

Optimización de las rutas para la intervención de pozos de petróleo

Miguel Jiménez-Carrión^{1*}, Luis Sánchez-Candela², Roxani Keewong-Zapata³, y José Bazán⁴

- (1) Facultad de Ingeniería Industrial, Dpto. Académico de Investigación de Operaciones, Universidad Nacional de Piura, Campus Universitario Miraflores, Castilla-Piura, Perú. (correo-e: mjimenezc@unp.edu.pe)
(2) Egresado de la Facultad de Ingeniería Industrial de la Universidad Nacional de Piura, (correo-e: lfsc_3@hotmail.com)
(3) Facultad de Ingeniería Industrial, Dpto. Académico de Ingeniería Mecatrónica, Universidad Nacional de Piura, Campus Universitario Miraflores, Castilla-Piura, Perú. (correo-e: rkeewongz@unp.edu.pe)
(4) Facultad de Ingeniería Industrial, Dpto. Académico de Ingeniería Industrial, de la Universidad Nacional de Piura (correo-e: jbazanc@unp.edu.pe)

* Autor a quien debe ser dirigida la correspondencia

Recibido Feb. 5, 2020; Aceptado Abr. 8, 2020; Versión final May. 12, 2020, Publicado Ago. 2020

Resumen

El objetivo es proponer un programa óptimo de intervención de pozos petroleros mediante la reasignación de rutas de los equipos SWAB en empresas dedicadas a estas operaciones. La primera fase hace uso del método de agrupamiento utilizando k-medias con el propósito de obtener un número de agrupamientos. La segunda fase utiliza algoritmos genéticos (AG) para encontrar el recorrido óptimo en cada agrupamiento de la primera fase y optimizar el número de agrupamientos. Los datos de tiempo de intervención, las coordenadas geográficas y las distancias de recorrido se obtuvieron visitando cada uno de los pozos. Los resultados muestran que el algoritmo propone un programa óptimo para el caso en estudio, obteniendo un ahorro de S/ 664 994.06 por año en estas operaciones. En conclusión, la implementación del algoritmo en dos fases (clústering y AG) es una solución para problemas de planificación de rutas en transporte con las características del proceso swab.

Palabras clave: k-medias; algoritmos genéticos; optimización; intervención de pozos; inteligencia artificial

Route optimization for oil well intervention

Abstract

The main objective of this research study is to propose an optimal oil well intervention program by reassigning the routes of SWAB teams in companies dedicated to these operations. The first phase makes use of the grouping method using k-means to obtain a number of groupings. The second phase uses genetic algorithms (GA) to find the optimal path in each cluster of the first phase and optimize the number of clusters. The intervention time data, the geographical coordinates of each well, and the travel distances were obtained by visiting each of the wells. The results show that the algorithm proposes an optimal program for the case under study, obtaining a saving of S/ 664 994.06 per year in these operations. In conclusion, algorithm implementation in two phases (clustering and GA) is a viable solution for solving problems in swab-like transport planification route processes.

Keywords: k-means; genetic algorithms; optimization; oil well intervention; artificial intelligence

INTRODUCCIÓN

Las empresas que se dedican a la actividad de servicios de extracción y transporte de petróleo con frecuencia realizan ajustes para mejorar la eficiencia de los procesos, tales como la reducción de tiempos de recorridos de las rutas y los costos operativos. Los recorridos que se siguen en estos tipos de actividades involucran nodos y aristas, cada nodo representa un pozo de petróleo a visitar y las aristas representan los caminos entre los pozos. El objetivo es recorrer todos los nodos de la red, partiendo de un mismo nodo de inicio, visitando una vez cada nodo y regresar al nodo de partida; esta situación configura el problema del Travel Salesman Problem (TSP). La complejidad del problema se incrementa, cuando los nodos están ubicados de forma muy dispersa y en grupos pequeños. Sobre el particular, en el problema de enrutamiento de vehículos (VRP), se plantea que una selección adecuada de la ruta del vehículo tiene un papel de influencia extenso para mejorar los intereses económicos y la adecuación de la planificación logística (Mohammed et al., 2017). Se ha observado que esta problemática se presenta en otras actividades como el caso de empresas comerciales que ofrecen el servicio Delivery.

Durante los últimos cincuenta años, el problema del agente viajero (TSP) se considera uno de los temas más importantes para explorar los problemas de optimización de la investigación operativa, principalmente debido a su naturaleza de formulación más pura, centrada en algunas de las aplicaciones más importantes relacionadas con la práctica, ocupaciones, tales como, por ejemplo, en algunas áreas de planificación industrial, logística de carga, enrutamiento de vehículos y configuración de máquinas conforme lo menciona (Júnior y Guimarães, 2019). Con relación a las variantes del TSP, las variantes de TSP y VRP se implementan utilizando el lenguaje de programación Delphi. Los métodos de TSP utilizados son heurística de inserción secuencial, heurística del vecino más cercano y enlace más barato. Las variantes de TSP que se implementan son múltiples (mTSP) y TSP con ventana de tiempo (TSPTW) (Wahyuningsih et al., 2016); el problema de enrutamiento de vehículo de entrega dividida (SDVRP) es una relajación del problema de enrutamiento de vehículo capacitado (CVRP) donde los clientes pueden ser asignados a múltiples rutas. Sobre el particular (Iv y Cavalier, 2012), desarrollan una nueva heurística de construcción para el SDVRP y se proporcionan resultados computacionales para treinta y dos conjuntos de datos de la literatura, la heurística de construcción se compara favorablemente con un método de generación de columnas y un método de dos fases, además, la construcción heurística es computacionalmente más rápida que los métodos mencionados.

En el mismo sentido y tratando de resolver la variante CVRP (Wang y Lu, 2009), propone un Algoritmo Genético Híbrido utilizando el método de vecino más cercano (NNA) incorporado al algoritmo de barrido (SA) para explicar simultáneamente las relaciones axiales y de radio entre los puntos de distribución con el depósito para generar una población cromosómica inicial bien estructurada y además empleó la metodología de superficie de respuesta (RSM) para optimizar la probabilidad de cruce y la probabilidad de mutación a través de experimentos sistemáticos. Por último, se obtuvo un algoritmo de barrido mejorado como algoritmo propuesto, con resultados satisfactorios; similarmente (Lin et al., 2009) aplica un algoritmo híbrido de recocido simulado y búsqueda tabú para resolver CVRP. Otras aplicaciones se encuentran en (Naderipour y Alinaghian, 2016), cuyos autores logran reducir 16% CO₂, NO_x y CO las emisiones, utilizando un algoritmo mejorado de optimización de enjambre de partículas en un problema de enrutamiento del vehículo dependiente del tiempo abierto (OTDVRP) y en (Zhang et al., 2014), se estudia el CVRP desde una perspectiva ambiental e introduce un nuevo modelo llamado problema ambiental de enrutamiento de vehículos (EVRP) con el objetivo de reducir el efecto adverso sobre el medio ambiente causado por el enrutamiento de vehículos. Un algoritmo híbrido de colonia de abejas artificiales (ABC) está diseñado para resolver el modelo EVRP.

Entre otras variantes se encuentra, el enrutamiento de la contaminación dependiente del tiempo (TDPRP) que consiste en enrutar una flota de vehículos para atender a un conjunto de clientes y determinar las velocidades en cada tramo de las rutas propuesto por (Franceschetti et al., 2017). Sin embargo, en este trabajo la variante que más se aproxima es mTSP y en esta investigación se trata de obtener el valor óptimo de viajantes, así como sus recorridos. El mTSP considera conocidos m agentes los cuales parten de una misma ciudad y recorren todas las ciudades una única vez y regresan a la ciudad de partida con el mínimo tiempo de recorrido, es uno de los problemas de optimización combinatoria más populares y ampliamente utilizado en la investigación operativa, es un problema de optimización NP-Hard; en consecuencia, se aplican enfoques heurísticos conforme lo manifiestan (Rostami, et al., 2015). Sin embargo, la variante que nos ocupa es diferente a lo buscado en mTSP, dado que nuestro proyecto de investigación se parte del tiempo máximo tolerable o permitido para recorrer todos los nodos de la red y se busca determinar el valor óptimo de m , número óptimo de multiagentes, para cumplirlo. Sobre el particular, no se conoce algoritmos que tenga la posibilidad de buscar el número óptimo de multiagentes a partir de conocer el tiempo máximo tolerable en recorrer todos los nodos, más aún cuando los nodos se encuentran dispersos y agrupados en pequeños grupos; por lo que es necesario hacer agrupamientos antes de encontrar el recorrido mínimo en

cada agrupamiento. En ese sentido (Jaradat et al., 2019), propone una solución para TSP utilizando Algoritmo de Firefly (FA) y agrupamiento de k-means. El enfoque propuesto comprende tres pasos principales: agrupar los nodos, encontrar la ruta óptima en cada grupo y volver a conectar los grupos. El primer paso usa el agrupamiento k-means para dividir los nodos en subproblemas, el segundo paso utiliza FA para encontrar la ruta óptima en cada grupo, finalmente reconectar todos los grupos y devolver la ruta entre ellos. Bajo el mismo enfoque de ruteo de vehículos, (Albornoz y Johns, 2011) propone un modelo de programación entera de localización y ruteo en la cual la solución óptima alcanzada es para instancias pequeñas y medianas, sin embargo para instancias grandes los mismo autores recomiendan el empleo de alguna estrategia alternativa; en el mismo sentido para un problema de diseño de rutas para vehículos (Ortega et al., 2009) implementan una metaheurística basada en múltiples sistemas de colonias de hormigas, cada una con una función objetivo organizada de manera jerárquica agregan programación de restricciones para determinar un movimiento factible a un nuevo cliente, para el problema de ruteo de vehículos con ventanas de tiempo y programación de carga (VRPTWSL).

La presente investigación se refiere a la reasignación de rutas para los equipos swab en el Lote VII, ubicado en la costa de Talara. El Lote VII, cuenta con 260 pozos para realizar el servicio de extracción de petróleo mediante el método de levantamiento artificial por swab o pistoneo, actualmente se cuenta con 10 equipos de swab (unidad de swab y cisterna). La empresa de transporte dedicada a los servicios de extracción y transporte de petróleo realiza un conjunto de actividades diarias en cada pozo que interviene, al ser concluidas se dirigen a otro pozo para realizar las mismas actividades y así sucesivamente hasta cumplir con los pozos asignados para cada día, (día 1: 115 pozos, día 2: 116 pozos y día 3: 117 pozos; y se repite). Sin embargo, muchas veces para poder cumplir tiene que asignar turnos nocturnos adicionales a los programados. Actualmente la empresa realiza la intervención de pozos basada en la experiencia y criterio de los operarios, no garantizando la mejor alternativa de las rutas empleadas en la intervención de los pozos afectando la eficiencia de la empresa. Es de gran importancia reasignar rutas que permita optimizar el programa actual de intervención de pozos. Muchas empresas han conseguido mejorar sus procesos, que antes eran realizados únicamente basados en la experiencia e intuición, gracias a la aplicación de modelos matemáticos; es por ello que la utilización de una variante del TSP propuesta como t-VRP, donde se encontrará el recorrido de cada vehículo en un tiempo máximo tolerable de 12 horas (el mismo que se corresponde con la jornada laboral en ese tipo de servicios; además, el alquiler del equipo swab es 12 horas por día); y herramientas prestadas de la inteligencia artificial para responder a la pregunta ¿Cómo la reasignación de rutas de los equipos optimiza el programa de intervención de pozos petroleros?, para ello es necesario localizar la ubicación de los pozos, hallar sus distancias, y medir el tiempo de intervención, es decir, el tiempo que permanece el equipo swab en cada pozo, para determinar el tiempo que consumen las rutas actuales y compararlas con el programa óptimo de intervenciones de los pozos.

El uso de herramientas de aprendizaje automático se ha ido intensificando con el propósito de que respondan a una mejor gestión de los recursos utilizados, en este contexto; se desarrolló una metodología apoyada en clústering entre otras herramientas para clasificar estudiantes participantes en ambientes virtuales de aprendizaje identificando relaciones entre la intensidad de uso de la plataforma y los resultados en los exámenes semestrales; encontrándose que el tiempo que un estudiante permanece en la plataforma no está relacionado con pertenecer al grupo de conocimiento alto (De-La-Hoz et al., 2019), así mismo (Castro et al., 2014) presentan una metodología que utiliza los algoritmos K-means entre otras, en el manejo de información geográfica permitiéndoles identificar diversos materiales de construcción logrando así realizar predicciones sobre materiales de construcción únicamente con la ubicación geográfica y cuantificando la intensidad de material.

Cómo el problema que se presenta tiene una alta complejidad primero porque se tiene que utilizar algoritmos de agrupamiento y en segundo lugar resolver el TSP, en el primer caso se utiliza el algoritmo K-means y en el segundo caso se utiliza los algoritmos genéticos; con relación a establecer agrupamientos existen métodos como el K-means el cual es uno de los más simples y conocidos algoritmos de agrupamiento, sigue una forma fácil y simple para dividir una base de datos dada en k grupos (fijados a priori). La idea principal es definir k centroides (uno para cada grupo) y luego tomar cada punto de la base de datos y situarlo en la clase de su centroide más cercano. El próximo paso es recalcular el centroide de cada grupo y volver a distribuir todos los objetos según el centroide más cercano, el proceso se repite hasta que ya no hay cambio en los grupos de un paso al siguiente, (Li y Wu, 2012); más delante los mismos autores manifiestan que este algoritmo depende bastante de los puntos iniciales y de la diferencia en la elección de las muestras iniciales que siempre conduce a resultados diferentes. Sobre el particular (Yang y Sinaga, 2019), sostienen que K-means tiene dificultades cuando a menudo existen componentes de características irrelevantes en los conjuntos de datos, que pueden causar un mal rendimiento; así como una alta dimensión de características en los conjuntos de datos por lo que proponen, reducir su dimensión para los algoritmos de agrupamiento. Además, el algoritmo K-means está basado en una función que minimiza la distancia entre el centroide y cada punto y, siempre usa el método de gradiente para obtener el extremo

mínimo. La dirección de búsqueda en el método de gradiente siempre es a lo largo de la dirección en la que disminuye la función, lo que conducirá al hecho de que cuando el punto focal del clúster inicial no es adecuado, todo el algoritmo se hundirá fácilmente en el punto mínimo local (Li y Wu, 2012). El problema del empleo de estos esquemas es que fallan cuando los puntos de un grupo están muy cerca del centroide de otro grupo. El algoritmo lleva a cabo los pasos que se muestran en la Figura 1, hasta alcanzar el criterio de convergencia (los objetos no se cambian de grupo, lo que significa que cada punto o nodo, encontró el centroide con distancia mínima).

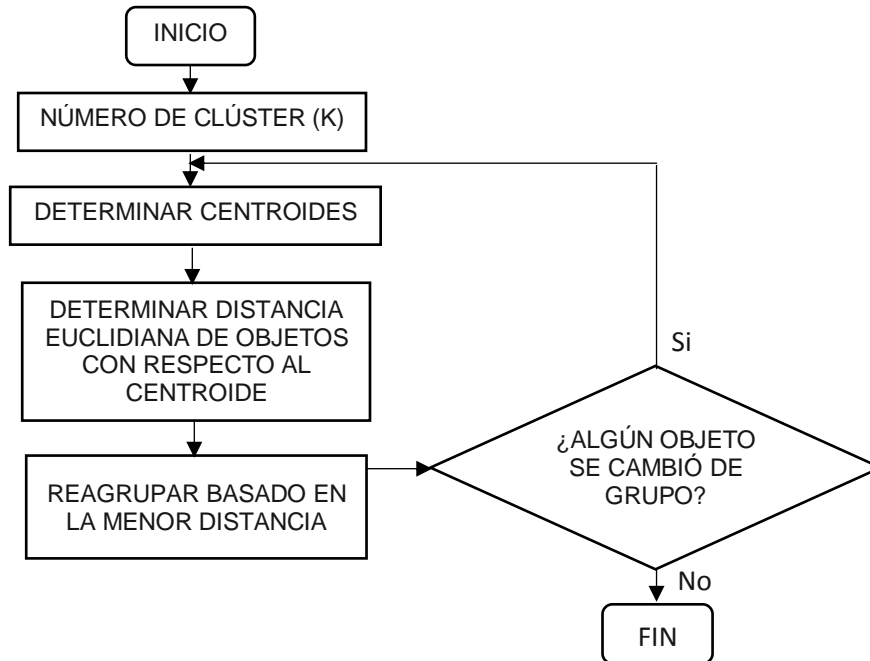


Fig. 1: Diagrama de flujo del algoritmo k-Means

Los Algoritmos Genéticos (AG's) son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización. Están basados en el proceso evolutivo de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza acorde con los principios de la selección natural y la supervivencia de los más fuertes. Holland (1975) en su trabajo *Adaptation in Natural and Artificial Systems* sostiene que la evolución de individuos (que representan soluciones) hacia valores óptimos del problema depende en buena medida de una adecuada codificación (García, 2008), y el rendimiento de los AG's depende en gran medida del tipo de operadores genéticos utilizados (Mulloorakam y Mathew, 2019). Así mismo, (Jiménez-Carrión, 2018) trabajando con un algoritmo genético simple manifiesta que la estrategia de una selección por torneo, unido a un porcentaje de mutación progresivo y el diseño dual de dos mecanismos de mutación mantienen una diversidad de individuos que evitan que el algoritmo genético simple se quede atrapado en un óptimo local. Así mismo, Goldberg (1989), en su trabajo *Genetic Algorithms in Search, Optimization and Machine Learning*, define los algoritmos genéticos como "métodos estocásticos de optimización global basados en los principios de la selección y evolución natural" (García, 2008); del mismo modo (Salgado y Afonso, 2019) sostienen que los algoritmos genéticos son combinados con una perspectiva egoísta de evolución de los genes para problemas de planificación de rutas; otras aplicaciones de los algoritmos genéticos relacionadas con la optimización de redes eléctricas se puede ver en (Anaut et al., 2009). Algoritmo genético multiobjetivo relacionadas con la optimización de sistemas energéticos en (Bastidas et al., 2010). Del mismo modo se implementó un Modelo de un algoritmo genético con selección discriminatoria de individuos bajo un esquema de ponderación de probabilidades de mutación en (Apaza, 2013).

Los AG's son procedimientos adaptativos para la búsqueda de soluciones en espacios complejos, inspirados en los procesos genéticos de los organismos naturales y en los principios de la evolución natural de poblaciones. La idea básica es mantener una población de cromosomas, los cuales representan soluciones candidatas a un problema concreto, que evolucionan con el tiempo a través de un proceso de competición y variación controlada. Cada cromosoma tiene una bondad o adaptación asociada, que describe la adecuación de la solución a la que representa. El proceso de competición, denominado mecanismo de selección utiliza estas adaptaciones para determinar los cromosomas que se usan para crear otros nuevos; Goldberg (1989) en su trabajo *Genetic Algorithm in Search, Optimization and Machine Learning*, menciona que los nuevos cromosomas se generan a través de los operadores genéticos denominados cruce y mutación (García, 2008).

Los AG's se han aplicado con éxito en problemas de búsqueda y optimización. Gran parte de este éxito se debe a su capacidad para explotar la información acumulada sobre un espacio de búsqueda, y así dirigir las siguientes búsquedas hacia los mejores subespacios. Ésta es su principal ventaja, sobre todo en espacios grandes, complejos y parcialmente definidos donde las técnicas clásicas de búsqueda no son apropiadas. Sobre el particular (Erazo, 2013) en su investigación "Propuesta de un programa de evaluación de pozos basado en la optimización del recorrido de los operadores de producción en el lote I, Talara, teniendo como objetivo general proponer un programa de recorrido de evaluación de pozos basado en la optimización del recorrido de los operadores de producción en el lote I, Talara; concluye la investigación obteniendo un beneficio de S/ 22 990.57, y recomienda "realizar estudios similares de investigación operativa, en los diversos campos de exploración y producción de petróleo, ya que este estudio, permite tener un mejor control de las operaciones haciéndolas óptimas".

Se han aplicado Algoritmos Genéticos y Templado Simulado para mejorar los indicadores de servicio en telecomunicaciones tales como la probabilidad de bloqueo y minimización de la utilización de la red, en razón a que las telecomunicaciones modernas están progresivamente sometidas al aumento de la demanda, lo que se traduce en mayores requerimientos en las redes fotónicas de transporte. Los resultados muestran que los algoritmos heurísticos son mejores bajo demanda dinámica de alta carga y que funcionan mejor bajo stress. Se concluye que los algoritmos genéticos dan la mejor solución a la demanda dinámica futura del transporte de datos (Rodríguez et al., 2014). Así mismo, se han aplicado algoritmos genéticos para minimizar la distancia recorrida en almacenes y centros de distribución donde se aplica el problema de conformación de lotes para la preparación de pedidos. Para esto, se propone una nueva representación de soluciones, en la cual cada gen de un cromosoma representa una orden de cliente a recuperar; se determinó que el algoritmo genético genera ahorros significativos en distancia recorrida y número de lotes respecto a una regla básica de conformación de lotes, especialmente en escenarios donde se exige conformar un mayor número de lotes. Se concluye que el algoritmo genético brinda soluciones eficientes en un tiempo computacional razonable, por lo cual se recomienda su implementación en ambientes operativos de almacenes y centros de distribución (Cano et al., 2018).

METODOLOGIA

Inicialmente se muestra la representación matemática del problema como una serie de subconjuntos de problemas TSP independientes, en el que cada variable X_{ij} en el modelo TSP, se asocia con una variable X_{ijk} para indicar a que subconjunto pertenece, el resto de los parámetros se mantienen idénticos. Esto hace que la Función objetivo del problema es aquella cuya sumatoria en todos los subconjuntos de agrupamientos se minimice la distancia total recorrida. En la formulación matemática del mTSP, Kiraly, y Abonyi, (2011) en su trabajo Optimization of Multiple Traveling Salesmen Problem by a Novel Representation based Genetic Algorithm, definen un modelo en el que, se optimiza una función de costo incluyendo el costo del recorrido total y el costo de asignar "m" agentes como una función lineal (Rostami et al., 2015). Luego se hizo la recolección de la data que permite implementar el código de la fase 1, el cual determina primero cuanto tiempo le toma a un solo agente recorrer todos los nodos de la red en un día determinado, luego se estima el número de agentes necesarios para no exceder el tiempo máximo tolerable y con esta información, finalmente se ejecuta el algoritmo de agrupamiento K-means. En razón a que en el agrupamiento de la fase 1, existan agrupamientos que excedan el tiempo máximo tolerable y otros que se encuentren muy por debajo de ese valor se ejecuta el código de la fase 2, y este lo que hace es optimizar el número de agrupamientos.

Paso 1: Representación matemática del problema

Como se planteó anteriormente, el TSP se define mediante el número de pozos "n" y la matriz de distancias $\|d_{ij}\|$ o la matriz de tiempos $\|t_{ij}\|$. La definición de recorridos válidos prohíbe conectar un pozo a sí mismo por lo que se penaliza con un valor muy alto a los elementos de la diagonal de la matriz de distancias o de tiempos. Un TSP es simétrico si $d_{ij}=d_{ji}$; o $t_{ij}=t_{ji}$. de lo contrario, el modelo TSP es asimétrico. Se define una variable binaria, x_{ijk} esta variable toma los siguientes valores:

x_{ijk} = toma el valor de 1, si se llega al nodo j desde el nodo i, en el subconjunto k; toma el valor de 0, en caso contrario.

El modelo mTSP, para $m=4$ subconjuntos, se da como: Función Objetivo, ecuación (1); Restricciones, ecuaciones (2) a la (17), $n= k_1 + k_2 + k_3 + k_4$; además, los pozos en cada subconjunto son mutuamente excluyentes, es decir los pozos solo pueden pertenecer a un solo agrupamiento. Un viaje redondo significa que es un recorrido cerrado, se sale del terraplén recorre el subconjunto de pozos asignados una sola vez y regresa al terraplén de donde partió, no está permitido la formación de sub recorridos (subredes). En realidad, un viaje redondo significa agregar restricciones del tipo: $u_i - u_j + k_m(x_{ijkm}) \leq k_m - 1$; $\square m=1,2,3,4$ donde u_i y u_j son enteros no negativos, para evitar sub recorridos.

$$\text{Min Recorrido} = \sum_{i=1}^{k1} \sum_{j=1}^{k1} d_{ijk1} x_{ijk1} + \sum_{i=1}^{k2} \sum_{j=1}^{k2} d_{ijk2} x_{ijk2} + \dots + \sum_{i=1}^{k4} \sum_{j=1}^{k4} d_{ijk4} x_{ijk4}; \quad d_{ij} = \infty \quad \forall i = j \quad (1)$$

$$\sum_{j=1}^{k1} x_{ijk1} = 1, i = 1, 2, \dots, k1 \quad (2)$$

$$\sum_{i=1}^{k1} x_{ijk1} = 1, j = 1, 2, \dots, k1 \quad (3)$$

$$x_{ij} = (0,1) \quad (4)$$

La solución forma un viaje redondo para un subconjunto de k1 pozos (5)

$$\sum_{j=1}^{k2} x_{ijk2} = 1, i = 1, 2, \dots, k2 \quad (6)$$

$$\sum_{i=1}^{k2} x_{ijk2} = 1, j = 1, 2, \dots, k2 \quad (7)$$

$$x_{ijk2} = (0,1) \quad (8)$$

La solución forma un viaje redondo para un subconjunto de k2 pozos (9)

⋮ ⋮ ⋮

$$\sum_{j=1}^{km} x_{ijkm} = 1, i = 1, 2, \dots, km \quad (14)$$

$$\sum_{i=1}^{km} x_{ijkm} = 1, j = 1, 2, \dots, km \quad (15)$$

$$x_{ijkm} = (0,1) \quad (16)$$

La solución forma un viaje redondo para un subconjunto de km pozos (17)

Gráficamente en la Figura 2, se observa un modelo TSP de 5 ciudades. Los nodos representan pozos, y las líneas que unen los pozos (arcos) representan rutas en dos sentidos que pueden ser distintas si el modelo es asimétrico. Así mismo, se observa que hay dos soluciones una con un solo agente y otra con dos agentes. Obsérvese que con dos agentes ambos salen del nodo 5 y los recorridos son: 5-1-2-5, 5-4-3-5.

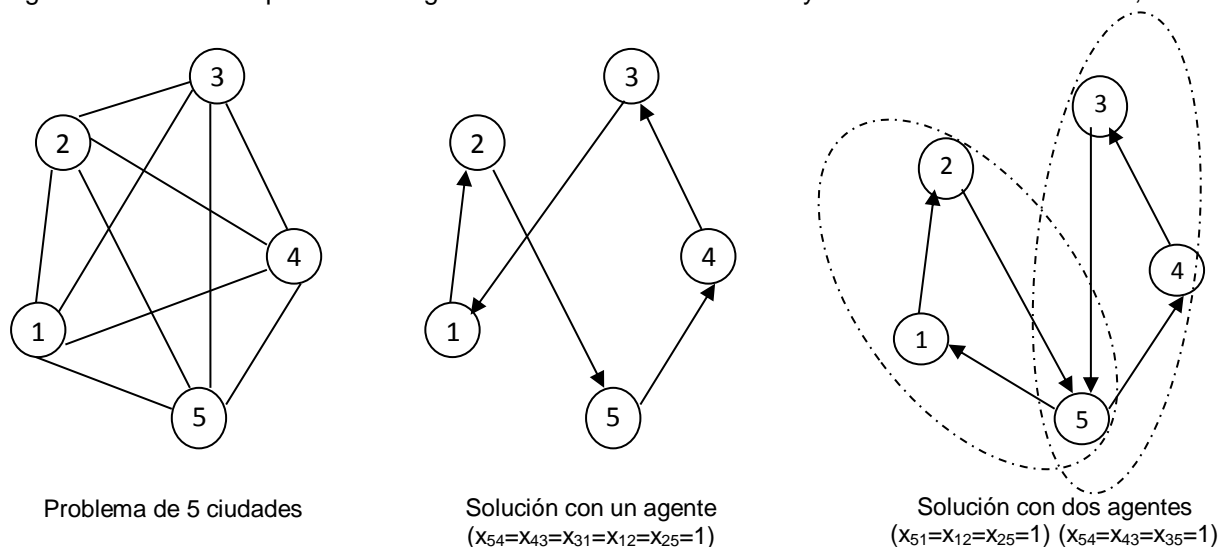


Fig. 2: Ejemplo del TSP con 5 ciudades, solución con 1 agente y con 2 agentes

Paso 2: Recolección de la data

La data se obtiene a partir de las coordenadas de localización de la latitud y la longitud obtenidas por GPS de cada pozo incluyendo el terraplén. Utilizando el software Google Earth se trazan las rutas entre los pozos para determinar las distancias, obteniendo finalmente 3 matrices con sus respectivas distancias, esta información se anota en la Tabla 1. Así mismo, se utiliza la Tabla 2 para recolectar la información relacionada con la localización de cada pozo medido en términos de latitud y longitud, tiempo en minutos durante la intervención de un pozo que consiste en el conjunto de actividades realizadas en éste como: los procedimientos de seguridad y medio ambiente (estacionarse teniendo en cuenta la manga de viento, colocar conos de seguridad, extintores, geomembranas, etc), y operación swab (armar el equipo y accesorios swab, operación de extracción del crudo, operación de llenado de cisterna, retiro de accesorios y equipo swab). Sin perder generalidad se ha estimado la velocidad promedio de viaje de cada equipo en el lote número VII, área de estudio del proyecto, en 25km/h; además la Tabla 2, registra el recorrido de cada equipo swab por lo que se tendrá 10 tablas, una para cada equipo swab; esto explica por qué la Tabla 2 llega a registrarse hasta 17 filas, algunos equipos swab llenaran todas las filas otras menos dependiendo de los pozos que le asignan.

Tabla 1: Matriz de distancias entre los pozos incluido el Terraplén, nodo de partida (Pozo 0) en Km.

Pozo	0	1	2	3	...	113	114	115
0					...			
1					...			
2					...			
3					...			
⋮	⋮	⋮	⋮	⋮	...	⋮	⋮	⋮
113					...			
114					...			
115					...			

Tabla 2: Tiempo en minutos de intervención del pozo, localización y recorrido del equipo swab

	Equipo TS:				Cisterna:	
	Winchero:				Cisternero:	
					Fecha:	
Nº	hora de llegada	hora de salida	pozo intervenido	coordenadas		tiempo de intervención del pozo
				Latitud	longitud	
0	-		Terraplén			
1						
⋮						
16						
17		-	Terraplén			

Paso 3: Desarrollo del código para el Algoritmo fase 1 (k-means)

La implementación del código para correr esta primera fase del algoritmo, lo que se hace es convertir el tiempo de intervención en cada pozo en distancia utilizando la velocidad media de 25 km/h, luego se suma esa distancia a la matriz de distancias entre los pozos incluyendo el terraplén para obtener una única matriz de distancias; luego se utiliza la matriz de distancias encontrada, con una función basada en algoritmos genéticos para determinar el recorrido mínimo que le toma a un solo equipo swab en recorrer todos los pozos asignados. Posteriormente se divide ese recorrido empleado por un solo agente entre 300 km, que representa el recorrido que un equipo swab haría en el lapso de 12 horas, a la velocidad media definida; esto permite determinar un valor inicial para el número necesario de equipos swab, es decir "m". Determinado el número de agrupamientos, el código hace una llamada a la matriz de coordenadas (latitud y longitud), para correr el algoritmo K-means, utilizando una función implementada en Matlab; obtenidos los agrupamientos se evalúa cada uno, haciendo uso nuevamente de la función basada en algoritmos genéticos aludida anteriormente. El resultado de esta primera fase es que algunos agrupamientos cumplen con el tiempo máximo tolerable, otros están muy por debajo de ese valor y por último otros agrupamientos exceden en demasía el tiempo máximo tolerable, demostrando que no es buena solución. Este estado, obliga a optimizar los resultados de la primera fase corrigiendo las desviaciones por arriba o por debajo de lo permitido, tratando de hacer trabajar cada equipo swab a tope es decir todo el día, en razón a que da lo mismo que trabaje pocas horas o el total, puesto que el alquiler del equipo swab es 12 horas por día.

Paso 4: Desarrollo del código para el Algoritmo fase 2 (Algoritmo Genético)

En este paso se ha definido en Matlab una función $[a, b] = \text{camino}(M)$, basado en algoritmos genéticos que determina el recorrido mínimo en kilómetros (parámetro "a") y la secuencia de pozos a recorrer saliendo del terraplén (parámetro "b") que es un vector; dándole como argumento una matriz de distancias "M". Utiliza un mecanismo de selección por torneo con 3 competidores, en la fase de reproducción se utiliza emparejamiento parcial con dos puntos de cruce, y en la fase de mutación se realiza 50% por inversión y 50% por intercambio cada vez que este se presenta; esta función es llamada tanto en el paso 3, para evaluar los clústeres formados, y con mayor intensidad en la fase 2. El código que permite optimizar el número de agrupamientos se inicia llamando al primer agrupamiento y obtiene su distancia, si ésta es menor de 300 km significa que aún le faltan pozos por visitar entonces entra en un bucle en el cual comienza adicionando un pozo y llama al pozo que se encuentra más cercano al centroide del primer agrupamiento, y ajusta los pozos del grupo a que perteneció, se actualiza el grupo actual y el grupo que cedió el pozo, recalculando la secuencia y su distancia recorrida, este bucle se ejecuta hasta que la distancia sea superior a 300 km. Cuando la distancia del agrupamiento es mayor de 300, significa que el agrupamiento excede el tiempo máximo tolerable y debe eliminar pozos del grupo, es aquí donde entra en un bucle, eliminando el pozo que se encuentra más lejos del centroide y determina además el grupo que está más cercano para entregarlo; igualmente actualiza ambos grupos, recalculando la secuencia y su distancia recorrida, este bucle se ejecuta hasta que la distancia del grupo sea menor a 300 km. Para los otros casos es decir cuando la distancia es igual a 300 km deja el grupo como está y pasa al siguiente grupo hasta terminar con todos los grupos. El resultado de la ejecución de este código es que se tiene agrupamientos cercanos a 300 km y grupos vacíos lo que significa que se ha optimizado el número de equipos swab.

Paso 5: Análisis de estabilidad de los datos proporcionados por el Algoritmo

La estabilidad del sistema en los resultados obtenidos se determina con un análisis de varianza usando la ecuación (18), después del paso 4, con el fin de establecer si existen diferencias significativas en los resultados. En esta ecuación el parámetro y_{ijkrm} representa la respuesta del algoritmo genético expresada en términos del recorrido mínimo en el clúster y su equivalente en horas; A_i , es el efecto causado por el i-ésimo factor A (Número de generaciones 1000, 1500 y 2000) y B_j es el efecto causado por el j-ésimo factor B (tamaño de la población 50, 100 y 150); $(C)_k$ representa el efecto causado por el k-ésimo factor C (porcentaje de mutación 0.01, 0.05, 0.10); $(D)_r$ representa el efecto causado por el r-ésimo factor D (porcentaje de cruce 0.85, 0.90, 0.95); $(AB)_{ij}$, $(AC)_{ik}$, $(AD)_{ir}$, $(BC)_{jk}$, $(BD)_{jr}$, $(CD)_{kr}$, representa el efecto causado por la interacción de dos factores; $(ABC)_{ijk}$, $(ABD)_{jir}$, $(ACD)_{jkr}$, $(BCD)_{jkr}$, representan el efecto causado por la interacción de tres factores; $(ABCD)_{ijkrm}$ representa el efecto causado por la interacción de los cuatro factores y n repeticiones; e_{ijkrm} representa el error causado por el i-ésimo factor A, j-ésimo factor B, k-ésimo factor C y r-ésimo factor D, en la réplica n. Para el algoritmo genético la información recolectada debe cumplir los requerimientos de independencia y normalidad requeridos. El análisis de varianza es realizado bajo un intervalo de confianza del 95% y 99%.

$$y_{ijkrm} = \mu + A_i + B_j + C_k + D_r + (AB)_{ij} + (AC)_{ik} + (AD)_{ir} + (BC)_{jk} + (BD)_{jr} + (CD)_{kr} + (ABC)_{ijk} + (ABD)_{jir} + (ACD)_{jkr} + (BCD)_{jkr} + (ABCD)_{ijkrm} + e_{ijkrm} \quad (18)$$

$i = 1,2,3; j = 1,2,3; k = 1,2,3; r = 1,2,3; n = 1,2,3,4$

Adicionalmente se realizó un análisis de Duncan para determinar el nivel en el cual cada factor proporciona mejor respuesta en el recorrido mínimo, que son las respuestas del algoritmo en dos fases.

ANÁLISIS Y DISCUSIÓN DE RESULTADOS

La ejecución del algoritmo K-means del paso 3, siempre conducen a resultados diferentes; en razón a que la elección de los puntos iniciales son aleatorios, lo cual es coincidente con lo manifestado por (Li y Wu, 2012), sin embargo, debido a que solo se utiliza las coordenadas de latitud y longitud (dos parámetros), no está influenciado por el problema de multidimensionalidad que menciona (Yang y Sinaga, 2019), es decir solo hay dos dimensiones. Los datos en la Tabla 3, ilustran una instancia de la salida del algoritmo K-means, para el programa de 115 pozos; observándose que la solución no es factible en razón a que hay agrupamientos que exceden el tiempo máximo tolerable, como el grupo 6, que emplea 16.38 hrs (409.4577/25), y el grupo 9, que emplea 13.27 hrs (331.3350/25); otros agrupamientos están muy por debajo del tiempo máximo tolerable como los grupos: 1, 2 y 5 con 6.04, 5.65 y 6.40 horas respectivamente, así mismo, hay agrupamientos que se les puede considerar aceptables por estar cercanos al tiempo máximo tolerable; la Figura 3, muestra la distribución de los agrupamientos de la instancia. El resultado es una solución no factible que necesita pasar a la fase 2.

Tabla 3: Instancia de la salida del código de la fase 1, programa 115 pozos

Grupo	Secuencia	Distancia	Cantidad	horas
1	[11-10-2-3-4-12-15-14]	151.0917	8	6.04
2	[116-110-112-113-114-111]	141.4743	6	5.65
3	[91-22-40-39-38-45-57-25-93-92-70-58-21]	262.4047	13	10.50
4	[77-107-76-75-80-74-79-106-53]	281.8960	9	11.26
5	[109-73-108-115-72-78-105]	160.0277	7	6.40
6	[6-5-17-16-18-7-9-19-20-23-24-8-90-89-88-86-87-85]	409.4577	18	16.38
7	[66-69-26-47-49-103-48-27-50-46-65-56-102]	246.5947	13	9.86
8	[67-31-29-30-35-37-36-32-33-34-28-68]	204.7717	12	8.19
9	[41-94-96-95-44-42-43-64-62-63-97-101-104-100-98-61-60-13-59-99]	331.6350	20	13.27
10	[52-51-81-55-82-54-83-84-71]	229.0440	9	9.16
	Recorrido de todos los agrupamientos (km)	2418.4475	115	96.74

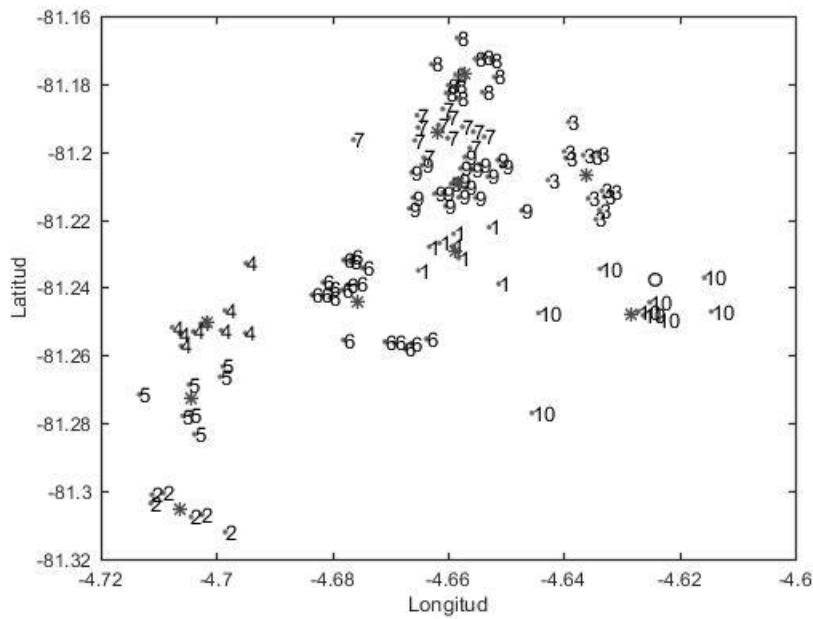


Fig. 3: Agrupamiento de la instancia de salida del K-means para 115 pozos, (o terraplén, * centroides)

La fase 2 del algoritmo, logra una solución factible y optimiza el número de agrupamientos para el programa de 115 pozos, logrando minimizar el número de equipos swab, reduce el recorrido total a 2412.3343 km, y el tiempo de recorrido, conforme se puede apreciar en la Tabla 4; así mismo, se muestra que el clúster 5 se encuentra vacío, indicativo de que no es necesario.

Tabla 4: Instancia de la salida del código de la fase 2, para el ejemplo de 115 pozos

Clúster	Secuencia	Distancia	Cantidad	horas
1	[99-59-13-60-61-98-100-104-101-97-63-62-43-42-64-96-94-41]	297.9943	18	11.92
2	[105-80-108-109-116-110-114-111-113-112-73-115-78]	268.7080	13	10.75
3	[21-58-70-92-93-69-25-46-57-45-38-39-40-22-91]	294.4200	15	11.78
4	[53-75-107-77-76-74-79-106-72]	294.7970	9	11.79
5	[]	0	0	0
6	[9-19-20-23-24-8-90-89-88-86-87-85]	289.1837	12	11.57
7	[102-56-65-103-28-67-31-29-30-36-48-27-50-47-49-26-66]	298.6043	17	11.94
8	[14-15-7-18-6-17-16-44-95-11-10-2-3-4-12]	294.0000	15	11.76
9	[84-54-82-55-52-5-35-37-32-33-34-68-71]	291.3377	13	11.65
10	[51-81-83]	83.2893	3	3.33
	Recorrido de todos los agrupamientos (km)	2412.3343	115	96.49

El algoritmo genético (paso 4) se evalúa según el análisis de varianza (ANVA) de la Tabla 5, la que muestra que, los factores principales A, B, y C, tiene una influencia altamente significativa de encontrar la solución óptima con el AG; sin embargo, el factor D no ejerce influencia significativa en los niveles estudiados para estos resultados. Con respecto a las interacciones se encuentra que cuando interactúan los factores A y C influyen significativamente en la solución del AG. Así mismo, cuando interactúan A, C y D también se encuentra influencia significativa en la solución; sin embargo, cuando interactúan los factores B y C la

influencia en la solución del AG, es altamente significativa. El resto de las interacciones no muestran influencia significativa en el resultado de la solución del AG. El coeficiente de variabilidad fue de 6.72% respecto de la media.

Tabla 5: Análisis de varianza en disposición factorial de 4 factores de tres niveles para el AG. (*) indica significación en el nivel 0.05, (**) indica significación en el nivel 0.01

Fuentes De Variación	Gl.	Sc.	C.M	Fc.	Sig.
Tratamientos	80	106695.99767			
A	2	3138.85342	1569.42671	4.80618	**
B	2	22396.13835	11198.06917	34.29273	**
C	2	47568.74664	23784.37332	72.83677	**
D	2	1019.48826	509.74413	1.56103	
A x B	4	3014.30300	753.57575	2.30773	
A x C	4	4091.41600	1022.85400	3.13237	*
A x D	4	1479.38900	369.84725	1.13261	
B x C	4	6858.75100	1714.68775	5.25102	**
B x D	4	2113.39800	528.34950	1.61801	
C x D	4	1317.00600	329.25150	1.00829	
A x B x C	8	662.30800	82.78850	0.25353	
A x B x D	8	662.04100	82.75513	0.25343	
A x C x D	8	6001.49200	750.18650	2.29736	*
B x C x D	8	1980.32100	247.54013	0.75806	
A x B x C x D	16	4392.34600	274.52163	0.84069	
Error	243	79350.06887	326.54349		
Total	323	186046.06654			
		CV =	6.72%		

El análisis de Duncan, Tabla 6, se evalúa para un nivel de significación del 5%, encontrándose que el número de generaciones indica que los niveles de 1500 y 2000 generaciones se comportan estadísticamente iguales, sin embargo, numéricamente el nivel de 1500 generaciones obtuvo el menor recorrido medio de 266.71 km. Con respecto al tamaño de la población todos los niveles son estadísticamente diferentes, sin embargo, el nivel de 150 individuos alcanzó el menor recorrido medio de 260.07 km; del mismo modo analizando el factor porcentaje de mutación los tres niveles se comportan estadísticamente diferentes pero el porcentaje que obtuvo el mejor promedio con 256.46 km en el recorrido fue el nivel de 10%. Aun cuando el cruzamiento o reproducción el ANVA no mostró significancia, el Duncan los muestra estadísticamente iguales, sin embargo, numéricamente el, $pC=0.95$ alcanzó el menor promedio con 267.00km. En la interacción NGxpM, el menor promedio lo tiene 200x0.10 con 249.69 km, del mismo modo la interacción TpxpM, que registró el menor promedio fue 150x0.10 con 254.22 km. Finalmente en la interacción AxCx D, los mejores promedios que se comportan estadísticamente iguales son: 2000x0.10x0.90 con 252.75 km, 1500x0.05x0.95 con 255.77 km, 1000x0.10x0.90 con 257.47 km, 1500x0.05x0.85 con 259.89 km, 1500x0.10x0.90 con 260.57 km, 2000x0.05 x0.90 con 260.64 km, 1500x0.10x0.85 con 260.66 km, 2000x0.05x0.85 con 261.94 km y 1000x0.10x0.85 con 262.01 km. Resumiendo, los mejores parámetros para el algoritmo son: número de generaciones 1500, tamaño de la población 150 individuos, porcentaje de mutación 10% y porcentaje de cruce 95%.

Determinados los parámetros del algoritmo y haciendo un análisis extensivo de manera integral (K-means y AG), para los programas 1, 2 y 3 en el cual, cada programa agrupa a 115, 116 y 117 pozos para extracción del crudo; se realizaron 10 repeticiones para cada caso, los resultados se muestran en la Tabla 7, donde se puede observar que el programa 1, tiene un recorrido promedio de 2406.0543 km con una desviación estándar de 4.9613, el programa 2, un recorrido promedio de 2399.3039 km con una desviación estándar de 8.8474, y el programa 3, muestra un recorrido promedio de 2375.6292 km y una desviación estándar de 9.5875, con lo que se concluye que es un algoritmo con una alta precisión que logra minimizar el número de equipos swab requeridos. En la tabla 8, 9 y 10, se muestran los mejores resultados (recorridos mínimos: 2399.6463, 2386.7273 y 2363.1442 km) para la empresa Transportes Saldarriaga, cada tabla comprende 6 columnas la primera columna indica el número de clústeres, la segunda columna indica el equipo swab asignado, la tercera columna contiene el número de pozos en cada clúster, la cuarta columna indica la secuencia que debe seguirse para recorrer los pozos, la quinta columna indica el tiempo recorrido más el tiempo de intervención, y finalmente la última columna muestra únicamente los kilómetros recorridos en cada clúster.

Tabla 6: Análisis de Duncan, a un nivel de significación del 0.05: (a) Factor A (NG), (b) Factor B (Tp), (c) Factor C (pM), Factor D (pC)

NG	Promedio	TP	Promedio	pM	Promedio	pC	Promedio
1500	266.71 a	150	260.07 a	0.10	256.46 a	0.95	267.00 a
2000	266.96 a	100	266.92 b	0.05	265.24 b	0.90	268.79 a
1000	273.43 b	50	280.11 c	0.01	285.41 c	0.85	271.32 a

(a) (b) (c) (d)

NG x pM	Promedio	TP x pM	Promedio
2000x0.10	249.69 a	150x0.10	254.22 a
1500x0.10	257.41 a b	100x0.10	254.26 a
1500x0.05	259.55 b	150x0.05	254.86 a
1000x0.10	262.30 b c	50x0.10	260.92 a b
2000x0.05	262.75 b c d	100x0.05	265.58 b c
1000x0.05	268.99 d	150x0.01	271.15 c d
1500x0.01	283.17 e	50x0.05	275.27 d e
1000x0.01	284.61 e	100x0.01	280.92 e
2000x0.01	288.44 e	50x0.01	304.15 f

(e) (f)

Tabla 7: Evaluación del algoritmo integral con 10 repeticiones

Repetición	115 pozos	116 pozos	117 pozos
1	2399.6463	2410.0803	2363.1442
2	2407.5523	2408.0073	2377.4160
3	2402.4043	2407.8363	2382.5159
4	2404.9883	2395.4013	2363.2970
5	2405.0683	2386.7273	2395.9080
6	2410.7373	2395.4013	2379.2450
7	2412.3343	2404.7133	2373.2340
8	2401.5783	2404.7133	2377.2179
9	2402.0803	2386.7273	2369.7050
10	2414.1533	2393.4313	2374.6090
Promedio	2406.0543	2399.3039	2375.6292
Desv. Std.	4.9613	8.8474	9.5875
mínimo	2399.6463	2386.7273	2363.1442
máximo	2414.1533	2410.0803	2395.9080

En la Tabla 8 (recorrido total 2399.6463 km), se observa que la solución obtenida por el algoritmo en dos fases para el programa N°1 (115 pozos), en cada ruta no exceden las 12 horas, sin embargo, equipo swab 4, utiliza casi todo el tiempo máximo tolerable (11h 55' 59"), y el equipo swab 9 el que menor tiempo emplea (04h 54' 18"), se advierte que se ahorrará un equipo swab; con respecto al recorrido efectuado este es de 247.412 km, que comparado con el recorrido actual de 322.846 km hay una reducción de 75.434 km.

Tabla 8: Tiempo total y distancia en la solución óptima para el Programa N°1 (115 pozos)

Nº clúster	Equipo swab	Cant	Pozos para recorrer	Tiempo total (h:m:s)	Kilómetros recorridos
1	1	15	[91 22 40 39 38 45 21 92 70 93 59 57 41 11 71]	10:59:08	23.857
2	2	17	[50 103 68 28 67 31 29 30 35 37 34 33 32 36 48 27 47]	11:34:17	24.078
3	3	13	[116 110 114 111 113 112 109 73 108 115 72 78 105]	11:05:06	39.625
4	4	10	[53 106 79 74 76 77 107 75 80 90]	11:55:59	31.664
5	5	14	[15 5 17 16 6 18 7 9 19 20 23 24 8 89]	11:45:18	25.126
6	6	16	[94 62 63 97 61 98 100 46 26 66 69 25 60 13 99 58]	11:40:48	27.836
7	7	16	[49 65 104 101 64 43 42 44 95 96 10 2 3 4 12 14]	11:37:55	31.216
8	8	10	[56 102 88 86 87 85 84 83 54 82]	10:26:19	37.215
9	--	0	[]	00:00:00	0
10	9	04	[51 81 55 52]	04:54:18	6.795
Total		115	Pozos	95:59:09	247.412

En la Tabla 9 (recorrido total 2386.7273 km), se observa que la solución obtenida por el algoritmo en dos fases para recorrer el programa N°2 (116 pozos), en cada ruta no excede la cantidad de 12:00:00 horas, cumpliendo con el tiempo máximo tolerable; con respecto al recorrido, el algoritmo necesita 9 equipos swab, uno para cada clúster, el equipo swab N°3 es al que le toma más tiempo en recorrer 13 pozos asignados con un tiempo total de 11 horas, 49 minutos y 18 segundos y el equipo swab N°9, es el que menor distancia recorre 13.9 km y emplea 2 horas, 25 minutos y 22 segundos; obteniéndose un recorrido total mínimo de 292.920 km respecto al recorrido actual que es de 408.031 km con 10 equipos, existiendo una reducción de un equipo swab y un ahorro en el recorrido de 115.111 km.

Tabla 9: Tiempo total y distancia en la solución óptima para el Programa N°2 (116 pozos)

Nº Clúster	Equipo Swab	Cant	Pozos para recorrer	Tiempo total (h:m:s)	Kilómetros recorridos
1	1	15	[62 52 34 32 51 107 42 23 41 100 94 57 102 64 98]	11:47:50	29.931
2	2	16	[101 36 63 25 27 26 29 31 33 43 30 28 24 95 22 37]	11:20:15	40.523
3	3	13	[80 114 115 111 112 116 77 91 75 110 90 78 74]	11:49:18	44.756
4	4	12	[53 92 76 81 56 54 38 65 104 11 10 73]	11:33:29	19.365
5	5	13	[85 86 82 83 88 87 84 89 2 5 6 3 4]	11:38:18	44.600
6	6	13	[71 19 20 70 69 49 47 58 67 68 45 21 40]	11:25:15	30.519
7	7	15	[117 108 109 79 113 8 7 9 48 50 46 72 44 61 93]	11:45:24	39.753
8	--	0	[]	00:00:00	0
9	8	16	[13 14 17 18 99 59 103 66 35 60 105 106 16 15 97 55]	11:42:59	29.573
10	9	03	[12 39 96]	02:25:22	13.900
Total		116	Pozos	95:28:09	292.920

La Tabla 10 (recorrido total 2363.1442 km) muestra la solución obtenida por el algoritmo en dos fases para recorrer el programa N°3 (117 pozos). Se necesitan 9 equipos swab uno para cada clúster, en el recorrido de cada ruta no se excede el tiempo máximo tolerable; el recorrido que proporciona el algoritmo con 9 equipos swab, es de 265.911 km, el cual es menor a los 376.103 km que recorren actualmente con 10 equipos. En este programa, también existe la disminución de 1 equipo swab. La nueva secuencia en el recorrido de los pozos disminuye en 110.192 km.

Tabla 10: Tiempo total y distancia en la solución óptima para el Programa N°3 (117 pozos)

Nº Clúster	Equipo swab	Cant	Pozos para recorrer	Tiempo total (h:m:s)	Kilómetros recorridos
1	1	14	[87 85 14 12 8 2 3 20 10 7 5 6 11 13]	11:20:59	24.573
2	2	13	[80 108 109 78 116 76 110 79 77 81 114 54 107]	11:39:53	36.204
3	3	17	[72 71 68 42 41 43 33 40 34 103 104 61 73 36 100 98 96]	11:57:00	29.168
4	4	14	[19 21 22 9 4 82 55 88 90 92 91 89 86 84]	11:24:51	39.102
5	5	17	[32 35 70 28 49 26 27 48 46 47 30 29 31 58 25 69 99]	11:35:37	31.922
6	6	12	[45 44 23 60 51 56 112 111 113 118 117 115]	11:56:19	47.634
7	7	16	[64 65 97 95 67 66 106 39 94 93 102 38 17 16 15 18]	11:59:09	26.314
8	--	0	[]	00:00:00	0
9	8	13	[59 50 83 52 24 37 74 101 63 62 105 57 75]	10:37:30	28.543
10	9	01	[53]	02:00:14	2.184
Total		117	Pozos	94:31:33	265.911

Estas mejoras, se traducen en un ahorro mensual en combustible de 27.17% (S/ 17 157.05), mano de obra 10% (S/ 4 500.00) y alquiler de equipo de transporte en 10% (S/ 33 000.00) que representa en términos monetarios un ahorro de S/ 54 657.05 cada 30 días lo que hace un total anual de S/ 664 994.06, como se puede apreciar en la tabla 11.

Tabla 11: Análisis económico entre la situación actual y la propuesta por el Algoritmo

Eficiencia: 4 km / gln Precio: 11.41 S/. / gln		Combustible				Ahorro %	
		Actual		Propuesta			
Nº días	Descripción	km	S/.	km	S/.		
10	Equipo swab P1	322.846	9209.18	247.412	7057.43	27.17	
	Cisterna para P1	322.846	9209.18	247.412	7057.43		
10	Equipo swab P2	408.031	11639.08	292.920	8355.54		
	Cisterna para P2	408.031	11639.08	292.920	8355.54		
10	Equipo swab P3	376.103	10728.34	265.911	7585.11		
	Cisterna para P3	376.103	10728.34	265.911	7585.11		
30	Costo total en combustible x mes		63153.21		45996.16		
Operador: 80 S/. / día Chofer: 70 S/. / día		Mano de Obra					10.00
		Actual		Propuesta			
Nº días	Descripción	Nº Ope.	S/.	Nº Ope.	S/.		
10	Operador de equipo swab P1	10	8000.00	9	7200.00		
	Chofer de cisterna P1	10	7000.00	9	6300.00		
10	Operador de equipo swab P2	10	8000.00	9	7200.00		
	Chofer de cisterna P2	10	7000.00	9	6300.00		
10	Operador de equipo swab P3	10	8000.00	9	7200.00		
	Chofer de cisterna P3	10	7000.00	9	6300.00		
30	Costo total en mano de obra x mes		45000.00		40500.00		
Eq. Swab 600 S/. / día Cisterna 500 S/. / día		Alquiler de Equipo				10.00	
		Actual		Propuesta			
Nº días	Descripción	Nº Equi.	S/.	Nº Equi.	S/.		
10	Equipo swab P1	10	60000.00	9	54000.00		
	Cisterna P1	10	50000.00	9	45000.00		
10	Equipo swab P2	10	60000.00	9	54000.00		
	Cisterna P2	10	50000.00	9	45000.00		
10	Equipo swab P3	10	60000.00	9	54000.00		
	Cisterna P3	10	50000.00	9	45000.00		
30	Costo Total en alquiler x mes		330000.00		297000.00		
Total, gastos operativos por mes:		438153.21		383496.16			
Ahorro total mensual:				54657.05			
Ahorro total anual:				664994.06			

CONCLUSIONES

De los resultados mostrados, su análisis y discusión, se obtuvieron las siguientes conclusiones sobre la implementación del Algoritmo: 1) La implementación del algoritmo en dos fases (clustering y AG), es una solución para problemas de planificación de rutas en transporte con las características del proceso swab. 2) En la fase 1 del algoritmo todos los clústeres mostraron que no eran buenas soluciones porque algunos excedían el tiempo máximo tolerable. 3) En la fase 2 se optimiza los resultados de la fase 1, corrigiendo las desviaciones fuera del tiempo máximo tolerable, y los clústeres que no se necesitan el algoritmo los deja vacíos; 4) Se alcanza un ahorro sustancial en combustible, mano de obra, así como en el alquiler de los equipos en la empresa de transporte.

REFERENCIAS

- Albornoz, V.M., y Johns, E.H., *Localización de Paraderos de Detención y Diseño Óptimo de Rutas en el Transporte de Personal*, doi: 10.4067/S0718-33052011000300014, *Ingeniare Rev. Chil. Ing.*, 19, 457–472 (2011)
- Anaut, D.O., di Mauro, G.F., Meschino, G., y Suárez, J.A., *Optimización de Redes Eléctricas mediante la Aplicación de Algoritmos Genéticos*, doi: 104067/S0718-07642009000400016, *Inf. Tecnol.*, 20 (2009)
- Apaza, V.M.A., "Modelo de un Algoritmo Genético con Selección Discriminatoria de Individuos bajo un Esquema de Ponderación de Probabilidades de Mutación", Tesis de grado, Universidad Católica San Pablo (2013)
- Bastidas, M.J., Bermúdez, R.F., Jaramillo, G.P., y Chejne, F., *Optimización Termoeconómica y Ambiental usando Algoritmos Genéticos Multiobjetivo*, doi: 10.4067/S0718-07642010000400006, *Inf. Tecnol.*, 21 (2010)
- Cano, J.A., Correa-Espinal, A.A., y Gómez-Montoya, R.A., *Solución del Problema de Conformación de Lotes en Almacenes utilizando Algoritmos Genéticos*, doi: 10.4067/S0718-07642018000600235, *Inf. Tecnol.*, 29, 235–244 (2018)
- Castro, A., Sifuentes, E., González, S., y Rascón, L.H., *Uso de Minería de Datos en el Manejo de Información Geográfica*, doi: 10.4067/S0718-07642014000500014, *Inf. Tecnol.*, 25, 95–102 (2014)

- De-La-Hoz, E.J., De-La-Hoz, E.J., y Fontalvo, T.J., *Metodología de Aprendizaje Automático para la Clasificación y Predicción de Usuarios en Ambientes Virtuales de Educación*, doi: 10.4067/S0718-07642019000100247, Inf. Tecnol., 30, 247–254 (2019)
- Erazo, O.M., "Propuesta de un Programa de Evaluación de Pozos basado en la Optimización del Recorrido de los Operadores de Producción en el Lote I, Talara", Tesis de grado, Universidad Nacional de Piura (2013)
- Franceschetti, A., y otros cinco autores, *A metaheuristic for the Time-Dependent Pollution-Routing Problem*, doi: 10.1016/j.ejor.2016.11.026, Eur. J. Oper. Res., 259, 972-972–991 (2017)
- García, C., "Algoritmos Genéticos Locales", Tesis Doctoral, Escuela Técnica Superior de Ingeniería Informática y Telecomunicación, Editorial de la Universidad de Granada, 34-35, ISBN: 978-84-691-7826-3, España (2008)
- Goldberg D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Longman Publishing Co., Inc. Boston, MA, USA. Tomado de García (2008)
- Holland J.H., *Adaptation in Natural and Artificial Systems*. The University of Michigan Press. Tomado de García (2008)
- Iv, J.H.W., y Cavalier, T.M., *A Construction Heuristic for the Split Delivery Vehicle Routing Problem*, doi: 10.4236/ajor.2012.22018, Am. J. Oper. Res., 2, 153–162 (2012)
- Jaradat, A., Matalkeh, B., y Diabat, W., *Solving Traveling Salesman Problem using Firefly algorithm and K-means Clustering*, doi: 10.1109/JEEIT.2019.8717463, 586–589 (2019)
- Jiménez-Carrión, M., *Algoritmo Genético Simple para Resolver el Problema de Programación de la Tienda de Trabajo (Job Shop Scheduling)*, doi: 10.4067/S0718-07642018000500299, Inf. Tecnol., 29, 299-314 (2018)
- Júnior, A.N., y Guimarães, L.R., *A greedy randomized adaptive search procedure application to solve the travelling salesman problem*, doi: 10.24867/IJEM-2019-3-243, Int. J. Ind. Eng. Manag., 10, 238–242 (2019)
- Kiraly, A. y Abonyi, J., *Optimization of Multiple Traveling Salesmen Problem by a Novel Representation based Genetic Algorithm*, Tomado de Rostami et al., (2015)
- Li, Y., y Wu, H., *A Clustering Method Based on K-Means Algorithm*, doi: 10.1016/j.phpro.2012.03.206, (Department of Computer Science Xinyang Agriculture College Xinyang, Henan 464000, China: Physics Procedia 25), 1104–1109 (2012)
- Lin, S.-W., Lee, Z.-J., Ying, K.-C., y Lee, C.-Y., *Applying Hybrid Meta-Heuristics for Capacitated Vehicle Routing Problem*, doi: 10.1016/j.eswa.2007.11.060, Expert Syst. Appl., 36, 1505–1512 (2009)
- Mohammed, M.A., Abd Ghani, M.K., y otros cuatro autores, *Solving Vehicle Routing Problem by Using Improved Genetic Algorithm for Optimal Solution*, doi: 10.1016/j.jocs.2017.04.003, J. Comput. Sci., 21, 255–262 (2017)
- Mulloorakam, A.T., y Mathew, N., *Combined Objective Optimization for Vehicle Routing Using Genetic Algorithm*. doi: 10.1016/j.matpr.2018.12.016, Mater. Today Proc., 11, 891–902 (2019)
- Naderipour, M., y Alinaghian, M., *Measurement, Evaluation and Minimization of CO₂, NO_x, and CO Emissions in the Open Time Dependent Vehicle Routing Problem*, doi: 10.1016/j.measurement.2016.04.043, Measurement 90, 443–452 (2016)
- Ortega, P., Oliva, C., Ferland, J., y Cepeda, M., *Multiple Ant Colony System for a VRP with Time Windows and Scheduled Loading*, doi: 10.4067/S0718-33052009000300013, Ingeniare Rev. Chil. Ing., 17, 393–403 (2009)
- Rodríguez, A.B., Gutiérrez, A.L., Rivera, L.A., y Ramírez, L.J., *Ruteo y Asignación de Longitud de Onda: Comparación de Algoritmos Genéticos y Templado Simulado*, doi: 10.4067/S0718-07642014000400003, Inf. Tecnol., 25, 13–18 (2014)
- Rostami, A.S., Mohanna, F., Keshavarz, H., y Hosseinabadi, A.A.R., *Solving Multiple Traveling Salesman Problem using the Gravitational Emulation Local Search Algorithm*, doi: http://dx.doi.org/10.12785/amis/090218, 9, 699-709 (2015)
- Salgado, P., y Afonso, P., *Evolutionary Genes Algorithm to Path Planning Problems*, doi:10.1007/978-3-030-17065-3_22, Adv. Intell. Syst. Comput., 942, 217–225. (2019)
- Wahyuningsih, S., Satyananda, D., y Hasanah, D., *Implementations of TSP-VRP Variants for Distribution Problem*, Global Journal of Pure and Applied Mathematics, ISSN: 09739750, 12, 723–732 (2016)
- Wang, C.-H., y Lu, J.-Z., *A Hybrid Genetic Algorithm that Optimizes Capacitated Vehicle Routing Problems*, doi: 10.1016/j.eswa.2008.01.072, Expert Syst. Appl., 36, 2921–2936 (2009)
- Yang, M.S., y Sinaga, K.P., *A Feature-Reduction Multi-View k-Means Clustering Algorithm*, doi: 10.1109/ACCESS.2019.2934179, IEEE Access IEEE 7, 114472 (2019)
- Zhang, S., Lee, C.K.M., y otros 3 autores, *Design and Development of a Hybrid Artificial Bee Colony Algorithm for the Environmental Vehicle Routing Problem*, doi: 10.1016/j.trd.2014.05.015, Transp. Res. Part Transp. Environ. 31, 85–99 (2014)