



Quinto
Congreso Nacional
de Riego y Drenaje
COMEII-AURPAES 2019

Septiembre 2019 | Mazatlán, Sinaloa



Artículo: COMEII-19002

Mazatlán, Sin., del 18 al 20

de septiembre de 2019

TRAZO DE REDES DE RIEGO MEDIANTE ALGORITMOS EVOLUTIVOS Y BIOINSPIRADOS: ALGORITMOS GENÉTICOS Y COLONIA DE HORMIGAS

Eduardo Jiménez Hernández^{1*}; Irineo L. López Cruz¹

¹Posgrado en Ingeniería Agrícola y Uso Integral del Agua. Universidad Autónoma Chapingo. Km. 38.5 carretera México - Texcoco, Chapingo, Estado de México. C.P. 56230.

eduardo.jimenez.eng@gmail.com - 595 133 0178 (*Autor de correspondencia)

Resumen

En este trabajo el trazo de redes de tuberías para riego se formuló como un problema de optimización combinatoria de variables discretas con restricciones. Con el objetivo de obtener redes de longitud mínima, se obtuvieron soluciones a este problema utilizando dos metaheurísticas, 1) Optimización con Colonia de Hormigas (ACO por sus siglas en inglés de *Ant Colony Optimization*), y 2) Algoritmos Genéticos (GA por sus siglas en inglés de *Genetic Algorithms*), mismas que se codificaron en el lenguaje de programación Python 2. El desempeño de ambas metaheurísticas se evaluó con redes de referencia usadas en la literatura y se obtuvieron soluciones iguales o cercanas a las óptimas. Para cada red de prueba se realizaron 20 ejecuciones de ambos métodos, en las que ACO demostró un desempeño superior, puesto que encontró la solución óptima en un número mayor de ejecuciones que GA, requiriendo para ello menos iteraciones.

Palabras clave: sistema de distribución de agua, redes de tuberías, ramales, optimización.



Introducción

Una red de distribución de agua es un sistema de elementos hidráulicos que se conectan para transportar agua desde la fuente de abastecimiento hasta los consumidores. De forma gráfica se representan mediante dos elementos básicos: nodos (bombas, válvulas, hidrantes, etc.) y uniones (tuberías). Dependiendo de su topología, se pueden clasificar en redes cerradas, si sus uniones forman algún tipo de “circuito cerrado”, o redes abiertas (también llamadas ramales, o redes con forma de árbol) cuando no contienen circuitos.

El proceso completo de planeación de una red de distribución de agua consiste de tres fases: a) trazo, b) dimensionamiento, y c) operación; y a pesar de que cada fase es dependiente de las otras, éstas se pueden formular y resolver como problemas independientes. Entonces el procedimiento de diseño de redes se lleva a cabo mediante iteraciones de estas tres fases (Eiger, Shamir, & Ben-Tal, 1994).

Aunque el ahorro debido a la optimización de la geometría (trazo) de las redes es generalmente mayor al que se obtiene por el dimensionamiento de sus componentes, la mayor parte de la investigación se ha enfocado en éste último, que es el problema más fácil (M. H. Afshar, 2006).

Gonçalves, Gouveia, & Pato (2014), mencionan que el problema de trazo de una red de distribución de agua es una instancia del problema de Árbol de Steiner, por lo tanto se puede clasificar como un problema NP-duro, puesto que es una generalización de un problema NP-duro conocido. Esto indica que, a pesar de que la formulación del trazo de una red es sencilla, los recursos de cómputo que se requieren para encontrar soluciones óptimas se incrementan con respecto al tamaño de la instancia del problema (número de nodos de la red). Para instancias grandes de estos problemas, y dado su complejidad, en ocasiones es imposible encontrar soluciones exactas, ya sea porque no existen métodos analíticos, o bien, porque tomaría mucho tiempo encontrarlas. Por esta razón se utilizan métodos heurísticos para encontrar soluciones aproximadas o subóptimas.

La palabra heurística proviene del griego *euriskein*, como la famosa palabra *eureka*, y quiere decir: encontrar. Se utiliza para caracterizar a las técnicas por las cuales se encuentra o se mejora la solución de un problema intratable. Se utilizan algoritmos heurísticos para obtener “buenas soluciones” (soluciones subóptimas o cercanas a la óptima global) de problemas cuyos algoritmos exactos de solución no son factibles en tiempos polinomiales. La palabra metaheurística, combina el prefijo *meta* del griego (que significa “más allá”, “a un nivel más alto”) con la palabra heurística, en otras palabras, se puede decir que metaheurística se refiere a una estrategia de nivel superior que guía a otras heurísticas para buscar soluciones factibles para problemas complejos.

La computación evolutiva es una de las principales ramas de metaheurísticas utilizadas para optimización. La idea de un algoritmo evolutivo es que: dada una población de individuos dentro de un ambiente que tiene recursos limitados, la competencia por dichos recursos causará selección natural (supervivencia del más apto). Teniendo una función a ser maximizada se puede crear un conjunto de soluciones candidatas al azar, y aplicando dicha función se puede obtener una medida abstracta de la aptitud de las soluciones, entre más alta mejor. Se seleccionan algunos de los candidatos con mas aptitud para procrear la siguiente generación. Esto se hace mediante la aplicación de operadores de recombinación y mutación. Los individuos nuevos son evaluados con



respecto a su aptitud y compiten con los viejos por un lugar en la siguiente generación. Este proceso se puede iterar hasta que se encuentre un candidato con la calidad suficiente (una solución) o hasta que se alcance un límite computacional previamente establecido. Entre los métodos que integran la computación evolutiva se encuentran: algoritmos genéticos, estrategias evolutivas, evolución diferencial, programación genética y programación evolutiva.

Bi et al. (2015) señalan que entre los principales algoritmos evolutivos que se han utilizado para resolver el problema de diseño de redes de distribución de agua, los algoritmos genéticos (GA por sus siglas en inglés de *Genetic Algorithms*) destacan por ser los más usados (M. H. Afshar, 2009; M. H. Afshar & Jabbari, 2008; Dandy, Simpson, & Murphy, 1996; Nicklow, Reed, & Savic, 2010; Simpson, Dandy, & Murphy, 1994; Walters & Lohbeck, 1993), sin embargo, su aplicación ha sido principalmente en problemas de prueba relativamente sencillos. Una revisión de todas las posibles aplicaciones que pueden tener los algoritmos genéticos en el manejo y planeación de recursos hídricos se puede encontrar en Nicklow et al. (2010).

Por otra parte se tienen algoritmos inspirados en procesos físicos, químicos y biológicos de la naturaleza, de los cuales la mayoría se inspira en procesos biológicos a los cuales se les llama: bio-inspirados. Una clase especial de estos algoritmos se ha desarrollado a partir de la inteligencia de enjambres, mismos que son los más populares. Algunos ejemplos son: optimización con colonia de hormigas, optimización con enjambre de partículas, búsqueda cuckoo, algoritmo de murciélagos y algoritmo de luciérnagas.

Los algoritmos de Optimización con Colonia de Hormigas (ACO por sus siglas en inglés de *Ant Colony Optimization*) han resultado ser una alternativa atractiva para el diseño óptimo de sistemas de distribución de agua ya que encuentra buenas soluciones usando una menor cantidad de evaluaciones de la función objetivo que GA (Maier et al., 2001, 2003). Una primera revisión de los problemas que se pueden resolver con ACO en sistemas de recursos hídricos es la realizada por Ostfeld (2011), sin embargo una referencia más completa se encuentra en A. Afshar, Massoumi, Afshar, & Mariño (2015).

En el diseño de redes de riego usualmente se requiere encontrar una solución para los problemas de trazo y dimensionamiento, ya sea de forma secuencial (Gonçalves & Pato (2000) y Gonçalves et al. (2014)); o simultánea, ambos enfoques han sido trabajados previamente por diversos investigadores utilizando diferentes métodos heurísticos.

Este trabajo tiene el objetivo de proporcionar evidencia de que ACO, al ser una metaheurística que ha demostrado buen desempeño en problemas de grafos, es una técnica que se puede aplicar eficientemente para obtener soluciones al problema de trazo de redes de tuberías para riego.

Materiales y Métodos

Los algoritmos genéticos son una técnica utilizada ampliamente y su funcionamiento se puede consultar en alguna de las referencias anteriores. La adopción de colonia de hormigas no es tan extensiva, por esta razón a continuación se describe brevemente.

Optimización con Colonia de Hormigas

ACO fue desarrollado por Dorigo, Maniezzo, & Colorni (1996), es un algoritmo de optimización combinatoria discreta basado en el comportamiento colectivo de hormigas en busca de alimento. Al inicio de la búsqueda las hormigas empiezan a explorar el ambiente de una forma que se presume casi aleatoria. El principio de funcionamiento de ACO se inspira en el hecho de que, después de un tiempo, esta colonia de hormigas es capaz de encontrar la ruta más corta desde su nido hasta una fuente de alimento. La inteligencia colectiva de la colonia se alcanza mediante una forma indirecta de comunicación que involucra el depósito en el suelo de una sustancia química, llamada feromona, por las hormigas que encuentran alimento de manera que las demás puedan seguir el rastro. Al paso del tiempo, las rutas más cortas, y más deseables, se refuerzan con cantidades más grandes de feromona, dado que requieren de menos tiempo para recorrerse, por lo que se convierten en la ruta dominante para la colonia.

Para aplicar ACO a un problema de optimización combinatoria, se requiere que el problema se represente como un grafo (M. H. Afshar & Mariño, 2006). Para esto considere que un grafo consistente de n puntos de decisión d_i ; $i = 1, 2, \dots, n$ donde cada punto de decisión está conectado a sus punto de decisión adyacentes mediante un conjunto de aristas o arcos. Por ejemplo, l_{ij} ; $j = 1, 2, \dots, NE_i$ representa un conjunto de arcos disponible en el punto de decisión i y NE_i representa el número total de arcos disponibles dicho punto de decisión. El algoritmo de ACO opera mediante la generación iterativa de una población de soluciones donde cada solución es representativa de la ruta que una hormiga ha viajado. Una hormiga genera una solución seleccionando un arco en cada punto de decisión con base en una regla de decisión probabilística. La regla original de decisión utilizada en el algoritmo de hormigas es:

$$p_{ij}(k, t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{j \in i}^{NE_i} [\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta} \quad (1)$$

donde $p_{ij}(k, t)$ es la probabilidad de que una hormiga k seleccione el arco l_{ij} estando en el punto de decisión i en la iteración t ; $\tau_{ij}(t)$ es la concentración feromona en el eje l_{ij} en la iteración t ; $\eta_{ij} = 1/c_{ij}$ es un valor heurístico que favorece a los arcos de costo mínimo local en l_{ij} , es decir, los arcos de distancia más corta; α y β son dos parámetros que controlan el peso relativo del rastro de feromona y el parámetro de sensibilidad heurística. El valor heurístico η_{ij} es análogo a proveer a las hormigas con visión y a veces se le llama visibilidad.

Una vez que un ciclo ha sido completado, el rastro de feromonas debe ser actualizado de manera que se refuercen las mejores soluciones. La forma general de la ecuación de actualización de feromonas es:

$$\tau_{ij}(t + 1) = \rho \tau_{ij}(t) + \Delta \tau_{ij} \quad (2)$$

donde $\tau_{ij}(t + 1)$ es la concentración de feromona en el arco l_{ij} en la iteración $t + 1$; $\tau_{ij}(t)$ es la concentración de feromona en el arco l_{ij} en la iteración t ; ρ es el coeficiente que representa el factor de persistencia de feromona, en ocasiones llamado factor de

evaporación de feromona; y el cambio en la concentración de feromona $\Delta\tau_{ij}$ está dado por Dorigo et al. (1996) como:

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k \quad (3)$$

donde m es el número total de hormigas y $\Delta\tau_{ij}^k$ es la feromona depositada por una hormiga k en el arco l_{ij} durante la iteración t . La concentración de feromona $\tau_{ij}(t)$ asociada al arco l_{ij} , intenta representar la tendencia que tiene la colonia a escoger el arco l_{ij} estando en el nodo i debida su aprendizaje. Es por ello que el rastro de feromona cambia durante la solución del problema para reflejar la experiencia adquirida por las hormigas. El factor de persistencia de feromona (ρ) es el mecanismo por el que se reduce la concentración del rastro de feromonas, permitiéndole a la colonia “olvidar arcos malos” e incrementar la probabilidad de seleccionar arcos que formen una solución óptima. El valor del factor de persistencia de feromona debe ser menor a uno para simular la evaporación de feromona. Su principal función es evitar el estancamiento, es decir, la situación en la que todas las hormigas terminan construyendo la misma solución. La cantidad de cambio en la concentración de feromona se define por Dorigo et al. (1996) como:

$$\Delta\tau_{ij} = \begin{cases} \frac{R}{f(S_k)}, & \text{si } l_{ij} \text{ se elije por la hormiga } k \\ 0, & \text{de otra manera} \end{cases} \quad (4)$$

donde $f(S_k)$ es el costo de la solución producida por la hormiga k y R es el factor de recompensa de feromona. Se debe hacer notar que la feromona se actualiza una vez que las hormigas han construido una solución. Además, las hormigas depositan una cantidad de feromona proporcional a la calidad de las soluciones que producen, de manera que a los arcos que se encuentren en soluciones de menor costo se les asignará más feromona y tendrán una mayor posibilidad de ser elegidos en futuras iteraciones. Los pasos de ejecución de ACO se muestran en la **Figura 1**.

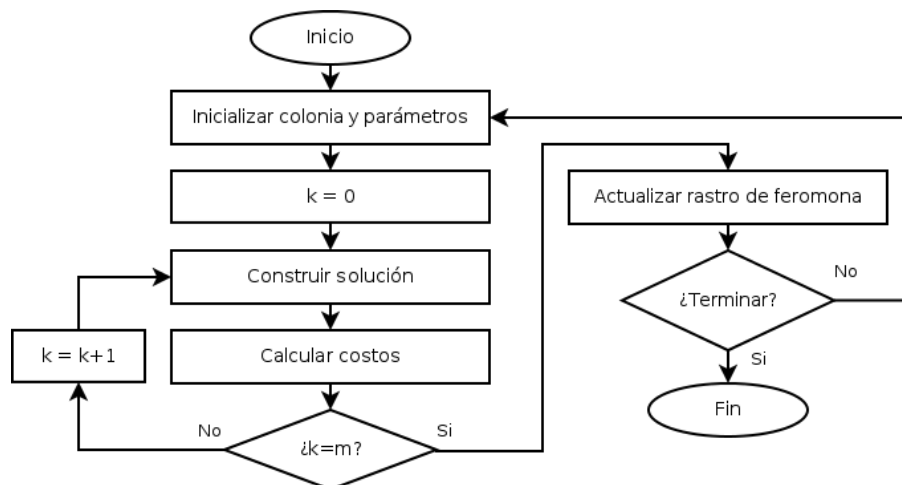


Figura 1. Diagrama de flujo del algoritmo de Optimización con Colonia de Hormigas, donde m es el número de hormigas de la colonia y k es la hormiga que construye una solución.

Formulación del problema

Aunque las redes de distribución de agua potable generalmente requieren circuitos cerrados para dar redundancia y asegurar la disponibilidad del agua, las redes de riego pueden ser abiertas y no tener circuitos en su topología. Estas redes son dendríticas ya que parten de uno o varios nodos de abastecimiento y tienen un conjunto de ramas que unen a los nodos fuente con los demás. Este trabajo se limitará a encontrar soluciones para el trazo óptimo de estas últimas. El trazo óptimo consiste en encontrar el subconjunto óptimo de conexiones entre sus nodos que proporcionen la red de longitud mínima, sujeto a un conjunto de restricciones hidráulicas. Formalmente, éste es un problema de optimización combinatoria de variables discretas con restricciones, cuya función objetivo se representa con la ecuación (5).

$$\min f(x_t) = \sum_{i=1}^P L_i \cdot \left(\sum_{j=i}^N pN_j + \sum_{j=1}^N pS_j \right) \quad (5)$$

donde $x_t = (x_1, x_2, \dots, x_N)$ es la topología de la red representada por una lista de adyacencia que contiene las conexiones entre los nodos de la red, $f(x_t)$ es la longitud total de las tuberías de la red, P es el número de tuberías de la red, N es el número de nodos de la red, y L_i es la longitud de la tubería i . Dado que los métodos utilizados para obtener soluciones a esta función contienen elementos estocásticos los valores de x_t pudieran contener redes o soluciones no factibles. Para descartar dichas soluciones se utilizan funciones de penalización que incrementan sus valores, alejando así esta solución del valor mínimo. En la función objetivo, pN es una penalización que se aplica cuando un nodo j se conecta con sí mismo, pS se aplica cuando el nodo j no se conecta con la fuente de abastecimiento.

Implementación de las metaheurísticas

Una de las metaheurísticas utilizadas fue GA, por ser una de las técnicas de computación evolutiva más utilizadas en problemas de planeación y manejo de recursos hídricos, y que ha dado buenos resultados en problemas similares. Los valores de los parámetros utilizados en GA se muestran en el **Cuadro 1**.

Cuadro 1. Valores de los parámetros usados en el algoritmo genético para la optimización del trazo y dimensionamiento de redes de tuberías para riego.

Parámetro	Valor
Población	200
Generaciones	1000
Operador de mutación	Intercambio (swap)
Operador de cruzamiento	En dos puntos
Operador de selección	Ruleta con escala lineal
Probabilidad de mutación	0.1
Probabilidad de cruzamiento	0.85

Por otra parte, se utilizó ACO por ser uno de los métodos inspirados en la naturaleza que mejor desempeño tienen para resolver problemas que se pueden representar mediante grafos. En el caso de ACO se utilizó el algoritmo *Ant System*, uno de los primeros algoritmos que se propusieron y cuya codificación es muy sencilla. El tamaño de la colonia es el número de hormigas artificiales y generalmente es igual a la dimensión del

problema a resolver; se permitieron 20 iteraciones, una tasa de evaporación de feromona $\rho=0.5$, $\alpha=1$ y $\beta=2$, con una feromona inicial de m/C^{nm} , siendo C^{nm} la longitud de una ruta generada con un algoritmo de vecinos más cercanos.

Las metaheurísticas GA y ACO se codificaron en el lenguaje de programación Python® 2.7. La organización de los módulos generados se muestra en la **Figura 2**, donde los bloques sombreados representan código de terceros utilizado en este proyecto y los bloques claros indican el código escrito como parte de este trabajo. Se codificaron las funciones objetivo en un módulo llamado *objectivefunctions.py* el cual se utiliza por el módulo principal *optimizer.py* para obtener soluciones a los problemas utilizando GA o ACO. Finalmente, se crearon algunos módulos auxiliares *utilities.py* y *graphics.py* para contener funciones de uso común y crear gráficas, respectivamente. Además, se codificaron algoritmos para el problema de dimensionamiento de la red utilizando el software EPANET 2 para simular el comportamiento hidráulico, sin embargo, este problema no se presenta en este trabajo. El código de terceros utilizado consiste en 1) la librería de funciones EPANET Toolkit, 2) *epamodule.py* que permite hacer uso del código de EPANET, escrito en ANSI C, directamente en lenguaje Python® a través del módulo *epanet-python* (<https://github.com/OpenWaterAnalytics/epanet-python>), y 3) una implementación de algoritmos genéticos llamada *pyevolve* 0.6rc1.

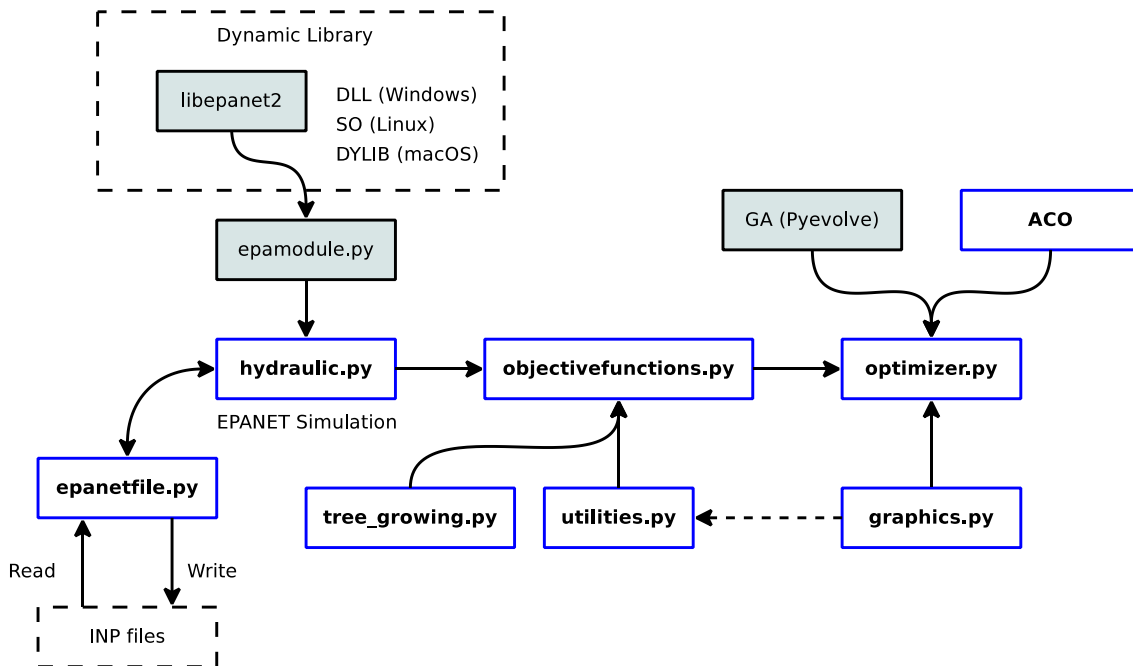


Figura 2. Diagrama de bloques del código realizado para el diseño óptimo de redes de riego.

El desarrollo y las pruebas se realizaron en una computadora con sistema operativo Linux 4.4, un procesador de 64 bits Intel® Core™ i7-4500 @ 1.80 GHz y 8 GB de RAM.

Redes de prueba

Las metaheurísticas se pusieron a prueba mediante la solución de redes de referencia usadas en la literatura. Se utilizan problemas sencillos cuya solución óptima es conocida

para evaluar si las metaheurísticas son capaces de encontrar dichas soluciones o al menos acercarse a ellas. Se realizó de esta manera debido a que no es posible asegurar que estos métodos encuentren o se acerquen a la solución óptima de redes con una gran cantidad de nodos y tuberías.

La primera red teórica utilizada, es la red de 3×3 originalmente propuesta por Geem, Kim, & Kim (2000) es una cuadrícula de 9 nodos, 12 conexiones posibles y una fuente de abastecimiento en el nodo 9 que tiene una elevación de 50 m. Mientras que los demás son nodos de demanda y tienen una elevación de cero. La carga de presión mínima requerida en los nodos es de 30 m, las tuberías tienen una longitud de 100 m y un coeficiente de Hazen-Williams de $C=130$. La configuración de esta red se muestra en la **Figura 3 a)**.

La segunda red de prueba es una red de 8×8 , o red de 64 nodos, fue propuesta y resuelta originalmente por Walters & Smith (1995) con base en problemas similares previamente propuestos por Walters & Lohbeck (1993). La gráfica base de esta red es una cuadrícula de 8×8 nodos que tiene una fuente de abastecimiento en el nodo 64.

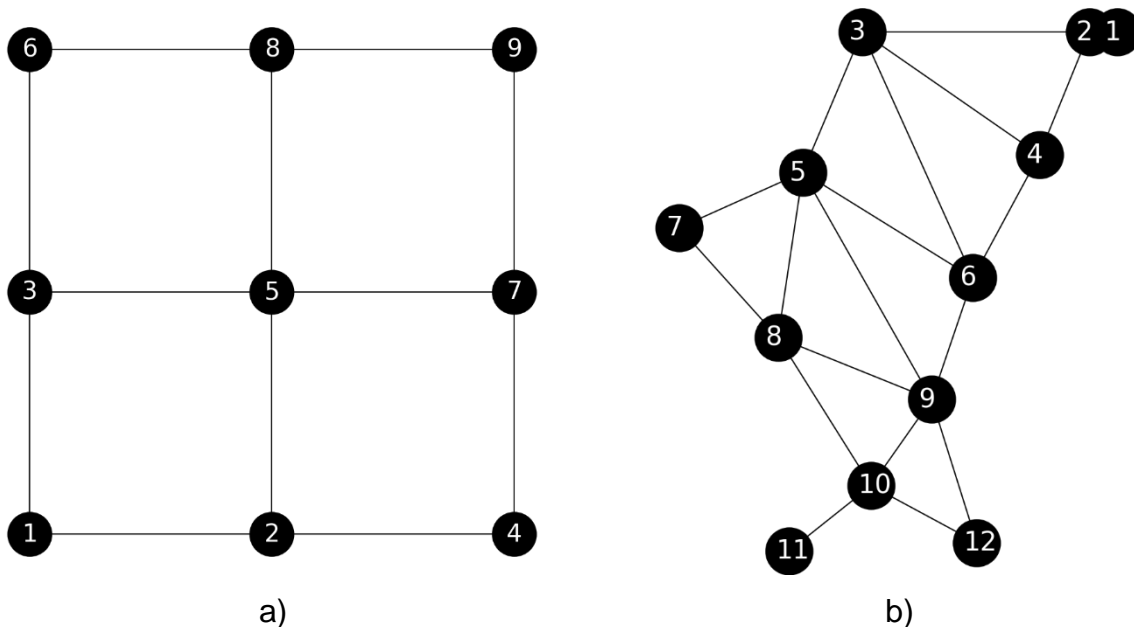


Figura 3. Configuración de dos de las redes de prueba utilizadas para evaluar el desempeño de las metaheurísticas GA y ACO; a la izquierda se muestran los nodos y conexiones de la red de 3×3 y; a la derecha los hidrantes de la red de riego de 12 nodos.

La última red de prueba es una red de riego de 12 nodos propuesta por Ángeles Montiel (2002) es de un proyecto que se localiza en el predio “La Noria” del Ejido Roque 1, en el municipio de Celaya, en el DR 085 La Begoña, Guanajuato. La fuente de abastecimiento es una toma del canal lateral 11+141 que conduce agua desde la presa derivadora La Soria, este punto se representa con el nodo 1. El material de las tuberías es PVC con un coeficiente de fricción de Hazen-Williams de $C=140$. La configuración de esta red se muestra en la **Figura 3 b)**.

conexiones posibles entre los nodos que tiene un tamaño de 2^{112} , ya que también se incluyen conexiones en diagonal entre los nodos.

Cuadro 2. Soluciones para el trazo de la red 3×3 obtenidas con las metaheurísticas Algoritmos Genéticos y Optimización con Colonia de Hormigas.

Ejecución	GA (200 ind., 500 gen.)			ACO ($m=5$, 10 iter., $\rho=0.5$)		
	Mínimo (m)	Diferencia (%)	Generaciones requeridas	Mínimo (m)	Diferencia (%)	Iteraciones requeridas
1	882.84	10.36	24	800	0	3
2	800.00	0.00	51	800	0	7
3	800.00	0.00	403	800	0	5
4	800.00	0.00	304	800	0	5
5	841.42	5.18	12	800	0	5
6	800.00	0.00	35	800	0	3
7	800.00	0.00	179	800	0	4
8	841.42	5.18	67	800	0	6
9	800.00	0.00	58	800	0	3
10	841.42	5.18	16	800	0	3
11	841.42	5.18	52	800	0	4
12	841.42	5.18	32	800	0	6
13	841.42	5.18	30	800	0	4
14	841.42	5.18	23	800	0	4
15	800.00	0.00	41	800	0	5
16	800.00	0.00	63	800	0	4
17	841.42	5.18	291	800	0	3
18	841.42	5.18	40	800	0	5
19	800.00	0.00	440	800	0	5
20	800.00	0.00	54	800	0	5
Promedio:	822.78	2.85	163*	800.00	0.00	4*
Desv. Estándar:	25.05	3.13	160.27	0.00	0.00	1.15

* Solo incluye las ejecuciones que encontraron la solución óptima

Usando ACO se la solución óptima global en el 100% de las ejecuciones realizadas, y se requirió de 15 iteraciones en promedio. En este caso se utilizó un tamaño de colonia de $m=10$ hormigas artificiales, 20 iteraciones y una tasa de evaporación de $\rho=0.5$. La red mostrada en la **Figura 5** es la mejor obtenida de 20 ejecuciones del algoritmo. Todas las soluciones óptimas para el trazo de esta red tienen este aspecto, aunque las conexiones pueden ser diferentes.

Por otra parte, GA fue incapaz de obtener soluciones factibles para este problema a pesar de que se usaron valores en sus parámetros mayores a los mostrados en el **Cuadro 1**, para el tamaño de población se usaron 100 y 200 individuos; y también se intentó con 500, 1000 y 10000 generaciones sin obtener éxito.

Para ayudar al algoritmo a encontrar las soluciones óptimas se utilizaron dos modificaciones. La primera consistió en reducir el espacio de búsqueda, de manera que se proporcionó un conjunto reducido de las conexiones permitidas entre los nodos de la red, utilizando una gráfica base que consiste de una cuadrícula con 112 conexiones para los 64 nodos, de manera que el espacio de búsqueda se redujo a 2^{112} combinaciones

posibles. La segunda modificación consistió en la exploración del espacio de búsqueda de las soluciones factibles, para ello se utilizó un *algoritmo de crecimiento de árbol* (Walters & Smith, 1995). Este algoritmo elige un nodo raíz de manera aleatoria y va agregando los nodos que se encuentren desconectados de manera iterativa hasta que finalmente se conectan todos en una red ramal. Todos los nodos que se van agregando se escogen mediante una función de probabilidades uniforme discreta y no se toma en cuenta la proximidad entre ellos. Auxiliándose de estas dos modificaciones, GA fue capaz de encontrar las soluciones óptimas para esta red desde la primera generación usando los parámetros recomendados.

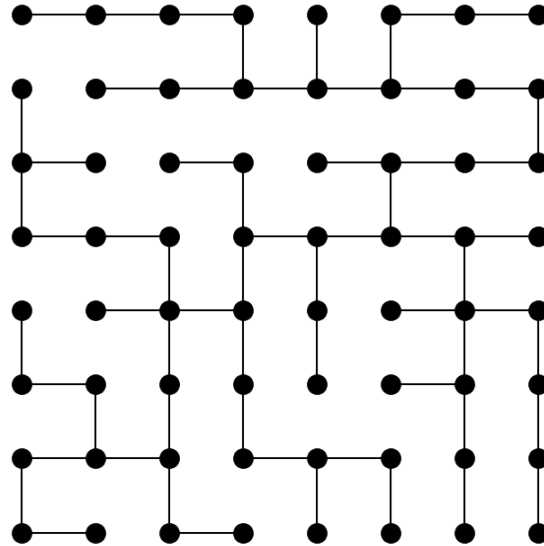


Figura 5. Solución óptima para el trazo de la red de 8 x 8 encontrada colonia de hormigas.

Red de riego de 12 nodos

Las soluciones para la red de riego propuesta por Ángeles Montiel (2002) no son tan fáciles de visualizar como en las otras redes tipo cuadrícula, debido a que la distribución de los nodos es irregular se tienen que usar algoritmos para encontrar la red de menor longitud. En este caso se usaron algoritmos de *Árbol de Expansión Mínima* para encontrar la solución óptima del trazo de la red. Los algoritmos usados fueron: a) Kruskal y b) Prim, los cuales hacen una búsqueda exhaustiva y conectan los nodos más próximos, de manera que se obtiene una red con topología de árbol de longitud mínima. La solución óptima alcanzada con ambos algoritmos tiene una longitud de 1,946.04 unidades.

En el **Cuadro 3** se presenta una comparación las soluciones obtenidas para el trazo de la red de 12 nodos con 20 ejecuciones de las metaheurísticas GA y ACO. Para cada ejecución se presenta la solución mínima obtenida, su diferencia con respecto al óptimo global y el promedio de generaciones o iteraciones en las que se obtuvo dicha solución.

Usando la metaheurística ACO se obtuvo la solución igual a la encontrada con los algoritmos de Kruskal y Prim. Con la configuración indicada se necesitó un máximo de 50 evaluaciones de la función objetivo. La mejor solución se encontró en 9 de las 20 ejecuciones realizadas, esto es el 45 % de las veces y se obtuvo en un promedio de 9 iteraciones por cada ejecución del algoritmo. El promedio de la solución mínima obtenida

resultó de 1,966.15 con una desviación estándar de 28.31, que es bastante cercano al óptimo.

Cuadro 3. Soluciones para el trazo de la red de 12 nodos obtenidas con Algoritmos Genéticos y Optimización con Colonia de Hormigas.

Ejecución	GA (200 ind., 1000 gen)			ACO ($m=5$, 10 iter., $\rho=0.5$)		
	Mínimo (m)	Diferencia (%)	Generaciones requeridas	Mínimo (m)	Diferencia (%)	Iteraciones requeridas
1	2619.59	34.61	56	1946.04	0.00	10
2	2546.37	30.85	172	1946.04	0.00	10
3	2391.37	22.88	807	1946.04	0.00	8
4	2165.37	11.27	705	2003.03	2.93	10
5	2503.00	28.62	410	1946.04	0.00	10
6	2497.63	28.34	144	1995.51	2.54	7
7	2291.96	17.78	147	1989.42	2.23	10
8	2607.94	34.01	629	1946.04	0.00	9
9	2374.41	22.01	276	1964.43	0.95	7
10	2479.81	27.43	132	1959.66	0.70	9
11	2381.38	22.37	235	1946.04	0.00	7
12	2443.90	25.58	141	2000.65	2.81	8
13	2481.09	27.49	891	2052.50	5.47	5
14	2922.94	50.20	114	1959.66	0.70	8
15	2416.58	24.18	212	1964.43	0.95	10
16	1987.72	2.14	557	1946.04	0.00	9
17	2479.66	27.42	317	1946.04	0.00	10
18	2542.22	30.64	682	1959.66	0.70	8
19	2093.71	7.59	364	1959.66	0.70	10
20	2264.88	16.38	470	1946.04	0.00	8
Promedio:	2424.58	24.59	373*	1966.15	1.03	9*
Desv. Est.:	203.39	10.45	256.86	28.31	1.45	1.12

* Solo incluye las ejecuciones que encontraron la mejor solución

Por otro lado, la metaheurística GA con los parámetros especificados en el **Cuadro 1** no pudo encontrar la solución óptima y en su lugar tuvo convergencia hacia una solución subóptima con un valor de 1,987.72, misma que únicamente se obtuvo en una de las 20 ejecuciones (5 % de las veces). GA tuvo un desempeño muy pobre puesto que requirió de 200,000 evaluaciones de la función objetivo para llegar a la solución subóptima. El promedio de la solución mínima obtenida es de 2,424.58 con una desviación estándar de 203.39, valores que se encuentran alejados de la solución óptima. En la **Figura 6** se muestran gráficamente las soluciones para el trazo de la red obtenidas con ambas metaheurísticas.

Aunque esta red se ha estudiado muy poco, se realizó una comparación con los resultados disponibles en la literatura. En el **Cuadro 4** se muestra el valor de las mejores soluciones reportadas y el método con el que se obtuvo dicho valor. Se puede observar que el autor original obtuvo una solución muy cercana al mínimo, sin embargo, dicha solución fue modificada manualmente para ajustarse a linderos parcelarios.

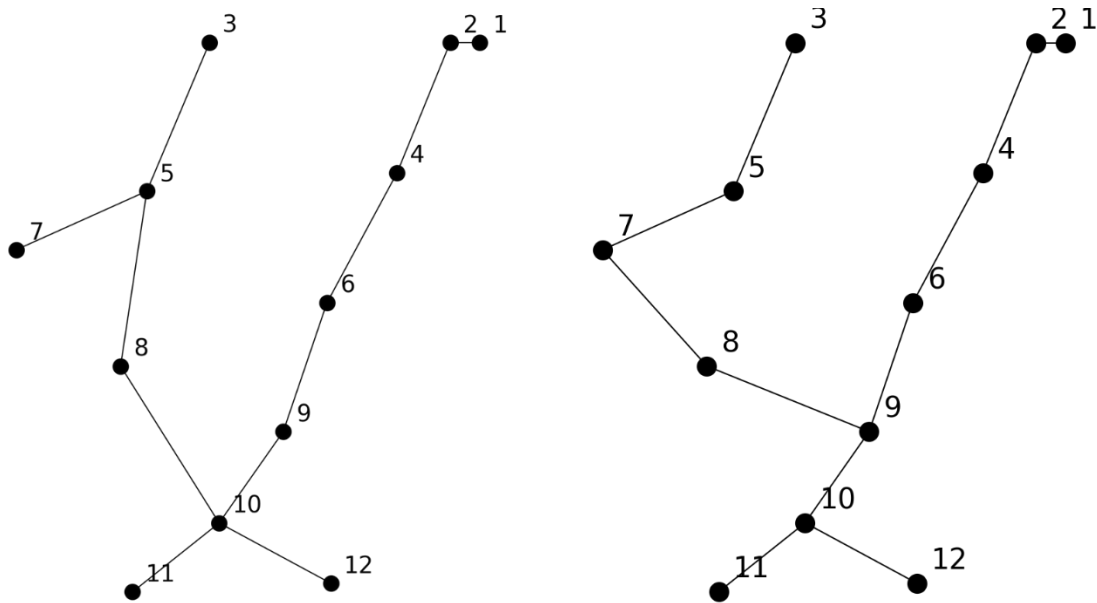


Figura 6. Mejores soluciones para el trazo de la red de 12 nodos, se presenta la mejor solución obtenida con GA longitud=1,987.72 (izq.) y ACO longitud=1,946.04 (der.).

Cuadro 4. Soluciones reportadas para el trazo de la red de riego de 12 nodos usando diferentes metaheurísticas: Programación Lineal (LP), Evolución Diferencial (DE), Colonia Artificial de Abejas (ABC), Algoritmos Genéticos (GA) y Colonia de Hormigas (ACO).

Solución	Método	Evaluaciones	Costo (Longitud, en m)
Ángeles Montiel (2002)	LP	No reportado	2,212.00
Ponce Pacheco (2013)	DE	1,100,000	1,946.04
Ponce Pacheco (2013)	ABC	1,100,000	2,113.86
Este trabajo	Kruskal, Prim	No aplica	1,946.04
Este trabajo	GA	200,000	1,987.72
Este trabajo	ACO	50	1,946.04

Entre la variedad de métodos que se han utilizado para solucionar el trazo de esta red, se puede observar que únicamente dos metaheurísticas fueron capaces de encontrar la solución óptima: ACO y DE. ACO es la más eficiente en cuanto al número de evaluaciones de la función objetivo requeridas para encontrar la solución óptima. Sin embargo, dado que no se reporta el número de evaluaciones mínimas en las que DE encuentra la solución, no es posible hacer una comparación justa puesto que las evaluaciones se obtuvieron directamente del número de ejecuciones del algoritmo.

Conclusiones

Se desarrolló un programa en lenguaje Python® 2 para obtener soluciones óptimas al problema de trazo de redes de riego mediante Optimización con Colonia de Hormigas (ACO). Las soluciones obtenidas se compararon con las de Algoritmos Genéticos (GA). Ambas metaheurísticas se evaluaron utilizando redes de prueba usadas comúnmente en



la literatura. La mayoría de las soluciones obtenidas fueron iguales o muy cercanas a las mejores reportadas hasta el momento.

Aunque las soluciones óptimas para el problema de trazo de las tres redes de prueba evaluadas se obtuvieron con ambos algoritmos, ACO tuvo un desempeño superior ya que dichas soluciones se obtuvieron en un porcentaje mayor de las ejecuciones, y requirió de un número menor de evaluaciones de la función objetivo con respecto a GA. Además, GA tuvo dificultad para encontrar soluciones a instancias grandes del problema de trazo, como la red de 64 nodos, en un tiempo aceptable por lo que se tuvieron que utilizar métodos auxiliares adicionales.

La aparente ventaja que tiene ACO sobre GA posiblemente se debe a que GA genera soluciones completamente al azar, mientras que ACO toma en cuenta la información geométrica del problema, incrementando la probabilidad de incluir segmentos de distancia mínima a las mejores soluciones del problema.

Lo anterior refuerza la hipótesis de que ACO es una técnica adecuada y eficiente generar soluciones óptimas para problemas de grafos, particularmente trazo de redes ramales, por lo que se recomienda fomentar su uso para obtener soluciones a este tipo de problemas.

Referencias Bibliográficas

- Ángeles Montiel, V. (2002) *Redes abiertas de tuberías para riego (Trazo, diseño, revisión y análisis)*. Chapingo, México: Universidad Autónoma Chapingo, pp 1-149.
- Afshar, A., Massoumi, F., Afshar, A., & Mariño, M. A. (2015). State of the art review of ant colony optimization applications in water resource management. *Water Resources Management*, 29, 3891–3904. <https://doi.org/10.1007/s11269-015-1016-9>
- Afshar, M. H. (2006). Application of a max-min ant system to joint layout and size optimization of pipe networks. *Engineering Optimization*, 38(3), 299–317. <https://doi.org/10.1080/03052150500476357>
- Afshar, M. H. (2009). Application of a compact genetic algorithm to pipe network optimization problems. *Transaction A: Civil Engineering*, 16(3), 264–271.
- Afshar, M. H., & Jabbari, E. (2008). Simultaneous layout and pipe size optimization of pipe networks using genetic algorithm. *The Arabian Journal for Science and Engineering*, 33(2B), 391–409. [https://doi.org/10.1061/\(ASCE\)1076-0342\(2005\)11:4\(221\)](https://doi.org/10.1061/(ASCE)1076-0342(2005)11:4(221))
- Afshar, M. H., & Mariño, M. A. (2006). Application of an ant algorithm for layout optimization of tree networks. *Engineering Optimization*, 38(3), 353–369. <https://doi.org/10.1080/03052150600557783>
- Bi, W., Dandy, G. C., & Maier, H. R. (2015). Improved genetic algorithm optimization of water distribution system design by incorporating domain knowledge. *Environmental Modelling & Software*, 69, 370–381. <https://doi.org/http://dx.doi.org/10.1016/j.envsoft.2014.09.010>
- Dandy, G. C., Simpson, A. R., & Murphy, L. J. (1996). An improved genetic algorithm for pipe network optimization. *Water Resources Research*, 32(2), 449–458.



<https://doi.org/10.1002/wrcr.20175/abstract/PERMISSIONS>

- Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics-Part B*, 26(1), 29–41.
- Eiger, G., Shamir, U., & Ben-Tal, A. (1994). Optimal design of water distribution networks. *Water Resources Research*, 30(9), 2637–2646.
- Geem, Z. W., Kim, T. G., & Kim, J. H. (2000). Optimal layout of pipe networks using Harmony Search. En *Proceedings of 4th International Conference on Hydro-Science and Engineering, Seoul, South Korea*.
- Gonçalves, G. M., Gouveia, L., & Pato, M. V. (2014). An improved decomposition-based heuristic to design a water distribution network for an irrigation system. *Annals of Operations Research*, 219(1), 141–167. <https://doi.org/10.1007/s10479-011-1036-7>
- Gonçalves, G. M., & Pato, M. V. (2000). A three-phase procedure for designing an irrigation system's water distribution network. *Annals of Operations Research*, 94(1), 163–179. <https://doi.org/10.1023/A:1018989721586>
- Maier, H. R., Simpson, A. R., Foong, W. K., Phang, K. Y., Seah, H. Y., & Tan, C. L. (2001). Ant colony optimization for the design of water distribution systems. En *World Water Congress 2001* (pp. 1–10). [https://doi.org/10.1061/\(ASCE\)0733-9496\(2003\)129:3\(200\)](https://doi.org/10.1061/(ASCE)0733-9496(2003)129:3(200))
- Maier, H. R., Simpson, A. R., Zecchin, A. C., Foong, W. K., Phang, K. Y., Seah, H. Y., & Tan, C. L. (2003). Ant Colony Optimization for Design of Water Distribution Systems. *Journal of Water Resources Planning and Management*, 129(3), 200–209. [https://doi.org/10.1061/\(asce\)0733-9496\(2003\)129:3\(200\)](https://doi.org/10.1061/(asce)0733-9496(2003)129:3(200))
- Nicklow, J., Reed, P., & Savic, D. (2010). State of the art for genetic algorithms and beyond in water resources planning and management. *Journal of Water Resources Planning and Management*, 136(4), 412–432.
- Ostfeld, A. (2011). Ant colony optimization for water resources systems analysis – review and challenges. En *Ant colony optimization - Methods and applications* (pp. 147–158). InTech. <https://doi.org/10.5772/13522>
- Ponce Pacheco, M. A. (2013). *Diseño de redes de riego mediante algoritmos de evolución diferencial y colonia artificial de abejas*. Universidad Autónoma Chapingo.
- Simpson, A. R., Dandy, G. C., & Murphy, L. J. (1994). Genetic algorithms compared to other techniques for pipe optimization. *J Water Resour Plann Manage*, 120(4), 423–443.
- Walters, G. A., & Lohbeck, T. (1993). Optimal layout of tree networks using genetic algorithms. *Engineering Optimization*, 22(1), 27–48. <https://doi.org/10.1080/03052159308941324>
- Walters, G. A., & Smith, D. K. (1995). Evolutionary design algorithm for optimal layout of tree networks. *Engineering Optimization*, 24(4), 261–281. <https://doi.org/10.1080/03052159508941193>