

# HAWANET: NUEVO SOFTWARE PARA LA OPTIMIZACIÓN DE REDES DE DISTRIBUCIÓN DE AGUA POTABLE UTILIZANDO ALGORITMOS EVOLUTIVOS

Jimmy H. Gutierrez-Bahamondes<sup>1</sup>, Daniel Mora-Melia<sup>2</sup>, P. L. Iglesias-Rey<sup>3</sup> y F. J. Martinez-Solano<sup>3</sup>

<sup>1</sup>Departamento de Ciencias de la Computación, Facultad de Ingeniería, Universidad de Talca, Chile.

<sup>2</sup>Departamento de Ingeniería y Gestión de la Construcción, Facultad de Ingeniería, Universidad de Talca, Chile.

<sup>3</sup>Departamento de Ingeniería Hidráulica y Medio Ambiente, Universitat Politècnica de Valencia, Valencia, España.  
E-mail: jgutierrezb@utalca.cl (G-B.J); damora@utalca.cl (M-M.D); piglesia@upv.es (I-R.P); jmsolano@upv.es (M-S.J)

## Introducción

El uso correcto del agua se ha vuelto de vital importancia y el estudio de sus sistemas es cada vez más relevante. Las redes de distribución de agua potable son sistemas hidráulicos que se componen de variados elementos, que en conjunto, permiten suministrar agua bajo ciertas restricciones de demanda.

Por lo tanto, cada día aumenta la necesidad de gestionar los recursos hídricos de una manera más eficiente. De esta forma, es esencial mejorar los sistemas de distribución de agua potable. Sin embargo, la optimización del diseño y operación de estas estructuras ha demostrado ser un conjunto de problemas muy complejos para las metodologías matemáticas tradicionales. Para intentar solucionar esto ha sido necesario recurrir a técnicas orientadas a la búsqueda de soluciones aceptables en periodos de tiempo razonables, como pueden ser los algoritmos heurísticos, que tienen uno de los mejores desempeños en este campo (Maier, et al., 2014).

Dentro de este contexto, diversas técnicas han sido testadas, pero no hay quórum acerca de cuál es la definitiva, especialmente cuando se abordan problemas en redes de tamaño real (Maier, et al., 2014), que por sus características requieren un gran número de variables de decisión. Todo esto, sumado a la no linealidad de los modelos matemáticos formulados y el gigantesco espacio de búsqueda de soluciones factibles, ha permitido clasificar este tipo de problemas como NP-Hard, principalmente por la alta complejidad de espacio de soluciones.

Por otro lado, los sistemas disponibles en el mercado que ofrecen la optimización de sistemas de agua potable son escasos, de pago o demasiado específicos. Además, otro problema importante es la calibración de los parámetros que guían los procesos de optimización de estos algoritmos (Geem, Z.W. and Sim, K.B., 2010).

Este trabajo presenta un nuevo software escalable que utiliza metodologías computacionales avanzadas para la optimización del diseño y operación de este tipo de redes.

## Metodología

Se desarrolló e implementó la versión preliminar de un software que utiliza algoritmos evolutivos para la optimización de distintos problemas de redes de abastecimiento de agua, como puede ser el dimensionado óptimo o el régimen de bombeo, cumpliendo en todo caso una serie de requerimientos de servicio, entre los que se incluyen condiciones mínimas de presión, velocidad, etc.

El software fue implementado en el lenguaje de programación Visual Basic.Net, utilizando la herramienta de programación Visual Studio 2013 a través de una metodología de desarrollo dinámica orientada a objetos.

Se diseñó una interfaz de comunicación con la biblioteca de enlace dinámico epanet2.dll (Toolkit para programadores desarrollado por EPANET) lo que permite realizar los cálculos necesarios en cada simulación hidráulica y conocer las condiciones de cada una de las variables involucradas inicialmente, durante y una vez finalizada cada simulación.

Como datos de partida o restricciones del problema, el software permite manejar presiones, velocidades, niveles de depósito, etc. Luego estos resultados son utilizados por el software en la evaluación de cada solución generada por los algoritmos implementados.

El programa cuenta con una interfaz gráfica que funciona bajo entorno Windows y se ha diseñado bajo un entorno amigable de modo que resulte fácil de utilizar incluso para usuarios con poca experiencia en algoritmos de optimización. La Figura 1 muestra algunas de las interfaces principales del software.

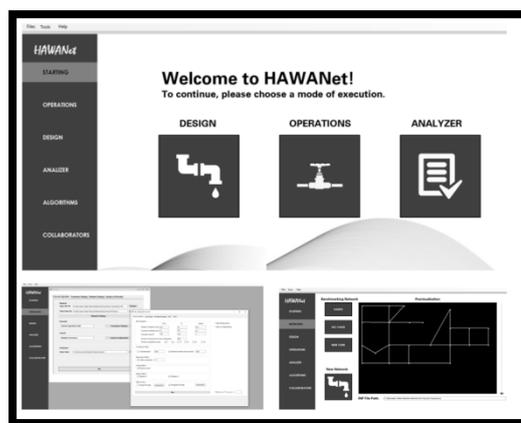


Figura 1.- Interfaz gráfica HAWANet.

El sistema construido ha sido diseñado modularmente utilizando una metodología de programación orientada a objetos en donde cada segmento de código ha sido encapsulado en diferentes clases, permitiendo que cualquier trabajo futuro se concentre en determinadas clases del código fuente sin alterar el resto. De esta manera HAWANet es considerada una herramienta totalmente escalable.

## Algoritmos implementados

Los algoritmos heurísticos implementados se describen brevemente a continuación:

**Genetic algorithm (GA)** Basado en la teoría de evolución, se genera una población de soluciones (individuos) con diferentes características la cual evoluciona seleccionando a los individuos con mejores aptitudes (menor costo de implementación) hasta detenerse según algún criterio previamente.

**Particle swarm optimization (PSO)** Este algoritmo intenta optimizar un problema a partir de una población de soluciones candidatas, denotadas como "partículas", moviendo éstas por todo el espacio de búsqueda según reglas matemáticas que tienen en cuenta la posición y la velocidad de las partículas.

**Local Search Algorithm (LS):** Única solución que evoluciona cada vez que se encuentra una mejor solución en el espacio de búsqueda, es muy propensa a caer en óptimos locales.

**Simulated Annealing Algorithm (SA):** Adaptación del algoritmo Local Search que permite aceptar soluciones peores con cierta probabilidad que van disminuyendo a medida que el número de iteraciones aumenta.

**Variable Neighbourhood Search (VNS):** Genera una solución única que es reemplazada cuando se encuentra una mejor entre las soluciones vecinas. Una de las grandes ventajas de este algoritmo está determinada por su cambio de vecindario cuando no encuentra mejoras, escapando de óptimos locales.

**Shuffled frog-leaping algorithm (SFLA):** Consiste en la imitación de una población de ranas que buscan alimento saltando entre piedras en su pantano. Cada salto provoca un cambio en la posición dentro del espacio de búsqueda. Los grupos de ranas evolucionan a medida que encuentran mejores soluciones.

Estas metodologías son genéricas y por tanto son utilizadas transversalmente tanto para la optimización en problemas de diseño como de operación. En cada uno de estos casos es necesario realizar un trabajo de calibrado para asegurar los mejores resultados, sobre todo frente a instancias donde el óptimo global del sistema es desconocido.

## Proceso de calibración

Cada algoritmo posee un conjunto de parámetros que afectan directamente su desempeño. Por consiguiente, es fundamental poder realizar un barrido que permita encontrar la mejor configuración, en cuanto a maximizar la posibilidad de encontrar mejores soluciones. HawaNet ofrece dos opciones, la primera opción es el ingreso manual, tal como muestra la Figura 2.

Figura 2.- Ingreso manual de Parámetros en GA.

La segunda alternativa permite asegurar que los resultados obtenidos sean estadísticamente aceptables produciendo un número alto de repeticiones para cada uno de los experimentos realizados. Además, esta opción automatiza el proceso de introducción de los parámetros del sistema a través de selección de rangos de procesamiento. Es decir, solo es necesario ingresar el valor mínimo, máximo y los intervalos de aumento para cada parámetro existente.

Figura 3.- Calibrado en HawaNet.

A partir de eso, el programa calcula el total de configuraciones y las ejecuta tantas veces como el usuario lo establezca. Además es posible determinar el número de ejecuciones paralelas que se utilizarán. La Figura 3 muestra la interfaz de procesamiento paralelo de HawaNet.

## Análisis estadístico

Una vez que todas las ejecuciones terminan, es posible volcar la información resultante a una nueva planilla Excel que automatiza el proceso de análisis estadístico de las soluciones a través de una Macro determinando el porcentaje de buenas soluciones que se han obtenido según parámetros determinados.

Una vez que los archivos son procesados, el sistema permite determinar libremente los umbrales de análisis (en términos porcentuales). De esta forma los usuarios podrán realizar comparaciones con el fin de obtener conclusiones que sirvan para mejorar la calibración de los algoritmos.

Para evitar la pérdida de datos que puede ocasionar alguna falla no pronosticada, el sistema crea un archivo Excel nuevo con cada vez que un proceso paralelo finaliza una configuración asegurando una copia de los resultados obtenidos anteriormente.

## Resultados

Se ha desarrollado una herramienta que representa un compilado de algoritmos metaheurísticos en un solo paquete. Está orientado al desarrollo académico.

El desempeño de la herramienta fue evaluado contrastando los resultados frente a los mejores valores conocidos en redes de benchmarking.

## Conclusiones

HawaNet es una herramienta de escritorio, creada para optimizar el diseño y operaciones de redes de distribución de agua potable. Este software utiliza algoritmos evolutivos para encontrar soluciones de buena calidad en periodos de tiempos razonables.

Apoyado en su capacidad de procesamiento paralelo puede reducir considerablemente los tiempos de calibración de los algoritmos implementados. Además, es totalmente escalable, lo que permite la incorporación de nuevas metodologías computacionales a sus funcionalidades.

HawaNet es una nueva herramienta que posee mucho potencial frente a la optimización del diseño y operación de redes de distribución de agua potable. Como desarrollo futuro, la idea es adaptarla progresivamente al análisis y diseño de sistemas cada vez más complejos.

## Referencias

**Maier, H. R. et al.** (2014) 'Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions', *Environmental Modelling & Software*, 62, pp. 271–299. doi: 10.1016/j.envsoft.2014.09.013.

**Geem, Z.W. and Sim, K. B.** (2010) 'Parameter-setting-free harmony search algorithm', *Applied Mathematics And Computation*, 217(8), pp. 3881–3889.