

# Comparison of Three Metaheuristics for Leak Diagnosis in Water Distribution Networks

L.A. Sánchez-Mejía\* L. Gómez-Coronel\*\* I. Santos-Ruiz\*  
S. Gómez-Peñate\*

\* *Tecnológico Nacional de México, I.T. Tuxtla Gutiérrez,  
TURIX-Dynamics Diagnosis and Control Group. Carretera  
Panamericana S/N, 29050 Tuxtla Gutiérrez, Mexico*

\*\* *Institut de Robòtica i Informàtica Industrial (IRI), UPC-CSIC,  
Llorens i Artigas, 4-6, 08028, Barcelona, Spain.*

**Abstract:** This article presents a comparison between three different metaheuristic methods: the genetic algorithm (GA), the simulated annealing (SA), and the particle swarm optimization (PSO) in an experimental setup for leak diagnosis in a water distribution network. First a detection stage estimates the value of the flow-rate residual between the input and the output of the water distribution network until it exceeds a preestablished threshold value. The estimated value of the leak is analyzed during a previous time window and when its value is stable around a constant value a detection alert is emitted. Then, an optimization algorithm is implemented. A cost function with two search variables (location and magnitude of the leak) is then defined. The optimization of the cost function is performed using the three metaheuristic methods. Finally, results using experimental data are presented: the computing time of each metaheuristic method, the estimated leaked outflow and the success/failure rate for each method.

**Keywords:** Leak Diagnosis, Metaheuristics, Water Distribution Network, Optimization, Parameter Estimation.

## 1. INTRODUCCIÓN

Las fugas en las redes de distribución de agua generan impactos adversos tanto ambientales como económicos. Como suelen ocurrir en tuberías subterráneas, son difíciles de detectar a simple vista, lo que prolonga su duración antes de ser reparadas. El enfoque convencional para el manejo de las fugas implica auditorías hídricas, detección y localización de fugas y reparación o reemplazo de tuberías.

Existen *métodos basados en hardware* que utilizan dispositivos acústicos, ópticos y magnéticos, entre otros. Aunque son precisos, suelen ser costosos y requieren mucho tiempo. Por otro lado, existen *métodos basados en software* que incluyen técnicas de análisis de datos y modelos matemáticos de la red. Estos métodos son menos costosos y se basan en mediciones de presión y caudal obtenidas de sensores colocados en ubicaciones fijas en la red. Las técnicas de detección de fugas basadas en datos están ganando popularidad, debido a que ofrecen mayor flexibilidad y precisión (Sánchez-Rivero et al., 2020; Quiñones-Grueiro et al., 2021), aunque su costo computacional suele ser elevado. Para abordar esta limitante, se ha propuesto la aplicación de metaheurísticas, las cuales son algoritmos de optimización con un costo computacional moderado que han demostrado ser efectivos en diferentes ámbitos.

Las metaheurísticas son algoritmos de búsqueda global que exploran eficientemente el espacio de soluciones, para

hallar soluciones óptimas o cuasi óptimas en un tiempo razonable (Keramat et al., 2023; Yousefi-Khoshqalb et al., 2023), emulando fenómenos naturales como la evolución o procesos físicos como el enfriamiento.

En las redes de distribución de agua, partiendo de mediciones de presión y caudal, las metaheurísticas pueden ser utilizadas para ajustar o “calibrar” los parámetros del modelo hidráulico, considerando que algunos de esos parámetros cambian con las fugas, y así predecir tanto la ubicación como el tamaño de estas. Trabajos previos han explorado el uso de metaheurísticas en la solución de diferentes problemas en redes de distribución de agua. Por ejemplo, Gómez-Coronel et al. (2022) usa el algoritmo genético (GA) para calibrar parámetros del modelo hidráulico difíciles o imposibles de medir directamente: el coeficiente de rugosidad de la tubería y el coeficiente de pérdidas menores en los accesorios. Durante la calibración el GA propone valores para ambos parámetros, se realizan simulaciones considerando los valores calibrados y un modelo de simulación en EPANET, y se comparan los valores de presión simulados y medidos en diferentes puntos buscando minimizar el error entre ellos. Gómez-Coronel et al. (2023) también usaron GA para calibrar los parámetros de un modelo de difusión de cloro en una red de distribución de agua, asegurando que las concentraciones simuladas estén en concordancia con las reales tras aplicar cloro en la estación de bombeo. Luego, se realizó una segunda calibración para estimar la concentración mínima necesaria para mantener los niveles dentro de los límites

establecidos por la normativa federal durante varios días de simulación. A pesar de los resultados satisfactorios, la investigación se limitó al uso de una sola metaheurística.

Dado que una buena estrategia para detectar y localizar fugas depende de tener una red de sensores confiable, en los trabajos desarrollados por Morales-González et al. (2021) y Yousefi-Khoshqalb et al. (2023) se discuten diferentes métodos basados en metaheurísticas para colocar óptimamente sensores en una red de distribución de agua, obteniendo un buen compromiso entre los resultados y los costos involucrados. En Keramat et al. (2023) se combina el uso de metaheurísticas y métodos basados en el descenso del gradiente para localizar hasta tres fugas en una tubería simple. En Rostami and Darvishi (2021) se utiliza una metaheurística denominada “búsqueda de la armonía” (*Harmony search*) usando como variable de decisión el número de nodo donde se encuentra la fuga. El caudal fugado se calcula usando una solución inversa de las ecuaciones de continuidad y conservación de la energía. En forma similar, en el estudio desarrollado por Shahhosseini et al. (2023) se usan dos metaheurísticas (GA y programación no lineal) para identificar los coeficientes de fuga en los nodos del sistema basándose en datos medidos de presión y caudal en diferentes puntos del sistema. Con respecto al diagnóstico de fugas, estas aplicaciones con metaheurísticas reportadas en la literatura se aplican en diferentes casos de estudio y fueron implementadas en equipos con distinta potencia computacional, lo que dificulta una comparación directa y objetiva entre ellas.

En este artículo se presenta una comparación de la aplicación de diferentes metaheurísticas en el diagnóstico de fugas en una red ramificada, a partir de mediciones de caudal y de presión en algunos nodos de la red. El artículo se compone como se explica a continuación: En la Sección 2 se describen las metaheurísticas a utilizar. En la Sección 3 se presenta la metodología para diagnóstico de fugas mediante metaheurísticas. En la Sección 4 se presentan y discuten los resultados. Finalmente, las conclusiones y propuestas de trabajos futuros que expandan esta investigación se presentan en la Sección 5.

## 2. FUNDAMENTO TEÓRICO

Las metaheurísticas son algoritmos para buscar soluciones óptimas o satisfactorias a problemas complejos cuando los métodos tradicionales son demasiado lentos o no pueden encontrar una solución. Matemáticamente, se trata de hallar el valor  $\theta = \theta^*$  que minimiza una función  $J(\theta)$  denominada *función de costo* o *función objetivo*:

$$\theta^* = \arg \min_{\theta} J(\theta) \quad (1)$$

Cuando las metaheurísticas se usan para estimar  $p$  parámetros del modelo de un sistema físico,  $\theta$  es un vector en  $\mathbb{R}^p$  y la función de costo  $J(\theta)$  es una métrica de error que mide la discrepancia entre un conjunto de mediciones  $y_i$  de las variables físicas del sistema y sus contrapartes  $\hat{y}_i(\theta)$  obtenidas mediante simulación asignando valores específicos a los parámetros. La métrica de error utilizada en este trabajo es el error cuadrático medio (MSE), definido por:

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i(\theta))^2, \quad (2)$$

donde  $N$  es el número de muestras utilizadas para ajustar el modelo. La región  $\mathcal{S}$  en  $\mathbb{R}^p$  donde es factible encontrar el mínimo absoluto de  $J(\theta)$  se denomina *espacio de búsqueda* y, en principio, cada  $\theta \in \mathcal{S}$  es una *candidata*.

En la red hidráulica utilizada en este trabajo, los parámetros a estimar contenidos en  $\theta$  son: rugosidad de la tubería, coeficientes de pérdidas y caudales de fuga asociados a cada nodo de la red. Las  $y_i$  son mediciones de presión en nodos específicos de la red, mientras que  $\hat{y}_i(\theta)$  son estimaciones de esas presiones obtenidas por simulación a partir del modelo hidráulico parametrizado por  $\theta$ .

### 2.1 Algoritmo genético

El algoritmo genético (GA) es una metaheurística inspirada en la evolución biológica. Entre la población (soluciones factibles), se busca al individuo con el mejor desempeño (el menor error) para solucionar una tarea. Para ello, en cada generación (iteración) se selecciona a los individuos más aptos para una *cruza* y *mutación* genética. El GA se resume en cuatro pasos (Mirjalili, 2018; Katoch et al., 2021):

1. *Creación*: Se genera una población inicial distribuida tan uniformemente como sea posible en la región de búsqueda. Cada individuo es un  *cromosoma* y cada componente de la solución es un  *gen* del cromosoma.
2. *Selección*: Se asigna una probabilidad a cada individuo para que sus genes sobrevivan a la siguiente generación. A los individuos más aptos, según la función de costo, se les asigna una mayor probabilidad.
3. *Cruza*: Una vez que los individuos más aptos son seleccionados, se mezclan los genes de dos soluciones *padres* para crear dos soluciones *hijas*.
4. *Mutación*: Modificación aleatoria de algunos genes que ayuda a mantener la diversidad en la población y evita que el algoritmo se quede atascado en mínimos locales, permitiendo la búsqueda del mínimo global.

El proceso se ejecuta iterativamente hasta un número máximo de generaciones o hasta que el desempeño de la nueva generación ya no sea significativamente superior a la anterior. Al cumplirse este criterio de terminación se obtiene como resultado al individuo de mejor desempeño, es decir, la solución (cuasi-)óptima. En la Fig. 1 se muestra gráficamente este proceso (Mirjalili, 2019; Katoch et al., 2021).

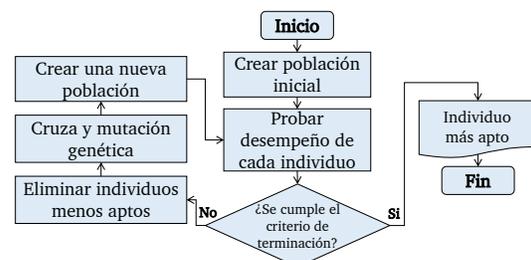


Fig. 1. Diagrama a bloques del GA.

### 2.2 Optimización por enjambre de partículas

La optimización por enjambre de partículas (PSO) es una metaheurística que se basa en el comportamiento colectivo de individuos trabajando conjuntamente para

mejorar tanto su desempeño individual como colectivo Cuevas et al. (2021), inspirado en el comportamiento social de pájaros y peces. El PSO es especialmente efectivo para problemas de optimización donde el espacio de búsqueda es grande y la función objetivo es compleja o no diferenciable. En forma resumida, el PSO contiene los siguientes pasos:

1. Crear una población aleatoria ( $P$ ) y establecer velocidades iniciales dentro de los límites especificados.
2. Evaluar la función objetivo para cada individuo en la población  $P$ .
3. Establecer la mejor posición personal ( $p_{best}$ ) y la mejor posición global ( $g_{best}$ ) de toda la población.
4. En cada iteración del algoritmo:
  - a) Ajustar la posición y velocidad de cada partícula.
  - b) Evaluar el valor de la función objetivo para la nueva posición.
  - c) Actualizar las mejores posiciones personales y la mejor posición global si se encuentra una mejora.
  - d) Finalizar las iteraciones si se cumple el criterio de paro; en caso contrario, continuar con el ciclo.

Como se muestra en la Fig. 2, en cada iteración se obtienen nuevas posiciones para las partículas a partir de una velocidad que se determina estableciendo una posición global y la mejor posición actual de cada partícula. La calidad de la población se determina evaluando los individuos con la función objetivo cada vez que toman una nueva posición. Al finalizar, se obtiene la mejor posición global que representa la solución óptima encontrada por PSO.

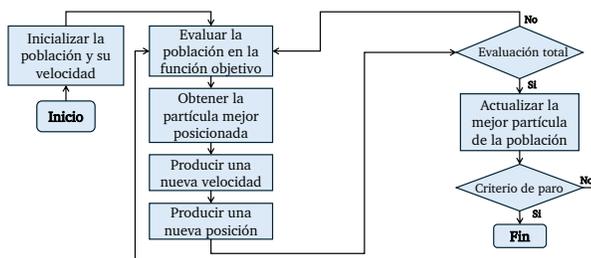


Fig. 2. Diagrama a bloques del PSO

### 2.3 Recocido simulado

El recocido simulado (SA) es un algoritmo de optimización que simula el proceso de enfriamiento lento de un metal. Comienza con una solución inicial y hace cambios aleatorios, aceptando tanto mejoras como empeoramientos con una probabilidad que disminuye con el tiempo. Esto permite explorar diversas soluciones y evitar quedar atrapado en óptimos locales. El proceso continúa hasta que se cumple un criterio de parada, como alcanzar un número máximo de iteraciones o una temperatura muy baja. En forma resumida, el SA contiene los siguientes pasos:

1. Se establece el problema a resolver junto con los parámetros del algoritmo, como la temperatura inicial  $T_0$  y el factor de enfriamiento  $\alpha$  Ávila et al. (2022).
2. Se crea una solución aleatoria como punto de partida.
3. Se repite el siguiente ciclo:

- a) Generar una nueva solución mediante perturbación de la solución actual.
- b) Evaluar la nueva solución.
- c) Decidir si aceptar la nueva solución basándose en una función de probabilidad que considera la diferencia en la función objetivo y la temperatura actual.
- d) Actualizar la solución actual si se acepta la nueva solución.
- e) Reducir la temperatura según una estrategia de enfriamiento.
- f) Finalizar las iteraciones si se cumple el criterio de paro; en caso contrario, continuar con el ciclo.

En general, el algoritmo converge hacia una solución óptima a medida que la temperatura disminuye gradualmente (Santa Chávez et al., 2014).

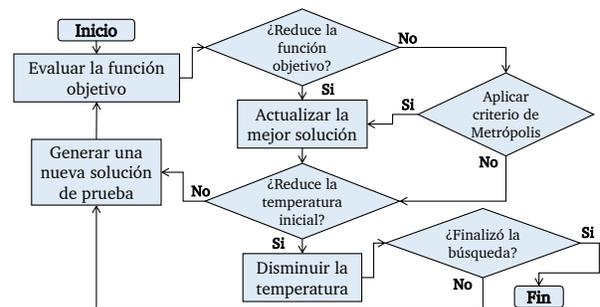


Fig. 3. Diagrama a bloques del SA

## 3. MATERIALES Y MÉTODOS

### 3.1 Descripción de la red hidráulica

El caso de estudio para probar las técnicas de diagnóstico de fugas es la red hidráulica mostrada en la Fig. 4. La red está construida con tubos de CPVC de 2 pulgadas, cédula 80, con diámetro interno de 48.6 mm. Las tuberías están dispuestas en dos camas con forma de serpentín separadas por un metro de elevación. La cama inferior (referida como la “línea principal de alimentación”) cuenta con dos ramificaciones que la conectan con la cama superior. El sistema se alimenta desde un tanque principal con una capacidad de 2500 L. El agua es impulsada en el sistema empleando una bomba centrífuga de 5 hp cuya potencia se regula mediante un variador de frecuencia para emular diferentes condiciones de alimentación a la entrada del sistema. La planta piloto se encuentra equipada con medidores de caudal en los puntos de entrada y salida de agua (FIT-01 aguas arriba; FIT-02, FIT-03 y FIT-04 en diferentes puntos aguas abajo) así como ocho medidores de presión (etiquetados PIT-01 a PIT-08) como puede verse en el diagrama de la Fig. 5. El agua que fluye aguas abajo se recicla al tanque principal, por lo que el nivel del mismo se mantiene aproximadamente constante en los experimentos.

Para simular experimentalmente las fugas, se dispone de cinco válvulas: tres en la línea principal de alimentación y una en cada uno de los ramales de la cama superior (ver Fig. 5). El agua que sale por estas válvulas en los experimentos también se recircula hacia el tanque principal.



Fig. 4. Red ramificada en la planta piloto.

### 3.2 Diagnóstico de fugas usando metaheurísticas

El diagnóstico de fugas consiste de dos etapas: en primer lugar, una etapa de *calibración* del modelo hidráulico en la que se asumen condiciones nominales de operación sin fuga donde se estiman parámetros no medibles (coeficiente de rugosidad, coeficientes de pérdidas menores) a partir de las mediciones de los sensores. Una vez que se tiene el modelo calibrado, se asumen sus parámetros como valores constantes y las meta-heurísticas se emplean en una segunda etapa de *diagnóstico* para estimar las variables de búsqueda asociadas con las fugas: su ubicación y magnitud. Se calibra simultáneamente para todas las ubicaciones posibles un caudal adicional que representa el caudal de fuga. Se propone que el algoritmo convergerá a valores de caudal fugado realistas según las condiciones físicas. Es decir, se asume que existen fugas en aquellos nodos, tuberías o sectores de la red en la que las magnitudes estimadas de los caudales de fuga son significativamente mayores que cero mientras que los nodos que no presentan fugas convergerán a valores de consumo cercanos a cero.

Para estimar los parámetros del modelo se minimiza la función de costo (2) que mide la discrepancia entre los valores medidos por los sensores y los estimados por la simulación con el modelo hidráulico. Las mediciones se toman en condición nominal sin fuga para múltiples presiones en el nodo de entrada (lo que se logra variando la potencia de bombeo). Durante la búsqueda de los valores óptimos de los parámetros, se establecen restricciones de frontera,  $\theta_{\min} \leq \theta \leq \theta_{\max}$ , según la cantidad y tipo de accesorios que se encuentran en cada línea de tubería.

---

#### Algoritmo 1: Calibración de parámetros

---

**ENTRADA:** Modelo hidráulico compatible con EPANET, con parámetros inicializados arbitrariamente.

**ADQUIRIR:** Mediciones de las variables relevantes del sistema.

**ESTABLECER:** los parámetros del área de búsqueda para la metaheurística.

**ITERACIÓN:**

1. Utilizar una metaheurística para minimizar el error entre las predicciones y las mediciones.
2. Iniciar un ciclo de optimización hasta que se alcance un criterio de convergencia.

**ASIGNAR:**

1. Proponer nuevos valores para los parámetros del modelo.
2. Simular el comportamiento del sistema con los parámetros propuestos.
3. Comparar las predicciones del modelo con las mediciones reales.

4. Calcular el Error Cuadrático Medio (RMSE) entre las predicciones y las mediciones.

5. Seleccionar los mejores valores de parámetros que minimizan el RMSE.

**CONDICIONES DE FINALIZACION:** Verificar si se alcanzó el número máximo de iteraciones o el umbral de error deseado.

**SALIDA:** Parámetros calibrados del modelo que minimizan el error (RMSE) entre las predicciones y las mediciones.

---

Una vez contando con los parámetros del modelo, etapa de diagnóstico se explica a continuación:

1. En primer lugar se calculan los valores de presión y caudal bajo condiciones nominales, es decir, sin fuga.
2. Dado que generalmente existen discrepancias entre los valores nominales y los valores actuales medidos en cada instante de tiempo, se calculan las diferencias entre ambos valores, denominadas *residuales*, que aumentan significativamente cuando ocurre una fuga. Sin embargo, no toda desviación que indique una caída de presión o una disminución instantánea en los caudales medidos aguas abajo es consecuencia de una fuga, pues estas variaciones pueden ser causadas por ruido en los sensores o por la turbulencia y los transitorios en el sistema. Por ello, se establecen de forma empírica unas desviaciones mínimas aceptables denominadas *umbrales de fuga*, de modo que cuando los residuales calculados exceden a esos umbrales se asume que ha ocurrido una fuga en alguna parte del sistema hidráulico. Los umbrales se asignan manualmente dependiendo de factores como el tamaño del sistema hidráulico, el flujo nominal y las magnitudes conocidas de las fugas más pequeñas a las que se espera que el sistema de diagnóstico sea sensible.
3. Una vez que se ha detectado la fuga se procede al diagnóstico. En esta etapa también se utiliza una *función de costo* para ser minimizada mediante metaheurísticas. Usando el *EPANET-MATLAB Toolkit* de OpenWater Analytics, se realiza una simulación hidráulica tomando como entradas las mediciones actuales de los sensores. Considerando a los caudales de fuga en los 5 nodos o regiones de la tubería como los parámetros sintonizables del modelo, las metaheurísticas ajustarán continuamente los caudales de fuga buscando minimizar las diferencias entre mediciones y simulaciones. El proceso de diagnóstico se resume en el Algoritmo 2.

---

#### Algoritmo 2: Diagnóstico de fugas

---

**ENTRADA:** Vector que contiene las mediciones instantáneas de los medidores de presión y caudal.

**ASIGNAR** las condiciones iniciales para la simulación al modelo hidráulico.

**ESTABLECER** los parámetros del área de búsqueda para la metaheurística.

**INICIA CICLO** hasta error mínimo

1. **PROPONER** valores para las demandas en las posiciones sospechosas de fuga.
2. **ASIGNAR** valores de fuga en los nodos correspondientes.
3. **RESOLVER** la hidráulica del sistema bajo estas condiciones.
4. **COMPARAR** los valores medidos y los simulados.
5. **EVALUAR** el error.

**FINAL CICLO**

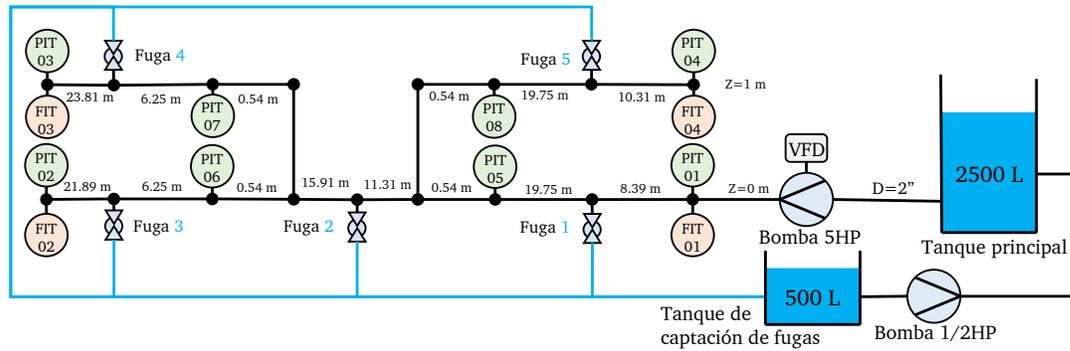


Fig. 5. Diagrama de Tuberías e Instrumentación (P&ID) de la planta piloto.

**COMPARAR** los valores de fuga calibrados e identificar la posición con el valor mayor.

**SALIDA:** Caudal de fuga estimado  $q_{leak}$  y nodo con fuga estimado  $n_l$ .

#### 4. RESULTADOS Y DISCUSIÓN

Las pruebas de diagnóstico de fugas se realizaron con 18 experimentos en la red hidráulica mostrada en la Fig. 4 para cinco ubicaciones de fuga. Se leyeron los sensores de presión y caudal durante dos minutos a una tasa de 10 muestras por segundo. Durante el primer minuto la red se mantiene sin fugas; al iniciar el segundo minuto, se activa una fuga abriendo la válvula correspondiente. En la Fig. 6, se muestran los resultados de un experimento para detectar una fuga ocasionada al abrir la válvula “Fuga 3”. La gráfica en la parte superior muestra el caudal de fuga calculado como la diferencia entre el caudal de entrada y la suma de los tres caudales de salida, asumiendo que no existe acumulación de agua en la red. A los 81 segundos de iniciar el experimento, 21 segundos después de abrir la válvula, se activó la alarma correspondiente y se mantuvo constante hasta el final del experimento (gráfica en la parte inferior de la Fig. 6). Las alarmas no se activan inmediatamente tras detectar el primer cambio en los caudales de fuga estimados por las metaheurísticas para evitar falsos positivos por las discrepancias instantáneas causadas por el ruido en los sensores o por seleccionar un umbral de fuga demasiado bajo.

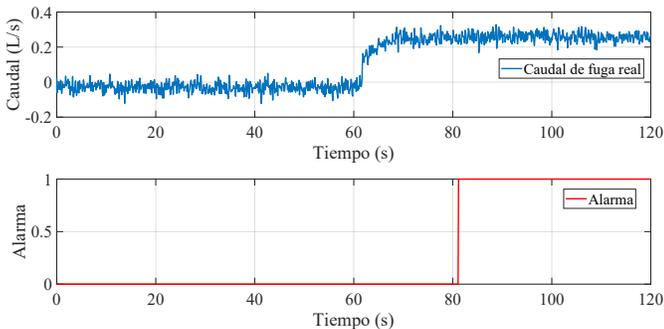


Fig. 6. Detección de una fuga en la posición “Fuga 3”

Para el GA se consideró una población de 50 individuos, un máximo de 80 generaciones, y una tolerancia de  $1 \times 10^{-8}$  en la función objetivo. Para el SA, se asignó una temperatura inicial de 50, con un máximo de 1000 iteraciones, y una tolerancia de  $1 \times 10^{-11}$ . Finalmente, para el PSO se seleccionaron enjambres de tamaño 100, con un

máximo de 100 iteraciones. Los parámetros se seleccionaron buscando la mejor relación entre costo computacional y precisión. Debido a la aleatoriedad intrínseca en la operación de los métodos metaheurísticos, las pruebas de diagnóstico se repitieron 5 veces para cada *dataset* para cada una de las metaheurísticas (GA, SA, PSO). La Tabla 1 muestra una comparación de la precisión promedio obtenida por cada uno de los algoritmos en las 5 pruebas realizadas en detectar correctamente la ubicación de la fuga.

Tabla 1. Exactitud en la detección de fugas con diferentes metaheurísticas

Fuga	GA	SA	PSO
1	100 %	85 %	75 %
2	100 %	50 %	0 %
3	89 %	85 %	100 %
4	100 %	100 %	100 %
5	100 %	100 %	100 %

De los resultados presentados de la localización en la Tabla 1 es posible apreciar cómo el GA es el método que proporciona la mayor precisión en la correcta detección de los escenarios de fuga, a excepción del tercer escenario. Por su parte el SA se desempeña de forma deficiente para localizar las fugas en las ubicaciones 1 a 3. Finalmente aunque el PSO logró identificar correctamente la fuga en la tercera posición en todos los experimentos realizados, se desempeña de forma poco satisfactoria para la primera ubicación de fuga. Debido a que las fugas 4 y 5 son suficientemente distinguibles debido a encontrarse en sectores más aislados de la red ningún método tuvo dificultad en localizarlas.

La Tabla 2 muestra la comparación de tres algoritmos en la estimación del caudal de fuga real en diferentes escenarios. Los valores reportados como  $Q_l$  son promedios de varios experimentos para cada caso de fuga, junto con el error de estimación promedio ( $e$ ) en cada caso.

La Tabla 3 muestra una comparación entre los tiempos de ejecución con las diferentes metaheurísticas utilizadas. La implementación se realizó en MATLAB corriendo en una computadora ROG Strix G513RM con sistema operativo Windows 11, procesador AMD Ryzen 7 6800H de 8 núcleos a 3.20 GHz, tarjeta gráfica NVIDIA GeForce RTX 3060, 1 TB de almacenamiento SSD, 16 GB de RAM y 24 GB de memoria virtual total.

Tabla 2. Estimación del caudal de fuga con diferentes metaheurísticas

Fuga ( $Q_i$ )	GA	SA	PSO
1 (0.4062 L/s)	0.4192 L/s ( $e = 3.21\%$ )	0.4238 L/s ( $e = 4.33\%$ )	0.2777 L/s ( $e = 31.64\%$ )
2 (0.3174 L/s)	0.3271 L/s ( $e = 3.06\%$ )	0.4231 L/s ( $e = 33.34\%$ )	0.4621 L/s ( $e = 45.59\%$ )
3 (0.2916 L/s)	0.2830 L/s ( $e = 2.95\%$ )	0.3032 L/s ( $e = 3.98\%$ )	0.2935 L/s ( $e = 0.65\%$ )
4 (0.3541 L/s)	0.3470 L/s ( $e = 2.00\%$ )	0.3624 L/s ( $e = 2.34\%$ )	0.3738 L/s ( $e = 5.57\%$ )
5 (0.3553 L/s)	0.3614 L/s ( $e = 1.71\%$ )	0.3744 L/s ( $e = 5.38\%$ )	0.3859 L/s ( $e = 8.61\%$ )

Tabla 3. Tiempos de ejecución con diferentes metaheurísticas

Método	Tiempo de ejecución
GA	24.43 s
SA	17.87 s
PSO	12.17 s

## 5. CONCLUSIÓN

Este artículo mostró una comparación entre diferentes métodos metaheurísticos en el diagnóstico de fugas en redes de distribución de agua. La metodología consta de dos etapas: en la primera se estiman los parámetros del modelo hidráulico del caso de estudio; en la segunda, las metaheurísticas se usan para estimar la ubicación y magnitud de la fuga. La validación experimental se efectuó en una red hidráulica ramificada a escala de laboratorio, considerando cinco escenarios de fuga. Los resultados demostraron que, entre las metaheurísticas probadas, el GA tiene el mejor desempeño en cuanto a la exactitud, aunque con un tiempo de ejecución mayor. El trabajo futuro se enfocará en una implementación en línea mediante cómputo en la nube e IoT para emitir alertas tempranas de fugas en tiempo real.

## AGRADECIMIENTOS

El trabajo ha sido financiado por el Tecnológico Nacional de México (TecNM) y por el Consejo Nacional de Humanidades, Ciencias y Tecnologías (Conahcyt). Los autores también agradecen el soporte científico de la Red Internacional de Control y Cómputo Aplicados (RICCA).

## REFERENCIAS

Cuevas, E., Fausto, F., Gálvez, J., and Rodríguez, A. (2021). *MATLAB: Computación metheurística y bioinspirada*. Alpha Editorial.

Gómez-Coronel, L., Santos-Ruiz, I., Torres, L., López-Estrada, F., and Delgado-Aguinaga, J. (2022). Model calibration for a hydraulic network using genetic algorithms. *Memorias del Congreso Nacional de Control Automático*, 146–251. doi: <https://doi.org/10.58571/CNCA.AMCA.2022.015>.

Gómez-Coronel, L., Delgado-Aguinaga, J.A., Santos-Ruiz, I., and Navarro-Díaz, A. (2023). Estimation of chlorine concentration in water distribution systems based on a genetic algorithm. *Processes*, 11(3). doi: [10.3390/pr11030676](https://doi.org/10.3390/pr11030676).

Katoch, S., Chauhan, S.S., and Kumar, V. (2021). A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80(5), 8091–8126. doi: [10.1007/s11042-020-10139-6](https://doi.org/10.1007/s11042-020-10139-6). <https://doi.org/10.58571/CNCA.AMCA.2024.013>

Keramat, A., Ahmadianfar, I., Duan, H.F., and Hou, Q. (2023). Spectral transient-based multiple leakage identification in water pipelines: An efficient hybrid gradient-metaheuristic optimization. *Expert Systems with Applications*, 224, 120021. doi: <https://doi.org/10.1016/j.eswa.2023.120021>.

Mirjalili, S. (2018). *Evolutionary Algorithms and Neural Networks*, volume 156 of *Studies in Computational Intelligence*. Springer Cham, 1 edition. doi:10.1007/978-3-319-93025-1. Published: 12 July 2018 (Hardcover), 26 June 2018 (eBook), 19 January 2019 (Softcover).

Mirjalili, S. (2019). *Genetic Algorithm*, 43–55. Springer International Publishing, Cham. doi:10.1007/978-3-319-93025-1\_4.

Morales-González, I.O., Santos-Ruiz, I., López-Estrada, F.R., and Puig, V. (2021). Pressure sensor placement for leak localization using simulated annealing with hyperparameter optimization. In *2021 5th International Conference on Control and Fault-Tolerant Systems (SysTol)*, 205–210. doi: [10.1109/SysTol52990.2021.9595369](https://doi.org/10.1109/SysTol52990.2021.9595369).

Quiñones-Grueiro, M., Ares Milián, M., Sánchez Rivero, M., Silva Neto, A.J., and Llanes-Santiago, O. (2021). Robust leak localization in water distribution networks using computational intelligence. *Neurocomputing*, 438, 195–208. doi: <https://doi.org/10.1016/j.neucom.2020.04.159>.

Rostami, I. and Darvishi, E. (2021). Combining inverse solution method and meta-heuristic algorithm to calculate the amount and location of leaks in water distribution networks. *Irrigation and Water Engineering*, 11(4), 87–104. doi:10.22125/iwe.2021.133686.

Sánchez-Rivero, M., Quiñones-Grueiro, M., Roseste Suárez, A., and Llanes Santiago, O. (2020). A novel approach for leak localization in water distribution networks using computational intelligence. In *Computational Intelligence in Emerging Technologies for Engineering Applications*, 103–122. Springer.

Santa Chávez, J.J., Meneses, C.A.P., and Echeverry, M.G. (2014). Algoritmo de recocido simulado aplicado al problema de secuenciamiento regular. *Avances Investigación en Ingeniería*, 11(1), 9–17.

Shahhosseini, A., Najarchi, M., Mahdi Najafzadeh, M., and Mirhoseini Hezaveh, M. (2023). Performance optimization of water distribution network using meta-heuristic algorithms from the perspective of leakage control and resiliency factor (case study: Tehran water distribution network, iran). *Results in Engineering*, 20, 101603. doi: <https://doi.org/10.1016/j.rineng.2023.101603>.

Yousefi-Khoshqalb, E., Nikoo, M.R., and Gandomi, A.H. (2023). Chapter 14 - optimal deployment of sensors for leakage detection in water distribution systems using metaheuristics. In S. Mirjalili and A.H. Gandomi (eds.), *Comprehensive Metaheuristics*, 269–291. Academic Press. doi: <https://doi.org/10.1016/B978-0-323-91781-0.00014-4>.

Ávila, C.A.M., Sánchez-Romero, F.J., López-Jiménez, P.A., and Pérez-Sánchez, M. (2022). Improve leakage management to reach sustainable water supply networks through by green energy systems. optimized case study. *Sustainable Cities and Society*, 83, 103994. doi: <https://doi.org/10.1016/j.scs.2022.103994>.