,71	22,00		237,32	21,00	
	150 x 50	235,73	375,85		240,73	200,59	
	300 x 100	238,31	5531,50		246,51	240,98	
II	80 x 50	63,07	91,33		66,54	27,14	
	160 x 100	64,76	1213,50		66,15	485,23	
	240 x 150	65,62	6223,93		63,40	825,87	
III	75 x 50	54,23	80,49		54,43	42,04	
	150 x 100	55,05	1021,90		55,01	301,46	
	225x 150	55,52	3009,85		56,72	611,24	

Se puede observar que para el problema I el PSO presenta buenos resultados en valores de energía, pero varía un poco en la distribución del material. En cuanto a tiempos de ejecución, se puede decir que, mientras aumenta el número de elementos finitos (discretización), el PSO emplea menos tiempos en comparación con la referencia y, además, se pueden observar soluciones más suavizadas.

Para el problema II la PSO encontró topologías con inclusión de ligeras barras delgadas al comparar con la referencia, pero las cuales desaparecen cada vez que se aumenta la malla. En cuanto a los valores de energía de deformación, se puede decir que son razonables.

Por último, para el problema III se puede apreciar que la PSO obtiene topologías y valores de energía bastantes similares a la referencia, además de menores tiempos de ejecución, lo que evidencia la capacidad de la PSO para encontrar puntos óptimos globales en cualquier tipo de problemas.

## V. CONCLUSIONES

En este trabajo el PSO se implementa con el fin de sustituir el criterio de optimalidad utilizado dentro del método de optimización topológica propuesto por Andreassen [1]. Como primer paso, el desempeño del PSO se evaluó en 19 funciones de pruebas. En los dos grupos de funciones estudiadas (con y sin restricciones) presentó alta precisión, exactitud y robustez para encontrar soluciones, como se evidencia en las tablas 2 y 4.

Luego se ha conseguido evaluar el desempeño de PSO en el problema de optimización topológica de estructuras bidimensionales en régimen elástico. En cuanto a las topologías (distribución de material) obtenidas por PSO, varían un poco en los problemas I y II en comparación con Andreassen [1], e incluyen unos elementos que no son muy relevantes y tienden a desaparecer cuando el problema se analiza con mallas más refinadas. Los valores de energía y tiempos de computo obtenidos por PSO son coherentes y satisfactorios (véase la Tabla 5), lo que otorga validez y aplicabilidad al trabajo aquí realizado.

En cuanto a la técnica empleada, se puede observar que la PSO muestra gran ventaja en tiempos de ejecución cuando los problemas se discretizan con mallas más refinadas (aumento de número de elementos), de manera que se evitan topologías complejas, con mayores bifurcaciones y agujeros entre las barras. Por otra parte, la PSO tiene versatilidad para analizar diferentes tipos de problemas con diversos tipos de mallas. Esto se ve reflejado en las distribuciones de material, los valores de energía y los tiempos logrados.

## REFERENCIAS

- [1] E. Andreassen, A. Clausen, M. Schevenels, B. S. Lazarov y O. Sigmund, "Efficient topology optimization in MATLAB using 88 lines of code", *Struct. Multidiscip. Optim.*, vol. 43, n.º 1, pp. 1-16, 2011. [En línea]. doi: 10.1007/s00158-010-0594-7
- [2] O. Sigmund, "A 99 line topology optimization code written in matlab", *Struct. Multidiscip. Optim.*, vol. 21, n.º 2, pp. 120-127, 2001. [En línea]. doi: 10.1007/s001580050176
- [3] V. J. Challis, "A discrete level-set topology optimization code written in Matlab", *Struct. Multidiscip. Optim.*, vol. 41, n.º 3, pp. 453-464, 2010. [En línea]. doi: 10.1007/s00158-009-0430-0
- [4] K. Suresh, "A 199-line Matlab code for Pareto-optimal tracing in topology optimization", *Struct. Multidiscip. Optim.*, vol. 42, n.º 5, pp. 665-679, 2010. [En línea]. doi: 10.1007/s00158-010-0534-6
- [5] J. Kennedy y R. Eberhart, "Particle swarm optimization", 1995 *IEEE, Int. Conf. Neural Networks (ICNN 95)*, vol. 4, pp. 1942-1948, 1995.
- [6] Y. Shi y R. C. Eberhart, "Parameter selection in particle swarm optimization", en *Evolutionary Programming VII*, 1998, pp. 591-600.
- [7] H.-Y. Sang, Q.-K. Pan y P. Duan, "Self-adaptive fruit fly optimizer for global optimization", *Nat. Comput.*, 2017. [En línea]. doi: 10.1007/s11047-016-9604-z
- [8] M. Jamil y X.-S. Yang, "A literature survey of benchmark functions for global optimisation problems", *Int. J. Math. Model. Numer. Optim.*, vol. 4, n.º 2, pp. 150-194, 2013. [En línea]. doi: 10.1504/IJMMNO.2013.055204
- [9] J. Liu, K. L. Teo, X. Wang y C. Wu, "An exact penalty function-based differential search algorithm for constrained global optimization", *Soft Comput.*, vol. 20, n.º 4, pp. 1305-1313, 2016. [En línea]. doi: 10.1007/s00500-015-1588-6
- [10] B. S. Lazarov y F. Wang, "Maximum length scale in density based topology optimization", *Comput. Methods Appl. Mech. Eng.*, vol. 318, pp. 826-844, 2017. [En línea]. doi: 10.1016/j.cma.2017.02.018