

Problemas de optimización en logística: una propuesta de solución aplicando el algoritmo Particle Swarm Optimization (PSO)

Jorge Hernán Cárdenas Lopera

Ingeniero Electrónico de la Universidad Pontificia Bolivariana. Maestro en Administración Empresarial con énfasis en estrategia del Instituto Tecnológico de Monterrey. Estudiante de Maestría en Física de la Universidad de Antioquia con trabajo de investigación de computación científica y algoritmos de optimización. Coordinador del área de Innovación y Nuevos Negocios de la Institución Universitaria Esumer. Medellín. Correo electrónico: innovacion.negocios2@esumer.edu.co.

Jorge Andrés Acosta Strobel

Tecnólogo en Comercio Internacional, Profesional en Negocios Internacionales y Especialista en Legislación Aduanera de la Institución Universitaria Esumer. Especialista en Paz y Desarrollo Territorial de la CUN. Magíster en Gestión de Ciencia, Tecnología e Innovación de la Universidad de Antioquia, Estudiante de Doctorado en Pensamiento Complejo en la Multiversidad Mundo Real Edgar Morín. Coordinador de programas de Negocios Internacionales de la Institución Universitaria Esumer. Medellín. Correo electrónico: Jorge.acosta@esumer.edu.co.

Recibido: 24/01/2023 - **Aceptado:** 21/02/2023 - **Publicado:** 30/03/2023

RESUMEN

El presente artículo tiene como propósito proponer herramientas de optimización para solucionar problemas logísticos a través de las ciencias computacionales. En este documento se utiliza el algoritmo *Particle Swarm Optimization* (PSO) como alternativa para solucionar problemas asociados a los sistemas logísticos en las organizaciones, con un énfasis en la optimización para el abastecimiento y transporte dentro de las empresas. Conforme a lo anterior, el trabajo está en dos epígrafes, los cuales son: Identificación del algoritmo PSO para los procesos logísticos y su aplicación conceptual a través de etapas, brindando como conclusión que las ciencias computacionales son fundamentales para los retos que están experimentando los sistemas logísticos en las organizaciones. En especial, la estructura del PSO permite de manera sencilla obtener mecanismos de optimización en el proceso de abastecimiento y transporte, pues sus etapas de aplicación son apropiadas para solucionar problemas logísticos.

Palabras clave: Algoritmo; Distribución; Logística; Particle Swarm Optimization (PSO); Transporte.

ABSTRACT

The purpose of this article is to propose optimization tools to solve logistical problems that allow through computational science. This document uses the Particle Swarm Optimization (PSO) algorithm as an alternative to solve problems associated with logistics systems in organizations, emphasizing optimization for supply and transportation within companies. In accordance with the above, the work is divided into two headings, which are: Identification of the PSO algorithm for logistics processes and its application through stages, providing as a

conclusion that computer science is fundamental to the challenges that logistics systems are experiencing in the organizations. Furthermore, the PSO structure allows for a simple optimization mechanism in the supply and transport process, since its application stages are well suited to solve problems in logistics.

Keywords: Algorithm; Distribution; Logistics; Particle Swarm Optimization (PSO); Transport.

1. INTRODUCCIÓN

A nivel global, la cadena de abastecimientos es fundamental para el desarrollo de las organizaciones, éstas intervienen de principio a fin en el funcionamiento de las compañías y puede generar pérdidas o ganancias en la entrega de bienes o servicios, conforme sea el caso de cada empresa. Cada estrategia establecida en el marco de los abastecimientos impacta directamente al consumidor final, por lo cual siempre es un proceso sensible en las empresas.

En problemas de los movimientos, recolección y entrega, en los procesos de suministros, muchas veces se evidencia el tema vehicular o de transporte en general, esto debido a las variables que se asocian a las carreteras, pavimentos, trasbordos, paros entre otros factores tal como lo manifiesta Savelsbergh y Sol (1995). En el caso de Colombia en particular, la infraestructura en particular, hace más evidentes las variables anteriormente establecidas, con el agravante de los altos costos económicos de transportes asociados por los peajes y las diferentes concesiones viales.

Conforme a lo anterior, existen herramientas de optimización para solucionar problemas logísticos que permiten, a través de las ciencias computacionales, generar apoyo para el desarrollo integral de las organizaciones, en especial, la gestión de abastecimientos en las diferentes líneas que lo solicite la compañía. Dentro de los procesos de optimización logística se puede hablar del algoritmo *Particle Swarm Optimization* (PSO), el que aplicado al ámbito de las ciencias administrativas propone la resolución de problemas de optimización para el abastecimiento y transporte, a través de soluciones candidatas conocidas como partículas, generando espacios de búsqueda asociados a diferentes variables de la distribución y gestión de mercancías.

Se debe indicar que el PSO, lleva a la generación y la vinculación de las ciencias de la computación a los sistemas logísticos integrales, los cuales demandan una innovación constante a partir de procesos de mejoramiento continuo dentro de las organizaciones. En este sentido, Rivera y Patiño (2021), indican que la gestión tecnológica permite el desarrollo de las compañías. Del mismo modo, Maggi et al. (2020), establecen que, en las compañías a nivel global, existe una falencia en la adopción de elementos que permitan la gestión tecnológica, generando de esta forma deficiencias en el desarrollo tecnológico. Estos factores a largo plazo impactan en el consumidor e incluso en la divulgación o difusión de productos, lo cual puede llegar a desembocar una alta tasa de mortalidad empresarial o productiva (Acosta y Montoya, 2019).

Al existir tanto grado de desconocimiento de las ciencias computacionales en los procesos administrativos, es necesario generar competencias que estimulen una concepción de la manipulación de algoritmos en pro de mejorar los procesos de las organizaciones.

Qasim y Yahya (2018), establecen que en los sistemas logísticos por lo general existen problemas en la clasificación de entradas o salidas de productos, lo cual ralentiza los procesos de optimización de recursos. Aun así, dichos autores también explican que el PSO, junto a un modelo de regresión logística, aporta a la gestión de clasificación y gestión de abastecimientos, teniendo herramientas que promuevan la automatización tal como lo es el *Warehouse Management Software* (WMS).

Conforme a lo anterior, el presente artículo busca realizar un acercamiento a los principios del algoritmo *Particle Swarm Optimization* (PSO), y cómo estos pueden aportar al desarrollo de las organizaciones en su sistema logístico. Para ello el texto está dividido en dos epígrafes que busca (i) contextualizar el algoritmo de Optimización por Enjambre de Partícula (PSO) e (ii) identificar las Etapas de PSO y cómo éstas pueden incidir en las organizaciones a través de los procesos logísticos.

2. MARCO TEÓRICO Y/O ANTECEDENTES

Dentro del *pipeline* de diseño comúnmente usado en ciencia e ingeniería, el modelamiento y la optimización son etapas críticas que garantizan la elección justificada de soluciones que satisfagan las demandas impuestas por un problema específico. Particularmente, con relación a los retos de diseño en logística, el modelamiento hace uso de datos recopilados acerca de la demanda, frecuencias, estacionalidad, así como costos de transporte y *warehousing*. Dicha etapa antecede la optimización, pues es allí donde se construyen los modelos matemáticos o algorítmicos que representan un sistema o proceso, por ejemplo, el modelo matemático con el cual se determina la distribución de productos en un almacén a partir de la demanda o rotación de inventario (Parsopoulos y Vrahatis, 2010). Durante esta etapa se pueden acumular gran cantidad de datos que deben ser agregados para identificar las variables o *features* del problema, que enriquezcan la posterior optimización.

En cuanto al proceso de optimización, preste precisa disponer del modelo o mecanismo de simulación para calcular resultados (variables de salida) a partir de unos parámetros de entrada (variables de entrada). Así pues, para un modelo estadístico que arroja datos sobre algunas variables de salida, como por ejemplo el costo y el tiempo en ruta de un camión distribuidor, se tendría como una entrada correspondiente la ubicación geográfica de los almacenes o la distancia en kilómetros (Lind et al., 2012). El objetivo de la optimización es encontrar las variables de entrada “correctas” que, al aplicarlas al modelo matemático, permite obtener unas salidas que satisfagan unas restricciones impuestas. Si se considera el tiempo de ruta como variable, podría establecer para ella una restricción o un límite máximo de X número de horas. Así mismo, de forma iterativa evaluar muchas configuraciones de almacenes y en cada caso se obtendrán diferentes valores de tiempo en ruta (Ruiz-Moreno et al., 2020). Intuitivamente se puede establecer que la mejor configuración de almacenes es aquella que arroje el tiempo en ruta por debajo de X horas, pero ¿qué sucede si esa configuración propuesta implica establecer quince (15) almacenes y el presupuesto disponible solo cubre el establecimiento de cinco (5) almacenes?

Lo anterior quiere decir que, el objetivo de optimización no solo es el tiempo en ruta, sino también el costo de la solución, es decir, se tendrían dos variables que considerar para encontrar la configuración que mejor se ajuste a ambos requerimientos. El problema puede crecer en complejidad si se evalúan más variables (como suele suceder) dentro del problema logístico. Con fines didácticos, se puede pensar en una variable más: el nivel de inventario por cada *warehouse* (Peláez y Acosta, 2021). Ese parámetro de salida del modelo debería estar sujeto a

una restricción, es decir un nivel máximo y/o mínimo. En ese caso la propuesta inicial de configuración de los almacenes, al ser evaluado en el modelo, este debe proporcionar valores específicos sobre tiempo en ruta, costo de la solución y nivel de inventario en cada almacén. Como se puede apreciar, encontrar el “punto óptimo” es el desafío de éste y de otros problemas asociados al proceso logístico (Arsham, 1994).

Por cuenta de situaciones similares a las expuestas anteriormente, la optimización se ha convertido en un objeto de estudio creciente en logística, abordando el uso de técnicas de aprendizaje automático (*Machine Learning*) y otros algoritmos de optimización, como *Simulated Annealing* y *Reed Deer Algorithm* (RDA) (Abdi et al., 2020), que aseguren resultados pertinentes y precisos; en consecuencia, que apoyen decisiones de inversión. Y es que gran parte de los problemas aquí presentados tienen como característica común, que se está trabajando con sistemas complejos de múltiples variables y múltiples objetivos de optimización, y que los modelos no necesariamente deben ser deterministas, sino que pueden llegar a ser de naturaleza estocástica. Por este motivo, es que se ha convertido en un eje de investigación la aplicación de métodos para resolver problemas de optimización usando algoritmos que se apoyen en datos históricos, como en el caso de sistemas supervisados de *Machine Learning*, o que se basen en el uso de muestras aleatorias, como los algoritmos genéticos o evolutivos (Bertsimas y Youssef, 2020).

El problema radica fundamentalmente en el desarrollo de un modelo que represente, de la manera más precisa posible, el fenómeno que queremos estudiar, y en disponer de un método de optimización para lograr el mejor desempeño posible del modelo propuesto. En este trabajo se presenta el algoritmo de *Optimización por Enjambre de Partículas* o *PSO* por sus siglas en inglés, que, por sus características, se ha utilizado en diferentes ámbitos de la ciencia e ingeniería de forma exitosa para trabajar sobre sistemas complejos. Nuestro objetivo no radica en la solución de un problema específico de optimización, sino más bien introducir un método innovador de optimización para la solución de múltiples problemas en el ámbito logístico a partir de una descripción del algoritmo y su abordaje desde un caso didáctico.

3. METODOLOGÍA O DESCRIPCIÓN DEL PROCESO

Esta investigación se fundamenta en la descripción y análisis de algunos problemas de optimización y del algoritmo PSO como alternativa de solución, a partir de la consulta bibliográfica de literatura acerca del método y de su pertinencia en el ámbito de la logística. Para la pesquisa de información se tuvieron en cuenta instrumentos por los cuales se caracteriza la revisión documental, a partir de repositorios de apoyo como Ebsco, Scopus y Dialnet (Ramírez-Carvajal et al., 2021), así como páginas de acceso asociadas al proceso de Optimización por Enjambre de Partículas. En el análisis se incluyen investigaciones que presentan soluciones a problemas de diseño de redes logísticas, así como casos del alcance de otras áreas del conocimiento que dan soporte al argumento sobre la capacidad y flexibilidad del algoritmo para enfrentar diferentes tipos de problemas.

Así mismo, se recurre a un método deductivo, con el que se abordan los conceptos generales de optimización y del algoritmo PSO, y a partir de ellos se construyen ejemplos de implementación simplificada del algoritmo sobre un problema logístico tratado previamente en la literatura.

Teniendo en cuenta lo anterior, para analizar los resultados y hallazgos se realizó un ejercicio en fases para comprender el PSO, en primera instancia, estudiando el algoritmo de

optimización y sus etapas de aplicación: inicialización de la población, evaluación inicial de la población, actualización de partículas y evaluación de nuevas partículas. Cada uno de esos pasos se confronta con casos documentados de problemas logísticos para brindar soluciones al objeto principal de estudio.

Con base en lo anterior, a través del estudio de bases de datos y el algoritmo, se le brinda al lector una aproximación a cómo el PSO se podría aplicar en las organizaciones que tengan problemas principalmente logísticos.

4. ANÁLISIS DE RESULTADOS O HALLAZGOS

a. ALGORITMO DE OPTIMIZACIÓN POR ENJAMBRE DE PARTÍCULA (PSO)

En la ciencia e ingeniería hay un interés permanente en la búsqueda de “la forma eficiente de hacer las cosas”, especialmente para la solución de problemas que involucran sistemas dinámicos, es decir, aquellos cuyo estado cambia con el tiempo. Si bien los paradigmas han cambiado respecto a la definición del concepto de optimización, se ha llegado a un entendimiento común sobre el mismo y su relación con el número de objetivos de optimización. Un gran número de problemas requieren que se optimice no solo una, sino múltiples variables que habitualmente compiten entre sí, es decir, optimizar una de las variables implica un sacrificio de otra(s) variable(s); así pues, pueden pensarse el costo y el tiempo en ruta como dos variables que son susceptibles de optimización, pero que, al mismo tiempo son antagonistas y compiten entre ellas. Lo anterior permite inferir que, las soluciones de optimización no son perfectas y, habitualmente, requieren compromisos entre las variables para llegar a un resultado que satisfaga los requerimientos del problema.

Así pues, en la búsqueda de métodos de optimización, durante la década de los noventa estuvieron en auge los métodos basados en gradiente descendente (Snyman, 2005), pero en años recientes ha habido un creciente interés en otros métodos, principalmente por cuenta de la aplicación exitosa de algoritmos basados en búsqueda aleatoria. Estos métodos se basan en la exploración aleatoria de soluciones propuestas en el espacio de parámetros del sistema (Arora, 2015). El advenimiento de los algoritmos de *Machine Learning* y la Inteligencia Artificial presuponen la disponibilidad de un gran número de herramientas susceptibles de ser aplicadas a problemas en logística, pero las cuestiones que requieren optimización deben tratarse con cuidado y hacer una selección considerada del método de optimización para el problema específico.

Podemos estudiar el transporte de carga urbano como un problema. La eficiencia depende de la planeación, organización, manejo y ejecución de la operación de transporte, pero también dependerá indirectamente de factores como la concentración de población, el crecimiento de la población de una ciudad y la creciente congestión en las rutas. Esto da indicios de estar frente a un sistema dinámico. Si sumamos a lo anterior la demanda continua por reducción de costos de instalación y operación, se puede notar que, los problemas logísticos, en general, se caracterizan por su a) variabilidad temporal, b) la frecuencia y la cobertura espacial como parámetros y, adicionalmente, c) un gran número de estos parámetros pueden estar en conflicto. Lo anterior llama la atención, pues el problema de optimización debe ser atendido a través del método adecuado frente a esas características, y no solo usando algoritmos que obedezcan al *hype* del momento.

Si analizamos todo lo planteado anteriormente, los problemas logísticos se representan en un espacio de parámetros multidimensional, es decir, se puede tener un número de grados de libertad

elevado y con ellos un alto número de restricciones. Es tarea del diseñador de la solución logística, inferir cuales son las variables independientes y dependientes del problema, y de construir un modelo que matemáticamente relacione dichas variables. En esta instancia, el modelo podría ser generado por un algoritmo de *Machine Learning* que arroje resultados a partir de datos históricos recopilados sobre variables del sistema, pero en ningún momento el modelo sustituye de forma directa la necesidad de un método o algoritmo de optimización.

Durante el proceso de optimización se debe construir lo que se conoce como *función objetivo*. Esta(s) ecuación(es) debe(n) relacionar las variables de diseño o de decisión, las que a su vez se someten a restricciones impuestas por el problema. Tomemos un caso en que se dispone de cinco alimentos, cada uno con un aporte nutricional específico de proteínas, vitaminas y calorías, entre otros (Arora, 2015). Cada alimento tiene un costo y queremos encontrar la cantidad de cada alimento que contribuya a la demanda nutricional requerida por una persona, teniendo como objetivo la minimización del costo total. En este caso, las variables de diseño son las cantidades de cada alimento representadas como x_1, x_2, x_3, x_4 y x_5 . Así pues el costo total sería,

$$\text{Costo Total} = 1.1x_1 + 1.2x_2 + 2x_3 + 1.3x_4 + 3x_5. \quad (1)$$

Esta ecuación corresponde a la función objetivo, donde los coeficientes son los costos de cada uno de los alimentos en cuestión.

Las restricciones surgen de la cantidad mínima requerida por persona de cada nutriente y la cantidad de nutriente aportada por cada alimento. Por ejemplo, si la persona requiere cien (100) unidades de vitaminas, y el alimento x_1 aporta solo nueve (9) unidades, se deben agregar otros alimentos que completen las noventa y un (91) unidades restantes de vitaminas. El desafío es encontrar la combinación y la cantidad de alimentos que satisfaga las restricciones al menor costo posible. Como podrá notar el lector, el problema puede ser complejo, pero se puede abordar de formas más sencillas al definir la función objetivo y las fórmulas que expresan apropiadamente las restricciones del problema.

Así pues, entendiendo que se pueden encontrar problemas con múltiples variables de entrada y salida, en los que la información de desempeño del sistema se condensa en la(s) función(es) objetivo y en las restricciones, queda pendiente analizar cuál es el método adecuado para seleccionar los valores de las variables de entrada, de tal forma que una vez aplicadas al modelo matemático del sistema, nos garantice el desempeño óptimo del mismo, conforme a las restricciones establecidas. Este asunto se puede exponer mejor a la luz del ejemplo anterior. Si bien se tienen restricciones y se cuenta con una función objetivo que relaciona las cantidades de alimento y aporte nutricional, no es claro que método se debe aplicar para obtener el menor costo posible a partir de la *minimización* de la fórmula del costo total.

Como se mencionó previamente, existen muchas alternativas en cuanto a métodos de optimización, particularmente considerando el arribo de las técnicas más avanzadas en ciencias de datos como *Machine Learning*. De todas esas alternativas, aquí se presenta el algoritmo PSO (Particle Swarm Optimization) como método con potencial para la solución de problemas logísticos dadas algunas de sus características, con los beneficios de reducción de tiempos de procesamiento y que es suficientemente flexible para ajustarlo a los requerimientos de nuestro problema. Este algoritmo ha sido estudiado en el ámbito logístico al compararlos con otros algoritmos híbridos y metaheurísticos en la solución de un problema de diseño de rutas (Abdi et al., 2020). Así mismo, ha sido validado ampliamente en otras áreas como robótica (Di Mario

et al., 2013), pronósticos en sismo resistencia (Mashayekhi et al., 2019) y la optimización de dispositivos de microondas para aplicaciones en radioastronomía (Cárdenas et al., 2022), lo que muestra el amplio espectro de aplicación del PSO.

El PSO está clasificado dentro de los algoritmos genéticos y, con más especificidad, como un algoritmo evolutivo, lo que significa que está basado en los mecanismos de la selección natural para discriminar las soluciones al problema de optimización (Arora, 2015), es un algoritmo potente, fácil de implementar y eficiente computacionalmente (Kennedy et al., 2001). Este algoritmo en particular estudia la inteligencia de enjambre (*Swarm Intelligence*), como las bandadas de aves o cardúmenes peces que, de forma cooperativa, buscan alimento, y en el que cada miembro del enjambre modifica su movimiento a partir del aprendizaje colectivo e individual (Wang et al., 2018). Para los casos de optimización, el enjambre (*población*), compuesto por muchos miembros (*partículas*), busca la mejor solución explorando un gran espacio de parámetros. Dicha exploración se da por el movimiento de cada miembro en el espacio persiguiendo la mejor solución global, es decir, la mejor solución evaluada por todo el enjambre.

La característica más relevante es lo que se conoce como *exploración aleatoria*, donde la población de partículas se mueve de forma aleatoria, pero con el paso del tiempo y de la experiencia acumulada, confluyen en la región de mayor interés dentro del espacio de solución. Volvamos al caso del problema de optimización de las vitaminas. El diseñador podría comenzar el proceso de optimización asumiendo que la leche, la que corresponde a la cantidad x_1 , es un producto que nunca puede faltar en la dieta. A partir de esa condición inicial, el experto puede ir seleccionando de forma iterativa algunos de los demás productos, y a través de algunos cálculos sencillos, llegaría a una combinación de productos en diferentes cantidades que le garanticen un costo total Y , inferior a un costo máximo establecido. Se pueden inferir fácilmente dos cosas. La primera es que, a pesar del dispendioso proceso manual, seguramente existe una solución mejor, es decir, una que garantiza un costo menor al hallado. La segunda es que, pueden existir un buen número de métodos (algoritmos) para explorar las posibles soluciones. En los algoritmos de exploración aleatoria, se generan posibles soluciones al problema que sean producto de un muestreo aleatorio. El componente genético o evolutivo se hace evidente en la forma como se van a seleccionar y a “mutar” las soluciones propuestas con el mejor ajuste (*fit*) a nuestro objetivo de optimización. Volviendo al caso didáctico, una forma de resolver el problema sería la siguiente: se selecciona una cantidad aleatoria de cada uno de los productos alimenticios, por ejemplo $x_1 = 1.2, x_2 = 2.3, x_3 = 0.5, x_4 = 1.1$ y $x_5 = 4$. Estos valores se introducen a la función de costo total arrojando un valor de \$18.5. Si suponemos que el valor máximo del costo es \$20, se podría concluir que tenemos una “buena” solución pues se ajusta a la condición de menor costo. En un segundo experimento, podría generarse una nueva propuesta de solución aleatoria como $x_1 = 4, x_2 = 0.3, x_3 = 0.7, x_4 = 2.1$ y $x_5 = 3.5$, cuyo valor de costo total sería \$19.39. Si bien es tentador asumir que la primera propuesta es la correcta, pues su costo total es menor, así mismo es posible que no cumpla con los requerimientos nutricionales de una persona, mientras que la segunda podría hacerlo.

El lector agudo podrá pensar que, un método de este tipo requiere de un número infinito de propuestas aleatorias de solución para garantizar que efectivamente se ha encontrado la mejor solución, y está en lo correcto. Este problema se resuelve a partir del concepto de evolución. En los algoritmos genéticos, se tiene una *población* que hereda sus características a miembros de la generación posterior, a partir de una evaluación del “éxito” de cada miembro. En un algoritmo como el PSO, se generan soluciones propuestas aleatorias, las que se evalúan con

una *función de fitness* que nos mida el “éxito”, es decir, que tan bien se ajusta una solución a todos los requerimientos.

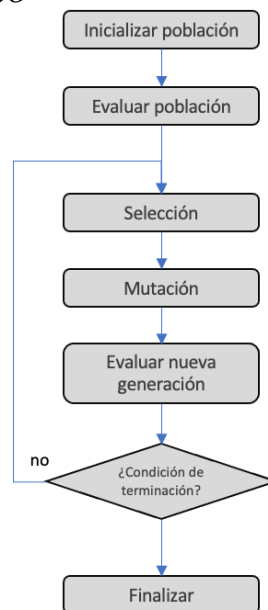
El resultado de la función de fitness nos permite emitir un veredicto sobre la calidad de cada solución, así pues, para el ejemplo usado hasta el momento, podríamos tener una función de fitness que, a partir del costo total obtenido y los aportes nutricionales con cada cantidad de alimento, califique las soluciones. Esto resuelve el problema mencionado sobre el número de propuestas aleatorias requeridas, pues en lugar de proponer un número infinito de soluciones aleatorias y evaluar cada una, el algoritmo genera una *población finita* de soluciones aleatorias, las que se evalúan y seleccionan, para que aquellas con el mejor fitness pasen sus características a una nueva generación en la población de soluciones.

b. ETAPAS DEL ALGORITMO PSO

Para profundizar en los detalles de aplicación del PSO, se puede abordar problemas más específicos de la logística como el modelamiento matemático para una red de distribución (Ruiz-Moreno et al., 2020), o el caso de modelamiento matemático para optimización de una red logística integrada (Khajavi et al., 2011). Estos ejemplos presentan modelos y soluciones a los problemas planteados, los que se pueden reinterpretar a la luz del algoritmo PSO. Para abordar el algoritmo con respecto a su aplicación en logística, se hará un desarrollo a partir de las etapas más generales del mismo, considerando el diagrama presentado en la gráfica 1.

Gráfica 1

Diagrama de flujo del algoritmo PSO



Nota. Elaboración propia.

PASO 1: INICIALIZACIÓN DE LA POBLACIÓN

El primer paso es la creación de una población de partículas, donde cada miembro de la población se construye como un vector cuyos elementos son las entradas del modelo. La palabra “población” significa un número grande de estos vectores que se crean aleatoriamente. El ámbito de aplicación define como lucen esos vectores, por ejemplo, en el campo de radio astronomía se requiere incrementar la sensibilidad de los receptores, lo que se logra

optimizando el desempeño electromagnético de algunos dispositivos de microondas. Dicha intervención se realiza al ajustar variables geométricas como ancho, alto, profundidad, y al adicionar cavidades a los dispositivos de microondas. Así pues, para este caso, un vector o partícula tiene todas las dimensiones que describen un diseño geométrico, es decir, el ancho y alto de la sección transversal, el largo del dispositivo y el número de ramificaciones, entre otras variables.

Volviendo al área logística y tomando el caso desarrollado por Khajavi et al. (2011), donde se utiliza un modelo *forward/reverse* MILP para diseñar la red de distribución, encontramos que el problema radica en determinar la locación, el número y capacidad de producción, distribución, recolección, y el flujo de producto entre los centros de producción y recolección/despacho. Una versión muy simplificada del problema tendría las variables similares a las presentadas en la tabla 1. Adicionalmente, hay algunos parámetros que se deben insertar al modelo, y que se presentan en la tabla 2 de manera reducida.

Tabla 1

Conjunto de variables reducido de un modelo para optimización de una red de distribución

Variable	Descripción
p	Periodos de tiempo de evaluación.
r_{kpw}	Tasa de retorno del producto w usado desde una zona k después de un periodo p .
d_{kpw}	Demanda del producto w en una zona k ocurrida en un periodo p .
WC_{jpw}	Costo de almacenamiento del producto w en el centro de distribución j en un periodo p .

Nota. Elaboración propia.

Tabla 2

Parámetros reducidos del modelo matemático para diseño de una red de distribución

Variable	Descripción
$CAX_{ij\ v p}$	Capacidad del modo de transporte v para transportar varios productos del centro de producción i al centro de distribución j en un periodo p .
o_j^n	Costo fijo de abrir un centro de distribución j con un nivel de capacidad n .
f_i^n	Costo fijo de abrir un centro de producción i con un nivel de capacidad n .

Nota. Elaboración propia.

Una partícula se representaría en este caso simplificado por un vector con la siguiente forma,

$$S_i = [p, r_{kpw}, d_{kpw}, WC_{jpw}, CAX_{ij\ v p}, o_j^n, f_i^n] \quad (2)$$

La inicialización de la población con cierta cantidad de partículas de forma aleatoria, daría como resultado un conjunto S de vectores, donde cada uno representa unos valores propuestos para las variables que se introducirán al modelo matemático, como los que se muestran en la tabla 3.

Tabla 3

Representación vectorial de las partículas para el algoritmo PSO

Partícula	Vector
S_1	$[p = 1, r_{kpw} = 0.6, d_{kpw} = 120, WC_{jpw} = 4, CAX_{ij\ vp} = 154, o_j^n = 15340, f_i^n = 45650]$
S_2	$[p = 3, r_{kpw} = 0.54, d_{kpw} = 200, WC_{jpw} = 6.4, CAX_{ij\ vp} = 202, o_j^n = 38965, f_i^n = 65901]$
S_3	$[p = 6, r_{kpw} = 0.45, d_{kpw} = 350, WC_{jpw} = 9, CAX_{ij\ vp} = 103, o_j^n = 25367, f_i^n = 56890]$
S_n	\vdots

Nota. Elaboración propia.

PASO 2: EVALUACIÓN INICIAL DE LA POBLACIÓN

Para este problema se plantea como objetivo la minimización de la función,

$$\begin{aligned} \text{Costo} = & \text{Fixed Opening Costs} + \\ & \text{Fixed and Variable Transportation Costs} + \\ & \text{Operation Costs} + \text{Warehousing Cost} \end{aligned} \quad (3)$$

Cada uno de los sumandos de la ecuación, se calcula a partir de la aplicación de otras ecuaciones del modelo (Khajavi et al., 2011), tomando los valores de cada uno de los vectores anteriores y así se obtiene el costo total de cada solución propuesta. Se considera como *mejor solución global*, aquel vector que nos arroje el menor costo y que, finalmente, se constituye en la tendencia que todas las demás partículas deberían seguir. Como se verá posteriormente, las partículas deben mutar de tal manera que, en conjunto, converjan en aquella solución que se considera la mejor para todas. Así pues, al aplicar la función de costo, se podría llegar a resultados como los siguientes valores supuestos,

$$\begin{aligned} \text{Costo de } S_1 &= 45.505.209,9 \\ \text{Costo de } S_2 &= 39.577.754 \\ \text{Costo de } S_3 &= 68.316.544,8 \\ &\vdots \end{aligned} \quad (4)$$

Los valores anteriores implican que la partícula S_2 es la que se clasifica con el mejor fitness global, lo que se convierte en un parámetro requerido para la operación del PSO y para la mutación de las partículas durante el resto del proceso iterativo posterior.

PASO 3: PROCESO ITERATIVO – ACTUALIZACIÓN DE PARTÍCULAS

Se debe recordar que, los principios bajo los que opera un algoritmo como el PSO son la exploración aleatoria y la mutación de los miembros de la población. Este proceso de cambio es iterativo y se prolonga durante el número de generaciones que se evalúe necesario según el objetivo de optimización. Hay casos reportados en los que 15-20 iteraciones (generaciones) son suficientes para lograr la convergencia (Cárdenas et al., 2022), mientras que otros problemas requieren que el número de iteraciones sea superior a 100.

La actualización de las partículas se realiza con el uso de lo que los investigadores han denominado la *fórmula de velocidad*. Si llamamos x_i a cada uno de los elementos de los vectores, donde $x_1 \equiv p$, $x_2 \equiv r_{kpw}$, etc., entonces la actualización o mutación de cada uno de estos elementos se da de acuerdo con la fórmula,

$$x_i(t + 1) = x_i(t) + v_i(t + 1) \quad (5)$$

Es decir, el valor en el tiempo presente (iteración actual) $x_i(t)$, se suma a una cantidad $v_i(t + 1)$, para dar como resultado, el nuevo valor de la variable. El valor de la velocidad $v_i(t + 1)$, codifica en su interior la información sobre la partícula con el mejor *fitness* global (p_g), el mejor *fitness* individual (p_i) y el valor de velocidad en la iteración inmediatamente anterior, $v_i(t)$, y se obtiene a partir de la siguiente función,

$$v_i(t + 1) = v_i(t) + c_1 r_1 (p_i - x_i) + c_2 r_2 (p_g - x_i) \quad (6)$$

Aquí r_1 y r_2 son variables aleatorias uniformemente distribuidas entre 0 y 1, y donde c_1 y c_2 son pesos denominados parámetros *cognitivo* y *social*, respectivamente. El parámetro cognitivo permite dar más peso significativo a la experiencia ganada por cada partícula, es decir la partícula tiende a mutar en la dirección de su mejor resultado personal sin considerar cual haya sido el mejor resultado global. Por otro lado, el parámetro social asigna mayor peso al mejor *fitness* global, lo que finalmente impulsa las partículas en la búsqueda conjunta de la mejor solución. Es necesario tener un balance entre ambos valores de tal forma que el cambio se haga progresivo para cada partícula y evitar caer en mínimos locales.

En el caso de aplicación, debemos actualizar cada una de las partículas S_i a partir de las fórmulas anteriores. Sabemos que el mejor *fitness* global le pertenece a S_2 . Si hacemos $c_1 = 1.8$ y $c_2 = 2.0$, entonces la ecuación de velocidad para la variable r_{kpw} de la primera partícula S_1 ,

$$v_{1 r_{kpw}}(t = 1) = 1.8 \times 0.65 \times (1 - 1) + 2.0 \times 0.18 \times (0.6 - 0.54) = 0.0216 \quad (7)$$

En este caso $v_i(t = 0) = 0$, puesto que no se tenía información preliminar sobre la velocidad para la partícula S_1 , pero es de esperar que, en futuras iteraciones, este valor de velocidad inicial si tenga un valor específico. Adicionalmente, r_1 y r_2 , al ser valores aleatorios, están cambiando continuamente a partir de una distribución uniforme; para este caso los podemos suponer como $r_1 = 0.65$ y $r_2 = 0.18$.

Este valor de velocidad se debe utilizar para actualizar las entradas de la partícula S_1 de la siguiente manera,

$$r_{kpw}(t + 1) = r_{kpw}(t) + v_{1 r_{kpw}}(t = 1) = 0.6 + 0.0216 = 0.6216 \quad (8)$$

El procedimiento debe aplicarse para cada componente de cada partícula, lo que nos entregaría un nuevo vector con la siguiente forma,

$$S_1(t + 1) = [p = 3, r_{kpw} = 0.6216, d_{kpw} = \dots]. \quad (9)$$

PASO 4: PROCESO ITERATIVO

Finalmente, la nueva generación de partículas es evaluada a la luz de la función de *fitness* para que, con ese resultado, se pueda seleccionar aquella nueva partícula que tiene el mejor *fitness* global. Es decir, a lo largo del proceso iterativo no se tiene una partícula específica que posea el título de la “mejor”, sino que dicho título estará cambiando de manos, y durante ese proceso el conjunto de partículas se irá aproximando a la “mejor solución para todas”, es decir la solución óptima. Esta propiedad se manifiesta en vectores que convergen con el tiempo, es

decir se asemejan más entre ellos, como se ejemplifica en la tabla 4, donde se ejemplifica cómo es que las variables de diferentes partículas convergen.

En términos prácticos, lo anterior significa que, al llegar al punto de convergencia de las partículas, se ha obtenido una solución propuesta que corresponde a un mínimo de la función de costo, es decir, es la combinación de entradas y parámetros que nos ofrece el mejor costo.

Tabla 4

Representación vectorial de las partículas para el algoritmo PSO

Partícula	Vector
S_1 generación=20	$[p = 4, r_{kpw} = 0.723, d_{kpw} = 175, WC_{jpw} = 5.9, CAX_{ij\ vp} = 103, \dots]$
S_2 generación=20	$[p = 4, r_{kpw} = 0.733, d_{kpw} = 175, WC_{jpw} = 5.9, CAX_{ij\ vp} = 102, \dots]$
S_3 generación=20	$[p = 4, r_{kpw} = 0.741, d_{kpw} = 177, WC_{jpw} = 6.1, CAX_{ij\ vp} = 103, \dots]$
S_n generación=20	⋮

Nota. Elaboración propia.

Al referirse a “un mínimo”, se hace con la intención de llamar la atención al problema que enfrenta este algoritmo y otros, incluyendo los de gradiente descendente, y es que, en el espacio de soluciones es posible encontrar más de un mínimo. Esto es particularmente evidente en problemas de multiobjetivo, en los frecuentemente se llega a lo que se conoce comúnmente como *mínimos locales*. Con el ánimo de solucionar este problema se ha recurrido a una versión modificada de la fórmula de velocidad (Parsopoulos y Vrahatis, 2010), en la que se incluye un término de inercia w . El nuevo parámetro permite controlar el valor de velocidad de tal forma que la exploración del espacio de soluciones se realice de forma más o menos lenta y así evitar la llegada y permanencia en esos mínimos locales.

$$v_i(t + 1) = wv_i(t) + c_1r_1(p_i - x_i) + c_2r_2(p_g - x_i) \quad (10)$$

Para finalizar, es importante resaltar el carácter heurístico del PSO pues, en la práctica, no solo es cuestión de disponer de un modelo matemático para el problema a resolver, sino que se debe realizar un proceso de configuración y tuneo de los hiperparámetros del algoritmo, de tal forma que se garantice la validez de los resultados obtenidos y el buen desempeño computacional. En algunos casos es necesario el desarrollo preliminar de pequeños modelos cuya solución sea conocida o que estén bien caracterizadas en la literatura, con el propósito de que, al aplicarle el algoritmo, se pueda evaluar su comportamiento y fiabilidad. Por ejemplo, para el caso de la optimización de dispositivos de microondas, la geometría del dispositivo se modifica durante el proceso de optimización, pero se dispone de diseños previos bien caracterizados en la literatura, permitiendo un punto de comparación o *benchmark* que aporte validez a los resultados que se obtendrán posteriormente (Cárdenas et al., 2022).

5. CONCLUSIONES

Los retos de diseño en los sistemas logísticos están asociados a su variabilidad temporal, así como a los parámetros de cobertura espacial y el gran número de objetivos de optimización antagonistas. El diseñador debe considerar dichos factores al formular un modelo para el sistema y, especialmente, al momento de llevar a cabo un proceso de optimización. Si bien en la literatura se encuentran diferentes herramientas para enfrentar dicha etapa, el algoritmo

genético PSO cuenta con suficiente validación en diferentes contextos de aplicación, con resultados que demuestran su flexibilidad y reducción de los tiempos de procesamiento.

Usando como referencia un modelo planteado en otras investigaciones para el diseño de redes de distribución, se muestra que sus parámetros y variables pueden llevar a la representación vectorial del algoritmo propuesto. Posteriormente, al contar con dicha representación y haciendo uso del diagrama de flujo general del algoritmo, se evidenció que, las etapas del PSO se ajustan adecuadamente para solucionar un problema logístico, a partir del principio de mutación de los vectores. Aún más, las características de los problemas logísticos pueden atenderse apropiadamente haciendo uso del principio de exploración aleatoria de soluciones, también incluidas en la estructura del PSO.

El algoritmo aporta como ventajas su fácil implementación, de tal manera que el experto puede enfocarse en actividades como construir los modelos matemáticos, ajuste fino del algoritmo y la posterior verificación de los resultados. El diseño de un sistema logístico se puede impactar positivamente al disponer de un mecanismo de optimización suficientemente flexible, como para que se pueda reajustar rápidamente a los cambios del modelo matemático y de los parámetros de configuración del sistema.

6. REFERENCIAS

Acosta, J. y Montoya, D. (2019). Revisión de la difusión de la innovación: casos de consolas de Nintendo. *Revista Espacios*, 40(1), 1-19. <http://es.revistaespacios.com/a19v40n01/a19v40n01p19.pdf>

Abdi, A., Abdi, A., Akbarpour, N., Amiri, A. S. y Hajiaghahi-Keshteli, M. (2020). Innovative approaches to design and address green supply chain network with simultaneous pick-up and split delivery. *Journal of Cleaner Production*, 250. <https://doi.org/10.1016/j.jclepro.2019.119437>

Arora, R. (2015). *OPTIMIZATION Algorithms and Applications*. Chapman & Hall.

Arsham, H. (1994). Modelos Deterministas: Optimización Lineal. <http://home.ubalt.edu/ntsbarsh/business-stat/opre/spanishd.htm>

Bertsimas, D. y Youssef, N. (2020). Stochastic optimization in supply chain networks: averaging robust solutions. *Optimization Letters*, 14(4), 839-855.

Cárdenas, J. H., Chaparro, G., Restrepo, Ó., Mena, P. y Monasterio, D. (16-20 de octubre de 2022). Particle Swarm Algorithm Applied to Quadrature Hybrid Multi-Branch Directional Coupler Optimization for ALMA Band 3. In *32nd IEEE International Symposium on Space THz Technology (ISSTT 2022)*. Baeza, Andalusia. <https://acortar.link/qZRYgL>

Di Mario, E., Talebpour, Z. y Martinoli, A. (2013). A comparison of PSO and Reinforcement Learning for multi-robot obstacle avoidance. 2013 IEEE Congress on Evolutionary Computation, Cancún, México, 2013, pp. 149-156. <https://doi.org/10.1109/CEC.2013.6557565>

- Khajavi, L., Seyed-Hosseni, S. M. y Makui, A. (2011). A mathematical model for optimization of an integrated network logistic design. *Management Science Letters*, 1(4), 415–426. <https://doi.org/10.5267/j.msl.2011.05.003>
- Kennedy, J., Eberh, R. y Shi, Y. (2001). *Swarm Intelligence*. Morgan Kaufmann
- Lind, D., Marchal, W. y Wathen, S. (2012). *Estadística aplicada a los negocios y la economía*. The McGraw-Hill.
- Maggi, C., Ramos, M. y Vergara, R. (2020). *Adopción de tecnologías digitales 4.0 por parte de pequeñas y medianas empresas manufactureras en la Región del Biobío (Chile)*. CEPAL.
- Mashayekhi, M., Harati, M. y Estekanchi, H. E. (2019). Development of an alternative PSO-based algorithm for simulation of endurance time excitation functions. *Engineering Reports*, 1(3), 1-15. <https://doi.org/10.1002/eng2.12048>
- Parsopoulos, K. y Vrahatis, M. (2010). *Particle Swarm Optimization and Intelligence. Advances and Applications*. IGI Global.
- Peláez, D. y Acosta, J. (2021). La importancia de la implementación de Warehouse Management System para los Centros de Distribución. *Revista CIES*, 12(1), 213-232. <http://revista.escolme.edu.co/index.php/cies/article/view/338>
- Qasim, O. y Yahya, Z. (2018). Feature selection using particle swarm optimization-based logistic regression model. *Chemometrics and Intelligent Laboratory Systems*, 182, 41-46. [doi:https://doi.org/10.1016/j.chemolab.2018.08.016](https://doi.org/10.1016/j.chemolab.2018.08.016)
- Ramírez-Carvajal, C., Praj, D. y Acosta-Strobel, J.-A. (2021). La relación triangular entre China, Taiwán y Estados Unidos en el periodo 2008-2018. *Revista Latinoamericana de Estudios de Seguridad*, (30), 92-106. <https://doi.org/10.17141/urvio.30.2021.4780>
- Rivera, J. y Patiño, J. C. (2021). La Gestión Tecnológica y su importancia en el Desarrollo de la Estrategia Corporativa. *Gesta*, (26), 5-8. <https://fondoeditorial.itm.edu.co/libros-electronicos/otras-publicaciones/gesta/Boletin-gesta-edicion-26.pdf>
- Ruiz-Moreno, S., Arango-Serna, M. D., Serna-Urán, C. A. y Zapata-Cortes, J. A. (2020). Modelo matemático para la optimización de la red de distribución de una empresa de transporte de paquetería y mensajería terrestre. *Revista DYNA*, 87(214), 248-257. <https://doi.org/10.15446/DYNA.V87N214.84679>
- Savelsbergh, M. y Sol, M. (1995). The General Pickup and Delivery Problem. *Transportation Science*, 29(1), 17-29. <https://doi.org/10.1287/trsc.29.1.17>
- Snyman, J. (2005). *Practical Mathematical Optimization. An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms* (2ª ed.). Springer.
- Wang, D., Tan, D. y Liu, L. (2018). Particle swarm optimization algorithm: an overview. *Soft Computing*, 22(2), 387-408. <https://doi.org/10.1007/s00500-016-2474-6>